IBM Software Group

IBM WebSphere Application Server v6

WebSphere Rapid Deployment

e business on demand.

IBM Confidential

Updated October 12, 2004

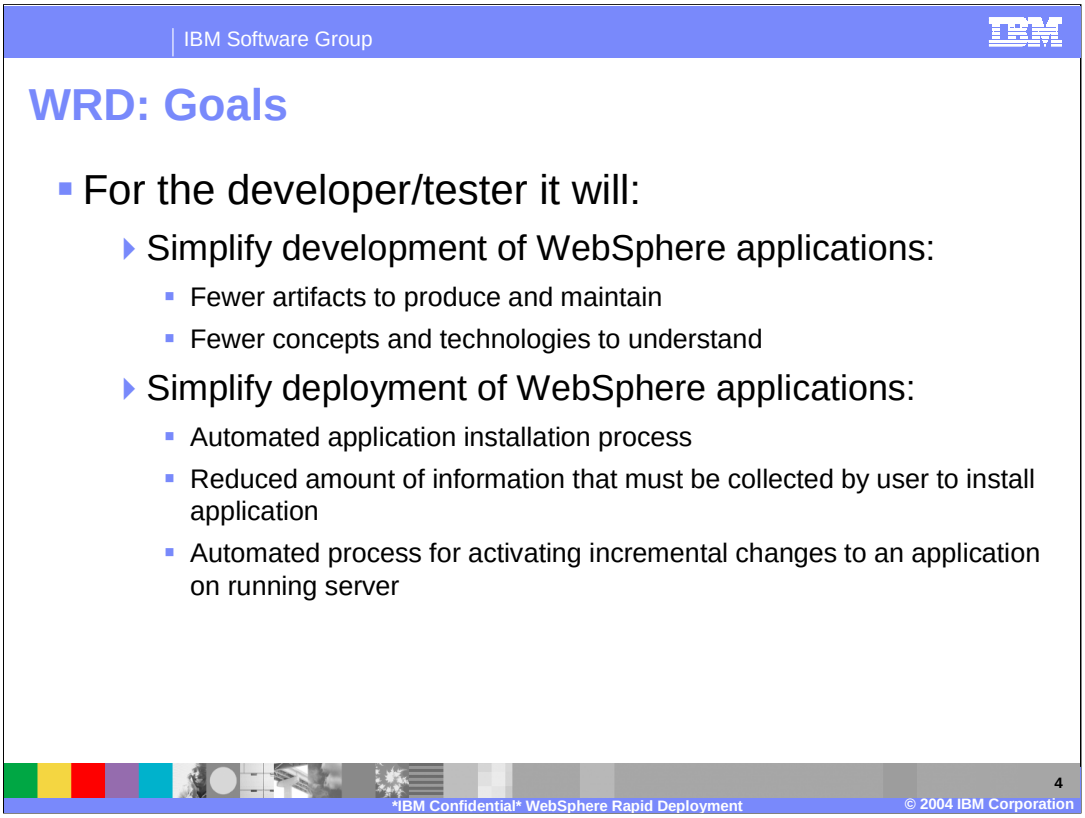
Agenda

- WebSphere Rapid Deployment Overview
  - WebSphere Rapid Deployment in general
  - Annotation-based Programming
  - Deployment Automation
- Setting up and using WRD for Deployment Automation
- Annotation-based Programming Details
- Usage Scenarios
- Summary

Section

WebSphere Rapid Deployment (WRD)

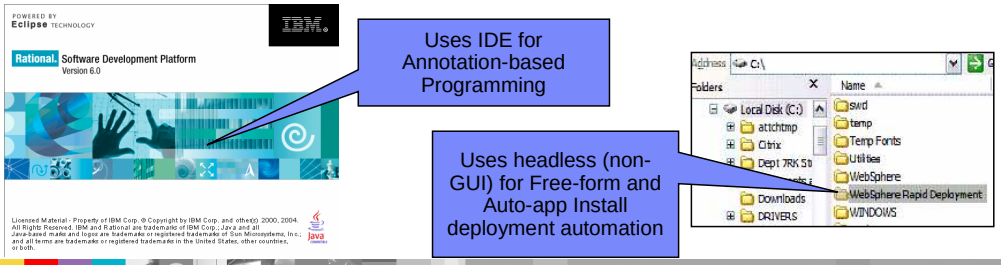
Overview



So what are the goals that WRD achieves? There are really two main points that WRD is trying to simply and improve. The first is the development of WebSphere applications. By maintaining fewer artifacts a developer can concentrate more on the business logic. The other is fewer concepts and technologies to learn and understand. As an example, WRD has a style called free-form (or by part application) that can construct a J2EE application from just simple artifacts like servlets. There is no need to understand the project structure of a J2EE application.

WRD: What is it and Where does it run?

- Comprised of following key concepts:
  - Annotation-based programming
  - Deployment automation
- WRD is a collection of Eclipse plug-ins:
  - **Annotation-based programming** – used within IBM Rational Web/Application Developer and Application Server Toolkit (AST) applications
  - **Deployment Automation:** Uses a headless (non-GUI) mode on an user defined file system directory, defined as a WRD workspace



WRD requires no changes on the Application Server – It uses existing application server administration function to deploy and control applications

WRD: Annotation-based Programming

- Developer adds metadata tags into application source code
  - Uses XDoclet tag syntax, where defined
- WRD uses the metadata to generate additional artifacts needed to run the application on the Application Server
- Minimizes number of artifacts a developer needs to create and understand – user maintains the single artifact

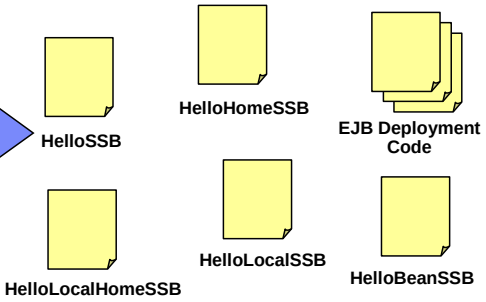
Single Java Source File with Annotation-based programming

```
package com.ibm.wrd;
 /**
  *  @ejb.session name="Hello" type="Stateless"
  * view-type=both  jndi-name="HelloBean"
  */
public class Hello
{
   /**
    * @ejb.interface-method view-type=both
    */
    public String hello(String name)
    {
        return "Hello: " + name;
}
}
```

Hello.java

Generates

Multiple Java Source Files and application artifacts



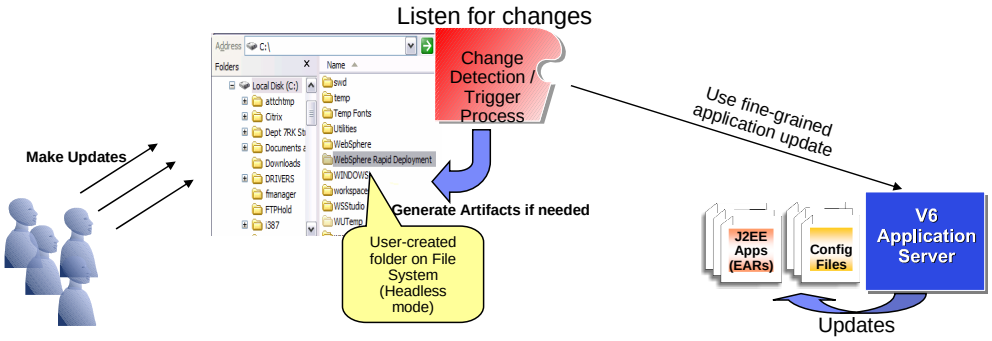
WRD: Deployment Automation

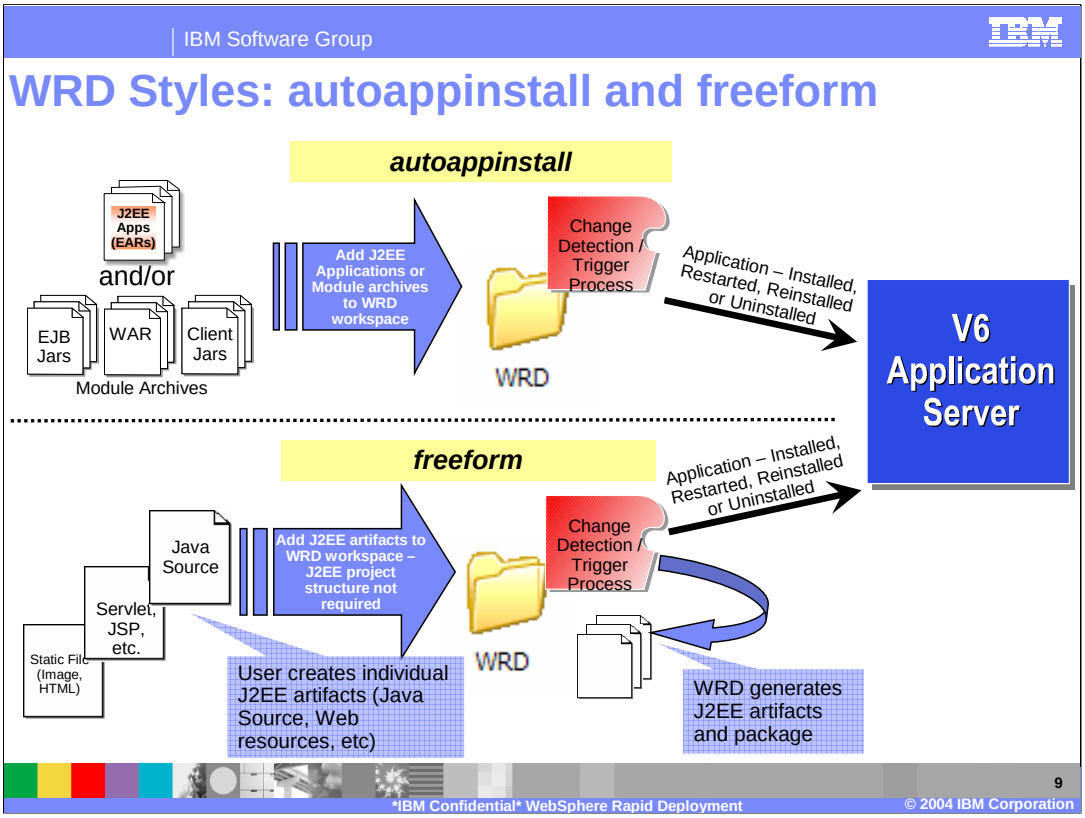
- Enable automatic installation of applications and modules onto a running WebSphere Server - local or remote servers
- Free form application development (initially only available in Headless mode)
  - Enables a “Hot Directory” concept for “file copy” and “Notepad” development and deployment
  - Constructs a well-formed EAR file from individual artifacts
  - Makes key decisions about default settings
- Support deployment of fine-grained application changes
- Goal of minimal application impact

Monitors a directory within workspace through Rational Application Developer (RAD) or with Eclipse User-Interface (UI)

WRD: Change Detection/Triggering Process

- Monitors the file system for changes in the WRD user workspace
- Drive processing operations based on the detection of change in artifacts of the application
  - Generates new application artifacts from existing artifacts
  - Drive deployment to the targeted WebSphere Application Server





Automated EAR Style

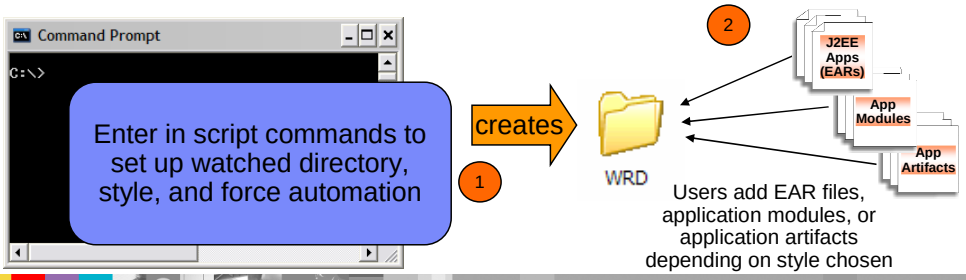
- Use this style for existing J2EE Application
- Copy an EAR or module archive into folder on file system
- Have application/module installed on server
- Assumed to be well-formed, pre-packaged J2EE application/module
- Application is either installed, restarted, reinstalled, or uninstalled
- Automates the application install wizard

Section

Setting up Deployment Automation

Setting up Deployment Automation

- Uses the Eclipse framework
- No User Interface (headless)
- Set of command line scripts
  - Tool included under the <WebSphere_Root>\bin directory
    - Actual implementation is stored under <WebSphere_Root>\deploytool directory
  - Two steps to using WRD
    - Configure the WRD Project (Name of project, style of project)
    - Enable WRD (Start monitoring designated project)

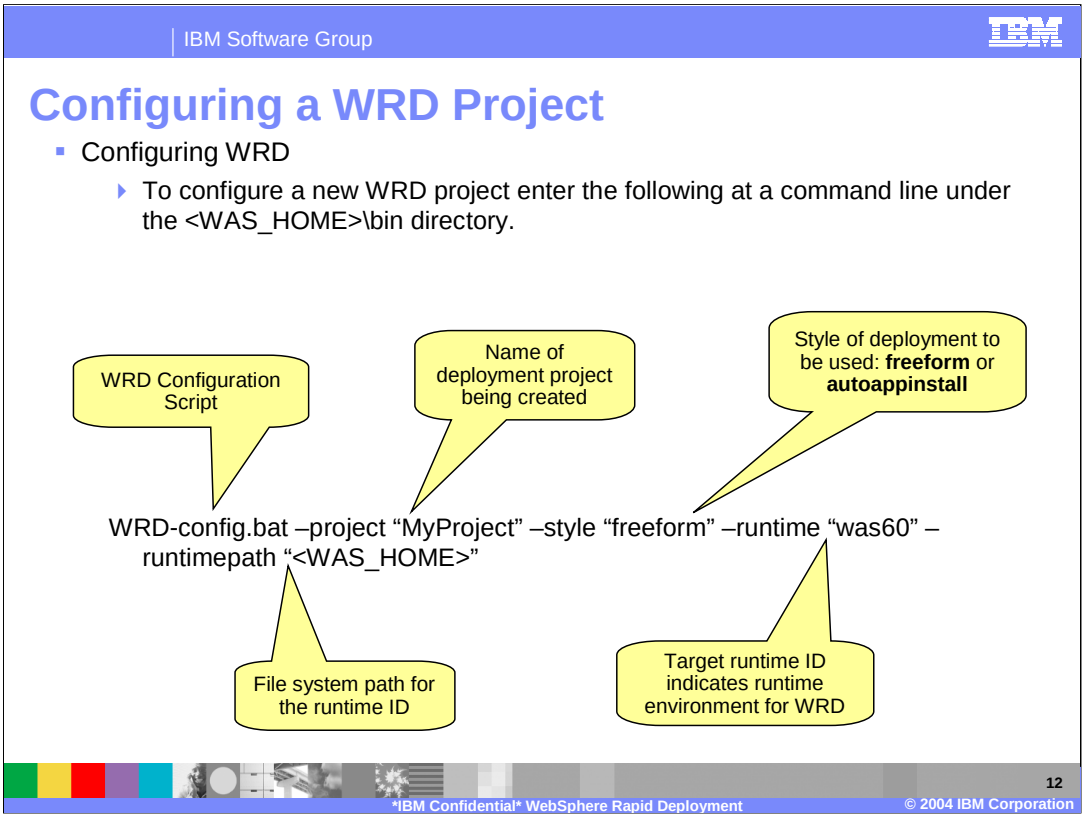


Set of command line scripts

WRD Configuration tasks - script allows user to create set of common WRD-enabled workspaces without launching Studio/ASTK in GUI mode

WRD Build Scripts – allow the user to execute a single pass of a build for a WRD-enabled workspace

Script **WRD.(bat/sh)** is used to start a WRD enabled Eclipse workspace in headless mode

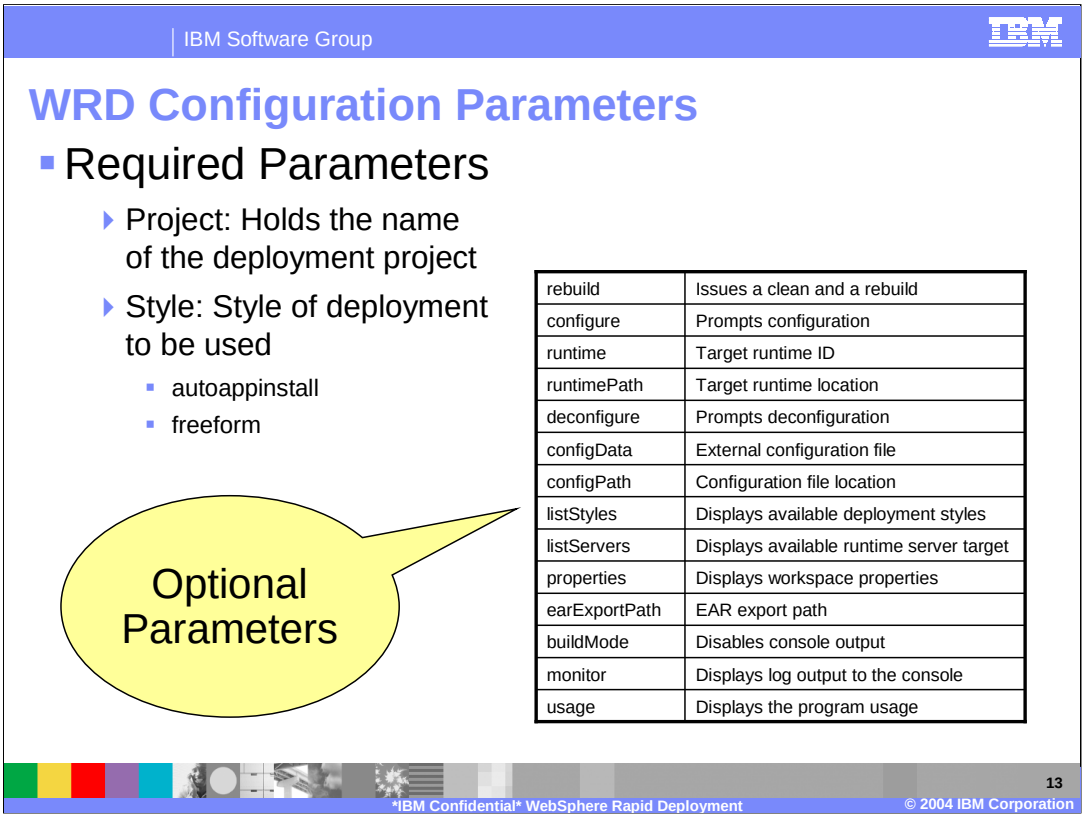


Set of command line scripts

WRD Configuration tasks - script allows user to create set of common WRD-enabled workspaces without launching Studio/ASTK in GUI mode

WRD Build Scripts – allow the user to execute a single pass of a build for a WRD-enabled workspace

Script **WRD.(bat/sh)** is used to start a WRD enabled Eclipse workspace in headless mode



--rebuild : Invokes a full clean and rebuild of the contents within the hot deployment project.

–configure : Opens an interactive console session for modifying any available parameters.

–runtime : The value of this parameter contains the target runtime ID that will indicate a particular runtime environment for WRD processing.

-runtimePath: The value of this parameter contains the filesystem path for the target runtime ID.

–configData : The value of this parameter points to the path of an xml configuration file that is used to drive the configuration session.

-configPath : The value of this parameter points to the destination path for the configuration file created for the session. If this path isn't provided, the default location will be within the root of the target project.

-listStyles : Lists all the available deployment styles and their descriptions.

-listServers : Lists all the available runtime server targets.

-properties : Displays the properties for a given rapid deployment project.

-earExportPath: Path to where EARs will be exported

-buildMode : Flags the application to turn off all console output. This may be useful during silent builds.

-monitor : Redirects all log output directly to the console. By default, all log output is written to an external log file.

-usage : Displays the program usage.

Running WRD

- After WRD is configured, invoke .bat/.sh script to enable monitoring of specified directory

  <profile_home>\bin> WRD.bat

- Additional parameters when running
  - -monitor: Run WRD with console output
  - -batch: Runs a build of WRD workspace and then shuts down

Section

Annotation-based Programming Details

Annotation-based Programming Details

- Comprised of two components
  - Definition of supported tags
    - Scope of tags
    - Artifacts the tags produce
  - Definition of processing builder
    - Provides mechanics for processing tags
    - Support annotations using Javadoc-style comments in Java Source file

Annotation-based Programming Details (cont.)

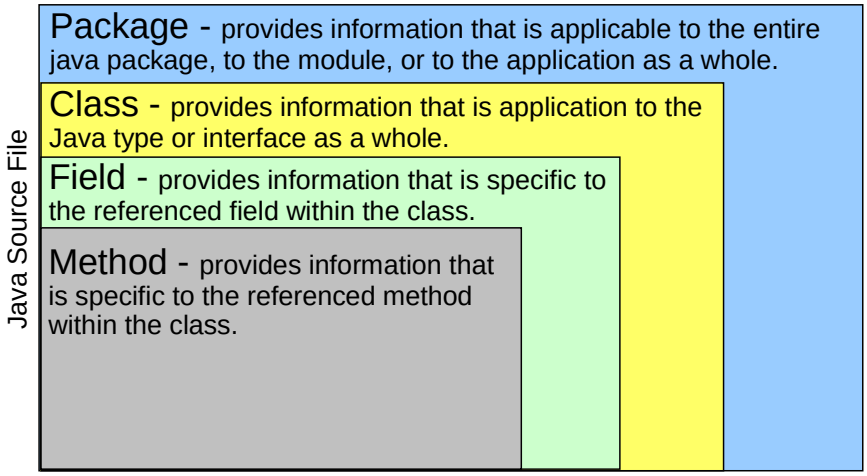
- Entered using an @-tag in comment block
- Code-assists features will be part of IBM Rational Application Developer
- Provides another option for development

Example

```
package com.example.wrd;
/**
 * @ejb.session name="Hello" type="Stateless"
   view-type=remote jndi-name="HelloBean"
 */
public class Hello {
   /**
    * @ejb.interface-method view-type=remote
    */
   public String hello(String name) {
     return "Hello: " + name;
   }
}
```

Scope Level

- Tags can be placed different scope levels



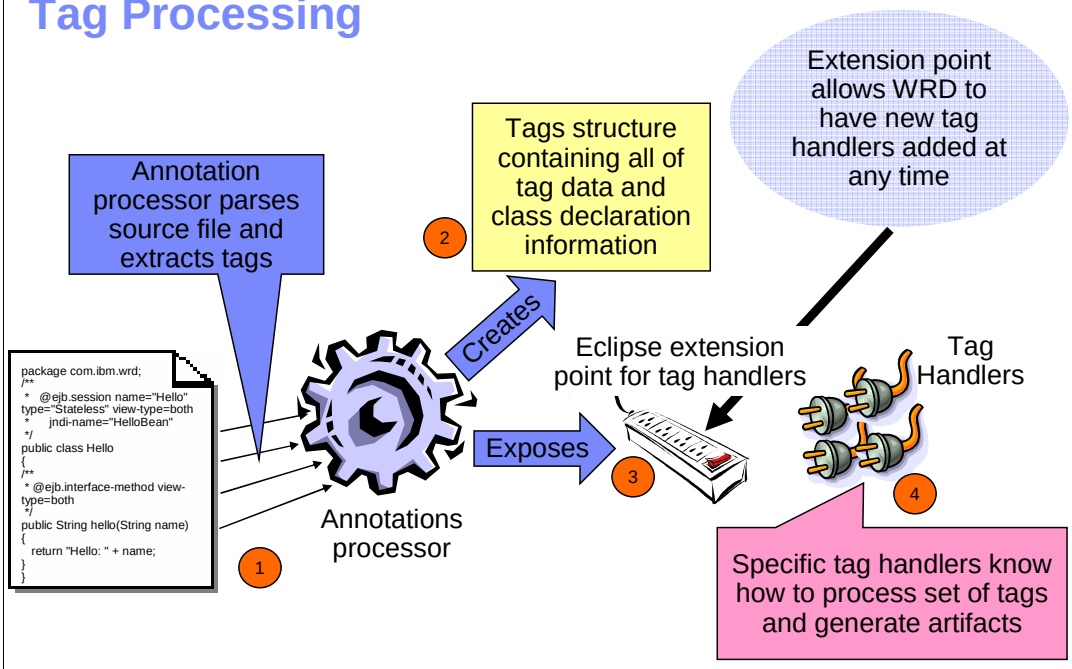
Tag Definitions

- Tags
  - EJB and Web (Supported in the UI)
  - Web Services (Supported only in headless mode)
- Adopts tag syntax used by XDoclet

Relationship to XDoclet: What's the difference?

- XDoclet is a popular open-source project
  - Supports annotation-based programming
  - Process annotations as part of build process, when all annotations are read and all artifacts generated
- Functional overlap, however different processing model
  - WRD supports incremental, on-demand processing
  - Will not directly leverage code from XDoclet project
- WRD adopts tag syntax used by XDoclet for J2EE 1.3
  - Will adopt J2EE 1.4 tags when XDoclet 2 is released
  - WRD contains proprietary tags for WebSphere specific development

Tag Processing



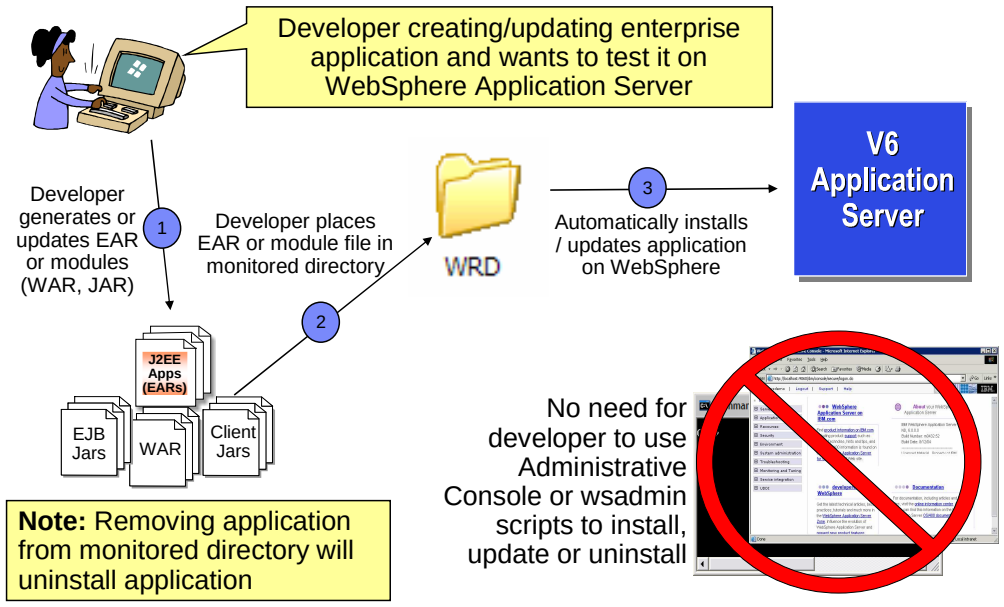
Annotation-based Programming: The Future

- JSR 175 (Metadata facility for the Java programming language)
  - Add metadata into the Java language
  - Standard set of tags for generating artifacts
- Eventually WRD will move to support this JSR (J2SE 1.5)

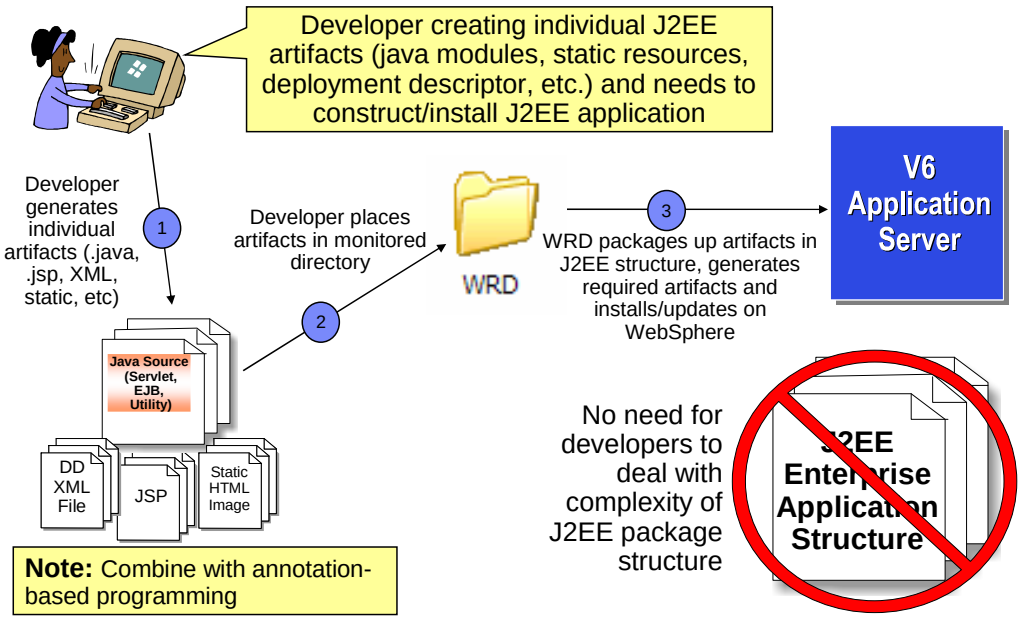
Section

Usage Scenarios

Scenario: Using autoappinstall Style

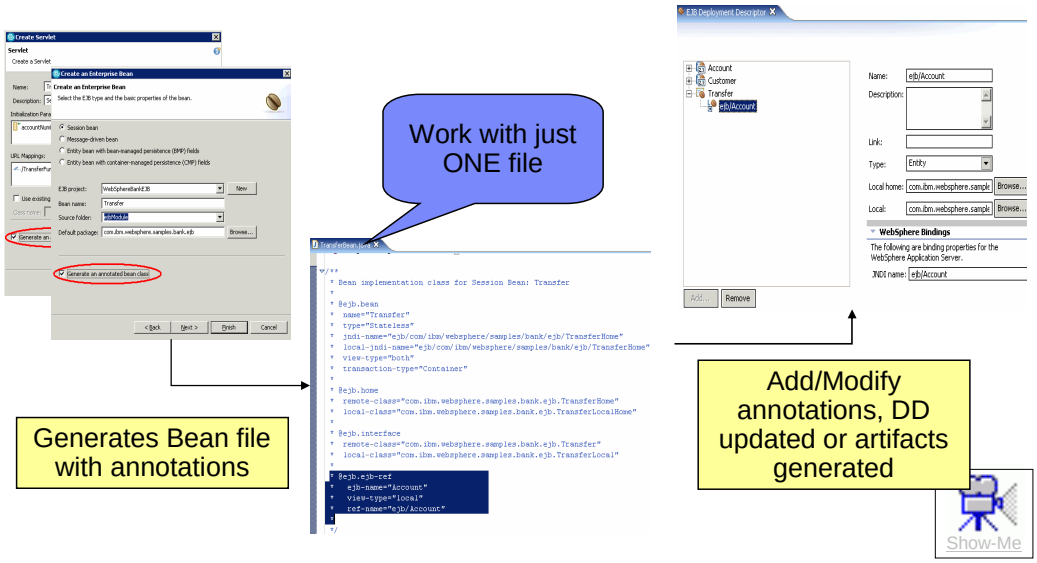


Scenario: Using freeform Style



Scenario: Annotations with Tooling products

- When creating EJBs/Servlets can generate annotations



Section

Summary

Summary

- Described WebSphere Rapid Deployment
- Deployment automation
  - Technical details
  - How to use
- Annotation-based programming feature
- Future of annotation-based programming and the progress in standardizing
- Usage Scenarios
- Real power of WRD comes when both functions are brought together and utilized

Template Revision: 10/01/2004 4:46 PM

Trademarks and Disclaimers

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

IBM	iSeries	OS/400	Informix	WebSphere
IBM(logo)	pSeries	AIX	Cloudscape	MQSeries
e(logo)business	xSeries	CICS	DB2 Universal Database	DB2
Tivoli	zSeries	OS/390	IMS	Lotus

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both. Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are
trademarks of Intel Corporation in the United States, other countries, or both. UNIX is a registered trademark of The Open Group in the United States and other countries. Linux is a registered trademark of Linus Torvalds. Other company, product and service names may be trademarks or service marks of others.

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 and/or changes in the product(s) and/or program(s) 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 (e.g., 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.

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.

This Page Intentionally Left Blank