



IBM Global Services - IBM eServer iSeries

J06
Web Tooling Beyond the Basics
Don Yantzi



Miami, FL

Oct 17-21, 2005

why "i"? it's simple.

© 2005 IBM Corporation

Agenda

- J2EE
 - A deeper look into J2EE
 - JDBC
 - J2EE Connector Architecture
 - Java Naming and Directory Interface
- Web Tooling
 - Web projects
 - J2EE Navigator and Hierarch Views
 - Struts
- Server Tooling
 - Creating new server configurations in the test environment
 - Configuring the test environment

Introduction

- There are many different technologies at your disposal when creating Web applications (and lots of acronyms to go with them!)
 - J2EE, JDBC, JCA, JMS, CSS, HTML, JSP, WAS, ...
 - Each has it's own useful purpose
- However, some are more common than others
 - CSS – Cascading Style Sheets for defining a consistent look and feel across all your web pages
 - HTML and JSP – Replace DDS as way to define the user interface
 - Struts – Great architecture to follow for your overall Web application
 - JDBC – Database access and stored procedure call using SQL
 - JCA – Java Connector Architecture for calling iSeries programs and service programs
- Purpose of this presentation is to cover the more common ones in greater detail and provide you with a foundation to explore the others



IBM Software Group

J2EE – The Technologies

WebSphere software



@business software

© 2003 IBM Corporation

J2EE - Components

- There are 4 main pieces to J2EE application model
 - ▶ Components
 - ▶ Containers
 - ▶ Services
 - ▶ Connectors

- Components (Modules)
 - You develop your code as components of a J2EE application
 - Many different types of components
 - Applets
 - Application clients (full graphical client)
 - Enterprise JavaBeans components (business logic)
 - Web components

J2EE - Containers

- Containers
 - Components run inside of a container
 - Containers are typically provided by system vendors like IBM
 - Web and EJB containers are provided with WebSphere Application Server
 - Provide services that can be used by used by the components which run in the container
 - Transaction support
 - Resource pooling
 - database connections
 - Often allow component behavior to be specified at deployment time instead of development time
 - Configuring which database to access
 - Maximum number of database connections

J2EE - Services

- Service Technologies
 - The J2EE specification defines standard APIs to access many common services
 - JDBC
 - Database-independent method for using SQL
 - Database provides provide JDBC drivers
 - IBM, Microsoft, Oracle, ...
 - Java Transaction API
 - Naming Service
 - Java Naming and Directory Interface (more on this later)
 - J2EE Connector Architecture
 - Java Message Service (JMS)
 - There are others, but these are the main ones

J2EE – Connector Architecture

- J2EE Connector Architecture
 - Provides a standard / portable API to use in Java components to access Enterprise Information Systems (EIS)
 - Typically provided by the EIS vendor
 - IBM provides connectors for
 - Calling an RPG or COBOL program
 - Accessing CICS

Everything is Components, Containers, Services and Connectors

***You develop your components using the help of the services
and connectors then deploy to a container!***



IBM Software Group

JDBC
J2EE Connector Architecture (JCA)
Java Naming and Directory Interface (JNDI)

WebSphere software



@business software

© 2003 IBM Corporation

JDBC

- Standard Java interface for running SQL
 - Independent of any single Database vendor
 - Works with DB2 UDB, Cloudscape, Informix, Microsoft SQL Server, Oracle, Sybase, ...
 - Lots of JDBC articles, books, web sites, ...
- Development time:
 - You write the code using JDBC and standard SQL
- Deployment time:
 - You specify which Database to use
 - This is configured in the "Web Deployment Descriptor" for you Web project (web.xml)
 - More on this latter!

JDBC Java Interfaces

- Use JDBC to:
 - Directly read, write and update DB2 UDB for iSeries using SQL
 - Call stored procedures written using RPG, COBOL or Java
- JDBC Terms
 - Connection (`java.sql.Connection`)
 - Live connection (session) with a specific database
 - Statements are associated with a Connection
 - Statement (`java.sql.Statement`)
 - Java interface used for executing SQL
 - PreparedStatement (`java.sql.PreparedStatement`)
 - Same as Statement, except it is precompiled for performance
 - Use PreparedStatement if you are running the same statement multiple times
 - CallableStatement (`java.sql.CallableStatement`)
 - Java interface used for calling stored procedures

JDBC Connection Pooling

- Problem:
 - In a typical web application there maybe 1000s of requests coming in every minute
 - There is a lot of overhead to create and close a connection to the database for every request
 - But you need a connection to run SQL queries
- Solution:
 - Use Connection pooling
 - Web App container creates JDBC connections in a pool
 - Instead of creating a Connection in your code you:
 - Ask the pool for a connection
 - Use the connection to run SQL statements
 - Return the connection to the pool so it can be reused

J2EE Connector Architecture (JCA)

- J2EE Connector Architecture provides a standard architecture for accessing various Enterprise Information Systems (EIS) from your Java application
 - RPG and COBOL programs or service programs
 - CICS
 - Enterprise Resource Planning (ERP) systems
- Resource Adapters
 - Provided by each vendor for their EIS system
 - Plugs into the application server and handles things like:
 - Communications
 - Transactions
 - Security
 - WDSO provides resource adapter for calling RPG and COBOL programs
 - `D:\WDSO\iseries\eclipse\plugins\com.ibm.etools.iseries.webtools_5.1.2\lib\iseriespgmcall.rar`

JNDI – How To Find Things

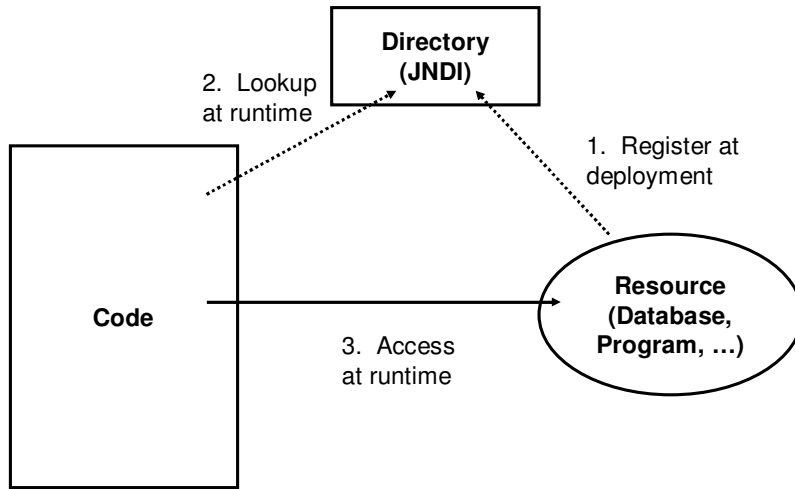
- Problem:
 - Many of the components to a web application are distributed across multiple servers
 - Components need to be dynamically changed or updated without having to modify the code and recompile
 - Changing a JDBC Database connection from the development database to the production database
- This is the domain of enterprise naming and directory servers
- Many different naming and directory server packages available
 - Need a standard way to interface with them so the code is not directly tied to a specific vendor's implementation

JNDI – How To Find Things

- Solution
 - Java Naming and Directory Interface (JNDI)
 - Allows developer to write programs that can lookup resources dynamically at runtime
 - Can easily change which database is used for JDBC without recompiling
 - Standard extension to the Java platform for connecting to and interfacing with naming and directory servers

- Use JNDI to locate other J2EE resources:
 - Database connections (JDBC)
 - RPG or COBOL program call resource adapters (JCA)
 - Message queue (JMS)

- JNDI is just an interface for locating services / components



Programming With JNDI

- Two steps to working with JNDI
 - ▶ Writing you application using JNDI to locate components
 - ▶ Or have one of the wizards generate the code
 - ▶ iSeries Program Call wizard
 - ▶ Database pages wizard
 - Configuring the naming and directory server with the components you need at runtime
 - ▶ For example: Define the Database connection and pooling information
 - ▶ In a Web application with is done by configuring the properties for the Web application server
 - ▶ More on this later...

Code Example: Using JNDI to Lookup JDBC Connection

```
// import JDBC Interfaces
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;

// import JNDI classes and interfaces
import javax.naming.Context;
import javax.naming.InitialContext;

...

// Retrieve JNDI context
Context initialContext = new InitialContext();
// Lookup JDBC DataSource using JNDI
DataSource datasource = (DataSource)
    initialContext.lookup("jdbc/customer");

// Use JDBC Data Source to run SQL query
Connection connection = datasource.getConnection();
Statement statement = connection.createStatement();
ResultSet results = statement.executeQuery("SELECT * FROM
CUSTOMER");

// Do something with the result set
```



IBM Software Group

J2EE – The Tools

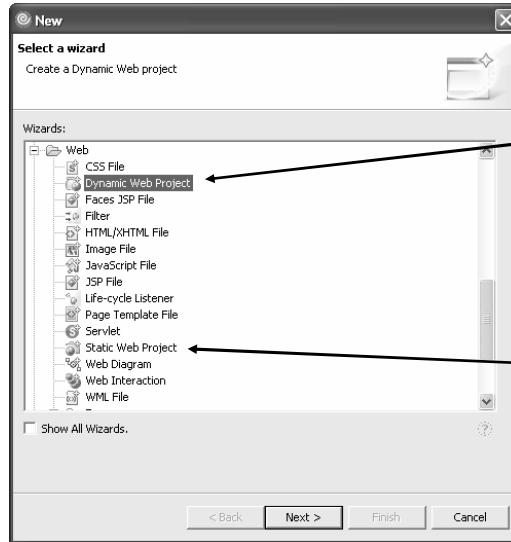
WebSphere software



@business software

© 2003 IBM Corporation

New Web Project Wizard



Use J2EE Web Project for Web applications with **dynamic content** (servlets and JSPs)

Use "Static Web Project" for Web sites with **no dynamic content** (just HTML and graphics)

New Web Project Wizard

New Dynamic Web Project
Create a standalone Dynamic Web project or add it to a new Application project.

Name: Sample Web Project
Project location: C:\Documents and Settings\yantzi.TOROLAB\IBM\rationalsdp6.1 Browse...

Hide Advanced <<
Servlet version: 2.4
Target server: WebSphere Application Server v6.0 New...

Add module to an EAR project.
EAR project: Sample Web ProjectEAR New...

Context Root: samples

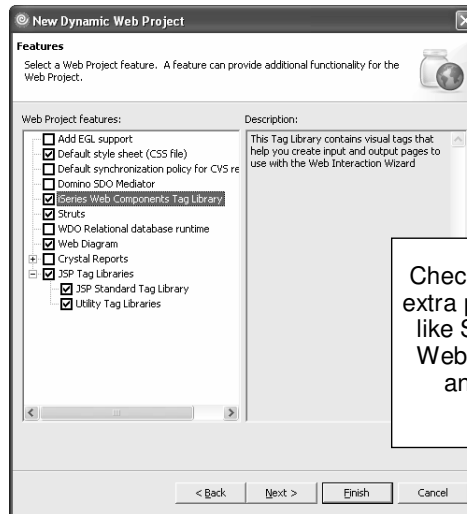
Back Next >

Select Servlet (J2EE) version and target application server

Context root is part of URL that identifies this web application:
[http://www.ibm.com/samples/...](http://www.ibm.com/samples/)

Each J2EE Web project is associated with an **Enterprise Application Resource (EAR)** project

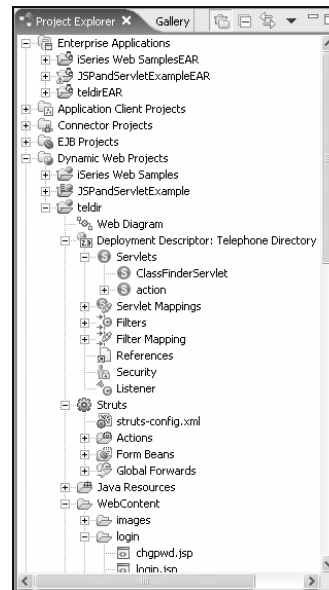
New Web Project Wizard



Checkboxes to add extra project support like Struts, iSeries Web Components and JSP Tag libraries

Project Explorer

- Project Explorer
 - Shows all of your J2EE projects and resources in the workspace
 - Enterprise applications projects
 - J2EE Connectors projects
 - Enterprise JavaBeans projects
 - Dynamic Web projects
 - Databases
 - Test servers
- Intelligent drill down
 - Expand Deployment descriptors
 - Delete, Add, Edit contents directly from view
- Provides context-based actions on selected objects



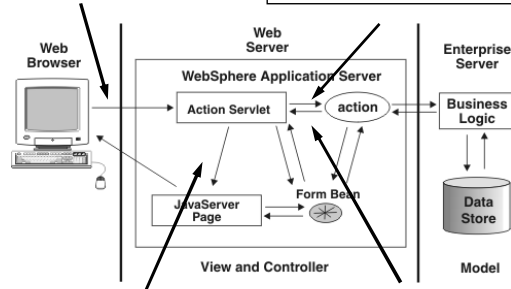
Struts

- What is Struts?
 - Open source framework for developing web applications
 - Sponsored by the Apache Software Foundation
 - Supports developing Web based applications that follow the Model-View-Controller (MVC) design
- How does it work?
 - Struts provides the Controller
 - You provide the Model and the View
 - Struts also provides:
 - Custom tag libraries for Internationalization (I18N)
- Struts is supported by the WebSphere Studio development tools and the WebSphere Application Server runtime

Struts Overview

1. Incoming request from Browser

2. Struts ActionServlet looks up the corresponding action class for the request, populates the a form bean with incoming data and passes the request to the action class



4. ActionServlet forwards the request to the corresponding JSP which sends result to Browser as an HTML page

3. Action class processes the request (using i5OS *PGMs and *SRVPGMs) and places results in form bean and returns to ActionServlet

How Does It Work - Controller

- Struts ActionServlet is the Controller
 - Uses configuration file (struts-config.xml) to determine:
 - ActionFormBeans
 - Uses the <form-bean> tag
 - Global Forwards
 - Uses the <forward> tag
 - ActionMappings
 - Uses the <action> tag
- What do you do?
 - Create an ActionFormBean to send data between view and model
 - Write an Action class for each request
 - Configure ActionMapping for each request

How Does It Work – Controller

- ActionForm (form bean)
 - Stores and validates data from incoming HTML pages
 - Transfers data between the view and the model
 - Can be stored in either the session or the request
 - Upon receiving a request, the controller populates the associated ActionForm with data from the request and forwards the form bean to the Action class
 - ActionForm can optionally perform validation on input
 - Override the method:
 - `validate(ActionMapping mapping, HttpServletRequest request)`
 - Struts handles redisplaying input page with error messages

How Does It Work - Model

- Action class
 - Handle error checking and invokes business logic (model)
 - This is the part you have to code!
 - Implement the method:
 - ```
public ActionForward execute(ActionMapping mapping,
 ActionForm form,
 HttpServletRequest request,
 HttpServletResponse response)
```
  - Return ActionForward instance to specify where control goes next
    - Typically a JSP to return results of Action to browser
    - Maps to a **Global Forward** (defined in struts-config.xml)
- ActionMapping
  - This is how the ActionServlet determines which incoming URL requests get mapped to which Action classes
  - ActionMappings are stored in the struts-config.xml file, requires the following info:
    - Incoming URI
    - Name of Action class
    - Name of the form bean used by this Action

## How Does It Work – View

- Struts includes tag libraries to help you
  - Create internationalized applications
    - Load in translated messages
    - Format dates and numbers for different locales
  - Automatically validate user input
    - Automatically redisplay input page with error messages from validation
  - Pre-fill HTML entry fields with data from your application

## Struts Tools in Development Studio Client

- Enable Web projects for Struts via project preference
  - Creates struts-config.xml
  - Adds Struts tag libraries to the project
- Struts node in the Project Explorer tree
- Wizards to create
  - Form beans
  - Action classes
- Special Struts Configuration File Editor
  - You don't have to know XML or the XML syntax used in the struts-config.xml

IBM eServer iSeries

## Struts Configuration File Editor

**Action Mappings**

Local Forwards

The following Action Mappings are defined for this config file

- /webIntSignonPromptAction
- /webIntChangePasswordAction
- /chgpwd
- /deleteprofile
- /getperson
- /login
- /logout
- /logout2
- /manage
- /manages
- /refresh
- /register
- /reportto
- /samemanager
- /search
- /update
- /wcl

Add, remove and edit Struts actions

Add Remove

**Action Mapping attributes**

Attributes of the selected Action Mapping

Type:  Browse... Edit...

Forward:

Include:

Parameter:

Display Name:

Description:

Default

**Form Bean Specification**

Form Bean Specifications of the selected Action mapping

Form Bean Name:

Scope:

Attribute:

Validate:

Prefix:

Suffix:

Input:

**Action Mapping Mapping Extensions**

Mapping Extension of the selected Action Mapping

Property:  Value:  Add Remove

View XML source for file

Action Mappings | Global Forwards | Form Beans | Data Sources | Source

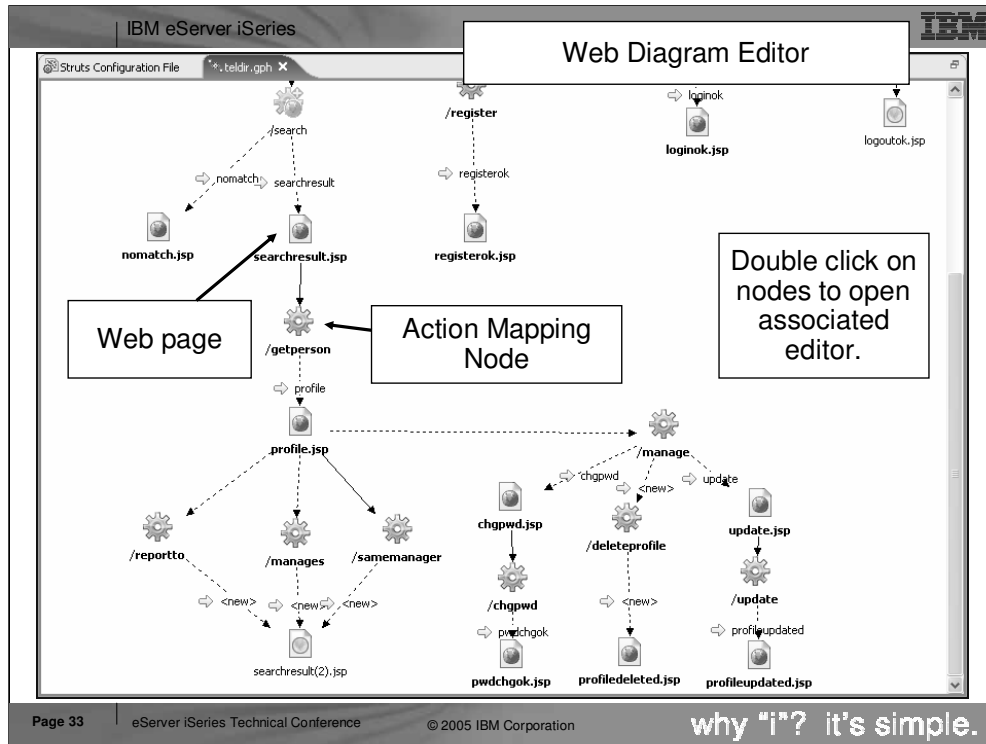
Page 31 eServer iSeries Technical Conference © IBM Corporation

why "i"? it's simple.

## Struts Tools in Development Studio Client

- Web Diagram Editor
  - Shows a graph view of your Struts based Web application
  - Can be used as a central point for working with the Web app
  - Useful for:
    - Adding new actions / form beans / JSPs
    - Editing existing actions / form beans / JSPs
    - Documenting overall architecture of the Web application
  - As parts are added, deleted or updated in the Web diagram editor the struts-config.xml file is updated with changes







IBM Software Group

# Server Tool

Configuring the Test Environment

**WebSphere** software



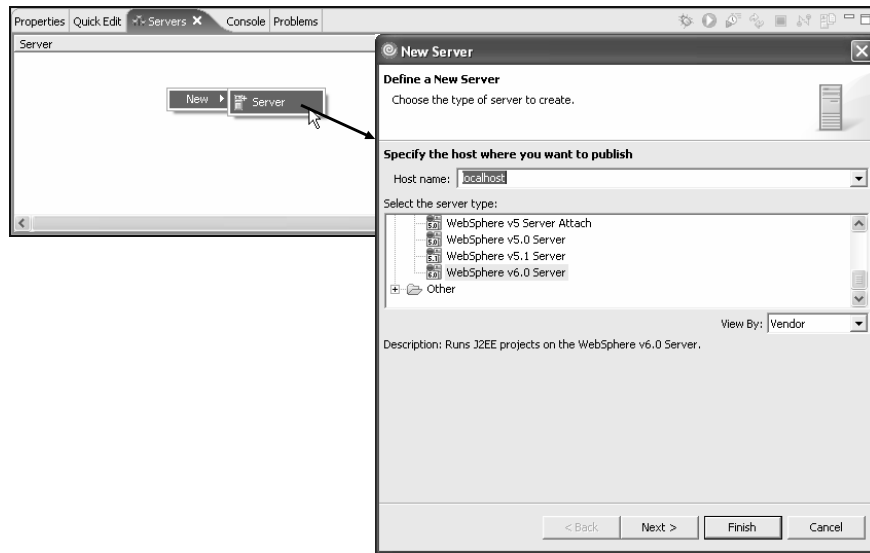
**@business software**

© 2003 IBM Corporation

## Servers

- The test environment uses “servers” to run and debug your web projects
  - \*\*\* These are Web Application Servers, not hardware servers
  - Run locally on PC or attach to remote server
  - The first time you test a Web project a server is created for you
  - Create your own, customize them and associate web projects with different servers
  
- Types of Servers and Server Configurations
  - WebSphere Application Server V6.0, V5.1 and V5.0
    - Included in WDS*c*
  - WebSphere Application Server Express V5.1 and V5.0
    - Included in WDS*c*
  - Apache Tomcat V5.0, V4.1, V4.0 and V3.2

## Creating a New Server



# Editing a Server Configuration

**Change to SOAP if you have problems starting your server**

**Basic startup properties of server can be modified here. More advanced properties are configured through normal admin interface....**

**Double click on the server to open it in the editor**

| Server                            | Host name | Status  |
|-----------------------------------|-----------|---------|
| WebSphere v6.0 Server @ localhost | localhost | Stopped |

# Configuring the Server

- New
- Open
- Delete
- Debug
- Start
- Profile
- Restart
- Stop
- Disconnect
- Publish
- Monitoring
- Add and remove projects...
- Enable/Disable EGL Debugging
- Run universal test client
- Restart universal test client
- Run administrative console
- Reconnect debug process
- Create tables and data sources
- Run external admin script

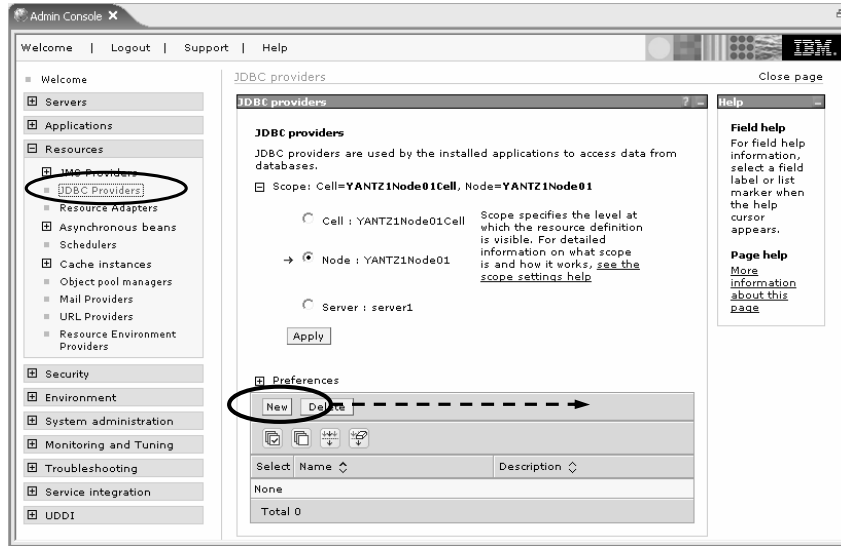
Server must be started first

Use the standard Web administration client to configure the test server

## JDBC: Defining Connection Pools in the Test Environment

- Earlier we looked at JDBC, what it is and why you would use it
- Now we will look at how to configure JDBC connection pools in the WebSphere test environment
  - First you need to add the required JDBC driver to the **providers list**
  - Then you can define a connection pool for the JDBC driver

# JDBC: Adding a Driver to the Provider List





## JDBC: Adding a Driver to the Provider List - 2

The screenshot shows the IBM Admin Console interface. On the left is a navigation tree with categories like Servers, Applications, Resources, Security, Environment, System administration, Monitoring and Tuning, Troubleshooting, Service integration, and UDDI. The main content area is titled 'JDBC providers' and contains a 'New' wizard. The wizard has a 'Configuration' tab and a 'General Properties' dialog box. The dialog box has three steps: Step 1: Select the database type (DB2), Step 2: Select the provider type (DB2 UDB for iSeries (Toolbox)), and Step 3: Select the implementation type (Connection pool data source). A callout box with a dashed arrow points to Step 2, containing the text 'Select the DB2 UDB for iSeries (Toolbox) driver'. On the right side of the wizard, there is a 'Help' section with 'Field help' and 'Page help' links.

The screenshot shows the IBM Admin Console configuration interface. On the left is a navigation tree with categories like Welcome, Servers, Applications, Resources, Security, Environment, System administration, Monitoring and Tuning, Troubleshooting, Service integration, and UDDI. The main area is titled 'Configuration' and contains a 'General Properties' section with the following fields:

- Scope:** cells:YANTZ1Node01Cell:nodes:YANTZ1Node01
- Name:** iSeries Toolbox Driver
- Description:** IBM Toolbox for Java JDBC Driver for remote DB2 connections on iSeries. This driver is recommended over the IBM Developer Kit for Java JDBC.
- Class path:** \${OS400\_TOOLBOX\_JDBC\_DRIVE\_R\_PATH}/jt400.jar
- Native library path:** (empty)
- Implementation class name:** com.ibm.as400.access.AS400JDBCConnectionPoolData\$

At the bottom of the configuration window are buttons for 'Apply', 'OK', 'Reset', and 'Cancel'. A callout box with a white background and black border contains the text: 'Accept the defaults and click Data sources'. In the top right corner of the configuration window, there is an 'Additional Properties' section with a link for 'Data sources (Version 4)'. On the far right, there is a 'Help' section with 'Field help' and 'Page help' links.

**Configuring A Data Source**

**JDBC providers**

**JDBC providers > iSeries Toolbox Driver > Data sources > WDS6 Demo**

A data source is used by the application to access data from the database. JDBC provider, which supplies the specific JDBC driver implementation class.

Configuration

Test connection

**General Properties**

- Scope: [cells:YANTZ1Node01:Cell:node01:YANTZ1Node01]
- Name: WDS6 Demo
- JNDI name: jdbc/wdsdemo
- Use this Data Source in container managed persistence (CMP)

**Additional Properties**

- Connection pool properties
- WebSphere application Server\_data source properties
- Custom properties

**Related Items**

- J2EE Connector Architecture (JCA) authentication data entries

**Description**

New JDBC datasource

Category

Data store helper class name

Select a data store helper class

Data store helper classes provided by WebSphere Application Server

**DB2 for iSeries data store helper**

[com.ibm.websphere.readadapter.DB2AS400DataStoreHelper]

Specify a user-defined data store helper

Enter a package-qualified data store helper class name

Component-managed authentication alias

Component-managed authentication alias: [YANTZ1Node01/yantzi...]

**JNDI name used in source code**

**Click here to customize pool properties (next slide)**

**Click here to define authentication information. Then select entry here.**

**iSeries hostname defined lower in page**

Page 43 | eServer iSeries Technical Conference | © 2005 IBM Corporation | why "i"? it's simple.

IBM eServer iSeries

Admin Console

Welcome | Logout | Support | Help

Welcome  
 Servers  
 Applications  
 Resources
 

- JMS Providers
  - Default messaging
  - WebSphere MQ
  - Generic
  - V5 default messaging
- JDBC Providers
- Resource Adapters
- Asynchronous beans
  - Schedulers
- Cache instances
  - Object pool managers
- Mail Providers
- URL Providers
- Resource Environment Providers

Security  
 Environment
 

- Virtual Hosts
- WebSphere Variables
- Shared Libraries
- Replication domains

Naming  
 System administration  
 Monitoring and Tuning

**JDBC providers**

JDBC providers > iSeries Toolbox Driver > Data sources > WDSF Demo >

**Connection pools**

Connection pool properties that can be modified to change the behavior of the J2C connection pool manager. Default values are provided for non-production use. Review and possible modification of these configuration values are recommended.

Configuration

| General Properties                                  | Additional Properties                                                        |
|-----------------------------------------------------|------------------------------------------------------------------------------|
| Scope<br>cells:YANTZ1Node01,Cell:nodes:YANTZ1Node01 | <input type="checkbox"/> <a href="#">Advanced connection pool properties</a> |
| Connection timeout<br>180 seconds                   | <input type="checkbox"/> <a href="#">Connection pool custom properties</a>   |
| Maximum connections<br>10 connections               |                                                                              |
| Minimum connections<br>1 connections                |                                                                              |
| Reap time<br>180 seconds                            |                                                                              |
| Unused timeout<br>1800 seconds                      |                                                                              |
| Aged timeout<br>0 seconds                           |                                                                              |
| Purge policy<br>EntirePool                          |                                                                              |

Apply OK Reset Cancel

**Field help**  
For field help information, select a field label or list marker when the help cursor appears.

**Page help**  
[More information about this page](#)

Page 44 eServer iSeries Technical Conference © 2005 IBM Corporation why "i"? it's simple.

IBM eServer iSeries

Admin Console

Welcome | Logout | Support | Help

One last thing, define value for OS400\_TOOLBOX\_JDBC\_DRIVER\_PATH on Environment > WebSphere Variables page

|                                |                                                                                                                |        |
|--------------------------------|----------------------------------------------------------------------------------------------------------------|--------|
| DB2UNIVERSAL JDBC DRIVER_PATH  |                                                                                                                | cells: |
| DB2 JDBC DRIVER_PATH           |                                                                                                                | cells: |
| DEPLOY_TOOL_ROOT               | \${WAS_INSTALL_ROOT}/deploytool/itp                                                                            | cells: |
| DRIVER_PATH                    | \${WAS_INSTALL_ROOT}                                                                                           | cells: |
| INFORMIX JDBC DRIVER_PATH      |                                                                                                                | cells: |
| JAVA_HOME                      | C:\Program Files\IBM\Rational\SDP\6.0\runtimes\base_v6/java                                                    | cells: |
| LOG_ROOT                       | \${USER_INSTALL_ROOT}/logs                                                                                     | cells: |
| MQJMS_LIB_ROOT                 | \${MQ_INSTALL_ROOT}/java/lib                                                                                   | cells: |
| MQ_INSTALL_ROOT                | \${WAS_INSTALL_ROOT}/lib/WMQ                                                                                   | cells: |
| MSSQLSERVER JDBC DRIVER_PATH   |                                                                                                                | cells: |
| ORACLE JDBC DRIVER_PATH        |                                                                                                                | cells: |
| OS400 NATIVE JDBC DRIVER_PATH  |                                                                                                                | cells: |
| OS400 TOOLBOX JDBC DRIVER_PATH | C:\Program Files\IBM\Rational\SDP\6.0\wrd_shared\ eclipse\plugins\com.ibm.etools.iseries.toolbox_6.0.0\runtime | cells: |
| STORE JDBC DRIVER_PATH         |                                                                                                                | cells: |

Page: 1 of 2 Total 28

Page 45 eServer iSeries Technical Conference © 2005 IBM Corporation why "i"? it's simple.

## J2EE Connector Architecture

- Earlier we looked at the J2EE Connector Architecture, what it is and why you would use it
- Now we will look at how to configure JCA in the WebSphere test environment
  - First you need to import the required **Resource Adapter**
  - Then define instance for your program

## JCA: Importing Resource Adapter

Admin Console X

Welcome | Logout | Support | Help

Resource adapters Close page

Resource adapters Help

**Resource adapters**

A resource adapter is an implementation of the J2EE Connector Architecture Specification that provides access for applications to resources outside of the server or provides access for an Enterprise Information System (EIS) to applications on the server. It can provide application access to resources such as DB2, CICS, SAP and PeopleSoft. It can provide an EIS with the ability to communicate with message driven beans that are configured on the server. Some resource adapters are provided by IBM; however, third party vendors can provide their own resource adapters. A resource adapter implementation is provided in a resource adapter archive file; this file has an extension of .rar. A resource adapter can be provided as a standalone adapter or as part of an application, in which case it is referred to as an embedded adapter. Use this panel to install a standalone resource adapter archive file. Embedded adapters are installed as part of the application install.

Scope: Cell=YANTZ1Node01Cell, Node=YANTZ1Node01

Cell : YANTZ1Node01Cell Scope specifies the level at which the resource definition is visible. For detailed information on what scope is and how it works, [see the scope settings help](#)

Node : YANTZ1Node01

Server : server1

Preferences

Select: Name

|                          |                                       |
|--------------------------|---------------------------------------|
| <input type="checkbox"/> | S16 JMS Resource Adapter              |
| <input type="checkbox"/> | WebSphere Relational Resource Adapter |

Total 2

Resource adapters are shipped in RAR files, click Install RAR

## Adding Resource Adapters to the Test Environment

Admin Console

Welcome | Logout | Support | Help

Resource adapters

Close page

Install RAR File

RAR files can be installed using two methods. You can choose to upload a RAR file from local file system or you can specify an existing RAR file on a server.

Path

Local path:

Specify path  
C:\Program Files\IBM\Rad\ Browse...

Server path:

Specify path

Scope  
Node  
YANT21Node01

Next Cancel

`%install_dir%\radi_prod\eclipse\plugins\com.ibm.etools.iseries.webtools.ae_6.0.0\lib\iseriespgmcall.rar`



**Resource adapters** > **iSeries Program Call Resource Adapter**

A resource adapter is an implementation of the J2EE Connector Architecture, resources outside of the server or provides access for an Enterprise Information System (EIS) to resources such as DB2, CICS, SAP and Pepl with message driven beans that are configured on the server. Some vendors can provide their own resource adapters. A resource adapter implementation file has an extension of .rar. A resource adapter can be provided as a stand-alone or as an embedded adapter. Use this panel to install a stand-alone resource adapter.

Configuration

**General Properties**

- Scope: [cells:YANTZ1Node01Cell:nodes:YANTZ1Node01]
- Name: iSeries Program Call Resource
- Description: [ ]
- Archive path: [\${CONNECTOR\_INSTALL\_ROOT}/iseriespgmcall.rar]
- Class path: [\${CONNECTOR\_INSTALL\_ROOT}/iseriespgmcall.rar]
- Native path: [ ]
- Thread pool alias: Default

**Additional Properties**

- J2C connection factories
- View Deployment Descriptor

Apply OK Reset Cancel

Page 49

IBM  
Simple.

IBM eServer Series Technical Conference © 2005 IBM Corporation

Resource adapters > User's Program Call Resource Adapter > J2E connection factories > WDSDemo Program

The connection factory represents one set of connection configuration resource-ref descriptors that refer to the ConnectionFactory, a list of configuration properties. In addition to the arbitrary set there are several standard configuration properties that apply connection pool manager in the application server run time.

Configuration

**Defining connection factory information**

**JNDI name used in source code**

**Same security information from JDBC provider**

**Click here to specify server information**

General Properties

- \* Scope: cell:/YANTZ1Node01/Cell:nodes/YANTZ1Node01
- \* Name: WDSDemo Program
- JNDI name: jca/wdsdemo
- Description:

Additional Properties

- Connection pool properties
- Advanced connection factory properties
- Custom properties

Related Items

\* Connection factory interface: javax.resource.cci.ConnectionFactory

Category:

Component-managed authentication alias: Component-managed authentication alias: YANTZ1Node01/yantzi

authentication

- Component-managed authentication alias (deprecated in V6.0, use resource reference settings instead)
- Resource reference (deprecated in V6.0, use resource reference settings instead)
- Resource ID
- Mapping-configuration alias (deprecated in V6.0, use resource reference authentication settings instead)
- (none)

Apply OK Reset Cancel

Page 50 why it's simple.



IBM Software Group

## Summary

**WebSphere** software



**@business software**

© 2003 IBM Corporation

## Summary

- Java 2 Enterprise Edition
  - Standards based model for developing applications in Java
    - Web applications
    - Enterprise applications
    - Client / Server applications
  - Supported by major Web Application Server vendors
  
- WebSphere Development Studio Client
  - Tools to make developing J2EE applications productive and easy
  - Customized editors for all the various technologies
    - Cascading style sheets, JSPs, HTML, animations, server configurations, Web deployment descriptor, ...
  - J2EE project navigator for easy management of your Web applications
  - Embedded test server for running, testing and debugging your Web applications

## Additional Resources

- J2EE Technologies  
<http://java.sun.com/j2ee>
- Struts  
<http://jakarta.apache.org/struts/>
- Rational Workbench Tools  
[www.ibm.com/rational](http://www.ibm.com/rational)
- IBM DeveloperWorks  
[www.ibm.com/developer](http://www.ibm.com/developer)



## Trademarks & Disclaimers

© IBM Corporation 1994-2002. All rights reserved.  
References in this document to IBM products or services do not imply that IBM intends to make them available in every country.  
The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

|                   |            |  |  |
|-------------------|------------|--|--|
| AS/400            | IBM (logo) |  |  |
| AS/400e           | iSeries    |  |  |
| e (logo) business | OS/400     |  |  |
| IBM               |            |  |  |

Lotus, Freelance Graphics, and Word Pro are registered trademarks of Lotus Development Corporation and/or IBM Corporation.  
Domino is a trademark of Lotus Development Corporation and/or IBM Corporation.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.  
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 trademarks of Microsoft Corporation in the United States, other countries, or both.  
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.  
SET and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.  
Other company, product and service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information in this presentation concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. Contact your local IBM office or IBM authorized reseller for the full text of the specific Statement of Direction.

Some information in this presentation addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. 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 IO 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.

Photographs shown are of engineering prototypes. Changes may be incorporated in production models.

## Disclaimer

- **Acknowledgment:**

This presentation is a collaborative effort of the IBM Toronto AS/400 Application Development presentation team, including work done by:

Don Yantzi, Phil Coulthard, George Farr, Claus Weiss, David Slater, Alison Butteril, Linda Cole

- **Disclaimer:**

The information contained in this document has not been submitted to any formal IBM test and is distributed on an as is basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customers' ability to evaluate and integrate them into the customers' operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will result elsewhere. Customers attempting to adapt these techniques to their own environment do so at their own risk.

- **Reproduction:**

The base presentation is the property of IBM Corporation. Permission must be obtained PRIOR to making copies of this material for any reason.