Application Development in a Mobile World

Duncan Winn
WebSphere Technical Sales
duncwinn@uk.ibm.com

3rd November 2011
Agenda

- My story so far
- Middleware and Application Server Architecture
- Enterprise Requirements
- Programming Models
- Mobile
My Story...
...a bit about me!
What is the most interesting or impressive bit of technology you've seen over the past year?
What do you think of when you hear IBM?
Smarter Solutions through hardware, software and services:

URL: ibm.com/start/uk

URL: 2010 UK&I Animation on YouTube
IT Career Milestones

• Software Developer / Porting Specialist
  – Porting IBM SW Products onto HP-UX Itanium and Solaris x86

• Management
  – Java Monitoring and Diagnostic tools for Java

• WebSphere Technical Specialist
  – Client Technical Professional / Technical Pre Sales
I'm paid to learn and talk!
What is Middleware?
What is Middleware?

“middleware: the kind of word that software industry insiders love to spew. Vague enough to mean just about any software program that functions as a link between two other programs...

... precise definitions can get all messy. Avoid using at all costs…”
What is Middleware?

“middleware: the kind of word that software industry insiders love to spew. Vague enough to mean just about any software program that functions as a link between two other programs...

... precise definitions can get all messy. Avoid using at all costs…”

“middleware: is at the core of IT and is vital to understand”
Middleware is Plumbing

- You don't really see it
- Industry standards (and specifications) make plumbing easier and safer
- It allows complex, dispirit parts to connect together to form a whole.
- You can concern yourself with more important things
What is Middleware?

Software that connects “software components” or “people and their applications”.

Consists of a set of services that allows multiple processes, running on one or more machines, to interact: Middleware can be stretched across multiple systems or applications.

Provides for interoperability between distributed architectures, which support and simplify complex distributed applications.

It includes **web servers**, **application servers**, **ESBs** and similar tools that support application development and delivery.

Middleware is especially integral to modern information technology based on XML, SOAP, Web services, and service-oriented architecture (SOA).

Middleware sits "in the middle" between the application code and the run-time infrastructure.

Middleware consists of a library of functions, enabling applications to page functions from a common library rather than re-create them per application.
Application Servers
Application Server Capabilities

• Build and deploy reusable application services quickly and easily.

• Run services in a secure, scalable, highly available environment, that traditionally would have been coded in your application.

• Connect software assets and extend their reach.

• Manage applications with ease.

• Grow as your needs evolve, reusing core skills and assets.
Enterprise Requirements
Can you list any?
Key Enterprise Considerations?

- Security
- System Health (RAS)
- Work Load Management
- Scalability and Throughput
- High Availability
- Disaster Recovery
- Performance
- Accessibility
- Response Times
- Resource Consumption
- Cost (TCO)
Considerations often forgotten...

- Reliability
- Availability
- Manageability
- Performance
- Automation
- Support
- Integration
- Security
- 3rd party components
- Backwards compatibility
- Availability
- Platform support
- Localization
- Value add products
- TCO vs. TCA
- Performance
- Documentation
- Scalability
- Data integrity
WebSphere Application Server – Focus areas

**Business Needs**

“Meet business objectives consistently, nimbly, cost-effectively”

“Enable applications to adapt to changing market conditions”

“Address extreme demands of clients & business models”

**Adoption Patterns**

Application Foundation

Intelligent Management

Extreme Transaction Processing
WebSphere Application Server Family
Multiple Business Models, Multiple Deployment Options

Customer Needs

WebSphere Application Server for Developers
WebSphere Application Server - Express
WebSphere Application Server Community Edition
Built on common WebSphere code
Built on open source technology

Capabilities

Ultimate scalability & performance; functional depth & breadth

Reduced acquisition costs; Small footprint...

Fast deployment of single app; low transaction volumes...

High transaction volumes, High Availability & Clustering, Centralized Administration, Advanced Web Services

WebSphere Application Server for z/OS
WebSphere Application Server Network Deployment
WebSphere Application Server Hypervisor Edition
WAS Tools Edition - Lowering Barriers to Developer Adoption
No charge WebSphere Application Server for use on developer desktop.


URL: WebSphere Tools Edition for Eclipse

- Install directly into the Eclipse environment already on developer’s machine
- Easily & quickly obtain Eclipse adapters for WAS environments (V7.0 and V8.0) via Eclipse updater
- Eclipse developers can now easily deploy newly developed apps to a WAS infrastructure

WAS for Developers - Tools Edition for Eclipse

- WAS 7 / WAS v8
- Development use license

WAS Developer Tools for Eclipse

- Server Adapter tools (v7, v8)

Available from developerWorks download site
WebSphere Application Server – Focus areas

**Business Needs**

“Meet business objectives consistently, nimbly, cost-effectively”

“Enable applications to adapt to changing market conditions”

“Address extreme demands of clients & business models”

**Adoption Patterns**

Application Foundation

Intelligent Management

Extreme Transaction Processing

**Application Foundation**

**Intelligent Management**

**Extreme Transaction Processing**
What is Intelligent Management?

Cloud
What does cloud mean to you?
Why Cloud Computing?

- “It takes me 3 months just to get a test environment”
- “Cannot take business opportunities as it is not practical to quickly provision IT for short-term opportunities”
- “Infrastructure budget: up to 90% maintenance, <10% innovation”
Typical WebSphere Application Server Environment

Applications deployed in Application Server “silos”

- Utilisation ≈ 15%
- How do I better utilise “unused” hardware?
- How do I provide good availability at peak?
- How do I reduce administration overhead?
- How do I increase flexibility?
- How do I do any of this without increasing risk?
Cloud Computing Delivery Models

**Private cloud**
- Privately owned and managed.
- Access limited to client and its partner network.
- Drives efficiency, standardization and best practices while retaining greater implementation control.

**Hybrid cloud**
- Access to client, partner network, and third party resources.

**Public cloud**
- Service provider owned and managed.
- Access by subscription.
- Delivers select set of standardized business process, application and/or infrastructure services on a flexible price per use basis.
Benefits of Private Cloud

Standardization and automation are key private cloud enablers

**Challenges**
- Low hardware utilization
- Heavily customized infrastructure
- Manual deployment and management processes

**Benefits**
- Increased utilization of infrastructure
- Standardized middleware
- Improved deployment speed
- Simplified applications management

![Diagram showing benefits and challenges of private cloud]

**VIRTUALIZATION** + **STANDARDIZATION** + **AUTOMATION** = Reduced Cost, Increased Flexibility
Speed the Development & Test Lifecycle Through Self Service Access to Repeatable Environments

IBM Workload Deployer & WAS Hypervisor Edition

Developer

1. Self service request

2. Rapidly access consistent & repeatable provisioned development & test environment
Application Virtualisation with Virtual Enterprise

Benefits*
- Consolidation ≈ 40%
- Utilisation ≈ 60%
- Failure impact ≈ -50%
- Administration ≈ -50%

Intelligent Infrastructure Management
- Meet SLAs by creating cluster capacity
- Health Management
- Application Management

* Illustrative values based on reduction of application server estate
Application Server Programming Models
Enabling Developers to Start With Open Source/Community Software & Benefit from IBM Value Add in Production
Enhanced developer productivity, user experiences, performance & integration:

**Enterprise JavaBeans (EJB) 3.1:** Enhanced developer productivity through simplification including testing outside of the application server, new timer support & asynch enhancements

**Contexts and Dependency Injection for Java (CDI) 1.0:** Faster time to value through tighter and simpler integration between Web & business logic tiers

**Java Persistence API (JPA) 2.0:** Enhanced developer ease of use & app performance through improved locking, mapping support & dynamic query construction

**Java Servlet 3.0:** Enhanced time to value through annotations and ease of integrating third party presentation frameworks

**Java API for RESTful Web Services (JAX-RS) 1.1:** Deliver better user experiences faster through integrated Web 2.0 programming model support

**JavaServer Faces (JSF) 2.0:** Enhanced developer productivity & end user experience through annotations & Facelets support

**Bean Validation 1.0:** Improved developer productivity through declarative means for describing validation constraints for data

**Java Architecture for XML Binding (JAXB) 2.2:** Improved performance via new default marshalling optimizations

**Enterprise Web Services 1.3:** Improved integration and reuse support

**Java API for XML-Based Web Services (JAX-WS) 2.2:** Developer productivity and security enhancements
Communications Enabled Applications (CEA)

Simply and rapidly add communications capabilities, like Click to Call and Cobrowsing, to any Web application leveraging existing skills and an SOA approach

Key Features:

Simplicity: 3 lines of code to add CEA into web app

Existing Skills: Java & JavaScript

Mobile Browser Widgets: Enable native look & feel

Telephony Access: REST & Web service interfaces to Make call, disconnect call & incoming call notifications

Web 2.0 Widgets: Customizable & extensible with iWidget support
  Click to Call
  Call Notifications
  Collaboration Dialog
  Contact Center Cobrowsing
  Peer to Peer Cobrowsing
  Two-way Synchronized Forms

PoC Friendly: Unit test environment & pre-tested with Avaya, Cisco & Nortel unified communications products

Ease of Access & Use: Integrated with WAS V8
OSGi Applications

Speed development, increase ease of use and reuse through the modularity, dynamism, and versioning capabilities of OSGi applied to web & enterprise applications

Key Features:

**Modular deployment and management:** Separate common libraries from application archives; manage them centrally and across many versions, concurrently

**Standards Based DI Framework:** POJO development model, with a container that manages injection of configuration, and controls activation & deactivation, integrated with the server

**In-place update:** Update applications modules without restarting the application

**Java Standards Layering:** Java standards such as transaction, security, & persistence can be mixed into the componentized apps as services

**SCA Integration:** Components can be decorated as SCA components to provide coarse grain SOA services
Web 2.0 & Mobile

Extend the reach of enterprise web applications across devices to deliver high quality user experiences

Enabling Mobile UI’s:
Dojo Core & Widget Infrastructure
Dojo Visualization
New Mobile Widget Library
Dojo Web Builder (Build optimization service)
Desktop & Mobile Demo Showcase

Accelerating Rich Internet Applications:
Touch-enabled desktop widgets
Maps components (tiled and vectors)
New Visualization widgets
Component updates: Dojo 1.6++, JAX-RS, etc

Common Mobile & RIA Building Blocks:
Directory Listing Service
File Upload Service (multipart)
Graphics Conversion Service (SVG/PNG/JPG/PDF)
Logging/Debug/Analytics Capture Service

Available as a Feature Pack supporting WAS v8, v7 & v6.1
Mobile: A New Frontier of Growth

Mobile B2C
- Increase customer satisfaction by enabling banking, insurance, and trading anywhere, anytime
- Reach customers in new ways through mobile applications, SMS, email

Mobile B2E & B2B
- Enable field employees for increased productivity
- Greater efficiency and accuracy in supply chain operations
- Exchange business information with partners securely

Mobile M2M
- Enable the exchange of data and events between businesses and machines
- Internet of Things - sensor events feeding information and driving a smarter
Why Is Mobile Different?

Mobile users require efficient and timely access to information.

Interactions are short and focused, interruptions are common.

Devices are often exclusively touch-based.

User interfaces must be easy and obvious.

- Screen real-estate is precious.
- Typing should be minimized.

- Applications must still be usable when out of wireless coverage.
- Timeliness of data must be communicated.
- Security is critical.
- Often used for monitoring as opposed to active consumption.
- Social interactions are important.
- Mobile hardware and user interfaces evolve much faster than the typical enterprise software cycle.

Mobile users today expect high-fidelity access to the same information they have on the desktop, presented in an easy-to-learn, mobile-friendly (often touch-friendly) format.
A spectrum of mobile development approaches

<table>
<thead>
<tr>
<th>Web Application</th>
<th>Mobile Web Application</th>
<th>Hybrid Mobile Application</th>
<th>Native Mobile Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Desktop and mobile using open web programming models</td>
<td>Mobile only using open web client programming models</td>
<td>Mobile only, app runs on device leveraging open web via JavaScript bridge</td>
<td>Mobile only, using native languages.</td>
</tr>
<tr>
<td>Limited to no device-specific functionality</td>
<td>Off-line capabilities</td>
<td>Native device capabilities</td>
<td>Native appearance, device capabilities, performance</td>
</tr>
</tbody>
</table>

Mobile Browser Execution

Application Store download and install

IBM Web 2.0 & Mobile Feature Pack for WebSphere

Existing SOA Enablement

Richness of Mobile Presentation / Services

Portability (cross-device reuse)

Maintenance Cost (TCO)
IBM Mobile Technology Preview Downloads

The IBM Mobile Technology Preview download is delivered as a zip file. It can be placed anywhere in the file system and unzipped to expose the contents listed below. The Overview document, included in the zip file, is a good starting place to learn about the IBM Mobile Technology Preview and the features provided. Please see the Getting Started Guide, linked from the Overview document, to familiarize yourself with the file contents, point you to the code, and give you a head start to running the samples.

Included with this Technology Preview are the following:

- **Hybrid Enablement Framework** for building cross-platform mobile applications with HTML, CSS, and JavaScript. The IBM Mobile Technology Preview is utilizing the PhoneGap open source framework as a means to quickly deliver hybrid capabilities. Other hybrid approaches exist and may be used in addition to, or instead of, PhoneGap in future technology preview deliveries.

- **Notifications** a RESTful notification service and a client side notification library enables enterprise logic to post notifications to one or more target devices.
Summary

• Application servers provide the **environment** to run **solutions** and **integrate** them with **every platform and system** as “business application services”.

• WebSphere Application Server is available on a wide range of platforms and in multiple packages to meet specific and varied business needs.

• Many requirements need to be considered when developing enterprise grade applications.

• Various programming models exist to speed mobile application development.

• Mobile offers a new frontier for growth within business and the IT industry.
Assignment
Assignment

Enterprise grade applications are built using a variety of programming models for example:

- JEE6 (including, but not limited to, EJB 3.1, JPA2.0, Java Servlet 3.0, and JAX-RS 1.1)
- OSGI
- SCA (Service Component Architecture)

The middleware platforms that support these applications require careful consideration of concepts such as load balancing, server utilisation (including virtualization and provisioning), high availability (including clustering and disaster recovery), security, ACID transactions and response times.

Compare and contrast the business and technical challenges of developing mobile applications using the above programming models and runtime capabilities, that are key to enterprise grade environments. Your essay shall show appreciation to the key concepts and use cases of the above technologies?