OSGi Alliance
From Vision to Execution (and back)
Jim Colson, Distinguished Engineer
IBM
A view of the market
Pervasive devices growing more than 10 times faster than PCs… 22 billion+ by 2008*

Nearly 70% of all enterprises will deploy a mobility/pervasive solution by 2005**

75% of knowledge workers mobile at least 25% of the time are using "pervasive" device (2005/2006)**

Enable the mobile workforce

Integrate data from physical assets

Support multiple device types over multiple networks seamlessly

Optimize business processes and asset utilization

*Source: IDC 2004

**Source: Meta Group 2003
Device Types

Platform
Aftermarket, On Device Developer Community
Extensible networked data aware applications
High back end affinity

Connectable
Built in networked data aware applications
Remote management and content
Limited back end affinity

Standalone
Conventional Embedded Market
No data network connectivity
No back end (data) affinity
Device Platform – The challenge

ISV’s, Developers

NEED to build their application to support MANY DEVICES

Enterprises & Service Providers

NEED APPLICATIONS that run on MANY DEVICES

OEMs, ODMs, Tier 1s

NEED APPLICATIONS that run on their devices
Device Platform – Success measures

NEED to build their application to support MANY DEVICES
ISV’s, Developers

Devices that reuse existing development skills
Common requirements vocabulary

Apps & services that run across evolving devices

Enterprises & Service Providers

Large ISV Community

OEMs, ODMs, Tier 1s

Widespread device deployment

Low total cost of ownership for devices, services, and applications.

NEED APPLICATIONS that run on MANY DEVICES

NEED APPLICATIONS that run on their devices

© copyright 2004 by OSGi Alliance. All rights reserved.
NEED to build their application to support MANY DEVICES

ISV’s, Developers

Devices that reuse existing development skills

Common requirements vocabulary

End-to-end tooling

Enabled by OSGi

Infrastructure

Enterprises & Service Providers

Apps & services that run across evolving devices

Large ISV Community

OEMs, ODMs, Tier 1s

Widespread device deployment

Portable middleware

Low total cost of ownership for devices, services, and applications.

NEED APPLICATIONS that run on MANY DEVICES
What’s new since OSGi WC 2003?
Creation of Mobile Expert Group (MEG)

• Created by OSGi Board of Directors on Friday, October 10, 2003.
  ➢ Formation driven by 2 key industry players Motorola & Nokia, with strong initial support by Deutsche Telekom, Espial, IBM & ProSyst Software
  ➢ Will define requirements and specifications to tailor and extend the OSGi Service Platform for mobile devices that are data-capable, and also capable of connecting to wireless networks
  ➢ Starting with development of an OSGi specification for the management of CDC Foundation based environments on mobile digital handsets

Expands the number of industries adopting and enhancing the OSGi Service Platform
Creation OSGi Users Forum (Japan)

Foundation: September 28, 2004
Founder Members: IBM Japan, Mitsubishi Electric, NEC, NTT, SHARP, TOSHIBA
Chairman: Prof. Hideyuki Tokuda (Keio University)
Objective: Cross industry business/technical information exchange fostering new services
Planned Activity: Regular workshops to drive a focus on promoting interoperability testing by member companies

ALPINE,Brainsellers.com, CLOUD NINE, ECHELON Japan, Espial, IBM Japan, Information Technology One, JAPAN PROCESS DEVELOPMENT, KAI Software, KDDI R&D Laboratories, Kochikuya, MAZDA, Melco Power Systems, Mitsubishi Electric

NEC, NTT Advanced Technology, NTT, NTT East, NTT Software, Oki Electric Industry, SANYO Electric, SEIKO EPSON, SHARP, System Warehouse, TEPCO, TOSHIBA, Toshiba Solution, Tosteem Inax Holding, YKK Corporation
What’s new with IBM and OSGi?
IBM Strategic thrust
On Demand Operating Environment

OSGi is key to Access Services for Users and Business
Access - **Extend** the programming model

Establish a “Client Side” Middleware Industry to enable “Server Managed Clients”

**Access - Extend the programming model**

- **Interaction Fidelity**
  - “Reactive”
  - Request & Response
  - None

- **Connection Fidelity**
  - Always
  - Mostly
  - Sometimes
  - Never

- **Adaptation Fidelity**
  - Gizmos (e.g. PDA)
  - Laptop
  - Desktop

*Current Programming Model*

© copyright 2004 by OSGi Alliance. All rights reserved.
Programming Model

SOA is the composition model for the MVC patterns across topologies

Dynamically manageable
Platform
Applications
Configuration
“Disconnectable” operations
Desktops and appliances

Controller
Store

View
Mediator
Model

Intermediate Tiers

Controller
Store

View
Mediator
Model

End Point

Controller
Store

View
Mediator
Model

Client Platform

Preserve Roles
Development
Deployment
Richer experiences
Local Interactions
Alternate View/Control choices

© copyright 2004 by OSGi Alliance. All rights reserved.
Managed Client Middleware

The power of one architectural model, one programming model, one consistent tool set.

IBM WORKPLACE

Industrial Controller
Telematics Controller
Smart Phone/PDA
Smartcard
Banking ATM
Laptop

BUSINESS PROCESSES

Monitoring Workflows Application Adapters

INFORMATION

Search, Analysis Integration

Data & Content

Managed Client Services
Interaction & Access Services
Collaboration Services
Business Context & Activities

FRONT-END INTEGRATION
BACK-END INTEGRATION
IBM Workplace Client Technologies

An architecture to enable the eco-system
Managed Client Services

- Value
  Enable simultaneous execution of applications and services on “fit for purpose” class libraries on a single JVM instance.
  Independent life-cycle management of services and applications on a single JVM instance
- Runtime (SMF + ….)
  OSGi Minimum Execution Environment
  J2ME (CDC based)
  J2SE
- Standards
  J2ME, J2SE, OSGi R3
Platform Management (push/pull)

• Value
  – Enable “On Demand” management of Applications ranging from passive distribution to active, intelligent management

• Runtime
  – Components
    • Agent Bundle
    • DMS Server

• Standards
  – OSGi,
  – OMA (SyncML DM)
Access Services

- **Value**
  - Extend backend programming model “out” to devices
- **Runtime**
  - Web Container
    - JSP 1.2 and Servlet 2.3
  - Relational Data Access using JDBC
    - DB2 Everyplace or Cloudscape with Data Synch
  - Assured Messaging using JMS 1.1
    - WebSphere MQ Everyplace
  - Web Services
    - Web Services Client (JSR 172) and Provider
  - XML Parsing
    - MicroXML a small footprint non-validating parser
    - XML4J a full featured parser with validation
  - Synchronization Services
    - SyncML Framework (OMA SyncML)
  - SCADA (Supervisory Control and Data Acquisition)
    - WebSphere MQ Telemetry Transport

- **Standards**
  - J2EE, J2ME, W3C,
  - Web Services, OMA
Interaction Services

• Runtimes
  – Components
    • Browser (HTML, WML)
    • Multi-modal browser (XHTML + VoiceXML (X+V))
    • Graphics (AWT, SWT, IcdUI, others)
    • JSPs/Servlets, Portlets
  – Aggregators
    • Enterprise Offering
    • MIDP
    • Browser
    • Custom
    • Portal
• Standards
  – J2SE, J2ME, W3C, Eclipse

© copyright 2004 by OSGi Alliance. All rights reserved.
Some Solution Patterns
Local Web Application with replicated data
extend existing web based applications
Embedded Gateway Data Collection
the “classic” OSGi solution pattern – with messaging
Client using Local/Global Web Services

*begin movement to application level SOA*

```
HTTP Server
Framework
Java Class Library
Java Virtual Machine

WAS

Global Address
Book Web Service

JDBC, DB2
Data Model
Local Address
Book Web Service

Web Services
Server Proxy

Web Svs, Client Stub
Address Book Application
```

© copyright 2004 by OSGi Alliance. All rights reserved.
WCT ME – What is it?

• An integrated SDK containing runtimes and tools...

  – Runtimes
    • *Pre-configured device middleware service platforms*
    • *Components for (composition/extension) of (new/existing) platforms*

  – Tools
    • *Packaging tools - composition*
    • *Development tools - creation*
    • *Analysis tools - optimization*
    • *Porting kits - movement*

• ... enabling an end to end ecosystem.
WCT ME – How is it packaged?

WCTME 5.7.1 CD

Micro Environment Toolkit for WebSphere Studio
Technologies (Miscellaneous)
WECE Core

Core(s) Tooling
3rd Party Tooling
C Tooling
UEI
Analyze Link (JXE) Compile (AOT)
Build & Package

WEME Core

WSDD 5.7.1 (LPP)

Update Sites
Micro Environment Toolkit for WebSphere Studio
Technologies
WECE
WEME
WSDD
QNX
Forum Nokia

© copyright 2004 by OSGi Alliance. All rights reserved.
WCT ME – Who is using it?
The value chain that enables the eco-system

- Semiconductors & OS Providers
- Platform & System Integrators
- Device Manufacturers

“Into” - Pre-Design / Embedded
Pre-enable new devices

“Onto” - Aftermarket / Download
Post install Platform, Apps & Data

Enterprises
Solutions
Applications & Services

Solution Integrators, ISVs

Client Platform
(onto)

Service Providers
Solutions
Applications & Services

WCT ME

© copyright 2004 by OSGi Alliance. All rights reserved.
WCT ME – How do you get it?

1. Evaluation License
2. My Device Application and Middleware License
3. Development License + Runtime License (either LPP or OEM)

Distribution

Shipping Solutions


Built in support: 20+ Operating Systems and 5 Instruction Sets including Windows and Linux for desktop

© copyright 2004 by OSGi Alliance. All rights reserved.
WCT ME – How does it connect?

Client

WCT
- JMS (MQe)
- Web Services
- (JDBC) DB2e or Cloudscape
- OSGi Device Agent
- SyncML Libraries
- Managed Services
- WECM

Server

Application (MDBs)
- MQe Gateway
- MQ Server

Application (Web Services)
- DB2e Sync Server

WEDM (Tivoli DM)
- SyncML Libraries
- WebSphere App Server
- WECM

Synchronize objects

Operate over secure, optimized, roaming network connections

Send and receive secure transactions

Consume and publish Web Services

Synchronize relational data

Install, configure, maintain and add software
Some testimonies from the eco-system
WCT ME – How does Lotus use it?

Workplace Client for Lotus Workplace Messaging and Documents

Lotus Workplace Applications
- Workbench based aggregator and other UI Components: Rich text editors, embedded browser
- WSRP Consumer
- WSRP Producer
- SWT Java apps
- SWT UI Components

Lotus RCP (common platform services)

Core Extension Point Framework
- JFace
- SWT
- Generic UI frame

Interaction Services
- Others
- Cloudscape Container
- xAction Agent
- OSGi Device
- SyncML Framework

Access/Platform Services

Managed Client Services
- Framework
- Java Class Library
- Java Virtual Machine

© copyright 2004 by OSGi Alliance. All rights reserved.
Gateway type deployments

Realtime pipeline flow

Remote offshore drilling

Remote UI/data collection

Pay as you drive insurance

© copyright 2004 by OSGi Alliance. All rights reserved.
Example ISV: Blue Martini

Blue Martini CRM applications

- Leverage the IBM Workplace Client Technology Micro Edition
- Deploy on intermittently connected mobile devices
- Enables mobile, guided selling interactions for increased productivity
  - Configurations
  - Quotes
  - Proposals
  - etc.
- Provide same UI and function as the web-connected applications

IBM Workplace Client Technology Micro Edition
- Database Mgmt.
- Transaction Mgmt.
- Platform Mgmt.
- Enterprise Access
- Embedded JVM

IBM WebSphere Application Server
- Database Mgmt.
- Transaction Mgmt.
- Platform Mgmt.
- Enterprise Middleware

Blue Martini Components

Interactive Selling Mobile

Interactive Selling Web

© copyright 2004 by OSGi Alliance. All rights reserved.
Example Enterprise: NISSAY

- **Who:** Nissay (Nippon Life Insurance Company) is Japan’s #1 insurance company

- **Why:** Extend their Insurance Adjustor application for their mobile field force (70,000+ employees) increasing productivity and customer satisfaction

- **What:** Allow Nissay’s mobile adjustors to input policy data when offline and seamlessly offer quotes or search for policies when connected

- **How:** IBM Workplace Client Technology Micro Edition is used to enable a “sometimes connected” environment (runtime and tools)

- **When:** Complete and in rollout.
Example: IBM Retail Store Integration Framework (SIF)

IBM Store Integration Framework supports both wired and wireless devices.
Design: The Personal Shopping Assistant hardware

• Key design requirements
  – Consumer-friendly
• Pick up & put away
• Easy to see
• Encourage self-scanning
• Fun for the whole family
  – Mobile
• Weight vs. battery life balance
• Battery life shouldn’t be a shopper concern
  – Retail environment hardened
• Hardened
• Spill proof
  – Location sensitive
• Precision: 4-6 feet
• Easy to install and maintain
Banking Opportunity
Architecturally, it looks like the retail store topology
OSGi WC 2004 IBM demonstrations

- Mobile Insurance Adjuster Application
  - One application across the Nokia 9500, PocketPC, and Windows Laptop
- Developer tooling
  - OSGi, Web Services, Eclipse, Extension Services
- RFID Solutions
  - TCG Trusted Platform Module
  - Integrated with WCT ME (OSGi) via partner Arcom hardware
- Telematics Solutions
  - Telematics Reference Implementation
- Cross-device stateful relocation of applications
  - Research enabling greater degrees of mobility and multi-device interaction
- Device Management (OMA DM and Mobile Expert Group direction)
  - WebSphere Everyplace Device Manager managing a Nokia 9500
- xCP
  - AACS rights management via OSGi bundles (includes trusted framework)
Looking forward…
What makes a complete device specification?

- Platform
- Connectable
- Standalone

On Device Programming Interfaces

Data Support
Protocol Support

User Experience

Connections

Physical Device

Infrastructure
## Platform specifications

<table>
<thead>
<tr>
<th>N</th>
<th>Data</th>
<th>Protocol</th>
<th>(S)(A) PI</th>
<th>Device</th>
<th>Connection</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Specifications**  **Constraints**
Client side middleware industry standards

- **Technical completeness** requires specification of the following:
  - Data Formats
  - Communication Protocols that deliver and consume those Data Formats
  - Programming Interfaces
  - declarative "Service" interfaces
  - imperative bindings

- **Interoperability completeness** requires the following:
  - **Control** (fair and transparent multi-lateral governance)
  - **Completeness** (see above)
  - **Compliance** (uniform adherence to the specs that maximizes interoperation)
  - **Cost** (equitable licensing of essential IPR; brand value flows equally to all "members")

SOA requires “Client” side middleware industry standards
An example – one view of the mobile constellation

- **SAX (API)**: `org.sax.*`
- **W3C (API, Data, Protocol)**: SOAP, HTML, DOM (`org.w3c.dom.java.*`), HTTP, XML, VoiceXML, X+V
- **JCP (static API)**: `java.*`; `javax.*`
- **W3C (API, Data, Protocol)**: SOAP, HTML, DOM (`org.w3c.dom.java.*`), HTTP, XML, VoiceXML, X+V
- **OSGi Alliance (Data, Protocol, SPI, dynamic API, service interface)**: `org.osgi.*`
- **Mobile Expert Group**
  - Execution Environment
  - Application Model
  - Service Programming Interfaces
  - UI Service
  - MIDP Container

**TCG (Data, Protocol)**
- **OMA (Data, Protocol)**
  - DRM
  - Data Sync
  - Device Mgmt
  - Location

**Eclipse.org (API, Data, Protocol)**
- SWT
- jFace
- RCP

**IETF (Data, Protocol)**
- TCP
- IP

**Netscape**
- SSL

**TCG Infrastructure WG**

**TCG Mobile WG**

© copyright 2004 by OSGi Alliance. All rights reserved.
Platform Specifications – What is the role of OSGi going forward?

• How do we manage the relationship between the constituent bodies of a platform?
• Does dynamic platform composition affect licensing models for component specifications?
• Open Source versus Open Standards?
• How “Open” does the standard need to be?
• Should OSGi specify other bindings for declarative services?
• Should we begin new verticals?

Lots done… Lots to do!
Questions?