Sun Java System Architecture: Java Enterprise System Technical Introduction
Dallas JUG: J2EE SIG 7/21/2004

Tom Barrett
Software Systems Engineer
Sun Microsystems - Dallas

thomas.barrett@sun.com
Session Objectives

- Introduce Sun's current thinking on application software infrastructure
- Discuss Java Enterprise System from an architectural perspective
- Stimulate discussion and feedback
- Incite you to learn more about Sun's software centerpiece: Java Enterprise System
Agenda
Sun Java System Architecture

• What is it?
• Why is it important?
• How are the key services fulfilled?
But First ... What's Java Enterprise System? Product Family
But First ... What's Java Enterprise System? Product Family

Full Java System Product Family
But First ...
What's Java Enterprise System? Product Family

Services Provided:
- Identity/Security
- Web and Application
- Portal
- Communication/Collaboration
- Availability
But First ... What's Java Enterprise System? Marketing Themes

- “Radical New Approach to Enterprise Software” - software/services for subscription on per employee or per citizen basis
- Simple – All products integrated with common installer and docs
- Predictable – Updates on a predictable cadence like Solaris
- Affordable – Dramatically lower acquisition cost
- Software Stack – network identity, web/application, portal, communication/collaboration, availability

Solaris 8 & 9 (SPARC and x86) and Red Hat Enterprise Linux AS 2.1
But First ...  
What's Java Enterprise System? Predictable Cadence

Security Services (end-to-end)

| Directory Srvr Enterprise Edition (includes dir-server, dir-proxy and id-sync) |
| Access Manager (formerly Identity) |
| Application Server PE |
| Application Server SE |
| Message Queue EE |
| Web Server |
| Messaging Server |
| Calendar Server |
| Instant Messaging Server |
| Portal Server |
| Portal Secure Remote Access |
| Sun Cluster |
| Portal Server Mobile Access |
| Application Server EE |
| Active Server Pages |
| Integration Server |
| Web Server Proxy |
| Identity Manager (includes Waveset Lighthouse and Meta-Dir) |

Roadmap Subject To Change
But First ...
What's Java Enterprise System? Java Enabler
"[JES] shows that there are finally some smart people at Sun thinking about doing things right"

"Most portals are really built of a dozen or so applications like the portal application framework, Web server, application server, calendar, e-mail, instant messaging, LDAP, single sign on and many others"

"[JES is a] clean integration of all of these tools that runs smoothly and is tested as a unit to discover integration problems"

"[JES is] not really innovative ... [but] a lot of the core business software is up and ready to use in a few hours"

"They [Sun] could be a market leader in their own J2EE market ... Stay tuned"
Agenda
Sun Java System Architecture

• What is it?
  – Standards-Based Building Blocks Approach
  – Taxonomy for Java Enterprise System Components
  – Collection of Service-Providing Capabilities

• Why is it important?
• How are the key services fulfilled?
What is it?
Building Blocks

- Java System Architecture is Sun's standards-based software architecture supported by Java Enterprise System
- Encompasses building and deploying content and services
What is it?
Open Standards-Based

Service/Content Creation, Assembly, and Deployment
UML, BPSS, WSDL, NetBeans

Service Integration
UDDI, ebXML, EDI, JMS, Java Connectors, SQL, JDBC, CORBA, JavaMail, FTP

Content Services
ESMTP, IMAP, POP, S/MIME, SMS, MMS, iCal, SIP, SIMPLE, WebDAV

Service Containers
JBI, WSCI, BPEL, Choreography

Service Delivery
RDF, RSS, WML, cHTML, J2ME, MIDP, Java Card, VoiceXML, OMA, JSR-124, JSR-168, JSR-172

ID/Sec/Mgmt: Liberty v1.1, LDAP, SP-DNA, DSML, UDDI, SASL, SAML, X.509, PKCS, PKIX, OCSP, CIM, CIM-SOAP, WBEM, Kerberos, IKE, JAAS, JCA/JCE, J2SE Policy/Perms, P3P, XML DSIG, XML Encrypt, Java Card, GSC-IS, JSR-177, XKMS, XACML, WS-Security

Throughout:
HTML, XHTML, HTTP(S), SSL/TLS, Java, J2SE, J2EE 1.4 (EJB, JSP, Servlets, JNDI, JMS, ...)
JAXM, JAXR, JAX-RPC, JAXB, JAXP, JMX, SOAP, WSDL, XML, XSLT, XML Schema, SAX, DOM, WS-I Basic Profile, WS-Reliability

underlined == emerging/future standard
What is it?
Our Collection of Software Service Enablers

Service/Content Creation, Assembly, Deployment
Sun Java Studio

Service Integration
- Sun Java System App Server, Integration Servers, Message Queue, Directory Server

Content Services
- Sun Java System Web Server, Messaging, Calendar, IM Servers

Service Containers
- Sun Java System Integration Servers
- Sun Java System Web, App Servers

Service Delivery
- Sun Java System Portal Server, Content Delivery Server

Identity, Security, and Management Services
- Sun Java System Directory Server, Identity Server, Management Framework, N1

Platform
- Solaris SPARC/IA32, Linux, Sun Cluster, N1
What is it?
Open Systems Platform

Integrated Stack

- Common installer
- System Testing
- Reference Architectures
- Standardized Documentation

Integratable Stack

- Partner products
- Standard and open APIs
Agenda
Sun Java System Architecture

• What is it?
• Why is it important?
  – Architecture for Future Systems
  – Evolution of Software Abstraction
• How are services supported?
Why is it important to you?
Past Architectures

- Server-based applications scales to perhaps 100s of users
- Desktop applications mean one copy of software mapped to each personal computer
Why is it important?
Modern Architecture for Future Systems

- Applications composed of loosely-coupled components
- Components deployed across the network
- Services must scale to 10,000s or millions of users
Why is it important?  
Modern Architecture for Future Systems

The Java™ Platform

Networked Components

Two fundamental platform choices

XML

The Windows Platform

The MVS™ Platform

App

Processor

O/S

OS400

AS400

MVS

370

Solaris™

SPARC®

Linux

x86

AIX

Power

Windows Server

x86

HP-UX

Precision
## Why is it important? Modern Architecture for Future Systems

A Look Ahead: Applications as Service Graphs

| Client/Server | Distributed Objects | Web Application | Web Service | Next | Next
|---------------|---------------------|-----------------|-------------|------|------
| ![Image](client_server.png) | ![Image](distributed_objects.png) | ![Image](web_application.png) | ![Image](web_service.png) | ![Image](next.png) | ![Image](next.png) |

Most of us are here
Why is it important? Modern Architecture for Future Systems

<table>
<thead>
<tr>
<th>Catch Phrase</th>
<th>The Network is the computer</th>
<th>Objects</th>
<th>Legacy to the Web</th>
<th>The Computer is the Network</th>
<th>Network of embedded things</th>
<th>Network of things</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scale</td>
<td>100s</td>
<td>1000s</td>
<td>10000000s</td>
<td>100000000s</td>
<td>1000000000s</td>
<td>1000000000s</td>
</tr>
<tr>
<td>Leaf Protocol(s)</td>
<td>X</td>
<td>X</td>
<td>+HTTP (+jVM)</td>
<td>+XML, Portal</td>
<td>+RMI</td>
<td>Unknown</td>
</tr>
<tr>
<td>Directory(s)</td>
<td>NIS, NIS+</td>
<td>+CDS</td>
<td>+LDAP(*)</td>
<td>+UDDI</td>
<td>+jini</td>
<td>+?</td>
</tr>
<tr>
<td>Session</td>
<td>RPC, XDR</td>
<td>+CORBA</td>
<td>+CORBA, RMI</td>
<td>+SOAP, XML</td>
<td>+RMI/jini</td>
<td>+?</td>
</tr>
</tbody>
</table>

Schematic

Client/Server Early Distributed Object Computing J2EE Web Services Current Java System Architecture Focus Technologies like Jini and JXTA
Why is it important?
Evolution of Software Abstraction: Traditional Thinking
Why is it important?
Evolution of Software Abstraction: Traditional Thinking

The Base Platform (J2SE) Means that Byte Codes are the New Binary

Base Java Platform

Legacy Binaries

Legacy Binaries

Old Developer Contract

Solaris

Linux

SPARC

x86
Why is it important? Evolution of Software Abstraction: New Realization

Layered Components Provide an Abstract Services Layer

Heightened abstraction level necessary to leverage potential of network computing

New Developer Contract

Legacy Binaries

Old Developer Contract

Why is it important?

Evolution of Software Abstraction: New Realization

Layered Components Provide an Abstract Services Layer

Heightened abstraction level necessary to leverage potential of network computing

New Developer Contract

Legacy Binaries

Old Developer Contract

Why is it important?

Evolution of Software Abstraction: New Realization

Layered Components Provide an Abstract Services Layer

Heightened abstraction level necessary to leverage potential of network computing

New Developer Contract

Legacy Binaries

Old Developer Contract
Agenda
Sun Java System Architecture

• What is it?
• Why is it important?
• How are the key services fulfilled?
  – Identity Management
  – Service Containers (web and application servers)
  – Service Integration (connectors, messaging, web services)
  – Content Services (email, instant messaging, calendaring)
  – Service Delivery (portals)
How are services fulfilled?

Identity Management

- **Service/Content Creation, Assembly, Deployment**
  - Sun Java Studio

- **Service Integration**
  - Sun Java System App Server, Integration Servers, Message Queue, Directory Server

- **Content Services**
  - Sun Java System Web Server, Messaging, Calendar, IM Servers

- **Service Containers**
  - Sun Java System Integration Servers

- **Service Delivery**
  - Sun Java System Portal Server, Content Delivery Server

- **Identity, Security, and Management Services**
  - Sun Java System Directory Server, Identity Server, Management Framework, N1

- **Platform**
  - Solaris SPARC/IA32, Linux, Sun Cluster, N1
How are services fulfilled?
Identity Management Challenge

- Web applications are proliferating
- Many applications developed as silos with their own security, provisioning and user management mechanisms resulting in:
  - Higher admin costs
  - Poor flexibility
  - Security risks
- Solution is a shared identity management to control the lifecycle of an identity and its relationship to business applications and services
How are services fulfilled?
Identity Service Role
How are services fulfilled?

Identity Key Features

- Single Sign-On
- Rule-Based Policy Engine
- Delegated Administration
- URL and J2EE policy agents
- SAML & Liberty Enablement
- JAAS authentication framework basis
- Java & XML interfaces

SAML (Security Assertion Markup Language) defines mechanisms to exchange authentication, authorization and nonrepudiation information, allowing single signon capabilities for Web services. Liberty is based upon SAML.
How are services fulfilled?

Identity Architecture

- **Portal Server**
- **Central Directory**
- **Certificate Services** & Management
- **Application Security**
- **Profiles / Attributes**
- **Identity Manager** (Formerly Waveset LightHouse)
- **Workflow Management**
- **PBX**
- **Messaging Server**
- **HR Database**
- **MS Active Directory**
- **Identity Provisioning & Synchronization**
- **Employee**
- **Supplier**
- **Customer**
- **Partner**
- **Access Manager** (Formerly Identity Server)
- **SSO**
- **Authorization**
- **Audit**
- **Federation**
- **Authentication**
- **Self-service**
- **Policies**
- **Admin**
- **Identity Management**
How are services fulfilled?
Identity Service Deployment

- Identity Service Deployment
- Directory Multi-Master Replication
- Identity Server Replication
- Policy Agents (URL and J2EE)

Directory Multi-Master Replication

Identity Server Replication
How are services fulfilled?
Service Containers

Service/Content Creation, Assembly, Deployment
Sun Java Studio

Service Integration
Sun Java System
App Server, Integration
Servers, Message Queue, Directory Server

Content Services
Sun Java System
Web Server, Messaging, Calendar, IM Servers

Service Containers
Sun Java System
Web Server
Sun Java System
Application Server

Service Delivery
Sun Java System
Portal Server, Content Delivery Server

Identity, Security, and Management Services
Sun Java System Directory Server,
Identity Server, Management Framework, N1

Platform
Solaris SPARC/IA32, Linux, Sun Cluster, N1
### How are services fulfilled?

**J2EE Container Services**

<table>
<thead>
<tr>
<th>Web Container</th>
<th>EJB Container</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Web Server</strong></td>
<td><strong>ORB</strong></td>
</tr>
<tr>
<td>JAXRPC/Web Services</td>
<td></td>
</tr>
<tr>
<td>JAXR</td>
<td>Admin/Management</td>
</tr>
<tr>
<td>Naming</td>
<td>Monitoring</td>
</tr>
<tr>
<td>Persistence/JDBC</td>
<td>Deployment</td>
</tr>
<tr>
<td>Transaction</td>
<td></td>
</tr>
<tr>
<td>Connector</td>
<td></td>
</tr>
<tr>
<td>Messaging</td>
<td></td>
</tr>
<tr>
<td>Java Mail</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td></td>
</tr>
<tr>
<td>Admin/Management</td>
<td></td>
</tr>
<tr>
<td>Monitoring</td>
<td></td>
</tr>
<tr>
<td>Deployment</td>
<td></td>
</tr>
</tbody>
</table>

- **HTTP(s)**
- **IIOP/SSL**

**GUI/CLI Tools**

- XML Registry
- Database
- EIS
- JMS Provider
- SMTP Server
- JACC Provider

- Web Service Endpoint

---

**Notes:**
- JAXR
- JAXRPC
- IIOP/SSL
- EIS
- GUI/CLI
How are services fulfilled?

J2EE Container Services: Application Server 7 Family

Enterprise Edition (EE) is not a first class citizen in Java Enterprise System yet, but a license to use is included.
How are services fulfilled?
J2EE Container Services: Application Server 7 Family

High performance interoperable Web Services

“Modular Innovation”
Web & EJB Containers

High performance, guaranteed messaging

Tool & Admin Extensions

Massively Upwardly Scalable Connection Handling

* Secure / Remote Administration
* Cluster Management

* Session State Availability
* RMI / HTTP Load Balancing
* Instance Failover / Recovery
How are services fulfilled?
Sophisticated Fault Tolerance Support with EE

Tier 1: Content

Tier 2: Business Logic

"Always On" Session State Repository
How are services fulfilled?
Sophisticated Fault Tolerance Support with EE

Session state is stored in a highly available data manager that is distributed across all servers.

If the sessions in any container fails, any other container can retrieve the session state so the application can continue from where it left off.
How are services fulfilled?
J2EE Container Services: Application Server 8 PE
How are services fulfilled?
J2EE Container Services: Application Server 8 SE & EE

• Improved Manageability & Usability
  • Cluster support in admin GUI & admin CLI
  • Remote JMX support
  • Improved Diagnostics / Logging
  • Solaris 10 Zones support – resource management
  • Monitoring enhancements in J2SE 1.5

• All-round Enhanced Performance
  • Continued WS performance lead
  • Major CMP / EJB performance improvements

• Target Availability: Q4 2004
How are services fulfilled?
Service Integration

Service/Content Creation, Assembly, Deployment
Sun Java Studio

Service Integration
Sun Java System
App Server, Integration
Servers, Message Queue,
Directory Server

Content Services
Sun Java System
Web Server, Messaging,
Calendar, IM Servers

Service Containers
Sun Java System
Web Server
Sun Java System
Application
Server

Service Delivery
Sun Java System
Portal Server, Content
Delivery Server

Identity, Security, and Management Services
Sun Java System Directory Server,
Identity Server, Management Framework, N1

Platform
Solaris SPARC/IA32, Linux, Sun Cluster, N1
How are services fulfilled?
Service Integration: Current Capabilities

- J2EE Connectors
- Asynchronous Messaging
- Web Services
How are services fulfilled?
Service Integration: J2EE Connectors

- Java Connector Architecture (JCA) does for Enterprise Information System (EIS) what JDBC does for databases
- Surfaces often arcane EIS capabilities in a standards-compliant way
- Provides tight coupling, high performance, high reliability and fine-grained access to “foreign” capabilities
- Links your J2EE container's connection pooling, transaction and security services to those of the target EIS
- Defines an Service Provider Interface (SPI) for plugging adapters into containers
- Documents a Common Client Interface (CCI) for application to interact with the adapter
- Third-party connector provider industry has emerged
How are services fulfilled?
Service Integration: Asynchronous Messaging

• Peer-to-peer exchange of messages
• Doesn't depend on all elements in distributed system being available at time of the transaction
• Java community embraced asynchronous messaging via Java Message Service (JMS)

Roles
– Provider: broker hosting the message service
– Client: producers and consumers of messages
– Domains: interaction styles include point-to-point (PTP) and publish-subscribe (Pub/Sub)
– Message: five types including text, map, bytes, stream, object and message (empty)
How are services fulfilled?

Service Integration: Messaging Architecture

- Utilizes JMS API to send and receive messages to/from destinations
- Client thread blocks sending persistent messages
- Client doesn't block otherwise

- ConnectionFactories – create connections to message service
- Destinations – represent physical locations (queues and topics) where messages are sent

- Queue (Point-to-Point):
  - Each message has only one consumer
  - Client can fetch message even if it it wasn't active when message was sent

- Topic (Pub/Sub):
  - Each message may have multiple consumers
  - Client must have been subscribed to topic when message was published
  - Client must be active in order receive messages (unless subscription is durable)
How are services fulfilled?
Service Integration: Web Services

1. Cross Platform JVM
   - Portable Logic
     - EJB
     - Servlet
   - Solaris
   - SPARC / Intel
   - Multi-Platform Deployable

2. Relational DB
   - Java Ent. System
   - Java
   - SQL Server
   - OS 390
   - Legacy

3. Service Registry/Repository
   - Browser/Portal

4. Business Process
   - SOAP/WSDL Interface
   - I2EE Connector
   - Single Platform

5. Windows
   - Single Platform
   - .NET
   - Embedded Logic (VB, C#)
   - Windows
   - Intel
   - Mainframe
How are services fulfilled?

Service Integration: Future Web Services

- Web services are accessible programmatically
- XML offers a lingua franca to support data interchange for heterogeneous environments
How are services fulfilled?

Service Integration: Future Web Services

Phases of Adoption

1. Statically-bound RPC-style interaction
2. Dynamically-bound on corporate intranets, limited, prearranged B2B on the Internet
3. Dynamically-bound, spontaneous interaction across the Internet
How are services fulfilled?
Service Integration Futures

Pluggable Artifacts & Engines
(not defined by JSR208)

WS-BPEL Engine
Rules Engine
Transform Engine
Routing Engine
“other” Engine

WS-I Binding
Email Binding
EIS Resource
EDI VAN Gateway
JMQ Gateway
“other” Binding

Deployment / Packaging SPI *

Machine SPI *

JBI Container

JBI Message Bus *

JBI Container

Binding SPI *

SOAP
SMTP
JCA
AS2
JMS
Proprietary

* Defined by JSR 208
How are services fulfilled?

Content Services

Service/Content Creation, Assembly, Deployment

Sun Java Studio

Service Integration

Sun Java System App Server, Integration Servers, Message Queue, Directory Server

Content Services

Sun Java System Web Server, Messaging, Calendar, IM Servers

Service Containers

Sun Java System Web Server

Sun Java System Application Server

Service Delivery

Sun Java System Portal Server, Content Delivery Server

Identity, Security, and Management Services

Sun Java System Directory Server, Identity Server, Management Framework, N1

Platform

Solaris SPARC/IA32, Linux, Sun Cluster, N1
How are services fulfilled?

Content Interfaces

- Web pages
- Portal channels
- Address books
- Calendars
- Email
- Instant messages
How are services fulfilled?
Content Services: Communication & Collaboration
How are services fulfilled?

Service Delivery

Service/Content Creation, Assembly, Deployment

Sun Java Studio

Service Integration

Sun Java System
App Server, Integration Servers, Message Queue, Directory Server

Content Services

Sun Java System
Web Server, Messaging, Calendar, IM Servers

Service Containers

Sun Java System
Web Server

Sun Java System
Application Server

Identity, Security, and Management Services

Sun Java System Directory Server,
Identity Server, Management Framework, N1

Platform

Solaris SPARC/IA32, Linux, Sun Cluster, N1

Service Delivery

Sun Java System
Portal Server, Content Delivery Server
How are services fulfilled?
Service Delivery: Portal Defined

- Goal is to render web application data so that it can be displayed with little or no additional processing on the client device
- Allows multiple sources of information to be displayed within a single page or set of pages
- Terminology:
  - Desktop: page holding the content
  - Channel: individual content sources presented as independent views and aggregated onto the desktop
- Comprehensive portal solution provides:
  - Location
  - Connection
  - Aggregation
  - Presentation
  - Communication
  - Personalization
  - Notification
  - Delivery
How are services fulfilled?
Service Delivery: Big Picture

Data No Matter Where It Resides
Aggregated and Personalized
Securely Delivered to Targeted Communities Via Any Device

- Enterprise, Legacy & Business Intelligence
- 3rd Party Data and Information Feeds
- Communication & Collaboration
- Web Pages & Links
- Process Automation Services

- Name/Password
- Role (Manager/Employee)
- Personal Assets
- Service Access
- Security

- Federation
- Identity
- Policy
- Provision
- Registry

- Identity Attributes
- Functions

- Authentication Mechanism
- Secure Boundary

- Employee
- Supplier
- Partner
- Customer
How are services fulfilled?
Service Delivery: Building Blocks

Secure Access to:
- Intranet File Servers, Legacy Apps
- Internal Web Apps
- User Desktops

Java System Portal Server
- Identity & Policy Management
- Personalization
- Knowledge Management & Search
- Content Management

Java System Identity Server

Java System Directory Server

Java System Web or Application Server

Virtually Any Device Access
- IBM, MS and Sun Groupware
- Connectivity
- VoiceXML
- WAP 2.0
Agenda Summary
Sun Java System Architecture

• What is it? (abstraction, product framework, open standards, integrated and integratable)
• Why is it important? (need to embrace componentry for immense, network-based applications)
• How are the key services fulfilled? (identity, containers, integration, content, delivery)
Thank you!

Sun Java System Architecture: Java Enterprise System Technical Introduction
Dallas JUG: J2EE SIG 7/21/2004

Tom Barrett
Software Systems Engineer
Sun Microsystems - Dallas

thomas.barrett@sun.com