

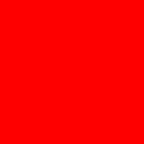
**ORACLE®**



**ORACLE®**

## **Java EE 7**

Linda DeMichiel  
Java EE Platform Lead



The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

# Agenda

- Java EE 7 theme
- Main features
- Component JSRs
- Transparency
- Status and Schedule

# Java EE 7 Theme: The Cloud

“We’re making Java EE ready for the cloud”

# Platform as a Service (PaaS)

*The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure..., but has control over the deployed applications and possibly application hosting environment configurations. (NIST)*

# Support for Platform-as-a-Service Model

- Next logical step in extending value proposition of Java EE platform
  - Provide a way for customers and users of Java EE to leverage private and public clouds
- PaaS support entails evolutionary change
  - New platform roles; some modifications to existing roles
  - Small programming model changes
  - Multi-tenancy
- Java EE 7 is the first step
  - An evolutionary step, not a radical one
  - Includes limited support for SaaS

# Java EE 7 Cloud Roadmap

- Define new roles to accommodate PaaS model
- Enable multitenancy
  - Containers and resource managers need to cooperate
- Add metadata
  - For service dependencies
  - For sharing
  - For configurability
- Applications declare themselves cloud-enabled
  - Additional restrictions possible
- Add useful APIs for cloud environment
  - JCache, JAX-RS client



# PaaS and Multi-tenancy: Some Models

## (1) PaaS Platform on Demand

- New runtime stack for each tenant

## (2) PaaS Multitenant Containers

- Isolated application partitions per tenant with shared runtime

## (3) SaaS Multitenant Applications

- Shared application instances, with tenant-specific customization

## (3\*) SaaS-limited

- Separate application instances, with tenant-specific customizations

# Multi-tenancy in Java EE 7

- Java EE 7 targets PaaS models 1, 2, 3\*
- Mantra: “one application instance per tenant”
  - Support for multiple isolated instances of the same application for different tenants
  - Each copy is deployed and customized for a single tenant
  - “SaaS-limited”
  - Allow shared resources
- “SaaS-proper” deferred to Java EE 8

# Tenants

- Tenants correspond to the unit of isolation
- Mapping to a tenant done by the container
  - E.g., using virtual servers
- Apps may be marked as Multitenancy-Enabled
  - Possible restrictions on application code
- Tenant identifier available to application
  - E.g., under `java:comp/tenantId`

# Multi-tenancy and Resources

- Separate database, shared nothing
  - Best isolation
  - Resource intensive
- Shared database, separate schema
  - Database provides isolation at schema-level
  - Less resource intensive
- Shared database, shared schema
  - Isolation happens at row level !
  - Least resource intensive

# Multi-tenancy and JPA

- JPA provider intermediates
  - Shared database, separate schema
    - Metadata allows reconfiguration of schema / table mapping
  - Shared database, shared schema
    - Provider can use tenant-id to provide isolation at row level
  - Consequences for programming model
    - Restrictions on native queries, direct JDBC access, etc.

# New and Updated Roles

## Cloud Vendor

- Java EE Product Provider
- PaaS Provider
- PaaS Account Manager
- System Administrator

## Cloud Customer

- PaaS Customer (Tenant)
- Application Submitter
- Deployer
- Application Administrator
- End-User

# Use Cases / Scenarios

- Java EE vendor becomes a PaaS product vendor
- A PaaS provider wants to support Java EE applications
- A PaaS customer writes a web application, deploys it to a cloud infrastructure, end users access it
- A software company writes an application, submits it to a PaaS provider, then any number of tenants sign up for the application, deploy it, their end users access it

# Other Topics on the Java EE 7 Agenda

- Alignment of Managed Beans across CDI, EJB, JSF, Servlet, ...
- Further simplifications for ease-of-development
- Pruning
  - JAX-RPC; EJB 2.X BMP, CMP; Deployment API (JSR 88)
- Update to Web Profile
- Improved configuration
- Add service metadata
  - – e.g., data source sharing, QoS
- Add useful APIs for cloud environment
  - JCache, JAX-RS client
- ...



# Java EE 7 JSRs

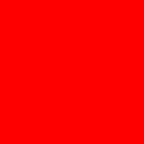
- Platform 7 / Web Profile 7
- JPA 2.1
- JAX-RS 2.0
- EJB 3.2
- JMS 2.0
- Servlet 3.1
- EL 3.0
- JSF 2.2
- CDI 1.1
- JCache 1.0 (JSR 107)
- Concurrency Utilities 1.0
- Bean Validation 1.1
- JSON 1.0
- Batch Processing 1.0
- State Management 1.0
- Common Annotations 1.2 MR
- JAX-WS 2.3 MR
- JTA 1.2 MR
- JSP 2.3 MR
- JASPIC 1.2 MR

# Transparency

- High level of transparency for all Java EE JSRs
- Using java.net projects to run our JSRs in the open
  - One java.net project per specification
  - E.g., javaee-spec, jpa-spec, jax-rs-spec, ejb-spec,...
- Publicly viewable Expert Group mailing list archive
- Users observer list gets copies of all emails to the EG
- Download area
- Wiki, source repository, JIRA, etc. at the group's discretion
- JCP.org private mailing list for administrative / confidential info
- Plan to move to JCP 2.8

# Status and Schedule

- Nearly all JSRs up and running
- Remaining ones to be filed in the coming weeks
- Final release target: Q3 2012
- Date-driven release: anything not ready will be deferred to Java EE 8



The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

**ORACLE®**