Java EE 5™ Introduction

Goals

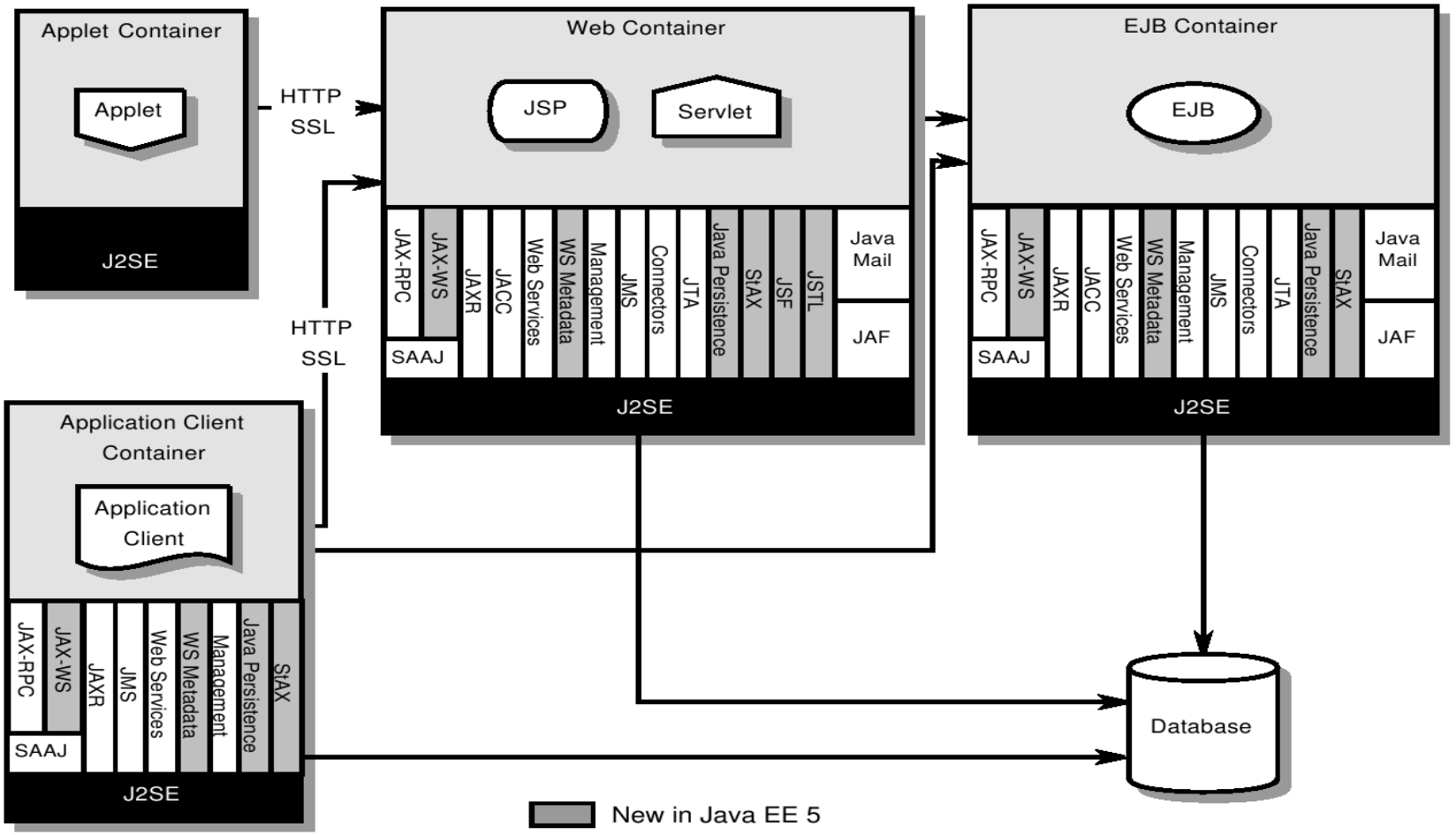
- Become familiar with the Java EE Architecture
  - Understand the purpose of Java SE, Java ME, and Java EE
  - Understand the purpose of each Java EE Container
  - Understand the purpose of each Java EE API
- Understand the difference between an API and Provider
- Understand there are significant variations in Java EE architectures

Java Technology Levels

- Java Platform, Standard Edition
  - Java SE (J2SE)
  - core language
- Java Platform, Micro Edition
  - Java ME (J2ME)
  - targeted at small devices
    - PDAs
    - cell phones
- Java Platform, Enterprise Edition
  - Java EE (J2EE)
  - targeted at enterprise deployments
    - persistence
    - distributed systems
    - web-based applications
    - transactions
    - security

Note: Java Technology went through a name change a while back.
- J2SE 5.0 stayed J2SE 5.0
- J2SE 6.0 is called Java SE 6
- J2EE 5.0 became Java EE 5
- J2ME became Java ME

Java EE Architecture



Java EE Architecture

- Does not imply physical partitioning into machines, processes, etc.
- No “all or nothing” requirement
  - use what you need

Containers

- Provide Java EE runtime environments with J2SE v5.0
- Provide required services to application components
  - example services: JMS, JTA
  - application components: Applets, Java applications, Servlets/JSPs, and EJBs
- Understand application component deployment formats
- Interposes between application components to transparently inject services required
  - transactions, security checks, resource pooling
- Synonymous with application server from the application developer point of view

Application Components

- Application Client
  - Java programs that execute outside of the EJB, Web, and Applet Containers
  - Typically GUI-based
- Applets
  - GUI components
  - typically execute in Web browser
- Web Applications
  - accessed through HTTP
  - primarily generate HTML and XML output
  - Servlets, JSPs, JSF, etc.
- Enterprise Java Beans (EJB)
  - execute in managed environment that include transactions
  - typically provide business logic for Java EE application

Resource Adapters

- System-level software component
- Extends functionality of Container
  - Provides connectivity with external resource managers or native language components
  - Implement existing (e.g., JDBC, JMS) or alternate APIs
- Plugin into Java EE environment to provide system-level support for
  - Resource Pooling
  - Transactions
  - Security

Standard Services

- Communications Services
  - HTTP
    - J2SE java.net package defines client-side API
    - Java EE defines server-side API defined within servlet, JSP, JSF, and Web service interfaces
  - HTTPS
    - use of HTTP over SSL for transport level security
  - Java Message Service (JMS)
    - API for messaging
    - support for publish-subscribe and point-to-point

CORBA Support Services

- RMI-IIOP
  - APIs (javax.rmi) to allow RMI-style programming independent of underlying protocol
    - J2SE native RMI Protocol (JRMP)
    - CORBA IIOP protocol
  - Permit Java EE application components to access CORBA services compatible with RMI programming restrictions
  - Permit CORBA clients to access EJB components
  - Using dependency injection versus JNDI lookup eliminates the need for applications to use most of API
- Java IDL
  - Permit Java EE application components to invoke CORBA services, independent of the RMI API

Coordination Services

- Java Naming and Directory Interface (JNDI)
  - API for naming and directory access
- Java Transaction API (JTA)
  - application-level API to demarcate transaction boundaries
  - provider-level API between transaction and resource managers

Persistence Services

- Java Database Connectivity (JDBC) API
  - API for connectivity with relational databases
- Java Persistence API
  - API for managing persistence and object/relational mapping
  - Required in Java EE
  - Can be used in J2SE environments (is part of Java SE 6)

XML and Web Service Services

- Java API for XML Processing (JAXP)
  - integrated API support for separate
    - SAX, DOM, and StAX (Streaming API for XML) XML parsing APIs
    - XSLT transform engines
- Java API for XML-based RPC (JAX-RPC)
  - legacy API for Web services
- Java API for XML Web Services (JAX-WS)
  - follow-on to JAX-RPC
  - primary API for Web services
  - includes SOAP and RESTful Web service bindings
  - uses Java Architecture for XML Binding (JAXB) to define Java to XML bindings
  - uses SOAP w/ Attachments API for Java (SAAJ) to manipulate low level SOAP msgs
- Web Services Metadata
  - defines Java language annotations for developing Web services
- Java API for XML Registries (JAXR)
  - API for client access to XML registry servers

Security Services

- Java Authentication and Authorization Service (JAAS)
  - enables services to authenticate and enforce access controls
  - Java implementation of standard Plugable Authentication Module (PAM)
- Java Authorization Service Provider Contract for Containers (JACC)
  - defines contract between Java EE application server and authorization service provider

System Services

- Java EE Connector Architecture (was J2CA)
  - provider-level interface for Resource Adapters to integrate Enterprise Information Systems into Java EE servers.
    - connection management
      - resource pooling
    - transaction management
      - integrate transactions across multiple resource managers
      - import transaction contexts into server
    - security access
    - thread management
      - allows resource adapter to allocate server threads for work
    - message delivery
      - integrate any messaging provider; not just JMS
    - optional, generic API between application program and resource adapter

Management/Deployment Services

- Management
  - Defines API for managing Java EE servers
  - Uses Java Management Extensions (JMX) API
- Deployment
  - Defines API between deployment tools and Java EE products

Other Services

- JavaMail
  - API and service provider for sending e-mail notifications
- JavaBeans Activation Framework (JAF)
  - API for handling MIME types

Java EE 5 APIs and Versions

- J2SE v5.0
  - JDBC, JNDI, RMI
- Java Persistence 1.0
- Common Annotations 1.0
- EJB3.0
- JMS 1.1
- JTA 1.1
- Servlet 2.5
- JSP 2.1
- StAX 1.0
- Web Services 1.2
- Web Services Metadata 2.0
- JAX-WS 2.0
- JAX-RPC 1.1
- SAAJ 1.3
- JAXR 1.0
- JSTL 1.2
- JSF 1.2
- JSP Debugging 1.0
- Java EE Management 1.1
- Java EE Deployment 1.2
- JACC 1.1
- Connector 1.5
- JavaMail 1.4
- JAF 1.1

Java EE Product Extensions

- Java EE products may supply implementations, in any form, for supplied Java EE APIs
  - different QOS, sizing, scaling, performance, cost, etc.
- Java EE products may provide additional APIs and protocols as extensions
- Java EE products may not add classes to Java API packages or change any properties of an existing Java API classes.
- Application use of product extensions is non-portable across Java EE products

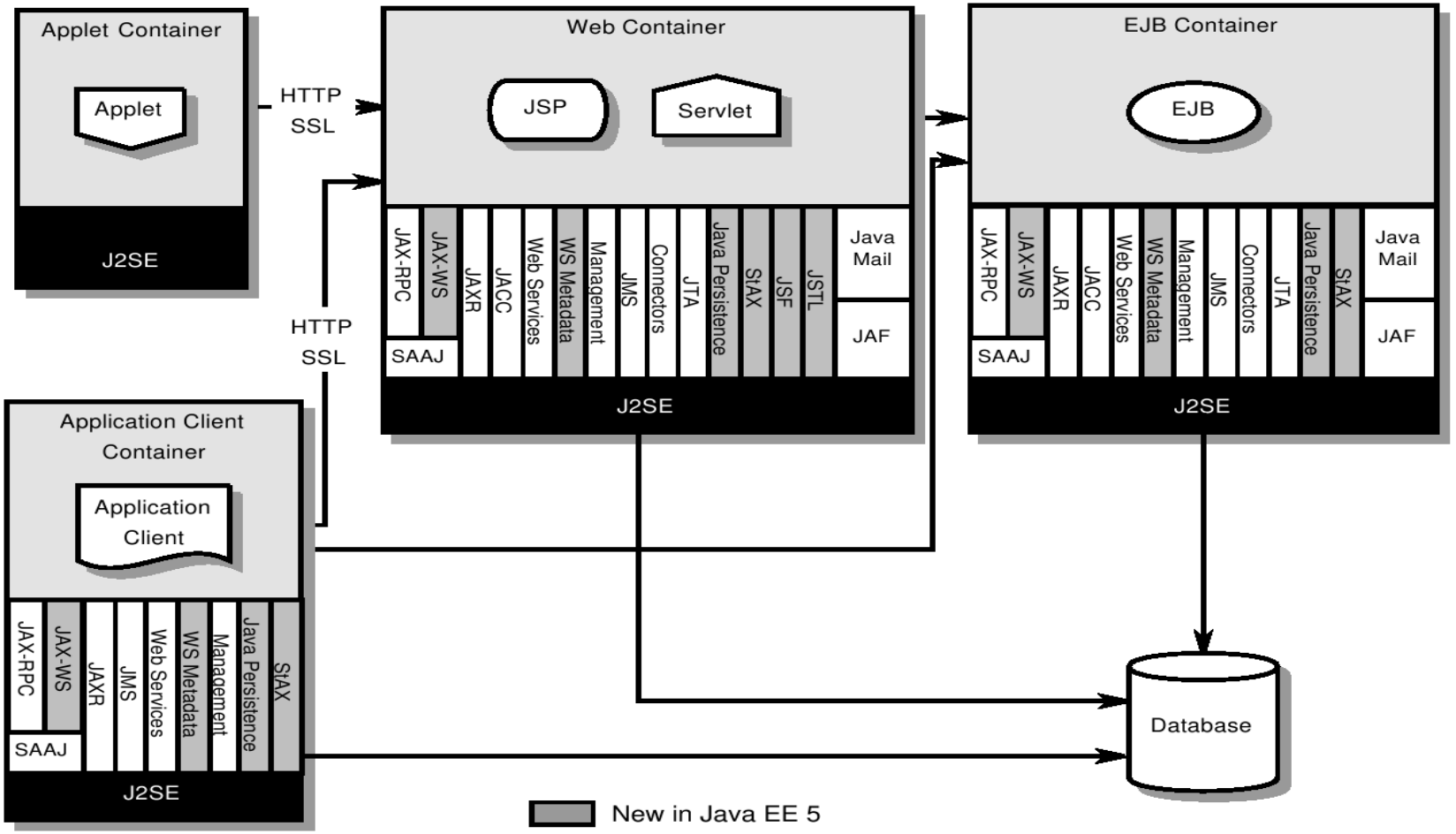
Java EE Roles

- Java EE Product Provider
  - vendor (e.g., Sun, JBoss, BEA, IBM)
- Application Component Provider
  - Java, HTML, etc. developer
- Application Assembler
  - integrates components into an application
- Deployer
  - integrates application with resources of the runtime
- System Administrator
  - monitors and tunes application after deployment
- Tool Provider
  - ex. Maven Cargo plugin
- System Component Provider
  - ex. external JMS provider

Java EE Release Highlights

- J2EE 1.3
  - Connector API
  - EJB (2.x) local interfaces and new CMP model
- J2EE 1.4
  - “Web services”
  - Management, Deployment, JAAS
- Java EE 5
  - “Ease of Development”
  - Annotations
  - Dependency Injection
  - Better defaults
  - Java Persistance API, StAX, JAX-WS
- Java EE 6
  - Profiles; making some components optional
  - More “Ease of Development”

Summary


Applet Container
Applet
J2SE
HTTP
SSL
HTTP
SSL
Web Container
JSP
Servlet
JAX-RPC
JAX-WS
JAXR
JACC
Web Services
WS Metadata
Management
JMS
Connectors
JTA
Java Persistence
StAX
JSF
JSTL
Java Mail
JAF
SAAJ
J2SE
EJB Container
EJB
JAX-RPC
JAX-WS
JAXR
JACC
Web Services
WS Metadata
Management
JMS
Connectors
JTA
Java Persistence
StAX
Java Mail
JAF
SAAJ
J2SE
Application Client Container
Application Client
JAX-RPC
JAX-WS
JAXR
JMS
Web Services
WS Metadata
Management
Java Persistence
StAX
SAAJ
J2SE
Database
New in Java EE 5

References

- Java™ Platform, Enterprise Edition 5 (Java EE 5) Specification (http://jcp.org/en/jsr/detail?id=244)