

Java Enterprise Edition

Einführung

Michael Lück

FOSSLC e.V.

Java Application Server Bootcamp, 2010

JEE Applikationen

- das interessanteste an JEE Applikationen ist, was sie **nicht** tun
 - Transaktionsverwaltung
 - Ressourcen Pooling
 - Verwaltung ihres Lebenszyklus
 - Authentisierung & Autorisierung
 - Persistierung (OR-Mapping u.ä.)
 - direkte Kommunikation mit anderen Anwendungen
- dennoch verfügen die Anwendungen über alle diese Funktionalitäten

JEE Applikationen

- das interessanteste an JEE Applikationen ist, was sie **nicht** tun
 - Transaktionsverwaltung
 - Ressourcen Pooling
 - Verwaltung ihres Lebenszyklus
 - Authentisierung & Autorisierung
 - Persistierung (OR-Mapping u.ä.)
 - direkte Kommunikation mit anderen Anwendungen
- dennoch verfügen die Anwendungen über alle diese Funktionalitäten

JEE Applikationen

- das interessanteste an JEE Applikationen ist, was sie **nicht** tun
 - Transaktionsverwaltung
 - Ressourcen Pooling
 - Verwaltung ihres Lebenszyklus
 - Authentisierung & Autorisierung
 - Persistierung (OR-Mapping u.ä.)
 - direkte Kommunikation mit anderen Anwendungen
- dennoch verfügen die Anwendungen über alle diese Funktionalitäten

JEE Applikationen

- das interessanteste an JEE Applikationen ist, was sie **nicht** tun
 - Transaktionsverwaltung
 - Ressourcen Pooling
 - Verwaltung ihres Lebenszyklus
 - Authentisierung & Autorisierung
 - Persistierung (OR-Mapping u.ä.)
 - direkte Kommunikation mit anderen Anwendungen
- dennoch verfügen die Anwendungen über alle diese Funktionalitäten

JEE Applikationen

- das interessanteste an JEE Applikationen ist, was sie **nicht** tun
 - Transaktionsverwaltung
 - Ressourcen Pooling
 - Verwaltung ihres Lebenszyklus
 - Authentisierung & Autorisierung
 - Persistierung (OR-Mapping u.ä.)
 - direkte Kommunikation mit anderen Anwendungen
- dennoch verfügen die Anwendungen über alle diese Funktionalitäten

JEE Applikationen

- das interessanteste an JEE Applikationen ist, was sie **nicht** tun
 - Transaktionsverwaltung
 - Ressourcen Pooling
 - Verwaltung ihres Lebenszyklus
 - Authentisierung & Autorisierung
 - Persistierung (OR-Mapping u.ä.)
 - direkte Kommunikation mit anderen Anwendungen
- dennoch verfügen die Anwendungen über alle diese Funktionalitäten

JEE Applikationen

- das interessanteste an JEE Applikationen ist, was sie **nicht** tun
 - Transaktionsverwaltung
 - Ressourcen Pooling
 - Verwaltung ihres Lebenszyklus
 - Authentisierung & Autorisierung
 - Persistierung (OR-Mapping u.ä.)
 - direkte Kommunikation mit anderen Anwendungen
- dennoch verfügen die Anwendungen über alle diese Funktionalitäten

JEE Applikationen

- das interessanteste an JEE Applikationen ist, was sie **nicht** tun
 - Transaktionsverwaltung
 - Ressourcen Pooling
 - Verwaltung ihres Lebenszyklus
 - Authentisierung & Autorisierung
 - Persistierung (OR-Mapping u.ä.)
 - direkte Kommunikation mit anderen Anwendungen
- dennoch verfügen die Anwendungen über alle diese Funktionalitäten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Wie geht das?

- Spezifikation von 4 Applikationstypen/Komponententypen
 - Application Clients
 - Applets
 - Web-Components
 - Enterprise Java Beans (EJB)
- Spezifikation von Ablaufumgebungen für die Applikationstypen: **Container**
 - kümmern sich um Deployment, Verwaltung und Ausführung von Anwendung eines Typs
 - bieten den Komponenten einheitliche Sicht auf APIs
 - injizieren benötigte Dienste in Instanzen der Komponenten
 - Transaktionsmanagement, Sicherheitsprüfungen, Ressourcen Pooling, ...
 - regeln Kommunikation zwischen Komponenten

Architektur

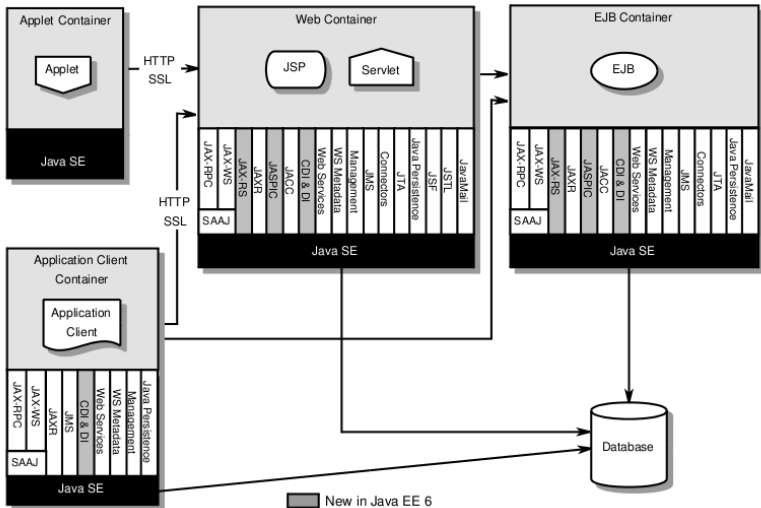


Abbildung: Java EE Architektur [Sun09]

- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit

- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit

- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit

- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit

- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit

- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit


- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit


- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit


- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit

JEE - Die Spezifikation

- JEE = Sammlung von Spezifikationen
 - Standardisierungen von Best Practices und quasi Standards
 - oftmals durch Community gesetzt
 - Standards werden auf einander abgestimmt
- offener Standard
 - erstellt durch den Java Community Process
 - Mitglieder können Vorschläge einreichen
 - Öffentlichkeit kann Drafts kommentieren
- Implementierung des Standards: Java Application Server
 - stellen insbesondere Container für die Applikationstypen bereit

-  Schwichtenberg, Holger ; Toth, Stefan ; Zörner, Stefan:
.NET 4.0 vs. Java EE 6: Aus der Vogelperspektive.
iX 4/2010

-  Sun Microsystems:
Java Platform. Enterprise Edition (Java EE) Specification, v6.
[http://jcp.org/aboutJava/communityprocess/final/
jsr316/index.html](http://jcp.org/aboutJava/communityprocess/final/jsr316/index.html).
Version: 2009

-  Wikipedia:
*Java Platform, Enterprise Edition — Wikipedia, Die freie
Enzyklopädie.*
[http://de.wikipedia.org/w/index.php?title=Java_
Platform,_Enterprise_Edition&oldid=69879857](http://de.wikipedia.org/w/index.php?title=Java_Platform,_Enterprise_Edition&oldid=69879857). –
[Online; Stand 11. April 2010]