



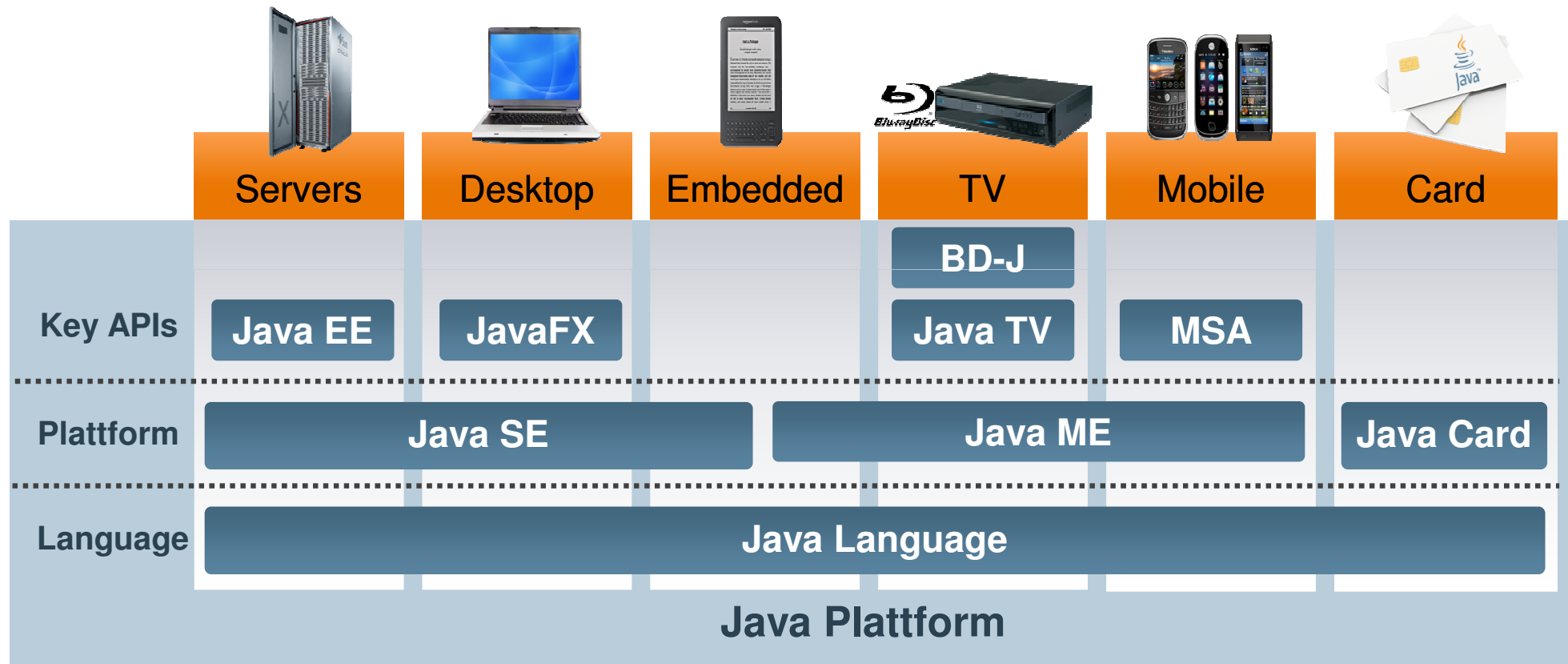
Java im Aufwind

Die evolutionäre Veränderung der Java Plattform

Wolfgang Weigend
Sen. Leitender Systemberater
Java Technologie und Architektur



Die Java Plattform



Wie sich die Java Plattform schrittweise verändert

- Java Strategie Treiber
- Java Community
- JDK 7 Review
- JDK 8 und darüber hinaus
- Entwicklungsumgebung
- JavaFX & OpenJFX
- Java EE & Ablaufumgebung
- Zusammenfassung

Java Strategie

- **Höchste Priorität für Java und zunehmendes Investment in die Plattform**
 - Kontinuierliche Bewegung in Richtung **”free & open”**
 - Verbesserter Support für Entwickler Community, Open Source Community und JCP
- **Breite Hersteller-Unterstützung für Java**
- **Evolutionäre Weiterentwicklung der Sprache**
 - Ideen reifen in ausgeprägten experimentellen Sprachen
 - Abwärtskompatibilität bleibt extrem wichtig für die Sprache

Zusammenarbeit mit der Community

- OpenJDK Mitglieder: IBM, Apple, SAP, Azul Systems, Twitter
- OpenJDK Community Richtlinien verabschiedet
- JVM Language Summit (Jährlich)
- JUG Leaders Call alle zwei Wochen
- JUG Leader Summit im Januar (Jährlich)
- Über 250 JUG´s aus 50 Ländern am JDK 7 Launch beteiligt
- Community Keynote auf der JavaOne 2012

Rückblick zum Java 7 Launch Event auf dem Java Forum Stuttgart 2011



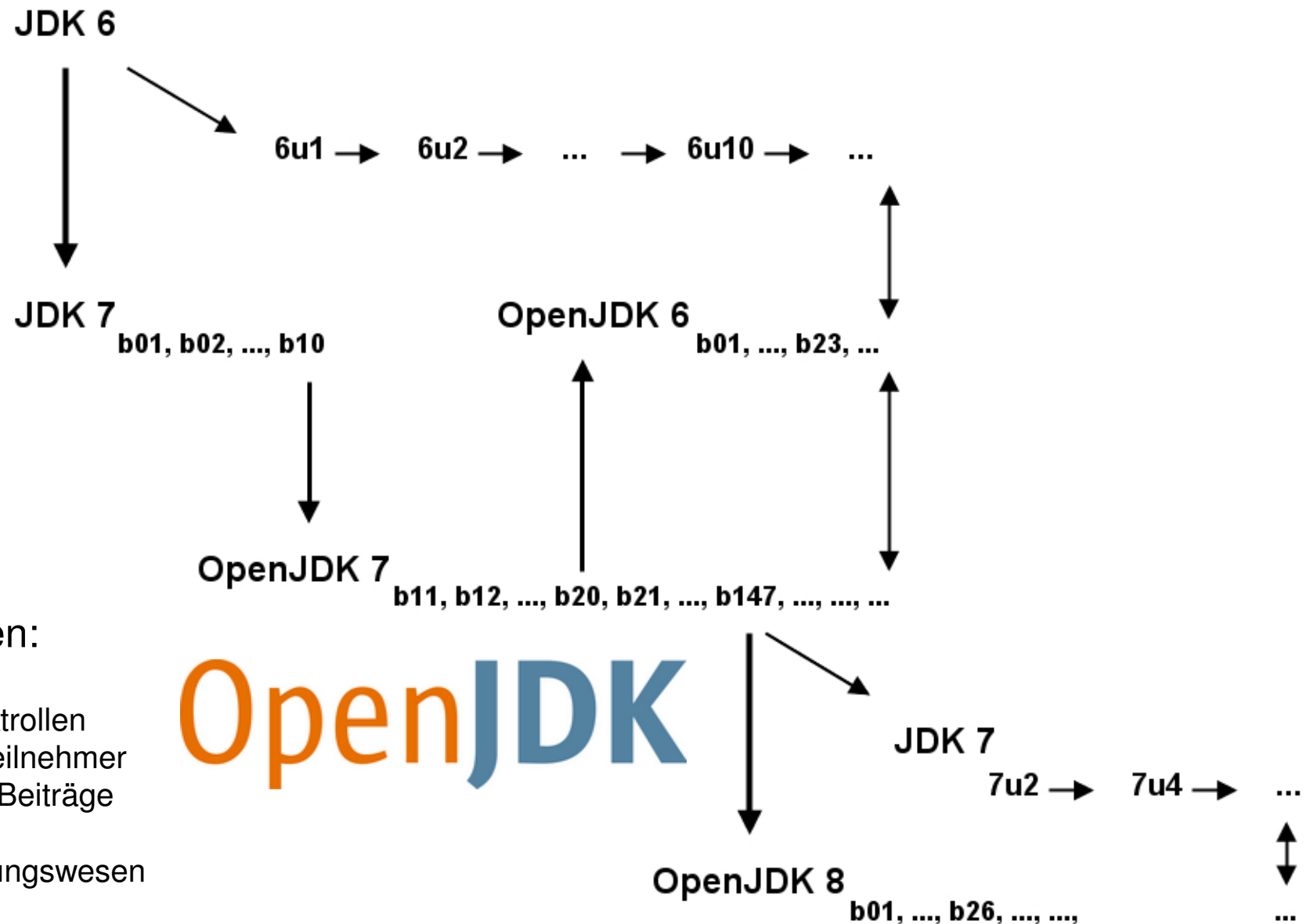
Introducing Java 7

MOVING JAVA FORWARD



Java 7 Satellite Broadcast Viewing im Hegel-Saal

OpenJDK Stammbaum



OpenJDK Richtlinien:

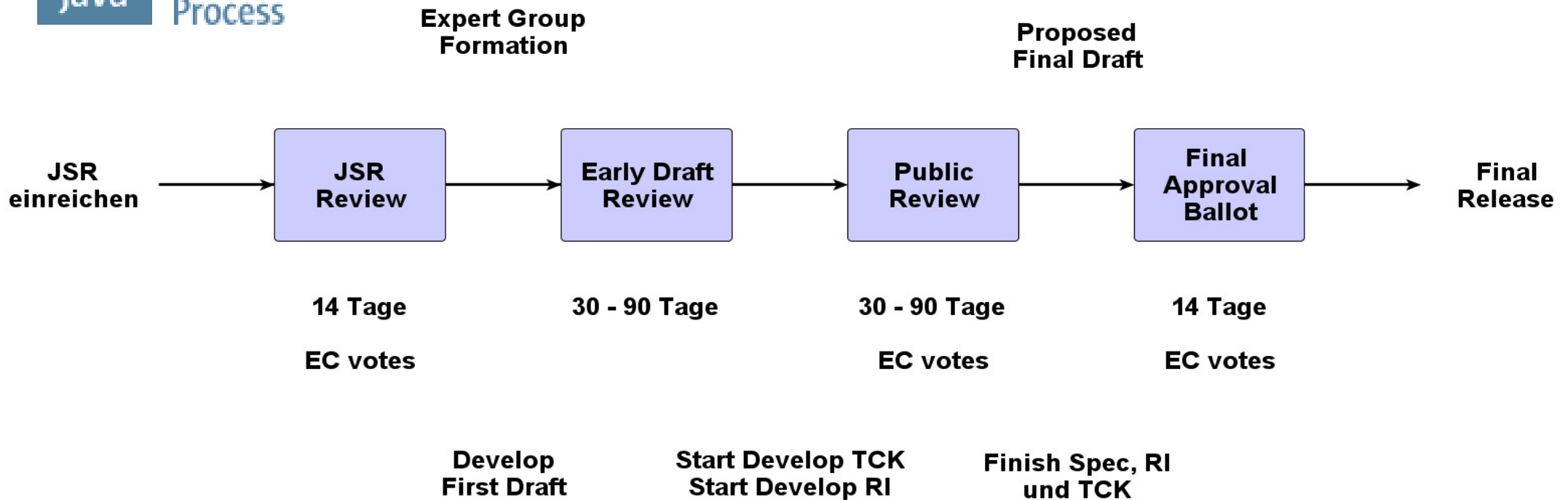
- Rollen & Gruppen
- Gruppenrollen & Projektrollen
- Projekte & OpenJDK Teilnehmer
 - Bug-Fixing & Code-Beiträge
- Governing Board
- Reporting & Entscheidungswesen
- Abstimmung

Wie sich Java verändert und anpaßt

Durch die Community - Für die Community

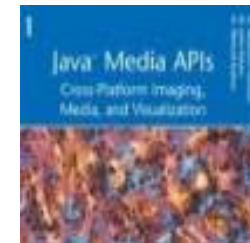


Java
Community
Process

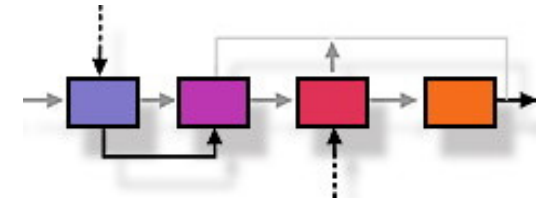


JCP wird reformiert: JSR-348

- Mehr Entwickler im Executive Committee
 - SOUJava
 - Goldman Sachs
 - London JavaCommunity
- JCP startet ein Programm zur Reform
 - JSR-348: Towards a new version of the JCP



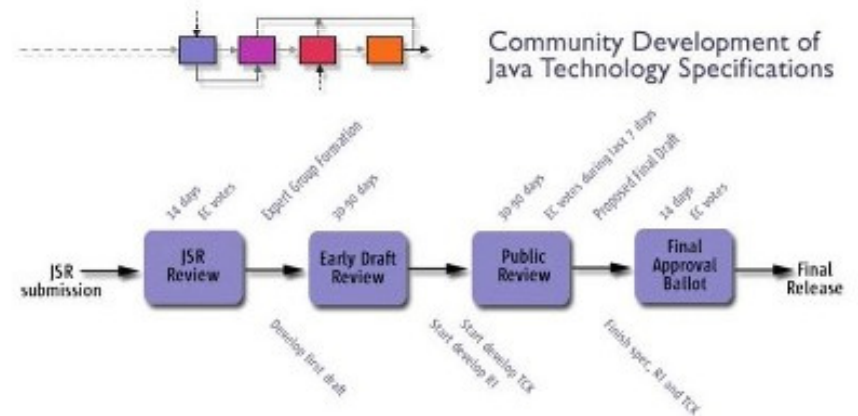
Executive Member Merge Projekt JSR-355



- **Merge-Projekt ist öffentlich und der zweite von drei JSR's zur Veränderung vom JCP Prozess**
- **Veränderungen im JCP Prozess Dokument und den bestehenden Komitee-Regeln mit dem Ziel der Verschmelzung beider Executive Komitee Mitglieder zu einem einzigen Executive Komitee**
- **Expert Group Mitglieder führen öffentlich nachvollziehbare Arbeitsweise fort**
 - Diskussion ist öffentlich
 - Arbeitsunterlagen und Protokolle werden ständig veröffentlicht
 - Issue Tracker

A major Revision of the Java Community Process: JSR-358

- Change of Java Specification Participation Agreement (JSPA)
- Independent Implementations (Not derived from the RI)
- Licensing and Open Source
- Transparency Requirements
- Compatibility Policy and TCKs
- The Role of Individual Members
- Patent Policy
- Intellectual Property flow
- Refactoring and Cleanup



Verwirklichte Pläne und neue Projekte

JDK 7

- Available for Windows, Linux, Solaris, embedded platforms
- Mac OS X Developer Preview Release JDK7u4

JVM Konvergenz

- Initial JRockit/HotSpot Konvergenz in JDK 7
- No more Perm-Gen
- Java Flight Recorder
- Memory-Leak Server

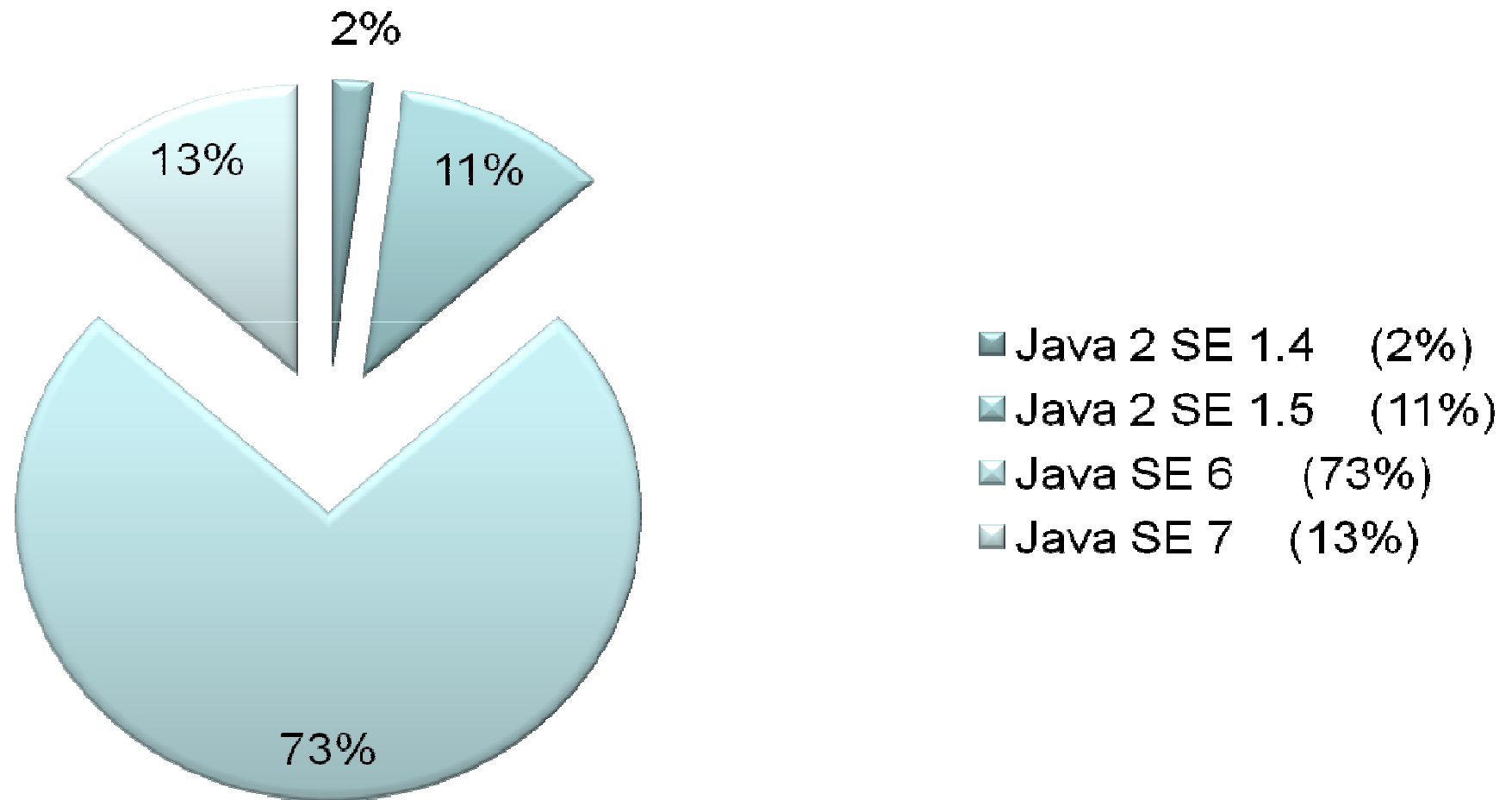
Open Source

- OpenJDK official Java SE 7 Reference Implementation
- OpenJDK project initiated for JDK 8 (JDK 8 b32)
- jdk8-dev@openjdk.java.net

Ziele der JVM Verschmelzung (HotRockit)

- Maximale Performance und Skalierbarkeit
- Multi-Core und Parallelisierungs Optimierungen
- Feingranulares Monitoring, Profiling und Management
- Extrem effiziente Garbage Collection

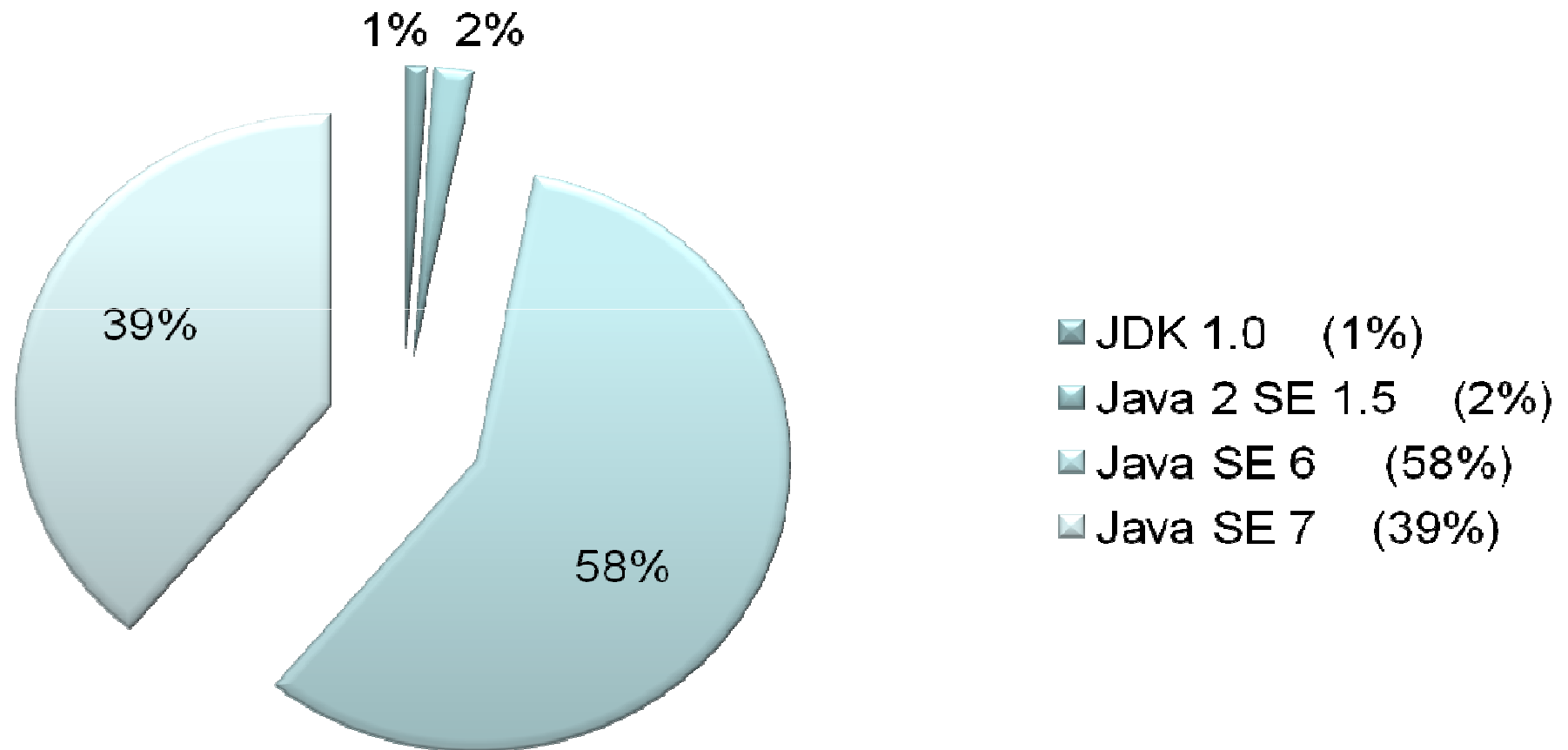
Welche Java-Version nutzen Sie beruflich?



Quelle: <http://it-republik.de/jaxenter/quickvote/results/1/poll/149>

Teilnehmer: 494

Welche Java-Version nutzen Sie privat hauptsächlich?

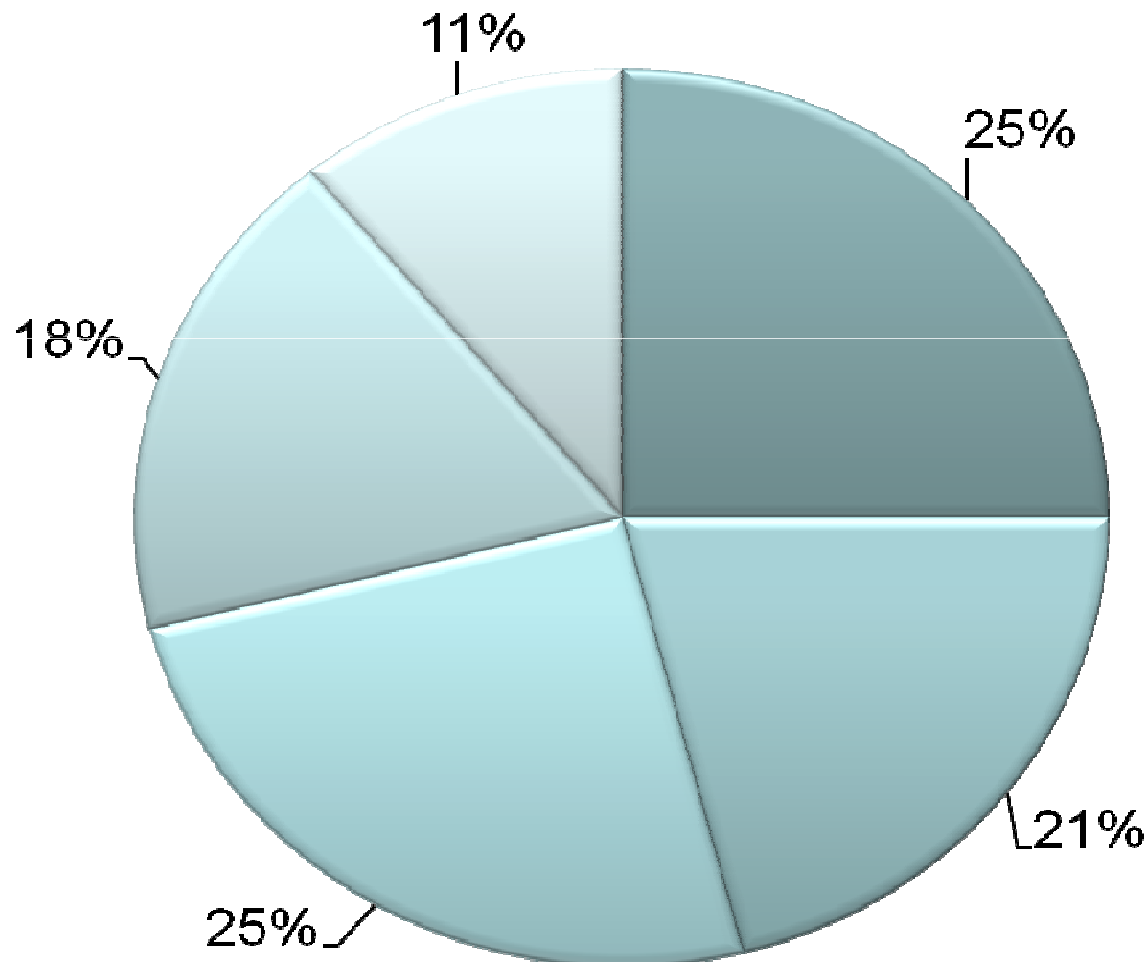


Quelle: <http://it-republik.de/jaxenter/quickvote/results/1/poll/151>

Teilnehmer: 318

Wird Java 7 eingesetzt?

Umfrage auf java.net: “*Have you tried out Java 7 yet?*”



- Yes, and I'm working with it regularly (25%)
- I've experimented with it a bit (21%)
- I plan to get started with Java 7 soon (25%)
- I'm waiting for a bug fix release (18%)
- No, and I don't plan to (11%)

JDK 7 Update Releases

Fehlerbereinigung

OpenJDK

- Projekt entwickelt Updates für JDK 7
- Sponsor ist die Build Group
- Mailing Liste lautet jdk7u-dev
 - Technische Diskussion zum JDK 7 Updates Projekt
 - Archive vorhanden
 - Nachrichten an alle Teilnehmer verschicken:
 - jdk7u-dev@openjdk.java.net
 - jdk7u-dev Subscription
- **Java SE 7 Update 2 Release verfügbar: JDK 7u5**
 - **Changesets und aktuelle Änderungen: JDK 7u6 build b14**
 - **Projekt Feedback Forum für Java SE Snapshots**
 - **Report a Bug oder Request a Feature**



JDK 7 Funktionsumfang

- Coin – Kleine Sprachverbesserungen (JSR-334)
- Dynamic Language Support (JSR-292)
- Concurrency und Collections Updates (JSR-166y)
- Netzwerk und File System (JSR-203)
- Sicherheit
- Internationalisierung
- Weitere Verbesserungen
- JVM Konvergenz



Languages Running on the JVM

Groovy

JRuby

...

...

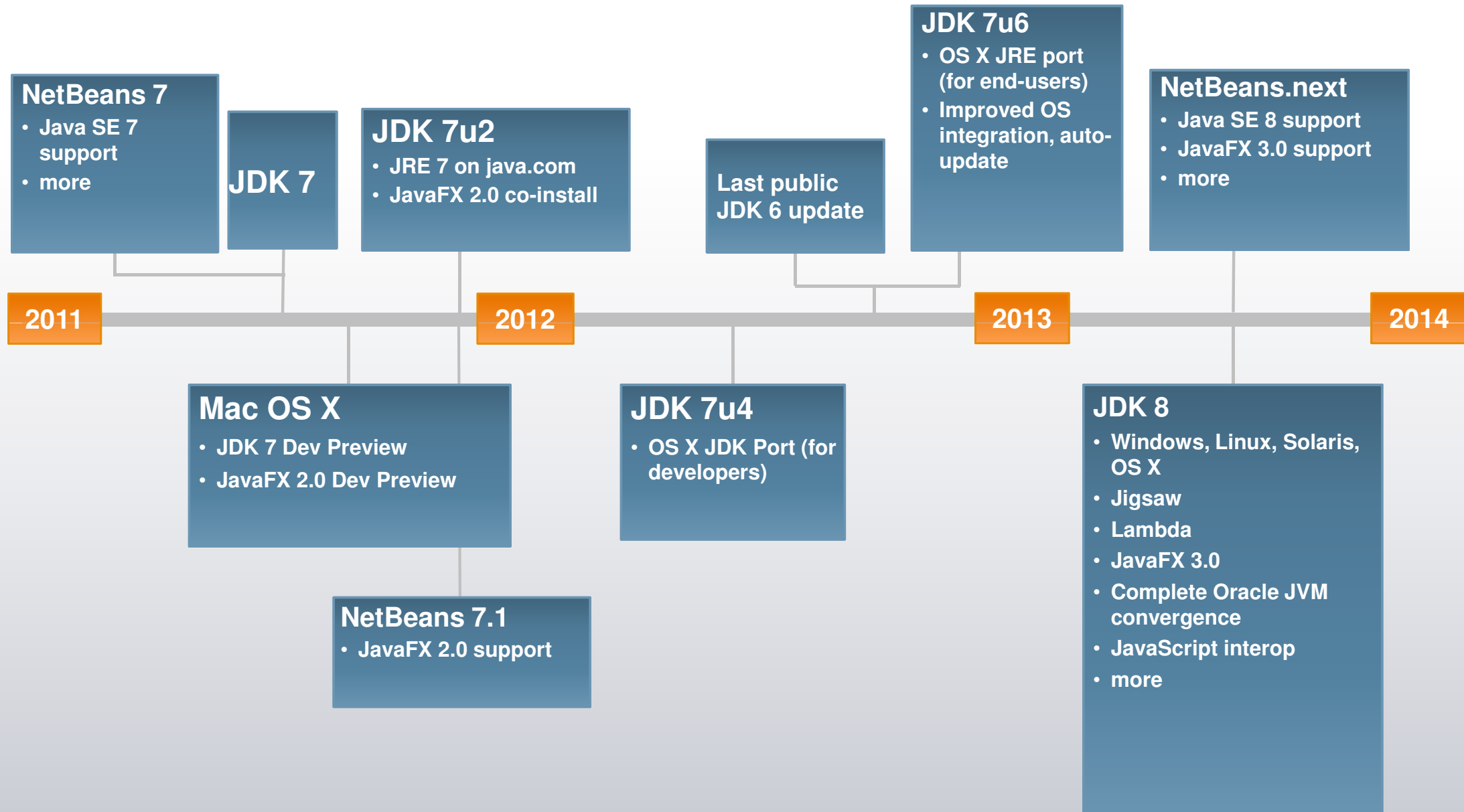
Scala

Clojure



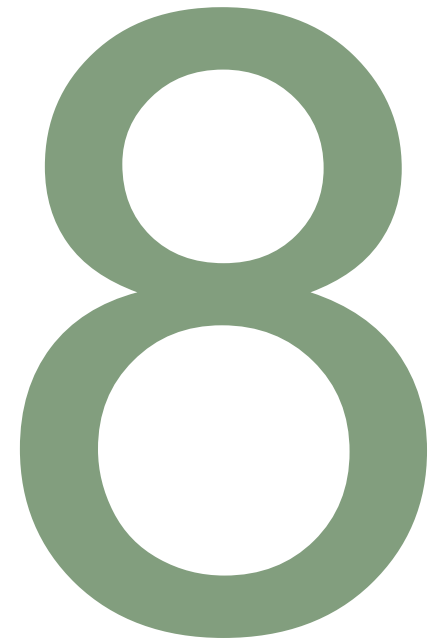
JavaScript

JDK Roadmap



JDK 8 geplant für September 2013

- Feedback der Community – **Zwischen den neuen JDK-Versionen werden 2 Jahre Zeit benötigt**
- Release Datum September 2013
- Funktionsumfang:
 - Projekt Jigsaw
 - komplette Plattform Modularisierung und Container-Unterstützung
 - Projekt Lambda (JSR-335)
 - JavaScript Interoperabilität
 - Device-Unterstützung
- Weitere kleine Sprachverbesserungen im Projekt Coin Teil 2



JDK 8 - September 2013

Thema	Beschreibung der geplanten Inhalte
Project Jigsaw	• Module system for Java applications and the Java platform
Project Lambda	• Closures and related features in the Java language (JSR-335) – Lambda Expressions – Interface Evolution – Concurrent Bulk Data Operations in Java collections APIs (filter/map/reduce)
Oracle JVM Convergence	• Complete migration of performance and serviceability features from JRockit, including Mission Control and the Flight Recorder
JavaFX 3.0	• Next generation Java Client
JavaScript	• Next-gen JavaScript-on-JVM Engine (Project Nashorn) • JavaScript/Java interoperability on JVM
Device Support	• Multi-Touch (JavaFX), Camera, Location, Compass and Accelerometer
Developer Productivity	• Annotations on types (JSR-308), Minor language enhancements
API and Other Updates	• Enhancements to Security, Date/Time, (JSR-310) Networking, Internationalization, Accessibility, Packaging/Installation
Open Source	• Open Development in OpenJDK, open source additional closed components

Lambda Ausdrücke

- **Vorschlag zum Sprachumfang Java SE 8**

- Die offizielle Syntax steht noch nicht fest!

- **Lambda Expressions (closures)**

```
/* (int x, int y) {return x+y; } */
```

- **Kleine Code-Anteile, winzige anonyme Methoden**
- **Parameter Liste → -> Operator → Expression od. Statements**

```
(String x) -> {return !x.isEmpty();}
```

- **Was hergeleitet werden kann, kann auch weggelassen werden**

```
x -> !x.isEmpty()
```

Modularisierung in JDK 8

Projekt Jigsaw Java Module System

```
module com.greetings @ 0.1 {  
    requires org.astro @ 1.2;  
    class com.greetings.Hello;  
}
```

```
package com.greetings;  
import org.astro.World;  
public class Hello {  
    public static void main(String[] args) {  
        System.out.println("Hello, " + World.name() + "!");  
    }  
}
```

```
module org.astro @ 1.2 { }
```

```
package org.astro;  
public class World {  
    private static native byte[] getName();  
    static {  
        System.loadLibrary("world");  
    }  
    public static String name() {  
        return new String(getName());  
    }  
}
```

JDK 9 und die Vision darüber hinaus

Interoperability

- Multi-language JVM
- Improved Java/Native integration

Cloud

- Multi-tenancy support
- Resource management

Ease of Use

- Self-tuning JVM
- Language enhancements

Advanced Optimizations

- Unified type system
- Data structure optimizations

Works Everywhere and with Everything

- Scale down to embedded, up to massive servers
- Support for heterogeneous compute models



NetBeans IDE 7.1

- Entwicklungswerkzeug für Desktop, mobile und Web Anwendungen
 - **Java EE 6, Java SE 6 (REST, CDI, JPA), Java SE 7, Swing, Java FX**
 - Java Editor support for Project Coin (Diamond Operator, Strings in switch, Multi-catch)
 - Bulk refactoring of projects and packages to Java SE 7
 - JavaFX 2.0 Full edit/compile/debug cycle support
 - Visual debugging of JavaFX apps
 - Java ME und Embedded
 - PHP, Ruby, Groovy, C / C++
 - HTML5 Editing, JSON Formatter
 - Debugger, Profiler, Refactoring
- Läuft auf MS Windows, Linux, Mac OS X und Solaris
- NetBeans IDE ist open-source und frei verfügbar
- Feature Liste & Builds
 - netbeans.org/community/releases/roadmap.html



ORACLE®

Unterstützung für eclipse IDE

- Eclipse Projekte mit Werkzeugen und Frameworks
- Ganzheitliche Unterstützung vom Software Development Lebenszyklus
 - Modellierung
 - Entwicklung
 - Deployment Werkzeuge
 - Reporting
 - Daten Manipulation
 - Testing und Profiling
- Primärer Fokus zur Entwicklung von Anwendungen
 - Java EE, Web Services und Web Anwendungen
- Eclipse Unterstützung für andere Programmiersprachen
 - C/C++, PHP, andere

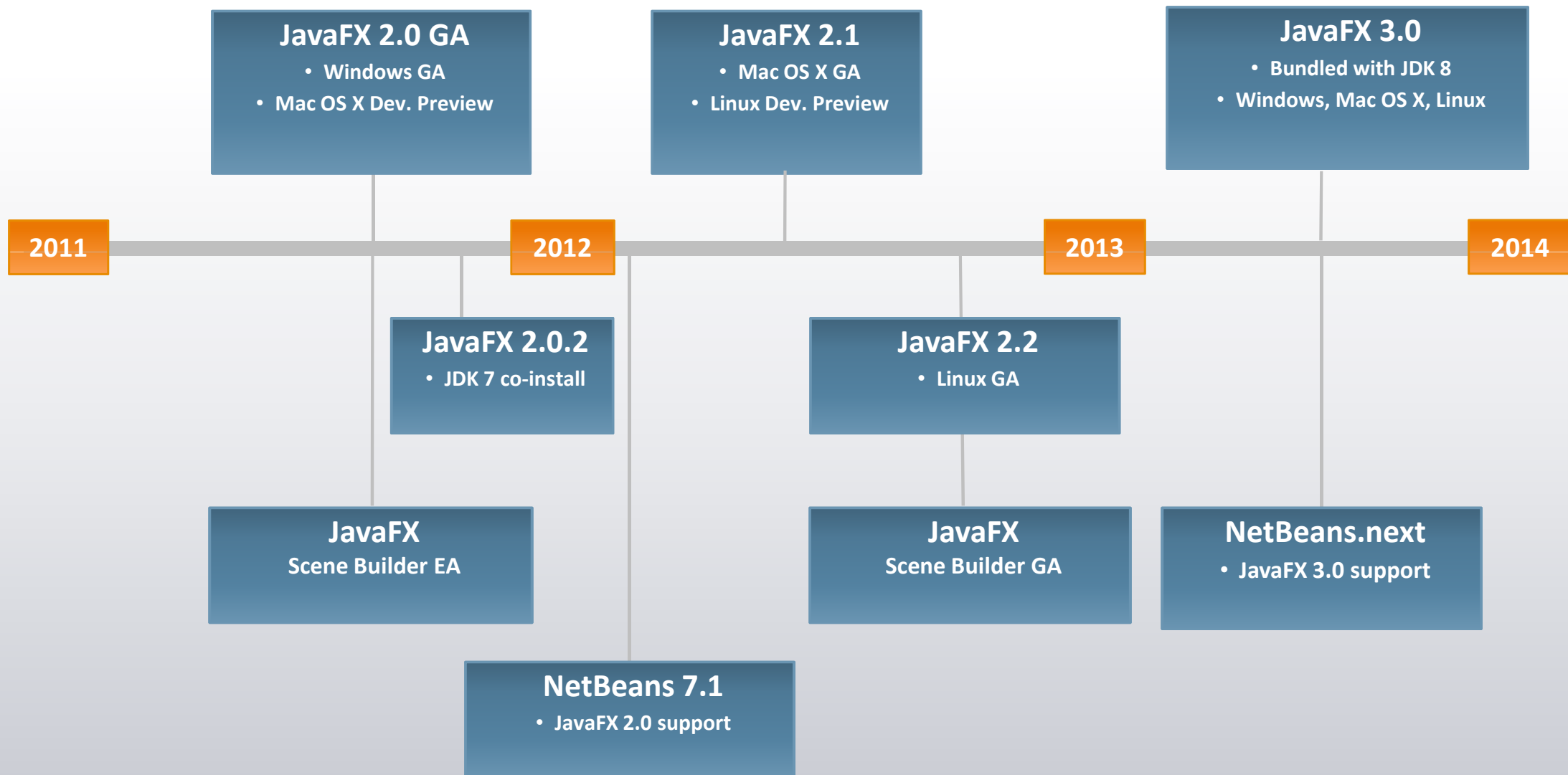


JavaFX 2.0

- **JavaFX als strategische UI für Rich-Client-Applikationen**
- **Einheitliche Applikationsentwicklung für Java und Web-Lösungen**
 - Browser Plug-in, Web Start, Native Executables
 - Hardware Accelerated Graphics (DirectX, OpenGL)
 - JavaFX wird mit HTML-DOM ausgestattet
 - JavaFX mit WebView für HTML5 Features (Web Sockets, offline Browsing, lokale Datenbank)
 - Leistungsfähige JavaScript Engine
 - JavaFX als Applet eingebettet in einer Web-Seite lauffähig
- **JavaFX 2.0 Plattform Sprachwechsel**
 - Java als native Sprache - anstatt JavaFX Script
 - JavaFX APIs in Java implementiert
 - Vorteile bei Verwendung von Generics, Annotations und Multithreading für JavaFX
- **JavaFX 2.1 verfügbar und mit NetBeans 7.1 unterstützt**
 - Migrationspfad für Swing- und SWT-basierte Anwendungen
 - JFXPanel Komponente ermöglicht das Einbinden von JavaFX Anwendungen in Swing
 - Open Source mit OpenJFX und im JCP standardisiert
- **JavaFX 2.2 Developer Preview build11 und Scene Builder 1.0**



JavaFX Roadmap



OpenJFX Project

- **OpenJFX is the open source** home of JavaFX development
- **Goal of OpenJFX is to build the next-generation Java client toolkit**
- **Project OpenJFX intends to file a JSR in the Java SE 9 timeframe and be part of the JDK**
- In 2011 the JavaFX toolkit was **completely rewritten in Java** and released by Oracle as JavaFX 2.0
- Oracle announced that it would **donate the JavaFX toolkit to the open source community** and by November 2011 the OpenJDK Community had agreed to take it on
- Project OpenJFX is currently focused on absorbing the existing code base from Oracle and **building a functioning community around it**
- Oracle's **code will be donated in phases** with the initial piece being the UI Controls classes

Benutzerumgebung und Client-Interaktion

- **HTML 5 Clients**

- Verwendung von bi-direktionaler Kommunikation zur Nutzung Event-gesteuerter Anwendungen
- Umfassendes, standardisiertes Client-Side-Programmiermodell:
 - APIs, schnelles JavaScript und standardisierte offline-Fähigkeiten

- **Cloud / PaaS**

- Vorhandene Building Blocks: WebSockets (Comet), JSON (JAX RS), RESTful WS
- Fehlende Benachrichtigung bei Datenänderungen, asynchrone Infrastruktur, ...

- **Programmier Modell**

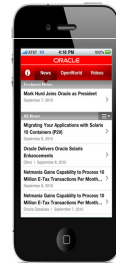
- Mehr als die Summe aller Einzelteile
- Web-native, bi-directionales Binding zur Cloud
- Vereinigung von Java ME, Java SE und Java EE

Ganzheitliche Betrachtung zur Unterstützung von dynamische Rich-Clients

HTML 5 Browser

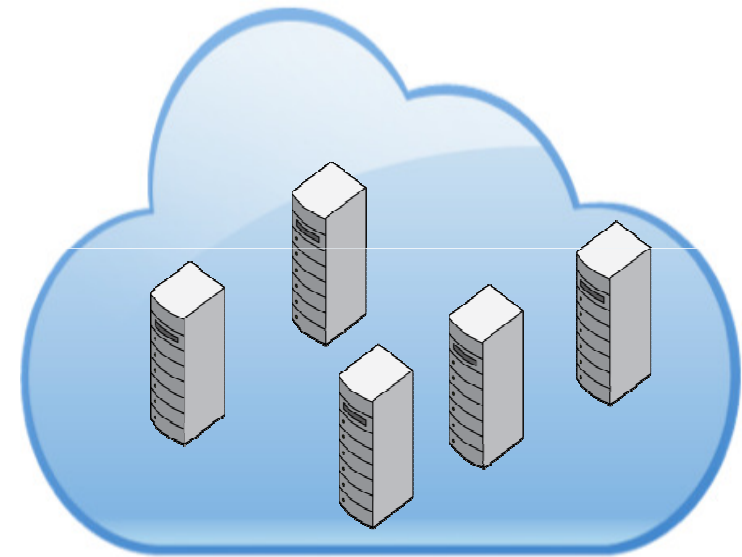


HTML & Java
hybride
Anwendungen



JSON over
WebSocket

Java
Anwendungen



Java EE PaaS

Wie sich die Entwicklung durch die Wolke schrittweise verändert

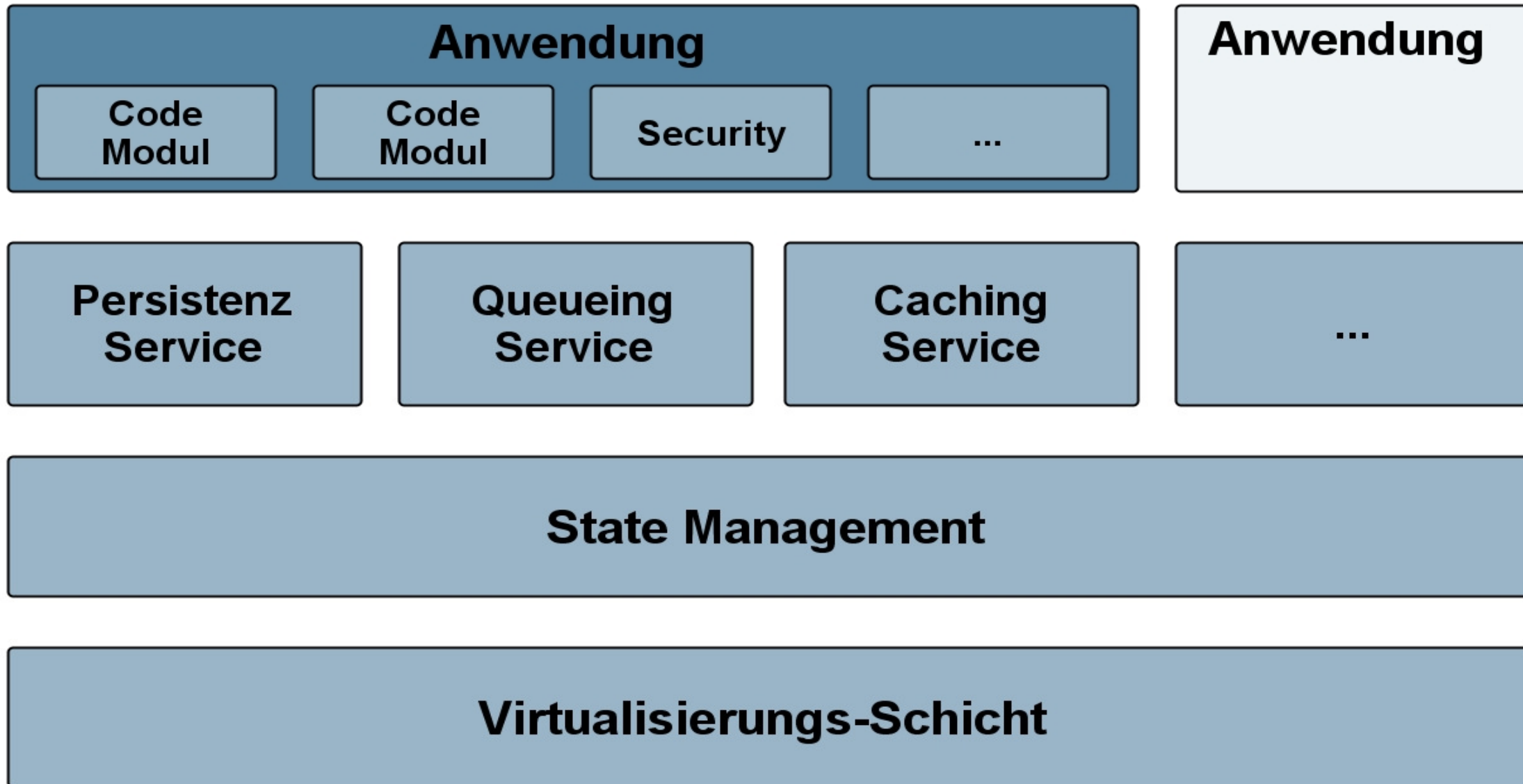
- **Entwickler für Unternehmensanwendungen wollen Cloud-Lösungen von der eigenen IT-Abteilung**
 - IaaS als neuer “Self-Service Data Center”
 - Unmittelbar, On-Demand Provisioning
 - Hosted, sichere Cloud Services
- **Virtualisierung ist ein wertvoller technischer “Building Block”, aber keine Plattform**
- **Entwickler schauen nach einem PaaS Standard für die nächste Generation Cloud-basierter Anwendungen**
 - Die Java EE Plattform eignet sich als PaaS Standard
- **Java EE hat bereits vergleichbare Herausforderungen für die IT gelöst**



Java EE 7 und PaaS

- **Definition neuer Rollen für die Plattform**
 - Anpassung und Adaption vom PaaS-Modell
- **Einführung von Metadaten für**
 - Service-Provisionierung und Konfiguration
 - Service-Qualität (QoS), Elastizität
 - Gemeinsam genutzte Applikationen und Ressourcen
 - Konfigurationsänderung und Anpassung (Customization)
- **Hinzufügen neuer API's für PaaS-Umgebungen**
 - JAX-RS Client API
 - Caching API
 - State Management
 - JSON, ..
- **Erweiterung existierenden APIs für Mandantenfähigkeit**

Java EE 7 PaaS Architektur



Java EE 7 Services

- PaaS Anwendungen konsumieren Services
 - Persistenz, Queueing, Caching, ...
- Service Metadaten ermöglichen einfache Nutzung beim Deployment in PaaS

```
@DataSourceDefinition(  
    name="java:app/jdbc/myDB",  
    className="oracle.jdbc.pool.OracleDataSource",  
    isolationLevel=TRANSACTION_REPEATABLE_READ,  
    initialPoolSize=5  
)
```

Java EE 7 Services

- PaaS Anwendungen konsumieren Services
 - Persistenz, Queueing, Caching, ...
- Service Metadaten ermöglichen einfache Nutzung beim Deployment in PaaS

```
@JMSConnectionFactory (  
    name="java:app/myJMSConnectionFactory",  
    resourceType="javax.jms.QueueConnectionFactory")
```

```
@JMSDestination(  
    name="java:app/myQueue",  
    resourceType="javax.jms.Queue")
```

Java EE 7 Services

- PaaS Anwendungen konsumieren Services
 - Persistenz, Queueing, Caching, ...
- Service Metadaten ermöglichen einfache Nutzung beim Deployment in PaaS

```
@ConnectorService (  
    name="java:app/myCustomConnector",  
    type="com.extraServices.customConnector.class",  
    properties = {...}  
)
```

Java EE 7 Services

- Erzeugte Entitäten provisionieren, verwalten und überwachen innerhalb der PaaS Ablaufumgebung
- Signifikante Software-Funktionalität die zur Ausführung einer Anwendung notwendig ist
- Typen
 - **Provisioniert:** Installiert, konfiguriert, von der Plattform verwaltet, Anwendungsbezogen oder gemeinsam verwendet
 - **Extern:** Enterprise-existent, Plattform kennt die Konfiguration
 - **Shared:** Mehrfache Verwendung in verschiedenen Umgebungen pro Benutzer, pro System
- Beispiele:
 - Java EE Applikations-Service
 - Datenbank
 - Load Balancer

Java EE 7 Services

- Service-Definition
 - Verwendet Metadaten zur Service-Provisionierung und Konfiguration
 - Spezifikation von funktionalen und nicht-funktionalen Service-Charakteristiken
 - Template-Abgleich
 - Explizite Template Spezifikation
- Service-Referenz
 - Repräsentiert die Abhängigkeit der Applikations-Komponente zum Service
 - Explizit: Benutzer-spezifisch durch Deployment Deskriptoren
 - Implizit und Auffinden (Discovery): Information ist im Archiv enthalten

Java EE 7 – Provisioning

Java EE Vorhandenes Modell

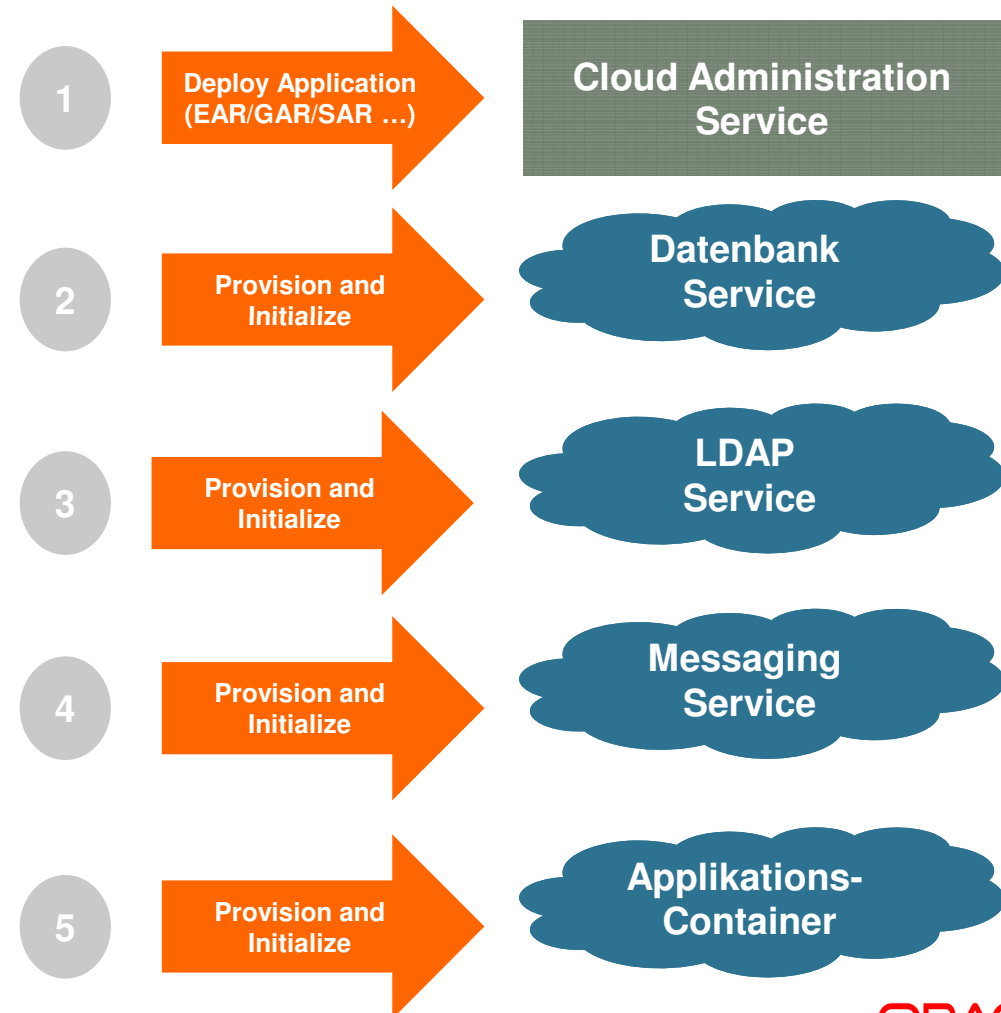
- Konfiguration Java EE Ressourcen – JDBC, JMS, etc.
- Deploy Application Archive (.EAR)

Java EE 7 Modell

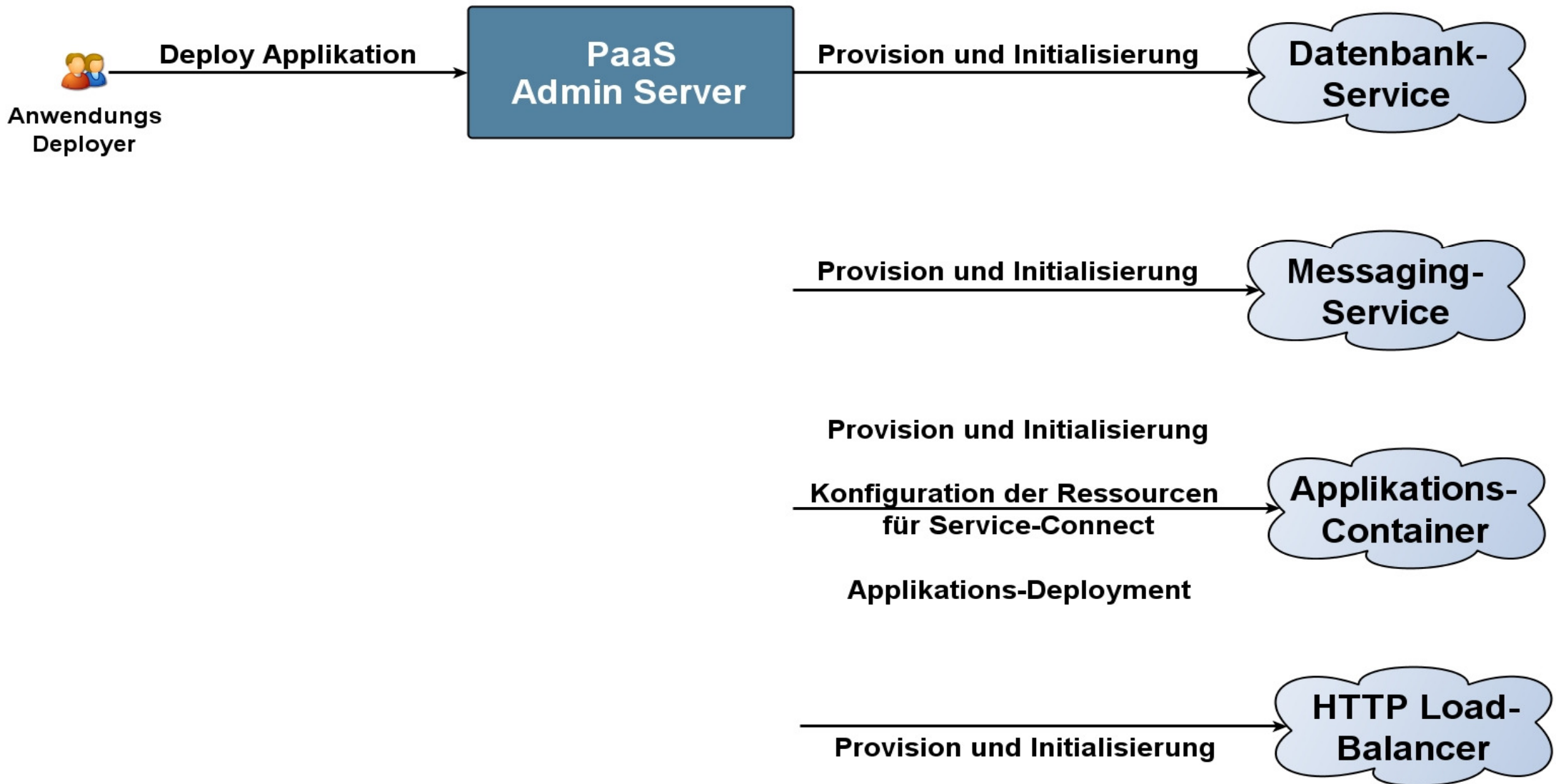
- Auto-Provision Services von Application Dependencies
e.g. Datenbank, LDAP

Extensible Deployment Modelle für Frameworks

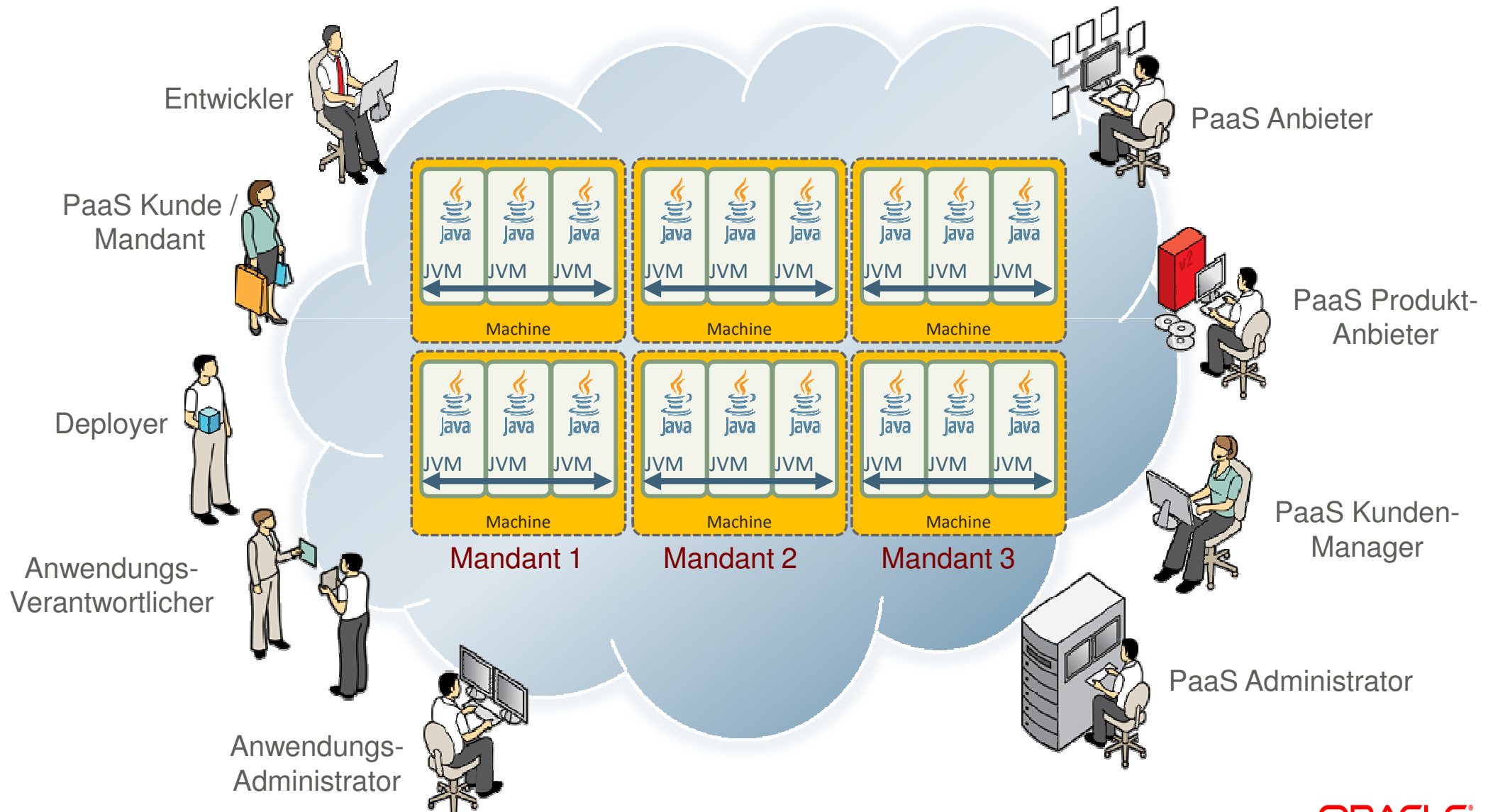
- Spring, Seam, etc.



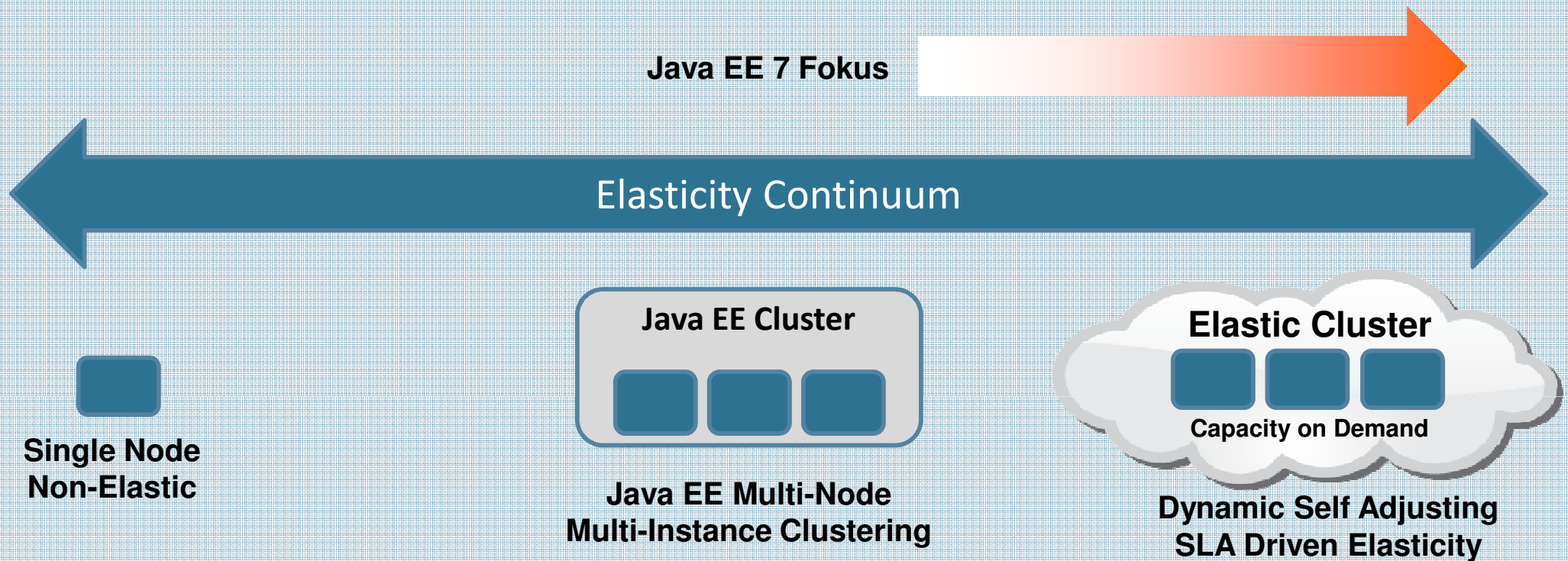
Java EE 7 – PaaS Applikations-Deployment



Java EE 7 - Rollenmodell



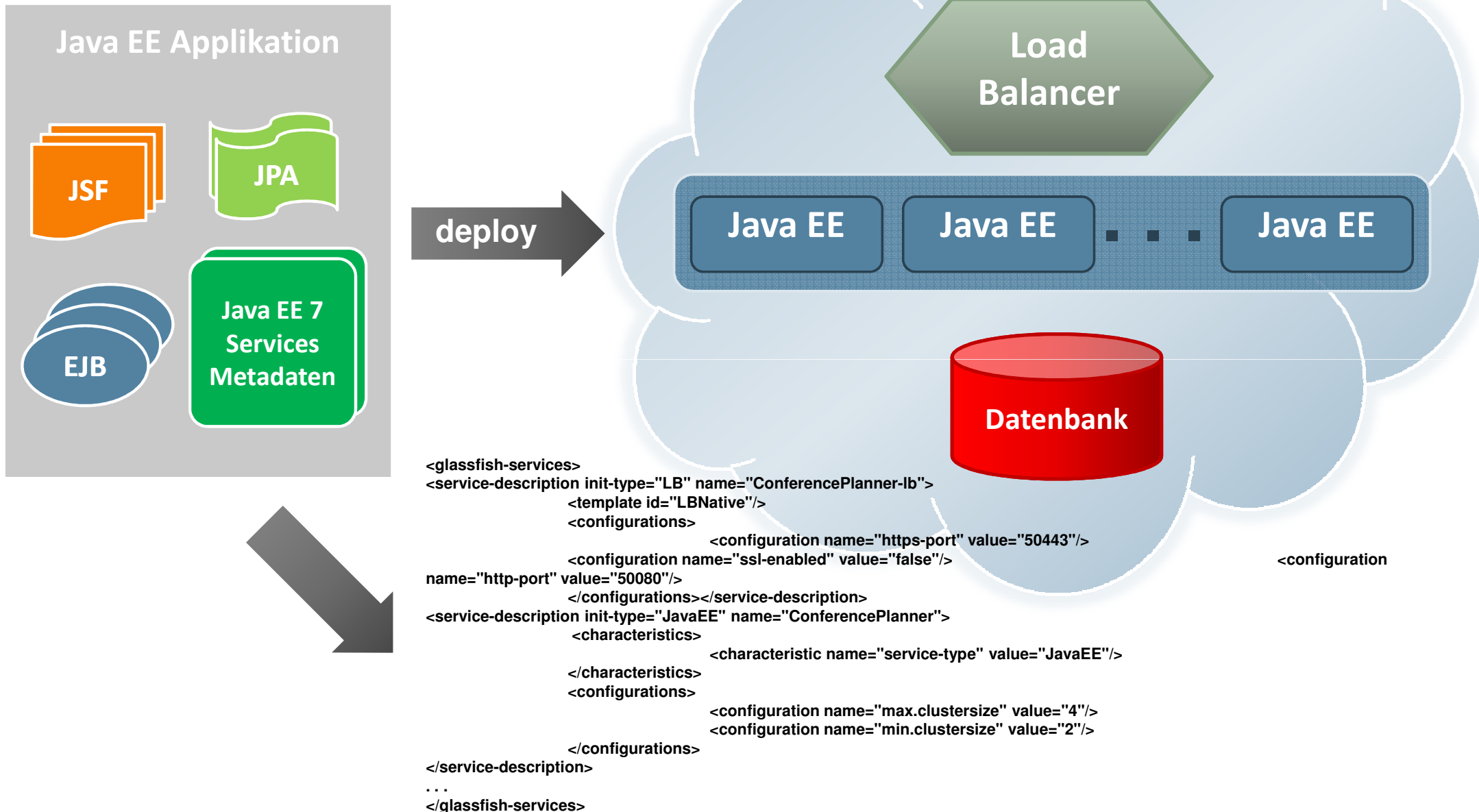
Java EE 7 – Elastizität



- **Elastisches Cluster – “Capacity On Demand”**
- **Eigenständiges Service Level Management**
 - Antwortzeit, CPU's, Speicherbedarf, Plattenverbrauch, ..
- **Deployment von “Single Machine” bis IaaS**

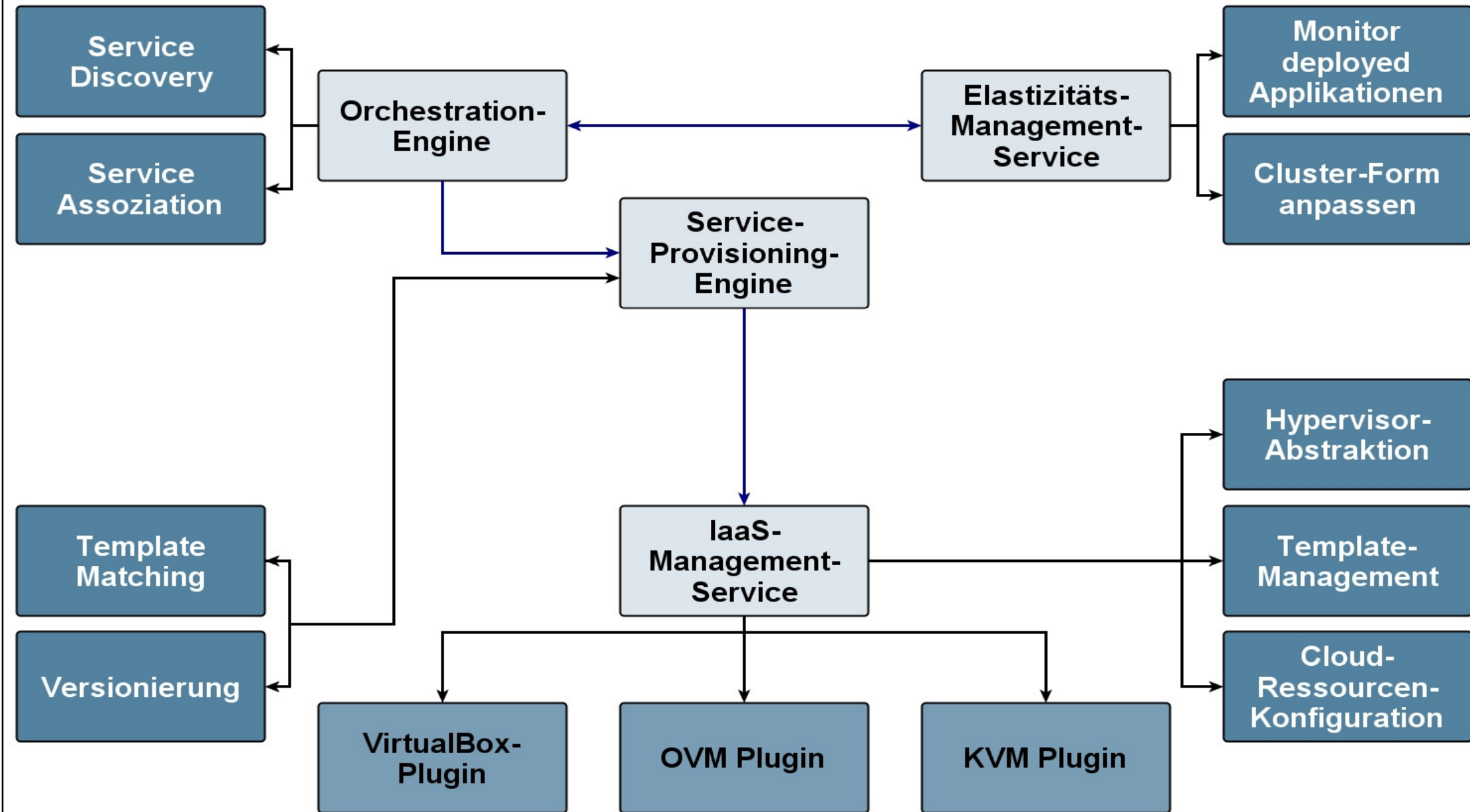
Java EE Beispiel mit GlassFish 4

<http://glassfish.org>



PaaS Enablement mit GlassFish

<http://glassfish.org>



Java EE 7 und Java EE 8 – Themenschwerpunkte

PaaS

- Provisioning
- Elastizität & eigenständige Skalierbarkeit
- Mandantenfähigkeit

Modularität *

- Building on Jigsaw
- Fokus auf OSGi interop
- Supporting Profiles & Modulare Applikationen

HTML5

- Emerging Web Standards erfordern ein Programmiermodell
- JSON, WebSockets, off-line, APIs & DOM

* Aligning with delivery schedules of Jigsaw in Java SE 8

Java EE 7 – Geplante Inhalte

Thema	Beschreibung/Inhalt
PaaS Enablement	• Service Definitions and Provisioning to enable Java as Platform as a Service • Enable Multi-tenancy in APIs
Web Profile	• Provide popular additions to the Web Profile including JAX RS 2.0 Support
JMS 2.0	• Simplify the programing model for building messaging based applications • Dependency Injection support
CDI	• Tighter Integration with JSF • Expand scope of container managed transactions • Expanded service metadata and improved configuration
Caching	• Provide APIs for accessing caching systems
Concurrency Utilities	• Support for Java concurrency APIs within the container
Pruning	• Allow vendors to optionally support older APIs • EJB CMP/BMP, JAX-RPC
Open Source and Transparency	• Open development under project GlassFish on java.net • Java EE 7 JSRs run in open with publicly viewable Expert Group mail archive

Java EE 7 JSR's

javaee-spec.java.net

- **Java EE Plattform 7 / Web Profile 7**
- Java Persistence API JPA 2.1 (JSR-338)
- Java API for RESTful Web Services JAX-RS 2.0
- JavaServer Faces JSF 2.2 (JSR-344)
- Servlet 3.1 (JSR-340)
- Enterprise JavaBeans EJB 3.2 (JSR-345)
- JavaServer Pages 2.3 MR
- Expression Language 3.0 (JSR-341)
- Java Messaging Service JMS 2.0 (JSR-343)
- Java API for XML-based Web Services JAX-WS 2.3 MR
- Dependency Injection for the Java Platform (JSR-330)
- Contexts and Dependency Injection for Java EE CDI 1.1
- Bean Validation 1.1 (JSR-349)
- Common Annotations 1.2 MR (JSR-250)
- Java Connector Architecture (JCA 1.6 mit JSR-322)
- JCache 1.0 (JSR-107)
- Concurrency Utilities 1.0 (JSR-236)
- State Management 1.0 (JSR-350)
- Batch Processing 1.0 (JSR-352)
- JTA 1.2 MR
- JASPIC 1.2 MR
- Java Web Sockets API 1.0 (JSR-356)
- JSON API 1.0 (JSR-353)

Java EE 7 JSR's

Architektursicht

CDI
Extensions

Web
Container
Extensions

JAX-RS 2.0

JSF 2.2

JSP 2.2
EL 3.0

Servlet 3.1

CDI 1.1; Common Annotations 1.2

Managed Beans

EJB 3.2

JPA 2.1

JTA 1.2

JMS 2.0

Bean Validation 1.1

JCache 1.0
(JSR-107)

Concurrency
Utilities 1.0

State
Management 1.0

Batch
Processing 1.0

JSON 1.0

Web Sockets 1.0

Zusammenfassung

- Die Java Plattform nutzt Innovationen im Ökosystem und wird sich weiterentwickeln
- Java Plattform liefert einen echten Mehrwert
- Unternehmen profitieren von existierenden Investitionen in Java EE
- Java SE 8 wird einige neue und größere Features beinhalten
- Die Ausrichtung für Java SE 9 läuft bereits



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.



Vielen Dank für Ihre Aufmerksamkeit!

Wolfgang.Weigend@oracle.com

