

12.–15.09.2010
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Enterprise Private Cloud Development

Wolfgang Weigend

Senior Leitender Systemberater Oracle Fusion Middleware

Agenda

- Cloud Computing
 - Eigenschaften und Anforderungen
 - Taxonomie und Betriebsmodelle
 - Cloud Plattform und Cloud Management (Werkzeuge)
- Private Cloud
 - Vorschlag zur Ausrichtung vom Rechenzentrum
 - Stufenweise Umsetzung und Ausblick
- Public Cloud
 - Amazon Cloud Services
 - Rackspace Referenz Architektur
- Programmiermodell für Services
 - SCA Komponente und Composite
- Zusammenfassung

Cloud Computing – Eigenschaften und geschäftliche Anforderungen

- **On Demand Self Services**
- **Netzwerk Zugriff**
- **Resource Pooling**
- **Rapid Elasticity**
- **Measured Services (Consumer & Provider)**
- **Geschäftliche Anforderungen:**
 - **Stabilität (Verfügbarkeit vom Geschäftsprozess und Cloud Computing)**
 - **Wirtschaftlichkeit (Individual-/Standard-Software und Service-Modelle)**
 - **Nachvollziehbarkeit (Berichte über Systemzustand)**

Technologie-Anforderungen von Cloud Anbietern

- **Service Level Management:** *Wie werden SLA´s und QoS garantiert, überwacht und nachvollziehbar?*
- **Skalierbarkeit:** *Wie wird Wachstum eingeplant?*
- **Leistung und Verfügbarkeit:** *Wie werden niedrige Durchlaufzeiten und geschäftliche Kontinuität sichergestellt?*
- **Anpaßbarkeit:** *Wie werden Anpassungen von Datenmodell, Prozessen und Benutzerschnittstelle erreicht?*
- **Integration:** *Wie werden bereits vorhandene Systeme (On-Premise) integriert?*
- **Sicherheit:** *Wie wird Datenverlust verhindert, oder unauthorisierter Zugriff Dritter vermieden? Wie werden regulatorische Vorgaben erfüllt?*
- **Niedrige Entwicklungs- und Wartungskosten:** *Wie kann eine zentrale Kodierungs-Basis für On-Premise und Public Cloud Betrieb benutzt werden?*

Cloud Taxonomie

Service Modelle

**Infrastruktur
IaaS**

**Plattform
PaaS**

**Software
SaaS**

Deployment Modelle

**Private
Cloud**

**Public
Cloud**

**Hybrid
Cloud**

**Community
Cloud**

Die Architekturschichten: SaaS, PaaS, IaaS

Software

- CRM On Demand
- Personalwesen
- Business Intelligence
- Transport Mgmt.
- Bedarfsplanung
- Product Life Cycle Mgmt.
- Universal Content Mgmt.
- Governance, Risk & Compliance

Plattform

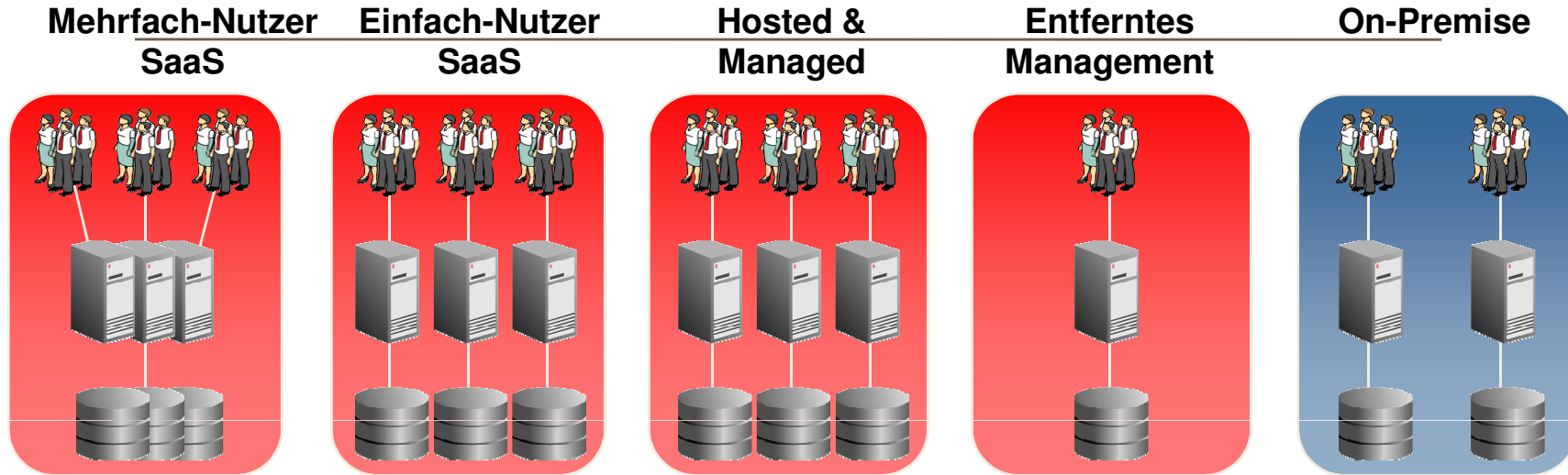
- Datenbank
- Middleware

Infrastruktur

- Bereitstellung von Infrastruktur und Management Services
- Unterstützung von Kunden beim Betrieb und Management ihrer Privaten Clouds



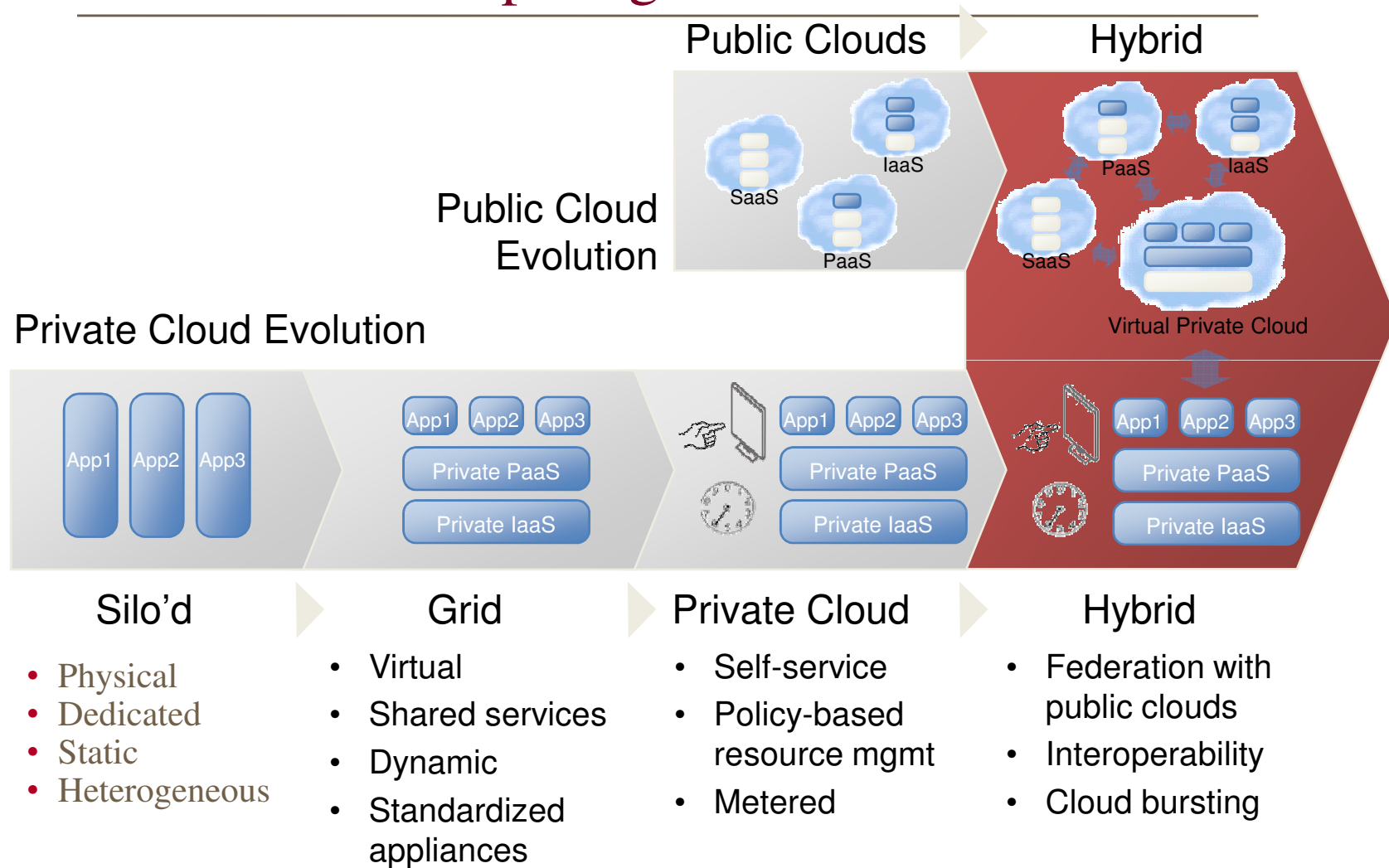
Flexible Betriebs-Optionen



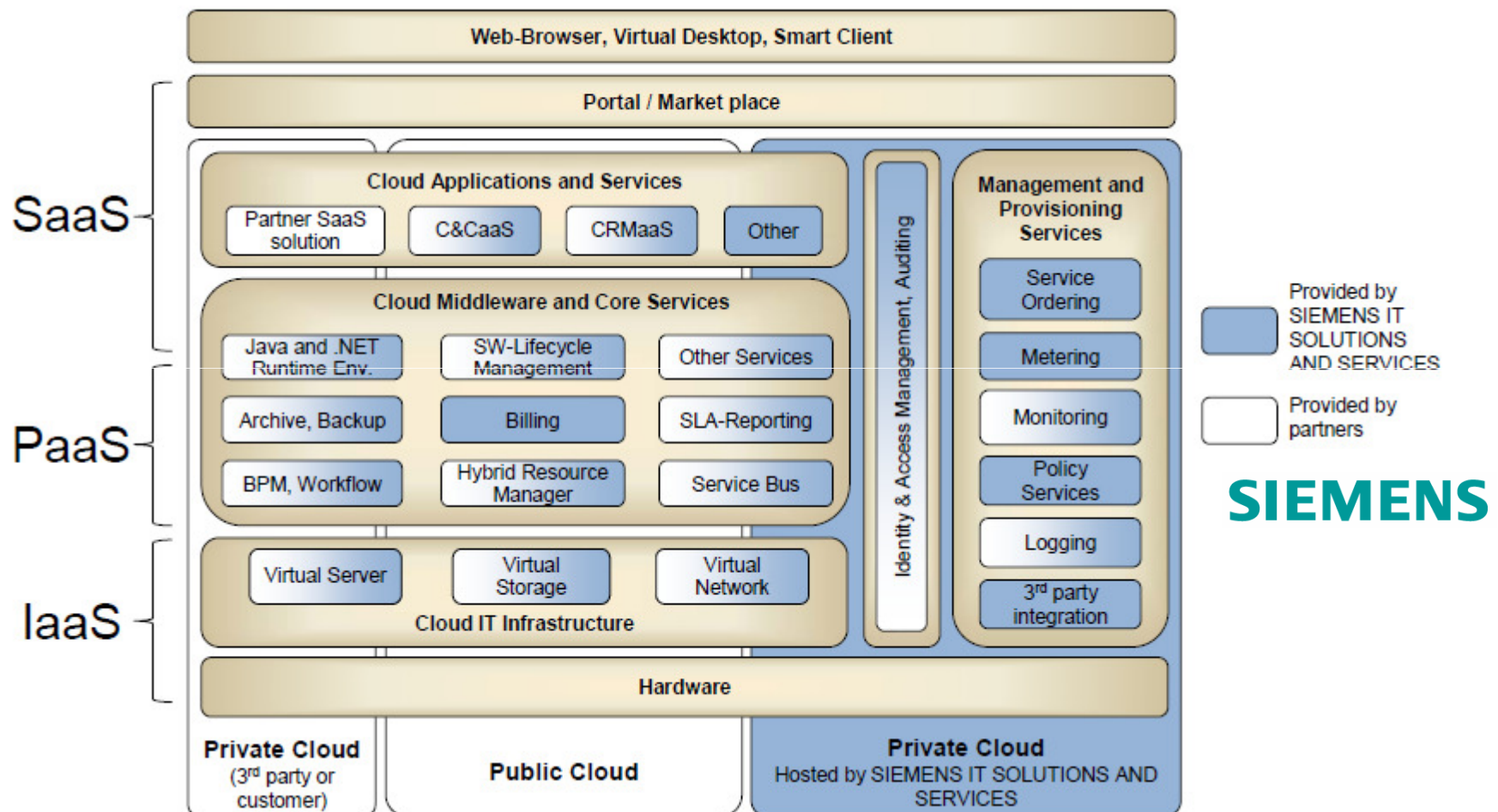
Public	Private
Zahlung-pro-Nutzung	Lizenziert
OpEx	CapEx & OpEx
Off-Premise	On-Premise
Management durch den Hersteller	Management durch den Kunden
Hersteller durchgeführte Wartung	Vom Kunden durchgeführte Wartung

* OpEx Betriebskosten als Ausgaben eines Unternehmens um einen operativen Geschäftsbetrieb sicherzustellen
 * CapEx Investitionsausgaben eines Unternehmens für längerfristige Anlagegüter

Unternehmensweiter evolutionärer Ansatz zum Cloud Computing

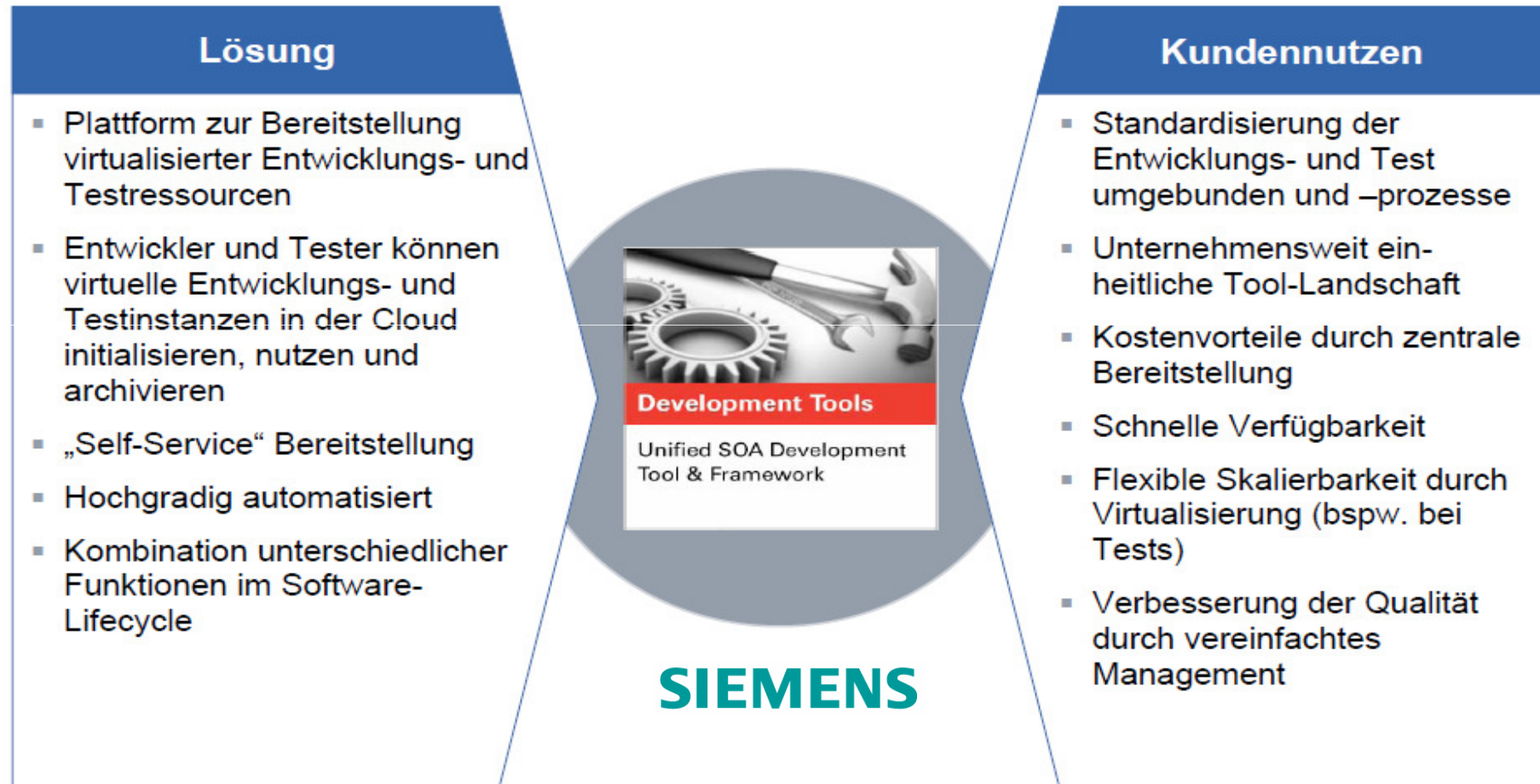


Eine hybride Cloud Architektur ist die Plattform für Community Clouds



„Oracle bietet einen der vollständigsten PaaS Stacks (Middleware und Entwicklungswerkzeuge) für Private Cloud Umgebungen.“

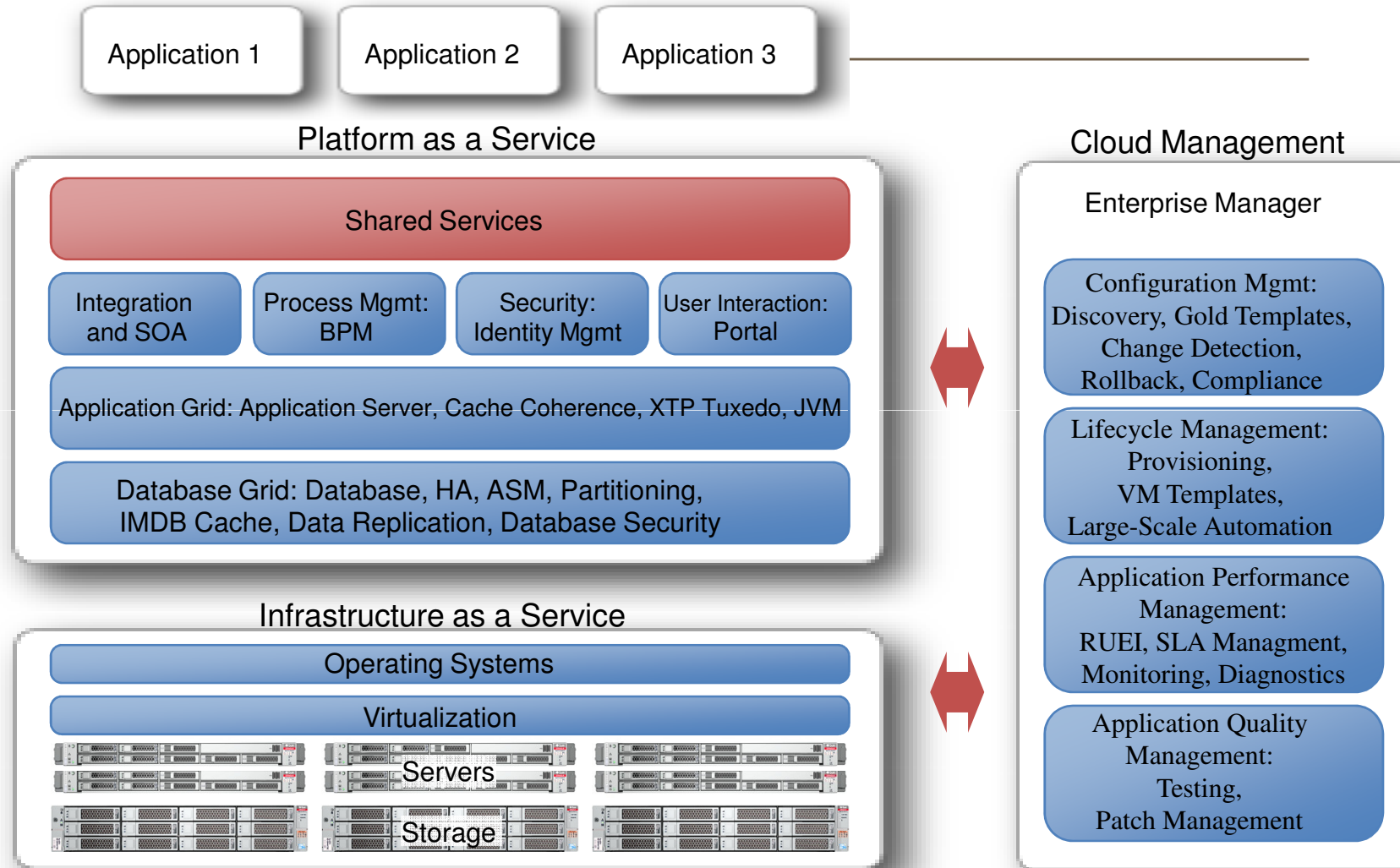
Gemeinsam mit Oracle baut Siemens ein Angebot für eine Test- und Development-Cloud Umgebung auf



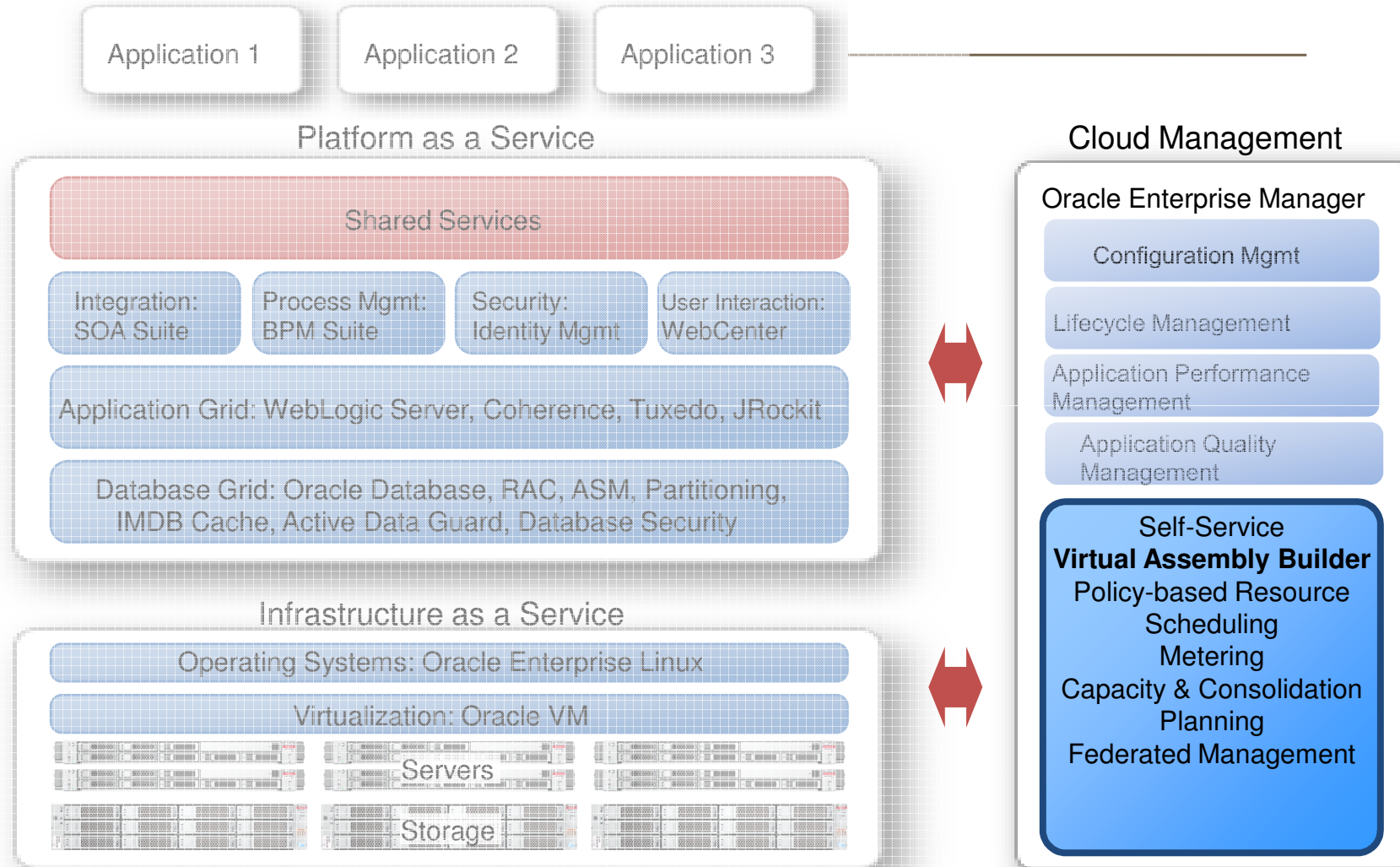
Quelle: Siemens IT Solutions and Services

© Siemens AG 2010. Alle Rechte vorbehalten.

Cloud Platform & Cloud Management



Cloud Management & Automatisierung



Private Cloud evolve Dev/Test provisioning

Virtual image distribution behind the firewall

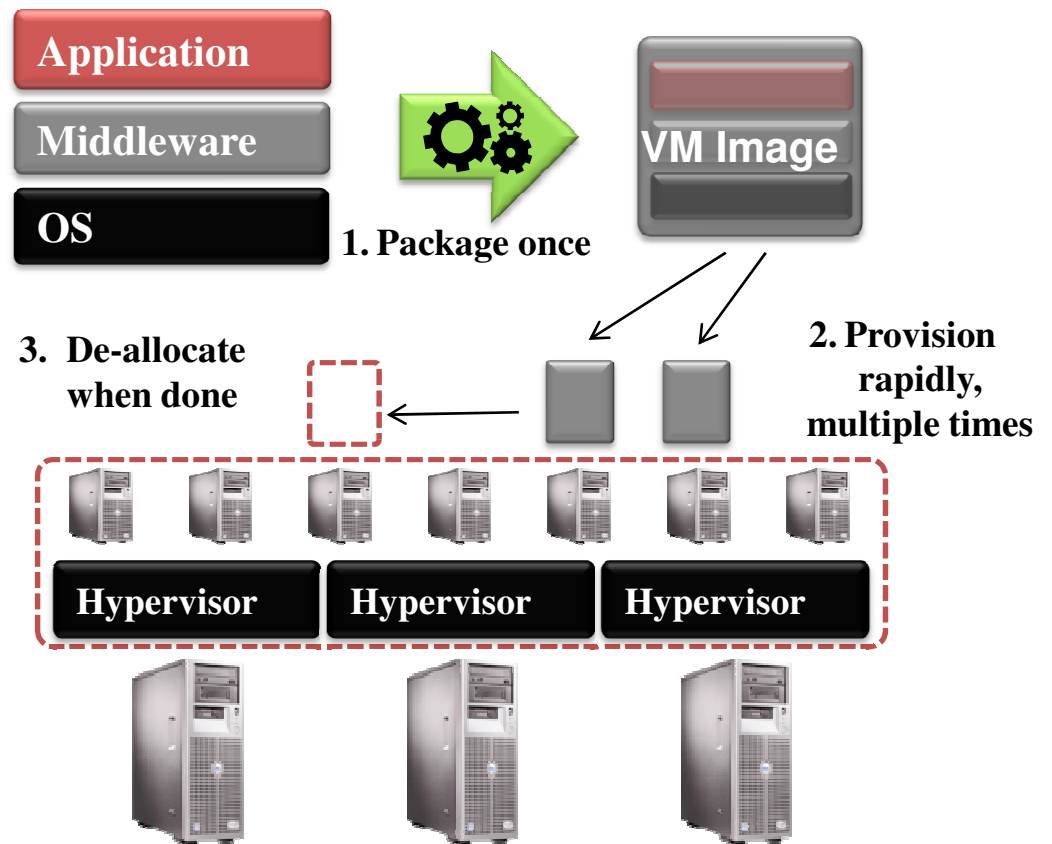
Traditional Dev/Test provisioning

For every dev/test environment:

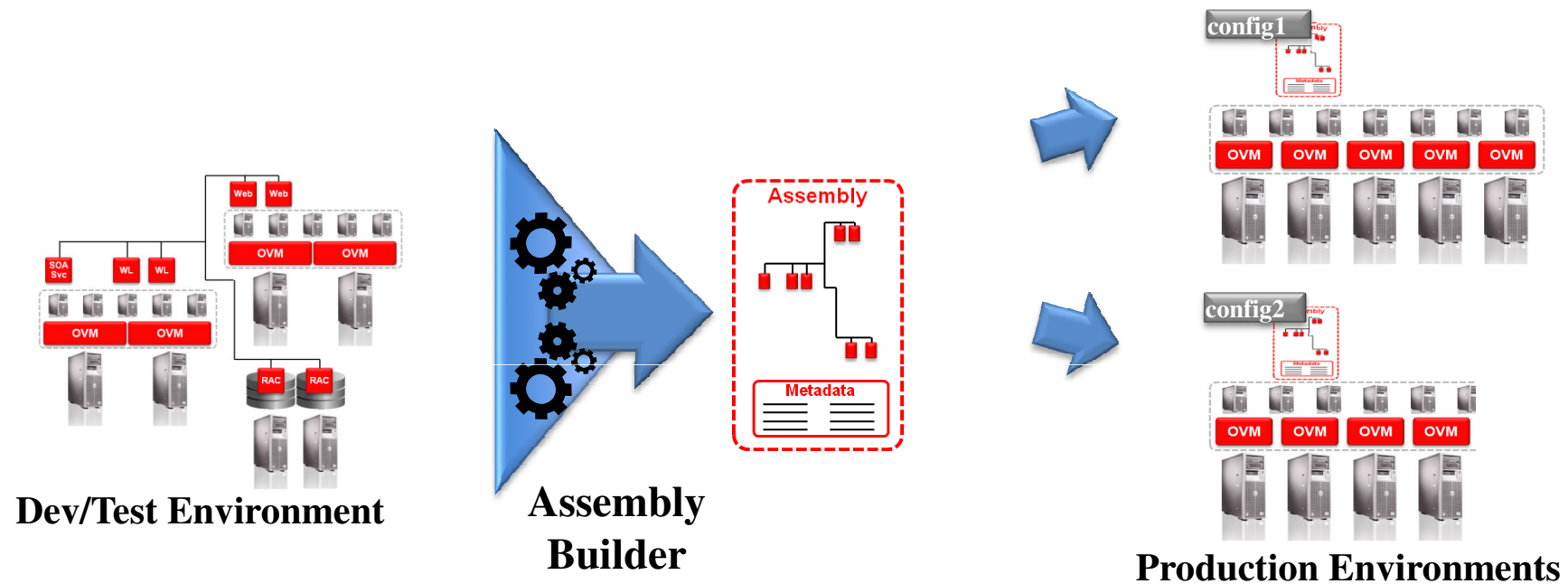
1. Procure and configure hardware
2. Install and configure OS
3. Install and configure dev (or test) environment



Provisioning with Virtual Machine Images



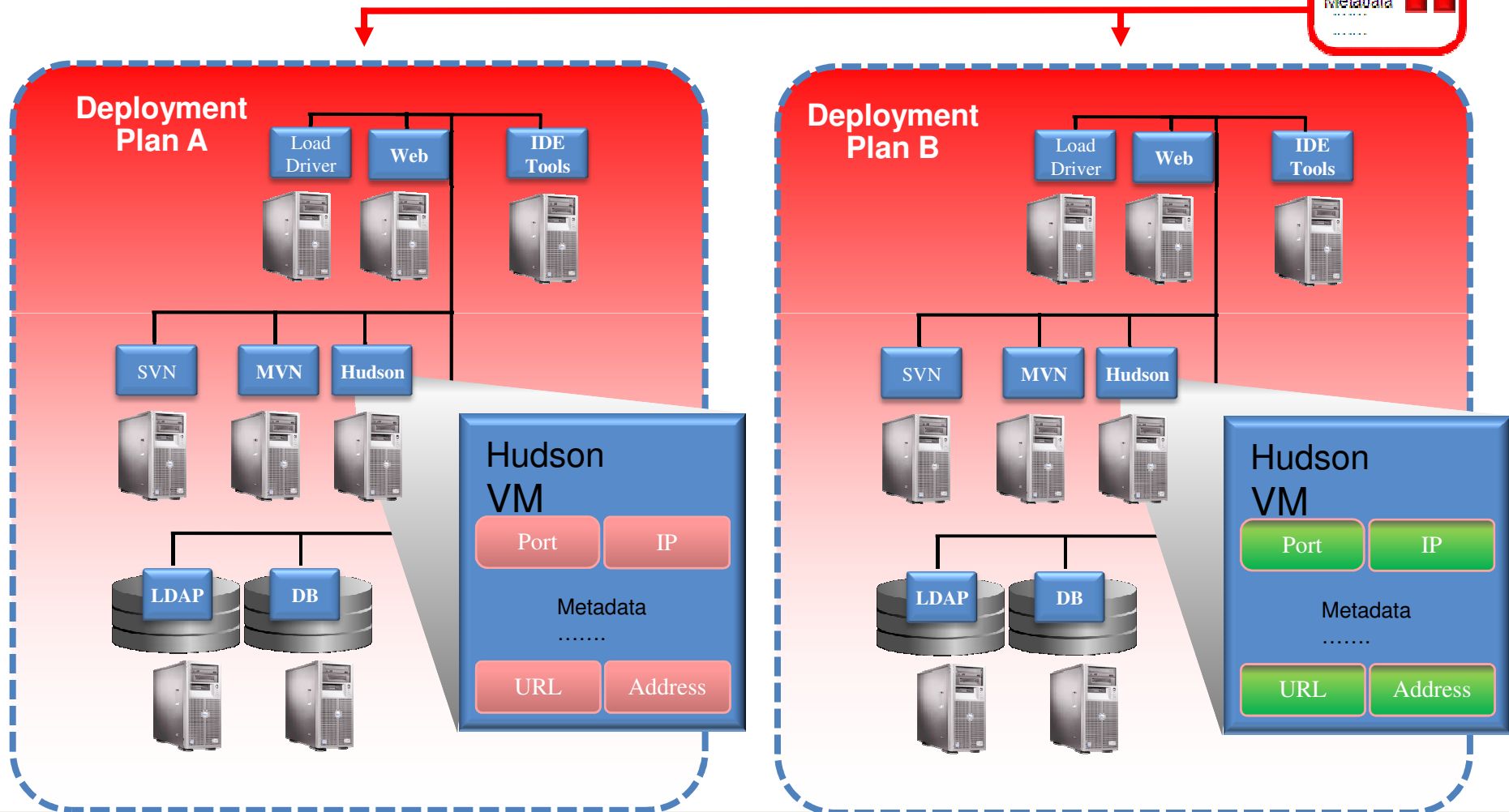
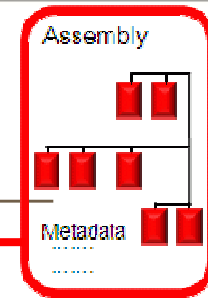
The Power of Assemblies



- **Package** up complex structure from dev/test and reconstitute in production
- **Minimize setup time and risk** of hard-to-debug configuration errors
- **Easily replicate** in production with minor variations
- Each production instance has **well-contained configuration** parameters for flexibility

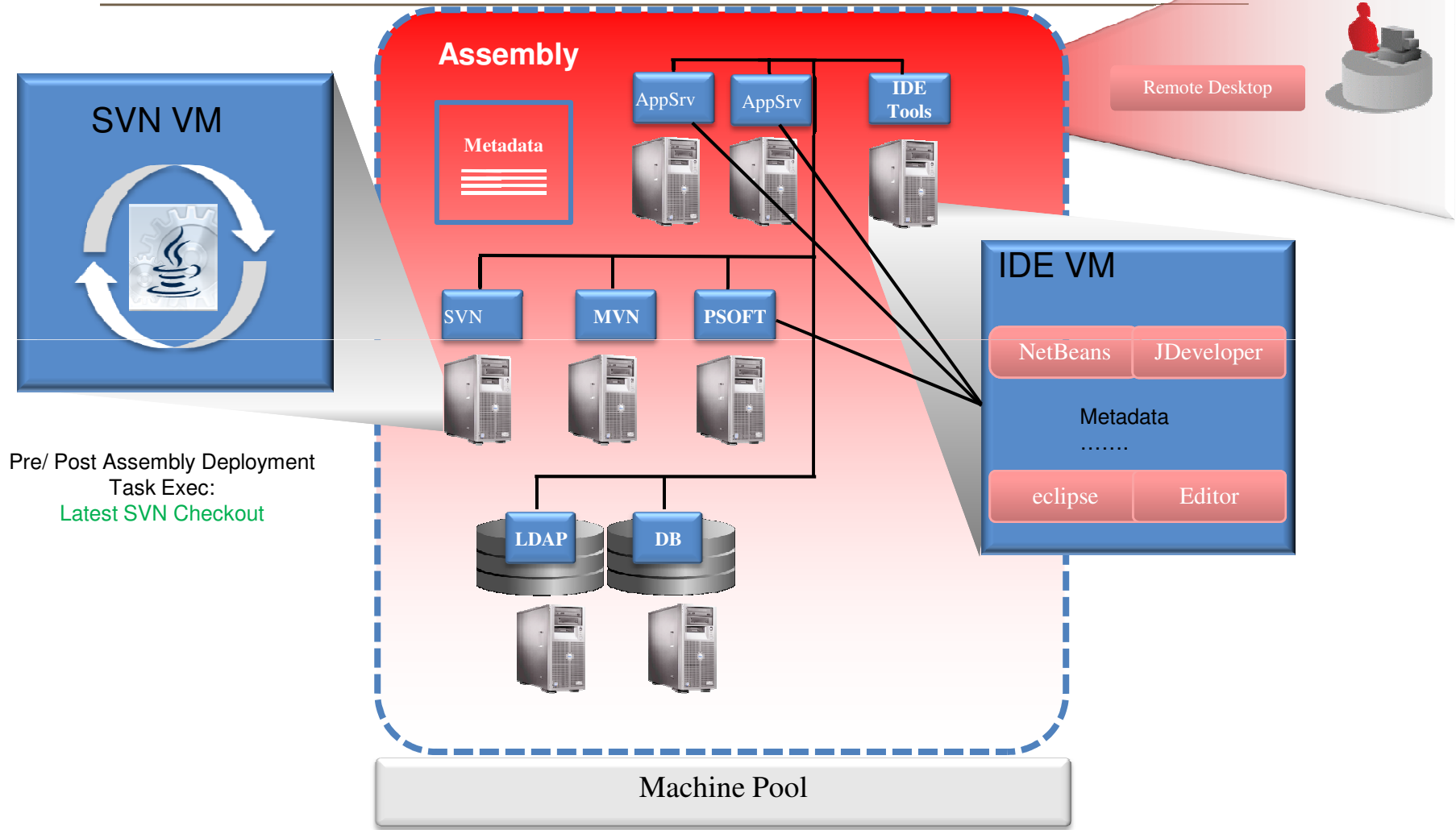
Eliminate environment mismatch errors

User-specified & dynamic late binding parameters



Remote Development Sandboxes

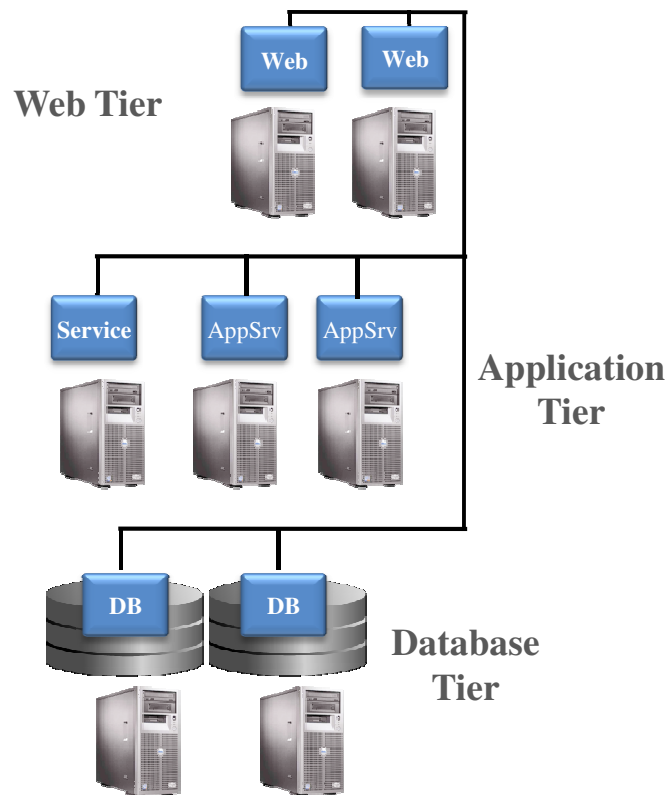
Check out new Assembly (e.g. daily), then refresh



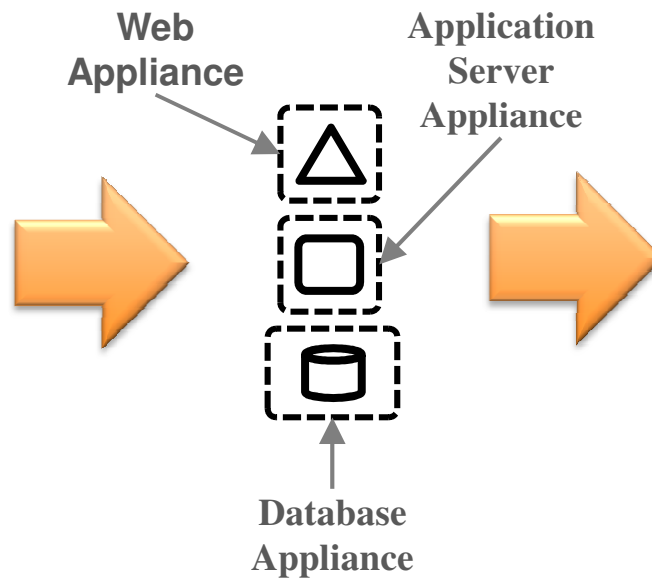
Private PaaS Deployment with OVAB

Deploy complete VM assemblies

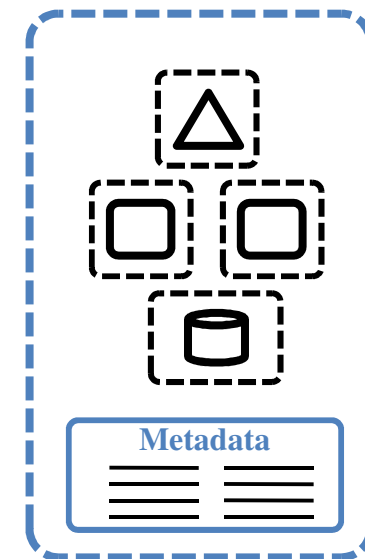
Reference System



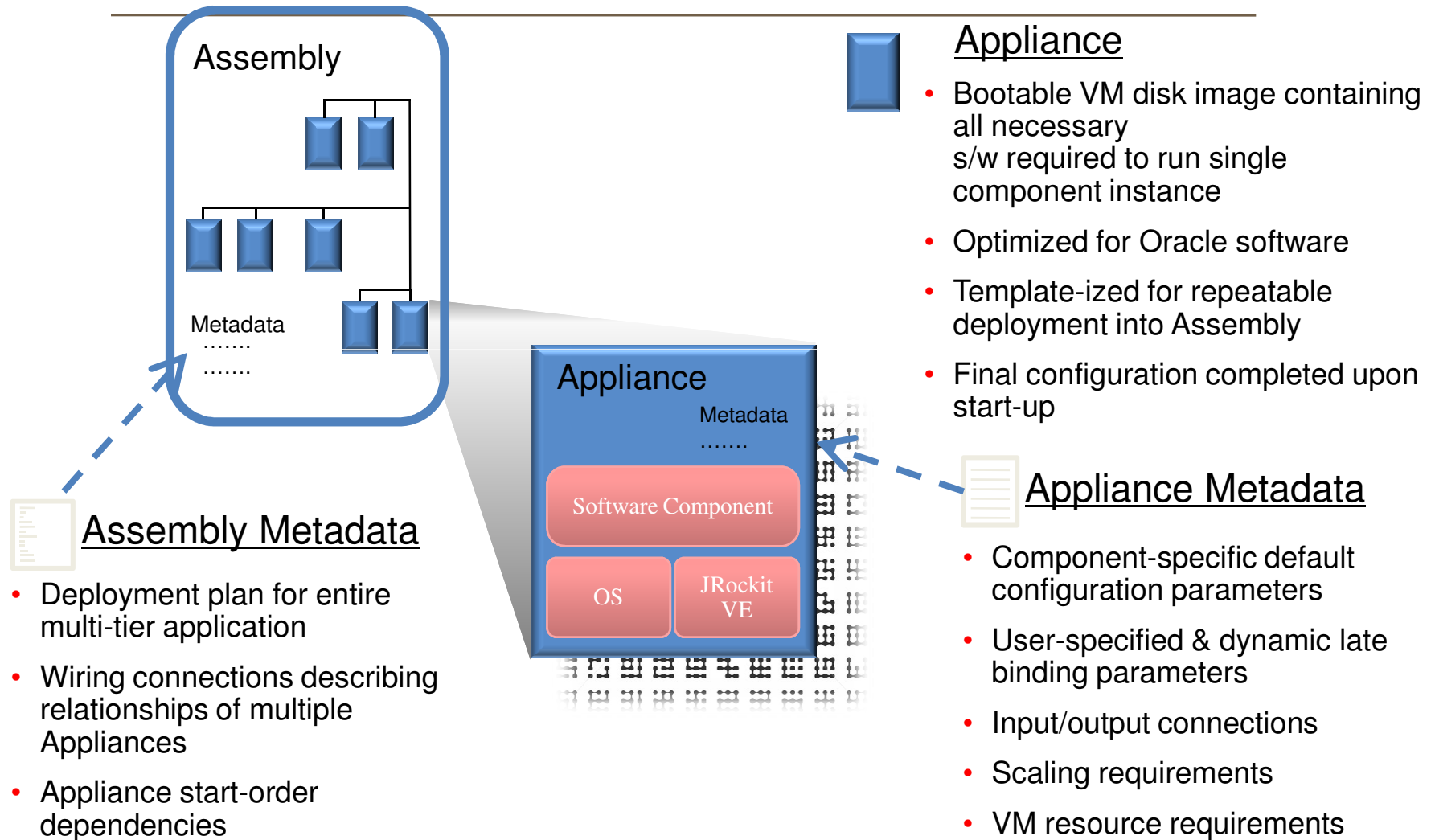
Virtualized Software Appliances



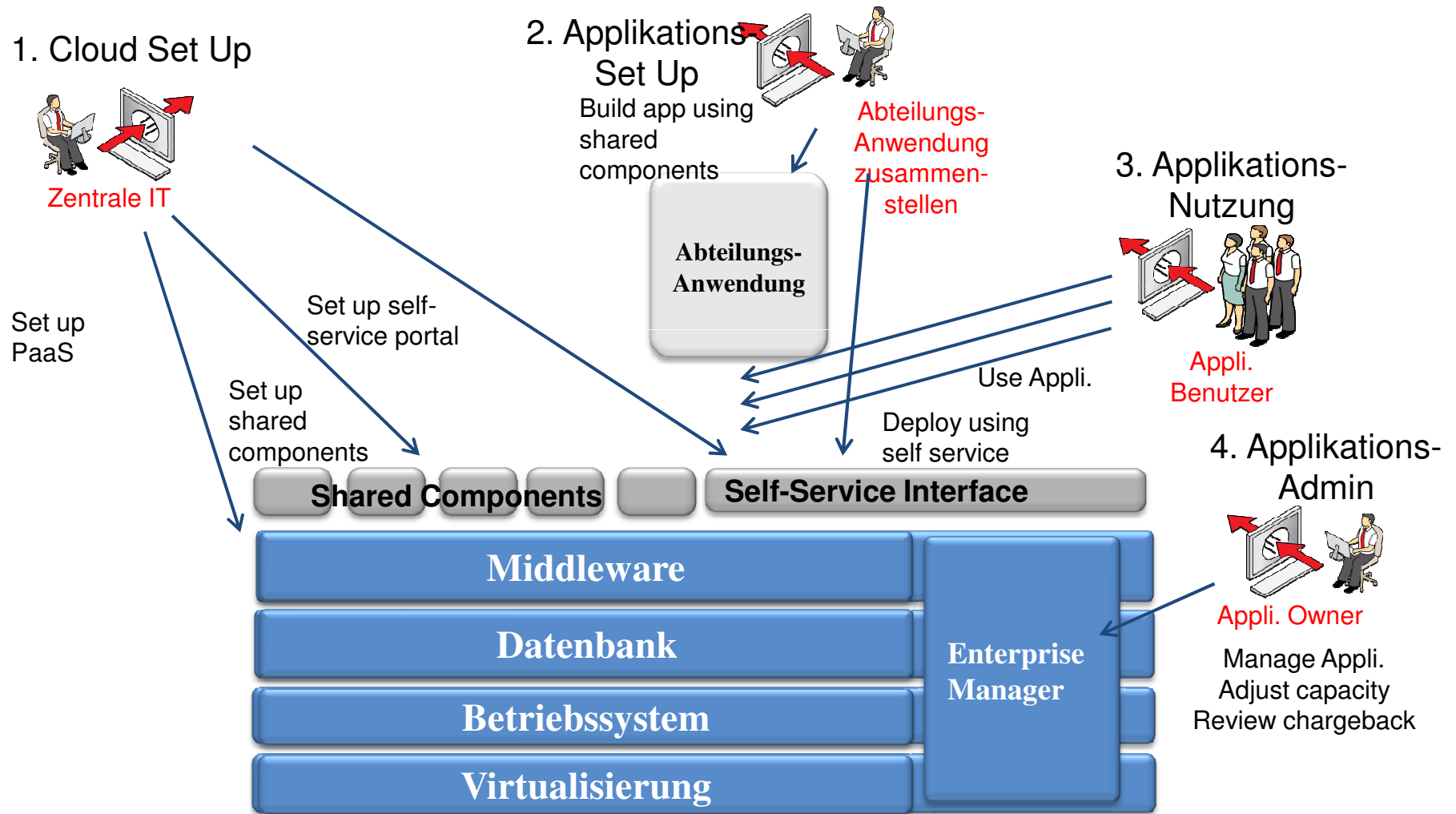
Assembly



Assembly Structure

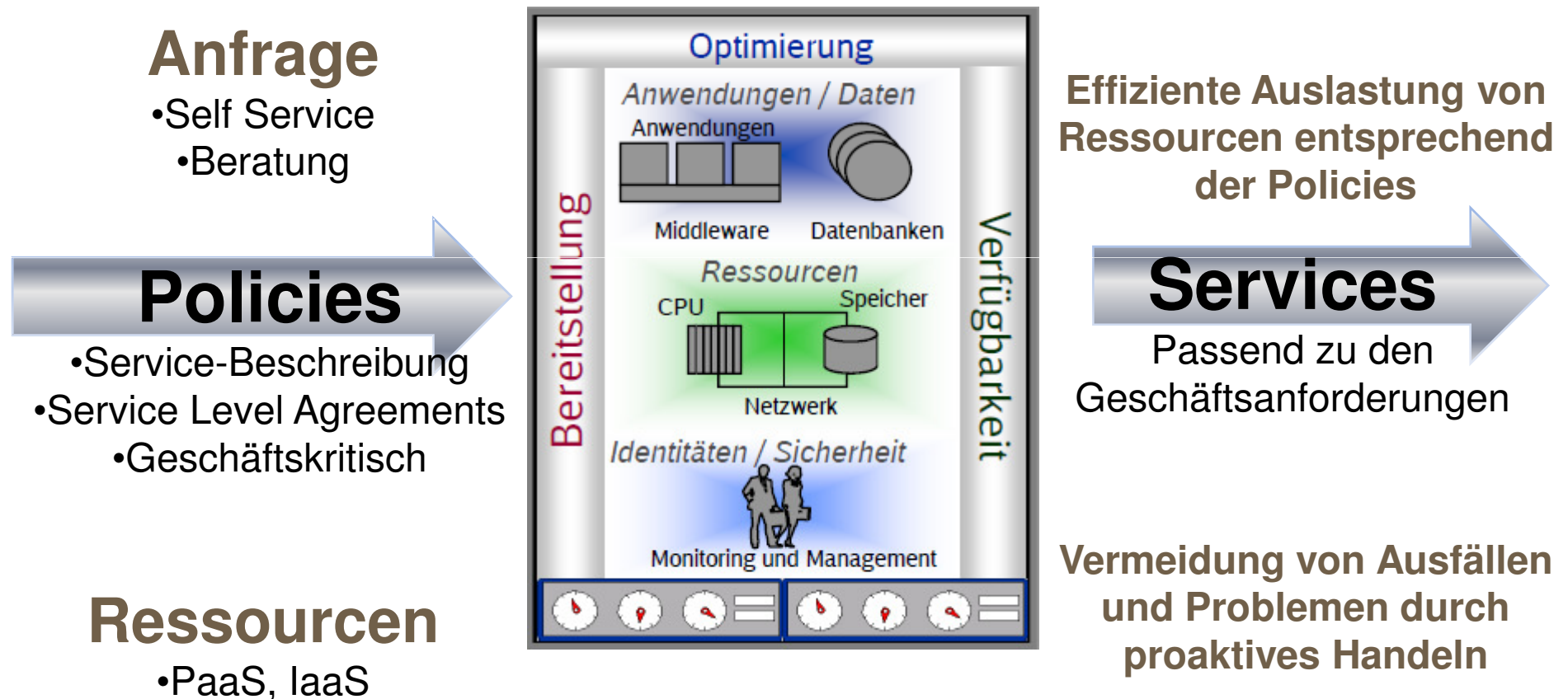


Private PaaS Lebenszyklus



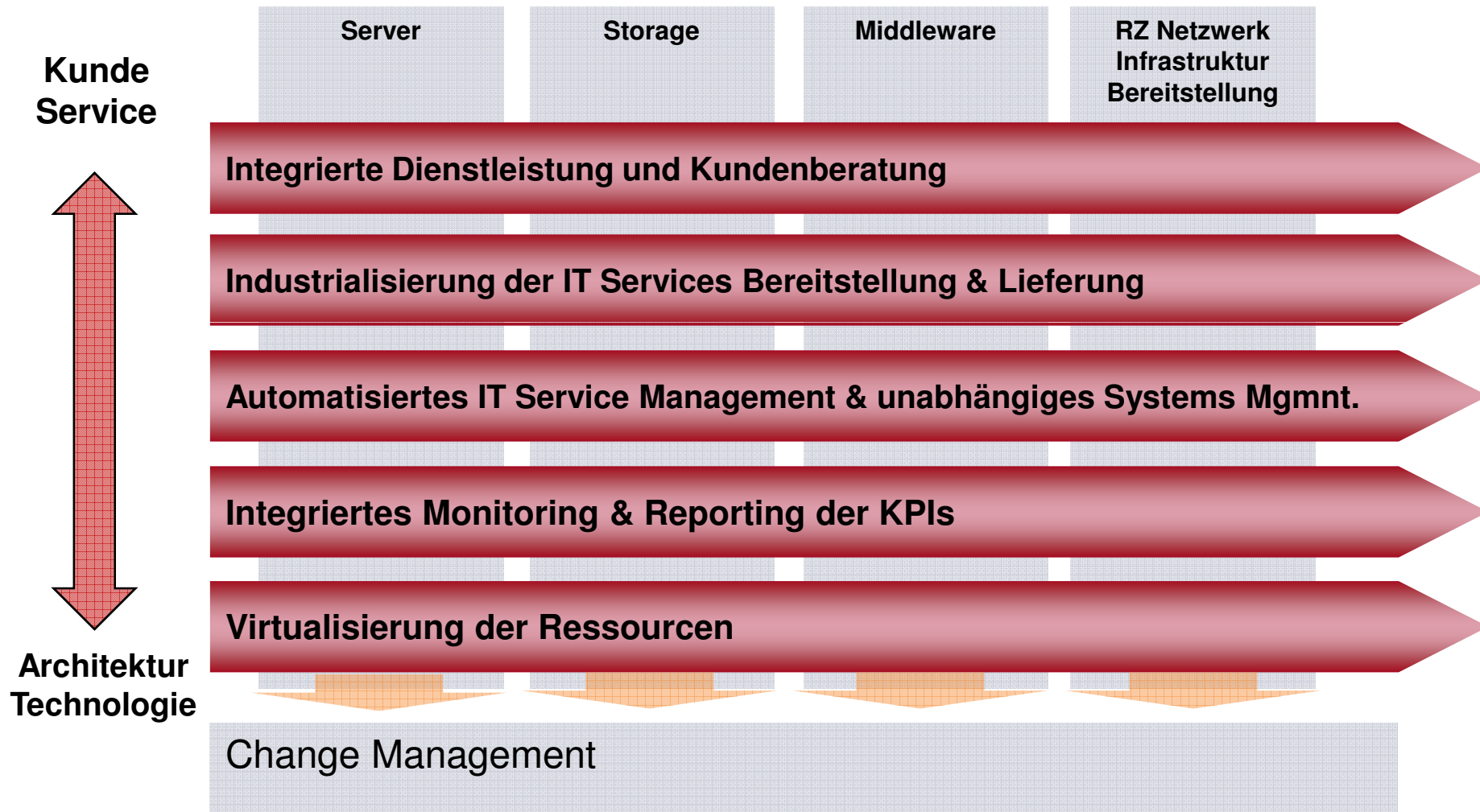
Private Cloud Ansatz

Das Rechenzentrum als proaktives System – Computing für die dynamische Bereitstellung von internationalen IT Services



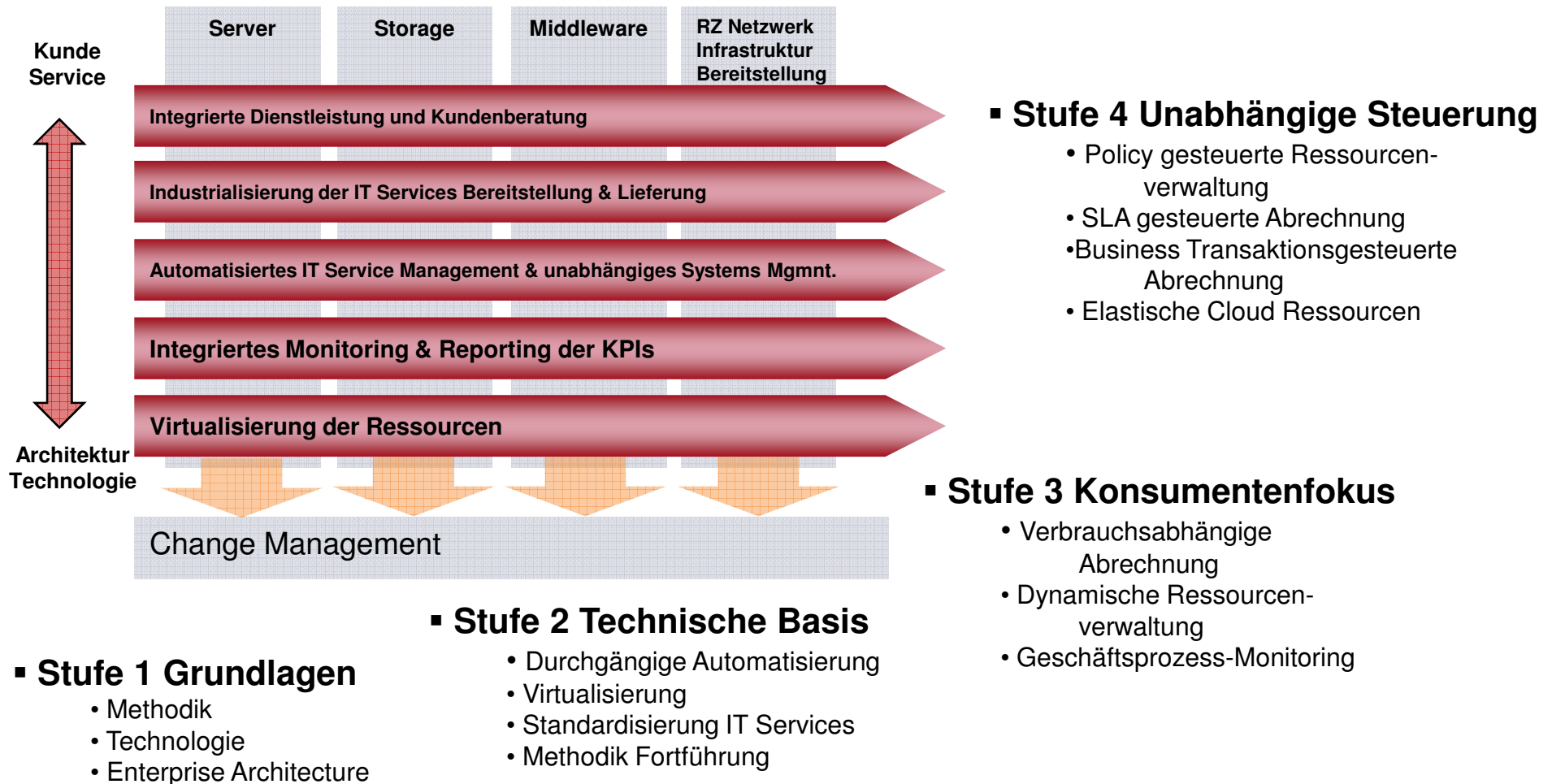
Private Cloud – Stufenweise Umsetzung (1)

Vorschlag zur künftige Ausrichtung vom Rechenzentrum



Private Cloud – Stufenweise Umsetzung (2)

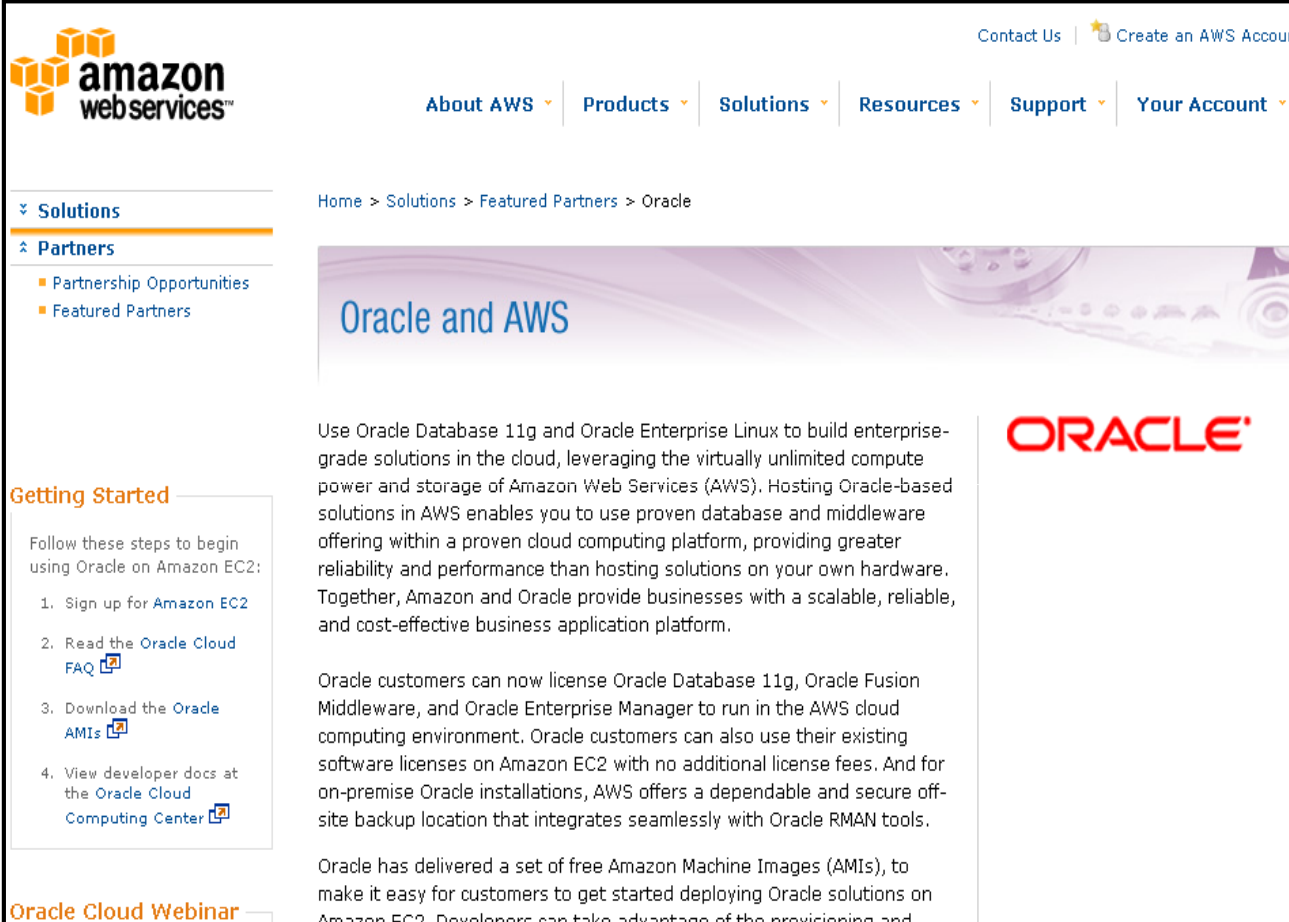
Vorschlag zu Inhalten und Kernthemen der einzelnen Stufen



Ausblick: Zukünftige Ausrichtung der Private Cloud

- *Übergang von bisherigen Anwendungen und existierenden Produkten zu Services im Betrieb mit dem Zweck der besseren Auslastung von Ressourcen*
- *Bereitstellung von internen dynamischen Entwicklungsumgebungen innerhalb der Private Cloud*
- *Automatischer und gesicherter Zugriff auf Ressourcen über Service Levels*
- *Automatische Nutzung von verteilten Ressourcen und Services über Börsen-ähnliche Mechanismen mittels Broker-Ansatz*
- *Künftiger Marktplatz für optimales Preis-Leistungsverhältnis von Services und deren Service Level Vereinbarungen*

Public Cloud Angebot von Amazon

amazon web services™

Contact Us | Create an AWS Account

About AWS | Products | Solutions | Resources | Support | Your Account

Home > Solutions > Featured Partners > Oracle

Oracle and AWS

Use Oracle Database 11g and Oracle Enterprise Linux to build enterprise-grade solutions in the cloud, leveraging the virtually unlimited compute power and storage of Amazon Web Services (AWS). Hosting Oracle-based solutions in AWS enables you to use proven database and middleware offering within a proven cloud computing platform, providing greater reliability and performance than hosting solutions on your own hardware. Together, Amazon and Oracle provide businesses with a scalable, reliable, and cost-effective business application platform.

Oracle customers can now license Oracle Database 11g, Oracle Fusion Middleware, and Oracle Enterprise Manager to run in the AWS cloud computing environment. Oracle customers can also use their existing software licenses on Amazon EC2 with no additional license fees. And for on-premise Oracle installations, AWS offers a dependable and secure off-site backup location that integrates seamlessly with Oracle RMAN tools.

Oracle has delivered a set of free Amazon Machine Images (AMIs), to make it easy for customers to get started deploying Oracle solutions on Amazon EC2. Developers can take advantage of the provisioning and

Getting Started

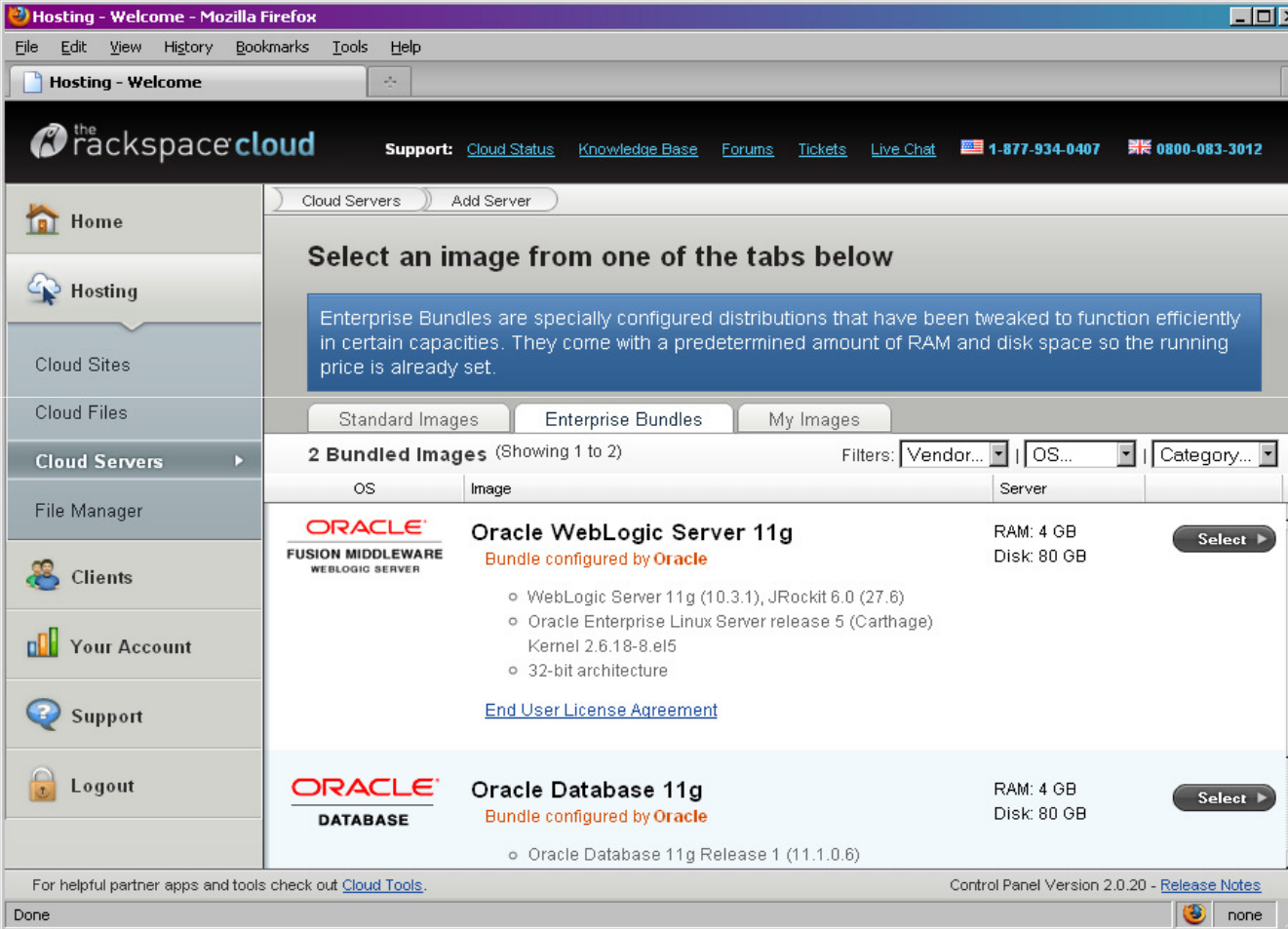
Follow these steps to begin using Oracle on Amazon EC2:

1. Sign up for [Amazon EC2](#)
2. Read the [Oracle Cloud FAQ](#)
3. Download the [Oracle AMIs](#)
4. View developer docs at the [Oracle Cloud Computing Center](#)

Oracle Cloud Webinar

- **Datenbank, Middleware, Enterprise Manager im Einsatz mit Elastic Compute Cloud (EC2)**
- **Amazon Machine Images (AMIs)**
- **Datenbank Secure Backup für Simple Storage Service (S3) und Amazon Block Storage (Mounted Filesystem)**

Public Cloud Hosting Technologie



Hosting - Welcome - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Hosting - Welcome

the **rackspace** cloud Support: [Cloud Status](#) [Knowledge Base](#) [Forums](#) [Tickets](#) [Live Chat](#) 1-877-934-0407 0800-083-3012

Home

Hosting

Cloud Sites

Cloud Files

Cloud Servers

File Manager

Clients

Your Account

Support

Logout

Cloud Servers Add Server

Select an image from one of the tabs below

Enterprise Bundles are specially configured distributions that have been tweaked to function efficiently in certain capacities. They come with a predetermined amount of RAM and disk space so the running price is already set.

Standard Images Enterprise Bundles My Images

2 Bundled Images (Showing 1 to 2) Filters: Vendor... | OS... | Category...

OS	Image	Server	
ORACLE FUSION MIDDLEWARE WEBLOGIC SERVER	Oracle WebLogic Server 11g Bundle configured by Oracle	RAM: 4 GB Disk: 80 GB	Select
	<ul style="list-style-type: none"> WebLogic Server 11g (10.3.1), JRockit 6.0 (27.6) Oracle Enterprise Linux Server release 5 (Carthage) Kernel 2.6.18-8.el5 32-bit architecture End User License Agreement		
ORACLE DATABASE	Oracle Database 11g Bundle configured by Oracle	RAM: 4 GB Disk: 80 GB	Select
	<ul style="list-style-type: none"> Oracle Database 11g Release 1 (11.1.0.6) 		

For helpful partner apps and tools check out [Cloud Tools](#). Control Panel Version 2.0.20 - [Release Notes](#)

Done

Self-service Public PaaS basierend auf Oracle VM, Oracle Enterprise Linux, Oracle Datenbank und Oracle WebLogic Server

Rackspace: Public PaaS verwendet Oracle Technologie

Anwendungsbeispiel

- Standardisierung mit OracleVM für Server Virtualisierung
- Implementiert PaaS mit der WebLogic Suite und der Oracle Datenbank

Umgebung

- Individuell erstellte Java-Anwendungen (Java EE)

Probleme

- Hosting-Anbieter – Derzeit wird Web-Applikations-Hosting nur mit Tomcat und MySQL angeboten
- Kunden fragen nach unternehmensweiter Infrastruktur (Enterprise Segment)

Technische Lösungskomponenten

- Oracle Produkt Portfolio – OracleVM, WebLogic Suite und Oracle Real Application Cluster (RAC)

Kernmerkmale der Komponenten

- Automatisierte Verteilung für die Middleware-Schicht und die Datenschicht
- Dynamische stufenlose Skalierbarkeit nach Bedarf für starke und schwache Belastung
- Komplettes Management
- Applikations Performance Management

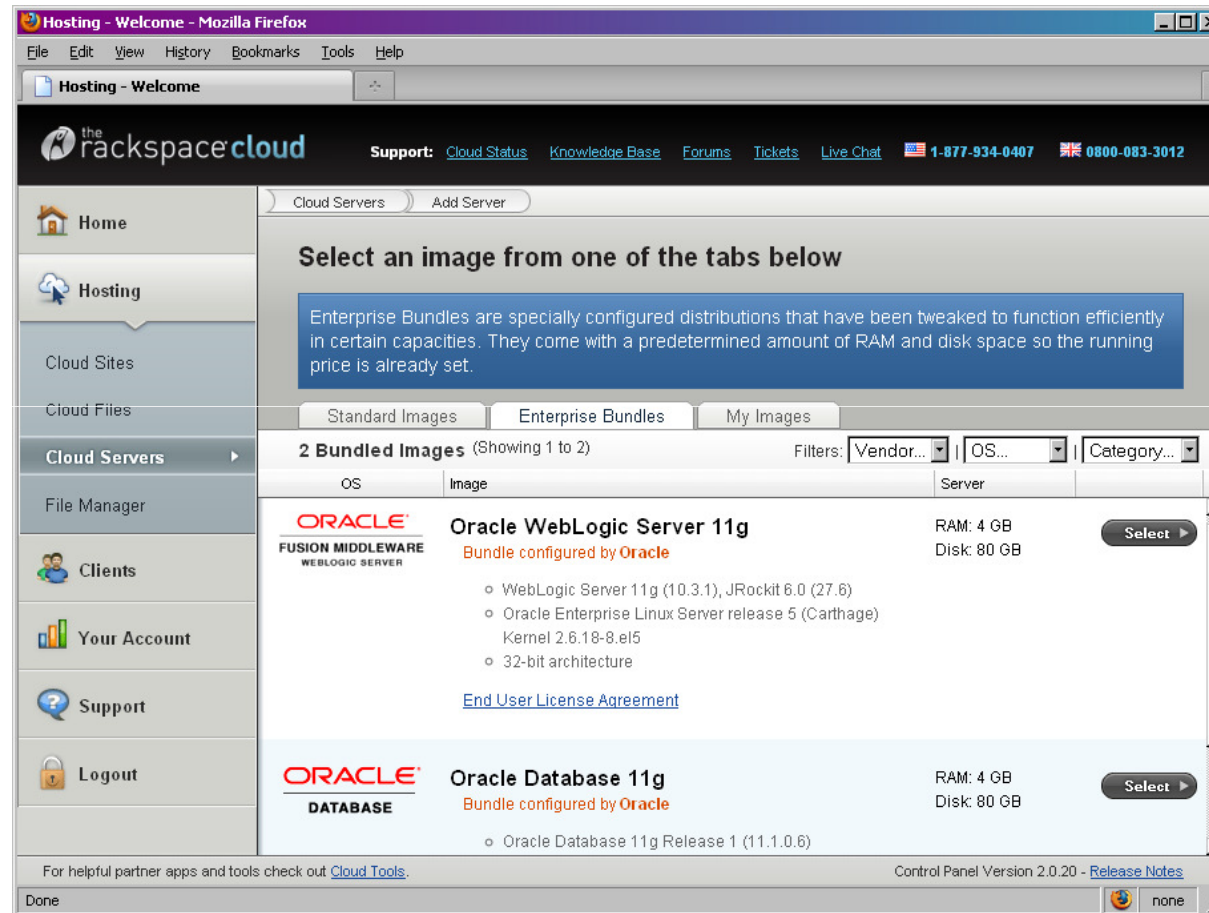
Kundenvorteile

- IaaS und PaaS Angebot
- Unternehmensweite Lösung (Enterprise Segment)
- Starke Marktpräsenz und Durchdringung

Rackspace's Hauptanforderungen

- Wiederholbaren Erfolg schaffen
- Aufbau einer Mehrfach-Nutzer Umgebung
- Plattform Lebenszyklus Management
- Skalierbarkeit nach Bedarf "On-demand"
- Plattform Stabilität
- Sicherheit der Plattform gewährleisten
- Messen und Monitoring
- Self-Service-Einrichtung etablieren
- Informationen zugänglich gestalten
- Management und Kontrolle
- Einheitliche Entwicklungs-, Test- und Betriebs-Umgebung

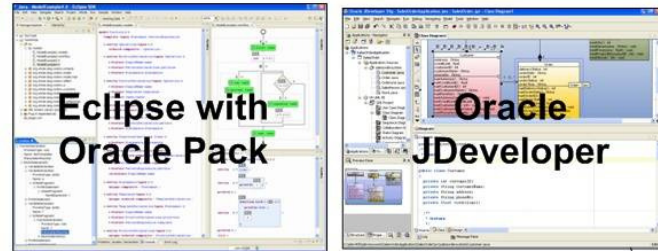
Rackspace Self-Services Public PaaS



Rackspace Cloud Sites einfach und schnell in zwei Schritten nutzen

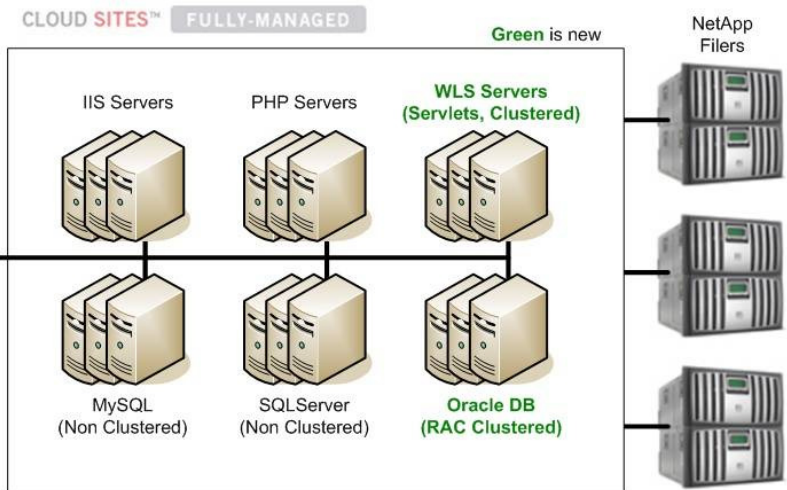
Rackspace's Cloud Angebot

Eclipse and Jdeveloper will be able to publish projects to the Rackspace Cloud Sites Infrastructure



Oracle recommends setting up the cloud sites admin infrastructure up so that the end user has as much control over their WLS Domain as possible without giving access to other accounts.

Public Internet



Single Instance of Oracle Enterprise Manager to manage the Cloud Site pieces. There will be a Rackspace super user account that can control all the DBs and WLS in the Cloud Site infrastructure. Initially, this is for Rackspace use only.



CLOUD SERVERS™



Shared Storage



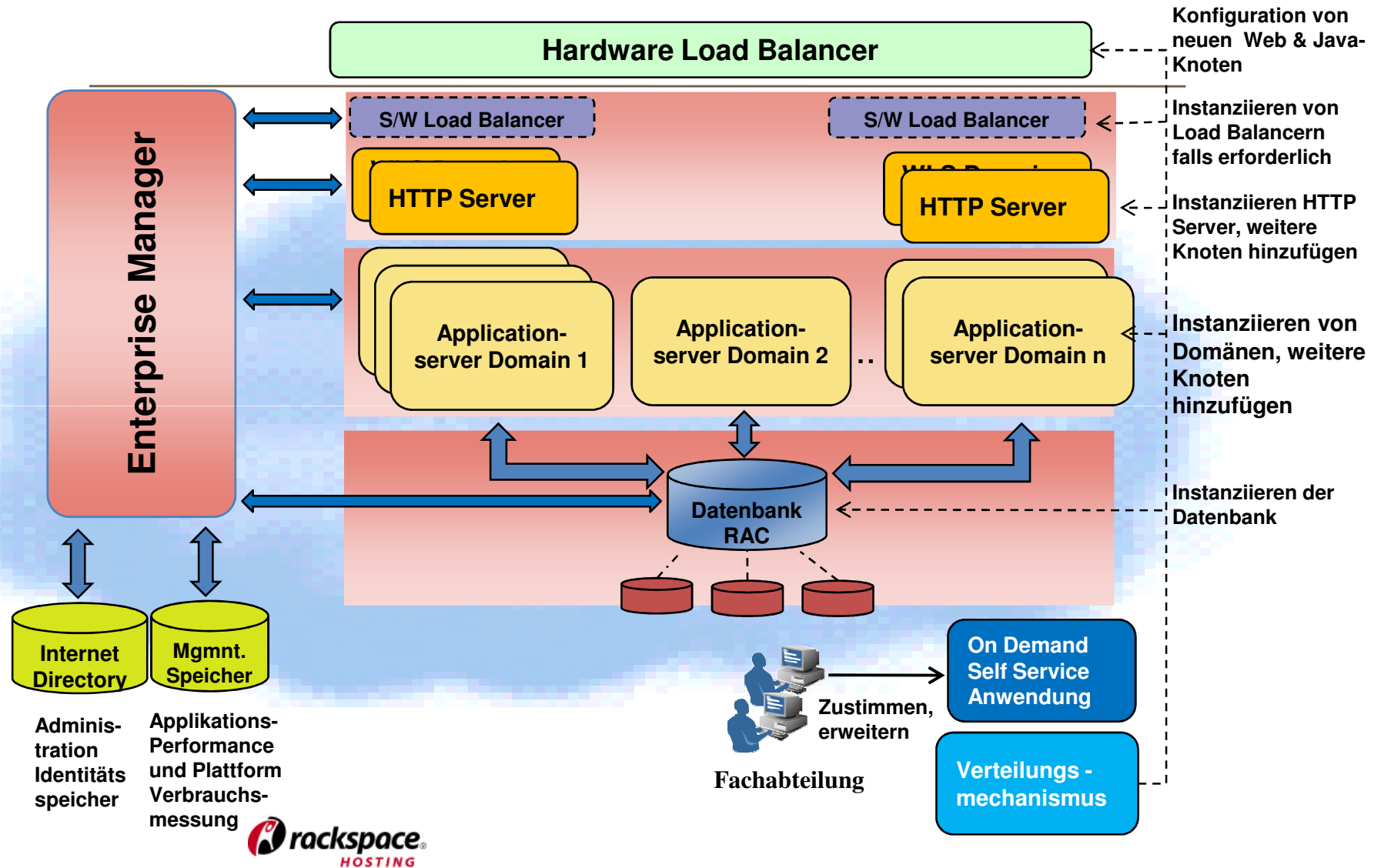
Images running on Intel machines using three different hypervisors

- Oracle WebLogic Server 10g Release 3 Template (x86 32 bit and 64 bit)
- Oracle Database Templates (x86 32 bit and 64 bit) - Both Oracle Database 11g (11.1.0.6) and 10g Release 2 (10.2.0.4)
- Oracle Enterprise Linux Templates (x86 32 bit and 64 bit)

- Oracle Enterprise Manager Grid Control Template (x86 32 bit)
- Oracle Fusion Middleware Service Oriented Architecture on WebLogic Server 9.2 Template (x86 32 bit)
- Oracle Fusion Middleware 10.1.3.4 Service Oriented Architecture on Oracle Containers for JEE 10.1.3.4 Template (x86 32 bit)
- Oracle Application Server 10g Release 3 Webcenter Template (x86 32 bit)



Rackspace Referenz Architektur



Wie ein Programmiermodell für Services aussehen könnte ..

- Ein Service-basiertes Programmiermodell leitet seine technische Umsetzung und Vision vom Grundkonzept eines Services ab
 - Ein Service ist im Wesentlichen eine Abstraktion einer für sich stehenden Softwarefunktion
 - Entwickler bauen Services, nutzen Services und entwickeln Lösungen, die Services beinhalten
 - Die Zusammensetzung (Komposition) von Services und deren Wiederverwendung zu integrierten Lösungen ist ein Schwerpunkt in dieser Betrachtung

Programmiermodell für Services

Kernelemente

- Service-Zusammenfügung
 - Eine technologie- und sprachunabhängige Darstellung für die Zusammensetzung von Services zu Geschäftslösungen
- Service-Komponenten
 - Eine technologie- und sprachunabhängige Darstellung eines Services, der auch aus anderen Services gebildet werden kann
- Service-Datenobjekte (Data Objects)
 - Eine technologie- und sprachunabhängige Darstellung einer Datenentität, die zwischen Services weitergereicht werden können

Service Component Architecture (SCA)

Vereinheitlichtes Programmiermodell für Services

- **Ein Modell für:**
 - die Implementierung von Komponenten
 - das Zusammenfügen von Komponenten zu Anwendungen
 - das Deployment (Verteilung) in (dezentrale) Laufzeitumgebungen

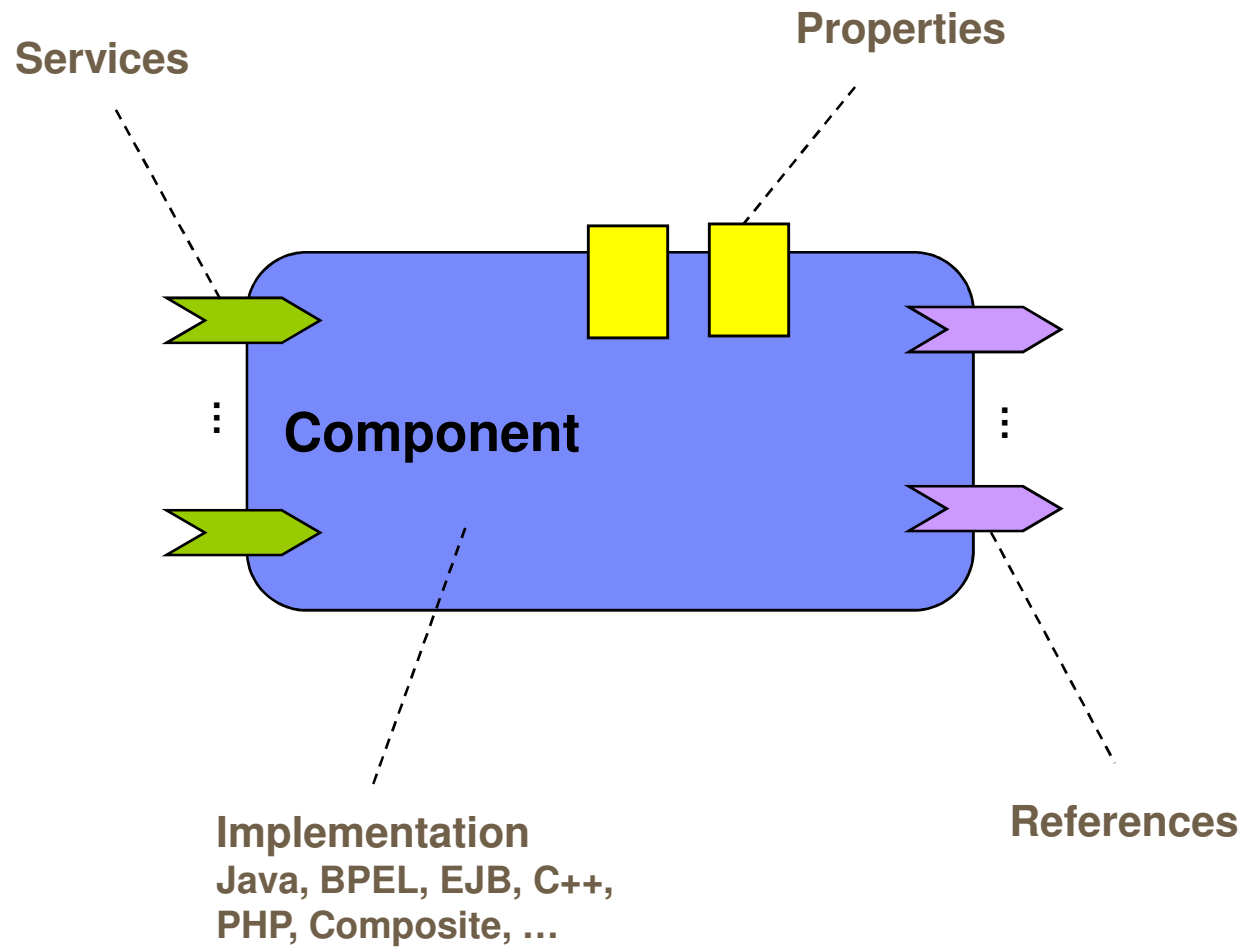
- Service-Komponenten stellen neue Services bereit bzw. konsumieren bestehende Services
- Trennung von Implementierung, Zusammenfügen, Zugriffsmechanismen:
 - Sprachneutral – Komponenten können in verschiedensten Sprache/Frameworks entwickelt werden, wie EJBs, Java POJOs, BPEL, COBOL, C++, ..
 - Technologieneutral – Komponenten nutzen verschiedenste Kommunikationsprotokolle und Infrastruktur, um Komponenten zu verbinden

Was ist eine Komponente?

Vergleich mit einem Kern

- Eine **Komponente (Component)** ist eine **konfigurierte Instanz** einer **Implementierung** einer **Businessfunktion**
- Businessfunktionen stehen anderen Komponenten als **Service** zur Verfügung
- Komponenten können von anderen Services abhängig sein (konsumieren andere Dienste); diese Abhängigkeiten werden **References** genannt
- Komponenten besitzen Eigenschaften, sogenannte **Properties**. Diese Datenwerte beeinflussen die Operationen der Businessfunktionen
- **Konfigurierte Instanz:**
 - die Komponente konfiguriert eine Implementierung
 - Mehr als eine Komponente kann ein und dieselbe Implementierung nutzen und konfigurieren

Was ist eine Komponente?

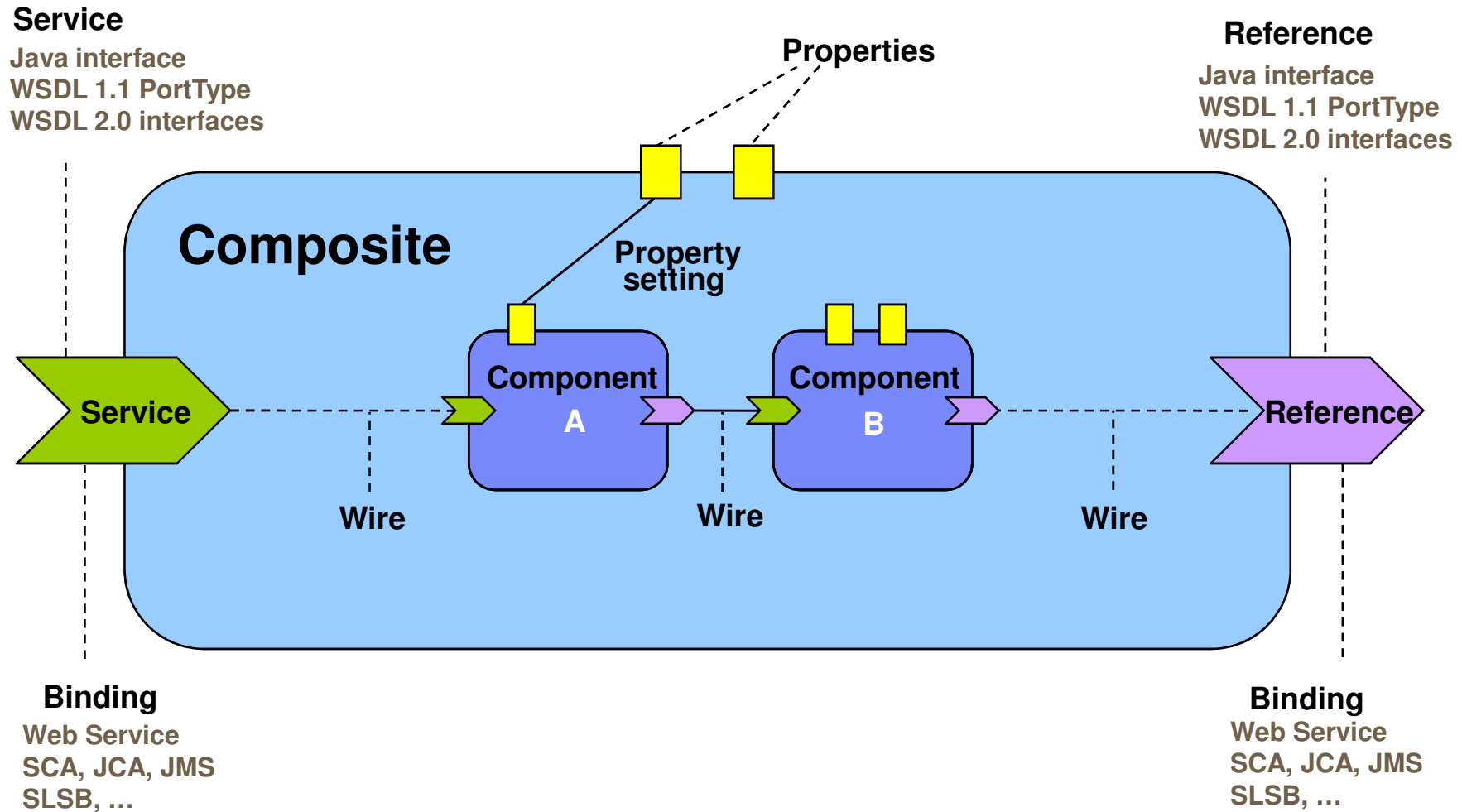


Was ist ein Composite?

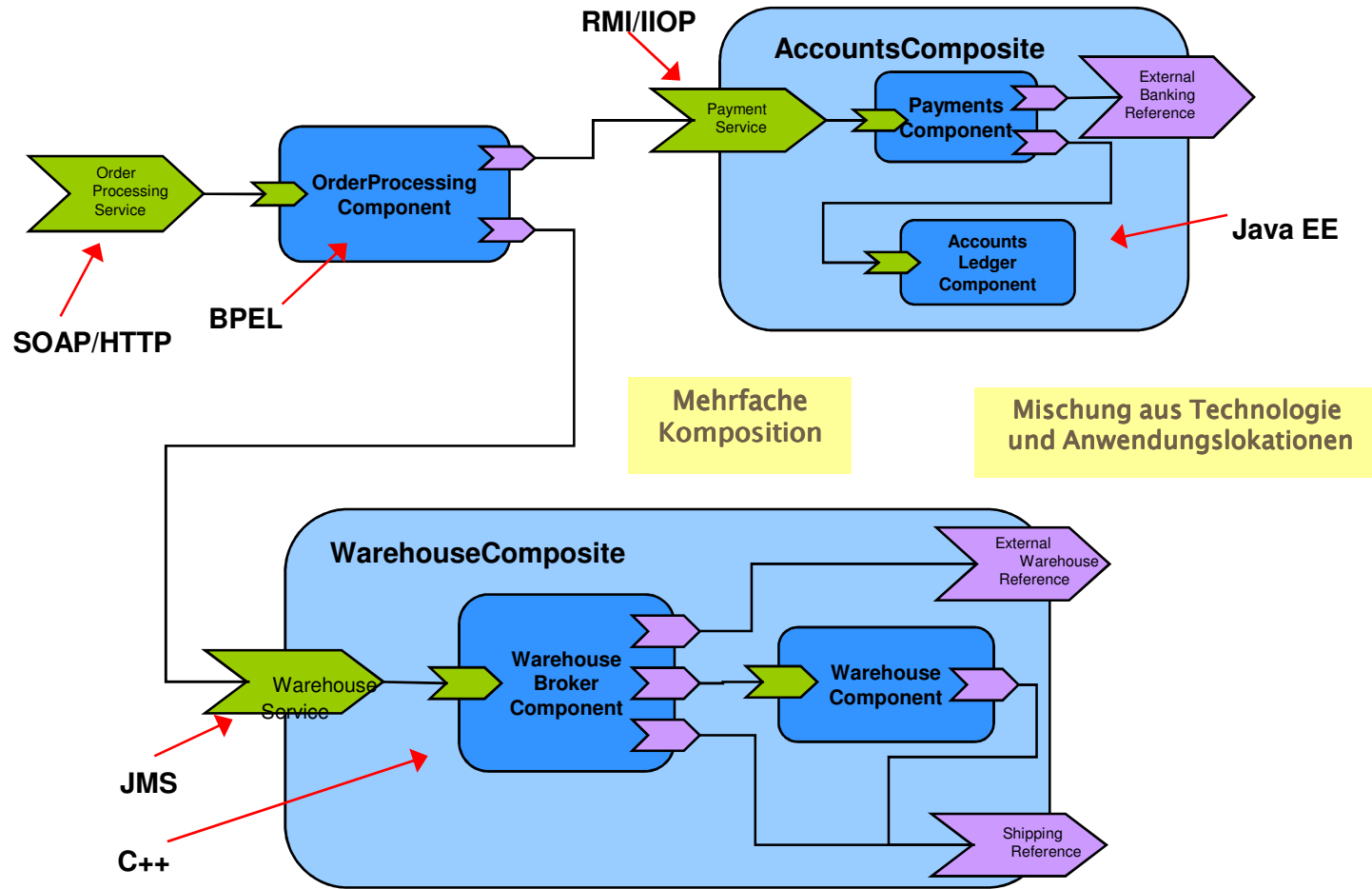
Vergleich mit einer Baueinheit

- SCA beschreibt den Inhalt und das Verbinden einer Applikation in sogenannten Composites (Bauteile, Zusammenbau, Baueinheiten, ...)
- Sie können enthalten
 - Components, Services, References, Property-Deklarationen, sowie
 - Verbindungen (wiring) zwischen den zuvor genannten Elementen
- Composites können selber Komponenten sein
- Composites bilden die Deployment-Einheit und werden verteilt in eine SCA Domain
 - SCA Domain repräsentiert eine Menge von Services für einen Geschäftsbereich

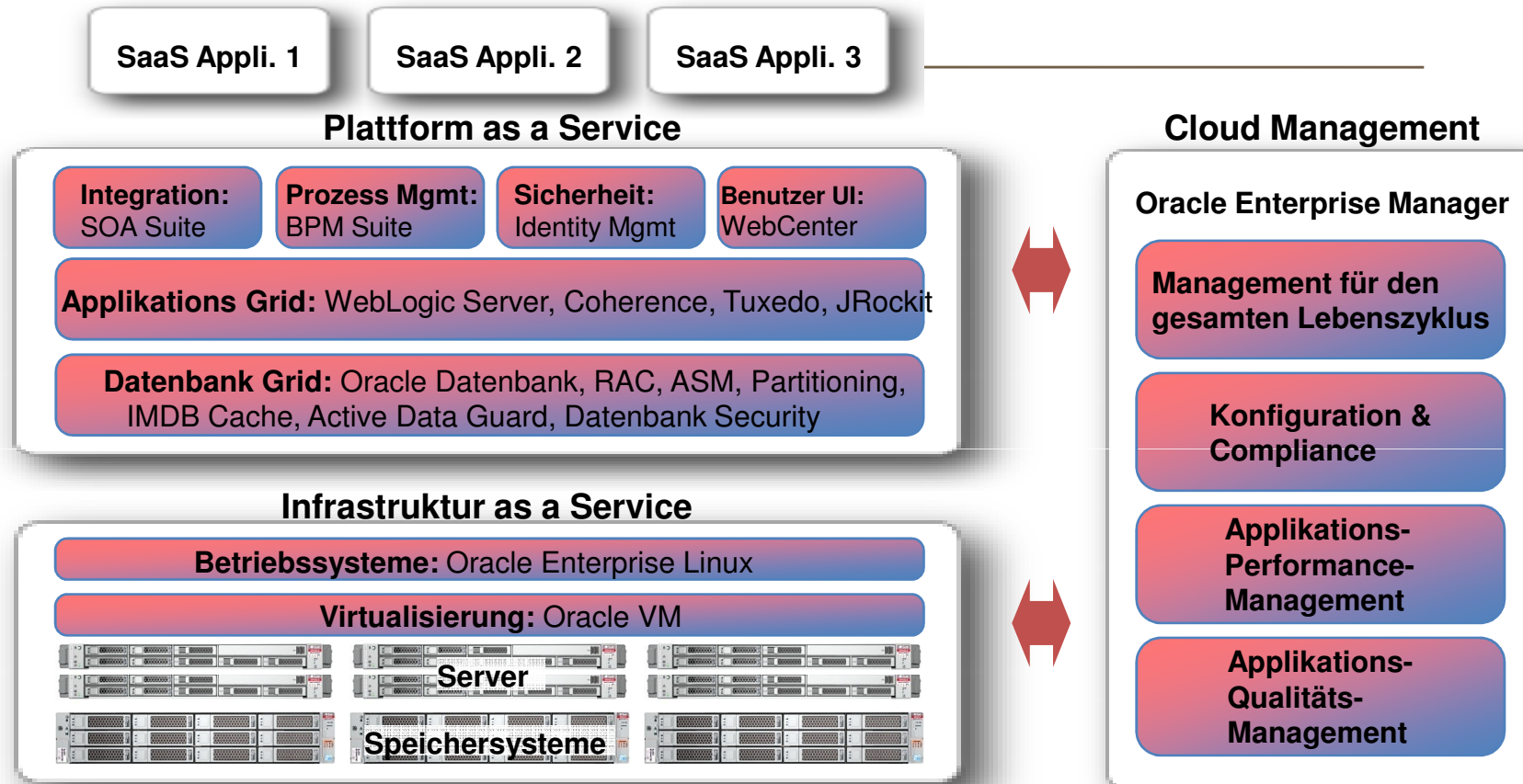
Was ist ein Composite?



SCA Domain / System



Oracle Cloud Plattform



- Umfassende Plattform für Aufbau, Betrieb und Management von SaaS Applikationen
- Service Anbieter für Zuverlässigkeit, Verfügbarkeit, Skalierbarkeit und Sicherheit
- Zentrale und einheitliche Plattform für “On-Premise” und SaaS-basierende Deployments
- Mandantenfähig für unterschiedliche Nutzerkreise

Einheitliche Plattform für Services

- **Ganzheitlich** – Die Plattform für SaaS bietet Partnern eine zentrale und integrierte Umgebung für On-Premise und Cloud-basierten Betrieb
- **Verfügbarkeit & Skalierbarkeit** – Mit der Unterstützung für Grid Computing, Virtualisierung und integriertem Management für die gesamte Plattform, wird mit SaaS eine höhere Verfügbarkeit, kontinuierliche und unterbrechungsfreie Services und beste Skalierbarkeit erreicht
- **Flexibilität** – Angeboten auf der Grundlage einer dynamisch zuschaltbaren und Standard-basierten Architektur die Flexibilität, individuelle Komponenten der Plattform auszuwählen
- **Schneller am Markt** – Mit einer standardbasierten und übergreifenden Plattform, wird die sogen. “Time-To-Market” beschleunigt, durch schnellere Entwicklungszeiten und reduzierte Ausgaben für kostenintensive Entwicklung, Integration und Testverfahren

Zusammenfassung: Cloud Computing

- **Eine Cloud Computing Strategie bedient mehrere Einsatzszenarien**
 - Technologie für den Aufbau von privaten Clouds oder den Betrieb in Public Clouds
 - Betrieb von Applikationen in privaten gemeinsam genutzten Service-Umgebungen oder über Public SaaS
- **Notwendigkeit ganzheitlicher Konzepte um unternehmensweite Rechenzentren auf dem evolutionären Weg zum privaten Cloud Service Anbieter zu unterstützen**
 - Erfahrungen aufgrund jahrelanger Investitionen in die Entwicklung von Grid-Technologie helfen dabei
 - Oracle verwendet Grid-Fähigkeiten in jeder Architekturebene und in den gesamten Produktschichten – Von der Infrastruktur bis zu den Anwendungen
- **Nützlich ist ein umfassendes Komponenten-Portfolio zum Aufbau und Management von Private Clouds und Public Clouds, von der Anwendung bis zum Speichersystem**

12.–15.09.2010
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Vielen Dank!

Wolfgang Weigend

Oracle