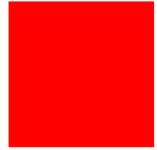


**ORACLE®**

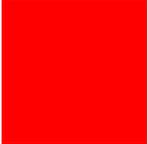
**Flexible Skalierbarkeit und extreme Performance mit Oracle Coherence**

**Wolfgang Weigend  
Sen. Leitender Systemberater**



# Agenda

- **Geschäftstreiber**
- **Weblogic Suite & WebLogic Application Grid**
- **Oracle Coherence Data Grid**
- **Zusammenfassung**

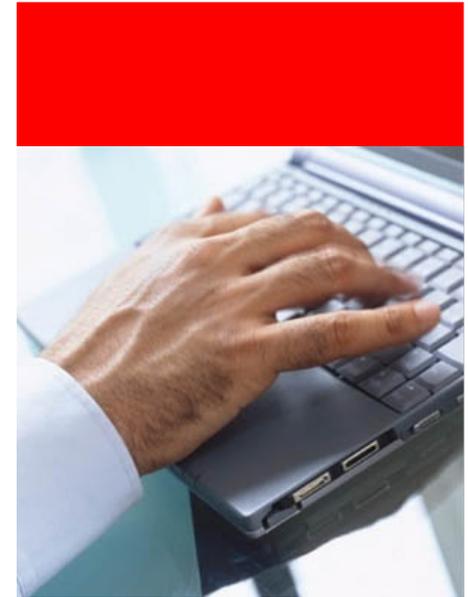


# Geschäftliche Anforderungen und Auswirkungen

- **Skalieren durch dynamische Änderungen (Allianzen)**
  - Dynamische, vorhersehbare Applikationsskalierbarkeit, angepasst an die Geschäftsanforderungen
  - Extremer Anstieg beim Datenzugriff, Volumen und Komplexität der Datennutzung
- **Kundenzufriedenheit erhöhen**
  - Bessere Applikationsleistung
  - Schneller Zugriff auf Daten, kürzere Antwortzeiten
- **Geschäftskontinuität ermöglichen**
  - Kontinuierliche Datenverfügbarkeit und Zuverlässigkeit
  - Service Level Agreements einhalten / überschreiten
  - Eindämmung von Kosten
- **Infrastruktur- und Entwicklungskosten reduzieren**
  - IT-Investitionen wirksam einsetzen, Grid Computing
  - Build vs. Buy

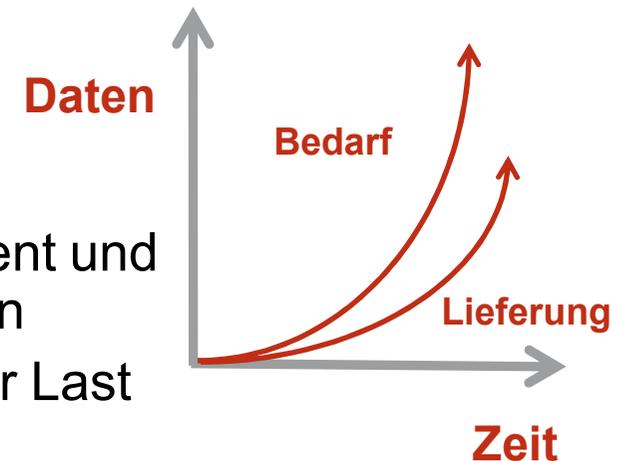
# Skalierbare Leistung und Transaktionsverarbeitung

- **Services und Events stellen neue Anforderungen an die Transaktionsverarbeitung**
  - Unbekannte Anforderungen (Business Geschwindigkeit, neue Zuordnung, Innovation)
  - Komplexität von Services nimmt zu (State), Realtime-Analyse
- **Skalierbare Leistung (High Performance)**
  - Zielsetzung ist, sicherzustellen, dass ein Service innerhalb der erforderlichen, definierten Antwortzeit auch in extremen Situationen reagiert und bei Bedarf adhoc nahtlos skalierbar ist
- **Transaktionsplattform**
  - Verteilte Transaktionsverarbeitung (Distributed Transaction Processing for Open Systems)
  - Extreme Transaktionsverarbeitung (XTP)



# Die IT bestimmt den Datenbedarf

- Virtualisierung
  - Zunehmender Bedarf an Datenquellen
  - Neuverteilung von Applikationen muß transparent und ohne Unterbrechnung vom Datenzugriff erfolgen
  - Gleichzeitige Versorgung bei mehrfach erhöhter Last nötig
- Service Infrastruktur
  - Starker Zugriff auf gemeinsame Ressourcen
  - Gemeinsamer Zugriff bedeutet kontinuierliche Verfügbarkeit und absolute Zuverlässigkeit
- Event Driven Architecture
  - Nur sinnvoll wenn auch Daten enthalten sind, damit Events in einen transaktionalen Rahmen kommen
  - Verbreitung von Daten betrifft alle Systeme



# Umdenken in der Applikationsinfrastruktur

## Enterprise Infrastruktur Anforderungen

- Skalierbarkeit – Capacity on Demand
- Leistung – Zero Latency
- Zuverlässigkeit – Transaktionale Integrität
- Verfügbarkeit – Kontinuierlich

## Hardware Kapazitäts Auswirkungen

- Rechenleistung: SMP/Multicore
- Memory Arrives: “In Memory Option”
- Netzwerkgeschwindigkeit: Gbe
- Storage: Flexibilität

## Enterprise Management Anforderungen

- Grid Automatisierung
- Service Level Management
- Application Performance Mgmt
- Provisioning

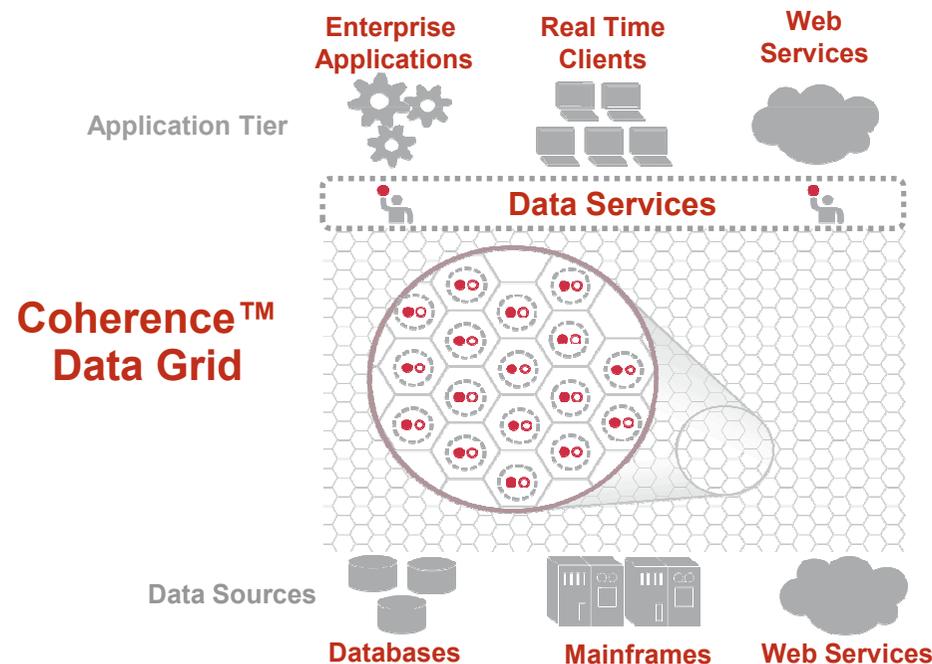
## Software Framework Belastung

- Service Oriented Architecture
- Web 2.0
- Event Driven Architecture
- Extreme Transaktionsvolumen

# Welche Maßnahmen können helfen?

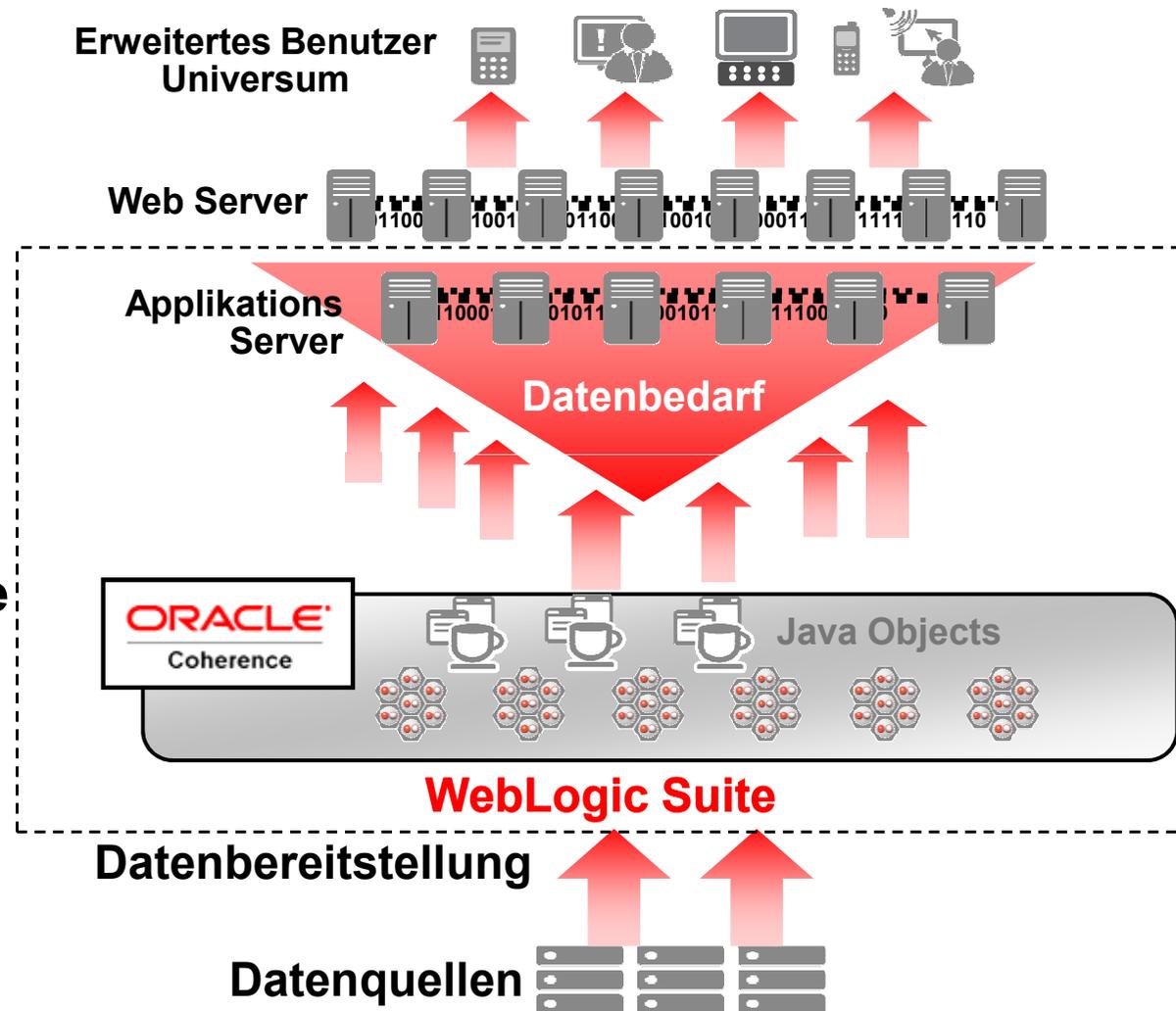
- Data Grid

- Liefert eine zuverlässige Datenschicht mit einer konsistenten Sichtweise
- Ermöglicht die dynamische Erhöhung der Datenkapazität mit Fehlertoleranz und Lastverteilung
- Stellt die erforderliche Verarbeitung und Skalierbarkeit der benötigten Datenkapazität sicher



# Lösungsansatz: WebLogic Suite mit Oracle Coherence als Datenvermittler (Data Broker)

- Oracle Coherence vermittelt die Datenversorgung mit dem Datenbedarf
- Data Grid im Middle-Tier mit Standard-Hardware skalieren

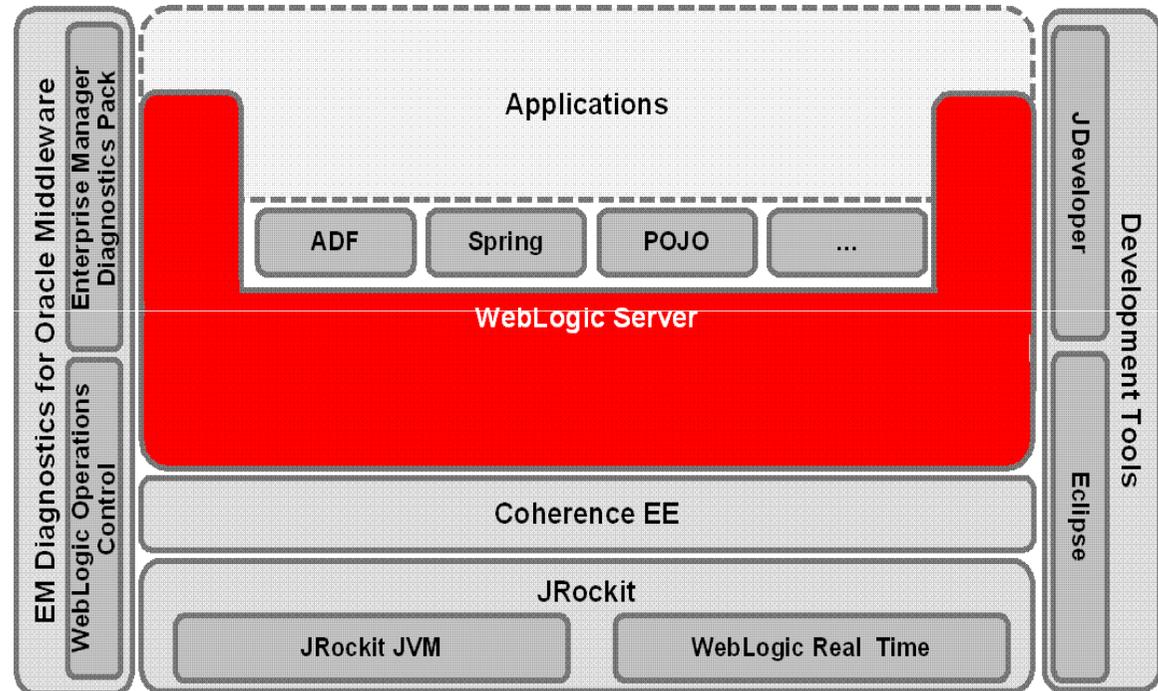


# WebLogic Suite

*Massive Skalierung mit hoher Service-Qualität*

## WebLogic Suite

**WL Enterprise Edition**  
+  
**Coherence EE**  
+  
**WL Operations Control**  
+  
**WebLogic Real Time**  
+  
**iAS EE**



# WebLogic Application Grid

Extreme Leistung, unbegrenzte Skalierbarkeit für die Applikationsinfrastruktur

## WebLogic Application Grid

Coherence Grid Edition

+

JRokit Real Time

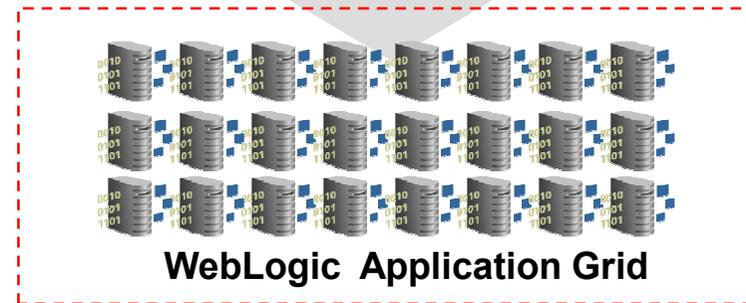
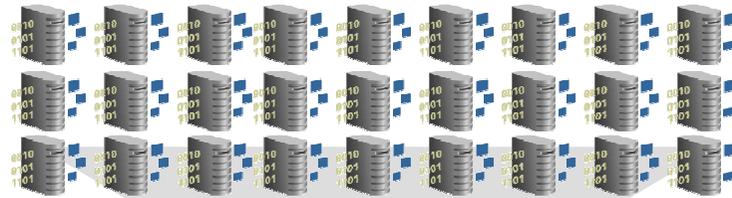
+

WebLogic Operations Control

+

EM Diagnostics Pack

WebLogic Server, Pure Java, .NET, Mainframe



WebLogic Application Grid



Legacy



Commodity



Databases

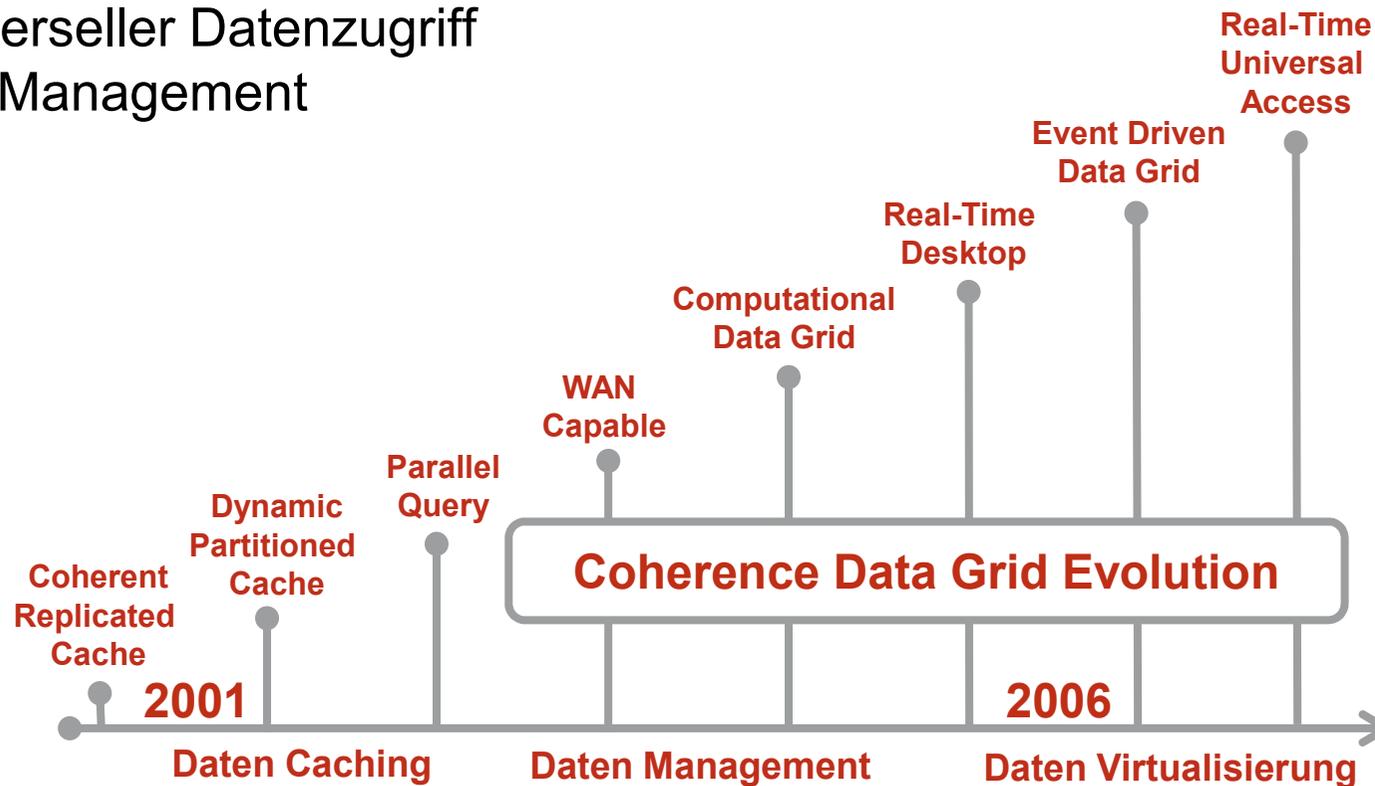


Virtualized Mainframes

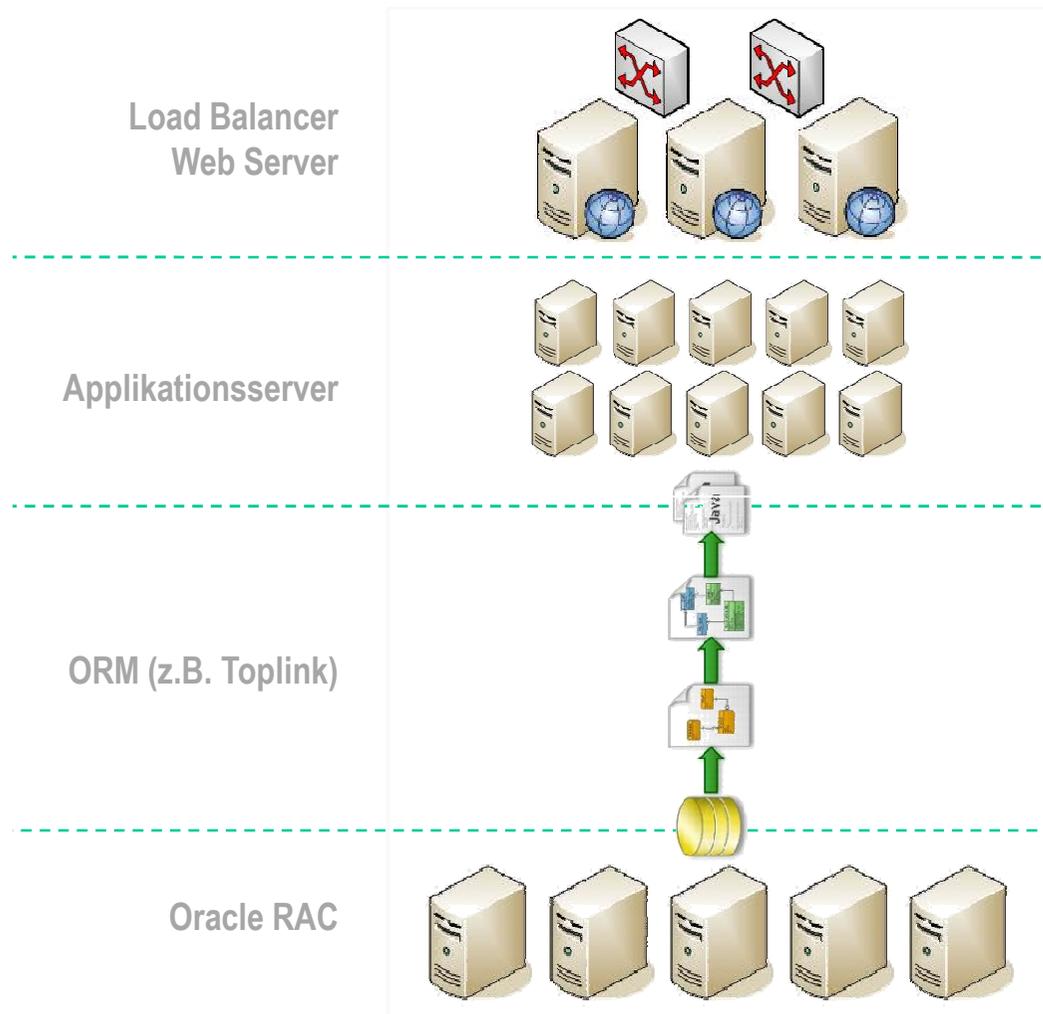


# Coherence Data Grid

- Zuverlässiges Design
- Vorhersagbare Skalierbarkeit
- Universeller Datenzugriff und Management

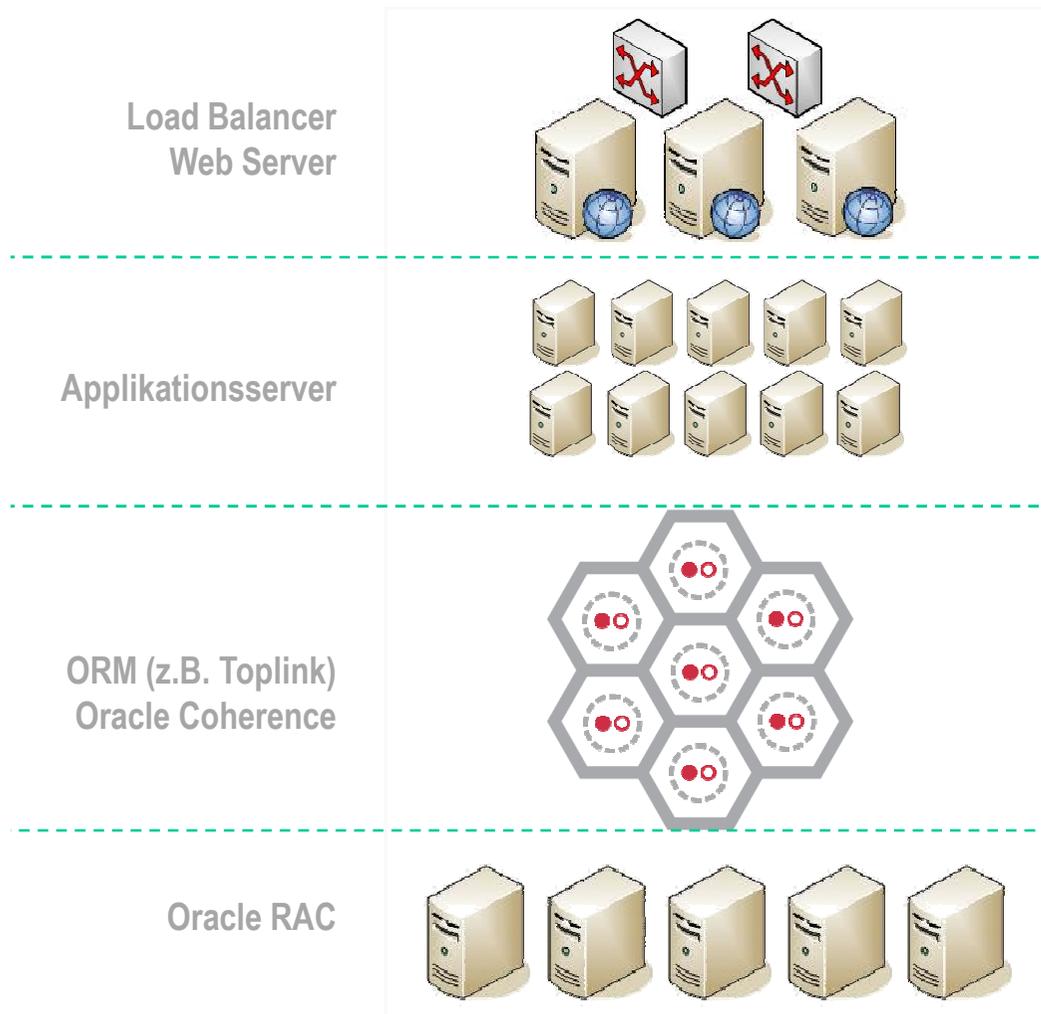


# Herausforderungen



- Skalierbarkeit der Applikationsschicht ist Herausforderung
  - Loosely-Coupled = Split Brain?
  - Transaktionsverarbeitung von Java
- ORM (Java → SQL Mapping) ist ein Bottleneck (Latency)
- In-Memory-Ansatz erforderlich um Antwortzeiten zu erfüllen (Disk-I/O Bottleneck)
- Verlust auch nur eines einzelnen Transaktionsstatus nicht akzeptabel (z.B. Trading)
- Session-States in heterogenen Umgebungen schwierig

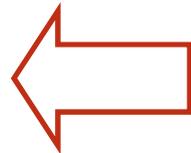
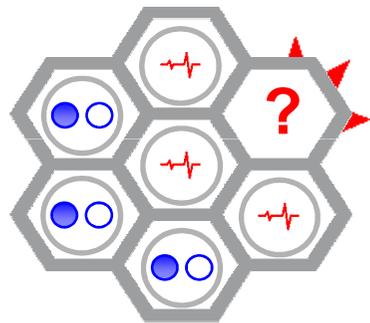
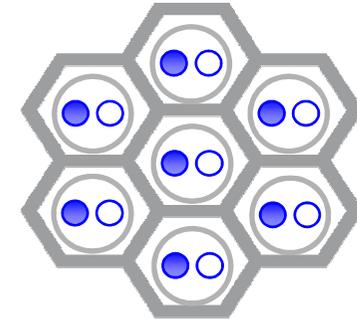
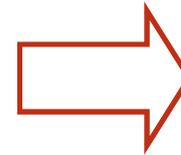
# Oracle Coherence: Data Grid



- Daten Virtualisierung mit Oracle Coherence: Caching von serialisierbaren Objekten im Grid
- Gemeinsame Nutzung von Daten (Read-Through, Read-Ahead)
- Parallele Abfragen, Indizes und Transaktionen im Cluster
- Entlastung der persistenten, unterliegenden Schicht (Write-Through, Write-Behind)
- Integriert in Fusion Middleware Toplink, Berkeley DB, TimesTen

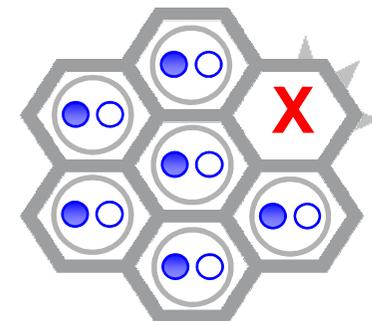
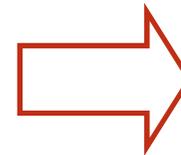
# Wie funktioniert Oracle Coherence?

- Daten werden gleichmäßig auf Clusterknoten verteilt
- Backup der Daten wird ebenfalls auf Knoten verteilt
  - Daten werden automatisch und synchron auf mindestens einen weiteren Server repliziert
- Logisch konsistente Sicht auf Daten von allen Knoten

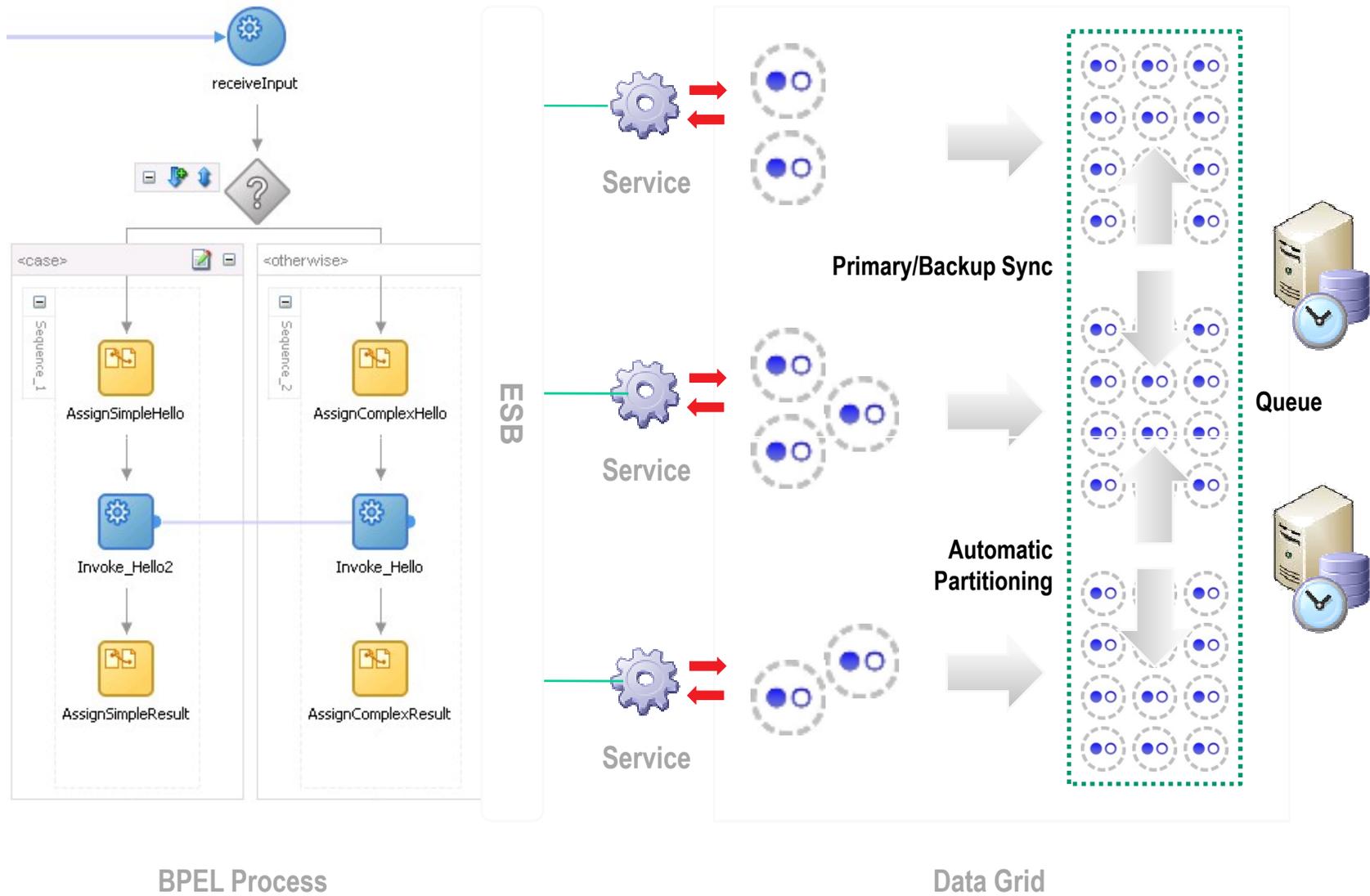


- Alle Knoten prüfen alle Knoten (Health-Check)
- Status bei Fehlern wird cluster-weit diagnostiziert

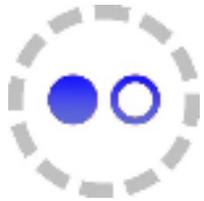
- Knoten wird aus dem Cluster entfernt
- Daten werden wieder gleichmäßig verteilt
- Kontinuierliche Operation: Kein Service-Abbruch oder Datenverlust wenn ein Server ausfällt



# Wie funktioniert Oracle Coherence?



# Oracle Coherence



## Caching

Applikationen nutzen Daten direkt aus dem verteilten, linear skalierbaren Data Grid. Anzahl der Zugriffe auf Backend-Systeme wird reduziert bzw. gezielt vermieden (Datenbanken, Mainframes etc). Write-Queue reduziert Last auf Backend-Systeme, Integration der Berkeley-DB für Cache-Overflow



## Analyse

Aggregationen, einfache und komplexe Abfragen können parallel im Data Grid clusterweit verteilt "In-Memory" ausgeführt werden



## Transaktionen

Applikationen können transaktionsrelevante Daten sicher, hochverfügbar und skalierbar im Data Grid verwalten

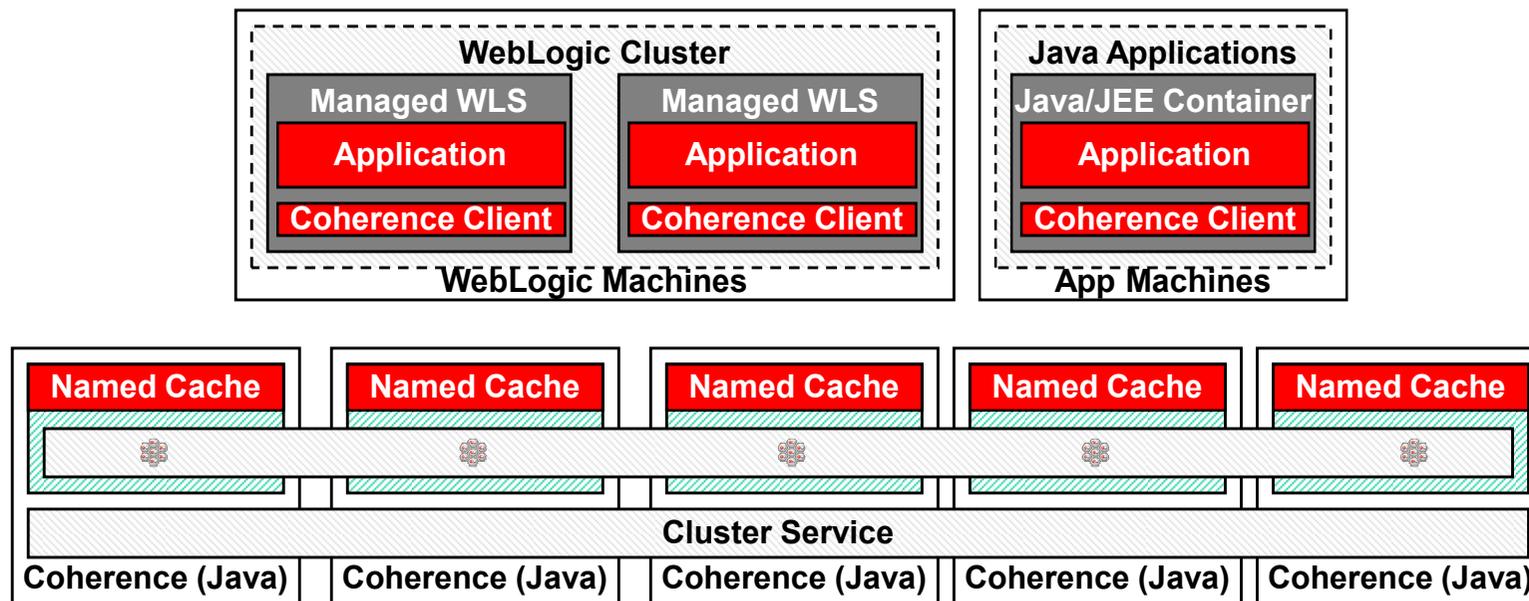


## Events

Automatisches Event-Handling, Änderungsbenachrichtigungen

# Kombination von Coherence und WebLogic Clustered Caching und Zustandsverteilung

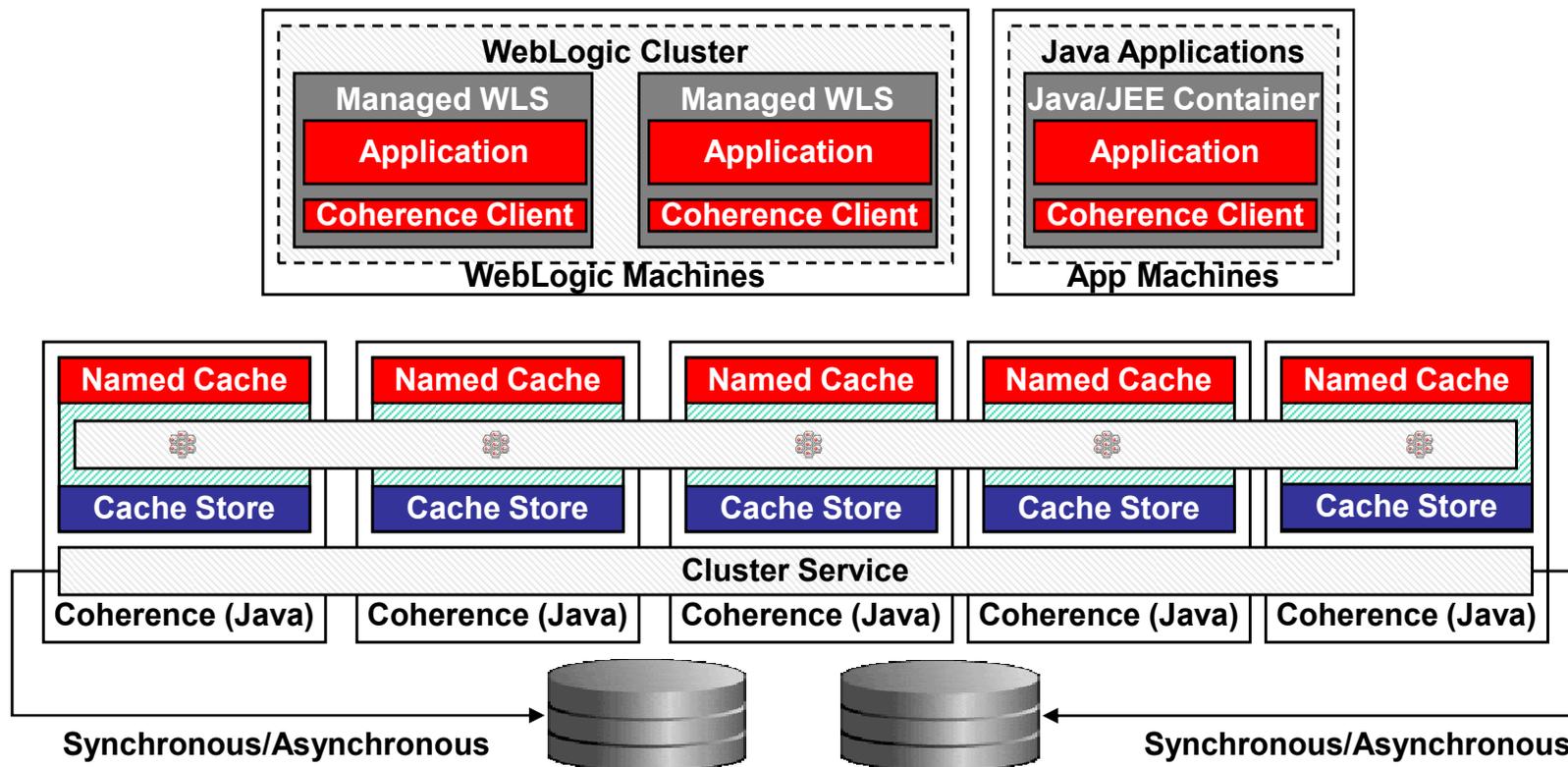
Verteilte Daten im Cache, unabhängige Zustandsverwaltung  
in heterogenen Java Umgebungen



# Kombination von Coherence und WebLogic

## Datenbank Schutzfunktion und Puffer

*Read through* und asynchrones *write through* zur Datenbank



# Zusammenfassung: Vorteile von Oracle Coherence und dem Application Grid

## Leistung

- Extreme, vorhersehbare Reaktionsleistung
- Unbegrenzte lineare Applikationsskalierbarkeit

## Reduzierte TCO

- Bessere Leistung durch die Migration auf Standard-Hardware
- Maximieren der Rechenzentrums-Nutzung mit dynamischen SLAs & Ressourcenmanagement

## Bessere Servicequalität

- Adaptive Applikationsinfrastruktur für die gewährleisteteste Service-Qualität liefern

The Oracle logo is displayed in a bold, red, sans-serif font. The letters are thick and blocky, with a registered trademark symbol (®) at the end of the word.

ORACLE®

**Vielen Dank für Ihre  
Aufmerksamkeit!**

Wolfgang.Weigend@oracle.com