

Verteilte Systeme

Prof. Dr. Martin Becke

CaDS - HAW Hamburg

Version 0.9

Inhalt

1 Ressourcen-Sharing

- Das Ziel
- Beispiele

2 Offenheit/ Openess

- Das Ziel
- Beispiele

3 Skalierung

- Das Ziel
- Dimensionen
- Unterziele
- Funktionale Skalierung
- Speedup

4 Transparenz

Ressourcen-Sharing

Ressourcen gemeinsam nutzen!

- ▶ Hardware, Software und Daten zwischen verschiedenen Knoten und Anwendungen im System teilen
- ▶ Kann Kosten und Komplexität reduzieren
- ▶ Erhöht Verwaltungsaufwand

Ressourcen-Sharing Beispiele

Aus der Praxis

- ▶ Virtualisierungstechnologie
- ▶ Blockchain-Technologie (Dezenrale Datenverwaltung)
- ▶ Cloud-Computing-Plattformen (z.B. AWS, Azure, Google Cloud)

Inhalt

- 1 Ressourcen-Sharing
 - Das Ziel
 - Beispiele
- 2 Offenheit/ Openess
 - Das Ziel
 - Beispiele
- 3 Skalierung
 - Das Ziel
 - Dimensionen
 - Unterziele
 - Funktionale Skalierung
 - Speedup
- 4 Transparenz

Offenheit

Adaptierbar sein!

- ▶ Systemdesign leicht zugänglich und anpassbar
- ▶ Gesteigerte Technologie und Plattformunabhängigkeit
- ▶ Standards und Spezifikationen für Wettbewerb

Offenheit

Beispiele

- ▶ Offene Standards für die Kommunikation
- ▶ Open-Source-Software
- ▶ Offene APIs
- ▶ Offene Datenformate

Inhalt

1 Ressourcen-Sharing

- Das Ziel
- Beispiele

2 Offenheit/ Openess

- Das Ziel
- Beispiele

3 Skalierung

- Das Ziel
- Dimensionen
- Unterziele
- Funktionale Skalierung
- Speedup

4 Transparenz

Skalierung

Das Ziel

- ▶ Mit wachsenden Anforderungen und steigenden Benutzerzahlen umgehen
- ▶ Nicht Leistung oder Stabilität verlieren
- ▶ Funktionalität und Leistungsfähigkeit bei Bedarf anpassbar

Skalierung

Horizontal und Vertikal

- ▶ Horizontal: Zusätzliche Knoten oder Server
- ▶ Vertikal: Mehr Ressourcen auf einem Knoten oder Server
- ▶ Load-Balancer spielen eine besondere Rolle

Skalierung

Unterziele

- ▶ Räumliche Skalierung
- ▶ Verwaltungs- oder administrative- Skalierbarkeit
- ▶ Funktionale Skalierung

Skalierung

Funktionale Skalierung

- ▶ Little's Law
- ▶ Amdahlsche's Law
- ▶ Gustafsons' Law

Skalierung

Speedup

- ▶ Idealer Speedup
- ▶ Realer Speedup
- ▶ Super-Speedup

Skalierung

Bedeutung Parallelisierung am Beispiel von Sort

- ▶ Quicksort
- ▶ Mergesort

Inhalt

- 1 Ressourcen-Sharing
 - Das Ziel
 - Beispiele
- 2 Offenheit/ Openess
 - Das Ziel
 - Beispiele
- 3 Skalierung
 - Das Ziel
 - Dimensionen
 - Unterziele
 - Funktionale Skalierung
 - Speedup
- 4 Transparenz

Transparenz

Einzelne Transparenzen

- ▶ Zugriff
- ▶ Lokalität
- ▶ Migration
- ▶ Replikation
- ▶ Fehler
- ▶ Ort
- ▶ Nebenläufigkeit/Skalierung