

# Verteilte Systeme

Prof. Dr. Martin Becke

CaDS - HAW Hamburg

Version 0.9

# Contents

- 1 Einleitung
  - Allgemein
  - Anwendung
- 2 VS in Hamburg
- 3 VS Definition
  - Definitionen in der Nutzung
  - Definitionen plural?
  - Abgrenzung II
  - Abstraktionsebenen
  - Aspekte und Sichten
- 4 Ziele
  - Allgemeine Ziele
  - VS Ziele

- ▶ Vernetzung und Zusammenarbeit von unabhängigen Computern
- ▶ Gemeinsame Lösung von Aufgaben
- ▶ Effektivere und effizientere Lösung von komplexen Anwendungen und Aufgaben
- ▶ Gemeinsame Nutzung von Rechenleistung und Speicherkapazität mehrerer Computer

- ▶ Weit verbreitet in vielen Bereichen der Industrie
- ▶ Wichtige Rolle bei der Entwicklung neuer Technologien und Anwendungen
- ▶ Besonders wichtig im Bereich der Künstlichen Intelligenz, des maschinellen Lernens und des Internet of Things

# Contents

- 1 Einleitung
  - Allgemein
  - Anwendung
- 2 VS in Hamburg
- 3 VS Definition
  - Definitionen in der Nutzung
  - Definitionen plural?
  - Abgrenzung II
  - Abstraktionsebenen
  - Aspekte und Sichten
- 4 Ziele
  - Allgemeine Ziele
  - VS Ziele

# VS in Hamburg

## Anwendungsbeispiele

- ▶ Smart Port in Hamburg
- ▶ Verteilte Systeme in Flugzeugen (Airbus)
- ▶ Real-Time Gross Settlement System (RTGS),  
Clearinghaus-System

# Contents

- 1 Einleitung
  - Allgemein
  - Anwendung
- 2 VS in Hamburg
- 3 VS Definition**
  - Definitionen in der Nutzung
  - Definitionen plural?
  - Abgrenzung II
  - Abstraktionsebenen
  - Aspekte und Sichten
- 4 Ziele
  - Allgemeine Ziele
  - VS Ziele

# Definitionen

## Beispiele

- ▶ Eine Sammlung unabhängiger Computer, die den Benutzern als ein kohärentes System erscheinen - Tanenbaum
- ▶ Mehrere unabhängige Computer, die miteinander kommunizieren und kooperieren, um gemeinsam eine Aufgabe zu erfüllen - Garg
- ▶ Ein Netzwerk aus autonomen Computern, die miteinander kommunizieren und koordiniert zusammenarbeiten, um eine gemeinsame Aufgabe zu erfüllen - Mukherjee



# Definitionen

## Definitionen plural?

- ▶ Wichtig: Nicht gemeinsam genutzter Speicher
- ▶ Definition muss im Prozess vervollständigt werden
- ▶ Eigenschaften wichtig für korrekten Lösungsbaum (Labyrinth)

# Abgrenzung Verteiltes und Monolithisches System

## Monolithisches System

- ▶ Vorteile:
  - ▶ Einfachere Architektur und Implementierung
  - ▶ Geringere Anforderungen an die Netzwerk- und Kommunikationsinfrastruktur
- ▶ Nachteile:
  - ▶ Begrenzte Skalierbarkeit und Verfügbarkeit
  - ▶ Anfälliger für Ausfälle
  - ▶ Kann zu Single-Point-of-Failure-Situationen führen

# Abgrenzung Verteiltes und Monolithisches System

## Verteiltes System

- ▶ Vorteile:
  - ▶ Bessere Skalierbarkeit und Verfügbarkeit
  - ▶ Robuster und widerstandsfähiger gegenüber Ausfällen
  - ▶ Ermöglicht die gemeinsame Nutzung von Ressourcen mehrerer Computer
- ▶ Nachteile:
  - ▶ Komplexere Architektur und höhere Komplexität in der Implementierung
  - ▶ Höhere Anforderungen an die Netzwerk- und Kommunikationsinfrastruktur

# Abgrenzung Großrechner

## Verteiltes System

- ▶ Mehrere unabhängige Computer vs. Großrechner: eine Computeranlage
- ▶ Höhere Skalierbarkeit und Verfügbarkeit vs. Großrechner: auf eine Computeranlage beschränkt
- ▶ Robuster und widerstandsfähiger gegenüber Ausfällen vs. Großrechner: anfälliger für Ausfälle
- ▶ Höhere Flexibilität in der Software-Entwicklung vs. Großrechner: oft auf eine bestimmte Architektur und Plattform beschränkt
- ▶ Großrechner erfordern oft teurere spezialisierte Hardware und Software

# Abstraktionsebenen

## Grundlage der Analyse

- ▶ Technologische Ebene
- ▶ Anwendungsebene

# Abstraktionsebenen

## AI und ITS

- ▶ ITS: Generisch bildet der Schwerpunkt die Entwicklung von Algorithmen und Protokollen
- ▶ ITS: Beispiele: Auswahl Kommunikations-Technologien, Architektur Infrastruktur
- ▶ AI: Generisch bildet der Schwerpunkt die Anwendung von vorhandenen Technologien
- ▶ AI: Beispiele aus Anwendungsbereich: Cloud, E-Commerce, Datenbanken

# Aspekte und Sichten

## Aspekte

- ▶ Skalierbarkeit und Ausfallsicherheit:
- ▶ Datenmanagement
- ▶ Orchestrierung und Deployment
- ▶ Sicherheit

# Aspekte und Sichten

## Sichten

- ▶ Architektursicht
- ▶ Prozesssicht
- ▶ Datensicht
- ▶ Sicherheitssicht
- ▶ Betriebssicht
- ▶ Entwicklersicht



# Einfluss von Sichten auf Entwicklung

## Beispiel DevOps

- ▶ DevOps ist eine agile Methode
- ▶ Zusammenarbeit zwischen der Entwicklung (Dev) und dem Betrieb (Ops)
- ▶ Enge Zusammenarbeit und Kommunikation zwischen den Teams
- ▶ Neue Werkzeuge wie: Continuous Integration (CI) und Continuous Delivery (CD)
- ▶ Hohe Einarbeitungskosten
- ▶ Hohe kulturelle Herausforderungen

# Contents

- 1 Einleitung
  - Allgemein
  - Anwendung
- 2 VS in Hamburg
- 3 VS Definition
  - Definitionen in der Nutzung
  - Definitionen plural?
  - Abgrenzung II
  - Abstraktionsebenen
  - Aspekte und Sichten
- 4 Ziele
  - Allgemeine Ziele
  - VS Ziele

# Allgemeine Ziele

## Beispiele

- ▶ Funktionalität
- ▶ Zuverlässigkeit
- ▶ Skalierbarkeit
- ▶ Leistung/ Effizienz
- ▶ Sicherheit
- ▶ Wartbarkeit
- ▶ Portabilität
- ▶ Benutzerfreundlichkeit
- ▶ Anpassbarkeit
- ▶ Kompatibilität

# Allgemeine Ziele

## Beispiele Ausfallsicherheit

- ▶ Mission-critical Anwendungen
- ▶ Hochverfügbare Anwendungen
- ▶ Business-kritische Anwendungen
- ▶ Nicht-kritische Anwendungen

# VS Ziele

## Ziele nach Tanenbaum

- ▶ Resource Sharing
- ▶ Offenheit (Openess)
- ▶ Skalierbarkeit
- ▶ Transparenz