

Verteilte Systeme

Aspekte aus der Praxis

Inhalt

1. Einleitung
2. Problemstellung
3. Lösungen
 - 3.1 Wiederholungen
 - 3.2 Schnittstellen-Design
 - 3.3 Engpässe
5. Ausblick

1. Einleitung (1)

Was sind verteilte Systeme ?

- Software ist örtlich auf verschiedene Rechner verteilt
- Software ist logisch auf verschiedene Prozesse verteilt
- Verteilte Software kommuniziert

1. Einleitung (2)

Bedeutung verteilter Software ?

- Parallelrechner
 - Steigerung der Rechenleistung
- Betriebliche Informationssysteme
 - Zugriff auf zentrale Ressourcen von verschiedenen Orten
- WebServices
 - Nutzung von Diensten im Internet

Inhalt:

1. Einleitung

2. Problemstellung

3. Lösungen

4. Ausblick

2. Problemstellung (1)

- Wiederholungen
 - bei Nicht-Verfügbarkeit
 - bei Kommunikationsfehlern
 - als Timeout-Behandlung
 - bei Ressourcenmangel

Inhalt:

1. Einleitung

2. Problemstellung

3. Lösungen

4. Ausblick

2. Problemstellung (2)

- Schnittstellendesign
 - komplexe Abhängigkeiten
 - syntaktische Beliebigkeit
- Engpässe
 - komplexes dynamisches Verhalten
 - Betrieb und Wartung

3. Lösungen

3.1 Wiederholungen

3.2 Schnittstellen - Design

3.3 Engpässe

Inhalt:

1. Einleitung
2. Problemstellung
3. Lösungen
4. Ausblick

Inhalt:

1. Einleitung
2. Problemstellung
3. Lösungen
4. Ausblick

3.1 Wiederholungen (1)

- Kommunikation als neue Ressource
 - Erfordert allfällige Wiederholungen
- Klare Rollenverteilung gemäß Kommunikationsmuster
 - Client - Server
 - Manager - Agent

Inhalt:

1. Einleitung
2. Problemstellung
3. Lösungen
4. Ausblick

3.1 Wiederholungen (2)

- Client - Server
 - Client fordert einen Dienst beim Server an
 - muss mit mangelhaftem Service rechnen
- Manager - Agent
 - Manager beauftragt Agent
 - Agent handelt autonom

3.1 Wiederholungen (3)

Inhalt:

1. Einleitung
2. Problemstellung
3. Lösungen
4. Ausblick

Rolle	Sicherung d. Komm.	Wiederholung
Client	-	+
Server	-	-
Agent	+	+
Manager	+	-

Inhalt:

1. Einleitung
2. Problemstellung
3. Lösungen
4. Ausblick

3.2 Schnittst.-Design

- Möglichst syntaktisch genau
- Aber Aspektorientierung beachten
 - keine Schnittstellenobjekte aus anderen Aspekten in eigene Software übernehmen
- Schnittstellenadapter verwenden
 - Umwandlung von Objekten anderer Aspekte in neutrale Formate
 - Schnittstellenobjekte serialisierbar!

Inhalt:

1. Einleitung
2. Problemstellung
3. Lösungen
4. Ausblick

3.3 Engpässe

- Trennung von Kurz- und Langläufer
 - Routing von Serviceanfragen auf verschiedene Anbieter
 - Routing konfigurierbar
- Skalierbarkeit
 - Mehrfache Instanzierung der Service-Anbieter
 - Instanzzahlen im Betrieb konfigurierbar

Inhalt:

1. Einleitung
2. Problemstellung
3. Lösungen
4. Ausblick

4. Ausblick

- Verteilung ist und bleibt interessant
 - WebServices
 - wenig syntaktische Bestimmtheit
 - erfordert hohe Disziplin für saubere Schnittstellen
 - Embedded Systems
- Ein altes Thema mit Zukunft !

Literatur

Brössler, Siedersleben „Softwaretechnik“,
Hanser-Verlag 1999

Inhalt:

1. Einleitung
2. Problemstellung
3. Lösungen
4. Ausblick