

Middleware

Rüdiger Kapitza
Jürgen Kleinöder

Friedrich-Alexander-Universität Erlangen-Nrnberg
Lehrstuhl Informatik 4 (Verteilte Systeme und Betriebssysteme)

`www4.informatik.uni-erlangen.de`

Wintersemester 07/08

Überblick

Organisatorisches

Kontakt

Inhalt der Veranstaltung

Vorlesungsbetrieb

Leistungskontrolle

Kontakt

Dozenten

- ▶ Rüdiger Kapitza
- ▶ Jürgen Kleinöder

Übungsbetreuung

- ▶ Michael Gernoth
- ▶ Julio Sincero
- ▶ Peter Ulbrich

Inhalt

Vorlesung

- ▶ Grundlagen der Objektorientierung
- ▶ Verteilte Systeme
- ▶ Verteilte Objekte - Java RMI, CORBA und .Net
- ▶ Fragmentierte Objekte
- ▶ Middlewareunterstützung für Fehlertoleranz
- ▶ Komponentenmodelle für Verteilte Systeme - EJB und OSGi
- ▶ Mobile Agenten
- ▶ Middleware für Peer-to-Peer Computing

Übung

- ▶ Objektorientierte Programmierung mit Java in verteilten Systemen
- ▶ Praktische Evaluierung aller vermittelten Konzepte:
Komponenten, fragmentierte Objekte, etc.

Vorlesungsbetrieb

Vorlesungstermin

- ▶ Dienstag 12:00 – 14:00 Uhr
- ▶ Ort: K1

Skript

- ▶ Skript der Vorlesung wird im WWW zur Verfügung gestellt
- ▶ `www4.informatik.uni-erlangen.de/Lehre/WS07/V_MW`

Vorlesungsbetrieb

Rückmeldungen und Fragen

- ▶ Geben Sie uns Rückmeldungen über den Stoff.
Nur so kann eine gute Vorlesung entstehen und gut bleiben.
- ▶ Stellen Sie Fragen!
- ▶ Machen Sie uns auf Fehler aufmerksam!
- ▶ Nutzen Sie auch außerhalb der Vorlesung die Möglichkeit, uns anzusprechen:
persönlich (Zi. 0.043 bzw. 0.037 im RRZE-Gebäude)
per E-Mail: rrkapitz,kleinoeder@informatik.uni-erlangen.de

Übung

Übungstermin

- ▶ Übungsbeginn ist Donnerstag, 18.10.2007
- ▶ Tafelübung Donnerstag 16:15 - 17:45, Raum H4

Inhalt der Tafelübungen

- ▶ Ergänzende und vertiefende Informationen zur Vorlesung
- ▶ Besprechung von Übungsaufgaben
- ▶ Klärung von Fragen, Diskussion
- ▶ Anmeldung zu den Übungen
- ▶ Web-Anmeldesystem Waffel

<https://waffel.informatik.uni-erlangen.de>

Leistungskontrolle

unbenoteter Schein

- ▶ 4 Stunden:
erfolgreiche Bearbeitung aller abzugebenden Übungsaufgaben
- ▶ 2 Stunden:
kurze mündl. Prüfung über den Vorlesungsstoff

benoteter Schein

- ▶ 2 Stunden:
mündl. Prüfung über den Vorlesungsstoff
- ▶ 4 Stunden:
erfolgreiche Bearbeitung der abzugebenden Übungsaufgaben +
mündl. Prüfung über Vorlesungs- und Übungsstoff

Leistungskontrolle

Diplomprüfung Informatik

- ▶ Vorlesungs- und Übungsstoff kann zusammen mit Betriebssysteme oder Verteilte Systeme in der Prüfung gewählt werden.
Prüfer: Prof. Schröder-Preikschat oder Jürgen Kleinöder

Diplomprüfung I&K

- ▶ Prüfung über Vorlesungs- und Übungsstoff – bei weniger als 20 Anmeldungen mündliche Prüfungen
- ▶ Bachelor- oder Masterprüfung CE
- ▶ mündl. Prüfung über Vorlesungs- und Übungsstoff
- ▶ Bearbeitung der Übungsaufgaben ist nicht verpflichtend, für eine erfolgreiche Prüfung aber unbedingt zu empfehlen!
- ▶ ECTS: 8 credit points

Überblick

Überblick

- Objektorientierte Programmierung
- Verteilte Systeme und Verteilte Objekte
- Middleware am Beispiel CORBA
- Fehlertoleranz und Middleware am Beispiel FT-CORBA
- Fragmentiertes Objektmodell
- Webservices
- Objektorientierte Komponentenmodelle
- Objektmobilität und mobile Agenten
- Peer-to-Peer und Middleware

Überblick

Objektorientierte Programmierung

- ▶ Motivation
- ▶ OOP – grundlegende Begriffe

Verteilte Systeme und Verteilte Objekte

- ▶ Grundlagen verteilter Systeme (Remote Procedure Call)
- ▶ OOP und verteilte Anwendungen
(Explizite vs. implizite Kommunikation, Transparenz der Verteilung)
- ▶ Java RMI
- ▶ Design Patterns für nebenläufige und verteilte Anwendungen

Überblick

Middleware am Beispiel CORBA

- ▶ Motivation und Architekturüberblick
- ▶ Der Object Request Broker (ORB) (IDL, Remote invocation, Dynamic invocation, Komponenten des ORB)
- ▶ Portable Object Adaptor
- ▶ CORBA Services

Fehlertoleranz und Middleware am Beispiel FT-CORBA

- ▶ Motivation
- ▶ Arten der Dienstreplikation (aktiv versus passive Replikation)
- ▶ Referenzen und Objektgruppen
- ▶ Architekturüberblick (Replication Manager, Factory, Fault Detector und Notifier)

Überblick

Fragmentiertes Objektmodell

- ▶ Motivation – Warum Alternativen zu Client/Server?
- ▶ Architekturüberblick
- ▶ Aspectix
- ▶ FORMI

Webservices

- ▶ Web-Anwendungen
- ▶ Kurzüberblick: XML, SOAP, WSDL, etc.

Objektorientierte Komponentenmodelle

- ▶ Komponentenmodelle (Grundkonzept, Übersicht)
- ▶ JavaBeans - ein Komponentenmodell für Java (Architektur, Properties, Events, Introspection)
- ▶ Enterprise Java Beans
- ▶ OSGi
- ▶ Jini - Middleware und ein verteiltes Komponentenmodell

Überblick

Objektmobilität und mobile Agenten

- ▶ Motivation
- ▶ Objektmigration am Beispiel des CORBA Lifecycle Service
- ▶ Mobile Agenten

Peer-to-Peer und Middleware

- ▶ Grundlagen Peer-to-Peer Systeme
- ▶ JXTA als Middleware für Peer-to-Peer Systeme