

Übungsblatt 5

zur Vorlesung

Verteilte Systeme/Ubiquitous Computing

Wintersemester 2007/2008

Achtung: Bitte beachten Sie auch die Web-Site der Vorlesung:
<http://www.mobile.ifi.lmu.de/Vorlesungen/ws0708/vs/>

Aufgabe 15: (H) Dienstvermittlungssysteme in der Praxis

- Stellen Sie die Dienstvermittlung innerhalb des “Jini” Systems der Firma Sun Microsystems vor.
- Beschreiben Sie das Grundkonzept von “Universal Plug and Play” (UPnP).
- Erklären Sie die Funktionalität des “Service Location Protocol” (SLP), das von der IETF in den RFCs 2165, 2608, 2609 und 3224 spezifiziert wurde.

Aufgabe 16: (T) Kontextverteilung

- Ein kontextsensitives System besteht aus Kontextquellen (KQ), kontextsensitiven Diensten (KSD) und Nutzern. Diese können jeweils dicht bei einander sein oder weit aus einander. Nehmen Sie an, es gäbe jeweils eine Kontextquelle, einen kontextsensitiven Dienst und einen Nutzer. Welche Kombinationen von Nähe und Entferntheit zwischen den drei Entitäten gibt es?
- Bei der Kontextverteilung ist wichtig, dass ein kontextsensitiver Dienst alle relevanten Kontextinformationen erhält und nur diese. Welche Arten von Relevanz kennen Sie? Erklären Sie diese.
- Was ist “localized scalability”? Wie müssen Kontextquellen, der kontextsensitive Dienst und die Nutzer zu einander angeordnet sein, damit “localized scalability” vorliegt?
- Welche Formen der Kontextbereitstellung kennen Sie?
- Beschreiben Sie, wie skalierbare Kontextverteilungsplattformen aufgebaut sind.

Aufgabe 17: (H) Uhren

$C_r(t)$ sei die Zeit zur UTC-Zeit t auf einem Rechner r . Der Hersteller der Uhr gibt die maximale Abweichung p der Uhr mit 10^{-12} an. Am Anfang sei $C_r(t) = t$. Die Rechneruhr laufe maximal, d.h. mit p , zu schnell. Wie viel geht sie nach einem Jahr vor?

Aufgabe 18: (H) Uhrensynchronisation

Gegeben seien drei Rechner A, B und C. Die Zeit wird in dem Format Stunden:Minuten:Sekunden, Millisekunden angegeben.

- Rechner A verfügt über die physikalische Uhrzeit (UTC-Zeit). Zum Zeitpunkt 10:05:02,081 sende Rechner B eine Anfrage an A nach der UTC-Zeit. Zum Zeitpunkt 10:05:02,093 auf B kommt die Antwort von A mit dem Wert 10:05:01,120 an. Es werde angenommen, dass Rechner A 2 Millisekunden für die Bearbeitung benötigt und Rechner B in vernachlässigbarer Zeit nach dem Erhalt des Ergebnisses seine interne Uhr korrigieren kann. Auf welche Zeit muss Rechner B nach dem Algorithmus von Christian gestellt werden, und wie wird die Uhrenaktualisierung realisiert?
- Rechner A fungiert als Zeitdämon nach dem Berkeley-Verfahren. Rechner A habe die Zeit 10:05:02,000, Rechner B 10:05:02,500 und Rechner C 10:05:01,800. Stellen sie die Prozesse dar, die nach dem Berkeley-Verfahren ablaufen.
- Nehmen Sie an, Rechner A will von Rechner B die Uhrzeit nach dem Network Time Protocol erfragen. Er sendet um 10:05:03,000 eine Nachricht los und empfängt eine Antwort um 10:05:04,000. Diese Antwort gibt an, dass Rechner B die Nachricht um 10:05:04,100 empfangen und um 10:05:04,700 gesendet hat. Wie groß ist der Zeitversatz σ zwischen Rechner A und B? Nehmen Sie an, dass die Zeiten für den Hin- und Rückweg gleich sind.
- Was ist eine logische Uhr?
- Wie funktioniert der Algorithmus von Lamport?

Aufgabe 19: (H) Wechselseitiger Ausschluss

- Welche Vor- und Nachteile hat ein zentraler Ansatz zur Realisierung eines wechselseitigen Ausschlusses?
- Gegeben seien wieder drei Rechner A, B und C. Alle drei Rechner haben ihre Uhrzeiten nun um eine totale Ordnung der Ereignisse erweitert und wollen auf ein gemeinsam genutztes Betriebsmittel zugreifen. Dabei sei der wechselseitige Ausschluss nach dem verteilten Algorithmus von Ricart und Agrawala organisiert. Gleichzeitig senden die Rechner folgende Anforderungen:

A: 10:06:03, B: 10:06:00, C: 10:06:07

Beschreiben Sie die zeitliche Reihenfolge der auftretenden Kommunikationsabläufe und rechnerinternen Entscheidungen, bis alle drei Rechner Zugriff auf das Betriebsmittel hatten. Kennzeichnen Sie auch die in den Warteschlangen gespeicherten Informationen, und markieren Sie aktive Rechner durch z.B. fette Kreisumrandungen.

- Welche Vor- und Nachteile hat der Algorithmus von Ricart und Agrawala?
- Gegeben sei folgende Situation: Rechner A wartet auf Rechner B. Rechner B wartet auf Rechner C und Rechner C wartet auf Rechner A. Wie würde ein solcher Deadlock mit dem Algorithmus von Chandy-Misra-Haas erkannt werden?