

Hinweise:

Bearbeitungszeit: 90 Minuten

Erlaubte Hilfsmittel: keine

Bitte notieren Sie Ihre Antworten ausschließlich auf dem Aufgabenblatt! Bei Bedarf benutzen Sie die Rückseite! Für Skizzen und Entwürfe steht ebenfalls die Rückseite zur Verfügung. Entwürfe, die nicht gewertet werden sollen, sind durchzustreichen.

Hinter jeder Aufgabe steht die Anzahl der Bewertungseinheiten (BE), die für diese Aufgabe vergeben werden. Es gibt insgesamt 45 BE. Zum Bestehen benötigen Sie also 22,5 BE. Die Zeitangabe entspricht der empfohlenen Zeit, die Sie sich mit der Aufgabe beschäftigen sollten, um in der Bearbeitungszeit von 90 Minuten fertig zu werden. Sie ist nur als Richtwert zu sehen.

Viel Erfolg!

Aufgabe 1: Thema: Transparenzforderungen

(6 BE, 10 min)

Student Otto programmiert ein Kundeninformationssystem für ein Einkaufszentrum. Hierfür greift er auf die Informationsrechner der einzelnen Läden zu. Die von Otto implementierte Bedienungsoberfläche sieht folgendermaßen aus:

Zunächst werden dem Kunden die verschiedenen Läden angezeigt. Wenn der Kunde sich für einen Laden näher interessiert, wird ihm die IP-Nummer des Servers und ein Benutzerkonto für diesen Rechner mitgeteilt, damit sich der Kunde dort einloggen kann. Sollte ein Rechner gerade nicht am Netz hängen, so wird das mitgeteilt, damit der Kunde nicht unnötig seine Zeit verschwendet. Einer der Läden hat seine Daten aus Sicherheitsgründen mehrfach auf verschiedenen Rechnern abgespeichert und kümmert sich auch um die Konsistenzhaltung. Für diesen Laden kann sich der Kunde unter mehreren Rechnern einen auswählen, auf dem er sich einloggen möchte. Um die unterschiedlichen Vorlieben der Kunden zu befriedigen, werden hierfür mindestens ein Windows- und mindestens ein Linux-Rechner zur Verfügung gestellt.

Nennen Sie mindestens drei Transparenzforderungen nach ISO, die Otto nicht beachtet hat und beschreiben Sie, was diese aussagen!

Lösung:

Ortstransparenz: Verberge, wo sich welche Ressource befindet.

Zugriffstransparenz: Verberge, wie man auf eine Ressource zugreift.

Ausfalltransparenz: Verberge, welche Ressource gerade ausgefallen ist.

Replikationstransparenz: Verberge, wie vielfach eine Ressource vorliegt.

Aufgabe 2: Thema: Client-Server-Architektur

(4 BE, 10 min)

Erklären Sie den Unterschied zwischen Callback und Polling!

Lösung:

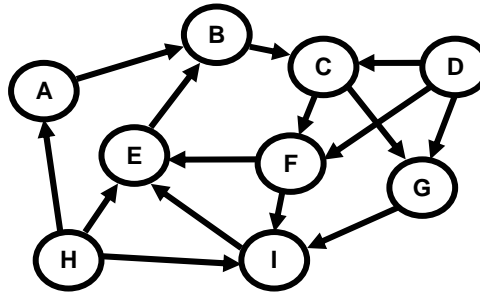
Beim Callback-Verfahren sendet der Client dem Server eine Anfrage und bricht danach die Verbindung ab. Bei Vorliegen der Antwort sendet der Server dem Client die Antwort in einer potentiell neuen Verbindung.

Beim Polling-Verfahren sendet der Client dem Server die Anfrage. Danach fragt er in regelmäßigen Abständen nach, ob die Antwort vorliegt und holt sie sich gegebenenfalls ab.

Aufgabe 3: Thema: Nebenläufigkeit

(8 BE, 15 min)

- a) Betrachten Sie den nachfolgenden Wartegraphen! Geben Sie einen Deadlock in diesem System an! Geben Sie an, wie **ohne** zentrale Steuerung dieser Deadlock erkannt werden kann. (5 BE, 10 min)



- b) Benennen und erläutern Sie ein weiteres Risiko von Nebenläufigkeit! (3 BE, 5 min)

Lösung:

- a) Ein Deadlock ist der Kreis B->C->F->E->B. Die Erkennung von Deadlocks ist äquivalent zur Erkennung von Kreisen im Wartegraphen. Der gegebene Deadlock kann erkannt werden, indem ein beliebiger Prozess des Kreises eine Erkennungsroutine startet und jedem Nachbarn seine Id zuschickt. Jeder Nachbar fügt den bisher zugeschickten Ids seine eigene hinzu und informiert alle Nachbarn. Genau dann, wenn ein Nachbar eine Liste zugeschickt bekommt, in der seine eigene Id bereits enthalten ist, dann befindet er sich in einem Deadlock.
- b) Ein weiteres Risiko sind Wettlauffeffekte, die sich dadurch auszeichnen, dass die Ausführung mehrerer Prozesse unterschiedliche Resultate hervorrufen kann in Abhängigkeit von der Reihenfolge, in der diese Prozesse beendet oder begonnen werden.

Aufgabe 4: Thema: Entfernte Aufrufe

(5 BE, 10 min)

Betrachten Sie einen Remote Procedure Call zwischen einem Client und einem entfernten Server:

Erklären Sie die Rolle von Stubs! Wo werden sie benötigt, im Client oder im Server oder in beiden? Was macht ein Stub?

Lösung:

Stubs sind Platzhalter für die aufrufende und aufgerufene Prozedur. Im Client sammeln sie die Parameter des Aufrufs und leiten das Ergebnis des Servers an die aufrufende Prozedur weiter, im Server führen sie den eigentlichen Prozeduraufruf durch und erhalten das Ergebnis. Zwischen den Stubs werden Aufruf und Ergebnis als serialisierte Nachricht (durch „Marshalling“ erzeugt) versendet.

Aufgabe 5: Thema Namensverwaltung

(7 BE, 15 min)

- a) Betrachten Sie die Domäne www.fh-wedel.de: Server A verwaltet die Domäne `www.fh-wedel.de`, Server B die Domäne `fh-wedel.de` und Server C die Domäne `de`. Aus den USA (Server D) kommt eine Anfrage an www.fh-wedel.de.
Beschreiben Sie die iterative Art der Namensauflösung! (5 BE, 10 min)
- b) Geben Sie jeweils ein Argument für die iterative und für die rekursive Namensauflösung an! (2 BE, 5 min)

Lösung:

- a) Iterative Namensauflösung: Anfrage von D an C, C teilt D die Adresse von B mit, Anfrage von D an B, B teilt D die Adresse von A mit, Anfrage von D an A, A antwortet D.
- b) pro iterative Namensauflösung: Wurzelknoten wird nicht so stark belastet.
pro rekursive Namensauflösung: Anfragewege meistens kürzer.

Aufgabe 6: Thema: Synchronisation von Daten

(8 BE, 15 min)

Für einen Datensatz, der in zehnfacher Kopie vorliegt, gebe es ein Lesequorum $N_R=6$ und ein Schreibquorum $N_W=5$. Erklären Sie an diesem Beispiel die Begriffe Lesequorum und Schreibquorum! Erklären Sie ferner die Begriffe Lese-Schreib-Konflikt und Schreib-Schreib-Konflikt! Kann sich einer dieser Konflikte bei den gegebenen Zahlen ergeben? Geben Sie für Ihre Antwort eine Begründung an!

Lösung:

Das Lesequorum gibt an, wie viele Kopien gelesen werden müssen, damit ein Datensatz übernommen werden kann, in diesem Beispiel also mindestens 6. Bei Nichtübereinstimmungen der Daten wird der aktuellste Datensatz übernommen. Das Schreibquorum gibt an, auf wie viele Kopien exklusives Schreibrecht bestehen muss, damit eine Datenaktualisierung an diesen Kopien vorgenommen werden darf, in diesem Beispiel also mindestens 5. Ein Lese-Schreib-Konflikt bedeutet, dass zu wenige Kopien gelesen werden, um sicherzustellen, dass immer die aktuellste dabei ist. Hierfür reicht es aus, dass $N_R+N_W>N$, was in diesem Fall gegeben ist. Ein Lese-Schreib-Konflikt kann also nicht auftreten. Ein Schreib-Schreib-Konflikt entsteht, wenn zur selben Zeit unterschiedliche Aktualisierungen vorgenommen werden sollen. Hierfür reicht es aus, dass $2N_W>N$, was in diesem Fall nicht gegeben ist. Ein Schreib-Schreib-Konflikt kann also auftreten, wenn zwei Prozesse auf jeweils verschiedenen 5 Kopien versuchen, eine Aktualisierung vorzunehmen.

Aufgabe 7: Thema: Konzepte zur Erzielung von Fehlertoleranz

(7 BE, 15 min)

- a) Gegeben seien 3 Server A, B und C. Jeder soll das auszuzahlende Gehalt des Angestellten Meier abspeichern. Meier hat die Möglichkeit, einmalig genau einen der Server zu manipulieren. Wir wissen leider nicht, welchen. Geben Sie ein fehlertolerantes Verfahren an, das sicherstellt, dass dennoch das richtige Gehalt für Meier in allen drei Servern abgelegt wird. (4 BE, 8 min)
- b) Definieren Sie die Kantenzusammenhangszahl in einem Graphen! Machen Sie eine Aussage zwischen der Kantenzusammenhangszahl und der maximal tolerierbaren Anzahl von Leitungsfehlern! (3 BE, 7 min)

Lösung:

- a) Jeder Server sendet dem anderen seine Version von Meiers Gehalt. Dann übernimmt jeder Server die Mehrheit der Daten. Da mindestens zwei richtig sein müssen, ist die Mehrheit immer das richtige Gehalt.
- b) Die Kantenzusammenhangszahl beschreibt, wie viele Kanten entfernt werden müssen, damit der Graph nicht mehr zusammenhängend ist. Die Kantenzusammenhangszahl-1 ist die Zahl von Leitungen, die gleichzeitig ausfallen dürfen, sodass immer noch jeder Server mit jedem anderen Server kommunizieren kann.