

TDTS04: Svar till tentamen 2010-06-10

av Juha.Takkinen@liu.se

Fråga 1

a) Ett distribuerat system har resurser fördelade på olika plattformar över ett nätverk men är transparent och ser ut som ett enda system för användaren.

b) Migreringstransparens innebär att en resurs kan ha flyttats utan att en användare märker det. Detta hanteras i rpc genom att man använder en katalogserver (eng. directory server) där alla resurser (fjärrtjänster, eng. remote services) finns listade med serveradress och portnummer. En användare kontaktar alltid katalogservern först för att lokalisera var en tjänst finns att tillgå. När en tjänst flyttas från en server till en annan så uppdateras informationen i katalogservern.

c) Dns-systemet är skalbart tack vare udp-protokollet (läser ej resurser för kommunikation), cachar (replikerade resursposter) och distribution (ansvaret för resursposter är delegerat).

Fråga 2

a) Lamportklockor används för att synkronisera en eller flera processer genom att ordna händelser via en relativ klocka som sätts baserat på två regler: i) om en händelse a i en process sker före en annan händelse b i samma process så kommer a före b och ii) om en händelse innebär att ett meddelande skickas från process A till process B så kommer händelsen a = "skicka meddelande" före händelse b = "ta emot meddelande", d.v.s. meddelandekommunikation har alltid en fördröjning över nätverket. Klockorna i processerna justeras när ett meddelande tas emot så att klockan i den mottagande processen alltid visar minst ett mer än klockan i den sändande processen.

b) Funktionen för uppdatering av replikerade resurser kan antingen i) skicka den data som ska uppdateras hos kopiorna, ii) skicka ett utgåendetmeddelande som anger vilken data som är inaktuell eller iii) skicka en eller flera uppdateringsoperationer som måste exekveras i varje kopia.

c) Påståendet är falskt. Bekräftelsen skickas först till primärkopian, som i sin tur sedan skickar en bekräftelse till ursprungsprocessen som förorsakade förändringen av data.

Fråga 3

a) Ett transient objekt existerar endast så länge något annat objekt använder det eller servern är igång. I EJB kan man implementera transienta objekt antingen som stateless session beans eller stateful session beans.

b) `HelloInterface.class` = definierar gränssnittet för fjärrservern (eng. remote interface), d.v.s. vilka fjärrtjänster (eng. remote services) som finns tillgängliga för klienter att använda i det objektbaserade distribuerade systemet

`HelloClient.class` = implementerar kommunikation med användaren och utför sedan rmi-anrop av de fjärrtjänster som användaren vill använda och som definieras i `HelloInterface.class`

`HelloServer.class` = innehåller objektservern för objektet ur klassen `Hello` som implementerar `HelloInterface` och ser till att objektet finns tillgängligt på ett specifikt portnummer på servern

`HelloImpl.class` = innehåller implementationen av gränssnittet som definieras i `HelloInterface.class`

`HelloImpl_Stub.class` = fil skapad av rmi-kompilatorn via kommandot `rmic`; utför marshaling/unmarshaling på parametrar på klientsidan som används i fjärrtjänsterna

c) IOR (Interoperable Object Reference) är systemet som Corba använder för att adressera objekt och speciellt mellan olika Corbasystem. Den taggade profilen innehåller all information som behövs för att aktivera objektet som referensen anger: `iiop-version` (protokollet för kommunikation mellan ORBar), värdnamn enligt dns-systemet (sträng eller ip-adress; eng. host), portnr, objektnyckel (vilken objektadapter som ska användas på mottagarsidan) och komponenter (ev. säkerhetsinformation eller vad som ska utföras om servern saknas).

Fråga 4

a) URL (Uniform Resource Locator) är en platsberoende referens till ett dokument om hur det kan nås och var, t.ex.

<http://www.ida.liu.se/~TDTS04/labs/index.sv.shtml>. URL:en består av namnrymden (här: `http`), dns-namnet på servern (`www.ida.liu.se` och ev. portnr) och namnet på dokumentet (inkl. katalogsökväg på servern: `~TDTS04/labs/index.sv.shtml`).

b) En klient som vill redigera ett dokument som finns på en WebDAV-server (Web Distributed Authoring and Versioning) kan välja antingen exklusivt lås (eng. exclusive; hela dokumentet är låst till en användare) eller delat lås (eng. shared; möjliggör samtidig redigering av flera användare på olika delar av ett dokument). Klienten får sedan en pollett (eng. token) från servern och servern håller reda på vilka polletter som finns hos vilka klienter. Klienten behöver sedan inte upprätthålla uppkopplingen med servern; när klienten vill ändra i dokumentet på servern så gör klienten en `http-post` med polletten inkluderad i anropet, som bevisar att klienten har rättighet att ändra i dokumentet.

c) Eftersom kopior av resurser (siddelar) hämtas när en klient begär det och har blivit omdirigerad till en cdn-server så använder Akamai'sys-

temet ett pull-baserat protokoll. (Rita figur som visar kommunikationen.)

Fråga 5

a) Antag persistent http-uppkoppling utan pipelining och att TCP kan skicka data i det sista steget i handskakningen. TCP-handskakning tar 1,5 RTT inklusive begäran av bas-html-sidan i http, d.v.s. sidan med referenser till de två objekten. Detta dokument tar ytterligare 0,5 RTT för att komma till klienten (ignorerar sändningstid, eng. transmission time).

De två objekten tar sedan $2 \times (0,5 \text{ RTT} + T)$ i nedladdningstid. T = sändningstid (eng. transmission time) för ett objekt = $(500 \times 1024 \times 8) \text{ b} / 10 \text{ Mbps} = 0,4 \text{ s}$. Total tid blir då $1,5 \text{ RTT} + 0,5 \text{ RTT} + 2 \times (0,5 \text{ RTT} + 0,4) \text{ s} = 1,5 \times 10 \text{ ms} + 0,5 \times 10 \text{ ms} + 1 \times 10 \text{ ms} + 0,8 \text{ s} = 3 \times 10 \text{ ms} + 0,8 \text{ s} = 0,83 \text{ s}$.

b) Påståendet är falskt. Tit-for-tat-mekanismen utförs mellan peers, d.v.s. klienterna som delar filer, eftersom bittorrentprotokollet använder sig av TCP, som finns i transportlagret. Routrar hanterar endast nätverkslagret och lägre lager.

c) i) A och NS är exempel på (typer av) resursposter.

ii) ns5.facebook.com fungerar som authoritative server i dns-systemet.

Fråga 6

a) UDP lägger på ett huvud på 8 byte på det den får från applikationslagret; huvudet innehåller information för multiplexning (portnummer) och en checksumma. Det implementerar inga andra tjänster som har med fragmentering/segmentering av meddelanden att göra. Alltså kommer UDP att skicka ett enda paket. Paketet är lika stort som datat från applikationslagret plus UDP-huvudet, d.v.s. 1008 byte.

b) i) Det första segment innehåller 980 B data (och 20 B protokollhuvud)

ii) B skickar endast en ACK till A eftersom de tre segmenten kommer inom en tidrymd av 500 ms; ACK:en är då en s.k. kumulativ ACK och bekräftar mottagandet av alla tre segmenten på samma gång.

c) Antag att $K = 1024 \text{ B}$, d.v.s tröskelvärde = 32768 B. TCP-sändaren kommer att efter 9 sändningsomgångar ha skickat $500 \text{ B} \times (1 + 2 + 4 + 8 + 16 + 32 + 64 + 128 + 129) = 500 \text{ B} \times 384 = 192\,000 \text{ B}$

Fråga 7

a) ARP-paketet kommer från värden längst upp till vänster i figuren, vilket syns på Sender MAC address och IP address i ARP-delen av

ethernetramen. Mottagarens ip-adress i ARP-paketet överensstämmer med det vänstra gränssnittet i routern i mitten av figuren, så värden vill ha reda på detta gränssnitts MAC-adress. Eftersom mottagar-MAC-adressen är satt till broadcast så kommer alla datorer i den vänstra halvan av nätverket att se paketet men endast routern läser av det. Routern kommer att svara med ett ARP-response-paket med sin MAC-adress som avsändare och skicka det som unicast till värden som frågade efter MAC-adressen. Värden kan sedan uppdatera sin ARP-tabell med den nya informationen.

b)

Steg	N'	D(B), p(B)	D(C), p(C)	D(D), p(D)	D(E), p(E)
0	A	10, A	3, A	-	-
1	AC	7, C	done	11, C	5, C
2	ACE	7, C		11, C	done
3	ACEB	done		9, B	
4	ACEBD			done	

Fråga 8

a) i) C kommer att upptäcka en kollision när ramen från A har nått fram till C eftersom C avlyssnar sin egen signal kontinuerligt vid sändning; signalen förändras vid kollision. Tiden för att skicka en bit från A till C = propageringstiden för första biten från A till C = $2000 \text{ m} / (2 \times 10^8) = 10 \text{ mikrosek.}$ Tiden $t_3 = t_1 + 10 \text{ mikrosek.} = 10 \text{ mikrosek.}$

ii) A upptäcker kollisionen när ramen från C har nått till A. Samma beräkning som i i) ovan ger tiden $t_4 = t_2 + 10 \text{ mikrosek.} = 13 \text{ mikrosek.}$

iii) Bittiden i ett 10 Mbps-ethernetnätverk = $1 / 10 \text{ Mbps} = 0,1 \text{ mikrosek.}$ Med $K = 1$ i BEB-algoritmen (Binary Exponential Back-off) så ska bittiden multipliceras med $K \times 512 = 512$, vilket ger 51,2 mikrosek. som väntetid.

b) Ram A: den har tre adresser: avsändare Stationens MAC- och ip-adress, mottagare 1: Accesspunktens MAC-adress och adress 2: routerns MAC-adress samt uppsalanvändarens ip-adress. Ram B: avsändare: Stationens ip-adress, accesspunktens MAC-adress och mottagare routerns MAC-adress samt uppsalaanvändarens ip-adress.