



Løsning på kontinuasjon i TDT4190 Distribuerte systemer
Onsdag 4. august 2004, 0900-1300

Det ønskes korte og konsise svar på hver av oppgavene. Det vesentlige er å kunne dokumentere forståelse, beherske prinsipper og se sammenhenger - ikke å kunne gjengi en mengde detaljer.

Der det synes å mangle noen opplysninger, må det angis hvilke antagelser som synes å være naturlige.

Oppgave 1 – Totaloversikt – 12.5%

- a) Angi presist hva et distribuert system er – og hva som ikke er et distribuert system

SVAR:

Et distribuert system er en samling av sammenkoplete, selvgående systemer. Flere fysisk adskilte systemer fremstår som étt logisk samhørende system. Delsystemene mangler felles klokke og felles lager. En felles oppfatning av tid, tilstand og verdier må således implementeres. Totalresultatet blir et partnerskap mellom delvis samvirkende, delvis uavhengige systemer.

En multiprosessor – med felles lager og felles klokke – er ikke et distribuert system

b) Beskriv kort fordeler og ulemper med distribuerte systemer

SVAR:

Fordeler:

- Deling av ressurser
 - Aksesserbarhet (En applikasjon mot flere ressurser og flere applikasjoner mot en ressurs)
- Ytelse på applikasjoner
 - Parallellitetsutnyttelse (Innen en applikasjon og mellom flere applikasjoner)
- Feiltoleranse
 - Pålitelighet (Fragmentering av applikasjoner og ressurser)
 - Tilgjengelighet (Replisering av applikasjoner og ressurser)
- Systemfokus
 - Større kapasitet
 - Bedre økonomi
 - Stegvis utvidelse
 - Ressurstilpasning i.h.t. organisasjonsforhold
 - Applikasjonstilpasning i.h.t. utførelsesforhold

Ulemper:

- Levende systemer
 - Trenger justerbarhet over lang tid
 - Trenger utvidbarhet i stor skala
- Åpne systemer
 - Behov for portabilitet av ulike applikasjoner
 - Behov for interoperabilitet mellom ulike systemer

Oppgave 2 – Modeller / Standarder – 12.5%

a) Angi presist hva mellomvare (middleware) er – og hva som ikke er mellomvare

SVAR:

Mellomvare er et programvare lag (med et generelt API) som konstrueres mellom globale applikasjoner og lokale ressurser slik at en gitt applikasjon kan nå en vilkårlig lokal ressurs, og en gitt ressurs kan nås av en vilkårlig applikasjon – hvor mellomvare laget står for ruting av forespørslene til riktig node og oversetting av forespørslene til riktig språk.

Generelle kommunikasjonslag – som håndterer hverken ruting eller oversetting – er ikke mellomvare

- b) Beskriv kort DCE (Distributed Computing Environment) arkitekturen

SVAR:

Mellomvare-type

- RPC-basert

DCE-tjenester

- Fundamentale tjenester
- Delingstjenester

Fundamentale tjenester

- Tråder: POSIX 1003.4a P-tråder
- RPC: En/flere bindingsmåter per tjeneste – En/flere grensesnitt per tjener
- Tid: Fysisk klokke, UTC-basert
- Navn: Hierarkisk, lokal og global – Interoperabel, X.500 og DNS
- Sikkerhet: Autentisering, Autorisering, Verifisering, Kryptering

Delingstjenester

- Sekundærlager filer: POSIX 1003.1 Filer – Disk basert
- Primærlager filer: BOOTP- og TFTP-basert
- Forvaltning: Høy tilgjengelighet via høy replisering

Oppgave 3 – Kommunikasjon / Synkronisering – 12.5%

- a) Angi presist hva vektorklokker er – og eksplisitt hvorfor de trengs

SVAR:

Vektorklokker brukes til å implementere et felles tidsbegrep. Vektorklokker holder oversikt over hvor mange hendelser i hver enkelt node som er kjent på en gitt node.

Vektorklokker trengs – ut over logiske klokker – til å skille mellom hendelser på ulike noder

- b) Beskriv kort en ekte distribuert algoritme for utvalgelse av en sekretærprosess (ikke av ringtypen)

SVAR:

Start Valg-rutine:

- 1a) Send Valg-melding til alle med høyere identifikator

Får Valg-melding (fra noen lavere):

- 2a) Send OK-melding til avsender
2b) Start Valg-prosess som over

Får OK-melding (fra noen høyere):

3a) Gi opp for andre

Får ingen melding (fra noen høyere):

4a) Send Sjef-melding til alle m/Egen identifikator

Oppgave 4 – Distribuerte filsystemer – 12.5%

- a) Angi presist hva filcaching er – og eksplisitt hvordan det utnyttes

SVAR:

Caching tilbyr temporære kopier av et dataelement nærmere bruksstedene enn det dataelementet i seg selv tilsier.

Disse kopiene tilsier økt sikkerhet i tillegg til økt nærhet

- b) Beskriv kort AFS (Andrew File System) arkitekturen

SVAR:

CMU AFS	
Globalt filsystem:	Felles
Cache type:	Hele filer
Konsistens:	Underretting
Oppdat. Semantikk:	Fil
Replikat type:	Kun RO
Orientering:	Feiltoleranse
Implementasjon:	Bruker + Kjerne

Oppgave 5 – Distribuerte databasesystemer – 12.5%

- a) Angi presist hva transaksjoner er – og eksplisitt hvorfor de trengs

SVAR:

Transaksjoner er programvare enheter som skal fungere som

- Arbeidsenheter – dvs. at alle leste verdier forblir lokalt brukbare underveis
- Samtidighetsenheter – dvs. at alle skrevne verdier forblir globalt synlige først etterpå
- Rekonstruksjonsenheter – dvs. at alle skrevne verdier forblir angrbare / bekreftbare

med henblikk på ekstern parallellitet og / eller interne feil.

Problemer som transaksjoner da håndterer er:

- Manglende atomiskhet – dvs. at transaksjoner ikke er udelelige
- Manglende konsistens – dvs. at transaksjoner ikke bevarer konsistens
- Manglende integritet – dvs. at transaksjoner ikke er isolerbare
- Manglende varighet – dvs. at transaksjoner ikke overlever sammenbrudd

- b) Beskriv kort en ekte distribuert algoritme for håndtering av vranglåser (ikke av unngåelsestypen)

SVAR:

Kantjaging (av oppdagingstypen)

- Hver node vedlikeholder en lokal vranglåsgraf
- Ved mistanke om vranglås, eller ved jevne mellomrom, forespør en vilkårlig node en annen node om dens lokale vranglåsgraf inneholder deler som kan kobles med den spørrende nodens lokale vranglåsgraf
- Forespørselen baseres på interessante deler av ens egen lokale vranglåsgraf, og forespørslene skjer rekursivt i mengden med involverte noder
- Den initierende noden vil til slutt ha skaffet seg de delene av de ulike vranglåsgrafene som samlet avdekker eventuelle vranglåser

Oppgave 6 – Distribuert delt lager – 12.5%

- a) Angi presist hva et distribuert delt lager er – og eksplisitt hvordan det utnyttes

SVAR:

Et distribuert delt lager tilsier en simulering av felles lager i en omgivelse uten felles lager

En multiprosessor – med fysisk delt lager – er lett å programmere (SW-messig). En multimaskin – uten fysisk delt lager – er lett å konstruere (HW-messig). Distribuert delt lager – simulert, logisk delt lager – søker det beste fra begge verdener.

- b) Beskriv kort likheter og ulikheter mellom LINDA og ORCA arkitekturene

SVAR:

LINDA:

- Kjøretidssystem simulering av delt lager (dvs. postbasert)
- Objektbasert overføring (altså ikke variabelbasert)
- Enkel objektmodell (PROLOG-lignende)

ORCA:

- Kjøretidssystem simulering av delt lager (dvs. postbasert)
- Objektbasert overføring (altså ikke variabelbasert)
- Full objektmodell (JAVA-lignende)

Oppgave 7 – Distribuert pålitelighet – 12.5%

- a) Angi presist hva distribuert feiltoleranse er – og hvilke alternativer til det som finnes

SVAR:

Feiltoleranse tilsier å maskere effektene av distribuerte feil på ulike måter, med tilhørende gradering av ytelse og funksjonalitet

En kan alternativt nærmest forhindre feil gjennom tilnærmet ubegrenset redundans. En kan også endog rette opp effektene av feil ved mer begrenset redundans.

- b) Beskriv kort hva majoritetsmekanismer brukes til i.f.m. distribuerte databaser

SVAR:

Med replisering og/eller kommunikasjonsfeil:

- Kan utnyttes til å utpeke partisjon for oppdatering (replisering & feil)
- Kan utnyttes til å sikre kollisjon av låser (kun replisering)
- Kan utnyttes til å minske blokkering i forsøk på å oppnå atomiskhet (kun feil)

Oppgave 8 – Distribuerte navnetjenester – 12.5%

- a) Angi presist hva iterativ navnenavigering er – og hvilke alternativer til det som finnes

SVAR:

Klienten selv mapper del for del av navnet i en sekvensiell prosess mot ulike tjenere

En kan bruke rekursiv navnenavigering som alternativ til iterativ navnenavigering, og hver av disse to metodene kan gå direkte mot bare en tjener eller flere ulike tjenere.

- b) Beskriv kort GNS (Global Name System) arkitekturen

SVAR:

- Tillater sammenslåing og restrukturering
- Tillater en fleksibel attributtstruktur

