

# Mellomvare for distribuert multimedia og mobilitet

INF 5040 høst 2006

foreleser: Frank Eliassen

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UIO

1

## Litteratur

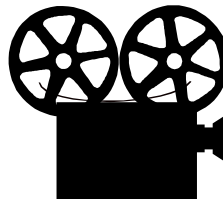
- q Tom Fitzpatrick, Gordon S. Blair, Geoff Coulson, Nigel Davies and Philippe Robin, “Supporting Adaptive Multimedia Applications through Open Bindings” (se <http://heim.ifi.uio.no/~frank/INF5040>)
- q Blair, Coulson, Robin, Papathomas: “An architecture for next generation middleware” (se <http://heim.ifi.uio.no/~frank/INF5040>)
- q Gjørven, Eliassen, Lund, Eide, Staehli., “Self-Adaptive Systems: A Middleware Managed Approach”, (se <http://heim.ifi.uio.no/~frank/inf5040>)

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UIO

2

## Utfordring fra multimedia og mobilitet til mellomvare

- Programmeringsmodell og systemstøtte for kontinuerlige media
  - strømmer (enkle og sammensatte)
- QoS forvaltning (management)
  - statisk og dynamisk
- Kontekstopptakelse og adaptivitet



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

3

## Betydning for mellomvare

- Generelle bilde for konvensjonell mellomvare ...
  - Understøtter IKKE kontinuerlige media
  - Tillater IKKE forvaltning av tjenestekvalitet
  - Tillater IKKE sanntidssynkronisering
  - Ingen eller begrenset støtte for kontekstopptakelse og adaptivitet

=> *Behov for forskning på dette området*

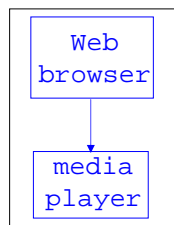


INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

4

## Streaming via nett-leser

- q Basert på “helper application”: media player
  - q Dekomprimering
  - q Fjerning av jitter
  - q Feilretting (rekonstruksjon av tapte pakker - FEC)

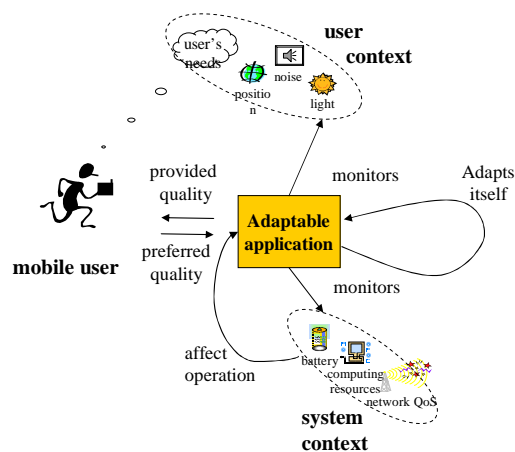


INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

5

## Applikasjoner for mobile omgivelser

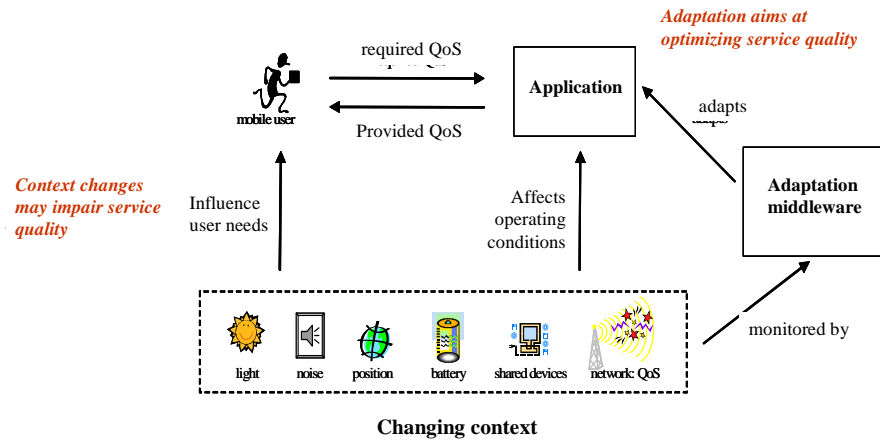
- q Mobil data-behandling krever spesielle løsninger på mange områder pga stadig og dynamisk endring i eksekverings-omgivelsene
  - q kontekstoppmerksomme applikasjoner
  - q adaptive applikasjoner



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

6

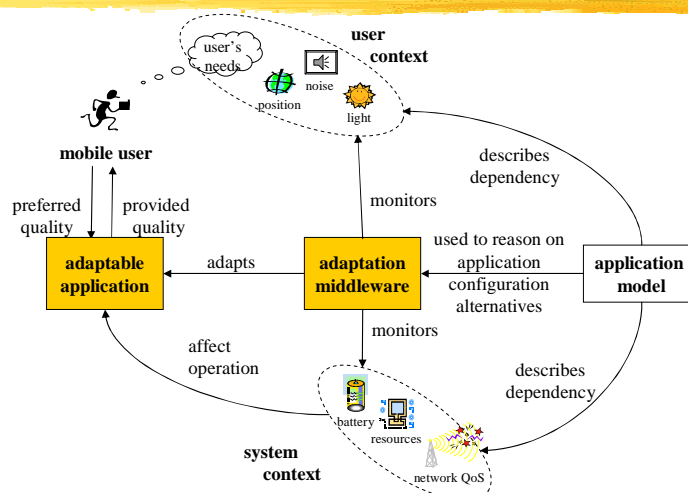
## Ekstern adaptasjonskontroll forenkler utvikling av mobile applikasjoner



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

7

## Ekstern adaptasjonskontroll krever at mellomvaren har tilgang til en modell av applikasjonen



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

8

## Mobil mellomvare krever støtte for heterogene datanett

- q Mobile enheter kan bevege seg mellom ulike teknologidomener med ulike karakteristika mhp bl.a. båndbredde, forsinkelse, og tap
- q Tilpasse seg endring
  - q På operativsystemnivå
  - q I mellomvareplattformen
  - q I applikasjonen



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

9

## Betydning for mellomvare

- q Mellomvaren må tilby en åpen arkitektur som tillater at den kan tilpasses dynamisk til endringer i den operasjonelle omgivelse
- q Generelle bilde for konvensjonell mellomvare ...
  - q Tilbyr generelt en svart-boks arkitektur
  - q Begrenset (ad hoc) åpenhet
    - CORBA: Åpenhet i essens kun uttrykt ved IIOP, POA og "interceptors"
  - q Tilbyr ikke *dynamisk* tilpassing

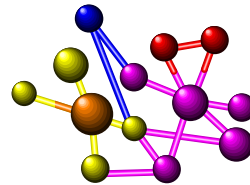


INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

10

## Støtte for mobilitet: Løsningen?

- q Konfigurerbar mellomvare og applikasjon
  - q Både mellomvaren og applikasjonen er komponent-baserte
    - jfr DCOM, EJB, CCM
- q Re-konfigurerbar mellomvare og applikasjon
  - q muliggjør inspeksjon av viktige hendelser (monitorering)
  - q muliggjør adaptasjon (tilpassing)
    - Endre adferden til eksisterende komponenter (parameterkonfigurering)
    - legge til nye komponenter, bytte ut komponenter, ...



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

11

## Distribusjonstransparens

- q Tradisjonelle rolle til mellomvareplattformer
  - q Filtrere bort problemer i forb. med distribusjon gjennom selektive transparens funksjoner
- q Den nye rollen
  - q mellomvare som en informasjonsleverandør i stedet for informasjonsskjuler
  - q mellomvare som en katalysator for adaptasjon

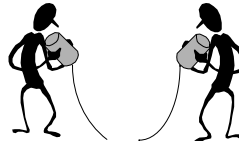


INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

12

## Eksempel på adaptasjon: I kommunikasjonssystemet

- Endre valg av protokoller
  - Rekonfigurer stakken til å passe nye/endrede betingelser i nettverket
- Endre protokoll parametre
  - Endre transmisjonsrate, policy for retransmisjon, buffer allokering, pakkestørrelse, etc.
- Andre optimaliseringer
  - optimalisere multicast ved å avbilde til ny nettverksteknologi ...



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

13

## Eksempel på adaptasjon: I mellomvare

- Sett inn nytt filter i datastrømmen
  - f.eks. MPEG komprimering for multimedia
- Endre parametre til et filter
  - f.eks. miks av I-, P- and B-rammer
- Andre strategier
  - Bruk “pre-fetching” eller “on-demand” caching
  - Endre prioritering/ scheduling av data
  - Sett inn delay-buffer for å redusere jitter



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

14

## Eksempel på adaptasjon: På høyere nivå

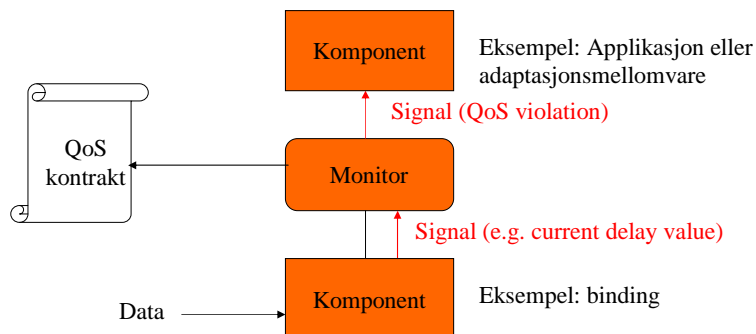
- q Restrukturer applikasjon
  - q Avlast prosessering til fjerne maskiner
  - q Bruk av proxy tjenere/filtre
- q Velge nye tjenester
  - q f.eks. basert på nærhet
- q Endre arbeidsmodus
  - q f.eks. fra RMI til mer asynkron interaksjon

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

15

## Beslutning om adaptasjon kan forårsakes av at QoS-kontrakter ikke overholdes

- q Kontekst/QoS overvåking ved monitorering



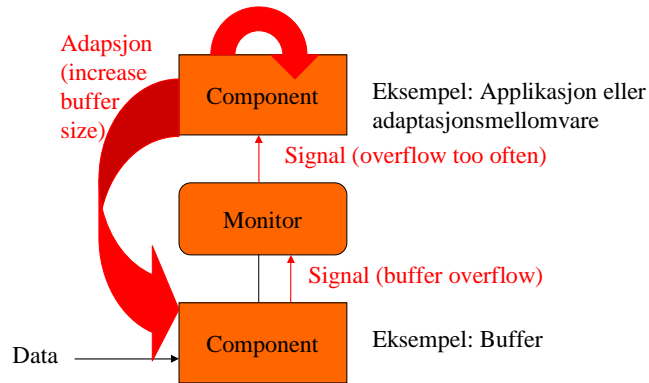
INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

16



## Adapsjon pga brudd på QoS kontrakt

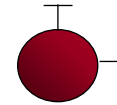
### q Monitorering og adapsjon (tilpassing)



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

17

## Adaptasjonsmellomvare krever åpne implementasjoner



### q Hva er en åpen implementasjon?

q Et system/en komponent med en åpen implementasjon tilbyr (minst) to grensesnitt, et basis-nivå grensesnitt til systemets/komponentens funksjonalitet, og et meta-nivågrensesnitt som avdekker aspekter av hvordan basis-nivå grensesnittet er implementert

- f.eks. i form av komponenter og bindinger mellom dem

### q Adapsjon gjennom åpne implementasjoner

q Meta-nivågrensesnittet tillater endring av aspekter av implementasjonen under kjøretid

- F.eks bytte ut komponenter eller endre deres adferd

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

18

## Case: ADAPT prosjektet

### Lancaster University

“The main aim of the Adapt Project is to investigate the development of distributed systems support to manage the differing levels of connectivity a mobile user will experience. More specifically, the Adapt Project will investigate the required support for adaptive multimedia applications which are capable of intelligently adapting to QoS fluctuations (e.g. through QoS re-negotiation, media scaling, etc).”

Tom Fitzpatrick et al,  
"Supporting Adaptive Multimedia  
Applications through Open Bindings"  
INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO



19

## Case: MULTE prosjektet

### UiT,UiO

#### q Goal:

- q develop flexible and adaptable middleware that supports a broad range of Quality-of-Service (QoS) requirements of distributed multimedia applications.
  - continuous media streams, enhanced interoperable stream multicast for heterogeneous environments, flexible connection management, and constrained latency high throughput

#### q Approach:

- q flexible/(re)configurable protocol system (Da CaPo) as basis for new ORB
- q F. Eliassen, T. Plagemann et al: *QoS management in the MULTE-ORB*, IEEE distributed systems online, March 2002, <http://dsonline.computer.org/middleware/articles/dsonline-MULTE-ORB.html>

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO



20

## ADAPT og MULTE interaksjonsmodell

- q Tre typer interaksjon mellom objekter [RM-ODP]:
  - q operasjonelle grensesnitt
    - klient objekter anroper operasjoner i grensesnittene til tjener objekt (jfr. RPC & RMI)
  - q strømgrensesnitt
    - kontinuerlige strømmer av dataflyter mellom produsent og konsument grensesnitt (jfr. multimedia)
  - q signalgrensesnitt
    - grensesnitt for å produsere eller konsumere sanntidshendelser

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

21



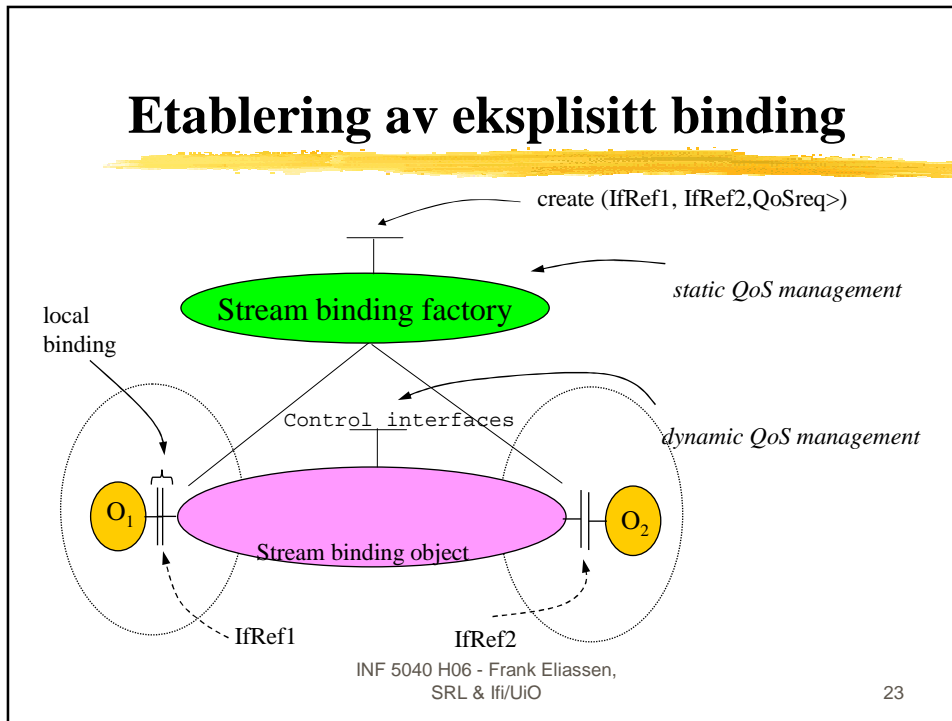
## Eksplisitte bindinger i ADAPT og MULTE

- q Eksplisitte bindinger i form av bindingsobjekt kan opprettes mellom kompatible grensesnitt (i motsetning til CORBA's implisitte bindinger)
  - q tre typer bindinger
    - Operasjonelle bindinger mellom operasjonelle grensesnitt
    - Strømbindinger mellom strømgrensesnitt
    - Signalbindinger mellom signalgrensesnitt
- q Statisk QoS forvaltning (forhandling, adgangskontroll og ressursreservering) innkapsles i operasjonen som oppretter bindingsobjektet
- q Dynamisk QoS forvaltning (monitorering, vedlikehold og kontroll, reforhandling) understøttes gjennom bindingsobjektets kontrollgrensesnitt (dets metanivågrensesnitt)

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

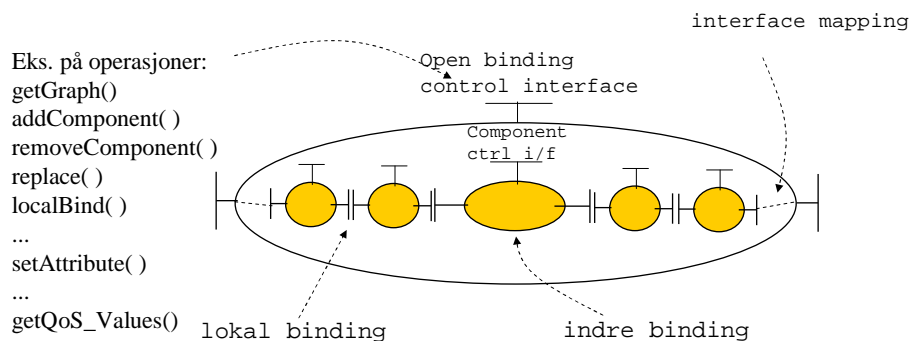
22

## Etablering av eksplisitt binding



## Åpne bindinger

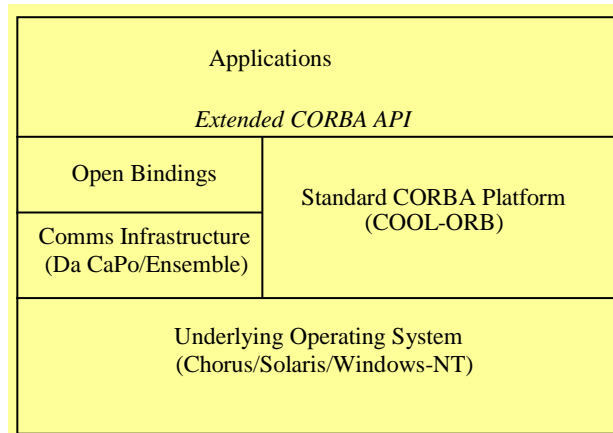
- En binding med en åpen implementasjon. Kontroll-grensesnittet er bindingens metanivågrensesnitt



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

24

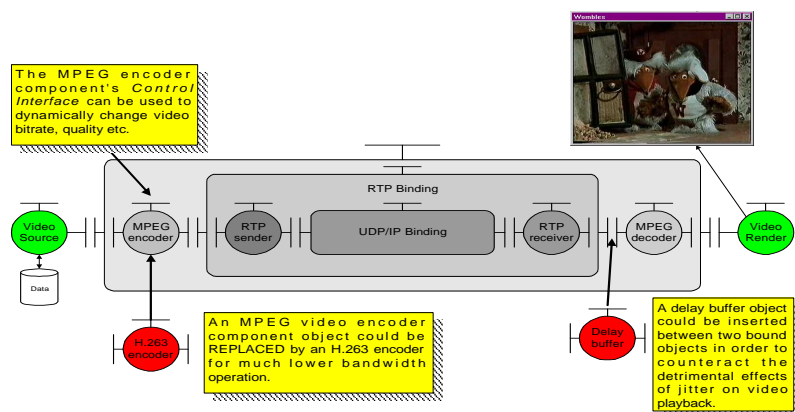
# Implementasjon av Adapt og MULTE plattformer



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

25

# Eksempel åpen binding og adaptasjon



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

26

## Begrensing ved adaptasjon gjennom åpne implementasjoner

- q Resonnering omkring adaptasjon (f.eks. inspeksjon av applikasjonen eller systemet) skjer direkte på implementasjon via meta-nivå-grensesnittet
- q Vanskelig å garantere integritet og konsistens når endringen allerede er gjort (mye kan gå galt)
- q En bedre tilnærming synes å være å resonnerer omkring adaptasjon på en *modell* av applikasjonen/systemet og ikke gjennomføre adaptasjonen før de foreslåtte endringene er kontrollert for korrekthet og integritet
- q Refleksivitet synes å kunne tilfredsstillere en slik tilnærming

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

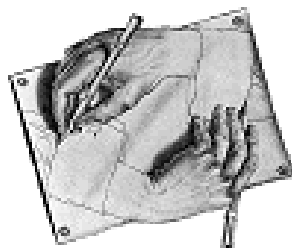
27

## Adaptasjon gjennom refleksivitet

- q Fra åpen implementasjon til refleksivitet

*Reflection* = *Open Implementation*  
+ *Causally Connected Self*  
*Representation (CCSR)*  
+ *Completeness*

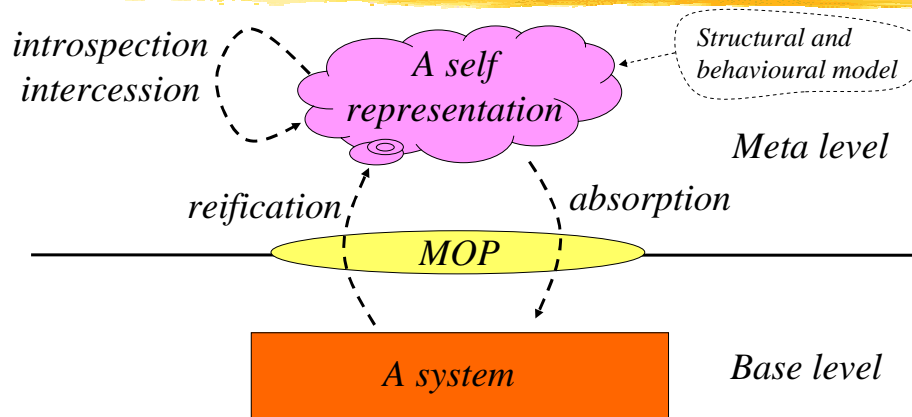
- q Grensesnittet til meta-nivået kalles gjerne *meta-objekt protokollen (MOP)*



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

28

## Reflektivt system: begrepsmessig model



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

29

## Hvorfor refleksivitet?

- q Støtte for introspeksjon (introspection)
  - q Muligheten til å inspisere strukturen eller adferden til et system
    - F.eks. dynamisk monitorering eller "accounting"
- q Støtte for adaptasjon (intercession, absorption)
  - q Dynamisk re-konfigurering (kort tidshorisont)
    - F.eks. endre protokoll konfigurasjon
  - q Evolusjon (lang tidshorisont)
    - F.eks. legge til ny multimediatjeneste

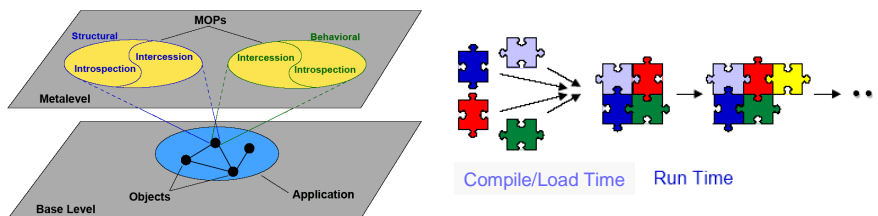


INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

30

# Open-ORBv1 reflektiv mellomvare: kombinerer refleksjon og komponenter

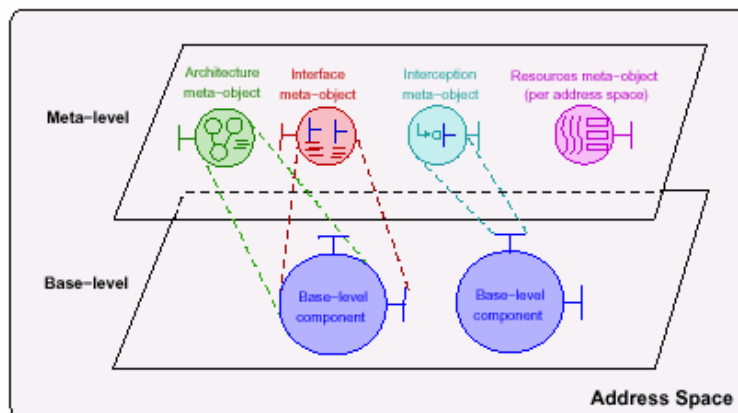
- ☐ Refleksjon
  - ☐ Bruk av refleksjon til å aksessere strukturen og adferden til den underliggende mellomvareplattformen
- ☐ Komponenter
  - ☐ Anvender komponent-orientert programmering på basis- og metanivå



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

31

# Open-ORBv1 Arkitekturen: En multi-model tilnærming



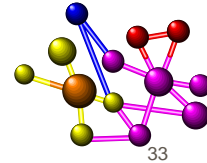
SRL & Ifi/UiO

32



## Strukturell (structural) refleksjon i Open-ORBv1

- Interface meta-model
  - støtter *introspeksjon* og *adapsjon* av den eksterne representasjon av en komponent
    - Required/provided interfaces
- Architectural meta-model
  - støtter *introspeksjon* og *adapsjon* av den underliggende programvarearkitektur
    - Programvarearkitektur = komponentgraf + arkitektoniske begrensninger



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

## Adferdsmessig (behavioural) refleksjon i Open-ORBv1

- Interception meta-model
  - støtter dynamisk innsetting av “interceptors” rundt komponentgrensesnitt
    - Pre-, post- og wrapper- metoder
- Resources meta-model
  - En per adresserom
  - støtter introspeksjon og adapsjon av allokering av ressurser og relatert ressurs forvaltning
    - basert på begreper som “tasks” og “resources”

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

34

## Eksempel på QoS forvaltning i OpenORBv1 [Andersen, 2002]

### Monitoring

**Event Collector** Observe behaviour of underlying functional components and generate relevant QoS events.

**Monitor** Collect QoS events and report abnormal behaviour to interested parties.

### Control

**Strategy Selectors** Select an appropriate adaptation strategy (i.e. strategy activator) based on feedback from monitors.

**Strategy Activators** Implement a particular strategy, e.g. by manipulating component graph.

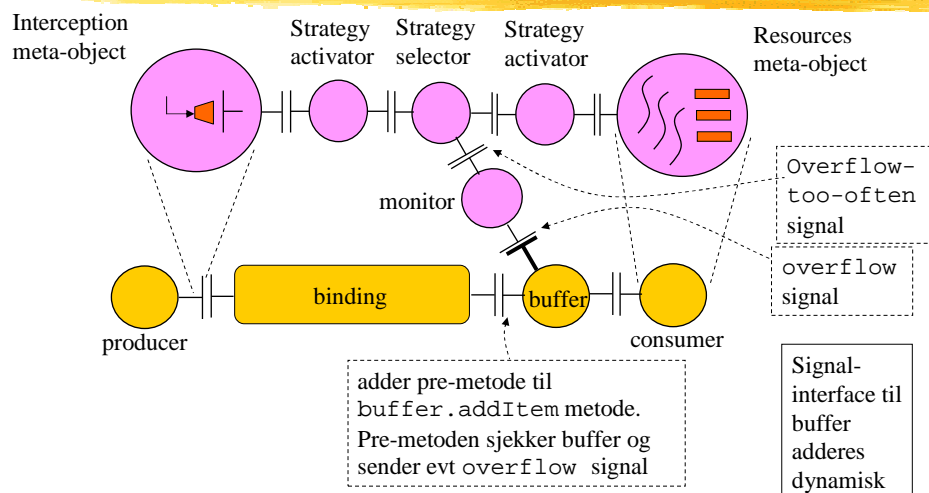
- *Dynamisk innsetting av management komponenter og binding til eksisterende (protokoll)komponenter*
- *Protokollkomponenter adapteres dynamisk til å kunne interagere med management komponenter*

[Andersen,2002] Anders Andersen, "OOPP: A Reflective Middleware Platform including Quality of Service Management", PhD thesis, University of Tromsø, 2002

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

35

## Eksempel på QoS forvaltning i OpenORBv1 (OOPP)



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

36

## Arkitektur-baserte tilnærminger

- q Basert på ideen om ekstern adaptasjonskontrol
  - q Ekstern adaptasjonskontrol krever en passende modell av det adaptive systemet for å resonnerer om det dynamiske adferd
- q Bruk av arkitekturmodeller som beskriver et systems overordnede komposisjon av komponenter, forbindelsene mellom dem, og andre egenskaper av interesse
- q Ideen er å gjøre modeller fra design-tid tilgjengelige for mellomvaren under kjøretid
- q Eksempler på "lokale" systemer ved UiO: QuA, MADAM
- q Skiller seg fra reflektive systemer ved at "absorption" ikke utføres automatisk av det reflektive systemet, men av generelle mellomvare-tjenester

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

37

## QuA approach: Middleware managed adaptation

- q Service request
  - q The user requests the type of service needed (specifying logical functionality) and requirements to non-functional properties (such as QoS).
- q Service planning and quality prediction framework
  - q discovery, context filtering, quality prediction and selection of the service type implementation alternative that best meets the end-users requirements given the current context
  - q based on recursive composition of service mirrors modelling applications and services
- q Adaptation middleware services can be designed based on the planning framework and the reflection facilities.

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

38

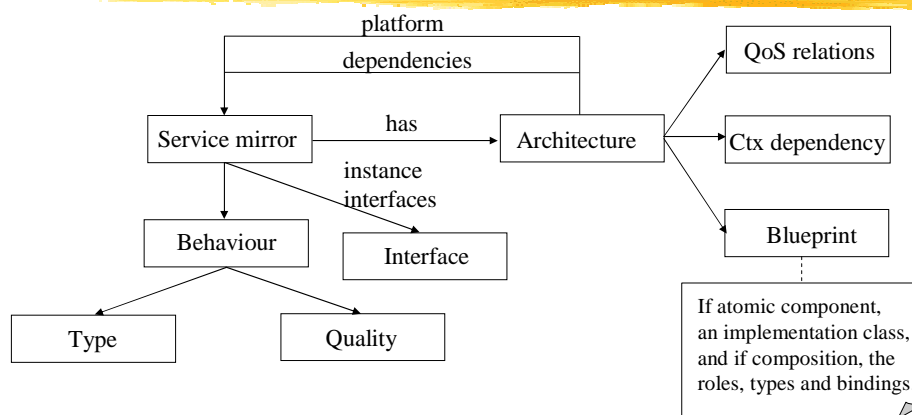
## QuA approach: Middleware managed adaptation

- q Service request
  - q The user requests the type of service needed (specifying logical functionality) and requirements to non-functional properties (such as QoS).
- q Service planning and quality prediction framework
  - q discovery, context filtering, quality prediction and selection of the service type implementation alternative that best meets the end-users requirements given the current context
- q For each service, the middleware provides a *service mirror*, which reflects the behavior, implementation and instantiation of a service
- q Adaptation middleware services can be designed based on the planning framework and the reflection facilities.

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

39

## A service mirror is a reflection of a QuA service, provided by the QuA middleware



- q Mirrors enable reflecting on services both before and during runtime, and facilitate keeping a clean separation between base level and meta level facilities.

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

40

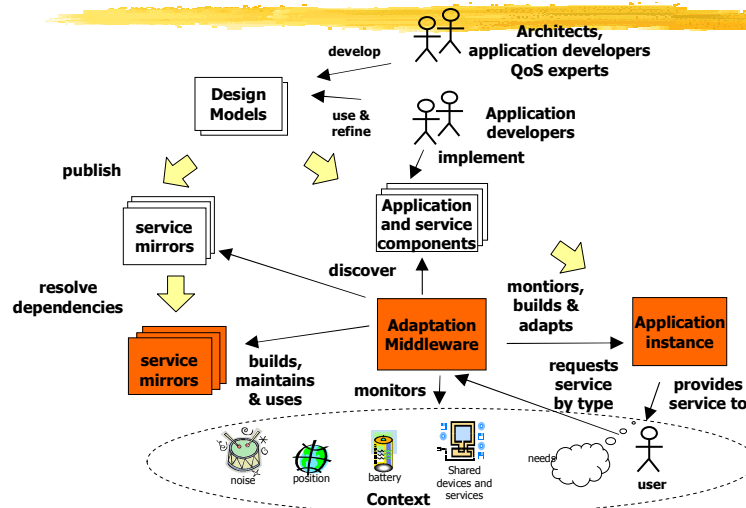
# Generic adaptation algorithm

1. On detecting context changes that may require adaption, (policy dependent) get the runtime model of the running service to be adapted
  2. Unresolve (some or all of) the dependencies of the service mirror of the running service (policy dependent) and perform service replanning on the now partially resolved service mirror
  3. If the result of service replanning imply changes to the running service (policy dependent), determine these changes by comparing the service mirror of the running service and the new mirror that resulted from step 2.
  4. Bring the running application into safe state and enforce the changes (if any) to the base level objects (reconfiguration).
- Like the planning algorithm, the adaptation algorithm uses meta information only.

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

41

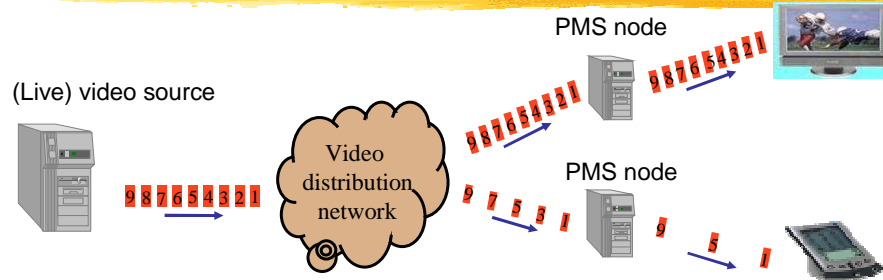
# Oversikt QuA rammeverk



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

42

## Example: the Personal Media Service (PMS)

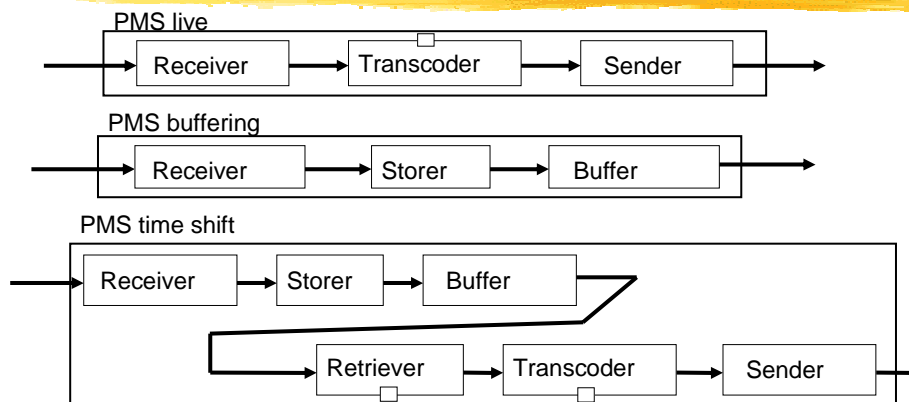


- q Receives a video stream from a provider
- q Transcodes the video feed to a format suitable for decoding by a client device
- q Must handle changing and scarce bandwidth availability, or a transitory loss of connection
- q Supports time-shift of output video stream (output video stream of PMS is temporarily behind input video stream)

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

43

## Implementing the Personal Media Service as an adaptable application using QuA



- q Based on the service planning framework, the adaptation algorithm switches between these three alternative configurations as the bandwidth fluctuates.

INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

44

# Oppsummering

- q Behov for effektiv mellomvare som kan understøtte et vidt spekter av krav inklusive multimedia og mobilitet
- q Løsningen ligger i mer åpne og adaptive arkitekturer
  - q Konfigurerbare
  - q Re-konfigurerbare
  - q Eks: ADAPT (& MULTE)
- q En rolle for refleksjon (eks: OpenORB)
  - q Relatert: arkitekturbaserte tilnærminger (eks: QuA)



INF 5040 H06 - Frank Eliassen,  
SRL & Ifi/UiO

45