

Start av Tau-suiten

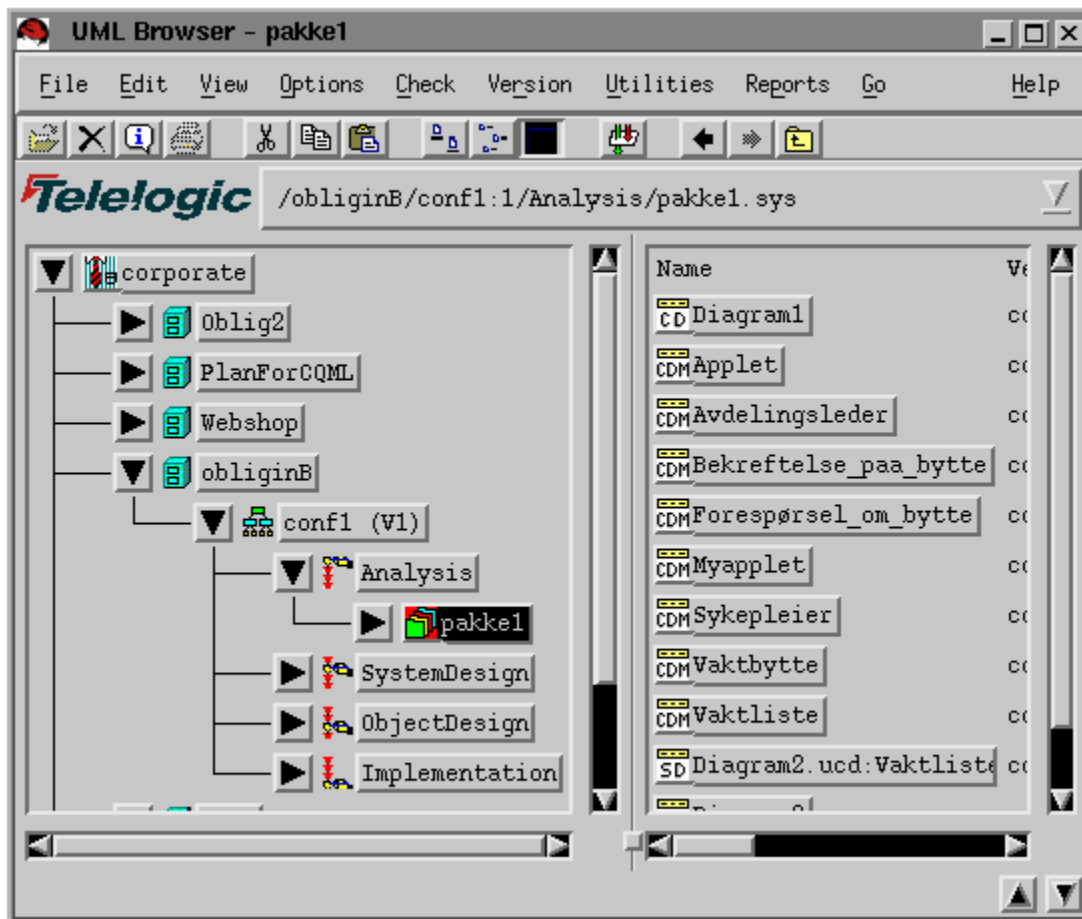
Tidligere har dette CASE-verktøyet blitt benyttet på Sun sine maskiner (solaris). Fra dette semesteret har IFI ikke lenger lisens på Tau-suiten gjennom solaris. I stedet har vi nå lisens på Tau-suiten gjennom Windows-plattformen.

Slik logger du deg inn:

- Man må koble opp mot Windows-serveren "windows.ifi.uio.no" via rdesktop.
- Fra Linux gjøres dette med kommandoen "windows windows"
- Fra Windows-PC'er gjøres dette med "Start -> Accessories -> Communications -> Remote Desktop Connection"
- Se etter Telelogic eller Tau UML suite.

Innføring

Tau-suiten er satt opp slik at alle grupper får tilordnet et prosjekt å jobbe på. Det er kun medlemmene av din prosjektgruppe som kan få tilgang til arbeidet dere har utført. Prosjektene som er laget, har de samme navnene som prosjektgruppene deres, dvs. noe slikt som inf3120pXX, og disse prosjektene finnes i vinduet til venstre på figuren under (under "corporate").



Kort innføring i oppbygning:

For å kunne jobbe med dette verktøyet, så må man ha et prosjekt som er tilordnet seg selv eller en prosjektgruppe. Disse prosjektene blir, som nevnt over, plassert under corporate. Corporate er det objektet som har tilgang til å kommunisere med repository, der all data i alle prosjekter blir lagret.

Når dere starter å jobbe på prosjektet deres, inneholder det ingenting, så dere må bygge opp en struktur (se eksempel-prosjektet med navn obliginB ovenfor). Denne strukturen består av følgende elementer:

conf1 (v1)

Dette nivået i strukturen (under prosjektet obliginB) tar seg av versjonskontroll og parallell utvikling. Her kan ulike brukere i samme prosjekt jobbe på sin egen versjon eller den samme.

4 faser

Tau støtter en utviklingsmodell basert på 4 faser: analyse, system design, objekt design, implementasjon.

Analyse: Denne fasen modellerer forretningssituasjoner; hva systemet skal gjøre, og ikke hvordan. Bruk av klassediagram, sekvensdiagram og bruksmønsterdiagram er mest aktuelt i denne fasen. Analysen skal være lite rettet mot tekniske aspekter ved systemet, som for eksempel programmeringsspråk, plattform, maskinvare etc.

System design: I denne fasen skal man ha fokus på design av applikasjonsarkitektur, identifisere subsystemer og eventuelt tilordne maskinvare til dem, mao. fokusere på hva som trengs for å løse problemet.

Objekt design: Denne fasen konsentrerer seg om to oppgaver: (1) velg UML kodegenerator og (2) gjør pakkeversjoner klar for kodegenerering.

Implementering Gjør ferdig kode som er generert.

Pakker

Om man ser på figuren under prosjektet obliginB, conf1, Analysis, så finner man under Analysis et nivå som heter pakke1. Dette nivået er det stedet hvor alle diagrammer for denne versjonen i analysefasen finnes (til høyre ser man alle diagrammene som tilhører denne pakken).