Modulær testing av komponent-aggregater basert på kontraktsorienterte spesifikasjoner

Ved å kombinere innsikt fra formelle metoder med pragmatiske metoder for systemutvikling, er håpet å finne en effektiv og anvendelig metode for å verifisere komposisjon ved hjelp av testing.

Verifikasjon av komposisjon

På høyeste nivå kan et system spesifiseres som en enhet. Denne spesifikasjonen kan dekomponeres (brytes ned til spesifikasjoner for komponenter). Med verifikasjon av komposisjon menes verifikasjon av at komposisjonen av de nye spesifikasjonene beskriver det samme systemet som den opprinnelige spesifikasjonen.

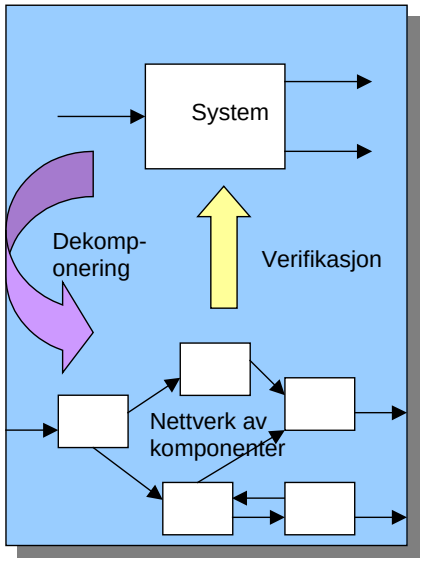
Dagens datamaskinbaserte systemer er i dag ofte store, åpne og distribuerte. Utvikling av systemer vil nesten alltid være modulær, i den forstand at systemet blir delt opp i moduler eller komponenter som spesifiseres og implementeres separat. Utviklingsprosesser består av mange personer som arbeider på forskjellige deler av systemet, og langt fra alle er eksperter på systemutvikling.

Dette stiller krav både til spesifikajons-metoder som brukes under system-utvikling:

- komponenter må spesifiseres på en slik måte at implementasjonen av dem kan skje adskilt fra implementasjon av andre deler av systemet
- spesifikasjonene må være hierarkiske, slik at det er mulig å beskrive systemet på ulikt abstraksjonsnivå
- det må være mulig å sikre konsistens mellom de ulike abstraksjonsnivåene
- spesifikasjonene være mulige å forstå for forskjellige grupper mennesker med ulik kompetanse og arbeidsfelt

Det stiller også krav til verifikasjons-metodene som brukes. Bl. a. er de nødt til å kunne håndtere parallellkomposisjon. Målet med dette arbeidet er å finne en metode for å verifisere komposisjon hvor

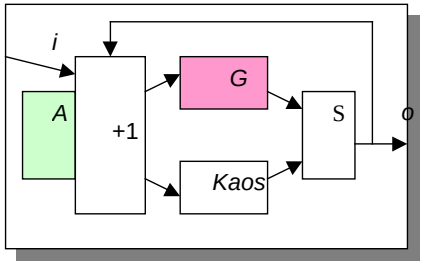
- verifikasjonen gjøre ved hjelp av testing
- det er tatt hensyn til kravene ovenfor



Kontraktspesifikasjoner
Kontraktspesifikasjoner (eller antagelse/-garanti-spesifikasjoner) er spesifikasjoner skrevet som en kontrakt

- en antagelse beskriver den oppførselen komponenten forventer av omgivelsene
- en garanti beskriver komponentens oppførsel dersom omgivelsene har den forventede oppførselen

Kontraktspesifikasjoner har sitt utspring fra formelle metoder, hvor spesifikasjoner skrives i logiske språk. Disse er ofte vanskelige å forstå og lite brukt. Her vil derfor antagelsen og garantien bli spesifisert med tilstandsdiagrammer, og komposisjon spesifisert med dataflyt-diagrammer.



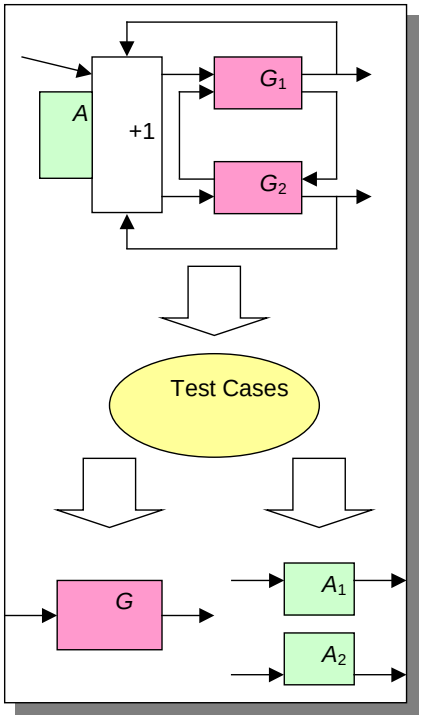
Som semantisk modell for tilstands-diagrammene brukes predikater over uendelige tidsavhengige strømmer av meldinger. Dersom antagelsen til en komponent holder til tidspunkt *t*, må garantien holde til tidspunkt *t*+1.

Semantikken er slik definert at garantien bare er interessant så lenge antagelsen holder og en tidsenhet lenger. Etter det kan man forvente kaotisk oppførsel fra komponenten.

Testing
Verifiaksjon av komposisjon kan gjøres ved hjelp av logiske beviser, men dette er vanskelig og ressurskrevende. Derfor er det et mål å kunne gjøre verifikasjon ved hjelp av konvensjonelle testeteknikker.

En opplagt måte å gjøre testingen på er å gjøre det direkte. Test cases genereres fra et nettverk av spesifikasjoner for kompo-nenter og prøves mot spesifikasjonen for systemet.

Martín Abadi og Leslie Lamport har formulert et komposisjonsprinsipp for det logiske spesifikasjonsspråket Temporal Logic of Actions som forenkler verifikasjon av komposisjon. Inspirert av dette er det mulig å konstruere et annet nettverk for testing av komposisjon.



Her blir kaosoppførselen unngått. ”+1”-semantikken gjør at test casene holder seg innenfor antagelsene ved hjelp av induksjon på tidsenheter. ”+1”-seman-tikken hjelper også med å sirkulariteter under verifikasjon

Dersom denne metoden kan implemen-teres i et testeverktøy, vil det forhåpentlig gi en mer effektiv metode enn direkte testing. Om det i praksis vil virke og bli mer effektivt gjenstår å se.

Mass Soldal Lund,
hovedfagsstudent
Institutt for Informatikk, UiO
SINTEF Tele og Data