

DOKTORAND: Paul Bernhard Beskow
GRAD: Philosophiae doctor
FAKULTET: Det matematisk-naturvitenskapelige fakultet
INSTITUTT: Institutt for informatikk
FAGOMRÅDE: Nettverk og Distribuerte Systemer
VEILEDERE: Carsten Griwodz, Pål Halvorsen
DISPUTASDATO: 4. mars 2013

AVHANDLINGENS TITTEL: *Parallel programming models and run-time system support for interactive multimedia applications*

Ressursbruken til moderne interaktive multimedia applikasjoner øker i takt med forbrukerens forventninger til stadig mer komplekse og intelligent multimediater tjenester. Nye hardware arkitekturer i form av flerkjerneprosessorer bidrar med de nødvendige ressursene, men for utviklere er det mer krevende å utforme parallelle og distribuerte applikasjoner enn deres sekvensielle motpart. Av denne grunn eksisterer det en rekke parallelle programmering modeller og teknikker for å minimere den kognitive belastningen som legges på utvikleren. Der mange av disse løsningene lar en utvikler tenke sekvensielt, men dra nytte av parallell og distribuert utførelse av arbeidsoppgavene. En begrensning i en rekke av disse eksisterende løsningene er deres manglende evne til å uttrykke vilkårlig komplekse arbeidsoppgaver. Prosesseringsgrafene av disse metodene, som uttrykker forholdet mellom beregning og kommunikasjon, er ofte begrenset til rettete asykliske grafer eller forhåndsbestemte løp. Videre er disse eksisterende løsningene ofte bare i stand til å uttrykke et delsett av de tilgjengelige parallelliserings domener som eksisterer. Dette begrenser effektiviteten til disse løsningene i forbindelse med multimedia prosessering, som ofte krever iterativ utførelse og i tillegg inneholder en rekke former for parallellitet. Vi hentet konsepter for utformingen av en programmerings modell og påfølgende kjøresystem fra tidligere arbeid og relatert arbeid og laget P2G, som er i stand til å støtte det nye domenet av interaktive multimedia applikasjoner. Med P2G forbedrer vi eksisterende verktøy og teknikker med støtte for vilkårlig komplekse prosesseringsgrafer, med sykler, forgreninger og tidsfrister, og gir støtte til data, oppgave, pipeline og nøstet parallellitet. Dette er oppnådd ved å automatisere viktige komponenter for parallellisering og distribusjon, slik som partisjonering, planlegging, agglomerering og synkronisering. Dette er bekreftet ved implementasjonen av en prototype som viser anvendbarheten og skalerbarheten av vår løsning.