

PongCorn – et OOP-spil

Projektdeltagere – Peter Tiedemann og Troels C. Damgaard

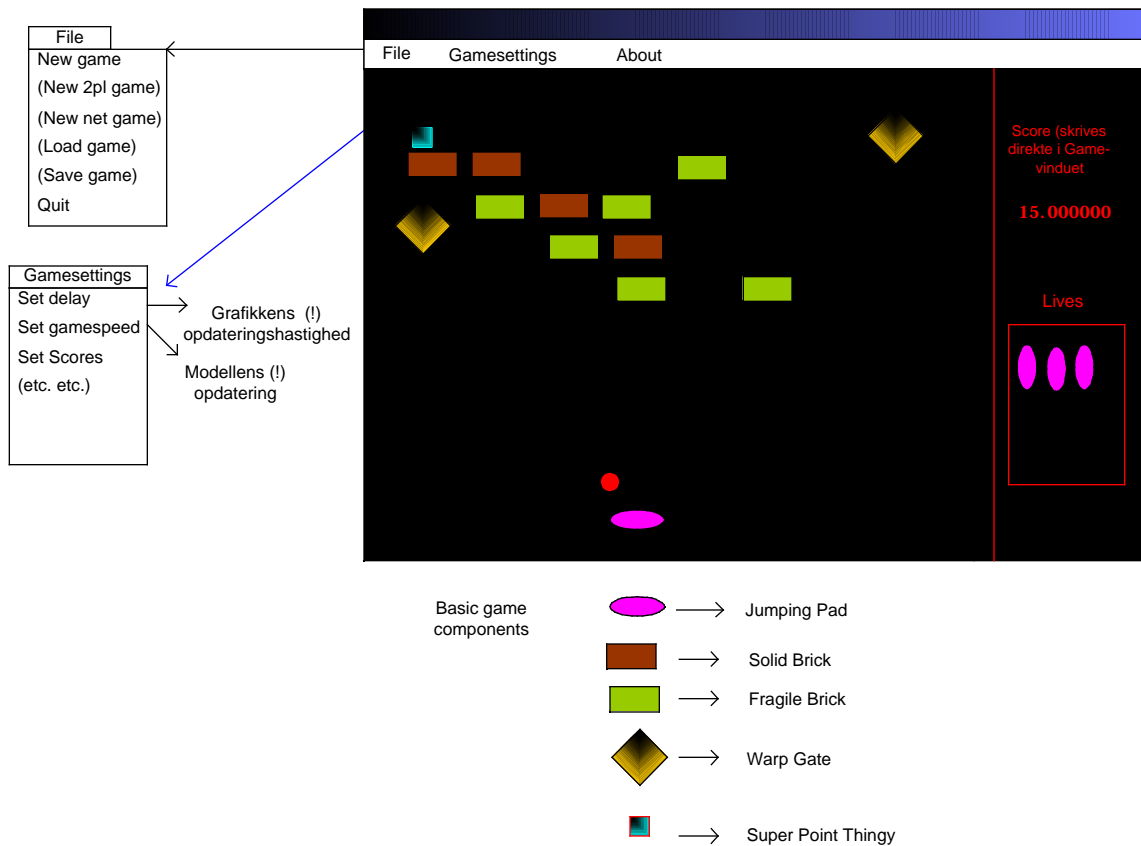
Motivation

Vi har valgt at designe og implementere et lille arkadespil, vi kalder PongCorn. Spil låner sig mange gange godt til objektorienteret design, idet der for det første indgår en række forholdsvis let identificerbare spilobjekter, der er underkastet en række en regler for interaktion. For det andet kræver spil både en grafisk brugergrænseflade, der repræsenterer den underlæggende model. Denne model skal løbende tage højde for input fra spilleren. Vi forsøger faktisk, at basere de overordnede design-beslutninger på Model-View-Controller design-konceptet.

Overblik over spillet

Spilleren styrer en lille 'trampolin' i bunden af et firkantet spilområde og forsøger, ved at bevæge denne horisontalt, at holde en bold fra at ramme bunden. Det gælder om at forsøge at få bolden til at ramme og ødelægge en række genstande, der svæver inden i spilrummet, før spilleren mister alle liv. Der mistes liv, hver gang bolden rammer bunden.

Nedenfor vises en skitse af et spil i aktion med en række spilobjekter angivet (*Basic game components*).



Disse genstande har som det ses forskellige former og, som navnene angiver, forskellige egenskaber. De simpleste og hyppigst forekommende genstande er to typer 'mursten' - *bricks*. Rammer bolden en *brick* reflekteres bolden. Er det en *fragile brick* kan den, afhængig af sin 'skrøbelighed' og antallet af berøringer destrueres og

hastighed nedsættes/forøges temporært, der opstår temporært flere bolde, spillerens *jumping pad* (trampolin) ændrer form (bliver større/mindre eller evt. radikalt anderledes) osv. osv.

Vi forestiller os (som en mulighed) at tilføje endnu et spilelement, idet *fragile bricks* i stedet for direkte at iværksætte spileffekten ved destruktion, kunne kreere (*spawn*) et nyt spilobjekt – et lille flag (med indikation af spileffekttype), der falder mod bunden af spilområdet. Ønsker spilleren den på flaget angivne spileffekt må han 'samle flaget op' med sin *jumping pad*.

Det giver også en række interessante problemstillinger omkring dynamisk kreering af objekter og dynamisk tilføjelse og sletning af spilleregler.

De to andre *game components* – *Warp Gate* og *Super Point Thingy* repræsenterer bare en lille del af de ideer til spilobjekter, vi kunne finde på at introducere i spillet. Deres funktion er vist ret åbenlys ud fra deres navne. En lille note omkring *Warp Gate* er dog, at vi foreløbig regner med bare, at vælge en tilfældig 'udgangsportal' for bolden efter dens teleportation... Derudover er det værd, at pointere at en *Warp Gate* ikke er *fragile*, dvs. den forsvinder ikke ved berøring med bolden. En *Super Point Thingy* derimod forsvinder altid ved første berøring.

Vi er allerede nået et stykke med designet – specielt af den underliggende spilmodel. Præsentationen af denne må dog vente til et senere tidspunkt, for at overholde 2-siders grænsen på denne lille projektbeskrivelse.

Peter Tiedemann
og
Troels C. Damgaard