

36.36

36.36.1

For det første har vi skeleton-koden med interfaces. Derefter er der civilization varianten hotciv som er indenfor hotciv domænet. Collaborating classes er kommunikationen med vores strategies og det er customizable da ikke afhænger af den specifikke implementation af strategies. igen spiller vores strategies ind da de både er et genbrug af designet og vores kompositionelle design lever op til framework karakteristika da vi har defineret et sæt strategies som arbejder sammen ligemeget den konkrete implementation og de kan kombineres som man har lyst til. Alt dette indenfor hotciv domænet der er en invariant af civ.

36.36.2

Frozen:

GameImpl – endOfTurn er et eksempel er et eksempel på frozen spot da den dikterer at blå er den sidste spiller inden en runde ender.

Udover det er vores CityImpl og frozen.

Hot:

strategies – man kan definere nye strategies for world, de forskellige age etc. og dermed tilføje specialiseret kode.

factories – definer hvilke hooks der er.

36.36.3

Vores abstract factory giver en inversion control da man specificerer konkrete implementationer af interfaces(strategy pattern fx getAttackStrategy-pattern) med andre ord man skal blot definere hvilke regler man vil bruge og så lave et nyt spil med en factory der har de regler og derefter klarer game og factory det selv.

36.36.4

separation:

et eksempel på separation kunne være moveUnit i GameImpl som bruger getAttackStrategy som hook interface. Den definerer variansen i kamp mere specifikt EpsilonCivAttack

Unification: vi har ikke nogen abstract methods. <--- **check om det er sandt!**

36.36.5

bemærk at kommentarer er talt med

Frozen code	Linier kode	
Framework + standard(*impl)	628+402=1030	
Variant	Linier kode	% af total
AlphaCiv	177	$(1-177)/(1030+177)=85\%$
BetaCiv	195	$(1-195)/(1030+177+195)=86\%$
GammaCiv	180	$(1-180)/(1030+177+180)=87\%$
DeltaCiv	209	$(1-209)/(1030+177+209)=86\%$
EpsilonCiv	222	$(1-222)/(1030+177+222)=85\%$

ZetaCiv	184	$(1-184)/(1030+177+184)=86\%$
SemiCiv	312	$(1-312)/(1030+177+312)=80\%$

36.45.1

Vi har identificere følgende viability points i MiniDraw:

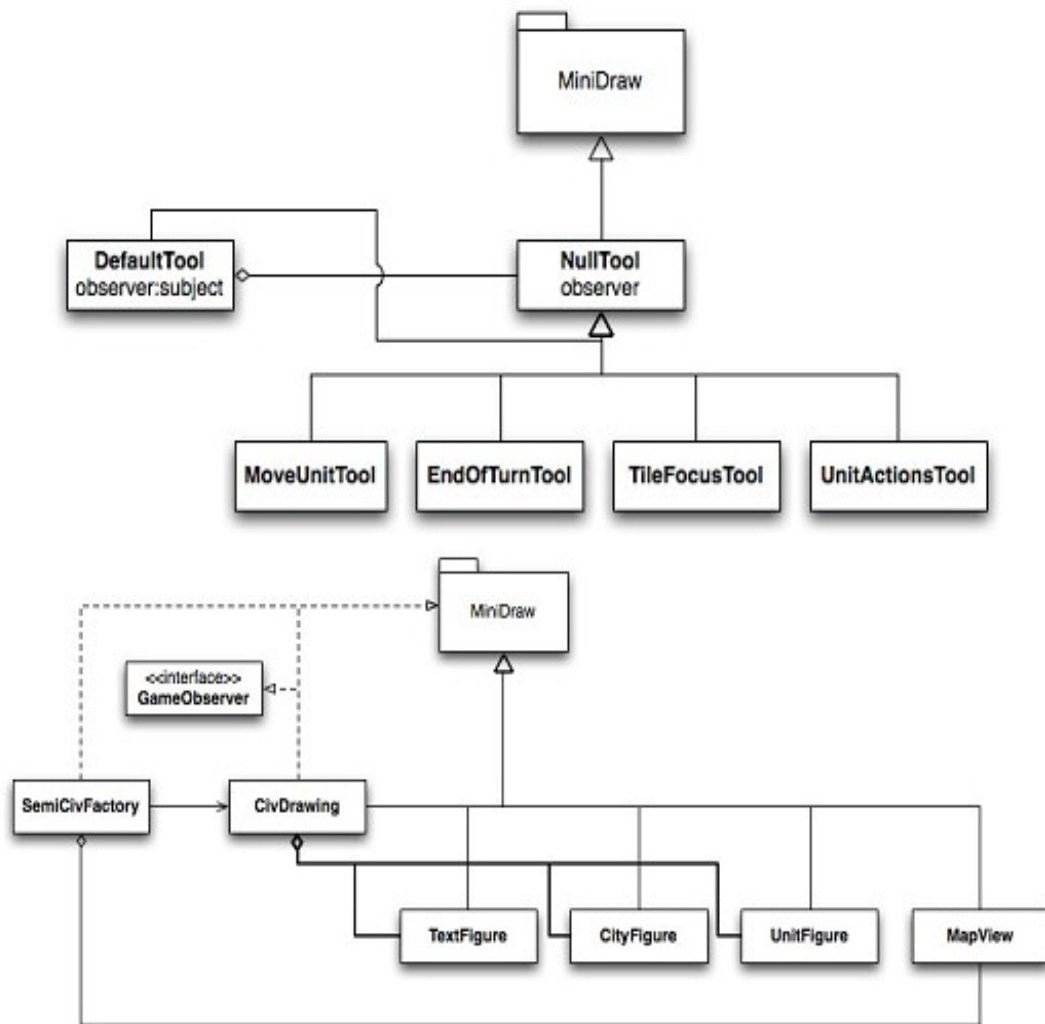
- Tool (NullTool)
- Drawing (Standard Drawing)
- Figure (ImageFigure)

Vi brugte disse til at implementere grafiske instrumentationer af spillet og tools til at interagere med spillet, hvoraf meget var givet på forhånd. Bland disse tools er MoveUnitTools der bruges når enheder skal bevæges på kortet. Yderligere har vi lave tools til "tile focus", "end of turn" og "unit actions" alle sammen samlet i et DefaultTool der er det primære tool.

Problemet med at vælge det rigtige tool afh. Funktionen løste vi ved at bruge Tool interfaces som "observer" og DefaultTool som subject i observer patternet.

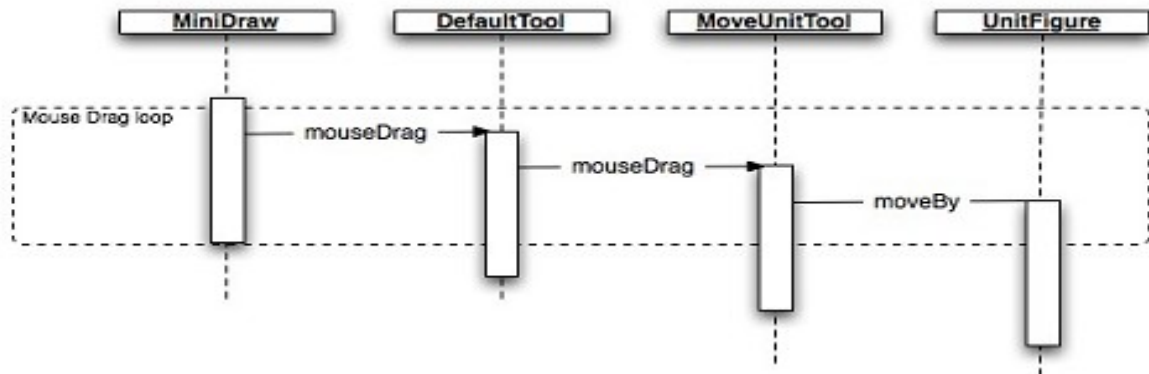
36.45.2

Tools & Observer:

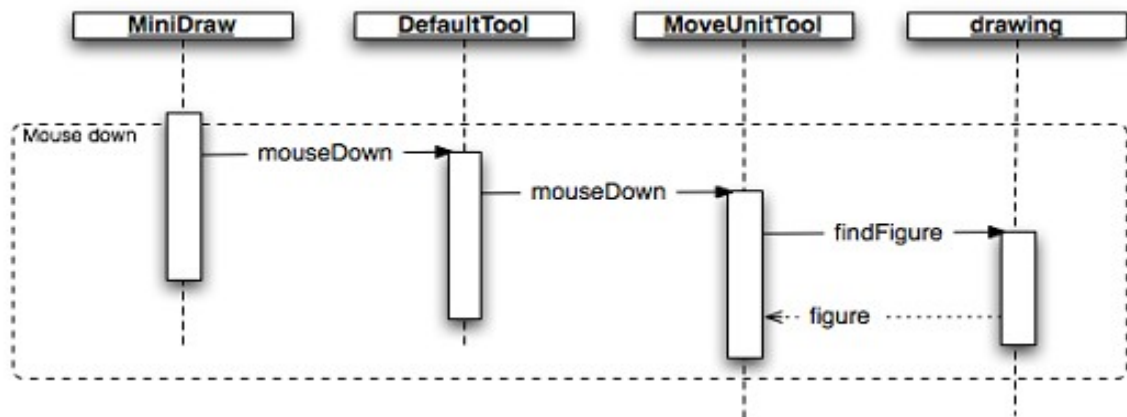


36.45.3

Drag Event



Mouse Down Event



Mouse Up Event

