

Håkon Stavik Rønning

Lyd i dataspill

Et historisk og praktisk perspektiv på strategier for musikalsk komposisjon og lyddesign i dynamiske audiovisuelle multimedier

Masteroppgave i Musikkteknologi

Trondheim, mai 2015

Sammendrag

I denne oppgaven oppsummeres historien til bruken av lyd i dataspill, og det spekuleres basert på denne, samt egne ønsker og erfaringer, omkring fremtiden til en del teknologier som potensielt kan gjøre bruk og produksjon av lyddesign og musikk til dataspill enklere. I tillegg til dette beskrives en del teori knyttet til bruken av lyd i multimedier, deriblant ulike strategier for komposisjon av dynamisk musikk og lyddesign til dataspill. Etter denne teoretiske delen beskrives to ulike spillprosjekter, og arbeidsprosessen med å lage lydeffekter og musikk til disse. Det reflekteres også rundt det helhetlige arbeidet med oppgaven, og hvordan det praktiske arbeidet kan sammenlignes med teorien i den første delen, samt hvilke utfordringer som har oppstått som følge av valget om å samarbeide med to eksterne spillprosjekter. Det viste seg ved ferdigstillingen av denne oppgaven at ingen av de to spillprosjektene har blitt ferdige i løpet av de nesten to årene som samarbeidene har pågått, og det har dermed ikke vært mulig å legge ved spillbare demoer av spillprosjektene med implementert lyd. Det er dermed sannsynlig at disse samarbeidene kan fortsette en god stund etter innleveringen av denne oppgaven, og frem mot spillenes ferdigstilling.

Forord

Denne oppgaven markerer avslutningen på min toårige mastergrad i musikkteknologi ved institutt for musikk ved *Norges teknisk-naturvitenskapelige universitet*(NTNU). Oppgaven er en praktisk-teoretisk oppgave som bygger på erfaringer fra mine fem år på musikkteknologistudiet. Den praktiske delen består av to ulike dataspillprosjekter som er gjennomført i samarbeid med to eksterne parter.

Jeg vil takke Øyvind Brandsegg for veiledning av masteroppgaven, samt Andeas Bergsland og Asbjørn Tiller for teori om lyddesign knyttet til faget *FVP3004 Lyddesign*, og Peter Svensson for teori om akustikk knyttet til fagene *TT3010 Audioteknologi og romakustikk*, og *TTT4170 Audioteknologi*. Jeg vil også takke Sigurd Saue for litteraturtips, og de andre lærerne og elevene ved masterprogrammet for nyttige innspill og tilbakemeldinger. Alle hos Kybernesis, og Ole-Ivar Holte hos Gridmedia, som jeg har jobbet sammen med på den praktiske delen har også vært viktige for gjennomføringen av denne oppgaven.

Håkon Stavik Rønning,
Trondheim, Mai 2015

Innhold

1	Innledning	5
1.1	Målsetting	5
1.2	Avgrensning	6
1.3	Tidligere arbeid	6
1.4	Forklaring og utdyping	7
2	Teori	9
2.1	Historien til lyd i dataspill	9
2.1.1	Avspillingsmetoder	10
2.1.2	Viktige spill og teknologier	11
2.2	Fremtiden til lyd i dataspill	18
2.3	Spillmediet	21
2.3.1	Audiovisuelle multimedier	21
2.3.2	Lyd i virkeligheten versus dataspill	25
2.3.3	Spillplattformer	30
2.4	Spillsjangre og musikk	32
2.4.1	Musikktyper	33
2.4.2	Spillertyper	34
2.4.3	Spillsjangre	35
2.5	Musikk i spill	39
2.5.1	Musikkens funksjon	41
2.5.2	Komposisjonsstrategier	42
2.5.3	Bruk av musikk i dataspill	49
2.6	Lyddesign	52
2.6.1	Klassifisering av lyder	52
2.6.2	Variasjon	55
3	Praktiske prosjekter	59
3.1	<i>Karavan</i>	59
3.1.1	Om Spillet	59

3.1.2	Musikk	62
3.1.3	Lyddesign	66
3.1.4	Implementasjon	69
3.2	<i>Shattered</i>	72
3.2.1	Om Spillet	72
3.2.2	Musikk	73
3.2.3	Lyddesign	76
3.2.4	Implementasjon	78
4	Anvendelse av teori i prosjektene	81
4.1	Anvendelse av teori i <i>Karavan</i>	81
4.1.1	Musikk	81
4.1.2	Lyddesign	82
4.1.3	Plattform	83
4.2	Anvendelse av teori i <i>Shattered</i>	83
4.2.1	Musikk	83
4.2.2	Lyddesign	84
4.2.3	Plattform	85
4.3	<i>Karavan</i> versus <i>Shattered</i>	86
5	Arbeidsprosess	89
5.1	Arbeidsprosess <i>Karavan</i>	89
5.2	Arbeidsprosess <i>Shattered</i>	90
5.3	Arbeidsprosess rapport	90
5.4	Refleksjon rundt arbeidsprosess	91
6	Avslutning	95
6.1	Erfaringer	95
6.2	Videre arbeid <i>Karavan</i>	96
6.3	Videre arbeid <i>Shattered</i>	96
A		99
A.1	Filer	99
A.2	Spill	100
A.3	Konsoller	101

Kapittel 1

Innledning

I denne oppgaven brukes flere ulike referanseformer. For litterære referanser brukes former [3, p. sidetall] med sidetall for konkrete deler, og uten sidetall for hele verker. For fotnoter brukes ¹. For dataspill og konsoller brukes formen Navn(produsent, årstall). Alle spill og litterære referanser er listet til slutt i dokumentet. Vedlegg refereres til som (Vedlegg: Navn på vedlegg).

Delkapitlene 2.1, 2.2, og 2.3.2 er ikke direkte relevante for diskusjonen i den praktiske delen av oppgaven, men gir en historisk kontekst til prosjektene, og bakgrunnsinformasjon om dataspillmediet som kan være av interesse.

1.1 Målsetting

Målet for denne oppgaven er å bedre forstå, og gi et helhetlig bilde av hvordan man i praksis kan bruke lyd i moderne dataspill.

Opgaven er todelt, i form av en teoretisk og historisk del, etterfulgt av en praktisk del. I den teoretiske og historiske delen vil jeg ta for meg historien og utviklingen til bruken av lyd i dataspill for å undersøke hvilke metoder, normer og tradisjoner som har dannet seg i løpet av mediets historie. Jeg vil ta for meg teori knyttet til akustikk, fysikk, lyddesign, og komposisjon av musikk som er relevant for dataspill. I den praktiske delen vil jeg sette denne teorien i praksis og beskrive hvordan jeg har jobbet og tenkt rundt to forskjellige spillprosjekter som jeg har vært med på i løpet av de siste to årene.

¹Eksempel på fotnote

1.2 Avgrensning

Det å jobbe med to spillprosjekter samtidig i den praktiske delen av oppgaven medfører både noen ulemper og noen fordeler. Den største fordelen med å jobbe på flere prosjekter samtidig er at disse kan sammenlignes med hverandre, og at det gir mer erfaring og større muligheter for bruk av teori i praksis. Flere prosjekter gir også en større sikkerhet med tanke på gjennomføring av den praktiske delen, i tilfelle ett av prosjektene skulle falle bort i løpet av arbeidet med oppgaven. Flere prosjekter fører derimot også til en større total arbeidsmengde, noe som gjør at mengden praktisk arbeid per prosjekt blir begrenset noe. Siden begge prosjektene i den praktiske delen av oppgaven har vært i startfasen av produksjonen er det naturlig at disse ikke vil bli helt ferdige før min oppgave skal ferdigstilles. Det er også naturlig at behovene til lyddesign og musikk i spillprosjektene forandrer og utvikler seg i løpet av prosessen, noe som gir noen utfordringer med tanke på å lage musikk og lyddesign såpass tidlig i prosessen. Basert på disse forutsetningene har jeg derfor begrenset musikken og lyddesignet på de to forskjellige prosjektene til det mest nødvendige, som mest sannsynlig holder seg uforandret. Dette arbeidet er foreløpig kun delvis implementert i spillprosjektene siden store deler av spillmekanikken er uferdig. Siden det på bakgrunn av dette ikke kan legges ved spillbare demoer av spillene, er mitt arbeid i prosjektene vist ved hjelp av lydfiler, illustrasjoner, og foreløpige implementeringer i FMOD Studio. Det vil dermed ikke bli gjennomført et totalt lyddesign med implementering til begge prosjektene i løpet av oppgaven.

1.3 Tidligere arbeid

Denne oppgaven bygger på tidligere arbeid jeg har gjort i min bacheloroppgave om lyd til dataspill, samt en rekke fag tatt i løpet av min bachelor og master. Jeg har også tatt inspirasjon fra tidligere arbeid ved instituttet, deriblant arbeidet gjort i sammenheng med dataspillet *Among the Sleep*, hvor Mats Haugland skrev sin masteroppgave om interaktiv musikk [5], og Kristian Mäkinen skrev sin masteroppgave om lyddesign [6]. I tillegg er inspirasjon hentet fra Håkon Nybøs masteroppgave om elektroakustisk filmlyd [8]. Teoretisk bakgrunnsstoff er hentet fra Karen Collins bok *Game Sound*[3], Winifred Phillips bok *A Composer's Guide to Game Music*[9], og Michel Chions bok *Audio-Vision* [2].

(Vedlegg: Bachelor)

1.4 Forklaring og utdyping

Tittelen for oppgaven *Lyd i Dataspill* er i seg selv relativt selvforklarende, men undertittelen *Et historisk og praktisk perspektiv på strategier for musikalsk komposisjon og lyddesign i dynamiske audiovisuelle multimedier* kan kreve litt ekstra utdyping av begreper. For å gjøre dette vil jeg begynne med å bryte ned undertittelen og forklare hvordan jeg definerer og bruker noen av de forskjellige begrepene i denne sammenhengen.

Begrepet dynamikk vil i denne sammenhengen brukes om medier som kan forandre seg basert på forhåndsbestemte regler. Jeg har også valgt å skille mellom to ulike former for dynamikk, og bruker begrepene interaktivitet og adaptivitet i tråd med Karen Collins definisjoner av begrepene[3, p. 4]. Interaktivitet vil i denne sammenhengen være et samspill mellom en bruker og et medium. Et medium vil presentere en bruker med informasjon via et brukergrensesnitt, og bruker vil interagere med dette, og mediet vil gi tilbakemelding til brukeren om en forandring. Adaptivitet brukes om dynamiske forandringer som baserer seg på noe annet enn en direkte interaksjon med mediet. Det som skiller adaptivitet fra interaktivitet er med andre ord at mens interaktivitet trenger en direkte påvirkning fra en bruker for å forandre seg, så vil adaptivitet kunne forandre seg uavhengig eller indirekte av brukeren.

Undertittelen til oppgaven bruker også uttrykket 'audiovisuelle multimedier'. Dette er sammensatt av to velkjente begreper, men en utdyping av disse kan allikevel være nyttig. 'Audiovisuell' brukes som en beskrivelse for noe som er både lydlig og visuelt. 'Multimedier' er medier for formidling av informasjon som er sammensatt av flere former for medier, deriblant tekst, lyd, grafikk, animasjon, og video. 'Audiovisuelle multimedier' er dermed sammensetting av auditive elementer som musikk, lydeffekter, og stemmeskuespill, med visuelle elementer som tekst, grafikk, animasjon og video. Audiovisuelle multimedier er et vidt begrep som omfavner de aller fleste dataspill, filmer, og mye kunst. Grunnen til at jeg har valgt å bruke begrepet audiovisuelle multimedier istedenfor å bare bruke begrepet dataspill er fordi mye av oppgaven gjelder tema som er generelle for alle audiovisuelle multimedier, og ikke bare spesifikt dataspill.

Dataspill er også et velkjent begrep, men kan defineres som et interaktivt multimedium som oppleves på en digital plattform. De aller fleste dataspill, med unntak av bl.a. de aller eldste dataspillene, er audiovisuelle av natur. Det betyr at de består av både auditive elementer som musikk, lydeffekter, og stemmeskuespill, samt visuelle elementer som tekst, grafikk, animasjon,

og video. Det som skiller dataspill fra andre audiovisuelle multimedier er at de er interaktive, og oppleves via en digital plattform. De fleste dataspill har også en form hvor spilleren prøver å oppnå et mål gjennom sine handlinger, og mottar belønning i form av poenger eller annen måloppnåelse.

Lyddesign er kort definert det å skaffe, manipulere, eller generere auditive data. Disse auditive data blir ofte brukt i auditive medier for å formidle informasjon, eller skape en ønsket stemning. Vanlige former for lyddesign er lydeffekter, foley, stemmeskuespill, og musikk.(se 2.6.1) Jeg har valgt å inkludere musikk som en del av lyddesign siden funksjonen ofte er den samme. Det kan argumenteres for at komposisjon av musikk ikke er lyddesign, men at innspillingen og prosesseringen av denne musikken er det.

Undertittelen til oppgaven betyr altså at jeg vil ta for meg hvilke ulike strategier for komposisjon av musikk og lyddesign i dynamiske medier som består av auditive og visuelle elementer, basert på normer og konvensjoner som har dannet seg i løpet av dataspillmediets historie. Jeg vil ta utgangspunkt i historien til bruk av lyd i dataspill og se på hvilke retningslinjer og muligheter dette har skapt for bruk av lyd i moderne dataspill.

Kapittel 2

Teori

2.1 Historien til lyd i dataspill

For å bedre forstå hvordan lyd brukes i moderne dataspill, og hvorfor ulike tradisjoner har oppstått, kan det være lurt å se på historien til bruken av lyd i dataspill. Lyddesign og musikk i dataspill har utviklet seg fra enkle signallyder implementert av programmererne av spillet, til å bli en stor industri med grupper av spesialister som jobber på hvert enkelt prosjekt for å skape en troverdig spillverden på lik linje med en moderne spillefilm. Dataspillindustrien har vokst enormt gjennom de siste tiårene, og har i større grad tiltrukket seg lyddesignere og komponister fra mer tradisjonelle medier som film og musikk, noe som har ført til en betraktelig økning i kvaliteten.

Tilgang på bedre verktøy og en generell forbedring i teknologi har gjort det enklere enn noensinne å lage dataspill. Dette, kombinert med en generasjon som har vokst opp med, og som i større grad velger å jobbe med dataspill, har ført til større konkurranse i et stadig mer mettet marked, og et større fokus på kvalitet. Det at disse verktøyene har blitt så enkle å bruke, samt at man har fått flere plattformer for digital distribusjon, har ført til en slags todeling av spillindustrien. På den ene siden har man de tradisjonelle spillselskapene som har laget spill i flere tiår, har hundrevis av ansatte, og har prosjekter med budsjetter på millioner. På den andre siden har man små grupper, eller enkeltpersoner, som lager spill med minimale budsjetter, og som distribuerer spillene sine uavhengig av de større spillselskapene.

For å få et historisk perspektiv på utviklingen av lyd i dataspill har jeg satt sammen en kronologisk liste over viktige dataspill og teknologier

som har bidratt med nyskapingner og satt standarder for bruken av lyd i dataspill. Jeg har også listet de vanligste avspillingsteknologiene som har preget lyden av dataspill. Jeg har i begrepet dataspill inkludert alle digitale former for interaktive multimedier, deriblant spill designet for personlige datamaskiner, digitale spilleautomater(arkadespill), spillkonsoller, mobiltelefoner, nettbrett, og andre håndholdte spillkonsoller. Jeg har dermed ekskludert elektromekaniske spill som flipperspill, enarmede banditter, og andre analoge spilleautomater. Jeg har også tatt for meg de ulike metodene for avspilling av lyd som er brukt i løpet av mediets historie.

2.1.1 Avspillingsmetoder

I løpet av dataspillmediets historie har teknologien for avspilling og generering av lyd forandret og utviklet seg. Her har jeg kort oppsummert noen av de vanligste teknologiene for avspilling og generering av lyd i dataspill.

Programmerbare lydgeneratorer

Programmerbare lydgeneratorer fungerer ved å skrive data til lydgeneratorens registre, som deretter brukes for å generere komplekse bølgeformer. Ved å kombinere ulike typer bølgeformer med ulike avspillingsfrekvenser og omhyllingskurver kan man skape en stor variasjon av lyder. Denne formen for lydgenerering var vanlig i de tidlige arkade- og konsollplattformene.[3, p. 10]

Subtraktiv syntese

Subtraktiv syntese fungerer ved å fjerne deler av spektrumet til en kompleks lyd ved hjelp av et filter. Subtraktiv syntese var vanlig i analoge synthesizere og programmerbare lydgeneratorer i tidlige arkade- og konsollplattformer. [3, p. 10]

Frekvensmodulasjon

Frekvensmodulasjon baserer seg på å bruke en tonegenerator for å modulere frekvensen til en annen tonegenerator. Ved å koble sammen er rekke tonegeneratorer kan man på denne måten skape svært komplekse og mer realistiske lyder enn ved tidligere teknologier.

Frekvensmodulasjon som lydsyntese ble utviklet av John Chowning ved Stanford University i løpet av 1967-68, og patentert i 1975. Teknologien ble etterhvert lisensiert til Yamaha og det ble utviklet en rekke lydkort som ble tatt i bruk i dataspillmaskiner i løpet av 1980-tallet. [3, p. 10]

Samplebasert syntese

Samplebasert syntese baserer seg på å spille av digitale opptak via en digital til analog konvertering(DAC) til analoge lydsignaler, som deretter kan forsterkes og spilles av. Samplebasert syntese ble tatt i bruk i datamaskiner i løpet av 1980-tallet og har blitt den vanligste formen for lydavspilling i moderne dataspill.[3, p. 11]

Granulær syntese

Granulær syntese er en form for samplebasert syntese, hvor samplene blir delt opp i svært korte mikrosamples kalt korn/grains. Ved å justere avspillingen av disse kornene i forhold til hverandre kan man skape kompliserte lyder og effekter. Granulær syntese kan være veldig ressurskrevende, og er foreløpig ikke spesielt utbredt i dataspill, men kan være et effektivt verktøy for å skape lyder til dataspill, som deretter spilles av som tradisjonelle samples. [3, p. 11]

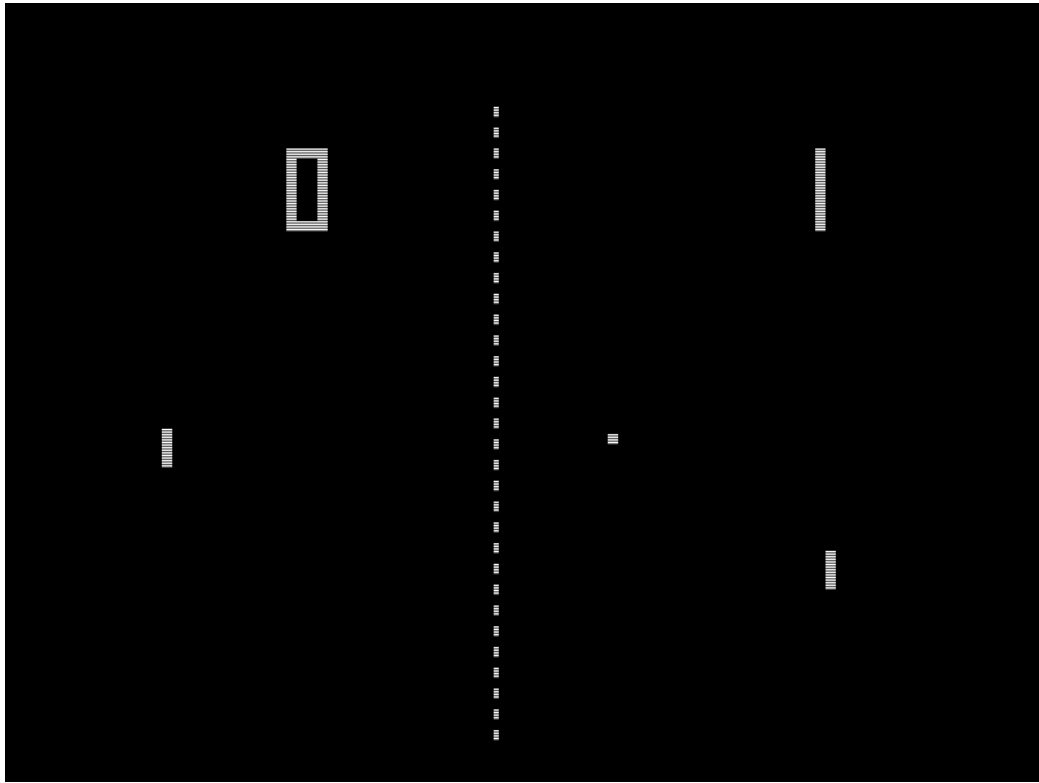
2.1.2 Viktige spill og teknologier

1971 - *Computer Space*(Nutting Associates, 1971)

Computer Space fra Nutting Associates er trolig det første masseproduserte dataspillet noensinne, og det første til å bruke lydeffekter. I spillet styrer man et romskip som har lyder for motorer, raketter, og eksplosjoner. [3, p. 8]

1972 - *Pong*(Atari Inc., 1972)

Tennisspillet *Pong* fra Atari er et av de første populære dataspillene til å bruke lyd. Spillet har svært enkle lydeffekter som signaliserer når ballen spretter i en tennisracket, i vegger, eller går ut av banen og gir poeng til motstanderen. [3, p. 8]



Figur 2.1: *Pong*

1975 - *Western Gun/Gun Fight* (Taito 1975)

Western Gun er trolig det første kommersielle spillet til å bruke musikk. Spillet er et western-skytespill, og bruker i tillegg til lydeffekter for skyting og treff musikk for å signalisere når en spiller har blitt skutt og motstanderen har fått et poeng.¹

1975 - FM syntese

Mellom slutten av 1970-tallet og starten av 1980-tallet blir frekvensmodulasjon (FM) syntese vanligere i dataspill, og gir muligheten til en mer kompleks tonegenerering enn tidligere. [3, p. 10]

¹Western Gun http://en.wikipedia.org/wiki/Gun_Fight sist besøkt 08.05.15

1977 - Atari VCS/2600(Atari Inc. 1977)

I 1977 blir Atari 2600 lansert, og er en av de tidlige hjemmekonsollene. Den har to audiokanaler summet til mono, muligheter for polyfoni, men et begrenset valg av bølgeformkombinasjoner, ulik stemming per kanal, og tungvint implementering som gjør det svært vanskelig å komponere musikk til den. [3, p. 20]



Figur 2.2: Atari 2600

1978 - *Space Invaders*(Taito, 1978)

Space Invaders er det første dataspillet til å bruke sammenhengende bakgrunnsmusikk. Det er også det første dataspillet til å bruke dynamisk musikk ved at det velkjente temaet øker i hastighet synkront med de invaderende romvesenene. [3, p. 12]

1978 - Magnavox Odyssey²(Magnavox, 1978)

Magnavox Odyssey² er en av de tidlige konsollene hvor spillene er lagret på et eksternt medium, noe som gir muligheten til å bytte mellom ulike spill på den samme konsollen. Det er også den første konsollen hvor ulike lydeffekter

og musikk kan lagres på det eksterne mediet, og kan dermed være unike for hvert spill.²

1980 - *Stratavox*(Sun Electronics, 1980) & *Berzerk*(Stern Electronics, 1980)

*Stratavox*³ er trolig det første spillet til å bruke talesyntese for å gi spillkarakterene stemmer, tett etterfulgt av spillet *Berzerk*⁴.



Figur 2.3: *Berzerk*

1980 - *Carnival*(Sega, 1980)

Arkadespillet *Carnival* er ett av de første til å bruke polyfonisk lyd, og kan dermed spille både musikk og lydeffekter samtidig. Spillet bruker programmerbare lydgeneratorer for å spille av lydene. [3, p. 12]

²Magnavox Odyssey² http://en.wikipedia.org/wiki/Magnavox_Odyssey2 sist besøkt 08.05.15

³Stratavox <http://en.wikipedia.org/wiki/Stratovox> sist besøkt 08.05.15

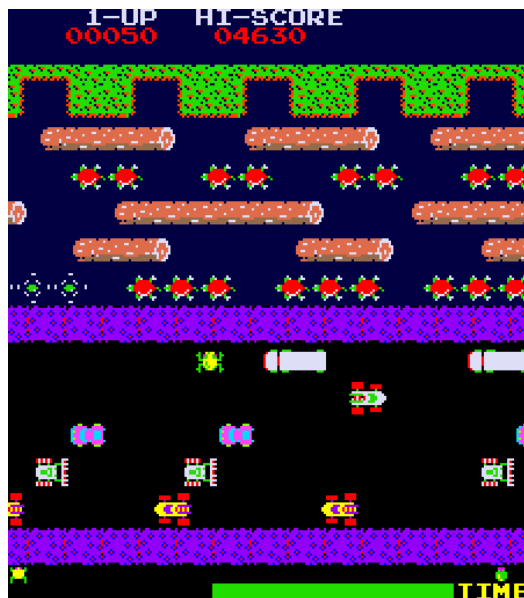
⁴Berzerk [http://en.wikipedia.org/wiki/Berzerk_\(video_game\)](http://en.wikipedia.org/wiki/Berzerk_(video_game)) sist besøkt 08.05.15

1980 - *Rally-X* (Namco, 1980)

Rally-X er trolig det første spillet til å bruke en digital til analog omformer, og kan dermed spille av samplede bølgeformer direkte, istedenfor å generere lyd via tradisjonelle tonegeneratorer. [3, p. 12]

1981 - *Frogger* (Konami, 1981)

Frogger er det første spillet som virkelig utnytter dataspillmediets muligheter for dynamisk musikk med flere ulike musikkspor som forandrer seg basert på spillerens handlinger. [3, p. 19]



Figur 2.4: *Frogger*

1983 - *Dragon's Lair* (Advanced Microcomputer Systems, 1983)

Dragon's Lair er det første arkadespillet til å ta i bruk laserdisk som lagringsmedium, noe som gir det muligheten til å spille av stereofoniske opptak av troverdig stemmeskuespill, effektlyder og foley.⁵

⁵Dragon's Lair http://en.wikipedia.org/wiki/Dragon's_Lair sist besøkt 08.05.15

1983 - MIDI

Musical Instrument Digital Interface (MIDI) blir definert i 1983, og er en effektiv måte for å programmere musikk til dataspill. MIDI baserer seg på kontrolldata som f.eks start, stop, frekvens, volum og instrumentkanal. Siden MIDI kun lagrer kontrolldata, bruker det svært lite minne i forhold til samplede lydfile. I 1991 blir MIDI-standarden oppdatert til General MIDI, som definerer en standardisert rekkefølge av 128 instrumenter og lydeffekter.[3, p. 48]

1984 - *Ballblazer*(LucasFilm Games, 1984)

Spillet *Ballblazer* fra LucasFilm Games er et tidlig eksempel på bruk av algoritmisk musikk i dataspill. Musikkens struktur blir bestemt av en algoritme som setter sammen serier av korte forhåndskomponerte 'riff' med dynamiske parametre for tempo, volum, pauser o.l. basert på vektet sannsynlighet. [3, p. 150]

1987 - MOD

Karsten Obarski skaper det første modulære filformatet MOD som blir brukt for å spille av musikk i dataspill, og det blir raskt standarden for musikk på Amiga konsollen. MOD-formatet består av et sett instrumenter i form av samples, en rekke mønstre for avspilling av sampleinstrumentene, og en liste som bestemmer rekkefølgen til mønstrene. Den første versjonen støttet 16 sampleinstrumenter, men blir etterhvert utvidet til å støtte 32, samt lagring av samples og mønstre i samme fil. Fordelen med MOD-formater over MIDI er at man kan lagre samples av instrumenter, noe som gir komponisten større frihet med tanke på valg av lyder, og sikrer at musikken låter likt på alle systemer.[3, p. 57]

1991 - iMUSE

LucasArts introduserer musikkssystemet Interaktiv Music Streaming Engine (iMUSE) bygget på MIDI i 1991. iMUSE er et system for dynamisk musikk med sømløse overganger mellom musikalske tema basert på hendelser i spillet. iMUSE har på mange måter satt standarden for dynamisk musikk i moderne dataspill. Det første spillet til å bruke iMUSE var *Monkey Island 2: LeChuck's Revenge*. [3, p. 51]



Figur 2.5: *Monkey Island 2: LeChuck's Revenge*

1991 - QSound

I 1990 lager QSound det første systemet for generering av tredimensjonal lyd. I 1991 lager de den første generasjon av sin QSound Chip for 3D-lyd til bruk i Capcoms arkadespill. ⁶

1991 - SNES(Nintendo, 1990)

Nintendo lanserer i 1991 sin konkurrent i 16-bit konsollgenerasjonen, 'Super Nintendo Entertainment System'(SNES). SNES'ens lydmodul inkluderer en 16-bit Sony digital signal prosessor som gir muligheten til å prosessere lyd med ulike effekter i sanntid. Den støtter 8 stereokanaler med programmerbar frekvens, volum og effekter, samt avspilling av MIDI direkte via lydkortet, som gjør implementeringen av musikk mye enklere. [3, p. 45]

⁶QSound <http://www.qsound.com/corporate/chronology.htm> sist besøkt 08.05.15

1994 - Mobil

Hagenuk introduserer i 1994 det første mobilspillet, med en versjon av *Tetris* på sin Hagenuk MT-2000⁷. Tre år senere lager Nokia sitt spill *Snake* som kommer installert på alle deres telefoner, noe som gjør spillet svært populært.[3, p. 78]

2.2 Fremtiden til lyd i dataspill

Basert på historien til lyd i dataspill kan det være verdt å vende blikket fremover, og spekulere litt i fremtiden til bruken av lyd i dataspillmediet. Basert på Moore's lov⁸, kan man forvente at forbedringer i teknologi og maskinkraft vil fortsette en stund fremover. Denne ekstra maskinkraften kombinert med effektiviseringer av programvaren vil gi oss muligheter til å bruke matematiske modeller og algoritmer som tidligere har krevd for mye maskinkraft til å brukes effektivt i sanntid. Disse forbedringene gjelder hovedsakelig forbedringer i fysiske modeller for simulering av lydgenerering, på samme måte som de siste årene har sett en stor forbedring og effektivisering av simulering av lys i sanntid i dataspill. Jeg vil her kort ta for meg noen av disse mulighetene og ønskene jeg ser for meg er relevante for lyddesign i dataspill.

Lydgenerering i sanntid

Allerede nå jobbes det med å generere lydeffekter i sanntid på grunnlag av interaksjon mellom virtuelle objekter basert på deres geometri og materialer. Det er med andre ord en fysisk modellering av hvordan lyd skapes i den virkelige verden. Men økt maskinkraft, effektiviseringer og forbedringer av fysiske modeller og algoritmer vil dette kunne gjøre det enklere å skape realistiske lydeffekter i dataspill, uten bruk av tradisjonelle lydeffekter som er spilt inn på forhånd. En del av dette er forsket på av forskningsgruppen GAMMA⁹ ved *University of North Carolina at Chapel Hill*.

⁷Hagenuk http://en.wikipedia.org/wiki/Hagenuk_MT-2000 sist besøkt 08.05.15

⁸Moore's lov http://en.wikipedia.org/wiki/Moore's_law sist besøkt 08.05.15

⁹GAMMA <http://gamma.cs.unc.edu/research/sound/> sist besøkt 08.05.15

Talesyntese

Foreløpig er ikke talesyntese fra tekst til lyd særlig utbredt i dataspill. Selv om man startet å bruke talesyntese i spill allerede på 1980-tallet, så har den begrensede kvaliteten og ‘robotaktige’ stemmelyden gjort at bruken har vært begrenset, og det har vært enklere å basere seg på stemmeskuespill for å skape troverdige stemmer. Etterhvert som denne teknologien utvikler seg, og kommer til et punkt hvor man kan syntetisere troverdige stemmer, som kan programmeres for å formidle følelser/skuespill, kombinert med en kunstig intelligens som gjør at datamaskinen kan kommunisere med en spiller i sanntid, vil det åpnes opp nye muligheter for å skape troverdige og dynamiske spillkarakterer. Selv om verken talesyntese eller kunstig intelligens i dag er på det stadiet at man kan overbevise en spiller om at den man snakker med er en virkelig person, som er kriteriet for den velkjente Turingtesten¹⁰ for kunstig intelligens, vil en spillers ‘willing suspension of disbelief’(se 2.3.1) kombinert med en troverdig kunstig intelligens og talesyntese kunne bidra til at dataspill får et helt nytt nivå av innlevelse og interaksjon. Selv om talesyntese ikke er så utbredt i dataspill i dag, har det potensialet til å kunne spare enorme summer med penger og tid i forhold til innspilling av stemmeskuespill på store prosjekter, og gjøre det mulig å ha stemmer i mindre spillprosjekter som ellers ville ha måtte basere seg på tekst, og åpner muligheten for en annen form for kommunikasjon med spilleren, siden maskinens del av samtalen ikke trenger å være planlagt på forhånd.

Relatert til talesyntese er også bruken av talegjenkjenning. Dagens talegjenkjenningsteknologi, som brukt i f.eks Apples kunstige intelligens Siri, kan tolke forhåndslagrede kommandoer og konvertere tale til tekst ganske nøyaktig. Ved å kombinere talegjenkjenning og talesyntese med en kunstig intelligens som kan tolke og bruke språk, kan man i fremtiden forhåpentligvis lage spillkarakterer som man kan holde en samtale med. Dette vil være ekstra nyttig i virtuell virkelighet, hvor et brukergrensesnitt for kommunikasjon vil bryte spillerens innlevelse, men vil også være nyttig i mer tradisjonelle dataspill for å øke spillverdenens troverdighet. Begrensningene i dagens teknologier for å gjøre dette mulig ligger stort sett i kvaliteten på den kunstige intelligensen når det gjelder tolkningen av språkets mening, samt troverdigheten til talesyntesen.

¹⁰Turing Test http://en.wikipedia.org/wiki/Turing_test sist besøkt 08.05.15

Klang basert på geometri

Akustikere har lenge brukt matematiske modeller for å regne ut romklangen og frekvensresponsen i et rom basert på rommets geometri og materialer. Med introduksjonen av datamaskiner og 3D-modeller har dette blitt enklere, men nøyaktige utregninger krever fortsatt vesentlig regnekraft. Det er nå mulig å regne ut disse modellene på forhånd, og legge dem til lyder i sanntid ved hjelp av konvolusjon¹¹. På denne måten kan man i dag få en romklang basert på rommets geometri, men denne klangen er kun statisk, og vil ikke endre seg om rommets geometri, materialer, innhold, eller lytterens posisjon forandrer seg. Med nok maskinkraft og effektive algoritmer vil man kunne oppdatere romklangen i spillverdenen i sanntid basert på omgivelsene til lytteren, og dermed skape en større realisme og innlevelse i spillet. Dette vil også være svært nyttig innenfor fagfelter som akustikk og arkitektur.

Musikk

Som dynamiske medier har dataspill noen ekstra utfordringer i forhold til lineære medier som film. Dataspill har ofte ingen fast varighet eller rekkefølge på hendelser, og noen spill varer i teorien så lenge en spiller ønsker å spille. Dette fører til at musikk som er komponert på forhånd, som nødvendigvis er av en begrenset varighet, raskt kan bli oppbrukt og føles repetitiv. En av løsningene på dette problemet er å bruke algoritmer for å komponere musikken basert på spillets handling. Denne teknikken kommer hovedsakelig med to ulemper. Den første ulempen er at det er vanskelig å lage musikk som føles naturlig og organisk ved hjelp av en algoritme, på samme måte som musikk komponert av et menneske. Den andre ulempen er at siden musikken skal komponeres i sanntid, må den også basere seg på virtuelle instrumenter for avspillingen. På grunn av begrenset maskinkraft blir kvaliteten og mengden virtuelle instrumenter som kan brukes i sanntid begrenset. Mer maskinkraft og forbedrede algoritmer for komposisjon, kombinert med mer realistiske og effektive virtuelle instrumenter vil gjøre det enklere å lage algoritmisk musikk for dataspill som oppleves som naturlige og spennende for lytteren. Mer om hvordan denne teknikken kan brukes i kapittel 2.5.2.

¹¹Konvolusjon <http://en.wikipedia.org/wiki/Convolution> sist besøkt 08.05.15

2.3 Spillmediet

2.3.1 Audiovisuelle multimedier

I sin bok *Audio-Vision*[2] definerer Michel Chion noen begreper som har blitt vanlige å bruke for å forklare forholdet mellom lyd og bilde i audiovisuelle medier. Chion har et hovedfokus på spillefilmer, men begrepene er like relevante for bruken av lyd i dataspill.

Synkrese

Et av begrepene han bruker er synkrese[2, p. 63]. Synkrese kan kort forklares som det fenomenet som gjør at to hendelser som skjer samtidig oppleves å ha en sammenheng med hverandre. Det er dette fenomenet som gjør at stemmeskuespill, foley, og effektlyder kan oppleves som troverdige, selv om lyden og bilde originalt ikke har den samme kilden, eller er tatt opp samtidig. For at synkresen skal fungere kreves det at synkroniseringen av bilde og lyd er innenfor en margin som gjør at det oppleves som at det er et naturlig sammenfall mellom kilden på bildet og kilden til lyden. Forskjellige kulturer har ulike marginer for hva som oppleves som troverdig, men en større realisme med tett synkronisering vil generelt oppleves som mer troverdig. Synkrese kan også brukes som et virkemiddel for å koble sammen bilde med lyder på en måte som ikke virker troverdig, men som gir en ekstra mening til innholdet ved å koble sammen to ellers urelaterte hendelser, slik som er tilfellet ved 'Added Value'.

Added value

Added value[2, p. 5] er et begrep Chion bruker for å beskrive når synkresen mellom en lyd og et bilde gir en ekstra mening i forhold til lyden eller bildet hver for seg. Med andre ord så er helheten større enn summen av delene. Dette fenomenet vil med synkrese av relaterte lyder skape en forsterkning eller fokus på en mening som allerede er synlig med bilde alene, men kan også brukes for å knytte sammen bilder med urelaterte lyder for å skape en ny mening. Synkroniserer man en karakter som går med vanlige skritt vil dette gi lytteren ekstra informasjon om karakterens ganglag, tyngde, tempo o.l. Bytter man derimot ut disse skrittene med en urelatert lyd, som f.eks lyden av en sprettende bordtennisball, vil dette gi en absurd og komisk effekt, og lytteren vil begynne å tenke over hva denne lyden sier om karakteren.

Diegese

Begrepet diegese[2, p. 73] brukes for å skille mellom det som finnes innenfor mediets verden, og det som er lagt til utenfor mediets verden. Musikk i filmer og dataspill kan ofte være både diegetisk og ikke-diegetisk. Man kan for eksempel se for seg en scene med en radio som spiller musikk. Dette vil være diegetisk musikk, som kan oppleves av karakterene i scenen. Om det derimot ikke finnes noe i scenen som produserer musikk, og lytteren hører musikk som ikke oppleves av karakterene i scenen, vil dette være et eksempel på ikke-diegetisk musikk. Diegetisk musikk brukes ofte for å skape en troverdig scene, som f.eks at det spilles høy elektronisk musikk i et diskotek, eller muzak¹² i en dagligvarebutikk. Ikke-diegetisk musikk brukes i større grad for å sette stemningen i en scene, eller signalisere om en viss karakter eller tema er relevant for scenen ved hjelp av ledemotiv¹³. Lyder kan gjerne bytte mellom å være diegetiske og ikke-diegetiske for å skape en spesiell effekt.

On/Off screen

Et annet begrep som ofte brukes er 'on screen' og 'off screen'. [2, p. 73] Dette brukes for å beskrive om et objekt i diegesen er i kameraets synsvinkel eller ikke. En radio som spiller musikk, og som er synlig for kameraet vil beskrives som 'on screen'. Om kameraet deretter snur seg vekk fra radioen slik at den ikke lenger er synlig, men musikken kan fortsatt høres, vil den da beskrives som 'off screen', selv om det er den samme radioen, og den samme musikken. Ikke-diegetisk lyd kan derimot ikke beskrives som 'on' eller 'off screen', siden den ikke finnes i scenens univers, og dermed ikke kan sees av kameraet uansett hvor det snur eller beveger seg.

Subjektiv/Objektiv lyd

Om en lyd er subjektiv eller objektiv kan brukes for å beskrive om en lyd oppleves som subjektiv, og kan kun høres av et enkelt lytteperspektiv, eller om lyden er objektiv og kan høres av alle. [2, p. 76] Begrepene intern og ekstern lyd brukes også om det samme, men siden dette kan skape forvirring med tanke på innendørs og utendørs lyd foretrekker jeg å bruke subjektiv og objektiv. Lytteperspektivet kan enten følge kameraet, eller være uavhengig. Begge disse mulighetene kan også kombineres med å ha et objektivt eller

¹²Muzak http://en.wikipedia.org/wiki/Elevator_music sist besøkt 12.05.15

¹³Ledemotiv <http://en.wikipedia.org/wiki/Leitmotif> sist besøkt 08.05.15

subjektivt lyttoperspektiv. Det er med andre ord fire generelle kombinasjoner av lyttoperspektiv og kameraerspektiv:

1. Objektivt og avhengig: Lyttoperspektivet opplever lyder slik de eksisterer i scenen i det punktet kameraet befinner seg.
2. Objektivt og uavhengig: Lyttoperspektivet opplever lyd slik den eksisterer i scenen i et annet punkt enn hvor kameraet befinner seg.
3. Subjektivt og avhengig: Lyttoperspektivet opplever lyd slik den oppleves for en karakter og kameraet følger karakterens perspektiv.
4. Subjektivt og uavhengig: Lyttoperspektivet opplever lyd slik den oppleves for en karakter, men kameraet er uavhengig av karakterens perspektiv.

Suspension of disbelief

Et litterært begrep som også er relevant for lyddesign er det filosofen Samuel Taylor Coleridge kaller ‘The willing suspension of disbelief’¹⁴. Dette begrepet brukes om at man som leser frivillig ser bort ifra usannsynligheter ved det litterære narrative. Det er dette fenomenet som gjør at man kan skape fantasiverdener som er ulik vår egen, men som fortsatt internt gir mening, og kan virke troverdig for en leser. Begrepet har etterhvert som det har blitt brukt om flere medier også omfavnet at man frivillig ser bort ifra mediets begrensninger når man skal dømme troverdigheten til innholdets premisser. En kjent strategi for å oppnå *suspension of disbelief* er å bruke den som opplever mediet sin mangel på kunnskap om et tema til sin fordel ved å f.eks bruke komplisert språk. Det er f.eks vanlig i sci-fi å forklare premisser som i virkeligheten ikke er mulige med kompliserte vitenskapelige begreper, og på denne måten maskere dette logiske bruddet ved å utnytte spillerens mangel på vitenskapelig kunnskap. I fantasy brukes ofte ‘magi’ for å bortforklare premisser som normalt ikke gir mening, på samme måte som vitenskap ofte brukes i sci-fi.

Dataspill er ofte begrensede av maskinvaren de er laget for, og har dermed et begrenset antall muligheter spilleren faktisk kan velge mellom. Dette maskeres ofte ved å gi spilleren en illusjon av valg, som gjør at spilleren føler at man står fritt til å velge, selv om man i praksis har svært få, eller ingen valgmuligheter. Spillere er ofte klar over denne mangelen på valg, men er fortsatt villig til å se bort ifra denne begrensningen med dataspillmediet,

¹⁴Suspension of disbelief http://en.wikipedia.org/wiki/Suspension_of_disbelief
sist besøkt 08.05.15

så lenge det *føles* som om de har et valg. Når det gjelder lyddesign vil også spillerens *suspension of disbelief* bidra til at lydeffekter oppleves som mer troverdig i sammenhengen av mediet, enn om de lyttes til på egenhånd. Lydeffekter trenger med andre ord ikke å være helt realistiske etterligninger av virkeligheten for å oppleves som troverdige. Dette gjelder også lyder hvor kilden i den virtuelle verdenen ikke finnes i den virkelige verdenen, og dermed ikke har noen tilhørende lyd som spilleren har kjennskap til fra det virkelige liv. I slike tilfeller er det viktig at lyden høres ut slik spilleren ser for seg at lyden hadde vært om det var virkelig. Noen av de mest kjente eksemplene på dette er fra sci-fi universet til *Star Wars*. Et av eksemplene på dette er det fiktive våpenet lyssabelen. Denne lyssabelen er basert på et tradisjonelt sverd, men har en plasmastråle istedenfor et blad. Disse våpnene har et svært distinkt lyddesign som brukes konsekvent gjennom serien, noen som gjør at de som kjenner til serien lett kan kjenne igjen lyden av en lyssabel, selv uten bilde eller andre assosiasjoner.

Innlevelse

En av de viktigste grunnene for å få spilleren til å oppnå en *suspension of disbelief* er for å øke graden av innlevelse i spillet. En høyere grad av innlevelse vil gjøre spillopplevelsen bedre, og gjøre innholdet i spillet mer virkningsfullt. Collins tar i *Game Sound* for seg en rekke aspekter knyttet til innlevelse i dataspill, og definerer innlevelse som en minkende kritisk distanse til det som vises, og en økende emosjonell tilknytning til det som skjer. Graden av innlevelse kan dermed variere fra en mild interesse for spillet til en total innlevelse hvor skillet mellom spilleren i den virkelige verden og karakteren i spillverdenen viskes ut. Innlevelsen i spill er sjeldent statisk, og kan variere i løpet av spillingen avhengig av spillerens innstilling til spillet, spillets handling, og distraksjoner fra tekniske feil eller omgivelsene. Total innlevelse er også sjeldent over lengre tid, og er delvis knyttet til fenomenet *flow* (se 2.5.1). Økt innlevelse er også mye av motivasjonen bak teknologier knyttet til virtuelle virkelighet slik som VR-headset som totalt dekker spillerens synsfelt, bruker hodetelefoner for lyd, og som oversetter spillerens hodebevegelser til kamerabevegelser i spillet. Ved å simulere spillverdenen på samme måte som man opplever den virkelige verden kan man enklere oppnå en høy grad av innlevelse og tilstedeværelse i spillverden og inngi en større grad av *suspension of disbelief* fra spilleren.

For å oppnå en høy grad av innlevelse spiller lyden en viktig rolle. En av måtene lyd brukes på for å øke innlevelsen er å maskere andre lyder

fra omgivelsene som kan distrahere fra spillet, og dermed bryte innlevelsen ved å trekke spilleren tilbake til den virkelige verden. Det er også viktig at lyddesignet virker troverdig innad i spillverdenen, og at det ikke er harde kutt i musikken eller andre tekniske feil som trekker oppmerksomheten vekk fra spilllets handling. Hvordan diegetiske lyder oppfører seg i det tredimensjonale rommet vil også påvirke graden av innlevelse. Den auditive simuleringen av spillverden vil ha en økt troverdighet jo nærmere denne er hvordan vi opplever lyder i den virkelige verden. Jo nærmere samsvar det er mellom lyden i spillet, og den virkelige verden, jo mindre grad av *suspension of disbelief* kreves av spilleren for å oppnå en høy innlevelse i spillet. Ved virtuell virkelighet, og tredimensjonale spill generelt, er det dermed viktig at lydenes plassering og nivå samsvarer med det visuelle for ikke å bryte spillerens innlevelse.

2.3.2 Lyd i virkeligheten versus dataspill

For å forstå hvordan lyd i dataspill fungerer er det en fordel å se på hvordan lyd fungerer i den virkelige verden, og hvordan dataspill prøver å emulere dette, samt hvordan den menneskelige hørselen fungerer. Når lydbølger beveger seg fra en lydkilde, gjennom et medium, til ørene hos en lytter vil en rekke faktorer spille inn på hvordan lytteren opplever lydkildens avstand, posisjon, og hastighet i forhold til seg selv. I tillegg vil lyden kunne si noe om omgivelsene lyden har beveget seg igjennom.

Avstand

Når lydbølger beveger seg gjennom et medium vil lydbølgene møte motstand og miste energi. Dette fører til at en lydkilde som er langt unna vil oppleves med mindre lydstyrke enn en tilsvarende lydkilde som er nærmere lytteren. I luft er denne sammenhengen lineær, det vil si at en lydkilde som er dobbelt så langt unna vil oppleves som halvparten så lydsterk.

Høyere frekvenser vil også miste energi raskere enn lavere frekvenser, noe som fører til at en lydkilde som er lengre unna vil oppleves som å ha mindre diskant enn en tilsvarende lydkilde som er nærmere.

Basert på disse fenomenene lærer mennesker seg å bedømme avstanden til lydkilder. For å kunne bedømme avstanden til lydkilden nøyaktig forutsetter det at man har kjennskap til lydkilden, og har hørt denne i ulike avstander tidligere slik at hjernen kan gjenkjenne lydkilden, og deretter sammenligne den opplevde lyden med den forventede lyden. Basert på forskjellen i nivå

og diskant mellom den opplevde og forventede lyden kan lytteren forsøke å bedømme avstanden.

For at mennesker skal greie å nøyaktig bedømme avstanden på denne måten forutsetter det også et visst forhold mellom lydkildens avstand og nivå. En lydkilde med et lavt nivå som er langt unna vil være for vanskelig å høre til å kjenne igjen og bedømme nøyaktig, mens en lydkilde som har et veldig høyt nivå og er veldig nærme vil risikere å skape et skadelig høyt lydtrykk for lytteren.

Lokalisering

En av de viktigste grunnene til at mennesker har to ører er for å kunne bedømme retningen til en lydkilde i forhold til seg selv. Denne egenskapen har vært evolusjonært svært viktig for å kunne jakte på andre dyr, samt for å unngå rovdyr. Dette har ført til at de fleste pattedyr har blitt svært flinke til å lokalisere lydkilder basert på hørselen.

Egenskapen å lokalisere en lydkilde ved å sammenligne to ulike lyttepunkter, som ørene våre er, baserer seg hovedsakelig på to fenomener. Det første, som forklart over, er at det opplevde lydnivået er avhengig av avstanden fra lydkilden til lytteren. Ved å sammenligne ørene, som er plassert med litt mellomrom, kan man bestemme hvilket av disse som er nærmest lydkilden, og i hvor stor grad.

Det andre fenomenet baserer seg på at en lydkilde som er langt unna vil bruke lengre tid på å nå frem til en lytter enn en lydkilde som er nærmere. Ved å ha to ører med litt avstand vil det øret som er lengst unna lydkilden registrere lyden litt etter det andre, som er nærmere lydkilden, og denne tidsforskjellen kan brukes for å gi informasjon om vinkelen mellom lydkilden og lytteren.

Basert på disse forskjellen i nivå og tid kan man altså ved å sammenligne to ulike ører bestemme hvilket som er nærmest, og i hvor stor grad det er nærmere, som igjen kan oversettes til lydkildens vinkel i forhold til lytteren. Denne teknikken kan plassere en lydkilde fra høyre til venstre i et plan, men den sier ingenting om lydkildens høydeplassering. Ørenes form, samt refleksjoner fra skuldre kan gi mennesker litt informasjon om lydkildens høyde, men mennesker er generelt svært dårlige til å bedømme den vertikale posisjonen til en lydkilde kun basert på hørsel, og bruker i større grad synet til dette.

Dopplereffekten

Dopplereffekten er et fenomen som oppstår når en lydkilde er i bevegelse i forhold til lytteren. Når en lydkilde beveger seg mot en lytter vil det punktet hvor hver enkelt lydbølge sendes ut fra lydkilden være litt nærmere lytteren for hver bølge, noe som fører til at nyere bølger begynner å ta igjen de tidligere bølgene sendt ut av lydkilden. Dette fører til at avstanden mellom bølgetoppene blir kortere, noe som tilsvarer en kortere bølgelengde, og dermed en høyere frekvens. Når en lydkilde beveger seg vekk fra en lytter vil det motsatte skje, og avstanden mellom bølgetoppene som kommer frem til lytteren blir lengre, noe som fører til en lavere frekvens. Det vil si at en lydkilde som beveger seg mot en lytter vil oppleves som å være transponert lysere i frekvensspekteret, mens en lydkilde som beveger seg vekk fra lytteren vil oppleves som å være transponert mørkere i frekvensspekteret. Graden av transponering vil være avhengig av hastigheten, opp til hastigheten når lyd hastigheten. Idet hastigheten overskrider lyd hastigheten vil lydkilden enten oppleves som et kraftig smell grunnet at alle lydbølgene tar igjen hverandre om lydkilden beveger seg mot lytteren, eller som stillhet om lydkilden beveger seg vekk fra lytteren grunnet at lydbølgene i sum beveger seg raskere vekk fra enn imot lytteren. Et velkjent eksempel på dopplereffekten er når en ambulanse med sirener kjører raskt forbi, og sirenen oppleves som å forandre tonehøyde.

Refleksjoner

Når en lydbølge beveger seg fra et medium til et annet vil lydbølgen bli delvis reflektert tilbake i det mediet det kom fra og delvis bli absorbert av det mediet det beveger seg til. Forholdet mellom mediene vil avgjøre hvor mye av lydbølgen som blir reflektert og hvor mye som blir absorbert. I et lukket rom vil den delen av lydbølgen som blir reflektert gå tilbake inn i rommet, og man kan se på det som en kopi av den originale lydkilden plassert i det punktet hvor refleksjonen inntraff, men med noe dempet energi. Denne prosessen fortsetter helt til all energien i lydbølgene blir borte. En enkel lydkilde som sender lyd ut i alle retninger i et lukket rom vil dermed raskt fylle rommet med et utallig antall refleksjoner fra alle retninger. Siden antallet refleksjoner er så høyt, og tiden mellom dem så kort, oppleves de for mennesker som en enkelt lyd som fyller rommet. Det er dette som er rommets klang.

Rommets størrelse og dimensjoner vil avgjøre hvor langt lyden må reise mellom hver refleksjon, noe som avgjør tiden mellom hver refleksjon. Jo lengre

tid mellom hver refleksjon, jo lengre tid bruker refleksjonene på å bruke opp all energien sin, og jo lengre tid tar det før klangen dør ut i rommet. Det er derfor større rom har en lengre romklang enn mindre rom, gitt at de har det samme forholdet mellom reflektert og absorbert lyd.

Materialet og tykkelsen til rommets tak, gulv, og vegger vil avgjøre hvilket forhold mellom refleksjon og absorpsjon rommet har, og vil dermed påvirke rommets klang på samme måte som rommets størrelse. Om det meste av lyden i rommet blir reflektert istedenfor absorbert vil refleksjonene ha mye energi, og det vil ta lengre tid før energien blir brukt opp, og klangen i rommet vil vare lengre. I motsetning vil et rom laget av et materiale hvor det meste av energien i lydbølgene blir absorbert og ikke reflektert vil refleksjonene ha lite energi, og romklangen vil vare kortere. Man kan med andre ord ha et stort rom, som absorberer mye, og dermed har en kort romklang, og et lite rom som reflekterer mye, og dermed har en lang romklang.

Når en lytter hører en lydkilde i et rom med klang, vil klangen blande seg med den direkte lyden fra lydkilden og vil kunne maskere lyden og gjøre den utydelig. Siden lyden av romklangen vil holde seg relativt jevn i hele rommet, men lyden direkte fra lydkilden vil dempes med avstand, kan lytteren bruke forholdet mellom direktelyden og klangen til å bidra med å bedømme avstanden til lydkilden. Klangen kan også gjøre at lytteren hører klangen før den direkte lyden, og dermed kan høre lydkilden før den ville blitt hørt om det ikke var for klangen. Dette er tilfellet om lydkilden er bak en stor gjenstand, eller bak et hjørne i en gang i forhold til lytteren. Om man beveger seg innover i en lang svingete hule av hard stein vil man for eksempel høre klangen fra hulen lenge før man kommer til lydkilden og hører den direkte.

Virkelighet versus dataspill

På samme måte som en realistisk grafikk vil føre til at spillet oppleves som mer virkelighetsnært, vil en troverdig lyd være viktig for å få spilleren til å leve seg inn i spillets verden. Spillets lyd vil nok for de fleste spillere påvirke opplevelsen mer subtilt enn spillets grafikk, siden mennesker generelt er veldig visuelle. De aller fleste vil enkelt bedømme om grafikken til et dataspill er realistisk eller ikke, og hvorfor, mens med lyd er det for de fleste mye vanskeligere å bedømme hvorfor lyden oppleves som troverdig eller ikke. Dette betyr ikke at det auditive er mindre viktig enn det visuelle for å skape troverdighet, men at det kan være vanskeligere å sette ord på hvorfor en lyd kan oppleves som mer troverdig enn en annen. Noe av grunnen er muligens

at mange har et større ordforråd og trening når det gjelder det visuelle, og mindre når det gjelder det auditive.

Det er også viktig å skille mellom hva som er realistisk og hva som oppleves som troverdig. Man har ofte andre forventninger til lyd og grafikk enn det som er realistisk. Grunnet normer opparbeidet innen spill og film gjennom flere tiår har man blitt opplært til hvordan noe skal se eller høres ut. Lyder har ofte blitt svært overdrevet for dramatisk effekt, men siden de har blitt gjentatt så mange ganger har de blitt etablert som den nye normen. Dette fører til at en faktisk realistisk fremstilling oppleves som nedtonet og dempet i forholdt til forventningene. Denne nye formen for overdrevet realisme kan vi kalle hyperrealisme,¹⁵ for å skille den fra en virkelighetsnær realisme. I likhet med hvordan dette begrepet brukes i andre medier som tegning, maleri, og skulpturer for å beskrive fremstillingen som er 'mer realistisk enn realismen', kan det også brukes om lyddesign som oppleves som en overdrivelse av virkeligheten. Denne hyperrealismen innebærer også imaginære hendelser som ikke kan oppleves i virkeligheten, men som oppleves som troverdige innenfor det fiksjonelle universet til en film eller spill.

Implementasjon

Implementeringen av lyder i et dataspill skjer som regel via en spillmotor som også har ansvaret for spillets fysikk, grafikk, animasjon o.l. Eksempler på populære spillmotorer er Unity¹⁶ og Unreal Engine¹⁷. Spillmotoren baserer lyden på lydobjekter, og et lytteobjekt. Lydobjektene er de objektene i spillverdenen som gir ifra seg lyd. Lytteobjektet er det objekter som fungerer som spillerens ører i spillverdenen, og er ofte knyttet til kameraet som fungerer som spillerens øyne i spillverdenen, slik at det er sammenheng mellom det spilleren ser, og det spilleren hører. Basert på plasseringen og bevegelse av lytteobjektet og de ulike lydobjektene vil spillmotoren bestemme lydnivåer, filtrering av frekvenser, plassering i stereobildet, dopplereffekt samt klang fra omgivelsene til alle lydkildene som er innenfor lytteobjektets akustiske horisont.

For å etterligne hvordan menneskets ører filtrerer lyd basert på lokasjon, kan man bruke en teknikk som kalles *head related transfer function* (HRTF)¹⁸ for

¹⁵Hyperrealisme <http://no.wikipedia.org/wiki/Hyperrealisme> sist besøkt 09.05.15

¹⁶Unity <https://unity3d.com> sist besøkt 08.05.15

¹⁷Unreal Engine <https://www.unrealengine.com> sist besøkt 08.05.15

¹⁸HRTF http://en.wikipedia.org/wiki/Head-related_transfer_function sist besøkt 08.05.15

å skape binaural audio. Denne teknikken kan skape en økt troverdighet av 3D-lyd når den brukes over hodetelefoner, da den gjenskaper den naturlige filtreringen som skjer akustisk i lydens møte med lytterens hode og ører. Den fungerer imidlertid dårlig over høyttalere, siden den da kommer i tillegg til lytterens egne ører. Teknikken fungerer med å konvolvare kildelyden med en impulsrespons, som ofte er spilt inn med bruk av et kunstig hode. Man bruker et sett med impulsrespons, samsvarende med de forskjellige vinklene lyden kan komme fra. Teknikken fungerer bedre jo nærmere samsvar det er mellom lytterens faktiske hode og hodet brukt til å generere impulsresponsene.

Implementeringen av lyd i et dataspill kan også gjøres ved hjelp av en egen lydmotor, ofte kalt *middleware*¹⁹. Det finnes en rekke ulike *middleware* for lyd beregnet på dataspill. Blant de mer populære løsningene er FMOD²⁰ og Wwise²¹. Dette er programmer som gir tilleggsfunksjoner som kan brukes sammen med spillmotoren, og forenkler implementeringen av lyd i spillet. *Middleware* gjør det enklere for en lyddesigner å jobbe uavhengig av resten av spillteamet, og gjør det mulig å gjøre endringer i lyddesignet uten å forandre på implementeringen i spillmotoren. *Middleware* for lyd har også flere muligheter og funksjoner enn det som finnes internt i spillmotoren, og gjør det enklere å skape et komplekst og dynamisk lyddesign. *Middleware* for lyd følger også ofte normer fra annen lyd- og musikkprogramvare, noe som gjør det enklere å overføre erfaring fra disse.

2.3.3 Spillplattformer

Dataspill finnes på en rekke ulike plattformer, som alle har sine fordeler og ulemper. Ulike plattformer stiller ulike krav med tanke på brukergrensesnitt og innhold, og de ulike plattformene har forskjellige muligheter og begrensninger med tanke på maskinvare. Når man utvikler et dataspill er det lurt å ta i betraktning hvilke plattformer spillet skal lanseres på, slik at man kan tilpasse innholdet til plattformens muligheter og begrensninger. De ulike plattformene som er vanlige for dataspill kan grovt deles inn i noen få ulike kategorier:

¹⁹Middleware <http://en.wikipedia.org/wiki/Middleware> sist besøkt 09.05.15

²⁰FMOD <http://www.fmod.org> sist besøkt 08.05.15

²¹Wwise <https://www.audiokinetic.com/products/wwise/> sist besøkt 08.05.15

Arkade

Arkader var tidlig i historien til dataspill den enkleste måten å oppleve dataspill på. Arkader har den fordelen at maskinvaren kan skreddersys til ett enkelt spill, noe som gir de optimal ytelse og muligheten til å ha originale og utradisjonelle brukerkontroller. Et eksempel på dette er arkader med kjørespill som bruker ratt og pedaler som kontroller.

Konsoll

Etterhvert som spillkonsoller har blitt bedre og billigere har de tatt over store deler av det tidligere arkademarkedet. Spillkonsoller har den fordelen at de kan bytte spill og kontroller, og dermed kan gi opplevelser tilsvarende det å ha flere arkader hjemme i sin egen stue. Konsoller har som regel en standardisert maskinvare, som gjør at spilldesignere kan optimalisere sine spill for denne maskinvaren. Dette medfører at maskinvaren som brukes i konsoller ofte er noen år gammel idet spillene lages. Dette er grunnen til at spillkonsoller ofte kommer i 'generasjoner' med noen års mellomrom.

PC

PC-er har muligheten til å oppgradere enkelte komponenter uavhengig av hverandre, og kan dermed oppgraderes så snart nye komponenter er tilgjengelige. Dette fører til at PC er den plattformen med nyest og kraftigst maskinvare. Siden det dermed er begrenset med standardisering er det vanskelig å optimalisere spill på tvers av alle PC-er, noe som kan føre til en del svinn av maskinkraft i forhold til andre plattformer.

Håndholdt

Håndholdte plattformer har ofte begrenset maskinvare i forhold til de andre plattformene, og ligger som regel en 'konsollgenerasjon' eller to bak de andre plattformene med tanke på regnekraft og minne. Håndholdte plattformer har ofte integrert kontroller og skjerm, eller baserer seg på trykkskjerm. Bruken av trykkskjerm har ofte ført til innovativ bruk av spillkontroll og nyskapende spillkonsepter som har gjort mobile spill svært populære.

Lyd på ulike plattformer

Når det gjelder bruken av lyd i dataspill på ulike plattformer er det som regel maskinvaren som setter begrensninger. De største utfordringene er mengden regnekraft og minne, samt maskinvare for avspilling av lyd. Maskinens regnekraft og minne setter fortsatt noen begrensninger på effektprosessering i sanntid, men det er ellers uproblematisk med tredimensjonal lyd og avspilling av flere parallelle lydspor på de aller fleste moderne plattformer. Når det gjelder høyttalere, hodetelefoner, og forsterkere er det liten standardisering, og resultatet kan variere betraktelig avhengig av hvilken maskinvare som brukes. Håndholdte plattformer har ofte innebygde høyttalere som har en begrenset eller ujevn frekvensrespons og minimal eller ingen stereobredde. Medfølgende hodetelefoner er også ofte av svært lav kvalitet, noe som vil begrense den lydlige opplevelsen.

Håndholdte plattformer brukes ofte i situasjoner hvor spilleren ikke har muligheten til, eller lyst til å høre spillets lyd. Spill laget for håndholdte plattformer burde derfor lages slik at de er spillbare uten lyd, noe som kan gjøre spill hvor vesentlige spillmekanikker baserer seg på lyd mindre egnet.

2.4 Spillsjangre og musikk

Etterhvert som det har blitt laget flere dataspill har det blitt naturlig å gruppere disse basert på likheter i stil. Det har dannet seg en rekke distinkte hovedsjangre av dataspill, med mange ulike undersjangre. Disse sjangrene kommer ofte med sin egen historie og konvensjoner for bruk av lyd og musikk. Denne inndelingen av spill i sjangre gjør det enklere for spillere å finne spill de liker, og reduserer læringskurven for nye spill som følger normene til sjangrene med tanke på kontroller og spillmekanikk.

Når man skal lage lyddesign til et spill er det lurt å være bevisst spillets sjanger, og hvilke konvensjoner og tradisjoner som følger med, slik at man ikke ubevisst bryter disse, og skaper en dårlig kombinasjon av det visuelle og auditive. Man kan også bruke disse konvensjonene til sin fordel ved å bevisst bryte disse for å oppnå en ønsket effekt.

Ulike spillsjangre har ofte musikk i bestemte musikk sjangre som er populære i den målgruppen spillet er laget med tanke på. For å finne ut hvilken musikk som kan passe i et spill kan man basert på spillets sjanger finne ut hvilke spillertyper den er mest populær hos, og hvilke musikktyper som er

mest populære hos disse spillertypene. Man burde selvfølgelig ta høyde for sammenhengen mellom typen musikk og spillets innhold, og ikke basere seg kun på målgruppens musikksmak.

Winifred Phillips har i sin bok *A Composer's Guide to Game Music*[9, p. 75] undersøkt sammenhengen mellom spillertyper, spillsjangre, og musikksjangre. Jeg skal her oppsummere hennes funn. En mer detaljert beskrivelse kan finnes i hennes bok.

Phillips baserer seg hovedsakelig på en studie av Chis Bateman og Richard Boon fra boken *21st Century Game Design*[1], hvor spillere deles inn i fire typer, hvor hver av disse kan være av typen 'dedikert' eller 'avslappet'. Disse spillertypene knyttes deretter til 11 ulike spillsjangre i varierende grad. I en annen studie knyttes tilsvarende personlighetstyper opp mot generell musikksmak. Ved å sammenligne disse studiene forsøker Phillips å undersøke sammenhengen mellom spillertyper, spillsjangre og musikkpreferanser. For å forstå sammenhengen og konteksten mellom disse ulike typene må vi først forklare hvordan undersøkelsene definerer de ulike spillertypene, spillsjangrene, og musikktypene.

2.4.1 Musikktyper

Rock

Rock omfatter her alle undersjangre av rock, metall, punk o.l. Spillere som foretrekker *rock* er i undersøkelsen karakterisert som mer introvert enn ekstrovert.

Urban

Urban brukes som et samlebegrep for hip-hop, rap, soul, og r&b. Spillere som foretrekker *urban* er i undersøkelsen karakterisert som mer ekstrovert enn introvert.

Pop/dance

Pop/dance brukes om moderne populærmusikk og elektronisk dansemusikk. Spillere som foretrekker *pop/dance* er i undersøkelsen karakterisert som mer ekstrovert enn introvert.

Elite

Elite samler sjangre som blir sett på som 'høykultur', deriblant jazz, klassisk/orkestral, og religiøs musikk. Spillere som foretrekker *elite* er i undersøkelsen karakterisert som mer introvert enn ekstrovert.

2.4.2 Spillertyper

Erobreren

Erobreren er en spillertype som spiller for å 'slå' spillet, og oppnå så mye som mulig. Musikktypene som er mest populære hos *erobreren* er *rock* og *elite*. *Erobreren* er introvert, og liker å planlegge.

Deltageren

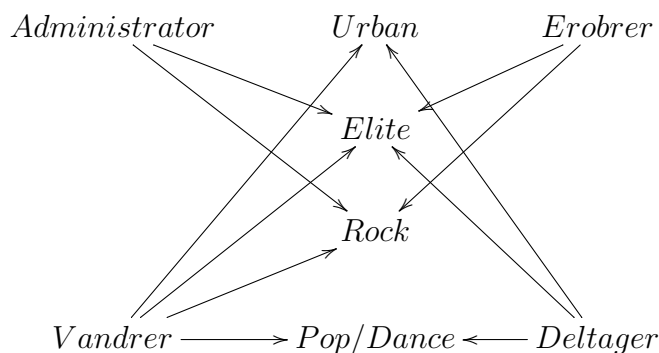
Deltageren er en sosial og emosjonell type, som spiller for å oppleve interaksjon med andre, samt fengende karakterer. Musikktypen mest populær hos *deltakeren* er *urban*, *pop/dance*, og *elite*. *Deltageren* er ekstrovert, og foretrekker å planlegge.

Administratoren

For *administratoren* er ikke målet å 'slå' spillet, men å lære alle ferdighetene som er nødvendig for å holde oversikt og kontroll over spillet. Musikktypene som er mest populære hos *administratoren* er *rock* og *elite*. *Administratoren* er introvert og liker ikke å planlegge.

Vandreren

For *vandreren* handler det ikke om å vinne eller lære alle ferdigheter, men om å utforske og fordype seg inn i spillverdenen mest mulig. *Vandreren* er åpen for alle kategoriene av musikk. *Vandreren* er ekstrovert og liker ikke å planlegge.



Figur 2.6: Relasjoner mellom spillertyper og musikktyper

Dedikert

Dedikerte spillere spiller mye spill, spiller i lengre perioder, og er villig til å lære seg kompliserte ferdigheter. De ser også på spill som en del av sin identitet.

Avslappet

Avslappede spillere spiller færre spill, spiller i kortere perioder, foretrekker spill som er mindre kompliserte, og ser på spill mer som en hobby enn som en livsstil.

2.4.3 Spillsjangre

Skytespill

Skytespill er som navnet tilsier spill hvor den viktigste spillmekanikken går ut på å skyte gjenstander eller fiender. Spillertypen Bateman og Boon mener har størst samsvar med skytespill er *erobreren*. *Erobreren* ønsker å beseire alle fiendene i spillet, noen som er det viktigste målet i de fleste skytespill. Når man ser på musikken i moderne skytespill ser man at disse ofte bruker episk orkester- og kormusikk kombinert med elementer fra rock og metall med tunge elgitarer og slagverk, noe som stemmer bra med *erobreren*s musikkpreferanser som er *elite* og *rock*.

Plattformspill

Plattformspill handler ofte om å komme seg igjennom en bane fra et startpunkt til et slutt punkt ved å unngå hindringer i form av fiender, feller, fallgroper o.l. Ifølge studien er plattformspill mest populære hos spillertypen *vandrer*. Studien mener at *vandrer* er åpen for alle de definerte musikktypene, noen som kan forklare den store variasjonen av musikk i plattformspill. Siden plattformspill ikke er begrenset til noen spesielle omgivelser, er musikken trolig i større grad knyttet til spillens miljø enn selve sjangeren.

Eventyrspill

Eventyrspill baserer seg på utforskning, gåteløsning, og en fengende historiefortelling. Eventyrspill har også ofte elementer av spenning, hvor spilleren må unngå farer eller beseire fiender. Spillertypen tettst knyttet til eventyrspill er *administratoren*. *Administratoren* foretrekker spill med et litt lavere tempo, hvor man ofte må huske store mengder informasjon og tenke logisk for å løse gåter. Bateman og Boon mener *administratoren* foretrekker musikktypene *rock* og *elite*, med andre ord de samme som *erobrer*, men når man ser på bruken av musikk i eventyrspill ser man at det er nesten utelukkende i kategorien *elite*. Dette har kanskje sammenheng med at eventyrspill ofte har et lavere tempo, noen som passer bedre med orkestral musikk enn rock.

Rollespill

Rollespill kan klassifiseres i to ulike kategorier, vestlige rollespill, og østlige rollespill. Begge sjangrene er basert på tidlige bordrollespill som *Dungeons & Dragons*²², men har utviklet seg parallelt i vesten (hovedsakelig USA) og østen (hovedsakelig Japan og Sør-Korea). Rollespill handler om å styre en karakter eller gruppe karakterer for å løse oppdrag og forbedre karakteren ved å skaffe bedre utstyr og samle erfaring for å øke ferdighetene til karakteren. Østlige rollespill har ofte et større fokus på historiefortelling og sosial interaksjon, mens vestlige rollespill fokuserer mer på spenning og forbedring av karakterene.

²²D&D <http://dnd.wizards.com> sist besøkt 08.05.15

Begge formene for rollespill tiltrekker *erobreren*, som er drevet av å skape en best mulig karakter og løse alle oppdragene. Vestlige rollespill er også populære hos *administratoren*, som liker kompleksiteten til forbedringssystemet. Østlige rollespill er derimot mer populære hos *deltageren*, som har et større fokus på historie og sosial interaksjon. Basert på dette er det ikke overraskende å se at det er vanlig med orkestral musikk og rock, som er felles for *erobreren* og *administratoren*, i vestlige rollespill, mens østlige rollespill i større grad trekker inn musikk av typen *pop/dance*, som populært hos *deltageren*.

Skrekkspill

Skrekkspill er spill hvor hovedmålet i spillet er å overleve mens man blir jaktet av skumle fiender. Skrekkspill har som regel en mørk atmosfære, og begrensede resurser tilgjengelig for spilleren. Spilleren styrer ofte en karakter med lite eller ingen trening i sniking eller slåssing, og fiendene har ofte et overnaturlig opphav, som demoner, zombier, spøkelser eller lignende. Spillertypen studien knytter til skrekkspill er *vandreren*. *Vandreren* er åpen for alle musikktypene, men skrekkspill brukes hovedsakelig musikk som passer med spillets mørke stemning. *Elite* og *rock* er dermed de vanligste typene musikk, med mørk orkestermusikk eller rock og metall.

Kjørespill

Kjørespill er spill som handler om kjøring/kappkjøring i forskjellige former, fra biler til romskip. Hovedfokus i kjørespill er hastighet og ferdighet. Disse spillene appellerer til avslappede spillere av alle spillertypene, med unntak av *deltageren*. Basert på dette ville man tro at kjørespill har en stor variasjon i musikktyper, men historien viser at de har en lang tradisjon med å bruke kjapp elektronisk dansemusikk. Denne formen for musikk brukes for å gi spilleren en følelse av hastighet og spenning. Dette valget av musikktype er stikk i strid med hva man skulle tro basert på spillertypene, men viser at spillsjangerens tradisjon og musikkens funksjon kan være viktigere enn spillerens musikkpreferanse.

Simulatorer og livssimulatorer

I simulatorspill har spilleren kontroll over en verden og dens innbyggere. Spillet emulerer en verden hvor spilleren ofte har gudelignende makt. Spillertypen som er vanligst for simulatorer er *administratoren*. *Administratoren*

liker muligheten til å holde oversikt over, og kontrollere en hel verden. De fleste simulatorspill har musikk av typene *rock* og *elite*, noe som passer godt med *administratorens* musikkpreferanser.

En undersjanger av simulatorspillene er livssimulatoren. Disse spillene er mer intime og personlige enn andre simulatorer, og fokuserer på velferden til enkeltindivider i motsetning til større grupper. Disse spillene har sjelden noen form for kamp, og handler om den sosiale interaksjonen med karakterene. Spillertypene som vanligst spiller denne formen for spill er *deltageren*. *Deltageren* er åpen for de fleste musikktyper, med unntak av *rock*. Det er dermed ikke overraskende at livssimulatorer som regel har vennlig og positiv musikk av typene *elite* og *pop/dance*.

Strategi

I likhet med simulatorspill har spilleren i strategispill kontroll over en verden med dens innbyggere og ressurser. I motsetning til andre simulatorer er derimot målet i strategispill å beseire motstandere og bygge seg et størst mulig imperium. Strategispill har dermed et større fokus på seier og ferdigheter. Det er dermed naturlig at *administratoren* er den vanligste spillertypen som spiller strategispill. I motsetning til vanlige simulatorer tiltrekker ikke strategispill *deltageren* eller avslappede spiller i stor grad, og musikken i strategispill er dermed hovedsakelig av typene *elite* og *rock*.

Gåte

Gåtespill er som navnet tilsier spill hvor målet er å løse gåter. Det finnes flere ulike former for gåtespill, og de opptrer ofte som elementer i andre spillsjangre. Gåtespill fungerer som hjernetrim, og tiltrekker seg spillere av flere spillertyper, deriblant *administratoren*, *vandreren*, og *deltageren*. Den spillergruppen som minst tiltrekkes av gåtespill er *erobreren*. Siden gåtespillene har en såpass stor variasjon i spillertyper, er det ingen tradisjon for spesifikke musikktyper, og det er viktigere at musikken passer stilen i hvert enkelt spill.

Slåssespill

Slåssespill handler om å beseire fiender i nærkamp ved hjelp av ulike angrep. Disse spillene er svært ferdighetsbaserte, og krever at spilleren velger rett

kombinasjon av angrep og forsvar til rett tid for å gjøre det bra. Slåssespill tiltrekker nesten utelukkende *erobreren*, og hovedsakelig av den dedikerte typen. Slåssespill bruker nesten utelukkende musikk av typen *rock*, noe som samsvarer bra med musikkpreferansen til *erobreren*, samt innholdet i spillet.

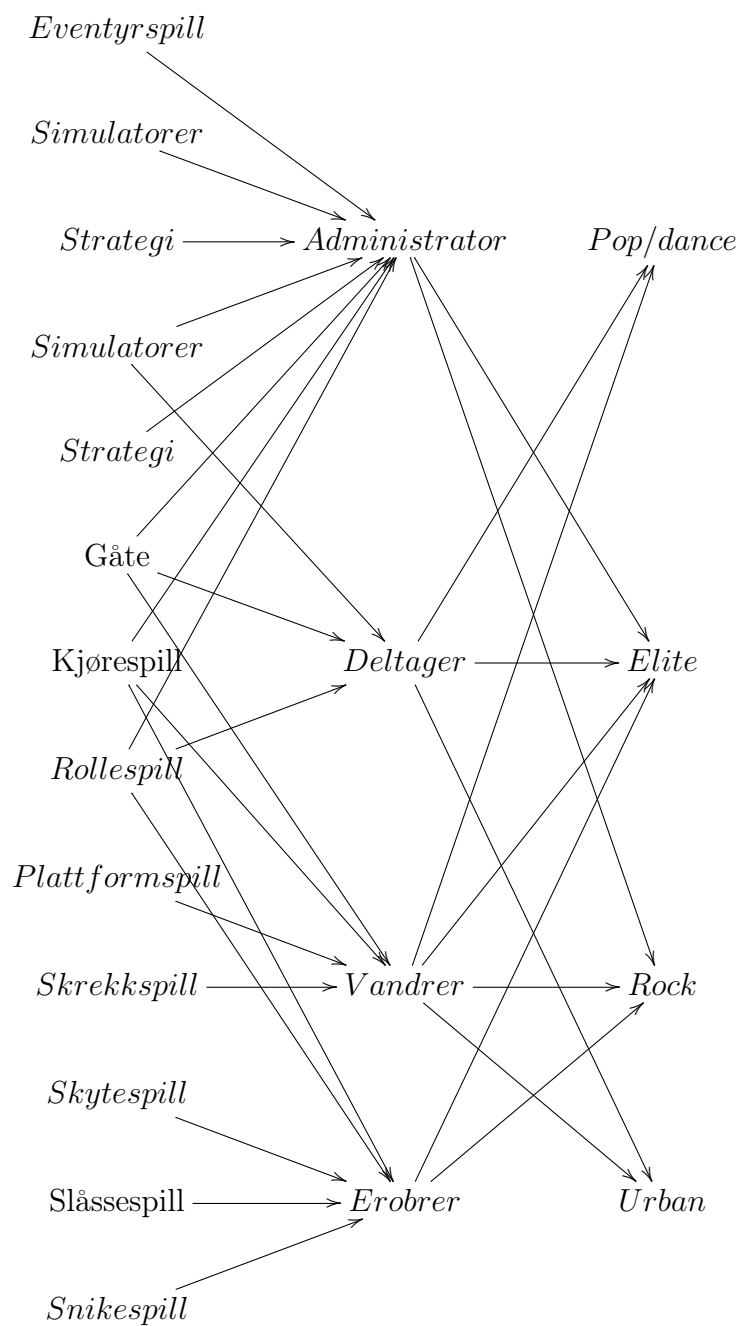
Snikespill

Snikespill er spill hvor målet for spilleren er å bevege seg igjennom baner, utføre oppdrag og ta ut fiender uten å bli oppdaget. Snikespill har likheter med skrekkspill i det at de ofte har en mørk stemning, og spilleren må unngå fiender og feller. I motsetning til skrekkspill har snikespill sjeldent overnaturlige eller skumle fiender, og spilleren styrer ofte en karakter som er en spesialist i sniking og slåssing, som f.eks en spion, tyv, eller leiemorder. Snikespill tiltrekker i likhet med slåssespill hovedsakelig dedikerte *erobrere*, og i en mindre grad avslappede *erobrere*. Det er dermed ikke overraskende at de vanligste musikktypene i snikespill er *elite* og *rock*. Ofte brukes musikk av typen *elite* mens spilleren er uoppdaget, og bytter raskt over til *rock* idet spilleren blir oppdaget og må rømme eller slåss.

2.5 Musikk i spill

Vi har tidligere sett på hvordan musikken i spill henger sammen med spillsjanger og spillertype. Her skal vi se videre på hvordan dette materialet kan informere arbeidet med utforming av musikk til dataspill. Dette med tanke på å lage musikk som i størst mulig grad treffer de ulike målgruppene for den typen spill man produserer musikk for, og som best mulig passer inn i spillets univers.

Forskjellige musikksjangre kommer i likhet med spillsjangre også med ulike konvensjoner, og er ofte knyttet til en bestemt instrumentering, tempo, rytme, og toneart. Musikk i dataspill kommer også med en ekstra utfordring for komposisjon av musikk med tanke på dynamikk. Dataspill har ofte ingen bestemt rekkefølge eller lengde på dets hendelser, noen som også burde tas høyde for i musikken.



Figur 2.7: Relasjoner mellom spillsjangre, spillertyper, og musikktyper

2.5.1 Musikkens funksjon

Noe av det man bør tenke over når man skal lage musikk til et annet medium er: hvorfor skal det være musikk? Hva er musikkens funksjon? Hvordan bidrar musikken til helheten?

Miljø/stemming

En av de grunnleggende funksjonene til musikk i dataspill er å dra spilleren inn i den virtuelle verdenen med en større tilstedeværelse ved å skape en troverdig stemming og miljø. Valget av musikk vil underbevisst fortelle spilleren mye om spillets miljø, og kan enkelt brukes til å sette en bestemt tidsperiode eller sted ved å spille på spillerens kjennskap til vår egen musikalske historie og tradisjoner. Selv om spillet kan foregå i en fiktiv verden, vil musikk fra et spesielt sted eller tidsperiode automatisk trekke assosiasjoner til dette stedet eller perioden. For at musikken skal ha denne effekten er det nødvendig at spilleren har en viss kjennskap til disse musikalske referansene. Det er dermed nødvendig å være bevisst spillernes referansenivå basert på hvilken kultur eller aldersgruppe målgruppen er i.

Flyt

En annen funksjon musikken i dataspill kan ha er å sette spilleren i en bestemt sinnstilstand. Målet er ofte å oppnå det som i psykologien kalles flyt, eller å være 'i sonen'. Mihaly Csikszentmihalyi, som først definerte begrepet *flow*²³, definerer det som en komplett fokusert motivasjon, med en ensrettet tilstedeværelse. I denne tilstanden er man komplett fokusert på øyeblikket, mister selvbevissthet, og har en endret tidssans. Selv om spiller ikke nødvendigvis oppnår en tilstand av flyt, vil musikken fortsatt hjelpe å holde spilleren fokusert på spillet, og ikke bli distraheret i like stor grad av andre lyder rundt seg.[4]

Signal

Musikken kan også brukes for å gi spilleren signaler om miljøet eller spilleren. Det er vanlig å bruke musikk for å signalisere spilleren om at han har blitt oppdaget av en fiende, har blitt skadet, eller når spilleren har oppnådd et

²³Flow [http://en.wikipedia.org/wiki/Flow_\(psychology\)](http://en.wikipedia.org/wiki/Flow_(psychology)) sist besøkt 08.05.15

bestemt mål o.l. Denne signaliseringen kan være i form av korte musikalske stikk, bytte av musikalsk tema, eller endringer i musikalsk intensitet.

I noen spill brukes fraværet av musikk som et signal til spilleren om at dette området av spillet er ferdig, og at det er på tide å bevege seg videre i handlingen. Denne funksjonen kaller Karen Collins i sin bok *Game Sound* en 'boredom switch'[3, p. 131]. Den motsatte effekten er når musikken brukes på tvers av scenene i spillet for å knytte handlingen sammen.

Temposetting

I mange spill brukes musikken aktivt for å sette tempoet og intensiteten i handlingen. I de fleste spenningsspill vil det være viktig å holde spilleren engasjert ved å holde tempoet oppe ved hjelp av musikken, selv om det tidvis er svingninger i spillintensiteten. Samsvaret mellom spillets handling og musikkens tempo og intensitet er også viktig for å skape en troverdig opplevelse for spilleren.

Spill generert basert på musikk

En del spill baserer spillmekanikken sin rundt musikken. Dette er ofte rytmespill hvor spilleren må spille på spesiallagede kontrollere i form av instrumenter eller dansematter i takt med musikken for å få poeng. Kjørespill hvor banen blir generert basert på musikkens rytmikk og intensitet har også blitt vanligere. Et eksempel på spill som bruker musikken for å generere innholdet i spillet er *Audiosurf* (Invisible Handlebar, 2008), som er et kjørespill hvor banen genereres basert på musikk valgt av spilleren.

2.5.2 Komposisjonsstrategier

Musikken i dataspill kan grovt deles inn basert på noen kriterier for hvordan denne musikken er komponert og spilles av. Det første kriteriet er om musikken er diegetisk eller ikke-diegetisk, det vil si om musikken høres innad i spillverdenen av karakterene, eller om den er lagt til 'over' handlingen i spillet. Diegetisk musikk i dataspill er som regel i form av tradisjonell lineær komposisjon, siden målet som regel er å emulere et annet medium innad i spillverdenen, som en radio, tv, eller lignende. Ikke-diegetisk musikk kan derimot også komponeres med en rekke andre strategier.



Figur 2.8: *Audiosurf*

Tradisjonell lineær komposisjon

Et annet kriterium er om musikken er lineær, dvs at variasjonen i musikkens er fastsatt på forhånd, eller om musikken er dynamisk, dvs om musikken varierer basert på endringer innad i spillet. Disse musikalske variasjonene kan være i struktur, instrumentering, intensitet, hastighet, tonalitet/pitch, effekter, eller hvilke som helst andre musikalske parametere, avhengig av kapasitetene til spillet og komponistens planlegging og kreativitet.

Dynamisk musikk

Jeg har valgt å skille mellom dynamisk musikk som er interaktiv, dvs musikk hvor variasjonene kan direkte påvirkes av spillerens handlinger, og dynamisk musikk som er adaptiv, og påvirkes indirekte av spillerens handlinger. Jeg vil for enkelthets skyld bruke begrepet dynamisk som et samlebegrep for både interaktiv og adaptiv musikk. Man kan f.eks se for seg en grense mellom

to områder i en spillverden som er knyttet til hvert sitt musikalske tema, og som spilleren fritt kan bevege seg mellom. Ved å bevege seg frem og tilbake mellom disse områdene kan spilleren direkte velge hvilke tema som blir spilt av. Dette er et eksempel på interaktiv musikk. Man kan også tenke seg et eksempel hvor musikken forandrer seg basert på spillerens helse. Om spillerens helse kommer under en viss grense kan man f.eks legge til ekstra instrumenter for å øke intensiteten og signalisere spilleren om fare. Dette er et eksempel på adaptiv musikk. Ofte brukes en kombinasjon av disse strategiene med blandinger av lineær, adaptiv, og interaktiv musikk i det samme spillet.

Valg av strategi og begrensninger

I deler av spillet hvor handlingen er av en bestemt lengde, som i sekvenser med planlagte hendelser, eller hvor spilleren beveger seg igjennom handlingen med en fast hastighet, kan det være naturlig å bruke en lineær komposisjonsstrategi, slik som er vanlig i film og andre lineære medier. Man kan også bruke lineær musikk som loops som bakgrunnsmusikk i menyer o.l. hvor det ikke er noen spesiell handling.

En adaptiv musikkstrategi kan være naturlig å bruke i deler av spill hvor musikken brukes for å signalisere spilleren om handling eller egenskaper ved spillverdenen ved å variere musikken i samsvar med ulike tilstander i spillet. Denne signaliseringen trenger ikke å være bevisst for spilleren, men kan fungere underbevisst ved å trekke oppmerksomhet mot viktige momenter, som f.eks når spilleren blir angrepet av en fiende, eller er nært ved å dø.

En musikkstrategi med interaktivitet kan brukes når spilleren skal få tilbakemelding på noe han velger å gjøre i spillverdenen. Ofte brukes det for å signalisere at spilleren har beveget seg inn i et nytt område av spillet, men brukes av og til også som en del av spillmekanikken ved at spilleren kan kontrollere musikken innad i spillverdenen, enten ved å skru av eller på en lydkilde med musikk, eller for å spille musikk ved å gjennomføre en rekke handlinger i riktig rekkefølge.

På grunn av dataspillmediets interaktivitet og dynamiske varighet må man komponere musikken med tanke på dette. Den enkleste strategien for å løse denne utfordringen er å spille av et lineært segment musikk i en loop inntil et visst kriterium oppfylles eller spillet avsluttes. Siden størrelsen på musikkfilen er begrenset av maskinvarens minne, er lengden og kvaliteten på filen nødvendigvis også begrenset. En økning i varighet eller oppløsning for lydfilen vil begge føre til en økt filstørrelse. En kort repeterende lydfile

vil ha et mindre potensiale for variasjon, og vil dermed raskere oppleves som kjedelig eller irriterende av lytteren. En lydfil med lav kvalitet vil også kunne oppleves som irriterende av lytteren over lengre perioder. Grunnet begrenset minne vil dermed økt lyd kvalitet eller lengde gå på bekostning av hverandre, noe som betyr av man må finne et kompromiss mellom disse. Selv om moderne spillmaskiner har store mengder minne, skal ofte dette deles mellom et stort antall elementer. Det er også begrensninger på hvor stort spillet kan være totalt, og ikke bare hvor mye minne det kan bruke om gangen. Det er derfor vanlig å prøve å begrense filstørrelsen på elementene så mye som mulig uten at det går for mye utover kvaliteten. Disse begrensningene gjelder spesielt på mobile og håndholdte spillmaskiner, men er også relevant for andre plattformer.

Vertikal og horisontal struktur

For å unngå problemet med lyttetretthet på grunn av statisk og ensformig musikk, men fortsatt bruke begrenset maskinminne, har man et par tilgjengelige strategier. En av disse strategiene er å bruke en rekke korte lineære klipp, som kan kombineres på ulike måter for å skape variasjon. Fordelen med dette er at antallet kombinasjoner øker eksponentielt med antallet lydklipp, men mengden plass bare øker lineært. Ulempen er at det krever mer arbeid for komponisten å lage klipp som passer sammen med hverandre, og jo flere klipp som skal kombineres, jo vanskeligere er det å få alle til å passe sammen. Denne teknikken med å kombinere korte lydklipp kan brukes på tre forskjellige måter, vertikalt, horisontalt, eller som en kombinasjon av vertikal og horisontal. Dette ligner på mange måter på 'Musikalisches Würfelspiel', som er en klassisk algoritmisk komposisjonsteknikk basert på forhåndskomponerte blokker av lik lengde.²⁴ [9, p. 185]

Horisontal kombinasjon har en rekke fordeler. Ved å kombinere lydklippene horisontalt etter hverandre på ulike måter kan man skape variasjon og i større grad unngå problemet med lyttetretthet grunnet repetisjon. Ved å gjøre dette begrenser man mengden lagringsplass brukt ved at den samme lydfila brukes hver gang den musikalske delen repeteres, istedenfor å lagre alle repetisjonene slik som i lineære lydklipp. Horisontal kombinasjon åpner også for at den musikalske strukturen kan påvirkes av spillet dynamisk.

Å kombinere lydklippene vertikalt vil si å synkronisert spille av flere lydklipp

²⁴Musikalisches Würfelspiel http://en.wikipedia.org/wiki/Musikalisches_Wurfelspiel sist besøkt 08.05.15

parallelt. Ved å legge til eller trekke ifra ulike lydklipp kan man skape et stort antall variasjoner av den samme musikken ved hjelp av relativt få lydklipp. Ved å gjøre dette begrenser man hvor mye plass man trenger for lagring ved å lagre alle elementene hver for seg, istedenfor å lagre alle mulige kombinasjoner av disse. Den vertikale kombinasjonen kan i likhet med horisontal kombinasjon også varieres dynamisk for å skape variasjon.

Både horisontal og vertikal kombinasjon av lydklipp krever at man kan skape overganger mellom, og loope filene sømløst. Vertikal kombinasjon bruker også flere lydfiler samtidig som må holdes synkroniserte, noe som kan være krevende på noen systemer med et stort antall filer, etterhvert som antallet filer blir så stort at systemet ikke rekker å gå igjennom alle filene før neste sample skal spilles av. I dette tilfelle må disse samplene enten hoppes over, eller utsettes. En utsettelse vil gjøre avspillingen av lydfilene asynkron, mens å hoppe over samples kan føre til hørbare artefakt i lyden. Det er også mulig å kombinere lydklipp både horisontalt og vertikalt samtidig, og disse teknikkene har dermed potensialet for stor variasjon med få filer. [9, p. 187]

Algoritmisk komposisjon

En annen strategi for å begrense bruken av lagringsplass, men uten å miste variasjon er å syntetisere musikken i sanntid basert på musikkdata komponert på forhånd, eller generert ved hjelp av algoritmisk komposisjon. Å syntetisere musikken basert på programmering var den eneste muligheten for å skape musikk tidlig i spillmediets historie, og har på mange måter preget tradisjonene og normene knyttet til å lage musikk for dataspill. Før bruken av samlede lydfiler var vanlig, ble musikken laget ved å styre tone- og støygeneratorer ved hjelp av koding. Ofte begrenset til noen få generatorer, bølgeformer, og tungvinn implementering var denne formen for musikk svært enkel, repetitiv, og med fokus på melodi. Etterhvert som teknologien utviklet seg dukket det opp standarder som MIDI og MOD, og det ble enklere å lage musikk på denne måten. Dataspillmusikk fra denne perioden har også dannet grunnlaget for sjangeren Chiptune²⁵, hvor man bruker denne formen for teknologi og estetikk til å lage musikk. Denne strategien for musikalsk komposisjon er fortsatt relevant for moderne dataspill, og spesielt på plattformer med begrenset maskinvare, eller i spill som har en 'retro' estetikk. Det er også mulig å bruke virtuelle instrumenter med samples av høy kvalitet fra virkelige instrumenter for å skape mer realistisk musikk styrt av data. Disse virtuelle instrumentene tar derimot ofte svært stor plass, og for

²⁵Chiptune <http://en.wikipedia.org/wiki/Chiptune> sist besøkt 08.05.15

musikk komponert på forhånd er det dermed mer økonomisk med tanke på lagringsplass å heller lagre musikken som lydfiler som spilles av istedenfor å inkludere instrumentene i spillet. For musikk komponert algoritmisk i sanntid kan det derimot være en relevant mulighet å bruke virtuelle instrumenter implementert i spillmotoren. [9, p. 209]

Tema

I musikkteorien brukes ofte begrepet tema om en gjenkjennelig melodi som en komposisjon bygger på. I film og dataspill brukes ofte en teknikk kalt ledemotiv for å danne ulike temaer. Ledemotiv er en teknikk som først ble tatt i bruk i klassisk musikk i romantikken, for eksempel i Richard Wagners operaer. Ledemotiv fungerer ved å gjentatt knytte en gjenkjennelig melodi eller harmoni til en stemning, karakter, hendelse, område, eller lignende. Fra moderne film og dataspill er kanskje de mest kjente eksemplene John Williams musikk til *Star Wars*.

Bruken av ledemotiv kan være et svært effektivt virkemiddel for å gi lytteren bestemte assosiasjoner. Ved å først definere et ledemotiv ved å gjenta dette i en spesiell kontekst, for deretter å benytte det i en annen kontekst, kan man knytte sammen disse elementene for å gi lytteren ønskede assosiasjoner.

I dataspill er det som en konsekvens av mediets dynamiske struktur naturlig å bruke musikalske ledemotiver eller tema knyttet til ulike steder i spillverdenen, karakterer som spilleren møter, eller hendelser som inntreffer ofte.

Når man skal komponere temaer til dataspill er det viktig å tenke igjennom graden av repetisjon og gjenkjennelighet. Om et tema er veldig gjenkjennelig, og repeteres for ofte, eller for lenge av gangen har dette potensialet til å irritere spilleren og skape lyttetretthet. For å unngå dette problemet kan man tenke seg til noen ulike løsninger.

Den enkleste løsningen på problemet med musikalsk lyttetretthet er å begrense bruken av musikk. Ved å kun bruke musikk når den har en prominent funksjon i spillets handling, kan man unngå problemet med lyttetretthet. På den andre siden kan fraværet av musikk gjøre resten av spillet mindre engasjerende, og plutselige innslag av musikk etter et langt fravær kan oppleves som kunstig, og bryte spillerens innlevelse i spillet. Man kan også løse problemet ved å variere temaenes melodier, instrumentering o.l.

En annen strategi er å bruke en mer ambient musikk som ikke bygger på lett gjenkjennelige melodier og harmonier, men som heller baserer seg på langsom utvikling av klangfarger. Denne formen for musikk tiltrekker mindre oppmerksomhet enn tematisk musikk, og kan dermed repeteres oftere, men er også mindre engasjerende.

Ved å bruke en algoritmisk komposisjonsstrategi kan man i stor grad unngå repetisjoner i musikken. På den andre siden gjør denne strategien det vanskeligere å definere faste tema, og resultatet blir ofte ikke spesielt musikalsk engasjerende.

Den beste løsningen er muligens å bruke en kombinasjon av alle disse tematiske strategiene. Ved å bruke temaer som har en klart gjenkjennelig melodi komponert på forhånd i segmenter av spillet hvor musikken har en fremtredende rolle kan man ha engasjerende gjenkjennelig musikk der hvor det trengs. Dette kan kombineres med algoritmisk komponerte eller ambiente temaer i segmenter av spillet hvor musikken har en mindre fremtredende rolle, og man vil dermed også kunne unngå problemet med lyttetretthet. Selv om denne kombinasjonsløsningen kanskje er den som vil fungere best fra spillerens perspektiv, vil det også være den løsningen som krever mest arbeid å gjennomføre.

Instrumentering

Som vi har sett tidligere er det ofte en tett kobling mellom spillsjanger og musikkjanger. Et av de viktigste elementene for å definere en musikkjanger er valget av instrumentering, med andre ord hvilke instrumenter som brukes, og hvordan. Instrumenteringen som velges vil også være viktig for å bidra til spillverdenens miljø og stemning. Man kan argumentere for at instrumenter med et harmonisk overtonespekter ofte vil oppleves som mildere enn instrumenter med et mer disharmonisk overtonespekter. Graden av harmoni, både i instrumentets overtonespekter, akkorder, og mellom instrumenter kan dermed brukes aktivt for å definere graden av aggressivitet i musikken.

I tillegg til tradisjonelle inndelinger basert på sjanger, material, eller eksitasjonsmetode kan man også grovt dele inn instrumentene i akustiske og elektriske. Mange instrumenter vil falle i en av kategoriene, men en del instrumenter vil oppleves som et sted imellom. Et piano vil f.eks oppleves, som et akustisk instrument, mens en synthesizer vil kunne oppleves som elektrisk ved bruk av 'kunstige' lyder, men akustisk ved bruk av troverdige

instrument-samples. En del musikksjangre er ofte kun akustisk eller elektrisk, men blandinger er også vanlige. Graden av prosessering og effekter på instrumenter vil også forandre instrumentets overtonespekter, og om det oppleves som elektrisk eller akustisk.

Valget av instrumentering vil på dette viset være en viktig del av konstruksjonen av spillverdener ved å bruke en troverdig instrumentering og dermed skape et troverdig miljø. Det vil f.eks oppleves som naturlig å bruke kun periodekorrekte akustiske instrumenter i et historisk spill satt til middelalderen, eller å bruke kun elektroniske instrumenter i et spill med cyberpunk-stil med handling satt i fremtiden.

2.5.3 Bruk av musikk i dataspill

I tillegg til musikkens ulike funksjoner som beskrevet tidligere, skriver Collins og Phillips[9, p. 145] også om en del forskjellige roller som musikk kan ha knyttet til spill.

Trailer

For å skaffe finansiering fra investorer og utgivere, eller reklamere for spillet til forbrukere skapes det ofte en trailer for spillet i form av en kort film. Siden denne traileren er lineær kan man komponere musikk til denne på samme måte som til tradisjonell film. I motsetning til spillefilm, hvor musikken ofte tilpasses til filmen, blir trailere ofte klippet for å matche musikken. I trailere brukes av og til også lisensiert musikk. Om man komponerer egen musikk for traileren, eller bruker musikk fra spillet, så er som regel hovedfunksjonen til musikken å informere om spillverdenens miljø og stemning, og gjøre potensielle spillere interessert i spillet

Globale spor

Globale spor er musikk som kan brukes hvor som helst i spillet. Disse sporene er ikke knyttet til et spesielt sted i spillverdenen, men er viktige for å sette den totale stemningen til spillet. Globale spor brukes ofte i tre forskjellige sammenhenger, menyer, stingers, og minispill.

Menyspor er musikk som spilles hver gang spilleren aktiverer en meny, og spillet pauses. Menymusikken er ofte rolig, og i form av et lineært segment

som loopes så lenge menyen er aktiv.

Stingers er korte musikalske tema som trigges ved spesielle hendelser i spillet. Vanlige eksempler er når en spiller oppnår et mål eller vinner en kamp, blir beseiret eller dør, eller når spilleren oppdager et nytt område eller samlegjenstand. Stingers kan også brukes ved overganger mellom ulike scener o.l. for å knytte disse sammen[3, p. 146], eller for å bryte opp og skape fornyet interesse i spillet. Hovedfunksjonen til stingers er å informere spilleren om en forandring i spillets tilstand. Navnet 'sting' eller 'stab' kommer trolig av at de ofte er korte og skarpe 'stikk' i handlingen.

Minispill er små spill som er integrert i handlingen til det større spillet. Ofte brukes slike minispill for å representere en handling i spillet, som f.eks ved å bruke et gåtespill for å hacke en datamaskin, eller dirke en lås. Disse minispillene har ofte et eget musikalsk tema som spilles hver gang minispillet er aktivt.

Hovedtema

Hovedtemaet er et av de viktigste temaene, og fungerer som spillets, eller til og med spillseriens musikalske signatur. Hovedtemaet er ikke knyttet til noe spesielt sted i spillverdenen, men brukes ofte under åpningssekvensen og eventuelt rulleteksten til spillet. Det er også vanlig å bruke hovedtemaet i trailere til spillet. Hovedtemaer har ofte en minneverdig melodi som er lett å kjenne igjen, slik at spillere blir raskt dratt inn i spillverdenen hver gang de hører temaet. Hovedtemaet fungerer som et slags ledemotiv for hele spillet. De viktigste funksjonene til hovedtemaet er å sette den generelle stemningen og tempoet for spillet.

Opplæringssekvens

Av og til trengs det egen musikk for spillets opplæringssekvens. En opplæringssekvens er en del av spillet hvor spilleren lærer seg hvordan spillets mekanikk og kontroller fungerer. Opplæringssekvensen er ofte en egen seksjon av spillet hvor spillet blir avbrutt av instruksjoner, og spilleren kan teste mekanikk og kontroll i trygge omgivelser. I dette tilfellet burde musikken ta høyde for avbrytelser i form av instruksjoner, og er ofte nærmere menymusikken i stemning og tempo. I andre spill er opplæringssekvensen en integrert del av spillets handling, og ofte fordelt utover spillets handling

etterhvert som ny spillmekanikk blir tilgjengelig for spilleren. I dette tilfelle er det kanskje ikke behov for en egen musikk for opplæringssekvensen.

Bakgrunns- og spenningsspor

Bakgrunns- og spenningsspor omfatter som regel store deler av musikken i et spill. Ikke alle spill har både bakgrunns- og spenningsspor. 'Avslappede' spill har ofte bakgrunnsmusikk, men ingen spenningsmusikk, mens en del spenningsspill har spenningsmusikk men ingen bakgrunnsmusikk.

Bakgrunnsmusikk er rolig musikk som brukes i seksjoner av spillet med lav intensitet. Hovedfunksjonen til bakgrunnsmusikken er å sette stemning/miljø, tempo/intensitet, samt å holde spilleren engasjert. Bakgrunnsmusikken er ofte knyttet til et spesielt sted i spillverdenen.

Spenningsmusikk brukes i seksjoner av spillet med høy intensitet. Spenningsmusikk brukes for å varsle spilleren om fare og skape spenning. Ofte er det egen spenningsmusikk knyttet til ulike typer fiender eller farer. Dataspill har også lang tradisjon med ledemotiv knyttet til sjefsfiender som er ekstra vanskelige å beseire.

Gåtespor

Gåtespor er musikk som brukes når spilleren skal løse en slags gåte for å komme seg videre i spillet. Gåtemusikken er ikke knyttet direkte til gåtespillsjangeren, men er knyttet til spillmekanikken gåteløsning i alle spillsjangre. Gåtemusikkens intensitet er knyttet til typen gåte som skal løses, og til konsekvensene om man ikke greier å løse gåten, men gåtemusikken blir sjeldent like intens som spenningsmusikken. Hovedfunksjonene til gåtemusikken er å skape spenning og flyt.

Planlagte hendelser

I spill blir ofte viktige deler av historien fortalt via forhåndsplanlagte hendelser som blir trigget når spilleren når et visst sted i spillets handling. Disse kan være i form av hendelser hvor spilleren ikke lenger kan interagere med spillet, og bare kan observere handlingen, eller via en serie korte sekvenser hvor spilleren kan interagere med handlingen kun ved visse tidspunkt. I tidlige dataspill besto disse planlagte hendelsene av bilder og tekst, og gikk senere over til korte videoklipp. Etterhvert har det blitt

vanligere å bruke spillmotoren til å lage disse hendelsene i sanntid, slik at den visuelle stilen ikke forandrer seg fra vanlig spilling, og gir muligheten for interaksjon. Siden disse hendelsene er planlagt på forhånd med fast lengde og rekkefølge kan man komponere musikk på lignende vis som tradisjonell filmkomposisjon, med andre ord, som en eller flere lineære sekvenser med fast lengde.

Tiltrekningsmodus

I den tiden hvor arkadespillene sto samlet i store spillehaller og konkurrerte om spillernes oppmerksomhet var det vanlig at spill som ikke var i bruk sto i tiltrekningsmodus for å trekke til seg potensielle spillere. Denne tiltrekningsmodusen finnes også i noen moderne spill, og kanskje spesielt i spillsjangre som har en historie som arkadespill. I disse moderne spillene fungerer tiltrekningsmoduset som en trailer med høydepunkter fra spillet, eller som underholdning når spillet er pauset. Musikken i denne modusen er som regel hovedtemaet fra spillet, eller en medley av annen musikk fra spillet.

2.6 Lyddesign

Når man lager lyddesign til film tar man ofte opp lyd på settet, og bruker dette som utgangspunkt for filmens lydspor og legger til foley, lydeffekter og stemmeskuespill. Når man skal lage lyddesign til animasjon eller dataspill har man ingen virkelige hendelser å ta utgangspunkt i, og hele lydsporet må dermed konstrueres fra grunnen av. I dataspill er det heller ingen fast rekkefølge på hendelser, noe som innebærer at lydsporet må kunne spilles av dynamisk i sanntid basert på spillerens handlinger. Spilleren har naturlig nok en begrenset mengde handlinger tilgjengelig i spillverdenen, som man kan planlegge og lage lyder for på forhånd. Disse lydeffektene kan dermed integreres med spillet slik at de blir spilt av til rett tid, og med rett prosessering basert på handlingen i spillet slik at de skaper et helhetlig og troverdig lyddesign.

2.6.1 Klassifisering av lyder

I likhet med musikk i dataspill, kan lydeffekter deles inn i diegetiske og ikke-diegetiske. De ulike formene for lyddesign kan i tillegg til diegetisk og ikke-diegetisk også grovt inndeles basert på deres funksjon i spillet. Bruken av

lyder kan grovt deles inn i brukergrensesnitt, miljølyder, karakterlyder/foley, og stemmeskuespill. I tillegg bruker jeg 'effektlyder' om mer generelle lyder som ikke passer inn i noen av disse kategoriene. Disse inndelingene er på ingen måte en fasit, eller gjensidig ekskluderende, men kan antyde noe om hvordan lydeffektene brukes i dataspillet.

Brukergrensesnitt

En stor andel av de ikke-diegetiske lydeffektene brukt i et dataspill er knyttet til brukergrensesnittet. Disse lydene er dermed vanlig å finne i menyer o.l. I en del dataspill vil det også finnes objekter i spillverdenen som har sitt eget brukergrensesnitt med lyder knyttet til seg. I disse tilfelle vil lydene knyttet til dette brukergrensesnittet kunne være diegetiske.

Lydeffekter knyttet til brukergrensesnitt brukes som en auditiv tilbakemelding til brukeren om at interaksjonen med spillet er registrert, eller for å informere om dataspillets tilstand. Disse lydene er ofte enkle korte signallyder. Når slike signallyder mangler, eller blir brukt inkonsekvent, kan brukeren bli usikker på om interaksjonen er registrert, noe som kan føre til irritasjon eller usikkerhet.

Miljølyder

Begrepet miljølyder bruker jeg om hovedsakelig statiske diegetiske lydeffekter knyttet til et område, som brukes for å informere om omgivelsene og sette stemning. Det er i tillegg til dette hovedsakelig to grunner til å bruke et kontinuerlig bakteppe av miljølyder i en spillscene. Det som kanskje er den viktigste grunnen er å skape en troverdig spillverden som gir en høy grad av innlevelse. Et unaturlig fravær av lyd, eller brå overganger mellom lyd og stillhet vil kunne bryte spillerens innlevelse i spillverdenen siden man i virkeligheten sjeldent opplever total stillhet. Den andre grunnen er for å jevne ut overgangene til og fra andre lydeffekter, slik at det totale lydbildet oppleves som et sammenhengende lydlandskap istedenfor som en serie oppbrutte lyder. Miljølyder kan være i form av enkel udefinert bakgrunnsstøy, eller kan være i form av atmosfæriske elementer som vind, bølger, dyr, eller andre naturlyder. Slike miljølyder kan brukes svært effektivt for å definere en setting, selv om ingen av elementene som høres faktisk er synlige i scenen. Dette krever naturlig nok at det er et visst samsvar mellom det visuelle og auditive. Jo høyere grad av dissonans det er mellom det visuelle og auditive, jo høyere grad av *suspension of disbelief* vil det kreves av spilleren for å oppleves som

troverdig. Det er derfor viktig å være bevisst dette når man skal lage lyd til en scene.

Karakterlyder

Karakterlyd kan brukes om diegetiske lydeffekter knyttet til en karakter, som tradisjonelt lages som foley²⁶. Foley er et begrep oppkalt etter Jack Donovan Foley, som generelt blir regnet som den første til å legge lydeffekter til film. Begrepet brukes vanligvis om lydeffekter som lages av en person i sanntid, synkronisert til et medium. Dette er ofte lyden av skritt, klær, våpen, korte stemmelyder o.l. Dette er ofte korte lyder som trigges basert på spillerens interaksjon med spillet. Karakterlyder trenger ikke nødvendigvis å være knyttet til spillerens karakter, og karakterene trenger heller ikke å være menneskelignende, men kan f.eks i et kjørespill være en bil.

Om karakterlydene skal klassifiseres som interaktive eller adaptive kan ofte være litt uklart. Når en spiller interagerer med spillet for å avfyre et våpen vil man som en direkte konsekvens av dette høre f.eks en skytelyd. På bakgrunn av dette vil man kunne argumentere for at dette er en interaktiv lyd. Om derimot en karakter i spillet som er styrt av datamaskinen via en kunstig intelligens gjør akkurat det samme, vil denne lyden være interaktiv fra perspektivet til den kunstige intelligensen, men ikke for spilleren. Vi kan dermed argumentere for at det er konteksten lyden trigges og oppleves i som avgjør om lyden er interaktiv, og ikke lyden i seg selv.

Stemme

Stemmeskuespill bruker jeg om alle lydeffekter som bruker stemme. Dette er hovedsakelig lyder med semantisk innhold, med andre ord språk.[2, p. 28] Stemmeskuespill innebærer innspilling av en karakters stemme for bruk i medier. Stemmeskuespill har tradisjoner fra hørespill sendt på radio, hvor skuespillere fremfører replikker. I film er det vanlig å spille inn replikker på nytt i postproduksjon for å ha større kontroll, for å få bedre lyd kvalitet, eller å legge til et alternativt språk, også kalt dubbing. Når man spiller inn replikker til film og karakteren er synlig(on screen), kreves det at replikkene tilpasses til filmen, men i animasjonsfilm eller dataspill kan animasjonene tilpasses replikkene. Grunnet den dynamiske naturen til dataspill er det som regel

²⁶Foley [http://en.wikipedia.org/wiki/Foley_\(filmmaking\)](http://en.wikipedia.org/wiki/Foley_(filmmaking)) sist besøkt 08.05.15

nødvendig å dele opp replikkene i korte segmenter som kan trigges sammen med en tilhørende animasjon.

Stemmeskuespill kan brukes både diegetisk og ikke-diegetisk. Ikke-diegetisk stemmeskuespill kan f.eks være i form av en fortellerstemme som kommenterer spillets handling, eller som formidler informasjon i hjelpetekst og i opplæringssekvensen av spillet. Diegetisk bruk av stemme vil hovedsakelig være spillkarakterenes replikker. Å bruke stemme i spill er en veldig effektiv måte å formidle informasjon på, som spilleren er vant med fra det virkelige liv, og vil om det gjennomføres rett føre til en økt grad av innlevelse, og gi spillverdenen en større troverdighet. Valget av dialekt eller sosiolekt²⁷ på stemmeskuespillet vil være viktig for å definere de ulike karakterene og miljø de kommer fra.

En ulempe med å bruke stemme i spill, og språk generelt, er at det krever at spilleren har kjennskap til språket. Om målgruppen for spillet skal være internasjonal, eller har redusert språkforståelse, som f.eks små barn, kan det være lurt å finne enn annen måte å kommunisere med spilleren på. Spill har ofte svært store mengder med replikker og ulike karakterer, noe som kan gjøre innspilling av stemmer til en lang og dyr prosess, spesielt om spillet skal være tilgjengelig på flere ulike språk. Denne store mengden replikker vil også bruke svært mye lagringsplass i forhold til tekst. En mulighet for å slippe unna denne lange og dyre innspillingsprosessen er å bruke talesyntese fra tekst til lyd. Dagens talesynteseteknologi er fortsatt litt begrenset med tanke på troverdighet og skuespill, og er derfor ikke så veldig mye brukt i dataspill i dag. Det er derimot trolig at dette vil bli mer utbredt etterhvert som talesynteseteknologien utvikler seg.

2.6.2 Variasjon

Når man skal skape et troverdig lyddesign er det viktig at de lydene man hører ikke oppleves som kunstige. En av årsakene til at lyder i dataspill ofte kan oppleves som kunstige er at man gjenkjenner repeterende lyder som den samme kildelyden. Når lyder skapes i virkeligheten vil det ofte være en rekke tilfeldigheter som lager naturlige variasjoner i lyden. Dette problemet kan løses ved å ha en egen kildelyd for hver gang lyden høres. Dette fungerer fint i statiske medier som film, men i interaktive medier som dataspill kan mange av lydene trigges et uendelig antall ganger. Dette kan potensielt løses ved å generere lyder i sanntid basert på hendelser i spillet, og dermed emulere

²⁷Sosiolekt <http://en.wikipedia.org/wiki/Sociolect> sist besøkt 08.05.15

hvordan lyder dannes i virkeligheten, men en slik løsning for å realistisk emulere dannelse av lyder basert på fysisk modellering av virkeligheten i sanntid er ikke tilgjengelig med dagens teknologi.

Den menneskelige korttidshukommelsen kan ifølge artikkelen *The Magical Number Seven, Plus or Minus Two*[7] av George A. Miller huske mellom fem og ni elementer samtidig. Basert på dette kan man argumentere for at om man har mer enn ni variasjoner av en repeterende lyd, vil en lytter i praksis ikke kunne kjenne igjen repeterende enkeltvariasjoner av en lyd, så lenge det er minst 8 andre variasjoner mellom de to repetisjonene. Man slipper dermed å ha et uendelig antall variasjoner av en lyd for å gi illusjonen av at det er det. Om man skal ha ni versjoner av alle repeterende lyder i et dataspill kan det rask bli et enormt antall lyder, og man kan rask møte spillplattformens begrensninger med tanke på lagringsplass.

En ofte brukt teknikk for å øke antallet variasjoner av en lyd, men begrense antallet nødvendige kildelyder er å variere kildelydens pitch eller amplitude tilfeldig innenfor et forhåndsbestemt intervall. I eksperimentell psykologi brukes begrepet ‘Just-noticeable difference’(JND), også kalt ‘Weber’s law’,²⁸ om den minste variasjonen som skal til for å være merkbar minst halvparten av gangene. For lyd er JND forskjellig fra person til person, og som en funksjon av lydens frekvens, amplitude, og klangfarge. Normale voksne kan stabilt merke forskjeller i pitch på rundt 25 cent (ca 1.45 prosent). Dette er dermed rundt den minste variasjonen av pitch som det er noe poeng i å bruke for at lyttere stabilt skal oppleve det som som en variasjon av kildelyden i over halvparten av tilfellene. Med variasjoner i pitch mindre enn dette vil lydene kunne tolkes som den samme, og hensikten med variasjonen er borte. Om variasjonen i pitch blir for stor vil man til slutt oppleve lyden som en ny type lyd istedenfor som en variasjon av den samme kildelyden. Hvor stor variasjon av pitch man burde bruke på en repeterende lyd for å skape en troverdig variasjon som oppleves som naturlig vil i stor grad være avhenging av kildelyden, men vil sjeldent være mindre en 25 cent, eller større en oktav. I de fleste tilfeller vil en variasjon innen et intervall på noen få semitoner oppleves som troverdig.

For amplitude er JND for mennesker rundt 1dB. Dette er med andre ord omtrent den laveste variasjonen som vil oppleves av en lytter. En økning på 6dB vil derimot oppleves som dobbelt så høyt. Om variasjonen oppleves som mer enn en dobling kan dette i mange tilfeller oppleves som unaturlig. Hvor stor variasjon i amplitude man skal bruke for å gi illusjonen av en naturlig

²⁸JND http://en.wikipedia.org/wiki/Just-noticeable_difference siste besøkt 08.05.15

variasjon vil i likhet med variasjon i pitch være avhenging av kildelyden, men vil i de fleste tilfeller være opp imot 6dB med 1dB intervall. Ved å kombinere variasjon i både pitch og amplitude kan man dermed skape et stort antall variasjoner med få antall kildelyder. Dette er svært nyttig for lyder som gjentas ofte, og i store deler av spillet, som f.eks skritt, skytelyder o.l.

Kapittel 3

Praktiske prosjekter

Jeg har jobbet med å komponere musikk og lyddesign til to konkrete spillprosjekter for PC/mobile plattformer. Disse prosjektene er *Corp Wars* laget av Kybernesis¹, og *Shattered* laget av Gridmedia². Begge spillprosjektene har flere musikalske temaer samt lyddesign. Min rolle i prosjektene har vært å komponere musikk og lyddesign, og å bidra med å implementere dette i spillene.

3.1 *Karavan*

3.1.1 Om Spillet

Fra jeg begynte å jobbe sammen med Kybernesis og frem mot slutten av min masteroppgave har spillprosjektet gjennomgått en rekke endringer. Det originale spillprosjektet jeg ble med på var et spillkonsept kalt *The Siege* i Kybernesis' egne spillsetting *Corp Wars*. Dette spillet var ment som et lite prosjekt for selskapets praktikanter som var ment til å vare noen måneder, men som har blitt utvidet betraktelig. Konseptet for *The Siege* er et slags 'Castle defense' strategispill hvor spilleren skal forsvare en festning mot angripende fiender ved å kontrollere festningens forsvar. Dette spillet er også tenkt å være koblet sammen med et annet spill *Territories* hvor disse festningene kan designes. Disse to spillene skal også kobles sammen med et tredje spill *Karavan*, som er et handels- og utforskningsspill hvor spilleren

¹Kybernesis <http://kybernesis.com> sist besøkt 10.05.15

²Gridmedia <http://gridmedia.com> sist besøkt 10.05.15



Figur 3.1: Konseptskisse for *Karavan*

kontrollerer en handelskaravane som skal utforske områder for å oppdage nye byer å handle med.

Jeg begynte originalt å lage musikk til det første spillet *The Siege*, men Kybernesis har i ettertid valgt å utsette dette spillet til fordel for *Karavan*. Siden spillene foregår i samme spillverden er de musikalske kravene til *Karavan* og *The Siege* mer eller mindre identiske. Dette har gjort det relativt uproblematisk å bruke musikken jeg originalt laget til *The Siege* i *Karavan* istedenfor. Kravene til lyddesign i de ulike spillkonseptene er også ganske like, og mye av lyden vil bli gjenbrukt mellom de ulike spillene i *Corp Wars*-settingen.

Introduksjon til *Corp Wars*

The year is 2250, the world has barely survived, not just one, but several apocalyptic events. In the start of the 21st century, geomagnetic storms ravaged earth's magnetic fields. Western economy collapsed, resulting in the North American Wars. Global warming melts almost all the ice in the world, making sea levels rise with 80 meters, which again creates drown several cities and even whole countries and creates swampland other places. Floods and storms ravage the world mercilessly and new viruses and bacterias emerge, killing of large amounts of the world's

population. The melting ice thins out the salt-density in the North Atlantic sea and stops the Gulf Stream along with Yellowstone finally going off armageddon-style. A new ice age comes upon the world covering most of northern Europe, Sibir, Canada and Northern part of USA, people is driven from their homes and millions die as they start fighting over land and food, forming city-states called Enclaves around resource-rich areas. At the peak of all the wars between the Enclaves, where all known technology is used in abundance, the magnetic poles decide to shift. Leaving the world in yet another chaos where no air-based communication, flight or satellite contact is possible.

Enter humanities last hope, The Ark! An arcology built by the western coast of Northern Africa, one of the few larger communities left on earth and infinitely more advanced than any other communities. There's a catch though, not everyone gets to live there and have to go through a screening process in Kyburgh, the mining city created by the builders of The Ark. And what do you have to do to qualify you ask? No one knows! But it is said that the best chance is to prove that you can take care of yourself and contribute resources to The Ark by building up a corporation.

(Utdrag fra designdokumentet til Corp Wars)



Figur 3.2: Konseptskisse for *Corp Wars*

Litt om *Corp Wars*

Kybernesis har laget sin egen spillverden kalt *Corp Wars*, hvor de har planlagt en serie av sammenhengende spill som vil påvirke hverandre. Denne spillverdenen er en post-apokalyptisk versjon av vår verden, satt langt frem i tid, hvor sivilisasjonen har kollapset grunnet naturlige og menneskeskapte katastrofer. Befolkningen i denne verdenen er delt inn i fire hovedgrupperinger, 'Corporate', 'Enklave', 'Nommo', og 'Wastelanders'. Corporates er etterkommerne av vestlig og asiatisk kultur, med store selskaper og mye teknologi. Enklave er en samling av selvforsynte bystater. Nommo er en samling av de ulike stammene som har overlevd i Afrika og samlet seg til ett rike. Wastelanders er en blanding av folk som har forlatt de andre hovedgrupperingene og lever i ødemarka.

3.1.2 Musikk

Musikken jeg har laget til *Corp Wars* har vært basert på de ulike grupperingene i spillverdenen. Musikken er tenkt å dynamisk endres fra atmosfæremusikk til spenningsmusikk. Musikken er tenkt å ha en rekke funksjoner. En av funksjonene til musikken er å signalisere spilleren om hvilke fiender som angriper, og hvor stor fare man er i. Ved å ha spesielle temaer eller musikalsk stil som konsekvent brukes når fiender fra ulike grupperinger angriper, kan spilleren raskt lære hvilke gruppering som angriper, kun basert på musikken. Ved at musikkens intensitet dynamisk tilpasser seg variabler som helse, kan man signalisere til spilleren om tilstanden til kampen. Tanken bak musikken er også at de ulike temaene skal informere spilleren om de ulike grupperingenes kulturer. De ulike temaene er alle laget med tempo og tonearter som gjør musikalske overganger mellom dem så enkle som mulig.

Nommo-tema

Temaet til Nommo er basert på afrikansk musikk og instrumentering, blandet med teknologiske lyder, for å symbolisere at Nommo-kulturen bygger på tradisjonelle afrikanske kulturer, men har utviklet seg videre og har integrert teknologi. Nommo-kulturen baserer også mange av de musikalske instrumentene sine på resirkulering av gamle gjenstander som blykkbokser o.l.

For å skape et musikalsk tema som passer med Nommo-kulturens blanding

av afrikanske tradisjoner og moderne teknologi, valgte jeg å blande mer tradisjonelle akustiske afrikanske instrumenter og stemmer med moderne elektroniske elementer. Temaet er delt inn i tre forskjellige lag, som kan fades mellom for å forandre temaets intensitet. I det første eksempelet som er vedlagt presenteres de ulike intensitetene av temaet etter hverandre. Først presenteres lag én alene, deretter lag én og to sammen, og til slutt alle tre lagene sammen. I spillet vil alle disse lagene være aktive samtidig, men balansen mellom dem vil justeres dynamisk basert på intensiteten til handlingen i spillet. I tillegg til dette eksemplet er en versjon av temaet hvor alle lagene presenteres hver for seg lagt ved.

(Vedlegg: Karavan/Nommo v1.wav, Nommo v2.wav)

Det første laget i temaet er basert på samplinger av et afrikansk strengeinstrument, og synging. Det originale opptaket av Xhosa-folket som samplene er hentet fra er gjort av Dave Dargie i Sør-Afrika en gang mellom 1985 og 1998.³ I tillegg til disse samplene er det brukt et sample av en teslaspole som brukes som er rytmisk element som representerer Nommoenes våpenteknologi som baserer seg på teslaspoler. Lag to av temaet introduserer en elektrisk synthesizer-bass, og perkusive elementer. Disse elementene brukes for å øke spenningen i temaet, og gi en mer futuristisk stemning. Lag tre av temaet øker intensiteten ytterligere ved å legge til enda flere rytmiske elementer som skaper en mer kompleks polyrytmikk med de tidligere elementene og gjør temaet mer kaotisk.

Corporates-tema

Temaet til Corporates er basert på rock/metal, og elektronisk musikk, som hovedsakelig har et vestlig opphav. Denne musikken skal være hard og kald for å symbolisere de store selskapene i *Corp Wars*-settingen som kun bryr seg om teknologi og profitt.

For å skape et tema som passer til Corporates kultur har jeg brukt kun instrumenter som kan klassifiseres som elektriske, og instrumentene er mye preget av forvrengning og støy. Temaet er i likhet med de tre andre delt inn i ulike lag som i spillet justeres i forhold til hverandre basert på intensiteten til handlingen. Det første laget av temaet består av relativt støyete synthesizere som introduserer de to ulike variasjoner av temaet. Lag to av temaet legger til et elektrisk piano, samt en LFO-modulerte synthesizer-bass som er karakteristisk for dubstepsjangeren. Lag tre legger til elektriske gitarer

³Xhosa <https://www.youtube.com/watch?v=H5ufUjLQBxk> sist besøkt 08.05.15

med mye forvrengning, og en trommemaskin.

Vedlagt er to eksempler av temaet, ett hvor lagene legges over hverandre, og ett hvor lagene blir presentert hver for seg.

(Vedlegg: Karavan/Corporates v1.wav, Corporates v2.wav)



Figur 3.3: Skjermdump fra *Karavan*-prototype

Enklave-tema

Temaet til Enklave er en blanding av tradisjonell korpsinstrumentering, blandet med mer moderne elektronisk dansemusikk. Dette skal vise at Enklave-kulturen bygger på gamle folkelige tradisjoner, men har også blitt påvirket og modernisert av teknologi.

For å skape Enklave-temaet har tanken vært å bruke kun tradisjonelle akustiske instrumenter, men det er av praktiske hensyn i første omgang gjort med sample-baserte virtuelle instrumenter. Det hadde vært en fordel å spilt inn så mye som mulig av instrumentene med ekte instrumenter, men det er ingen konkrete planer om å gjøre dette foreløpig.

Enklave-temaet er i likhet med de andre delt inn i tre ulike lag. Det første laget presenterer en enkel trompetmelodi. De andre laget legger til flere blåsere, som forsterker rytmikken i temaet og gjør det mer funk-aktig med rytmiske 'hits'. Lag tre legger til slagverk basert på det velkjente 'amen break'

samplet, og en tuba som spiller veldig rytmisk, som en basslinje fra mer elektroniske sjangre som dubstep eller drum and bass. En av de viktigste inspirasjonene for Enklave-temaet er bandet Hot Botz Brass Band⁴ som spiller musikk lignende elektronisk dansemusikk med korpsinstrumentering.

Vedlagt ligger to eksempler på Enklave-temaet. En versjon hvor de ulike lagene presenteres hver for seg, og en hvor lagene legges over hverandre slik som de vil bli brukt i spillet.

(Vedlegg: Karavan/Enklave v1.wav, Enklave v2.wav).

Wastelander-tema

Temaet til Wastelanders skal være en blanding av de andre kulturenes musikk, men med mindre bruk av teknologi. Wastelanders-temaet er dermed en blanding av slagverk, som representerer det afrikanske perkusive elementet og Enklaves dansemusikk, banjo, kontrabass, og elgitar som representerer Corporates hardere vestlige stil, og tuba som representerer Enklaves korpstradisjon. Siden temaet skal blande elementer fra de tre andre temaene brukes det en blanding av både akustiske og elektriske instrumenter.

Det første laget i temaet består av en elektrisk gitar, i likhet med Corporate-temaet, men bruken en forvrengning som oppleves som noe 'varmere'. Det andre laget består av en banjo, en kontrabass, og en tuba. Det tredje laget legger til et tradisjonelt slagverk.

Tanken bak kombinasjonene av instrumenter i temaet er inspirert av musikk fra sørstatene i USA som baserer seg på kombinasjoner av ulike musikalske tradisjoner og instrumenter. På samme måte er musikken fra Wastelander kulturen en blanding av musikalske tradisjoner og instrumenter fra de andre kulturene i spillverdenen.

Vedlagt er to ulike versjoner av Wastlander-temaet som presenteres de ulike lagene. Den første versjonen består av de ulike lagene hver for seg, og den andre versjonen lagene lagt over hverandre slik som de blir brukt i spillet.

(Vedlegg: Karavan/Wastelanders v1, Wastelanders v2)

⁴Hot Botz Brass Band <http://www.reverbnation.com/hotbotzbrassband> sist besøkt 11.05.15



Figur 3.4: *Wastelander buggy*

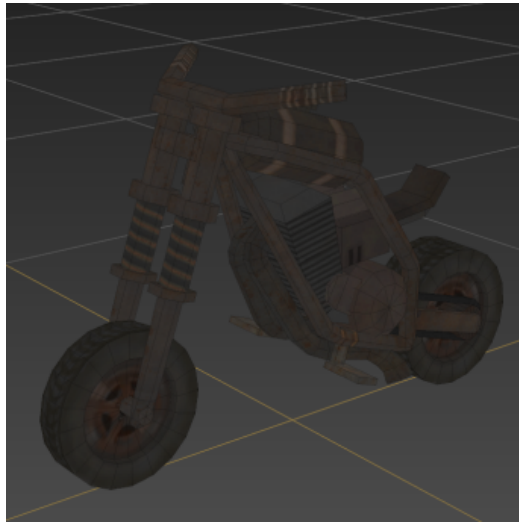
3.1.3 Lyddesign

Lyddesignet i *Corp Wars* er hovedsakelig sentrert rundt kjøretøy og våpen. I kamp kontrollerer man kjøretøy som skal forsvare seg med våpen, og man blir angrepet av fiender i kjøretøy med våpen. Til kjøretøyene har jeg laget lydeffekter for små og store kjøretøy. Disse kan varieres dynamisk for å signalisere forandring i kjøretøyets hastighet. I tillegg til dette har jeg laget lyder for skyting av små og store våpen, små og store eksplosjoner, og bruken av penger. De fleste lydeffektene er laget i Csound, men noen er også gjort som opptak. Det er foreløpig ingen konkrete planer for stemmeskuespill i *Corp Wars*, men det er mulig at det vil bli behov for dette etterhvert som spillene utvikler seg. Alle lydeffektene som foreløpig er laget til *Karavan* er samlet i et FMOD Studio-prosjekt slik de er implementert i spillet. De ulike lydeffektene er også beskrevet under.

(Vedlegg: Karavan/FMOD Karavan)

Maskingeværskudd

En av lydeffektene jeg har laget til *Karavan* er maskingeværskudd. Denne er laget i to versjoner, en for store maskingevær og kanoner knyttet til tankser o.l.(Vedlegg BigTurret), og en for mindre gevær knyttet til mindre kjøretøy og personell (Vedlegg SmallTurret). De to ulike versjonene er relativt like, med unntak av tonehøyde. Lydeffekten er laget som ett enkelt skudd, slik



Figur 3.5: *Wastelander bike*

at den kan trigges synkront med skuddhendelsen i spillet, og dermed ha variabel repetisjons-hastighet og varighet avhengig av våpentype. Lydeffekten vil trigges som både enkeltskudd, korte burst, og som full auto. Tanken bak implementeringen er å bruke en tilfeldig variasjon i pitch og nivå for hver gang lydeffekten trigges, slik at den ikke oppleves som monoton og unaturlig ved gjentatte repetisjoner. Lydeffekten er laget i programmeringsspråket Csound. (Vedlegg: Karavan/Csound/ bigTurret, smallTurret)

Ekspløsjoner

Ekspløsjonslydeffekten er i likhet med maskingeværskuddene laget i to ulike versjoner, en liten eksplosjon og en stor eksplosjon. Den lille eksplosjonen er tenkt for granater, missiler o.l, mens den store eksplosjoner er tenk for eksploderende kjøretøy o.l. Eksplosjonene vil også spilles av med tilfeldig pitch og nivå for å skape variasjon. Eksplosjonseffektene er laget i Csound. (Vedlegg: Karavan/Csound/ bigExplosion, smallExplosion)

Kjøretøy

I *Karavan* er det foreløpig to hovedtyper kjøretøy, store lastebiler og tankser, og mindre kjøretøy som biler og motorsykler. Til disse har jeg laget to ulike lydeffekter. Disse lydeffektene er litt mer kompliserte enn de andre, og er sammensatt av to ulike lydfile. En av lydfilene representeres lyden av

kjøretøyet når det står stille, med motoren i gang. Den andre lydfilen representerer lyden av kjøretøyet i fart med bevegende hjul/belter i kontakt med bakken. Kjøretøyenes hastighet i spillet kontrollerer deretter miksingene mellom disse to loopende lydfilene. I tillegg til denne miksingene blir lydfilenes pitch/hastighet økt i samsvar med hastigheten, for å gi inntrykket av at kjøretøyet hastighet øker, og motoren får et høyere omdreiningstall. De fire lydfilene som er brukt til de to ulike kjøretøytypene er alle laget i Csound. (Vedlegg: Karavan/Csound/ drivingBig, drivingSmall, engineBig, engineSmall)



Figur 3.6: *Wastelander truck*

Penger

I *Corp Wars* verdenen finnes det en egen valuta kalt Kash. Det er denne valutaen spillerne kan bruke i spillene for å kjøpe diverse ressurser og oppgraderinger av kjøretøy o.l. Når spilleren bruker disse pengene spilles det av en lydeffekt for å informere spilleren om at transaksjonen er gjennomført. Det ble først gjort forsøk på å lage en pengelydeffekt i Csound, slik at lydeffekten var i samme stil som de resterende lydeffektene, men dette viste seg å gi lite troverdige resultater. Det ble derfor valgt å gjøre et opptak av faktiske mynter istedenfor. Det ble gjort en rekke opptak av fallende mynter, og de to beste opptakene ble valgt ut. Siden denne lyden potensielt skal høres ganske ofte, og er en veldig distinkt lyd, valgte vi å bruke to lyder isteden for bare en for å skape et ekstra grad av variasjon. Hver gang lydhendelsen

for bruk av penger trigges velges en av disse lydfilene tilfeldig, og spilles av med en tilfeldig variasjon i pitch og nivå.

(Vedlegg: Karavan/ Kash1.wav, Kash2.wav)

Lyddesign-estetikk i forhold til visuell estetikk

Den visuelle stilen i spillet er ‘low poly’ med enkle 3D-modeller, og kan dermed minne om litt eldre spill. Lyddesignet er derfor laget på lignende måte som i dataspill fra perioden før bruken av digitale samples var utbredt og lydene ble syntetisert basert på kode ved hjelp av programmerbare lydgeneratorer og FM-syntese. Selv om lyddesignet jeg har laget blir implementert som samples, og ikke syntetisert i sanntid, har de fortsatt noe av den samme estetikken som disse tidlige spillene. Selv om estetikken på mange måter minner om eldre spill, er det visuelle fortsatt tredimensjonalt, og det er derfor ønskelig at den auditive opplevelsen samsvarer med dette, og at lyder har en troverdig bevegelse i spillverdenen med tanke på avstand og posisjon.

3.1.4 Implementasjon

Corp Wars-spillene lages i spillmotoren Unity, og det er planlagt å lansere dem til PC og mobil. Siden *Karavan* skal lanseres på flere ulike plattformer var det viktig å finne en løsning for implementeringen av lyden som fungerer for alle plattformene, og som gjorde det enklest mulig å implementere. Unity har innebygget en del grunnleggende audiofunksjonalitet, men for å få til et mer komplekst og dynamisk lyddesign kreves det programmering, bruk av eksterne plugins, eller *middleware*. Av *middleware* støtter Unity blant annet FMOD Studio og FMOD, samt en rekke plugins som kan utvide lydfunksjonaliteten⁵.

Musikken i spillet skal være dynamisk, og baserer seg på en kombinasjon av horisontal og vertikal komposisjon. Den horisontale variasjonen er mellom de ulike temaene knyttet til de fire ulike grupperingene i spillverdenen. Den vertikale variasjonen er basert på at hver av de fire temaene har tre ulike lag av instrumenter, hvor miksing av lagene kontrolleres av intensiteten til handlingen i spillet til enhver tid. Siden musikken først ble laget for spillet *The Siege*, hvor spilleren mer eller mindre konstant blir angrepet av fiender, er musikken laget med tanke på en kampsituasjon, og er dermed rask og

⁵Unity Asset Store <https://www.assetstore.unity3d.com> sist besøkt 08.05.15

intensiv. Siden *Karavan* består av spillmekanikker knyttet til både kamp og utforskning, er det naturlig å bruke kampmusikken originalt laget for *The Siege*, og bruke den i kampsituasjonene i *Karavan*. Om denne musikken ender opp med å bli brukt i kun *Karavan*, byttet ut med ny musikk og brukt i *The Siege*, eller blir brukt i begge spillene er foreløpig usikkert, og vil bli bestemt etterhvert som de to spillene utvikler seg. Siden alle spillene i *Corp Wars*-serien foreløpig er ganske tidlig i produksjonsfasen er lite fastsatt enda.

I startfasen av prosjektet brukte jeg programmeringsspråket C# internt i Unity for å lage et system for å kontrollere det dynamiske musikksporet. Dette systemet fungerte for så vidt greit til å kontrollere miksing mellom de ulike temaene og lagene, men er ikke nødvendigvis så brukervennlig eller enkelt å modifisere.

(Vedlegg: *Karavan/MusicLogic.cs*)



Figur 3.7: *FMOD Studio*

De nyere versjonene av Unity har inkludert støtte for FMOD Studio, og ikke kun FMOD Ex, som i tidligere versjoner av Unity. Basert på dette, kombinert med min erfaring med bruk av FMOD fra min bacheloroppgave, bestemte vi oss for å bruke FMOD Studio til implementeringen av lyden i *Karavan*. Integrasjonen av FMOD Studio i Unity fungerer for alle plattformene spillet er planlagt for, og gjør det enklere å lage et mer komplekst lyddesign. For å integrere FMOD Studio og Unity kreves det litt arbeid og programmering for å spille av og kontrollere lydhendelser i FMOD fra Unity, men når dette er satt

opp er det enklere å legge til nye lyder og redigere gamle lyder uten å påvirke resten av spillet eller forandre kode. Dette gjør det også mye enklere for en lyddesigner og spillprogrammerer å jobbe parallelt på det samme prosjektet uten å komme i veien for hverandre.

Integreringen av FMOD Studio i Unity består av fire hovedsteg. Det første steget er å ha et prosjekt i FMOD Studio med de ønskede lydhendelsene. Disse lydhendelsene legges deretter til i en eller flere 'banker' i FMOD Studio, og prosjektet bygges. Det andre steget er å sette opp Unity for å kunne bruke disse bankene med lydhendelser fra FMOD Studio. For å gjøre dette må en spesiell pakke importeres i det aktuelle Unity-prosjektet. Denne pakken kan lastet ned fra hjemmesiden til FMOD.⁶ Når denne pakken er importert dukker en egen FMOD-meny opp i oppgavelinjen til Unity. Det tredje steget er å importere bankene som ble bygget fra FMOD Studio inn i Unity via FMOD-menyen. Når dette er gjort vil lydhendelsene være tilgjengelige i Unity. For å oppdatere lydhendelsene må disse endres i FMOD Studio, prosjektet bygges, og bankene importeres eller oppdateres på nytt i Unity via FMOD-menyen. Det er også mulig å sette opp prosjektene til å automatisk oppdateres, slik at lydhendelser kan endres i FMOD Studio og ta effekt i Unity i sanntid, men det er ikke gjort for dette prosjektet. Det fjerde steget er å bruke disse lydhendelsene via kode i Unity. Dette kan gjøres ved å skrive et eget script, eller bruke ett av eksemplene som er inkludert i pakken importert i steg to. Det er også mulig å ta utgangspunkt i disse scriptene og modifisere de etter behov.

Musikken i FMOD kontrolleres fra Unity via to parametre som er koblet til tema og intensitet. Når parameteren for tema går over en bestemt grenseverdi, vil FMOD skape en overgang fra det nåværende musikalske temaet til det nye musikalske temaet. Parameteren for intensitet fungerer ved å gradvis forandre miksen mellom de ulike lagene i temaet basert på en forhåndsbestemt kurve. En økende intensitet fra null til én vil f.eks føre til at det første laget i det nåværende temaet fades inn til maks, deretter lag to, og deretter lag tre. Temaet er satt opp slik at de ulike lagene alltid spilles av synkronisert, og overganger mellom ulike temaer vil fortsette med samme intensitet.

I tillegg til eksempelfilene av temaene har jeg lagt ved FMOD prosjektet slik som det er implementert i spillet. Dette kan åpnes i FMOD Studio for å se oppsettet av temaene, og teste hvordan dette fungerer i sanntid. Alle temaene og lagene er samlet i en enkel lydhendelse. Ved å spille av lydhendelsen og

⁶FMOD Studio Unity integration <http://www.fmod.org/download/> sist besøkt 11.05.15

justere parameterene for tema og intensitet kan man få inntrykk av hvordan oppsettet fungerer i spillet.

Det resterende lyddesignet er også implementert via FMOD. Dette gjør det enklere å variere enkelte lydhendelser med tilfeldig pitch og nivå uten å påvirke andre lydhendelser. På de korte enkeltstående lydene som skudd, eksplosjoner, penger brukes det et enkelt oppsett i FMOD hvor hver enkelt lydhendelse får en tilfeldig pitch og nivå innen forhåndsbestemte marginer for hver avspilling. For kjøretøylydene er det et litt mer komplisert oppsett. Som tidligere nevnt består kjøretøylydene av to lyder, en for motoren, og en for kontakt med bakken. Når kjøretøyet står stille høres kun motorlyden, men etterhvert som farten øker, vil bakkelyden fades inn. I tillegg til dette justeres også hastigheten/pitchen til lydene, slik at en høyere hastighet oppleves som en høyere intensitet med høyere omdreiningstall for motoren.

(Vedlegg: Karavan/FMOD Karavan)

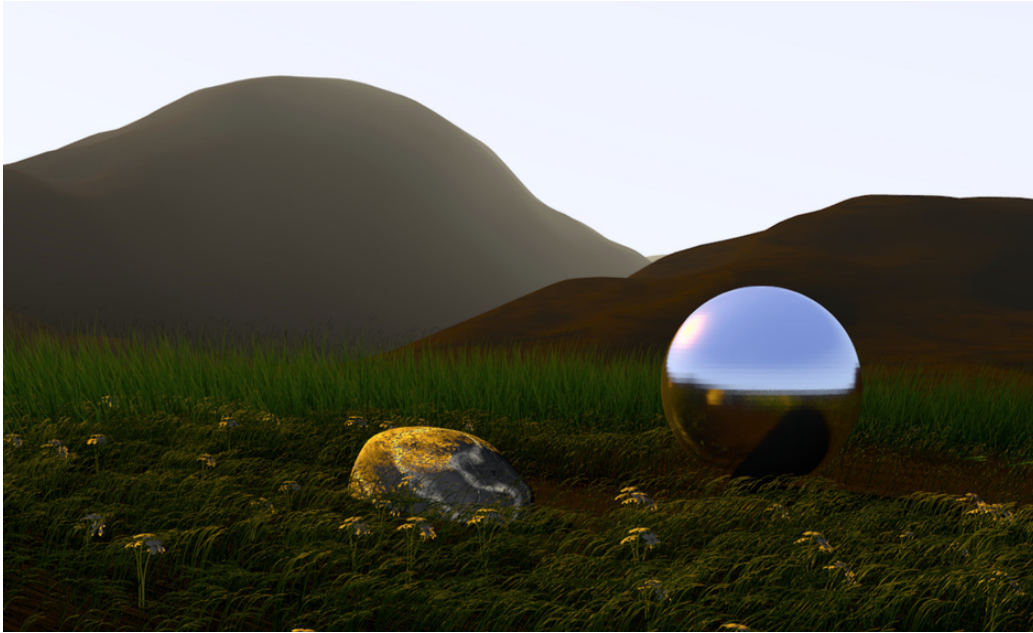
3.2 *Shattered*

3.2.1 Om Spillet

Jeg kom i kontakt med Gridmedia via instituttet, etter at de tok kontakt for å se om noen var interessert i å lage musikk og lydeffekter til et spillprosjekt kalt *Shattered*. Ettersom denne forespørselen kom like etter at jeg hadde begynt på mastergraden, og hadde planer om å jobbe med lyd i dataspill, var det naturlig å bli enig om et samarbeid og utarbeide en oppgavebeskrivelse (Vedlegg: *Shattered*/Oppgavedefinisjon.pdf).

Litt om *Shattered*

Shattered er et lite fysikksimulerings dataspill. Brukeren styrer en kule som kan trille rundt i en verden bestående av mange spennende opplevelser. Vi tror det blir svært viktig at *Shattered* består av spennende og interessante opplevelser, mye grafikk, mange gøyale puzzles", mange fysikkeffekter, musikk og lydeffekter, m.v. Vi ser for oss å ha flere 'levels' hvor bakgrunnsverdenen og 'theme' blir endret betydelig, som f.eks. på en sjørøverøy, i en Inka jungelby, i en egyptisk ørkenby, osv. I *Shattered* vil brukeren få mange underholdende timer med spennende opplevelser og 'puzzles'.



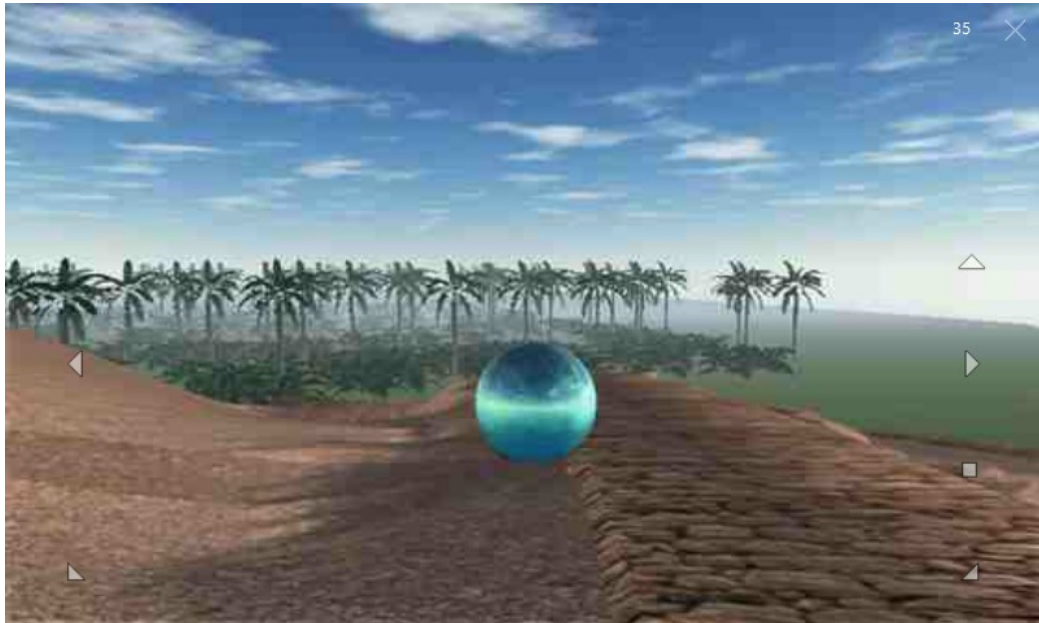
Figur 3.8: Konseptskisse for *Shattered*

(Hentet fra designdokumentet til spillet, Vedlegg: Shattered/ Beskrivelse.pdf)

3.2.2 Musikk

Musikken i *Shattered* baserer seg på ulike musikalske temaer, som er knyttet til ulike steder/nivåer i spillverdenen. Det første planlagte utkastet til spillet består av fem nivåer med ulike kombinasjoner av musikk. Det første nivået består av et fjell, og en jungel. Planen var å ha et musikalsk tema knyttet til fjellet med ‘musikk som blåsende vind’, og et tema knyttet til jungelen med ‘musikken full av jungellyder’. For det andre nivået var planen å bruke den samme musikken som i det første nivået, i tillegg til et sjørøvertema. Nivå tre skulle være knyttet til Inkariket, og dermed ha et musikalsk tema knyttet til dette, samt bruk av jungeltemaet. Temaet for nivå fire er Egypt, og skal bestå av en pyramide og en labyrint. Det var dermed planer om et musikalsk tema for pyramiden, og ett for labyrinten inne i pyramiden. Planen for nivå fem var et tema med ‘vind og hagl’.

Siden det første nivået av spillet skulle lages først, og gjennomføringen av de andre nivåene fortsatt var litt usikre, var fjell og jungel-temaene prioritert først. På grunn av at Gridmedia har hatt ansvaret for å implementere musikken i spillet, har det vært vanskelig å lage et system for dynamisk



Figur 3.9: Fjellnivå i *Shattered*-prototype

variasjon av musikken. Musikken er derfor hovedsakelig komponert med tanke på kun horisontal variasjon mellom de ulike temaene, med lineær looping med forhåndsbestemte struktur av hvert enkelt tema. Temaene er også laget med tempo og skalaer som gjør musikalske overganger mellom temaene enklere.

Det ble laget flere utkast til de forskjellige temaene, og vi kom til slutt frem til en versjon av fjelltemaet og jungeltemaet som ble godkjent av Gridmedia som ferdig til å brukes. I tillegg til disse to temaene er også noen av utkastene til de andre temaene foreløpig brukt i tillegg for å teste implementasjonen.

Fjelltema

Fjelltemaet er laget med tanke på omgivelsene rundt spilleren i det man starter spillet. I starten av spiller befinner man seg på toppen av et fjell, og temaet begynner derfor rolig med et vindinstrument. Tanken bak vindinstrumentet er at dette i tillegg til å være et musikalsk element skal fungere som en miljølyd som informerer om omgivelsene.

Strukturen i temaet er laget med tanke på at dette skal kunne loopes i spillet. Siden temaet potensielt kan være aktivt over lengre perioder, avhengig av hvor lang tid spilleren bruker til neste område, er det viktig at temaet ikke repeterer for ofte, og inneholder nok variasjon til at det ikke fører til



Figur 3.10: Jungelnivå i *Shattered*-prototype

lyttetretthet hos spilleren. Slik temaet er strukturert i eksemplet som er vedlagt varierer intensiteten syklisk ved å legge til og trekke ifra instrumenter. Temaet består totalt av fire ulike variasjoner. Den første variasjonen er kun et vindinstrument som fungerer som miljølyd. I den andre delen legges det til synthesizere. Den tredje delen introduserer en trommemaskin, og den fjerde delen legger til en rytmisk synthesizer melodi.

(Vedlegg: *Shattered/Fjell.aiff*)

Jungeltema

Jungeltemaet er i likhet med de andre temaene i *Shattered* laget med tanke på omgivelsene musikken skal brukes i. Det andre nivået i spillet har en jungel-tematikk, og det musikalske temaet for dette området bruker dermed en del elementer knyttet til dette. Det første musikalske elementet i jungeltemaet er ment som en jungel-atmosfære med lyden av insekter o.l. Denne lyden brukes både som en diegetisk miljølyd for området, og som et ikke-diegetisk musikalsk element. Etter jungel-atmosfæren introduseres 'jungel-trommer' som også kan tolkes som både diegetiske og ikke-diegetiske. Disse kan i en diegetisk kontekst tolkes som at jungelen er full av trommende fiender, noe som gjør spillet mer intenst og spennende. Etter disse instrumentene introduseres en rekke elektroniske elementer, som en

trommemaskin, synthesizer-bass, og synthesizer-melodi. Disse elektroniske instrumentene har kun en musikalsk funksjon, og fører temaet over i en mer *pop/dance* type musikk.

Jungeltemaet er i likhet med fjelltemaet laget med tanke på å potensielt kunne deles inn i ulike lag som kan brukes i en vertikal komposisjonsstrategi basert på intensitet, men bruker foreløpig kun som en lineær loop med fast struktur.

(Vedlegg: Shattered/Jungel.aiff)

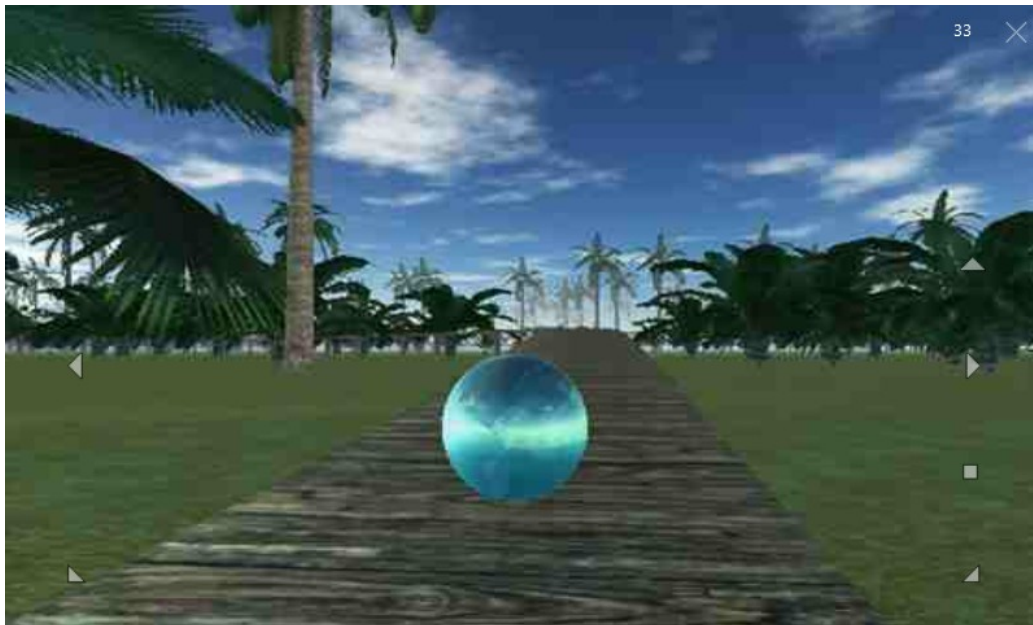
Diegese

Ved å bruke miljølyder som musikalske elementer i musikken knyttes musikken og lyddesignet sammen, og skillet mellom diegetisk og ikke-diegetisk blir delvis uklart. Vindlyden i fjelltemaet kan beskrives som en miljølyd som passer med omgivelsene, som er diegetisk, men spiller også en melodi som passer resten av musikken, som er ikke-diegetisk. På samme måte kan jungellydene og jungeltrommene i jungeltemaet tolkes som både diegetiske og ikke-diegetiske. På denne måten oppfylles ønsket om å skape en sammensmelting av miljølyder og musikk. Dette er gjort både av praktiske hensyn for å begrense antall lydspor, og dermed mengden nødvendig lagringsplass med tanke på plattformen spillet skal lanseres på, og som et kreativt virkemiddel for å skape musikk med tilknytning til de ulike nivåene i spillverdenen.

3.2.3 Lyddesign

Av lyddesign til *Shattered* er det grunnet begrenset tid og prioritering av musikken bare noen av de planlagte lydeffektene som foreløpig har blitt laget. De lydeffektene som er blitt prioritert er de som er knyttet til kulens bevegelser, og som dermed høres igjennom mesteparten av spillet. Siden kula skal kunne bevege seg på to forskjellige måter, stillestående rotasjon og rulling, som skal tilpasses alle de ulike lagene kula kan bevege seg på ender man opp med å trenge en del lyder. I tillegg til dette trengs lyder for når kula kolliderer med ulike gjenstander, og når kula begynner å gå i oppløsning og knuser. De lydeffektene som er laget for kula er rotasjon og rulling på tre og stein. Eksempler på hvordan lydeffektene skal implementeres er gjort i FMOD for å enkelt demonstrere disse, selv om implementasjonen i det endelige spillet ikke vil bruke FMOD.

(Vedlegg:Shattered/ FMOD Shattered)



Figur 3.11: Rulling på treverk i *Shattered*-prototype

Rotasjon

Lydene for rotasjon spilles av når kulen står stille på ett sted, men roterer for å endre retning. For å skape disse lydene er det brukt et tomt syltetøyglass som er skrapet mot ulike underlag. Utfordringen med denne lydeffekten er å skape en sammenhengende jevn lyd som kan loopes. Lydeffekten for rotasjon på stein er laget ved å skrape/rottere glasset på betong, og lydeffekten for rotasjon på tre er laget ved å skrape/rottere glasset på en treplate. På grunn av at betongen er mer ujevn og mindre glatt enn treverket er lyden for stein noe mer ujevn enn den for tre. Opptakene er gjort med en mikrofon med karakteristikk av typen hypertyre, en såkalt shotgun-mikrofon, for å minske mengden bakgrunnsstøy. Bruken av en slik shotgun-mikrofon gjør det enklere å gjøre opptak med høy kvalitet uten å trenge et helt lydstudio, og passer bra til å gjøre opptak hvor kun deler av lyden er i fokus, som for eksempel på et filmsett.

(Vedlegg: *Shattered*/Rotasjon Stein.wav, Rotasjon Tre.wav)

Rulling

Lydeffektene for kulens rulling er også laget for stein og tre. Disse lydene spilles av i en loop når kula ruller. Valget av lydfil baserer seg på underlaget kula ruller på. Kulens hastighet skal også dynamisk bestemme pitchen og nivået til avspillingen av lydfilen. Når kula begynner å rulle vil lydfilen fades inn, og pitchen vil øke med hastigheten innen for et forhåndsbestemt intervall. Når kula stopper vil lydfilen fades ut. Opptakene er i likhet med rotasjonslydeffektene gjort med en shotgun-mikrofon for å isolere lydkilden og begrense bakgrunnsstøy. Lydene er skapt ved å rulle et tomt syltetøyglass langs et betonggulv, og på en treplate.

(Vedlegg: Shattered/Rulling Stein.wav, Rulling Tre.wav)

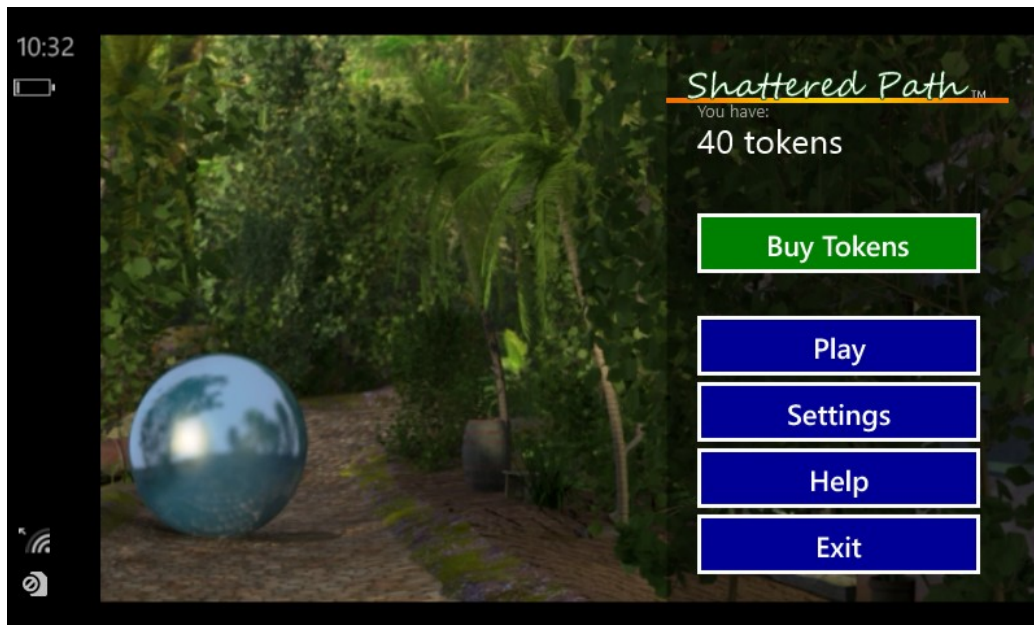
Lyddesign-estetikk i forhold til visuell estetikk

Den visuelle estetikken i *Shattered* er tenkt som en virkelighetstro fysikk-simulering med ‘.høy kvalitets/avansert 3D grafikk, lyssetting, effekter og skyggelegging’(vedlegg shattered beskrivelse). Lyddesignet er tenkt å bidra til denne realistiske estetikken, og er dermed basert på lydopptak fra den virkelige verden, uten overdreven redigering eller effekter som kan oppleves som for ‘hyperrealistiske’. Spillet er også visuelt tredimensjonalt, og planen er derfor også at lyden skal samsvare med det visuelle og oppleves som tredimensjonal med realistisk nivå basert på avstand, posisjon o.l.

3.2.4 Implementasjon

Shattered implementeres for Windows Phone 8.1, og er laget i Gridmedia sin interne spillmotor Geelix Game Engine. Spillet skal også strømmes fra en server til en klient på brukerens mobiltelefon. Strømmingen fungerer ved at kontrollsignaler sendes fra klienten på brukerens mobiltelefon til serveren, som deretter strømmer video og lyd tilbake til brukeren. Denne løsningen gir muligheter til å kjøre programvare som vanligvis krever for mye ressurser til å kjøre på en mobiltelefon, men har også noen begrensninger basert på nettverkforbindelsen med tanke på latens mellom brukerens interaksjon, og respons fra systemet, samt kvaliteten på video- og lyd-strømmen.

På grunn av at spillet er laget med tanke på denne strømmeløsningen i Geelix Game Engine, og kun foreløpig er planlagt for Windows Phone 8.1 er det dessverre ikke mulig å legge ved en spillbar demo av spillprosjektet.



Figur 3.12: *Shattered meny*

Gridmedia ønsket heller ikke at noen av spillet kildekode skulle legges ved denne rapporten. Spillet er planlagt å etterhvert bli tilgjengelig via Windows Phone Store. Siden implementeringen av lyden i spillet skal skje via DirectX i Geelix Game Engine, noe som jeg ikke har noen erfaring med, vil selve implementeringen gjøres av Gridmedia.

Kapittel 4

Anvendelse av teori i prosjektene

4.1 Anvendelse av teori i *Karavan*

4.1.1 Musikk

Karavan er hovedsakelig et strategispill, med elementer av spenning/eventyr. Basert på teorien diskutert tidligere er spillertypen *administratoren* tettest knyttet til spillsjangrene eventyr og strategi. Strategispill spilles også hovedsakelig av spillere som kan beskrives som dedikerte. Den tydeligste spillertypen i målgruppen til *Karavan* er dermed *administratoren* av typen dedikert. Musikksjangeren som er tettest assosiert med eventyrspill er *elite*, og musikkjangrene knyttet til strategispill er *elite* og *rock*.

Det er dermed naturlig for *Karavan* å ha musikk som faller inn under disse kategoriene. Undersøkelsen som beskrives i teoridelen trekker ingen slutninger om hvordan spillertypens beskrivelse som dedikert eller avslappet påvirker spillertypens musikkpreferanse, men man kan kanskje argumentere for at en spiller med en mer ekstrem spillestil ofte vil foretrekke mer ekstrem musikk. På bakgrunn av dette kan man argumentere for at dedikerte spillere heller mer mot *rock* enn *elite*, og avslappede spillere heller mer mot *elite* enn *rock*. Dette er kun basert på antagelser, og ikke på data, men virker å stemme overens med hvor ekstreme de ulike spillsjangrene er, i forhold til hvilke musikkjangre som er knyttet til disse.

Om man ser på musikken som er laget til spillet er det en del variasjon i sjanger, men ingen av temaene passer naturlig inn i typene som er definert i teorien. Corporates-temaet har elementer av *rock*, men også elektroniske

elementer som kan passe inn under *pop/dance*. Nommo-temaet har også noen elektroniske elementer som kan knyttes til *pop/dance*, men de afrikanske elementene passer ikke naturlig inn i noen av sjangerdefinisjonene. Enklave-temaet har elementer som kan passe inn i både *urban*, *pop/dance* og *elite*. Brassband elementene kan kanskje klassifiseres som *elite*, men funk-elementene og de elektronika-inspirerte elementene passer bedre i *urban* eller *pop/dance*. Wastelander temaet har elementer som passer inn i *rock*, og elementer som passer inn i *urban*.

Det er med andre ord lite klart samsvar mellom musikkjangrene brukt i spillet, og musikkpreferansen til den mest typiske spillertypen i målgruppen basert på spillets sjanger. Det er noe overlapp mellom spillertypepreferanser og musikkjangre, men dette har ikke vært den avgjørende faktoren for valg av musikk ved designet av spillet. Kravene til spillets musikk er designet mer med tanke på å passe inn i historien til spillverdenen, og samsvare med den musikken de ulike grupperingene i spillverdenen hadde laget. Funksjonen til musikken er dermed i større grad brukt for å bygge spillets stemning og miljø, enn som underholdning for den største målgruppen til spillet.

Komposisjonstrategien som er valgt for musikken i *Karavan* er en kombinasjon av horisontal og vertikal struktur. De forskjellige temaene er organisert horisontalt, med de ulike intensitetene for hvert tema organisert vertikalt. På denne måten kan man enkelt bytte mellom ulike temaer og intensiteter.

4.1.2 Lyddesign

Lyddesignet jeg har laget til *Karavan* er stilisert for å passe opp imot den visuelle stilen til spillet, og på den måten sammen skape et troverdig helhetlig uttrykk. Lydene er også tenkt å brukes i 3D, og skal følge de ulike objektene lyden er koblet imot. Det er også brukt variasjon i pitch og nivå for å unngå problemer med lyttetretthet, og for å skape et mer naturlig lyddesign. Det er foreløpig ikke laget noe miljølyd til spillet, men det er sannsynlig at dette vil bli lagt til etterhvert som spillet utvikler seg.

Lydeffektene er diegetiske, med unntak av pengelyden som er knyttet opp imot brukergrensesnittet. Klassifiseringen av de andre lydeffektene er litt i grenseland mellom karakterlyder og generelle effektlyder avhengig av konteksten de brukes i.

4.1.3 Plattform

Karavan er tenkt først som et PC spill, men skal trolig også lanseres på mobil etterhvert. Siden Unity støtter de fleste populære mobile plattformene lanseres spillet muligens på flere av disse, men hovedfokuset blir trolig først på Android, som ved fullføringen av denne oppgaven har den største markedsandelen av operativsystemer for mobile plattformer, etterfulgt av iOS som har den nest største markedsandelen. Valget om å fokusere på de meste populære plattformene først kan ha mye å si for spillets suksess, men medfører også en større grad av konkurranse og økt mulighet for et mettet spillmarked på plattformen.

PC er potensielt den kraftigste dataspill-plattformen, og det er dermed ikke så mange begrensninger på maskinkraft og minne som på de andre plattformene. PC-er har derimot ikke standardisert maskinvare eller programvare, noe som gjør at dataspill laget for PC burde kunne tilpasse seg ulike operativsystemer og maskinvarer. På grunn av dette er det vanligvis brukeren sitt ansvar å forsikre seg om at systemet støtter spillet, og ikke produsenten. Produsenten bør selvfølgelig prøve å tilpasse spillet slik at det er spillbart på så mange systemer som mulig.

Siden *Karavan* har brukt spillmotoren Unity kombinert med *middleware*-programvaren FMOD er implementeringen av lyd i spillet relativt enkelt å gjennomføre. Wwise og Unity gjør det også enkelt å bygge prosjektet til ulike plattformer med tanke på begrensninger knyttet til de disse. Med tanke på lyden er det dermed ikke så mange ekstra utfordringer til å bygge spillet for mobil enn til PC.

4.2 Anvendelse av teori i *Shattered*

4.2.1 Musikk

Shattered har elementer fra flere ulike spillsjangre, deriblant gåte, plattform, og kjørespill. Spillertypen tettest knyttet til plattformspill er *vandrereren*. *Vandrereren* er åpen for mye forskjellig musikk, noe som kanskje kan forklare at plattformspill ikke har noen konkret tradisjon for musikalsk sjanger. Det kan også tenkes at åpenheten i musikk sjanger er det som tiltrekker en viss type spillere. Gåtespill spilles av en rekke ulike spillertyper, deriblant *administratoren*, *vandrereren*, og *deltageren*. Basert på undersøkelsen om spillertyper har heller ikke gåtespill noen fast tradisjon for valg av musikalsk

sjanger. Kjørespill tiltrekker også flere ulike spillertyper, som regel av typen avslappet. Basert på dette skulle man tro at kjørespill heller ikke har noen fast musikalsk sjanger, men i motsetning til gåte og plattformspill har kjørespill en lang tradisjon for bruk av elektronisk dansemusikk. Dette kan ha en sammenheng med at kjørespill ofte kan ha en litt monoton spillopplevelse, og musikken kan brukes for å gjøre denne mer spennende. *Shattered* er ikke et tradisjonelt kjørespill, men har overlappende elementer med tanke på ferdighetene som kreves for å kontrollere kulen og unngå at denne knuses. Basert på hvordan *Shattered* passer inn i de ulike spillsjangrene vil det dermed være naturlig at spillet bruker musikk som passer med spilleverdenens omgivelser, og eventuelt elementer fra elektronisk dansemusikk.

Om man ser på temaene som er laget for *Shattered* passer de alle hovedsakelig inn under *pop/dance*, med en del elementer knyttet til spillverdenens ulike nivåer. Fjelltemaet har kun elektroniske synthesizer og trommemaskin-elementer, noe som gjør at det kan beskrives som elektronisk dansemusikk. Vindinstrumentet brukes også som en miljølyd for å knytte musikken til spillnivåets omgivelser. Jungeltemaet er også basert på synthesizere og trommemaskiner, og er svært rytmebasert. Instrumentet med jungellyder og instrumentet med jungeltrommer knytter også musikken sammen med spillnivåets omgivelser på samme måte som vindinstrumentet i fjelltemaet.

Komposisjonsstrategien brukt for musikken i *Shattered* er en horisontal struktur med loopende lineære segmenter. Hvert av temaene er laget for å kunne loope sømløst. Man kan deretter bytte mellom temaene ved å fade over til et annet tema. Temaene er komponert med tanke på å også kunne bruke en horisontal struktur basert på intensitet, men de er foreløpig kun brukt som lineære looper med forhåndsbestemt struktur. Temaene er også komponert i samme tempo og compatible tonearter slik at overganger mellom dem skal kunne gjøres så sømløse som mulig.

4.2.2 Lyddesign

Lyddesignet som foreløpig er laget til *Shattered* er kun karakterlyder knyttet til kulens bevegelser. Karakterene er planlagt å implementeres som 3D-lyder i spillmotoren, slik at disse står i stil med den visuelle delen av spillet. Lyddesignet er laget for å stå i stil med den mer realistiske estetikken som har vært målet for *Shattered*.

I tillegg til karakterlydene er det integrert miljølyder i musikken, og det er dermed ikke planer om å bruke noen egne miljøspor. Det er foreløpig ingen

lyder knyttet til brukergrensesnitt, eller stemmelyder. Spiller vil trolig ikke trenge noen stemmelyder, men det er stor sannsynlighet for at lyder knyttet til brukergrensesnittet vil bli nødvendig.

4.2.3 Plattform

Med tanke på det lydlige har *Shattered* hatt en del begrensninger knyttet til plattform. Spillet skal i første omgang kun lanseres på mobil, noe som medfører en del begrensninger på maskinvare. Det er også laget for Windows Phone, som slik det er ved fullføringen av denne oppgaven er blant de minste av de populære mobilplattformene med tanke på markedsandel, noe som kan ha en del å si for mengden og kvaliteten på tilgjengelige verktøy. Valget med å fokusere på en av de mindre plattformene kan også ha mye å si for spillets suksess. En liten plattform betyr færre antall potensielle spillere, men medfører også mindre konkurranse og mindre sannsynlighet for et mettet marked for spill på den plattformen.

Mobiltelefoner kan ha ganske varierende maskinvare, og er blant de spillplattformene med minst maskinkraft og minne. En stor andel av den tilgjengelige maskinkraften og minne brukes til det grafiske, og andelen til overs for lyd kan bli ganske liten. Manglende maskinkraft kan begrense mengden prosessering av lyder i sanntid som 3D-lyd og effekter. Mengden minne vil begrense antallet og kvaliteten på lydfilene som kan brukes. Siden *Shattered* bruker en løsning for fjernstrømming av spillet slipper man unna en del av disse problemene, men skaper også noen nye. Siden spillet skal kjøres på en server har man potensielt mer minne og maskinkraft, men dette vil være avhengig av antallet spillere på serveren. Man skaper også noen nye problemer knyttet til nettverket med tanke på hastighet, latens, og databruk.

Som nevnt i teoridelen burde dataspill laget for mobile plattformer kunne spilles helt uten lyd. I *Shattered* er ikke noe av lyddesignet kritisk for spillopplevelsen, men bidrar til og underbygge det visuelle. Spillet kan dermed fint spilles uten lyd, selv om lyden vil gi opplevelsen ekstra verdi.

Det å bruke en egenprodusert spillmotor skaper også noen utfordringer. Å lage sin egen spillmotor gir store muligheter for implementeringen av lyd, men gjør at dette ofte må gjøres på et mye lavere nivå med tanke på programmering enn med kommersielle spillmotorer. Dette gjør at implementeringen av lyd krever mer programmering, kan bli vanskeligere, og tar lengre tid.

Med tanke på plattformen er ikke den måten musikken foreløpig er

implementer på den mest effektive. Slik det er nå er hvert musikalske tema laget som en lineær loop som inneholder hele den musikalske strukturen i en enkelt fil. Dette betyr at det er en del gjentakelse av de samme delene flere ganger i løpet av filene, og filene blir ganske lange og tar stor plass. For å unngå dette problemet kunne man ha lagret alle de ulike musikalske delene kun en gang, enten som egne filer, eller sekvensielt i en felles fil, deretter kunne man brukt kode for å spille av rett del til rett tid. Dette sparer en del lagringsplass, og gir muligheten til å dynamisk forandre musikkens struktur, men skaper også en del problemer med tanke på synkronisert avspilling og overganger mellom de ulike delene. Av praktiske hensyn har vi valgt å kun bruke lineære segmenter med forhåndsbestemt struktur, selv om dette bruker mer lagringsplass og gir mindre muligheter for dynamisk musikk.

4.3 *Karavan* versus *Shattered*

Karavan og *Shattered* er på mange måter ganske ulike dataspill. Begge spillene er i 3D, og er laget med tanke på en mobil plattform, men ellers er spillsjanger, målgruppe, estetikk, og implementering ganske så ulike.

Med tanke på spillsjanger er *Karavan* hovedsakelig et spenning/eventyr og strategi-spill, mens *Shattered* hovedsakelig et gåte/plattform/kjørespill. *Karavan* er i tillegg satt i fremtiden, og har en veldig stilisert estetikk, mens *Shattered* har en mer realistisk estetikk, og har temaetikk knyttet til historiske steder fra fortiden. Basert på disse forskjellene i både sjanger, estetikk, og setting er det ganske tydelig at de to spillene har ulike målgrupper. Av spillertyper er målgruppen for *Shattered* hovedsakelig *vandrerer*, mens det for *Karavan* er *administratoren*. *Shattered* er også laget hovedsakelig med tanke på avslappede spillere, mens *Karavan* fokuserer i større grad på dedikerte spillere.

Denne forskjellen i sjanger, setting og estetikk gjenspeiles også i valget av lyder. Musikalsk bruker *Shattered* kun elektroniske instrumenter, noe som kan tolkes mer som futuristisk, mens *Karavan* bruker en større grad av akustiske instrumenter. Dette kan med det første virke rart med tanke på spillenes miljø, siden *Shattered* baserer seg på fortid og *Karavan* er satt i fremtiden. I *Shattered* er dette fordi sjangeren av musikken ikke er direkte påvirket av miljøet, men i større grad av spillets sjanger. Det er en del elementer i musikken knyttet til miljøet, men ikke når det gjelder valg av instrumenter med tanke på akustisk versus elektrisk. I *Karavan* er musikken er blanding er tradisjonell musikk fra de ulike grupperingene, kombinert med innflytelser

fra den moderne teknologien som har overlevd katastrofene. Det er dermed naturlig å bruke en blanding av akustiske og elektriske instrumenter, siden gamle instrumenter kan brukes i fremtiden, mens nye instrumenter ikke kan brukes i fortiden.

Lyddesignet er i begge spillene laget med tanke på å gjenskape et troverdig tredimensjonalt rom, men i *Shattered* er lydeffektene laget med tanke på høyest mulig grad av realisme, kontra det mer stiliserte lyddesignet til *Karavan*. I begge tilfellene er målet at lyddesignet skal stå i stil med den visuelle estetikken. I *Shattered* betyr dette et lyddesign med høy grad av realisme, mens i *Karavan* betyr dette et mer stilisert lyddesign inspirert av spill fra en tidligere epoke i spillhistorien. I begge spillene er lytteperspektivet objektivt og avhengig av kamerater. Det vil si at den diegetiske lyden som høres er den som kan høres av alle ved kameraets posisjon i scenen.

Målet for implementeringen er i begge spillene ganske like med tanke på dynamisk musikk og lyddesign, men selve implementeringen er noe ulik. Ulikheter i valg av programvare gjør implementeringsprosessen en del enklere i *Karavan* enn i *Shattered*. I *Karavan* blir de dynamiske egenskapene til lydeffektene og musikken i stor grad tatt hånd om av programvaren, mens i *Shattered* må mye av denne funksjonaliteten programmeres selv. Disse forskjellene er en grunn til at implementeringen i *Karavan* har kommet mye lengre, og er mer kompleks enn i *Shattered* ved fullføringen av denne oppgaven. Noe av grunnen til dette kan også være valg av plattform. Siden Windows Phone er en betydelig mindre plattform enn PC, Android eller iOS, er det muligens den med dårligst støtte med tanke på implementering av lyd.

Kapittel 5

Arbeidsprosess

5.1 Arbeidsprosess *Karavan*

Arbeidet i samarbeid med Kybernesis har gått veldig bra, og har vært en positiv opplevelse. Kybernesis har vært flinke til å definere konkrete oppgaver som må gjennomføres i deres prosjekt. For å definere og delegere oppgaver har vi brukt Asana¹, som er en plattform for administrasjon av gruppesamarbeid via Internett. Ved å definere oppgaver, gi tilbakemeldinger, og markere oppgaver som ferdig eller uferdig på Asana, har det vært relativt enkelt å holde oversikt over fremgangen til prosjektet, og få tilbakemeldinger på de oppgavene man har jobbet med.

Min arbeid i prosjektet har hovedsakelig bestått av å jobbe med de oppgavene jeg har blitt tildelt, laste opp utkast av lyder til Asana, og få tilbakemeldinger fra resten av gruppa. Denne prosessen har deretter blitt gjentatt for hver oppgave til den har blitt godkjent som ferdig. I tillegg til å jobbe på denne måten har Kybernesis også arrangert en del møter og 'workshops' hvor så mange av oss som mulig har møttes på kontorene til Kybernesis og jobbet sammen, noe som gjør prosessen med iterering over oppgaver og tilbakemelding raskere. I tillegg til kommunikasjon via Asana har vi også brukt chatte-programmene hipchat og skype for raskere kommunikasjon.

Prosjektet jeg har jobbet med hos Kybernesis har gjennomgått en del endringer i løpet av arbeidsprosessen, noe som er å forvente av denne typen prosjekt, og såpass tidlig i prosessen. Dette har derimot ført til at arbeidet med den grafiske biten av spillet ikke har kommet skikkelig i gang mens jeg

¹Asana <https://asana.com/> sist besøkt 08.05.15

har jobbet med lyd, og lyden er dermed basert mer på konseptskisser og tekstlige beskrivelser enn på spillet som det er realisert.

5.2 Arbeidsprosess *Shattered*

Samarbeidet med Gridmedia har også gått greit, men har muligens slitt med litt vanskeligere kommunikasjon enn hos Kybernesis. Arbeidsprosessen har bestått i at jeg har jobbet med musikk og lyddesign basert på beskrivelsen av spillet som laget av Gridmedia. Jeg har deretter sendt lyder og musikk til Gridmedia på mail, og fått tilbakemeldinger fra Gridmedia på mail. Denne prosessen har vært gjentatt for hver lyd og musikalsk tema til vi har kommet frem til et endelig resultat. Vi har også hatt en del møter for å planlegge spillet og min rolle i prosjektet. Etttersom jeg begynte på musikken før spillet var programmert har musikken hovedsakelig vært basert på tekstlige beskrivelser av temaene fra Gridmedia. Bakgrunnen for at spillet ble påbegynt relativt sent var fordi Gridmedia har fokusert sine ressurser på utviklingen av Geelix Game Engine og tilhørende program for produksjon av spillnivåer.

5.3 Arbeidsprosess rapport

I den første delen av arbeidet med oppgaven brukte jeg hovedsakelig tiden på det praktiske arbeidet, og lesing av teori som kunne være relevant til oppgaven. Mot midten av oppgaven fokuserte jeg på det praktiske arbeidet, samt noe skriving av den teoretiske delen av rapporten parallelt. Mot slutten av arbeidet har ferdigstilling av rapporten vært i fokus. Ikke all teorien har vært direkte relevant for den praktiske delen av oppgaven, men jeg har allikevel valgt å favne litt bredere i teoridelen siden dette kan være relevant i det videre arbeidet med disse prosjektet. Det vil også være relevant for fremtidige spillprosjekter jeg vil arbeide med, og gir meg en mer solid plattform for videre arbeid innen feltet. Selve arbeidsprosessen med rapporten har bestått i at jeg har laget utkast til deler av rapporten, som deretter er sendt til veileder for tilbakemelding. Denne prosessen har vært gjentatt etterhvert som rapporten har vokst. Tilbakemelding på rapporten er også gitt av veileder ved møter.

5.4 Refleksjon rundt arbeidsprosess

I løpet av prosessen med denne oppgaven har jeg gjort meg en del tanker om det å jobbe på to ulike praktiske prosjekter samtidig, i tillegg til dokumentering og rapportskrivning. Helt fra starten av oppgaven var det ganske klart at det kom til å bli en delt praktisk og teoretisk oppgave. Da jeg startet på oppgaven var omfanget av den praktiske delen noe uklart. Den originale planen var å enten finne en ekstern partner som skulle stå for design av spillet, eller designe mitt eget spillprosjekt. Fordelen med å lage et eget spillprosjekt, i motsetning til et eksternt samarbeid, er at man har større kreativ kontroll og oversikt over prosjektets helhet. Å designe et eget spillprosjekt vil derimot kreve en større andel av den tilgjengelige tiden til andre hensyn enn det lydlige, som jo er det hovedsakelige fokuset til denne oppgaven. Å designe et eget prosjekt gjør også at man har større kontroll over fremdriften i prosjektet, og er den eneste som er ansvarlig for gjennomføringen. Når man samarbeider med eksterne partnere kan man aldri være helt sikker på fremgangen i prosjektet, eller når det vil blir ferdigstilt.

Etterhvert som jeg fikk kontakt med de eksterne partnerene, og fikk avklart min rolle i prosjektene falt valget på å kun jobbe med de eksterne prosjektene istedenfor å designe et eget prosjekt. Jeg valgte også å samarbeide med to ulike prosjekter istedenfor å kun fokusere på ett av dem. Dette valget førte til en økt total arbeidsmengde, men også en viss sikkerhet med tanke på gjennomføringen av den praktiske delen av oppgaven. Jeg følte også at samarbeid med eksterne parter var en viktig erfaring med tanke på arbeid med senere prosjekter, og at to prosjekter gir oppgaven en større bredde både teoretisk og praktisk. De to prosjektene har ulikt antall personer med ulike personligheter, designes for ulike plattformer, og har tydelige forskjeller i stilart. Disse forskjellene gir meg viktige erfaringer med arbeid i ulike gruppestørrelser, personlighetstyper, plattformer, og stilarter som er nyttige for fremtidige prosjekter.

De to eksterne prosjektene har ved ferdigstillingen av denne oppgaven begge ganske mye gjenstående arbeid på selve spillene, så det lydlige arbeidet jeg har gjort har kun blitt delvis implementert, siden en del av de elementene lydene er knyttet til ikke eksisterer enda i spillene. Selv om jeg har jobbet med to prosjekter føler jeg selv at jeg har fått gjort litt lite på den praktiske delen av oppgaven, men siden prosjektene ikke har kommet lengre enn de har, hadde det ikke hatt så mye å si for implementeringen om jeg hadde fått gjort mer. Det kunne nok vært mer effektivt å kommet inn i spillprosjektene senere i prosessen, slik at en større del av spillet var ferdigstilt. Dette hadde

begrenset mengden forandringer i spilldesignet etter at man har begynt på lyden, og det hadde gjort det enklere å lage sammenheng mellom det visuelle og auditive. Det hadde også gjort iterering med testing av ulike lydlige idéer implementert i spillet enklere, og dermed få et mer helhetlig bilde av spilllets lyddesign. Det har også vært vanskelig å få testet de dynamiske elementene, når de delene av spillet som skal styre dynamikken ikke eksisterer enda.

På den andre siden har det å være med fra starten av prosjektene gitt meg muligheten til å kunne påvirke designet av spillene, og gitt muligheten til å tilpasse spillene til det ønskede lyddesignet og løsning for implementering. Det har også gjort det mulig å sette et større fokus på den lydlige delen av opplevelsen og skape større plass til lyddesignet. I *Karavan* har lyddesignet delvis vært begrenset av å måtte passe inn i en eksisterende spillverden, men det har også vært åpent for tolkning. Det har også i stor grad vært mulig å bestemme hvordan lyddesignet skal implementeres og brukes. I *Shattered* har designet vært mer fastsatt, og det har vært vanskelig å tilpasse designet med tanke på musikalsk sjanger og implementering.

En ekstra potensiell utfordring med å samarbeide med eksterne partnere er at disse i en del tilfeller kan ha retningslinjer som begrenser delingen av materialer knyttet til prosjektet før dette er klar for lansering. Dette er både med tanke på prosjektets publisitet, og for å beskytte misbruk av materialet med tanke på kopiering, etterligninger o.l. På grunn av dette, samt valg av plattform, har det ikke vært mulig å legge ved en spillbar demo av *Shattered* eller deler av spillmotoren eller strømmeløsningen som er relevant for implementeringen av lyden.

De to ulike prosjektene har begge hatt både noen fordeler og ulemper med tanke på arbeidsprosessen. I prosjektet i samarbeid med Kybernesis har det vært enkelt å kommunisere med de andre som har jobbet på prosjektet, og det har vært enkelt å kunne implementere og teste ulike idéer og løsninger i spillmotoren, siden det er valgt å bruke en spillmotor med god dokumentasjon og støtte for ulike plattformer. Den største ulempen har vært at spillet har forandret design i løpet av prosessen, og at det ikke har kommet langt nok i utviklingsprosessen til å kunne teste mitt arbeid i implementert i spillet.

Gjennomføringen av *Shattered* har også gått ganske tregt, og implementeringen av lyd har dermed ikke begynt før helt mot slutten av arbeidet med min oppgave. Spillet er også laget i en egenprodusert spillmotor, som foreløpig kun har støtte for én spillplattform. Dette har gjort det tilnærmet umulig for meg å implementere lyden, og jeg har dermed ikke kunne testet mitt arbeid i praksis. I tillegg til dette har samarbeidet vært preget av noe tung kommunikasjon, og mye av tiden har gått med til å diskutere formaliteter. En

ekstra ulempe har vært Gridmedias noe restriktive retningslinjer for deling av materialer knyttet til Geelix Game Engine og *Shattered*. Dette har gjort det vanskelig for meg å dele arbeidet jeg har gjort i kontekst av spillet siden det ikke finnes en løsning for å spille spillet med lyd på PC direkte. For å spille det på PC kreves det Windows 8 og Microsoft Visual Studio Express 2013 for Windows, eller Windows Phone Power Tools for WP8.1. Gridmedia har også uttrykt ønske om at kildekode til spillet eller klienten ikke legges ved denne oppgaven.

Kapittel 6

Avslutning

6.1 Erfaringer

Arbeidet knyttet til denne oppgaven har vært svært lærerikt for meg, både med tanke på teori og praktisk samarbeid. Det har vært ekstra spennende å kunne samarbeide med eksterne partnere, og bidra til, og få viktige erfaringer fra, faktiske spillprosjekter. Det har også vært en lærerik utfordring å jobbe i større grupper med tanke på kommunikasjon, siden prosjektene blir såpass store at ikke alle kan ha oversikt over alt som blir gjort. Med tanke på kommunikasjon har det sannsynligvis vært enklere siden det meste har foregått i Trondheim. Siden flesteparten som har jobbet på prosjektene har vært i samme by har det vært relativt enkelt å arrangere møter ved behov. Til tross for dette har store deler av kommunikasjonen på begge prosjektene foregått via Internett.

En viktig erfaring jeg tar med meg videre fra arbeidet med disse to praktiske prosjektene er at selv om det som komponist og lyddesigner kan være positivt å være med fra starten av et prosjekt, er det på bakgrunn av hvor lang tid et spillprosjekt ofte tar kanskje mer effektivt å komme inn i prosjektet på et senere tidspunkt, hvor mesteparten av spillets design er ferdigstilt, og man kan jobbe opp imot en prototype av spillet og får testet implementeringen av lydene. Det kan med andre ord være greit å være med fra starten av prosjektet, gitt at man har mulighet til å være med på hele prosessen. Om man har begrenset tid til å jobbe på et prosjekt, slik som er tilfellet for denne oppgaven, vil det kanskje være best å bruke denne tiden i slutfasen av prosjektet, i motsetning til i starten eller midten slik som har vært tilfelle for de to prosjektene jeg har jobbet på i den praktiske delen av oppgaven.

Bakgrunnen for å jobbe med to prosjekter i startfasen var at jeg ikke kunne finne noen andre prosjekter som var nærmere sluttfasen å samarbeide med. Hadde et prosjekt som var nærmere fullføring vært tilgjengelig ville dette trolig blitt valgt istedenfor ett eller begge av de prosjektene jeg har jobbet med. Et prosjekt nærmere sluttfasen ville ført til en større trygghet med tanke på gjennomføringen av den praktiske delen, og det hadde dermed kanskje ikke vært behov for å jobbe på to spillprosjekter. På den andre siden ville det å jobbe på kun ett prosjekt begrenset mulighetene for sammenligning, erfaringer, og bredden på relevant teori.

6.2 Videre arbeid *Karavan*

Spillet *Karavan* er fortsatt et stykke fra å være ferdigstilt, og trenger sannsynligvis både mer musikk og lyddesign før det blir ferdig. Det vil dermed være naturlig for meg å fortsette samarbeidet med Kybernesis for å fullføre arbeidet med *Karavan*. Det er også muligheter for at jeg fortsetter samarbeidet med Kybernesis på andre spill i *Corp Wars*-serien etter at dette prosjektet blir ferdig. Etterhvert som *Karavan*-prosjektet utvides, og nye elementer legges til, vil også behovet for mer lyd øke. Det kan godt tenkes at det vil bli behov for opptak av stemmeskuespill i løpet av prosjektet. Det vil også bli behov for en del lyddesign knyttet til brukergrensesnittet, siden den eneste ikke-diegetiske lydeffekten som foreløpig er laget er for bruk av penger i spillet. Kybernesis er har også kontakt med andre musikere fra blant annet et tidligere prosjekt, så det kan godt tenkes at arbeidet med musikk blir delt, eller at andre komponister får ansvar for musikk, og at jeg fortsetter arbeidet med lyddesign, slik at *Corp Wars* får en mer helhetlig lydlig stil.

6.3 Videre arbeid *Shattered*

I likhet med *Karavan*-spillet, gjenstår det en del arbeid før *Shattered* er klar for lansering. Det gjenstår foreløpig å ferdigstille musikk til noen av de allerede planlagte nivåene. Det er også planer om å utvide spillet med flere nivåer etter lanseringen av den første versjonen av spillet. I sammenheng med disse nye nivåene vil det være behov for ny musikk. Med tanke på å ha en sammenhengende musikalsk stil i spillet er det dermed naturlig for meg å fortsette samarbeidet med Gridmedia på *Shattered*. Med tanke på lyddesign gjenstår det en del arbeid, både med produksjon eller anskaffelse

av lyder, samt implementeringen av disse i spillet. Det er mulig at de resterende lydeffektene skaffes via lydbiblioteker eller lignende, og at jeg kun fortsetter med musikken til spillet, siden dette var den primære oppgaven min i prosjektet.

Tillegg A

A.1 Filer

Bachelor.zip
Karavan/Corporates v1.wav
Karavan/Corporates v2.wav
Karavan/Csound/bigExplosion.csd
Karavan/Csound/bigExplosion.wav
Karavan/Csound/smallExplosion.csd
Karavan/Csound/smallExplosion.wav
Karavan/Csound/bigTurret.csd
Karavan/Csound/bigTurret.wav
Karavan/Csound/smallTurret.csd
Karavan/Csound/smallTurret.wav
Karavan/Csound/drivingBig.csd
Karavan/Csound/drivingBig.wav
Karavan/Csound/drivingSmall.csd
Karavan/Csound/drivingSmall.wav
Karavan/Csound/engineBig.csd
Karavan/Csound/engineBig.wav
Karavan/Csound/engineSmall.csd
Karavan/Csound/engineSmall.wav
Karavan/Csound/smallExplosion.csd
Karavan/Csound/smallExplosion.wav
Karavan/Csound/smallTurret.csd
Karavan/Csound/smallTurret.wav
Karavan/Enklave v1.wav
Karavan/Enklave v2.wav
Karavan/FMOD Karavan

Karavan/Kash1.wav
Karavan/Kash2.wav
Karavan/MusicLogic.cs
Karavan/Nommo v1.wav
Karavan/Nommo v2.wav
Karavan/Wastelanders v1.wav
Karavan/Wastelanders v2.wav

Shattered/Beskrivelse.pdf
Shattered/Fjell.aiff
Shattered/FMOD Shattered
Shattered/Jungel.aiff
Shattered/Oppgavedefinisjon.pdf
Shattered/Rotasjon Stein.wav
Shattered/Rotasjon Tre.wav
Shattered/Rulling Stein.wav
Shattered/Rulling Tre.wav

A.2 Spill

Utvikler. (Årstall). Navn. Utgiver(e).

Advanced Microcomputer Systems. (1983). *Dragon's Lair*. Cinematronics.
Atari Inc.(1972). *Pong*. Atari Inc.
Invisible Handbar. (2008) *Audiosurf*. Steam.
Konami. (1981). *Frogger*. Konami/Sega-Gremlin.
LucasFilm Games. (1984). *Ballblazer*. Atari Inc.
Namco. (1980). *Rally-X*. Namco/Midway.
Nutting Associates. (1971). *Computer Space*. Nutting Associates.
Sega. (1980). *Carnival*. Sega/Gremlin.
Sun Electronics. (1980). *Stratavox*. Taito.
Stern Electronics. (1980). *Berzerk*. Stern Electronics.
Taito. (1975). *Western Gun*.Taito/Midway.
Taito. (1978). *Space Invaders*. Taito/Midway/Leisure & Allied Industries.

A.3 Konsoller

Utvikler. årstall. Navn

Atari Inc. (1977). Atari VCS/2600.

Magnavox. (1978). Magnavox Odyssey2.

Nintendo. (1990). Super Nintendo Entertainment System.

Bibliografi

- [1] Bateman, C. and Boon, R. (2006). *21st century game design*. Hingham, Mass.: Charles River Media.
- [2] Chion, M., Gorbman, C. and Murch, W. (1994). *Audio-vision*. New York: Columbia University Press.
- [3] Collins, K. (2008). *Game sound*. Cambridge, Mass.: MIT Press.
- [4] Csikszentmihalyi, M. (1990). *Flow*. New York: Harper & Row.
- [5] Haugland, M. (2012). *Interaktiv musikk i spill*. Trondheim: NTNU.
- [6] Mäkinen, K. (2012). *Lyddesign som illusjonistisk virkemiddel i interaktive medier*. Trondheim: NTNU.
- [7] Miller, G. (1956). *The Magical Number Seven, Plus or Minus Two*. The Psychological Review, 1956, vol. 63, pp. 81-97. <http://cogprints.org/730/1/miller.html> sist besøkt 08.05.15
- [8] Nybø, H. (2012). *Elektroakustisk filmlyd, Teori og praksis på grensen mellom filmlyd og elektroakustisk komposisjon*. Trondheim: NTNU.
- [9] Phillips, W. (2014). *A Composer's Guide to Game Music*. Cambridge, Mass: MIT Press.

Figurer

2.1	<i>Pong</i>	12
2.2	Atari 2600	13
2.3	<i>Berzerk</i>	14
2.4	<i>Frogger</i>	15
2.5	<i>Monkey Island 2: LeChuck's Revenge</i>	17
2.6	Relasjoner mellom spillertyper og musikktyper	35
2.7	Relasjoner mellom spillsjangre, spillertyper, og musikktyper	40
2.8	<i>Audiosurf</i>	43
3.1	Konseptskisse for <i>Karavan</i>	60
3.2	Konseptskisse for <i>Corp Wars</i>	61
3.3	Skjermdump fra <i>Karavan</i> -prototype	64
3.4	<i>Wastelander buggy</i>	66
3.5	<i>Wastelander bike</i>	67
3.6	<i>Wastelander truck</i>	68
3.7	<i>FMOD Studio</i>	70
3.8	Konseptskisse for <i>Shattered</i>	73
3.9	Fjellnivå i <i>Shattered</i> -prototype	74
3.10	Jungelnivå i <i>Shattered</i> -prototype	75
3.11	Rulling på treverk i <i>Shattered</i> -prototype	77
3.12	<i>Shattered meny</i>	79