



KUNGL
TEKNISKA
HÖGSKOLAN



TRITA-NA-E03159 • CID-256 • ISSN 1403-0721 • Department of Numerical Analysis and Computer Science

Manusverktyg för interaktiva produktioner

Jenny-Li Örsell

Master Thesis, CID, Stockholm



CID, CENTRE FOR USER ORIENTED IT DESIGN

Jenny-Li Örsell

Manusverktyg för interaktiva produktioner

Report number: TRITA-NA-E03159, CID-256

ISSN number: ISSN 1403 - 0721 (print) 1403 - 073 X (Web/PDF)

Publication date: Stockholm, 2003

Reports can be ordered from:

CID, Centre for User Oriented IT Design

NADA, Department of Numerical Analysis and Computer Science

KTH (Royal Institute of Technology)

SE- 100 44 Stockholm, Sweden

Telephone: + 46 (0)8 790 91 00

Fax: + 46 (0)8 790 90 99

E-mail: cid@nada.kth.se

URL: <http://cid.nada.kth.se>

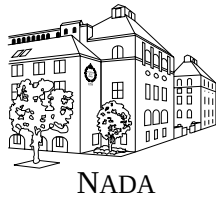


**KTH Numerisk analys
och datalogi**

Manusverktyg för interaktiva produktioner

Jenny-Li Örsell

TRITA-NA-E03159



Numerisk analys och datalogi
KTH
100 44 Stockholm

Department of Numerical Analysis
and Computer Science
Royal Institute of Technology
SE-100 44 Stockholm, Sweden

Manusverktyg för interaktiva produktioner

Jenny-Li Örsell

TRITA-NA-E03159

Examensarbete i människa-datorinteraktion om 20 poäng
vid Programmet för datateknik,
Kungliga Tekniska Högskolan år 2003
Handledare på Nada var Björn Thuresson
Examinator var Yngve Sundblad

MANUSVERKTYG FÖR INTERAKTIVA PRODUKTIONER

SAMMANFATTNING

Många manusförfattare inom interaktiva medier använder idag samma verktyg som vanliga författare när de skriver, dvs. papper och penna eller ordbehandlare. Dessa verktyg är enkla att använda men de är inte utformade för icke-linjära berättelser med alternativa handlingar eller för texter med många referenser till andra medieformer. Med traditionella verktyg blir interaktiva manus svåröverskådliga och detta kan i slutändan inverka negativt på de resulterande produktionerna.

Denna rapport innehåller en studie i interaktivt berättande och summerar fyra intervjuer med personer som skriver för interaktiva medier. Den beskriver även några existerande applikationer inom området. Förundersökningen visar att det behövs en annorlunda typ av manusverktyg och att inga sådana verktyg ännu finns. Vidare beskriver rapporten utvecklingen av en generell och objekt-orienterad modell för interaktiva berättelser inkluderande konceptet *situation*. Modellen omvandlas senare till en datorbaserad prototyp av ett manusverktyg, vilket inte bara visar att denna typ av verktyg är möjliga att skapa utan även hur de kan vara beskaffade.

SCRIPT TOOLS FOR INTERACTIVE PRODUCTIONS

ABSTRACT

Today, many scriptwriters for interactive media use the same tools for writing as common writers i.e. pen and paper or word processors. These tools are easy to use, but they are not designed for non-linear stories with alternate plots or for texts with lots of references to other types of media. With traditional tools interactive scripts become messy and difficult to survey, which in the end can affect the resulting productions in a negative way.

This thesis contains a study of interactive storytelling and summarizes four interviews with persons writing for interactive media. It also gives a tour of some existing applications in the area. The preliminary investigation shows that a different type of scripting tool is needed and that no such tool can be found today. Furthermore, the thesis describes the development of a general object-oriented model for interactive stories including the concept of *situation*. The model is then transformed into a computer-based prototype of a script tool, which not only shows that this kind of tool is possible to create but also how it could be designed.

FÖRORD

Det är en vacker sommardag; solen skiner från en strålande klarblå himmel och gräset är lockande grönt. Alla som kan är ute och njuter av det härliga vädret. Alla? Nej, inte riktigt... Bakom ett par neddragna persienner i ett litet dammigt rum kan man hitta en ung kvinna djupt involverad i ett nyligen inköpt dataspel. Entusiastiskt har hon precis börjat bygga sitt eget äventyr och idéerna sprudlar kring vad som ska hända. Men på något sätt rinner äventyrets berättelserådar hela tiden iväg åt olika håll, slår knut på sig själva och krockar med varandra och det är ett fasligt sjå med att hålla rätt på allting. Med en djup suck ger kvinnan upp sitt äventyr för dagen och går ut.

Den unga kvinnan är naturligtvis jag och berättelsen slutar inte riktigt här. Efter någon veckas funderande och testande med vanliga ordbehandlingsprogram får jag klart för mig att det behövs någon form av verktyg för att hålla rätt på alla trådar i min interaktiva berättelse. Ännu en vecka senare hittar jag en gammal anteckning från en föreläsning med Magnus Seter, manusförfattare för bl.a. dataspel, där han uttrycker exakt samma tanke. Plötsligt vet jag vad mitt examensarbete ska handla om. Jag ska bygga det – verktyget som håller ordning på interaktiva manus.

Denna rapport beskriver mitt examensarbete i ämnet människa-dator-interaktion vid institutionen för numerisk analys och datalogi (Nada) på Kungliga Tekniska Högskolan. Arbetet har genomförts mellan november 2002 och juni 2003 och motsvarar 20 akademiska poäng. Uppdragsgivare är Interaktiva Institutet (II).

Jag skulle speciellt vilja tacka mina två handledare Björn Thuresson (Nada) och Peter Becker (II) för många och långa diskussioner som fört arbetet åt rätt håll; ibland framåt och ibland bakåt. Jag skulle också vilja tacka Magnus Seter, manusförfattare vid Houdini Creations AB, för att så frikostigt ha upplåtit sin tid åt mina frågor och funderingar. Även mina två andra intervjupersoner Mathias Gullbrandson och Rex Brådhe har försett mig med insikter som jag inte velat vara utan. Sist men inte minst vill jag tacka Nicklas Kittelmann för hans oändliga tålamod och hans generositet med sina tekniska kunskaper.

INNEHÅLLSFÖRTECKNING

INLEDNING	1
Bakgrund	1
Syfte med arbetet.....	1
PROBLEMDEFINITION	3
Frågeställningar.....	3
Fokus och valda begränsningar.....	3
FÖRSTUDIE	6
Litteratur om manusförfattarens uppgifter.....	6
Litteratur om datorn som problemens lösning.....	7
Litteratur om interaktivt manusskrivande	8
Intervjuer	9
Intervju med Magnus Seter	9
Intervju med Rex Brådhe	12
Intervju med Mathias Gullbrandson	13
Om rollspel.....	14
Om lajv.....	15
Om multimedial och interaktiv teater	16
Om ett äventyrsspel.....	18
Om andra medieformer	19
Ergodisk litteratur	19
Textbaserade dataspel	20
Interaktiv film	20
Programvara	21
HyperStudio.....	21
Storyspace.....	21
Dramatica Pro	21
Adventure Game Studio	22
Text Adventure Development System	22
Erasmatron	22
ARBETSFÖRLOPP	24
Val av arbetsmetod och angreppssätt	24
Analys av interaktiva berättelsers uppbyggnad.....	25
Berättelsers interaktiva struktur	25
En objektorienterad begreppsvärld	27
Val av terminologi	29
Prototyp av manusverktyg.....	30
Val av teknik	31
Från terminologi till datastruktur	32
Från datastruktur till funktionsanalys.....	33
Från funktionsanalys till gränssnitt	36
Iterationer i arbetet.....	39
Ny intervju med Magnus Seter.....	40
Svåra beslut	42
Ändringar	43

Refuserade förslag.....	44
Medvetna begränsningar i prototypen	45
SLUTSATSER OCH DISKUSSION	48
Behövs ett manusverktyg?	48
Litteraturen.....	48
Intervjupersonerna.....	48
Rajkas rapport.....	48
Finns det manusverktyg?	49
Är det möjligt att skapa ett manusverktyg?	49
Hur kan ett manusverktyg se ut?	49
Ett datorbaserat program.....	49
Lämplig modell	50
Lämplig datastruktur och teknik.....	51
Lämplig funktionalitet	51
Lämpligt gränssnitt.....	51
Motsvarar konceptet behoven?.....	51
Olika manustyper.....	52
Olika tekniska system.....	53
Användarorientering	53
Framtid.....	54
KÄLLFÖRTECKNING	55
BILAGA 1: FRÅGOR INFÖR INTERVJU MED SETER	58
BILAGA 2: RAJKAS MODELLFÖRSLAG	59
BILAGA 3: OM ROLLSPEL	60
BILAGA 4: MAZERMINDS DATABASSTRUKTUR	62
BILAGA 5: UTDRAK UR ETT MANUS	64
BILAGA 6: FLER INTERVJUFRÅGOR	66

INLEDNING

För att få en bättre förståelse av examensarbetet, vad som låg till grund för det och vad det gick ut på börjar rapporten med att beskriva arbetets bakgrund och vilka intentioner jag hade när det påbörjades.

Bakgrund

Under 70-talet föddes många av våra moderna, interaktiva berättandeformer. År 1974 gavs det första kommersiella rollspelet, *Dungeons & Dragons* av Gygax, Arneson och Blume ut [Darlington 98]. Under 1976 skapades det första dator-baserade äventyrsspelet *Adventure* av Woods och Crowther och det första *Multi-User Dungeon* skapades av Trubshaw och Bartle 1979-80 [Aarseth 97].

Idag, 30 år senare, har manusskrivande för interaktiva produktioner gått från undergroundrörelse och experimentverkstad till att bli ett yrke. Dess likheter med det traditionella författandet av film- och teatermanus är relativt stora och många som verkar inom det ”nya” området kommer ursprungligen från näraliggande och äldre mediebranscher. Det verkar därför inte helt otroligt att en hel del rutiner och arbetssätt vandrat med personer som gått från den ena branschen till den andra – med varierande resultat.

Manusskrivande för interaktiva produktioner skiljer sig nämligen avsevärt från författande av vanliga manus på vissa punkter. I en interaktiv produktion kan publiken (i denna rapport även benämnd som användaren eller spelaren beroende på sammanhang) ofta välja och påverka händelseförloppet. Så länge publikens möjliga val är få och påverkan på berättelsen begränsad går det ofta bra att skriva manuset linjärt. Att läsa ett sådant manus blir ungefär som att hoppa fram och tillbaka i en uppslagsbok. Men ju fler alternativa berättelsespår i produktionen som finns att ta hänsyn till och ju större och mer komplicerad påverkan publiken har på berättelsen desto svårare blir det att skriva ett linjärt manus. Ofta slutar det i en svårläst härva av referenser och villkor.

Som det ser ut nu använder sig författare till interaktiva produktioner av samma hjälpmedel som vanliga manusförfattare, dvs. papper och penna eller ordbehandlingsprogram. Dessa verktyg är enkla att använda, men då de inte är utformade för att ta hänsyn till exempelvis stickspår och alternativa slut blir resultatet ofta en oöverskådlig papperslunta.

Manus skrivna med klumpiga verktyg riskerar att bli just klumpiga. Mångfalden missgynnas, kvalitén blir lägre och även kommunikationen med dem som ska förverkliga manuset blir lidande. Det är märkligt att den yrkesgrupp som förser en stor del av underhållningsindustrin med själ och tanke saknar ett relevant verktyg.

Syfte med arbetet

Jag hade redan från början en teori om att det behövs bättre verktyg för att modellera icke-linjära, interaktiva berättelser. Den teorin ville jag få bekräftad genom att undersöka både vad som redan finns i form av skrivstöd och hur en tänkt målgrupps behov och arbetsmetoder kan se ut.

Men jag ville gå längre. Att projicera en flerdimensionell berättelse på ett två-dimensionellt papper har uppenbarligen sina nackdelar. Det bästa sättet att beskriva flerdimensionella (eller icke-linjära) berättelser är att presentera dem i

ett flerdimensionellt (eller icke-linjärt) medium. En dator hanterar lätt linjär text men även icke-linjära referenser i form av exempelvis hyperlänkar och villkor i form av programkod. Datorn borde vara idealisk för detta ändamål. Jag ville därför också undersöka om det går att skapa programvara för att underlätta arbetet för interaktiva manusförfattare och ta reda på hur denna programvara skulle kunna se ut.

PROBLEMDEFINITION

En god ansats till att utreda något är att ställa sig inte bara en, utan massor av frågor och sedan sovra och sälla bland dessa. Det är också lämpligt att besluta vad som inte ska ingå i utredningen. Följande två stycken beskriver de frågor och begränsningar som gällde för examensarbetet.

Frågeställningar

Genom att formulera ett antal delfrågor och i klassisk söndra-och-härska-teknik besvara dem en efter en hoppades jag kunna uppnå syftet med mitt arbete. Frågorna sammanfattade inte bara målet i ett antal punkter utan gav även något konkret att utgå ifrån i det fortsatta arbetet. Listan nedan beskriver packningen i min mentala frågeryggsäck vid arbetets början:

- Finns det något uttalat behov av ett manusverktyg?
- Finns det kanske redan något manusverktyg för interaktiva produktioner? Om inte, vad använder man sig istället av? Finns det några näraliggande verktyg för andra sorters manus som kan vara intressanta att titta på?
- Om det finns ett uttalat behov men inga redan existerande verktyg anpassade för ändamålet, är det då så att det helt enkelt inte går att bygga manusverktyg? Är uppgiften alldeles för komplex eller löser man saker på så olika sätt att ett gemensamt verktyg för olika medieformer inte är möjligt att konstruera?
- Vad är en interaktiv berättelse? Vad är inte en interaktiv berättelse? Vilka olika berättandeformer ska examensarbetet behandla? Går det att extrahera några allmänna beståndsdelar ur interaktiva berättelser?
- Hur ser manusförfattarens arbetssituation ut? Vad har han eller hon för behov och krav på sina arbetsredskap idag? Vilka önskemål har manusförfattaren om ett framtida verktyg?
- Vilka fördelar och nackdelar finns det med att skapa ett verktyg i datormiljö?
- Vilken funktionalitet bör finnas med, hur ska gränssnittet se ut och finns det andra krav på system eller liknande som bör tillgodoses?
- Vilka begränsningar, förutsättningar och problem kommer att uppstå då arbetet går framåt och vilken riktning kommer processen att ta av de val som då måste göras?

Fokus och valda begränsningar

För att inte uppgiften redan från början skulle svälla ut över alla bredder var ett tydligt uttalat fokus och ett par inledande begränsningar lämpliga. Efterhand som arbetet fortskred och nya fakta kom i dagen skulle fler begränsningar komma att komplettera de första.

Målet var att stödja skrivandeprocessen för de författare som skriver manus för interaktiva, berättande produktioner.

Att "stödja skrivandeprocessen" kan innebära många saker. I detta fall låg fokus på att underlätta organisationen och överblickandet av manus och inte på att stödja kreativiteten eller att skapa ett bättre ordbehandlingsprogram. Att

stödja kreativiteten (exempelvis med en karaktärsgenerator) och därigenom försöka ersätta fantasin hos en mänsklig hjärna vore inte bara ett vanskligt projekt utan även onödigt då den kreativa biten är just det som är mest tilltalande i yrken som manusförfattare. Att skapa ett bättre ordbehandlingsprogram än exempelvis Word inom ramarna för ett examensarbete skulle därtill vara en prestation som vida överskred de flestas förmåga.

En annan viktig fråga var vilken del av skrivandefasen som skulle stödjas. I de flesta multimedieprojekt ligger manusdesignen mellan konceptutvecklingsfasen och produktionen. Generellt sett kan man säga att manusdesignfasen inbegriper att skriva ett antal utkast som i allt högre detaljnivå specificerar produktionen [Stansberry 97].

Först skrivs en synopsis (på engelska *content treatment*; ett begrepp mer känt från filmindustrin) som är en tidig skiss av innehåll, tema och interaktionsmöjligheter. Här beskrivs karaktärerna, stämningen, huvudstoryn och dess lösning [Stansberry 97]. På detta stadium i skrivprocessen verkar det inte vara något problem att skriva linjärt. Det är också här frågor som *Vilka motiv har karaktärerna?*, *Hur löser man uppgiften?* och *Vilken är den bakomliggande berättelsen?* besvaras. Detta steg är till stor del en kreativ fas och verktyg som stöder denna aspekt av manusförfattande finns redan på marknaden.

Ett annat viktigt dokument som skrivs behandlar den interaktiva strukturen. Det skapas ibland av en informations- eller interaktionsdesigner, men inte sällan är detta samma person som manusförfattaren. Den interaktiva strukturen kan sägas vara en disposition för hur man tar sig igenom produktionens innehåll och den illustreras gärna med t.ex. flödesscheman. Designen av den interaktiva strukturen är kritisk på så sätt att den påverkar både kontinuiteten och logiken i produktionen och den narrativa presentationen [Stansberry 97].

När man slutligen lägger ihop berättelsen med den interaktiva strukturen, med dialoger och med anvisningar för ljud och bild etc. får man ett produktionsmanus. Till detta kan även tillkomma tekniska specifikationer, storyboards med skisser och andra dokument [Stansberry 97]. Jag valde att inrikta arbetet så att man kan skapa produktionsmanus där narrativa element, interaktiv struktur och anvisningar för olika medieformer samexisterar.

Den huvudsakliga målgruppen var manusförfattare. Men ett manus fungerar ofta som den specifikation efter vilken ett helt team av programmerare, grafiker, ljudläggare etc. arbetar och är dessutom en presentation av den kommande produktionen inför både projektledare och kunder. Därför var det lätt att halka in på andras behov och områden som marknadsföring och samarbetsdokument. Jag valde in i det längsta att undvika detta, men i vissa fall sammanföll naturligtvis de övriga intressenternas behov med manusförfattarens.

Att förstå manusets roll i interaktiva och berättande produktioner var en central punkt i examensarbetet som vållade en del huvudbry. En del interaktiva produktioner som jag från början räknade in i området för examensarbetet hade snarare en uppsättning regler än ett manus i vanlig mening. Även litteraturen pekade på att det för produktioner med lite eller inget *content development* (exempelvis arkadspel) sällan skrivs ett traditionellt manus [Stansberry 97]. Ytterligare några produktioner som valdes bort var sådana som inte var primärt berättande utan kanske mest interaktiva, t.ex. hemsidor.

Slutligen fanns det produktioner där berättelsens manus skrevs samtidigt som den producerades; jämför textbaserade äventyr som "är" sitt manus. Även ett antal medieformer med mindre uttalad tradition av interaktivitet gick bort och jag valde att koncentrera mig på nyare berättandeformer (s.k. *new media* [Manovich 01]).

FÖRSTUDIE

Bland de frågor som ställdes i arbetets inledande fas var det de som behandlade om och vad som redan var gjort inom området som först måste besvaras. Frågan om manusförfattarens arbete och situation var också viktig, men även en viss orientering i de skilda världar som ämnet skulle beröra var nödvändig. Det första steget i strategin blev därför att undersöka vilka svar och nya uppslag litteraturen, den tänkta målgruppen och den för tillfället existerande programvaran kunde ge mig. Kanske skulle jag någonstans finna stöd för mina teorier?

Litteratur om manusförfattarens uppgifter

Manusförfattare arbetar ofta i ett av två lägen: berättande eller förklarande. Antingen berättar han/hon sin historia eller förklarar hur något – säg interaktionen med en karaktär eller övergången från en plats till en annan – fungerar. Författaren ska inte bara skapa ett intressant innehåll utan sköter också uppdelningen av innehållet och navigationen mellan produktionens olika delar [Stansberry 97]. En första slutsats blir då att ett manusverktyg både ska låta författaren skriva i det och samtidigt på ett tydligt sätt ska visa hur det skrivna är uppbyggt och hur det hänger ihop. Det räcker inte med att skriva – författaren ska även kunna kontrollera flödet i berättelsen.

Ett känt problem i interaktiva sammanhang är att författarens kontroll över berättelsen är omvänt proportionell mot antalet vägar genom den [Bolter 91]. Ju fler valmöjligheter en användare eller spelare har desto fler fall måste författaren täcka och desto större är risken för att motsägelser och ologiska hopp ska uppstå. Samtidigt kan ett stort antal valmöjligheter ge fler infallsvinklar i produktionen, mindre förutsägbarhet, bättre anpassning till olika individers handlingsmönster och en känsla av frihet för användaren. Det ligger nära till hands att tro att mängden beslut och vägar i en interaktiv produktion har en positiv inverkan på dess narrativa rikedom. Även om detta inte alltid stämmer (interaktion för interaktionens skull ger sällan något mervärde till en berättelse) är naturligtvis interaktionen en mycket viktig beståndsdel i manusförfattarens arbete. Att hjälpa författaren att skapa och hålla rätt på flera lösa berättelseträdar torde vara en viktig uppgift för ett manusverktyg.

Det kan vara mycket svårt att avgöra hur bra ett manus är bara genom att läsa det. Manusets innehåll ligger utstrött över flera sidor och bildar en väv av korsreferenser där innebörden i texten beror på logik och rumsliga förhållanden som inte alltid är uppenbara. Dessutom skymmer ett otal instruktioner och koder sikten för berättelsen. En nödvändig strategi för att få en bättre bild av manuset är att bygga en prototyp av det [Stansberry 97]. Prototypen kan vara väldigt enkel, t.ex. i form av en s.k. *text engine* där den skrivna texten ackompanjeras av något slags navigationsmöjlighet. Genom att testa sitt manus får författaren möjlighet att uppdatera och revidera det innan det går till produktion. En prototyp höjer kvalitén på produktionen och hjälper manusförfattaren i hans/hennes arbete. Den underlättar även förståelsen för andra inblandade i arbetet och kan användas i mötet med målgruppen – innan den dyra produktionen har kommit igång. Slutsatsen denna gång är att ett bra manusverktyg borde hjälpa författaren att skapa en prototyp.

Litteratur om datorn som problemens lösning

Geometrin hos en interaktiv berättelse blir lätt komplex. Ju fler avsnitt som skrivs och ju fler kopplingar som görs mellan avsnitten desto svårare är det för författaren att få en överblick över allting. Man kan likna författaren vid en matematiker som försöker förstå ett tredimensionellt objekt genom att studera ett antal tvådimensionella projektioner av det. I en interaktiv berättelse blir varje specifik väg genom den en projektion av dess geometri [Bolter 91]. Först när man studerat alla vägar (projektioner) och deras relationer till varandra har man en chans att få en bild av hela berättelsen. Problemet idag är att många interaktiva berättelser har långt fler dimensioner än tre: *I det här rummet kan spelaren välja dörren till höger eller dörren rakt fram eller ta hissen en våning upp eller ta tidsmaskinen till 1931 eller avsluta spelet*. Papper och ordbehandlare stöder främst tvådimensionella projektioner: *Fortsätt läsa här eller gå till någon av rubrikerna Vänster, Hissen och Tidsmaskinen*. Interaktiva manus blir då fådimensionella renderingar av flerdimensionella objekt [Stansberry 97]. I datorstödda manusverktyg kan man däremot skapa projektioner i olika antal dimensioner. Genom att välja att projicera en del, men kanske inte alla, aspekter av en interaktiv berättelse i en vy kan författaren snabbt få klart för sig inte bara att saker hänger ihop utan även hur de hänger ihop.

Vår litterära teknik har historiskt sett förespråkat att vi ska skriva och läsa linjärt. Den tryckta litteraturen har ibland tvingats arbeta mot sitt medium för att kunna vara ämnes- och relationsorienterad [Bolter 91]. Även ordbehandlingsprogram plattar ut texten och förutom tillgång till hierarkiska rubriker ger de skribenten dåligt stöd för att greppa textens struktur. En intressant lösning som framförts är tanken om en s.k. *outline processor*, i vilken textstycken kan ersättas med symboler [Bolter 91]. I vårt multimediala samhälle kan detta naturligtvis utökas till att gälla även andra medieformer som t.ex. ljud och animationer. Ett bra exempel på hur man lyckats föra över en tryckt publikation till ett mindre linjärt media, dela in det ämnesmässigt med lättillgängliga relationer och samtidigt krydda det med nya medieformer markerade med symboler är *Nationalencyklopedin*, som numera finns i Internet-version [NE 03]. Kanske är det så här vi ska skriva multilinjärt?

Under studierna av kurslitteraturen snubblade jag över en spännande definition av skrivande [Bolter 91]:

To write is to do things with topics – to add, delete, and arrange them. The computer /.../ giving visual expression to our acts.

Att skriva skulle således inte bara handla om att fylla meningar med ord, utan även om att sortera och ordna texten och att behandla den som grupper av objekt. Vidare stod det att:

[A] hypertext consists of topics and their connections, where again the topics may be paragraphs, sentences /.../ words or /.../ graphics. A hypertext is like a printed book that the author has attacked with a pair of scissors.

Drastiskt? Ja. Men att med teknikens hjälp klippa upp texten i ämnesorienterade stycken och sedan koppla samman dem på ett meningsfullt sätt kunde vara lösningen på problemet med det flödesorienterade, linjära skrivandet. Datorn framstod mer och mer som den miljö i vilken ett manusverktyg bäst skulle frodas. Att skriva linjärt, hierarkiskt, i lager [Bolter 91] eller

ämnesorienterat – i datorn skulle alla skrivsätt kunna vara lika naturliga och att läsa det skulle vara lika bekvämt.

Litteratur om interaktivt manusskrivande

En vanlig fråga som dyker upp i manusdesignfasen är vilket format ett manus ska ha. Formatet beror förstås mycket på materialet självt [Stansberry 97] och vad för slags manus det är som ska skrivas. I ett rollspel är manuset nästan alltid platsbaserat och i en klassisk radioteater består manuset kanske mest av dialog. Vissa manus har inte en lika tydlig tradition i fråga om format och stil; dataspelsmanus varierar relativt mycket och lånar gärna sitt format av närbesläktade medier. Förutom det dialog- och platsbaserade skrivsättet finns en mängd mer eller mindre stilrena varianter. Ett vanligt format är det som inspirerats av filmens värld och som grundar sig på vad publiken ser och hör. Den stora variationen av stilar skulle kunna lägga krokben för ett manusverktyg genom sitt höga krav på flexibilitet. Samtidigt är en bortrationalisering av olika format otänkbar och värmandet av mångfalden borde vara en självklarhet. Men är det möjligt?

Något som ofta nämns som en utmaning är att skriva interaktiv dialog [Stansberry 97]. Att ge publiken möjlighet att välja mellan ett antal repliker när de talar är en vanlig företeelse i framför allt dataspel. Men hur skriver man en dialog med flera olika svarsalternativ linjärt, när svaren i sin tur pekar åt olika håll? Svaret är att det inte går att skriva interaktiv dialog linjärt. Istället kan man ta till knep som att rita ett flödesschema över dialogen med hänvisningar till en linjär text med repliker. Med färger, former, pilar och siffror i flödesschemat kan man markera vem som säger vad, vilken reaktion en replik får och vilka villkor som gäller för ett svarsalternativ. Det är dock lätt att se att en lång och interaktivt komplex dialog med flera inblandade karaktärer och konsekvenser skulle bli väldigt rörig att rita upp. Antingen får man ta ett större papper eller hitta andra sätt att angripa dialogen. Men hur skulle det gå till? Man skulle kunna se den interaktiva dialogen som ett specialfall av en interaktiv berättelse. Både berättelsen och dialogen handlar om att ge publiken möjlighet att påverka förloppet genom olika val. Om man lyckades hitta ett sätt att skriva interaktiva berättelser icke-linjärt kanske man samtidigt skulle finna ett sätt att skriva interaktiv dialog.

Den enklaste formen av interaktion är direkt och statisk: Varje gång användaren trycker på knapp A kommer händelse A utspelas och om användaren trycker på knapp B utspelas händelse B. Utgången är strikt specificerad och beror på användarens direkta handlingar. Denna design föredras av utvecklare av användarvänliga verktyg eftersom det ger en högre förutsägbarhet och användaren en känsla av kontroll. I spel och mer underhållningsorienterade system ses däremot oväntade händelser ofta som en utmaning [Schneiderman 98] och som en positiv del av upplevelsen. Där vill man gärna göra interaktionen mer komplex och avhängig på andra förhållanden än de momentana. Spelarens tidigare beteende kan då påverka att ett tryck på knapp B i vissa fall istället utmynnar i händelse C. Denna form av interaktiv struktur kallas villkorade satser (*conditional linking*) och ger författaren möjlighet att öka komplexiteten hos en berättelse [Stansberry 97]. Ett exempel på hur denna teknik kan användas hittar man i den danska spelproduktionen Blackout [Deadline 97]: *Vid ett tillfälle hamnar spelaren på en bakgata där ett gäng ligister hotar. Spelaren har fyra handlingsalternativ att välja på, men*

bara ett av dem leder till något positivt. Det "rätta" alternativet är det som är mest konsekvent med spelarens tidigare beteendemönster: En tuff karaktär bör välja det aggressiva handlingsalternativet för att undgå misshandel medan en poetisk karaktär klarar sig undan med hjälp av sin talang som talare.

Datorns förmåga att hålla rätt på flera variabler samtidigt och värdera dem lämpar sig bra i dessa sammanhang. Då villkorade satser bidrar till berättandets dynamik borde ett manusverktyg åtminstone ge författaren möjlighet att sätta villkor för olika händelser, kanske även kunna värdera variabler.

Intervjuer

För att få en bättre orientering i världen av interaktivt berättande planerade jag in tre intervjuer i förundersökningen. Syftet med intervjuerna var även att få träffa målgruppen i verkligheten och ta del av dess värdefulla förstahandsinformation om manusförfattarens arbetssituation och behov. De tre intervjuerna gjordes med personer med stor erfarenhet inom sina respektive medieområden och med ett uttalat intresse för manusskrivandets process. Rent generellt fick intervjuerna snarare formen av diskussioner där vi bl.a. avhandlade interaktivitet, multimedialitet och manusskrivandets natur.

Intervju med Magnus Seter

Den första intervjun gjordes med Magnus Seter, manusförfattare och spel-designer på Houdini Digital Creations. Han föreläser om interaktivt berättande och speldesign och har tidigare även skrivit manus för rollspel. Seter visade sig mycket intresserad av mina planer på ett manusverktyg för interaktiva berättelser och resultatet av intervjun kom att påverka det fortsatta arbetet i hög grad. I Bilaga 1 återfinns en lista på de frågor jag förberedde inför intervjun.

Manusförfattaren

Seter hade svårt att hitta några specifika egenskaper som skulle kunna karaktärisera honom och hans författarkollegor i dataspelsbranschen. Vissa författare kunde arbeta som typiska skribenter medan andra mer liknade regissörer i sitt arbete. En del tänkte i bilder medan andra tänkte i strukturer när de skrev. Många var kvinnor och något färre var män. Även de tekniska kunskaperna hos manusförfattare varierade. Seter ansåg att en majoritet av manusförfattarna skriver digitalt. Därmed fick jag ett lämpligt mått på vilken datakunskap ett tänkt manusverktyg kunde kräva av sina användare. Ett så intuitivt gränssnitt som möjligt och mycket vägledning vid svårare moment skulle vara nödvändigt i verktyget.

Två av manusförfattarens uppgifter och problem

En av de viktigaste uppgifterna för manusförfattare idag enligt Seter är att portionera ut berättelsens innehåll i lagom stora avsnitt. I ett icke-linjärt manus som är skrivet linjärt är det svårt att få överblick och bedöma hur välportionerad texten är. Det gäller att undvika att innehållet i kapitel ett räcker för spelande i tio timmar medan innehållet i kapitel två endast räcker för en halvtimmes spelande.

Ytterligare ett problem för författare är den stora mängd versioner av manuset som skapas i iterationerna då det skickas mellan författaren och kunden. I början tar tiden mellan att kunden faktagranskat och kommit med önskemål

om ändringar till att manuset är rättat av författaren kanske en hel arbetsdag. I slutet av manusfasen kan en ny iteration påbörjas var 30:e minut. Ett manusverktyg skulle behöva funktionalitet för robust versionshantering och möjlighet att spåra gjorda ändringar. Eftersom rättningarna till slut blir så små tyckte Seter att det skulle vara praktiskt om manuset var delbart. Då kunde man skicka endast en liten del med rättningen istället för hela manusstrukturen varje gång kunden skulle uppdateras.

Manusets delar

Seter sa sig arbeta mycket med byggstenar som kapitel, avsnitt och scener och sorterade gärna dessa i trädstrukturer. Även begrepp som modul, uppdrag och uppgift kunde förekomma för att beteckna de hierarkiska delarna i ett manus. Att ha en överenskommen mall för manusets struktur och formatering har betydelse för hur tydligt manuset är för övriga inblandade inom ett projekt.

För att hålla rätt på de olika ingredienserna i ett manus kan scener numreras efter vilket kapitel och avsnitt de hör till. Bilder, ljud och andra objekt namnges med olika koder som identifierar dem, deras typ och kanske deras position i manuset. Ett objekt kan exempelvis heta B52:7_2, vilket betyder att det är en bild (B) med identitetsnummer 52 i kapitel 7, scen 2. I de flesta större projekt finns en utsedd person som har ansvar för att namngivningen och dess standard följs på ett korrekt sätt [Stansberry 97]. I mindre projekt är denna person – inte helt överraskande – ofta manusförfattaren själv. Det är tydligt att ett viktigt moment för författaren är att hålla rätt på saker och ting; vad de är, vart de hör och när de kan dyka upp.

Actionspel kontra äventyrsspel

På frågan om det fanns någon skillnad mellan t.ex. actionspel och äventyrsspel svarade Seter att det troligtvis skrivs manus för båda sorterna men i olika hög grad. Actionspel kan vara mer teknikbaserade och då blir det ofta programmerare som tar över manusskrivarrollen. Äventyrsspel är manusintensiva och berättelsestyrda och kräver därför en utsedd manusförfattare. Skillnaden ligger kanske i mängden slumphändelser och mängden av händelser som ”måste” ske (för berättelsens logik). Slumphändelser och automatiska mekanismer är vanliga i actionspel och skapas främst genom programmering, medan manusförfattaren styr och beskriver sådana händelser som har stor betydelse för berättelsen.

Roller och personer i ett projekt

Författarens roll tenderar att överlappa andra roller i multimedieprojekt. Eftersom författaren är den som har bäst insikt i produktionen och på sätt och vis är beskyddaren av berättelsens själ är en roll som projektledarassistent inte helt ovanlig. Att författaren ibland även fungerar som interaktionsdesigner har nämnts tidigare och som den med mest ledighet i slutet av projekt kan uppgifter som testare och liknande också förekomma. Frågan uppstår om det är möjligt och önskvärt att ge stöd för författaren även i dessa roller?

Att skriva manus är ett kollaborativt arbete. Det är många som ska titta på manuset ur sitt specifika perspektiv och ge förslag på ändringar. Olika personer vill ha ut olika saker ur manuset; programmerarna vill veta tekniska detaljer, kunden vill granska fakta och grafikerna vill ha ut bildanvisningar. Under intervjun kom vi fram till att en lämplig funktion skulle vara att kunna filtrera ut dessa fakta ur manuset till respektive person. För att stödja möjligheten för flera att titta på manuset från sitt perspektiv diskuterade vi

ytligt ett antal tekniska lösningar, bl.a. en manusdatabas med flera klienter och ett därtill hörande webbgränssnitt.

Projektgång och designdokument

I ett projekt arbetar författaren mycket i början och lite i fortsättningen. Ett projekt kunde enligt Seter se ut enligt följande:

Projektspecifikation⇒ Idéarbete⇒ Manus skrivs⇒ Utvärdering⇒ Produktion

De designdokument man arbetar med i början av projektet varierar naturligtvis, men Seter föreslog ändå följande lista av dokument (sorterad i tidsordning):

- Idébeskrivning eller projektspecifikation är det (korta) dokument som skrivs först. Exempel: *Kunden vill ha ett spel om Drottningholms slott för barn, och huvudkaraktärerna ska vara två spöken.*
- Synopsis. Ett dokument i storleken ett till tre A4 med yttlig beskrivning av bl.a. spelets stämning, karaktärer och huvudstory.
- Enklare struktur. Ett dokument som ger en grov överblick över hur spelet är uppbyggt. Exempel: *Slottet ska ha fyra vingar och ett torn i vilket vardera av de fem kapitlen utspelar sig.*
- Storyboard/skisser. Någon har sagt att illustratören är manusförfattarens bästa vän och det gäller definitivt i denna fas av arbetet. En storyboard består av små skisser över hur slutresultatet kan se ut och visualiserar manusförfattarens idéer.
- Teknisk specifikation. Skrivs samtidigt som storyboard och har betydelse för vilken teknik man kan använda sig av i spelet. Exempel: *Spelet ska innehålla tecknade animationer men inga videoklipp.*
- Ett s.k. *overall structure*. En disposition eller ”skelettet” för manuset.
- Manuset. Innehåller det mesta, bland mycket annat anvisningar för ljud och bild, filhänvisningar, beskrivningar av interaktionsmöjligheter och repliker för röstskådespelare.

Dagens verktyg – nackdelar och fördelar

Idag skriver man oftast manus i ordbehandlingsprogram som Word på både Mac och PC, men även annan programvara som kalkylprogrammet Excel används. En stor nackdel Seter ser med Word är att det inte klarar av långa manus p.g.a. de många fältkoder man arbetar med. Formatmallarna används bl.a. till att markera vilken sorts text det är fråga om (t.ex. instruktioner eller ordagrann lydelse) och förutom att Word är resurskrävande och beräknings- tungt tappar programmet lätt formateringen. Det är heller inte lämpat för att skriva interaktiv dialog i, något som Seter speciellt efterlyser. Ordbehandlings- programmen stödjer flödesorienterat skrivande medan det saknas verktyg för att skriva manus objektorienterat. I ett objektorienterat manusverktyg baserat på strukturer kan karaktärer, föremål och platser manipuleras och ges egenskaper var för sig. Seter ville inte ha en ordbehandlare utan snarare ett prototyping-verktyg i vilket man kunde bygga en klickbar prototyp av texten. De fördelar han såg med Word var främst funktioner som rättstavning och att kunna söka och ersätta text.

Önskvärd funktionalitet

Vi talade om att det vore praktiskt med bibliotek för olika objekt i manuset, t.ex. ett karaktärsbibliotek. På skärmen skulle man kunna ”dra” en karaktär från biblioteket till de scener karaktären skulle ingå i. Vid större projekt med flera manusförfattare skulle ett gemensamt bibliotek ha fördelen att ge författarna samma objekt att arbeta med och eliminera risken att t.ex. en karaktär beskrivs som sur av en författare och glad av en annan. Överhuvudtaget borde manusverktyg stödja skrivandeprocesser där flera författare är inblandade i samma projekt.

Under intervjun framkom en rad andra, blandade förslag på funktionalitet i ett framtida manusverktyg:

- Sökfunktion. Vill författaren plocka bort ett föremål i en scen ska han/hon också kunna ta bort alla andra förekomster av föremålet i andra scener på ett enkelt sätt.
- Möjlighet till hierarkiskt strukturerande av berättelsen. Jag var först tveksam till Seters entusiasm över trädstrukturer eftersom det liknade linjärt skrivande allt för mycket (jfr rubriker och underrubriker). Interaktiva berättelser är snarare uppbyggda som icke-linjära nätverk och det som saknas nu är ett sätt att skriva nätverksorienterat. Kravet på flexibilitet i skrivandet och en idé om att det kanske skulle gå att förena de båda skrivsätten fick mig att ändra inställning.
- En anpassningsbar struktur för projekt. Seter hade en idé om att man kanske redan i början av ett projekt ska välja vilken struktur det ska ha från ett antal olika modeller. Men att tvinga författaren att välja mellan ett antal statiska, färdigstöpta former har sina nackdelar. En förvald struktur med inbyggd flexibilitet vore kanske ett bättre alternativ.
- En funktion för att hålla reda på vad som finns kvar att göra/skriva. Ett typiskt exempel är då författaren måste hålla rätt på löst hängande berättelsetrådar som ska sammanfogas med övriga berättelsen. Två lösningsförslag som diskuterades under intervjun var en ”Att göra”-lista och möjligheten att kvittera det som är färdigt med t.ex. checkrutor.

Hur ett manus ser ut

I slutet av intervjun lånade Seter mig en del av ett manus han skrivit [Seter 02:b] och ett utdrag ur detta manus finns i Bilaga 5.

Intervju med Rex Brådhe

Den andra intervjun i förundersökningen gjordes med Rex Brådhe. Brådhe har en bakgrund som skådespelare och producent för SVT Utbildning och SVT Interaktiva Medier. Jag fick höra att han tidigare lekt med tanken att skapa ett skrivstöd för interaktiva medier och nyfiken på hans tankar kring ett sådant projekt stämde vi möte.

Manus, deras innehåll och språk

Under intervjun talade vi om generella skillnader i manusskrivande för teater, film och interaktiva medier. För teater skrivs nästan bara dialog och ibland enstaka scenanvisningar. Inom film skrivs förutom detta relativt detaljerade bildanvisningar som specificerar vad som ska synas på filmduken. För inter-

aktiva produktioner i sin tur skriver manusförfattaren i stort sett "allt": bild-anvisningar, ljudanvisningar, dialog, valalternativ m.m.

Teatermanus har en traditionell utformning som de flesta inblandade kan läsa, men manus för interaktiva produktioner är mer komplexa och därmed mer svårlästa. Brådhe hade upplevt att manus inte lästes p.g.a. att de var svårbegripliga, torra och fulla av "onödiga" fakta. Han tog upp vikten av att kunna levandegöra berättelsen och att hitta ett gemensamt manusspråk som alla medverkande kan förstå. Att införa en standard för hur manus ska se ut och sprida den skulle dock vara en knepig uppgift. Mitt lösningsförslag var istället att man skulle låta de inblandade extrahera just den information som intresserade dem ur manuset. All "onödig" och svårbegriplig information skulle filtreras bort och författaren skulle inte behöva skriva mer än ett manus.

Förslag på funktionalitet

Brådhe förespråkade att ett manusverktyg skulle arbeta med byggstenarna ord, bild och ljud, dvs. baseras på vad som sägs, syns och hörs i produktionen. Det han sa påminde väldigt mycket om hur man arbetar med och ser på film, och jag var inte helt säker på att det skulle fungera för alla sorters manus. Brådhe menade vidare att berättelsen styrs av handlingar och att handlingar i sin tur skapar händelser. Han tyckte sig även se att det fanns en vattendelare i hur man ser på interaktiva produktioner. Vissa människor ser det som en slags film som rullar på medan andra ser det som handlingar som kan göras efter varandra. Det första synsättet är passivt, medan det andra är aktivt.

Brådhe hade själv funderat på att skapa ett manusverktyg men stött på problem och var inte längre säker på om projektet var möjligt att genomföra. Han hade ändå ett antal konkreta förslag på funktionalitet som han gärna delade med sig av:

- Författaren ska kunna beskriva situationer i verktyget.
- Författaren ska kunna skriva dialog.
- Det ska vara möjligt att komplettera texten med andra medier, t.ex. en bild som förklarar vilken slags skog författaren är ute efter eller ett ljud som karaktäriserar en önskvärd stämning i miljön.
- Författaren ska kunna hålla rätt på vart de olika berättelseträdarna löper.
- Verktyget ska stödja direktmanipulation.
- Det ska vara möjligt att zooma ut och in i sitt perspektiv, från helhetssyn till detaljnivå.

Vi diskuterade den revolution för skrivandet som skedde då ordbehandlaren kom och konkurrerade ut skrivmaskinen. Från att ha varit tvungen att skriva varje sida uppfifrån och ner kunde skribenten plötsligt börja skriva mitt i sin berättelse, gå tillbaka, ändra och hoppa fram igen. Det var en stor frihet för alla som skrev när ordbehandlingsprogrammen kom och ett steg mot icke-linjärt skrivande. Analogin till mitt manusverktyg var uppenbar; det fria skrivandet måste stödjas.

Intervju med Mathias Gullbrandson

Mathias Gullbrandson hittade jag genom att han stått som konstnärlig ledare och projektledare för lajvet *Virtual Galileo* [Virtual Galileo 02] som ägde rum

någon månad före examensarbetets början. Han hade även varit inblandad i en rad andra lajvprojekt, bl.a. *Futuredrome* [Futuredrome 03] i Kinnekulle med ca 2000 deltagare. Med Mathias talade jag om att skriva manus för lajv, hur interaktiviteten tar sig i uttryck i dessa sammanhang samt om hur man utvecklar karaktärer för ett lajv. En del av det material som framkom under intervjun samt en kort introduktion till ämnet återfinns under rubriken *Om lajv* längre fram i denna text.

Om rollspel

Rollspel är en unik spelform som uppfanns i början av 70-talet av Gary Gygax och hans vänner i USA. Det sägs ha sina rötter i militära strategispel där tennfigurer simulerar krigsmanövrer och händelseförlopp, men skillnaderna mellan de båda spelformerna är relativt stora. Det första kommersiella rollspelet, *Dungeons & Dragons*, gavs ut 1974 och finns än idag på marknaden i reviderad form [Darlington 98]. Det har en fiktiv medeltidsvärld som tema men det finns även rollspel som utspelar sig i framtiden, i vampyrsmåhällen, i rymden, i vilda västern-miljöer osv.

Frågan vad ett rollspel är och hur det fungerar skulle kräva en lång förklaring, men en bra introduktion för novisen återfinns i Bilaga 3. Rent generellt spelas de flesta rollspel i grupper om 4-10 personer där minst en person utses till spelledare. Spelledaren är den som håller rätt på och skapar den värld spelet utspelar sig i. Övriga spelare skapar varsin karaktär med vissa egenskaper och färdigheter för den värld spelledaren håller i. Allt sker med hjälp av muntliga framställningar, papper, penna, olika sorters tärningar och gedigna regelböcker. Att spela rollspel innebär inte att delta i fysiska tävlingar utan handlar snarare om sociala och interaktiva berättar-event där alla hjälps åt att skapa en spännande historia.

Till varje rollspel (i meningen regelverk och världsbild) finns en mängd äventyr att köpa. Ett äventyr, som ofta består av ett texthäfte och kanske någon tillhörande karta, är som ett manus för spelledaren. Det beskriver miljön i den värld spelarna befinner sig i, intrigen, stämningen, vad och vilka man kan träffa på i miljön och mycket annat. Spelledaren kan i många fall välja att läsa upp beskrivningar direkt ur häftet eller själv formulera egna, mer levande beskrivningar. En del saker håller spelledaren hemliga för spelarna tills deras karaktärer upptäcker dem. Eftersom spelarna ofta har en oändlig mängd handlingsalternativ att välja på hänger spelets kvalitet lika mycket på spelledarens improvisations- och berättarförmåga som på manusets utformning.

Rollspelarnas karaktärer rör sig fysiskt i den fiktiva världen. Det har till följd att äventyren skrivs platsorienterat. Huvudrubrikerna i texthäftet motsvarar de intressanta platser karaktärerna kan besöka i spelet. Under varje rubrik finns förutom platsbeskrivningen också beskrivningar av de föremål och karaktärer som återfinns på platsen. Även intressanta handlingsalternativ som för äventyret framåt/bakåt beskrivs, liksom villkor och konsekvenser för dessa handlingar. Exempel: *För att spelarna ska klara av den onde magikern måste de använda trollstaven som ligger i urnan på kyrkogården. Träder de in i hans laboratorium utan staven i sin ägo kommer de förvandlas till stoft med 50 % säkerhet.*

Ett av problemen med platsorienterade äventyr är att föremål kan flyttas och karaktärer rör på sig. Då måste antingen den som skriver äventyret beskriva objektet varje gång/på varje ställe det är aktuellt eller så får spelledaren bläddra

fram och tillbaka i häftet för att hitta beskrivningen han/hon söker. Om en scen påverkar en annan måste det anges i båda scenerna för att äventyret ska bli så läsbart som möjligt [Seter 02:a]:

En spelledare måste kunna läsa texten från början till slut och hänga med i händelsernas flöde /.../. För att göra det ännu svårare måste läsaren när som helst kunna hoppa till vilken del av texten som helst för att hitta viktig information. Och när han spelar äventyret med sina spelare är händelsernas flöde troligtvis inte samma som det flöde som presenteras i texten. [Seter 02:a]

Rollspelande är i högsta grad berättande på interaktiv basis och brottas därför med flera av de problem som beskrivits tidigare i denna rapport. Samtidigt har många rollspel oerhört utvecklade regelsystem med mängder av variabler som ska läggas ihop, multipliceras eller slumpas fram i olika situationer. Vill man skapa ett eget äventyr väntar en mängd tärningsslag och räknande. För att hjälpa spelledare och dem som gör egna äventyr finns det idag en hel del datorstödda verktyg, t.ex. RolePlayingMaster [RPM 03] och Campaign Suite [Twin Rose 03]. Dessa verktyg har främst funktioner för att användaren enkelt ska kunna justera siffror och värden och generera regelriktiga karaktärer och föremål. Det viktigaste bidraget är dock att man här kan närma sig sitt äventyr på ett objektorienterat sätt istället för flödesorienterat som tidigare: I ett fönster kan man t.ex. se alla föremål och deras egenskaper, i ett annat kan man rita kartor och i ett tredje kan man generera monster. Genom att sortera gränssnittet efter aktiviteter och objekt lyckas rollspelsverktygen skapa ordning i kaoset.

Om lajv

Lajv (från engelskans *live*) är kanske närmast släkt med interaktivt drama, men det förklaras ofta som ”levande rollspel”. Rent generellt kan man säga att om ett vanligt rollspel spelas av 4 personer utrustade med papper och penna kring ett bord så kan ett lajv ha 400 deltagare utrustade med rekvisita som ringbrynjor och kopparkittlar och utspela sig i en riktig skog. Av praktiska skäl har lajv inte lika många detaljerade regler och man använder heller inte tärningar för att slå fram resultat av t.ex. en strid. Betoningen ligger snarare på att framställa (tids-)typiska miljöer och med fysiska medel gynna det sociala samspelet mellan deltagarna.

Lajv har liksom rollspel ett skrivet manus, men spelledaren är här ersatt av ett antal skådespelare som genom interaktion med övriga deltagare styr handlingen i önskad riktning. Följande uppställning av lajvmanusets delar grundar sig på vad som framkom under intervjun med Mathias Gullbrandson:

- Bakgrundshistoria. ”Detta har hänt”.
- Beskrivning av spelarenan, dess stämning och instruktioner för scenografin.
- En dramaturgi eller kronologi där större händelser beskrivs och hur olika drivkrafter påverkar dem.
- Regler för lajvets deltagare.

- Rollbeskrivningar för ensemblen (skådespelarna). Eftersom huvudrollerna har störst betydelse och är mest utpräglade spelas de i huvudsak av skådespelare som driver handlingen framåt.
- Rollbeskrivningar för vanliga deltagare. I beskrivningen ingår förutom grupptillhörighet och egenskaper även rollkaraktärens motiv, rädslor och drivkrafter. Beroende på hur kritisk rollen är för dramaturgin specificeras den olika noga och ibland måste t.o.m. karaktärens handlingsval i olika situationer beskrivas.
- Beskrivningar av grupper av karaktärer, t.ex. medlemmarna i släkten Montague (från *Romeo och Julia*). Karaktärerna får automatiskt vissa egenskaper genom sitt medlemskap i olika grupper. Vill en deltagare spela en mindre avgörande karaktär i lajvets berättelse och mer fritt utforma sin roll kan han/hon välja en mindre, gruppbounden roll. Denna roll baseras då på en grupptillhörighet snarare än på en egen, mer specificerad karaktärsbeskrivning.

Precis som i rollspel kan man se att lajvmanus har en del publika och en del interna delar: *Alla deltagare vet att släkten Montague och Capulet ligger i fejd med varandra (gruppbeskrivning), däremot är det ingen förutom skådespelarna som från början vet om och i så fall vilken av släkterna som kommer att segra (dramaturgi).* Dessutom finns det en mängd rollbeskrivningar som är privata: *Om du träffar en man vid namn Romeo kommer du att förälska dig i honom oavsett hans rang eller börd. Håll detta hemligt.*

En intressant poäng med lajvmanus är att de till stor del är karaktärsorienterade. Intrigen ligger i huvudsak placerad i rollbeskrivningarna och det är karaktärernas drivkrafter som för handlingen framåt. På samma sätt har även grupperingar av karaktärerna betydelse för berättelsens dynamik. Att de sammanflätade händelserna i ett lajv sker konstant och parallellt med varandra kan göra det svårt att skriva ett manus. Risken för att olika händelser krockar, inte inträffar alls eller kommer ur fas med varandra är påtaglig när lajvet ska utövas i praktiken. Dramaturgin i ett lajvmanus beskriver därför i första hand övergripande, episka händelser som är (relativt) förutbestämda. Den interaktiva biten återfinns främst på det episodiska mikroplan där lajvets deltagare improviserar sina roller mot varandra.

När det skrivs om lajv har det hittills oftast handlat om Tolkieninspirerade helger i skogsmiljö eller kostymdraman/mordintriger på en herrgård. Deltagarna och rekvisitan har befunnit sig rent fysiskt på spelplatsen. I takt med att ny teknik och nya lajvteman introducerats har denna tradition brutits. Hösten 2002 genomfördes *Virtual Galileo* [Virtual Galileo 02], det första i raden av svenska lajv med delvis virtuell scenografi. Här kunde deltagare hemifrån datorn styra en avatar att samspela med de riktiga människorna på spelplatsen. Blandningen av olika medieformer hos *Virtual Galileo* är dock inte unik och vad detta multimediala projekt utmynnar i får framtiden utvisa. Möjligheten att planera och skriva instruktioner för fler medieformer kommer troligtvis att efterfrågas och det i sin tur påverkar manusets utseende.

Om multimedial och interaktiv teater

Jag fick i min hand en rapport av förundersökningen till projektet *Digitala Teaterrum* skrivet av Peter Rajka [Rajka 96]. Undersökningen syftade till att

skapa underlag för utvecklingen av digitala planeringsverktyg inom i första hand den teatrala världen. I rapporten ställdes frågor som *Hur ska de teatrala representationstyperna och deras strukturer modelleras på ett för datorbaserad representation tillämpbart sätt?* Här eftersträvades en generell modell för planering av händelser (t.ex. scenbyten och repliker) oavsett media eller datatyp och ett gränssnitt där den detaljerade och multimediala kompositionen kunde presenteras. Trots idogt sökande hade Rajka bara funnit ett fåtal projekt och verktyg som någorlunda rörde datorbaserad planering för det skapande arbetet inom teater.

Jag såg genast paralleller till mitt arbete – att försöka finna en mer generell modell för planeringen (dvs. manusskrivandet) av händelser, oavsett mediatyp, i interaktiva berättelser. När vi talar om multimedia förknippar vi det främst med det senaste årtiondets tekniska framfart. Då är det lätt att glömma att teater har varit multimedialt mycket längre än så med klassiska musik-, ljus- och ljudeffekter. Virtuellt och digital scenografi är måhända spännande, men det är fortfarande bara ett komplement till de redan existerande medieformerna. Inte heller är lineariteten i teatermanus så självklar:

Dramatexten är visserligen också en linjär text i rent fysisk mening, men dess strukturer visar hierarkiska och kausala samband, så som indelningen i akter, scener, situationer och repliker, jämte samband mellan olika handlingar. Kronologin är inte alltid så spikrak. [Rajka 96]

Resultatet av projektet Digitala Teaterrum skulle komma att bli en datorbaserad tredimensionell verkstad för att experimentera med scener och aktörer i olika uppsättningar [Becker, Hoogland 97]. I rapporten lade Rajka märke till att då man förlägger teatern till ett digitalt rum och gör den interaktiv börjar gränserna mot dataspel, *Virtual Reality* och *Augmented Reality* flyta. Huruvida det var interaktiviteten eller multimedialiteten som fick dessa gränser att flyta framgick inte av rapporten. Däremot stärktes känslan av att anledningen till att dessa uttrycksformer (liksom lajv) liknar varandra mer och mer är att de alla strävar efter att efterlikna och modellera verkligheten. Men är det meningen att vi ska skriva manus för verkligheten? Kanske är den fråga som ställs i pjäsen *Det allra viktigaste* [Unga Klara 03] inte helt obefogad: *Teater är livet, men är inte livet en teater?* Slutsatsen av resonemanget var att jag inte borde söka en specifik, fördefinierad metafor för interaktivt manusskrivande. Om olika medieformer börjar likna varandra och verkligheten allt mer är det inte längre aktuellt att tala om partitur eller scen när man kan vara mer generell. Det borde gå att skapa en vokabulär inspirerad inte bara av teatern, musiken eller rollspelandet utan av livet självt.

I slutet av undersökningsrapporten ger Rajka ett förslag på en modell för hur man ska kunna klassificera och strukturera olika typer av händelse-representationer. Rajka påpekar dock noga att det inte är fråga om en exakt specifikation av datastrukturer utan endast en principiell utgångspunkt för vidare arbete och diskussion. Modellen bygger på fyra olika klasser av händelserepresentation och i Bilaga 2 återger jag ett urval med de delar i modellen som på olika sätt återkommit och inspirerat mitt arbete.

Förutom modellskissen gör Rajka två intressanta observationer i sin rapport. Dels att en datorbaserad representation har fördelar som snabb förflyttning mellan olika nivåer/representationsformer genom t.ex. länkning och att presentationen av dessa nivåer kan göras samtidigt och synkront. Dels att ett

datorbaserat (teater-)verktyg är lämpligt för planering och test, inte för att producera syntetiska föreställningar. Båda dessa observationer torde gälla för detta examensarbete.

Om ett äventyrsspel

Det finns naturligtvis hur många dataspel som helst att titta på för den som vill studera interaktivt berättande. I synnerhet den genre som benämns äventyrsspel är känd för sitt stora innehåll av berättande, men det finns även en del att lära av spel i andra genrer. Min orientering i dataspelsvärlden med dess stilarter och mekanismer var relativt god inför examensarbetet. Därför valde jag att endast titta på ett spel i förundersökning och på så sätt få mer tid över för de områden jag visste mindre om. Även det faktum att många spel bygger på liknande berättande- och interaktionstekniker och att den allmänna kännedomen om dataspel är så pass utbredd stödde mig i detta beslut.

Valet föll till slut på *Neverwinter Nights* [BioWare 03], ett spel med karaktären av både äventyrsspel och rollspel. Med dess representativa drag fungerade det utmärkt som exempel för att belysa de vanligast förekommande funktionerna och begreppen i sin genre. Samtidigt gav *Neverwinter Nights* en unik inblick i sin egen uppbyggnad genom det tillhörande program, Aurora, i vilket den hugade spelaren kan konstruera egna avancerade banor och äventyr. Med Seter hade jag talat om hur författare skriver manus för dataspel och hur andra sedan producerar det. I Aurora fick jag möjlighet att se bakom kulisserna när berättaren själv producerar och manipulerar sitt äventyrsspel. Det slog mig att gränsen mellan att skriva och bygga ett äventyr inte är så stor.

Nedan listas ytterligare ett antal observationer jag gjorde i studiet av *Neverwinter Nights*. Det är ett axplock med de egenskaper hos spelet som jag ansett haft högst relevans för examensarbetet. Bland dessa finns både egenskaper som är vanliga i de flesta dataspel och egenskaper som är unika för just det valda exemplet:

- Spelet är areabaserat. Mellan större areor finns ingen visuell övergång (skärmen blir svart när ett nytt område laddas) och spelarens karaktär kan alltid agera i en area oberoende av vad som händer i närbelägna areor. Inuti en area kan det också finnas områden som är separerade ifrån varandra (t.ex. med en ravin) men där spelaren ändå kan se hur ytorna förhåller sig till varandra. Areabegreppet är på så sätt rätt splittrat. Dessutom går det att markera en del av en area och skapa en utlösare av den så att t.ex. en karaktär som går på mattan i ett rum utlöser en fälla. Frågan som uppstår är hur man ska se på areor rent tekniskt. Kan en area innehålla andra areor eller gränsar den istället till andra areor?
- I spelet görs en skillnad på föremål som kan plockas upp (t.ex. ett svärd) och föremål spelaren kan interagera med men inte flytta (t.ex. en byrå). Vissa objekt "är" miljön (en äng eller ett hus), vissa objekt är en del av en interaktiv miljö (ett träd eller en brunn) och vissa objekt i sin tur är föremål i miljön (äpplen). Det är intressant att man här valt att dela in kategorin föremål i tre nya kategorier.
- Då och då inträffar olika slumphändelser, t.ex. att det börjar regna eller att spelarens karaktär blir rånad när han/hon befinner sig i ett slumkvarter. Det går att programmera hur hög chansen/risken är för olika slumphändelser.

- Både karaktärer och föremål har ett stort antal (olika) egenskaper och justerbara värden för dessa. Många, men inte alla, av dessa värden beror på vilken grupp karaktären eller föremålet tillhör. Ju mer erfaren en karaktär blir desto fler egenskaper kan han/hon förvärva.
- Tekniken bakom dialoger kan användas för att interagera med saker. Ett exempel är att spelarens karaktär kan "tala" till en dörr och som svar få en lista på möjliga handlingsalternativ såsom att knacka på den, att öppna den, att dyrka upp den eller att slå in den. Man kan med andra ord tala till föremål och inte bara till karaktärer.
- Beroende på egenskaper hos spelarens karaktär kan vissa dialogalternativ vara strukna. Ännu vanligare är att dialogalternativ dyker upp först då spelarens karaktär utfört vissa uppdrag. Det finns med andra ord villkor (och konsekvenser) knutna till repliker. Detta gör att skrivandet blir ekonomiskt och författaren kan återanvända dialoger. Exempel: Istället för att skriva två mycket likartade dialoger för fallet att en karaktär är man respektive fallet där karaktären är en kvinna kan författaren välja att skriva en enda dialog där de replikalternativ som skiljer de båda fallen är villkorade efter karaktärens kön.
- Det finns två sorters karaktärer, dels spelets programmerade karaktärer och dels karaktärer som styrs av riktiga spelare. När en spelare talar med spelets karaktärer får han/hon välja en av flera färdigskrivna repliker. När spelaren talar med karaktärer som spelas av andra spelare används tangentbordet för inmatning av fritext. Det finns därmed två sätt för en spelare att kommunicera.

Om andra medieformer

Listan på olika medieformer där interaktivt berättande utgör en viktig del skulle kunna göras lång. I förundersökningen har jag bl.a. tittat på dataspel, rollspel, lajv och interaktiv teater. I detta stycke skulle jag vilja nämna några av de områden jag inte hann studera specifikt för examensarbetet:

Ergodisk litteratur

Det finns en hel del litteratur som kommit att kallas ergodisk litteratur eller cybertext. (Se [Aarseth 97] för diskussion kring begreppen.) Sådan litteratur är ofta, men inte alltid, uppdelad i mindre sekvenser och innebörden och upplevelsen av texten beror mycket på hur den läses. Det kan handla om texter där endast ett urval av de ingående sekvenserna läses, texter där sekvensernas ordning kan kastas runt eller texter utan specifik början och slut.

Exakt vilken litteratur som kan anses vara ergodisk är svårt att avgöra. Idag skulle vi kanske främst förknippa det med hypertexter på Internet där textfragment länkas till varandra. Men även om persondatorerna och deras genombrott radikalt ändrat förutsättningarna för att skriva och läsa interaktivt har ergodisk litteratur förekommit mycket längre än så i vanliga tryckta media. En av de äldsta texter som brukar nämnas i dessa sammanhang är *I Ching*, en kinesisk text för spådomar skriven runt 1000-talet f. Kr. [Aarseth 97]. Ett mer nutida exempel är soloäventyren/rollspelsböckerna om *Ensam Vargen* från mitten av 80-talet. Där uppmanades läsaren t.ex. att [Dever, Chalk 85]:

*Om du vill prata med gästgivaren vid disken, se [stycke] 342.
Om du vill försöka vinna ett par Gulddaler på armbrytning se
[stycke] 276.*

Textbaserade dataspel

Textbaserade äventyr hörde till de första dataspelen och såldes i stora mängder under 80-talet. I slutet av 80-talet började dataspel med avancerad grafik, animationer och ljud konkurrera med dessa och några år senare ansågs text-äventyrens tid vara slut rent kommersiellt. Trots att de textbaserade spelen försvann från den kommersiella arenan för så länge sedan finns det fortfarande en livskraftig hobbyverksamhet med människor som både spelar och skriver nya äventyr [Roberts 03].

De textbaserade spelen är en egen kategori av interaktiv fiktion som liknar både ergodisk litteratur och grafiska dataspel. I ett textbaserat äventyr presenteras spelaren för textuella beskrivningar av alla objekt och händelser i berättelsen. Interaktionen sker genom att spelaren matar in korta ord från en fördefinierad vokabulär som sedan hanteras av en speciell kommandotolk. Ett typiskt exempel på hur interaktionen kan gå till ser ut så här (kommandon och variabler skrivna med kapitäl):

[Dator]> *Du befinner dig i en mörklagd källarlokal. Det luktar mögel och väggarna är fläckiga av någon slags vitaktig beläggning. I ett hörn står en trasig pappkartong.*

[Spelare]> UNDERSÖK PAPPKARTONG

[Dator]> *När du närmar dig pappkartongen hör du ett ilsket fräsande och en svart rätta slinker mellan dina fötter. I kartongen ligger en gulnad tidning.*

[Spelare]> TA TIDNING

[Dator]> *Du tar upp den gulnade tidningen. Den är så skör att bitar av den faller i golvet.*

De flesta textbaserade spel är skrivna för ett av tre spridda system med varsin kommandotolk: TADS, Inform och Hugo [Britton 03]. Trots att spelen är textbaserade är det inte helt ovanligt med äventyr där texten kompletterats med bilder och ljud. Dessa äventyr ska inte förväxlas med grafiska dataspel utan kan snarare jämföras med böcker med tillhörande illustrationer.

Även om jag inte studerade textbaserade dataspel inför examensarbetet hade jag en viss tidigare erfarenhet av dem och verktyg för att skapa dem. Bl.a. vill jag nämna Per Wilhelmssons spel *The Longest Week* [Wilhelmsson 02] som med sina inslag av tidsresor visade mig hur komplicerad struktur en interaktiv berättelse kan ha i textbaserade spel.

Interaktiv film

Ungefär ett år före examensarbetet läste jag en kortkurs om interaktiv film med Chris Hales på Dramatiska Institutet. Min erfarenhet från denna kurs pekar på att interaktiv film i mångt och mycket liknar multimedial och interaktiv teater. Ur ett manusperspektiv krävs för båda fallen att författaren vill ha möjlighet att ge instruktioner för olika medieelement, kunna skriva dialog, beskriva interaktionsmöjligheter och ange villkor och konsekvenser av dessa.

Programvara

Ett av syftena med förundersökningen var att ta reda på om det redan fanns programvara för författare som skriver interaktiva manus. Då var det inte verktyg i form av vanliga ordbehandlingsprogram eller applikationer för att rita upp flödesscheman jag i första hand sökte efter. Istället ville jag undersöka de program som hade mest likheter med planering och skrivande av interaktiva berättelser.

Många av de applikationer jag fann liknade varandra rätt mycket och därför listas endast ett urval med de mest intressanta fynden nedan. Bland de funna programmen fanns en stor mängd verktyg för linjärt skrivande manus-författare inom i huvudsak film- och teaterbranschen. Dessa program representeras i urvalet av Dramatica Pro, medan övriga applikationer snarare är inriktade mot skapandet av färdiga (interaktiva) produktioner.

Resultatet av mitt omfattande letande måste därmed sägas vara negativt. Inget verktyg för att skriva manus för interaktiva, icke-linjära och multimediala berättelser kunde hittas – denna gång. Det är naturligtvis fortfarande möjligt att ett sådant verktyg existerar.

HyperStudio

HyperStudio är ett program där användaren kan skapa presentationer som kombinerar text, grafik, ljud och hyperlänkar. HyperStudio är främst utvecklat för utbildningssammanhang och ska enligt beskrivning göra elever till "multimedia authors" [Hyperstudio 01]. Trots denna framställning handlar det snarare om att användare får skapa färdiga presentationer i stil med vad man kan åstadkomma med PowerPoint än att skriva berättelser. Som verktyg skulle HyperStudio kunna fungera för planering och skissande av en produktion, däremot är det inte lämpligt att skriva komplexa manus i.

Storyspace

Storyspace är ett verktyg för bl.a. författare som skriver interaktiv fiktion i form av hypertext. Det är en miljö som lämpar sig för att skriva och länka komplexa texter och som ger möjlighet till att visuellt överblicka hur dessa är ordnade. Storyspace fokuserar på skrivandeprocessen och organisationen av information och är därför lämpligt för flera allmänna användningsområden t.ex. storyboarding, forskning, författande etc. [Eastgate 02]. För manus för interaktiva berättelser skulle det kunna fungera som den miljö i vilken en text-baserad prototyp av berättelsen görs. Storyspace saknar dock en del specifika stöd som manusförfattare för interaktiva produktioner behöver, t.ex. att hantera, söka, beskriva och namnge objekt och referenser till andra media.

Dramatica Pro

Dramatica Pro [Screenplay 03] är ett program i första hand tänkt för manus-författare inom film och teater men som även fungerar för vanliga roman-författare. Det tar användaren från idé till färdigt manus med betoning på att underlätta det kreativa arbetet med att utveckla karaktärer, teman, händelser och att hitta en väl fungerande struktur för dramaturgin. Dramatica Pro är intressant eftersom det har en så tydlig inriktning på att utveckla innehållet i en berättelse. Däremot passar det inte att skriva interaktiva berättelser eller produktionsfärdiga manus i.

Adventure Game Studio

Adventure Game Studio (AGS) låter användaren skapa sina egna äventyr liknande det tidiga 90-talets spel från Sierra och LucasArts. Programvaran är kostnadsfri och består av en utvecklingsmiljö i vilken det går att rita upp grafik och lägga till interaktionsmöjligheter och en motor med vilken äventyren körs. AGS sköter hanteringen av funktioner som att spara/ladda spel och låter användaren göra en exekverbar fil av sitt äventyr [Jones 03]. För att skapa enklare spel är AGS idealiskt men det är inte praktiskt att bygga prototyper i och fungerar inte alls för den som vill skriva ett manus.

Text Adventure Development System

Text Adventure Development System (TADS) är ett programmeringssystem speciellt designat för den som vill skriva äventyrsspel eller interaktiv fiktion [Roberts 03]. Systemet består av ett programmeringsspråk, en kompilator, en tolk, en avlusare (eng. *debugger*) och ett bibliotek med en mängd fördefinierade objekt som förekommer i äventyrsspel. TADS kan användas både för renodlade textbaserade äventyr och för spel som blandar text, grafik och ljud. Systemets multimedia-del använder HTML (Hypertext Mark-up Language) för formatering av text vilket ger författare möjlighet att kombinera och kontrollera text, bilder, audio, ramar och länkar. Biblioteket har en detaljerad världsmodell med objekt som:

- Rum.
- Aktörer.
- Föremål som kan bäras.
- Föremål som är fixa i miljön.
- Dörrar.
- Behållare, ytor och saker som kan öppnas och stängas.
- Fordon och farkoster.
- Nummerskivor, strömbrytare och knappar.
- Kläder m.m.

TADS är liksom många av de listade programmen här till för att skapa färdiga produktioner i. Spelutvecklare som skriver i TADS rekommenderas att skriva ett manus först:

The first step is designing the game. Using a system like TADS, which makes it very easy to get started with a game, you may be tempted to jump right in and start programming your game, but you should instead turn off your computer and get out a pad of paper. [Roberts 02]

TADS kan möjligtvis fungera som ett skissverktyg men för ändamålet att skriva manus saknar det en del funktioner. Bl.a. har användaren ingen översikt över berättelsen och kan heller inte skriva interna beskrivningar till alla objekt.

Erasmatron

Erasmatron [Crawford 03] är en utvecklingsmiljö skapad av speldesignern Chris Crawford för konstruktion av interaktiva s.k. berättelsevärldar (eng. *storyworlds*). En författare eller konstruktör kan här skapa en ensemble av upp

till 30 aktörer och ge dem namn och utseende och specificera varje aktörs speciella personlighetsdrag. Till detta kan han/hon beskriva 62 olika scener där dramat ska utspela sig och placera ut upp till 126 aktiva föremål.

Konstruktören kan därefter programmera aktörerna att utföra handlingar som beskrivs som verb med tillhörande subjekt och objekt. För varje handling kan aktörerna tilldelas ett antal alternativa mothandlingar beroende på deras personlighet, relationer, känslor och historia. På detta sätt kan berättelsens upphovsman skapa ett övergripande händelseförlopp som framskrider enligt en tidsbestämd sekvens eller som utlöses av specifika dramatiska vändpunkter.

Tonvikten i Erasmatron ligger på interaktionen mellan karaktärerna, deras personlighetsdrag och emotionella tillstånd. Det mesta av det som händer i programmet består just av att aktörer gör saker med varandra [Crawford 97]. De handlingar konstruktören skapar är på sätt och vis dramats regler men de specificerar inte nödvändigtvis vilka händelser som verkligen kommer att inträffa. Resultatet av konstruktörens arbete i Erasmatron kan upplevas med hjälp av en annan programvara – Erasmaganza [Crawford 03]. Där kan spelaren se aktörerna, läsa texter som beskriver vad dessa säger och gör och interagera med dem enligt de handlingar konstruktören definierat.

Erasmatron har ett objektorienterat förhållningssätt och stödjer interaktivt berättande genom att händelsestrukturen kan byggas som ett nätverk. Dess tydliga karaktärsorientering kan dock vara ett hinder för vissa berättelser och då resultatet stöps i en viss form (som inte är text) passar det heller inte för att skriva manus i.

ARBETSFÖRLOPP

Efter att ha undersökt interaktivt berättande i olika former och fått en uppfattning om vilka arbetsmetoder och problem manusförfattare har att göra med var det dags att skrida till verket. Nya frågor hade uppstått på vägen och väntade på svar. Vad skulle jag göra med all information? Hur skulle den behandlas? Vart skulle den leda mig?

Val av arbetsmetod och angreppssätt

Förundersökningen var en förutsättning för det fortsatta arbetet. Där hade jag studerat flera olika uttrycksformer för att hitta likheter och skillnader mellan dem och deras sätt att närma sig interaktivt berättande. Jag hade gjort intervjuer för att komma närmare min målgrupp. Jag hade sökt efter liknande projekt som skulle kunna ge mig vägledning och viktigast av allt: Jag hade funnit ett behov av det jag ville utveckla.

Den information jag fått behövde nu struktureras. I mer än ett fall hade mina källor pekat på ett objektorienterat förhållningssätt (intervjun med Seter samt [Bolter 91] och [NE 03]) och ett första steg skulle kunna vara att titta på om interaktiva berättelser kan modelleras efter sina ingående objekt eller byggstenar. När jag identifierat ett lämpligt antal byggstenar planerade jag att över-sätta dem till objekt i en datastruktur. Ett annat viktigt moment skulle vara att hitta en gemensam terminologi för de begrepp jag skulle handskas med. Det var nödvändigt att ha ett fastslaget språk med vilket jag kunde kommunicera mina tankar.

Förundersökningen resulterade i en mängd förslag på funktionalitet hos ett manusverktyg och blev på så sätt ett slags allmän funktionsanalys. Till-sammans med en mer specifik funktionsanalys för varje objekt i datastrukturen skulle den ge riktlinjer för vad den tänkta användaren skulle kunna göra i programmets gränssnitt. Efter att ha provat på manusskrivande, läst teorier om samt diskuterat interaktivt berättande längtade jag efter ett konkret system att testa. Hittills hade jag inte funnit ett enda och nu beslöt jag mig för att bygga ett själv. Med en prototyp som utkast och förslag skulle debatten berikas med nya idéer och insikter. Det skulle finnas något att samlas kring som var lättare att förstå än en skriven specifikation och det skulle visa vad som var möjligt och inte möjligt att göra.

För att testa prototypen hade jag ett riktigt manus att tillgå [Seter 02:b] och detta ville jag använda för att försäkra mig om att allt som kunde representeras på papper gick att representera i min prototyp. Jag provade också att själv skriva ett antal mindre äventyr i och för prototypen. Målsättningen var inte att skapa ett färdigt resultat, utan att skapa något att utgå ifrån i vidare diskussioner med målgruppen.

Magnus Seter hade varit positiv till ett andra möte och var nyfiken på vad jag skulle komma fram till. Vi bokade därför en ny intervju i slutet av prototyp-fasen då vi tillsammans kunde diskutera prototypens egenskaper och då jag lovade att återknyta till resultatet från den första intervjun.

Hela arbetsfasen kom att präglas av eftertanke och iterativ utveckling. Varje idé bollades med minst en, ibland tre personer. Inför varje stort beslut fanns mer än ett förslag och när jag kommit till ett resultat ändrade jag perspektiv för att se om det höll även ur en annan synvinkel. Jag skickade frågor via e-post till

intervjupersonerna för att i efterhand vara säkra på att jag uppfattat dem rätt och för att få deras uttalanden förtydligade. Varje vald begränsning i funktionalitet eller arbetsmängd övervägdes nog så att den inte skulle ha någon oönskad bieffekt för framtida utveckling. Nya idéer fogades till gamla, och gamla idéer refuserades för nya. Trots att arbetet ofta tog två kliv bakåt för varje steg framåt kunde jag i efterhand se att jag närmat mig mitt mål.

Analys av interaktiva berättelsers uppbyggnad

Det första steget i arbetsfasen gick ut på att titta på interaktiva berättelser och hitta gemensamma strukturer att utgå ifrån. Helt medvetet präglades mitt synsätt mer av kunskap om berättelser än om berättandets dramatiska uppbygg. Även om det fanns programvara som arbetade med abstrakta drama-begrepp som t.ex. ”motiv”, ”antagonist” och ”subhandling” [Screenplay 03] hade intervjuerna pekat på att jag hellre skulle ha ett konkret och objekt-orienterat perspektiv på berättelselementen.

Berättelsers interaktiva struktur

Mångfalden av interaktiva strukturer i berättande produktioner är större än man kanske skulle kunna tro. För att hålla ett öga på vad manusverktyget skulle kunna representera sammanställde jag en lista på alla de strukturer jag hade erfarenhet av. Listan, som följer nedan, är troligtvis inte komplett och bara ett sätt av många att kategorisera interaktiva produktioner på. Vidare vill jag gärna påpeka att variationen inom varje listpunkt är enorm – två produktioner med samma struktur kan uppfattas som helt olika beroende på andra variabler som spelar in. I praktiken blandar man ofta flera strukturer i en interaktiv produktion.

- **Enkel linjär struktur.** Denna struktur har kritiserats en del och inte helt setts som ”äkta” interaktion eftersom användarens möjlighet att påverka berättelsen begränsas till valet ”fortsätt”. Strukturen kan liknas vid en rak landsväg där vandraren på vissa platser har möjlighet att stanna upp och vila. I denna grupp hittar man ”statiska” berättelser som t.ex. CD-ROM-produktionen *Ceremony of Innocence* [RealWorld 98].
- **Trädstruktur.** En trädstruktur är som ett träd där varje gren delar sig i nya kvistar och där ingen kvist någonsin växer ihop med de andra igen. Att skriva en interaktiv berättelse på detta sätt är oekonomiskt och storleksbegränsande eftersom antalet nya vägar relativt antalet val användaren får göra snabbt blir mycket stort.
- **Parallell flervalsstruktur.** Strukturen liknar ett vägnät med flera vägskäl där vägarna antingen delar på sig eller går ihop igen och där vandraren endast kan gå i en riktning. Ett enkelt exempel på parallell flervalsstruktur där spelaren inte tillåts komma tillbaka till de situationer han/hon har besökt är leken *Sten, sax och påse*. För varje nytt ”vägskäl” finns tre vägar, och även om en väg visar sig bättre än de andra (spelaren vinner omgången) kommer spelaren alltid till samma nya vägskäl oavsett vilken väg han/hon tar.
- Läger man till möjligheten för spelaren att återkomma till en tidigare valsituation och kanske göra ett annorlunda val får man en **ringformad struktur**. Det kan vara tydligt, som att en modifierbar slinga spelas om och om igen. Ringstrukturen kan också vara underordnad en annan struktur

som i t.ex. dataspel *Baldurs Gate II* [BioWare 02] (som i första hand är en parallell flervalstruktur) där spelaren i många fall kan ändra sina val om t.ex. gruppmedlemmar och gå tillbaka till tidigare uppdrag. Det finns en fin men viktig skillnad mellan vad som anses vara samma valsituation och vad som anses vara likadana valsituationer.

- I en **nätverksstruktur** är de flesta vägar förbundna med varandra, flera gångriktningar är möjliga och vandraren kan lätt ta sig tillbaka till ett vägsål. Det gör det mer komplicerat för författaren att skapa en sammanhängande berättelse men ger å andra sidan användaren större handlingsfrihet. Ett enkelt exempel på nätverksstruktur är en hemsida med en meny som alltid ger möjlighet att hoppa till alla andra sidor som hör till sajten.
- **Villkorade strukturer** kan vara vilken som helst av föregående strukturer där val- och händelsealternativ kräver att vissa förutsättningar är uppfyllda. Det kan handla om värdet på variabler som styr spelkaraktärens egenskaper, vilken tid situationer utspelar sig, hur många val spelaren tidigare har gjort, slumpen, andra spelares val, antal utförda uppgifter etc. Villkorade strukturer ökar dynamiken i interaktiva produktioner enormt och är mycket vanliga i dessa sammanhang.
- I en **struktur med fri interaktion** kan mängden val- och handlingsalternativ en spelare har inte rimligen specificeras i förhand. Rollspel, vars struktur typiskt präglas av fri interaktion, låter oftast spelaren lösa problem på fler sätt än vad manusförfattaren har möjlighet att förutse. Tittar man på interaktiv dialog illustreras denna struktur ytterligare: Interaktiv dialog i dataspel fungerar ofta så att spelaren har ett fåtal repliker att välja på ("ja" eller "nej"). I ett textbaserat spel där spelaren har tillgång till en begränsad vokabulär ("go" och "west" kan kombineras till "go west") är möjligheterna lite fler. Tänker man sig ett program med en avancerad s.k. *parser*, t.ex. *Eliza* [Aarseth 97], där användaren i stort sett kan skriva in vilken replik som helst med sitt tangentbord har gränsen mot fri interaktion passerats. Fri interaktion behöver (och kan) dock inte skrivas av en manusförfattare. Därför behandlas inte denna struktur vidare i denna rapport.

Att rita pärlor och trådar för att visualisera en produktions interaktiva struktur var till en början en lockande tanke. Jag tänkte mig att om författaren kunde se sitt manus som en pärlväv eller en *mindmap* skulle han/hon direkt få en god överblick över det. Det visade sig dock rätt snart att denna tanke inte var möjlig att genomföra i praktiken. De flesta dataspel har en mycket komplex struktur och skulle snarare bli en härva än en överskådlig väv eller karta om man försökte rita upp dem. Många situationer skulle troligtvis inte gå att rita upp alls eftersom bilder är tvådimensionella och interaktiva berättelser flerdimensionella. Följande frågor, varav de flesta förblev obesvarade, fick mig att stryka min pärlvävsmetafor:

- Hur ser väven ut då spelaren måste klara exempelvis tre av fem uppdrag innan han/hon får gå vidare?
- Hur ritas händelser upp som kan inträffa lite när som helst? Hur ritas händelser upp som måste ske inom två checkpunkter?
- Hur representeras uppdrag som är valfria att ta/inte ta? Vill man se möjliga händelser och nödvändiga händelser i samma vy?

- Hur förenklat vill författaren se sitt manus? Ska bara stora och/eller sammansatta händelser visas? Ska det gå att zooma in och ut ur strukturen?
- Hur gör man om spelaren kan röra sig både i rum och tid?
- Interaktionen kan ske på flera plan. Valalternativ som ”spara” eller ”titta i ryggsäcken” eller ”res över kartan” är typiskt alltid möjliga medan val som ”undersök klockan” och ”gå till höger” beror på vilken situation spelaren befinner sig i. Valalternativen kan sägas höra till separata interaktionsspår och frågan är hur dessa ritas upp i förhållande till varandra.
- Hur ser den interaktiva strukturen i ett strategispel som t.ex. *Command and Conquer* ut?

En objektorienterad begreppsvärld

Ett andra viktigt steg i kartläggandet av interaktiva berättelser var att identifiera deras gemensamma byggstenar. Jag hade stött på många olika begrepp och nu gällde det att med objektorienterade ögon sälla ut de som var mest allmängiltiga och överordnade. Jag koncentrerade mig på att plocka fram de byggstenar som jag trodde hade störst betydelse först och hoppades att de sedan skulle peka på vilka mindre byggstenar som kunde vara relevanta. Sällningen var naturligtvis ett högst subjektivt urval men inspirerades av mina resultat från förundersökningen. Följande begrepp/byggstenar identifierades:

- Rumsbegreppet. I alla produktioner och berättelser finns någon form av plats, area, scen, bakgrund, arena eller miljö där händelserna utspelar sig. Det kan röra sig om allt ifrån en huvudmeny i ett dataspel till en scen med kulisser i en teaterpjäs eller en uppbyggd medeltida by i ett lajv. Med någon form av karta eller (då två dimensioner inte räcker till) listning av grannområden skulle författaren kunna få en överblick över hur olika platser är förbundna med varandra.
- Personbegreppet. En berättelse, interaktiv eller ej, skulle vara otänkbar utan någon form av invånare. Invånarna skänker liv åt berättelsen och styrs i interaktiva sammanhang ofta av en publik. Publikens perspektiv i förhållande till berättelsens invånare och dess påverkan på berättandet har varit föremål för diskussion länge och perspektivet är intressant ur dramasympunkt eftersom det kan utnyttjas i den kreativa delen av berättandeprocessen. I lajvsammanhang där aktörer och publik är desamma skiljer man på skådespelare och deras roller. Användare i virtuella världar representeras av avatarer och i rollspel talar man om spelare/spelledare och deras respektive karaktärer. För detta arbete valde jag att inte representera spelare/publik på något sätt eftersom de inte ingår som aktörer i manuset. Jag valde också att inte skilja på olika slags karaktärer i interaktiva berättelser trots att det förekommer skillnader mellan rollerna i både lajv och rollspel. Skillnaderna består här i vem som spelar en karaktär, inte i direkta olikheter mellan karaktärerna.
- Sakbegreppet. Ord som ”rekvisita” och ”inventory” säger något om att det i de flesta (interaktiva) berättelser finns föremål att laborera med. De kan delas, öppnas, kombineras, bäras, manipuleras, vara ett hinder, explodera osv. Att som i *Neverwinter Nights* [Bioware 03] dela upp föremål i tre kategorier beroende på interaktionsmöjlighet kändes inte naturligt. Jag

utgick ifrån att en kategori skulle räcka långt och att den, om nödvändigt, kunde kompletteras senare.

- Medlemsbegreppet. Inom i synnerhet rollspel och lajv är möjligheten att gruppera personer/karaktärer mycket viktig. En karaktär i ett rollspel kan t.ex. höra till ett skrå och en kyrka, vara av en specifik ras och höra till ett visst samhällsskikt. I dataspel är det också vanligt att föremål är av en viss typ, t.ex. är juveler och mynt värdeföremål som det går att handla med och revolverar och svärd är vapen med vilka rollkaraktärer kan försvara sig. I ett manusverktyg måste därför både personer och saker kunna grupperas och de måste kunna ha medlemskap i flera grupper samtidigt.
- Mediereferenser. Brådhe talade om byggstenarna ord, bild och ljud och Rajka nämnde i sitt lösningsförslag [Rajka 96] arkiv med ljussättningar, ljud och projektioner. Båda pekade på det faktum att manusförfattaren ofta vill referera till bilder, kartor, ljud, animationer etc. i sin text. Det är inte något som är unikt för interaktiva manus, men det är ett så pass viktigt och ofta krångligt moment att det inte kan förbises. Till en enkel sak som t.ex. en replik kan författaren vilja ange inte bara vad som sägs utan hur det sägs, av vem det sägs och vad ljudfilen heter där man kan höra repliken uttalas av en röstskådespelare. Vilka typer av media författaren vill använda sig av är förstås olika från projekt till projekt och enklast blir det om författaren själv kan få definiera sina medietyper.
- Egenskaper och beskrivningar. Oavsett medieform vill manusförfattaren ha en möjlighet att specificera egenskaper hos sina karaktärer och föremål och skriva anvisningar för miljöer, bildelement, händelser etc. Tyvärr skulle det inte gå att skapa en komplett lista på alla möjliga egenskaper (och värden dessa skulle kunna anta). Enklast vore därför att till en början inte se egenskaper som ett fristående begrepp utan som en del av (textuella) beskrivningar. Vidare måste det gå att skilja på publika beskrivningar som alla får läsa, och privata eller interna beskrivningar ämnade för t.ex. författarens kollegor, spelledaren eller en specifik skådespelare.
- Hierarkistruktur. När man i böcker talar om kapitel och avsnitt talar man inom teatern om akter och scener. I dataspel kan uppdelningen modul/uppdrag/uppgift förekomma och i lajv finns det en möjlig gräns att dra mellan episka händelser och episodiska händelser. Det är tydligt att längre texter och produktioner måste kunna delas in i mindre stycken som kan ordnas hierarkiskt.
- Händelse- och handlingsbegreppet. Brådhe uttryckte träffande hur en interaktiv berättelse styrs av handlingar, som i sin tur skapar händelser. Handlingarna (eller valen) görs aktivt av produktionens användare. Händelserna är oftast en reaktion eller konsekvens av användarens interaktion eller passivitet. Även om de båda begreppen tycks vara motsatser i ett passivt/aktivt perspektiv skulle man kunna säga att handlingar på sätt och vis också är händelser. I den tidigare nämnda pärlvävsmetaforen skulle en handling representeras av en pärla medan en händelse skulle representeras av tråden mellan två pärlor.
- Kausalitetsbegreppet, som i sin tur består av villkor och konsekvenser. Varje handling eller händelse i en interaktiv berättelse kan ha ett villkor knutet till sig. Typiskt är att en händelse har villkoret att en specifik

handling måste utföras för att händelsen ska kunna inträffa: *Dörren öppnas endast om du har tryckt ner dörrhandtaget*. På samma sätt har handlingar, men även händelser konsekvenser: *Slänger din medspelare en bytta vatten över din rollkaraktär eller om det börjar regna i berättelsen blir din karaktär blöt och frusen*. Ofta är villkor och konsekvenser det lim som binder samman händelser och handlingar med varandra.

- Dialogbegreppet. Dialog är en kombinationsform av handling och händelse vilket gör att även repliker kan ha villkor och konsekvenser knutna till sig. Dialoger har ofta en mycket enklare interaktiv struktur än den övriga produktionen och därför kan det gå att visualisera dialoger på ett enklare sätt än vanliga händelser och handlingar.
- Tidsbegreppet. Så här långt kommen kände jag att det var dags att gå vidare i examensarbetet. Tidsbegreppet var intressant och nödvändigt att studera, men kanske en lämplig begränsning i detta läge. Jag kunde heller inte se att det nämnvärt skulle påverka övriga begrepp.

Val av terminologi

A final difficulty rests in the fact that interactive design is a young field with a terminology that is not always consistent, involving the cross-disciplinary use of language. [Stansberry 97]

Som vi precis sett kan man finna flera allmänna begrepp i interaktiva berättelser. Dessa begrepp har olika namn beroende på vilken form av berättande det rör sig om. I förklaringen av rumsbegreppet radades det t.ex. upp sex olika benämningar (plats, area, scen, bakgrund, arena, miljö) för i stort sett samma sak. Mångfalden av termer och deras ibland något olika betydelse förvirrade mig redan från början. Jag beslöt därför att skapa en egen terminologi för de begrepp och fenomen jag ville handskas med. Med ett överenskommet språk skulle kommunikationen bli tydligare och förhoppningsvis skulle manusverktygets tillgänglighet förbättras.

För att hitta ett väl fungerande språk ställde jag en del krav på vokabulären. För det första måste den vara så intuitiv som möjligt. Konkreta ord inspirerade av verkligheten hade här en fördel framför abstrakta begrepp. "Plats" och "föremål" är mer intuitiva än "område" och "objekt". Terminologin måste dessutom vara tillräckligt entydig för att inga ord skulle förväxlas med konkurrerande begrepp. Ord som "person" ströks eftersom det skulle kunna beteckna både en riktig människa och den roll han/hon antagit i en berättelse. Vidare ställde jag krav på att terminologin skulle ha en viss allmängiltighet. Begrepp som "dialog" fungerade i alla sammanhang medan "kapitel" direkt förde tankarna till litterära media. Jag utgick i första hand från etablerade ord och undvek nya, okända uttryck. Varför tala om "påföljd" eller "händelse-resultat" när det naturliga "konsekvens" redan fanns?

Jag hade planerat att använda engelska som huvudspråk i manusverktyget vilket gjorde att jag fick skapa en terminologi både för svenska och engelska sammanhang:

- Rumsbegreppet benämndes "area" på både svenska och engelska. Refuserade förslag var "plats" (vilket var för konkret, det passade inte som beteckning för t.ex. en huvudmeny), "rum", "miljö", "scen" (en scen är inte

bara en omgivning utan även ett annat ord för en situation) samt engelskans "location" och "scene".

- Personbegreppet benämndes "karaktär" på svenska, "character" på engelska. Refuserade förslag var "person", "roll" och "avatar".
- Sakbegreppet fick namnet "föremål" respektive "item". Andra förslag var "sak", "objekt" och "thing".
- Medlemsbegreppet betecknades genom att det fanns "grupper" (av personer eller föremål), på engelska (item- och character-) "groups".
- Mediereferenser var en punkt som kanske borde ha ägnats mer tanke. En referens kallades på svenska "dokument", på engelska "document", vilket inte riktigt stämde. Referenser pekade nämligen inte bara på en fil eller ett dokument, utan även på textuella anvisningarna för filen. Varje dokument var dessutom av en "dokumenttyp"/"documenttype".
- Hierarkibegreppet benämndes med ordet "modul" på alla nivåer. En modul kunde vara submodul till en modul och huvudmodul åt en annan. Förslag som "kapitel", "avsnitt", "moment" och "scen" ströks. Den engelska varianten var naturligtvis "module".
- "Handling" och "händelse" behöll sina namn även om det fanns andra förslag såsom "val", "valalternativ", "händelseförlopp" och "skeende". På engelska hette begreppen "action" och "event" med refuserade förslag som "choice", "selection" och "happening".
- Kausalitetsbegreppets "villkor" och "konsekvens" behöll också sina namn med andrahandsvalen "förutsättning" och "påföljd" eller "reaktion". Översatt blev det "condition" och "consequence".
- "Dialog" och "replik" var givna vinnare liksom "dialogue" och "line".
- Jag valde inget särskilt ord för att beteckna publiken/användaren, men eftersom jag ofta utgick från dataspelsvärlden i mitt examensarbete var "spelare" den term som förekom flitigast i mitt språk. Naturligtvis skulle det ha behövts ett bättre ord för detta, i synnerhet i denna rapport.

Utöver vokabulären ovan skapade jag ett nytt och mycket centralt begrepp: "Situation." **En situation är ett tillfälle då vissa förutsättningar gäller och då det finns ett eller flera möjliga valalternativ och händelseförlopp som kan ta spelaren till en ny situation.** En situation kan med andra ord rymma flera konkurrerande handlingar och händelser. Beroende på vilka villkor som för tillfället är aktuella inträffar antingen en händelse eller så ges spelaren vissa handlingsalternativ. Konsekvensen av en handling eller händelse är att spelaren hamnar i en ny situation med nya förutsättningar.

Prototyp av manusverktyg

I förundersökningen hade jag fått höra att det kanske inte var möjligt att skapa ett verktyg för interaktiva manus. För att bevisa motsatsen och konkretisera mina idéer och teorier bestämde jag mig för att bygga en prototyp av ett manusverktyg. Prototypen skulle visa var svårigheterna låg men även vad som var möjligt med teknikens hjälp. Projektet och prototypen döptes till Mazermind för att återknyta till författarens roll som hjärnan bakom en berättelse och till labyrinthen som en metafor för interaktiva system. Även

kopplingen till orden ”maze” (förvirring) och ”amaze” (förundra) spelade in i namnvalet.

Val av teknik

Min första uppgift var att välja den teknik projektet skulle grundas på. Följande stycke presenterar vilka alternativ jag hade, vilka val jag gjorde och vilka fördelar, nackdelar och begränsningar dessa val innebar.

Språk

Valet av programmeringsspråk stod mellan tre alternativ: C++, Java eller ett språk för webbt teknik som t.ex. ASP (Active Server Pages).

Fördelarna med C++ var två: Dels hade jag tillgång till ett ramverk med kod för grundläggande funktioner och dels hade jag tillgång till support. Ramverket bestod av verksamhetsobjekt som kunde sköta kopplingen mot en databas. Det bestod även av grafiska komponenter som kunde förstå de data verksamhetsobjekten returnerade från databasen. Funktioner för databaskontroll och grafik skulle vara tidsödande att skriva själv men samtidigt nödvändiga i min prototyp och här passade ramverket som handen i handsken. Tillsammans med möjligheten till handledning av Nicklas Kittelmann, en civilingenjör med stor erfarenhet av programmering av databasapplikationer, gjorde detta att C++ var mitt förstahandsval som programmeringsspråk. Den största nackdelen med valet var att manusverktyget inte skulle vara tillgängligt på alla plattformar utan tvingade författare att byta till PC från t.ex. Mac. En annan nackdel var att jag aldrig tidigare programmerat i C++ och att det därför skulle gå åt en del tid till inläring.

Det andra alternativet, att skriva Mazermind i Java, hade just fördelen att det skulle fungera på de flesta plattformar. De författare som använde Mac skulle kunna fortsätta med det. Jag hade även programmerat i Java tidigare, men kände mig en aning ringrostig. Utan liknande support och ramverk som alternativet med C++ erbjöd mig räknade jag med att det skulle ta längre tid att skapa en prototyp. Dessutom skulle jag inte få lära mig ett nytt språk.

Att skriva kod i ASP hade jag tidigare erfarenhet av och det var en lösning som även skulle passa för Mac-användare. Mazermind skulle då endast behöva installeras på en enda dator, oavsett antal författare eller användare som var inblandade i ett projekt. Manusverktyget krävde dock att denna dator skulle innehålla en webbserver och installationen skulle inte på långa vägar vara lika enkel som om verktyget var ett program skrivet i C++ eller Java. Den största nackdelen var ändå att gränssnittet skulle bli långsamt och segt p.g.a. längre laddnings- och uppdateringstider.

Utvecklingsmiljö

Grafiken i det ramverk jag skulle använda mig av grundade sig på Microsoft Foundation Class; en standard för C++-programmering. Standarden användes i Microsofts utvecklingsmiljö Visual C++, i vilken också ramverket var skrivet. Eftersom jag dessutom hade tidigare och positiva erfarenheter av att arbeta i systemprogrammet Visual InterDev beslöt jag mig för att använda Visual C++ (6.0) för utvecklingen av Mazermind.

Databaslösning

Det var ingen tvekan om att manusverktyget skulle generera mängder av data som behövde klassificeras efter olika mallar. Därför var det en lika självklar

som objektorienterad lösning att lagra informationen i en databas. Frågan var snarare vilken databaslösning som kvalificerade sig bäst i detta fall. Tekniken skulle inte bara användas av mig under utvecklingsarbetet utan även vara en del av det verktyg som installerades på slutanvändarens datorsystem. Detta ställde ytterligare krav på enkelhet och lättillgänglighet. Jag hade två huvudkandidater för databaslösningen: Access och SQL Server.

SQL Server var det avancerade alternativet med hög prestanda både gällande mängden data som kunde lagras och mängden användare som samtidigt kunde arbeta mot databasen. Produkten var dock både komplicerad och dyr och tiden för att installera databasen låg på flera timmar. Varken kravet på enkelhet eller lättillgänglighet uppfylldes särskilt bra av SQL Server varför jag valde att titta närmare på det andra alternativet.

En Access-databas skulle ha sämre prestanda, vilket kunde bli problem om många användare samtidigt arbetade med samma projekt. Av intervjun med Seter hade jag dock förstått att det var ovanligt med produktioner med så många författare inblandade att den lägre prestandan överhuvudtaget skulle märkas. Troligtvis skulle inte heller den något sämre lagringskapaciteten vara något problem utan databasen skulle gott och väl svälja alla manus. En Access-databas bestod av en enda fil, vilket skulle göra den enkel att hantera. Access var ett billigt program och ett lättillgängligt sådant eftersom det ingick i samma programpaket (Office) som de verktyg författare redan använde sig av.

För prototypen valde jag att använda Access, men som Mazermind byggdes stödde den även en lösning med SQL Server.

Från terminologi till datastruktur

Det objektorienterade perspektivet och valet av teknik banade väg för hur översättningen från terminologi till datastruktur skulle gå till. Den digitala struktur som bäst lämpade sig för att representera min begreppsvärld var en databas. I den databas som skapades för Mazermind omvandlades många av de identifierade begreppen från terminologin (ovan) till varsin egen tabell: *MzItem*, *MzItemGroup*, *MzArea*, *MzDocument*, *MzDocumentType*, *MzModule*, *MzCondition*, *MzConsequence* och *MzSituation*. Medlemskap i grupper, grannrelationer mellan areor och associationer till föremål behövde också representeras av egna tabeller: *MzGroupConnection*, *MzMapConnection* och *MzItemConnection*. I Bilaga 4 finns en vy över den kompletta databasstrukturen.

För att kunna representera begreppen som tabeller i en databas måste jag ur ett användarperspektiv fastställa vilka egenskaper som karaktäriserade varje begrepp. De flesta objekt skulle t.ex. behöva kunna namnges av författaren och hade därför attributet *Name*. Mycket av författarens arbete skulle dessutom gå ut på att beskriva olika saker. Eftersom en del beskrivningar var publika (skulle till slutproduktion) och andra var av mer privat natur (hörde till arbetsprocessen) valde jag att skapa attributen *DescriptionExternal* och *DescriptionInternal*. Ett abstrakt begrepp som "Villkor" skulle inte gå att undersöka av en slutanvändare och hade därför heller inte någon publik/extern beskrivning, medan något så konkret som ett föremål kunde ha både en intern beskrivning för dess modellerare (*Bananen ska bestå av gula polygoner*) och en extern beskrivning ägnad slutanvändaren (*Föremålet du betraktar tycks vara en gulaktig frukt med söt doft*). För att alla objekt i databasen skulle gå att hålla rätt på begåvades de med ett automatiskt genererat och unikt identifikationsnummer (*ID*). Ett identifikationsnummer för varje objekt skulle dessutom vara tids-

ekonomiskt vid sökning efter objekt i tabellerna. Funktioner som versionshantering och administration av användare förbereddes genom tabellerna *Tag* och *User*. Alla objekt fick också attribut som kopplade dem till versionshanteringen, dvs. egenskaper som visade när och av vem objektet senast ändrades (*Version*, *SavedDateTime*, *SavedByUserID*, *CheckedOutDateTime* och *CheckedOutByUserID*).

Att gå direkt från terminologi till färdiga databastabeller var varken ett troligt eller önskvärt arbetsförlopp. Att hitta begrepp, namnge dem och representera dem digitalt var en iterativ process som krävde både förmåga att behålla fokus på större sammanhang och en ödmjukhet inför att göra ändringar. En av skillnaderna mellan terminologin och datastrukturen var att de två olika begreppen föremål och karaktär rymdes i samma tabell i datorn (*MzItem*). De båda begreppen hade precis likadana egenskaper och efter en viss tvekan beslöt jag att de, rent tekniskt sett, kunde representeras i samma format. För att ändå kunna skilja på begreppen i gränssnittet gavs de ett attribut, *Type*, som angav om det var ett "item" eller en "character". På så sätt fanns även en möjlighet att i framtiden inte bara skilja på karaktärer och föremål utan även på karaktärer och föremål av olika typer (jfr rollspelandets *playing character* och *nonplaying character*).

En annan förändring i datastrukturen var situationsbegreppet med de båda tabellerna *MzSituation* och *MzSitpart*. En situation har egenskaper som i vilken area och i vilken del (modul) av manuset den inträffar. Dessa egenskaper sköttes av den första tabellen. Varje situation består i sin tur av flera möjliga händelseförlopp. Dessa förlopp motsvarade rader i den senare tabellen. Varje händelseförlopp fick attribut som vilken situation det tillhörde, vilket villkor och vilken konsekvens som gällde och vilken handling/händelse som förknippades med förloppet. I databasen hamnade handlingar och händelser i samma tabell; *MzAction*. Vidare skulle en handling/händelse kunna vara sammansatt av flera delhändelser som t.ex. när flera animationer spelas efter varandra. Även för dessa handlings- och händelseatomer fanns en tabell, kallad *MzActionPart*.

I förundersökningen hade en tanke dryftats om hur likt det är att skriva interaktiva dialoger och att skriva interaktiva berättelser. Den idén fick nu liv då det visade sig att jag inte behövde skapa några nya tabeller för dialoger och repliker. Istället kunde en karaktärs (eller ett föremåls för den delen) tur att tala representeras som en situation. Situationens händelseförlopp motsvarade då olika möjliga repliker karaktären hade att välja på. Precis som med handlingar och händelser kunde varje replik ha ett villkor och en konsekvens (t.ex. en motreplik).

Varje tabell och begrepp i databasen hade sina speciella egenskaper och särdrag. De vanligaste egenskaperna hade att göra med vilken typ objektet var av, vilken förälder i hierarkin objektet hade och vilken area/modul/villkor/fil som associerades med objektet. Med det fanns även unika egenskaper såsom konsekvenstabellens *NextSituationID* vilken pekade på vilken situation som skulle komma härnäst och föremålstabellens *Number* som angav antalet föremål (t.ex. *30 stycken silverpenningar*).

Från datastruktur till funktionsanalys

I förundersökningen hade jag fått en mängd förslag på hur ett manusverktyg skulle kunna se ut och vad det skulle kunna göra. Jag hade hela tiden haft

dessa förslag i bakhuvudet men kände nu att det var dags att sammanfatta dem innan jag gick vidare. En passande form för sammanfattningen skulle vara en funktionsanalys; en metod ursprungligen tagen från industridesignen men som nu även förekommer frekvent inom utvecklingen av informations-system. En funktionsanalys lämpar sig för att summera och strukturera information i ett tidigt skede och visar vad en produkt skall göra men inte hur det ska gå till [Löwgren, Stolterman 99].

Efter sammanfattningen gjorde jag även en mer detaljerad funktionsanalys för de ingående tabellerna/begreppen i databasen. Denna funktionsanalys gjordes relativt tidigt i examensarbetet, innan databasen fått sitt slutliga utseende. Därför kom inte hanteringen av grupper med och inte heller lågprioriterade områden som projektadministration och versionshantering. När gränssnittet började byggas blev det uppenbart att det skulle behöva göras förändringar i listan på specifika funktioner.

Sammanfattande funktionsanalys

Sammanfattningsvis skulle manusverktyget underlätta för författaren att:

- skriva icke-linjärt. (Verktygets huvudfunktion.)
- överblicka manuset, förstå dess uppbyggnad och kontrollera flödet i berättelsen. (Nödvändig funktion.)
- skapa ett manus ur vilket ingående beståndsdelar och medietyper lätt kan extraheras. (Nödvändig funktion.)
- presentera manuskriptet så att slutresultatet framgår tydligare. (Nödvändig funktion.)
- se själva berättelsen trots all logik, alla referenser och beskrivningar. (Nödvändig funktion.)
- skriva interaktiv dialog. (Nödvändig funktion.)
- uttrycka villkor för händelser. (Nödvändig funktion.)
- sortera och dela upp både berättelsen och dess ingående objekt. (Nödvändig funktion.)
- numrera och namnge objekt (efter en standard). (Önskvärd funktion.)
- söka och ersätta skriven text. (Önskvärd funktion.)
- komma ihåg vad som är kvar att skriva. (Önskvärd funktion.)
- hålla reda på versioner. (Önskvärd funktion.)
- samarbeta med andra författare. (Önskvärd funktion.)
- komplettera texten med bilder och liknande. (Önskvärd funktion.)

Specifik funktionsanalys

Följande lista med förslag på specifika funktioner för varje begrepp i databasen gjordes:

- För varje **area** ska det gå att se och ändra dess namn, dess interna och externa beskrivning, vilken modul den tillhör (om den hör till någon modul) och till vilka dokument den är kopplad. Samma sak gäller för de föremål och karaktärer som befinner sig i arean och för de situationer som utspelar sig i arean. Att kunna se alla areor tillsammans i en grafisk vy där

deras relationer visualiseras skulle vara önskvärt. Primärt ska det åtminstone gå att se vilka grannareor en area har och hur den är kopplad till dessa (t.ex. riktning).

- **Föremål och karaktärer** ska kunna särskiljas i gränssnittet. Deras egenskaper som namn, antal, intern och extern beskrivning ska vara synliga och redigerbara. Länkar till objekt som vilken area de befinner sig i och i vilka moduler de förekommer ska också gå att redigera. Eventuellt skulle en funktion för att skapa och ändra ett föremåls innehåll/en karaktärs ägodelar (och tvärtom) vara bra att ha. Det ska också gå att förflytta sig till de objekt (t.ex. dialoger och beskrivande dokument) som förknippas med föremålet/karaktären.
- Varje **modul** ska ha ett namn, en extern beskrivning och en intern beskrivning som alla ska gå att redigera. Till varje modul ska det vara möjligt att associera dokument, areor, föremål, karaktärer och situationer och det ska vara lätt att snabbt förflytta sig till dessa objekt. Dessutom bör det vara möjligt att se och ändra moduluhierarkin.
- För **situationer** ska det gå att se och ändra deras namn och deras interna beskrivning. Dessutom ska det gå att se/ändra/hoppa till den area och modul situationen utspelar sig i liksom till associerade dokument och till närliggande situationer. Att kunna se och redigera vilka händelseförlopp (sitparts) som ingår i situationen är nödvändigt. Samtidigt vore det önskvärt att direkt kunna gå till händelseförloppens separata delar (villkor, konsekvenser etc.).
- Ett händelseförlopp (**sitpart**) är snarare ett samlingsobjekt än ett eget begrepp. Därför behöver det endast funktioner för att kopplas till en situation, ett villkor, ett action-objekt (handlingar och händelser) och en konsekvens. Naturligtvis ska dessa vara lätta att nå från händelseförloppet.
- För ett **villkor** ska det finnas möjlighet att redigera rubriken (namnet) och den interna beskrivningen. Det ska även vara lätt att nå dess händelseförlopp och förloppets övriga delar i gränssnittet. Dessutom bör det vara enkelt att välja bland existerande villkor eftersom de troligtvis kommer att återanvändas ofta.
- Utöver den funktionalitet som gäller för villkor ska man då man skapar en **konsekvens** kunna ange/ändra/hoppa till den nya situation konsekvensen leder till.
- Förutom funktionalitet som delas av villkor och konsekvenser ska man för en **action** (handling/händelse) kunna se/ändra dess ingående delhändelser och -handlingar och deras inbördes ordning.
- Varje **actionpart** ska ha en intern beskrivning (t.ex. *Miranda säger med glad röst*) och en text/extern beskrivning (t.ex. *Hejsan och välkomna!*). Dessutom ska det vara möjligt att koppla ett dokument till en actionpart (t.ex. ljudfilen där Miranda säger frasen) och naturligtvis ändra eller hoppa till detta dokument.
- Det ska vara möjligt att se hela **dialoger** med alla dess repliker. Nya repliker ska gå att lägga till och ta bort. **Repliker** ska vara redigerbara och gå att koppla till ett villkor, en konsekvens och en person eller ett föremål som säger repliken.

- **Dokument**, dvs. mediereferenser ska kunna ses översiktligt. De ska kunna sorteras efter typ och kanske filtyp. De ska kunna länkas/associeras till de flesta andra objekt och det ska vara lätt att nå dessa objekt från dokumentet. För varje dokument ska det vara möjligt att se dess namn/beteckning, dess dokumenttyp, dess filnamn, dess filtyp och dess interna beskrivning. Dessa egenskaper ska kunna redigeras och kopieras.
- **Dokumenttyper** ska kunna definieras av användaren.

Från funktionsanalys till gränssnitt

The Object-Action Interface Model (OAI)[Schneiderman 98] går ut på att den som konstruerar ett interaktivt system först identifierar målgruppens aktiviteter och vilka objekt målgruppen arbetar med. Därefter kan konstruktören skapa representationer av dessa objekt och aktiviteter i datorn och eventuellt ordna dem i hierarkier. Mitt arbetssätt hade hittills följt *OAI*-modellens första del och nu när funktionsanalysen var färdig var det dags för en digital representation av mina resultat. Funktioner skulle implementeras och jag måste välja en passande layout för Mazermind.

Målgruppens roll

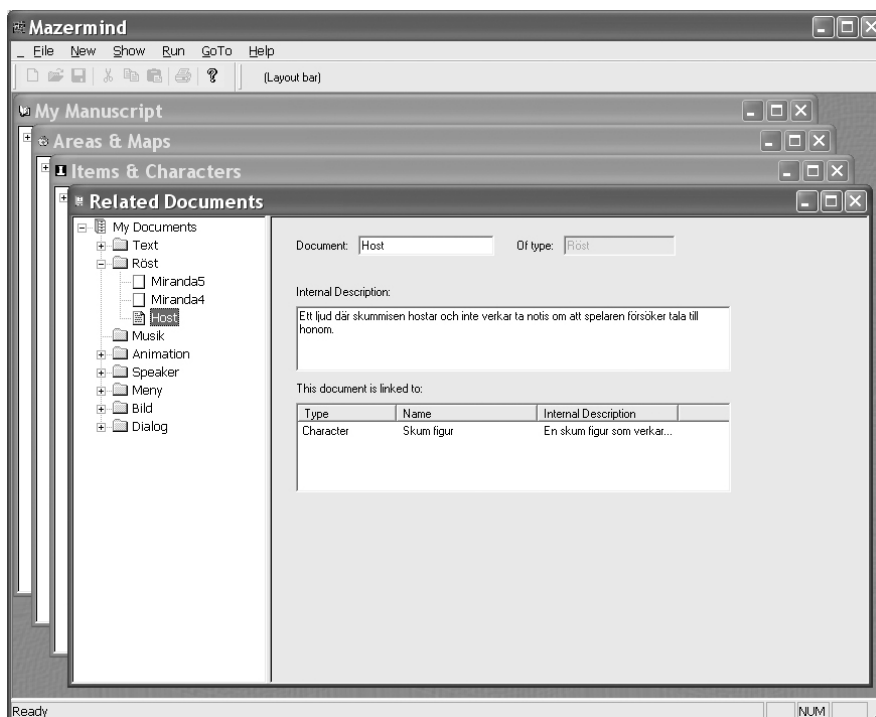
Gränssnittsdesignen följde så långt möjligt de riktlinjer som finns i litteraturen om människa-datorinteraktion. Däremot gjordes designen inte i samarbete med verktygets tänkta målgrupp. En av följderna härav blev att gränssnittet sorterades efter objekt och deras samhörighet i högre grad än det sorterades efter användaren och dennes uppgifter och aktiviteter. Fördelen med den något mer objektorienterade än aktivitetsorienterade designen var att verktyget bättre synliggjorde den begreppsvärld författaren skulle arbeta med. Aktörens roll var heller inte bortglömd; förstudien och intervjuerna hade specificerat vad en användare skulle vilja göra med begreppen och verktygets funktionalitet återspeglade majoriteten av dessa önskemål. Ett intressant alternativ hade dock varit att i gränssnittet skapa en verktygslåda ur vilken funktioner kunde tas fram och användas på objekten som kunde ligga i bibliotek. Ett sådant förfarande skulle dock kräva att ungefär samma aktiviteter kunde utföras på alla objekt.

Metafor

Rent konceptuellt sett bestod Mazermind av tre delar: en del att skriva i, en del att presentera i och en del att söka och extrahera information i. Jag valde att börja med delen för skrivande eftersom den stod närmast författarens roll. Inledningsvis hade jag planerat att verktygets uppbyggnad skulle följa någon slags metafor. En metafor är ett kraftigt verktyg för att snabbt ge användaren en bild av var saker finns och hur de fungerar. Enligt [Kristof, Satran 95] ska en metafor för att fungera dels vara känd av användaren och dels passa rent konceptuellt på det nya innehållet. Här hade jag problem eftersom Mazerminds tänkta användare kom från vitt skilda håll med olika begreppsvärldar i bagaget. Dessutom var objekten i verktyget en blandning från flera olika berättartraditioner och medier vilket gjorde det svårt att hitta en metafor som passade för mer än en delmängd av dem. I t.ex. programmet Director [Macromedia 03] används en metafor inspirerad av teater och film, men den metaforen fungerar inte för att exempelvis gruppera föremål och beskriva konsekvenser. Jag beslöt till slut att inte använda mig av någon speciell metafor utan istället vara så konkret som möjligt.

Menyer och ikoner

Med det val av teknik jag gjorde blev Mazerminds gränssnitt mycket likt den typiska Windows-standard med menyer i överkanten och ett grått arbetsfält över större delen av skärmen (se figur 1). Funktionernas och menyernas namn valdes så att de skulle vara så intuitiva, konsekventa, koncisa och distinkta som möjligt [Schneiderman 98]. Menylisten följde delvis den standard som många program använder sig av med t.ex. Arkiv-menyn längst till vänster. Placeringen av menyvalen gjordes så att de mest frekvent använda alternativen låg så nära till hands som möjligt för användaren [Kristof, Satran 95]. Även möjligheten att nyttja kortkommandon och en verktygslist med ikoner för de vanligaste funktionerna ökade tillgängligheten [Schneiderman 98]. För att tillgodose att olika personer föredrar olika kognitiv stil kompletterade jag menyvalen med ikoner som sedan återkom i de respektive fönstren. Ikonerna utformades så att de skulle vara konkreta och distinkta från varandra och så att de gav visuell återkoppling vid markering [Schneiderman 98]. Frånvaron av konventioner för att skapa ikoner till nya funktioner fick dock till följd att en del av dem ändå inte blev så konkreta och självklara som man kunde ha önskat; ikonerna för objekten "situation" och "sitpart" var typiskt sådana fall.

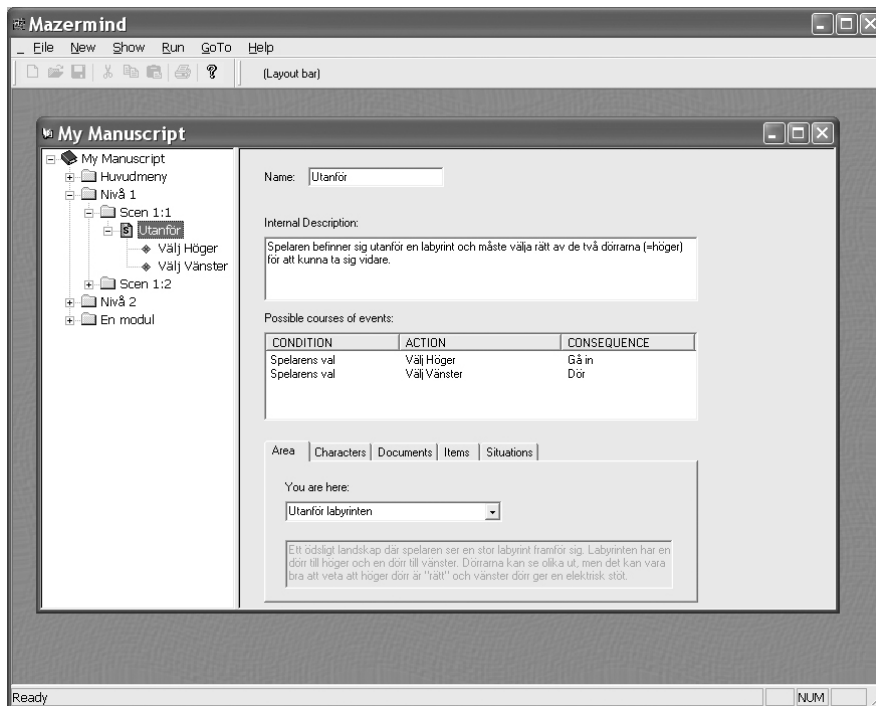


Figur 1. Mazerminds gränssnitt med fyra olika fönster öppna.

Fönster och trädstrukturer

Applikationens huvudpart bestod av fyra separata fönster: *My Manuscript*, *Areas & Maps*, *Items & Characters* och *Related Documents* (se figur 1). Meningen från början var inte att ha flera fönster eftersom en fönsterhantering skulle ta en del av författarens uppmärksamhet från skrivandet. Enligt funktionsanalysen skulle dock varje begrepp behöva en egen vy med fält för specifika egenskaper och aktiviteter (utom föremål och karaktärer vars vyer kunde se likadana ut). Dessutom hade jag flera hierarkier som måste representeras

visuellt. Den viktigaste hierarkin var kanske den med moduler, situationer, sitparts, villkor, handlingar, händelser och konsekvenser (se figur 2). Därtill kom hierarkin för grupper och föremål/karaktärer samt hierarkin med dokumenttyper och dokument. Dessa tre hierarkier fick bli varsin trädstruktur i varsitt fönster. Areor, som inte ingick i någon hierarki, fick också ett eget fönster. Kanske skulle det senare visa sig att man ville sortera dessa i grupper också. Alla fönster byggdes upp med en trädstruktur till vänster (även area-fönstret för konsekvensens skull [Schneiderman 98]) där användaren kunde se och arbeta med begreppen på ett mer översiktligt sätt. Till höger fanns alltid en vy där användaren kunde ändra på egenskaper för det enskilda markerade objektet (se figur 2). På detta vis fanns en möjlighet att välja mellan två olika detaljeringsnivåer i arbetet.



Figur 2. Fönstret My Manuscript visar till vänster hierarkin med moduler, situationer och möjliga händelseförlopp. Till höger visas vyn för en vald situation.

Vyer

Objekt högt upp i hierarkin hade vyer med listor på sina ingående objekt på lägre nivåer. T.ex. hade alla dokumenttyper en lista över dokument av just dess typ i vyn. För att underlätta sökandet efter objekt ytterligare gjordes alla listor enkelt sorterbara för användaren [Schneiderman 98]. Andra likheter mellan vyerna var att de alla innehöll ett liknande fliksystem för associerade objekt. Genom att dubbelklicka på ett associerat objekt öppnades ett nytt fönster där det associerade objektets egenskaper omedelbart kunde redigeras. På så sätt minskades avstånden mellan objekt ur ett navigationsmässigt perspektiv vilket var en återkommande punkt i både funktionsanalysen och facklitteraturen [Kristof, Satran 95]. Vidare var jag noga med att hålla vyerna rena från överflödiga funktioner [Kristof, Satran 95] och att alla titlar till textfälten skulle vara tydliga att förstå [Schneiderman 98].

Utan att gå in för mycket på alla detaljer vill jag slutligen nämna att villkor, actions (handlingar och händelser) och konsekvenser omvandlades från tre vyer till ett fliksystem i samma vy för att tydligare visa att de hörde ihop. Denna samlingsvy bestod egentligen av en sitpart (ett händelseförlopp). Eftersom en sitpart var ett något abstrakt begrepp och författaren inte skulle behöva handskas med det i onödan lät jag sitpartens action automatiskt få namnge hela objektet. Dialogvyn var också speciell och utformades som en trädstruktur med repliker och ett tillhörande fliksystem med replikernas villkor, talare, innehåll och konsekvenser.

Presentationsdelen

Inom ramen för examensarbetet ville jag väldigt gärna hinna med en bit på manusverktygets presentationsdel. Det var kanske det viktigaste enskilda partiet av Mazermind och en av dess fördelar framför ett vanligt ordbehandlingsprogram. Under arbetets gång fick presentationsdelen smeknamnet Evaluatorn eftersom den tog det skrivna manuset, evaluerade det och genererade HTML-sidor länkade till varandra av det. Varje situation, area, föremål etc. omvandlades till en egen sida med olika attribut utskrivna och med länkar till associerade objekt. I manusverktygets skrivdel ordnades den textuella beskrivningen av situationer (dvs. berättelsen) hierarkiskt i moduler. I presentationsdelen kunde texten istället studeras interaktivt i en nätverksliknande (icke-linjär) struktur. Att presentationen skedde i form av HTML-sidor hade även fördelen att den som ville titta på och utvärdera manuset (t.ex. en projektledare, testare eller kund) inte skulle behöva lära sig eller skaffa ett nytt program. Istället skulle författaren kunna skicka alla sidor med e-post eller lägga upp dem lösenordsskyddat på en lämplig webbserver.

Designen av Evaluatorns gränssnitt gick tyvärr bortom gränsen för mitt examensarbete men det var tydligt att den ibland måste skilja sig från skrivdelens design. Vid t.ex. presentationen av en situation vore det kanske lämpligt att inte bara visa upp dess attribut utan även komplettera dem med en area-beskrivning.

Iterationer i arbetet

Arbetsförloppet gick som tidigare nämnts inte linjärt framåt. Begreppsvärlden hade växt fram i takt med förstudiens intervjuer och undersökning av olika medier. Terminologin skapades för att spegla den funna begreppsvärlden, men nya begrepp tillkom och gamla termers betydelse förändrades då datastrukturen och gränssnittet utvecklades. Processen att skapa en datastruktur var också allt annat än linjär då gränssnittets utseende och innehåll ställde nya krav på vilka egenskaper och tabeller strukturen skulle ha. Funktionsanalysen gjordes i samband med att datastrukturen byggdes och utgjorde en grund för gränssnittsdesignen. Prototypens utseende och funktionalitet ändrades i sin tur av den andra intervju jag hade med Magnus Seter och av de tester jag gjorde genom att mata in korta manusfragment i Mazermind. Inte ens de begrepp jag hittat i början var huggna i sten.

Denna rapport, skriven enligt en linjär och hierarkisk struktur precis som de manus den behandlar, skulle inte kunna beskriva alla turer i det iterativa arbete jag gjort på ett fullt läsbart sätt. Däremot kan den beskriva vilka speciella vägskalet jag besökt, vilka stigar som lett mig tillbaka så att jag varit

tvungen att ändra mitt vägval och vilka vägar jag helt enkelt valt att inte ta. Men först lite om min andra intervju med Magnus Seter.

Ny intervju med Magnus Seter

När större delen av arbetet med prototypen var färdig stämde jag åter möte med Magnus Seter. Jag hoppades att han skulle kunna ge mig återkoppling på prototypens trovärdighet och en känsla för om den var realistisk eller ej. Var idéerna och deras implementation på väg åt fel eller rätt håll? På ett mer konkret plan ville jag diskutera de krav och önskemål vi pratat om vid den första intervjun och i vilken utsträckning jag lyckats möta dem. Diskussionen och en gemensam genomgång av prototypen skulle kunna ge mig sådana förslag på förändringar och ny funktionalitet som jag inte hade tillgång till genom riktiga användartester. Inför intervjun skrev jag ner ett antal specifika frågor till Seter som uppstått under arbetet med prototypen. Dessa frågor finns i Bilaga 6.

Intervjun gav mycket input och Seter verkade försiktigt positiv till Mazermind. Det var dock rätt mycket att gå igenom på de få timmar intervjun varade och en del av frågorna fick besvaras med e-post vid ett senare tillfälle. Seter ansåg att verktyget ”helt klart är på rätt väg” men att komplexiteten skulle kunna skrämma de manusförfattare som skriver linjärt. Någon form av hjälp för användaren att komma in i programmet och dess funktioner skulle vara nödvändig.

Seter hade många förslag och idéer om specifika funktioner i prototypen, bl.a. tyckte han att termen ”document” inte passade för att beteckna en medie-referens. Namnvalet gav felaktiga associationer till t.ex. Word-dokument och rena filer. I prototypens konsekvensflik kunde författaren välja nästa situation ur en lista och här föreslog Seter att alternativet ”Quit” skulle finnas med för att beteckna att spelet tog slut.

Vad gällde fliken ”Action” var det inte helt klart vilket innehåll den förväntades ha och risken fanns att författaren skulle placera handlingar och händelser i konsekvens- eller villkorsfliken istället. Som det nu såg ut innehöll konsekvensfliken förändringar av enkla variabler (*Spelaren får tre nya liv*) medan actionfliken tog hand om spelarens valalternativ (*Spelaren kan öppna dörren*), dialogstarter (*Spelaren startar en dialog med draken*) och berättelsestyrda händelser med tillhörande mediereferenser (*Spela animation A15*). Seter tyckte att detta var otydligt och funderade först på ett separat ”Display documents”-fält för mediereferenser. Sedan föreslog han att fältet för handlingar och händelser skulle omvandlas till ett ”Action”- och ett ”Reaction”-fält, kanske på två nya flikar. Ett händelseförlopp skulle då kunna fördelas enligt följande:

- Condition: *Spelaren har nyckeln.*
- Action: *Spelaren väljer att försöka öppna kistan.*
- Reaction: *Animation då kistan öppnas.*
- Consequence: *Spelaren får tio guldpengar och går vidare till situation 4711.*

Under intervjun diskuterade vi en hel del om hur detaljerad funktionalitet Mazermind kunde ha innan manusskrivandet började likna programmering. Ett exempel som Seter tyckte ”nog är över gränsen” var att författaren kunde specificera föremåls ägare och innehåll i manusverktyget. Jag höll med om att specialknappar och -funktioner ofta var onödiga och att man kunde klara sig

långt på de redan befintliga, fria textbeskrivningarna av objekt. Seter menade att han personligen tilltalades av programmeringsstuket även om han inte trodde att majoriteten av alla manusförfattare skulle göra det. Denna åsikt födde tanken att manusverktyget kunde ha ett standardläge och ett avancerat läge med mer detaljerad funktionalitet för att tillfredsställa olika användare. Fördelen med att skriva sitt manus i ett avancerat läge skulle bl.a. vara att Evaluatorn kunde göra mer intelligenta beräkningar vid tester. Men jag tvekade ändå inför att skapa en intelligent Evaluator eftersom testaren av ett spel vill kunna ta sig fram fort och kanske fuska ibland för att få läsa/uppleva alla situationer.

Seter hade fler åsikter om Evaluatorn, bl.a. att det måste finnas funktionalitet för att se var i äventyret man befinner sig och vilka platser/situationer man har och inte har besökt. En enkel lösning som t.ex. en typisk webbmeny skulle underlätta användarens navigation och orientering mycket. Ytterligare en önskan om att externa beskrivningar skulle stå före interna beskrivningar i Evaluatorn uttalades. Denna önskan omformulerades sedan till att författaren i detalj skulle kunna välja vad de genererade sidorna visade upp. Då skulle ett annat problem Seter nämnde kunna undvikas, nämligen att kunder korrekturläser saker de inte ska bedöma (t.ex. programmeringsinstruktioner).

Seter ville att mediereferenser (documents) skulle ha ett fält för "vem" och ett fält för "vad" (t.ex. *Miranda* och *springer över en äng*). Då skulle det vara lättare att hitta alla dokument där en viss person eller ett visst föremål ingår. I enlighet med diskussionen om skrivandets närmande till programmering valde jag dock att vänta med denna uppdelning. Istället för att belasta författaren med ytterligare inmatningsfält kunde problemet kanske lösas genom intelligenta sökningar. Ett annat refuserat förslag handlade om en kompassros med vilken användaren enkelt genom ett klick skulle kunna välja ut- och ingångar till en area. Problemet var här att det skulle finnas ett oändligt antal möjliga passager och inga av dem kunde anses som standard.

Jag ställde frågan vad de långa och konstiga filbeteckningarna i ett manus var till för; en bild med en pillerburk kunde t.ex. heta "voa_7_1_U301b25" [Seter 02:b]. Seter svarade att de främst var till för att författaren snabbt skulle hitta och förstå vilket dokument det rörde sig om när han/hon talade med t.ex. kunder eller medarbetare. Seters kollegor blev ofta tokiga på det komplexa namngivningssystemet och han medgav att även han kunde tycka att det var jobbigt att arbeta med. Något som var extra irriterande var då man ville ha samma dokument på två ställen i manuset. Eftersom det låg på olika ställen fick det två olika namn och om dokumentet måste ändras fick det göras två gånger. Efter en diskussion kom vi fram till att det kunde vara uppskattat om man i manusverktyget kunde specificera en mall för hur dokument av en viss typ skulle namnges. Då kunde namnen genereras automatiskt och ville man flytta ett objekt skulle dess namn uppdateras av sig självt.

När vi talade vidare kom Seter med det eleganta förslaget att alla flikar med associerade objekt redan vid namnet skulle visa om de innehöll något så att författaren slapp bläddra i tomma flikar. Flikarna skulle också kunna visa upp hur många objekt de innehöll, t.ex. *Characters* (3). En annan viktig funktion som jag inte tänkt på var att författaren måste ha möjlighet att skriva ut sitt manus och få det i pappersform. Antingen på samma sätt som ett linjärt manuskript ser ut idag, eller som utskrifter av de genererade webbsidorna. Seter ville att alla sitparts/händelseförlopp skulle ha en titel, vilket de inte hade vid tiden för

intervjun. Jag löste detta (som tidigare nämnts) genom att låna titeln från förloppets ingående action. Vi var också överens om att det för funktioner och situationer som alltid var åtkomliga i ett spel borde finnas en speciell samlingsmodul, t.ex. en modul med namnet "Huvudfunktioner" eller "Allmänna funktioner".

Svåra beslut

Under arbetets gång stötte jag på en del problem som var större eller svårare än andra. Alla löstes inte, och en del beslut var heller inte optimala. Den första stora nöten jag hade att knäcka var att hitta gemensamma drag för de olika orienteringar inom berättande som fanns. Hur kunde jag skapa en modell som fungerade för areabaserade rollspel, dialogbaserade interaktiva teaterpjäser och tidsbaserade dataspel samtidigt? Var noder och länkar (eller pärlväven) en fungerande metafor? När jag till slut fann situationsbegreppet var jag länge tveksam till att det var svaret på mina frågor. Även terminologin gav mig på vissa punkter gråa hår. Ordet "document" visade sig t.ex. inte vara ett lyckat val. Ännu svårare var det med situationsbegreppet där alternativet "scen" länge spökade som en term betecknande ett tillstånd med vissa förutsättningar. Kraven på att orden skulle vara både konkreta och distinkta krockade gärna och ordet "situation" var i mitt tycke inte helt intuitivt.

Att bestämma vilka attribut och tabeller som skulle ingå i databasstrukturen var också knepigt. Varje koppling mellan tabellerna måste övervägas för sig, eftersom de direkt motsvarade associationer användaren skulle vilja göra. En abstrakt koppling i databasen, t.ex. att en sitpart (händelseförlopp) hade attributet *ItemID*, kunde rent konkret betyda att det i gränssnittet gick att koppla en replik till en karaktär eller ett föremål. De objekt som gav mig mest huvudbry var sådana som kunde kopplas till nästan vad som helst men ibland med olika betydelse: mediereferenser och föremål/karaktärer.

När gränssnittet mejslades fram blev det tydligt att vissa inmatningsfält och begrepp inte var så entydiga och självklara som de borde vara. Exempelvis var skillnaden mellan den interna och externa beskrivningen av saker inte särskilt uppenbar. Vissa objekt hade endast en intern beskrivning och även när möjligheten till att skriva en extern beskrivning fanns var det inte säkert att den alltid skulle utnyttjas. Detta orsakade i sin tur följdfrågan om Evaluatorn alltid borde visa upp alla beskrivningarna eller inte. En annan oklarhet var skillnaden mellan villkor, handling/händelse och konsekvens: Händelsen *spelaren får nyckeln i sin ägo* skulle kunna vara konsekvensen av en handling men också villkoret för en annan handling. Ytterligare en nivå ner i hierarkin stötte jag på problemet att en action både kunde vara en handling eller flera händelser. Skulle Seters förslag på ett action- och ett reactionfält göra arbetet tydligare eller krångligare?

Andra spørsmål var hur jag skulle representera funktioner som alltid kunde väljas och händelser som kunde inträffa lite när som helst, s.k. slumphändelser. Även Evaluatorns nivå av intelligens i förhållande till hur mycket och detaljerad information författaren måste mata in var ett ämne som oroade mig. Hur noga och reglerat ville användaren egentligen kunna klicka sig igenom sitt äventyr? En sista olöst fråga var hur låg upplösning gränssnittet skulle stödja för att den stora mängden fönster och inmatningsfält skulle få plats på skärmen. Prototypen fick egentligen inte plats på 800×600 pixlar men samtidigt ville jag inte ställa för höga krav på användarens utrustning.

Ändringar

I projektet var jag ibland tvungen att gå tillbaka och göra ändringar för att få ett bättre slutresultat. Detta stycke beskriver några av de viktigaste modifikationerna och varför de gjordes.

Intervjun med Mathias Gullbrandson visade att karaktärerna i ett lajv ofta hör till en typ av individer med liknande egenskaper och attribut. Detta fick mig att inse något som jag inte tänkt på tidigare: Karaktärer (och föremål) måste kompletteras med ett nytt gruppbegrepp. Det dröjde heller inte länge efter detta förrän jag förstod att varje föremål och karaktär skulle kunna höras till flera olika grupper och att varje grupp måste kunna ordnas i en hierarki med andra grupper. Förutom lajv stödde både rollspel och dataspel dessa tankar.

I ett tidigt skede av arbetet skapade jag en terminologi att kommunicera med. Ju längre tiden gick desto fler nya termer behövdes och några av de refuserade förslagen återupptod med ny betydelse. Ett av alternativen till ordet "händelse", nämligen "händelseförlopp", visade sig vara till användning när begreppet sitpart lanserades i databasstrukturen. En sitpart bestod inte enbart av en händelse, utan av ett sammansatt förlopp av ett villkor, en action och en konsekvens. För varje situation fanns flera möjliga sitparts eller, bättre uttryckt, flera möjliga händelseförlopp. Även ordet "action" förändrade sin mening från att i terminologin bara beteckna en handling till att beteckna både handlingar och händelser i databasen.

Från allra första början fanns överhuvudtaget inte begreppet action eller handling/händelse med i databasstrukturen. Istället fanns en "situationsrad" som senare blev en sitpart och som pekade på ett föregående villkor och en efterföljande konsekvens. För att göra det hela mer lättöverskådligt och för att standardisera funktionerna för att hämta data skapades så småningom en action-tabell jämlik med tabellerna för villkor och konsekvenser. Då förvandlades sitpart till ett abstrakt samlingsobjekt som kunde göras osynligt för användaren. En annan förändring i databasstrukturen var att den länge var mer detaljerad än dess slutliga version. Två tabeller som ströks var *Condition-Part* och *ConsequencePart*. Dessa tabeller höll reda på delvillkor och delkonsekvenser och deras inbördes förhållande, t.ex. kunde ett villkor se ut enligt följande:

IF (Spelarens erfarenhetspoäng > 20) AND (ÄgarenAv(Den magiska lampan) = Spelaren) OR (Spelaren tappar putsduken på den magiska lampan) THEN...

Eftersom en sådan struktur på villkor (och konsekvenser) mer liknade programmering än vanligt manusförfattande valde jag att helt ersätta den med fritextbeskrivningar. Jag trodde inte att vinsterna i t.ex. en smartare Evaluator skulle uppväga nackdelarna med ett så tillkrånglat och för många främmande skrivsätt.

I gränssnittet gjordes flera små förändringar kontinuerligt; ikoner och texter byttes ut, nya fält kom till och flyttades runt och trädvyerna ändrade utseende. Från början skapades ett modulfönster med moduler och situationer ordnade i en hierarki och ett situationsfönster med situationer, sitparts, villkor, actions och konsekvenser ordnade i en annan hierarki. Det var bekvämt att kunna se situationer på två olika sätt i två olika trädvyer, men det innebar också extra arbete med fem fönster istället för fyra. För att minska fönsterhanteringsarbetet för författaren slog jag till sist ihop modul- och situationsfönstret (och deras respektive trädvyer) till ett fönster med namnet *My Manuscript*. Eftersom den

resultaterande hierarkin nu blev mycket djup och litteraturen förespråkade grunda hierarkier [Kristof, Satran 95] valde jag att plocka bort ikonerna för villkor, actions och konsekvenser ur trädvyn. De tre objekten låg ju redan i samma vy, fast på olika flikar.

Refuserade förslag

En del av de idéer jag hade visade sig inte vara bra, eller ens genomförbara. Men lika viktigt för mitt arbete som det var att upptäcka vad som var möjligt var det att hitta gränserna för vad som inte var möjligt (eller lämpligt). Nedanstående punktlista visar ett axplock av de tillfällen då jag fick tänka om:

- Alldeles i början trodde jag att alla händelser kunde definieras av var och när de inträffade. I och med att en interaktiv berättelse gärna har flera stickspår med alternativa situationer för samma rum och tidpunkt visade det sig att de fyra dimensioner vi är vana vid från den verkliga världen inte skulle räcka till. Jag var tvungen att hitta ett annat sätt att modellera den interaktiva världen.
- Länge önskade jag mig en tidsaxel, kanske med flera olika händelsestråkar, där den interaktiva berättelsen skulle gå att visualisera med alla händelser i ordning. Tyvärr visade det sig att interaktiva berättelser ofta inte är så ordnade att de går att visualisera, än mer sällan är de ordnade efter någon form av tid (verklig eller virtuell).
- I funderingarna kring hur gränssnittet skulle se ut kom idén från förundersökningen upp om att författaren skulle kunna projicera en del, men kanske inte alla, aspekter av berättelsen i en vy. Svårigheten att ordna berättelser och deras händelsestråkar visuellt hade redan visat sig och jag försökte hitta ett nytt sätt att visa upp äventyr. Ett av alla förslag som kom upp var att användaren skulle kunna se vyer med två valfritt kombinerade objekt: En vy med karaktärer och karaktärer skulle visa upp alla dialoger, en vy med karaktärer och föremål skulle visa upp ägarförhållanden osv. Förslaget i sin helhet refuserades snart men levde delvis kvar i tanken på en kartvy där alla areor och deras grannrelationer kunde ritas upp.
- Jag hade en idé om att situationer skulle kunna sorteras i fler än ett modulsystem. Ett äventyr skulle exempelvis kunna var uppdelat i ett modulsystem för kapitel, ett för areor och ett för tidsepoker. Svårigheten låg i att visualisera flera hierarkier med samma objekt i gränssnittet. Dessutom var det tveksamt om författare verkligen efterfrågade sådan funktionalitet.
- Mer än en gång dök tanken upp att det till kopplingar mellan olika areor skulle gå att sätta villkor. Då skulle Evaluatorn snabbt kunna avgöra vilka ingångar och utgångar som var tillgängliga från den area testspelaren befann sig vid. Invändningen mot denna tanke var dels att det skulle ge författaren merarbete, dels att det skulle tvinga honom/henne till ett area-baserat förhållningssätt. Jag höll fast vid att villkor var något som endast skulle kunna sättas för handlingar och händelser i situationer.
- Föremål kan vara associerade med specifika händelser, t.ex. en magisk lampa och händelsen att en ande dyker upp ur dess innandöme. Det tidiga förslaget att varje föremål skulle ha en mängd händelser kopplade till sig var kanske intuitivt ur ett perspektiv men det krånglade också till hur datastrukturen för händelser skulle se ut. Istället skapades den mer allmänna

situationsmodellen där olika objekt, däribland föremål, kunde kopplas till en situation.

- Mazermind gav användaren möjlighet att skapa kopplingar av skilda slag mellan olika objekt. Ett föremål kunde exempelvis vara kopplat till flera dokument såsom en bild och ett ljud. Då skulle det vara praktiskt om det gick att göra en beskrivning av kopplingen, t.ex. *detta dokument motsvarar ljudet då föremålet används* och *detta dokument visar föremålet när det är öppnat*. Idén refuserades eftersom det kunde bli tungt för manusförfattaren att inte bara beskriva vanliga objekt i berättelsen utan även mer eller mindre abstrakta kopplingar mellan objekten. Verket skulle hållas på ett så konkret och objektorienterat plan som möjligt och denna idé var ett gränssfall. Det reella användningsområdet av denna funktionalitet var osäker och jag beslöt att låta idén vila tills användartester kunde visa om den var intressant eller ej.

Medvetna begränsningar i prototypen

Tiden för byggandet av Mazermind var knapp; drygt en och en halv månad hade jag på mig att skapa en prototyp som kunde visa de mest vitala idéerna om ett manusverktyg. Det var därför helt naturligt att vissa tankar och funktioner måste prioriteras bort för att ge tid åt andra. En del av de bortprioriterade idéerna var mindre viktiga för förståelsen av verktyget och en del av dem tog för lång tid i anspråk för att hinnas med under examensarbetet. Följande stycke sammanfattar de begränsningar jag valde att göra och den funktionalitet som fanns med i planerna men vars realisering fick läggas på framtiden.

Mazermind gjordes i en engelsk version för att passa en så stor målgrupp som möjligt. När funktionaliteten började bli färdig ville jag dock gärna översätta applikationen så att användaren t.ex. kunde välja svenska som språk istället. Ytterligare en inställning som kunde vara praktisk att ha med var möjligheten att välja hur nya fönster skulle öppnas i verktyget. Om användaren dubbelklickade på exempelvis ett föremål i en lista, ville han/hon då att föremålet skulle visas i ett nytt, i samma eller ett annat öppnat fönster? Även möjligheten att välja bort vissa objekt ur sitt manus kunde vara behändig. Ibland ville författaren kanske inte beskriva areor eller så bestod äventyret bara av dialog och då skulle vissa objekt vara onödiga att visa upp i gränssnittet.

Som tidigare nämnts prioriterade jag bort att skapa ett tidsbegrepp i programmet. Tills vidare fick tidsangivelser läggas in i den externa och interna beskrivningen av situationer och situationernas inbördes förhållande fick anges genom villkor och konsekvenser för händelser. Vid en utveckling av prototypen skulle det ändå vara viktigt att undersöka hur ett tidsbegrepp skulle kunna se ut, om det var nödvändigt och hur det skulle påverka gränssnittet mot användaren.

Även om jag hann slutföra utformningen av ett fungerande fönster för areor hade jag inte ens börjat tänka på ett gränssnitt för att visa deras inbördes relationer. Att ha en kartvy var inte nödvändigt för att författaren skulle kunna hantera grannskap mellan areor men det skulle troligtvis underlätta hans/hennes arbete om möjligheten till visuell återkoppling fanns. Frågan om det trots allt inte gick att skapa en begränsad visuell översikt över situationerna och deras kopplingar var heller inte färdigbearbetad.

Mazermind blev – trots mitt krav på en intuitiv applikation – ett komplext verktyg i jämförelse med t.ex. vanliga ordbehandlingsprogram. För att hjälpa den tänkta användaren skulle det behövas stöd i form av hjälptexter för att förklara begrepp eller inmatningsfält och guider (eng. *wizards*) som steg för steg kunde leda användaren rätt. En minneshjälp i form av ett digitalt block i vilket noter om objekt eller handlingen kunde antecknas så att de inte glömdes bort under skrivandets gång skulle också underlätta det kognitiva arbetet för författaren. Ingen av dessa funktioner fanns med i prototypen men de skulle naturligtvis vara mycket viktiga att få med i en färdig produkt. Även typiska ordbehandlingsfunktioner som att ändra typsnitt, söka och ersätta text planerades.

Prototypen gjordes så att man skulle kunna se vilka egenskaper objekt tilldelats och vilka funktioner som kunde utföras på dem. En del av funktionerna implementerades på riktigt så att t.ex. egenskaper gick att ändra och objekt kunde namnges. Men som i alla prototyper var det också många funktioner som inte fungerade eller fanns med i Mazermind. De viktigaste av dessa var funktioner för att skapa nya objekt, att ta bort gamla objekt och att flytta objekt med direktmanipulation i deras respektive hierarkier. En annan önskvärd funktionalitet var att kunna exportera information och objekt från ett projekt till ett annat.

I den andra intervjun med Magnus Seter dök tanken upp om att skapa ett avancerat läge i manusverktyget för de användare som föredrog ett mer detaljerat förhållningssätt i sitt arbete. Förslaget kom upp i slutet av fasen för prototypbygget och hann därför aldrig realiseras. En av de extra funktioner det avancerade läget skulle kunna ha var möjligheten att mer noggrant specificera villkor och konsekvenser. (Se diskussionen om delvillkor under rubriken *Ändringar* ovan.) En annan intressant utökning vore om författaren kunde skapa variabler för olika egenskaper och bestämma variabelernas (gräns-) värden. Då kunde Evaluatorn hålla reda på egenskaper hos olika objekt och varna om variablerna antog felaktiga värden.

En fundering jag hade var att dokumenttyper inte bara skulle samla dokument under ett visst namn (t.ex. *Animationer*) utan även kunde definiera vilka och hur många objekt som kopplades till dokument av respektive typ. Exempelvis skulle en teatermanusförfattare kanske vilja skapa en egen dokumenttyp för gester. Varje dokument av typen ”gest” skulle då referera till en handling på scenen och alltid vara kopplad till en (1) karaktär i manuset. Om författaren hade möjlighet att i förväg definiera dokumenttyper skulle han/hon i fortsättningen för varje mediereferens ha ett färdigt formulär att fylla i. I exemplet ovan skulle det finnas ett (1) inmatningsfält för att ange vilken karaktär som skulle göra gesten på scen. Dessutom skulle applikationen kunna hålla reda på vilka saker som behövde fyllas i och det skulle bli lättare att söka efter dokument kopplade till ett visst objekt (t.ex. efter gester hörande till en viss roll).

Ett verktyg som kunde generera namn på dokument var något som kunde bespara författaren tråkigt och icke-kreativt arbete. Det var funktionalitet som borde finnas i Mazermind och jag hade funderingar kring hur man kunde tänkas vilja döpa sina dokument och vad som skulle hända om namnmallen ändrades mitt i ett projekt. Längre än till funderingar kom jag inte men i en fortsättning på examensarbetet skulle namngenerering vara ett prioriterat område.

Evaluators utseende och funktionalitet kom inte särskilt långt utan var snarare på skisstadiet när jag slutade bygga prototypen. Utöver att användarens orientering, navigering och översikt behövde underlättas skulle det vara praktiskt om författaren snabbt kunde ta sig från Evaluatorn till verktygets skrivdel för att t.ex. rätta stavfel eller sakfel som han/hon hittat. Även en journal som kunde visa vilka handlingar och händelser som inträffat skulle här vara praktisk.

Action-fältet (eller det alternativa action- och reaction-fältet) var inte helt tydligt och det behövdes en bättre märkning för att användaren skulle kunna se vad som var en handling och vad som var en händelse (kanske också en dialogstart). Dessutom måste det finnas restriktioner för att inte användaren skulle kunna lägga till mer än en handling per händelseförlopp. I t.ex. ett data-spel kan en spelare inte göra något utan att direkt hamna i en ny situation. Däremot kan flera händelser avlösa varandra utan att spelaren har möjlighet att påverka situationen. Även restriktioner för villkor måste finnas med för att inte två konkurrerande händelseförlopp med olika konsekvenser ska kunna inträffa samtidigt. Någon form av grafiskt gränssnitt (exempelvis med spår för olika dokumenttyper) för att ordna och visa hur händelser och valalternativ ligger i förhållande till varandra inom ett händelseförlopp kunde också vara behändigt.

Den sista punkten jag tar upp här får exemplifiera de val jag gjorde i designen av datastrukturen och vilka begränsningar de kunde medföra. Jag beslöt bl.a. att en situation högst kunde kopplas till en area. Situationer utan area var möjliga att tänka sig, eftersom det innebar att de kunde inträffa var som helst. Situationer kopplade till en area var än mer naturliga. Det var svårt att tänka sig en situation som inträffar på två platser, men det var möjligt att författare skulle vilja återanvända en situation från en likartad plats eller att samma sak skulle kunna hända på två olika ställen. Detta var alltså inte möjligt att göra i manusverktyget, däremot var det möjligt att kopiera en situation, ge den ett nytt namn och placera den i en ny area.

SLUTSATSER OCH DISKUSSION

Nog finns det mål och mening i vår färd – men det är vägen som är mödan värd.

Med detta citat av Karin Boye [Boye 42] skulle jag vilja säga att även om examensarbetets resultat – en fungerande prototyp – i sig var intressant, så var det vägen dit som hade störst betydelse. Det var en väg som jag fått privilegiet att bryta och en väg som förhoppningsvis skulle kunna locka nya upptäcktsresande. Alla svårigheter, ändrade planer och beslutsalternativ jag erfor under arbetets gång hade betydelse för vilka slutsatser som kunde dras. Summan av upplevelsorna och deras slutsatser var det egentliga resultatet av min expedition – inte var jag till sist hamnade.

Behövs ett manusverktyg?

En av de första frågorna jag ställdes inför var om det fanns ett behov av manusverktyg för interaktiva produktioner. Om så inte vore fallet skulle det vara meningslöst att gå vidare utan att helt ändra examensarbetets inriktning först. Frågan var kritisk och jag anlidade flera källor innan jag lät mig övertygas om att ett behov av ett manusverktyg faktiskt förelåg.

Litteraturen

Litteraturen identifierade det linjära skrivandet som ett problem i sammanhang där antalet möjliga händelseförlopp i en berättelse är fler än ett. Likaså bekräftade litteraturen att det är problem med att läsa icke-linjära berättelser skrivna i linjära media [Bolter 91]. Detta tolkade jag som att manus för interaktiva produktioner kräver en teknik eller miljö där de kan skrivas och läsas icke-linjärt. Vidare fann jag i litteraturen stöd för att författare i multimedia-projekt måste (ha verktyg för att) kunna bygga modeller av sina manuskript [Stansberry 97].

Intervjupersonerna

Magnus Seter var inte nöjd med de ordbehandlare och alternativa program som fanns tillgängliga för manusförfattare för dataspel. Han uttryckte en önskan om bättre verktyg för interaktivt manusskrivande. I synnerhet efterlyste Seter möjligheten att skriva interaktiv dialog på ett smidigare sätt och helst ville han ha ett verktyg för att skapa prototyper av manus innan de gick till produktion.

Brådhe upplevde att det saknades bra skrivstöd för interaktiva medier. Det fanns inga traditioner eller någon utvecklad standard för hur ett manus för interaktiva produktioner (film, teater eller dataspel) skulle se ut och detta såg han som ett problem. Brådhe hade själv funderat på att skapa ett lämpligt verktyg för ändamålet.

Mathias Gullbrandson nämnde inte något specifikt behov av ett manusverktyg, däremot en fundering kring stöd för planeringen och skapandet av roll-karaktärer.

Rajkas rapport

Rajka beskrev i sin rapport att det för flera olika användarkategorier behövdes generella verktyg för datorbaserad aktionsplanering. I undersökningen inför projektet Digitala Teaterrum [Rajka 96] hade han letat efter en modell för kontroll och planering av händelser, oavsett media och datatyp, och efter en

applikation som kunde presentera kompositionsarbetet. Rapporten visade att det varken fanns en sådan modell eller applikation och att man skulle behöva konstruera dessa själv. Den avslutades istället med ett förslag på hur den förra av dessa skulle kunna se ut (se Bilaga 2).

Finns det manusverktyg?

Trots idogt sökande fann jag inget verktyg utformat för just de arbetsuppgifter och behov som karaktäriserar manusförfattare för interaktiva produktioner. Däremot fann jag en hel del hjälpmedel för liknande aktiviteter, bl.a. verktyg för att skriva (utkast till) linjära manus, t.ex. Dramatica Pro [Screenplay 03], och program för att skapa färdiga interaktiva produktioner som t.ex. Aurora [BioWare 03] och Adventure Game Studio [Jones 03]. Inte heller tydde något på att andra skulle ha hittat det verktyg jag sökte:

There simply isn't anything out there that lets you create interactive storytelling. [Crawford 03]

Är det möjligt att skapa ett manusverktyg?

Innan jag påbörjade konstruktionen av prototypen var jag inte helt säker på att det ens skulle vara möjligt att skapa ett manusverktyg. Rex Brådhe hade försökt men gett upp p.g.a. svårigheterna och det faktum att det inte fanns något verktyg trots att behovet av ett sådant förekom kunde betyda att det inte var ett genomförbart projekt. Med byggandet av Mazermind visade jag att det rent teoretiskt var möjligt att konstruera en generell modell av interaktiva berättelser och att skapa ett datorbaserat verktyg för att presentera och hantera denna modell. Huruvida det som fungerade i teorin även skulle vara praktiskt användbart återstod dock att se.

Hur kan ett manusverktyg se ut?

Med utgångspunkt från mina erfarenheter av arbetet med prototypen presenterar jag i detta stycke ett antal förslag på hur ett manusverktyg bör vara beskaffat. Jag vill betona att förslagen inte är färdiga lösningar utan snarare starka rekommendationer som stöds av observationer gjorda under studiet av olika interaktiva ämnesområden.

Ett datorbaserat program

Examensarbetet granskade inte några särskilda metoder för att skriva manus för interaktiva produktioner och inbegrep heller inga tester med icke-digitala medier. Trots detta vågar jag med stöd av mina skrivna källor [Bolter 91] och [Stansberry 97] påstå att manusverktyg kan dra en oslagbar mängd fördelar av att baseras på datorteknik. Datorn är ett medium där icke-linjära och flerdimensionella strukturer kan uttryckas på ett naturligt – i det här fallet flerdimensionellt – sätt och där komplexa nätverk och hierarkier enkelt låter sig organiseras och kontrolleras.

När man talar om ett datorbaserat verktyg förs tankarna direkt till någon form av programvara, och det var (en prototyp av) ett sådant program som konstruerades under examensarbetet. Ett intressant alternativ till detta hade varit att jag istället utvecklat ett programmeringsspråk i stil med det Text Adventure Development System [Roberts 03] som beskrevs i förundersökningen. Man skulle också kunna tänka sig att jag skulle ha skapat en

vokabulär baserad på exempelvis XML (Extensible Markup Language) med taggar som markerar vad manustextens delar är av för typ.

Utifrån mina intervjuer kom jag fram till att manusförfattare för interaktiva produktioner inte nödvändigtvis har avancerade kunskaper om datormediet. Ett verktyg för skrivandet av manus borde därför vara så intuitivt och lätt-tillgängligt som möjligt. Denna slutsats var inte förenlig med idén att manusförfattare skulle lära sig programmera i ett nytt språk. Däremot fanns inga hinder för att speciellt anpassade gränssnitt byggda ovanpå programmeringsspråk skulle kunna utgöra utmärkta miljöer för författare att skriva manus i. Mazermind tog på sätt och vis tillvara på denna idé genom möjligheten att med ett klick exportera hela manuset till XML.

Min rekommendation är att ett manusverktyg bör ha formen av ett datorbaserat program vars gränssnitt är enkelt att använda och som bygger på en underliggande och kanske mer avancerad struktur.

Lämplig modell

Den struktur som ska ligga till grund för ett manusverktyg ska både spegla den begreppsvärld som återfinns i interaktiva berättelser och de uppgifter manusförfattaren har i sin yrkesroll.

I stycket *En objektorienterad begreppsvärld* diskuteras en modell över vilka byggstenar som ingår i de flesta interaktiva berättelser. Denna modell är ett subjektivt urval men många av dess element återkommer i andra, liknande uppställningar. En del skulle kunna tycka att modellen inte är tillräckligt detaljerad (t.ex. finns inget begrepp för att benämna passager mellan areor) eller att den begränsar berättandet (en area kan inte vara en karaktär eller tvärtom). Samtidigt skulle en tillräckligt detaljerad modell för alla synsätt och berättandeformer vara omöjlig att skapa och den skulle snarare vara i vägen för dem som inte vill ha en bestämd modell skriven på näsan. Genom att istället ha en datastruktur som tillåter slutanvändaren att definiera egna begrepp (t.ex. *akter*) och grupper av objekt (t.ex. *verktyg* eller *animationer*) går det att uppnå en högre och nödvändig grad av flexibilitet.

En annan punkt i modellen som också kan ifrågasättas är dess mest centrala begrepp. Vi erinrar oss att *en situation är ett tillfälle då vissa förutsättningar gäller och då det finns ett eller flera möjliga valalternativ och händelseförlopp som kan ta spelaren till en ny situation*. Det är ett abstrakt och konstruerat koncept och därför inte helt självklart. Situationsbegreppet knyter ihop en rad andra vitala begrepp som villkor, handling, händelse och konsekvens och beskriver den för interaktiva berättelser så karakteristiska kausaliteten. Oavsett om just det situationsbegrepp jag utvecklade är optimalt eller ej behövs det objekt som kan agera motor för att berättelsens handling ska skrida framåt. Varje modell av en interaktiv begreppsvärld bör innehålla begrepp med funktionalitet liknande den hos det i rapporten presenterade situationsförslaget.

Av skäl som står förklarade i stycket *Fokus och valda begränsningar* valde jag att inte skapa en modell inriktad på de dramaturgiska skeendena i berättelser. En sådan modell hade gett ett helt annorlunda resultat och skulle i sig kunna vara ett eget examensarbete. Istället skapade jag en objektorienterad och relativt konkret modell som var lätt att översätta till datastrukturer.

Lämplig datastruktur och teknik

Den datastruktur som ska ligga till grund för ett manusskrivarprogram bör motsvara den uppsättning av begrepp som modellerar interaktiva berättelser. Den bör eventuellt också spegla de handlingar en manusförfattare kan tänkas vilja utföra på objekt hörande till begreppsvärlden. Eftersom den utvecklade modellen troligtvis är objektorienterad är det lämpligt om även den underliggande tekniken är det. Stöd i form av en databas rekommenderas här starkt.

Tekniken bör rent allmänt vara lätt att hantera (i form av t.ex. installationsarbete för slutanvändaren) och ska helst fungera på flera plattformar. Den får gärna redan vara spridd så att de flesta användare slipper köpa nya extra-program för att manusverktyget ska fungera. Det kan också vara lämpligt om tekniken tillåter andra personer att granska manusförfattarens arbete utan att för den skull behöva specifik programvara för detta. De teknikval som gjordes i detta examensarbete är ett exempel på hur majoriteten av dessa kriterier kan uppfyllas.

Lämplig funktionalitet

Redan i arbetets förundersökning kom jag fram till en lämplig förteckning över vilken funktionalitet ett manusverktyg borde ha. Denna förteckning gäller fortfarande och du finner den i dess helhet under stycket *Sammanfattande funktionalitetsanalys*.

Utöver de punkter som där tas upp vill jag även betona vikten av att bibehålla enkelheten i skrivandet och att skapa ett manusverktyg som är så intuitivt som möjligt. Ett verktyg för berättande bör anpassa sig efter författarens skrivsätt vad gäller orientering (plats-, dialog-, tids-) och perspektiv (hierarkiskt, nätverksbaserat). Dessutom är det önskvärt med stor frihet för användaren att själv välja hur manustexten ska delas upp och vilken detaljnivå den ska ha.

Lämpligt gränssnitt

Då gränssnittet hos Mazermind inte hunnit genomgå några regelrätta användartester avstår jag från att spekulera i hur ett lämpligt gränssnitt hos ett manusverktyg bör se ut. Flera alternativ till hur utformningen ska gå till är möjliga och det är svårt att i förväg avgöra vilket som skulle vara mest praktiskt. Denna rapport har tidigare beskrivit ett aktivitetsorienterat gränssnitt med en verktygslåda och ett bibliotek som alternativ till det i prototypen använda relationsorienterade gränssnittet med objekt i hierarkiska vyer. Det finns självklart fler varianter som väntar på att utforskas.

Det bör kanske påpekas att av de allra minsta designbesluten som gjordes för Mazermind är inte alla redovisade i rapporten eftersom risken för att de kommer att ändras vid vidare kontakt med målgruppen är överhängande.

Motsvarar konceptet behoven?

Efter förrättat värv och med resultatet i handen kan man fråga sig om konceptet möter de behov det från början skulle tillfredsställa eller om det måste till en omdefiniering av dessa. Att konstatera att "Det bidde en rock" när det var ett par byxor som efterfrågades är kanske inte så lustigt, men en utvärdering är ändå nödvändig för att se om arbetet kunde ha gjorts på ett bättre sätt. Samtidigt bör man hålla i minnet att den utvecklade prototypen i

sig inte är en slutgiltig lösning utan endast ett första steg i (förhoppningsvis) rätt riktning.

Olika manustyper

I förundersökningen studerade jag flera områden för vilka det skrivs manus: rollspel, lajv, multimedial teater, dataspel m.fl. Varje område hade sina specifika behov och även om jag hoppades kunna möta alla dessa med ett och samma verktyg var det inte särskilt troligt att det skulle vara möjligt. I några fall visade det sig också att det inte fanns något behov eller att de krav som ställdes redan var uppfyllda av existerande programvara. Vi ska titta närmare på varje område:

- Rollspelsmanus kräver att författaren i detalj kan specificera och räkna ut olika egenskaper och värden. Dessa krav uppfylls av redan existerande programvara men inte av min prototyp. Samtidigt saknas det verktyg för att skriva rollspelsmanus där själva berättandet (och inte räkneregler eller variabelsättning) står i fokus. Mazermind skulle med en del utökad funktionalitet kunna utgöra ett sådant verktyg.
- Lajv har en struktur där interaktiviteten inte ligger lika mycket i dess manus som det ligger i hur dess deltagare väljer att utföra handlingen i manuset. Ändå krävs det av författaren att han/hon måste förutse när berättelsen kan ta annorlunda vändningar och hur man ska stävja eller underlätta dessa vändningar. I lajv finns i likhet med rollspel ett behov av att kunna beskriva karaktärer, föremål och platser. Det är dock inte lika detaljerat vilket gör att Mazermind här har större potential att fungera i dess nuvarande form. Frågan kvarstår ändå om inte ett manusverktyg för lajv borde ha en större fokusering på utvecklingen av karaktärer eftersom det är karaktärerna som för berättelsen framåt.
- För traditionell teater och film finns det en mängd datorbaserade verktyg för skrivandet av manus. Dessa verktyg kan i en viss utsträckning nyttjas av de former av teater/film som inte är linjära och där publiken har en mer aktiv roll i berättandet. Den i examensarbetet utvecklade prototypen visar sig dock ha en god förmåga att stödja just det som de traditionella verktygen är sämre på; interaktivt berättande och möjligheten att skriva instruktioner för valfria medieformer.
- Dataspeisförfattare saknar helt särskilda verktyg att skriva sina manus i. Deras behov av ett hjälpmedel har studerats ingående och även styrt stora delar av detta arbete. Därför torde Mazermind väl uppfylla kraven från denna typ av författare.
- För att skriva och skapa interaktiv fiktion eller textbaserade äventyr finns det redan både ett antal språk och program till hands. För denna form av litteratur/spel är det heller inte lika självklart att det ska skrivas ett manus eller ens en synopsis. I de fall man ändå skulle vilja skriva manus för ett textbaserat äventyr skulle Mazermind troligtvis fungera väl. Det är även möjligt att skapa en form av interaktiv fiktion direkt i prototypen (eftersom den genererar webbsidor) men det är inte ett medvetet planerat användningsområde.

Utöver diskussionen kring olika berättandeformer är det intressant att veta hur väl prototypen stöder olika orientering och perspektiv i skrivandet. Vissa

manus, t.ex. rollspelsmanus, utgår ifrån vilken plats händelserna utspelar sig i. Andra manus, t.ex. teatermanus, kan vara nästan helt skrivna som dialog. Ännu fler varianter är manus som bygger på vilken karaktär spelaren för tillfället är och manus där tidpunkten är det mest väsentliga. Alla dessa orienteringar (area-, dialog-, karaktärs- och tids-) stöds av Mazermind genom den fritt formbara modulhierarkin och genom det situationsbaserade berättandet där flera sorters objekt kan associeras till en situation. Även olika slags interaktiva strukturer stöds av prototypen vilket gör att författaren kan välja om han/hon vill ha t.ex. ett hierarkiskt eller ett nätverksliknande perspektiv på sitt manus.

Naturligtvis kan det ändå finnas strukturer och orienteringar som jag inte tänkt på och som inte fungerar lika smidigt i Mazermins system.

Olika tekniska system

Prototypen av manusverktyget uppfyllde alla krav på att tekniken skulle vara enkel, lättillgänglig och billig utom i ett avseende. Den fungerade endast i Windows-miljö vilket måste sägas vara en stor begränsning för alla manusförfattare med vana av Mac- eller Unixdatorer.

Användarorientering

Utvecklingen av manusverktyget var inte särskilt användarorienterad, även om applikationens slutanvändare var med i processen och hade synpunkter som påverkade arbetet i hög grad. Eftersom inget liknande verktyg någonsin gjorts tidigare var det i detta skede mer intressant att se vilken funktionalitet som var möjlig, vilken begreppsvärld som var lämplig och om det fanns något allmängiltigt och fungerande perspektiv på interaktivt berättande. Ändå är det berättigt att framföra en viss kritik mot att användarorienteringen inte kom tillräckligt tidigt och inte heller var så framträdande i arbetsprocessen. Förvisso har jag precis inventerat hur väl prototypen motsvarade behovet från målgrupper med olika berättartraditioner men samtidigt borde kartläggningen av olika användare och deras behov redan ha gjorts i förundersökningen. Som det nu var fick jag aldrig ett riktigt grepp om exakt vilka krav jag i första hand borde uppmärksamma.

Å andra sidan var utvecklingen av Mazermind i hög grad funktionalitetsorienterad och grundade sig på de fakta jag fick reda på från mina intervju-personer. Även det faktum att jag stämde av ett nytt möte med Magnus Seter för att gå igenom implementationen av de funktioner som vi diskuterat tidigare pekar på att jag följde ett iterativt arbetssätt om än i mycket liten omfattning. I rapporten har jag skrivit om intervjuer och även redovisat listor med frågor skrivna för dessa. Trots detta språkbruk har det inte handlat om typiska situationer med en intervjuare och en intervjuad person utan snarare om diskussioner mellan jämbördiga och intresserade människor enligt principen om partnerskap [Beyer, Holtzblatt 97]. Jag har också följt en mängd generella riktlinjer [Gould m.fl. 97] för att öka systemets användbarhet genom att:

- Definiera målgruppens problem.
- Identifiera slutanvändarnas arbetsuppgifter.
- Sätta mig in i målgruppens (tekniska) kunskap och förmåga.
- Skissa på användarscenario.

- Själv testa författarrollen.
- Skriva funktionalitetsanalyser.
- Konstruera en prototyp.
- Kontinuerligt göra ändringar vid behov.

Om Mazermind går in i ett nästa utvecklingssteg från prototyp till produkt kommer ett användarorienterat arbetssätt vara än mer viktigt för att ett lyckat resultat ska uppnås. Då är Goulds fyra principer [Gould m.fl. 97] om iterativ och integrerad design, tidiga och kontinuerliga tester/revideringar samt fokus på användaren en utmärkt plattform att utgå ifrån.

Framtid

Förutsättningarna för att skapa en kommersiellt gångbar produkt av Mazermind är kanske inte så stora eftersom dess tänkta målgrupp är relativt begränsad. Samtidigt bör man inte bortse ifrån att det – som denna rapport tydligt visar – finns ett äkta och skriande behov av ett manusverktyg för interaktiva medier. Den skapade prototypen utgör en utmärkt utgångspunkt för framtagandet av en färdig applikation. **Därför torde alla med intresse av att utveckla berättandet inom den interaktiva underhållningsindustrin ha anledning att sträva efter att driva detta arbete vidare.**

KÄLLFÖRTECKNING

Alla Internet-baserade källor verifierades senast 2003-07-01.

[Aarseth 97] AARSETH, E.J. 1997. *Cybertext. Perspectives on Ergodic Literature*. The Johns Hopkins University Press. ISBN 0-8018-5578-0

Intressant bok som bl.a. tar upp den ergodiska litteraturens historia med en mängd referenser till verk och genrer inom ämnet.

[Becker, Hoogland 97] BECKER, P & HOOGLAND, R. 1997. *Bakom Kulisserna. August Strindbergs Pelikanen*. Liber Förlag.

En CD-ROM-produktion som presenterar planeringsverktyget 3D-Teatern med utgångspunkt från en uppsättning av A. Strindbergs pjäs *Pelikanen*.

[Beyer, Holtzblatt 97] BEYER, H. & HOLTZBLATT, K. 1997. *Principles of Contextual Inquiry*. Ur Contextual Design. Morgan Kaufmann Publishers.

Avsnittet handlar om en metod för systemdesign som ger bättre insikter om slutanvändaren och dennes arbetssituation.

[BioWare 02] BioWare. 2002. Officiell webbsida för Baldurs Gate II.

http://www.bioware.com/games/shadows_amn/

[BioWare 03] BioWare. 2003. Officiell webbsida för Neverwinter Nights.

<http://nwn.bioware.com/>

[Bolter 91] BOLTER, J.D. 1991. *Writing Space. The Computer, Hypertext, and the History of Writing*. Lawrence Erlbaum Associates. ISBN 0-8058-0428-5

En bok som tar upp problemet med att skriva icke-linjär litteratur i linjära (tryckta) media och hur datorn är lösningen som låter oss skriva interaktiva texter.

[Boye 42] BOYE, K. 1942. *I rörelse*. Ur Härdarna. Albert Bonniers förlag. ISBN 91-0-056159-2

[Britton 03] BRITTON, S. 2003. Personlig hemsida om interaktiv fiktion med bl.a. länkar till flera spel. <http://www.igs.net/~tril/if/>

[Crawford 03] CRAWFORD, C. 2003. Webbsida för programvaran Erasmatron och Erasmaganza. <http://www.erasmatazz.com>

[Crawford 97] CRAWFORD, C. 1997. Online-intervju med Chris Crawford. <http://www.omnimag.com/archives/chats/em080497.html>

[Darlington 98] DARLINGTON, S. 1998. The History of Role-Playing. *Places to go, people to be*. Nummer 1/1998. <http://ptgptb.org/0001/history1.html>

Utmärkt artikel i nio delar om rollspelets historia.

[Deadline 97] Deadline Games. 1997. Officiell webbsida med en beskrivning av spelet Blackout. <http://www.deadline.dk>

[Dever, Chalk 85] DEVER, J. & CHALK, G. 1985. *Ensamma Vargen 2: Eld över vattnet*. Äventyrsspel. ISBN 91-7898-001-1

[Eastgate 02] Eastgate Systems. 2002. Webbsida för programvaran Storyspace. <http://www.eastgate.com/Storyspace.html>

[Futuredrome 03] 2003. Webbsidan för lajvet Futuredrome 2002 och dess uppföljare Afterworld 2003. <http://www.futuredrome.com>

- [**Gould m.fl. 97**] GOULD, J.D. BOIES, S.J. & UKELSON, J. 1997. *How To Design Usable Systems*. Ur Handbook of Human-Computer Interaction. North-Holland. ISBN 0444818626
- Kapitel med en beskrivning av fyra principer för användbar systemdesign.
- [**Hyperstudio 01**] Roger Wagner Publishing. 2001. Webbida för Hyperstudio. <http://www.hyperstudio.com>
- [**Jones 03**] JONES, C. 2003. Hemsida för Adventure Game Studio. <http://www.adventuregamestudio.co.uk/>
- [**Kristof, Satran 95**] KRISTOF, R., SATRAN, A. 1995. *Interactivity by Design*. Adobe Press. ISBN 1-56830-221-5
- Ganska enkel bok med bilder och råd om hur man bör utforma information, interaktion och gränssnitt i datorbaserade interaktiva produkter; främst hemsidor.
- [**Löwgren, Stolterman 99**] LÖWGREN, J. & STOLTERMAN, E. 1999. Design Methodology and Design Practice. *Interactions*. Nummer 1/1999. <http://www.acm.org/interactions/>
- [**Macromedia 03**] Macromedia. 2003. Webbida för Director MX. <http://www.macromedia.com/software/director/>
- [**Manovich 01**] MANOVICH, L. 2001. *The Language of New Media*. The MIT Press. ISBN 0-262-13374-1
- [**NE 03**] *Nationalencyklopedien*. 2003. Internet-versionen. <http://www.ne.se>
- [**TSR 96**] *Player's Handbook for the AD&D Game*. 1996. TSR. ISBN 0-7869-0329-5
- [**Rajka 96**] RAJKA, P. 1996. *Undersökningar för projektet "Digitala Teaterrum"*. En opublicerad förundersökningsrapport inom projektgruppen Digitala Teaterrum.
- [**RealWorld 98**] RealWorld Multimedia. 1998. *Ceremony of Innocence*. ISBN 0952086476
- [**Roberts 02**] ROBERTS, M. J. 2002. Text Adventure Development System Manual, Chapter 9. <http://tela.bc.ca/tads/>
- [**Roberts 03**] ROBERTS, M. J. 2003. Officiell webbsida för Text Adventure Development System. <http://www.tads.org>
- [**RPM 03**] RolePlayingMaster. 2003. Programvarans officiella webbsida. <http://www.roleplayingmaster.com>
- [**Schneiderman 98**] SCHNEIDERMAN, B. 1998. *Designing the User Interface*. Addison Wesley. ISBN 0-201-69497-2
- Ett standardverk om människa-datorinteraktion och gränssnittsdesign.
- [**Screenplay 03**] Screenplay Systems. 2003. Webbida för Dramatica Pro. <http://www.dramatica.com/>
- [**Seter 02:a**] SETER, M. 2002. Att skriva äventyr för rollspel. *Sverox*. Nummer 23/2002.
- [**Seter 02:b**] SETER, M. 2002. *Grundläggande omvårdnad, modul 7*. Houdini Digital Creations.
- Manuskript för en undervisningsproduktion för omvårdnadsutbildningar.

[Stansberry 97] STANSBERRY, D. 1997. *Labyrinths. The Art of Interactive Writing and Design*. Wadsworth Publishing. ISBN 0-534-51948-2

Verket fokuserar på hur man skriver manus och strukturerar interaktionsdesignen för olika multimediprojekt inom utbildning, underhållning och i affärssammanhang. Många konkreta exempel från författarens egna arbeten.

[Twin Rose 03] Twin Rose Software. 2003. Forum och produktinformation om Campaign Suite. <http://twinrose.net>

[Unga Klara 03] *Det allra viktigaste*. 2003. Pjäsk skriven av Nikolaj Jevreinov, regisserad av Suzanne Osten och uppförd på Unga Klara i Stockholm. <http://www.stadsteatern.stockholm.se/ungaklara/ungaklara.asp>

[Virtual Galileo 02] 2002. Den officiella webbsidan för Virtual Galileo. <http://virtual-galileo.nu/>

[Wilhelmsson 02] WILHELMSSON, P. 2002. Projektredogörelse för The Longest Week med möjlighet att ladda ner spelet. <http://www.student.nada.kth.se/~d98-pwi/TheLongestWeekWeb/Projektredog.htm>

BILAGA 1: FRÅGOR INFÖR INTERVJU MED SETER

Följande frågor ställdes under den första intervjun med Magnus Seter:

- Berätta om dig själv och din bakgrund!
- Varför är du intresserad av ett manusverktyg för interaktiva produktioner?
- Vilka (yrkesgrupper) arbetar du med?
- Vilka andra roller har du förutom manusförfattare i ett projekt?
- Finns det några generella likheter mellan olika manusförfattare och i så fall vilka?
- Hur mycket datakunskaper kan man förvänta sig av en manusförfattare?
- Finns det några spel/manusverktyg på marknaden?
- Vilka hjälpmedel använder man sig idag av då man skriver? (Föreslå Word, Post-It etc.)
- Vilka fördelar har dessa verktyg?
- Vilka nackdelar har dessa verktyg och vilken funktionalitet saknar de?
- Har du några visioner om hur ett manusverktyg skulle kunna se ut?
- Vilka nödvändiga funktioner skulle ett manusverktyg minst ha för att vara användbart?
- Vem skrivs ett manus för?
- Hur ser ett manus ut idag? Består det bara av text, eller ingår andra medieformer i det?
- Vad är svårast med att skriva ett manus förutom att vara kreativ? (Föreslå visualisering, att hålla reda på saker etc.)
- Vad är viktigast då man skriver ett manus? (Föreslå att hålla ihop berättelsens struktur, kunna länka till andra medieformer etc.)
- Finns det någon metafor man arbetar utefter? (Föreslå tidslinje och karta.)
- Vilka delar av en berättelse arbetar man med och vilka har störst betydelse? (Föreslå karaktärer, miljöer, dialoger, händelser och valalternativ.)
- När i ett projekt skriver man ett manus?
- Finns det några allmänna arbetssätt och dokument i dataspelsprojekt?
- Beskriv hela skrivandeprocessen; milstolpar och vad man utgår ifrån i de olika stegen!
- Vilka relationer har de olika dokumenten som skrivs i en produktion, t.ex. storyboards, synopsis...
- Är arbetet med att skriva olika dokument iterativt eller seriellt eller..?
- Vad är skillnaden mellan ett actionspel och ett spel med mer traditionellt berättande (t.ex. äventyrsspel) ur manusperspektiv?
- Du har även skrivit manus för rollspel, ser du några likheter och olikheter med det och att skriva manus för dataspel?
- Kan man få se på ett manus?

BILAGA 2: RAJKAS MODELLFÖRSLAG

Följande lista är ett urval från ett modellförslag av Peter Rajka [Rajka 96] som beskriver hur man ska kunna klassificera och strukturera olika typer av händelserepresentationer inom teatern. Rajka påpekar att det inte är fråga om en exakt specifikation av datastrukturer utan endast en principiell utgångspunkt för vidare arbete och diskussion. Modellen bygger på fyra olika klasser av händelserepresentation:

- Klassen av händelser, som är en generell och abstrakt klass med övergripande egenskaper för alla typer av händelser:
 - Händelsens namn.
 - Händelsebeskrivning.
 - Successionslänkar till nästa/föregående händelseobjekt på samma nivå.
 - Interna hierarkilänkar till över- och underordnade händelser i samma klass, t.ex. från en scen till en akt.
 - Referenslänkar till associerade händelseobjekt.
- Klassen av fabel med följande egenskaper:
 - Personnamn.
 - Personbeskrivning.
 - Namn på miljö eller föremål.
 - Beskrivning av miljö eller föremål.
- Klassen av dramastruktur (dramatexten):
 - Rollista med konkreta namn.
 - Scenanvisningar.
 - Strukturelement av olika typ (t.ex. akt, scen, replik).
- Klassen av spelstruktur (föreställningen):
 - Scenografiska konstruktioner (t.ex. dekor, möbler).
 - Ljus (länkar till ljusmoment).
 - Ljud (länkar till t.ex. ljudfiler).
 - Projektioner (länkar till t.ex. rörliga bildfiler).

Händelserepresentationen kompletteras med andra funktioner som bl.a. ett bibliotek uppdelat på olika arkiv för ljud, ljus etc. och en verkstad där bibliotekets element kan konstrueras och modifieras.

BILAGA 3: OM ROLLSPEL

Följande text är tagen ur Player's Handbook [TSR 96] som är ett regelverk för spelare av rollspelet Advanced Dungeons & Dragons. Texten förklarar hur ett klassiskt rollspel fungerar:

The Real Basics

Imagine that you are playing a simple board game, called Snakes and Ladders. Your goal is to get from the bottom to the top of the board before all the other players. Along the way are traps that can send you sliding back toward your starting position. There are also ladders that can let you jump ahead, closer to the finish space. So far, it's pretty simple and pretty standard.

Now let's change a few things. Instead of a flat, featureless board with a path winding from side to side, let's have a maze. You are standing at the entrance, and you know that there's an exit somewhere, but you don't know where. You have to find it.

Instead of snakes and ladders, we'll put in hidden doors and secret passages. Don't roll a die to see how far you move; you can move as far as you want. Move down the corridor to the intersection. You can turn right, or left, or go straight ahead, or go back the way you came. Or, as long as you're here, you can look for a hidden door. If you find one, it will open into another stretch of corridor. That corridor might take you straight to the exit or lead you into a blind alley. The only way to find out is to step in and start walking.

Of course, given enough time, eventually you'll find the exit. To keep the game interesting, let's put some other things in the maze with you. Nasty things. Things like vampire bats and hobgoblins and zombies and ogres. Of course, we'll give you a sword and a shield, so if you meet one of these things you can defend yourself. You do know how to use a sword, don't you?

And there are other players in the maze as well. They have swords and shields, too. How do you suppose another player would react if you chance to meet? He might attack, but he also might offer to team up. After all, even an ogre might think twice about attacking two people carrying sharp swords and stout shields.

Finally, let's put the board somewhere you can't see it. Let's give it to one of the players and make that player the referee. Instead of looking at the board, you listen to the referee as he describes what you can see from your position on the board. You tell the referee what you want to do and he moves your piece accordingly. As the referee describes your surroundings, try to picture them mentally. Close your eyes and construct the walls of the maze around yourself. Imagine the hobgoblin as the referee describes it whooping and gambling down the corridor toward you. Now imagine how you would react in that situation and tell the referee what you are going to do about it.

We have just constructed a simple role-playing game. It is not a sophisticated game, but it has the essential element that makes a role-playing game: The player is placed in the midst of an unknown or dangerous situation created by a referee and must work his way through it.

This is the heart of role-playing. The player adopts the role of a character and then guides that character through an adventure. The player makes decisions, interacts with other characters and players, and, essentially, "pretends" to be

his character during the course of the game. That doesn't mean that the player must jump up and down, dash around, and act like his character. It means that whenever the character is called on to do something or make a decision, the player pretends that he is in that situation and chooses an appropriate course of action.

Physically, the players and referee (the DM) should be seated comfortably around a table with the referee at the head. Players need plenty of room for papers, pencils, dice, rule books, drinks, and snacks. The referee needs extra space for his maps, dice, rule books, and assorted notes.

The Goal

Another major difference between role-playing games and other games is the ultimate goal. Everyone assumes that a game must have a beginning and an end and that the end comes when someone wins. That doesn't apply to role-playing because no one "wins" in a role-playing game. The point of playing is not to win but to have fun and to socialize.

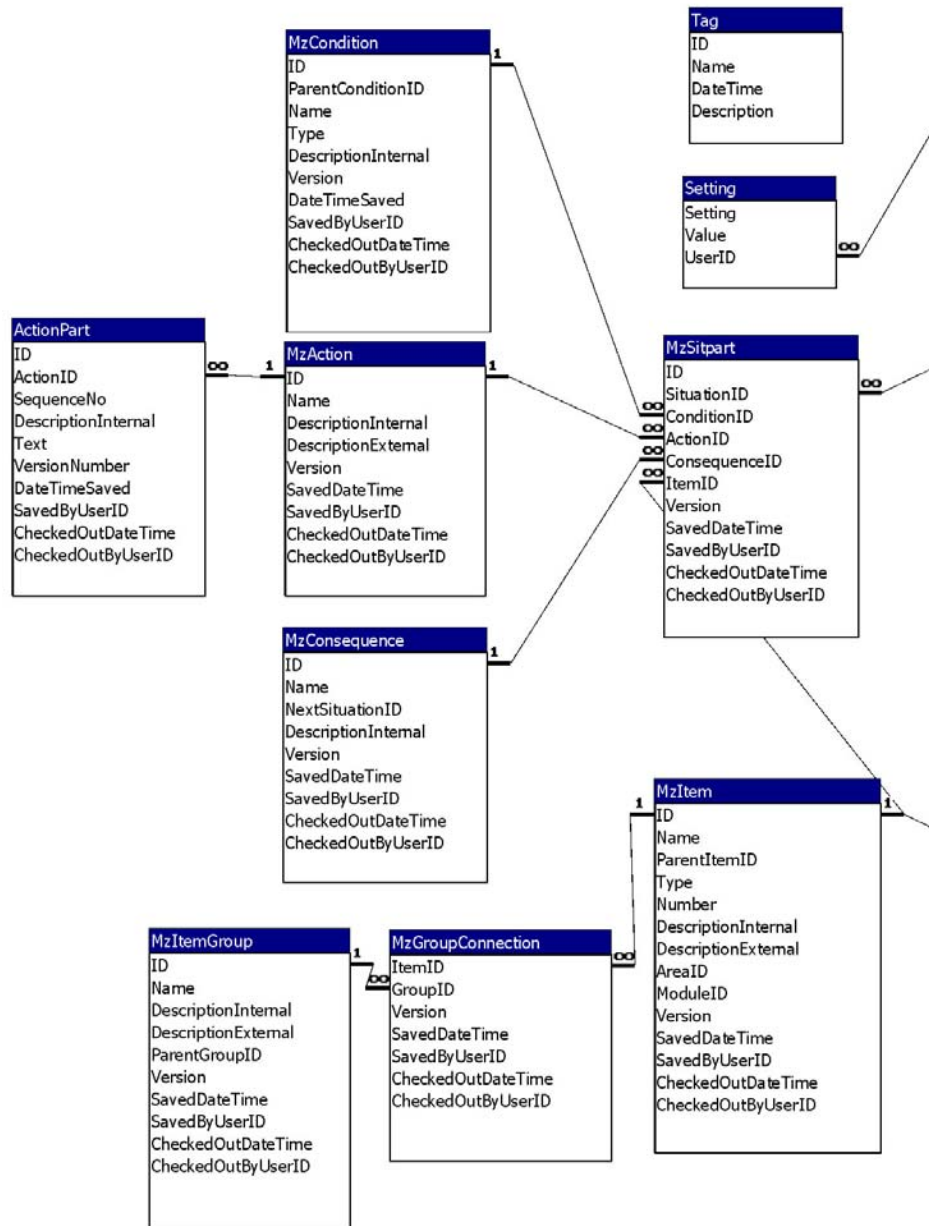
An adventure usually has a goal of some sort: protect the villagers from the monsters; rescue the lost princess; explore the ancient ruins. Typically, this goal can be attained in a reasonable playing time: four to eight hours is standard. This might require the players to get together for one, two, or even three playing sessions to reach their goal and complete the adventure.

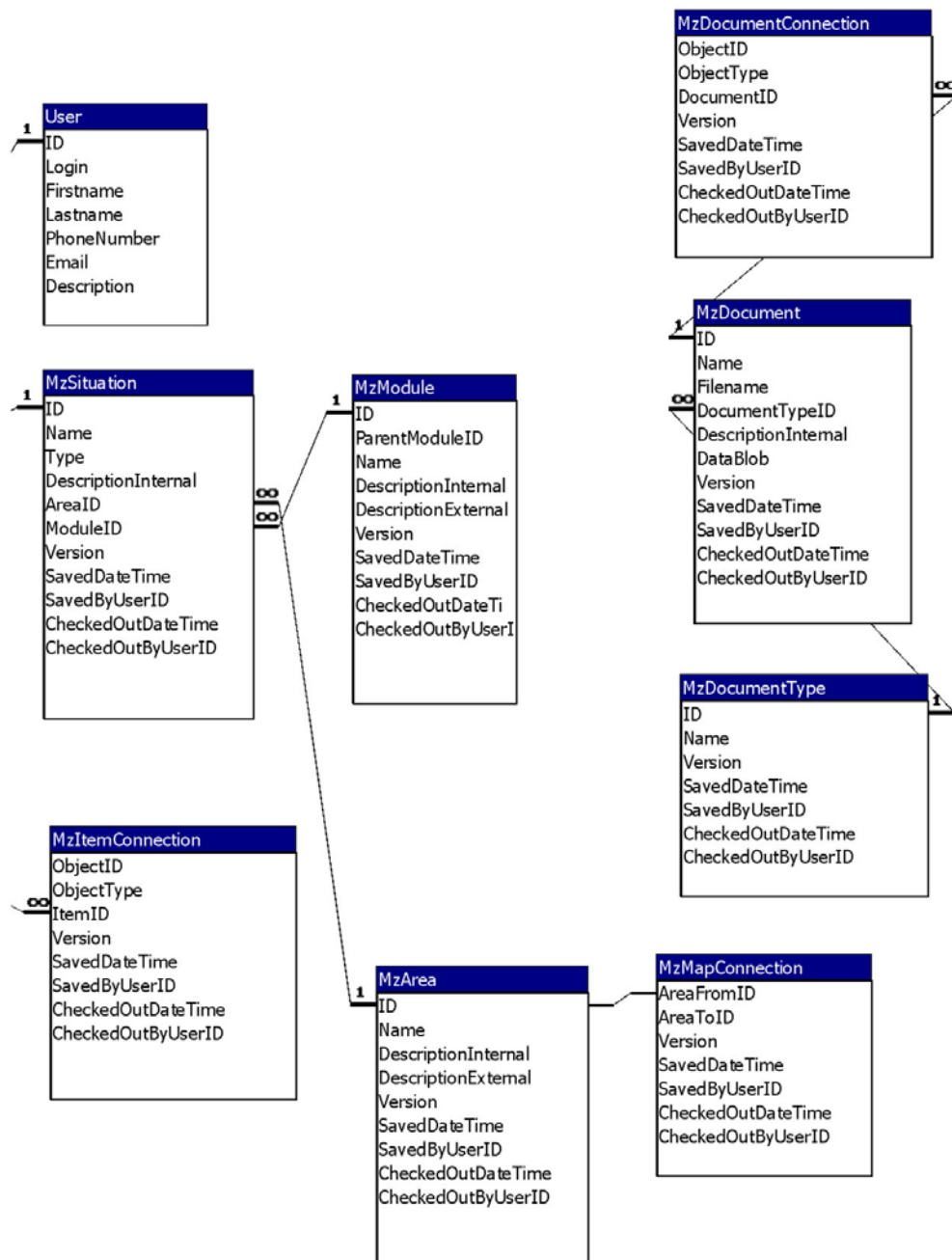
But the game doesn't end when an adventure is finished. The same characters can go on to new adventures. Such a series of adventures is called a campaign.

Remember, the point of an adventure is not to win but to have fun while working toward a common goal.

BILAGA 4: MAZERMINDS DATABASSTRUKTUR

Följande två bilder visar de viktigaste delarna och relationerna i Mazerminds databasstruktur. Lägg gärna märke till den ”ryggrad” som sträcker sig från *ActionPart* till *MzModule*.





BILAGA 5: UTDRAG UR ETT MANUS

Följande dokument är ett utdrag ur undervisningsproduktionen ”Grundläggande omvårdnad” [Seter 02:b]. Utdraget publiceras med särskilt tillstånd av författaren.

02-02-22 © Houdini Digital Creations 89/148

Ajdå. Termometer för mätning i ändtarm ska inte användas någon annanstans. Försök igen.

[momentet temperaturmätning börjar om]

Del 3

[om två eller färre fel:]

Text voa_7_1_U301t14
Bra, du mätte temperaturen - då går vi vidare!

Speaker voa_7_1_U301s20 [Sebastian]
Bra, du mätte temperaturen - då går vi vidare!

[tona över till extro]

[om mer än två fel:]

Text voa_7_1_U301t15
Du mätte temperaturen, men inte helt utan problem. Vill du försöka igen eller gå vidare?

Speaker voa_7_1_U301s21 [Sebastian]
Du mätte temperaturen, men inte helt utan problem. Vill du försöka igen eller gå vidare?

- Bild voa_7_1_U301b57 [”OK-knapp”]

Text voa_7_1_U301t16
Försök igen!

Text voa_7_1_U301t17
Gå vidare.

[om ”försök igen”: momentet temperaturmätning börjar om – om ”Gå vidare”: tona över till Extro]

Avslutning/extro

- Bild voa_7_1_U301b58 [Emma sitter i uppfälld säng med kuddar bakom ryggen med en termometer i munnen. Utdragsbordet är utfällt och hon lägger patient. Solen lyser in genom det öppna fönstret. Sebastian kommer med frukost]
- Bild voa_7_1_U301b59 [Textruta i övre vänstra hörnet – på känt serietidningsmanér]

Text voa_7_1_U301t18
Två dagar senare...

Speaker voa_7_1_U301s22 [Sebastian]
Hejsan, hur känns det i dag, så?

Speaker voa_7_1_U301s23 [Emma (pratar med termometer i munnen)]
Jotack, bra – febern är nog borta.
[”Okack, ra – federn e no otta nu”]

Slut

Eleven går till Uppdragsmenyn.

voa_7_1_U302 Uppgift 2 Puls

2a

Pulsen är den tryckvåg som bildas när blodet slungas ut från hjärtat. Man känner hjärtats takt. Pulsen kan kännas på olika delar av kroppen. Det vanligaste stället att känna pulsen på är på handledens insida. Andra ställen som pulsen kan kännas på är på halsen, vid tinningen och i ljumsken. Normalt ligger pulsen hos en vuxen på 60-80 slag/minut i vila. Barn har högre puls. Pulskontroll genomförs normalt på en minut.

Hur lär jag mig ta pulsen på en person och vad innebär det att ta puls?

Eleven ska kunna kontrollera, registrera och iaktta puls.

Vad ska man iaktta vid pulstagning?

- Vad handlar uppgiften om?
Miranda kommer in till Östen som larmat eftersom han känt dubbelslag och blivit orolig.
- Etablera scenen/scenförändringar
Östens sovrum.
- Definiera roller som förekommer i uppgiften
Miranda, Östen (Sebastian i spelmomenten).

Problemställning/utmaning/intro

- Bild voa_7_1_U302b01 [Miranda sitter på sängkanten hos Östen som ser bekymrad ut]

Speaker voa_7_1_U302s01 [Miranda]
Hur är det, Östen?

Speaker voa_7_1_U302s02 [Östen (orolig)]
Jag får såna otäcka dubbelslag, jag undrar om det är nåt med hjärtat...

Speaker voa_7_1_U302s03 [Miranda (lugnande)]
Ja, extraslag kan vara obehagliga. Vi ska se vad du har för puls...

BILAGA 6: FLER INTERVJUFRÅGOR

Följande frågor skrevs inför den andra intervjun med Magnus Seter.

- Kan det vara en idé med namnmallar för dokumenttyper så att dokumentens namn kan genereras?
- Vad har fil/dokumentnamnen för funktion?
- Hur (Vem) beskriver man att ett ljud ska gå över två bilder (eller tvärtom)?
- Hur beskriver man ett val som nästan alltid kan göras, t.ex. att gå till en huvudmeny?
- Vilka funktioner som Word tillhandahåller saknar du mest i prototypen av Mazermind?
- Arbetar man som författare med flera manus samtidigt?
- Vill man ha en pappersversion av sitt manus?
- Hur komplext verktyg vill man ha? Jämför man prototypen med t.ex. programvara för rollspelshantering (RolePlaying Master) är skillnaden himmelsvid.
- Kan du tänka dig att testa verktyget igen?
- Känner du andra manusförfattare som kan vara intresserade av och ha synpunkter på Mazermind?
- Hur låter namnet "Mazermind"?