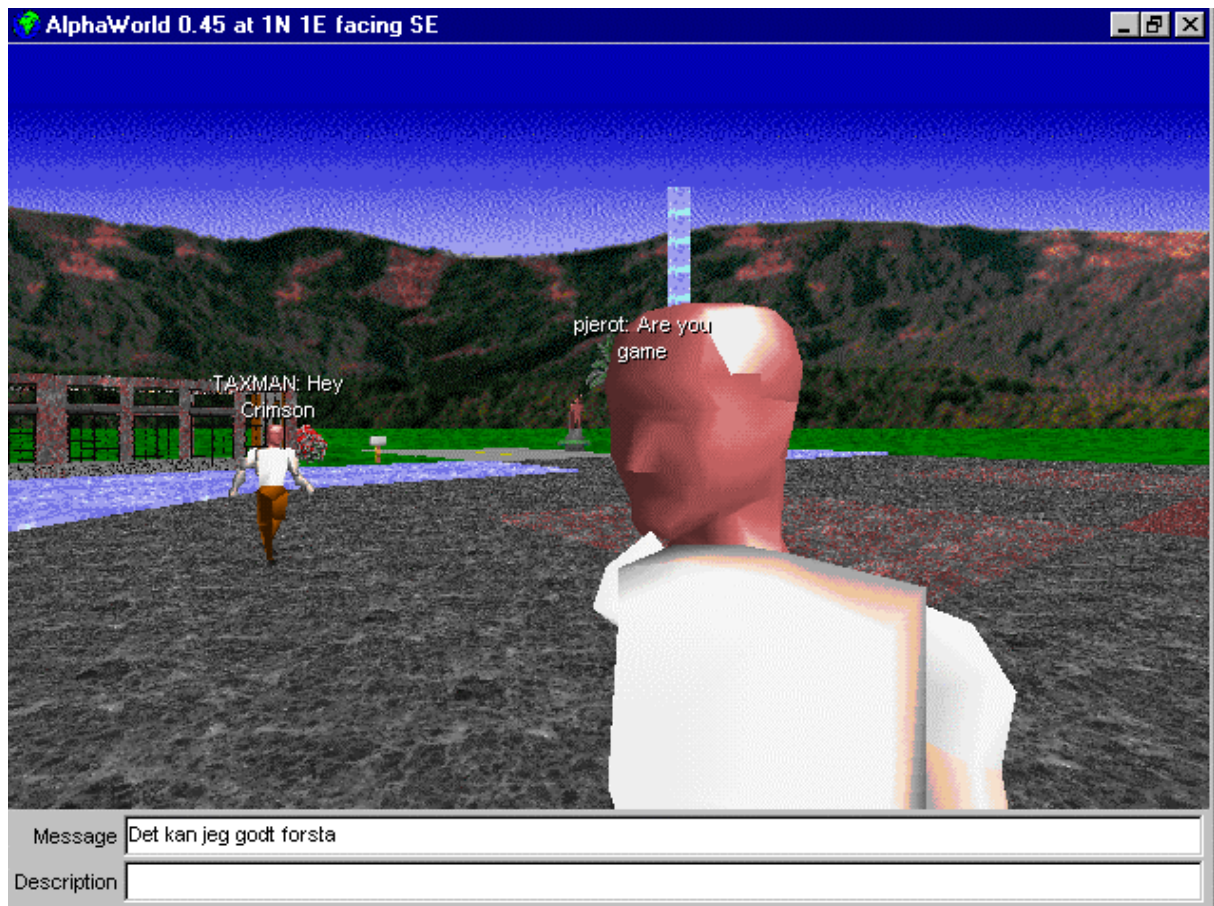


Fra HTML til VRML



**Miniprojekt i faget "World Wide Web"
Handelshøjskolen i København
21. december 1995**

**Thomas Hensing, 130968
Søren Tjørnov, 151068
Birger Larsen, 050669**

Indholdsfortegnelse

1. Virtual Reality Modeling Language	3
1.1 Hvad er VRML?	3
1.2 Hvordan virker VRML?	4
1.3 Hvad består VRML sproget af?	5
1.3.1 Nodes	5
1.3.2 Node syntaks	7
1.3.3 Scene graphs og noders virkefelt	7
1.4 VRML browsere	8
1.4.1 WebSpace	8
1.4.2 WorldView	10
1.4.3 WebFX	10
1.5 VRML editorer	11
1.5.1 Virtual Home Space Builder	12
1.5.2 Fountain	13
1.6 VRML testscene	14
1.7 Virtual Reality analyse af VRML	16
1.7.1 Navigation ved hjælp af hoved og kropsbevægelse	16
1.7.2 Visualisering af scener i Head-mounted-displays	16
1.7.3 2D animation	17
1.7.4 Manipulation/bevægelse af objekter	17
1.7.5 3D animation	17
1.7.6 Collision detection	17
1.7.7 Autonome objekter	17
1.7.8 Flere samtidige deltagere i scener	17
1.7.9 Audio	18
1.7.10 Andre sanser	18
1.7.11 Konklusion	18
1.8 Fremtiden for VRML	18
1.8.1 Krav til et 3D grafiksprog, der understøtter adfærd	20
1.8.2 Konklusion	21
2. VRML mediet og HTML mediet	22
2.1 Styrker ved HTML mediet	22
2.2 Styrker ved VRML mediet	22
2.2.1 Intuitiv navigation i informationsbjerge	22
2.2.2 Formidling af 3D informationer	24
2.2.3 Det sociale aspekt	26
2.3 Det nye og det gamle medie	30
2.4 Hvordan vil brugen af HTML og VRML finde sted i fremtiden?	32
2.4.1 Scenarie 1: VRML indlejret i HTML	32
2.4.2 Scenarie 2: HTML og VRML eksisterer side om side	32
2.4.3 Scenarie 3: VRML med transparent brug af indlejret HTML	33
2.4.4 Konklusion	33
3. Konklusion	34
Bilag 1 - sourcekode til VRML scene	35

Indledning

I 1989 udviklede Tim Berners-Lee på Center for European Particle Physics (CERN) en hypertext brugergrænseflade, der senere blev til det allesteds nærværende World Wide Web. Før fremkomsten af WWW har adgangen til information på internettet været hæmmet af tunge/klodsede brugergrænseflader, der bl.a. krævede, at brugeren skulle huske og indtaste lange sekvenser af IP-numre, filstier, kryptiske UNIX-kommandoer og filnavne. Med WWW blev navigation på nettet langt mere intuitiv. WWW's hypertext brugergrænseflade tillader brugeren hurtigt at overskue og navigere i store informationsmængder. Overskueligheden fremmes ved, at tekst kan suppleres med billeder, lyd, videoklip m.m. Navigationen lettes ved at brugeren kan klikke på udvalgte emneord i HTML dokumenter og få vist mere detaljeret information om disse emneord i andre relaterede HTML dokumenter.

At WWW er en succes, er der i dag ingen der bestrider. På få år har det fuldstændigt udkonkurreret etablerede internet-brugergrænseflader som Gopher, FTP, WAIS m.fl. Ikke desto mindre arbejdes der i øjeblikket seriøst på at skabe det, som nogen vil kalde en erstatning for og andre en overbygning til WWW - kendt under forkortelsen VRML. Virtual Reality Modeling Language er et 3D scenebeskrivelses sprog, der er tænkt som en Virtual Reality brugergrænseflade til internettet. I udviklingen af VRML sproget deltager bl.a. så prominente navne som Silicon Graphics, Sony, Digital Equipment Corporation m.fl.

Vi vil i denne opgave forsøge at blotlægge det nye medies muligheder. Endvidere vil vi sammenholde VRML mediet med det nuværende WWW medie baseret på HTML dokumenter. Vi får således følgende problemformulering:

Vi ønsker at afdække anvendelsesmulighederne i VRML mediet, og specielt hvilken rolle det nye medie vil det få i forhold til HTML mediet.

Vi har opdelt opgaven i to dele. I kapitel 1 kigger vi nærmere på teknologien bag det nye medie. Specielt analyserer vi VRML sproget, værktøjer til at browse og editere VRML scener. Endvidere analyserer vi VRML ud fra en generel Virtual Reality synsvinkel og ser på VRML's fremtidsudsigter. I kapitel 2 slipper vi det teknologiske perspektiv og anlægger istedet en medieorienteret synsvinkel. Specielt analyserer vi styrker og svagheder for henholdsvis VRML mediet og HTML mediet. Vi afslutter med en vurdering af hvordan brugen af det nye og det gamle medie vil finde sted i fremtiden.

1. Virtual Reality Modeling Language

I dette kapitel beskrives og diskuteres teknologien bag VRML mediet, herunder VRML sproget, VRML browsere, VRML editorer m.m. Afsnittet skal give læseren en basal forståelse af vigtige begreber og komponenter i VRML teknologien og dermed danne fundamentet for kapitel 2., hvor vi foretager en sammenlignende analyse af HTML og VRML.

I afsnit 1.1 defineres VRML begrebet. I afsnit 1.2 diskuteres hvordan VRML virker i praksis. I afsnit 1.3 foretages en mere detaljeret diskussion af VRML sproget. I afsnit 1.4 diskuteres VRML browsere. I afsnit 1.5 diskuteres VRML editorer. I afsnit 1.6 beskrives en simpel VRML scene, som vi har lavet for at illustrere mulighederne i sproget. I afsnit 1.7 diskuteres VRML ud fra en Virtual Reality synsvinkel. Specielt diskuteres mangler i VRML i forhold til Virtual Reality teknologien i al almindelighed. Endelig afsluttes i afsnit 1.8 med en vurdering af den fremtidige udvikling af VRML mediet.

1.1 Hvad er VRML?

Virtual Reality Modeling Language er et sprog til at beskrive tredimensionelle grafiske figurer og verdener. Det hedder i VRML terminologi en *scene*. VRML indholder som de fleste andre scenebeskrivelsessprog faciliteter til at beskrive geometriske former, teksturer, lyskilder osv. Det som adskiller VRML fra de mange andre scenebeskrivelsessprog er, at det er skræddersyet til brug på internettet.

VRML sproget muliggør, at man kan forbinde én VRML scene med andre VRML scener på internettet. Dette foregår ved, at man kan relatere en figur i én scene til en anden scene. Klikker man på en figur i en scene der peger på en anden scene, vil man blive ført til den anden scene. Man kan på den måde hoppe rundt på internettet gennem et net af forbundne VRML scener. VRML er altså et 3D hypermedie, hvor man kan skifte fra scene til scene, på samme måde som når man klikker på et anchor i HTML, og det valgte HTML dokument hentes og vises.

Bortset fra evnen til at forbinde scener på tværs af nettet, kan man ikke sige, at VRML sproget er banebrydende i forhold til andre scenebeskrivelsessprog. Opfinderne af VRML ønskede ikke at opfinde den dybe tallerken igen, og valgte derfor at bygge sproget på et allerede eksisterende sprog, nemlig Open Inventor fra Silicon Graphics. Open Inventor er et værktøj som Silicon Graphics benytter til at lave tredimensionelle grafiske prototyper med, og det er det scenebeskrivende sprog fra Open Inventor og det tilhørende filformat som udgør grundstenen i VRML. Grunden til, at de ikke valgte et andet sprog var, at Silicon Graphics valgte at stille sprog og filformat fra Open Inventor til fri afbenyttelse for andre virksomheder og privatpersoner. På den måde kom opfinderne hurtigt i gang, og kunne koncentrere sig om at udvikle den første parser, som de også havde fået den grundlæggende sourcekode til af SGI.

Oprindeligt var VRML ikke tænkt som et selvstændigt sprog, men som en udvidelse af HTML sproget, således at HTML også kunne vise 3D grafik. Men man indså hurtigt, at visning af 3D grafik adskilte sig så fundamentalt fra visning af tekst og 2D grafik, at det krævede udviklingen af et separat sprog, separate browsere og editorer. Der hører en masse historier til hvorledes VRML i øvrigt udviklede sig frem til den første version, men vi vil nøjes med at nævne, at version 1.0 af VRML blev vedtaget i sommeren 1995, og version 1.1 i november 1995. Version 2.0 forventes at blive vedtaget i maj 1996.

Selv om der foreligger en endelig specifikation af sproget, så betyder det ikke at syntaksen er beskrevet fyldestgørende. Der er ganske vidst udført et stort arbejde med at dokumentere VRML, f.eks. er der fremkommet en BNF¹ beskrivelse af VRML 1.0, men i nogle tilfælde er man nødt til at søge hjælp i den originale manual til Open Inventor (der ligger til grund for VRML sproget) for at få en komplet oversigt over visse VRML kommandoers syntaks.

1.2 Hvordan virker VRML?

I det følgende vil vi bl.a. beskrive, hvordan VRML hentes, genereres og vises, hvordan det fungerer ift. HTML, samt hvordan VRML kommunikerer med Web-servere.

VRML virker stort set på samme måde som det velkendte HTML. VRML er altså ikke et programmeringssprog, der bliver kompileret og eksekveret, men et beskrivende sprog, der fortolkes af en browser. VRML 'koden' er således fundamentet for den scene, der dannes af browseren. I en VRML scene kan man, ligesom med farvet, understreget tekst i et HTML dokument, klikke på et objekt, der peger videre på en anden VRML scene, hvorefter man sendes (teleportes) over i denne næste VRML scene osv. Objekter i VRML scener kan også pege på HTML dokumenter. Støder man på en URL, der peger på et HTML dokument, da vil man oftest have brug for at få vist dette i en HTML browser² Omvendt kan man også, i et HTML dokument, støde på en VRML scene, som man skal have vist i sin VRML browser. Et typisk samspil mellem VRML og HTML kunne se således ud:

HTML browser: "Hov, her er en VRML fil. Hej, VRML browser, jeg har en VRML fil, du skal vise..."

VRML browser: "OK. Sådan ser den ud...(scenen vises)...Hov, brugeren klikkede på et objekt, der peger på et HTML dokument. Hallo HTML browser!, dette skal du vise..."

HTML browser: "Ja tak. Sådan ser det ud....(dokument vises)..." osv. osv.

VRML scener er absolut statiske³, dvs. at der ikke kan ændres på en scenes udseende, efter at den er blevet indlæst, parset og renderet af browseren. Alle objekterne i en VRML scene er opbygget af forbundne punkter, der danner polygoner og dermed komplette grafiske objekter og verdener. Hvorledes en figur/et objekt genereres (renderes) er op til browseren. De fleste browsere kan renderere VRML scener i wireframe, hvilket giver en betydelig hastighedsgevinst, da generering af overflader på polygoner (shading) er ret ressourcekrævende. Da VRML filer oftest er en del større end HTML filer, er de, for at mindske overførselstiden, komprimeret på GZIP-format⁴, som ens browser skal kunne dekomprimere.

Tiden, der bruges til parsing og rendering af VRML scenen og alle dens inline scener, afhænger selvfølgelig af computer- og modemhastighed. Det er dog i højere grad end HTML et spørgsmål om hvor hurtig computeren, idet beregning af 3D grafik er meget krævende.

Skulle overførslen, parsingen og renderingen imidlertid gå for langsomt, kan man vha. VRML's såkaldte lazy-loading-funktion indledningsvist få vist scenens objekter som tomme

¹ Backus/Naur syntaks beskrivelser

² Samspillet mellem VRML og HTML browser kan være forskelligt, afhængig af hvilken type VRML browser man anvender. Mere herom i afsnit 1.4 om VRML browsere.

³ VRML scenerne er alle statiske i VRML version 1.0.

⁴ Et komprimeringsformat der normalt kun benyttes på UNIX-computere

kasser (bounding boxes) i de rette dimensioner og på de rette positioner. Man behøver således ikke vente på, at alle scenens objekter parses og renderes endeligt, før man kan påbegynde sin navigation i scenen. De færdige objekter dukker så op et for et under ens navigation. En stor, stor fordel, hvis scenen indeholder hundredvis af objekter.

En anden væsentlig egenskab i VRML, der kan spare én for tidsrøvende navigation gennem store, tunge scener på en langsom computer, er scene-forfatterens muligheden for at forud-definere forskellige (interessante) positioner, som man kan springe direkte til. Ens synsfelt/-position i VRML scener er givet ved et kamera, man styrer rundt i scenen. Ved at springe direkte til en anden kameraposition, kan man således spare en lang rejse frem til disse positioner.

Vender vi blikket mod Web-servere, så forholder det sig så smart, at disse kan håndtere VRML filer stort set uden nogen ændring. I princippet fungerer en Web-server på samme måde, uanset om den håndterer HTML dokumenter, VRML scener eller andre filformater. En 'dialog' mellem en server og en VRML browser kunne typisk se således ud:

VRML browser: "Hallo server, har du filen ABC.WRL?"

Web-server: "Ja. Den er på 55 Kb, fra 951203 og er en VRML fil. Vil du ha' den?"

VRML browser: "Ja tak, send mig den..."

Web-server: "Her kommer den...(sender fil)...værsgo'..."

VRML browser: "Tak..." (viser ABC.WRL-scenen)

Efter den ovenstående overordnede beskrivelse af, hvordan VRML fungerer, vil det være oplagt at gå lidt 'tættere på'. I afsnittet nedenfor beskrives selve VRML sproget og syntaksen mere detaljeret.

1.3 Hvad består VRML sproget af?

I nedenstående diskuteres VRML sproget. Først gennemgås nodes, der er VRML sprogets byggesten. Derefter gennemgås syntaksen for nodes og endelig diskuteres noders virkefelter.

1.3.1 Nodes

En VRML scene består af en række enheder, der i VRML sproget kaldes **nodes**. En node kan være f.eks. være et kamera, en lyskilde eller en geometrisk shape som en kube. Noderne kan arrangeres i en hierarkisk struktur, ved hjælp af særlige groupnodes. Det totale sæt af noder i en scene kaldes for en **scene graph**. I alt indeholder VRML sproget 36 forskellige nodes til at opbygge scener med. De kan inddrages i følgende grupper:

- Shape - specificerer geometriske former
- Property - specificerer egenskaber som farve, overflademateriale m.m. for andre noder
- Transform - specificerer flytning, rotation, skalering m.m. for andre noder
- Camera - specificerer synsfelter (kameraer) i scenen
- Lights - specificerer forskellige typer lyskilder i scenen
- Group - sammenbinding af noder.
- WWWInline - indlejring af andre VRML filer.

Shapes gruppen består bl.a. af en række geometriske former som kube, kegle, cylinder, sfære m.fl., men rummer derudover også noder til at generere 3D tekst og vilkårligt komplekse geometriske shapes, som f.eks. landskaber, planter og menneskekroppe.

Property indeholder først og fremmest funktioner der fastsætter *properties* som overflademateriale, farve, skrifttype og textures for formene. En ellers kedelig sofa kan betrækkes med et spraglet blomsterbetræk, vægge kan gøres gennemsigtige, fotografier af bedstemor og bedstefar kan dekorere væggene osv.

Transform nodes indeholder bl.a. funktioner til at flytte, rotere og skalere shapes, lights og cameras.

Camera anvendes bl.a. til at styre opsætningen af et eller flere foruddefinerede synsfelter, kaldet cameras, i scenen. En scene kan have indlagt vilkårligt mange cameras, som brugeren kan vælge imellem. Kameraets position og orientering bestemmer, hvilket udsnit af scenen brugeren ser. En scene kan f.eks. indeholde et rum med et skulptur. Udover det primære camera kan være indlagt et alternativt camera, som altid kigger direkte på skulpturen. Ønsker man i en håndvending et close-up af skulpturen, kan man i stedet for at bruge tid på at manøvrere det primære camera hen foran skulpturen, vælge det alternative camera, der kigger på skulpturen.

Lights findes i VRML i tre forskellige udgaver: Punktlis, retningsbestemt lys og spotlights. **Punkt lys** kan sammenlignes med en sol. Lyset udgår fra et bestemt punkt i alle retninger. Punkt lys har en position men ingen retning. **Retningsbestemte lyskilder** kaster derimod deres lys i en bestemt retning. De har ikke nogen position, men belyser scenen fra et punkt uendeligt langt borte. **Spotlights** fungerer som en teaterprojektør. De har en position som punkt lys og en retning som retningsbestemt lys. Men hvor retningsbestemt lys lyser uendeligt, aftager intensiteten af spotlighten med afstanden fra lyskilden. Endvidere har en spotlight, som en teaterprojektør, et fokusområde. Dvs. at lyskilden kun belyser et afgrænset område foran sig.

Group bruges til at markere grupper af noder. En af de mest brugte sprogenheder er separator-noden. Den binder nodes sammen, således at f.eks. en kube og de handler der skal foretages på den ikke bliver blandet sammen med andre (grupper af) noder. En anden meget vigtig node i denne gruppe er WWWAnchor, der anvendes til at relatere en shape i en scene til en anden scene på internettet. Klikker man på en shape med musepointeren, bliver man transporteret til den scene, den er relateret til. Dette muliggør, at man kan hoppe rundt på internettet gennem et net af forbundne scener. En shape kan også relateres til andet end scener, f.eks. HTML dokumenter, lydfiler, videosekvenser osv. Klikker man på en shape, der er relateret til et HTML dokument, vil VRML browseren starte en WWW browser op, som f.eks. Netscape eller Mosaic med det HTML dokument, som shapen er relateret til.

WWWInline noden muliggør, at man kan sammensætte en scene bestående af vilkårligt mange andre scener placeret på nettet. Man kan forestille sig en simpel scene indeholdende et rum med en sofa og en lænestol. Basisscenen, der indeholder selve rummet, kan ligge på én server, mens sofaen og lænestolen f.eks. kunne ligge på andre servere. Måden man definerer denne distribuerede scene på, er ved at erklære sofa og lænestol som WWWinline referencer i basisfilen. I praksis betyder det, at man i basisscene-filen angiver, at på position x,y,z skal en scene fra andre servere indsættes. VRML browseren henter sofa- og lænestolscenen fra de andre

servere, som den indsætter i basisscenen. Browsers render⁵ basis-scenen først, og inline-scenerne efterhånden som de hentes ind.

1.3.2 Node syntaks

En node består af følgende bestanddele:

- **Objekttype.** Som tidligere beskrevet kan en node være en shape, en transformation eller lignende.
- **Parametre** som adskiller noden fra noder af samme type. Det kan for eksempel være angivelse af radius for en sphere-node, eller forskellige teksturer til de enkelte noder. Parametrene hedder *fields*. Der kan være mellem 0 og flere fields pr. node.
- **Et navn som identificerer noden.** Det er ikke påkrævet at en node har et navn, men det kan være praktisk, idet det bliver muligt for hjælpeapplikationer at identificere indholdet af en scene med stor nøjagtighed. Navne behøver ikke at være unikke.
- **Child nodes.** Det er muligt at opbygget et objekthieraki ved at lade noder indeholde andre noder. *Parent nodes* traverserer deres børn når scenen bliver renderet. Noder med tilhørende child nodes betegnes som *group nodes*.

Syntaksen for en node er ganske ligefrem:

```
objectname objecttype { fields children }
```

De eneste elementer som skal indgå, er `objecttype` og `{}`. Alle øvrige elementer er optionelle. Se kodeeksempel i næste afsnit for hvorledes en typisk gruppe af noder fungerer sammen.

Der findes 16 forskellige typer *fields*, og grundlæggende kan de deles op i to klasser. Fields som kan indeholde en enkelt værdi (alle navne begynder med SF), og fields som kan indeholde multiple værdier (alle navne begynder med MF).

Alle noder kan selvfølgelig ikke tage alle fields som argumenter. Vi vil alligevel beskrive alle SF-fields over een kam. Først og fremmest findes en række datatyper, der er kendt fra normale programmeringssprog, og som i sig selv bør være selvforklarende. Det drejer sig om typerne boolean, enum, flag, float, long int og string. Hertil kommer en række 3d-grafik relaterede typer, som 2D vektor, 3D vektor, farve, 2D pixel-image og rotation.

1.3.3 Scene graphs og noders virkefelt

Som nævnt ovenfor er alle noder i en scene arrangeret i en scene graph. Den holder styr på rækkefølgen af noderne, samt noders virkefelt. For property nodes kan man tale om et virkefelt. Virkefeltet bestemmer hvilke andre nodes, der influeres af property noden. Generelt gælder det, at noder aldrig kan influere tidligere noder, kun efterfølgende. Endvidere kan en node kun påvirke noder på samme niveau som sig selv, eller noder der befinder sig under den selv i nodehierakiet. Nedenfor er vist et simpelt VRML program, der illustrerer denne tankegang:

⁵ Rendering vil sige at tegne et billede af en 3D model

```

#VRML V1.0 ascii

material      { # Påvirker alle efterfølgende nodes
                diffuseColor 1 1 0 # Farv efterfølgende shapes gule
            }
separator     { # Grupperer nedenstående 2 noder som en samlet node.

                # Position i scene for sfære
                Translation {translation 0 4 0 } #påvirker kun efterflg. sfære
                sphere {
                    radius 10}
            }
Cylinder      { # Farves gul, men placeres ikke samme sted som sfære
                radius 5}

```

Første linie fortæller browseren, at der er tale om en VRML fil og at det er version 1.0. Næste linie er en kommentar. Alle kommentarer indledes med #. Da material noden er angivet allerførst og på øverste niveau i nodehierakiet, får den effekt for alle efterfølgende noder. Dvs., at alle efterfølgende shapes farves gule. Separator noden grupperer to noder i samme gruppe: En property node (translation) og en shape node (sphere). Separator noden flytter disse to noder et niveau ned i nodehierakiet, således at noder højere oppe i hierakiet ikke påvirkes af dem. Dvs. at sfærens placering i scenen, der fastsættes med translation noden ikke påvirker kubens, der defineres senere i filen.

1.4 VRML browsere

Der findes efterhånden et pænt udbud af VRML browsere på markedet. Disse kan overordnet placeres i tre grupper: 1) De, der fungerer som hjælpeapplikationer til HTML browsere, 2) De, der fungerer som stand-alone browsere og 3) De, der kører som indlejrede applikationer i HTML browsere. Fælles for disse er, ligesom i HTML browsere, at de har en række funktionsknapper øverst på skærbilledet, der springer en scene frem/tilbage, kalder home-scenen, reloader en scene eller stopper genereringen af en scene.

Browsere er på nuværende tidspunkt (december 1995) oftest beta-releases, og kan hentes rundt omkring på forskellige WWW-sites. Nedenfor vil et lille, men ret repræsentativt, udsnit af de mest udbredte VRML browsere blive beskrevet. Silicon Graphics / Template Graphics Software's "WebSpace" tilhører hjælpeapplikationsgruppen, Intervista's "WorldView" er en stand-alone-browser og Paper Software's "WebFX" fungerer som en VRML browser indlejret i en HTML browser.

1.4.1 WebSpace

Silicon Graphics / Template Graphics Software's WebSpace⁶ er en hjælpeapplikation, der kaldes fra en HTML browser, når denne støder på en VRML fil.

For at ens HTML browser, f.eks. NetScape, kan starte WebSpace korrekt op, kræver det, at WebSpace konfigureres i NetScape som en hjælpeapplikation, på samme måde som andre filformater, der ikke understøttes af HTML browseren selv, skal videresendes til en hjælpeapplikation. Der skal anføres en sti til WebSpace, samt MIME-type (x-world/x-vrml) og file-extension for VRML filer (wrl). I forhold til hjælpeapplikationer som f.eks. en grafik-viewer, så er VRML hjælpe-browsere, herunder også WebSpace, dog mere avancerede end disse. De

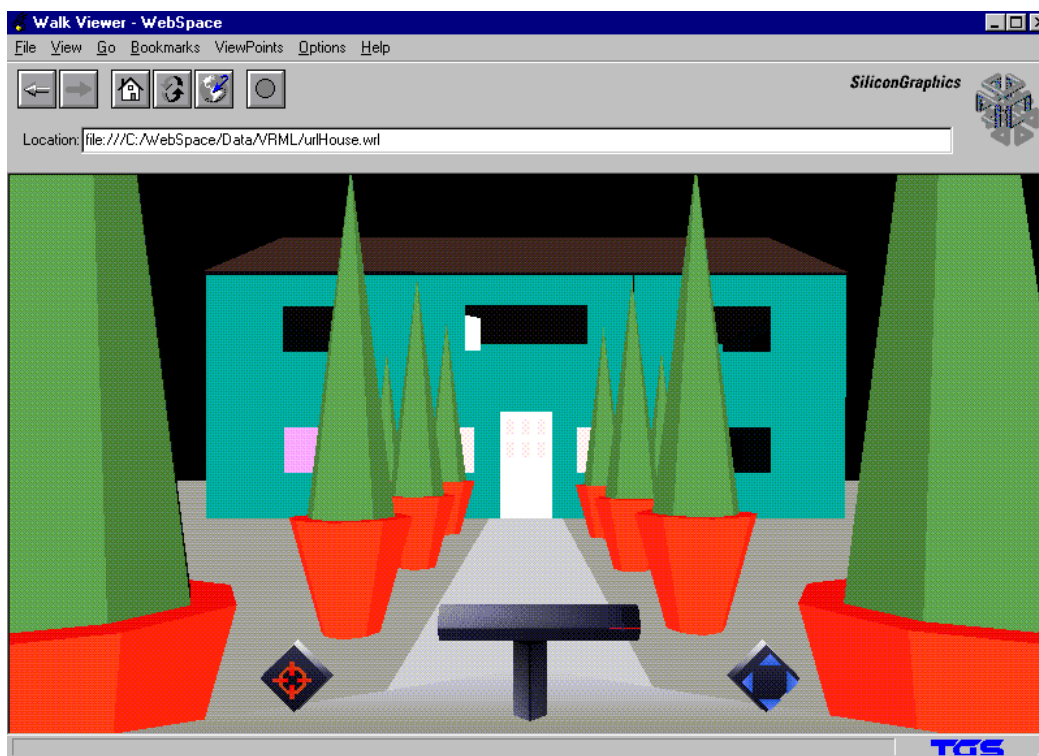
⁶ WebSpace kan findes på <http://webpace.sgi.com>

modtager ikke bare informationer fra, men sender også informationer til Web-browseren, når brugeren i VRML scenen klikker på en shape, der peger på en ny scene.

VRML browsere har, ligesom alle de velkendte HTML browsere, en statusline. Fører man muse-pointeren henover et objekt, der peger videre på en anden scene, da vises dennes URL i statuslinien. Iøvrigt kan man i VRML, i modsætning til i HTML, knytte yderligere tekst til URL'erne, således at der i statuslinien kan fortælles mere om, hvad denne URL kan byde på. Klikker man herefter på objektet, transporteres man hen til den angivne URL. Peger URL'en på et HTML dokument, ryger man således fra WebSpace og tilbage til sin 'egentlige' browser, f.eks. Netscape Navigator.

WebSpace har to navigationsformer, der anvendes i forskellige situationer under ens surfen rundt i VRML scenerne, nemlig Walkmode og Examinemode. I Walkmode bevæger man sig selv rundt i scenen; i Examinemode bevæger/drejer man et objekt rundt i scenen.

I Walkmode, som anvendes i scener med flere objekter, er der på instrumentbrættet nederst i synsfeltet en styrepind, som man kan gribe fat i med muse-pilen. Frem- og tilbage-bevægelse og drejen mod højre og venstre styres med denne. Styrepinden bevæger sig, når man trækker i den. Vil man op/ned/højre/venstre uden at 'køre frem eller tilbage', bruges en pileknap hertil. Skal synsfeltet tippes opad eller nedad, gøres dette ved hjælp af en slags tippe-hjul, og ønsker man at flyve direkte hen mod et objekt i scenen, klikker man først på en 'spring-til'-knap og derefter på objektet. Herved bringes man halvvejs frem til objektet.



Figur A: Webspace i Walkmode

Examinemode anvendes udelukkende i scener, hvor der kun er et objekt. I en sådan scene skal man typisk ikke navigere rundt, men snarere undersøge objektet nærmere. Dette kan gøres ved hjælp af instrumentbrættets kugle, der styrer objektet. Ligesom man i walkmode greb fat i

styre-pinden med muse-pilen, tager man i examinemode fat i kuglen med musen og dreje den rundt for at bevæge objektet, man undersøger. Man kan zoome ud fra og ind på objektet med en drejeknap. Examinemode har som walkmode også en 'spring-til'-knap og en pile-knap til op/ned/højre/venstre-bevægelse.

1.4.2 WorldView

Intervista's WorldView⁷ er en såkaldt stand-alone VRML browser. Den kan altså fungere alene, uden tilknytning til en HTML browser. Bevæger man sig rundt i VRML scener og klikker på en URL til et HTML dokument, så er det dog nødvendigt, at man har sat WorldView op til at kalde en HTML browser, til at vise dette HTML dokument. På samme måde skal man have sat WorldView op til at kalde sine egne hjælpeapplikationer, når der er brug herfor, f.eks til afspilning af lydfile eller visning af videosekvenser.

Der er også mulighed for at sætte WorldView til at render i tre detaljeringsgrader. På en lynhurtig PC vil man oftest, også i store, tunge scener, køre med høj detaljeringsgrad. Men på en mindre kraftig PC er det en stor fordel, at man kan sætte detaljeringsgraden til low eller kun generere objekterne vha. wire-frames.

WorldView har to navigationsformer, der iøvrigt ligger meget tæt på WebSpace's. De kaldes World View og Model View. World View svarer stort set til WebSpace's Walkmode, hvor man selv bevæger sig omkring. Instrumentbrættet med styrepind er erstattet af et vindue med forskellige knapper med pile på. Klikker man på disse med muse-pilen, bevæges eller drejes man i de respektive retninger, som knappens pile angiver. Vinduet med knapperne kan iøvrigt placeres på skærmen, hvor det passer en bedst.

Model View fungerer ligesom Webspace's Examinemode. Knap-panelet man anvender, er det samme som i World View, men i stedet for selv at bevæge sig rundt ved brugen heraf, er det objektet, der bevæges rundt.

1.4.3 WebFX

Paper Software's WebFX⁸ fungerer totalt indlejret i f.eks. QMosaic eller NetScape Navigator. Der kræves ingen opsætning mellem WebFX og HTML browseren, og man ser kun én browser, der så at sige har én fod i HTML og én i VRML.

I WebFX er der ikke en skarp opdeling mellem de to navigeringsmodes beskrevet i både WebSpace og WorldView. Man trykker blot på højre museknap for at flytte et objekt (examinemode) og på venstre for at flytte sig selv (walkmode).

På det seneste er flere VRML browser kommet i stand-alone versioner. Silicon Graphics' WebSpace i stand-alone version hedder WebSpace Navigator og Paper Software's WebFX hedder WebFX Explorer.

Vores indtryk af de tre browsere er, at navigationen med WebSpaces kontrolpind er meget intuitiv, mens Worldviews kontrolpanel er utroligt irriterende og langsommeligt at anvende. WebFX's brugergrænseflade er ganske udemærket. Mht. hastighed så er Webspace langsomt,

⁷ WorldView kan findes på <http://www.webmaster.com:80/vrml/wvwin32s/>

⁸ WebFX kan findes på <http://www.paperinc.com/>

men til gengæld den som render scener mest korrekt, mens WebFX er den hurtigste af de tre browsere.



Figur B: WebFX med kube med levende billeder

1.5 VRML editorer

VRML editorer anvendes til at konstruere VRML scener. Der findes pt. omkring 10-15 dedikerede VRML editorer, hvoraf de fleste endnu er betaversioner. Editorerne tilbyder mere eller mindre avancerede metoder til at opbygge scener. Nogen editorer har specialiseret sig i skabe de overordnede rammer for scener. Disse er velegnede til at konstruere rum, bygninger, landskaber osv. Andre editorer minder mere om en gammeldags 3D modelleringspakke og er mere velegnede til at konstruere detaljerne i VRML scener. Men fælles for dem alle er selvfølgelig, at de kan tilknytte forme i en VRML scene til andre VRML scener, HTML dokumenter osv.

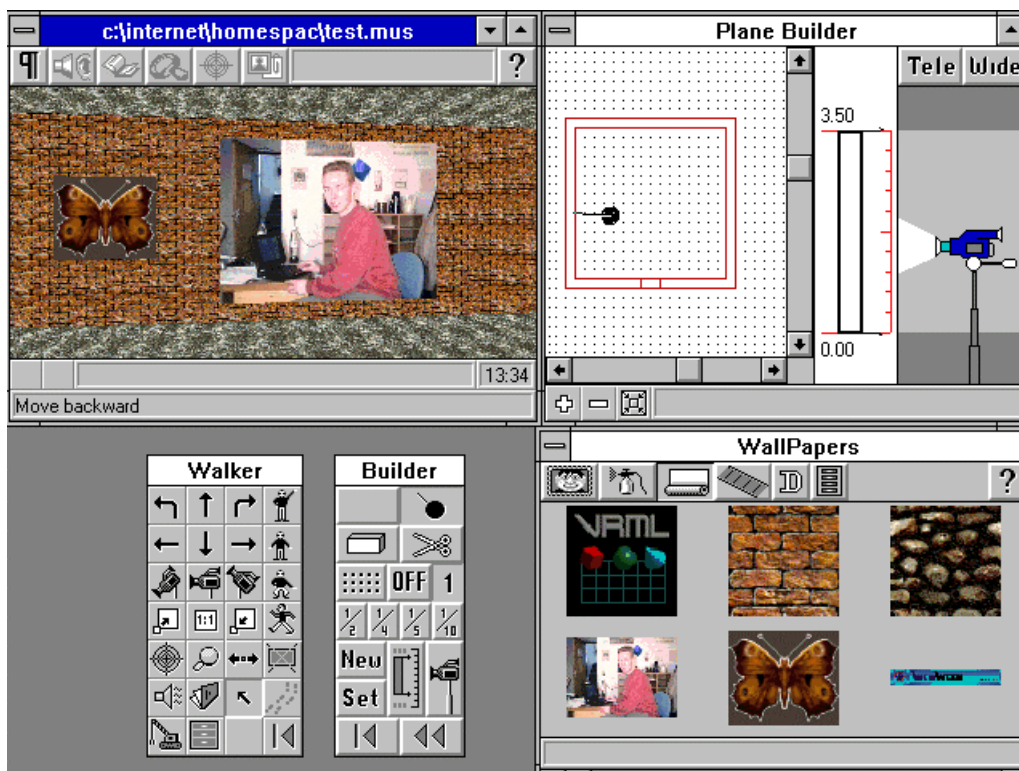
Hvor VRML browsere ofte er gratis, så koster mange af editorene penge, omend prisen er forholdsvis lav. Typisk omkring US \$500 for en licens. En kommerciel version af VRML editoren Virtual Home Space Builder sælges f.eks. for US \$495.

Vi vil i nedenstående beskrive to eksempler på VRML editorer. Den ene, Virtual Home Space Builder, er et typisk eksempel på en editor, der primært er anvendelig til at skabe de overordnede scene rammer, mens den anden, Fountain, kan anvendes til at skabe mere detaljerede geometriske shapes.

1.5.1 Virtual Home Space Builder

Virtual Home Space builder (herefter VHSB) er ifølge Paragraph⁹, der har udviklet programmet, et værktøj der tillader almindelige mennesker at konstruere deres eget hjem i Cyberspace. Dette skal nok tages med et gran salt, men ihvertfald kan man med programmet forholdsvis nemt konstruere scener, der indeholder rum og bygninger. Rummene kan man dekorere med vægtapeter og vægmalerier, men derudover har man ikke andre muligheder for at indrette dem. Man kan f.eks. ikke placere en stol i rummene eller træer foran bygninger osv. Det sætter selvfølgelig nogen begrænsninger for programmets anvendelsesmuligheder. Paragraph lover dog, at man i fremtidige versioner også kan importere 3D objekter fra andre pakker. Dette vil muliggøre at man også kan møblere de konstruerede rum. Men indtil da, så er VHSB's ubetingede styrke, at man meget hurtigt kan fremstille grundkonstruktioner som rum og bygninger i scener. Derfor anvendes programmet ofte i kombination med andre editorer. Først skabes bygninger og rum med VHSB og derefter udfyldes de med mere avancerede 3D shapes, som sofaer, stole osv. i andre VRML editorer.

Programmet er helt klart rettet mod ikke-programmører, idet konstruktionen af rum og bygninger sker i en overskueligt grafisk brugergrænseflade (se nedenstående skærbillede).



Figur C: VRML editoren Virtual Home Space Builder

Øverst til venstre ses **viewer** vinduet, der viser scenen man er i gang med at konstruere. Øverst til højre ses **plane builder** vinduet, hvor i bygningskonstruktionen foregår. Konstruktionen foregår ved, at man først tegner bygningens fundament i to dimensioner og derefter angiver højden på væggene i bygningen. I nederste højre hjørne ses **chooser** vinduet hvor i man kan vælge mellem en række tapeter og billeder til at dekorere vægge med. VHSB under-

⁹ Virtual Home Space Builder kan findes på <http://www.paragraph.com>

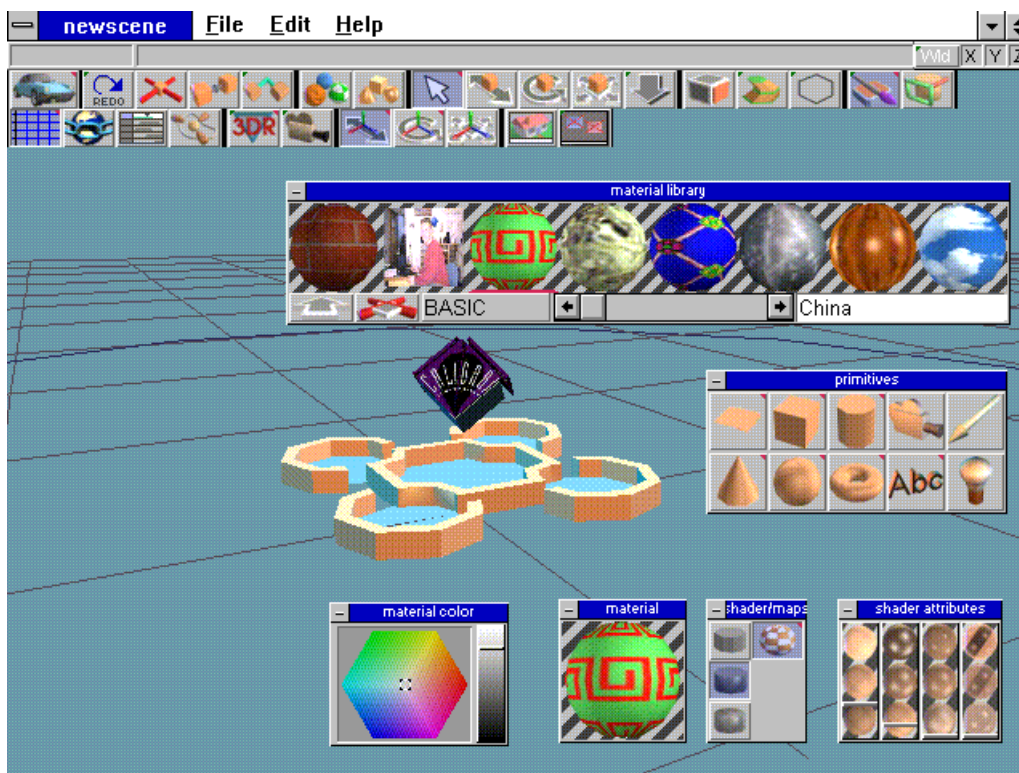
støtter en lang række filformater, så man kan bruge andre billeder og textures end programets egne. Udover de beskrevne vinduer har man to toolbars til sin rådighed: En **walker toolbar**, der anvendes til at navigere i scenen og en **builder toolbar**, der indeholder en række værktøjer til at bygge scenen med.

Billeder på væggene kan meget let linkes til andre scener, ved at klikke på billedet og derefter indtaste en URL. Den kan så pege på andre scener, HTML dokumenter, lydfiler eller noget helt fjerde.

Vi har anvendt VHSB til at konstruere en simpel VRML scene (se afnit 1.6) og må konstatere at programmet er nemt at gå til, omend det lader noget tilbage at ønske i funktionalitet. Eller sagt med andre ord: Det som VHSB kan, kan det godt. Til gengæld kan det ikke så meget.

1.5.2 Fountain

Hvad VHSB mangler af funktioner, har Fountain¹⁰ til gengæld i overflod. Udover almindelige geometriske shapes som kube, sfære, cylinder, pyramide osv. kan man skabe 3D tekst objekter i alle TrueType fonte, opsætte lyskilder, kameravinkler o.m.a. Endvidere kan geometriske shapes beklædes med mange forskellige typer materialer. Materiale egenskaber som transparens og lysintensitet kan justeres og endelig kan man importere shapes fra andre værktøjer. Alt ialt finder man mange af de faciliteter, som man normalt forbinder med 3D modelleringsværktøjer som f.eks. 3D studio og AutoCad.



Figur D: VRML editoren Fountain

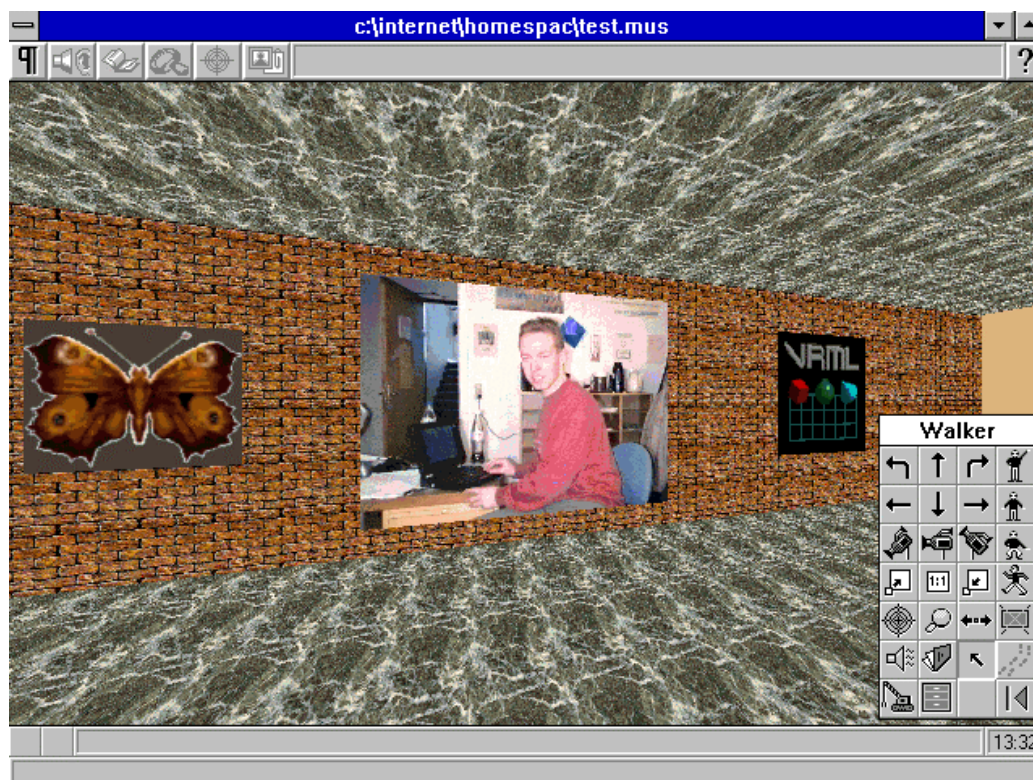
¹⁰ Fountain kan findes på <http://www.caligari.com/1vltwo/2homebld.html>

Det eneste, som adskiller Fountain fra rigtige 3D modelleringsværktøjer er muligheden for at tilknytte shapes til andre scener. Som i VHSB foregår det ved at klikke på shapen og derefter indtaste URL'en for den scene shapen skal relateres til.

Selv om Fountain kan langt flere ting end VHSB, så har VHSB dog et væsentligt fortrin. Det er nemt at anvende. Man når hurtigt frem til et synligt resultat. Forskellen mellem de to svarer lidt til forskellen mellem et 4. generations databasesprog og et 3. generationssprog som f.eks. Pascal.

1.6 VRML testscene

Projektgruppen har lavet en VRML testscene for der igennem at opnå den mest basale erfaring med konstruktion af VRML scener. Scenen indeholder en enkelt bygning og en sfære pålagt et månetexture placeret foran bygningen. Sfæren er linket til BigMac projektets HTML home-page. Man kan afprøve dette ved at dobbeltklikke (evt. enkeltklik alt afhængig af den anvendte browser) på sfæren. I bygningen hænger der på væggene en række billeder med links til projektdeltagernes homepages. De kan selvfølgelig også afprøves ved at dobbeltklikke (evt. enkeltklikke) på billederne. Endelig kan man ved at klikke på skiltet ved indgangen til bygningen sende en kommentar til projektgruppen. Se nedenstående figur 5 og 6.



Figur E: Testscene lavet i Virtual Home Space Builder

Scenen er konstrueret ved først at skabe grundkonstruktionen i Virtual Home Space Builder og derefter at håndkode mindre tilføjelser i koden. Som allerede nævnt understøtter VHSB ikke geometriske shapes som f.eks. en sfære, kuber, cylindre osv. Derfor er sfæren indsat manuelt i VRML koden. Koden til sfæren kan ses nedenfor:

```

WWWAnchor{
  name "http://www.econ.cbs.dk/webkursus/projects/BigMac/index.html"

  Separator {
    Texture2{
      filename "moon.gif"
      model DECAL
    }

    Translation { translation 3 0 1 } #Placering i scenen
    Sphere { radius 300 }
  }
}

```

WWWAnchor noden angiver, at den efterfølgende text i {} definerer et link til en URL, i dette tilfælde BigMac projektets HTML homepage. I name attributten i linie to angives URL'en. Herefter følger den shape, som skal vises i scenen, i dette tilfælde en sfære. Først defineres det ved hjælp af texture2 noden, at sfæren skal have pålagt et texture. Derefter placeres den ved hjælp af translation noden og endelig defineres selve sfæren med en radius 300. Den komplette VRML sourcekode til testscenen kan findes i bilag 1.



Figur F: Testscenen renderet af WebFX

Testscenen kan findes på følgende URL:

<http://www.econ..cbs.dk/webkursus/projects/BigMac/scene/bigmac.wrl>

Testscenen er afprøvet med en række forskellige browsere, hvoraf WebFX fra Paper absolut er den, som har givet det bedste resultat (se figur 6). Med andre browsere kan man risikere ikke at kunne se textures på væggene, ligesom farverne kan kikse.

1.7 Virtual Reality analyse af VRML

Efter selv et kort studie af VRML finder man hurtigt ud af, at det er forholdsvis simpelt og unuanceret sprog. I dette afsnit vil vi analysere VRML sproget ud fra en generel Virtual Reality synsvinkel, for der igennem at afdække svagheder ved VRML i dets nuværende udformning, dvs version 1.0. Sammenligner man VRML miljøet med almindelige Virtual Reality miljøer kan man umiddelbart konstatere at der er en række Virtual Reality faciliteter, der mangler i VRML miljøer. Disse kan inddeles efter hvor teknologisk vanskelige de er at indlægge i VRML miljøer. De teknologisk mindre vanskelige er:

- Navigation ved hjælp af hoved og kropsbevægelse
- Visualisering af scener i HMD'er (Head Mounted Displays)
- 2D animation

Dertil kommer en række faciliteter, der i modsætning til ovenstående, er langt mere krævende at indlægge i VRML miljøer.

- Manipulation/bevægelse af objekter
- Collision detection
- 3D animation
- Autonome objekter
- Flere samtidige deltagere i scener
- Audio
- Andre sanser

Når vi i det nedenstående diskuterer mangler i VRML miljøer, skal man være opmærksom at mangler både kan skyldes VRML sproget og VRML browsere. Vi vil derfor for hver mangel specificere om det vedrører sproget, browseren eller begge.

1.7.1 Navigation ved hjælp af hoved og kropsbevægelse

Navigation i Virtual Reality foregår ideelt set ved hjælp af hoved og kropsbevægelser. Hoved og kropsbevægelser bestemmer hvad brugeren ser i den virtuelle verden. Ingen af de tilgængelige VRML browsere understøtter pt. tracking af kropsbevægelser. Navigation foregår udelukkende ved hjælp af en almindelig 2D mus. Det gør at navigationen bliver mindre intuitiv. Dette er en mangel ved VRML browserne og ikke VRML sproget.

1.7.2 Visualisering af scener i Head-mounted-displays

Egentlig er der ikke noget til hinder for at anvende HMD'er i VRML miljøer. Men fordi der er ikke er nogen browsere, der understøtter navigation ved hjælp af hovedbevægelser, anvendes HMD'er stort set ikke i VRML miljøer¹¹. VRML scener kan derfor ikke opleves som ægte 3D visualisering. Væsentligste problem herved er, at hjernens evne til at danne dybdeperspektiv, ved hjælp af stereoskopisk syn, ikke udnyttes fordi afbildningen ikke sker i HMD'er med separate billedkilder til hvert øje. Dette gør f.eks. interaktion med objekter meget besværlig, fordi man ikke præcist kan fornemme, hvor langt borte et objekt er. Dette problem er egentlig afledt af det forrige og kan derfor karakteriseres som et browser problem.

¹¹ Problemet er, at selv om brugeren kan anvende et HMD til at se scenen, så kan navigation i scenen ikke ske ved hovedbevægelser, men kun gennem f.eks. en 2D mus, som er svær at styre med et totalt omsluttende HMD.

1.7.3 2D animation

Med 2D animation menes egentlig levende billeder som f.eks. video og TV. Idag understøtter VRML, som tidligere beskrevet anvendelsen af textures i scener. En mur kan f.eks. pålægges et murstens texture. Men ideelt set bør man også kunne pålægge levende billeder (dynamiske textures) på shapes i scenen. Man kan forestille sig at et TV i en scene, hvor der kører en rigtig TV kanals udsendelser på skærmen. Der findes allerede enkelte eksempler på lignende scener. WebFX browseren leveres med en demoscene der viser en kube, hvorpå der på siderne afspilles Michael Jackson's video "Black or White" (se figur 2). Men endnu understøtter VRML sproget ikke officielt 2D animation.

1.7.4 Manipulation/bevægelse af objekter

Man kan ikke i VRML, som man ellers ofte ser i Virtual Reality miljøer, tage fat om, hive i eller skubbe et objekt. Et af Virtual Realitys varetegn er en virtuel hånd der kan bevæge sig rundt i virtuelle verdener og gribe fat i objekter. Den virtuelle hånd styres typisk af brugeren ved hjælp af en såkaldt dataglove, der registrerer personens fysiske håndbevægelser, videretransmitterer dem til computeren som omdanner de fysiske bevægelser til virtuelle handlinger. Men den mulighed har man ikke i VRML scener og det skyldes at objekters position og orientering i scenen kan ikke ændres. Sidstnævnte begrænsning medfører en anden væsentlig mangel, nemlig at objekter ikke kan bevæge sig selvstændigt rundt i scenen. Har man en VRML scene, der viser en by, kan man altså ikke have biler der kører på vejene, mennesker der går på fortovene osv. Denne mangel gør at mange VRML scener virker en smule spøgelsesagtige. Manglende mulighed for at bevægelse af objekter i scenen er et VRML sprog problem.

1.7.5 3D animation

Hvor forrige punkt vedrører bevægelse af objekter, så handler 3D animation om at objekter i en scene ændrer form. Et primitivt eksempel er en sfære der vibrerer. Et mere avanceret eksempel kan være animerede virtuelle personer, der "går" rundt i en scene. Dvs. ikke bare flytter sig rundt, men bevæger ben og arme, som ved rigtig gang. Understøttelse af 3D animation er primært et VRML sprogproblem.

1.7.6 Collision detection

Collision detection sørger for, at brugeren ikke kan gå igennem objekter, som vægge, møbler osv. VRML understøtter ikke collision detection. Selv om collision detection ikke er kritisk for illusionen, er det dog forstyrrende, at få lov til at gå gennem mure/vægge. Collision detection skal implementeres i VRML sproget.

1.7.7 Autonome objekter

Autonome objekter er objekter i scenen, der har en selvstændig adfærd. Autonome objekter forudsætter at man kan specificere for det enkelte objekt hvordan det skal reagere på hændelser i omgivelserne. Et eksempel kan være en virtuel bankboks, der kun åbner hvis man indtaster den rigtige kode på display'et placeret på boksens front. Simulering af autonome objekter er et VRML sprog problem.

1.7.8 Flere samtidige deltagere i scener

Der er ingen mulighed for at interagere med andre mennesker i VRML scener. Hermed afskærer man sig helt fra det sociale aspekt ved Virtual Reality. Ideelt set burde man kunne

møde og kommunikere med andre personer i VRML scener. Dette er specielt interessant i internet sammenhæng, da VRML kunne være fundamentet for et helt ny kommunikationsform mellem mennesker over hele kloden. Samtidige deltagere i en scene er et VRML sprog problem.

1.7.9 Audio

Lyd understøttes ikke af VRML, hvilket må siges at være en væsentlig mangel. Lyd tilskrives stadig større betydning for oplevelsen af en virtuel verden. Lyd har endvidere det fremtrin for den visuelle afbildning at det er en 360 graders sansning, mens synssansen kun opfanger 180-200 grader horisontalt og omkring 60-80 grader vertikalt af omgivelserne. Lyd er derfor et vigtigt supplement til synssansen, der ikke bør ignoreres. Audio kræver et helt nyt sæt af noder indlagt i VRML sproget.

1.7.10 Andre sanser

Følesansen og den kinestetiske sans understøttes heller ikke i VRML. Disse er mindre kritiske for illusionsopfattelsen end høresansen og synssansen. Der er generelt ikke mange Virtual Reality miljøer der understøtter disse, hvorfor det er tvivlsomt om VRML nogensinde kommer til det.

1.7.11 Konklusion

Sammenligner man VRML med almindelig Virtual Reality, kan man helt overordnet konstatere at man med VRML i første omgang har koncentreret sig om det helt basale, nemlig den visuelle afbildning af scener. Ambitionsniveauet er med vilje lagt lavt i den første version for hurtigst muligt at få vedtaget den første version. Det betyder at en lang række ellers almindelige Virtual Reality faciliteter endnu ikke er understøttet.

VRML scener fremstår som statiske, døde verdener sammenlignet med virtuelle verdener kendt fra andre Virtual Reality miljøer. Det skyldes primært at objekter ikke kan bevæge sig, at man ikke kan manipulere med objekter, at man kan gå gennem mure og at man ikke kan møde andre personer i VRML scener. Man kan betragte VRML scener men ikke ændre i dem. Eneste mulighed for interaktion er at bevæge sig rundt og klikke på links til andre scener.

Denne fatale mangel ved sproget er man selvfølgelig klar over og man arbejder på at udbedre ovennævnte mangler i fremtidige versioner. Som nævnt kan manglerne grupperes i faciliteter der er vanskelige og faciliteter der er mindre vanskelige at indlægge i VRML miljøer. Der er god grund til at formode at de mindre vanskelige indenfor ganske kort tid vil kunne findes i VRML miljøer. De mere vanskelige vil vi vende tilbage til i nedenstående afsnit **Fremtiden for VRML**. Her vil vi gennemgå forskellige løsningsteknikker, der i øjeblikket diskuteres i VRML miljøet.

1.8 Fremtiden for VRML

Som tidligere nævnt arbejdes der i øjeblikket på at få udbygget antallet af funktioner i VRML. Arbejdet foregår ved, at enkeltpersoner eller virksomheder udarbejder forslag til, hvad de mener der skal tillægges VRML i version 2.0. Forslagene ligger offentligt tilgængelige på Inter-

net¹², og de samles af VAG - The VRML Architecture Group. Der afholdes en række forberedende tekniske møder i løbet af vinteren 1996, og i april vil gruppen fremkomme med et forslag. Det er derefter muligt at kommentere forslaget, som vil lede frem til en endelig version i maj 1996. En række store virksomheder, som f.eks. Sony og Silicon Graphics, har allerede fundet sammen om et fælles forslag, som det forventes der vil blive skævet meget til.

Der foreligger selvfølgelig en række forslag til at forbedre og udvide antallet de eksisterende 36 noder og deres fields i moderat grad, og f.eks. give mulighed for brug af de mest benyttede grafikformater, herunder også MPEG og AVI. Listen over forslag er meget omfattende, og forslagene er meget varierede. Der er dog en lang række funktioner, der går igen i forslagene.

En af de mest efterspurgte funktioner er undetstøttelse af adfærd. Det dækker over et vidt begreb, men kan måske bedst beskrives ved, at det f.eks. er muligt at ændre udseendet af et objekt dynamisk - også efter at noden er blevet fortolket. De fleste er enige om, at der skal kunne tilknyttes en shape for script til objekter. Med script menes der en programdel, der bliver eksekveret i tilfælde af, at der bliver foretaget en bestemt handling på objektet. I scriptet er det beskrevet hvorledes objektet skal agere, fuldstændig som metoder i objektorienteret sammenhæng. Dette giver også mulighed for, at objektet kan tillægges realistisk fysisk adfærd, som f.eks. tyngde. Også 3D animationer, som f.eks. en kube der kontinuerligt trækker sig sammen og derefter udvider sig, vil kunne implementeres via tilknyttede scripts. Der skal også være funktioner der kan ændre ved scene-stukturen, f.eks. at skabe eller slette noder og flytte objekter rundt på scenen.

En meget ligefrem måde at forestille sig kontrol af kollision mellem objekter er, at et objekt kontrollerer om det er kollideret med et et andet objekt. Det er også den måde, at kollisioner mellem camera-noder og objekter detekteres. Men hvad skal styre, at to objekter undersøger om de er kollideret med hinanden, hvis der er tale om autonome objekter uden tilknytning til hinanden? Man kan f.eks. gøre det ved at definere sensor-noder, der registrerer ændringer i tid, positioner og rum. Man kan tale om en slags radar-noder, der overvåger afgrænsede udsnit af en scene. Meddelelser om hændelser skal kanaliseres videre til de berørte objekter, for at de kan beslutte hvorledes de skal handle. Hertil kan man lave router-noder, der er noder der sender beskeder mellem objekter. At to objekter er kollideret, er blot een af mange besked-typer.

Muligheden for at flere personer deltager i delte scener er umiddelbart muliggjort ved hjælp af de overfor beskrevne teknikker - så længe det da foregår på den samme server. Men hvis al interaktion skulle foregå på een server, eller hvis alle servere skulle vide hvor alt ligger i cyberspace, ville de hurtigt bukke under. Derfor er den såkaldte Cyberspace Protokol blevet skitseret (og forsøgsmæssigt implementeret). Alle servere ved hvor de befinder sig, og har suveræn kontrol over deres lille bid af cyberspace. De ved også en lille smule om hvordan cyberspace ser ud, og ved helt præcist hvem de skal spørge for at få mere at vide, nemlig deres naboservere. En server er aldrig passiv, men vil konstant sørge for at holde sig orienteret om hvorledes cyberspace ser ud på de omkringliggende servere.

¹² Et repræsentativt udsnit kan findes på:

http://www.yahoo.com/Entertainment/Virtual_Reality/Virtual_Reality_Modeling_Language__VRML_/Reference_Material/

Der er også forslag til hvordan lyd kan medtages, og forslagene hertil er meget detaljerede. Her tales der selvfølgelig om at indlemme en række af de etablerede lydformater (f.eks. .wav eller .ra, men også at tilpasse lyden til den tredimensionale verden, og lade den være afhængig af retning og afstand, samt at benytte avancerede akustiske effekter som beregning af ekko fra omkringliggende objekter, afhængigt af deres overflader og bevægelse. Det hele samles under begrebet audio-rendering. En række af forslagene til audio-rendering er ret urealistiske med de nuværende computerressourcer, på samme måde som det vil være urealistisk for de nuværende browsere at danne de ofte meget beregningstunge scener vha. af raytracing¹³.

1.8.1 Krav til et 3D grafiksprog, der understøtter adfærd

Vores beskrivelser af mangler ved VRML, og de forslag til udbyggelser som foreligger, har foregået på et forholdsvist konkret niveau. De har taget deres udgangspunkt i generelle Virtual Reality betragtninger, samt ønsker fra industrien til hvordan man i første omgang kan udvide et simpelt og unuanceret sprog. En række af de nævnte forslag til udvidelser af VRML griber dog fat i uløste problemstillinger, som der er blevet forsket i lang tid før VRML fremkom. Men det viser helt klart, at industrien ser muligheder i VRML, og gerne vil være til at udbredet det.

Vi har afgrænset os fra at sammenligne VRML med andre grafiksprog eller teknikker til at downloade eksekverbare programmer lokalt, som f.eks. det nyligt fremkomne Java fra Sun. Vi finder det dog relevant at afslutte med at analysere VRML i forhold til en række kriterier, som et tredimensionelt grafiksprog med håndtering af adfærd må forventes lever op til, hvis det skal kunne benyttes som en universal Internetstandard, der i sidste ende måske udvikler sig til til en reel virtuel verden - cyberspace. I et af forslagene til fremtidige forbedringer af VRML opstilles der en liste af kriterier:

1. Det skal understøtte hovedparten af de computere der er tilsluttet Internet
2. Have evnen til at downloade sikker kode, der kan eksekveres lokalt
3. Have protokoller for at udføre fjernprocedurekald
4. Understøtte tidskritisk adfærd
5. Understøtte objektorientering, med fleksibel nedarvnings-semantik

ad 1: Dette punkt bør være selvforklarende, men VRML har den bagdel, at det kræver en platformspecifik browser.

ad 2: I de seneste år er der fremkommet nogle forslag til, hvordan man kan downloade kode, der under sikre forhold kan eksekveres i lukkede afsnit af systemet. Det går selvfølgelig ikke, at et fremmed program sletter en fil eller lægger operativsystemet ned. Man kan have det synspunkt, at det i det hele er meningsløst at have en VRML browser, da det browserelement der skal fortolke den downloadede kode lige så godt kan følge med koden! Det er en antagelse, der har rod i de principper, der ligger bag teknikker som Java eller OpenDOC, der helt bryder med den traditionelle opdeling i afgrænsede programmer. Igen kommer VRML lidt til kort her, da det er browseren der har den fulde kontrol over hvorledes koden skal eksekveres, men det er dog muligt at downloade kode.

¹³ Teknik til at danne fotorealistiske grafikbilleder

ad 3: Fjernprocedurekald er nødvendige, for at en handling i een scene (der foregår på een server) der har virkning for et objekt i en anden scene (der foregår på en anden server) kan udføres. Det skal være muligt for to uafhængige servere at vedligeholde hver deres scene. Hvis man f.eks. slår to flasker (der kommer fra hver sin scene på hver sin server) mod hinanden, og de begge knuses, så er det scripts til objekter på to forskellige servere der skal udføres. VRML understøtter ikke fjernprocedurekald, men der arbejdes på at indføre protokoller, hvor scener er opdelt i delelementer.

ad 4: Med understøttelse af tidskritisk adfærd menes der f.eks. synkronisering af objekters adfærd. Forslagene om indførsel af objektscripts og overhændelse af beskeder mellem objekter muliggør dette. Men det kan også dreje sig om tilpasning af begrænsede computerressourcer. Hvor der er et bestemt krav til hvor mange gange en scene skal gentages i sekundet (af hensyn til realismen) kan browseren nøjes med at foretage en simpel flatshading af baggrundsobjekter, medens objekter der er i fokus bliver phong-shadet. De nuværende browsere har dog ikke den beskrevne teknik, eller andre elementer til understøttelse af tidskritisk adfærd.

ad 5: Tilbage står punkt 5. der heller ikke opfyldes. Der er der andre sprog der gør, og for VRMLs vedkommende ser det ud som om der også vil blive gjort noget ved det, idet der foreligger forslag om at gøre VRML objektorienteret. Der tales om muligheden for definition af prototype-noder (der ikke nødvendigvis bliver vist i scenen), hvor fields, scripts eller hele noder kan nedrives. I relation hertil skal det også være muligt at noder f.eks. blokerer hinandens scripts, så de kan synkronisere hinandens adfærd. Objektorientering åbner også mulighed for multimetoder, hvor en samling af objekter i fællesskab afgør hvilken scriptmetode i objekthierarkiet der skal benyttes, således at parametrene i kaldet passer bedst muligt til metoden.

1.8.2 Konklusion

VRML i sin nuværende version lever kun op til meget få af de nævnte kriterier. Det vi har ville vise med dette afsnit er, hvad der skal til, for at man har et sprog hvormed man kan bygge cyberspace-lignende miljøer.

Det er vores vurdering, at VRML indenfor ganske kort tid (1 år fra nu) vil understøtte flere personer i samme scene. Adfærd, lyd og understøttelse af objekter med scripts vil formodentlig dukke op indenfor 2-3 år, mens distribuerede scener (cyberspace protokollen) ligger noget længere ude i fremtiden.

Ovenstående vurdering er baseret på de forslag vi har læst, samt annoncering af produktlanceringer inden for de kommende måneder, som f.eks. VRML+, der er en ikke-standardiseret extension til VRML udviklet af virksomheden bag AlphaWorld (se afsnit 2.2.3). VRML+ understøtter bl.a. flere deltagere i samme scene, samt adfærd.

2. VRML mediet og HTML mediet

I dette afsnit vil vi slippe det teknologiske perspektiv og istedet anlægge en medieorienteret synsvinkel. I afsnit 2.1 og 2.2 vil vi analysere styrker og svagheder for henholdsvis HTML mediet og VRML mediet. I afsnit 2.3 diskuterer vi om det nye medie vil opluge/inkorporere HTML mediet. Endelig vil vi i afsnit 2.4 opstille en række scenarier for hvorledes vi forestiller os at fremtidig anvendelse af de to medier vil foregå.

I nedenstående afsnit forudsætter vi, at de i afsnit 1.7 fundne mangler er blevet udbedret. Vi arbejder således med en idealiseret form af VRML mediet, der først kan påregnes at være en realitet om 2-3 år.

2.1 Styrker ved HTML mediet

Historisk set findes der enorme mængder tekstuel information i verden, og det vil der fortsat findes, også i fremtiden. Der vil også komme ny tekstuel information til. *Megen* tekstuel information. Det er derfor, at HTML har sin berettigelse, og også vil have det i fremtiden. Man kan dog diskutere, hvorvidt HTML er det rette medie til at formidle alle de typer information, som det gør.

Vi benytter os i denne forbindelse af to begreber: *Navigation-information* og *target-information*. Target-information er den information, som er målet for en søgning. Det kan f.eks. være en artikel på HTML format. Navigation-information er det, vi bruger til at navigere rundt på Internet med, for at kunne finde frem til målet for vores informationssøgning. Navigation-information er altså noget vi benytter os af undervejs.

HTML er godt til at vise target-information, som f.eks. tekstuel information. Det er VRML ikke så godt egnet til. Vi vil senere skitsere, hvorledes HTML kan opluges fuldkomment af VRML. Det er dog kun i den yderste konsekvens - et egentligt cyberspace. I praksis er det nok de færreste der kan forestille sig, at man læser dokumenter på en virtuel skærm i VRML. I det mindste ikke så længe det skal foregå på en almindelig pc skærm.

På Internet bliver både navigation-information og target-information vist i det samme medie, HTML. Navigation mellem target-information er struktureret vha. anchors i HTML, og til en vis grad ved brug af URL'er. Den brug er historisk betinget. Man skal se på, hvad man havde, før HTML kom frem. HTML er ulige nemmere at navigere med end f.eks. gopher og IP-numre. Der er måske umiddelbart ikke den store forskel på gopher og hyperlinks, men effekten forstærkes i betydelig grad, hvis man benytter sig af billeder og lyd til at ledsage og markere linkene.

2.2 Styrker ved VRML mediet

2.2.1 Intuitiv navigation i informationsbjerge

Virtual Reality teknologien i almindelighed er af mange spået et stort potentiale som værktøj til at håndtere store informationsmængder, også kaldet informationsbjerge. Der findes pt. en lang række eksempler på Virtual Reality brugergrænseflader til visualisering af informations-

bjerge, hvoraf Information Visualizer projektet på Xerox PARC¹⁴, måske er det mest kendte. Information Visualizer er en Virtual Reality brugergrænseflade, der kan visualisere store mængder hierakiske og lineære, data. Man har bl.a. anvendt det til at visualisere store netværks filstrukturer, store koncerners organisationsstrukturer m.m.

Håndtering af informationsbjerge er specielt interessant i VRML og Internet-sammenhæng, fordi internettet udgør det vel nok største menneskefrembragte informationsbjerg. Internettet kan opfattes som et kæmpe mængde af ustrukturerede abstrakte informationer. Som nævnt i ovenstående anvender man idag HTML dokumenter til at strukturere internettets informationer. Både til at vise navigation-information og target-information.

Der er ingen tvivl om at HTML dokumenter er velegnede til at formidle target-information af tekstuel og visuel karakter. Derimod kan der rejses tvivl om, hvorvidt navigation på internettet er optimal ved brug af hyperlinks. Selv om HTML og hyperlinks var en voldsom forbedring i forhold til eksisterende internetbrugergrænseflader som FTP, Telnet, Gopher m.fl., er der en række ulemper ved at anvende HTML hyperlinks til navigation på internettet:

- Det er svært at overskue en større mængde sammenhængende HTML dokumenter arrangeret i et dokumenthierarki med mange niveauer. Det kan være en lovsamling, en virksomheds homepage eller noget helt tredje. Overblikket er begrænset til det niveau i hierakiet, hvorpå man befinder sig. Man kan ikke se nedad i hierakiet, idet man ikke har mulighed for at forudse hvor megen information, der gemmer sig bag ved et link (bortset fra selvfølgelig ved at følge linket). Er man endvidere hoppet ind på et af de nedre niveauer i et dokumenthierarki, har man det tilsvarende problem med at se, hvad der gemmer sig på øvre niveauer i hierakiet.
- Har man bevæget sig gennem mange HTML dokumenter for at nå frem til noget target-information, er det senere vanskeligt at huske hvordan man fandt informationen. Ganske vist forsøger HTML browserne at lette navigationen ved bl.a. at markere tidligere besøgte links med en iøjnefaldende farve, og man kan da gemme et dokument som et bogmærke, så man altid kan finde tilbage til det igen. Men hvad nu hvis man ikke lige fik markeret dokumentet som et bogmærke, eller man så noget undervejs, som man ikke fik kigget nærmere på. Så er det svært at huske mere end 3-4 klik tilbage i HTML dokumenter.

Vender vi tilbage til VRML, så har dette medie nogle interessante egenskaber i relation til ovennævnte ulemper.

Overblik over store dokumenthierakier kan i VRML f.eks. opnås ved at visualisere et dokumenthieraki i et 3D dokumenttræ á la Information Visualizers kegletræer. Her vises et filsystems filer og biblioteker som klaser af rektangler frit hængende i rummet. Øverst er rodbiblioteket. Fra dette udgår "snore" til andre klaser af filer liggende i niveaueu under roden. Fra disse klasse udgår igen snore til filer længere nede i hierakiet. Zoom funktionen i Information Visualizer gør, at man kan zoome helt ud og dermed får et overblik over samtlige filer, eller helt ind på den enkelte fil, så den udfylder hele synsfeltet. På samme måde kunne man vise HTML dokumenter som dokumentklaser i et dokumenttræ, som man kunne zoome ind og ud på, alt efter om man ønskede overblik eller detalje.

¹⁴ Stuart K. Card m.fl., "Information Visualisation using 3D interactive animation", Communications of the ACM, april 1993, Vol. 36, No. 4: 57-71.

En anden metode til at skabe overblik finder man i det eksisterende VRML produkt VRserver¹⁵, der kan repræsentere en servers filstruktur i en VRML scene. VRServer konstruerer dynamisk gange, værelser, halls, døre og vægmalerier, der tilsammen repræsenterer filstrukturen på serveren (se figur 7). Brugere kan vandre gennem gangene og åbne døre til interessante filrum, der igen indeholder døre til andre filrum. Programmet tillader endvidere serveradministratoren til en vis grad at skræddersy serverens 3D repræsentation til individuelle ønsker. Ideen med at visualisere et filsystem i 3D er også set i produktet Filesystem Navigator fra Silicon Graphics, hvor man kan visualisere filsystemer som bygninger på en slette.



Figur G: VRserver, med filer og kataloger repræsenteret som billeder og døre

Mht. navigation på Internet, så har VRML også her en fordel i forhold til HTML. Vi husker simpelthen 3D rumlige repræsentationer bedre end endeløse rækker af tekstdokumenter. Vi er biologisk hardwired til at agere i et 3D rum, og ikke til at overskue og bevæge os i en abstrakt verden, som et 2D dokumenthierarki vitterligt er. Der er meget stor forskel på, om man finder noget target-information ved at klikke sig gennem 10-20 HTML dokumenter, eller om man går gennem en virtuel by, ind i et rødt hus med et piletræ udenfor, og inde i huset går op ad den snoede trappe til højre i lokalet, for inde i et rum at finde et sæt af dokumenter. Små detaljer, som en bygnings farve eller trædørens metaldørhammer, indprenter sig meget nemmere end store mængder tekst. Vi husker bedre en gåtur end en kliktur!

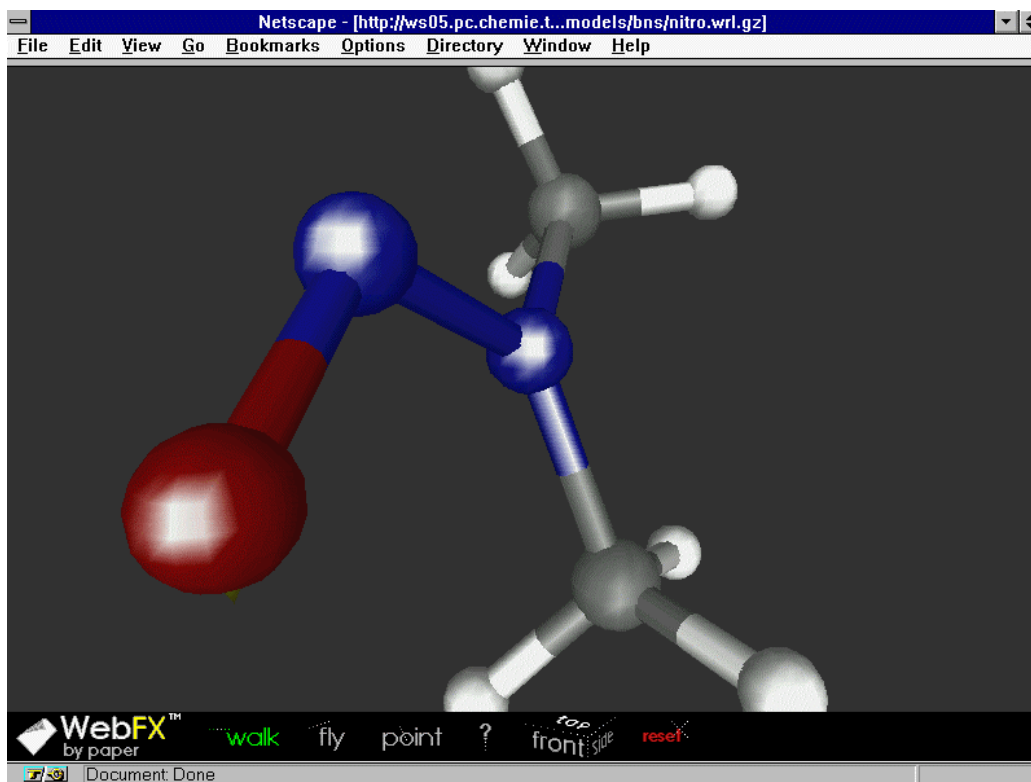
2.2.2 Formidling af 3D informationer

Hvor forrige punkt indebar en fortolkning af abstrakte informationer til rumlige strukturer, så vedrører dette punkt formidling af informationer, der er tre-dimensionelle af natur. Dvs., at

¹⁵ VRServer kan findes på: <http://www.tenet.net>

der ikke sker en fortolkning, men snarere en formidling af information. Nedenfor gives en række eksempler på hvorledes man allerede idag anvender VRML til formidling af 3D information.

Visualisering af 3D datasæt bliver stadig mere anvendt i forskningsmiljøer. Kombinationen af interaktiv 3D grafik og WWW's distributionsmuligheder åbner helt nye perspektiver for videnskabelig publicering. Man kan nu dele ikke bare beskrivende tekst og billeder, men også de faktiske forskningsresultater med omverdenen. På "Institut für Physikalische Chemie" i Darmstadt anvender man VRML til at offentliggøre forskning omkring enzymet Cytochrome P450, som er særlig interessant indenfor kræftforskningen (se figur 8). Der er et stort udvalg af scener med proteiner, aminosyrer og andre molekyler med relation til førnævnte enzym¹⁶.



Figur H: Enzymet Cytochrome P450 illustreret i VRML

Indenfor geologien anvendes VRML til bl.a. at visualisere digitaliserede terrændata. Hos MountainTop::Computing, der har specialiseret sig i geologisk visualisering, finder man en VRML scene med en jordklode¹⁷. Områder på kloden, hvortil der findes digitaliserede terrænmodeller, er markeret med røde firkanter. Klikker man på en rød firkant, får man præsenteret en digital terrænuddgave af det område, som firkanten dækker.

Kommercielle organisationer som Grundig og Intel anvender VRML til præsentation af deres produkter. På Grundig's VRML server kan man se de sidste nye fjernsynsmodeller m.m.¹⁸,

¹⁶ Darmstadts molekyler findes på: <http://ws05.pc.chemie.th-darmstadt.de/vrmlG/>

¹⁷ Terræn jordkloden findes på <http://www.gwha.com/~jym/demWorld.wrl.gz>

¹⁸ Grundig's WWW adresse er <http://www.asia-online.com/grundig>

mens man hos Intel kan finde produktinformation om den nye Pentium Pro processor i en VRML scene, der skal forestille at være et udsnit af den nye processors kredsløb¹⁹.

Arkitekturen er måske et af de områder, hvor VRML vil få den største gennemslagskraft. VRML giver arkitekten mulighed for at præsentere realistiske 3D modeller af potentielle bygningsværker for kollegaer, kunder osv. før byggeriet påbegyndes. Kunder kan i VRML vandre rundt i den virtuelle bygning og får derved et langt bedre beslutningsgrundlag, end de kunne have fået med perspektivtegninger og CAD udskrifter. Hos firmaet Lightscape²⁰ anvender man VRML til at visualisere både eksisterende og ikke-eksisterende bygninger. Her finder man f.eks. en model af Jerusalem City Hall. Desværre fylder den 12.5 MB. Så vi kan desværre ikke bringe et billede af den her.

Indenfor kunstverdenen kan man anvende VRML til at udstille kunstværker i virtuelle museer og gallerier. I Virtual Ceramics Gallery²¹ kan man se både virkelige og virtuelle keramiske værker. I "The Rosetti Room" kan man selv indrette et rum med nogen af historiens mest berømte malerier²². Først bliver man i et HTML dokument præsenteret for en række malerier, som man kan klikke på, hvis man ønsker dem inkluderet i rummet. Man har også begrænsede muligheder for at styre den øvrige indretning af rummet, f.eks. om der skal være en sofa, en bogreol osv. Når man har foretaget de nødvendige valg, kan man igangsætte genereringen af ens helt egen VRML scene og gå rundt i sin eksklusivt udvalgte kunstsamling.

2.2.3 Det sociale aspekt

WWW, som vi kender det idag, er overvejende et statisk read-only medie. Statisk fordi kommunikationen i mediet primært består i at læse skrivebetyttede HTML dokumenter fra nettet. Endvidere er muligheden for kommunikation med andre mennesker begrænset til e-mail og Internet Relay Chat. På WWW støder man ikke på andre mennesker, der impulsivt har mulighed for at gøre, sige eller informere om noget uventet.

At der her-og-nu (on-line) "er nogen i den anden ende", har en enorm betydning for et medies succes. Kevin Hughes²³ kommer med den (ret let-forståelige!) sammenligning, at folk næsten udelukkende kommer på den lokale pub for at snakke, diskutere, spille, hygge sig og i det hele taget være i kontakt med og møde andre mennesker. Og næsten aldrig for at sidde for sig selv og bladrelæse i en masse tekst. WWW mediet er velegnet til søgning af information, men ikke til kommunikation mellem mennesker.

Forestiller man sig derimod, at man via internettet kunne møde andre mennesker fra hele verden, se deres ansigtsudtryk, høre deres tale, selv tale med dem, opleve ting sammen med dem osv., ville man for alvor åbne op for den globale kommunikation mellem mennesker.

En af visionerne med VRML mediet er netop, at man skal kunne møde og interagere med andre personer overalt på nettet. Efterhånden som mulighederne/redskaberne for social adfærd udbygges/forbedres i VRML mediet, vil der derfor ske en markant vækst i brugen af WWW

¹⁹ Intel's WWW adresse er <http://www.intel.com/>

²⁰ Lightscares adresse er <http://www.lightscape.com/>

²¹ Det virtuelle keramik galleri kan findes på http://www.bekkoame.or.jp/~inoue_k/VCM3.html

²² The Rosetti Room findes på <http://jefferson.village.virginia.edu/~dfa4y/dgr/>

²³ Kevin Hughes, "From Webspace to Cyberspace", Enterprice Integration Technologies, 1995, side 13. Bogen kan findes på <http://www.eit.com/~kevinh/cspace/#description>

som fundament for denne 'egentlige kommunikation', muligheden for at praktisere "social adfærd", frem for blot at benytte WWW til informationssøgning, forskning, markedsføring eller lignende.

Undersøgelser²⁴ har vist, at kun ca. hver 5. WWW-bruger primært anvender det til kommunikation, f.eks. i form af E-mails, IRC (Internet Relay Chat) eller nyhedsgrupper. I betragtning af, at internettet jo netop forbinder flere og flere mennesker med hinanden, så er denne 5.-del en forbløffende lille andel. Et WWW med en brugergrænseflade, der understøtter "ægte" social adfærd vil givet blive det et spændende sted at mødes - ikke bare med dem man kender og til daglig udveksler E-mail med, men også med ukendte personer fra andre lande og fremmede kulturer.

WWW vil selvfølgelig fremover stadig i høj grad blive brugt til informationssøgning. Men med fremkomsten og videreudviklingen af VRML vil WWW undergå sin måske mest radikale forandring, idet det vil gå fra at være et medie til kommunikation mellem menneske og maskine, til at være et kollektivt, socialt medie, der understøtter kommunikation mellem mennesker. Et måske ikke helt urealistisk bud kunne være, at WWW om fem år istedet for informationssøgning primært anvendes til "social adfærd".



Figur I: WorldsChat: Væggen, hvorfra man vælger sin avatar

Tæt på Cyberspace: WorldsChat og AlphaWorlds

Der findes allerede idag nogle ret funktionsdygtige prototyper på programmer, som via WWW simulerer kunstige 3D-verdener, f.eks. WorldsChat og AlphaWorld. Menneskene

²⁴ Bl.a. en undersøgelse lavet af Survey-Net; <http://www.survey.net/inet1r.html>

repræsenteres heri som tre-dimensionelle figurer- avatars - der kan gå rundt imellem hinanden, foretage forskellige handlinger, tale sammen, lytte, kigge mv.

I disse 3D-verdener er man helt klart meget tættere på at være i et egentligt "cyberspace", end hvad man hidtil har haft mulighed for. Dette også selvom programmerne kører op imod servere, der afvikler "lukkede verdener".

Men enhver med pc, WWW-adgang og WorldsChat eller AlphaWorld installeret, kan let betrede det grønne cyber-græs i disse facinerende verdener. Vi har afprøvet begge disse prototype-programmer og beskriver dem kort nedenfor. De giver begge et godt bud på, hvad fremtidige flerbruger VRML verdener bl.a. vil kunne byde på. De giver begge i en betydelig grad mulighed for social adfærd.

WorldsChat

I Worlds Chat²⁵ går man rundt på en virtuel rumstation sammen med andre internetbrugere fra hele verden. Man kan se de andre deltagere i rummet, og de kan se en. Hvordan man vil repræsenteres vælger man selv blandt et skønsomt udvalg af eventyrskikkelser, almindelige personskikkelser og tegneseriefigurer mv. (se figur 9). Møder man nogle interessante virtuelle personer, kan man kommunikere med dem via tastaturet i et separat tekstvindue (se figur 10).



Figur J: WorldsChat: Nogle figurer/personer står og taler sammen

²⁵ Worlds Chat kan hentes på <http://www.worlds.net>

AlphaWorld

AlphaWorld²⁶ minder på mange områder om WorldsChat. Man kan møde andre internetbrugere repræsenteret som menneskeskikkelser, kommunikere med dem via tastaturet, gå ture med dem osv. Der er dog nogle væsentlige forskelle. Hvor WorldsChat foregår på en rumstation, så er AlphaWorld et kæmpeområde bebygget med huse, veje træer m.m. Man får fornemmelsen af, at det er uendeligt stort, idet man kan blive ved med at vandre i en vilkårlig retning uden at støde på en grænse.

Ved ankomsten lander man i AlphaWorlds centrum (ground zero), hvor man som nybegynder farer forvildet rundt og prøver at komme i kontakt med nogen af de mange andre nyankomne. Mere erfarne AlphaWorld beboere bevæger sig målrettet ud i byen.



Figur K: AlphaWorld: Tre personer står og snakker sammen foran et hus (hvor Poerfie i dette tilfælde er os/brugeren)

AlphaWorld adskiller sig endvidere fra WorldsChat ved, at det er internetbrugere selv, der bebygger AlphaWorld. Enhver kan slå sig ned i AlphaWorld i et af de mange ubebyggede områder og bygge sit eget hus. Man har en stor mængde byggesten til rådighed, der kan sættes sammen til endog temmelig flotte 3D modeller af huse (se figur 12). En pudsigt lille finesse er, at man kan have en rigtig postkasse foran sit hus. Klikker besøgende på denne, kan de sende en e-mail til husets beboer, hvis han ikke er hjemme.

At man vitterligt kan møde folk fra hele verden fik vi bekræftet gennem en del besøg i byen. Man kan endog også møde danskere og nordmænd. På figur 11 ses en samtale mellem os (poerfie) og en dansker (pjerot) og en nordmand (Paulman). På figur 12 er vi gået indenfor i hu-

²⁶ AlphaWorld kan hentes på <http://www.worlds.net>

set, der ses udefra på den første figur. Læg mærke til den detaljerede møblering, med både pejs og levende ild.

Figur 11 og 12 er begge taget i en speciel tilstand, hvor man svæver over sin egen krop, for at kunne få bedre udsyn. Figuren på forsiden af opgaven er lavet i en mere normal tilstand, hvor man er inde i sin egen krop, og ser ud igennem sine "øjne".



Figur L: AlphaWorld: Vores avatar er nu spadseret ind i huset, og betragter det levende ildsted, statuen, det hyggelige sofahjørne og de flotte blomster

2.3 Det nye og det gamle medie

Historisk set har nye medier altid forsøgt at inkorporere/opsluge tidligere medier. Fjernsynet kunne gengive tale, billeder og tekst. Multimediecomputeren kunne ligesom fjernsynet gengive tale, billeder, tekst osv., men var derudover et interaktivt medie. Multimediecomputeren muliggjorde, at man pludseligt selv kunne bestemme hvad man gerne ville se og høre og hvornår. Nu er der så kommet en nyt medie VRML, som principielt tilbyder både multimedie computerens formidling af tekst og billeder, men derudover også interaktiv formidling af rumlige 3D informationer.

Der synes altså at være en tendens til, at nyt medie, for at slå an, helst skal kunne tilbyde det som tidligere medier kan og så gerne lidt til. Medier synes at ekspandere ud af akserne tid og rum, og der stræbes mod stadig mere virkelighedsnære tids- og rumlighedsopfattelser. Man kan gruppere medier efter hvordan de forholder sig til tid og rum. Jævnfør nedenstående tabel²⁷:

²⁷ Listen af medier er bestemt ikke udtømmende.

Rum Tid	1D	2D	3D
Statisk	?	Maleri, fotografier	Hologrammer
Dynamisk	Bøger	Video/TV	ikke-interaktiv 3D computergrafik
Nutid	Internet Relay Chat	Multimedie / Videotelefoni	Virtual Reality / VRML

1D angiver, at mediet anvendes til at formidle en-dimensionale informationer, som f.eks. tekst eller tale. 2D angiver, at mediet anvendes til at formidle to-dimensionale informationer, som f.eks. billeder eller video. 3D angiver, at mediet anvendes til at formidle tre-dimensionale informationer som f.eks. kemiske molekyler, eller arkitekttegninger.

Statisk tid angiver, at mediet kun kan formidle øjebliksinformation. Dynamisk tid angiver, at mediet kan formidle informationer spredt over tid. Nutidsmedier involverer understøttelse af kontinuerlig interaktion mellem medie og personen der anvender mediet.

Tabellen skal forstås på den måde, at jo længere man bevæger sig mod højre og nedad, desto mere omsluttende medier finder man, og i jo højere grad evner de at inkorporere tidligere medier. Konsekvensen af dette bliver, at Virtual Reality og dermed VRML bliver det hidtil mest omsluttende medie mennesket har frembragt. Dette nye medie evner at omslutte alle kendte medier.

Man kan så stille spørgsmålet ved, om VRML mediet vil udkonkurrere HTML mediet? Kigger man på historien, kan man se talrige eksempler på, at fremkomsten af nye medier har fremkaldt voldsomme diskussioner, om hvorvidt det nye medie ville udkonkurrere ældre medier eller ej. Da fjernsynet fremkom, mente datidens "eksperter", at bogen med tiden ville blive overflødiggjort, fordi folk ville læse bøger i fjernsynet. Samme diskussion har verseret omkring WWW og vil nu fortsætte med VRML mediet. Nogen vil hævde, at al information skal formidles igennem VRML, andre at mediet kun har begrænset anvendelighed osv. Vi tror, at selv om VRML mediet muligvis vil slå igennem, med samme kraft som HTML mediet, så vil det ikke erstatte dette eller andre medier, men snarere komplementere disse. Selvfølgelig vil nogle ting flyttes fra eksisterende medier til det nye medier. Men de gamle medier vil stadig have deres berettigelse.

Michael Benedikt udtrykker et lignende synspunkt på side 124 i sin bog "Cyberspace - first steps":

"...just as printing did not replace but displaced writing, and writing did not replace but displaced storytelling, and just as movies did not replace theater, nor television and movies... cyberspace will not replace either objective reality or dreaming and thinking in their historical modes... Each of these earlier media and activities will move over a little, as it were, free - indeed obliged- to become themselves, more involved in their own artistry and usefulness. Each will be dislocated in certain dimensions, but freed in others, as Innis, Mcluhan and Carpenter so clearly saw."

I næste afsnit vil vi forsøge at formulere nogle mere konkrete fremtidsscenarier, der viser hvorledes vi tror at VRML mediet og HTML mediet vil blive anvendt fremover.

2.4 Hvordan vil brugen af HTML og VRML finde sted i fremtiden?

I dette afsnit vil vi beskrive, hvorledes vi forestiller os at HTML og VRML medierne vil blive brugt i tilknytning til hinanden i fremtiden. Til dette vil vi illustrere tre scenarier, hvor Internet-browsing foregår med udgangspunkt i een af de to medier, eller i en kombination.

2.4.1 Scenarie 1: VRML indlejret i HTML

Vi forestiller os, at HTML i fremtiden vil understøtte VRML på samme måde, som det i dag understøtter 2D grafikformater som JPEG. På samme måde som man i dag kan angive et tag der refererer til en grafikfil, kan man forestille sig et tag, der refererer til en VRML fil, og samtidig angiver dimensionerne på firkanten. Umiddelbart efter at VRML filen er indlæst, vil det ligne et almindelig 2D grafikbillede, med mindre den da animerer. Man kan forestille sig, at der på undersiden af den omkransende firkant er placeret nogle simple manipulationsfunktioner, f.eks. et par knapper, der virker som Examine-mode i WebSpace. Det er herved muligt at dreje figuren, uden at den øvrige tekst dækkes, eller på anden måde påvirkes af, at man foretager nogle handlinger på VRML filen. Hvis man ønsker at undersøge VRML scenen yderligere, klikker man blot på VRML figuren, og skærbilledet maksimeres, på samme måde kan man i dag lave et link, fra et lille thumb-nail grafikbillede, til en version i fuld størrelse. Ved denne handling kommer der også mere avancerede navigationsmetoder til rådighed.

Dette var et eksempel på, hvorledes en HTML browser kan indkapsle en ny teknologi. Der findes mange eksempler på, at både gamle og nye teknikker er blevet indkapslet på lignende vis. Et godt eksempel er, at det i Netscape Navigator er muligt at afvikle en FTP-session (det er godt nok indeholdt i den underliggende URL-protokol).

Man kan allerede i dag finde HTML sider, hvori VRML indlejret. Man kan i denne forbindelse se VRML som 3D grafik, hvor enkelte 2D grafikformater allerede understøttes i HTML. Vi har tidligere beskrevet, hvorledes man ud fra MIME-type kan angive eksterne programmer til at behandle data på filformater, som browseren ikke understøtter.

2.4.2 Scenarie 2: HTML og VRML eksisterer side om side

Dette er faktisk den situation vi i dag befinder os i. Hvis en browser ikke selv kan håndtere VRML, kan den starte en ekstern VRML viewer. Men det kan også foregå den anden vej rundt, hvor man en VRML viewer starten en ekstern HTML browser. Alle de browsere vi analyserede tidligere, virker på denne måde.

Hovedparten af den information der i dag findes på tekstformat, vil være uegnet at overføre til VRML format. Vi forestiller os dog alligevel, at der kan opstå en parallel til World Wide Web, hvor navigation udelukkende foregår ved hjælp af VRML. Det vil ske, netop fordi VRML er meget bedre til at navigere rundt i informationsbjerge med end en HTML browser. Vi har tidligere vist et simpelt eksempel på, hvorledes en virtuel homepage kan se ud. Links til andre VRML filer eller HTML dokumenter kan knyttes til grafik, shapes eller elementer.

World Wide Web opstod for ganske få år siden, men har udviklet sig eksplosivt. Migreringen fra tekstbaserede systemer, som f.eks. gopher, veronica og ftp, til en grafisk brugergrænseflade, som f.eks. Mosaic, er foregået meget hurtigt. Men det betyder ikke, at den er overstået endnu, eller vil være det inden for de første par år. UNI-C sælger f.eks. stadig abonnementer til den klassiske tekstbaserede brugergrænseflade. På samme måde vil VRML ikke overtage fra

HTML med ét slag, men der vil i en årrække være to parallelle brugergrænseflader, en 3D og en 2D.

2.4.3 Scenarie 3: VRML med transparent brug af indlejret HTML

Den ultimative bruge af VRML finder man selvrølgelig den dag, hvor brugen af HTML i VRML dokumenter er fuldstændig transparent for brugeren. Når man har en HTML fil (eller et hvilket som helst andet tekstfilformat) i VRML, vil den blive vist eksempelvis som et ægte dokument, eller på en virtuel grafisk skærm, lavet i VRML. Vi taler her om et egentligt cyberspace. Grunden til, at VRML stadig vil understøtte HTML og alle andre filformater er, at informationer den dag i dag lagres på en lang række formater, som det ikke nødvendigvis vil være nyttigt at konvertere til VRML eller tilsvarende.

2.4.4 Konklusion

Vi forestiller os at scenarie 2, det som idag er det mest udbredte, langsomt vil blive afløst af scenarie 1 og 3. Scenarie 3 vil formodentlig være det man fremover vil forbinde med at surfe på internettet. Altså i et net af forbundne VRML scener, hvori man kan støde på HTML dokumenter, der vises i VRML scenen. Scenarie 1 vil finde anvendelse i de tilfælde, hvor man har behov for at supplere overvejende tekstuel information med 3D information. Det kunne f.eks. være i et videnskabeligt paper om molekylærbiologi, hvori tekst kan ledsages af VRML modeller af molekyler.

3. Konklusion

I forbindelse med diskussionen af styrker og ulemper for henholdsvis HTML mediet og VRML mediet fandt vi at information med fordel kan opdeles i navigation-information og target-information. HTML mediets primære styrke er formidling af target-information, mens VRML mediets primære styrke er formidling af navigation-information.

VRML mediet kan gennem 3D visualiserings teknikker skabe overblik over store dokumentkomplekser. Endvidere synes mennesker at have nemmere ved at huske navigation gennem rumlige strukturer end gennem abstrakte dokumenthierakier som HTML dokumenter.

Hvor HTML mediet kan betegnes som statisk og uden mulighed for interaktion mellem mennesker, så synes VRML mediet at rumme stort potentiale som formidler af kommunikation mellem mennesker.

Vores analyse af VRML og HTML mediets respektive styrker tyder på at det nye medie har en række kvaliteter, der kan give det samme gennembrud, som HTML mediet selv fik ved fremkomsten af WWW. Dog tror vi ikke at det nye medie fuldstændigt vil overfløddiggøre HTML mediet, men snarere supplere det.

Mere specifikt mener vi, at man kan forestille sig 3 scenarier for anvendelse af de to medier. Af disse 3 scenarier mener vi, at det er scenarie 3, der i fremtiden vil blive det foretrukne måde at surfe på internettet. Dvs. at man navigerer på internettet gennem et net af VRML scener, der kan indeholde HTML dokumenter, som vises i VRML scenen

Endelig mener vi, at det vi kalder det sociale aspekt, vil få en radikal indflydelse på udbredelsen af det nye medie. VRML mediet vil kunne skabe grundlag for en helt ny dimension af global kommunikation mellem mennesker. Man kan måske udtrykke det som et skift fra det vi idag kalder 'informationssamfundet', over til hvad man kan kalde 'kommunikationssamfundet'.

Bilag 1 - sourcekode til VRML scene

```
#VRML V1.0 ascii
Separator{
  ShapeHints {vertexOrdering CLOCKWISE shapeType SOLID}
  LightModel {model BASE_COLOR}
  Coordinate3{
    point[
      512 896 1024, 5120 896 1024, 5120 896 5632, 512 896 5632, 512 0 1024, 512 0 5632,
      2560 0 5632, 2560 0 1024, 3072 0 1024, 3072 0 5632, 5120 0 5632, 5120 0 1024,
      2560 0 1280, 3072 0 1280, 4864 28 1280, 3072 28 1280, 3072 810 1280, 4864 810 1280,
      2560 28 1280, 768 28 1280, 768 810 1280, 2560 810 1280, 3072 632 1280, 2560 632 1280,
      4864 28 5376, 4864 810 5376, 768 28 5376, 768 810 5376, 2560 632 1024, 3072 632 1024,
      768 28 2252, 768 28 2611, 768 273 2611, 768 273 2252, 768 28 2852, 768 581 2852,
      768 581 2611, 768 581 2252, 768 810 2852, 768 810 2252, 768 28 4185, 768 28 4627,
      768 225 4627, 768 225 4185, 768 667 5376, 768 667 4627, 768 667 4185, 768 735 5376,
      768 735 4185, 768 28 3730, 768 735 3730, 768 97 3730, 768 97 2852, 768 735 2852,
      3583 0 1024, 4108 0 1024, 4108 126 1024, 3583 126 1024, 5120 651 1024, 4108 651 1024,
      3583 651 1024, 3583 896 1024, 3072 896 1024, 2560 896 1024,
    ] # 64 points
  }
  TextureCoordinateBinding {value PER_VERTEX_INDEXED}
  TextureCoordinate2{
    point[
      7.59375 0, 10.3984 0, 10.3984 1.91406, 7.59375 1.91406, 12.2812 0, 12.2812 4.32031,
      10.3984 4.32031, 7.59375 4.32031, 12.2812 6.10938, 7.59375 6.10938, 0 0,
      0 6.10938, 22.6953 0, 26.1484 0, 26.1484 1.53906, 22.6953 1.53906, 32 0,
      32 4.99219, 26.1484 4.99219, 22.6953 4.99219, 32 5.52344, 22.6953 5.52344,
      19.1406 0, 19.1406 5.52344, 19.1406 0.539062, 12.2812 0.539062, 12.2812 5.52344,
      32 6.10938, 1 0, 1 1, 0 1, 32 32, 0 32, 23.9922 0, 28.0938 0, 28.0938 0.984375,
      23.9922 0.984375, 36 0, 36 5.08594, 28.0938 5.08594, 23.9922 5.08594, 36 7,
      23.9922 7, 20 0, 20 7, 16 0, 16 7, 0 7, 16 4.9375, 20 4.9375,
    ]
  }
  MaterialBinding {value PER_FACE_INDEXED}
  Material{
    ambientColor 0 0 0
    emissiveColor 0 0 0
    diffuseColor[
      0.313726 0.25098 0.188235, 0.25098 0.501961 0.454902, 0.862745 0.705882 0.486275,
      0.74902 0.74902 0, 0.407843 0.329412 0.407843, 0.54902 0.501961 0, 0.611765 0 0,
      0.470588 0.0784314 0.25098, 0.0784314 0.298039 0.282353, 0.784314 0.580392 0.486275,
      0.658824 0.752941 0.580392, 0.313726 0.376471 0.188235, 0.407843 0.439216 0.423529,
    ] # 13 colors
  }
}

WWWAnchor{
  name "http://www.econ.cbs.dk/webkursus/projects/BigMac/index.html"

  Separator { # The red sphere
    Texture2{
      filename "moon.gif"
      model DECAL
    }

    # Material {
    #   diffuseColor 1 0 0 # Red
    # }
    Translation { translation 3 0 1 }
    Sphere { radius 300 }
  }
}

Separator{
  IndexedFaceSet{
    coordIndex[
      0,1,2,3,-1, 4,5,6,7,-1, 8,9,10,11,-1, 12,6,9,13,-1, 14,15,16,17,-1, 18,19,20,21,-
1,
      22,23,21,16,-1, 24,14,17,25,-1, 26,24,25,27,-1, 7,12,23,28,-1, 13,8,29,22,-1,

```

```

28,23,22,29,-1, 12,13,15,18,-1,
] # 13 faces
materialIndex[
0, 1, 1, 1, 2, 2, 2, 3, 2, 4, 4, 5, 6,
]
}
Separator{
Texture2{
filename "bl50.bmp"
model DECAL
}
IndexedFaceSet{
coordIndex[
1, 30,31,32,33,-1, 31,34,35,36,-1, 37,35,38,39,-1, 19,30,39,20,-1, 40,41,42,43,-
1, 41,26,44,45,-1, 46,44,47,48,-1, 49,40,48,50,-1, 34,49,51,52,-1, 53,47,27,38,-
1,
] # 10 faces
textureCoordIndex[
0,1,2,3,-1, 1,4,5,6,-1, 7,5,8,9,-1, 10,0,9,11,-1, 12,13,14,15,-1,
13,16,17,18,-1, 19,17,20,21,-1, 22,12,21,23,-1, 4,22,24,25,-1, 26,20,27,8,-1,
]
materialIndex[
3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
]
}
}
WWWAnchor{
name "http://www.econ.cbs.dk/webkursus/students/hensing/index.html"
Separator{
Texture2{
filename "hensing.jpg"
model DECAL
}
IndexedFaceSet{
coordIndex[
52,51,50,53,-1,
] # 1 faces
textureCoordIndex[
10,28,29,30,-1,
]
materialIndex[
3,
]
}
}
}
WWWAnchor{
name "http://www.econ.cbs.dk/webkursus/students/tjornov/"
Separator{
Texture2{
filename "64x64.bmp"
model DECAL
}
IndexedFaceSet{
coordIndex[
43,42,45,46,-1,
] # 1 faces
textureCoordIndex[
10,28,29,30,-1,
]
materialIndex[
3,
]
}
}
}
}
WWWAnchor{
name "http://www.econ.cbs.dk/webkursus/students/bila92ab/index.html"
Separator{
Texture2{
filename "hsb.bmp"
model DECAL
}
IndexedFaceSet{
coordIndex[
33,32,36,37,-1,
] # 1 faces
textureCoordIndex[
10,28,29,30,-1,
]
}
}
}
}

```

```

        ]
        materialIndex[
        3,
        ]
    }
}
Separator{
    Texture2{
        filename "stdcr010.bmp"
        model DECAL
    }
    IndexedFaceSet{
        coordIndex[
        19,14,24,26,-1,
        ] # 1 faces
        textureCoordIndex[
        10,16,31,32,-1,
        ]
        materialIndex[
        7,
        ]
    }
}
Separator{
    Texture2{
        filename "stdcr010.bmp"
        model DECAL
    }
    IndexedFaceSet{
        coordIndex[
        20,27,25,17,-1,
        ] # 1 faces
        textureCoordIndex[
        10,16,31,32,-1,
        ]
        materialIndex[
        8,
        ]
    }
}
Separator{
    Texture2{
        filename "bl60.bmp"
        model DECAL
    }
    IndexedFaceSet{
        coordIndex[
        54,55,56,57,-1, 55,11,58,59,-1, 60,58,1,61,-1, 8,54,61,62,-1, 4,7,63,0,-1,
        28,29,62,63,-1,
        ] # 6 faces
        textureCoordIndex[
        33,34,35,36,-1, 34,37,38,39,-1, 40,38,41,42,-1, 43,33,42,44,-1, 10,45,46,47,-
1,
        48,49,44,46,-1,
        ]
        materialIndex[
        9, 9, 9, 9, 9, 9,
        ]
    }
}
WWWAnchor{
    name "mailto:birgerl@inet.uni-c.dk"
    Separator{
        Texture2{
            filename "message2.bmp"
            model DECAL
        }
        IndexedFaceSet{
            coordIndex[
            57,56,59,60,-1,
            ] # 1 faces
            textureCoordIndex[
            10,28,29,30,-1,
            ]
            materialIndex[
            9,
            ]
        }
    }
}

```

