Æliens

preface

This book provides a concise and comprehensive introduction to multimedia. It arose out of the need for material with a strong academic component, that is material related to scientific research.

Indeed, studying multimedia is not only fun. Compare it with obtaining a driver license. Before you are allowed to drive on the highway, you have to take a theory exam. So why not take such an exam before entering the multimedia circus.

Don't complain, and take the exam. After all it makes you aware of the rules governing the (broadband) digital highway.

themes and variations

So, who is this book meant for? It is meant for the student or reader who is looking for a quick introduction to the main topics in multimedia. The twelve chapters provide a concise overview of the themes and trends in current multimedia practice and research.

The themes and variations addressed in this book may be summarized as follows.

themes and variations

- *digital convergence – all for one, one for all*
- *broadband communication – entertainment*
- *multimedia information retrieval – as an afterthought?*
- *multimedia and game application(s) – from design to development*

To explain in somewhat more detail, *digital convergence* may be characterized as the coming together of data (including audio, video and information) in a possible multitude of platforms, to which these data are delivered by a variety of (broadband) communication channels. In fact, the increasingly powerful communication infrastructure due to the popularity of the Internet and the World Wide Web, leads to an almost universally accessible multimedia (information) repository, for which (unfortunately) the notion of (multimedia) information retrieval seems to have occurred only as an afterthought. Digital content design is only one step in the process of multimedia application development. Important issues in multimedia application development are, apart from project maanagement, data representation, navigation, presentation and usability.

An underlying thought that motivated the writing of this book is that somehow the gap between *authoring* and *retrieval* should be bridged. In other words, either by developing the technology for extracting features or attributes from multimedia objects, or by applying content annotation for such objects, multimedia information retrieval should be considered as a necessary asset to make a multimedia web an effective information repository. In multimedia applications, such as the *digital dossier* we introduce in chapter 10, the data representation must accomodate meta-information, to support effective navigation and search.

Another line of thought, that became more clear during the writing of the book is concerned with the aesthetics of (interactive) applications. You will find more on this in chapters 11 and 12, that deal with game development.

what do you need to learn

When taking up multimedia as a subject of study, you may ask yourself what you need to know and learn about it. In general, what this book presents is

> a collection of concepts, a number of facts, some history, potential applications and application areas, a brief overview of standards (some of which are still being developed), technology issues, as well as some scattered insights on visual design, application development and the relevance of multimedia and games.

Let me be frank with you. There is too much information to be digested in a first course. Nevertheless, after studying this book you will have an introduction to multimedia that should be viable for the rest of your (academic) career.

Now, don't hesitate, put yourself to the test and check which phrases and acronyms you are familiar with in the lists given for the subjects of *digital convergence*, *broadband communication* and *information retrieval*.

digital convergence

- concepts – *digital revolution*
- facts – *from the entertainment industry*
- history – *from Pong to Big Brother*
- applications – *infotainment*
- standards – *MPEG, RM3D, SMIL*
- technology – *TV, PC, DVD*

How did you succeed thus far? If you did well, try the second round and test yourself in what detail you have have knowledge about technologies mentioned.

broadband communication

- concepts – *Quality of Service*
- facts – *compression is needed*
- history – *the internet*
- applications – *entertainment and communication*

- standards – *HTTP, TCP/IP, RTP*
- technology – *cable, (X)DSL*

Finally, check to what extent you master the vocabulary of multimedia information retrieval.

multimedia information retrieval

- concepts – *features, precision, recall*
- facts – *the problem is utterly complex*
- history – *from text to multimedia*
- applications – *digital libraries*
- standards <– *distance metrics*
- technology – *indexing & algorithms*

If you are working online, you may click back to the text in the book that explains these notions. Just to make sure whether your impression of familiarity was justified.

assignment(s)

I strongly believe that practical work is necessary, also for academics, to get a good grasp on multimedia and game development. Even if your interest is purely intellectual, it pays off to make your virtual hands dirty and indulge in making a compelling presentation.

As an assignment, consider making a presentation that offers an

Annotated Tour in Amsterdam

Amsterdam is the place where I live, and where our students take their courses. You may find it more convenient or natural to replace Amsterdam with a location of your choice.

Online, you will find an elaborated version of the assignment, including an extended description, a working plan, deliverables and hints. In essence though, the intent of the assignment is to make a compelling, not to say artistic, presentation, and to explore the realm of multimedia rethorics.

As a tool you may choose, for example, Flash or the flex 2 SDK, which is freely available.

examination

Despite the fact that some consider the practical aspects of multimedia to be exclusively relevant, the intellectual aspects of multimedia should not be ignored.

Consider the following question, which is directly related to one of the themes underlying this book, that is the complementarity of authoring and retrieval:

> *Give a short description of the contents and structure of your presentation. Indicate how the information contained in your presentation can be made accessible (for example in search).*

This question can only be answered when the student has a sufficient level of experience, insight and knowledge of the field, and is able to relate theory and practice.

Each chapter contains a brief list of questions that may be used as a checklist, to see if you have sufficient knowledge of a particular area. These questions may also be used to prepare exams! The questions are meant to test for insight, that is the ability to discuss a somewhat broader theme, and knowledge of concepts and technology, covering definitions, applications, historical facts, as well as the technological infrastructure enabling the deployment of multimedia applications.

In addition to the regular material, the book also contains a number of examples and sections indicating *research directions*. These sections are not meant to be part of the exam, but might provide the student with suggestions for projects or further research. Moreover, both the discussions in the *research directions* and the material in the appendices presents a vision on what multimedia should be. In effect, I have a strong preference for a programmatic approach to (intelligent) multimedia, as outlined in appendix E. Nevertheless, the bulk of the (regular) material is relevent also for readers with a rather different opinion on what consitutes the *essence of multimedia*.

how to use this book

The intended audience for this book is

intended audience(s)

- students (beginning and advanced)
- instructors
- professionals and interested laymen

The course notes were explicitly written for first year Computer Science and Information Science students. The Information Science students are expected to choose the specialisation *Multimedia and Culture*, a curriculum provided by the department of Mathematics and Computer Science of the Faculty of Sciences of the VU University Amsterdam. The course has a practical part and a theoretical part, which in combination takes 2-4 weeks, full time study. The book covers the theoretical part. The online version gives a skeleton assignment that may adapted by the one responsible for the course. The online version contains all the material needed for giving a multimedia course, that is

multimedia course

- presentations for all chapters, including the preface in dynamic HTML slides
- presentable versions of the MPEG-4 standard, and other relevant material
- possible exam questions, with back links into the text for quick learning and review
- seven sample lectures, with additional explanation for the instructor

One additional remark may be made. This is (so to speak) 'a book with an attitude'. It is slightly authorative and directive towards the students, telling them to learn the facts and 'do the exam'. Some students take refuge to learning the 'keywords and phrases'. They are even helped in this respect, since the text uses a 'graphic' layout to emphasize important points, and to allow for a quick recognition of chunks of relevant material.

the artwork

Although a book about multimedia does not need to be a multimedia artefact itself, it seemed better to include illustrations, to avoid the impression of a 'dry' book. Since I did not want to include any redundant diagrams or pictures, I decided to use a personal selection from the history of visual design, games, computer art and video art, not only to spice up the book but also to give the reader a collection of interesting samples. Each chapter starts with illustrations setting the *visual theme* of the chapter. All other illustrations are, in one way or another, related to the examples or the text of that chapter. Brief comments about the artwork, and an explanation of the visual theme, can be found at the end of each chapter.

about the author

At some point you may wonder whether the author is qualified or authorized to write about a particular subject and, in this particular case, to publish a book about such an elusive notion as multimedia.

Let me give you some personal history. Way back in the seventies, I did a degree in painting at the Gerrit Rietveld Academy in Amsterdam. At the same time, I did a master's philosophy, where I graduated in the field of aesthetics on a comparative study on theories of imagination and creativity, reading writers such as Kant, Husserl and Sartre. Then I got an interest in computer music, after listening to a concert of Xenakis in Paris, and started to work on a PDP-15 (with 4K of memory) at the Institute of Sonology in Utrecht. Leaving all philosophy and traditional art behind, I learned programming, studied AI and theoretical computer science. Some eight years later, I obtained my Ph.D. in computer science and started my academic career. After working in software engineering, and in particular object-oriented software development, I was asked, at the end of the millenium, to set up a collection of multimedia courses, since by then multimedia was coming in vogue as an academic subject. These courses, which include the introduction multimedia, Web3D authoring, intelligent virtual environments, a multimedia casus, and recently also visual design, are reflected in this book.

about the book

What started as a (not so) gentle introduction to multimedia, has grown into a rich (at times somewhat idiosyncratic) collection of topical material about multimedia and game development. Borrowing a phrase from the politics of the seventies, at some point, apparently, the professional became personal, and the personal professional. Nevertheless, the book may still be read as an introduction. It is written in a concise and compact manner, supported by the slides format, which allows for presentation of the material in class, and is illustrated by a variety of images, taken from the arts, design, and multimedia and game projects.

acknowledgements

This book is the result of developing the course notes for an *introduction to multimedia* for first year Computer Science and Information Science students. Hence, first of all, I like to thank the students that had to endure all the rough drafts of this material, and perhaps not to forget my experiment(s) with the presentation format of it. Further I like to thank Harrie van der Lubbe and Sander Lammers for developing the manual for Director and their support in developing the practical assignment. Also, I like to thank Martin Kersten from CWI for allowing me to join his Multimedia Database Systems research group as a guest for a period of about two years, and Alex van Ballegooij for his active involvement in the RIF project and his coding effort for the *slide* PROTOs, used to produce the presentation slides for this book and described in appendix B. Also from CWI, I like to thank Lloyd Ruttledge, Lynda Hardman and Jacco van Ossenbruggen, for their effort in thinking about the multimedia course in its initial stages, and Lloyd and Jacco for their involvement in some of the practical work, and Jacco in particular for his knowledge of hypermedia systems that he shared with me during the period that he was my Ph.D. student. From CWI, I like to thank Zsofi Ruttkay for her general interest in 'my projects'. From the VU, I like to thank Andy Tanenbaum for allowing me to use his material on digital video, Gerrit van der Veer for taking the initiative for *Multimedia and Culture*, Zhisheng Huang for his excellent contributions to the WASP project, Johan Hoorn for our spirited cooperation, and Claire Dormann, for our discussions on the direction the *Multimedia and Culture* curriculum should take, and for sharing her thoughts on persuasive technology with me. I also like to thank Tatja Scholte and IJsbrand Hummelen from ICN (Netherlands Institute for Cultural Heritage) for their contributions to the *multimedia casus*, and Gaby Weijers and Bart Rutten from Montivideo for their cooperation and all the video material they so generously provided. Thanks are due to Mark Veldhuijzen van Zanten, Jaap Stahlie, Peter van Kessel, Federico Campanale and Katelijne Arts for providing me with material. Special thanks goes to the student members of the 2004 autumn *multimedia casus* group, for their collective work on the *abramovic dossier*, and to Rutger van Dijk for rekindling my interest in C++/DirectX programming by his youthful enthousiasm. I also should not forget two students from the Computer Science master Multimedia, who were exceptional in their dedication and skills, Winoe Bhikharie, who acted (among others) as the manager in the development of the VU-Life game described in 11.2, and Marco Bouterse, who excelled in both DirectX and Half-life shader programming. Also I need to mention Marek van de Watering, a student from Multimedia and Culture, who often surprised me by his sincerity and sensitivity. No doubt, I owe thanks to Gaynor Redvers-Mullon who made a serious attempt to encourage me to get the best out of this manuscript, even though at some point I decided to do it in my own way. Further, I also wish to thank Dhaval Vyas, at that time PhD. student at

VU, for his interest in my projects and for sharing his thoughts on the *panorama* project, znd Zeljko Obrenovic, postdoc at CWI, for his enthusiasm, his technical skills and his willingness to assist me with my teaching. I thank Pier Vellinga, who I googled as the spin-doctor of climate change, for involving me in Clima Futura, and not in the least for the pleasant cameraderie we enjoyed during the effort to obtain the dutch national prize for academic communication, and Frans Feldberg, for involving me in the VU @ Second Life effort, an equally refreshing interdisciplinary partnership, with which we got into the 8 o'clock nation TV news, as planned! Last but not least, I owe Yiwen Wang (王依文) my deepest gratitude for providing the motivation to continue the work, and to strive for beauty and pleasure, in a serious way.

Finally, I must mention that I owe much insight and material to (among others) the following books and articles: Subrahmanian (1998), Forman and Saint John (2000), Chang and Costabile (1997), Ossenbruggen (2001), Vasudev and Li (1997), Hughes (2000), Grau (2003), Kress and van Leeuwen (1996), and not to forget Zielinski (2006). As in any intellectual endeavor, intellectual ancestry can hardly be praised enough. So let me briefly indicate, for each chapter, some of the sources that provided me with inspiration, insight and material:

1. Forman and Saint John (2000), Davenport (2000), Jain (2000).
2. Chang and Costabile (1997), Ossenbruggen (2001), Hughes (2000).
3. Vasudev and Li (1997), Koenen (2000), Visser and Eliens (2000).
4. Luna (2003), Adams (2003), Fernando and Kilgard (2003)
5. Subrahmanian (1998), Baeza-Yates and Ribeiro-Neto (1999).
6. Subrahmanian (1998), McNab et al. (1997), Kersten et al. (1998).
7. Subrahmanian (1998), Fluckiger (1995),
8. Fluckiger (1995), Ballegooij and Eliens (2001), Huang et al. (2002).
9. McCuskey (2002), Bolter and Grusin (2000),
10. Chapman and Chapman (2004a), Chapman and Chapman (2004b), Hughes (2000),
11. Sherrod (2006), Grau (2003),
12. Juul (2005), Arnheim (1957), Hawkins (2005), Kress and van Leeuwen (1996).

The material in sections 4.3, 7.1, 7.3, chapter 8, sections 9.3 and 10.2, and section 11.2 reflect my own research efforts. The other material has all been diligently collected from (among others) the sources mentioned.

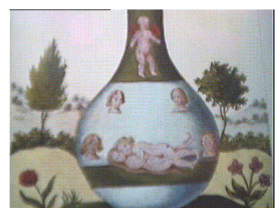
topical media & game development

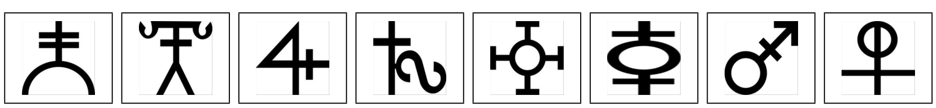
part i. digital convergence

more than the art of turning base metals into gold, alchemy is a system of cosmic symbolism

perfect solutions

1. digital culture
2. hypermedia information spaces





2

reading directives In these first chapters, we will explore the notion of multimedia, look at it from a historical perspective and discuss in somewhat more detail the issue of convergence, the mix and interchange of media that has become possible in the digital era. In the second chapter, we will give a precise definition of information spaces, and an overview of the history of hypermedia.

Essential sections are section 1.2, which characterizes digital convergence in a more precise way, and sections 2.1 and 2.2, which respectively characterize information spaces and hypermedia. Section 2.3 may safely be skipped by readers not interested in the philosophy of media and creation.

perspectives The topics treated in this part can be looked at from multiple perspectives. When you write a paper about any of these topics, as suggested in *projects* paragraph, you should be aware of from which perspective you tackle your subject.

2

In summary, we can in a non-exhaustive way, distinguish between the following perspectives:

perspectives – digital convergence

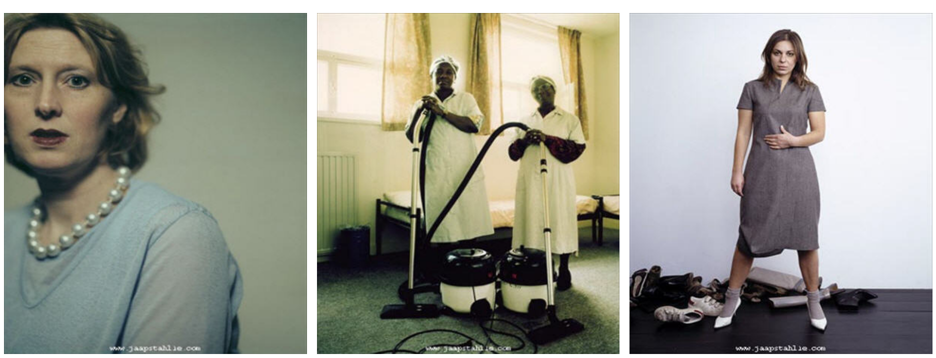
- *historical* – media development timeline
- *sociological* – communication relations
- *psychological* – experience limits
- *aesthetical* – dynamics of creation
- *technical* – divergence & competition
- *philosophical* – re-mediation
- *commercial* – what is the economic model?

As an example, the psychological perspective deals with an interesting issue, namely how much information can we digest and what are the limits to our perceptual system that determines whether the experience offered by a virtual reality interface is really effective.

essay topics The issues treated in these chapters may be used as topics for an essay. As a hint, here are a few titles:

- digital convergence and the future of mobile multimedia
- media @ home – the windows media center
- media art – merging technology and aesthetics

For a first essay, I would suggest a paper no longer than 5 pages. If there are technical details that you do not want to omit, then consider an appendix of 2 to 3 pages. For hints on how to approach writing a paper, see appendix 5.



3

the artwork

1. alchemy – an illustration from a book about alchemy, from which also the quote is taken, the quote is explained in the *afterthoughts*.
2. signs – ancient chemical symbols, van Rooijen (2003), p. 171, 172.
3. photographs – Jaap Stahlie[1], from portrait series.

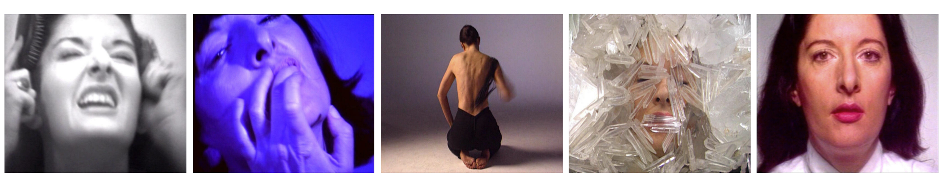
[1] www.jaapstahlie.com

1. digital culture

life is becoming digital <xx>

learning objectives *After reading this chapter you should be able to define the notion of multimedia, recount the history of digital entertainment, explain the concept of digital convergence, discuss the future of cyberspace, and speculate about the commercial viability of mobile multimedia.*

We live in the digital era, Negroponte (1995). We are surrounding ourselves with gadgets and we are consuming immense amounts of information, that is increasingly being delivered to us via the Internet. We play games, and we still watch (too much) television. Some of us watch televion on our PCs, and may be even looking forward to watch television on their mobile phone. This is multimedia. For others, the PC is still a programmable machine. Being able to program it might earn you a living. Understanding multimedia, however, might even provide you with a better living. In this chapter, we study what trends may currently be observed in the creation and delivery of multimedia information, and we explore what impact the digital revolution may have from a commercial perspective.



1

1.1 entertainment and experience

The question of *what is multimedia* is rather elusive. We may, nevertheless, look at how the phrase *multimedia* is used, and how the concept *multimedia* is related to other concepts. as in the concept graphs that may be obtained with the Visual Thesaurus[2], providing as input *multimedia*.

[2] www.visualthesaurus.com

We then see that the notion of multimedia is related to *systems*, in particular interactive and hypermedia systems, and indirectly also to the notion of *transmission*, which will even become more apparent when we inspect the graph for the concept of *medium*, depicted in figure X below.

However, although this gives us some indication of how to position *multimedia* in the larger area of computer applications, in particular when exploring the *systems* node, it does not so much tell us what multimedia is all about.

From the perspective of human cognition, we may look at how multimedia contributes to our understanding of ourselves and the world around us. Traditionally, three levels of cognitive functioning are distinguished, Bruner (1972), corresponding with three levels of meaning:

levels of meaning

- actionary level – action and movements
- sensory/iconic level – images and impressions
- symbolic level – language and mathematics

Multimedia is clearly (most strongly) related to the sensory/iconic level, although for games one could say there is also a strong relation with the actionary level, and to some extent (for both multimedia and games) with the symbolic level.

For a more serious and deep understanding of how multimedia artefacts provide meaning and what role they play in our daily life, or how that meaning is affected by social contexts, we need to take recourse to *semiotic theory*, which is now one step too far, buth which we will look at in chapter 12.

Antother perspective from which to understand the meaning of *multimedia*, is to look at the function of media in our society, or, in other words, how *multimedia* is situated in our cultural institutions.

Consider this quote from the preface of all of all MIT books in the *Leonardo* series:

cultural convergence

> *The cultural convergence of art, science, and technology provides ample opportunity for artists to challenge the very notion of how art is produced and to call into question its subject matter and its function in society.*

Although the quote is about *art*, it is essentially related to *multimedia*, to the extent that the quote refers to *media art*. The MIT Media Lab[3] is one of the worlds most famous institutes in the field of multimedia. The *Leonardo* series is a collection of authoritive books on multimedia and related topics, which includes Zielinski (2006), Grau (2003), Wilson (2002).

To understand the position of (computer supported) media in our society, we may observe following Zielinski (2006): there are two forces, political and technological, and there is, currently, a trend towards standardization and uniformity

standardization and uniformity

1. Telematic media were incorporated very quickly in the globalization strategies of transnational corporations and their political administrators and they became increasingly dependent on existing power structures.

[3]www.media.mit.edu/

2. At the other end of the scale, there were individuals, or comparatively small groups, who projected great hopes onto these networks as a testing ground for cultural, artistic and political models that would give greater prominence and weight to divergence and plurality.

This reflects what Zielinski (2006) calls the *advanced media paradox*, facilitating heterogeneity and immersion on the one hand, and striving for universalisation on the other hand, as demanded by the centers of technological and political power.

Leaving the socio-political arena, we may in some sense predict the tension between *convergence* and *divergence*, by looking at the meaning context of the concept of *convergence*, again using the Visual Thesaurus, where we find that not only notions such as *overlap* and *occurrence* are related to it, but also the complementary concept of *divergence*. However, instead of speculating on the meaning of words, it might be more worthwhile to look at what we may consider to be the recent history of multimedia, entertainment.

entertainment

In november 2000, a theme issue of the Scientific American appeared, featuring a number of articles discussing (digital) entertainment in the era of digital convergence. Let's start with a quote:

Scientific American (november 2000)

> *The barriers between TV, movies, music, videogames and the Internet are crumbling. Audiences are fetting new creative options. Here is what entertainment could become if the technological and legal hurdles can be cleared ...*

Moreover, the editors made some wildly speculative claims, such as *digitizing everything audio and video will disrupt the entertainment industry's social order*, and *the whole concept of holding a CD or movie in your hand will disappear once d-entertainment is widely available.* To some extent this seems already to be true, as for example the music industry can painfully testify to.

Underlying the importance of entertainment in the era of digital convergence is the premisse governing an entertainment economy, which may be stated as

there is no business without show business

Additionally, the authors of the introduction to the theme issue speculate that *the creation of content will be democratized*, due to the availability of low cost digital movie cameras and PC video editors. Producing a video movie is now possible for just a few thousend euro or dollars. However, given the aesthetic ignorance of the average individual making video movies, it seems doubtful that this will hold true for entertainment in general.

In that same issue of the Scientific American, Gloria Davenport, a pioneer in the field of multimedia, presents list of applications characterizing the evolution of digital entertainment, Davenport (2000):

evolution of digital entertainment

- 1953: Winky Dink (CBS) – interactive television, drawing exercise
- 1972: Pong (Atari) – ping-pong on computer screen
- 1977: Adventure – text-based interactive fiction
- 1983: Dragon's Liar – laser-disc technology 3D game
- 1989: SimCity – interactive simulation game
- 1989: Back to the Future – the Ride
- 1993: Doom – 3D action game
- 1995: The Spot – interactive web-based soap opera (Webisodic)
- 1999: IMAX3D – back to Atlantis (Las Vegas)
- 2000: Big Brother – TV + around the clock Web watch + voting
- 2001: FE Sites – fun enhanced web sites

It is interesting to note that *Big Brother*, which was originally created by a Dutch team, has become a huge success in many countries. Although the integration with the web was limited, it may be seen as the start of a number of television programs with web-based interaction facilities.

digital experience

The list compiled by Gloria Davenport suggests, a convergence towards an 'ultimate digital experience', Now, what does *digital experience* mean?

In a special issue of the Communications of the ACM, about the next 1000 years of computing, Ramesh Jain makes the following observation, Jain (2000):

> *The desire to share experiences will be the motivating factor in the development of exciting multimedia technology in the foreseeable future.*

Considering the variety of means we have at our disposal to communicate, as reflected in the list below, we may wonder whether our current technology really stands out as something special.

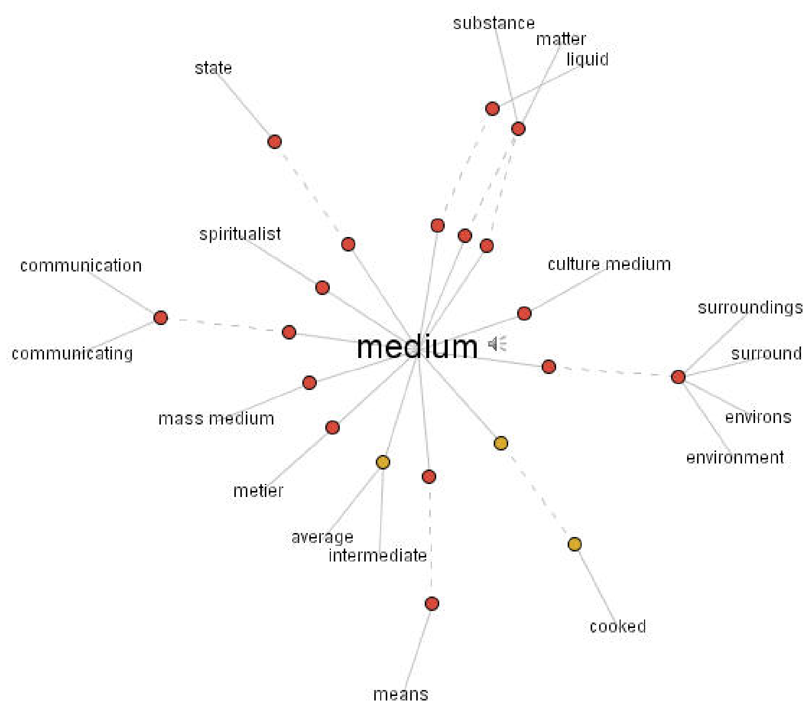
communication technology

- *oral* – communicate symbolic experiences
- *writing* – record symbolic experiences
- *paper* – portability
- *print* – mass distribution
- *telegraph* – remote narrow communication
- *telephone* – remote analog communication
- *radio* – analog broadcasting of sound
- *television* – analog A/V broadcasting
- *recording media* – analog recording
- *digital processing* – machine enhancement
- *internet* – multimedia communication

According to Ramesh Jam, internet-based multimedia communication differs from earlier communication technology in that it somehow frees the message from the medium. Reflecting on Marshall McLuhan phrase – *the medium is the message* – he observes that:

the medium was the message when only one medium could be used to communicate messages.

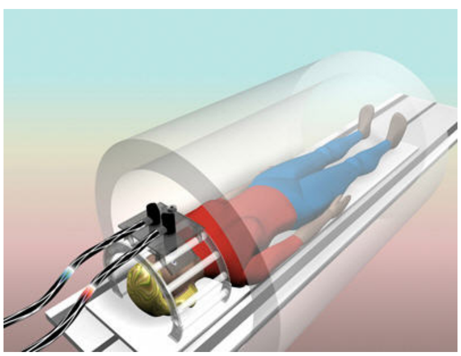
Now, that the Internet allows the synthesis and rendering of information and experiences using whatever is the most appropriate media to convey the message, the message is, as Jain phrases it, just the message, and the medium is just the medium. In other words, the medium itself does not seem to constrain what message can be conveyed. Looking at the documentary *Fahrenheit 9/11* though, we may seriously doubt whether this is true. Although it is possible to gain knowledge about the alliances that underly politics, even in the age of the internet, the television campaigns seem to be more dominant in affecting the general publics opionion about global politics than anything else, due to the conventional formats of presentation and editing.



2

Let's once more look at a graph, above, indicating the concept relations for the notion of *medium*. What strikes me as important are the relations with the distinct concepts of *substance*, *communication*, *environment*, and *intermediate*. In some respects the notion of *medium*, underlying the plural use of it in *multimedia* is comparable to the notion of *ether*, which was once seen as a vehicle for the transport of broadcasted information. But I also like to stress the 'substantial' aspect of multimedia, as a material for design and creation, similar to paint.

The basic issue here is what is a medium and how does it affect, or even shape our experience(s). Following Ramesh Jain, we may speculate that the range of sensory information offered by multimedia applications may become much richer than is currently the case, and we may then predict that there will be a tremendous progress in presentation technology, multisensory presentation technology! Clearly, from a technological perspective there seems to be no limit, except those imposed by our own phantasy. However, it should be equally obvious that compelling experiences rely on carefully staged presentations, and as such require an entirely new discipline of design.



VR for pain relief

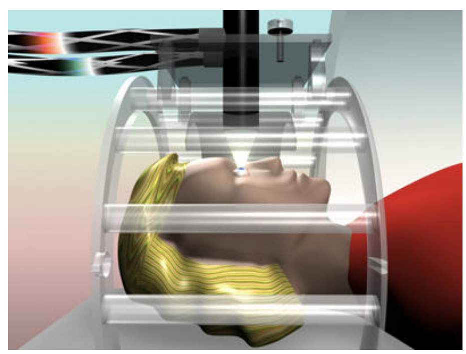


image delivery system

3

example(s) – ***VR for pain relief***

The research project fMRI Research on Virtual Reality Analgesia[4] at the Human Interaction Laboratory (Washington) has explored the use of VR to reduce the agonay of taking MRI scans. The U.W Radiology Digital Imaging Science Centers wide field of view magnet-friendly virtual reality image delivery system makes it possible for volunteers and patients to have the illusion of going into virtual reality during fMRI brain scans. As explained on the website, the image on the left above, shows a woman in virtual reality during an fMRI brain scan, looking into a custom magnet-friendly virtual reality goggles. VR images from projectors in another room are carried to the participant in the form of light (photons, no electrons) via optic fiber image guides. The participant has the illusion of going inside the virtual world, allowing researchers to measure what happens to her brain when she reports reductions in pain during VR. The white cage-like structure around the womans head, in the image on the right, shows fMRI receiver coils used by the

[4]www.hitl.washington.edu/research/magnet

fMRI brain scanner to collect the information about changing patterns of brain activity.

Another project investigating the use of VR techniques for pain distraction can be found at the site of the Virtual Environments[5] of the Georgia Institute of Technology, Atlanta.

research directions– *the face of cyberspace*

The notion of *cyberspace* was introduced in William Gibson's novel *Neuromancer*, that appeared in the early 1980's, signifying a vast amount of (digital) data that could be accessed only through a virtual reality interface that was controlled by neuro-sensors. Accessing data in *cyberspace* was not altogether without danger, since data protection mechanisms (including firewalls, as we call them nowadays) were implemented using neuro-feedback. Although the vision expressed in *Neuromancer* is (in our days) still futuristic, we are confronted with a vast amount of information and we need powerful search engines and visualisation techniques not to get lost. So what is the reality of *cyberspace* today?

... cyberspace is a construct in terms of an electronic system.

as observed by Vivian Sobschack, 1996, quoted from Briggs and Burke (2001), p. 321. On reflection, our (electronic) world of today might be more horrendous than the world depicted in *Neuromancer*. In effect,

cyberspace

> *television, video cassettes, video tape-recorder/players, video games, and personal computers all form an encompassing electronic system whose various forms interface to constitute an alternative and absolute world that uniquely incorporates the spectator/user in a spatially decentered, weakly temporalized and quasi-disembodied state.*

All these gadgets make us dizzy, stoned with information and fried by electromagnetic radiation. However, the reality of everyday computer use is (fortunately?) less exciting than the images in *Neuromancer* suggest. User interfaces are usually tiresome and not at all appealing. So except for the fanatic, the average user does easily get bored. Would this change when virtual reality techniques are applied pervasively? What is virtual reality?

virtual reality

> *virtual reality (is) when and where the computer disappears and you become the 'ghost in the machine' ...*

In other words, virtual reality is a technology that provokes immersion, sensuous immersion, supported by rich media and powerful 3D graphics. In our age of information, we may wonder how all that information should be presented. Rephrasing the question, we may ask what are the limits of the digital experience, or more importantly, what should be the norm: 3D virtual environments, plain text, or some form of XP?

[5] www.gvu.gatech.edu/virtual

1.2 technological developments

Let's see if we are able to give a more precise characterization of *digital convergence.* In their introduction to the theme issue of the Scientific American, Forman and SaintJohn locate the beginning of digital convergence, historically, at the 1939 New York World Fair, and more in particular the RCA Pavillion, which should be considered as the formal debut of television broadcast. They observe that

history

> *the receiver at the RCA Pavillon was way ahead of its time, it was a combination of television - radio - recorder - playback - facsimile - projector ...*

Moreover, they remark that this *in hindsight suggests that we humans have a fundamental desire to merge all media in one entity.*

By way of definition we may state, following Forman and SaintJohn, that digital convergence is:

digital convergence

> the union of audio, video and data communication into a single source, received on a single device, delivered by a single connection

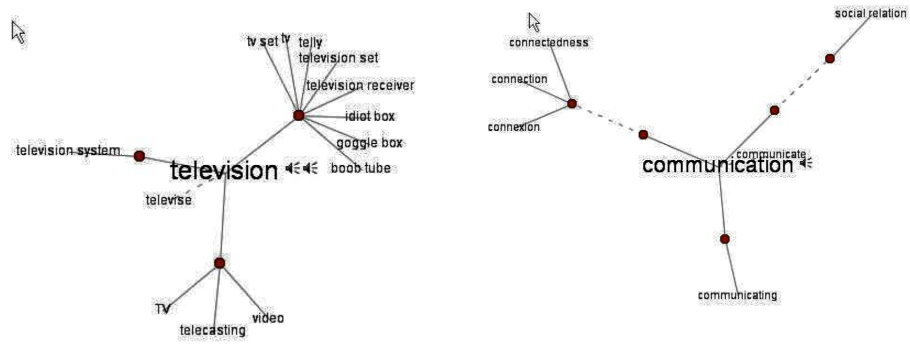
And, as they say, *predicted for decades, convergence is finally emerging, albeit in a haphazard fashion.*

Taking a somewhat closer look, we may discern subsidiary convergences with respect to content, platform and distribution:

subsidiary convergences

- *content* – audio, video, data
- *platform* – PC, TV, internet, game machine
- *distribution* – how it gets to your platform

Here, Forman and SaintJohn continue by speculating that if compatibility standards and data protection schemas can be worked out, all d-entertainment will converge into a single source *that can shine into your life on any screen, whereever you are ...* However, observe that the number of competing standards and architectures is enormous, and that apart from the technical issues involved it is not entirely clear what business model should underly such convergence. In computer shops, there PCs with TV receivers are sold in the range of 1000-2000 euro. This does not include the screen. They come with either the XP Home or Windows Media Center. One of the first in this line of machines, in the higher prices range, was the Sony W1.



4

TV or PC

It is fair to say that no device has changed the way we live so dramatically as television. Television, for one, has altered the way we furnish our living rooms, not to speak about the time we waste watching the thing. Comparing the graphs for *television* and *communication*, we immediately see that their underlying concepts are very different. And more specifically, the association of television with a phrase such as idiot box may raise doubt whether the promise of convergence, which does include communication as an essential feature, will ever become a reality.

Now, we may wonder what interactive television and enhanced televison have to offer us. Looking back, we may observe that it takes some time for the new possibilities to catch on. For example, interactive television was introduced in 1970, but apparently people did not want to communicate with the broadcaster. As another example of enhanced televison, take Big Brother. Although many people watched Big Brother when it first appeared on television, the willingness of the audience to react other than by phone was (apparently) somewhat disappointing. Perhaps, in the Netherlands this was due to the fact that only a fraction of the PC owners was, at that time, permanently online.

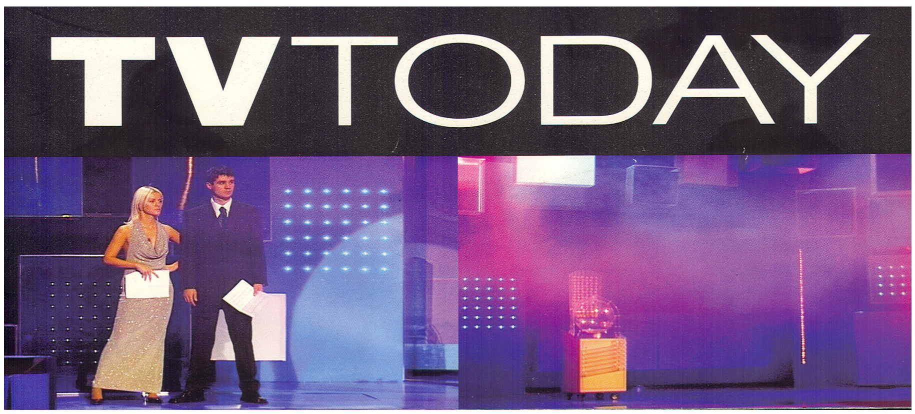
In spite of the failed experiments, Forman and SaintJohn state, somewhat optimistically, that *the convergence of digital content, broadcast distribution and display platforms create the* big *convergence of d-entertainment and information with feedback supporting human interactivity.*

Before looking at *digital television* more closely, let's summarize what digital convergence involves:

convergence

- *content* – 2D/3D graphics, data, video, audio
- *distribution* – broadcast, wireless, DVD, internet, satelite, cable
- *platform* – PC, television, game machine, wireless data pad, mobile phone

This summary indicates the technical opportunities, and the possible functinal extensions that may enhance our use of television, computer, game console and mobile phone. As concerns digital television, we may come up with some immediate advantages, such as enhanced resolution, a multiplication of channels, and (more relevant to the issue of convergence) interactive television.



exposition on the history of TV in Institute for Time-based Arts/Montevideo[6]

5

To get you failiar with some common acronyms, when speaking about (digital) television, we must make a further distinction between:

- HDTV – high definition television
- SDTV – standard definition television
- ITV – interactive television

As further discussed in chapter 3, we have (standard) codecs for d-TV, in particular MPEG-2, for recording digital video, and MPEG-4, for high-quality streaming video on the internet, both from the Motion Picture Expert Group, that enable the effective delivery of digital video, possibly in combination with other content.

Unfortunately, experts disagree on what might become the most suitable appliance or platform to consume all those digital goodies. Here is a list of possible choices:

a *killer* d-TV appliance ...

- DVD player/recorder – 400.000 sold in 2 years, 2h of MPEG-2 video
- personal television – TiVo, Replay-TV (MPEG-2 cache)
- game machine – Sony PS 2, X-Box

Will we prefer to watch stored video, instead of live televison broadcasts? Will the Internet be able to compete with traditional television broadcasting. Will DelayTV or Replay-TV, which allows you to watch previous broadcasts at a time that suits you become popular? Will an extended game machine or PC replace your television? Currently,we must observe that streaming media (still) have rather poor resolution.

Leaving game machines aside, will it then be the TV or PC that will become our platform of choice? Forman and SaintJohn observe:

TV or PC

> *The roadblock to the Entertainment PC could be the PC itself. Even a cheap TV doesn't crash or freeze. The best computers still do.*

However, they conclude that it might make sense to adopt a programmable PC that can support competing TV standards, rather than construct a stack of TV peripherals. Nevertheless, there are a number of problems that occur when we (collectively) choose for the PC as our platform for d-entertainment. Should we have thin clients, for example based on the Sun/Java platform or so-called fat clients based on some version of Microsoft windows> How do we handle the fact that the current internet protocols are not robust, and how can we provide what is known as *quality of service*? Should we adopt any of the proprietary architectures and codecs, such as RealVideo, QuickTime, Windows media, or should we adhere to an open standard such as MPEG-4?

Evidently, the situation becomes even more complex when we just consider the range of alternatives for connectivity, that is for possible ways of distributing contents:

distribution

- *telephone network* – from 0.5 - 2 Mbps to 60 Mpbs (2.5km)
- *broadcast TV* – 6 MHz / 19 Mbps (4 channels MPEG HDTV)
- *cable TV* – hybrid fiber-optic coaxial cable 6 Mbps
- *fixed wireless* – 2 Mbps (radiotowers + rooftop antenna), phones/handhelds
- *satellite* – downloads to 100kbps, modem for uploads ...

Most probably, convergence with respect to distribution will not result in one single way of being connected, but rather a range of options from which one will be selected transparently, dependent on content and availability.

Let's stay optimistic, and ask ourselves the following question:

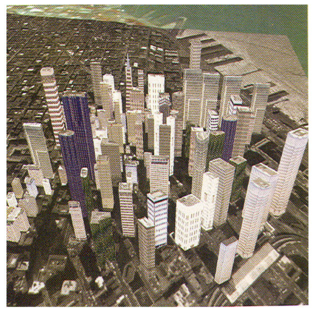
what will we do with convergence once we have it?

One possible scenario, not too unlikely after all, is to deploy it for installing computing devices everywhere, to allow for, to name a few, smart houses, smart clothes, or, in other words, to create a smart world. I wonder what a smart world will look like. In the end we will have to wait and see, but whatever will emerge

we will watch

That is to say, it is not likely that we will have a world without television. Television as we are used to it seems to be the dominant paradigm for d-entertainment, for both the near and distant future.

Berkeley mesh



San Francisco view

augmented terrain map

6

example(s) – *visible world*

Just imagine that every visible place on earth would be accessible in a virtual world. Researchers of the Georgia Institute of Technology[7], Atlanta, have developed software for the semi-automated construction of detailed interactive urban environments, that takes data from multiple sources, including geo-corrected imagery from aerial photography and satelites and ground-based close-ups, Rosenblum and Macedonia (2002).

The challenge here is to collect data from multiple sources and convert this into models, and perhaps even more difficult, to make the models visible so that they can be navigated in an interactive fashion. Recently, the Georgia group teamed up with a group from Berkeley to develop more complex models (images on the left), and together they are working on automating the extraction of information from aerial pictures (image on the right), in particular the detection of groups of trees, and height estimation.

There are many applications for such technology, including urban planning, emergency response, tourism and entertainment, military operations, traffic management, construction and maintenance, mobile services, citizen-government relations, and (not in the least) games.

The next step might be to connect the cameras, that are already there in many of these places, to the model, to observe what happens there in real life. But, somehow, this vision becomes frightening.

However, if you want to give it a try yourself, and populate the virtual globe with your own creations, go download the viewer and editing tool from *Google Earth*:

Google Earth

- Earth – earth.google.com
- SketchUp – sketchup.google.com/download.html

and read the tutorials!

[7]www.gvu.gatech.edu/datavis/research

research directions– *technological determinism*

Although there are many technical issues involved in (digital) multimedia, as exemplified in the issues that play a role in digital convergence, a technical perspective alone does not suffice. Each technological innovation has its consequences on our social life. Conversely, each trend in society might result in the adoption or development of new technology. Looking at the history of the media, we may observe that media become *materials* in our social processes. Or, as phrased in Briggs and Burke (2001):

media as materials

> *each medium of communication tended to create a dangerous monopoly of knowledge*

For example (Briggs and Burke (2001), p. 8) for Christians, images where both a means of conveying information and a means of persuasion, that is part of the rethorics of institutionalized religion.

Looking at our age, and the media that have come into existence in the previous century (radio, television, ...), Briggs and Burke (2001) observe that:

technological determinism

> *technological determinism was not the answer, ... more attempts were to be made to provide answers about the social consequences of television than had ever been asked about radio.*

In effect, underlying all developments in the media (including the computer) we may assume a basic need for information. A rather problematic need, for that matter:

information

> *Information became a major concern anywhere during the late 1960 and 1970s where there was simultaneous talk both of 'lack of information' and 'information saturation'.* Briggs and Burke (2001), p. 555

Nowadays, we regard information as a commodity. Train schedules, movies, roadmaps, touristic information, stock prices, we expect it all to be there, preferably online, at no cost. No information, no life. Information drives the economy. Upwards and downwards!

1.3 multimedia applications

In many stores there is a multimedia section. In some stores you will see B-movies being announced as *multimedia topper*. In other stores, the multimedia sections has a large offering of computer peripherals, ranging from DVD-RW drives to webcams and TV on PC hardware. Elsewhere you may buy authoring packages to organize your cell-phone photos, your family photo and video album, to create your personal archive on DVD. All this might make you wonder whether multimedia is serious business. See figure (a) and (b), illustrating our personal *memex*, as explained below.

But more seriously, what is the commercial impact multimedia and in particular digital convergence may have? And, perhaps equally important, why should we be interested in this from, I must say, an academic perspective?

In this last section of the introductory chapter, we will look at some popular press items related to new media, mixed media (in particular the merge of TV and internet) and mobile multimedia. We will then briefly reflect on what significance these issues have from our academic perspective.

7

new media

As you may be read in the newspapers in the beginning of this century, large investments have been made (by both cable and telephone companies) to improve the technological infrastructure for the new media. Simultaneously, joint ventures have arisen between content developers and providers, as with the Dutch Endemol company.

Now, what does the popular press have to say about all these developments. Here is one comment, from a Dutch newspaper:

Peter Greven 23/3/2001 (Volkskrant)

> new media sucks – people like new technology. they don't like new media.

The translation from Dutch is, admittedly, mine. It says, in other words, that people like to receive the old stuff on new gadgets, but that they are not willing to pay for any new sort of services. For example, when considering the smart video recorder, that uses a disk cache for storing MPEG coded versions of broadcasts, just think of other gadgets and services that didn't make it or that are encountering problems in being accepted. Some famous examples from the past are the videofoon, videotext, cd-i, and DCC.

Perhaps the reason for these failures is the *trial-and-error* method,, also referred to as the spaghetti method, that is being followed in developing new media. As characterized by Jan van Dijk, of a dutch university in the east of the Netherlands (Twente), the spaghetti method consists of throwing a plate against the wall, and see what will stick to the wall. In other words, just throw your product on the market and see whether it will stick. Perhaps that is not the right method to be followed. But can you think of a better one?

In many cases 'the market', that is the people using a service, do not behave as expected. For example in Sweden, the upload of material far exceeded download, which is contrary to the assumptions underlying ADSL.

TV meets the Web

At first sight it seems promising to develop mixed media. As an example, a dutch agency announced services to support the integration of TV and the Web, promising the integration of

www.tvmeetstheweb.com

> *streaming media (audio and video), interactive gaming, virtual reality and 3D animation, interactive TV programming, interactive advertising, video on-demand, webcasting and multimedia*

In 2000 they issued a report sketching the European broadband landscape. Quoting from this report: *The advent of broadband Internet access, which has been available in the US for some time but is only now beginning to make inroads into Europe, makes a whole range of new services possible. As download speeds have increased and more bandwidth has become available, the possibility of delivering screen-based content such as films, television programs and music has moved a step closer to mass market usage.* With respect to the adoption of cable or DSL in Europe, they observe that despite the fact that cable companies have gained firm ground, there is an even larger number of conventional telephone lines, around 180 million. In contrast, there are only 15 million cable subscribers, giving DSL a large potential audience. Matthijs Leendertse, co-author of the report, observes: *Gaining competitive advantage and future revenue in Europe's broadband landscape will depend heavily on a company's ability to offer integrated services: access (fixed and wireless) and content. It is virtually impossible at this point for one single company to offer these services on a pan-European level. This means that companies need to find partners to fill the gaps in their offerings.* Let me assure you, at the moment you will be reading this the battle is still going on!

mobile multimedia

Let's look at another potential hype. In 2000, Webnoize published a report (by Matt Bailey), entitled *Wireless Entertainment: What Is It Worth?*, which introduces the *wireless web*, and predicts that *young media junkies* will demand music videos and animations, and listen to wirelessly streamed music.

The intent of the report is to investigate whether investments in the mobile entertainment are justified. The report examines how providers of music and video services can benefit from the wireless delivery of multimedia. Using survey evidence, pricing information from new wireless networks and interviews with industry visionaries, the report analyzes supply and demand to build an economic and business model for mobile multimedia.

Apart from the need to invent some business model, there are a number of strategic questions to be answered in order to estimate the risk of making investments in this direction. Following Bailey, we may list questions such as:

strategic questions

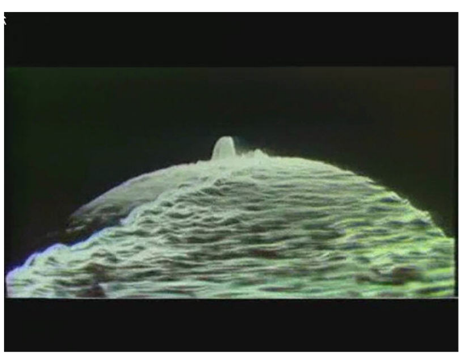
- how quickly will wireless connectivity speeds improve?
- what is the demand for services that deliver music and video to wireless devices?
- how can suppliers of multimedia services monetize demand for wireless access?
- how much will it cost to stream multimedia content to wireless devices now and in 2006?
- are consumers willing to compromise quality for lower cost?

And more. If you are interested whether anyone is willing to take such risks and invest in mobile multimedia, just look at what players were involved.

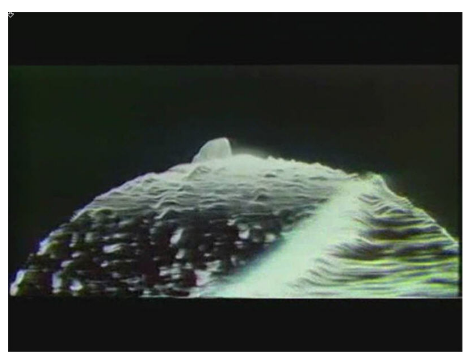
the players

> *Alltel, AT&T Wireless, AtomShockwave, Cingular Wireless, Clear Channel, HitHive, Ifilm, Infinity, KDDI, Liquid Audio, LMIV, Mannesmann, MP3.com, MTV, NetCom, Myplay, Nortel Networks, NTT DoCoMo, Omnitel, Sprint, Telefonica, Telstra, Vitaminic, Verizon Wireless, Virgin Megastores, Vodafone, Voicestream.*

Now make up your mind, and ask yourself the question whether multimedia is worth your (intellectual) investment.



Vasulka Objects (1)



Vasulka Objects (2)

8

the academic perspective

Being sensitive to hype is only too human. So also academics may be fascinated by new trends. and get distracted by rumors on the market. Breaking loose from this fascination, we may ask ourselves what are the real issues, and what makes multimedia interesting. Let me start with answering the latter question first. As an academic subject, multimedia is interesting because it offers such an intriguing mix of subjects, including multimedia technology, exploratory design and scientific validation. Commercially, it is safe to say that the volume of entertainment related multimedia content, including games, music and infotainment is substantial, and hence its economic interest is indisputable. But what are the real issues?

One of the examples of multimedia applications I will present in the last chapter is an application in the domain of cultural heritage. For this domain we have developed so-called *digital dossiers* containing a representation of the work(s) of a particular artist as well as information that characterizes the work in its historic and cultural context, needed for the re-exposition or installation of the work. Problems facing the developer of a digital dossier cover the interaction of the user with the dossier, the presentation of both textual and multimedia information, and facilities for search and navigation. And there are technical issues, such as which codecs to select for the videos and how to manage the content included in the dossier. Developing a dossier is not as one might naively think the creation of content only, but rather involves designing the functionality of the application as well.

Generalizing from the domain of cultural heritage to the area of infotainment and multimedia information systems, where an integrated presentation of textual and multimedia information must be achieved, we may boldly state that designing the functionality of the application is the most crucial issue, and as such of primary academic interest. All other topics, including multimedia technology, compression algorithms, software engineering, multimedia platform support and information retrieval techniques, may be regarded in some loose sense to be subservient to the issue of design.

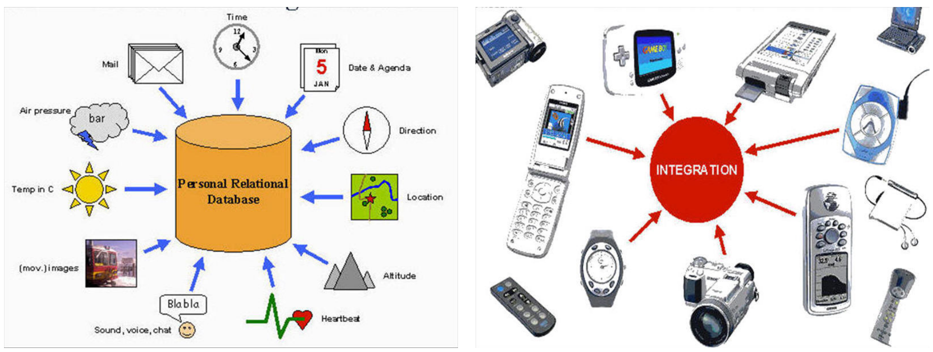
digital art As the illustrations in the text testify, another personal motivation for being involved in multimedia comes from the area of digital art. And, with students I observe a similar interest in the potential digital content authoring offers as a vehicle for personal expression.

One of the artists of which I included material in this book is Woody Vasulka, who was a pioneer in the early days of video and computer art. In an interview, held in 1985 with Rene Coelho, the founder of Montevideo[8], Vasulka explained his fascination with the scan processor and later the video computer by stating that it allowed him to *invent the image*. Still, howver, as he said, in some sense traditional painting acted as a visual reference system by which to judge the images produced with the new technology. Later in the interview, he observed that after some time he became bored with the images produced this way, and he started to feel the need to include more narrative in his work. His wife, Steina Vasulka, with whom he founded the Kitchen, a gathering place for new media artists in New York in the 1970s, remarked that in the early phase she was struck by the fact that *the material was so friendly*, that is how easy it was to express your ideas.

These words suffice to emphasize the importance of the motivation you might

[8] www.montevideo.nl

get out of the material, to be susceptible to as Brancusi phrases it *the rethorics of the material*, even when you are an academic.



(a) personal events (b) gadgets

9

example(s) – *personal memex*

Just imagine that your would store all your photographs, SMS messages, emails, and in addition to that record your physiological condition, using body-wearable sensors. These data can then be uploaded to your PC, and later to a mass storage server, so that they can be used in your medical dossier, to improve your performance in your favorite sport, or to augment your memory when recollecting stories about your holidays or travels. Impossible? Not at all, Disk space will be cheap. Your body may act as a network to connect the body wearable devices, and, after all, most of the gadgets do already exist! Besides, the idea is not new. See section 2.2 for early visions of the *memex*.

research directions– *the information society*

There is no doubt about it, we live in an information society. But do we know what an information society is?

In Briggs and Burke (2001) (p. 187), the functions of the media are summarized as

functions of media

information, education, entertainment

So, perhaps, we could better state that we live in a *media society*. So far, in the latter part of the previous century, television has dominated our lives, and observe that (following Ernie Kovack, cited from Briggs and Burke (2001)):

medium

television is a medium 'because it is neither rare nor well done'

Back to the main issues, what is an *information society*? According to Briggs and Burke (2001):

information society

> *the new term 'information society' gave form to a cluster of hitherto more loosely related aspects of communication – knowledge, news, literature, entertainment, all exchanged through different media and different media materials – paper, ink, canvas, paint, celluloid, cinema, radio, television and computers. From the 1960s onwards, all messages, public and private, verbal and visual, began to be considered as 'data', information that could be transmitted, collected, recorded, whatever their point of origin, most effective through electronic technology.*

So, from the varieties of perspectives we have discerned, including technological perspectives, societal perspectives and psychological perspectives, we must investigate the problem of communication:

communication

- *what* – content
- *who* – control
- *whom* – audience (how many)

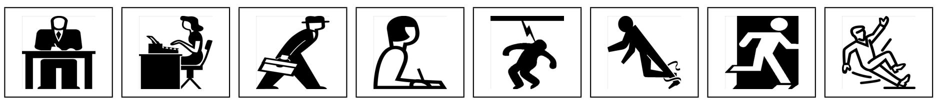
That is, simply, who says what to whom in what channel with what effect?! The remainder of the book will, however, will treat these issues mainly from a technological perspective. In the chapters that follow, we will enquire after the technological assumptions that make an information society possible.

1.4 development(s) – convergence in second life

第二生命

> Second Life seems to be overtaking the world. In the whole range of cummunity-building platforms, Second Life stands out as an immersive 3D world with an almost stunning adoption, by both individuals, companies and institutions, followed attentively by the Press. Not entirely without an understanding of the value of press coverage, the VU University Amsterdam decided to create presence in Second Life, by creating a virtual campus, to realize a (virtual) community of learners, VUSL. And, indeed, we succeeded in being the first university in The Netherlands with presence in Second Life and, as hoped, this was covered in the 8 o'clock nation-wide TV news.

More substantial than getting into a nation-wide television broadcast, however, is our aim to communicate our institutional goals, *creating a community of learners*, by creating a virtual campus in Second Life, offering *an information portal* as well as *a meeting point*, in a media platform that is widely adopted by our target community. Virtual presence in Second Life, obviously, is not enough. The relatively long history of virtual worlds has shown that lack of interesting content and functionality easily leads to boredom, desinterest, and hence *churn*, users dropping off. As a consequence, there is a need for sustainable functionality, that both motivates people to come back and participate, and, otherwise why choose Second Life, makes essential use of the 3D immersive environment offered by Second Life. In this paper, we will explore how to use web services in meaningful compositions or mashups to enhance our presence in Second Life, and create a community where visitors actively participate in both education and research,



10

questions

digital convergence

1. Sketch the developments in *multimedia*. What do you expect to be the commercial impact of multimedia in the (near) future?

concepts

2. Explain what is meant by *digital convergence*.
3. Which kinds of *(digital) convergence* do we have?
4. Discuss the relation between the *medium* and the *message*.

technology

5. Give a brief sketch of the development of *digital entertainment*.
6. Characterize: HDTV, SDTV, ITV.
7. Discuss convergence with respect to *platforms*.
8. Discuss convergence with respect to *delivery*.

projects & further reading As a project, consider the development of a Java-based mobile game using J2ME, see Morrison (2005), or a web-based game using Visual Basic .NET, see Santos Lobao and Hatton (2003).

You may further explore multiplatform game development, and find arguments to choose for either Java-based or managed code based implementations.

For further reading, I advice to have a look at the special issues of the Scientific American, American, and the CACM on the next 1000 years of computing, CACM (2001), and, for getting an idea where this all leads to, Schneidermann's *Leonardo's laptop*, Shneiderman (2003).

the artwork

1. photographs of art works by Marina Abramovic, *Art must be beautiful*, *Blue period*, *Dissolution*, *Dozing consciousness*, *In between*, with (pending) permission from Montevideo[9]. See also section 10.2.
2. *medium*, according to the Visual Thesaurus[10].

[9] www.montevideo/nl
[10] www.visualthesaurus.com

3. fMRI Research on Virtual Reality Analgesia[11], see section 1.1.
4. *television* and *communication*, according to the Visual Thesaurus.
5. TV Today, exhibition at Montevideo, februari 2005.
6. visible world – taken from Rosenblum and Macedonia (2002), see section 1.2.
7. personal event database and personal gadgets, from Freeband[12] project.
8. *Thomas Lips 1975*, *Thomas Lips 1993*, from Marina Abramovic, with permission from Montevideo.
9. *scanlines* from Woody Vasulka[13], 197x, with permission from the artist.
10. signs – people, van Rooijen (2003), p. 254, 256.

The work of Marina Abramovic has a strong *existential* flavor. It has also served as the material for a case study in developing a digital artist dossiers, the *abramovic dossier*, discussed in section 10.2. The work of Woody Vasulka is of a more *experimental* character, and shows the joy of discovering the possibilities of the, at the time, new electronic and digital tools and materials.

[11] www.hitl.washington.edu/research/magnet
[12] www.freeband.nl
[13] www.vasulka.org

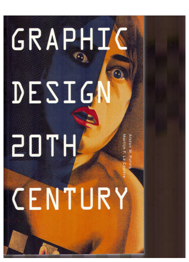
2. hypermedia information spaces

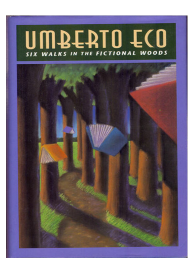
everything must be intertwinkled <xx>

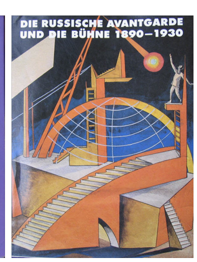
learning objectives

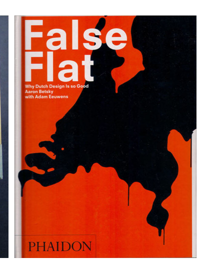
After reading this chapter you should be able to define information spaces in a precise manner, position the hypertextual capabilities of the web in a historical perspective, explain the difference between multimedia and hypermedia, and argue why computational support for narrative structure in multimedia applications is desirable.

However entertaining it might be presented to you, underlying every multimedia presentation there is an information space. That is to say, irrespective of the medium, there is a message. And being confronted with a message, we might want to inquire for more information. In this chapter, we will define the notion of information space more precisely. We will extend this definition to include information hyperspaces, by looking at the history of hypertext and hypermedia. Finally, we will discuss visualisation as a means to present (abstract) information in a more intuitive way, and we will reflect on what is involved in creating compelling multimedia.









1

2.1 information and data

Current day *multimedia information systems* distinguish themselves from older day information systems not only by what information they contain, that includes multimedia objects such as images and sounds, but also by a much more extensive repertoire of query mechanisms, visual interfaces and rich presentation facilities. See Chang and Costabile (1997).

Preceding the advent of multimedia information systems, which include networked multimedia systems as discussed in section 6.3, we have seen advances in

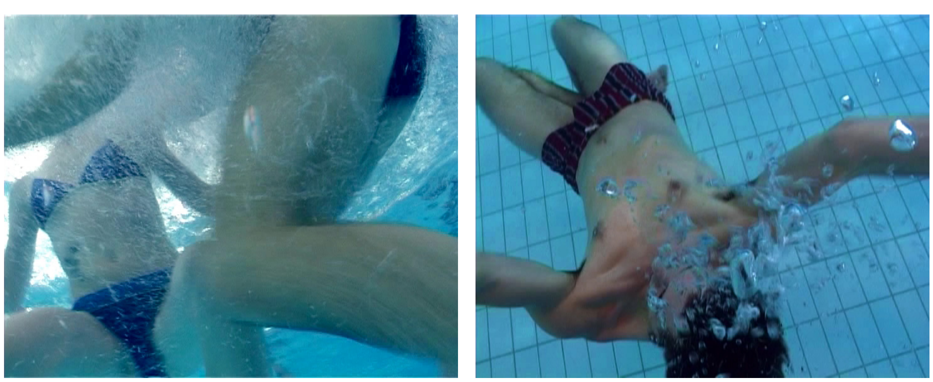
multimedia information systems

- *storage technology – multimedia databases*
- *wideband communication – distribution accross networks*
- *parallel computing – voice, image and video processing*
- *graphic co-processors – visual information with high image quality*

Now, the class of *multimedia information systems* is, admittedly, a large one and includes applications and application areas such as:

geographical information systems, office automation, distance learning, health care, computer aided design, scientific visualization, and information visualization.

Nevertheless, irrespective of what technology is used for storage and retrieval, multimedia information systems or multimedia databases impose specific requirements, with respect to: the size of data, synchronisation issues, query mechanisms and real-time processing.
Partly, these requirements concern the efficiency of storage and retrieval and partly they concern aspects of usability, that is the way information is presented to the user. In particular, we can think of a great number of query mechanisms that our multimedia information system of choice is expected to support: free text search, SQL-like querying, icon-based techniques, querying based on ER-diagrams, content-based querying, sound-based querying, query by example, and virtual reality techniques.



2

logical information spaces

But before thinking about the optimal architecture of multimedia information systems or the way the information is presented to the user, let's consider in what way a multimedia (information) system or presentation may be considered an *information space*.

As a tentative definition, let's assume that

> *an information space is a representation of the information stored in a system or database that is used to present that information to a user.*

This may sound too abstract for most of you, so let's have a look at this defininition in more detail.

First of all, observe that when we speak of representation, and when we choose for example a visual representation, then the representation chosen might be either the users conceptualization of the database, or a system generated visualization. In principle the same holds for a text-based representation, but this is far less interesting because the options in choosing a representation and presenting it to the user are much more limited.

Unfortunately, the phrase *representation* is also somewhat vague. To be more precise, we must distinguish between a *visual information space* (for presentation), a *logical information space* (in which we can reason about abstract information objects) and a *physical information space* (where our concrete multimedia objects are stored).

Summarizing we have:

- *physical information space* – images, animations, video, voice, ...
- *logical information space* – abstract database objects
- *presentational information space* – to present information to the user

Our visual information space, our presentation space, as you may prefer to call it, might reflect the logical information space in a symbolic manner by using diagrams, icons, text and possibly visualzations, or, going one step further, it may also mimic the logical information space by using virtual reality, as discussed in chapter 8.

Now we can give a more precise definition of the notion of information space, in particular *logical information spaces*:

> *a logical information space is a multidimensional space where each point represents an object from the physical information space (read database).*

First of all, observe that when we speak of dimensions we might also speak of attributes that can take either continuous, numerical, discrete or logical values. So, concretely, these attributes may be directly or indirectly related to information stored in the database, and hence we can give a more precise definition of the notion of (multimedia) information objects, queries and *cues* (in the logical information space):

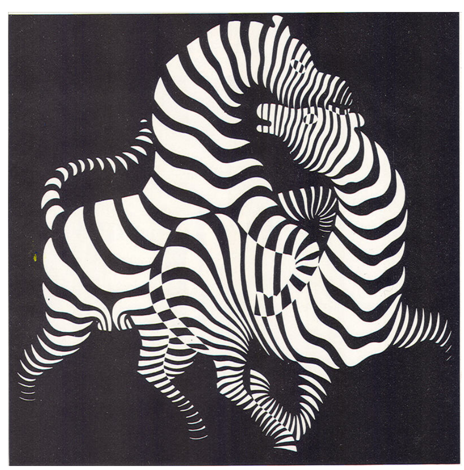
- *information object* – a point in the (logical) information space
- *query* – an arbitrary region in this information space
- *clue* – a region with *directional information*, to facilitate browsing

The notion of *clue* is actually quite interesting, since both examples and queries may be regarded as clues, that facilitate browsing through the contents of an information space. As an example, just think of the situation that, when looking for another notebook, you want something that is similar to the the thing you've previously seen, but that has an additional video output slot that may be connected to your TV.

Also, clues are needed to allow for *query by example*. In this case you need to help the user to define a query in the logical information space, so that the system can construct an *optimal query* to search for the desired object(s) in the physical information space.

When we regard *the information retrieval problem* to be the construction of the *optimal query* with respect to the examples and clues presented by the user, then we may characterize the *optimal query* as the one that will retrieve the largest number of relevant database objects within the smallest possible region in the (logical) information space.

extensions Given the stratification, that is levels or layers, of information systems discussed above, we can think of improvements or extensions on each level. At the physical layer, for example networked multimedia, in a client/server architecture, see 6.3. At the logical layer, as an information hyper space,consisting of chunks and hyperlinks, as explained in section 2.2. And at the presentation layer a virtual reality interface, representing for example the physical location of student records, somewhere at a virtual campus [x], as further explored in chapter 8. Each of these improvements or extensions can be regarded as a technological or scientific adventure in it's own right.



example(s) – *e-flux*

Do you recognize this?

> When we visit a contemporary art exhibition, we find ourselves before the works, which are often quite difficult to interpret, and we observe them without understanding the process that generated them. Between a chopped-up cow immersed in formaldehyde and a replica of the Pope blindsided by a meteorite, it's legitimate to ask questions.

To provide a counter-force the exhibiton Project Room[14] *challenges the usual exhibition routine and decides to not exhibit executed art works but rather offers ten self-interviewing videos by as many artists, who speak openly about a piece they are working on, or a visionary project they want to realize, or about their creative process.*

In other words, this is about works of art with no physical manifestion. It is an interesting issue whether this would still count as a *work of art*. And for multimedia, is there multimedia without a physical manifestation, with sensorily impressing the user/client. Do you remember the children story, the *New Clothes of the Emperor*?

research directions– *universal interchange*

Technology changes rapidly. Just think about the development of the PC in the last two decades of the previous century. And applications change rapidly too. At the time of writing the web does barely exist for ten years. Information spaces, on the other hand, from a sufficiently abstract perspective at least, should be rather stable over time. So the question is, *how can we encode information content in an application-independent way?* As a remark, application-independence implies technology-independence. The answer is, simply, XML. The next question then should be, what is XML and why is it more suitable for encoding information then any of the other formats, such as for example relational tables.

The first question is not so difficult. There are many sources from where an answer may be obtained. Perhaps too many. A good place to start is the XML FAQ (Frequently Asked Questions) at the Web Consortium site:

www.w3.org/XML/1999/XML-in-10-points

> *XML is a set of rules (you may also think of them as guidelines or conventions) for designing text formats that let you structure your data.*

More specifically, XML may be characterized as follows:

XML in 10 points

1. XML is for structuring data
2. XML looks a bit like HTML
3. XML is text, but isn't meant to be read
4. XML is verbose by design

[14] www.e-flux.com

5. XML is a family of technologies
6. XML is new, but not that new
7. XML leads HTML to XHTML
8. XML is the basis for RDF and the Semantic Web
9. XML is license-free, platform-independent and well-supported

Perhaps not all of these points make sense to you at this stage. So let me first indicate that XML has in fact quite a long history. XML is the successor of SGML (the Structured Generalized Markup Language) that was developed in the 1980s to encode documents (such as airplane manuals) in an application-independent manner. SGML is not a language itself, but a descritpion of how to create a content description language, using tags and attributes (as in HTML). In fact, HTML is an application of SGML, using tags with attributes both for formatting and hyperlinks. In other words, SGML is a meta language. And so is XML. Since everything got messy on the web, XML was proposed (as a subset of SGML) to make a clear distinction between content and presentation. Presentation aspects should be taken care of by stylesheets (see below) whereas the content was to be described using and XML-based language.

Now, why is XML a suitable format for encoding data? That question is a bit harder to answer. One of the reasons to use XML might be that it comes with a powerful set of related technologies (including facilities to write stylesheets):

related technologies

- Xlink – hyperlinks
- XPointer – anchors and fragments
- XSL – advanced stylesheets
- XSLT – transformation language
- DOM – object model for application programmer interface
- schemas – to specify the structure of XML documents

These technologies (that are, by the way, still in development) provide the support needed by applications to do something useful with the XML-encoded information. By itself, XML does not provide anything but a way to encode data in a meaningful manner. Meaning, however, comes by virtue of applications that make use of the (well-structured) data.

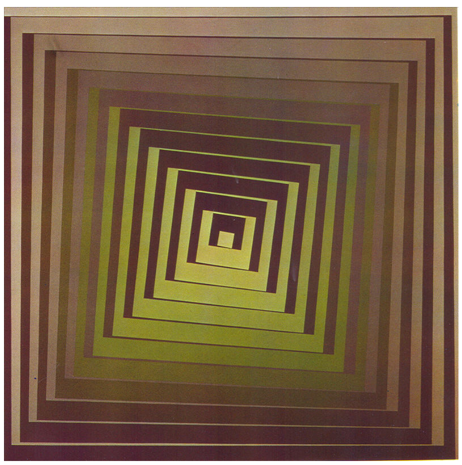
In summary, XML and its related technologies provide the means to

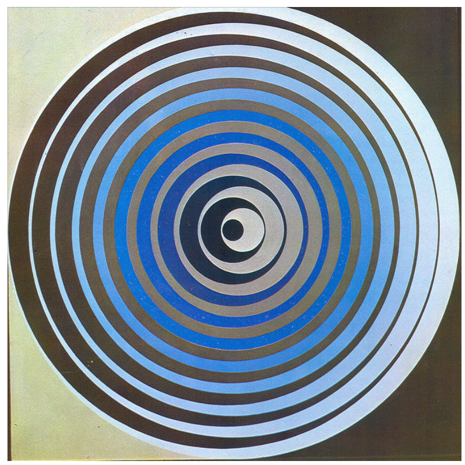
XML

- separate data from presentation
- transmit data between applications

Actually, the fact that XML was useful also for arbitrary data interchange became fully apperent when XML was available. To get an impression of what XML is used for nowadays, look at www.xml.org.

This leaves us with the question of why XML is to be preferred over other candidate technolgies, such as relational databases and SQL. According to Kay (2001), the answer to that question is simply that XML provides a richer data structure to encode information. In the multimedia domain we see that XML is widely adopted as an encoding format, see section ??. For an example you might want to have a look at MusicXML, an interchange format for notation, analysis, retrieval, and performance applications, that is able to deal with common Western musical notation as used from the 17th century onwards. In appendix we will explore how XML might be useful for your own multimedia application by treating some simple examples.





4

2.2 hypermedia

Given an information space we may turn it into an information hyperspace, that is, following Chang and Costabile (1997),

information hyperspace

> the logical information space may further be structured in a *logical information hyperspace*, where the clues become hyperlinks that provide directional information, and the information space can be navigated by the user following directional clues.

In other words, information is chunked, and each chunk is illustrated or made accessible by an example (hypernode).

Now, what exactly does *information hyperspace* mean? To answer this question, let's briefly look at the history of hypertext and hypermedia.

history

- 1945 – Vannevar Bush (Memex) – as we may think, Bush (1995)
- 1963 – Douglas Engelbart (Augment) – boosting the human intellect Engelbart (1963)
- 1980 – Ted Nelson (Xanadu) – everything is intertwinkled, Nelson (1980)

Vannevar Bush' seminal paper *As we may think* may be regarded as the origin of what is known as *hypertext* with which, even if you don't know the phrase, every one of you is familiar, since it is (albeit in a rather simple way) realized in the web.

The phrase *hypertext* was invented by Ted Nelson (not patented, as far as I know), who looked for a less constraining way to organize information then was common in the educational system he grew up with. But before that, Douglas Engelbarth, who incidently invented the mouse, developed the Augment system to, as he said, *boost the human intellect*. What for, you may ask. Let me quote the series of flashes that Engelbarth went through, according to *Dust or Magic* Hughes (2000):

- *flash 1*: we are in trouble (human mankind)
- *flash 2*: we need to boost mankind's ability to deal with complex urgent problems
- *flash 3*: aha, graphic vision surges forth of me ...
- *flash 4*: hypermedia – to augment the human intellect
- *flash 5*: augment (multimedia) workstation – portal into an information space

classification of hypermedia

Perhaps it is good to know that Vannevar Bush wrote his article when working for an information agency in the second world war period. From that perspective, we can easily see that hypermedia (combining hypertext and multimedia) were thought of an instruments of intelligence.

Basically, hypermedia systems must be able to deal with:

hypermedia systems

- components – *text, graphics, audio, video*
- links – *relations between components*
- presentation – *structured display*

Far from being a definition, this characterization gives some insight in what functionality hypermedia systems must support. Recall that dealing with complex information is what hypermedia is all about.

Is this a natural way to deal with information? Just think about how you are taught to deal with information and how you actually go about with it. Speaking about Ted Nelson, Hughes (2000) observed that *he realized that this intertwingularity was totally at odds with the education system he spent so long in and had been so uncomfortable with.* Quoting Ted Nelson himself from his book *Literary Machines*:

> *A curriculum promotes a false simplification of any subject, cutting the subject's many interconnections and leaving a skeleton of sequence which is only a charicature of its richness and intrinsic fascination.*

Judge for yourself. Would you prefer to have an 'immersive' course in multimedia rather than a more or less ordered collection of abstractions?

True enough, the visions of the pioneers of hypermedia where overwhelming. Nevertheless, the concept of hypermedia, that is non-linear media with machine-supported links, or *'text' as a network*, found an application in a large variety of systems, see McKnight et al. (1991).

classification of hypermedia systems

- macro-literary systems – *publishing, reading, criticism*
- problem exploration tools – *authoring, outlining, programming*
- browsing systems – *teaching, references, information*
- general hypermedia technology – *authoring, browsing, collaboration*
- embedded hypermedia – *CASE, decision support, catalogs*

An example of a hypermedia system that has extensively been used in education, for example biology and chemistry classes, is the Brown University Intermedia system of which supports so-called *information webs*, consisting of *documents* and *links*, that could both be retrieved by specifying attribute, allowing in this way for respectively both filtered content and conditional navigation. An interesting aspect of this system is that the user may create *maps*, that is structures containing documents and links, which form a personalized version of the web of information for a specific user, superimposed on the information space offered by the system.

5

Dexter Hypertext Reference Model

After many years of developing ideas and exploring implementations, one group of experts in the field came together and developed what is commonly known as the *Dexter Hypertext Reference Model*, named after the location, actually a pub, where the meetings were held. The Dexter model offers an abstract description of *hypertext*. It made a distinction between *components*, *anchors* within components and *links* between components, attached to anchors. The model was meant as a reference standard against which existing and future hypertext systems could be compared.

Components have the following attributes:

component

- content – *text, graphics, video, program*
- attributes – *semantic description*
- anchors – *(bi-directional) links to other documents*
- presentation – *display characteristics*

The Dexter Hypertext Model has been criticised from the beginning. Among others, because *compound documents*, that is documents having subcomponents, where not adequately dealt with. And also because it did not accomodate multimedia (such as video) content very well. In practice, however, the Dexter model has proven to be even somewhat overambitious in some respects. For example, the web does (currently) not support bi-directional links in a straightforward manner.

Amsterdam Hypermedia Model

When looking for alternatives, a Dutch multimedia research group at CWI proposed to extend the Dexter model with their own multimedia model (CMIF), an extension for which they coined the name *Amsterdam Hypermedia Model*.

Let's look at the (CMIF) multimedia model first:

(CMIF) multimedia model

- data block – *atomic component*
- channel – *abstract output device*
- synchronization arc – *specifying timing constraints*
- event – *actual presentation*

What strikes as an immediate difference with respect to the hypertext model is the availability of *channels*, that allow for presenting information simultaneously, and so-called *synchronization arcs*, that allow the author to specify timing constraints. Also, events are introduced in the model to deal with user interactions.

With respect to authoring, the model supports a declarative approach to specifying sequential and parallel compounds, that is in what order specific things must be presented and what may ocuur simultaneously. Again, channels may be employed to offer a choice in the presentation, for example a dutch or english account of a trip in Amsterdam, dependent on the preferences of the (human) viewer.

The Amsterdam Hypermedia Model (AHM) extends the Dexter Hypertext Reference Model in a rather straigthforward way with channels and synchronization arcs.

Amsterdam Hypermedia Model

- contents – *data block*
- attributes – *semantic information*

- anchors – *(id, value)*
- presentation – *channel, duration, ...*

Obviously, the difference between Dexter and AHM is primarily the more precise definition of *presentation characteristics*, by introducing *channels* as in the (CMIF) multimedia model. Another (major) difference lies in the characterization of compounds. Each compound has one or more children, or subcomponents. Subcomponents may act as the source or destination of synchronization arcs. Each component obtains a start-time, that may result from parallel or sequential composition and synchronisation arcs.

Another interesting concept introduced by the Amsterdam Hypermedia Model is the notion of *context*. What happens when you click on a link? Does everything change or are only some parts affected? Then, when you return, does your video fragment start anew or does it take up where you left it? Such and other issues are clarified in the Amsterdam Hypermedia Model, of which we have omitted many details here.

It is perhaps interesting to know that the Amsterdam Hypermedia Model has served as a reference for the SMIL standard discussed in section 3.2. If you want to know more about the Amsterdam Hypermedia Model, you may consult Ossenbruggen (2001) or Hardman et al. (1994).

6

example(s) – *hush*

In the *hush*[15] we explore a variety of hypermedia applications. In fact already in 1994 we developed a SGML-based browser with *applets* in Tcl/Tk, Applications and SGMLWEB. Somehow, we did a lot with music with optimistic titels such as *Bringing music to the We*, Ossenbruggen & Eliens (1994) and more pessimistic ones such as *Jamming (on) the Web*, Eliens *et al.* (1997). The acronym *hush* stands for *hyper utility shell*. Many of the projects with *hush* were student projects, in which we studied operational support for hypermedia applications. Although we used SGML for markup, we did not have any specific document model, as in CMIF. An overview and rationale of *hush* is given in Eliens (2000). A significant part of the *hush* software is being reused in the ViP system, that is discussed in section 4.3, albeit with an entirely different presentation technology.

[15]www,cs,vu.nl/∼eliens/online/hush

research directions– *computational models*

Today, hypermedia functionality is to some extent embedded in almost all applications. However, to realize the full potential of hypermedia, and in effect the networked multimedia computer, there are still many (research) issues to be resolved. To get an impression of the issues involved, have a look at the famous seven hypermedia research issues formulated by Halasz.

research issues

- *search and query* – for better access
- *composition* – for imposing structure
- *virtual structures* – on top of existing structures
- *computation* – for flexibility and interaction
- *versioning* – to store modification histories
- *collaborative work* – sharing objects with multiple users
- *extensibility and tailorability* – to adapt to individual preferences

See Ossenbruggen (2001), section 2.3 for a more extensive description. Although the research issues listed above were formulated quite early in the history of hypermedia, as a reflection on the requirements for second-generation hypermedia systems, they remain valid even today. Without going into any detail with respect to the individual research issues, I rather wish to pose the grand encompassing research issue for the networked multimedia computer: *What is the proper computational model underlying hypermedia or, more generally, for applications that exploit the networked multimedia computer in its full potential*? Some directions that are relevant to this issue will be given in section which deals with the multimedia semantic web.

2.3 multimedia authoring

It is tempting to identify a presentation with the information space it presents. This is what users often do, and perhaps should do. When that happens, the presentation is effective. But you must remember that the actual presentation is just one of the many possible ways to engage a user in exploring an information space. Making the choice of what to present to the user is what we understand by *(multimedia) authoring.*

Authoring is what we will discuss in this section. Not by giving detailed guidelines on how to produce a presentation (although you may look at the online assignment for some hints in this respect), but rather by collecting wisdom from a variety of sources.

7

visualization

Let's start with our explorations by looking at the problem of *visualisation* with a quote from David Gelernter, taken from Shneiderman (1997):

visualization

> *Grasping the whole is a gigantic theme, intellectual history's most important. Ant vision is humanity's usual fate; but seeing the whole is every thinking person's aspiration.* David Gelernter, Mirror Worlds 1992

Now, consider, there are many ways in which the underlying information space may be structured, or speaking as a computer scientist, what data types may be used to represent the (abstract) information.

data types

- *1-D linear data* – text, source code, word index
- *2-D map data* – floor plan, office layout
- *3-D world* – molecules, schematics, ...
- *temporal data* – 1 D (start, finish)
- *multi-dimensional data* – n-dimensional (information) space
- *tree data* – hierarchical
- *network data* – graph structure

The *visualisation problem* then is to find a suitable way to present these structures to the user. Basicall, following Shneiderman (1997), there are two paradigms to present this information:

- *interactive* – overview first, zoom and filter, then details on demand
- *storytelling* – as a paradigm for information presentation

Storytelling may be very compelling, and does not force the user to interact. On the other hand, storytelling may lead to information consumerism alike to television enslavement.

An interaction paradigm that combines 'storytelling' with opportunities for interaction, as for example in the *blendo* approach discussed in section 3.2, would seem to be most favorable. Interaction then may result in either changing the direction of the story, or in the display of additional information or even transactions with a third party (for example to buy some goodies).

persuasive technology

Whatever your target audience, whatever your medium, whatever your message, you have to be convincing if not compelling.

In the tradition of *rethorics*, which is the ancient craft of convincing others, a new line of research has arisen under the name of *persuasive technology.* In the words of my collegue, Claire Dormann, persuasion is:

persuasion

- a communication process in which the communicator seeks to elicit a desired response from his receiver
- a conscious attempt by one individual to change the attitudes, beliefs or behaviours of another individual or group individual through the transmission of some messages.

In other words, *the purpose of persuasion is to accomplish one of the following goals: to induce the audience to take some action, to educate the audience (persuade them to accept to accept information or data), or to provide the audience with an experience.* In the area of multimedia, one may think of many applications. Quoting Claire Dormann, *in* interactive media*, the field of application of persuasive technology ranges from E-commerce, social marketing (like an anti-AIDS campaign) to museum exhibits. Also E-commerce provides an obvious example. To convince people to buy more, more persuasive messages and technologies are developed through the use of humorous and emotional communication, agents (such as price finders) or 3D representations of products and shops. For health campaigns (or any campaign of your choice) one can imagine 3D information spaces with agents presenting different point of views and where users are given different roles to play. In a museum you might want to highlight key points through innovative and fun interactive exhibits.* Although the subject of *persuasive technology* is far less technology-oriented than the name suggests, multimedia (in a broad sense) form an excellent platform to explore *persuasion.* As concerns multimedia authoring, set yourself a goal, do the assignment, explore your capabilities, convey that message, and make the best of it.

(re)mediation

What can you hope to achieve when working with the new media? Think about it. Are the new media really new? Does anyone want to produce something that nobody has ever seen or heard before? Probably not. But it takes some philosophy to get that sufficiently clear.

In Bolter and Grusin (2000), the new media are analyzed from the perspective of remediation, that is the mutual influence of media on eachother in a historical perspective. In any medium, according to Bolter and Grusin (2000), there are two forces at work:

(re)mediation

- *immediacy* – a tendency towards transparent immersion, and
- *hypermediacy* – the presence of referential context

Put in other words, immediacy occurs when the medium itself is forgotten, so to speak, as is (ideally) the case in realistic painting, dramatic movies, and (perhaps in its most extreme form) in virtual reality. Hypermediacy may be observed when either the medium itself becomes the subject of our attention as in some genres of modern painting, experimental literature and film making, or when there is an explicit reference to other related sources of information or areas of experience, as in conceptual art, many web sites, and also in CNN news, where apart from live reports of ongoing action, running banners with a variety of information keep the viewers up to date of other news facts.

Now, the notion of *remediation* comes into play when we observe that every medium draws on the history of other media, or even its own history, to achieve a proper level of immediacy, or 'natural immersion'. For example, Hollywoord movies are only realistic to the extent that we understand the dramatic intent of cuts, close-ups and storylines, as they have been developed by the industry during the development of the medium. As another example, the realism of virtual reality can only be understood when we appreciate linear perspective (which arose out of realistic Renaissance painting) and dynamic scenes from a first person perspective (for which we have been prepared by action movies and TV).

Even if you may argue about the examples, let it be clear that each (new) medium refers, at least implicitly, to another medium, or to itself in a previous historic phase. So, what does this mean for new media, like TV or virtual reality?

Let's start with virtual reality. Bolter and Grusin (2000) comment on a statement of Arthur C. Clarke

> *Virtual Reality won't merely replace TV. It will eat it alive.*

by saying that ... *he is right in the sense that virtual reality remediates television (and film) by the strategy of incorporation. This strategy does not mean that virtual reality can obliterate the earlier visual point-of-view technologies, rather it ensures that these technologies remain as least as reference points by which the immediacy of virtual reality is measured.*

So, they observe "paradoxically, then, remediation is as important for the logic of transparency as it is for hypermediacy". Following Bolter and Grusin (2000), we can characterize the notions of immediacy and hypermediacy somewhat more precisely.

immediacy

- epistemological: transparency, the absence of mediation
- psychological: the medium has disappeared, presence, immersion

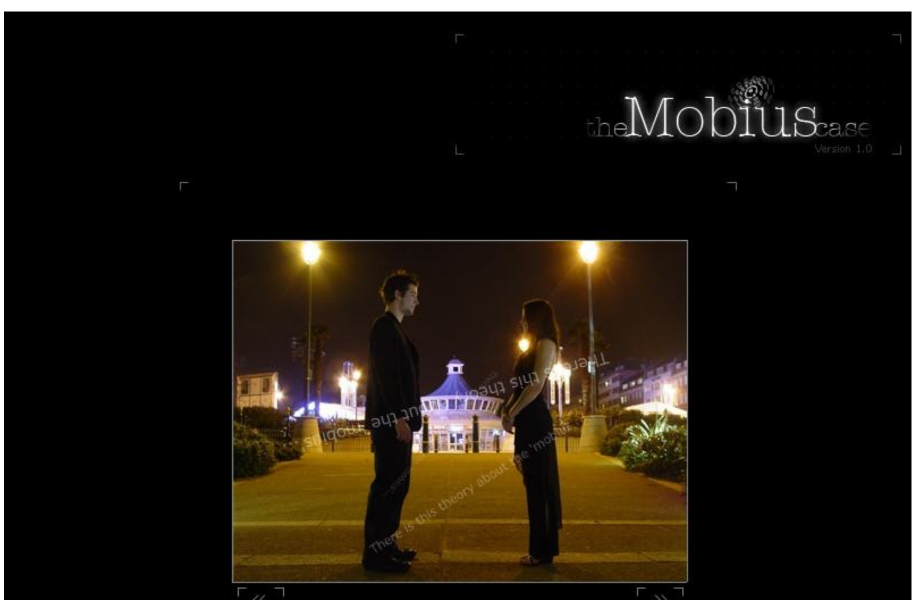
hypermediacy

- epistemological: opacity, presence of the medium and mediation
- psychological: experience of the medium is an experience of the real

Now, sharpen your philosophical teeth at the following statement, taken from Bolter and Grusin (2000), p. 224:

> *Convergence is the mutual remediation of at least three important technologies – telephone, televison and computer – each of which is a hybrid of technical, social and economic practice, and each of which offers its own path to immediacy.*
>
> *The telephone offers the immediacy of voice or the interchange of voices in real-time.*
>
> *Television is a point-of-view technology that promises immediacy through its insistent real-time monitoring of the world.*
>
> *The computer's promise of immediacy comes through the combination of three-dimensional graphics, automatic (programmed) action, and an interactivity that television can not match.*
>
> *As they come together, each of these is trying to absorb the others and promote its own version of immediacy.*

Let us once more come back to virtual reality and its possible relevance in our information age, Bolter and Grusin (2000), p. 225:: *in the claim that new media should not be merely archival but immersive, the rhetoric of virtual reality finally enters in, with its promise of the immediacy of experience through transparency.* . So, with respect to the new media, we may indeed conclude: *what is in fact new is the particular way in which each innovation rearranges and reconstitutes the meaning of earlier elements. What is new about media is therefore also old and familiar: that they promise the new by remediating what has gone before. The true novelty would be a new medium that did not refer to the other media at all. For our culture, such mediation without remediation seems to be impossible.*



8

example(s) – *mobius*

Rurger van Dijk, a former student of mine, has implemented an interactive story in *flash*. The story is a romance, told with images displaying scenes from the life of the players, a young man and a young women. The user can choose perspectives, either the man's or woman's, and watch the story from that point of view. The story is both non-linear and circular. The scenes can be connected in various way, and order is not compulsory.

research directions– *narrative structure*

Where do we go from here? What is the multimedia computer, if not a new medium? To close this section on multimedia authoring, let us reconsider in what way the networked multimedia computer differs from other media, by taking up the theme of convergence again. The networked multimedia computer seems to remediate all other media. Or, in the words of Murray (1997):

convergence

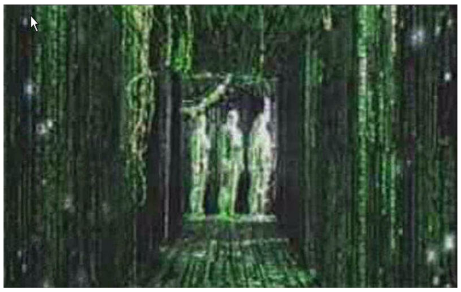
> *... merging previously disparate technologies of communication and representation into a single medium.*
>
> *The networked computer acts like a telephone in offering one-to-one real-time communication, like a television in broadcasting moving pictures, like an auditorium in bringing groups together for lectures and discussion, like a library in offering vast amounts of textual information for reference, like a museum in its ordered presentation of visual information, like a billboard, a radio, a gameboard and even like a manuscript in its revival of scrolling text.*

In Murray (1997), an analysis is given of a great variety of computer entertainment applications, varying from shoot-em-up games to collaborative interactive role playing. Murray (1997) identifies four essential properties that make these applications stand out against the entertainment offered by other media, which include books and TV. Two key properties determine the interactive nature of computer entertainment applications:

interactive

- *procedural* – 'programmed media' ...
- *participatory* – offering agency

All applications examined in Murray (1997) may be regarded as 'programmed media', for which interactivity is determined by 'procedural rules'. With *agency* is meant that the user can make active choices and thus influence the course of affairs, or at least determine the sequence in which the material is experienced.



9

Another common characteristic of the applications examined is what Murray (1997) calls *immersiveness.* Immersiveness is determined by two other key properties:

immersive

- *spatial* – explorable in (state) space
- *encyclopedic* – with (partial) information closure

All applications are based on some spatial metaphor. Actually, many games operate in 'levels' that can be accessedonly after demonstrating a certain degree of mastery. Networked computer applications allow for incorporating an almost unlimited amount of information. Some of the information might be open-ended, with storylines that remain unfinished. Closure, then, is achieved simply by exhaustive exploration or diminishing attention.

multimedia authoring Coming back to the question what the 'new medium', that is the networked multimedia computer, has to offer from the perspective of multimedia authoring, two aspects come to the foreground:

multimedia authoring

- narrative format
- procedural authorship

The narrative format is incredibly rich, offering all possibilities of the multimedia computer, including 3D graphics, real-time sound, text. In short, everything up to virtual reality. But perhaps the most distinguishing feature of the new medium is that true authorship requires both artistic capabilities as well as an awareness of the computational power of the medium. That is to say, authorship also means to formulate generic computational rules for telling a story while allowing for interactive interventions by the user. Or, as phrased in Murray (1997), the new *cyberbard* must create prototypical stories and formulaic characters that, in some way, lead their own life and tell their stories following their innate (read: programmed) rules. In section **??** and appendix , we will present a framework that may be used as a testbed for developing programmed narrative structures with embodied agents as the main characters.

2.4 development(s) – semantic mashups

mashup(s)

- substituting a single pragmatism for ideal design
- light weight programming models

web 2.0 design pattern(s)

- web 1.0 – the web as platform
- web 2.0 – architecture of participation
- web 3.0 – data is the (intel) inside

blogosphere / perma link / track back / page ran

10

questions

information spaces

1. (*) What factors play a role in the development of *multimedia information systems*? What research issues are there? When do you expect the major problems to be solved?

concepts

2. Define the notion of *information spaces*?
3. Indicate how multimedia objects may be placed (and queried for) in an *information (hyper) space*?
4. Characterize the notion of *hypermedia*.

technology

5. Discuss which developments make a large scale application of multimedia information systems possible.
6. Give a characterization of an object, a query and a clue in an *information space*.
7. Describe the *Dexter Hypertext Reference Model*.
8. Give a description of the *Amsterdam Hypermedia Model*.

projects & further reading As a project, I suggest the development of a virtual tour in a city, museum or other interesting locatoion.

You may further explore the implementation of traversal within a context, taking into account the history of navigation when backtracking to a particular point, issues in hyperlinking and interaction in multimedia applications, and computational support for narratives.

For further reading I advice you to take a look at the history of hypermedia and the web, using online material from the W3C[16], or the history of media as accounted for in Briggs and Burke (2001) and Bolter and Grusin (2000).

the artwork

1. book covers – Desing, Eco (1994), Avantgarde, Kunst, Betsky (2004)
2. Federico Campanale[17] – Oxygen, fragments from video installation, 2004
3. Vasarely – Diehl 1973.
4. Vasarely – Diehl 1973.
5. Vasarely – Diehl 1973.
6. Federico Campanale – Oxygen, more fragments.
7. student work – from *introduction multimedia* 2000.
8. Rutger van Dijk – *mobius*, interactive story, opening screen, see section 2.3.
9. edgecodes – screenshots, see section 2.3
10. signs – people, van Rooijen (2003), p. 244, 245.

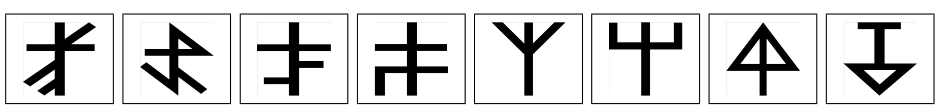
The work of Vasarely has served as an example for many contemporary digital artists. It is playful, mat may be characterized also as *formalist*. The highly aesthetic video work of Federico Campanale who, as he told me was strongly influenced by vasarely in his early years, shows a similar combination of formalism and playfulness. The interactive story by Rutger van Dijk has a rather different atmosphere, it is highly romantic, with slick graphics. The musea sites are included to point to the existence of (an increasing number) of virtual tours.

[16] www.w3c.org
[17] www.blue-frame.com

part ii. delivery & presentation

if you linger for a long time in one place you'd almost think there must be something there
wittgenstein

3. codecs and standards
4. multimedia platforms



2

reading directives In this part we will look at the issues involved in delivery and presentation, primarily from a technical perspective. We will argue the importance of codecs (read compression), and we will discuss the criteria for selecting a particular codec, as well as the standards that have been developed for packaging multimedia content in an effective way. In chapter 4, we will discuss multimedia presentation platforms, and we will look at the Microsoft DirectX 9 platform in somewhat greater detail.

Essential sections are section 3.1, which introduces codecs, 3.2, which discusses the MPEG-4 and SMIL standards and section 4.1, with puts the development of high-end multimedia platforms in a historical perspective. Sections 3.3 and 4.3 can safely be skipped on first reading.

perspectives As you can see below, the topics introduced in this part are not only relevant from a technical perspective. Other perspectives are equally valid:

perspectives – delivery & presentation

- technical – codec selection
- political – market vs. consortium
- sociological – digital services
- legal – copyright protection
- scientific – experience design
- computer science – computational support
- futuristic – global & personal information
- commercial – WMV, Quicktime, RealONE

For example, the issues of copyrights and copyright protection are hot topics, since the rise of the internet is obviously a threat to the tradional industries of music and film distribution.

essay topics Since many of the interesting topics will only be hinted, you may select on or more topics for further investigation and study. As essay titles I would suggest:

- multimedia standards – MPEG4
- XML-based multimedia – SMIL
- multimedia technology – the DirectX 9 toolbox

When you write the essay, then assess first from which perspective you will tackle the subject. When you approach the material from a technical perspective, then make sure that you do understand the technical issues in sufficient detail.

3

the artwork

1. logo – a drawing by Soutine, it is (almost) my personal logo, and also decorates the cover of Eliens (2000).
2. signs – property marks, van Rooijen (2003), p. 76, 77.
3. photographs – Jaap Stahlie[18], commissioned work.

[18] www.jaapstahlie.com

3. codecs and standards

without compression delivery is virtually impossible

learning objectives

After reading this chapter you should be able to demonstrate the necessity of compression, to discuss criteria for the selection of codecs and mention some of the alternatives, to characterize the MPEG-4 and SMIL standards, to explain the difference between MPEG-4 and MPEG-2, and to speculate about the feasibility of a semantic multimedia web.

Without compression and decompression, digital information delivery would be virtually impossible. In this chapter we will take a more detailed look at compression and decompression. It contains the information that you may possibly need to decide on a suitable compression and decompression scheme (codec) for your future multimedia productions. We will also discuss the standards that may govern the future (multimedia) Web, including MPEG-4, SMIL and RM3D. We will explore to what extent these standards allow us to realize the optimal multimedia platform, that is one that embodies digital convergence in its full potential. Finally, we will investigate how these ideas may ultimately lead to a (multimedia) semantic web.

1

3.1 codecs

Back to the everyday reality of the technology that surrounds us. What can we expect to become of networked multimedia? Let one thing be clear *compression is the key to effective delivery*

There can be no misunderstanding about this, although you may wonder why you need to bother with compression (and decompression). The answer is simple. You need to be aware of the size of what you put on the web and the demands that imposes on the network. Consider the table, taken from Vasudev and Li (1997), below.

media	uncompressed	compressed
voice 8k samples/sec, 8 bits/sample	64 kbps	2-4 kbps
slow motion video 10fps 176x120 8 bits	5.07 Mbps	8-16 kbps
audio conference 8k samples/sec 8bits	64 kbps	16-64 kbps
video conference 15 fps 352x240 8bits	30.4 Mbps	64-768 kbps
audio (stereo) 44.1 k samples/s 16 bits	1.5 Mbps	128k-1.5Mbps
video 15 fps 352x240 15 fps 8 bits	30.4 Mbps	384 kbps
video (CDROM) 30 fps 352x240 8 bits	60.8 Mbps	1.5-4 Mbps
video (broadcast) 30 fps 720x480 8 bits	248.8 Mbps	3-8 Mbps
HDTV 59.9 fps 1280x720 8 bits	1.3 Gbps	20 Mbps

You'll see that, taking the various types of connection in mind

(phone: 56 Kb/s, ISDN: 64-128 Kb/s, cable: 0.5-1 Mb/s, DSL: 0.5-2 Mb/s)

you must be careful to select a media type that is suitable for your target audience. And then again, choosing the right compression scheme might make the difference between being able to deliver or not being able to do so. Fortunately,

images, video and audio are amenable to compression

Why this is so is explained in Vasudev and Li (1997). Compression is feasible because of, on the one hand, the statistical redundancy in the signal, and the irrelevance of particular information from a perceptual perspective on the other hand. Redundancy comes about by both spatial correlation, between neighboring pixels, and temporal correlation, between successive frames.

The actual process of encoding and decoding may be depicted as follows:

codec = (en)coder + decoder

signal $\rightarrow$ source coder $\rightarrow$ channel coder (encoding)

signal $\leftarrow$ source decoder $\leftarrow$ channel decoder (decoding)

Of course, the coded signal must be transmitted accross some channel, but this is outside the scope of the coding and decoding issue. With this diagram in mind we can specify the *codec design problem*:

> *From a systems design viewpoint, one can restate the codec design problem as a bit rate minimization problem, meeting (among others) constraints concerning: specified levels of signal quality, implementation complexity, and communication delay (start coding – end decoding).*

2

compression methods

As explained in Vasudev and Li (1997), there is a large variety of compression (and corresponding decompression) methods, including model-based methods, as for example the object-based MPEG-4 method that will be discussed later, and waveform-based methods, for which we generally make a distinction between lossless and lossy methods. Hufmann coding is an example of a lossless method, and methods based on Fourier transforms are generally lossy. Lossy means that actual data is lost, so that after decompression there may be a loss of (perceptual) quality.

Leaving a more detailed description of compression methods to the diligent students' own research, it should come as no surprise that when selecting a compression method, there are a number of tradeoffs, with respect to, for example, coding efficiency, the complexity of the coder and decoder, and the signal quality. In summary, the follwoing issues should be considered:

tradeoffs

- *resilience to transmission errors*
- *degradations in decoder output – lossless or lossy*
- *data representation – browsing & inspection*
- *data modalities – audio & video.*
- *transcoding to other formats – interoperability*
- *coding efficiency – compression ratio*
- *coder complexity – processor and memory requirements*
- *signal quality – bit error probability, signal/noise ratio*

For example, when we select a particular coder-decoder scheme we must consider whether we can guarantee resilience to transmission errors and how these will affect the users' experience. And to what extent we are willing to accept degradations in decoder output, that is lossy output. Another issue in selecting a method of compression is whether the (compressed) data representation allows for browsing & inspection. And, for particular applications, such as conferencing, we should be worried about the interplay of data modalities,in particular, audio & video. With regard to the many existing codecs and the variety of platforms we may desire the possibility of transcoding to other formats to achieve, for example, exchange of media objects between tools, as is already common for image processing tools.

compression standards

Given the importance of codecs it should come as no surprise that much effort has been put in developing standards, such as JPEG for images and MPEG for audio and video.

Most of you have heard of MP3 (the audio format), and at least some of you should be familiar with MPEG-2 video encoding (which is used for DVDs).

Now, from a somewhat more abstract perspective, we can, again following Vasudev and Li (1997), make a distinction between a *pixel-based approach* (coding the raw signal so to speak) and an *object-based approach*, that uses segmentation and a more advanced scheme of description.

- *pixel-based* – MPEG-1, MPEG-2, H3.20, H3.24
- *object-based* – MPEG-4

As will be explained in more detail when discussing the MPEG-4 standard in section 3.2, there are a number of advantages with an object-based approach. There is, however, also a price to pay. Usually (object) segmentation does not come for free, but requires additional effort in the phase of authoring and coding.

MPEG-1 To conclude this section on codecs, let's look in somewhat more detail at what is involved in coding and decoding a video signal according to the MPEG-1 standard.

MPEG-1 video compression uses both *intra-frame analysis*, for the compression of individual frames (which are like images), as well as. *inter-frame analysis*, to detect redundant blocks or invariants between frames.

The MPEG-1 encoded signal itself is a sequence of so-called I, P and B frames.

frames

- I: intra-frames – independent images
- P: computed from closest frame using DCT (or from P frame)
- B: computed from two closest P or I frames

Decoding takes place by first selecting I-frames, then P-frames, and finally B-frames. When an error occurs, a safeguard is provided by the I-frames, which stand on themselves.

Subsequent standards were developed to accomodate for more complex signals and greater functionality. These include MPEG-2, for higher pixel resolution and data rate, MPEG-3, to support HDTV, MPEG-4, to allow for object-based compression, and MPEG-7, which supports content description. We will elaborate on MPEG-4 in the next section, and briefly discuss MPEG-7 at the end of this chapter.

example(s) – *gigaport*

GigaPort[19] is a project focussing on the development and use of advanced and innovative Internet technology. The project, as can be read on the website,

[19] www.gigaport.nl/info/en/about/home.jsp

focuses on research on next-generation networks and the implementation of a next-generation network for the research community.

Topics for research include:

GigaPort

- optical network technologies - models for network architecture, optical network components and light path provisioning.
- high performance routing and switching - new routing technologies and transport protocols, with a focus on scalability and stability robustness when using data-intensive applications with a high bandwidth demand.
- management and monitoring - incident response in hybrid networks (IP and optical combined) and technologies for network performance monitoring, measuring and reporting.
- grids and access - models, interfaces and protocols for user access to network and grid facilities.
- test methodology - effective testing methods and designing tests for new technologies and network components.

As one of the contributions, internationally, the development of optical technology is claimed, in particular *lambda* networking, networking on a specific wavelength. Locally, the projects has contributed to the introduction of fibre-optic networks in some major cities in the Netherlands.

research directions– *digital video formats*

In the online version you will find a brief overview of *digital video technology*, written by Andy Tanenbaum, as well as some examples of videos of our university, encoded at various bitrates for different viewers.

What is the situation? For traditional television, there are three standards. The american (US) standard, NTSC, is adopted in North-America, South-America and Japan. The european standard, PAL, whuch seems to be technically superior, is adopted by the rest of the world, except France and the eastern-european countries, which have adopted the other european standard, SECAM. An overview of the technical properties of these standards, with permission taken from Tanenbaum's account, is given below.

system	**spatial resolution**	**frame rate**	**mbps**
NTSC	704 x 480	30	243 mbps
PAL/SECAM	720 x 576	25	249 mbps

Obviously real-time distribution of a more than 200 mbps signal is not possible, using the nowadays available internet connections. Even with compression on the fly, the signal would require 25 mbps, or 36 mbps with audio. Storing the signal on disk is hardly an alternative, considering that one hour would require 12 gigabytes.

When looking at the differences between streaming video (that is transmitted real-time) and storing video on disk, we may observe the following tradeoffs:

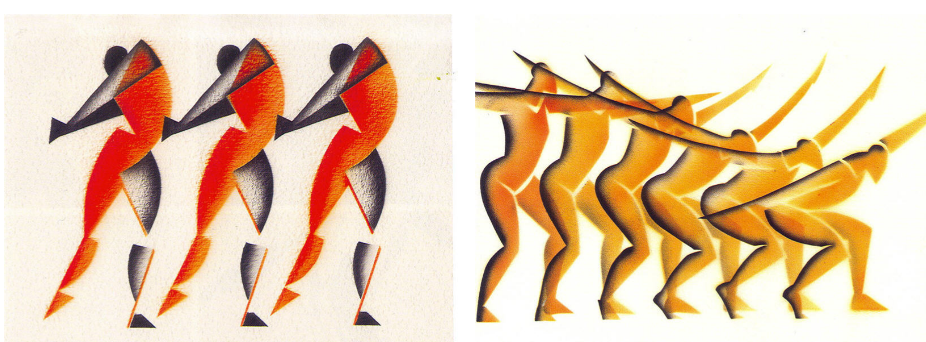
item	**streaming**	**downloaded**
bandwidth	equal to the display rate	may be arbitrarily small
disk storage	none	the entire file must be stored
startup delay	almost none	equal to the download time
resolution	depends on available bandwidth	depends on available disk storage

So, what are our options? Apart from the quite successful MPEG encodings, which have found their way in the DVD, there are a number of proprietary formats used for transmitting video over the internet: Quicktime, introduced by Apple, early 1990s, for local viewing; RealVideo, streaming video from RealNetworks; and Windows Media, a proprietary encoding scheme fromMicrosoft. Examples of these formats, encoded for various bitrates are available at Video at VU.

Apparently, there is some need for digital video on the internet, for example as propaganda for attracting students, for looking at news items at a time that suits you, and (now that digital video cameras become affordable) for sharing details of your family life.

Is digital video all there is? Certainly not! In the next section, we will deal with standards that allow for incorporating (streaming) digital video as an element in a compound multimedia presentation, possibly synchronized with other items, including synthetic graphics. Online, you will find some examples of digital video that are used as texture maps in 3D space. These examples are based on the technology presented in section **??**, and use the streaming video codec from Real Networks that is integrated as a rich media extension in the *blaxxun* Contact 3D VRML plugin.

comparison of codecs A review of codecs[20], including Envivio MPEG-4, QuickTime 6, RealNetworks 9 en Windows Media 9 was published januari 2005 by the European Broadcast Union[21]. It appeared that The Real Networks codecs came out best, closely followed by the Windows Media 9 result. Ckeck it out!



3

[20] www.ebu.ch/trev_301-samviq.pdf

[21] www.ebu.ch/trev_home.html

3.2 standards

Imagine what it would be like to live in a world without standards. You may get the experience when you travel around and find that there is a totally different socket for electricity in every place that you visit.

Now before we continue, you must realize that there are two types of standards: *de facto* market standards (enforced by sales politics) and committee standards (that are approved by some official organization). For the latter type of standards to become effective, they need consent of the majority of market players.

For multimedia on the web, we will discuss three standards and RM3D which was once proposed as a standard and is now only of historical significance.

standards

- XML – eXtensible Markup Language (SGML)
- MPEG-4 – coding audio-visual information
- SMIL – Synchronized Multimedia Integration Language
- RM3D – (Web3D) Rich Media 3D (extensions of X3D/VRML)

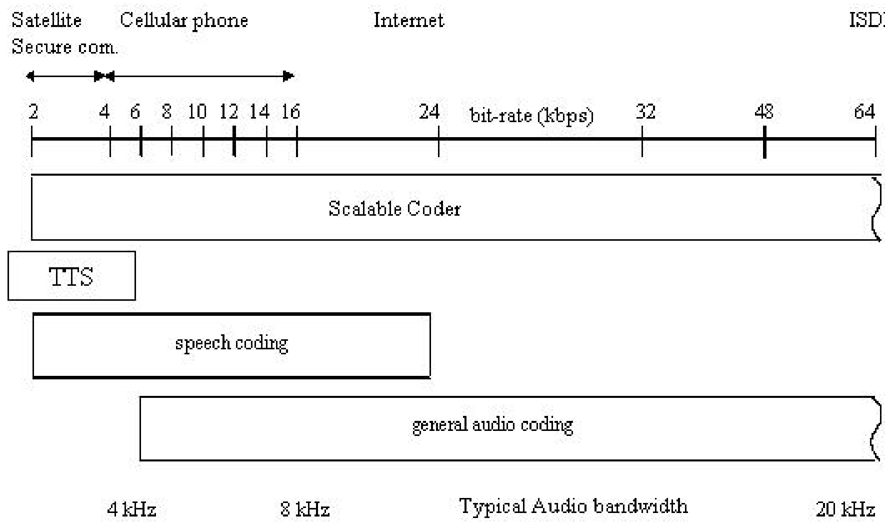
XML, the *eXtensible Markup Language*, is becoming widely accepted. It is being used to replace HTML, as well as a data exchange format for, for example, business-to-business transactions. XML is derived from SGML (Structured Generalized Markup Language) that has found many applications in document processing. As SGML, XML is a generic language, in that it allows for the specification of actual markup languages. Each of the other three standards mentioned allows for a syntactic encoding using XML.

MPEG-4 aims at providing "the standardized technological elements enabling the integration of production, distribution and content access paradigms of digital television, interactive graphics and multimedia", Koenen (2000). A preliminary version of the standard has been approved in 1999. Extensions in specific domains are still in progress.

SMIL, the *Synchronized Multimedia Integration Language*, has been proposed by the W3C "to enable the authoring of TV-like multimedia presentations, on the Web". The SMIL language is an easy to learn HTML-like language. SMIL presentations can be composed of streaming audio, streaming video, images, text or any other media type, W3C (2001). SMIL-1 has become a W3C recommendation in 1998. SMIL-2 is at the moment of writing still in a draft stage.

RM3D, *Rich Media 3D*, is not a standard as MPEG-4 and SMIL, since it does currently not have any formal status. The RM3D working group arose out of the X3D working group, that addressed the encoding of VRML97 in XML. Since there were many disagreements on what should be the core of X3D and how extensions accomodating VRML97 and more should be dealt with, the RM3D working group was founded in 2000 to address the topics of extensibility and the integration with rich media, in particular video and digital television.

remarks Now, from this description it may seem as if these groups work in total isolation from eachother. Fortunately, that is not true. MPEG-4, which is the most encompassing of these standards, allows for an encoding both in SMIL and X3D. The X3D and RM3D working groups, moreover, have advised the MPEG-4 commitee on how to integrate 3D scene description and human avatar animation in MPEG-4. And finally, there have been rather intense discussions between the SMIL and RM3D working groups on the timing model needed to control animation and dynamic properties of media objects.



4

MPEG-4

The MPEG standards (in particular 1,2 and 3) have been a great success, as testified by the popularity of mp3 and DVD video.

Now, what can we expect from MPEG-4? Will MPEG-4 provide *multimedia for our time*, as claimed in Koenen (1999). The author, Rob Koenen, is senior consultant at the dutch KPN telecom research lab, active member of the MPEG-4 working group and editor of the MPEG-4 standard document.

> *"Perhaps the most immediate need for MPEG-4 is defensive. It supplies tools with which to create uniform (and top-quality) audio and video encoders on the Internet, preempting what may become an unmanageable tangle of proprietary formats."*

Indeed, if we are looking for a general characterization it would be that MPEG-4 is primarily

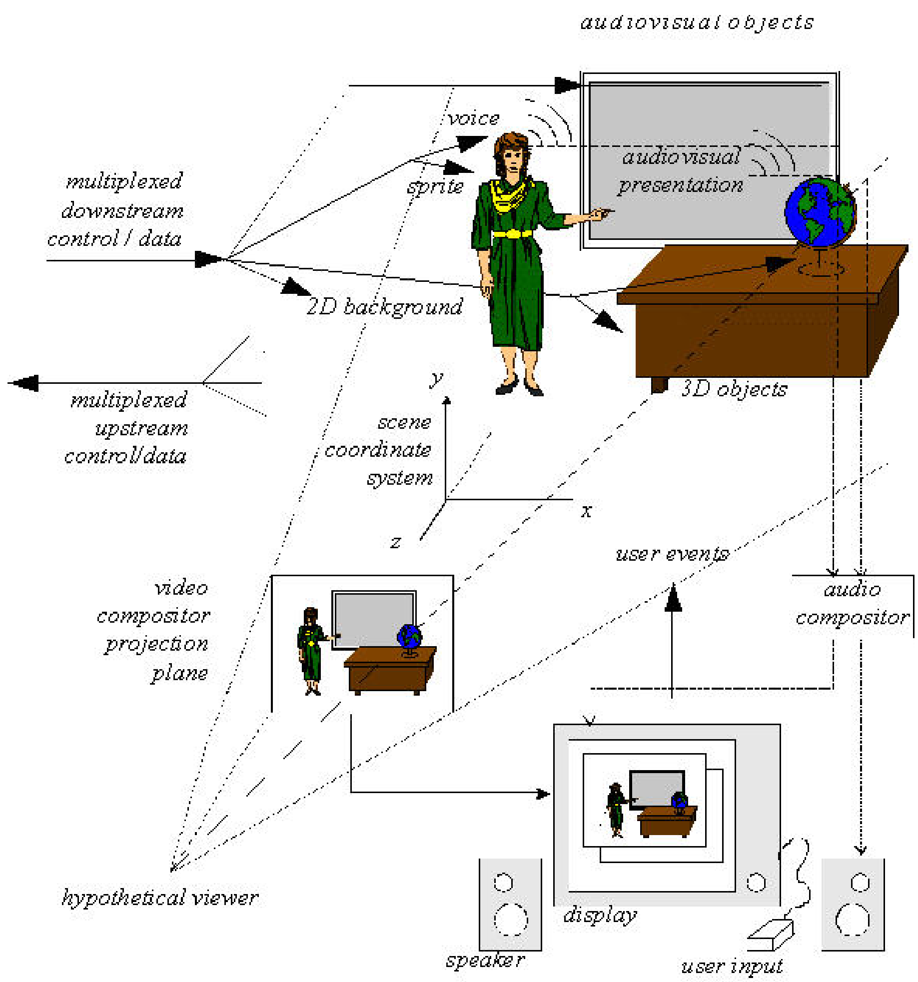
MPEG-4

a toolbox of advanced compression algorithms for audiovisual information and, moreover, one that is suitable for a variety of display devices and networks, including low bitrate mobile networks. MPEG-4 supports scalability on a variety of levels:

scalability

- *bitrate* – switching to lower bitrates
- *bandwidth* – dynamically discard data
- *encoder and decoder complexity* – signal quality

Dependent on network resources and platform capabilities, the 'right' level of signal quality can be determined by selecting the optimal codec, dynamically.



media objects It is fair to say that MPEG-4 is a rather ambitious standard. It aims at offering support for a great variety of audiovisual information, including still images, video, audio, text, (synthetic) talking heads and synthesized speech, synthetic graphics and 3D scenes, streamed data applied to media objects, and user interaction – e.g. changes of viewpoint.

Let's give an example, taken from the MPEG-4 standard document.

example

> *Imagine, a talking figure standing next to a desk and a projection screen, explaining the contents of a video that is being projected on the screen, pointing at a globe that stands on the desk. The user that is watching that scene decides to change from viewpoint to get a better look at the globe ...*

How would you describe such a scene? How would you encode it? And how would you approach decoding and user interaction?

The solution lies in defining *media objects* and a suitable notion of composition of media objects.

media objects

- *media objects* – units of aural, visual or audiovisual content
- *composition* – to create compound media objects (audiovisual scene)
- *transport* – multiplex and synchronize data associated with media objects
- *interaction* – feedback from users' interaction with audiovisual scene

For 3D-scene description, MPEG-4 builds on concepts taken from VRML (Virtual Reality Modeling Language, discussed in chapter 7).

Composition, basically, amounts to building a *scene graph*, that is a tree-like structure that specifies the relationship between the various simple and compound media objects. Composition allows for placing media objects anywhere in a given coordinate system, applying transforms to change the appearance of a media object, applying streamed data to media objects, and modifying the users viewpoint.

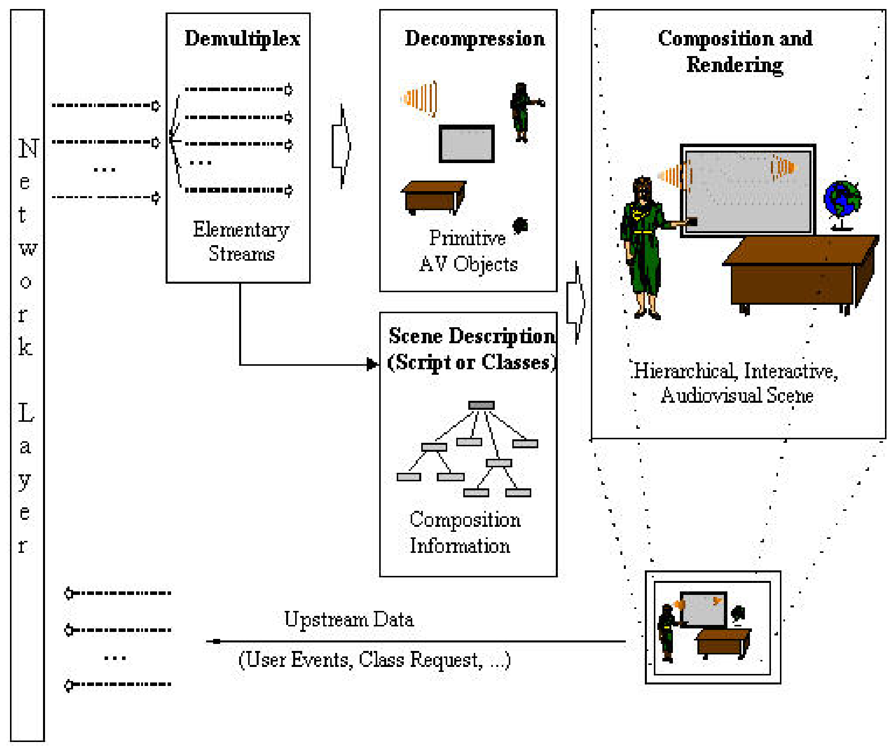
So, when we have a multimedia presentation or audiovisual scene, we need to get it accross some network and deliver it to the end-user, or as phrased in Koenen (2000):

transport

> *The data stream (Elementary Streams) that result from the coding process can be transmitted or stored separately and need to be composed so as to create the actual multimedia presentation at the receivers side.*

At a system level, MPEG-4 offers the following functionalities to achieve this:

- BIFS (Binary Format for Scenes) – describes spatio-temporal arrangements of (media) objects in the scene
- OD (Object Descriptor) – defines the relationship between the elementary streams associated with an object
- *event routing* – to handle user interaction

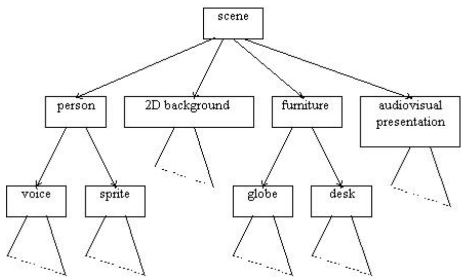


6

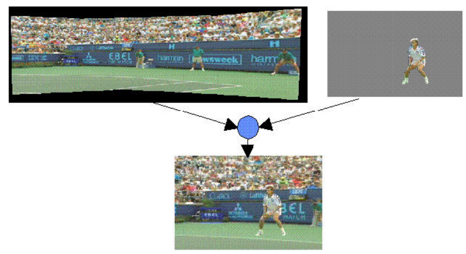
In addition, MPEG-4 defines a set of functionalities For the delivery of streamed data, DMIF, which stands for

Delivery Multimedia Integration Framework

that allows for transparent interaction with resources, irrespective of whether these are available from local storage, come from broadcast, or must be obtained from some remote site. Also transparency with respect to network type is supported. *Quality of Service* is only supoorted to the extent that it ispossible to indicate needs for bandwidth and transmission rate. It is however the responsability of the network provider to realize any of this.



(a) scene graph



(b) sprites

7

authoring What MPEG-4 offers may be summarized as follows

benefits

- *end-users* – interactive media accross all platforms and networks
- *providers* – transparent information for transport optimization
- *authors* – reusable content, protection and flexibility

In effect, although MPEG-4 is primarily concerned with efficient encoding and scalable transport and delivery, the *object-based* approach has also clear advantages from an authoring perspective.

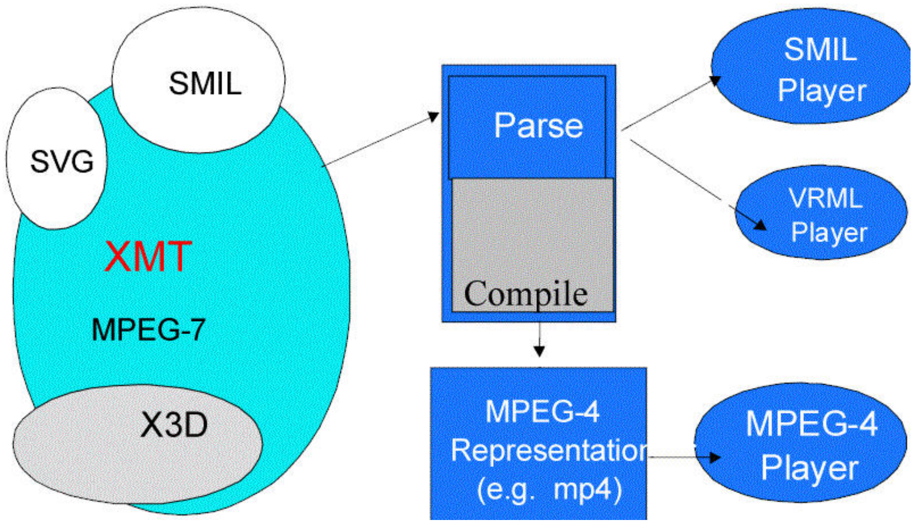
One advantage is the possibility of reuse. For example, one and the same background can be reused for multiplepresentations or plays, so you could imagine that even an amateur game might be 'located' at the centre-court of Roland Garros or Wimbledon.

Another, perhaps not so obvious, advantage is that provisions have been made for

managing intellectual property

of media objects.

And finally, media objects may potentially be annotated with meta-information to facilitate information retrieval.



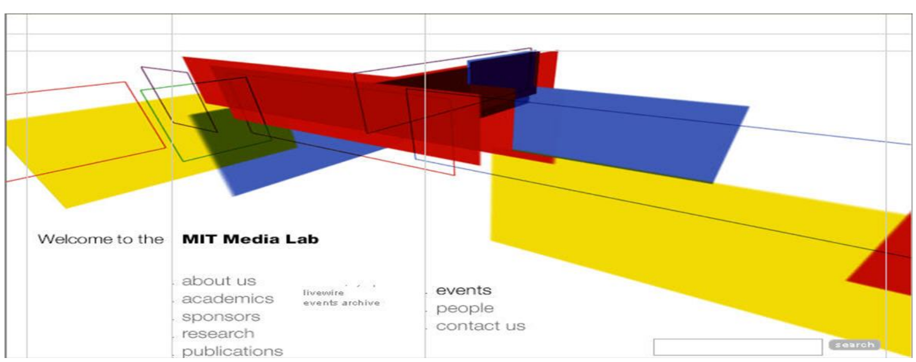
8

syntax In addition to the binary formats, MPEG-4 also specifies a syntactical format, called XMT, which stands for *eXtensible MPEG-4 Textual format*.

XMT

- XMT contains a subset of X3D
- SMIL is mapped (incompletely) to XMT

when discussing RM3D which is of interest from a historic perspective, we will further establish what the relations between, respectively MPEG-4, SMIL and RM3D are, and in particular where there is disagreement, for example with respect to the timing model underlying animations and the temporal control of media objects.



9

example(s) – *structured audio*

The Machine Listening Group[22] of the MIT Media Lab[23] is developing a suite of tools for *structered audio*, which means *transmitting sound by describing it rather than compressing it.* It is claimed that tools based on the MPEG-4 standard will be the future platform for computer music, audio for gaming, streaming Internet radio, and other multimedia applications.

The structured audio project is part of a more encompassing research effort of the Music, Mind and Machine Group[24] of the MIT Media Lab, which *envisages a new future of audio technologies and interactive applications that will change the way music is conceived, created, transmitted and experienced*,

SMIL

SMIL is pronounced as *smile*. SMIL, the Synchronized Multimedia Integration Language, has been inspired by the Amsterdam Hypermedia Model (AHM). In fact, the dutch research group at CWI that developed the AHM actively participated in the SMIL 1.0 committee. Moreover, they have started a commercial spinoff to create an editor for SMIL, based on the editor they developed for CMIF. The name of the editor is GRINS. Get it?

As indicated before SMIL is intended to be used for

TV-like multimedia presentations

The SMIL language is an XML application, resembling HTML. SMIL presentations can be written using a simple text-editor or any of the more advanced tools, such as GRINS. There is a variety of SMIL players. The most wellknown perhaps is the RealNetworks G8 players, that allows for incorporating RealAudio and RealVideo in SMIL presentations.

parallel and sequential

> *Authoring a SMIL presentation comes down, basically, to name media components for text, images,audio and video with URLs, and to schedule their presentation either in parallel or in sequence.*

Quoting the SMIL 2.0 working draft, we can characterize the SMIL presentation characteristics as follows:

presentation characteristics

- The presentation is composed from several components that are accessible via URL's, e.g. files stored on a Web server.
- The components have different media types, such as audio, video, image or text. The begin and end times of different components are specified relative to events in other media components. For example, in a slide show, a particular slide is displayed when the narrator in the audio starts talking about it.

[22] sound.media.mit.edu/mpeg4
[23] www.media.mit.edu
[24] sound.media.mit.edu

- Familiar looking control buttons such as stop, fast-forward and rewind allow the user to interrupt the presentation and to move forwards or backwards to another point in the presentation.
- Additional functions are "random access", i.e. the presentation can be started anywhere, and "slow motion", i.e. the presentation is played slower than at its original speed.
- The user can follow hyperlinks embedded in the presentation.

Where HTML has become successful as a means to write simple hypertext content, the SMIL language is meant to become a vehicle of choice for writing *synchronized hypermedia*. The working draft mentions a number of possible applications, for example a photoalbun with spoken comments, multimedia training courses, product demos with explanatory text, timed slide presentations, onlime music with controls.

As an example, let's consider an interactive news bulletin, where you have a choice between viewing a weather report or listening to some story about, for example, the decline of another technology stock. Here is how that could be written in SMIL:

example

```
<par>
   <a href=" #Story"> <img src="button1.jpg"/> </a>
   <a href=" #Weather"> <img src="button2.jpg"/></a>
    <excl>
        <par id="Story" begin="0s">
          <video src="video1.mpg"/>
          <text src="captions.html"/>
        </par>

        <par id="Weather">
          <img src="weather.jpg"/>
          <audio src="weather-rpt.mp3"/>
        </par>
    </excl>
</par>
```

Notice that there are two *parallel* (PAR) tags, and one *exclusive* (EXCL) tag. The *exclusive* tag has been introduced in SMIL 2.0 to allow for making an exclusive choice,so that only one of the items can be selected at a particular time. The SMIL 2.0 working draft defines a number of elements and attributes to control presentation, synchronization and interactivity, extending the functionality of SMIL 1.0.

Before discussing how the functionality proposed in the SMIL 2.0working draft may be realized, we might reflect on how to position SMIL with respect to the many other approaches to provide multimedia on the web. As other approaches we may think of *flash*, dynamic HTML (using javascript), or java applets. In the SMIL 2.0 working draft we read the following comment:

history

> *Experience from both the CD-ROM community and from the Web multimedia community suggested that it would be beneficial to adopt a declarative format for expressing media synchronization on the Web as an alternative and complementary approach to scripting languages.*
>
> *Following a workshop in October 1996, W3C established a first working group on synchronized multimedia in March 1997. This group focused on the design of a declarative language and the work gave rise to SMIL 1.0 becoming a W3C Recommendation in June 1998.*

In summary, SMIL 2.0 proposes a *declarative format* to describe the temporal behavior of a multimedia presentation, associate hyperlinks with media objects, describe the form of the presentation on a screen, and specify interactivity in multimedia presentations. Now,why such a fuzz about "declarative format"? Isn't scripting more exciting? And aren't the tools more powerful? Ok, ok. I don't want to go into that right now. Let's just consider a *declarative format* to be more elegant. Ok?

To support the functionality proposed for SMIL 2.0 the working draft lists a number of modules that specify the interfaces for accessing the attributes of the various elements. SMIL 2.0 offers modules for animation, content control, layout, linking, media objects, meta information, timing and synchronization, and transition effects.

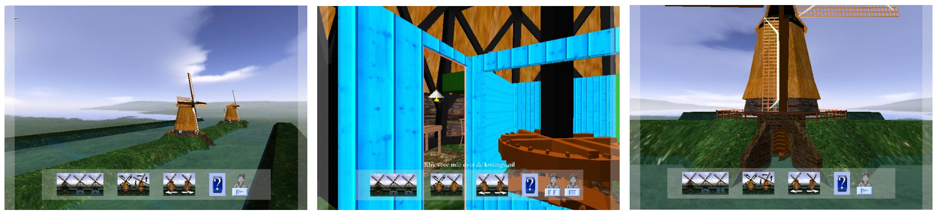
This modular approach allows to reuse SMIL syntax and semantics in other XML-based languages, in particular those that need to represent timing and synchronization. For example:

module-based reuse

- SMIL modules could be used to provide lightweight multimedia functionality on mobile phones, and to integrate timing into profiles such as the WAP forum's WML language, or XHTML Basic.
- SMIL timing, content control, and media objects could be used to coordinate broadcast and Web content in an enhanced-TV application.
- SMIL Animation is being used to integrate animation into W3C's Scalable Vector Graphics language (SVG).
- Several SMIL modules are being considered as part of a textual representation for MPEG4.

The SMIL 2.0 working draft is at the moment of writing being finalized. It specifies a number of language profiles topromote the reuse of SMIL modules. It also improves on the accessibility features of SMIL 1.0, which allows for, for example,, replacing captions by audio descriptions.

In conclusion, SMIL 2.0 is an interesting standard, for a number of reasons. For one, SMIL 2.0 has solid theoretical underpinnings in a well-understood, partly formalized, hypermedia model (AHM). Secondly, it proposes interesting functionality, with which authors can make nice applications. In the third place, it specifies a high level declarative format, which is both expressive and flexible. And finally, it is an open standard (as opposed to proprietary standard). So everybody can join in and produce players for it!



10

RM3D – not a standard

The web started with simple HTML hypertext pages. After some time static images were allowed. Now, there is support for all kinds of user interaction, embedded multimedia and even synchronized hypermedia. But despite all the graphics and fancy animations, everything remains flat. Perhaps surprisingly, the need for a 3D web standard arose in the early days of the web. In 1994, the acronym VRML was coined by Tim Berners-Lee, to stand for *Virtual Reality Markup Language*. But, since 3D on the web is not about text but more about worlds, VRML came to stand for *Virtual Reality Modeling Language*. Since 1994, a lot of progress has been made.

www.web3d.org

- VRML 1.0 – *static 3D worlds*
- VRML 2.0 or VRML97 – *dynamic behaviors*
- VRML200x – *extensions*
- X3D – *XML syntax*
- RM3D – *Rich Media in 3D*

In 1997, VRML2 was accepted as a standard, offering rich means to create 3D worlds with dynamic behavior and user interaction. VRML97 (which is the same as VRML2) was, however, not the success it was expected to be, due to (among others) incompatibility between browsers, incomplete implementations of the standards, and high performance requirements.

As a consequence, the Web3D Consortium (formerly the VRML Consortium) broadened its focus, and started thinking about extensions or modifications of VRML97 and an XML version of VRML (X3D). Some among the X3D working group felt the need to rethink the premisses underlying VRML and started the Rich Media Working Group:

groups.yahoo.com/group/rm3d/

> *The Web3D Rich Media Working Group was formed to develop a Rich Media standard format (RM3D) for use in next-generation media devices. It is a highly active group with participants from a broad range of companies including 3Dlabs, ATI, Eyematic, OpenWorlds, Out of the Blue Design, Shout Interactive, Sony, Uma, and others.*

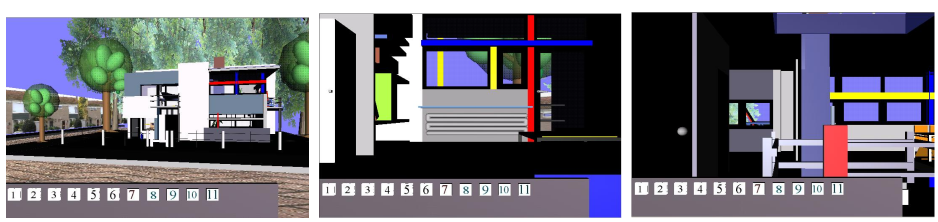
In particular:

RM3D

> *The Web3D Consortium initiative is fueled by a clear need for a standard high performance Rich Media format. Bringing together content creators with successful graphics hardware and software experts to define RM3D will ensure that the new standard addresses authoring and delivery of a new breed of interactive applications.*

The working group is active in a number of areas including, for example, multi-texturing and the integration of video and other streaming media in 3D worlds.

Among the driving forces in the RM3D group are Chris Marrin and Richter Rafey, both from Sony, that proposed *Blendo*, a rich media extension of VRML. Blendo has a strongly typed object model, which is much more strictly defined than the VRML object model, to support both declarative and programmatic extensions. It is interesting to note that the premisse underlying the Blendo proposal confirms (again) the primacy of the TV metaphor. That is to say, what Blendo intends to support are TV-like presentations which allow for user interaction such as the selection of items or playing a game. Target platforms for Blendo include graphic PCs, set-top boxes, and the Sony Playstation!



11

requirements The focus of the RM3D working group is not *syntax* (as it is primarily for the X3D working group) but *semantics*, that is to enhance the VRML97 standard to effectively incorporate rich media. Let's look in more detail at the requirements as specified in the RM3Ddraft proposal.

requirements

- *rich media* – audio, video, images, 2D & 3D graphics (with support for temporal behavior, streaming and synchronisation)
- *applicability* – specific application areas, as determined by commercial needs and experience of working group members

The RM3D group aims at interoperability with other standards.

- *interoperability* – VRML97, X3D, MPEG-4, XML (DOM access)

In particular, an XML syntax is being defined in parallel (including interfaces for the DOM). And, there is mutual interest and exchange of ideas between the MPEG-4 and RM3D working group.

As mentioned before, the RM3D working group has a strong focus on defining an object model (that acts as a common model for the representation of objects and their capabilities) and suitable mechanisms for extensibility (allowing for the integration of new objects defined in Java or C++, and associated scripting primitives and declarative constructs).

Notice that extensibility also requires the definition of a declarative format, so that the content author need not bother with programmatic issues.

The RM3D proposal should result in effective 3D media presentations. So as additional requirements we may, following the working draft, mention: high-quality realtime rendering, for realtime interactive media experiences; platform adaptability, with query functions for programmatic behavior selection; predictable behavior, that is a well-defined order of execution; a high precision number systems, greater than single-precision IEEE floating point numbers; and minimal size, that is both download size and memory footprint.

Now, one may be tempted to ask how the RM3D proposals is related to the other standard proposals such as MPEG-4 and SMIL, discussed previously. Briefly put, paraphrased from one of Chris Marrin's messages on the RM3D mailing list

> *SMIL is closer to the author* and *RM3D is closer to the implementer*.

MPEG-4, in this respect is even further away from the author since its chief focus is on compression and delivery across a network.

RM3D takes 3D scene description as a starting point and looks at pragmatic ways to integrate rich media. Since 3D is itself already computationally intensive, there are many issues thatarise in finding efficient implementations for the proposed solutions.

12

timing model RM3D provides a declarative format formany interesting features, such as for example texturing objects with video. In comparison to VRML, RM3D is meant to provide more temporal control over time-based media objects and animations. However, there is strong disagreement among the working group members as to what time model the dynamic capabilities of RM3D should be based on. As we read in the working draft:

working draft

> *Since there are three vastly different proposals for this section (time model), the original <RM3D> 97 text is kept. Once the issues concerning time-dependent nodes are resolved, this section can be modified appropriately.*

Now, what are the options? Each of the standards discussed to far provides us with a particular solution to timing. Summarizing, we have a time model based on a spring metaphor in MPEG-4, the notion of cascading time in SMIL (inspired by cascading stylesheets for HTML) and timing based on the routing of events in RM3D/VRML.

The MPEG-4 standard introduces the *spring metaphor* for dealing with temporal layout.

MPEG-4 – spring metaphor

- duration – minimal, maximal, optimal

The spring metaphor amounts to the ability to shrink or stretch a media object within given bounds (minimum, maximum) to cope with, for example, network delays.

The SMIL standard is based on a model that allows for propagating durations and time manipulations in a hierarchy of media elements. Therefore it may be referred to as a cascading modelof time.

SMIL – cascading time

- time container – speed, accelerate, decelerate, reverse, synchronize

Media objects, in SMIL, are stored in some sort of container of which the timing properties can be manipulated.

```
<seq speed="2.0">
   <video src="movie1.mpg" dur="10s"/>
   <video src="movie2.mpg" dur="10s"/>
   <img src="img1.jpg" begin="2s" dur="10s">
            <animateMotion from="-100,0" to="0,0" dur="10s"/>
   </img>
   <video src="movie4.mpg" dur="10s"/>
</seq>
```

In the example above,we see that the speed is set to *2.0*, which will affect the pacing of each of the individual media elements belonging to that (sequential) group. The duration of each of the elements is specified in relation to the parent container. In addition, SMIL offers the possibility to synchronize media objects to control, for example, the end time of parallel media objects.

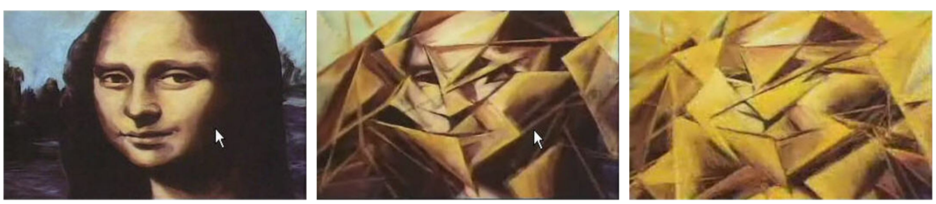
VRML97's capabilities for timing rely primarily on the existence of a *TimeSensor* thatsends out time events that may be routed to other objects.

RM3D/VRML – event routing

- *TimeSensor* – isActive, start, end, cycleTime, fraction, loop

When a *TimeSensor* starts to emit time events, it also sends out an event notifying other objects that it has become active. Dependent on itsso-called *cycleTime*, it sends out the fraction it covered since it started. This fraction may be send to one of the standard interpolators or a script so that some value can be set, such as for example the orientation, dependent on the fraction of the time intercal that has passed. When the *TimeSensor* is made to loop, this is done repeatedly. Although time in VRML is absolute, the frequency with which fraction events are emitted depends on the implementation and processor speed.

Lacking consensus about a better model, this model has provisionally been adopted, with some modifications, for RM3D. Nevertheless, the SMIL cascading time model has raised an interest in the RM3D working group, to the extent that Chris Marrin remarked (in the mailing list) *"we could go to school here"*. One possibility for RM3D would be to introduce *time containers* that allow for a temporal transform of their children nodes, in a similar way as grouping containers allow for spatial transforms of their children nodes. However, that would amount to a dual hierarchy, one to control (spatial) rendering and one to control temporal characteristics. Merging the two hierarchies, as is (implicitly) the case in SMIL, might not be such a good idea, since the rendering and timing semantics of the objects involved might be radically different. An interesting problem, indeed, but there seems to be no easy solution.



13

example(s) – *rich internet applications*

In a seminar held by *Lost Boys*, which is a dutch subdivison if Icon Media Lab[25], *rich internet applications* (RIA), were presented as the new solutions to present applications on the web. As indicated by Macromedia[26], who is one of the leading companies in this fiwld, *experience matters*, and so plain html pages pages do not suffice since they require the user to move from one page to another in a quite unintuitive fashion. Macromedia presents its new line of *flash*-based products to create such *rich internet applications*. An alternative solution, based on general W3C recommendations, is proposed by BackBase[27]. Interestingly enough, using either technology, many of the paricipants of the seminar indicated a strong preference for a backbuuton, having similar functionality as the often used backbutton in general internet browsers.

---

[25] www.iconmedialab.com

[26] www.macromedia.com/resources/business/rich_internet_apps/whitepapers.html

[27] www.backbase.com

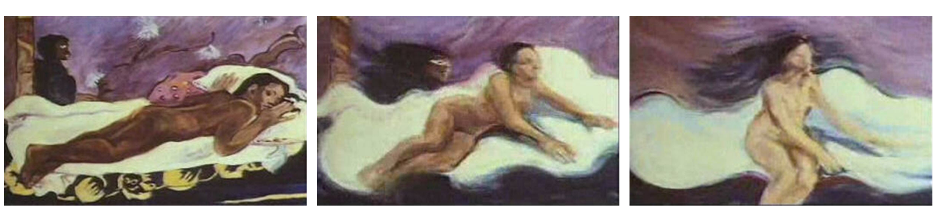
research directions– *meta standards*

All these standards! Wouldn't it be nice to have one single standard that encompasses them all? No, it would not! Simply, because such a standard is inconceivable, unless you take some proprietary standard or a particular platform as the defacto standard (which is the way some people look at the Microsoft win32 platform, ignoring the differences between 95/98/NT/2000/XP/...). In fact, there is a standard that acts as a glue between the various standards for multimedia, namely XML. XML allows for the interchange of data between various multimedia applications, that is the transformation of one encoding into another one. But this is only syntax. What about the semantics?

Both with regard to delivery and presentation the MPEG-4 proposal makes an attempt to delineate chunks of core fuctionality that may be shared between applications. With regard to presentation, SMIL may serve as an example. SMIL applications themselves already (re)use functionality from the basic set of XML-related technologies, for example to access the document structure through the DOM (Document Object Model). In addition, SMIL defines components that it may potentially share with other applications. For example, SMIL shares its animation facilities with SVG (the Scalable Vector Graphics format recommended by the Web Consortium).

The issue in sharing is, obviously, how to relate constructs in the syntax to their operational support. When it is possible to define a common base of operational support for a variety of multimedia applications we would approach our desired meta standard, it seems. A partial solution to this problem has been proposed in the now almost forgotten HyTime standard for time-based hypermedia. HyTime introduces the notion of *architectural forms* as a means to express the operational support needed for the interpretation of particular encodings, such as for example synchronization or navigation over bi-directional links. Apart from a base module, HyTime compliant architectures may include a units measurement module, a module for dealing with location addresses, a module to support hyperlinks, a scheduling module and a rendition module.

To conclude, wouldn't it be wonderful if, for example, animation support could be shared between rich media X3D and SMIL? Yes, it would! But as you may remember from the discussion on the timing models used by the various standards, there is still to much divergence to make this a realoistic option.



14

3.3 a multimedia semantic web?

To finish this chapter, let's reflect on where we are now with 'multimedia' on the web. Due to refined compression schemes and standards for authoring and delivery, we seemed to have made great progress in realizing *networked multimedia*. But does this progress match what has been achieved for the dominant media type of the web, that is text or more precisely textual documents with markup?

web content

- *1st generation* – hand-coded HTML pages
- *2nd generation* – templates with content and style
- *3rd generation* – rich markup with metadata (XML)

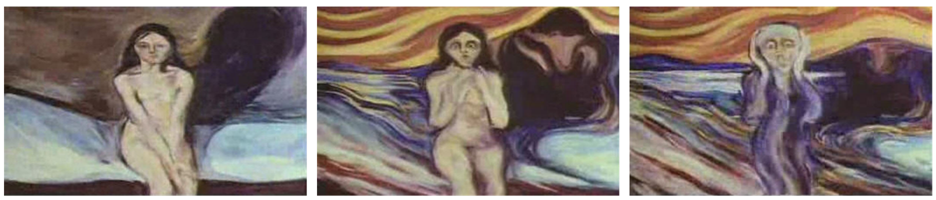
Commonly, a distinction is made between successive generations of web content, with the first generation being simple hand-coded HTML pages. The second generation may be characterized as HTML pages that are generated on demand, for example by filling in templates with contents retrieved from a database. The third generation is envisaged to make use of rich markup, using XML, that reflects the (semantic) content of the document more directly, possibly augmented with (semantic) meta-data that describe the content in a way that allows machines, for example search engines, to process it. The great vision underlying the third generation of web content is commonly refered to as the *the semantic web.* which enhances the functionality of the current web by deploying knowledge representation and inference technology from Artificial Intelligence, using a technology known as the *Resource Description Framework* (RDF). As phrased in Ossenbruggen et. al. (2001), the semantic web will bring

structure to the meaningful content of web pages,

thus allowing computer programs,such as search engines and intelligent agents, to do their job more effectively. For search engines this means more effective information retrieval, and for agents better opportunities to provide meaningful services.

A great vision indeed. So where are we with multimedia? As an example, take a *shockwave* or *flash* presentation showing the various musea in Amsterdam. How would you attach meaning to it, so that it might become an element of a semantic structure? Perhaps you wonder what meaning could be attached to it? That should not be too difficult to think of. The (meta) information attached to such a presentation should state (minimally) that the location is Amsterdam, that the sites of interest are musea, and (possibly) that the perspective is touristic. In that way, when you search for touristic information about musea in Amsterdam, your search engine should have no trouble in selecting that presentation. Now, the answer to the question how meaning can be attached to a presentation is already given, namely by specifying meta-information in some format (of which the only requirement is that it is machine-processable). For our *shockwave*or *flash* presentation we cannot dothis in a straightforward manner. But for MPEG-4 encoded material, as well as for SMIL content, such facilities are readily available.

Should we then always duplicate our authoring effort by providing (meta) information, on top of the information that is already contained in the presentation? No, in some cases, we can also rely to some extent on content-based search or feature extraction, as will be discussed in the following chapters.



15

Resource Description Framework – the Dublin Core

The Resource Description Framework, as the W3C/RDF[28] site informs us *integrates a variety of applications from library catalogs and world-wide directories to syndication and aggregation of news, software, and content to personal collections of music, photos, and events using XML as an interchange syntax.* The RDF specifications provide, in addition *a lightweight ontology system to support the exchange of knowledge on the Web.*

The Dublin Core Metadata Initiative[29] is an open forum engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models.

What exactly is meta-data? As phrased in the RDF Primer[30]

meta data

> Metadata is data about data. Specifically, the term refers to data used to identify, describe, or locate information resources, whether these resources are physical or electronic. While structured metadata processed by computers is relatively new, the basic concept of metadata has been used for many years in helping manage and use large collections of information. Library card catalogs are a familiar example of such metadata.

The Dublin Core proposes a small number of elements, to be used to give information about a resource, such as an electronic document on the Web. Consider the following example:

Dublin Core example

```
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns #"
   xmlns:dc="http://purl.org/dc/elements/1.1/"
   xmlns:dcterms="http://purl.org/dc/terms/">
```

[28] www.w3.org/RDF
[29] dublincore.org
[30] www.w3.org/TR/rdf-primer

```
<rdf:Description rdf:about="http://www.dlib.org/dlib/may98/miller/05miller.html">
 <dc:title>An Introduction to the Resource Description Framework</dc:title>
  <dc:creator>Eric J. Miller</dc:creator>
  <dc:description>The Resource Description Framework (RDF) is an
   infrastructure that enables the encoding, exchange and reuse of
   structured metadata. rdf is an application of xml that imposes needed
   structural constraints to provide unambiguous methods of expressing
   semantics. rdf additionally provides a means for publishing both
   human-readable and machine-processable vocabularies designed to
   encourage the reuse and extension of metadata semantics among
   disparate information communities. the structural constraints rdf
   imposes to support the consistent encoding and exchange of
   standardized metadata provides for the interchangeability of separate
   packages of metadata defined by different resource description
   communities. </dc:description>
 <dc:publisher>Corporation for National Research Initiatives</dc:publisher>
  <dc:subject>
   <rdf:Bag>
     <rdf:li>machine-readable catalog record formats</rdf:li>
     <rdf:li>applications of computer file organization and
      access methods</rdf:li>
   </rdf:Bag>
  </dc:subject>
  <dc:rights>Copyright ľ 1998 Eric Miller</dc:rights>
  <dc:type>Electronic Document</dc:type>
  <dc:format>text/html</dc:format>
  <dc:language>en</dc:language>
 <dcterms:isPartOf rdf:resource="http://www.dlib.org/dlib/may98/05contents.html"/>
 </rdf:Description>
</rdf:RDF>
```

Items such as *title*, *creator*, *subject* and *description*, actually all tags with the prefix *dc*, belong to the Dublin Core and are used to give information about the document, which incidentally concerns an introduction to the Resource Description Framework. The example also shows how *rdf* constructs can be used together with the Dublin Core elements. The prefixes *rdf* and *dc* are used to distinguish between the distinct namespaces of respectively RDF and the Dublin Core.

The Dublin Core contains the following elements:

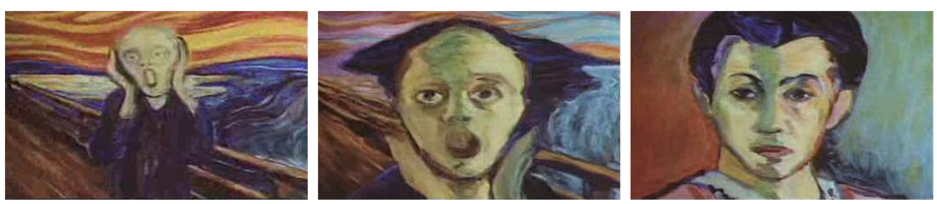
Dublin Core[31]

- *title* – name given to the resource
- *creator* – entity primarily responsible for making the content of the resource
- *subject* – topic of the content of the resource
- *description* – an account of the content of the resource
- *publisher* – entity responsible for making the resource available

[31] dublincore.org/documents/dces

- *contributor* – entity responsible for making contributions to the content of the resource
- *date* – date of an event in the lifecycle of the resource
- *type* – nature or genre of the content of the resource
- *format* – physical or digital manifestation of the resource
- *identifier* – unambiguous reference to the resource within a given context
- *source* – reference to a resource from which the present resource is derived
- *language* – language of the intellectual content of the resource
- *relation* – reference to a related resource
- *coverage* – extent or scope of the content of the resource
- *rights* – information about rights held in and over the resource

In section 10.3 we discuss an application in the domain of cultural heritage, where the Dublin Core elements are used to provide meta information about the information available for the conservation of contemporary artworks.



16

research directions– *agents everywhere*

The web is an incredibly rich resource of information. Or, as phrased in Baeza-Yates and Ribeiro-Neto (1999):

information repository

> *The Web is becoming a universal repository of human knowledge and culture, which has allowed unprecedented sharing of ideas and information in a scale never seen before.*

Now, the problem (as many of you can acknowledge) is to get the information out of it. Of course, part of the problem is that we often do not know what we are looking for. But even if we do know, it is generally not so easy to find our way. Again using the phrasing of Baeza-Yates and Ribeiro-Neto (1999):

browsing & navigation

> *To satisfy his information need, the user might navigate the hyperspace of web links searching for information of interest. However, since the hyperspace is vast and almost unknown, such a navigation task is usually inefficient.*

The solution of the problem of *getting lost in hyperspace* proposed in Baeza-Yates and Ribeiro-Neto (1999) is information retrieval, in other words *query & search*. However, this may not so easily be accomplished. As observed in Baeza-Yates and Ribeiro-Neto (1999), The main obstacle is the absence of a well-defined data model for the Web, which implies that information definition and structure is frequently of low quality. Well, that is exactly the focus of the semanics web initiative, and in particular of the Resource Description Framework discussed above.

Standardizing knowledge representation and reasoning about web resources is certainly one (important) step. Another issue, however, is how to support the user in finding the proper resources and provide the user with assistance in accomplishing his task (even if this task is merely finding suitable entertainment).

What we need, in other words, is a unifying model (encompassing both a data model and a model of computation) that allows us to deal effectively with web resources, including multimedia objects. For such a model, we may look at another area of research and development, namely *intelligent agtents*, which provides us not only with a model but also with a suitable metaphor and the technology, based on and extending object-oriented technology, to realize intelligent assistance, Eliens (2000).

For convenience, we make a distinction between two kinds of agents, *information agents* and *presentation agents*.

information agent

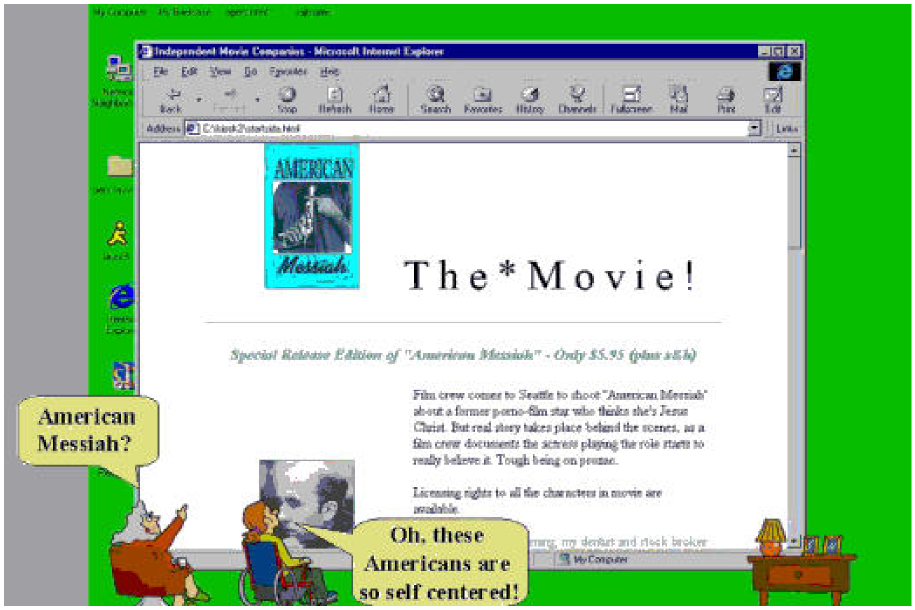
- gather information
- filter and select

Information agents are used to gather information. In addition, they filter the information and select those items that are relevant for the user. A key problem in developing information agents, however, is to find a proper representation of what the user considers to be relevant.

presentation agent

- access information
- find suitable mode of presentation

Complementary to the information agent is a *presentation agent* (having access to the information gathered) that displays the relevant information in a suitable way. Such a presentation agent can have many forms. To appetize your phantasy, you may look at the vision of *angelic guidance* presented in Broll et. al (2001). More concretely, my advice is to experiment with embodied agents that may present information in rich media 3D. In section **??**, we will present a framework for doing such experiments.



17

navigating information spaces Having *agents everywhere* might change our perspective on computing. But, it may also become quite annoying to be bothered by an agent each time that you try to interact with with your computer (you know what I mean!). However, as reported by Kristina Höök, even annoyance can be instrumental in keeping your attention to a particular task. In one of her projects, the *PERSONAS* project, which stands for

> *PERsonal and SOcial NAvigation through information spaceS*

the use of agents commenting on people navigating information space(s) is explored. As a note, the plural form of *spaces* is mine, to do justice to the plurality of information spaces.

As explained on the *PERSONAS* web site, which is listed with the acronyms, the *PERSONAS* project aims at:

PERSONAS

> *investigating a new approach to navigation through information spaces, based on a personalised and social navigational paradigm.*

The novel idea pursued in this project is to have agents (*Agneta* and *Frieda*) that are not helpful, but instead just give comments, sometimes with humor, but sometimes ironic or even sarcastic comments on the user's activities, in particular navigating an information space or (plain) web browsing. As can be read on the *PERSONAS* web site:

Agneta & Frieda

> *The AGNETA & FRIDA system seeks to integrate web-browsing and narrative into a joint mode. Below the browser window (on the desktop) are placed two female characters, sitting in their livingroom chairs, watching the browser during the session (more or less like watching television). Agneta and Frida (mother and daughter) physically react, comment, make ironic remarks about and develop stories around the information presented in the browser (primarily to each other), but are also sensitive to what the navigator is doing and possible malfunctions of the browser or server.*

In one of her talks, Kristina Höök observed that some users get really fed up with the comments delivered by *Agneta* and *Frieda*. So, as a compromise, the level of interference can be adjusted by the user, dependent on the task at hand.

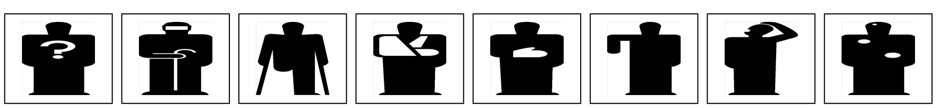
Agneta & Frieda

> *In this way they seek to attach emotional, comical or anecdotal connotations to the information and happenings in the browsing session. Through an activity slider, the navigator can decide on how active she wants the characters to be, depending on the purpose of the browsing session (serious information seeking, wayfinding, exploration or entertainment browsing).*

As you may gather, looking at the presentations accompanying this *introduction to multimedia* and Dialogs, I found the *PERSONAS* approach rather intriguing. Actually, the *PERSONAS* approach is related to the area of *affective computing*, see Picard (1998), which is an altogether different story.

The *Agneta* and *Frieda* software is available for download at the *PERSONAS* web site.

3.4 glueing it all together



18

questions

codecs and standards

1. (*) What role do standards play in *multimedia*? Why are standards necessary for compression and delivery. Discuss the MPEG-4 standard and indicate how it is related to other (possible) standards.

concepts

2. What is a *codec*?
3. Give a brief overview of current multimedia standards.

4. What criteria must a *(multimedia) semantic web* satisfy?

technology

5. What is the *data rate* for respectively (*compressed*) voice, audio and video?
6. Explain how a *codec* functions.
7. Which considerations can you mention for choosing a compression method?
8. Give a brief description of: XML, MPEG-4, SMIL, RM3D.

projects & further reading As a project, you may think of implementing for example JPEG compression, following Li and Drew (2004), or a SMIL-based application for cultural heritage.

You may further explore the technical issues on authoring DV material, using any of the Adobe[32], mentioned in appendix E. or compare

For further reading I advice you to take a look at the respective specifications of MPEG-4 and SMIL[33], and compare the functionality of MPEG-4 and SMIL-based presentation environments. An invaluable book dealing with the many technical aspects of compression and standards in Li and Drew (2004).

the artwork

1. costume designs – photographed from *Die Russchische Avantgarde und die Buhne 1890-1930*
2. theatre scene design, also from (above)
3. dance Erica Russel, Wiedermann (2004)
4. MPEG-4 – bits rates, from Koenen (2000).
5. MPEG-4 – scene positioning, from Koenen (2000).
6. MPEG-4 – up and downstream data, from Koenen (2000).
7. MPEG-4 – left: scene graph; right: sprites, from Koenen (2000).
8. MPEG-4 – syntax, from Koenen (2000).
9. MIT Media Lab[34] web site.
10. student work – *multimedia authoring I*, dutch windmill.
11. student work – *multimedia authoring I*, Schröder house.
12. student work – *multimedia authoring I*, train station.
13. animation – Joan Gratch, from Wiedermann (2004).
14. animation – Joan Gratch, from Wiedermann (2004).
15. animation – Joan Gratch, from Wiedermann (2004).
16. animation – Joan Gratch, from Wiedermann (2004).
17. Agneta and Frieda example.
18. signs – people, van Rooijen (2003), p. 246, 247.

[32] www.adobe.com/tutorials
[33] www.w3c.org/AudioVideo
[34] medai.mit.edu

Both the costume designs and theatre scene designs of the russian avantgarde movement are *expressionist* in nature. Yet, they show humanity and are in their own way very humorous. The dance animation by Erica Russell, using basic shapes and rhythms to express the movement of dance, is to some extent both solemn and equally humorous. The animations by Joan Gratch use *morphing*, to transform wellknown artworks into other equally wellknown artworks.

4. multimedia platforms

with DirectX 9 digital convergence becomes a reality

learning objectives

After reading this chapter you should be able to characterize the functionality of current multimedia platforms, to describe the capabilities of GPUs, to mention the components of the Microsoft DirectX 9 SDK, and to discuss how to integrate 3D and video.

Almost 15 years ago I bought my first multimedia PC, with Windows 3.1 Media Edition. This setup included a video capture card and a 4K baud modem. It was, if I remember well, a 100 Mhz machine, with 16 Mb memory and a 100 Mb disk. At that time, expensive as it was, the best I could afford. Some 4 years later, I acquired a Sun Sparc 1 multimedia workstation, with a video capture card and 3D hardware accelerator. It allowed for programming OpenGL in C++ with the GNU gcc compiler, and I could do live video texture mapping at a frame rate of about one per second. If you consider what is common nowadays, a 3Ghz machine with powerful GPU, 1 Gb of memory, a 1.5Mb cable or ADSL connection and over 100 Gb of disk space, you realize what progress has been made over the last 10 years.

In this chapter, we will look in more detail at the capability of current multimedia platforms, and we will explore the functionality of the Microsoft DirectX 0 platform. In the final section of this chapter, I will then report about the work I did with the DirectX 9 SDK to implement the ViP system, a presentation system that merges video and 3D.

1

4.1 developments in hardware and software

Following Moore's law (predicting the doubling of computing power every eighteen months), computer hardware has significantly improved. But perhaps more spectacular is the growth in computing power of dedicated multimedia hardware, in particular what is nowadays called the GPU (graphics processing unit). In Fernando and Kilgard (2003), he NVIDIA GeForce FX GPU is said to have 125 million of transistors, whereas the Intel 2.4GHz Pentium 4 contains only 55 million of transistors. Now, given the fact that the CPU (central processing unit) is a general purpose, or as some may like to call it, *universal* device, why is it necessary or desirable to have such specialized hardware, GPUs for graphics and, to be complete DSPs (digital signal processors) for audio?

a little bit of history

Almost evryone knows the stunning animation and effects in movies made possible by computer graphics, as for example the latest production of Pixar, *The Incredibles*. Such animation and effects are only possible by offline rendering, using factories of thousands of CPUs, crunching day and night to render all the frames.

At the basis of rendering lies traditional computer graphics technology. That is, the transformation of vertices (points in 3D space), rasterization (that is determining the pixel locations and pixel properties corresponding to the vertices), and finally the so-called raster operations (determining whether and how the pixels are written to the framebuffer). OpenGL, developed by SGI was the first commonly available software API (application programmers interface) to control the process of rendering. Later, Microsoft introduced Direct3D as an alternative for game programming on the PC platform.

The process outlined above is called the *graphics pipeline*. You put models, that is collections of vertices, in and you get (frames of) pixels out. This is indeed a simplification in that it does not explain how, for example, animation and lighting effects are obtained. To gain control over the computation done in the graphics pipeline, Pixar developed Renderman, which allows for specifying transformations on the models (vertices) as well as operations on the pixels (or fragments as they are called in Fernando and Kilgard (2003)) in a high level language. As vertex operations you may think of for example distortions of shape due to a force such as an explosion. As pixel operations, the coloring of pixels using textures (images) or special lighting and material properties. The languages for specifying such vertex or pixel operations are collectively called *shader* languages. Using offline rendering, almost anything is possible, as long as you specify it mathematically in a computationally feasible way.

The breakthrough in computer graphics hardware was to make such shading languages available for real-time computer graphics, in a way that allows, as Fernando and Kilgard (2003) phrase it, 3D game and application programmers and real-time 3D artists to use it in an effective way.

Leading to the programmable computer graphics hardware that we know today, Fernando and Kilgard (2003) distinguish between four generations of 3D accellerators.[35]

[35] The phrase GPU was introduced by NVIDIA to indicate that the capabilities of the GPU far exceed those of the VGA (video graphics array) originally introduced by IBM, which is

4 generations of GPU

- Before the introduction of the GPU, there only existed very expensive specialized hardware such as the machines from SGI.
- The first generation of GPU, including NVIDIA TNT2, ATI Rage and 3dfx Voodoo3, only supported rasterizing pre-transformed triangles and some limited texture operations.
- The second generation of GPUs, which were introduced around 1999, included the NVIDIA GeForce 2 and ATI Radeon 7500. They allowed for both 3D vertex transformations and some lighting, conformant with OpenGL and DirectX 7.
- The tird generation GPUs, including NVIDIA GeForce 3, Microsoft Xbox and ATI Radeon 8500, included both powerful vertex processing capabilities and some pixel-based configuration operations, exceeding those of OpenGL and DirectX 7.
- Finally, the fourth generation of GPUs, such as the NVIDIA GeForce FX and ATI Radeon 9700, allow for both complex vertex and pixel operations.

The capabilities of these latter generations GPUs motivated the development of high level shader languages, such as NVIDIA Cg and Microsoft HLSL. High level dedicated graphics hardware programming languages to control what may be called the programmable graphics pipeline.

the (programmable) graphics pipeline

Before discussing shading languages any further, let's look in some more detail at the graphics pipeline. But before that you must have an intuitive grasp of what is involved in rendering a scene.

Just imagine that you have created a model, say a teapot, in your favorite tool, for example Maya or 3D Studio Max. Such a model may be regarded as consisting of polygons, let's say triangles, and each vertex (point) of these triangles has apart from its position in (local) 3D space also a color. To render this model it must first be positioned in your scene, using so-called world coordinates. The *world transformation* is used to do this. The world transformation may change the position, rotation and scale of your object/model. Since your scene is looked at from one particular point of view we need to apply also a so-called *view transformation*, and to define how our view will be projected on a 2D plane, we must specify a *projection transformation*. Without going into the mathematical details, we may observe that these transformations can be expressed as 4x4 matrices and be combined in a single matrix, often referred to as the *world view projection matrix*, that can be applied to each of the vertices of our model. This combined transformation is the first stage in the process of rendering:

graphics pipeline

1. vertex transformation – apply world, view, projection transforms
2. assembly and rasterization – combine, clip and determine pixel locations
3. fragment texturing and coloring – determine pixel colors
4. raster operations – update pixel values

---

nothing more than a dumb framebuffer, requiring updates from the CPU.

The second phase, roughly, consists of cleaning up the collection of (transformed) vertices and determining the pixel locations that correspond to the model. Then, in the third phase, using interpolation or some more advanced method, coloring and lighting is applied, and finally a sequence of per-fragment or pixel operations is applied. Both OpenGL and Direct3D support among others an alpha (or transparency) test, a depth test and blending. The above characterized the fixed function graphics pipeline. Both the OpenGL and Direct3D API support the fixed function pipeline, offering many ways to set relevant parameters for, for example, applying lights, depth and texturing operations.

To understand what the programmable graphics pipeline can do for you, you would best look at some simple shader programs. In essence, the programmable pipeline allows you to perform arbitrary vertex operations and (almost) arbitrary pixel operations. For example, you can apply a time dependent morphing operation to your model. Or you can apply an amplification to the colors of your scene. But perhaps more interestingly, you can also apply an advanced lighting model to increase realism.

A simple morphing shader in ViP, see section 4.3.

2

a simple shader

When I began with programming shaders myself, I started with looking at examples from the DirectX SDK. Usually these examples were quite complex, and my attempt at modifying them often failed. Being raised in theoretical computer science, I changed strategy and developed my first shader program called *id*, which did nothing. Well, it just acted as the identity function. Then later I used this program as a starting point for writing more complex shader programs.

The *id* shader program is written in the DirectX 9 HLSL (high level shader language), and makes use of the DirectX Effects framework, which allows for specifying multiple vertex and pixel shaders, as well as multiple techniques and multiple passes in a single file.

The program starts with a declaration, specifying the global names for respectively the texture and the world/view/projection matrix. Also a texture sampler is declared, of which the function will become clear later.

HLSL declarations

```
texture tex;
float4x4 wvp;        // World * View * Projection matrix

sampler tex_sampler = sampler_state
{
   texture = /<tex/>;
};
```

It then defines, respectively, the vertex shader input and output, as structures. This declaration follows the standard C-like syntax for specifying elements in a structure, except for the identifiers in capitals, which indicate the semantics of the fields, corresponding to pre-defined registers in the GPU data flow.

vertex shader data flow

```
struct vsinput {
   float4 position : POSITION;
   float3 normal : NORMAL;
   float2 uv : TEXCOORD0;
};
struct vsoutput {
   float4 position   : POSITION;   // vertex position
   float4 color    : COLOR0;     // vertex diffuse color
   float2 uv  : TEXCOORD0;  // vertex texture coords
};
```

When the *vs_id* function, given below, is called, the input arguments are filled by the registers corresponding to the semantics pf the input structure. Similarly, the output results in setting the registers corresponding to the semantics of the output structure.

vertex shader

```
vsoutput vs_id( vsinput vx ) {
   vsoutput vs;

   vs.position = mul(vx.position, wvp);
   vs.color = color;
   vs.uv = vx.uv;

   return vs;
}
```

The *vs_id* function does exactly what the fixed graphics pipeline would do when transforming vertices. It applies the transformation to the vertex and passes both color and texture sampling coordinates to the pixel shader.

The pixel shader has a single color as output, which is obtained by sampling the texture, using the (interpolated) vertex color to modify the result.

pixel shader

```
struct psoutput
{
    float4 color : COLOR0;
};


psoutput ps_id( vsoutput vs )
{
    psoutput ps;

    ps.color = tex2D(tex_sampler, vs.uv) * vs.color;

    return ps;
}
```

Note that the *tex_sampler* comes from the global declaration above. The function *text2D* is a built-in for obtaining a color value from the sampler.

Finally, for each technique and each pass within a technique, in our case one technique with one pass, it must be indicated which function must be used for respectively the vertex shader and the pixel shader.

technique selection

```
technique render_id
{
    pass P0
    {
        VertexShader = compile vs_1_1 vs_id();
        PixelShader  = compile ps_2_0 ps_id();
    }
}
```

To make actual use of this program, the effect must be invoked from a DirectX or OpenGL program using the interface offered by the API.

Examples of Impasto, see examples – impasto

3

a morphing shader Slightly more complex is an example adapted from the morphing shader that can be found in ATI's Rendermonkey. To make a shader that morphs a cube into a ball and back, you must manipulate the vertices and the normals of the cube. For this to work your cube must have sufficient vertices, which is a property you can set in the tool that you use to make a cube.

morphing (vertex) shader

```
float3 spherePos = normalize(vx.position.xyz);
float3 cubePos = 0.9 * vx.position.xyz;

float t = frac(speed * time);
t = smoothstep(0, 0.5, t) - smoothstep(0.5, 1, t);

// find the interpolation factor
float lrp = lerpMin + (lerpMax - lerpMin) * t;

// linearly interpolate the position and normal
vx.position.xyz = lerp(spherePos, cubePos, lrp);
vx.normal = lerp(sphereNormal, cubeNormal, lrp);

// apply the transformations
vs.position = mul(wvp, vx.position);
```

The example uses the built-in function *lerp*, that performs linear interpolation between two values using an interpolation factor between 0 and 1.

color amplification As an example of a pixel shader, look at the fragment defining an amplification of coloring below. It merely amplifies the RGB components of the colors when this exceeds a certain treshold.

coloring (pixel) shader

```
float4 x = tex2D(tex_sampler, vs.uv);
if (x.r > x.g && x.r > x.b) { x.r *= xi; x.g *= xd; x.b *= xd; }
else  if (x.g > x.r && x.g > x.b) { x.g *= xi; x.r *= xd; x.b *= xd; }
else  if (x.b > x.r && x.b > x.g) { x.b *= xi; x.r *= xd; x.g *= xd; }
ps.color = x;
```

When you develop shaders you must keep in mind that a pixel shader is generally invoked far more often than a vertex shader. For example a cube can be defined using 12 triangles of each tree vertices. However, the number of pixels generated by this might be up to a million. Therefore any complex computation in the pixel shader will be immediately noticable in the performance. For example, a slightly more complex pixel shader than the one above makes my NVIDIA GeForce FX 5200 accelerated 3 GHz machine drop to 5 frames per second!

rendering of van Gogh painting with Impasto

4

example(s) – *impasto*

IMPaSTo[36] is a realistic, interactive model for paint. It allows the user to create images in the style of famous painters as in the example above, which is after a painting of van Gogh. The *impasto* system implements a physical model of paint to simulate the effect of acrylic or oilpaint, using Cg shaders for real-time rendering, Baxter et al. (2004).

research directions – *the art of shader programming*

At first sight, shader programming seems to be an esoteric endeavor. However, as already indicated in this section, there are a number of high level languages for shader programming, including NVIDIA Cg and Microsoft HLSL. Cg is a platform independent language, suitable for both OpenGL and Direct3D. However, counter to what you might expect also Microsoft HLSL can be used for the OpenGL platform when you choose the proper runtime support.

To support the development of shaders there are, apart from a number of books, some powerful tools to write and test your shaders, in particular the already mentioned ATI Rendermonkey tool, the CgFx tool, which both produce HLSL code, as well as the Cg viewer and the effect tool that comes with the Microsoft DirectX 9 SDK.

Although I am only a beginning shader programmer myself, I find it truly amazing what shaders can do. For a good introducion I advice Fernando and Kilgard (2003). Futher you may consult Engel (2004a), Engel (2004b) and Engel (2005). Written from an artist's perspective is St-Laurent (2004).

[36] gamma.cs.unc.edu/IMPaSTo

Stacks and stacks of books on DirectX

5

4.2 DirectX 9 SDK

Many of the multimedia applications that you run on your PC, to play games, watch video, or browse through your photos, require some version of Direct X to be installed. The most widely distributed version of Direct X is 7.0. The latest version is the october release of 2004. This is version 9c. What is DirectX? And, more importantly, what can you do with it? In the DirectX documentation that comes with the SDK, we may read:

DirectX

> Microsoft DirectX is an advanced suite of multimedia application programming interfaces (APIs) built into Microsoft Windows; operating systems. DirectX provides a standard development platform for Windows-based PCs by enabling software developers to access specialized hardware features without having to write hardware-specific code. This technology was first introduced in 1995 and is a recognized standard for multimedia application development on the Windows platform.

Even if you don't use the DirectX SDK yourself, and to do that you must be a quite versatile programmer, then you will find that the tools or plugins that you use do depend on it. For example, the WildTangent[37] game engine plugin makes the DirectX 7 functionality available through both javascript and a Java interface. So understanding what DirectX has to offer may help you in understanding and exploiting the functionality of your favorite tool(s) and plugin(s).

DirectX 9.0 components

In contrast to OpenGL, the DirectX SDK is not only about (3D) graphics. In effect, it offers a wide range of software APIs and tools to assist the developer of multimedia applications. The components of the DirectX 9 SDK include:

DirectX 9 components

- Direct3D – for graphics, both 2D and 3D
- DirectInput – supporting a variety of input devices
- DirectPlay – for multiplayer networked games
- DirectSound – for high performance audio

[37] www.wildtangent.com

- DirectMusic – to manipulate (non-linear) musical tracks
- DirectShow – for capture and playback of multimedia (video) streams

In addition there is an API for setting up these components. Also, DirectX supports so-called *media objects*, which provide a standard interface to write audio and video encoders, decoders and effects.

Altogether, this is a truly impressive and complex collection of APIs. One way to become familiar with what the DirectX 9 SDK has to offer is to start up the sample browser that is part of the SDK and explore the demos. Another way is to read the online documentation that comes with the SDK, but perhaps a better way to learn is to make your choice from the large collection of introductory books, and start programming. At the end of this chapter, I will provide some hints about how to get on your way.

Direct3D

In the DirectX 9 SDK, Direct3D replaces the DirectDraw component of previous versions, providing a single API for all graphics programming. For Direct3D there is a set of simple, well-written tutorials in the online documentation, that you should start with to become familiar with the basics of graphics programming in DirectX.

Direct3D tutorials

- tutorial 1: creating a device
- tutorial 2: rendering vertices
- tutorial 3: using matrices
- tutorial 4: creating and using lights
- tutorial 5: using texture maps
- tutorial 6: using meshes

Before you start working with the tutorial examples though, you should acquire sufficient skill in C++ programming[38] and also some familiarity with Microsoft Visual Studio .NET.

One of the most intricate (that means difficult) aspects of programming Direct3D, and not only for novices, is the creation and manipulation of what is called the *device*. It is advisable to take over the default settings from an example, and only start experimenting with more advanced setting after you gained some experience.

DirectSound – the *drumpad* example

The DirectX SDK includes various utility libraries, for example the D3DX library for Direct3D, to simplify DirectX programming.

As an example of a class that you may create with DirectSound, using such a utility library, look at the *drumpad* below. The *drumpad* class can be integrated

[38] The DirectX 9 SDK also offers APIs for C# and VisualBasic .NET. See the *research directions* at the end of this section.

in your 3D program, using DirectInput, to create your own musical instrument. The header of the class, which is, with some syntactical modifications, taken from the SDK samples section, looks as follows:

```
class drumpad {
public:
    drumpad()
    ~drumpad();
    bool    initialize( DWORD dwNumElements, HWND hwnd );
    bool    load( DWORD dwID, const TCHAR* tcszFilename );
    bool    play( DWORD dwID );
protected:
    void    CleanUp();
    CSoundManager* m_lpSoundManager;
    CSound **      m_lpSamples;
};
```

The interface offers some methods for creating and destroying a *drumpad* object, initialization, loading sounds and for playing the sounds that you loaded. The *CSoundManager* is a class offered by the utility library for DirectSound.

The *play* function is surprisingly simple.

```
bool drumpad::play( DWORD id ) {
    m_lpSamples[id] -> Stop();
    m_lpSamples[id] -> Reset();
    m_lpSamples[id] -> Play( 0, 0 );
    return true;
}
```

The *id* parameter is a number that may be associated with for example a key on your keyboard or some other input device. Using the *drumpad* class allows you to make your own VJ program, as I did in the system I will describe in the next section. In case you are not familiar with either C++ or object-oriented programming, you should study object-oriented software development first. See for example Eliens (2000).

DirectShow

DirectShow is perhaps the most powerful component of the DirectX SDK. It is the component which made Mark Pesce remark that with the DirectX 9 SDK *digital convergence has become a reality*.[39] A technical reality, that is, Pesce (2003).

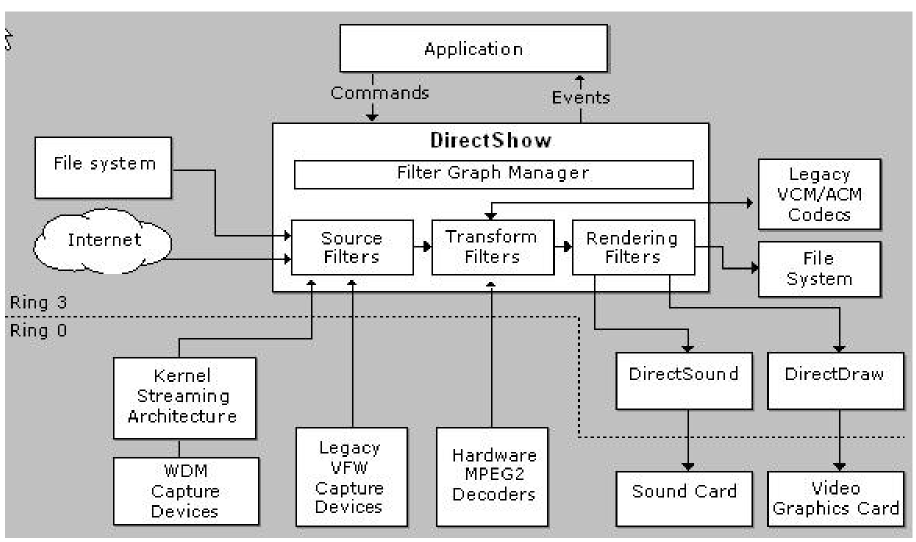
As we have seen in chapter 3, working with multimedia presents some major challenges:

multimedia challenges

[39] It is historically interesting to note that Mark Pesce may be regarded as the inventor, or initiator, of VRML, which was introduced in 1992 as the technology to realize a 3D web, as interlinked collection of 3D spaces.

- volume – multimedia streams contain large amounts of data, which must be processed very quickly.
- synchronization – audio and video must be synchronized so that it starts and stops at the same time, and plays at the same rate.
- delivery – data can come from many sources, including local files, computer networks, television broadcasts, and video cameras.
- formats – data comes in a variety of formats, such as Audio-Video Interleaved (AVI), Advanced Streaming Format (ASF), Motion Picture Experts Group (MPEG), and Digital Video (DV).
- devices – the programmer does not know in advance what hardware devices will be present on the end-user's system.

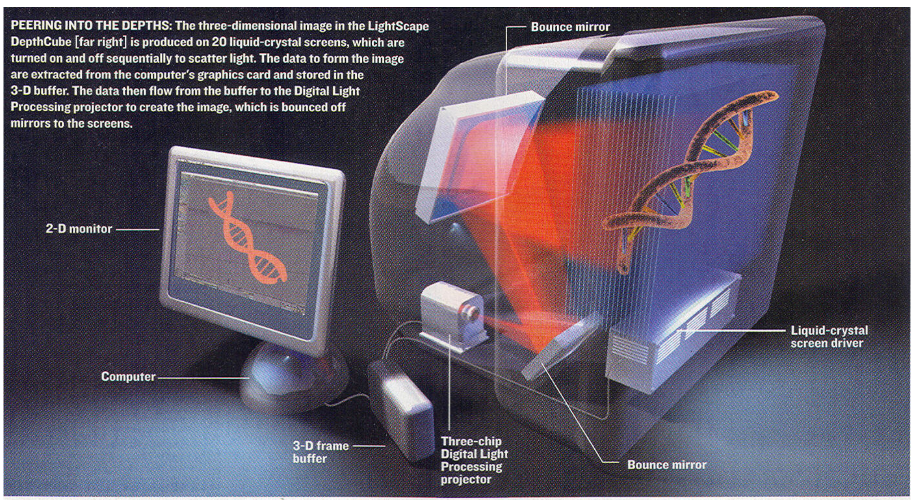
The DirectShow component was designed, as we learn from the online documentation, to address these challenges and to simplify the task of creating applications by isolating applications from the complexities of data transports, hardware differences and synchronization. The solution DirectShow provides is a modular architecture that allows the developer to set up a data flow graph consisting of *filters*. Such filters may be used for capturing data from, for example, a video camera or video file, for deploying a codec, through the audio compression manager (ACM) or video compression manager (VCM) interfaces, and for rendering, either to the file system or in the application using DirectSound and DirectDraw and Direct3D.



6

The diagram above, taken from the DirectX 9 SDK online documentation, shows the relations between an application, the DirectShow components, and some of the hardware and software components that DirectShow supports.

An interesting and convenient feature of the filter-based dataflow architecture of DirectShow is *SmartConnect*, which allows the developer to combine filters by indicating constraints on media properties such as format. The actual connections then, which involves linking input pins to output pins, is done automatically by searching for the right order of filters, and possibly the introduction of auxiliary filters to make things match.

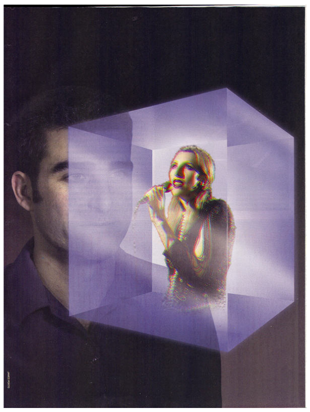


DepthCube, see example(s) – *3D vision*

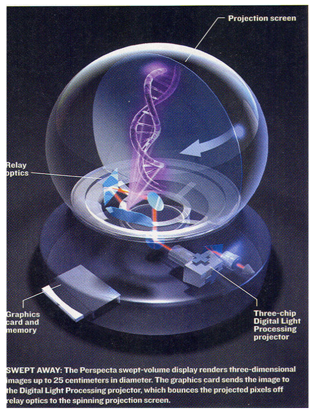
7

DirectX application development

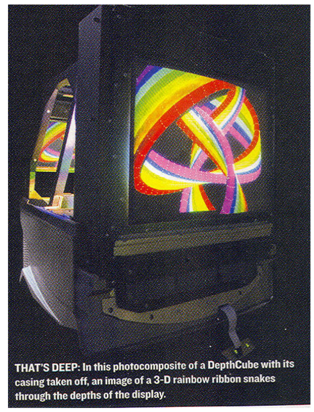
The examples that come with the DirectX'9 SDK use an application utility library, which includes a general application class that takes care of most of the details of creating an application and rendering window, initialization and event handling. For each of the SDK components there are numerous examples, ranging in difficulty from beginners to expert level. There are also a number of examples that illustrate how to mix the functionality of different SDK components, as for example the projection of video on 3D, which we will discuss in more detail in the next section.



3D vision



Perspectra



DepthCube

8

example(s) – *3D vision*

Have you ever wondered how it would feel to be in Star Trek's holodeck, or experience your game in a truly spatial way, instead of on a flat LCD-display. In Sullivan (2005), an overview is given of technology that is being developed to enable volumetric display of 3D data, in particular the Perspecta swept-volume display (middle) and LightSpace DepthCube (right), that uses a projector behind a stack of 20 liquid-crystal screens.

The first approach of displaying volumetric data, taken by the Perspecta swept-volume display, is to project a sequence of images on a rotating sheet of reflective material to create the illusion of real volume. The psychological mechanism that enables us to see volumes in this way is the same as the mechanism that forces us to see motion in frame-based animation, at 24 frames per second, namely *persistence of vision*.

LightSpace DepthCube uses a stack of 20 transparent screens and alternates between these screens in a rapid way, thus creating the illusion of depth in a similar way. In comparison with other approaches of creating depth illusion, the solutions sketched above require no special eyewear and do not impose any strain on the spectator due to unnatural focussing as for example with autostereoscopic displays.

For rendering 3D images on either the Perspecta or DepthCube traditional rendering with for example OpenGL suffices, where the z-coordinate is taken as an indication for selecting a screen or depth position on the display. Rendering with depth, however, comes at a price. Where traditional rendering has to deal with, say 1024x748 pixels, the DepthCube for example needs to be able to display 1024x748x20, that is 15.3 million, *voxels* (the volumetric equivalent of a pixel) at a comparable framerate.

research directions– *the next generation multimedia platform*

Factors that may influence your choice of multimedia development platform include:

- platform-dependence – both hardware and OS
- programming language – C/C++, Java, .NET languages
- functionality – graphics, streaming media
- deployment – PC/PDA, local or networked, web deployment

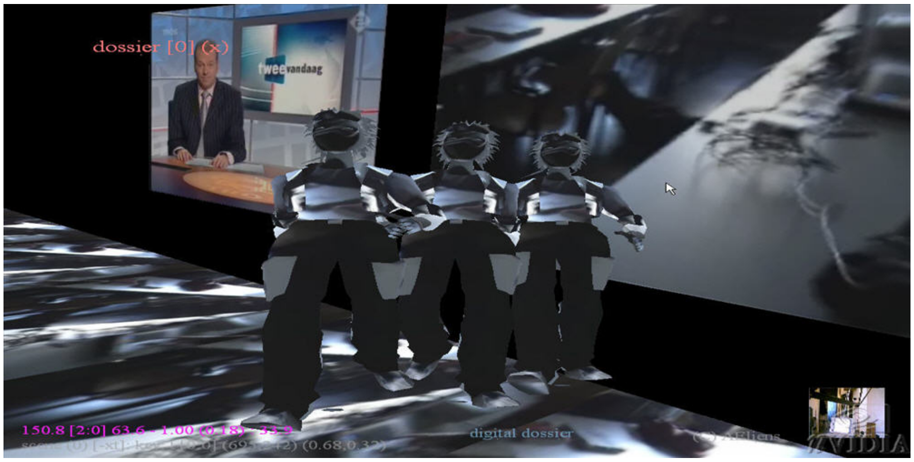
A first dividing line is whether you prefer to develop on/for Linux or Microsoft windows. Another dividing line, indeed, is your choice of programming language, C/C++, Java or .NET languages. Another factor that may influence your choice is the functionality you strive for. For example, Managed DirectX, for the .NET languages, provides only limited support for DirectShow and does not allow for capturing live video from a DV camera. And finally, it matters what deployment you wish to target for, mobile phone, PDAs or PCs, and whether you plan to make stand-alone applications or applications that must run in a web browser.

Apart from the hard-core programming environments such as the Microsoft DirectX 9 SDK, the Java Media Framework, OpenGL with OpenML extensions for streaming media, or the various open source (game development) toolkits, there are also high-level tools/environments, such as Macromedia Director MX, that allow you to create similar functionality with generally less effort, but also less control. In appendix E, a number of resources are listed that may assist you in determining your choice.

Given the range of possible options it is futile to speculate on what the future will offer. Nevertheless, whatever your choice is, it is good to keep in mind, quoting Bill Gates:

> *Software will be the single most important force in digital entertainment over the next decade.*

It should not come as a surprise that this statement is meant to promote a new initiative, XNA, which as the announcement says *is targeted to help contain the skyrocketing development costs and allow developers to concentrate on the unique content that differentiates their games.*



Animation in front of television news in ViP

9

4.3 merging video and 3D

In june 2003, I was approached by a theatre production company to advice on the use of "VR in theatre". As described in more detail in section 9.3, I explored what technology was available to realize such VR-augmented theatre. These explorations resulted in the development of the ViP system, that I once announced as follows:

www.virtualpoetry.tv

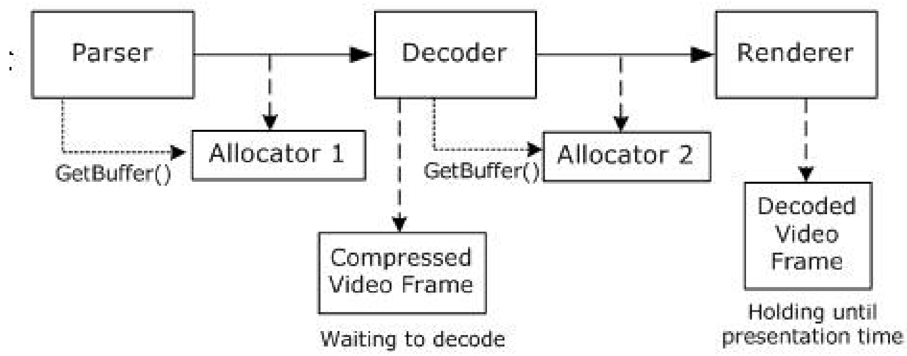
> *The ViP system enhances your party with innovative multimedia presentations.*
>
> *It supports multiple webcams and digital video cameras, mixed with video and images, enhanced by 3D animations and text, in an integrated fashion.*
>
> *For your party, we create a ViP presentation, with your content and special effects, to entertain your audience.*

In the course of time, I continued working on the system and, although I do not use it for parties, but rather for enlivening my lectures, it does include many of the features of a VJ system, such as the *drumpad* described in 3.2.

The major challenge, when I started its development, was to find an effective way to map live video from a low/medium resolution camera as textures onto 3D geometry. I started with looking at the ARToolkit but I was at the time not satisfied with its frame rate. Then, after some first explorations, I discovered that mapping video on 3D was a new (to some extent still experimental) built-in feature of the DirectX 9 SDK, in the form of the VMR9 (video mixing renderer) filter.

the Video Mixing Renderer filter

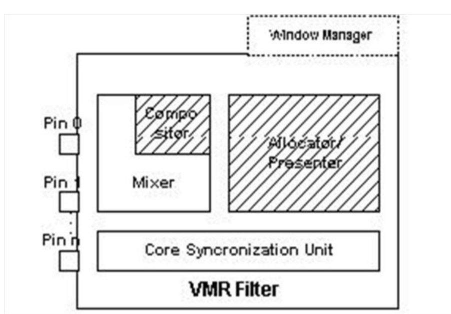
The VMR filter is a compound class that handles connections, mixing, compositing, as well as synchronization and presentation in an integrated fashion. But before discussing the VMR9 in more detail, let's look first at how a single media stream is processed by the filter graph, as depicted in the figure below.



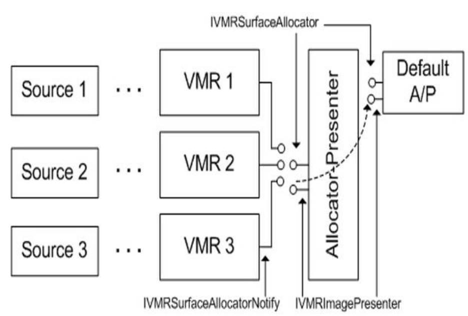
10

Basically, the process consists of the phases of parsing, decoding and rendering. For each of these phases, dependent on respectively the source, format and display requirements, a different filter may be used. Synchronization can be either dtermined by the renderer, by pulling new frames in, or by the parser, as in the case of live capture, by pushing data on the stream, possibly causing the loss of data when decoding cannot keep up with the incoming stream.

The VMR was originally introduced to allow for mixing multiple video streams, and allowed for user-defined *compositor* and *allocator/presenter* components.



(a) VMR filter



(b) multiple VMRs

11

Before the VMR9, images could be obtained from the video stream by intercepting this stream and copying frames to a texture surface. The VMR9, however, renders the frames directly on Direct3D surfaces, with (obviously) less overhead. Although the VMR9 supports multiple video pins, for combining multiple video streams, it does not allow for independent search or access to these streams. To do this you must deploy multiple video mixing renderers that are connected to a common allocator/presenter component, as depicted on the right of the figure above (b).

When using the VMR9 with Direct3D, the rendering of 3D scenes is driven by the rate at which the video frames are processed.

Lecture on digital dossier for Abramovic, in ViP

12

the ViP system

In developing the ViP system, I proceeded from the requirement to project live video capture in 3D space. As noted previously, this means that incoming video drives the rendering of 3D scenes and that, hence, capture speed determines the rendering frame rate.

I started with adapting the simple *allocator/presenter* example from the DirectX 9 SDK, and developed a scene management system that could handle incoming textures from the video stream. The *scene* class interface, which allows for (one-time) initialization, time-dependent compositing, restore device setting and rendering textures, looks as follows:

```
class scene {
public:
    virtual int init(IDirect3DDevice9*);  // initialize scene (once)
    virtual int compose(float time);  // compose (in the case of an
  animation)
    virtual int restore(IDirect3DDevice9*);  // restore device settings
    virtual int render(IDirect3DDevice9* device, IDirect3DTexture9*
  texture);
protected:
```

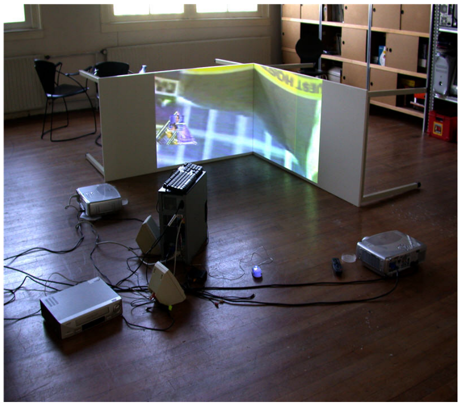
```
...
};
```

The scene graph itself was constructed as a tree, using both arrays of (sub) scenes as well as a dictionary for named scenes, which is traversed each time a video texture comes in. The requirements the scene management system had to satisfy are further indicated in section 9.3. Leaving further details aside, observe that for the simple case of one incoming video stream, transferring the texture by parameter suffices.

Later on, I adapted the *GamePlayer* which uses multiple video mixing renderes, and then the need arose to use a different way of indexing and accessing the textures from the video stream(s). So, since it is good practice in object-oriented software engineering to suppress parameters by using object instance variables, the interface for the *scene* class changed into:

```
class scene {
public:
        virtual int load();   // initialize scene (once)
        virtual int compose();  // compose (in the case of an animation)
        virtual int restore();  // restore device settings
        virtual int render();  // display the (sub) scene
protected:
...
};
```

Adopting the scene class as the unifying interface for all 3D objects and compound scenes proved to be a convenient way to control the complexity of the ViP application. However, for manipulating the textures and allocating shader effects to scenes, I needed a global data structure (dictionaries) to access these items by name, whenever needed. As a final remark, which is actually more concerned with the software engineering of such systems that its functionality per se, to be able to deal with the multiple variant libraries that existed in the various releases of DirectX 9, it was needed to develop the ViP library and its components as a collection of DLLs, to avoid the name and linking clashes that would otherwise occur.



installation

reality of TV news

13

example(s) – *reality of TV news*

The *Reality of TV news* project by Peter Frucht uses an interesting mix of technology:

- live video capture from the device of an external USB2.0 TV card
- live audio capture from the soundcard (line in)
- display of live audio and video with java3D (had to be invented)
- autonomous 3D objects with a specified lifetime
- collision behaviour (had to be invented)
- changing of texture-, material- and sound characteristics at runtime
- dual-screen display with each screen rotated toward the other by 45 degrees about the Y-axis
- 3D sound

In the project, as phrased by Peter Frucht, *the permanent flow of the alternating adverts and news reports are captured live and displayed in a 3D virtual-reality installation. The currently captured audio and video data is displayed on the surface of 3D shapes as short loops. The stream enters the 3D universe piece by piece (like water drops), in this way it is getting displaced in time and space - news reports and advertising will be displayed partly in the same time. By colliding to each other the 3D shapes exchange video material. This re-editing mixes the short loops together, for instance some pieces of advertising will appear while the newsreader speaks.*

The software was developed by Martin Bouma, Anthony Augustin and Peter Frucht himself, with jdk 1.5, java3d 1.31, Java Media Framework 2.1.1e. The primary technological background of the artist, Peter Frucht, was the book *CodeArt*[40], Trogemann & Viehoff (2004), by his former professor from the Media Art School in Cologne, Germany. The book is unfortunately only available in German, and should be translated in English!

[40] java.khm.de

research directions– *augmented reality*

In the theatre production that motivated the development of the ViP system, the idea was to have wearable LCD-projection glasses, with a head-mounted low resolution camera. This setup is common in *augmented reality* applications, where for example a historic site is enriched with graphics and text, laid on top of the (video rendered) view of the site. Since realtime image analysis is generally not feasible, either positioning and orientation information must be used, or simplified markers indicating the significant spots in the scene, to determine what information to use as an overlay and how it should be displayed.

The ARToolkit[41] is an advanced, freely available, toolkit, that uses fast marker recognition to determine the viewpoint of a spectator. The information that is returned on the recognition of a marker includes both position and orientation, which may be used by the application to draw the overlay graphics in accordance with the spectator's viewpoint.

Augnented reality is likely to become a hot thing. In april 2005 it was featured at BBC World[42], with a tour through Basel.

4.4 development(s) – gaming is a waste of time

14

questions

multimedia platforms

1. What components does a multimedia platform consist of? Discuss both hardware and software components.

concepts

2. Characterize the functionality of current multimedia platforms.
3. Explain the notions of vertex shader and pixel shader.
4. Indicate what solutions exist for merging video and 3D graphics.

technology

5. Characterize the capabilty of current GPUs.
6. What does HLSL stand for? Give some examples of what it is used for.
7. What are the components of the DirectX 9 SDK?
8. Explain how the VMR9 works. Give an example.

[41] artoolkit.sourceforge.net
[42] www.bbcworld.com/content/template_clickonline.asp?pageid=665&co_pageid=3

projects & further reading As a project, I suggest the development of shader programs using Rendermonkey[43] or the Cg Toolkit[44], or a simple game in DirectX.

You may further explore the possibilities of platform independent integration of 3D and media, by studying for example OpenML[45]. For further reading, among the many books about DirectX, I advice Luna (2003), Adams (2003) and Fay et al. (2004).

the artwork

1. dutch light – photographs from documentary film Dutch Light[46].
2. ViP – screenshot, with morphing shader, see section 4.3.
3. impasto – examples, see section 4.1
4. impasto – after a painting of van Gogh, using Cg shaders,
5. 3D vision, from Sullivan (2005), see example(s) section 4.2.
6. idem.
7. photographs of DirectX and multimedia books, by the author.
8. DirectX – diagram from online documentation.
9. ViP – screenshot, with the news and animations.
10. DirectX – diagram from online documentation.
11. DirectX – diagram from online documentation.
12. ViP – screenshot, featuring Abramovic.
13. Peter Frucht – *Reality of TV news*, see section 4.3.
14. signs – people, van Rooijen (2003), p. 248, 249.

The theme of the artwork of this chapter is *realism*. In the documentary *dutch light*, it was investigated whether the famous *dutch light* in 17th century painting really existed. The photographs shown here are a selection of shots that were taken on particular locations over a period of time. However, as an art historian formulated it in the documentary: *dutch light* is nothing but a *bag of tricks* shared by dutch 17th century painters. The examples from *impasto* demonstrated that, after all, realism is an arbitrary notion.

[43] www.ati.com/developer/RenderMOnkey
[44] www.nvidia.com/cg
[45] www.khronos.org/openml
[46] www.dutchlight.nl

part iii. multimedia information retrieval

.. my history might well be your future ...
ted nelson

5. information retrieval
6. content annotation
7. information system architecture

2

reading directives In the following chapters we will discuss how we can make the various media formats, including text, images, audio and video amenable to search, either by analyzing content or by providing explicit meta information. For video, in particular, we develop a simple annotation logic that captures both the story line and the actors, that is persons and objects, that figure in it.

Essential sections are section 5.1, that characterizes scenarios for information retrieval, section 5.3, that introduces standard information retrieval concepts stemming from text search, section 6.4, that defines the aforementioned annotation logic, and section 7.2, that gives an outline of an abstract multimedia data format.

Section 6.3 is rather technical and may safely be skipped. Also sections 5.2, 6.1 and 7.3 may be skipped on first reading.

perspectives Apart from the many technical issues in information retrieval, perhaps the human interaction issues are the most urgent. As possible perspectives to look at these issues. consider:

perspectives – multimedia information retrieval

- application(s) – digital dossier
- psychological – focus
- experimental – user interaction
- algorithmic – (information) access
- system – unified presentation space
- presentation – embodied agents
- search – semantic annotation
- commercial – future systems

As you will see in the *research directions* given for each section, there are many proposals to improve interaction, for example the use of 3D virtual environments as an alternative way of presenting information.

essay topics For further study you may want to look at algorithms for analyzing content, annotation schemes for pareticular application domains, or the presentation issues mentioned before. Possible essay titles are:

- searching the web – searching for images, video and sound
- finding a tune – mobile music search services

Since the retrieval problem seems to be rather intractable in a general fashion, you should limit your discussion to a specific domain, for example retrieval in the domain of cultural heritage, and relate technical issues to the requirements of users in that particular domain.

3

the artwork

1. kata – japanese martial arts picture.
2. signs – japanese coats of arms, van Rooijen (2003), p. 140, 141.
3. photographs – Jaap Stahlie[47], two early experiments (left, and right)

[47]www.jaapstahlie.com

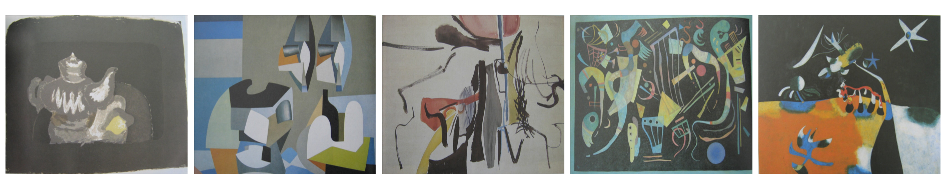
5. information retrieval

information retrieval is usually an afterthought

learning objectives

After reading this chapter you should be able to describe scenarios for information retrieval, to explain how content analysis for images can be done, to characterize similarity metrics, to define the notions of recall and precision, and to give an example of frequence tables, as used in text search.

Searching for information on the web is cumbersome. Given our experiences today, we may not even want to think about searching for multimedia information on the (multimedia) web. Nevertheless, in this chapter we will briefly sketch one of the possible scenarios indicating the need for multimedia search. In fact, once we have the ability to search for multimedia information, many scenarios could be thought of. As a start, we will look at two media types, images and documents. We will study search for images, because it teaches us important lessons about content analysis of media objects and what we may consider as *being similar*. Perhaps surprisingly, we will study text documents because, due to our familiarity with this media type, text documents allow us to determine what we may understand by effective search.



1

5.1 scenarios

Multimedia is not only for entertainment. Many human activities, for example medical diagnosis or scientific research, make use of multimedia information. To get an idea about what is involved in multimedia information retrieval look at the following scenario, adapted from Subrahmanian (1998),

Amsterdam Drugport

> *Amsterdam is an international centre of traffic and trade. It is renowned for its culture and liberal attitude, and attracts tourists from various ages, including young tourists that are attracted by the availability of soft drugs. Soft drugs may be obtained at so-called coffeeshops, and the possession of limited amounts of soft drugs is being tolerated by the authories.*
>
> *The European Community, however, has expressed their concern that Amsterdam is the centre of an international criminal drug operation. Combining national and international police units, a team is formed to start an exhaustive investigation, under the code name* Amsterdam Drugport.

Now, without bothering ourselves with all the logistics of such an operation, we may establish what sorts of information will be gathered during the investigation, and what support for (multimedia) storage and (multimedia) information retrieval must be available.

Information can come from a variety of sources. Some types of information may be gathered continuously, for example by video cameras monitoring parking lots, or banks. Some information is already available, for example photographs in a (legacy database) police archive. Also of relevance may be information about financial transactions, as stored in the database of a bank, or geographic information, to get insight in possible dug traffic routes.

From a perspective of information storage our informatio (data) include the following media types: images, from photos; video, from surveillance; audio, from interviews and phone tracks; documents,from forensic research and reports; handwriting, from notes and sketches; and structured data, from for example bank transactions.

We have to find a way to store all these data by developing a suitable multimedia information system architecture, as discussed in chapter 6. More importantly, however, we must provide access to the data (or the information space, if you will) so that the actual police investigation is effectively supported. So, what kind of queries can we expect? For example, to find out more about a murder which seems to be related to the drugs operation.

retrieval

- *image query* – all images with this person
- *audio query* – identity of speaker
- *text query* – all transactions with BANK Inc.
- *video query* – all segments with victim
- *complex queries* – convicted murderers with BANK transactions
- *heterogeneous queries* – photograph + murderer + transaction
- *complex heterogeneous queries* – *in contact with* + murderer + transaction

Apparently, we might have simple queries on each of the media types, for example to detect the identity of a voice on a telephone wiretap. But we may also have more complex queries, establishing for example the likelihood that a murderer known by the police is involved, or even *heterogeneous* queries (as they are called in Subrahmanian (1998)), that establish a relation between information coming from multiple information sources. An example of the latter could be, *did the person on this photo have any transactions with that bank in the last three months*, or nore complex, *give me all the persons that have been in contact with the victim (as recorded on audio phonetaps, photographs, and video surveillance tapes) that have had transactions with that particular bank.*

I believe you'll have the picture by now. So what we are about to do is to investigate how querying on this variety of media types, that is images, text, audio and video, might be realized.

2

research directions– *information retrieval models*

Information retrieval research has quite a long history, with a focus on indexing text and developing efficient search algorithms. Nowadays, partly due to the wide-spread use of the web, research in information retrieval includes modeling, classification and clustering, system architectures, user interfaces, information visualisation, filtering, descriptive languages, etcetera. See Baeza-Yates and Ribeiro-Neto (1999).

Information retrieval, according to Baeza-Yates and Ribeiro-Neto (1999), deals with the representation, storage, organisation of, and access to information items. To see what is involved, imagine that we have a (user) query like:

find me the pages containing information on ...

Then the goal of the information retrieval system is to retrieve information that is useful or relevant to the user, in other words: *information that satisfies the user's information need.*

Given an information repository, which may consist of web pages but also multimedia objects, the information retrieval system must extract syntactic and semantic information from these (information) items and use this to match the user's information need.

Effective information retrieval is determined by, on the one hand, the *user task* and, on the other hand, the *logical view* of the documents or media objects that constitute the information repository. As user tasks, we may distinguish between *retrieval* (by query) and *browsing* (by navigation). To obtain the relevant information in retrieval we generally apply *filtering*, which may also be regarded as a ranking based on the attributes considered most relevant.

The logical view of text documents generally amounts to a set of index terms characterizing theb document. To find relevant index terms, we may apply operations to the document, such as the elimination of stop words or text stemming. As you may easily see, full text provides the most complete logical view, whereas a small set of categories provides the most concise logical view. Generally, the user task will determine whether semantic richness or efficiency of search will be considered as more important when deciding on the obvious tradeoffs involved.

information retrieval models In Baeza-Yates and Ribeiro-Neto (1999), a great variety of information retrieval models is described. For your understanding, an information retrieval model makes explicit how index terms are represented and how the index terms characterizing an information item are matched with a query.

When we limit ourselves to the classic models for search and filtering, we may distinguish between:

information retrieval models

- boolean or set-theoretic models
- vector or algebraic models
- probabilistic models

Boolean models typically allow for *yes/no* answers only. The have a set-theoretic basis, and include models based on fuzzy logic, which allow for somewhat more refined answers.

Vector models use algebraic operations on vectors of attribute terms to determine possible matches. The attributes that make up a vector must in principle be orthogonal. Attributes may be given a weight, or even be ignored. Much research has been done on how to find an optimal selection of attributes for a given information repository.

Probabilistic models include general inference networks, and belief networks based on Bayesan logic.

Although it is somewhat premature to compare these models with respect to their effectiveness in actual information etrieval tasks, there is, according to Baeza-Yates and Ribeiro-Neto (1999), a general consensus that vector models will outperform the probabilistic models on general collections of text documents. How they will perform for arbitrary collections of multimedia objects might be an altogether different question!

Nevertheless, in the sections to follow we will focus primarily on generalized vector representations of multimedia objects. So, let's conclude with listing the advantages of vector models.

vector models

- attribute term weighting scheme improves performance
- partial matching strategy allows retrieval of approximate material
- metric distance allows for sorting according to degree of similarity

Reading the following sections, you will come to understand how to adopt an attribute weighting scheme, how to apply partial matching and how to define a suitable distance metric.

So, let me finish with posing a research issue: *How can you improve a particular information retrieval model or matching scheme by using a suitable method of knowledge representation and reasoning?* To give you a point of departure, look at the logic-based multimedia information retrieval system proposed in Fuhr et al. (1998).

5.2 images

An image may tell you more than 1000 words. Well, whether images are indeed a more powerful medium of expression is an issue I'd rather leave aside. The problem how to get information out of an image, or more generally how to query image databases is, in the context of our *Amsterdam Drugport* operation more relevant. There are two issues here

- obtaining descriptive information
- establishing similarity

These issues are quite distinct, although descriptive information may be used to establish similarity.

descriptive information

When we want to find, for example, all images that contain a person with say sunglasses, we need to have of the images in our database that includes this information one way or another. One way would be to annotate all images with (meta) information and describe the objects in the picture to some degree of detail. More challenging would be to extract image content by image analysis, and produce the description (semi) automatically.

According to Subrahmanian (1998), content-based description of images involves the identification of objects, as well as an indication of where these objects are located in the image, by using a *shape descriptor* and possibly *property descriptors* indicating the pictorial properties of a particular region of the object or image.

Shape and property descriptors may take a form as indicated below.

shape

- bounding box – (XLB,XUB,YLB,YUB)

property

- property – name=value

As an example of applying these descriptors.

example

> shape descriptor: XLB=10; XUB=60; YLB=3; YUB=50
> property descriptor: pixel(14,7): R=5; G=1; B=3

Now, instead of taking raw pixels as the unit of analysis, we may subdivide an image in a grid of cells and establish properties of cells, by some suitable algorithm.

definitions

- image grid: $(m * n)$ cells of equal size
- cell property: (Name, Value, Method)

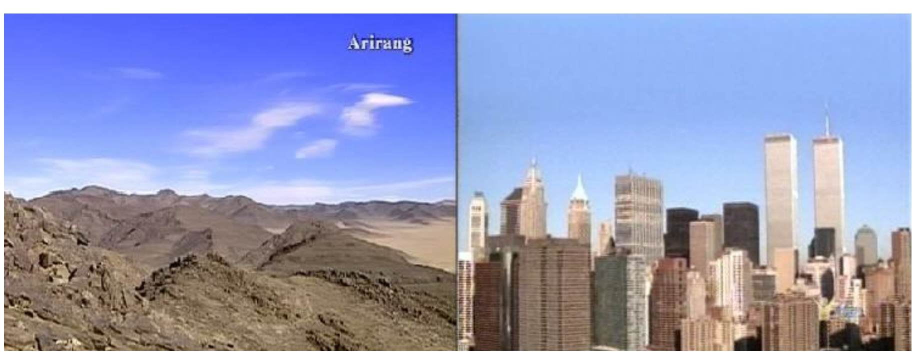
As an example, we can define a property that indicates whether a particular cell if black or white.

example

> property: (bwcolor,{b,w},bwalgo)

The actual algorithm used to establish such a property might be a matter of choice. So, in the example it is given as an explicit parameter.

From here to automatic content description is, admittedly, still a long way. We will indicate some research directions at the end of this section.



3

similarity-based retrieval

We need not necessarily know what an image (or segment of it) depicts to establish whether there are other images that contain that same thing, or something similar to it. We may, following Subrahmanian (1998), formulate the problem of similarity-based retrieval as follows:

> *How do we determine whether the content of a segment (of a segmented image) is similar to another image (or set of images)?*

Think of, for example, the problem of finding all photos that match a particular face.

According to Subrahmanian (1998), there are two solutions:

- *metric approach* – distance between two image objects
- *transformation approach* – relative to specification

As we will see later, the transformation approach in some way subsumes the metric approach, since we can formulate a distance measure for the transformation approach as well.

metric approach What does it mean when we say, the distance between two images is less than the distance between this image and that one. What we want to express is that the first two images (or faces) are more alike, or maybe even identical.

Abstractly, something is a distance measure if it satisfies certain criteria.

metric approach

distance $d : X \rightarrow [0, 1]$ is distance measure if:

$$\begin{aligned} &\mathrm{d(x,y) = d(y,x)} \\ &\mathrm{d(x,y) \leqslant d(x,z) + d(z,y)} \\ &\mathrm{d(x,x) = 0} \end{aligned}$$

For your intuition, it is enough when you limit yourself to what you are familiar with, that is measuring distance in ordinary (Euclidian) space.

Now, in measuring the distance between two images, or segments of images, we may go back to the level of pixels, and establish a distance metric on pixel properties, by comparing all properties pixel-wise and establishing a distance.

pixel properties

- objects with pixel properties $p_1, \ldots, p_n$
- pixels: $(x, y, v1, \ldots, v_n)$
- object contains w x h (n+2)-tuples

Leaving the details for your further research, it is not hard to see that even if the absolute value of a distance has no meaning, relative distances do. So, when an image contains a face with dark sunglasses, it will be closer to (an image of) a face with dark sunglasses than a face without sunglasses, other things being equal. It is also not hard to see that a pixel-wise approach is, computationally, quite complex. An object is considered as

complexity

> a set of points in k-dimensional space for k = n + 2

In other words, to establish similarity between two images (that is, calculate the distance) requires n+2 times the number of pixels comparisons.

feature extraction Obviously,we can do better than that by restricting ourselves to a pre-defined set of properties or features.

feature extraction

- maps object into s-dimensional space

For example, one of the features could indicate whether or not it was a face with dark sunglasses. So, instead of calculating the distance by establishing color differences of between regions of the images where sunglasses may be found, we may limit ourselves to considering a binary value, yes or no, to see whether the face has sunglasses.

Once we have determined a suitable set of features that allow us to establish similarity between images, we no longer need to store the images themselves, and can build an index based on feature vectors only, that is the combined value on the selected properties.

Feature vectors and extensive comparison are not exclusive, and may be combined to get more precise results. Whatever way we choose, when we present an image we may search in our image database and present all those objects that fall within a suitable *similarity range*, that is the images (or segments of images) that are close enough according to the distance metric we have chosen.

4

transformation approach Instead of measuring the distance between two images (objects) directly, we can take one image and start modifying that until it exactly equals the target image. In other words, as phrased in Subrahmanian (1998), the principle underlying the transformation approach is:

transformation approach

> *Given two objects* o1 *and* o2*, the level of* dissimilarity *is proportional to the (minimum) cost of transforming object* o1 *into object* o2 *or vice versa*

Now, this principle might be applied to any representation of an object or image, including feature vectors. Yet, on the level of images, we may think of the following operations:

$to_1, \ldots, to_r$ – translation, rotation, scaling

Moreover, we can attach a cost to each of these operations and calculate the cost of a transformation sequence TSby summing the costs of the individual operations. Based on the cost function we can define a distance metric, which we call for obvious reasons the *edit distance*, to establish similarity between objects.

cost

- $cost(TS) = \Sigma_{i=1}^{r} cost(to_i)$

distance

- d(o,o') = min { cost(TS) | TS in TSeq(o,o') }

An obvious advantage of the *edit distance* over the pixel-wise distance metric is thatwe may have a rich choice of transformation operators that we can attach (user-defined) cost to at will.

For example, we could define low costs for normalization operations, such as scaling and rotation, and attach more weight tooperations that modify color values or add shapes. For face recognition, for example, we could attribute low cost to adding sunglasses but high cost to changing the sex.

To support the *transformation approach* at the image level, our image database needs to include suitable operations. See Subrahmanian (1998).

operations

rotate(image-id,dir,angle)
segment(image-id, predicate)
edit(image-id, edit-op)

We might even think of storing images, not as a collection of pixels, but as a sequence of operations on any one of a given set of base images. This is not such a strange idea as it may seem. For example, to store information about faces we may take a base collection of prototype faces and define an individual face by selecting a suitable prototype and a limited number of operations or additional properties.

5

example(s) – *match of the day*

The images in this section present a *match of the day*, which is is part of the project *split rpresentation* by the Dutch media artist Geert Mul. As explain in the email sending the images, about once a week, *Television images are recorded at random from satellite television and compared with each other. Some 1000.000.000 (one billion) equations are done every day.*

. The *split representation* project uses the image analyses and image composition software *NOTATION*[48], which was developed by Geert Mul (concept) and Carlo Preize (programming ∓ software design).

research directions – *multimedia repositories*

What would be the proper format to store multimedia information? In other words, what is the shape multimedia repositories should take? Some of the issues involved are discussed in chapter , which deals with information system architectures. With respect to image repositories, we may rephrase the question into *what support must an image repository provide, minimally, to allow for efficient access and search?*. In Subrahmanian (1998), we find the following answer:

image repository

- *storage* – unsegmented images
- *description* – limited set of features
- *index* – feature-based index
- *retrieval* – distance between feature vectors

And, indeed, this seems to be what most image databases provide. Note that the actual encoding is not of importance. The same type of information can be encoded using either XML, relational tables or object databases. What is of importance is the functionality that is offered to the user, in terms of storage and retrieval as well as presentation facilities.

[48]homepage.mac.com/geertmul2

What is the relation between presentation facilities and the functionality of multimedia repositories? Consider the following mission statement, which is taken from my research and projects page.

mission

> *Our goal is to study aspects of the deployment and architecture of virtual environments as an interface to (intelligent) multimedia information systems <black>...*

Obviously, the underlying multimedia repository must provide adequate retrieval facilities and must also be able to deliver the desired objects in a format suitable for the representation and possibly incorporation in such an environment. Actually, at this stage, I have only some vague ideas about how to make this vision come through. Look, however, at chapter and appendix for some initial ideas.

6

5.3 documents

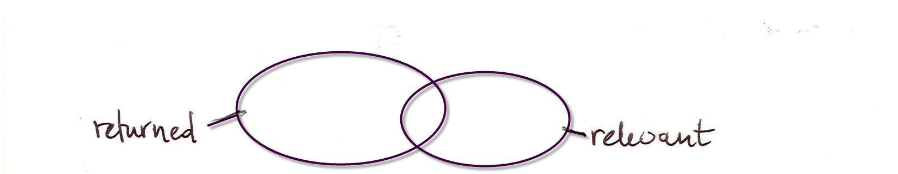
Even in the presence of audiovisual media, text will remain an important vehicel for human communication. In this section, we will look at the issues that arise in querying a text or document database. First we will characterize more precisely what we mean by effective search, and then we will study techniques to realize effective search for document databases.

Basically, answering a query to a document database comes down to string matching. However, some problems may occur such as synonymy and polysemy.

problems

- synonymy – topic T does not occur literally in document D
- polysemy – some words may have many meanings

As an example, *church* and *house of prayer* have more or less the same meaning. So documents about churches and cathedrals should be returned when you ask for information about 'houses of prayer'. As an exampleof polysemy, think of the word *drum*, which has quite a different meaning when taken from a musical perspective than from a transport logistics perspective.



precision and recall

Suppose that, when you pose a query, everything that is in the database is returned. You would probably not be satisfied, although every relevant document will be included, that is for sure. On the other hand, when nothing is returned, at least you cannot complain about non-relevant documents that are returned, or can you?

In Subrahmanian (1998), the notions of *precision* and *recall* are proposed to measure the effectiveness of search over a document database. In general, precision and recall can be defined as follows.

effective search

- precision – how many answers are correct
- recall – how many of the right documents are returned

For your intuition, just imagine that you have a database of documents. With full knowledge of the database you can delineate a set of documentsthatare of relevance to a particular query. Also, you can delineate a set that will be returned by some given search algorithm. Then, *precision* is the intersection of the two sets in relation to whatthe search algorithm returns, and *recall* that same intersection in relation to what is relevant. In pseudo-formulas, we can express this as follows:

precision and recall

precision = (returned and relevant) / returned
recall = (returned and relevant) / relevant

Now, as indicated in the beginning, it is not to difficult to get either perfect recall (by returning all documents) or perfect precision (by returning almost nothing). But these must be considered anomalies (that is, sick cases), and so the problem is to find an algorithm that performs optimally with respect to both precision and recall.

For the total database we can extend these measures by taking the averages of precision and recall for all topics that the database may be queried about.

Can these measures only be applied to document databases? Of course not, these are general measures that can be applied to search over any media type!

frequency tables

A *frequency table* is an example of a way to improve search. Frequency tables, as discussed in Subrahmanian (1998), are useful for documents only. Let's look at an example first.

example

term/document	d0	d1	d2
snacks	1	0	0
drinks	1	0	3
rock-roll	0	1	1

Basically, what a frequency table does is, as the name implies, give a frequency count for particular words or phrases for a number of documents. In effect, a complete document database may be summarized in a frequency table. In other words, the frequency table may be considered as an index to facilitate the search for similar documents.

To find a similar document, we can simply make a word frequency count for the query, and compare that with the colums in the table. As with images, we can apply a simpledistance metric to find the nearest (matching) documents. (In effect, we may take the square root for the sum of the squared differences between the entries in the frequence count as our distance measure.)

The complexity of this algorithm may be characterized as follows:

complextity

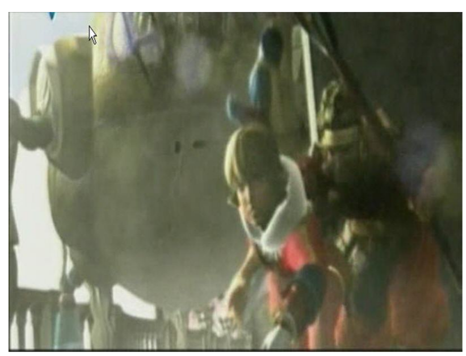
compare term frequencies per document – O(M*N)

where M is the number of terms and N is the number of documents. Since both M and N can become very large we need to make an effort to reduce the size of the frequency table.

reduction

- stop list – irrelevant words
- word stems – reduce different words to relevant part

We can, for example, introduce a *stop list* to prevent irrelevant words to enter the table, and we may restrict ourselves to including *word stems* only, to bring back multiple entries to one canonical form. With some additional effort we could even deal with synonymy and polysemy by introducing, respectively equivalence classes, and alternatives (although we then need a suitable way for ambiguation). By the way, did you notice that frequency tables may be regarded as feature vectors for documents?



7

research directions– *user-oriented measures*

Even though the reductions proposed may result in limiting the size of the frequency tables, we may still be faced with frequency tables of considerable size. One way to reduce the size further, as discussed in Subrahmanian (1998), is to apply *latent sematic indexing* which comes down to clustering the document database, and limiting ourselves to the most relevant words only, where relevance is determined by the ratio of occurrence over the total number of words. In effect, the less the word occurs, the more discriminating it might be. Alternatively,the choice of what words are considered relevant may be determined by taking into account the area of application or the interest of a particular group of users.

8

user-oriented measures Observe that, when evaluating a particular information retrieval system, the notions of precision and recall as introduced before are rather system-oriented measures, based on the assumption of a user-independent notion of relevance. However, as stated in Baeza-Yates and Ribeiro-Neto (1999), different users might have a different interpretation on which document is relevant. In Baeza-Yates and Ribeiro-Neto (1999), some user-oriented measures are briefly discussed, that to some extent cope with this problem.

user-oriented measures

- *coverage ratio* – fraction of known documents
- *novelty ratio* – fraction of new (relevant) documents
- *relative recall* – fraction of expected documents
- *recall effort* – fraction of examined documents

Consider a reference collection, an example information request and a retrieval strategy to be evaluated. Then the *coverage ratio* may be defined as the fraction of the documents known to be relevant, or more precisely the number of (known) relevant documents retrieved divided by the total number of documents known to be relevant by the user.

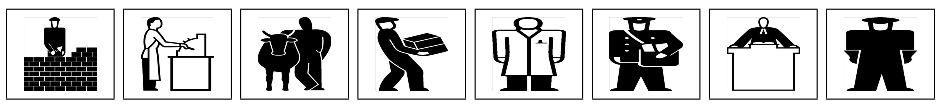
The *novelty ratio* may then be defined as the fraction of the documents retrieved which were not known to be relevant by the user, or more precisely the number of relevent documents that were not known by the user divided by the total number of relevant documents retrieved.

The *relative recall* is obtained by dividing the number of relevant documents found by the number of relevant documents the user expected to be found.

Finally, *recall effort*may be characterized as the ratio of the number of relevant documents expected and the total number of documents that has to be examined to retrieve these documents.

Notice that these measures all have a clearly 'subjective' element, in that, although they may be generalized to a particular group of users, they will very likely not generalize to all groups of users. In effect, this may lead to different retrieval strategies for different categories of users, taking into account levelof expertise and familiarity with the information repository.

5.4 development(s) – tags, labels and descriptors



9

questions

information retrieval

1. (*) What is meant by the *complementarity of authoring and retrieval*? Sketch a possible scenario of (multimedia) information retrieval and indicate how this may be implemented. Discuss the issues that arise in accessing multimedia information and how content annotation may be deployed.

concepts

2. How would you approach *content-based description* of *images*?
3. What is the difference between a *metric* approach and the *transformational* approach to establishing similarity between images?
4. What problems may occur when searching in text or document databases?

technology

5. Give a definition of: *shape descriptor* and *property descriptor*. Give an example of each.
6. How would you define *edit distance*?
7. Characterize the notions *precision* and *recall*.
8. Give an example (with explanation) of a *frequency table*.

projects & further reading As a project, you may implement simples image analysis algorithms that, for example, extract a color histogram, or detect the presence of a horizon-like edge.

You may further explore scenarios for information retrieval in the cultural heritage domain. and compare this with other applications of multimedia information retrieval, for example monitoring in hospitals.

For further reading I suggest to make yourself familiar with common techniques in information retrieval as described in Baeza-Yates and Ribeiro-Neto (1999), and perhaps devote some time to studying image analisis, Gonzales and Wintz (1987).

the artwork

1. artworks – ..., Miro, Dali, photographed from Kunstsammlung Nordrhein-Westfalen, see artwork 2.
2. left Miro from Kunst, right: Karel Appel
3. *match of the day* (1) – Geert Mul
4. *match of the day* (2) – Geert Mul
5. *match of the day* (3) – Geert Mul
6. *mario ware* – taken from gammo/veronica[49].
7. *baten kaitos – eternal ways and the lost ocean*, taken from gammo/veronica.
8. idem.
9. signs – people, van Rooijen (2003), p. 252, 253.

The art opening this chapter belongs to the tradition of 20th century art. It is playful, experimental, with strong existential implications, and it shows an amazing *variety of styles*.

The examples of *match of the day* by Geert Mul serve to illustrate the interplay between *technology and art*, and may also start you to think about what *similarity* is. Some illustrations from games are added to show the difference in styles.

[49] www.gammo.nl

6. content annotation

video annotation requires a logical approach to story telling

learning objectives

After reading this chapter you should be able to explain the difference between content and meta information, to mention relevant content parameters for audio, to characterize the requirements for video libraries, to define an annotation logic for video, and to discuss feature extraction in samples of musical material.

Current technology does not allow us to extract information automatically from arbitrary media objects. In these cases, at least for the time being, we need to assist search by annotating content with what is commonly referred to as meta-information. In this chapter, we will look at two more media types, in particular audio and video. Studying audio, we will learn how we may combine feature extraction and meta-information to define a data model that allows for search. Studying video, on the other hand, will indicate the complexity of devising a knowledge representation scheme that captures the content of video fragments. Concluding this chapter, we will discuss an architecture for feature extraction for arbitrary media objects.

1

6.1 audio

The audio media type covers both spoken voice and musical material. In this section we will discuss audio signal, stored in a raw or compressed (digital) format, as well as similarity-based retrieval for musical patterns.

In general, for providing search access to audio material we need, following Subrahmanian (1998), a data model that allows for both mdeta-data (that is information about the media object) and additional attributes of features, that we in prinicple obtain from the media object itself, using feature extraction.

audio data model

- *meta-data* – describing content
- *features* – using feature extraction

As an example of audi meta-data, consider the (meta-data) characterization that may be given for opera librettos.

example

singers – (Opera,Role,Person)
score – ...
transcript – ...

For signal-based audio content, we have to perform an analysis of the audio signal for which we may take parameters such as frequency, velocity and amplitude. For the actual analysis we may have to break up the signal in small windows, along the time-axis. Using feature extraction, we may characterize (signal-based) properties such as indicated below.

feature extraction

- *intensity* – watts/m^2
- *loudness* – in decibels
- *pitch* – from frequency and amplitude
- *brightness* – amount of distortion

For a more detailed treatment of signal-based audio content description, consult Subrahmanian (1998).

In the following we will first give an overview of musical search facilities on the web and then we will discuss similarity-based retrieval of musical patterns in somewhat more depth in the section on *research directions*, In section 6.3, we will have a closer look at feature extraction for arbitrary media types.

2

musical similarity

Although intuitively obvious, how can we characterize musical simularity? And perhaps more importantly, how can we compute the extent to which one piece of music or a melody line is similar to another piece of music or melody line. As concerns musical content, at least for most genres, it appears that

According to Selfridge (1998), we should focus primarily on *melody*, since

> *"It is melody that makes music memorable: we are likely to recall a tune long after we have forgotten its text."*

Other features, content-based as well as descriptive, may however be used as additional filters in the proces of retrieval.

Melodic searching and matching has been explored mainly in the context of bibliographic tools and for the analysis of (monophonic) repertories Hewlett and Selfridge-Field (1998). As described in section , many of these efforts have been made available to the general public through the Web. Challenges for the near future are, however, to provide for melodic similarity matching on polyphonic works, and retrieval over very large databases of musical fragments.

In this section we will look in somewhat more detail at the problem of melodic similarity matching. In particular, we will discuss representational issues, matching algorithms and additional analysis tools that may be used for musical information retrieval.

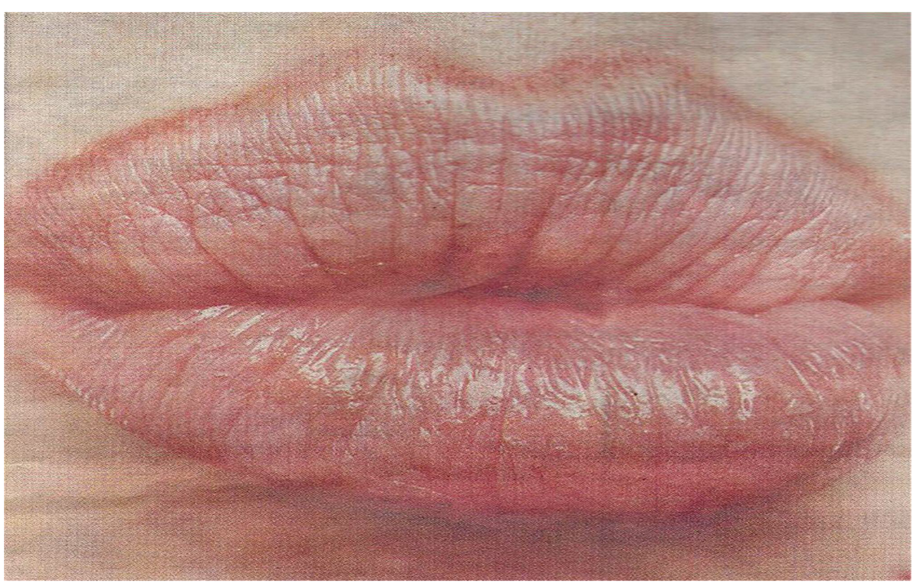
melodic similarity Consider the musical fragment *Twinkle, twinkle little star* (known in the Dutch tradition as *"Altijd is Kortjakje ziek"*), which has been used by Mozart for a series of variations Mozart (1787). Now, imagine how you would approach establishing the similarity between the original theme and these variations. As a matter of fact, we discovered that exactly this problem had been tackled in the study reported in Mongeau and Sankoff (1990), which we will discuss later. Before that, we may reflect on what we mean by the concept of a *melody*. In the aforementioned variations the original melody is disguised by, for example, decorations and accompaniments. In some variations, the melody is distributed among the various parts (the left and right hand). In other variations, the melody is only implied by the harmonic structure. Nevertheless, for the human ear there seems to be, as it is called in Selfridge (1998), a *'prototypical'* melody that is present in each of the variations.

When we restrict ourselves to pitch-based comparisons, melodic similarity may be established by comparing profiles of pitch-direction (up, down, repeat) or pitch contours (which may be depicted graphically). Also, given a suitable representation, we may compare pitch-event strings (assuming a normalized pitch representation such as position within a scale) or intervallic contours (which gives the distance between notes in for example semitones). Following Selfridge (1998), we may observe however that the more general the system of representation, the longer the (query) *string* will need to be to produce meaningful discriminations. As further discussed in Selfridge (1998), recent studies in musical perception indicate that pitch-information without durational values does not suffice.

representational issues Given a set of musical fragments, we may envisage several reductions to arrive at the (hypothetical) prototypical melody. Such reductions must provide for the elimination of confounds such as rests, repeated notes and grace notes, and result in, for example, a pitch-string (in a suitable representation), a duration profile, and (possibly) accented note profiles and harmonic reinforcement profiles (which capture notes that are emphasized by harmonic changes). Unfortunately, as observed in Selfridge (1998), the problem of which reductions to apply is rather elusive, since it depends to a great extent on the goals of the query and the repertory at hand.

As concerns the representation of pitch information, there is a choice between a base-7 representation, which corresponds with the position relative to the tonic in the major or minor scales, a base-12 representation, which corresponds with a division in twelve semitones as in the chromatic scale, and more elaborate encodings, which also reflect notational differences in identical notes that arise through the use of accidentals. For MIDI applications, a base-12 notation is most suitable, since the MIDI note information is given in semitone steps. In addition to relative pitch information, octave information is also important, to establish the rising and falling of melodic contour.

When we restrict ourselves to directional profiles (up, down, repeat), we may include information concerning the slope, or degree of change, the relation of the current pitch to the original pitch, possible repetitions, recurrence of pitches after intervening pitches, and possible segmentations in the melody. In addition, however, to support relevant comparisons it seems important to have information on the rhythmic and harmonic structure as well.



3

example(s) – *napster*

Wasn't it always your dream to have all your music free? Napster[50] was the answer. (But not for long.) Napster is, as we learn in the WikiPedia[51], *an online music service which was originally a file sharing service created by Shawn Fanning. Napster was the first widely-used peer-to-peer music sharing service, and it made a major impact on how people, especially college students, used the Internet. Its technology allowed music fans to easily share MP3 format song files with each other, thus leading to the music industry's accusations of massive copyright violations. The service was named Napster after Fanning's nickname.* However, Napster has been forced to become commercial. So the question is: is there life after napster? Well, there is at least open source!

research directions – *musical similarity matching*

An altogether different approach at establishing melodic similarity is proposed in Mongeau and Sankoff (1990). This approach has been followed in the Meldex system McNab et al. (1997), discussed in section . This is a rather technical section, that may be skipped on first reading. The approach is different in that it relies on a (computer science) theory of finite sequence comparison, instead of musical considerations. The general approach is, as explained in Mongeau and Sankoff (1990), to search for an optimal correspondence between elements of two sequences, based on a distance metric or measure of dissimilarity, also known more informally as the *edit-distance*, which amounts to the (minimal) number of transformations that need to be applied to the first sequence in order to obtain the second one. Typical transformations include *deletion*, *insertion* and *replacement*. In the musical domain, we may also apply transformations such as *consolidation* (the replacement of several elements by one element) and *fragmentation* (which is the reverse of consolidation). The metric is even more generally applicable by associating a weight with each of the transformations. Elements of the musical sequences used in Mongeau and Sankoff (1990) are pitch-duration pairs, encoded in base-12 pitch information and durations as multiples of 1/16th notes.

The matching algorithm can be summarized by the following recurrence relation for the dissimilarity metric. Given two sequences $A = a_1, \ldots, a_m$ and $B = b_1, \ldots, b_n$ and $d_{ij} = d(a_i, b_j)$, we define the distance as

$$d_{ij} = min \begin{cases} d_{i-1,j} + w(a_i, 0) & \text{deletion} \\ d_{i,j-1} + w(0, b_j) & \text{insertion} \\ d_{i-1,j-1} + w(a_i, b_j) & \text{replacement} \\ d_{i-k,j-1} + w(a_{i-k+1}, \ldots, a_i, b_j).\ 2 \leqslant k \leqslant i & \text{consolidation} \\ d\_i - 1, j - k + 1 + w(a\_i, b\_j - k + 1, \ldots b\_j)\ 2 <= k <= j & \\ & \text{fragmentation} \end{cases}$$

with

$$d_{i0} = d_{i-1,0} + w(a_i, 0),\ i \geqslant 1 \qquad \textit{deletion}$$

[50] www.napster.com

[51] en.wikipedia.org/wiki/Napster

$$d_{0j} = d_{0,j-1} + w(0, b_i),\ j \geqslant 1 \qquad \textit{insertion}$$

and $d_{00} = 0$. The weigths $w(\_, \_)$ are determined by the degree of dissonance and the length of the notes involved.

The actual algorithms for determining the dissimilarity between two sequences uses dynamic programming techniques. The algorithm has been generalized to look for matching phrases, or subsequences, within a sequence. The complexity of the algorithm is $O(mn)$, provided that a limit is imposed on the number of notes involved in consolidation and fragmentation.

Nevertheless, as indicated in experiments for the Meldex database, the resulting complexity is still forbidding when large databases are involved. The Meldex system offers apart from the (approximate) dynamic programming algorithm also a state matching algorithm that is less flexible, but significantly faster. The Meldex experiments involved a database of 9400 songs, that were used to investigate six musical search criteria: (1) exact interval and rhythm, (2) exact contour and rhythm, (3) exact interval, (4) exact contour, (5) approximate interval and rhythm, and (6) approximate contour and rhythm. Their results indicate that the number of notes needed to return a reasonable number of songs scales logarithmically with database size McNab et al. (1997). It must be noted that the Meldex database contained a full (monophonic) transcription of the songs. An obvious solution to manage the complexity of searching over a large database would seem to be the storage of prototypical themes or melodies instead of complete songs.

indexing and analysis There are several tools available that may assist us in creating a proper index of musical information. One of these tools is the Humdrum system, which offers facilities for metric and harmonic analysis, that have proven their worth in several musicological investigations Huron (1997). Another tool that seems to be suitable for our purposes, moreover since it uses a simple pitch-duration, or *piano-roll*, encoding of musical material, is the system for metric and harmonic analysis described in Temperley and Sleator (1999). Their system derives a metrical structure, encoded as hierarchical levels of equally spaced beats, based on preference-rules which determine the overall likelihood of the resulting metrical structure. Harmonic analysis further results in (another level of) *chord spans* labelled with roots, which is also determined by preference rules that take into account the previously derived metrical structure. As we have observed before, metrical and harmonic analysis may be used to eliminate confounding information with regard to the 'prototypical' melodic structure.

4

6.2 video

Automatic content description is no doubt much harder for video than for any other media type. Given the current state of the art, it is not realistic to expect content description by feature extraction for video to be feasible. Therefore,to realize content-based search for video, we have rely on some knowledge representation schema that may adequately describe the (dynamic) properties of video fragments.

In fact, the description of video content may reflect the story-board, that after all is intended to capture both time-independent and dynamically changing properties of the objects (and persons) that play a role in the video.

In developing a suitable annotation for a particular video fragment, two questions need to be answered:

video annotation

- what are the interesting aspects?
- how do we represent this information?

Which aspects are of interest is something you have to decide for yourself. Let's see whether we can define a suitable knowledge representation scheme.

One possible knowledge representation scheme for annotating video content is proposed in Subrahmanian (1998). The scheme proposed has been inspired by knowledge representation techniques in Artificial Intelligence. It captures both static and dynamic properties.

video content

video v, frame f
f has associated objects and activities
objects and activities have properties

First of all, we must be able to talk about a particular video fragment v, and frame f that occurs in it. Each frame may contain objects that play a role in some activity. Both objects and activities may have properties, that is attributes that have some value.

property

property: name = value

As we will see in the examples, properties may also be characterized using predicates.

Some properties depend on the actual frame the object is in. Other properties (for example sex and age) are not likely to change and may considered to be frame-independent.

object schema

(fd,fi) – frame-dependent and frame-independent properties

Finally, in order to identify objects we need an object identifier for each object. Summing up, for each object in a video fragment we can define an *object instance*, that characterizes both frame-independent and frame-dependent properties of the object.

object instance: (oid,os,ip)

- *object-id* – oid
- *object-schema* – os = (fd,fi)
- *set of statements* – ip: name = v and name = v IN f

Now, with a collection of object instances we can characterize the contents of an entire video fragment, by identifying the frame-dependent and frame-independent properties of the objects.

Look at the following example, borrowed from Subrahmanian (1998) for the *Amsterdam Drugport* scenario.

frame	objects	*frame-dependent properties*
1	Jane	has(briefcase), at(path)
-	house	door(closed)
-	briefcase	
2	Jane	has(briefcase), at(door)
-	Dennis	at(door)
-	house	door(open)
-	briefcase	

In the first frame Jane is near the house, at the path that leads to the door. The door is closed. In the next frame, the door is open. Jane is at the door, holding a briefcase. Dennis is also at the door. What will happen next?

Observe thatwe are using predicates to represent the state of affairs. We do this, simply because the predicate form *has(briefcase)* looks more natural than the other form, which would be $has = briefcase$. There is no essential difference between the two forms.

Now, to complete our description we can simply list the frame-independent properties, as illustrated below.

object	*frame-independent properties*	value
Jane	age	35
	height	170cm
house	address	...
	color	brown
briefcase	color	black
	size	40 x 31

How to go from the tabular format to sets of statements that comprise the object schemas is left as an (easy) exercise for the student.

Let's go back to our *Amsterdam Drugport* scenario and see what this information might do for us, in finding possible suspects. Based on the information given in the example, we can determine that there is a person with a briefcase, and another person to which that briefcase may possibly be handed. Whether this is the case or not should be disclosed in frame 3. Now, what we are actually looking for is the possible exchange of a briefcase, which may indicate a drug transaction. So why not, following Subrahmanian (1998), introduce another somewhat more abstract level of description that deals with *activities*.

activity

- activity name – id
- statements – $role = v$

An activity has a name, and consists further simply of a set of statements describing the *roles* that take part in the activity.

example

{ giver : Person, receiver : Person, item : Object }
giver = Jane, receiver = Dennis, object = briefcase

For example, an *exchange* activity may be characterized by identifying the *giver*, *receiver* and *object* roles. So, instead of looking for persons and objects in a video fragment, you'd better look for activities that may have taken place, by finding a matching set of objects for the particular roles of an activity. Consult Subrahmanian (1998) if you are interested in a further formalization of these notions.

5

video libraries

Assuming a knowledge representation scheme as the one treated above, how can we support search over a collection of videos or video fragments in a video library.

What we are interested in may roughly be summarized as

video libraries

which videos are in the library
what constitutes the content of each video
what is the location of a particular video

Take note that all the information about the videos or video fragments must be provided as meta-information by a (human) librarian. Just imagine for a moment how laborious and painstaking this must be, and whata relief video feature extraction would be for an operation like *Amsterdam Drugport*.

To query the collection of video fragments, we need a query language with access to our knowledge representation. It must support a variety of retrieval operations, including the retrieval of segments, objects and activities, and also property-based retrievals as indicated below.

query language for video libraries

- *segment retrievals – exchange of briefcase*
- *object retrievals* – all people in v:[s,e]
- *activity retrieval* – all activities in v:[s,e]
- *property-based* – find all videos with object oid

Subrahmanian (1998) lists a collection of video functions that may be used to extend SQL into what we may call VideoSQL. Abstractly, VideoSQL may be characterized by the following schema:

VideoSQL

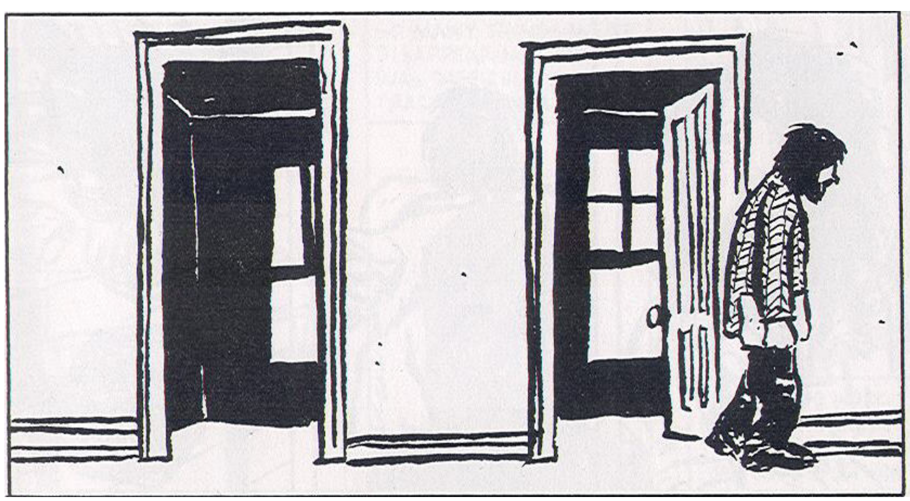
SELECT – v:[s,e]
FROM – video:<source><V>
WHERE – term IN funcall

where *v:[s,e]* denotes the fragment of video *v*, starting at frame *s* and ending at frame *e*, and *term IN funcall* one of the video functions giving access to the information about that particular video. As an example, look at the following VideoSQL snippet:

example

SELECT vid:[s,e]
FROM video:VidLib
WHERE (vid,s,e) IN VideoWithObject(Dennis) AND
    object IN ObjectsInVideo(vid,s,e) AND
    object != Dennis AND
    typeof(object) = Person

Notice that apart from calling video functions also constraints can be added with respect to the identity and type of the objects involved.



6

example(s) – *video retrieval evaluation*

The goal of the TREC[52] conference series is to encourage research in information retrieval by providing a large test collection, uniform scoring procedures, and a forum for organizations interested in comparing their results. Since 2003 their is an independent *video* track devoted to research in automatic segmentation,

[52]trec.nist.gov

indexing, and content-based retrieval of digital video. In the TRECVID[53] 2004 workshop, thirty-three teams from Europe, the Americas, Asia, and Australia participated. Check it out!

7

research directions– *presentation and context*

Let's consider an example. Suppose you have a database with (video) fragments of news and documentary items. How would you give access to that database? And, how would you present its contents? Naturally, to answer the first question, you need to provide search facilities. Now, with regard to the second question, for a small database, of say 100 items, you could present a list of videos thatb matches the query. But with a database of over 10.000 items this will become problematic, not to speak about databases with over a million of video fragments. For large databases, obviously, you need some way of visualizing the results, so that the user can quickly browse through the candidate set(s) of items.

Pesce (2003) provide an interesting account on how *interactive maps* may be used to improve search and discovery in a (digital) video library. As they explain in the abstract:

> *To improve library access, the Informedia Digital Video Library uses automatic processing to derive descriptors for video. A new extension to the video processing extracts geographic references from these descriptors.*
>
> *The operational library interface shows the geographic entities addressed in a story, highlighting the regions discussed in the video through a map display synchronized with the video display.*

[53]www-nlpir.nist.gov/projects/trecvid

So, the idea is to use geographical information (that is somehow available in the video fragments themselves) as an additional descriptor, and to use that information to enhance the presentation of a particular video. For presenting the results of a query, candidate items may be displayed as icons in a particular region on a map, so that the user can make a choice.

Obviously, having such geographical information:

> *The map can also serve as a query mechanism, allowing users to search the terabyte library for stories taking place in a selected area of interest.*

The approach to extracting descriptors for video fragments is interesting in itself. The two primary sources of information are, respectively, the spoken text and graphic text overlays (which are common in news items to emphasize particular aspects of the news, such as the area where an accident occurs). Both speech recognition and image processing are needed to extract information terms, and in addition natural language processing, to do the actual 'geocoding', that is translating this information to geographical locations related to the story in the video.

Leaving technical details aside, it will be evident that this approach works since news items may relevantly be grouped and accessed from a geographical perspective. For this type of information we may search, in other words, with three kinds of questions:

- *what* – content-related
- *when* – position on time-continuum
- *where* – geographic location

and we may, evidently, use the geographic location both as a search criterium and to enhance the presentation of query results.

mapping information spaces Now, can we generalize this approach to other type of items as well. More specifically, can we use maps or some spatial layout to display the results of a query in a meaningful way and so give better access to large databases of multimedia objects. According to Dodge and Kitchin (2002), we are very likely able to do so:

> *More recently, it has been recognized that the process of spatialization – where a spatial map-like structure is applied to data where no inherent or obvious one does exist – can provide an interpretable structure to other types of data.*

Actually, we are taking up the theme of *visualization*, again. In Dodge and Kitchin (2002) visualizations are presented that (together) may be regarded as an *atlas of cyberspace.*

atlas of cyberspace

> *We present a wide range of spatializations that have employed a variety of graphical techniques and visual metaphors so as to provide striking and powerful images that extend from two dimension 'maps' to three-dimensional immersive landscapes.*

As you may gather from chapter 7 and the *afterthoughts*, I take a personal interest in the (research) theme of *virtual reality interfaces for multimedia information systems*. But I am well aware of the difficulties involved. It is an area that is just beginning to be explored!

8

6.3 feature extraction

Manual content annotation is laborious, and hence costly. As a consequence, content annotation will often not be done and search access to multimedia object willnot be optimal, if it is provided for at all. An alternative to manual content annotation is (semi) automatic feature extraction, which allows for obtaining a description of a particular media object using media specific analysis techniques.

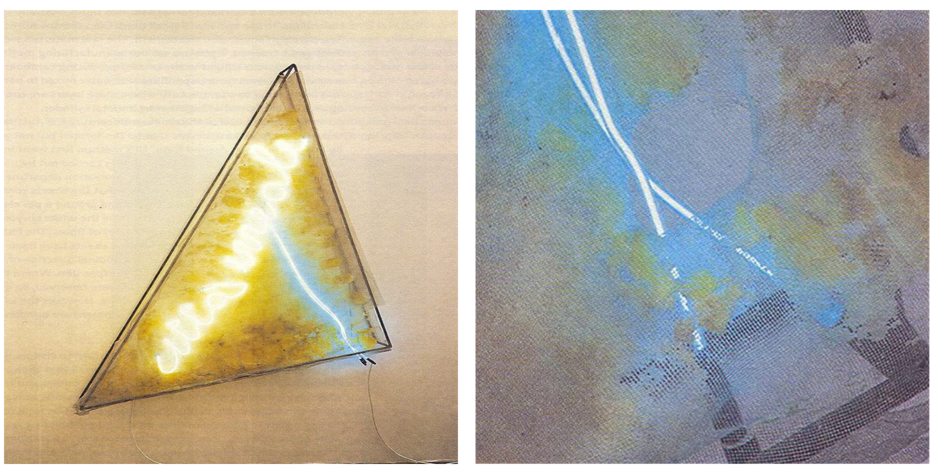
The Multimedia Database Research group at CWI has developed a framework for feature extraction to support the *Amsterdam Catalogue of Images* (ACOI). The resulting framework for feature extraction is known as the ACOI framework, Kersten et al. (1998).

The ACOI framework is intended to accomodate a broad spectrum of classification schemes, manual as well as (semi) automatic, for the indexing and retrieval of arbitrary multimedia objects. What is stored are not the actual multimedia objects themselves, but structural descriptions of these objects (including their location) that may be used for retrieval.

The ACOI model is based on the assumption that indexing an arbitrary multimedia object is equivalent to deriving a grammatical structure that provides a namespace to reason about the object and to access its components. However there is an important difference with ordinary parsing in that the lexical and grammatical items corresponding to the components of the multimedia object must be created dynamically by inspecting the actual object. Moreover, in general, there is not a fixed sequence of lexicals as in the case of natural or formal languages. To allow for the dynamic creation of lexical and grammatical items the ACOI framework supports both *black-box* and *white-box* (feature) detectors. Black-box detectors are algorithms, usually developed by a specialist in the media domain, that extract properties from the media object by some form of analysis. White-box detectors, on the other hand, are created by defining logical or mathematical expressions over the grammar itself. Here we will focus on black-box detectors only.

The information obtained from parsing a multimedia object is stored in a database. The feature grammar and its associated detector further result in updating the data schemas stored in the database.

formal specification Formally, a feature grammar G may be defined as $G = (V, T, P, S)$, where V is a collection of variables or non-terminals, T a collection of terminals, P a collection of productions of the form $V \rightarrow (V \cup T)$ and S a start symbol. A token sequence ts belongs to the language $L(G)$ if $S \stackrel{*}{\longrightarrow} ts$. Sentential token sequences, those belonging to $L(G)$ or its sublanguages $L(G_v) = (V_v, T_v, P_v, v)$ for $v \in (T \cup V)$, correspond to a complex object C_v, which is the object corresponding to the parse tree for v. The parse tree defines a hierarchical structure that may be used to access and manipulate the components of the multimedia object subjected to the detector. See Schmidt et al. (1999) for further details.



9

anatomy of a feature detector

As an example of a feature detector, we will look at a simple feature detector for (MIDI encoded) musical data. A special feature of this particular detector,that I developed while being a guest at CWI, is that it uses an intermediate representation in a logic programming language (Prolog) to facilitate reasoning about features.

The hierarchical information structure that we consider is defined in the grammar below. It contains only a limited number of basic properties and must be extended with information along the lines of some musical ontology, see Zimmerman (1998).

feature grammar

```
detector song; ## to get the filename
detector lyrics; ## extracts lyrics
detector melody; ## extracts melody
detector check;  ## to walk the tree

atom str name;
atom str text;
atom str note;

midi: song;

song: file lyrics melody check;

file: name;

lyrics: text*;
melody: note*;
```

The start symbol is a *song*. The detector that is associated with *song* reads in a MIDI file. The musical information contained in the MIDI file is then stored as a collection of Prolog facts. This translation is very direct. In effect the MIDI file header information is stored, and events are recorded as facts, as illustrated below for a *note_on* and *note_off* event.

```
event('twinkle',2,time=384, note_on:[chan=2,pitch=72,vol=111]).
event('twinkle',2,time=768, note_off:[chan=2,pitch=72,vol=100]).
```

After translating the MIDI file into a Prolog format, the other detectors will be invoked, that is the *composer*, *lyrics* and *melody* detector, to extract the information related to these properties.

To extract relevant fragments of the melody we use the melody detector, of which a partial listing is given below.

melody detector

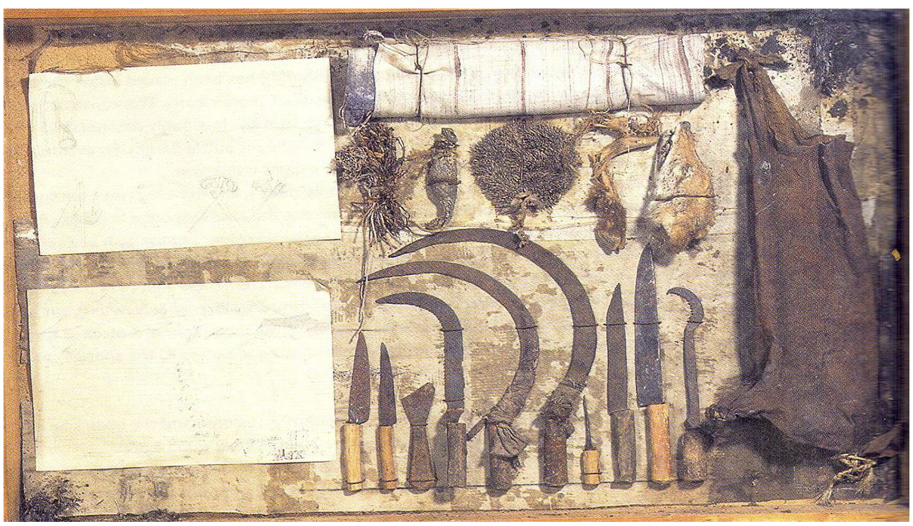
```
int melodyDetector(tree *pt, list *tks ){
char buf[1024]; char* _result;
void* q = _query;
int idq = 0;

  idq = query_eval(q,"X:melody(X)");
  while ((_result = query_result(q,idq)) ) {
        putAtom(tks,"note",_result);
        }
```

```
  return SUCCESS;
}
```

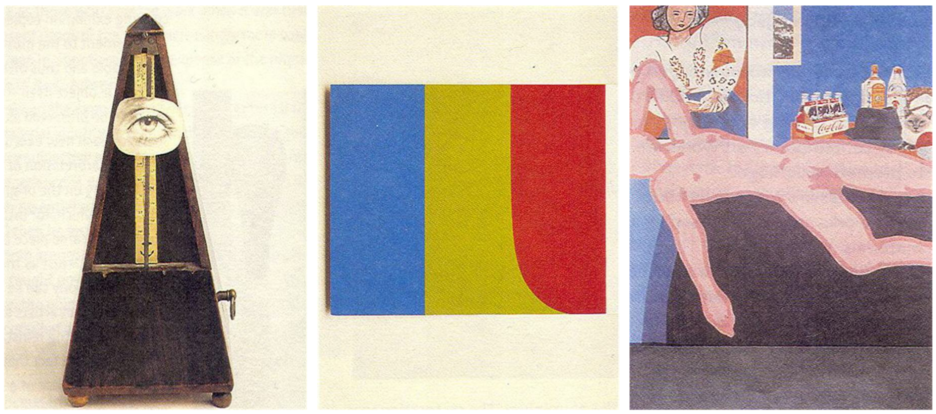
The embedded logic component is given the query `X:melody(X)`, which results in the notes that constitute the (relevant fragment of the) melody. These notes are then added to the tokenstream. A similar detector is available for the lyrics.

Parsing a given MIDI file, for example *twinkle.mid*, results in updating the database.



10

implementation The embedded logic component is part of the *hush* framework, Eliens (2000). It uses an object extension of Prolog that allows for the definition of native objects to interface with the MIDI processing software written in C++. The logic component allows for the definition of arbitrary predicates to extract the musical information, such as the melody and the lyrics. It also allows for further analysis of these features to check for, for example, particular patterns in the melody.



11

example(s) – *modern art: who cares?*

The artworks shown above are taken from Hummelen and Sillé (1999), which bundles the experiences and insights resulting from studying the preservation of contemprary art, under the title: *modern art, who cares?* This project was a precursor to the INCCA[54] that provided the input to our *multimedia casus*, which is introduced in chapter 10.

Both the INCCA project and the related *Open Archives Initiative*[55], focus on making meta-information available on existing resources for the preservation of contemporary art and cultural heritage in general, including reports, case studies and recordings of artworks, that is images, videos and artists interviews.

12

research directions– *media search*

There is a wealth of powerful search engines on the Web. Technically, search engines rely either on classification schemes (as for example Yahoo) or content-based (keyword) indexing (as for example Excite or AltaVista). Searching on the Web, nowadays, is moderately effective when text-based documents are considered. For multimedia objects (such as images or music) existing search facilities are far less effective, simply because indexing on category or keywords can not so easily be done automatically. In the following we will explore what search facilities there are for music (on the web). We will first give some examples of search based on keywords and categories, then some examples of content-based search and finally we will discuss a more exhaustive list of musical databases and search facilities on the Web. All search facilities mentioned are listed online under *musical resources*.

[54]www.incca.org
[55]www.openarchives.org

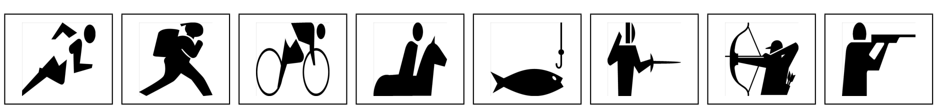
keywords and categories For musical material, in particular MIDI, there are a number of sites that offer search over a body of collected works. One example is the Aria Database, that allows to search for an aria part of an opera based on title, category and even voice part. Another example is the MIDI Farm, which provides many MIDI-related resources, and also allows for searching for MIDI material by filename, author, artist and ratings. A category can be selected to limit the search. The MIDI Farm employs voting to achieve collaborative filtering on the results for a query. Search indexes for sites based on categories and keywords are usually created by hand, sometimes erreonously. For example, when searching for a Twinkle fragment, Bach's variations for Twinkle were found, whereas to the best of our knowledge there exist only Twinkle variations by Mozart Mozart (1787). The Digital Tradition Folksong Database provides in addition a powerful lyrics (free text) search facility based on the AskSam search engine. An alternative way of searching is to employ a meta-search engine. Meta-search engines assist the user in formulating an appropriate query, while leaving the actual search to (possibly multiple) search engines. Searching for musical content is generally restricted to the lyrics, but see below (and section).

content-based search Although content-based search for images and sound have been a topic of interest for over a decade, few results have been made available to the public. As an example, the MuscleFish Datablade for Informix, allows for obtaining information from audio based on a content analysis of the audio object. As far as content-based musical search facilities for the Web are concerned, we have for example, the Meldex system of the New Zealand Digital Library initiative, an experimental system that allows for searching tunes in a folksong database with approximately 1000 records, McNab et al. (1997). Querying facilities for Meldex include queries based on transcriptions from audio input, that is humming a tune! We will discuss the approach taken for the Meldex system in more detail in research directions section, to assess its viability for retrieving musical fragments in a large database.

music databases In addition to the sites previously mentioned, there exist several databases with musical information on the Web. We observe that these databases do not rely on DBMS technology at all. This obviously leads to a plethora of file formats and re-invention of typical DBMS facilities. Without aiming for completeness, we have for example the *MIDI Universe*, which offers over a million MIDI file references, indexed primarily by composer and file length. It moreover keeps relevant statistics on popular tunes, as well as a hot set of MIDI tunes. It further offers access to a list of related smaller MIDI databases. Another example is the aforementioned Meldex system that offers a large collection of tunes (more than 100.000), of which a part is accessible by humming-based retrieval. In addition text-based search is possible against file names, song titles, track names and (where available) lyrics. The Classical MIDI Archive is an example of a database allowing text-based search on titles only. Results are annotated with an indication of "goodness" and recency. The Classical Themefinder Database allows extensive support for retrieval based on (optional) indications of meter, pitch, pitch-class, interval, semi-tone interval and melodic contour, within a fixed collection of works arranged according to composer and category. The index is clearly created and maintained manually. The resulting work is delivered in the MuseData format, which is a rich (research-based) file format from which MIDI files can be generated, Selfridge (1997). A site which collects librarian information concerning music resources is the International Inventory of Music Resources (RISM), which offers search facilities over bibliographic records for music manuscripts, librettos and secondary sources for music written after c.a. 1600. It also allows to search for libraries related to the RISM site. Tune recognition is apparently offered by the Tune Server. The user may search by offering a WAV file with a fragment of the melody. However, the actual matching occurs against a melodic outline, that is indications of rising or falling in pitch. The database contains approx. 15.000 records with such pitch contours, of which one third are popular tunes and the rest classical themes. The output is a ranked list of titles about which the user is asked to give feedback.

discussion There is great divergence in the scope and aims of music databases on the Web. Some, such as the RISM database, are the result of musicological investigations, whereas others, such as the MIDI Farm, are meant to serve an audience looking for popular tunes. With regard to the actual search facilities offered, we observe that, with the exception of Meldex and the Tune Server, the query facilities are usually text-based, although for example the Classical Themefinder allows for encoding melodic contour in a text-based fashion.

6.4 development(s) – expert recommendations



13

questions

content annotation

1. (*) How can video information be made accessible? Discuss the requirements for supporting video queries.

concepts

2. What are the ingredients of an *audio data model*
3. What information must be stored to enable search for video content?
4. What is *feature extraction*? Indicate how feature extraction can be deployed for arbitrary media formats.

technology

5. What are the parameters for *signal-based (audio) content*?
6. Give an example of the representation of *frame-dependent* en *frame-independent* properties of a video fragment.
7. What are the elements of a query language for searching in video libraries?
8. Give an example (with explanation) of the use of *VideoSQL*.

projects & further reading As a project, think of implementing musical similarity matching, or developing an application retrieving video fragments using a simple annotation logic.

You may further explore the construction of media repositories, and finding a balance between automatic indexing, content search and meta information.

For further reading I advice you to *google* recent research on video analysis, and the online material on search engines[56].

the artwork

1. works from Weishar (1998)
2. faces – from www.alterfin.org, an interesting site with many surprising interactive toys in *flash*, javascript and html.
3. mouth – Annika Karlson Rixon, entitled *A slight Acquaintance*, taken from a theme article about the body in art and science, the Volkskrant, 24/03/05.
4. story – page from the comic book version of *City of Glass*, Auster (2004), drawn in an almost tradional style.
5. story – frame from Auster (2004).
6. story – frame from Auster (2004).
7. story – frame from Auster (2004).
8. *white on white* – typographical joke.
9. modern art – *city of light* (1968-69), Mario Merz, taken from Hummelen and Sillé (1999).
10. modern art – *Marocco* (1972), Krijn Griezen, taken from Hummelen and Sillé (1999).
11. modern art – *Indestructable Object* (1958), Man Ray, *Blue, Green, Red I* (1964-65), Ellsworth Kelly, *Great American Nude* (1960), T. Wesselman, taken from Hummelen and Sillé (1999).
12. signs – sports, van Rooijen (2003), p. 272, 273.

Opening this chapter are examples of design of the 20th century, posters to announce a public event like a theatre play, a world fair, or a festival. In comparison to the art works of the previous chapter, these designs are more strongly *expressive* and more simple and clear in their *message*. Yet, they also show a wide variety of styles and rethorics to attract the attention of the audience. Both the faces and the mouth are examples of using body parts in contemporary

[56] www.searchtools.com/tools/tools-opensource.html

art. The page of the comic book version of *City of Glass*, illustrates how the 'logic' of a story can be visualised. As an exercise, try to annoyaye the sequence of frames from the *City of Glass* can be described using the annotation logic you learned in this chapter. The modern art examples should interesting by themselves.

7. information system architecture

effective retrieval requires visual interfaces

learning objectives

After reading this chapter you should be able to dicuss the considerations that play a role in developing a multimedia information system, characterize an abstract multimedia data format, give examples of multimedia content queries, define the notion of virtual resources, and discuss the requirements for networked virtual environments.

From a system development perspective, a multimedia information system may be considered as a multimedia database, providing storage and retrieval facilities for media objects. Yet, rather than a solution this presents us with a problem, since there are many options to provide such storage facilities and equally many to support retrieval. In this chapter, we will study the architectural issues involved in developing multimedia information systems, and we will introduce the notion of media abstraction to provide for a uniform approach to arbitrary media objects. Finally, we will discuss the additional problems that networked multimedia confront us with.

1

7.1 architectural issues

The notion of *multimedia information system* is sufficiently generic to allow for a variety of realizations. Let's have a look at the issues involved.

As concerns the database (that is the storage and rerieval facilities), we may have to deal with homegrown solution, commercial third party databases or (even) legacy sources. To make things worse, we will usually want to deploy a combination of these.

With respect to the information architecture, we may wish for a common format (which unifies the various media types), but in practice we will often have to work with the native formats or be satisfied with a hybrid information architecture that uses both media abstractions and native media types such as images and video.

The notion of media abstraction, introduced in Subrahmanian (1998), allows for uniform indexes over the multimedia information stored, and (as we will discuss in the next section) for query relaxation by employing hierarchical and equivalence relations.

Summarizing, for content organisation (which basically is the information architecture) we have the following options:

content organisation

- *autonomy* – index per media type
- *uniformity* – unified index
- *hybrid* – media indexes + unified index

In Subrahmanian (1998), a clear preference is stated for a uniform approach, as expressed in the *Principle of Uniformity*:

Principle of Uniformity

> *... from a semantical point of view the* content *of a multimedia source is independent of the source itself, so we may use statements as meta data to provide a description of media objects.*

Naturally, there are some tradeoffs. In summary, Subrahmanian (1998) claims that: metadata can be stored using standard relational and OO structures, and that manipulating metadata is easy, and moreover that feature extraction is straightforward. </lp> Now consider, is feature extraction really so straightforward as suggested here? I would believe not. Certainly, media types can be processed and analysis algorithms can be executed. But will this result in meaningful annotations? Given the current state of the art, hardly so!

research directions– *the information retrieval cycle*

When considering an information system, we may proceed from a simple generic software architecture, consisting of:

software architecture

- a database of media object, supporting
- operations on media objects, and offering
- logical views on media objects

However, such a database-centered notion of information system seems not to do justice to the actual support and information system must provide when considering the full information retrieval cycle:

information retrieval cycle

1. specification of the user's information need
2. translation into query operations
3. search and retrieval of media objects
4. ranking according to likelihood or relevance
5. presentation of results and user feedback
6. resulting in a possibly modified query

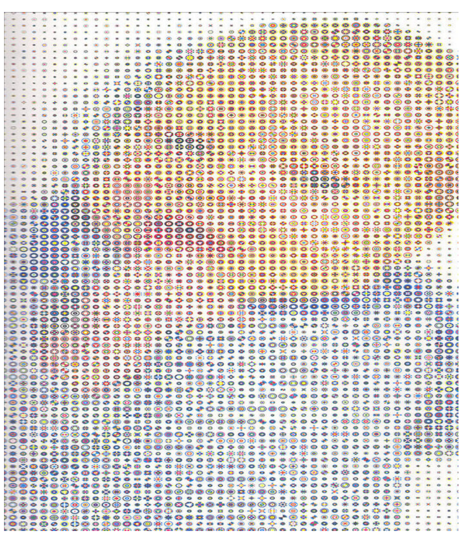
When we look at older day information retrieval applications in libraries, we see more or less the automation of card catalogs, with search functionality for keywords and headings. Modern day versions of these systems, however, offer graphical userinterfaces, electronic forms and hypertext features.

When we look at the web and how it may support digital libraries, we see some dramatic changes with respect to the card catalogue type of applications. We can now have access to a variety of sources of information, at low cost, including geographically distributed resources, due to improved networking. And, everybody is free to make information available, and what is worse, everybody seems to be doing so. Hence, the web is a continuously growing repository of information of a (very) heterogeneous kind.

Considering the web as an information retrieval system we may observe, following Baeza-Yates and Ribeiro-Neto (1999), that:

- despite high interactivity, access is difficult;
- quick response is and will remain important!

So, we need better (user-centered) retrieval strategies to support the full information retrieval cycle. Let me (again) mention someof the relevant (research) topics:*user interfaces, information visualisation, user-profiling and navigation.*



2

7.2 media abstractions

Let's have a closer look at media abstractions. How can we capture the characterization of a variety of media types in one common media abstraction. A definition of such a media abstraction is proposedin Subrahmanian (1998). Leaving the formal details aside, a media abstraction has the following components:

media abstraction

- *state* – smallest chunk of media data
- *feature* – any object in a state
- *attributes* – characteristics of objects
- *feature extraction map* – to identify content
- relations – to capture state-dependent information
- (inter)relations between 'states' or chunks

Now, that characterization is sufficiently abstract, and you may wonder how on earth to apply this to an actual media database.

However, before giving some examples, we must note that the *feature extraction map* does not need to provide information about the content of a chunk of media data automatically. It may well be a hand-coded annotation.

Our first example is an image database.

example – image database

states: { pic1.gif,...,picn.gif }
features: names of people
extraction: find people in pictures
relations: left-of, ...

In an image database it does not make much sense to speak about relations between 'states' or chunks of media data, that is the images.

For our next example though, video databases, it does make sense to speak about such relations, since it allows us to talk about scenes as sequences of frames.

example – video database

states: set of frames
features: persons and objects
extraction: gives features per frame
relations: frame-dependent and frame-independent information
inter-state relation: specifies sequences of frames

Now, with this definition of media abstractions, we can define a simple multimedia database, simply as

simple multimedia database

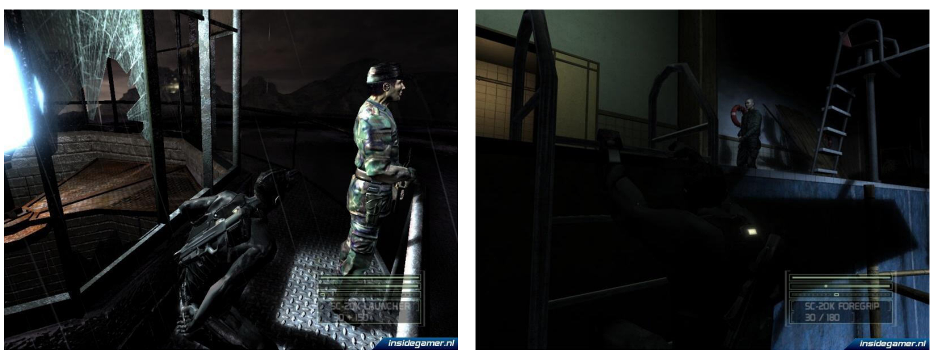
- a finite set M of media abstractions

But, following Subrahmanian (1998), we can do better than that. In order to deal with the problems of *synonymy* and *inheritance*, we can define a structured multimedia database that supports:

structured multimedia database

- *equivalence relations* –to deal with synonymy
- *partial ordering* – to deal with inheritance
- *query relaxation* – to please the user

Recall that we have discussed the relation between a 'house of prayer' and 'church' as an example of synonymy in section 4.3. As an example of inheritance we may think of the relation between 'church' and 'cathedral'. Naturally, every cathedral is a church. But the reverse does not necessarily hold. Having this information about possible equivalence and inheritance relationships, we can relax queries in order to obtain better results. For example, when a user asks for cathedral in a particular region, we could even notify the user of the fact that although there are no cathedrals there, there are a number of churches that may be of interest. (For a mathemathical characterization of structured multimedia databases, study Subrahmanian (1998).)



3

query languages Having media abstractions, what would a query language for such a database look like? Again, following Subrahmanian (1998), wemay extend SQL with special functions as indicated below:

SMDS – functions

Type: object $\mapsto$ type
ObjectWithFeatures: $f \mapsto \{o \mid \text{object } o \text{ contains } f\}$
ObjectWithFeaturesAndAttributes: $(f, a, v) \mapsto \{o \mid o \text{ contains } f \text{ with } a = v\}$
FeaturesInObject: $o \mapsto \{f \mid o \text{ contains } f\}$
FeaturesAndAttributesInObject: $o \mapsto \{(f, a, v) \mid o \text{ contains } f \text{ with } a = v\}$

Having such functionswe can characterize an extension of SQL, which has been dubbed SMDS-SQL in Subrahmanian (1998), as follows.

SMDS-SQL

SELECT – media entities

- m – if m is not a continuous media object

- $m : [i, j]$ – m is continuous, i, j integers (segments)
- $m.a$ – m is media entity, a is attribute

FROM

- <media><source><M>

WHERE

- term IN funcall

As an example, look at the following SMDS-SQL snippet.

example

```
SELECT M
FROM   smds source1 M
WHERE  Type(M) = Image AND
       M IN ObjectWithFeature("Dennis") AND
       M IN ObjectWithFeature("Jane") AND
       left("Jane","Dennis",M)
```

Note that *M* is a relation in the image database media abstraction, which contains one or more images that depict Jane to the left of Dennis. Now, did they exchange the briefcase, or did they not?

When we do not have a uniform representation, but a hybrid representation for our multimedia data instead, we need to be able to: express queries in specialized language, and to perform operations (joins) between SMDS and non-SMDS data.

Our variant of SQL, dubbed HM-SQL, differs from SMDS-SQL in two respects: function calls are annotated with media source, and queries to non-SMDS data may be embedded.

As a final example, look at the following snippet:

example HM-SQL

```
SELECT M
FROM smds video1, videodb video2
WHERE M IN smds:ObjectWithFeature("Dennis") AND
      M IN videodb:VideoWithObject("Dennis")
```

In this example, we are collecting all video fragments with Dennis in it, irrespective of where that fragment comes from, an (smds) database or another (video) database.

research directions– *digital libraries*

Where *media abstractions*, as discussed above, aremeant to be technical abstractions needed for uniform access to media items, we need quite a different set of abstraction to cope with one of the major applications of multimedia information storage and retrieval: digital libraries.

According to Baeza-Yates and Ribeiro-Neto (1999), digital libraries will need a long time to evolve, not only because there are many technical hurdles to be overcome, but also because effective digital libraries are dependent on an active community of users:

digital libraries

> *Digital libraries are constructed – collected and organized – by a community of users. Their functional capabilities support the information needs and users of this community. Digital libraries are an extension, enhancement and integration of a variety of information institutions as physicalplaces where resources are selected, collected, organized, preserved and accessed in support of a user community.*

The occurrence of digital libraries on the web is partly a response to advances in technology, and partly due to an increased appreciation of the facilities the internet can provide. From a development perspective, digital libraries may be regarded as:

> *... federated structures that provide humans both intellectual and physical access to the huge and growing worldwide networks of information encoded in multimedia digital formats.*

Early research in digital libraries has focussed on the digitization of existing material, for the preservation of our cultural heritage, as well as on architectural issues for the 'electronic preservation', so to speak, of digital libraries themselves, to make them "immune to degradation and technological obsolescence", Baeza-Yates and Ribeiro-Neto (1999).

To bring order in the variety of research issues related to digital libraries, Baeza-Yates and Ribeiro-Neto (1999) introduces a set of abstractions that is know as the 5S model:

digital libraries (5S)

- *streams*: (content) – from text to multimedia content
- *structures*: (data) – from database to hypertext networks
- *spaces*: (information) – from vector space to virtual reality
- *scenarios*: (procedures) – from service to stories
- *societies*: (stakeholders) – from authors to libraries

These abstractions act as "a framework for providing theoretical and practical unification of digital libraries". More concretely, observe that the framework encompasses three technical notions (streams, structures and spaces; which correspond more or less with data, content and information) and two notions related to the social context of digital libraries (scenarios and societies; which range over possible uses and users, respectively).

For further research you may look at the following resources:

D-Lib Forum – http://*www.dlib.org*
Informedia – http://*www.informedia.cs.cmu.edu*

The D-Lib Forum site gives access to a variety of resources, including a magazine with background articles as well as a test-suite that may help you in developing digital library technology. The Informedia site provides an example of a digital library project, with research on, among others, video content analysis, summarization and in-context result presentation.

4

7.3 networked multimedia

For the end user there should not be much difference between a stand-alone media presentation and a networked media presentation. But what goes on *behind the scenes* will be totally different. In this section, we will study, or rather have a glance at, the issues that play a role in realizing effective multimedia presentations. These issues concern the management of resources by the underlying network infrastructure, but may also concern authoring to the extent that the choice of which media objects to present may affect the demands on resources.

To begin, let's try to establish, following Fluckiger (1995), in what sense networked multimedia applications might differ from other network applications:

networked multimedia

- real-time transmission of continuous media information (audio, video)
- substantial volumes of data (despite compression)
- distribution-oriented – e.g. audio/video broadcast

Naturally, the extent to which network resource demands are made depends heavily on the application at hand. But as an example, you might think of the retransmission of television news items on demand, as nowadays provided via both cable and DSL.

For any network to satisfy such demands, a number of criteria must be met, that may be summarized as: throughput, in terms of bitrates and burstiness; transmission delay, including signal propagation time; delay variation, also known as jitter; and error rate, that is data alteration and loss.

For a detailed discussion of criteria, consult Fluckiger (1995), or any other book on networks and distributed systems. With respect to distribution-oriented multimedia, that is audio and video broadcasts, two additional criteria play a role, in particular: multicasting and broadcasting capabilities and document caching. Especially caching strategies are of utmost importance if large volumes of data need to be (re)transmitted.

Now, how do we guarantee that our (networked) multimedia presentations will come across with the right quality, that is free of annoying jitter, without loss or distortion, without long periods of waiting. For this, the somewhat magical notion of *Quality of Service* has been invented. Quoting Fluckiger (1995):

Quality of Service

> Quality of Service *is a concept based on the statement that not all applications need the same performance from the network over which they run. Thus, applications may indicate their specific requirements to the network, before they actually start transmitting information data.*

Quality of Service (*QoS*) is one of these notions that gets delegated to the other parties, all the time. For example, in the MPEG-4 standard proposal interfaces are provided to determine *QoS* parameters, but the actual realization of it is left to the network providers. According to Fluckiger (1995) it is not entirely clear how *QoS* requirements should be interpreted. We have the following options: we might consider them as hard requirements, or alternatively as guidance for optimizing internal resources, or even more simply as criteria for the acceptance of a request.

At present, one thing is certain. The current web does not offer *Quality of Service*. And what is worse, presentation formats (such as for example *flash*) do not cope well with the variability of resources. More specifically, you may get quite different results when you switch to another display platform

5

virtual objects

Ideally, it should not make any difference to the author at what display platform a presentation is viewed, nor should the author have to worry about low-quality or ill-functioning networks. In practice, however, it seems not to be realistic to hide all this variability from the author and delegate it entirely to the 'lower layers' as in the MPEG-4 proposal.

Both in the SMIL and RM3D standards, provisions are made for the author to provide a range of options from which one will be chosen, dependent on for example availability, platform characteristics, and network capabilities.

A formal characterization of such an approach is given in Subrahmanian (1998), by defining *virtual objects*.

virtual objects

- $VO = \{(O_i, Q_i, C_i) \mid 1 \leqslant i \leqslant k\}$

where

- $C_1, \ldots, C_k$ – mutually exclusive conditions
- $Q_1, \ldots, Q_k$ – queries
- $O_1, \ldots, O_k$ – objects

In general, a virtual object is a media object that consists of multiple objects, that may be obtained by executing a query, having mutually exclusive conditions to determine which object will be selected. Actually, the requirement that the conditions are mutually exclusive is overly strict. A more pragmatic approach would be to regard the objects as an ordered sequence, from which the first eligible one will be chosen, that is provided that its associated conditions are satisfied.

As an example, you may look at the Universal Media proposal from the Web3D Consortium, that allows for providing multiple URNs or URLs, of which the first one that is available is chosen. In this way, for instance, a texture may be loaded from the local hard disk, or if it is not available there from some site that replicates the Universal Media textures.

6

networked virtual environments

It does seem to be an exageration to declare *networked virtual environments* to be the ultimate challenge for networked multimedia, considering that such environments may contain all types of (streaming) media, including video and 3D graphics, in addition to rich interaction facilities. (if you have no idea what I am talking about, just think of, for example, Quake or DOOM, and read on.) To be somewhat more precise, we may list a number of essential characteristics of networked virtual environments, taken from Singhal and Zyda (1999):

networked virtual environments

- *shared sense of space* – room, building, terrain
- *shared sense of presence* – avatar (body and motion)
- *shared sense of time* – real-time interaction and behavior

In addition, networked virtual environments offer

- *a way to communicate* – by gesture, voice or text
- *a way to share ...* – interaction through objects

Dependent on the visual realism, resolution and interaction modes such an environment may be more or less 'immersive'. In a truly immersive environment, for example one with a haptic interface and force feedback, interaction through objects may become even threathening. In desktop VEs, sharing may be limited to the shoot-em-up type of interaction, that is in effect the exchange of bullets.

Networked virtual environments have a relatively long history. An early example is SIMNET (dating from 1984), a distributed command and control simulation developed for the US Department of Defense, Singhal and Zyda (1999). Although commercial multi-user virtual communities, such as the *blaxxun* Community server, may also be ranked under networked virtual environments, the volume of data exchange needed for maintaining an up-to-date state is far less for those environments than for game-like simulation environments from the military tradition. Consider, as an example, a command and control strategy game which contains a variety of vehicles, each of which must send out a so-called *Protocol Data Unit* (PDU), to update the other participants as to their actual location and speed. When the delivery of PDUs is delayed (due to for example geographic dispersion, the number of participants, and the size of the PDU), other strategies, such as *dead reckoning*, must be used to perform collision detection and determine possible hits.

To conclude, let's establish what challenges networked virtual environments offers with respect to software design and network performance.

challenges

- *network bandwidth* – limited resource
- *heterogeneity* – multiple platforms
- *distributed interaction* – network delays
- *resource management* – real-time interaction and shared objects
- *failure management* – stop, ..., degradation
- *scalability* – wrt. number of participants

Now it would be too easy to delegate this all back to the network provider. Simply requiring more bandwidth would not solve the scalability problem and even though adding bandwidth might allow for adding another hundred of entities, smart updates and caching is probably needed to cope with large numbers of participants.

The distinguishing feature of networked virtual environments, in this respect, is the need to

manage dynamic shared state

to allow for real-time interaction between the participants. Failing to do so would result in poor performance which would cause immersion, if present at all, to be lost immediately.

7

example(s) – *unreal*

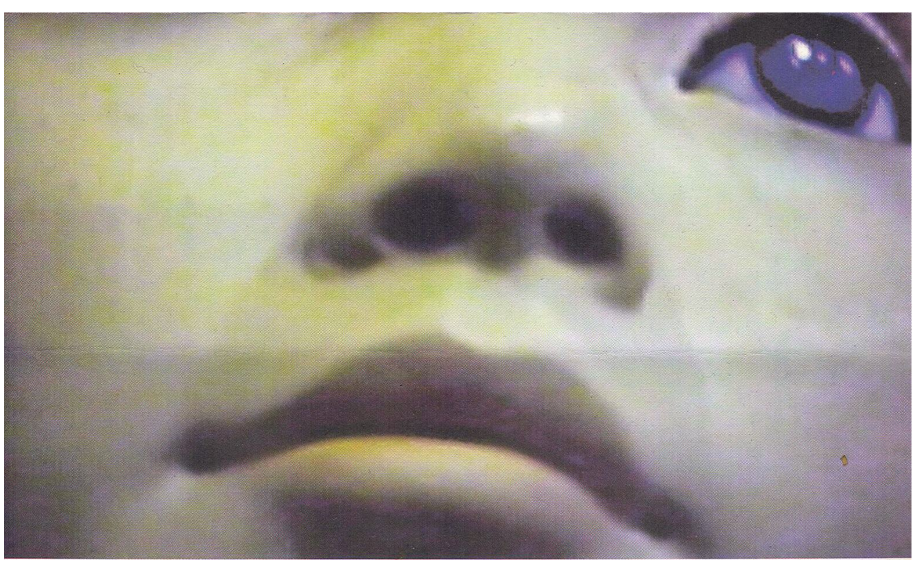
Unreal Tournament[57] is a highly popular multiplayer game. The storyline is simple, but effective: *It's the year 2362. The most anticipated Tournament ever is about to take place, dwarfing the spectacle and drama of previous events. The finest competitors ever assembled prepare to lay waste to their opponents and claim the Tournament Trophy for themselves.*

There a a number of roles you can associate with:

> the corrupt, thunder crash, iron guard, juggernauts, iron skull, sun blade, super nova, black legion, fire storm, hellions, bloof fist, goliath

An interesting feature of the Unreal Tournament games is that they can be andapted and even be re-programmed[58] by the users themselves, has has been done for example for the *Mission Rehearsal Exercise* discussed in section 9.2.

scripting: www.gamedev.net/reference/list.asp?categoryid=76



[57] www.unrealtournament.com
[58] www.unrealtournament.com/ut2004/screenshots.php

8

research directions– *architectural patterns*

Facing the task of developing a multimedia information system, there are many options. Currently, the web seems to be the dominant infrastructure upon which to build a multimedia system. Now, assuming that we chose the web as our vehicle, how should we approach building such a system or, in other words, what architectural patterns can we deploy to build an actual multimedia information system? As you undoubtly know, the web is a document system that makes a clear distinction between *servers* that deliver documents and *clients* that display documents. See Eliens (2000), section 12.1. At the server-side you are free to do almost anything, as long as the document is delivered in the proper format. At the client-side, we have a generic document viewer that is suitable for HTML with images and sound. Dependent on the actual browser, a number of other formats may be allowed. However, in general, extensions with additional formats are realized by so-called *plugins* that are loaded by the browser to enable a particular format, such as *shockwave*, *flash* or *VRML*. Nowadays, there is an overwhelming number of formats including, apart from the formats mentioned, audio and video formats as well as a number of XML-based formats as for example SMIL and SVG. For each of these formats the user (client) has to download a plugin. An alternative to plugins (at the client-side) is provided by Java *applets*. For Java applets the user does not need to download any code, since the Java platform takes care of downloading the necessary classes. However, since applets may be of arbitrary complexity, downloading the classes needed by an application may take prohibitively long.

The actual situation at the client-side may be even more complex. In many cases a media format does not only require a plugin, but also an applet. The plugin and applet can communicate with eachother through a mechanism (introduced by Netscape under the name LiveConnect) which allows for exchanging messages using the built-in DOM (Document Object Model) of the browser. In addition, the plugin and applet may be controlled through Javascript (or VBscript). A little dazzling at first perhaps, but usually not to difficult to deal with in practice.

Despite the fact that the web provides a general infrasructure for both (multimedia) servers and clients, it might be worthwhile to explore other options, at the client-side as well as the server-side. In the following, we will look briefly at:

- the Java Media Framework, and
- the DLP+X3D platform

as examples of, respectively, a framework for creating dedicated multimedia applications at the client-side and a framework for developing intelligent multimedia systems, with client-side (rich media 3D) components as well as additional server-side (agent) components.

Java Media Framework The Java platform offers rich means to create (distributed) systems. Also included are powerful GUI libraries (in particular, Swing), 3D libraries (Java3D) and libraries that allow the use and manipulation of images, audio and video (the Java Media Framework). Or, in the words of the SUN web site:

java Media Framework[59]

> *The JavaTM Media APIs meet the increasing demand for multimedia in the enterprise by providing a unified, non-proprietary, platform-neutral solution. This set of APIs supports the integration of audio and video clips, animated presentations, 2D fonts, graphics, and images, as well as speech input/output and 3D models. By providing standard players and integrating these supporting technologies, the Java Media APIs enable developers to produce and distribute compelling, media-rich content.*

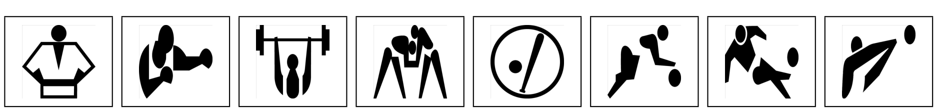
However, although Java was once introduced as the *dial tone of the Internet* (see Eliens (2000), section 6.3), due to security restrictions on applets it is not always possible to deploy media-rich applets, without taking recourse to the Java plugin to circumvent these restrictions.

DLP+X3D In our DLP+X3D platform, that is introduced in section **??** and described in more detail in appendix , we adopted a different approach by assuming the availability of a generic X3D/VRML plugin with a Java-based External Authoring Interface (EAI). In addition, we deploy a high-level ditributed logic programming language (DLP) to control the content and behavior of the plugin. Moreover, DLP may also be used for creating dedicated (intelligent) servers to allow for multi-user applications.

The DLP language is Java-based and is loaded using an applet. (The DLP jar file is of medium size, about 800 K, and does not require the download of any additional code.) Dua, again, to the security restrictions on applets, additional DLP servers must reside on the site from where the applet was downloaded.

Our plugin, which is currently the *blaxxun* VRML plugin, allows for incorporating a fairly large number of rich media formats, including (real) audio and (real) video., thus allowing for an integrated presentation environment where rich media can be displayed in 3D space in a unified manner. A disadvantage of such a unified presentation format, however, is that additional authoring effort is required to realize the integration of the various formats.

7.4 development(s) – clients versus servers



9

[59] java.sun.com/products/java-media/jmf/reference/api

questions

information system architecture

1. (*) What are the issues in designing a *(multimedia) information system architecture*. Discuss the tradeoffs involved.

concepts

2. What considerations would you have when designing an architecture for a multimedia information system.
3. Characterize the notion of *media abstraction*.
4. What are the issues in *networked multimedia*.

technology

5. Describe (the structure of) a video database, using *media abstractions*.
6. Give a definition of the notion of a *structured multimedia database*.
7. Give an example (with explanation) of querying a *hybrid multimedia database*.
8. Define (and explain) the notion of *virtual objects* in *networked multimedia*.

projects & further reading As a project, you may implement a multi-player game in which you may exchange pictures and videos, for example pictures and videos of celebrities.

Further you may explore the development of a data format for text, images and video with appropriate presentation parameters, including postioning on the screen and intermediate transitions.

For further reading you may study information system architecture patterns[60], nd explore the technical issues of constructing server based advanced multimedia applications in Li and Drew (2004).

the artwork

1. examples of dutch design, from Betsky (2004).
2. idem.
3. sceenshots – from *splinter cell: chaos theory*, taken from Veronica/Gammo[61], a television program about games.
4. screenshots – respectively *Sekken 5*, *Sims 2*, and *Super Monkey Ball*, taken from insidegamer.nl[62].
5. screenshots – from Unreal Tournament[63], see section 7.3.
6. idem.
7. idem.
8. *resonance* – exhibition and performances, Montevideo[64], april 2005.

[60] www.opengroup.org/architecture/togaf8-doc/arch/p4/patterns/patterns.htm
[61] www.gammo.nl
[62] http://www.insidegamer.nl
[63] www.unrealtournament.com/ut2004/screenshots.php
[64] www.montevideo.nl

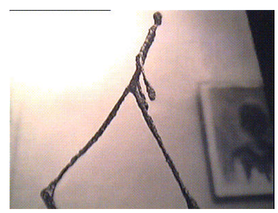
9. signs – sports, van Rooijen (2003), p. 274, 275.

Opening this chapter are examples of *dutch design*, taken from the book *False Flat*, with the somewhat arrogant subtitle *why is dutch design so good?*. It is often noted that dutch design is original, functional and free from false traditionalism. Well, judge for yourself.

The screenshots from the various games are included as a preparation for chapter 9, where we discuss realism and immersion in games, and also because multiplayer games like *Unreal Tournament* have all the functionality a serious application would ever need.

part iv. applications & tools

a journey of a thousand miles begins with the first step
chinese/japanese proverb



8. virtual environments
9. digital content creation
10. application development

2

reading directives In this part we will look in more detail at virtual environments as an interface to complex multimedia information spaces. In chapters 9 and 10, we will consider the issues that come into focus when creating digital content, and more in general, when developing a multimedia application. In chapter 10, the final chapter, we will look at some examples of multimedia application development.

Essential sections are section 8.1, which argues how virtual reality interfaces may be relevant, and sections 9.2 and 10.2, which provide examples of multimedia application development. Dependent on your experience you may skip sections 9.1 and 10.1, which provide rules of thumb for respectively content creation and application development.

perspectives Even in a more practical sense there are many perspectives that may characterize your academic stance. Design and application development, obviously, does not only consist of aesthetic or technical issues. So, non-exhaustively, you may look at such issues from the following perspectives:

perspectives – multimedia applications

- technical – algorithmic effects
- sociological – stakeholders and teamwork
- tool selection – Maya vd 3DSMAX
- political – negotiating support
- scientific – experience design
- computer science – tools and technologies
- artistic – portfolio as a design product

For example, the issues you may come across in an actual project may have to do more with people then any thing else, in other words may be more of a political nature, than of an aesthetic nature.

essay topics Even when you are primarily interested in the practice of developing digital content, it might well pay of to reflect on more theoretical issues. For example, consider writing an essay about:

- 2D vs 3D aesthetics animation – stills, sequences and stories
- elements of style – diversity and confluence
- models of creativity – a critical evaluation

In writing about such issues you should always beware of the risk of abstract speculation. So, look for examples in the domain of art, design or street culture to demonstrate your point.

3

the artwork

1. walking figure – sculpture by Alberto Giacometti, Hohl (1971).
2. signs – meteorological symbols, van Rooijen (2003), p. 214, 215.
3. photographs – Jaap Stahlie[65], commissioned work, using traditional non-digital techniques.

[65] www.jaapstahlie.com

8. virtual environments

augmented virtuality acts as an intelligent looking glass

learning objectives

After reading this chapter you should be able to characterize the notion of virtual context, discuss the issue of information retrieval in virtual environments, explain what is meant about intelligent multimedia and discuss the potential role of intelligent agents in multimedia applications.

From a user perspective, virtual environments offer the most advanced interface to multimedia information systems. Virtual environments involve the use of (high resolution) 3D graphics, intuitive interaction facilities and possibly support for multiple users. In this chapter, we will explore the use of (desktop) virtual environments as an interface to (multimedia) information systems. We will discuss a number of prototype implementations illustrating, respectively, how paintings can be related to their context, how navigation may be seen as a suitable answer to a query, and how we can define intelligent agents that can interact with the information space. Take good notice, the use of virtual environments as an interface to information systems represents a major challenge for future research!

1

8.1 virtual context

Imagine that you walk in a museum. You see a painting that you like. It depicts the Dam square in 17th century Amsterdam. Now, take a step forwards and suddenly you are in the middle of the scene you previously watched from some distance. These things happen in movies.

Now imagine that you are walking on the Dam square, some sunday afternoon in May 2001, looking at the Royal Palace, asking yourself is this where Willem-Alexander and Maxima will get married. And you wonder, what did this building and the Dam square look like three centuries ago. To satisfy your curiosity you go to the Royal Museum, which is only a half hour walk from there, and you go to the room where the 17th century city-scape paintings are. The rest is history.

We can improve on the latter scenario I think. So let's explore the options. First of all, we may establish that the Dam square represents a rich information space. Well, the Dam Square is a 'real world' environment, with it has 700 years of (recorded) history. It has a fair amount of historical buildings, and both buildings and street life have changed significantly over time.

So, we can rephrase our problem as

how can we give access to the 'Dam square' information space

But now we forget one thing. The idea underlying the last scenario is that we somehow realize a seamless transition from the real life experience to the information space. Well, of course, we cannot do that. So what did we do?

2

Look at the screenshot from our *virtual context* prototype. You can also start the VRML demo version that is online, by clicking on the screenshot. What you see is (a model of) the Dam square, more or less as it was in 2001. In the lower part, you see a panel with paintings. When you click on one of these painting, your viewpoint is changed so that you observe the real building from the point of view from which the painting was made. Then using the controls to the right of the panel, you can overlay the real building with a more or less transparent rendering of the painting. You can modify the degree of transparency by turning the dial control. You may also make the panel of paintings invisible, so that it does not disrupt your view of the Dam and the chosen overlay.

In other words, we have a VR model of Dam square and a selection of related paintings from the Royal Museum, that are presented in aa panel from which the user can choose a painting. We deploy viewpoint adjustment, to match the selected painting, and we use overlay of paintings over buildings, in varying degrees of transparancy, to give the user an impression of how the differences between the scene depicted in the painting and the actual scene in (the virtual) reality.

We have chosen for the phrase *virtual context* to characterize this prototype, since it does express how virtual reality technology enables us to relate an information space to its original context.

From the perspective of virtual reality, however, we could also have characterized our prototype as an application of *augmented virtual reality*, since what we have is a virtual reality model of a reallife location that is augmented with information that is related to it, (almost) without disrupting the virtual reality experience. In summary, we may characterize our approach as follows.

augmented virtual reality

- give user sense of geographic placement of buildings
- show how multiple objects in a museum relate to eachother
- show what paintings convey about their subject, and how

Considering the fact that many city-scape paintings of Amsterdam have been made, many of which are in the Royal Museum, and that paintings may say many things about their subject, we believe that our approach is viable for this particular instance. The augmented virtual reality approach would also qualify as a possible approach to cultural heritage projects, provided that sufficient pictorial material is available or can be reconstructed.

Although we were quite satisfied with what we accomplished, there are still many things that can be done and also a number of open problems. Guided tours are a wellknown phenomenon. But how to place them in our virtual context is not entirely clear. As another problem, our approach does not seem suited to account for buildings that do no longer exist. Another thing we have to study is how to change the temporal context, that is for example change from a model of the dam in 2001 to a model of the Dam in 1850. We would then also like to have 'viewpoint transitions' over space and time!

Finally, to give better access to the underlying information space we must also provide for textual user queries, and find an adequate response to those queries.

<name a=ex-t-7-1>

VRML To realize our prototype we used VRML, which limits us to medium quality desktop VR. At this stage, VRML is a good option, since it is a relatively stable format with a reasonable programmatic model. In short, what VRML offers is

VRML

- declarative means for defining geometry and appearance
- prototype abstraction mechanism
- powerful event model
- relatively strong programmatic capabilities

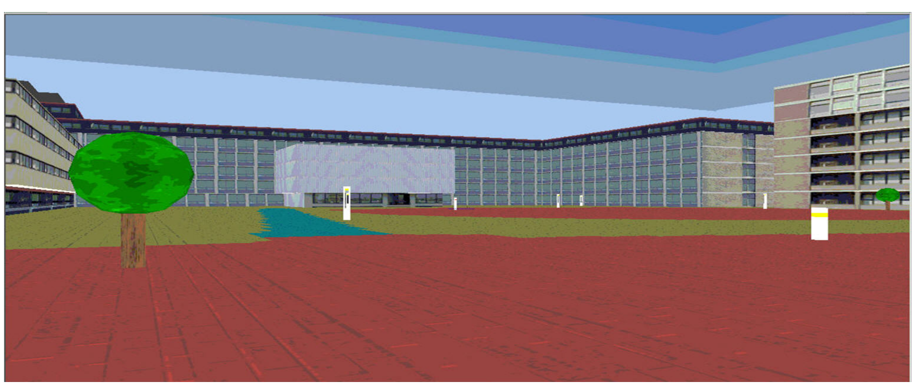
Although VRML allows for writing models (including geometry and appearance) using a plain text editor, many tools support export to VRML. As a consequence, often tools are used to create more complex models.

In addition, VRML allows for defining prototype abstractions, so reuse of models and behavior can be easily realized.

Defining dynamic behavior involves the routing of events that may come from a variety of built-in sensors (for example a TimeSensor for animations) to scripts or so-called interpolators, that allow for the manipulation of geometry and appearance parameters of the model.

In particular, the use of scripts or the *External Authoring Interface* (EAI), that allows for defining behavior in Java, is essential for realizing complex behavior.

Summarizing, VRML is a sufficiently rich declarative language for defining 3D scenes, with a relatively powerful programming model for realizing complex behavior. Some may think that VRML is dead. It isn't. The underlying model is endorsed in both the X3D and RM3D standards, simply since it has proven its worth.



3

research directions– *augmented virtuality*

Given an information space, there is a duality between information and presentation. For an audience or user to be able to digest a presentation, the amount of information must be limited. Effective presentation, moreover, requires the use of proper rethorics (which may be transcoded as *ways of presenting*) that belong to the medium. Using VR, which is (even in its desktop format) a powerful presentation vehicle, one should always beware of the question *what is it good for?* Generally one may ask, what is the added value of using VR? In an abstract fashion the answer should be, to bridge the gap between information content and presentation. Or, in other words, to resolve the duality between information and presentation!

Let's look at an example, a site about archeology, announced as a site offering *Virtual Archeology*. Perhaps it is good to bring to your attention that the *virtual*, in Computer Science, means nothing but another level of indirection to allow for a (more) flexible usage of entities or objects. See Eliens (2000), section 1.2.

virtual archeology

- variety of archeological sites
- various paths through individual site
- reconstruction of 'lost' elements
- 'discovery' of new material
- glossary – general background knowledge

For a site about archeology, *virtual* means the ability to present the information in a number of ways, for example as paths through a particular site, with the possibility to explore the reconstruction of lost or perished material, and (for students) to discover new perspectives on the material. In addition, for didactic reasons there may also be a glossary to explain concepts from archeology.

Now, how would you construct such a site about virtual archeology? As a collection of HTML pages and links? It seems that we can do better, using VR and rich interaction mechanisms!

So, what is meant by *augmented virtuality*? Nothing that hasn't been expressed by the notion of *augmented virtual reality*, of which an example has been given in this section. The phrase *augmented virtuality* itself is just one of those potentially meaningless fancy phrases. It was introduced simply to draw your attention to the duality between information and presentation, and to invite you to think about possible ways to resolve this duality.

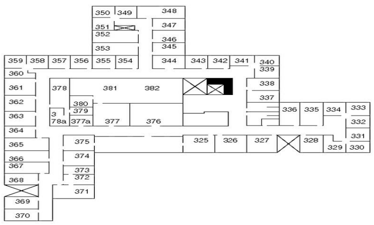
8.2 navigation by query

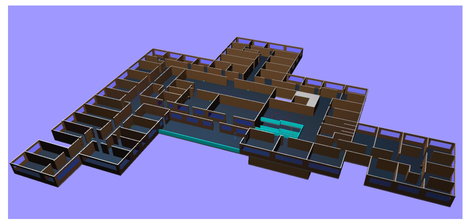
Virtual worlds form (in itself) a rich repository of multimedia information. So, when working on the musical feature detector, sketched in sectin 6.3, the thought occurred to ask funding for a research project on information retrieval in virtual worlds. This project is called RIF, which stands for

RIF

Retrieval of Information in Virtual Worlds using Feature Detectors

For the RIF project, we decided to develop a small multi-user community of our own, using the *blaxxun* Community Server. Then, during the development of our own virtual environment, the question came upof how to present the results of a query to the user. The concept we came up with was *navigation by query*, and in this section we will look at the prototype we developed to explore this concept.





On the left is the 2D map of the third floor of CWI, on the right the model generated from it.

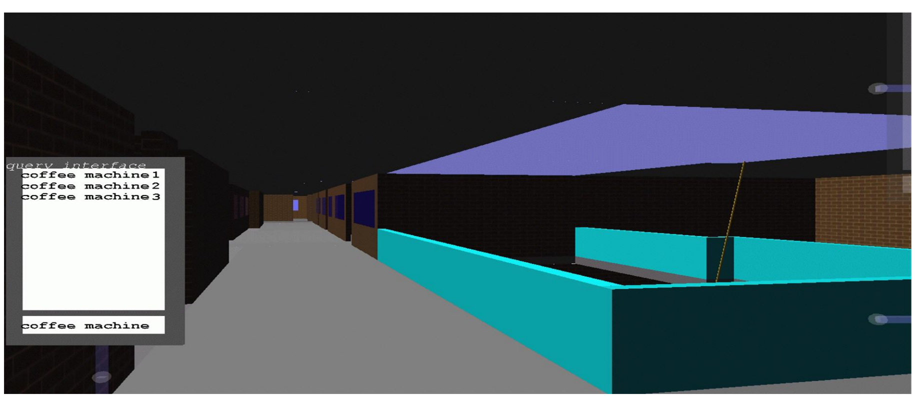
4

case study – CWI

For our prototype, we took one of the worlds of our virtual environment, the third floor of the CWI. The reason for this is that we were (at the time) doing our research there, and so there were plenty locations of interest, such as the rooms of our collegues, the printer room, and not to forget, the coffee corner.

We started out by taking a map of the third floor, and developed a model of it, using a tool developed by a student, who needed such a tool for realizing his game *Out of the Dark*.

When dwelling around in (this part of) our virtual environment, the user may pose (arbitrary) queries, for example *where is the coffee machine*.



5

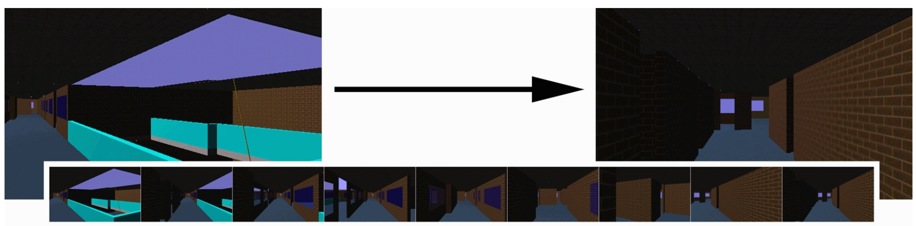
Remind, that after a few hours of research, coffee might be needed to get fresh ideas!

As a result, the user is then so to speak taken by the hand and led to one of the coffee machines that can be found on the third floor. In effect, with knowledge of the layout of the building a viewpoint transformation is executed, in a tempo that allows the user to

explore and discover

the (model of the) third floor of the CWI.

The idea is rather straightforward. Some have asked us why *navigation by query* might be useful. Well, simply, it seems to offer an interesting alternative to navigation by explicit interaction and navigation in the formof a guided tour. Our primary goal in developing the prototype, however, was to see whether navigation by query is feasible, and under what conditions.



6

information in virtual worlds

Developing the prototype has forced us to think more explicitly about what information is available in virtual worlds, and (perhaps more importantly) how to gain access to it. So the question we asked ourselves was

what are we searching for?

Now, in a virtual world, such as the ones built with VRML, we can distinguish between the following types of information: viewpoints, that is positions in the world from where interesting things can be looked at or accessed in any other way; areas of interest, where those intersting things are located; objects, that may provide information or offer particular kinds of functionality; persons, that is other users thatare visiting the world; and even text, which might be on billboards or slides.

Some of this information is, so to speak, hard-wired in the model and may be accessed anytime, in some cases even by scanning the VRML file. Other information, however, is of a more dynamic nature, since it might be due to the presence of multiple users, the execution of scripts, or events that happen in response to user interaction. Some information may even be explicitly hidden, such as for example the actions one should take in solving a puzzle or playing a game.

When the virtual world is loaded, all the information (or at least most of it) is present in the so-called scenegraph, the structure thatis built to render the world. Using the software interface to access the scengraph (which is usually browser-specific), we can look for annotations, node types and textual content to extract information from the world. This information may then be stored in a database, and be reused later for other users and queries. In principle, more advanced techniques could be used to extract information from the materials used, and even from textures and geometry.

presentation issues

In our prototype, we aimed at solving the question how to present the results of a query, using navigation. First of all, we had to

choose a metaphor

for navigation. Dependent on the object of interest a viewpoint can be selected. For a viewpoint, it is just that viewpoint. For an area of interest, the viewpoint selected must enable the user to view the area, and when objects or persons are chosen, care must be taken not to block the users' view by some obstacle.

Now answering a query then comes down to planning a suitable route and apply a series of viewpoint transformations along that route.
Not surprisingly, the navigation metaphor we chose was

walking

as the preferred mode of viewpoint transformtions.

the prototype

The structure of the prototype is depicted in the figure below.

In realizing the prototype,we made the following (simplifying) assumptions.

We avoided a number of difficulties by choosing for explicit annotations (which indicate locations and areas of interest), and by avoiding the intricacies of route planning and advanced text processing.

The requirements laid down before hand just stated that we would have a database and that we would avoid superfluous user interface elements. Instead, we used control and input panels written in VRML, in order to provide a 3D(pseudo-immersive) interface.

Now, our assumptions may in principle be relaxed. For example, annotation might be done incrementally by users that visit the world or to some extent even automatically, by using feature extractors. Instead if explicit maps, we may dynamically create maps based on users' navigation patterns. And, instead of simple keyword matching, we may apply more advanced text retrieval techniques. But this is left as future work. Anyway, we were satisfied that we could state the following conclusions:

conclusions

- navigation by query is feasible and may help users to find locations and objects
- determining suitable navigation routes without an explicitly defined map is hard

As is often the result with good research, you solve one problem and a number of other problems come up. So, one of the questions that remains was: how can we improve on navigation? What additional navigation support can we provide?

research directions– *extended user interfaces*

Is desktop VR a suitable candidate as an interface technology for multimedia information systems? And if so, what needs to be done to apply this technology effectively?

At first sight, our vision of applying VR as an interface to multimedia systems seems to be doomed to fail. As Ben Schneiderman, in a keynote for the Web3D Symposium 2002, observes:

3D GUI

> *Wishful thinking about the widespread adoption of three-dimensional interfaces has not helped spawn winning applications. Success stories with three-dimensional games do not translate into broad acceptance of head-tracking immersive virtual reality. To accelerate adoption of advanced interfaces, designers must understand their appeal and performance benefits as well as honestly identify their deficits. We need to separate out the features that make 3D useful and understand how they help overcome the challenges of dis-orientation during navigation and distraction from occlusion.*
>
> Ben Shneiderman

So, even if advanced (3D) user interfaces might be useful, there are a number of questions to raise. Again, following Ben Schneiderman:

> *Does spatial memory improve with 3D layouts? Is it true that 3D is more natural and easier to learn? Careful empirical studies clarify why modest aspects of 3D, such as shading for buttons and overlapping of windows are helpful, but 3D bar charts and directory structures are not. 3D sometimes pays off for medical imagery, chemical molecules, and architecture, but has yet to prove beneficial for performance measures in shopping or operating systems.* Ben Shneiderman

In particular, according to Schneiderman, we must beware of *tacky 3D*, gadgets in 3D space that are superfluous and only hindering the user to perform a task. Well-spoken and based on adequate observations! Nevertheless, at this stage, we should (in my opinion) adopt a slightly more liberal attitude and explore in what ways the presentation of (multimedia) information could be augmented by using (desktop) VR. But enough about *augmentation*. Let's discuss technology, and investigate what is required for the effective deployment of VR from the point of view of intelligent agents!

7

8.3 intelligent agents

Visitors in virtual environments are often represented by so-called avatars. Wouldn't it be nice to have intelligent avatars that can show you around, and tell you more about the (virtual) world you're in.

Now, this is how the idea came up to merge the RIF project, which was about information retrieval, with the WASP project, another acronym, which stands for:

WASP

Web Agent Support Program

The WASP project aims at realizing intelligent services using both client-side and server-side agents, and possibly multiple agents. The technical vehicle for realizing agents is the language DLP, which stands for

DLP

Distributed Logic Programming

Merging the two projects required providing the full VRML EAI API in DLP, so that DLP could be used for programming the dynamic aspects of VRML worlds.

background Historically, the WASP project precedes the RIF project, but we started working on it after the RIF project had already started. Merging these two projects had more consequences than we could predict at the time. The major consequence is that we shifted focus with respect to programming the dynamcis of virtual environments. Instead of scripts (in Javascript), Java (through the EAI), and even C++ (to program *blaxxun* Community Server extensions), we introduced the distributed logic programming language DLP as a uniform computational platform. In particular, for programming inteligent agents a logic programming language is much more suited than any other language. All we had to do was merge DLP with VRML, which we did by lifting the Java EAI to DLP, so that function calls are available as built-ins in the logic programming language.

When experimenting with agents, and in particular communication between agents, we found that communication between agents may be used to maintain a shared state between multiple users. The idea is simple, for each user there is an agent that observes the world using its 'sensors' and that may change the world using its 'effectors'. When it is notified by some other agent (that is co-located with some other user) it can update its world, according to the notification. Enough background and ideas. Let's look at the prototypes that we developed.

8

multi-user soccer game

To demonstrate the viability of our approach we developed a multi-user soccer game, using the DLP+VRML platform.

We chose for this particular application because it offers us a range of challenges.

multi-user soccer game

- *multiple (human) users* – may join during the game
- *multiple agents* – to participate in the game (e.g. as goalkeeper)
- *reactivity* – players (users and agents) have to react quickly
- *cooperation/competition* – requires 'intelligent' communication
- *dynamic behavior* – sufficiently complex 3D scenes, including the dynamic behavior of the ball

Without going into detail, just imagine that you and some others wish to participate in a game, but there are no other players thatwant to join. No problem, we just add some intelligent agent football players. And they might as well be taken out when other (human) players announce themselves.

For each agent player, dependent on its role (which might be *goal-keeper*, *defender*, *mid-fielder* and *forward*), a simple coginitive loop is defined: sensing, thinking, acting. Based on the information the agent gets, which includes the agent's position, the location of the ball, and the location of the goal, the agents decides which action to take. This can be expressed rather succinctly as rules in the logic programming formalism, and also the actions can be effected using the built-in VRML functionality of DLP.

Basically, the VRML-related built-ins allow for obtaining and modifying the values of *control points* in the VRML world.

control points

- get/set – position, rotation, viewpoint

These control points are in fact the identifiable nodes in the scenegraph (that is, technically, the nodes that have been given a name using the DEF construct).

This approach allows us to take an arbitrarily complex VRML world and manipulate it using the control points. On the other hand, there are also built-ins that allow for the creation of objects in VRML. In that case, we have much finer control from the logic programming language.

All in all we estimate that, in comparison with other approaches, programming such a game in DLP takes far less time than it would have taken using the basic programming capabilities of VRML.

agents in virtual environments

Let us analyse in somewhat more detail why agents in virtual environments may be useful. First of all, observe that the phrase *agents in virtual environments* has two shades of meaning:

agents in virtual environments

- virtual environments with embedded autonomous agents
- virtual environments supported by ACL communication

where ACL stands for *Agent Communication Language*. Our idea, basically is to use an ACL for realizing shared objects, such as for example the ball in the soccer game.

The general concept op multi-user virtual environments (in VRML) has been studied by the *Living Worlds Working Group*. Let's look at some definitions provided by this working group first. A *scene* is defined as a geometrically bounded, continuously navigable part of the world. Then, more specifically a *world* is defined as a collection of (linked) scenes.

Now, multi-user virtual environments distinguish themselves from single-user virtual environments by allowing for so-called *Shared Objects* in scenes, that is objects that can be seen and interacted with by multiple independent users, simultaneously. This requires synchronization among multiple clients, which may either be realized through a server or through client-to-client communication.

Commonly, a distinction is made between a *pilot* object and a *drone* object.

Shared Object

- *pilot* – instance that will be replicated
- *drone* – instance that replicates pilot

So, generally speaking, pilot objects control drone objects. There are many ways to realize a pilot-drone replication scheme. We have chosen to use agent technology, and correspondingly we make a distinction between *pilot agents*, that control the state of a shared object, and *drone agents*, that merely replicate the state of a shared object.

Since we have (for example in the soccer game) different types of shared objects, we make a further distinction between agents (for each of which there is a pilot and a drone version). So, we have *object agents*, which control a single shared object (like the soccerball). For these agents the pilot is at the server, and the drone is at the client. We further have agents that control the users' avatars, for which the pilot at user/client side, and the drone either at the server or the client. Finally, we have autonomous agents, like football players, with their own avatar. For those agents, the pilot is at the server, and the drones at the clients.

Now, this classification of agents gives us a setup that allows for the realization of shared objects in virtual environments in an efficient manner. See Huang et al. (2002) for details.

The programming platform needed to implement our proposal must satisfy the following requirements.

programming platform

- VRML EAI support
- distributed communication capabilities (TCP/IP)
- multiple threads of control – for multiple shared objects
- declarative language – for agent support

So, we adapted the distributed logic programming language DLP (which in its own right may be called an agent-oriented language *avant la lettre*), to include VRML capabilities. See the online reference to the AVID project for a further elaboration of these concepts.

9

PAMELA The WASP project's chief focus is to develop architectural support for web-aware (multi) agent systems. So, when we (finally) got started with the project we developed a taxonomy along the following dimensions:

taxonomy of agents

- 2D/3D – to distinguish between text-based and avatar embodied agents
- client/server – to indicate where agents reside
- single/multi – as a measure of complexity

A classification along these dimensions results in a lattice, with as the most complex category a *3D-server-multi-agent system*, of which the distributed soccer game is an example. See Huang et al. (2000).

When we restrict ourselves to *3D-client-single-agent systems*, we may think of,for example, navigation or presentation agents, that may help the user to roam around in the world, or that provide support for presenting the results of a query as objects in a 3D scene.

Our original demonstrator for the WASP project was an agent of the latter kind, with the nickname *PAMELA*, which is an acronym for:

PAMELA

Personal Assistent for Multimedia Electronic Archives

The PAMELA functional requirements included: autonomous and on-demand search capabilities, (user and system) modifiablepreferences, and multimedia presentation facilities. It was, however, only later that we added the requirement that PAMELA should be able to live in 3D space.

In a similar way as the soccer players, PAMELA has control over objects in 3D space. PAMELA now also provides animation facilities for its avatar embodiment.

To realize the PAMELA representative, we studied how to effect facial animations and body movements following the *Humanoid Animation Working Group* proposal.

H-Anim

- control points – joints, limbs and facial features

The H-Anim proposal lists a number of control points for (the representation of the) human body and face, that may be manipulated upto six degrees of freedom. Six degrees of freedom allows for movement and rotation along any of the X,Y,Z axes. In practice, movement and rotation for body and face control points will be constrained though.

presentation agent Now, just imagine how such an assistant could be of help in multimedia information retrieval.

presentation agent

> Given any collection of results, PAMELA could design some spatial layout and select suitableobject types, including for example color-based relevance cues, to present the results in a scene. PAMELA could then navigate you through the scene, indicating the possible relevance of particular results.

persuasion games But we could go one step further than this and, taking inspiration fromthe research field of *persuasive technology*, think about possible persuasion games we could play, using the (facial and body) animation facilities of PAMELA:

persuasion games

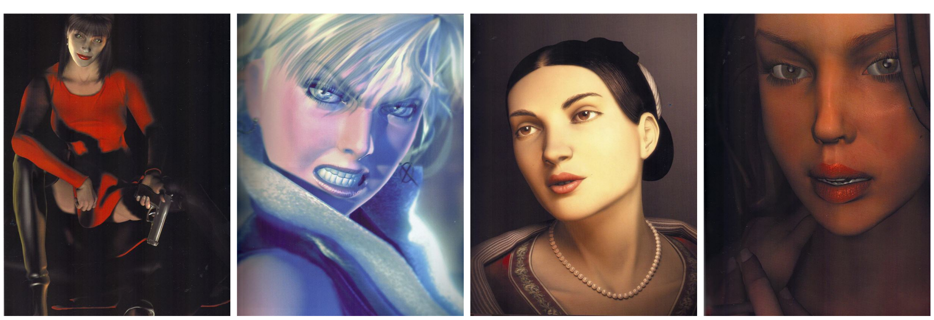
- single avatar persuasive argumentation
- multiple avatar dialog games

Just think of a news readerpresenting a hot news item. or a news reader trying to provoke a comment on some hot issue. Playing another trick on the PAMELA acronym, we could think of

PAMELA

Persuasive Agent with Multimedia Enlightened Arguments

I agree, this sounds too flashy for my taste as well. But, what this finale is meant to express is, simply, that I see it as a challenge to create such synthetic actors using the DLP+VRML platform.



10

research directions– *embodied conversational agents*

A variety of applications may benefit from deploying embodied conversational agents, either in the form of animated humanoid avatars or, more simply, as a 'talking head'. An interesting example is provided by *Signing Avatar*, a system that allows for translating arbitrary text in both spoken language and sign language for the deaf, presented by animated humanoid avatars. Here the use of animated avatars is essential to communicate with a particular group of users, using the sign language for the deaf.

Other applications of embodied conversational agents include e-commerce and social marketing, although in these cases it may not always be evident that animated avatars or faces actually do provide added value.

Another usage of embodied conversational agents may be observed in virtual environments such as Active Worlds, *blaxxun* Community and Adobe Atmosphere. Despite the rich literary background of such environments, including Neil Stephenson's Snow Crash, the functionality of such agents is usually rather shallow, due to the poor repertoire of gestures and movements on the one hand and the restricted computational model underlying these agents on the other hand. In effect, the definition of agent avatars in virtual environments generally relies on a proprietary scripting language which, as in the case of *blaxxun* Agents, offers only limited pattern matching and a fixed repertoire of built-in actions.

In contrast, the scripting language for *Signing Avatar* is based on the H-Anim standard and allows for a precise definition of a complex repertoire of gestures, as exemplified by the sign language for the deaf. Nevertheless, also this scripting language is of a proprietary nature and does not allow for higher-order abstractions of semantically meaningful behavior.

scripting behavior In this section we introduced a software platform for agants. This platform not only offers powerful computational capabilities but also an expressive scripting language (STEP) for defining gestures and driving the behavior of our humanoid agent avatars.

The design of the scripting language was motivated by the requirements listed below.

STEP

- *convenience* – for non-professional authors
- *compositional semantics* – combining operations
- *re-definability* – for high-level specification of actions
- *parametrization* – for the adaptation of actions
- *interaction* – with a (virtual) environment

Our scripting language STEP meets these requirements. STEP is based on dynamic logic Harel (1984) and allows for arbitrary abstractions using the primitives and composition operators provided by our logic. STEP is implemented on top of DLP,

As a last bit of propaganda:

DLP+X3D

> The DLP+X3D platform provides together with the STEP scripting language the computational facilities for defining semantically meaningful behaviors and allows for a rich presentational environment, in particular 3D virtual environments that may include streaming video, text and speech.

See appendix D for more details.

evaluation criteria The primary criterium against which to evaluate applications that involve embodied conversational agents is whether the application becomes more effective by using such agents. Effective, in terms of communication with the user. Evidently, for the *Signing Avatar* application this seems to be quite obvious. For other applications, for example negotiation in e-commerce, this question might be more difficult to answer.

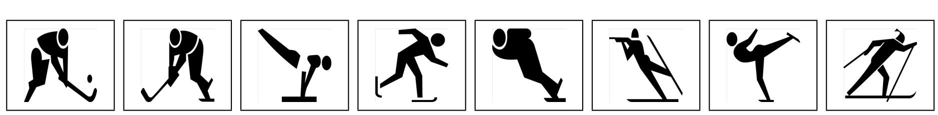
As concerns the embedding of conversationl agents in VR, we might make a distinction between *presentational VR*, *instructional VR* and *educational VR*. An example of educational VR is described in Johnson et al. (2002). No mention of agents was made in the latter reference though. In instructional VR, explaining for example the use of a machine, the appearance of a conversational agent seems to be quite natural. In presentational VR, however, the appearance of such agents might be considered as no more than a gimmick.

Considering the use of agents in applications in general, we must make a distinction between *information agents*, *presentation agents* and *conversational agents*. Although the boundaries between these categories are not clearcut, there seems to be an increasing degree of interactivity with the user.

From a system perspective, we might be interested in what range of agent categories the system covers. Does it provide support for managing information and possibly information retrieval? Another issue in this regard could be whether the system is built around open standards, such as XML and X3D, to allow for the incorporation of a variety of content.

Last but not least, from a user perspective, what seems to matter most is the naturalness of the (conversational) agents. This is determined by the graphical quality, as well as contextual parameters, that is how well the agent is embedded in its environment. More important even are emotive parameters, that is the mood and style (in gestures and possibly speech) with which the agents manifest themselves. In other words, the properties that determine whether an agent is (really) convincing.

8.4 development(s) – the metaverse revisited



questions

virtual environments

1. (*) Discuss how *virtual environments* may be used for giving access to *(multimedia) information*. Give a brief characterization of *virtual environments*, and indicate how *information (hyper) spaces* may be projected in a virtual environment.

concepts

2. What is meant by *virtual context*?
3. Give an example of *navigation by query*, and indicate its possible advantages.
4. Discuss the deployment of *(intelligente) navigation agents*.

technology

5. Give a brief characterization of: VRML.
6. What is a *viewpoint transformation*?
7. What kinds of navigation can you think of?
8. How may intelligent avatars be realized? Give an example.

projects & further reading As a project, I suggest the implementation of storytelling in virtual environments, with (possibly) an embodied agent as the narrator. You may further explore or evaluate the role of agents in multimedia applications and virtual environments.

For further reading in (real) VR I advice Sherman and Craig (2003), and for gaining an understanding in story telling and applications you may try to get hold of the proceedins, of TIDSE 2003[66], and TIDSE 2004[67].

the artwork

1. another series of *dutch light*[68].
2. *virtual context* – Dam Square, Amsterdam, see 8.1.
3. VU Campus in VRML – student project.
4. CWI 3th floor, floormap and model, see 8.2..
5. query – on 3th floor of CWI.
6. navigation – on 3th floor of CWI.
7. soccer game – image from WASP project, see section 8.3.
8. *digital beauties* – taken from Wiedermann (2002).
9. *digital beauties* – taken from Wiedermann (2002).
10. signs – sports, van Rooijen (2003), p. 276, 277.

[66]www.zgdv.de/TIDSE03
[67]www.zgdv.de/TIDSE04
[68]www.dutchlight.nl

Another sequence of *dutch light*, opening this chapter, is meant to make you wonder about *realism*. Is virtual reality less 'real'? With a reference to section 2.3, where I quoted Bolter and Grusin (2000) on *re-mediation*, I may remark that the graphic style chosen for presenting the virtual environment strongly determines whether the environment is experienced as 'realistic'. In our culture this is generally a *photorealistic style*, as for example in the *Mission Rehearsal Exercise* discussed in the next chapter, section 9.2. The *digital beauties* are not only a pleasure to look at, but do also display a wide range of postures and styles.

9. digital content creation

post-modern design allows for sampling

learning objectives

After reading this chapter you should be able to mention some basic rules of digital content creation, discuss what criteria your portfolio should meet, describe how you would approach the design of a logo, explain the notion of user-centered design, and characterize the issues that play a role in dveloping multimedia for theatre.

Whether your ambition is to become a professional designer or not, also for students of information science and computer science, a course in visual design is a must, I think.

In this chapter, we will treat various aspects of digital content creation. The first section discusses how to approach visual design and gives a number of basic design assignments, that can be used to get experience with visual design. Section 2 discusses the issue of workflow and tools, and investigates how design fits in with the process of developing multimedia applications. In the final section, I will elaborate on a theatre project I was involved in, for which I had to develop an augmented reality application.

1

9.1 visual design

When you are trained as a visual artist, as I once was (in the pred-digital era), you must do many basic exercises with form and color, learn the skills of drawing and painting, including sill lifes and portraits. And, to graduate you must make an exhibition of your work over the years.

In the digital era, things have changed. The skills that you need to learn, which include the use of tools for digital content creation, as well as the artistic goals, I would say. What has not changed, however, is the need for basic exercises and the presentation of your work, that is the creation of a portfolio.

In this section, I will give an outline of the course *visual design*, that I started in februari 2005. This outline is meant to give you a general idea of how to approach design, and to give you some hints on how to acquire the skills needed to act as a designer. If you have the opportunity to take a course in visual design, then don't hesitate and do it!

2

perspective(s)

The overall goal of the *visual design* course is to establish some basic *aesthetic awareness*, by providing suitable exercises and assignments. In addition, the student is supposed to become familiar with the craft of design, which necessarily, but not exclusively, involves the use of tools and techniques.

To accomodate for the various interests and backgrounds of the students, we distinguish between several *tracks* or perspectives, as summarized below:

track(s) – perspective

- styling – concept and presentation
- digital content – material, animation
- tech track – special effects

To illustrate the various perspectives, I invited guest speakers who showed their work and talked about their approach.[69] The process of design is very complex, ranging from conceptual explorations and sketching up to the stage of finalizing delivery. It is also very personal. However, as each of the speakers testified to, a significant portion of the time and effort goes into negotiating with the client. Whether it involves taking photographs for an advertisement or setting up a campaign, it takes a lot of back and forth to get an idea of what the client wants. In the agency of one of the speakers Mark Veldhuijzen van Zanten, they have created roles to help eachother come up with the right ideas.

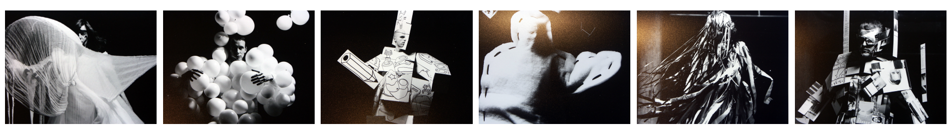
www.178aardigeontwerpers.nl

[69] A selection of their works is hown in this book.

- *e-motionist* – make emotion, rhythm and movement flow together
- *chaoticus* – who sees chaos within order
- *formologist* – who approaches the fabrication of forms as an art
- *infonaut* – who moves in the twilight zone of information and meaning
- *transformator* – who transforms images and concepts into new matter

Although slightly ironic, and in practice not so clear-cut, these roles give you an idea what cognitive modes are involved in bringing an original concept to a stage of finalization.

Coming up with an idea and sketching require a more reflective cognitive mode, whereas finalizing a design requires a more experience-related cognitive mode. It is important to choose the right tools to work with, dependent on which phase you're in, for example paper and pencil when you are still sketching and Photoshop or maya when you put your ideas into production.



3

deliverable(s)

However interesting the process may be, design is not about process but about product(s). Such a product may be, dependent on what you ar good at, one of the following:

products

- web site – e.g. conference, campaign (browse)
- 2D/3D animation – promotion/ad (temporal sequence)
- virtual space – game/infotainment (navigate)
- ebook – story (sequential experience)

As indicated between brackets, each of the products favors a particular mode of interaction. Although *interaction* is not an aspect of visual design as such, it is an important aspect to take into account. In section 9.2, we will look in more detail at the issue of interaction and usability in general.

One easily overlooked issue in a design project, is the creation of a portfolio. There may easily be some confusion here with regard to what should be considered the product of design, one of the items in the list above, or your portfolio. The answer is simple. Both! As a record of the process of design, the portfolio is itself a product of design.

portfolio – design as a product

- concept(s)
- sketches & explorations

- finalized products
- evaluation & reflection

What criteria should a portfolio meet? Well, nowadays it is not only common to have your portfolio on the web, I have been told that you cannot do without it. So, first of all, your portfolio should be web-friendly. And your work should not be too many clicks away! And, secondly, it should give sufficient insight in what you have to offer, so that a potential client can decide whether it is worth the effort to contact you.

In our visual design course, I require that the portfolio contains a description of the concept(s) underlying the design, sketches and explorations as well as the finalized products. It is also required, after all it is an academic course, to provide an evaluation and some reflection in the form of an essay on a topic such as *2D versus 3D aesthetics*, *animation techniques*, or (more theoretical), *elements of style* or *theories of creativity.*

www.jaapstahlie.com

> In my perception a portfolio is about the
> past and I feel much more related to the
> present especially in my work as a
> photographer. To me the relation with
> the present and the subject/assignment directs
> my creativity, the experiences over the
> past draw my skills. My challenge is to be
> truly inspirred, to be present in the present.

However, as illustrated by the motivation Jaap Stahlie gives with his portfolio (above), it is perhaps wise not to overdo it!

There are basic exercises, obligatory for all students, and a final assignment, where you have a choice between three productions, each with a different supervisor. In addition, as explained in the guidelines, all students must write an essay, and give a presentation in class. For deadlines, see the schedule. There will be periodic checks on the status of your work. Each year there will be recommended themes.

assignment(s)

In the *visual design* course there are basic exercises, obligatory for all students, and a final assignment, where the student has a choice between three productions.

basic exercises

1. develop a logo
2. create a sign
3. design a collage
4. write a story

For the final assignment, there is a choice between the following assignments: developing a house style, creating a non-linear visual story, and designing a suitable game environment For this assignment) the students are allowed to work in groups. However, the contribution of each individual must be reflected in his/her own portfolio.

4

regulation(s)

The first requirement when working in an area such as visual design is that you acquire sufficient self-discipline to find the challenge in the assignments and to complete the tasks involved. Since a department of computer science is not the natural habitat for a course in visual design, I have laid down some strict rules:

rules

- be present – 2 omissions max.
- be in time – hard deadlines
- be online – have your portfolio available
- be creative – don't steal without a reason/mentioning
- be smart – there is no 2nd chance

These rules may well apply when you work, after you graduated, as an individual designer/developer or as a member of a team in some agency. To finish this section, I may remark that design is an interesting field, full of implicit (not always so obvious) wisdom and apparent paradoxes. Whatever you do, deliver! Silence is lethal. And, as another item of colloquial wisdom, be authentic, but only if possible!

5

research directions– *on creativity*

One of the assignments in the *visual design* course is an essay. As one of the recommended topics we have a reflection on theories of creativity. As such it is not a training in creativity[70]. This section contains some random thoughts on the processes and products of design, and ends with the provoking statement *there is no theory of creativity.* You may, however, try to find some counter-arguments, for example in the line of Hewlett and Selfridge-Field (1998).

multimediocrity Multimedia is a promising technology, and (nowadays) affordable. So we see that multimedia (which includes 3D-graphics, video and sound) is increasingly being used, also in information visualisation. But what is it good for? To quote Hughes (2000):

> *multimedia's promise is terribly generalized, it simply lets you do anything.*

As with any new technology, the early multimedia productions (in particular CDROM and CD-I) were not optimal with respect to (aesthetic) quality. To quote Hughes (2000), again:

shovelware – *multimediocrity*

> ... far from making a killing, it looked as if the big boys ... had killed the industry by glutting the market with inferior products.

Perhaps the industry in the late eighties did not have the right business model. But, then again, what are the chances of multimedia in our time. One more quote from Hughes (2000):

> if multimedia is comparable to print then yes, we'd be crazy to expect it to mature in a mere ten years.

[70] www.goshen.edu/%7Emarvinpb/arted/tc.html

eliminating complexity So now, in the new millenium, we are (sadder and wiser) in a position to approach the effective deployment of multimedia afresh. What we look for is aesthetic quality. How do we find it? Easy enough, just be authentic.

> *"Learning how to not fool ourselves is, I'm sorry to say, something that we haven't specifically included in any particular course that I know of. We just hope you've caught it by osmosis."* Richard Feynman

Authentic in creating multimedia means, apart from not fooling yourselves, that you must become aware of the message orinformation you want to convey and learn to master the technology to a sufficient degree. But beware, an effective multimedia presentation is not the same as scientific argumentation:

the media equation

> *We regularly exploit the media equation for enjoyment by the willing suspension of our critical faculties. Theatre is the projection of a story through the window of a stage, and typically the audience gets immersed in the story as if it was real.*

These quotes, as well as the following one have been taken from an online essay on *eliminating complexity* which provides an argument against inessential gadgets and spurious complexity and bells and whistles in whatever you can think of, including user interfaces and scientific theories. Back to the subject, what does *master the technology to a sufficient degree* mean? Just remember that what you do is a form of engineering.

> *"engineering is the art of moulding materials we do not wholly understand ... in such a way that the community at large has no reason to suspect the extent of our ignorance."* A. R. Dykes.

In other words, learn the tool(s) that you are using to a degree that you master the basics and easily cut through its apparent magic.

theories of creativity Producing multimedia, in whatever form, has an element of craftsmanship. But, given the need for aesthetic quality, whatever way you approach it, there is an element of creativity. That means, you're in for a challenge. And, to quote Hughes (2000),

> *The best thing is to empower yourself. But before you can do that, you need to understand what you are doing – which is a surprisingly novel thing to do.*

Now it is tempting to look for a set of guidelines and rules that give you a key to creativity. So let me be straight with you:

there is no theory of creativity

On the other hand, there are techniques for producing ideas. And some recommend a sequence of steps, such as:

steps

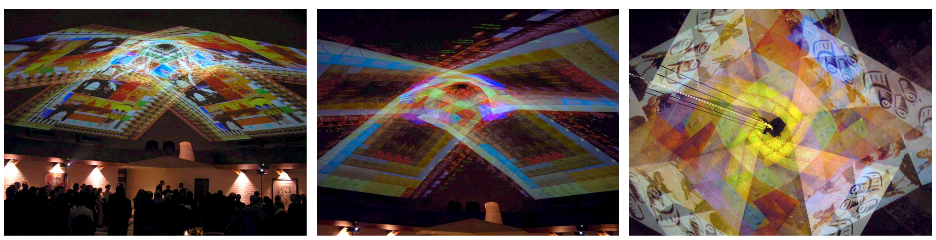
> browse, explore; chew it over; incubation, let it rest; illumination (YES); verification,*does it work?*

And in addition, still following Hughes (2000), there are some general rules:

general rules

- *if you aim to please everybody, you will please nobody*
- *constraints come with the territory, you must learn to love them*
- *emotional charge is the key to success*

Now, if you'd ask me, I would say, just
make your virtual hands dirty.



6

9.2 designing the user experience

In a time in which so much information is available as in ours, we may make statements like

postmodern design

> ... postmodern design is of a highly reflective nature ... appropriating design of the past ... in other words, sampling is allowed but no plagianarism

One interpretation might be that it is impossible to be original. But another interpretation might be that it is undesirable to be (totally) original. In other words, as discussed in section 2.3, it is necessary that your design contains references to not only some real situation but also to other work, so that it can be understood and experienced by the user/spectator/client whatever you like to call the people that look at your work. As observed in Sherman and Craig (2003), designing for multimedia does not take into account only the technological or aesthetic issues, but also constraints on what people can perceive and what the experiential context is in which the work is presented, which may be re-phrased more plainly as what expectations the user has.

game design

Let us consider how these observations affect one of the project assignments for our *visual design* course. Also for *game design*, there are several options, dependent on the track the student is most comfortable with.

game design

- style – develop concept, plot and visual assets for *a game of choice*
- content – develop environments, models and animations for *a game of choice*
- effects – develop models, textures and special effects (shaders) for *a game of choice*

To explain, *style* may be considered to involve the whole gamut of concepts, plot and genre, as well as the visual assets or props of the game, those things by which the game differentiates itself from other games. Since the reader of this book will probably be more familiar with games than the author, there is no need to expand on these issues. *Content* is concerned with the actual game environment, including the models and animations. Finally, *effects*, to simplify things, is everything else, those things that are visual but does not belong to the story line or game environment.

Games, perhaps more than any other multimedia application, are appealing, not because they are useful, although they might be, but because the user gets emotionally involved, not to say addicted. Now, following Norman (2004),

> did you ever wonder why cheap wine tastes better in fancy glasses?

Exactly, because cheap glasses do not give us the same emotion. It is, indeed, a matter of style!

Obviously, games are played for fun. As applications, games may be classified as seductive, which is, see section 2.3, stronger than persuasive. Norman (2004) distinguishes between four categories of pleasure.

seduction

- physio-pleasure – of the body
- socio-pleasure – by interaction with others
- psycho-pleasure – due to use of the product
- ideo-pleasure – reflecting on the experience

In other words, games are seductive, or fun to play, because they arouse any combination of pleasure from the categories above. Which combination depends on the kind or genre of game. Quoted from Norman (2004), but originally from Wolf we can list, not exhaustively, the following genres of video game:

genre(s)

> Abstract, Adaptation, Adventure, Artificial Life, Board Games, Capturing, Card Games, Catching, Chase, Collecting, Combat, Demo, Diagnostic, Dodging, Driving, Educational, Escape, Fighting, Flying, Gambling, Interactive Movie, Management Simulation, Maze, Obstacle Course, Pencil-and-Paper Games, Pinball, Platform, Programming Games, Puzzle, Quiz, Racing, Role-Playing, Rhythm and Dance, Shoot Em Up, Simulation, Sports, Strategy, Table-Top Games, Target, Text Adventure, Training Simulation, and Utility.

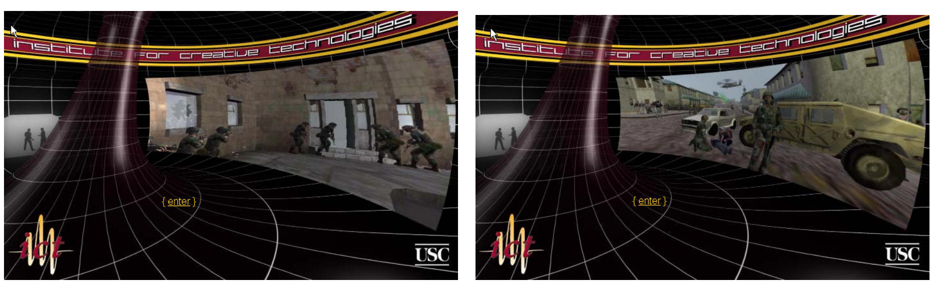
When you develop a game it is good to reflect on what genre your game belongs to, because that will directly affect the user's expectation when playing the game, due to the conventions and rules that exist within a particular genre. For video games, which can be characterized as *a mixture of interactive fiction with entertainment*, interaction is evidently another distinguishing factor in determining the success of the game.

Corresponding to the kind of pleasure a user may experience, Norman (2004) distuinguishes between three levels of design:

levels of design

- visceral – what appeals to our intuition (*affordance*)
- behavioral – is all about use (*performance*)
- reflective – its all about message, culture and meaning

Of these, the latter two should be rather obvious, although we will elaborate on the notion of *usability* later on. But what does *affordance* mean, and how is it related to our intuition.



7

affordance – ecology of behavior

The notion of affordance[71] has a long history. According to Don Norman, the word "affordance" was invented by the perceptual psychologist J. J. Gibson, to refer to the properties of the world that 'invite' actions, as for example a char invites one to sit. Originally, however, the notion of affordance dates back to the beginning of the 20th century, when it was used in phenomenologist philosophy to describe how the world around us presents itself as meaningful. Affordance, in other words, is a concept that explains why it seems natural for us to behave in a particular way, either because it is innate, as the reflex to close one's eyes by sudden exposure to light, or because we have learned that behavior, as for example surfing the web by clicking on links. In game or product design, thinking about 'affordance' may help us to find the most natural way to perform certain actions. Natural is in this context perhaps not the best phrase. What we must take into account is what is perceived as an affordance, and how actions fit in with what we may call an exology of behavior (with the system).

[71] www.jnd.org/dn.mss/affordances-and-design.html

How does this diversion in abstract philosophy help us design better games? To answer this question, I'd like to recount my visit at the Virtual Humans Workshop, held in october 2004 at the Institute of Creative Technologies[72], in Los Angeles. Of interest in particular is the ICT Games Project:

ICT Games Project

> The goal of the ICT games project is to develop immersive, interactive, real time training simulations to help the Army create a new generation of decision-making and leadership-development tools.

As further explained on the website:*with the cooperation of the U.S. Army Research, Development and Engineering Command, Simulation Technology Center (RDECOM STC), Training and Doctrine Command (TRADOC), and commercial game development companies, ICT is creating two training simulations that are intended to have the same holding power and repeat value as mainstream entertainment software.*

The two training applications developed by ICT are:

- Mission Rehearsal Exercise – to solve a potential conflict after a car accident
- Language Training Simulation – to learn how to contact local leaders in arabic

The *mission rehearsal exercise* is situated in former Yugoslavia. The trainee is confronted with the situation after a car accident in which a boy got injured. The mother of the boy is furious, and a potentially hostile crowd is waiting. An army convoy is on its way to a nearby airport, and needs to pass the crossing where the accident took place. The trainee must decide what to do and give appropriate orders, knowing that the wrong decision may lead to serious trouble.

The *language training simulation* is situated in the Middle-East, and is meant to teach the trainee not only some basic arabic but also proper ways of conduct, in conformance with local customs to gain confidence.

Both applications are highly realistic, with impressive graphics.[73] The both support speech input. The challenge in both simulation games was to come up with a natural way to indicate to the trainee what options for actions were available. Natural means, in this context, that it should fit within the simulation or game environment. Obviously, a menu or a row of pushbuttons does not fit naturally within such an environment, and would break what we have previously, in section 2.3 called 'immersion'.

I was invited at ICT for the Virtual Humans Workshop because of my involvement with embodied conversational agents (ECAs), as discussed in section 8.3. The topic of the workshop was, among others, to investigate whether the notion of affordance could help in analyzing and evaluating the interaction of a user/trainee with the simulation game. These were the questions we tackled:

Virtual Humans Workshop[74]

---

[72] www.ict.usc.edu

[73] A lot of attention has been devoted to creating the models and environments. Both simulations are implemented using the Unreal game engine.

[74] www.ict.usc.edu/∼vhumans/2004/

- Is it more appropriate to construct a frame of analysis that encompasses both user and ECA in a single interaction graph?
- Is it fitting to think in terms of a fixed graph that the user comes to recognize, or is the graph itself a dynamic structure?
- Is it even appropriate to focus on "affordances to act," or is it more fitting to consider cues that influence the mental interpretations that lead to action (e.g., affordances of control, affordances of valence of potential outcomes, etc.)? How does this relate to intrinsic motivation?

This workshop was a follow-up on a seminar in Dagstuhl on Evaluating Embodied Conversational Agents[75], where we discussed the tipics of interaction and affordance in a special interest group. In the *research directions*, I will further discuss an evaluation study that we did on agent-supported navigation in a virtual environment.

Back to our question, how can affordance help us in designing a game? In the *mission rehearsal exercise*, described above, it would be much more easy to have a menu with all available options listed. However, such amenu would defeat the purpose of the simulation, since such menus will not likely occur in real life. Immersion is, in other words, necessary to maintain the emotional involvement with the application, and affordance is the key to immersion. But, although it sounds like an answer, it does rather lead to another question, *how can we define the usability of a game*?

8

usability and fun

In interaction design there is a clear, yet unresolved, tension between usability and fun. Usability is, despite the many disputes, a well-defined notion:

usability (ISO DIS 9241-11)

> ... the effectiveness, efficiency and satisfaction with which specified users can achieve particular goals in particular environments ...

This is the ISO DIS 9241-11 definition, cited from Faulkner (2000). In sectiona 10.3 we will further investigate usability as a means to evaluate systems from an interaction perspective. Now, I wish to focus on why artefacts or games might be

[75] wwwhome.cs.utwente.nl/~zsofi//eeca

appealing even if these same aspects may compromise usability in the traditional interpretation.

In describing a fancy juice squeezer, designed by Philip Starck Norman (2004) observes, following KS, that is:

emotional involvement

- entices by diverting attention – unlike the common
- delivers surprising novelty – not identifiable to its function
- goes beyond obvious needs and expectations – it becomes something else
- creates an instinctive response – curiosity and confusion

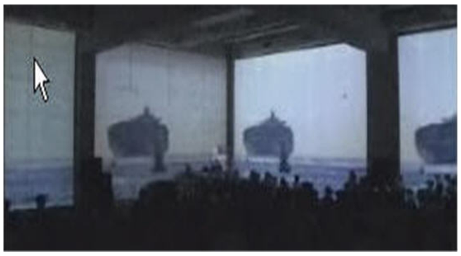
The phrase *satisfaction* in the definition of usability above seems somewhat meagre to explain the emotional involvement with games, and even inappropriate as one realizes that, in the *mission rehearsal exercise*, frustration might actually be beneficial for the learning experience.

9

example(s) – *visual sensations*

The dutch *visual sensations*[76] festival is an annual contest for VJs. In 2005, in cooperation with the festival, a parallel seminar seminar was held discussing the topic of the history of VJ-ing, a aplenary discussion of the relation between club-VJs and the established art circuit. In addition there were two guest speakers, Geert Mul and Micha Klein[77], both visual artists who also have a ten-years experience as VJ.

[76] www.visualsensations.nl
[77] www.michaklein.com




10

Above is another work of Geert Mul, in cooperation with DJ Speedy J. It was shown a dance event in cooperation with Rotterdam Maritime Museum. On the right, the cranes are swinging on the rhythm of the music.

The portfolio of Geert Mul[78] starts with a quote from Arnheim (1957):

form and content

> Very often people assume that "form" as a concept is the opposite of something called "content". This assumption implies that a poem or a musical piece or a film is like a jug. An external shape, the jug, contains something that could just as easily be held in a cup or pail. Under this assumption, form becomes less important than whatever it is presumed to contain.
>
> We do not accept this assumption. If form is the total system, which the viewer attributes to the film, there is no inside or outside. Every component functions as part of the overall pattern that is perceived. Thus we shall treat as formal elements many things that some people consider content. From our standpoint, subject matter and abstract ideas all enter into the total system of the artwork (....)

I totally agree with this. And perhaps this is why I have a preference for artworks that are slightly out of the main stream of tradional art.

research directions– *engaging with fictional characters*

What do you need to evaluate your game or multimedia application? There are many ways to gain insight in how your system is being used, see section 10.3. But if you want to establish functional properties of a multimedia application, for example the effectiveness of using an agent in navigating a virtual environment, in a scientifically more rigorous way, you need to have:

experimental validation

- a theory – in our case: PEFiC
- a test scenario – for example, memory tasks in a digital dossier
- the technology – to realize applications

[78] e.mac.com/geertmul2

In this section, I will briefly describe our efforts in experimentally validating the use of ECAs in virtual environments. As technology, we use our *intelligent multimedia technolgy*, described in sectiona 8.3 and appendix E. So what must be explained is the theory we adopt and the test scenarios we use.

PEFiC is a theory developed by Johan Hoorn and Elly Konijn, to explain *Perceiving and Experiencing Fictional Characters*, see PEFIC. The PEFiC theory may serve as the basis for the experimental evaluation of user responses to embodied agents. In summary, PEFiC distinguishes between three phases, encoding, comparison and response, in analyzing the user's behaviour towards an agent. Encoding involves positioning the agent (or fictitious character) on the dimensions of ethics (good vs bad), aesthetics (beauty vs ugliness) and epistemics (realistic vs unrealistic). Comparison entails establishing personal relevance and valence towards the agent. Response, finally, determines the tendency to approach or avoid the character, in other words involvement versus distance.

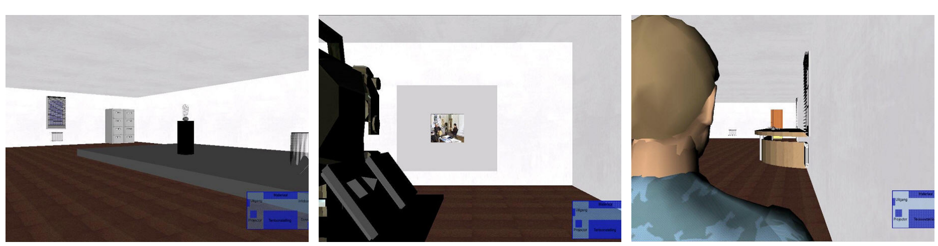
In general, having a virtual environment, there is, for developing test scenarios, a choice between:

validation scenario(s)

- navigation – pure interactivity
- guided tours – using some narrative structure
- agent-mediated – navigation and guided tours

For our application, a virtual environment of an artist's atelier, we have three experimental conditions, navigation without an agent, navigation with a realistic agent and navigation with a cartoon-like (unrealistic) agent. To ensure that these conditions can be compared, the acrtual information encountered when using the application is in all conditions the same.

The independent variable in our experiment, the degree of realism of the agent, corresponds with the epistemic and to some extent the aesthetic dimension of appraisal in the PEFiC theory. As dependent variables we have, among others, user satisfaction, believability, that is estimated usefulness of the agent, and also the extent to which the relevant information is retained.



11

The application is a *digital dossier* for the Dutch artist Marinus Boezem. The spatial metaphor we used for the dossier is the artist's atelier. We created a virtual environment containing a display of the artworks, in 3D, a file cabinet with textual information, a workbench for inspecting the artist's material, and a video projector, with which the user can display a video-recorded interview with the artist.

The actual task to be performed by the user is to learn what constraints do apply to the installation of one of the artworks, *Stone and Feather*:

Stone and Feather

- feather: 70 cm, from ostrich, curved
- stone: 13.5 cm, white marble
- position: alignment with pedestal, no glue
- environment: 50 lux of light max.

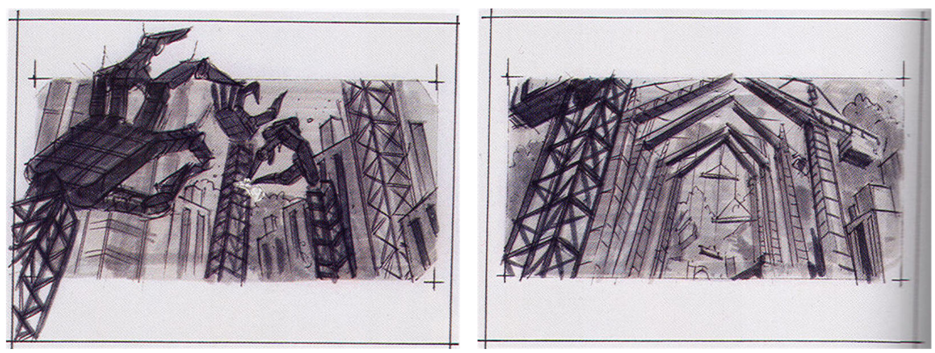
The items mentioned in this list must be reproduced by the user in a subsequent memory test, and in another experiment the user must be able to choose the right materials and reconstruct the artwork.

Our assumption in designing this test scenario was that the gestural nature of positioning the artwork will be favorable for the condition with a gesturing agent, whereas believability will be positively affected by the degree of realism of the agent.

12

9.3 multimedia augmented theatre

In june 2003, I was asked to advise on the use of VR in a theatre production of the Odyssee. Lacking experience in this field, I accepted the invitation to participate with some reluctance, since at the time I didn't have any clue what the VR for the theatre production should look like. Nevertheless, I took the invitation as a challenge and started looking for appropriate hardware, bothering collegues for information on mixed reality art productions, and downloading code to explore software technologies. Many hurdles ware to be taken. We had to deal with organizational issues, such as finding the money for financing the actual production (which unfortunately never came through), finding the right people (students, in our case) to select material and contribute to the code; aesthetic issues, in particular to determine which approach to take to reach at an effective solution; and not in the least technical issues, to realize the production on a sufficiently efficient low-cost platform.



13

context – the Odyssee theatre production

The Odyssee[79]. theatre production was initiated by Ground Control[80], as a successor of previously succesful theatrical spectacles, including an open air performance of Faust[81]. In effect, two performances of the Odyssee were planned, an out-door (external) version, involving real ships at the shore of a lake, and an in-door (internal) version, to be played in temporarily empty office buildings. The in-door version is meant to give a more psychological rendering of the Odyssee Entanaclaz (2003), where the travels of Ulysses are experienced by the audience as a confrontation with themselves. Our contribution was asked for the in-door version, to enhance the experience of the audience with additional VR.

The Odyssee is a wellknown account of the travels of Ulysses leaving Troje, in 24 episodes ending in his return to Ithaca and his reunion with Penelope. The actual theatre production takes 12 parts which are played in 12 successive rooms through which the audience, subdivided in small groups, is guided one room after another for about five minutes per room. Our initial idea was to add information in the form of text and images, to direct the interpretation of the audience towards a particular perspective. In that beginning stage, somewhat optimistically, we

[79]www.odyssee2004.nl
[80]www.ground-control.org
[81]www.faust2002.nl

planned to offer multiple perspectives to each participant, in an individualized manner, dependent on the actual focus of attention of the individual participant.

initial ideas – VR and augmented reality: Our first problem was to find suitable hardware, that is see-through goggles. Searching the Internet gave us the name of a relatively nearby company, Cyber Mind NL[82], that specialized in entertainment VR solutions. Both price-wise and in terms of functionality semi-transparent see-through glasses appeared to be no option, so instead we chose for simple LCD-projection goggles with a (head-mounted) low-resolution camera. This solution also meant that we did not need expensive head orientation tracking equipment, since we could, in principle, determine focus using captured image analysis solutions such as provided by the AR Toolkit[83]. Moreover, captured video feed ensured the continuity and reactiveness needed for a true (first-person perspective) VR experience.

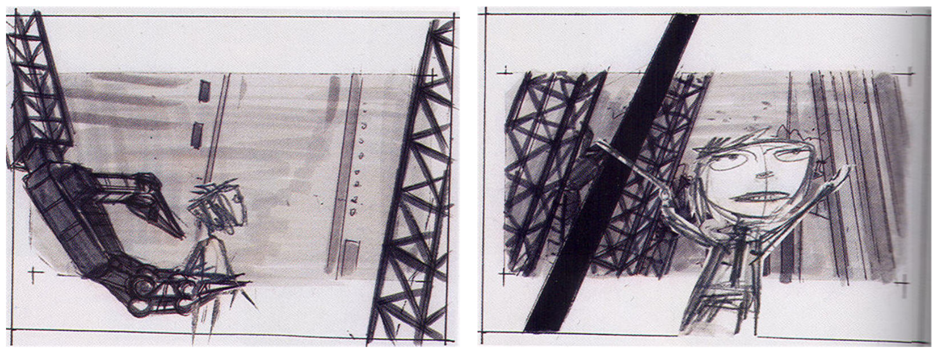
Augmented or mixed reality[84] is an interesting area of research with many potential applications. However, in the course of the project we dropped our ambition to develop personalized presentations using image analysis, since we felt that the technology for doing this in a mixed reality theatre setting is simply not ripe, and instead we concentrated on using the captured video feed as the driver for text and image presentation. In addition, we developed image manipulation techniques to transform the (projection of the) captured video, to obtain more implicit effects, as to avoid the explicit semantic overload resulting from the exclusive use of text and images.

technological constraints – the DirectX platform: After a few experiments with the AR Toolkit, it soon appeared that the frame rate would not be sufficient, on the type of machines our budget would allow for. Moreover, reading the AR Toolkit mailing list, marker tracking in a theatrical context seemed to be more or less unfeasible. So, we shifted focus to the DirectX SDK 9[85], both for video capture and projection in 3D. The DirectX toolkit is a surprisingly functional, and very rich technology for multimedia applications, supporting streamed video, including live capture, 3D object rendering and precise synchronisation between multimedia content-related events. At that time, and still at the time of writing, our own *intelligent multimedia technology*[86] was no option, since it does not allow for using live video capture and is also lacking in down-to-the-millisecond synchronisation.

After exploring texture mapping images copied from the incoming captured video stream, we decided to use the VMR-9 *video mixing renderer* introduced in DirectX 9, that allows for allocating 3D objects as its rendering surface, thus avoiding the overhead of explicit copies taken from a video processing stream running in a separate thread. See section 4.3. Although flexible and efficient,

[82] www.cybermind.nl
[83] www.hitl.washington.edu/research/shared_space/download
[84] www.se.rit.edu/~jrv/research/ar
[85] www.microsoft.com/windows/directx
[86] www.intelligent-multimedia.net

DirectX is a low-level toolkit, which means that we had to create our own facilities for processing a scenegraph, world and viewpoint transformations, and, even more importantly, structuring our mixed reality presentations in time.

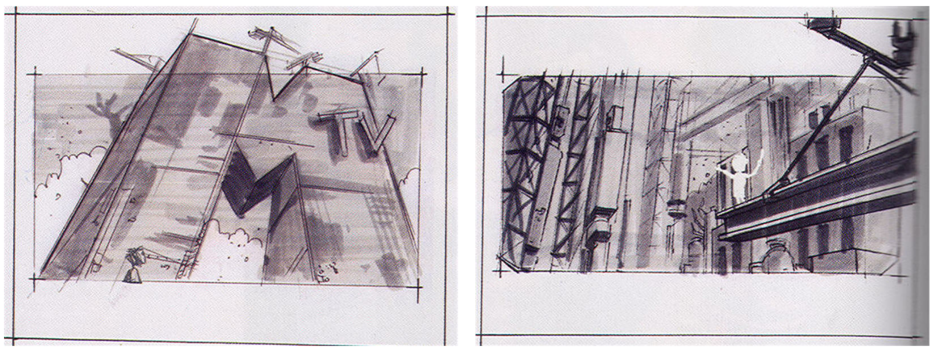


14

structuring time – maintaining 'see-through' aesthetics

One of the problems we encountered in discussing what we conveniently may call the VR with the producer of the Odyssee theatre performance was the high expectancy people have of VR, no doubt inspired by movies as the Matrix and the like. In mixed reality applications, manipulating persons, warps in space, and basically any intensive image analysis or image manipulation is simply not possible in real time. Moreover, there is a disturbing tendency with the layman to strive for semantic overload by overlaying the scene with multiple images and lines of text, thus obscuring the reality captured by the camera and literally blocking the participants view and awareness of the scene. Basically, as a guideline, we tend to strive for 70% visibility of the scene, 20% image or projection transformations and only 10% of information in the form of text and images.

The total duration of our presentation is only 2 minutes, or 118 seconds to be precise. We made a subdivision in 4 scenes, with transitions inbetween, hierarchically ordered in a tree-like structure. Initially, we abstracted from the actual duration, by taking only the fraction of the time passed (in relation to the total duration) as an indication for which scene to display. However, when the development reached its final stages, we introduced actual durations that allowed us to time the sequence of scenes to the tenth of a second. In addition, we used multiple layers of presentation, roughly subdivided in background captured image, the transformed captured image projected on 3D objects, and, finally, pictures and text. These layers are rendered on top of eachother, triggered in a time-based fashion, semi-independent of one another. The frame rate varies between 20 and 30, dependent on the number of images simultaneously used for texturing. Our final mixed reality theatre application may be considered a prototype, awaiting to be put to the test by the audience.



15

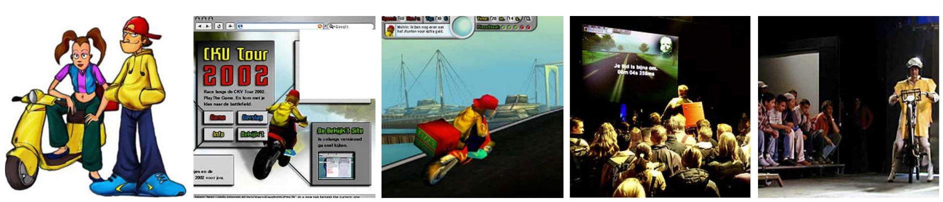
lessons learned – our explorations revisited: Altogether, the development of the mixed reality theatre application has been quite an experience, in multiple ways.

Not in the least it has been (and still is) a challenge to explain the possibilities of mixed reality applications to the layman, that do not take the abstractions we use in our daily academic life for granted.

To be frank, it also has opened my eyes to what some consider 'politically incorrect' technology, in other words Microsoft DirectX, originally developed as game technology, and no doubt a rich toolbox for real life multimedia applications.

Reinventing the wheel is not as simple as it seems. Nevertheless, developing scenegraph processing facilities and the appropriate timing mechanisms for controlling the mixed reality presentation was, apart from being a rekindling of basic skills, a learnful experience.

Finally, even before putting the application to the test, considering the aesthetics of mixed reality theatre productions, it may be called an eye-opener to realize how important reality is, and how meaningless explicit semantics (in the form of text and images) may become. Rather, our explorations were an incentive to further explore more implicit (graphical) modifications of the captured scene to convey meaning.



16

example(s) – *pizza boy*

Did you like to go to a theatre play, when you were sixteen? Peter van Kessel, one of the speakers in our *visual design* develeped a game *pizza boy* to overcome the resistance of secondary school students to theatre. The game, developed by Peter's agency Headland[87], introduced elements of the theatre play in a game setting. The player must deliver pizzas, riding a scooter, and finds him/herself in a situations where he/she must enact a role or perform some actions, related to the theatre play. Visiting the theatre, there is a MIDI-enabled scooter with which the students can play the game, in preparation for the actual play (image on the right in the figure above). Peter reported that the game did indeed help to overcome the scepticism these yougsters had with theatre.

The game was implemented by the Little Chicken Game Company[88], with Virtools. It was available for download at the website of the governmental department that initiated the development of the game, but became such a huge success, also with other people than the original target group, that it had to be taken offline.

research directions– *computational art*

I started studying computer music in 1979. For about four years, I worked on a system for real-time sound synthesis and algorithmic composition. Some years later, I wrote an essay that I presented in 1986 at the First Symposium for Electronic Arts, in Utrecht. The essay was the published in Leonardo, Eliens (1988). Now, almost 20 years later, I reread the abstract, which I OCRed from the original hardcopy:

Computational Art[89]

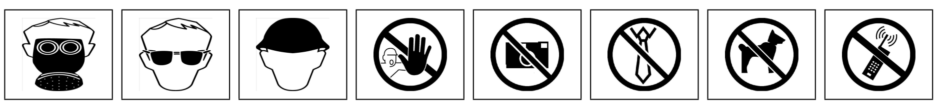
> The author conducts a simple thought experiment investigating the existence and scope of 'computational art': the utilization of the computer in the visual arts and music. In the experiment he sets the task of constructing an artifact that is capable of producing works of art. Since it appears that the artifact needs at least the capability of imagination, he queries the nature of images and imagery and argues that imagination is strongly intentional. Next he introduces the concept of notational systems, since they seem to govern the artistic activity of (not exclusively) machines. Confronted with the question of whether we are able to develop a computational analogue for taste, he finds that notational systems prove to be necessary for mediating the method of production of an artwork and the appraisal of its artistic value. Furthermore, the author shows that there are certain epistemological limits to the creativity of an imaginative device. Although the outcome of this hypothetical construction task clearly denies the possibility of an autonomously creative artifact, there seems to be no reason to worry about the opportunities for computational art: the computer appears to be a

[87] www.headland.nl
[88] www.littlechicken.nl/en
[89] www.cs.vu.nl/~eliens/archive/documents/computational-art.pdf

> unique tool in exploring the possibilities of artistic production, guided by artists.

Today, despite the immense increase in computational power and the overwhelming amount of multimedia peripherals, the computer is still not able to produce art autonomously. But as a tool it is about to take over the entire industry, ranging from traditional publishing to film production and distribution. And, perhaps more tangible, as a tool for the creation of media content it is becoming ever better!

9.4 development(s) – ubiquitous design



17

questions

digital content creation

1. (*) What are the considerations in developing digital content?

concepts

2. What guidelines can you give for the process of design?
3. What is a portfolio? And, what criteria should it meet?
4. What is to be understood by affordance? And, why is affordance important?

technology

5. How would you characterize the following items: logo, sign, collage, story?
6. Characterize the elements of game design.
7. What factors play a role in emotional involvement?
8. Explain how time constraints may be incorporated in the scene graph.

projects & further reading As a project, you may develop a dialog engine for non-linear interactive story telling or a *collage* generator, that produces artworks from a collection of images.

You may further explore the various presentation platforms, and assess the tradeoffs with respect to the support they offer for authoring.

For further reading, I suggest to study interaction design patterns[90]. It is also worthwhile to get some books on modern art, to gain some knowledge about art and design.

the artwork

[90] www.visi.com/∼snowfall/InteractionPatterns.html

1. street logos – images from Manco (2004).
2. photograph of oilpaint box.
3. Mark Veldhuijzen van Zanten – the six roles in their *agency*[91].
4. Mark Veldhuijzen van Zanten – to design for the *salon*, periodic lounge evenings in musea and art institutes.
5. Geert Mul – interactive multimedia installation.
6. Geert Mul – multimedia installation in dutch consulate in India.
7. website of Institute of Creative Technologies[92], showing scenes from *Mission rehearsal Exercise* (MRE).
8. street logos – more images from Manco (2004).
9. website for Visual Sensations[93], a yearly VJ contest in the Netherlands, developed by the agency of mark Veldhuijzen van Zanten.
10. Geert Mul – *Harbour Sound & Vision*, 1999
11. screenshots from virtual atelier of Marinus Boezem.
12. left: *don't spit*, a chines poster against spitting during the SARS period, taken from dutch newspaper; right: filmteckarna, Wiedermann (2004).
13. sketches – from filmteckarna, Wiedermann (2004).
14. sketches – from filmteckarna, Wiedermann (2004).
15. sketches – from filmteckarna, Wiedermann (2004).
16. game – *pizza boy*, developed by Headland[94], see 9.3.
17. signs – health and safety, van Rooijen (2003), p. 258, 259

The artwork for this chapter is meant to emphasize *context*. The *street logos* opening this chapter, as well as the work of Mark veldhuijzen van Zanten and Geert Mul, must be experienced in a context to fully appreciate their meaning.

Also for the MRE application, it is the context, in this case the stress and anxiety of a war situation, that determines the impact. The photorealistic graphic style of MRE, wellknown by the trainees from other games, is meant to strengthen the experience of *immersion*. Notice that the street logos assume an almost iconic character.

[91] 167aardigeontwerpers.nl
[92] www.ict.usc.edu
[93] www.visualsensations.nl
[94] www.headland.nl

10. application development

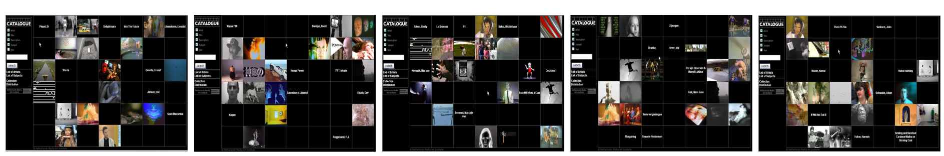
learn the craft, break through the magic of engineering

learning objectives

After reading this chapter you should be able to discuss the multimedia development process, to indicate the need for information system support in the cultural heritage domain, to characterize the notion of digital dossier, to provide solutions for navigating complex information spaces, and to discuss the data representation issues involved.

As you gather from reading this book, the field of multimedia is widely divergent. However, when you develop a multimedia application, you will find that all topics treated so far will become relevant. There will be a need to mix multiple media formats. You will have to find suitable codecs for your video. You will be asked whether search is possible. And, not the least important, you will have to balance navigation and presentation.

This chapter is based on the work we, that is my students, have been doing in the domain of cultural heritage. In the first section, we will introduce the notion of *digital dossier* and outline our general approach. We will then in section 2 look at some examples, and describe how we deploy concept graphs as a universal navigation tool for complex information spaces. Finally, in section 3, we will explore the options for presenting multimedia material and discuss the design issues as well as the technical issues that have arisen in the course of our work.



1

10.1 multimedia casus

You can learn a great deal about technology, but there is no meaning to that unless the technology is applied to produce something worthwhile. In this final chapter, the outline of a *multimedia casus* will be presented, that is a course in which students face the challenge of creating a veritable (intelligent) multimedia information system.

In the studyguide, the course is described as follows.

multimedia casus

> *The assignment in the* multimedia casus *is to develop a virtual environment for some cultural or governemental institute or company. The practicum takes the form a stage, in which external supervision plays an important role.*
>
> *In the* multimedia casus*, techniques learned in previous courses will be applied to create the application. At the start of the course the actual assignment will be determined.*
>
> *Examples of possible assignments are: the development of a virtual exposition hall for the Dutch Royal Museum of the Arts, a virtual city square, which gives information about both the present and the past, a virtual shop, with online buying facilities, or an online broker, which offers facilities for inspecting houses.*

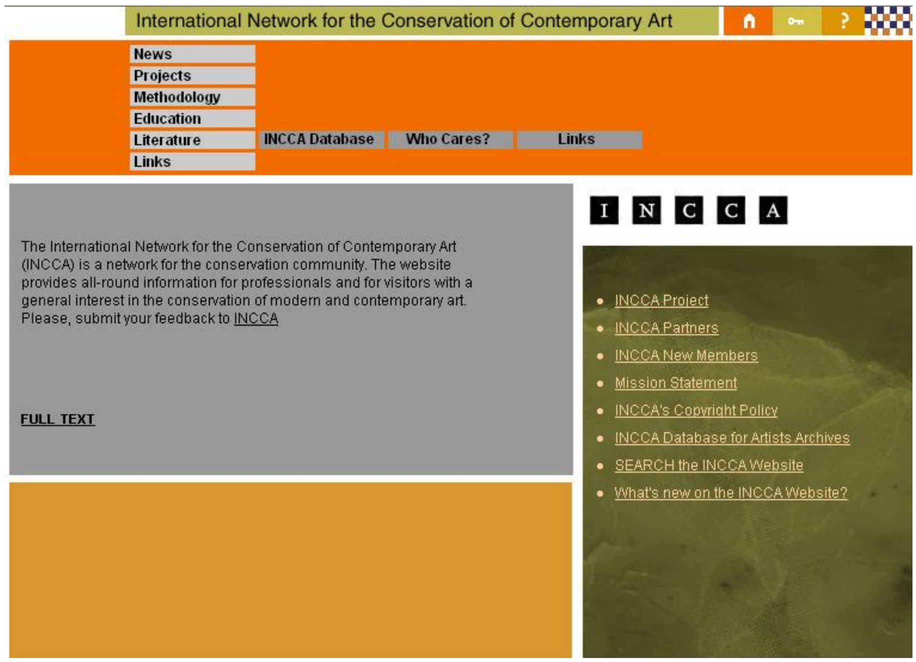
In effect, the availability of a representative of a cultural institute, industry, or governmental department is crucial, otherwise the assignment might easily degrade to the type of toy assignments so common in academia. Now, what is the challenge in such an assignment?

augmented information In the *research directions* of section 8.1 the notion of *augmented virtuality* was introduced to clarify the duality between *information* and *presentation*. More in particular, it was argued that the use of VR makes no sense unless there is some added value, that is by using the rich presentation and interaction facilities that come with this technology.

In an abstract fashion, we may rephrase the assignment as follows:

> Given an information space, create a VR that resolves the duality between information and presentation, using *intelligent multimedia* technology. The VR must offer access to all relevant information entities, organized in a suitable spatial layout, and must allow for presentations from a variety of perspectives, making full use of graphical and rich media facilities.

Below, we will see how this may work out for a concrete assignment.



front page of the INCCA website

2

project assignment – *present a complex information space*

Art is an interesting and complex phenomenon. No art, no culture! Hence, the preservation of collections of artworks is of crucial importance. The ICN (Netherlands Institute for Cultural Heritage) is a government-funded institute for the preservation of (dutch) cultural heritage. ICN gives advice, organises courses, does research, etcetera.

ICN is actively involved in the preservation of modern art, being project leader for INCCA (International Network for the Conservation of Contemporary Art), in the person of Tatja Scholte.

INCCA

> *In 1999, a group of eleven international modern art museums and related institutions applied to the European Commission (Raphael Programme) under the umbrella International Network for the Conservation of Contemporary Art (INCCA). The INCCA project was accepted and work started in January 2000 led by the organiser, the ICN (Netherlands Institute for Cultural Heritage) and the co-organiser, Tate, London.*

The objectives of INCCA are phrased as follows.

objectives

> *INCCA's most important set of objectives, which are closely interlinked, focuses on the building of a website with underlying databases that will facilitate the exchange of professional knowledge and information. Furthermore, INCCA partners are involved in a collective effort to gather information directly from artists.*

The INCCA web site contains a wealth of information about contemporary artists, as well as links to virtual collections of the works of a variety of artists, as for example Mondriaan. The way the virtual Mondriaan collection is presented is interesting in itself. It is a running display with iconic representations of his paintings. The speed of the display varies with the user's mouse movement, and at any time the user may select a painting to obtain more information about it. This particular site suggests where our *intelligent multimedia* approach may fit in.

Returning to the INCCA project once more, as its mission statement we read:

mission

> *INCCA's guiding mission is to collect, share and preserve knowledge needed for the conservation of modern and contemporary art.*

By now, the outlines of our assignment should become clear. Our information space is information about modern and contemporary artists, in the form of digital representations of there work, photographs, audio recordings from interviews and written text. The project assignment is to organize (part of) this material in a virtual environment and to include interaction facilities that highlight particular aspects of this information.

At this stage it would be too ambitious to cover all the material in the INCCA database, so we should restrict ourselves to one or more smaller case studies. The challenge, obviously, is to create presentations with a solid narrative structure and to augment the presented material in a suitable manner, using *intelligent multimedia* technology. What is *suitable*, is part of the challenge!

project management – *roles*

Can the challenge, stated above, be met? Well, there are many ways the project may loose its focus, or fail alltogether. Students should be aware of the fact that the challenge is real and that failure would bring about shame.

Since there are no golden rules for project management, the students themselves are responsible for keeping the project on track. In other words, project management is part of the experience. Here is a checklist.

checklist

- *roles* – create a team
- *project goal* – develop a vision
- *production* – construct the assets
- *quality assesment* – test and control
- *delivery* – present and archive

- *manage* – all along
- *document* – track project's history

The rule of the supervisor should be minimal, as a critical third party. The students work as a group, and they should take responsibility as a group, including the management of the project, assigning roles, and keeping track of progress. In such an approach *intervision* (students supervise one another) is a necessary mechanism in judging the final result of the project.

judgement

- *group* – (2) effort, 5 (product), 3 (documentation)
- *individual* – (4) responsibility, (3) productivity, (3) quality

On a scale of 0-10, both the group result and the individual efforts may be assigned a mark with proper weights, as indicated above. In addition, target deliverables should be defined to assure that the project meets its deadlines and to inspect the nature and quality of the students' work.

deliverables

- *group* – project plan, design, project report, product
- *individual* – detailed weekly account of activities

Dependent on the time available a schedule should be defined indicating when the deliverables should be ... delivered.

schedule

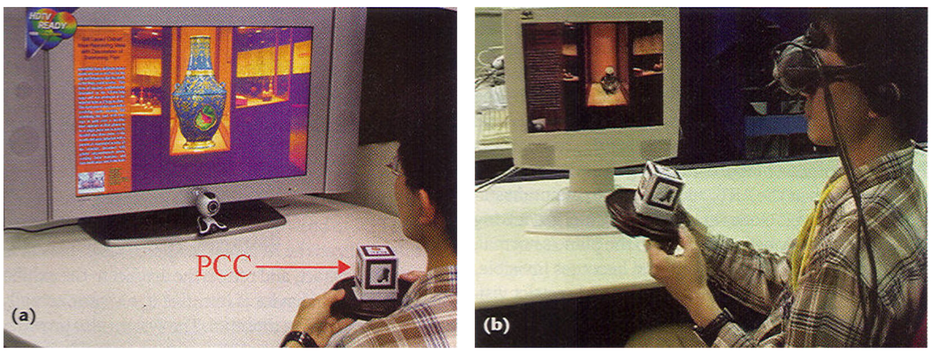
1. project organisation
2. project definition
3. planning and design
4. construction and development
5. integration and delivery
6. presentation and archiving

Is this a realistic setup? It should be. Besides, it is not the supervisor's responsibility, is it? It is first of all the responsibility of the students themselves!

peer reviews – ***to control group dynamics*** Whether you are a student or responsible for supervising projects, this account of how our *multimedia casus* is organized should give you some indications of what you may encounter in a team project. But apart from the organizational issues, you should be aware of the group dynamics, that is the individual relations and clusters of persons that emerge during the process of development. In general it takes some time before the various roles are established, that is who takes leadership, who takes notes during meetings, and who does most of the technical work. Also, sometimes some of the more creative members of the tem are overshadowed by some of the more outspoken ones. Some people simply need to learn to assert themselves!

For a supervisor, it is often quite difficult to assess the contribution of each individual. I remedy this by having both informal peer reviews, in the group, as well as formal peer reviews, where marks must be given for *responsibility*, *productivity* and *quality of the work*. The order in which the students to be reviewed are presented to the reviewer is random, to avoid any bias due to presentation order.

I started using peer reviews about ten years ago, then informally, because I noticed that students could be much more direct in their criticisms that I dared to be. Recently, following suggestions from my collegue Johan Hoorn, who is an experimental psychologist, we formalized the procedure and introduced peer reviews for other courses as well, including the *visual design* course.



physical control cube — head-mounted display

3

example(s) – *tangible virtual museum*

Don't touch that! Keep your hands off! This is what you often hear parents shouting at their children in a museum or gallery exhibit. More often, however, precious artifacts, ceramics, porcelain or bronzes, are stored away in glass showcases, precluding any kind of physical interaction, and many times a proper look as well.

To remedy this situation, researchers from the Academia Sinica and the National Cheng Kung University have developed a *tangible photorealistic virtual museum*, a system for real-time interaction with photorealistic museum artifacts, which allows for an immersive experience using tangible interfaces, in the form of a 3D control cube (image left), Rosenblum and Macedonia (2005). The display is a kiosk-like system showing a panoramic view of the exhibit, augmented with a collection of perspective photographs for each object. The user may examine any of the objects by using a handheld control cube (PCC) to control size and rotation of the object.

As indicated, the system is not 3D mesh-based but image-based, which allows for high resolutions on mid-range platforms, which would not be feasible according to the authors, when using 3D modelling techniques.

research directions– *metaphors and interaction style*

Given a problem statement as the one above, to present information about contemporary artists, how would you proceed? You might start by asking potential users, or stakeholders, how they would like the system to be. The answer you will get this way is likely to be disappointing. They will probably tell you that it must be like something they already know. So it might be better to rely on your own intuition and find a creative solution by choosing a fitting metaphor.

Let me give an example. In creating the digital dossier, a notion that will be explained in the next section, for the artist Marinus Boezem, as presented in the *research directions* of section 9.2, we choose the artist's atelier as a metaphor, and we used the spatial layout of the atelier as an organizational principle for presenting the information. In this, indeed very naturalistic, approach, we used pedestals to present the artworks, a file cabinet to present the textual information and a video projector to present the video recorded interview with the artist. The extent to which the virtual atelier does represent the artist's atelier faithfully is not important, in this context. What is important is whether the spatial metaphor did function as a valid organizational principle for presenting the information.

Instead of arguing whether this is the case or not, or whether the graphics chosen were right, etcetera, I would rather like to refer you to the literature, so that you can investigate the issues involved yourself.

In Preece et al. (1994), it is observed that interface metaphors act as conceptual models to support particular tasks. For office tasks, for example, we have the wellknown *desktop metaphor*. Preece et al. (1994) lists a number of such metaphors, for a variety of application domains:

application area	metaphor	familiar knowledge
operating environment	desktop	office tasks
spreadsheets	ledger sheet	columnar table
object-oriented environment	physical world	real world
hypertext	notecards	organization of text
learning environment	travel	tours, guides, movement
file storage	piles	categorizing
multimedia environments	rooms	spatial structures
cooperative work	multi-agents	travel agents, servants

In the most right column it is indicated why the metaphors should work, assuming real world situations that we are familiar with.

In some cases it is necessary to speak of a *composite metaphor*. For example, scrollbars are not easily to be found on your natural desktop. Form a cognitive perspective then, we may speak of multiple mental models.

When we look at what interaction styles are supported from a more technical perspective, we have following Preece et al. (1994), the following options:

interaction styles

- command entry
- menus and navigation
- forms fills and spreadsheets

- natural language dialog
- direct manipulation

However, each of these interaction styles may somehow be incorporated in the representation that we adopt for our metaphor.

2D vs 3D Surprisingly, each year that I start with another *multimedia casus* group, there is a discussion whether the application should be in 2D, using traditional web technology or *flash*, or 3D, using VRML or any other suitable 3D technology. My answer to the students objections, which can partly be explained by the fact that they fear the complexity of 3D, is flatly that anything that can be done in 2D can be done in 3D. But looking at the list of interaction styles above, I am tempted to add that a 3D representation allows for a more rich repertoire of interaction styles, such as spatial navigation. It would be interesting to investigate to what extent the interaction styles used in game playing can be incorporated in 'more serious' applications.

10.2 digital dossier(s)

After a first round of the *multimedia casus*, in which the students produced an application giving an overview of the INCCA information archive, the participants, but only incidental information about the artists and their artworks, we decided to focus on case studies of individual artists, and we introduced the notion of *digital dossier*:

digital dossier

> *Create a VR that realizes a* digital dossier *for a work of a particular artist. A* digital dossier *represents the information that is available for a particular work of art, or a collection of works, of a particular artist. The* digital dossier *should be* multimedia-enhanced*, that is include photographs, audio and other multimedia material in a compelling manner.*

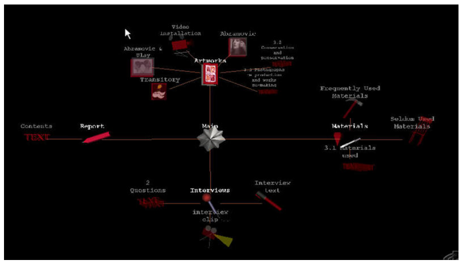
Like a medical dossier, the *digital dossier* was meant to give the information about the artist and the works of art readily at hand, so that it could effectively be used for the task of conservation and the re-installation of the artworks.

Since we were in doubt whether the phrase *dossier* actually existed in the English language, we looked it up in a dictionary:

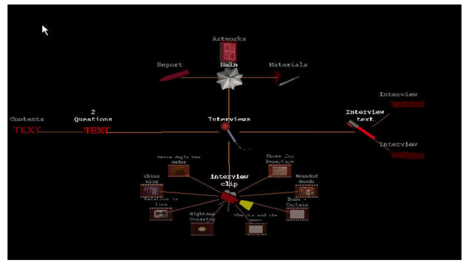
Webster New World Dictionary

- dossier (dos-si-er) [Fr < dos (back); so named because labeled on the back] a collection of documents concerning a particular person or matter
- archive – 1) a place where public records are kept ... 2) the records, material itself ...

We chose for the phrase *digital dossier*, and not for archive or library, to stress that our focus lies on presentational aspects. Although issues of data representation and content management are clearly important, our primary interest was with issues of presentation and navigation.



main node



interviews

4

the *abramovic dossier*

For the 2004 autumn group, we decided to take the work of Marina Abramovic, a serbian-dutch artist who became wellknown in the seventies with performances with her partner Ulay, and has since then produced numerous installations, videos and performances with what I would like to call 'high existential impact'. The directive with which the students where set to work was, quoting Ted Nelson:

> *everything must be highly intertwinkled*

Since virtual museums are by now a common phenomenon, and the virtual atelier for Marinus Boezem may be considered to be just a variant of this, the 2004 autumn group decided to explore alternative ways of presentation and navigation.

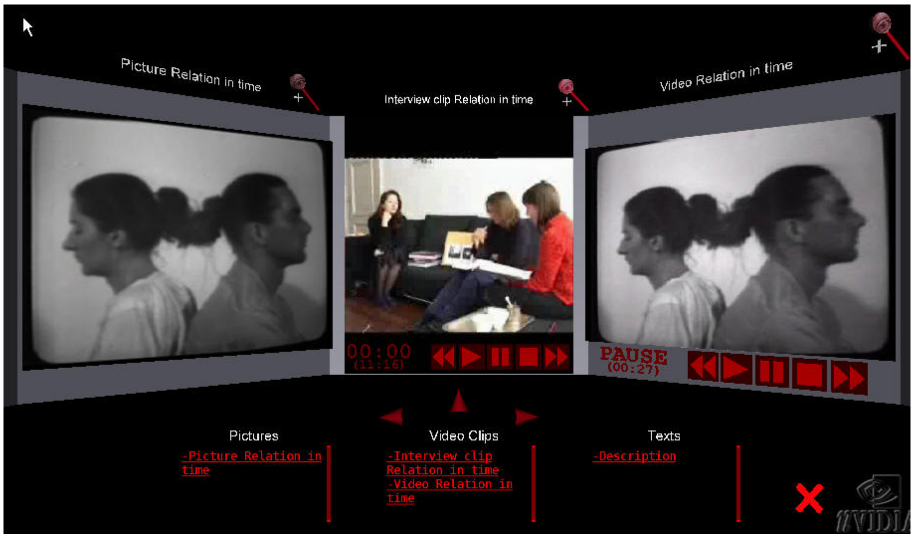
As material for the *abramovic dossier* there was an interview with Marina Abramovic from ICN, made in cooperation with the Dutch Foundation for the Visual Arts, and a great collection of videos from Montevideo. In addition, a transcription of the contents of the interview made by Michela Negrini, a student of media art at the University of Amsterdam, who also provided an interpretation as well as a categorization of the works of art. Given the material and the categories along which this material was classified, the students decided to explore the use of concept graphs as an instrument for navigating the information space.

navigation – *concept graphs*

The reader has already encountered concept graphs in chapter 1, when the notions of multimedia, medium, television and communication were explained by indicating their relations to other concepts.

Concept-relation graphs are a familiar tool in linguistics and have also been used for a long time in Artificial Intelligence to describe the semantic relationships in complex domains. As a navigation instrument it is, to my knowledge only used in a kanji learning tool[95] and the Visual Thesaurus[96].[97]

[95] www.rikai.com/perl/KanjiMap.pl?
[96] ualthesaurus.com
[97] The Visual Thesaurus allows also for invoking Google image or document search from any of the elements of the concept graph.



presentation of video clips from Marina Abramovic

5

After the initial idea was there, one of the students of the group, Olaf van Zon, an AI student, managed to get a first version of a 3D concept graph working in VRML. This prototype implementation demonstrated the potential of the concept graph as a navigation instrument in the *abramovic dossier*.

presentation – *gadgets*

The original idea of presenting information, that is the actual interview, the videos and images of the works of art, as well as the textual information, was to use *rooms*, where the information could be projected on the walls. The *room* metaphor, which obviously stems from the virtual museum approach, did however not seem appropriate since it conflicted with the concept graph used for navigation. After some discussion, information rooms were abandoned in favor of *information gadgets*, that could be expanded from and collapsed into the concept graph.

In the original *abramovic dossier*, the presentation gadget consists of three panes that can simultaneously show a video of the work, the interview, that is the fragment in which Abramovic speaks about that particular work, and the textual information related to the work and the interview. However, it appeared that in some cases there was not enough information, because the work was not spoken about in the interview, and in other cases there was too much information, for example multiple recordings or text documents. It was then decided to extend the presentation gadget with lists of alternative material that the user could select from and direct to one of the panes for further inspection.

To enable the user to focus on one of the panes, for example to get a better view of the video material a zoom in/out button was provided. All these enhancements, however, did complicate the interaction, as became clear when the *abramovic dossier* was presented at Montevideo.

In the course of the project, another interesting presentation feature was added, namely the reconstruction of one of the video installations in 3D, incidentally demonstrating the advantages of using 3D.

reconstruction – *recreating the installation*

In discussing the *abramovic dossier* with Bart Rutten from Montevideo, who provided us with all the video material, another project was mentioned which was concerned with 3D-recordings/models of existing installations. Having full confidence in the technical capabilities of my students, I promised to show that such a reconstruction of an installation would naturally fit within our approach.

Reconstruction of Terra della Dea Madre *in VRML.*

6

The installation for which the reconstruction was made is *Terra dea degli madre*, and installation with two chairs and a television, which was exhibited in the Stedelijk Museum of Amsterdam, in 1986. As a starting point, we took a video produced at the time of the exhibition, which shows the installation in an exposition room in the Stedelijk Museum, and which contains, apart from comments from Abramovic, also the video shown on the televison in the installation.

At this point, we can only speculate how useful such a reconstruction can be as a tool for the conservator responsible for the re-installation, to play around with the presentation parameters, the positioning in space, the overall size, light and ambient effects.

style issues – *how to improve the dossier*

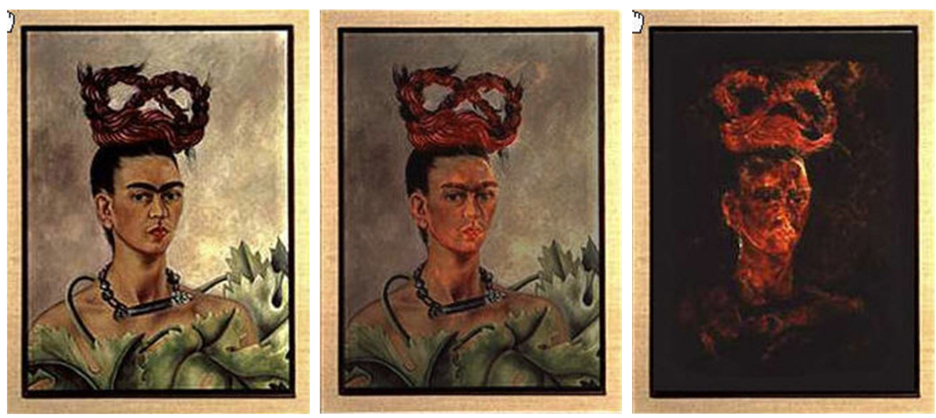
The *abramovic dossier* does also provide a facility for search, as well as online help. However, as already mentioned, when demonstrating the application to the interested parties, that is ICN and Montevideo, a number of issues came along, that I will here summarize as a list of questions:

style issues

- what icons should be used to identify the elements of the concept graph?
- what categories and relationships are most appropriate?
- how should the information be displayed, simultaneously or more focussed?

- how do we allow the user to choose between multiple information items?
- how do we avoid visually disturbing elements?

Obviously, although the *abramovic dossier* was very positively received, these issues must be dealt with to make it a success. Having a first prototype, we need to rethink our application, not only with regard to its style of presentation, but as we will discuss in section 10.3, also in terms of its underlying data representation.



no light | half light | full light

7

example(s) – *conservator studio*

Ever thought of becoming a conservator? Seattle Artmuseum's Conservator Studio[98] gives you the opportunity to explore this career options:

> Explore four paintings from the Mexican Modernism exhibition through the eyes of a conservator (what's a conservator? you'll find that out too!). You'll have a new perspective on the paintings as well as how they are handled and prepared for display.

The illustrations above show what occurs when manipulating *transmitted light* on the painting *Self-Protrait with Braid*, oil on canvas, from the Mexican painter Frida Kahlo. As explained in the accompanying text: *when a light is shone through this painting one can see that the hair and the flesh areas are painted with thin layers of paint.*

These series of images are part of an interactive *flash* application developed by the Seattle Artmuseum to engage the general audience in the conservation of art, and to arouse an interest in art in general. The application allows the user to experiment with the various techniques used for the analysis and conservation of oil paintings.

[98] www.seattleartmuseum.org/exhibit/interactives/mexicanModernism/enter.asp

research directions– *establishing usability*

In the March 2005 volume of CACM, an assessment is given of the current state of *user-centered design* practice. User-centered design is, quoting UCD, *a multi-disciplinary design approach based on an active involvement of users to improve the understanding of user and task-requirements, iterative design and evaluation.* In the article, which is based on a survey among user-centered design practitioners, user-centered design is claimed to have been beneficial for, among others, customer satisfaction and enhanced ease of use. Other measures mentioned are mostly relevant for e-commerce applications, which, as the authors observe, *have greatly bolstered the the appeal of usability and user-centered design, as users can take their business elsewhere with just one mouse click.*

In our case, the competition is fortunately less threatening. Nevertheless, usability issues such as legibility of text, ease in navigation and adequate task support are equally relevant. As a first step after completing the *abramovic dossier*, we have developed a test-plan and a sample task, and (the students) executed two test-sessions with participants from ICN and Montevideo, who where asked to work with the system thinking aloud. The test-sessions were recorded on video, and the participants were requested to complete a questionnaire.

In UCD, a list of approaches is given, which were reported to have been used by the respondents of the survey:

user-centered design methods

> field studies, user requirement analysis, iterative design, usability evaluation, task analysis, focus groups, formal/heuristic analysis, user interviews, prototype (without user testing), surveys, informal expert review, card sorting, participatory design

The three most frequently used methods in this list are, respectively, iterative design, usability evaluation and task analysis. These three methods were also considered to be important by the respondents. Frequently used, but not considered to be as important, were informal expert reviews. And less frequently used, but considered important, were field studies. This distinction can, according to UCD, attributed to cost-benefit trade-offs, since clearly field studies are much more costly.

Usability evaluation looks, according to Preece et al. (1994) to issues such as:

usability evaluation

- *learnability* – time and effort to reach level of performance
- *throughput* – the amount of work done
- *flexibility* – accomodating changes in the task
- *attitude* – of users to the system

To conclude this section, let's take a closer look at task analysis.

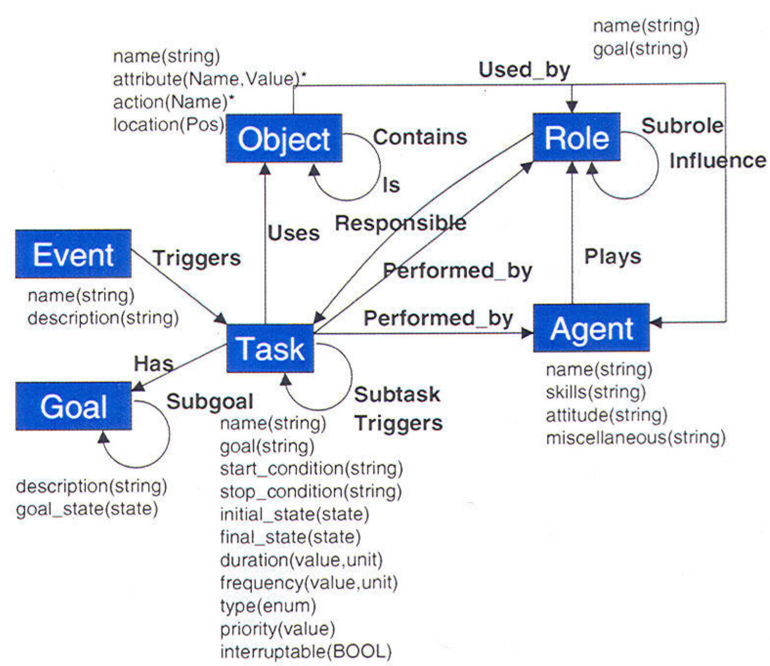
task analysis Task analysis may be characterized as the decomposition of a task into subtasks or steps, to arrive at a sufficiently detailed description of the task and its relation to the environment.

In Welie et al. (1998), a description is given of what might be understood as the task world ontology, the concepts and relations that play a role in performing a task analysis. The main concepts figuring in the task world ontology are, following Welie et al. (1998):

task world ontology

- *task* – activity performed by an agent to reach a certain goal
- *goal* – a desired state in the task world or system
- *role* – a meaningful collection of tasks
- *object* – refers to a physical or non-physical entity
- *agent* – an entity that is considered active
- *event* – a change in the state of the task world

As indicated in the diagram above, these concepts are related in various ways. Example relations include *uses*, *triggers*, *plays*, *performed_by*, *has*, etcetera.

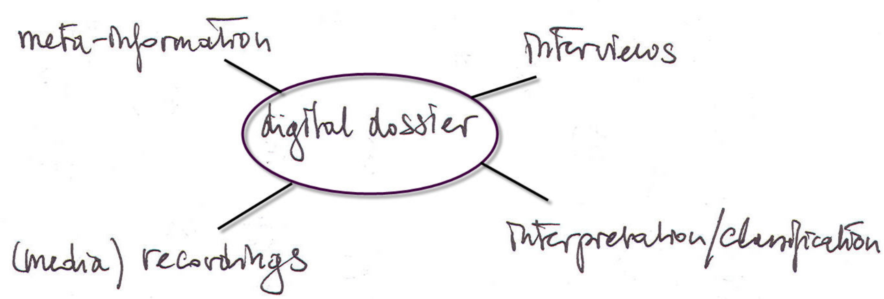


8

Creating a task model based on this, or a similar, ontology may help us understand what a user needs to accomplish and how this may be supported by an information system. As such, creating a task model should be considered to be an essential ingredient of the software engineering life cycle, Eliens (2000).

10.3 representation & interaction

In re-thinking the *abramovic dossier*, we first needed to re-establish what are our goals in developing this application and what are our primary data sources. The goal, first of all, is to support conservators in their task of preserving contemporary art, and to assist them with the re-installation of such artworks.



Our primary data sources are *meta-information*, coming from the INCCA initiative, and video-recorded artist *interviews*, which were initiated by ICN as a means to record information about contemporary art that would otherwise be lost. In addition we media-material, including images and video, that may be regarded as *recordings* of the works of art, as well as the textual *interpretations* and classifications that exist, or may be constructed from this material.

At this point, I may remark that one of the pitfalls in creating a dossier is to get trapped in the visually salient features of the dossier, the presentation of the artworks themselves, and forget about the primary focus of the dossier, to make all information accessible in an adequate manner.[99]

For our next generation of digital dossiers, we decided to take the following steps:

next generation dossier(s)

1. adaptation of representation to Dublin Core (+ annotation needed for presentation)
2. XML-based content management, with php forms (extending what we have now)
3. there should also be a possibility to present the information and material in a 'plain' web format
4. as well as in (a new version of) 3D dossiers
5. we should think about the proper presentation parameters.

Dublin Core is the standard used in the INCCA initiative, to record meta-information about existing information sources. See section 3.3 for a description of the Dublin

[99] For many cultural heritage applications, which aim to present art to the layman, presenting the artwork is the primary focus, and giving access to the information context generally comes second.

Core element set and the Resource Description Framework (RDF) on which it is based.

For the *abramovic dossier*, a collection of record-like structures was developed, together with a simple content-management tool, written in PHP. This content-management system must be adapted to be compatible with the Dublin Core-based resource descriptions.

Further, we decided that, along with the 3D presentation of the dossier, it would be worthwhile to develop a conversion tool that produces standard web-technology based presentations as well. This approach allows us to assess the usability merits of the 3D dossiers in a comparative way.

Finally, as I indicated before, an important issue that must be resolved concerns the proper presentation parameters. What do we present to the user? And how do we allow the user to interact with the material presented?

content management and data representation

For developing the *abramovic dossier*, we have a fixed number of record-like structures:

structures

- Video – to display video fragment, including interviews
- Picture – to present pictures of the artwork
- Artwork – contains all information connected to a work of art
- TextItem – to present text, from the interview or any other source
- MaterialItem – to present information about material(s) used
- GroupNode – to combine nodes in the concpet graph
- Information – acts as the outer container for all nodes

All these structures support a set of common attributes, including *shortName*, *longName*, *ID*, *connectedNodesIDs*, and *description*. In addition the *Video*, *Picture* and *Picture* have fields allowing to show a preview image. And the *Video*, *Picture* and *TextItem*, also have a *url* field giving access to the actual information item.

The *Information* and *GroupNode* structures are used for creating the top-levels of the concept graph, whereas the other structures, such as the *Video* and *TextItem* give access to for example a fragment of an interview and its transcription.

Below an example is given of the data underlying the concept graph of the *abramovic dossier*:

concept graph

```
Information {
 informationNodes [
  GroupNode {
      ID "MAIN"
      shortName "Main"
      longName "Main"
      urlModel "models/conceptGraph/main/modelMain.wrl"
```

```
        description [ "Central information node" ]
        connectedNodesIDs [ "ARTWORKS", "KEYWORDS",
                           "INTERVIEWS", "REPORT" ]
        }
    GroupNode {
        shortName "Artworks"
        longName "Artworks"
        description [ "Node that connects to all the artworks" ]
        ID "ARTWORKS"
        connectedNodesIDs [ "MAIN", "TRANSITORY",
                  "ULAY", "VIDEOINSTALLATION", "ABRAMOVIC" ]
        urlModel "models/conceptGraph/artworks/artworksGroup.wrl"
        }
        # # ...
  ]
  }
```

The *Information* node collects all available nodes, and takes care of connecting the individual nodes, based on the information specified for each node.

As an example of an *Artwork* node, that is an element of the list of nodes in the *Information* node, look at:

```
Artwork {
  shortName "Terra degla Dea Madre"
  longName "Terra degla Dea Madre"
  description ["15:40 min, colour, sound."]
  ID "AV24"
  connectedNodesIDs ["VIDEOINSTALLATION", "DTV24",
        "TTV24", "PV24", "CV24", "VV24", "G0"]
  urlPreviewImage "images/previewImages/AV24.jpg"
  widthPreviewImage 479
  heightPreviewImage 349
}
```

This node is connected to many other nodes, giving access to the information items that belong to it, such as the video clips of the interview, shown below.

```
Video {
  ID "CV24"
  shortName "Interview clip Terra degla Dea Madre"
  longName "Interview clip showing Terra degla Dea Madre"
  url "interviewclips/interview_terra_degla.avi"
  width 320
  height 360
  urlPreviewImage "images/previewImages/interview_terra_degla.jpg"
  widthPreviewImage 320
  heightPreviewImage 240
  description [""]
```

```
  connectedNodesIDs ["CLIP", "AV24"]
}
```

In the *url* field of this declaration, the actual video file is indicated, which should be displayed at a resolution of 320x360, as specified in the *width* and *height* fields.

And finally, as an example of a *TextItem*, consider:

```
TextItem {
  shortName "Instruction"
  longName "Green Dragon Lying instructions for the public."
  description ["Text explaining the way the public has to interact with the
    artwork."]
  ID "ITO05"
  connectedNodesIDs ["AO05", "INTERACTION"]
  url "text/AO05_instruction.txt"
}
```

For constructing the *abramovic dossier*, Tim Verweij developed the content management tool, that allows the user to browse and edit existing nodes, and to insert new nodes into the graph.

integration with the Dublin Core

The Dublin Core is a general resource description formalism, that allows for specifying resources in a variety of domains. See section 3.3. For INCCA the Dublin Core was chosen, not because it is the most suitable formalism, but because it may serve as the least common denominator, and agreement on anything else simply seemed to be impossible. As a reminder, the Dublin Core provides the following elements:

Dublin Core[100]

- *title* – name given to the resource
- *creator* – entity primarily responsible for making the content of the resource
- *subject* – topic of the content of the resource
- *description* – an account of the content of the resource
- *publisher* – entity responsible for making the resource available
- *contributor* – entity responsible for making contributions to the content of the resource
- *date* – date of an event in the lifecycle of the resource
- *type* – nature or genre of the content of the resource
- *format* – physical or digital manifestation of the resource
- *identifier* – unambiguous reference to the resource within a given context
- *source* – reference to a resource from which the present resource is derived

[100]dublincore.org/documents/dces

- *language* – language of the intellectual content of the resource
- *relation* – reference to a related resource
- *coverage* – extent or scope of the content of the resource
- *rights* – information about rights held in and over the resource

Descriptions of items in the *digital dossier* should incorporate these elements, together with the attributes needed for the insertion of items in the concept graph and the presentation parameters, that are necessary for displaying the (media) material. Technically, the namespaces supported by RDF does allow for merging these different types of annotations. However, the challenge here is to derive the presentation attributes automatically, and to come up with a reasonable default for inserting these items in the concept graph.

location of *Tower of Babel* project

9

intelligent guidance – I-GUARD

Although digital archives or digital libraries[101] are by no means a new phenomenon, our concept of *digital dossiers* contains a number of innovative elements. A digital dossier provides a unified information and presentation space. In this sense it differs significantly from a digital archive with a traditional web interface, where navigation and presentation are distinct. Digital dossiers allow to a much greater extent for an immersive experience of the information related to works of art. As such it is reminiscient to explorations in *virtual archeology*[102], our to our notion of *virtual context*, presented in section 8.1.

Working out the issues indicated above, that is the integration with the Dublin Core and providing suitable content management, is a matter of diligent software engineering. But what can we further do to support the construction of digital dossiers and improve the usability of such dossiers? And what are the scientific issues, worth to be investigated?

To indicate the research issues, let me first expand the cope of our project and re-define the goal of our research:

I-GUARD

[101] www.ifla.org/II/etext.htm
[102] library.thinkquest.org/18261/?tqskip1=1

> Contemporary art is an intrinsic part of our cultural heritage. Installations, performances, video and other forms of media art, as for example *web art*, have the interest of a small group of adherents, but are in comparison with more traditional art forms, far more difficult to present to a general audience. Another problem presents itself, due to the type of materials used and the context-specific aspects of these art forms, in the conservation of the works.
>
> In our research we address the issue of providing access to these contemporary art forms from a wide variety of perspectives, ranging from the interested layman to the expert that has to deal with archiving, conserving and the possible re-installation of the art works.

The acronym I-GUARD stands for *Intelligent Guidance in Artist's Digital Dossiers*, and refers to a project the aim of which is to arrive at a general framework for artist's digital dossiers, that provide intelligent guidance to both the expert user, responsible for the future re-installation of the work(s), and the interested layman, that wishes to get acquainted with a particular work or collection of works. In general, there are two techniques that we can apply to provide such guidance:

intelligent guidance

- filtering the information space according to the user's perspective, and
- intelligent agents, that (pro) actively aid the user in searching the information space.

Filtering the information space may be done by using techniques from formal linguistics to restrict the concept graph that defines the navigation structure, that is by stating assumptions with respect to the relevance of particular (linguistic) categories or elements from a user's perspective. Intelligent agents is an approach stemming from artificial intelligence which allows for providing guidance in a variety of ways, possibly even in an embodied form using a face or humanoid figure to give suggestions to the user on what interactions to perform. With the latter type of guidance we have already experimented in the Marinus Boezem dossier, as described in section 9.2. So let's look at what natural language technology has to offer.

natural language: Having a concept graph as a generic navigation device, it still remains a problem how to fill the concept graph with meaningful content, and how to indicate meaningful relations between the concepts and aspects covered by the nodes of the concept graph. In the *abramovic dossier* this was done by hand, based on information derived from a transcripion of an interview with the artist. (provided to us by ICN). Interviews with artists is one of the means ICN deploys to gain knowledge needed for the conservation of contemporary artworks. Such interviews provide a rich source of textual information, that includes both general viewpoints on the artist's oeuvre as well as specific constraints that adhere to the (re) installation of the work(s) of art.

What we should strive for is to derive both structure and content of the concept graph for a particular dossier (semi) automatically. Using a basic lexicon of terms and phrases related to contemporary art we should be able to generate a representation of the textual information that may serve as a basis for constructing the concept graph. This representation must contain an enumeration of the concepts, the relation between occurrences of concepts, as well as a reference to the work(s) of art to which the concepts apply.

Natural language processing technology may not only serve for the static analysis of the material, when the digital dossier is created, but also dynamically when the dossier is being used, to aid the user in finding relevant information. Research issues here are, on the one hand, the interpretation of user input (that is, loosely structured natural language), and on the other hand, filtering the concept graph representing the information space in such a way that it adequately reflects the user's interest or perspective.

In summary, from a research perspective, digital dossier(s) concern the following issues:

digital dossier(s)

- representation of information of one or multiple works of art,
- presentation of that information in a *rich media presentation environment*,
- intelligent navigation and interaction, and
- support for interaction with loosely-structured natural language.

And to conclude, *digital dossiers* will on the one hand contribute to making contemporary art forms accessible to a larger audience and on the other hand are explicitly meant to support the complex task of the conservation and reinstallation of works of art in an effective manner.

outside view of *Tower of Babel* project

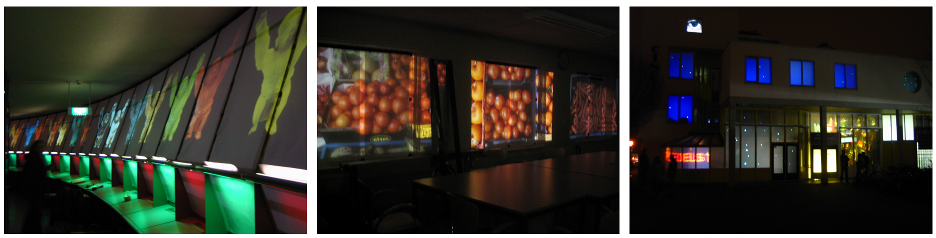
10

example(s) – *Tower of Babel*

In the *Tower of Babel*[103] project, shown above, multimedia material was projected from within buildings, on the windows, to the outside. Local citizens in a neighbourhood in Amsterdam where approached to submit material that expressed

[103] www.torenvanbabel.info

their emotions of daily life, with the question *what moves you*. The text and photograms could be submitted either by email or SMS. Also workshops were held, during which particpants could develop material. This material was then edited and prepared for projection, using 40 carroussel dia-projectors, taking about 2000 images, and six beamers projecting images and video. Also sound material, that was collected in the same manner, was being used during the projection.



Inside view of Tower of Babel *project.*

11

The centre of the location[104], a somewhat impoverished neighbourhood near the centre of Amsterdam, is a building dating from 1926, originally an antroposofic temple, that once served as a cinema for avant-garde movies, and is now being used as a library. The buildings surrounding it are, if not split up into apartments, being used as a local youth centre, a city archive and another library.

research directions– *media art*

In a recent symposium on the preservation of contemporary media art, a number of institutions presented their projects, ranging from more technical topics, such as the *conservation of videotapes*[105] and the *mass storage of digital material*[106] to the conceptual issues in *capturing new media*[107], the variety of *media formats*[108] and the need to record and maintain *meta data*[109] about the artworks and related information.

To get an idea what the phrase *media art* encompasses, have a look at the circumscription given in the WikiPedia[110]: *new media art* is a generic term used to describe art related to, or created with, technology invented or made widely available since the mid-20th Century, including technology stemming from telecommunications, mass media and digital modes of delivery the artworks Below, the disciplines that belong to this form of art are listed, together with their entries in the WikiPedia, in an abbreviated form:

(new) media art

[104] www.alphons.net/panos/tolstraat.html
[105] www.montevideo.nl/en/pdf/CONSERVERING_1tm80.pdf
[106] www.ichim.org/ichim03/PDF/128C.pdf
[107] www.v2.nl/Projects/capturing/summary.html
[108] www.variablemedia.net
[109] www.incca.org
[110] en.wikipedia.org/wiki/New_Media_art

- *audio art* – no definition available
- *computer art* – any art in which computers played a role in production or display of the artwork.
- *digital art* – art created on a computer in digital (that is, binary) form.
- *electronic art* – entry to game producer, should be Leonardo[111].
- *generative art* – art or design generated, composed, or constructed through computer software algorithms, or similar mathematical or mechanical autonomous processes
- *hacktivism* – the writing of code, or otherwise manipulating bits, to promote political ideology
- *interactive art* – a piece of art that involves the spectator in some way.
- *internet art* – art or, more precisely, cultural production which uses the Internet as its primary medium and, more importantly, its subject.
- *performance art* – art where the actions of an individual or a group at a particular place and in a particular time, constitute the work.
- *robotic art* – page does not exist
- *software art* – is an intersection of two almost non-overlapping realms: software and art.
- *video art* – is a subset of artistic works which relies on "moving pictures" and is comprised of video and/or audio data.
- *video game art* – involves the use of a computer game for the creation of a digital artwork.

By the nature of the WikiPedia, to which every user can contribute entries, this list nor the defining entries are by any means authorative. Nevertheless, it does provide an overview and may serve as a starting point for further research.

10.4 development(s) – hybrid multimedia applications

12

questions

application development

[111] mitpress2.mit.edu/e-journals/Leonardo

1. (*) What information system support is needed in the domain of cultural heritage? Explain what considerations playa role in developing multimedia applications in this domain.

concepts

2. What phases do occur in the multimedia application development process?
3. Characterize the notion of digital dossier.
4. Explain the distinction between navigation and presentation, and discuss possible solutions for combining them.

technology

5. What elements do you include in your checklist when you set up a project?
6. What is a concept graph? And, how may it be used for navigation?
7. What structures do you need to represent the information in a cultural heritage application? Describe what descriptive features these structures must have.
8. What elements does the Dublin Core have? How can the these elements be integrated with for example the descriptive features of video?

projects & further reading As a project, develop a data format for text, images and video in XML, and implement stylesheets in XSLT to convert the format for display, for example in HTML frames or using SMIL.

You may further explore the formulation of criteria for selecting software and tool support for developing multimedia applications.

For further reading I suggest, apart from the manuals and learning materials that come with your tools, to study example projects and in particular the workflow, that is the dependencies between stages in the production, as for example explained in McCuskey (2002).

the artwork

1. website of Montevideo Collection Catalogue[112]. To avoid being parochial here, I should also mention similar institutes abroad, such as Electronic Arts Intermix[113] from New York, USA, and org.uk[114]>LUX[115], from London, UK.
2. website of INCCA[116].
3. tangible virtual museum – from Rosenblum and Macedonia (2005), see section 10.1.
4. digital dossier – concept graph for *abramovic dossier*, see section 10.2.
5. digital dossier – presentation gadget in *abramovic dossier*, with video of *Relation in Time*, with Ulay.
6. digital dossier – installation *Terra dea degli madre*, as 3D model.

[112] catalogue.montevideo.nl
[113] www.eai.org
[114] www.rockonflash.com/demos/RockOnFlashLogoDemo/asdoc/RockOnFlashLogo/org/uk.html
[115] www.rockonflash.com/demos/RockOnFlashLogoDemo/asdoc/RockOnFlashLogo/org/uk.html
[116] www.incca.org

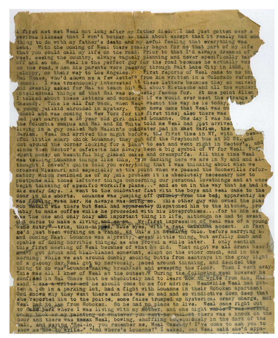
7. conservator's studio – *Self-Portrait with Braid*[117], see section 10.2
8. diagram – task world ontology, Welie et al. (1998).
9. tower of babel – location where the event took place, see below.
10. tower of babel – projection of *tower of babel* project, see section 10.3, submitted by Katelijne Arts. The project is a concept of Katelijne Arts, Tineke Goemans, Franka van de Goor, Leidi Haaijer en Bert Vogels.
11. tower of babel – a view from the inside of the building.
12. signs – sports, van Rooijen (2003), p. 278, 279.

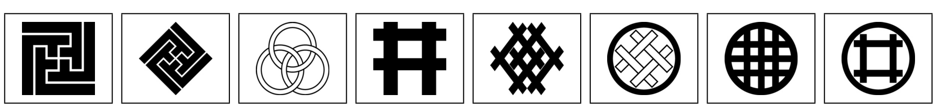
The artwork for this chapter is selected to emphasize *variety* and *experiment*. The collection of Montevideo contains a great number of works from the early history of video art, including the works of Nam June Paik and Bill Viola. Yet, despite the experimental flavor of these works, contemporary media art shows a strong sense of *context*, *experience* and *communication*, as demonstrated for example in the *tower of babel* project. The issues of preservation we dealt with in this chapter, may now, to conclude this chapter, be summarized as: how do we preserve the *context of experience* of contemporary media art?

[117] www.seattleartmuseum.org/exhibit/interactives/mexicanModernism/enter.asp

part v. game development

man is a playful animal
johan huizinga





2

reading directives In the final two chapters of this book, we will look at game development, with an emphasis on both implementation and design. In particular, we will discuss the motivation for using game technology to develop serious application, which in summary may be characterized as the requirement to allow for *immersion*, understood as experiental involvement of the user.

Essential sections are 11.2, in which we describe the development of VU-Life using the Source Half-Life 2 SDK, and 12.1, which presents a semi-formal game model, that may serve as a reference for designing more complex games.

perspectives Game development is essentially teamwork, and generally involves both artists as well as more technical developers. In addition, with to the commercial impact of games, many factors influence the design and implementation of games. As a consequence, we may look at game development from (among others) the following perspectives:

perspectives – game development

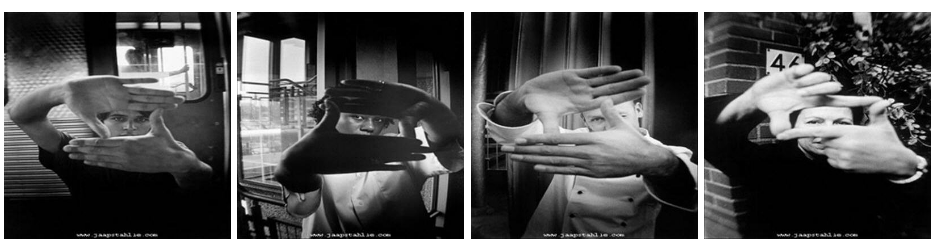
- artistic – plot, narrative, style
- technical – choice of game engine (SDK)
- sociological – sharing within game communities
- tool selection – supporting the workflow
- commercial – success factors
- management – teamwork

It is only very recently that game development became a topic with academic credibility. From an academic perspective, game technology may be regarded as enabling technology, that is related to many areas of computer science, including graphics, computer organization, distributed systems and software engineering.

essay topics In accordance with the variety of perspectives, an essay may focus on the commercial aspects of games, or, for example, software engineering aspects, or the design of community games. Consider writing an essay about:

- the success of games – criteria for comparison
- game engine architecture – options for extensibility
- community games – models for sharing information

When writing, make sure that you provide enough information about the actual history of games. And, remind that games require real-time performance, whereas the movies that may have triggered your phantasy may get away with effects that require many hours of processing. Afterall, games are about interaction!



3

the artwork

1. manuscript – used as a desktop by my favorite student.
2. signs – abstract symbols, van Rooijen (2003), p. 214, 215.
3. photographs – Jaap Stahlie[118], commissioned work.

[118] www.jaapstahlie.com

11. game technology for serious applications

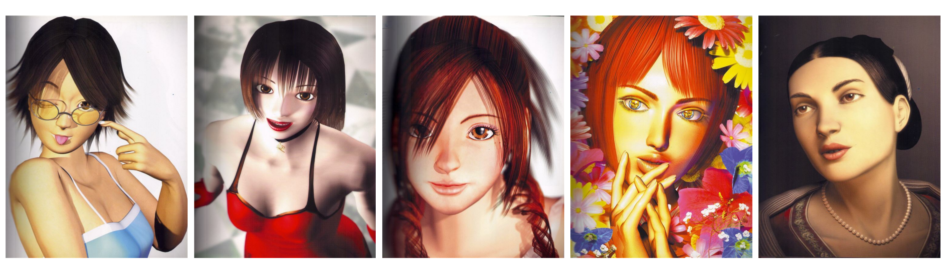
immersion does not require illusion but involvement

learning objectives

After reading this chapter you should have an idea how to approach the development of a moderately complex game, and you should also be able to discuss the notion of immersion and argue why using game technology is relevant for serious applications.

Game playing is fundamental to human life. Not only for entertainment, but also to acquire the necessary skills for survival. Game playing can take a variety of forms, but nowadays the dominant game paradigm is undoubtedly the interactive video game, to be played on a multimedia-enhanced PC or game console. Currently, games are being (re) discovered in the academic field, another serious areas of society, as excellent means for both the transfer of knowledge and, perhaps more importantly, for attitude change.

In this chapter we will look at the various issues in developing a game, and more specifically, in section 11.2 at requirements for a promotional game for our faculty and the issues that came up when giving a masterclass game development for high school students using this game. Finally, we will sketch the history of immersive systems, in particular panoramas, in section 11.3, and we will discuss how immersion is to be realized in a game context.



1

11.1 constructing a game

game playing

> According to Huizinga: "in the game we confront a function of the living creature which cannot be determined either biologically or logically".

From Kress and van Leeuwen (1996)

visual culture

> Games are an increasingly important element in our visual culture.

game programming

- gameplay programming
- game engine programming

2

elements of game design

pre-production, production, post-production
plan/blueprint
In the medium of game creation, we can capture fun in two areas:

fun

- in the general flow of the game experience and
- in the individual moments during a playing session.

of a certain player's unique experience

what is a game?

> Let's begin with a simple definition of what a game might be: a game is a series of processes that takes a player to a result, Schuytema (2007).

interesting decisions, skill aspect of a game

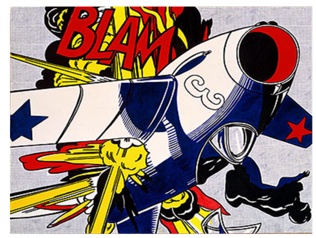
Let's postulate a slightly more robust definition for an interactive electronic game, Schuytema (2007):

interactive electronic game

A game is a play activity comprised of a series of actions and decisions, constrained by rules and the game world, moving towards an end condition. The rules and the game world are delivered by electronic media and controlled by a digital program. The rules and game world exist to create interesting situations to challenge and oppose the player. The player's actions, his decisions, excitement, and chances, really, his journey, all comprise the "soul of play". It is the richness of context, the challenge, excitement, and fun of a player's journey, and not simply the attainment of the end condition that determines the success of the game.

varoom

thung



blam

battle

- confrontation on well-established area
- delimited in space/time
- audience/participants who judge victory/loss

game design team

design team

- managers
- producers
- programmers
- testers
- designers

game software architecture

From Sherrod (2006):

game engine

- rendering system – 2D/3D graphics
- input system – user interaction
- sound system – ambient and re-active
- physics system – for the blockbusters
- animation system – motion of objects and characters
- artificial intelligence system – for real challenge(s)

success factors

a brief history of games

From Sanchez-Crespo Dalmau (2004)

a brief history of game programming

- phase i: before space war – hardwired
- phase ii: spacewar on atari – console with game
- phase iii: game console and PC – seperate game development
- phase iv: shakedown and consolidation – player code in data files
- phase v: advent of the game engine – user level design
- phase vi: the handheld revolution – the GameBoy
- phase vii: the cellular phenomenon – larger installed user base
- phase viii: multiplayer games – from MUD to Everquest

remarks:

1) tennis for two William Higinbotham Brookhaven National Labs New York,
1950'
2) Steve Russell, 1961, MIT, spacewar, two player game on Digital PDP-1
3) Atari VCS, Apple II, IBM PC (Dos)
4) Donkeykong, Pacman -> Nintendo
5) Doom -> Valve Halflife
6) Gameboy with well-established collection of game
7) NTT Docomo I-Mode, Samurai Romanesque
8) MUD (1979), MULE (1983), Ultima/Everquest 1600 hours/year

near future: a truly cinematic gaming experience

3

Screens from Samurai Romanesque.

4

example(s) – *samurai romanesque*

Samurai Romanesque, available on Japan's NIT DoCoMo packet-switched i-mode network, is an example of a mobile game with a large following. This massive multi-player game is developed by the japanese game developer Dwango. It runs on the Java 2 platform Micro Edition (J2ME). Players take, as we read in Krikke (2003) a virtual journey through 15-th century Japan, engage other players in real-time battles, visit historical towns and villages, practice the art of Zen, engage in romances and even can have children. This massive multiplayer role-playing game can accomodate up to half a million players, and is accounted to be a huge success in Japan. *Samurai Romanesque* is an example of a mobile game incorporating features such as position awareness, player history, chatting, and effective graphics. In Krikke (2003), it is further explained how the technology with which the game is implemented positions itself in the *battle for mobile cyberspace*.

research direction(s)– *serious games*

play, learn, become[119]

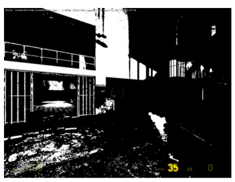
> Serious games and simulations are poised for a second revolution. Today's children, our workforce and scientists are increasingly playing, learning, and inventing in visually intensive "virtual" environments. In our increasingly experiential economy, immersive educational and training solutions are needed to advance the workforce of tomorrow. Game-based learning and technologies meet this challenge.

peace maker[120]

> Q: With the lion's share of strategy games on the market being devoted to ending a conflict through violence, why was it important to you to emphasize the need for a peaceful solution?
>
> A: When we started to work on the project and looked around at other video games, we encountered the notion that war is much more challenging and

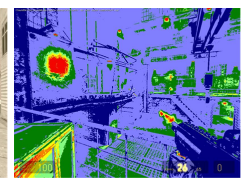
[119] www.virtualheroes.com
[120] seriousgamessource.com/features/feature_071806_peacemaker.php

conflict is essential to engage players. Many people we talked to perceived peacemaking as mere negotiations, where a group of diplomats sit at a table for lengthy discussions and sign agreements. We tried to shed light on what we see as the other side of peacemaking how challenging it is for a leader to gain trust and understanding in the face of constant violence. How difficult it is to execute concessions, while your own population is under stress or feeling despair. In a sense, peacemaking can be more complicated, sophisticated and rewarding than war making, and it is a message that we would like to convey to young adults, the future generation of leaders.


black/white


greyscale


sepia


thermal

5

11.2 game @ VU

In june 2005 we started with the development of a game, nicknamed VU-Life 2, using the Half-Life 2 SDK. We acquired a Cybercafe license for Half-Life 2, with 15 seats, because we would like to gain experience with using a state-of-the-art game engine, and we were impressed by the graphic capabilities of the Half-Life 2 Source game engine.

After some first explorations, we set ourselves the goal:

- to develop a game that could be used for promoting our institute, and
- to prepare a masterclass game development for high-school students.

Our first ideas concerning a game included a game in which the subject chases a target, a game where the subject has to escape, and an adventure game. In the end we decided for a less ambitious target, namely to develop a game which gives the subject information about our institute, by exploring a realistic game environment, representing part of our faculty. As an incentive, a simple puzzle was included which gives the subject information on how to obtain a 'hidden treasure', to be found in a specific location in the game environment. See section 2 for more information on this.

fig. 1: VU-Life 2 – opening screen

With only about eight months time, we decided to do a feasibility study first, to gain experience with the Half-Life 2 SDK technology, and to determine whether our requirements for the game and the masterclass could be met.

For the VU-Life 2 game, we can summarize our requirements as follows:

- the game must provide information about the faculty of sciences of the VU,
- the game environment must be realistic and sufficiently complex, and
- the interaction must be of a non-aggressive, non-violent, nature.

The last requirement has to do with the fact that the VU is by its origin a Christian university, so that anny agressive or violent interaction could hardly be considered to be an appropriate theme for a promotional game.

For the masterclass, we stated the following requirements:

- it must be suitable for beginners, in particular high school students,
- it must explain basic texture manipulation, and
- offer templates for modifying a game level, and finally
- there must be a simple (easy to understand) manual.

The format for a masterclass for high-school students at our institute is three times two hours of instruction. The goal is to attract (more) students for the exact sciences. However, if the masterclass would be too complex, we would run the risk to chase potential students away, which would be highly counter-productive.

In this paper we will report our experiences in developing the VU-Life 2 game and the associated masterclass[121]. The structure of this paper is as follows. In section 2 we will give an impression of the VU-Life 2 game by presenting a typical usage scenario. In the sections that follow, we will discuss the technical issues encountered in developing the VU-Life 2 game, and the assignments for

[121] www.cs.vu.nl/~eliens/game

the masterclass. In section 4, we will moreover describe the documentation we developed for the masterclass. In section 5 we will discuss the lessons we learned and in particular our experiences in presenting the masterclass to high-school students. And finally, we will draw our conclusions by giving a summary of our efforts and indicating our plans for the future.

VU-Life 2 – the game

To give an impression of the game and how we used the Source game engine and the associated Half-Life 2 SDK, let's start with a typical game scenario, illustrated with a walkthrough.

fig. 2(a) lecture room (b) lecture room (c) student office

When starting VU-Life 2, the player is positioned somewhere in the game environment, such as a lecture room, fig. 1(a). In the front left corner of the lecture room, middle right of fig. 1(a), there is a place marked as an information spot. The information spot corresponds with one of the nine in the top right of the screen. The player is expected to detect this correspondence by exploring the game environment. The nine squares together form a puzzle, indicating, when all squares are filled, where the hidden treasure can be found. In other words, when the player visits all the nine information spots contained in the game environment, the player has solved the puzzle and may proceed to obtain the hidden treasure.

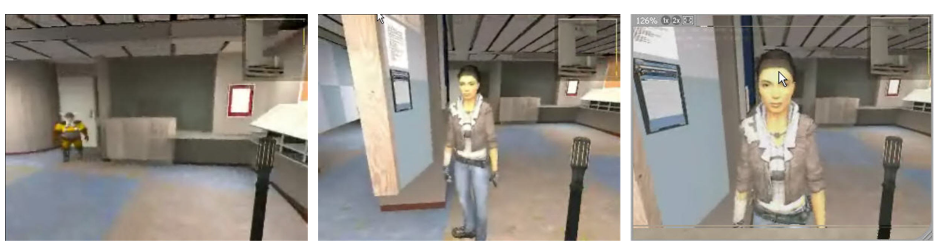


fig. 3(a) student office (b) student office (c) student office

To visit all the information spots, the player has to explore the game environment, including another lecture room, fig. 1(b), the student administration office, figs. 1(c) and 2, and the student dining room, fig. 3. While exploring the game environment, the player may read information about the curriculum, meet other students, fig. 2, and encounter potentially dangerous individuals, fig. 3(b).

fig. 4(a) restaurant (b) restaurant (c) restaurant

As illustrated in figs. 1-3, the puzzle squares will gradually become filled, and when complete, the combined puzzle squares will indicate the location of the hidden treasure, which is the 7th row of chairs of the lecture room in fig. 1(b).

Despite the fact that we intended to create a non-violent game, we must admit that the hidden treasure actually consists of obtaining the power to use weapons. Form our observations, and this was exactly what motivated us to include this feature, the use of weapons proved to be a most enjoyable aspect for the high school students playing the VU-Life 2 game, in particular when allowed to play in multi-user mode.

using the Half-Life 2 SDK – technical issues

The VU-Life 2 team had no prior experience with the Half-Life 2 Source SDK. Therefore we started by exploring three aspects of the Source SDK: level design with the Hammer editor, making game modifications, and importing (custom) models into Half-Life 2. During the exploration of these aspects we came across various technical issues, which we will discuss below.

level design First, we made various smaller levels. Each level was compiled and tested seperately so that it worked fine as a standalone level. The idea was to combine them, that is to create one large world containing the smaller levels. However, the initial coupling caused several compiling errors. After analyzing the errors, some important restrictons for building (large) levels became clear.

In the second part of the level compilation process called VVIS, a visibility tree of the level is made. This tree is used to tell the renderer what to draw from a given (player) viewpoint in the level. The amount of used brushes (the default shapes for creating a level) determine the size of the visibility tree. The bigger the tree, the longer VVIS will take to build the visibility tree at compile time and the more work the renderer has to determine what to draw at runtime. Therefore, the standard brushes should only be used for basic level structure. All other brushes that do not contribute to defining the basic level structure should be tied to so-called *func_detail* entities. This makes VVIS ignore them so that they do not contribute to the visibility tree, thus saving compiling and rendering time.

In addition, there is a (hardcoded) maximum to the number of vertices/faces you can use for a level. Each brush-based entity contributes to the number of vertices used. It is possible, however, to reduce the number of vertices used by converting brush-based objects to entities. This is done outside of the Hammer level editor with the use of 3D modelling software and the appropriate conversion tools.

With the above mentioned restrictions in mind we were able to create a relatively large level that more or less realistically represents the faculty of exact sciences of the VU campus. The key locations are, as partially illustrated in figs. 2-4, restaurant (fig. 4), lecture room S111 (fig. 2(a)), lecture room KC159 (fig. 2(b)), student office (figs. 2(c) and 3), multimedia room S353 (not shown).

To give an impression of the overall size of the *VU.vmf* game level, as map information we obtained 6464 solids, 41725 faces, 849 point entities, 1363 solid entities, and 129 unique textures, requiring in total a texture memory of 67918851 bytes (66.33 MB).

game modifications Since a multi-user environment was required. we chose to modify the Half-Life 2 Deathmatch source code, The biggest challenge for modifying the code was finding out how to implement the features for VU-Life 2. To this end, relevant code fragments were carefully studied in order to find out how the code is structured and works. Furthermore, by experimenting, it was possible to get the features working. Below is a list of features for the VU-Life 2 Mod.

- *Player properties* – Players start out immortal, meaning that they cannot "die" while exploring the world. Furthermore, continuous sprinting is enabled, which allows the player to walk around faster.
- *Puzzle HUD* – When the player starts out, the puzzle HUD is the only HUD element displayed.
- *Puzzle setter* – Allows puzzle parts to be displayed on the puzzle HUD.
- *Weapon enabler* – Allows weapons to be enabled/disabled for the player. Enabling the weapons also enables damage, and swithes from the puzzle HUD to the default Half-Life 2 HUD, which displays weapon and damage information along with a crosshair.

importing models Getting a model into the Half-Life 2 environment requires two steps:

- The model must be exported to the custom Valve format smd
- The model must be compiled from *smd* to *mdl* format

The first step required finding the correct plugin that allowed a conversion to the *smd* format. The second step required using Valve tool *studiomdl* and defining a *qc* file, which is used to specify properties for the compiled model. The default Valve tool *studiomdl.exe* proved to be difficult to work with, because it requires a lot of parameters have to be set. By using the StudioMDL 2.0 GUI, compiling the *smd* file was very easy. It sets the appropriate parameters, allowing the user to focus on the compiling of the model.

the masterclass – instruction and assignments

The masterclass consisted of three sessions, two hours each. In the first session, the (high school) students were given an overview and general instructions on how to accomplish the assignments, and were then set to play the VU-Life 2 game.

The assignments, as already indicated in section 1, were:

1. to modify an existing game level by applying different textures,
2. to create objects within an existing game level, and
3. (for advanced students only) to create a new level.

More complex assignments, such as creading a Mod, were considered to be outside of the scope of this masterclass.

The overview and instructions given in the first session included:

- an overview of the history of games,
- a general introduction on modelling characters and objects,
- the use of the Hammer editor, and finally,
- an explanation of the assignments.

The history of games encompassed historic landmarks such as Pong, Tetris and The Sims, as well as a brief discussion of current games like Worlds of Warcraft, and Half-Life 2.

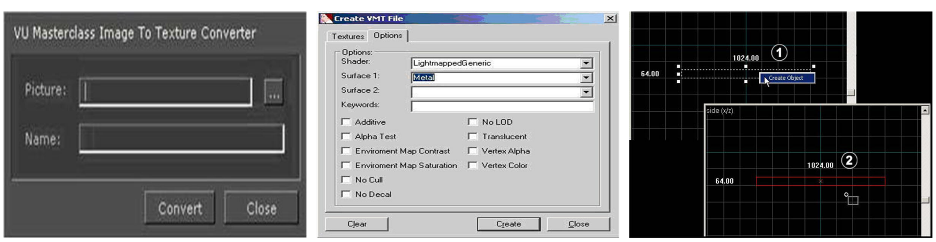


fig. 5(a) converter (b) VMT tool (c) camera

In the introduction on modelling an overview was given of the major tools, like Maya and 3DSMax, as well as a brief explanation of notions such as vectors, polygons, textures, lights, and skeleton-based animation.

Both the explanation of the use of the Hammer and the assigments were explicitly meant as a preparation for session two, in which the students started working on their assignments.

In addition to the oral overview and instructions, the students were given a manual, that was made available in paper as well as online, to prepare themselves for the assignments. The homework for the second session was to make pictures suitable for the application as textures in the *masterclass room*, which is depicted in fig. 5(a).

To allow the students to easily apply their textures, a texture conversion tool, fig. 4(a), was offered, that converts and image file into a texture for a particular location in the game level based on keywords, e.g. *mc_floor* for the texture on the floor of the *multimedia room*. Alternatively the students could use the VMT-Edit tool, fig. 4(b), and apply the texture using the Hammer editor, figs. 4(c) and 5.

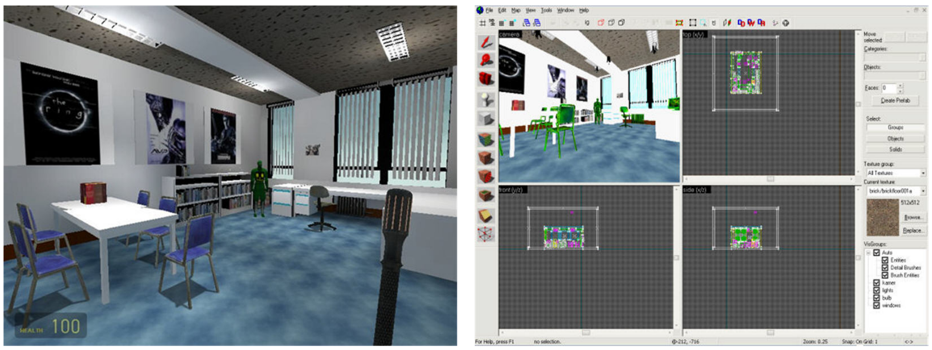


fig. 6(a) masterclass room (b) room in Hammer editor

The introduction on how to use the Hammer editor covered the basic tools, including the

- *block tool* – for creating simple object,
- *selection tool* – to select objects for texturing,
- *entity tool* – to select dynamic or interactive objects, and the
- *texture tool* – to apply textures to an object;

as well as how to compile a level into a map ready for play, including an explanation of the BSP (world), VIS (visibility), and RAD (radiosity) components.

The students were explicitly told that the assignments did not involve any programming, creating game AI, or modelling. (To learn these aspects of game development, they were simply adviced to sign up for our curriculum.) Instead, we told them, use your phantasy and be creative!

lessons learned

In the second session, the high school students started working with great fervour, see fig. 7.

Somewhat surprisingly, all students worked directly from the (paper) manual, rather than consulting the online documentation, or the help function with the tool.

fig. 7: masterclass at work

In retrospect, what appeared to be the main difficulty in developing the masterclass was to create challenging assignments for every skill level. In our case, the basic skill level (modifying textures of a template level) allowed the high school students to start immediately. By having optional advanced assignments like creating your own objects, you can keep all students interested, since there are assignments to match the various skill levels.

competition To stimulate the participants in their creativity, we awarded the best result, according to our judgement, with a VU-Life 2 T-shirt and a CD with Half-Life 2. The results varied from a music chamber, a space environment, a *Matrix* inspired room, and a messy study room. We awarded the *Matrix* room with the first prize, since it looked, although not very original, the most coherent.

example(s) – *dead media*

civilisation

> Media are special cases within the history of civilisation. They have contributed there share to the gigantic rubbish heaps that cover the face of our planet or to the mobile junk that zips through outer space.

dead media project

> Together with like-minded people, in 1995, Bruce Sterling started a mailinglist (at that time still an attractive option) to collect *obsolete software*. This list was soon expanded to collect *dead ideas*, or *dead artifacts*, and systems from the *history of technical media*: inventions that appeared suddenly and disappeared just as quickly, which dead-ended and were never developed further; models that never left the drawing board; or actual products that were bought and used and subsequently vanished into thin air.

machines can die

> Sterling's project confronted burgeoning phantasies about the immortality of machines with the simple facticity of a continuously growing list of things that have become defunct.

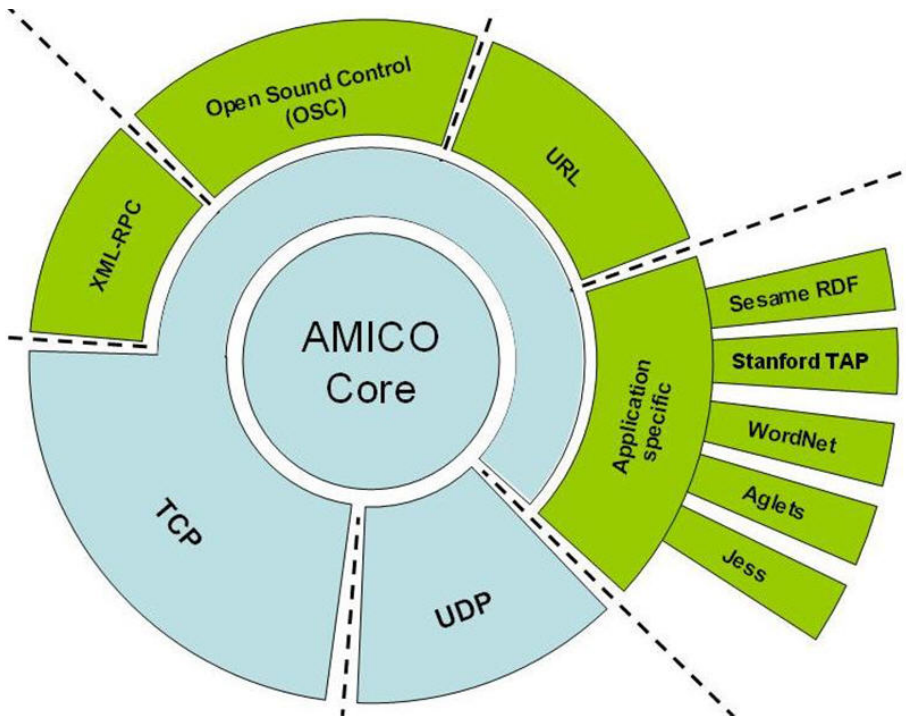
machines can die

> Once again, romantic notions of technology and of death were closely intertwined in the *Dead Media* Project.

research direction(s)– *open source game engine components*

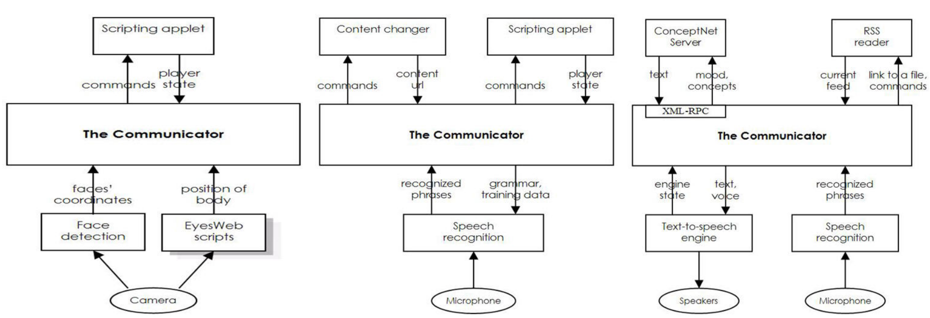
Delta3D[122]

[122] www.delta.org



6

multi-modal interaction – AMICO



camera input voice input news reader

7

11.3 immersion is not illusion

perspectives The notion of perspective is an interesting notion itself, since it describes both

- the organisation of the image, as well as
- the (optimal) point of view of the viewer.

This intricate relation between viewer and image, dependent on perspective, implies that when looked at from the 'wrong point of view', there will be a distortion of the image. The 'normal' perspective, as we know it, is the 'central' perspective. However, there are variants of perspective that force the viewer into an abnormal point of view, as for example with anamorphismes.

In a multi-dimesnional space often a change of perspective, that is a change of point of view, suffices for the correction of a reducing or distorting projection. Just image how a plane is projected as a line on an orthogonal surface, and a line as a point.

From Kress and van Leeuwen (1996):

realism

documentary modality of black and white realism ...

www.lichtensteinfoundation.org/

8

visual grammar

Grammar goes beyond formal rules of correctness. It is a means of representing *patterns of experience*. It enables human beings to build a mental picture of reality, to make sense of their experience of what goes on around them and inside them.

immersion versus illusion

analogon of reality

> Certainly, the image is not the reality but at least it is its perfect *analogon* and it is exactly this analogical perfection which, to our common sense, photography. This can be seen as the special status of the photographic image, it is a message without a code. Roland Barthes, cited from Kress and van Leeuwen (1996), p. 23

immersion

> The concept of immersion when implemented as an artwork surrenders most of the essential properties of an artwork.

Grau (2003), p. 319

properties of artwork(s)

- form
- structure
- function
- processuality
- statement

virtual reality

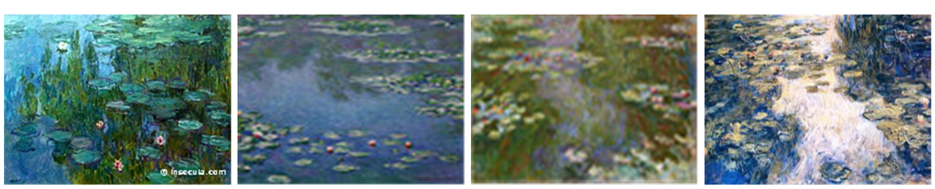
> The idea of *virtual reality* only appears to be without a history: in fact, it rests firmly on historic art traditions, which belongs to a discontinuous movement of seeking illusionary image spaces.

ecstatic transport

> Using contemporary image techniques, immersive art very often visualizes elements that can best be described as Dionysian: ecstatic transport and exhilaration.

collective memory

> It is an apparent feature of the concept of immersion that it engages with the spatial and pictorial concentration of the awareness of one's own people, the formation of collective identity through powerful images that occupy the function of memory.



9

new media

> A consequence of the constitutive function of artistic-illusionary utopias for the inception of new media of illusion is that the media are both a part of the history of culture and of technology.

10

realism versus naturalism

realism

> A *realism* is produced by a particular group as an effect of the complex of practices which define and constitute that group.

naturalism

> Each realism has its *naturalism*, that is a realism that is a definition of what counts as real, a set of criteria for the real, and it will find its expression in the *right*, the best, the most *natural* form of representing that kind of reality, be it a photograph or a diagram.

dominant paradigm

> The dominant standard by which we judge visual realism (and hence *visual modality*) remains for the moment, naturalism as conventionally understood, *photorealism*.

Kress and van Leeuwen (1996): *realism is a social construct*

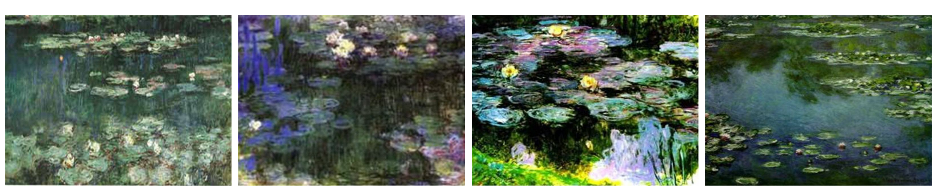
involvement– relationship(s) with application(s)

Sidney Fels, UIU04 keynote, *designing intimate experience*

Observation: "people form relationships with objects external to their own self".

aesthetics of interaction

- response – object disembodied from self
- control – self embodies object
- reflection – self disembodied from object
- belonging – object embodies self



11

example(s) – *Monet's Nympheas*

perso.orange.fr/art-deco.france/nympheas.htm

About Monet's Waterlilies Panorama in Giverny, from Grau (2003):

mass medium

> Thus, one year after Monet's death and fifty years after his *Impression soleil levant*, a late example of modern art reached the changed artistic landscape of the 1920's, transported in a derivative of *the* mass medium for images in the 19th century.

research directions– *information art*

See Wilson (2002).

Quote from MIT Press, Leonardo series:

cultural convergence

> *The cultural convergence of art, science, and technology provides ample opportunity for artists to challenge the very notion of how art is produced and to call into question its subject matter and its meaning in society.*

from Grau (2003).

tele-presence

- notions of artificial life
- fusion with (infinite) virtual image worlds
- transformation of self into digital data

human aspirations

> *Telepresence also combines the contents of three archetypal areas of human aspirations: automation, virtual illusion and metaphysical views of the self.*

cybergnosis

> What is being preached is the phantasm of union in a global net community, cybergnosis, salvation through technology, disembodied as a post-biological scattering of data that lives forever.

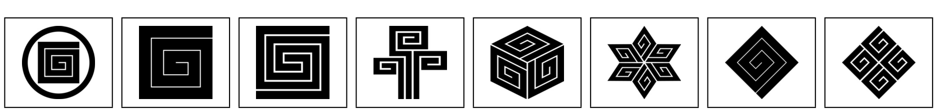
zealots

> What we observe are hyperzealots of a new technoreligion running wild, zapping, excerpting and floating in cyberspace.

aesthetics

> Since the eighteenth century, aesthetic theories have regarded *distance* as a constitutive element of reflection, self-discovery and the experience of art and nature.

11.4 development(s) – game design patterns



12

questions

game technology for serious applications

1. (*) What are the elementary steps in game development? Discuss the role technology plays in determining the game development project trajectory.

concepts

2. What phases can you distinguish in the actual development of a game?
3. Give arguments pro and con the use of a game engine.
4. Discuss the notion of immersion, and explain why immersion does not necessarily imply illusion.

technology

5. Characterize the built-in functionality that comes with a game engine.
6. Give a characterization of the tools that come with the Source Half Life 2 SDK.
7. Give a brief description of the history of immersive environments and application.
8. Discuss, on a suitable level of abstraction, the immersive features of games.

projects & further reading As a project, develop a non-violent game using the Source SDK. For example, you may develop an application that gives a community of users access their personal collections of photographs.

One interesting feature to explore is the use of narratives, that is a kind of guided tour that gives a user an overview of the collection of photographs by means of a story, taking (in other words) the user by the hand in navigating the gane space.

For further reading I suggest, apart from the manuals and learning materials that come with the Source SDK, books on game development such as Luna (2003). XXX

the artwork

1. *digital beauties* – taken from Wiedermann (2002).
2. Masereel, social realist works
3. Roy Lichtenstein, 1962
4. Masereel, social realist works
5. images from *Samurai Romanesque*, see section 1.3
6. HalfLife 2 shader programming
7. VU-Life 2 – opening screen
8. VU-Life 2 – screenshots
9. VU-Life 2 – screenshots
10. VU-Life 2 – screenshots
11. VU-Life 2 – tools
12. VU-Life 2 – tools

13. VU-Life 2 – masterclass
14. diagram AMICO core
15. diagram AMICO applications
16. Roy Lichtenstein, 1962, Stillives
17. Monet, Nympheas
18. Monet, Nympheas
19. Monet, Nympheas
20. signs – abstract, van Rooijen (2003), p. 146, 147.

12. towards an aesthetics for interaction

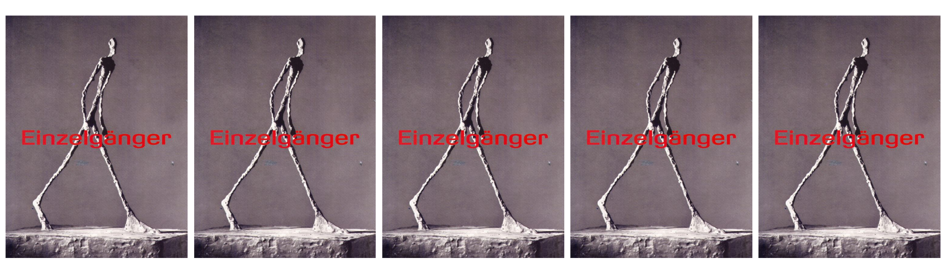
experience is determined by meaning

learning objectives

After reading this chapter you should have an understanding of the model underlying game playing, and the role of narratives in interaction. Furthermore, you might have an idea of how to define aesthetic meaning in a cultural context, and apply your understanding to the creative development of meaningful interactive systems.

As in music, the meaning of interactive applications is determined, not only by its sensory appearance, but to a high extent by the structure and functionality of the application. This observation may, also, explain, why narratives become more and more important in current video games, namely in providing a meaningful context for possible user actions.

In this chapter, we take an interactive game-model extended with narrative functionality as a starting point to explore the aesthetics of interactive applications. In section 12.1, we will introduce a model for interactive video games, and in section 12.2 we will present a variety of rules for the construction of narratives in a game context. Finally, in section 12.3, we will characterize the notion of meaning from a traditional semiotics perspective, which we will then apply in the context of games and interactive multimedia applications.



1

12.1 a game model

game theory perspectives

- system – (formal) set of rules
- relation – between player and game (affectionate)
- context – negotiable relation with 'real world'

dictionary

2

classic game model

- *rules*: formal system
- *outcome*: variable and quantifiable
- *value*: different valorisation assignments
- *effort*: in order to influence the outcome
- *attachment*: emotionally attached to outcome (hooked)
- *consequences*: optional and negotiable (profit?)

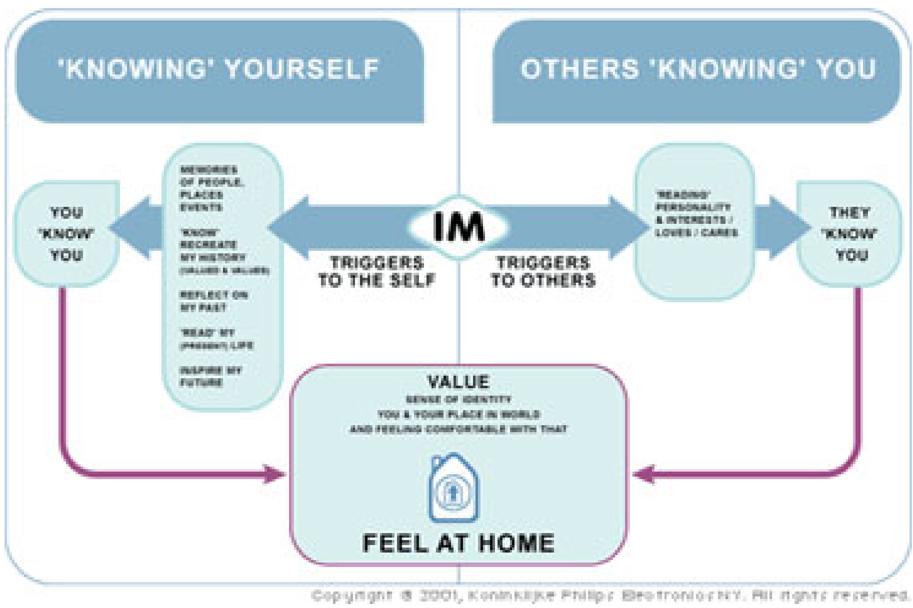
rules vs fiction

> Game fiction is ambiguous, optional and imagined by the player in uncontrollable and unpredictable ways, but the emphasis on fictional worlds may be the strongest innovation of the video game.

3

theory of interaction

Are *games* relevant for a theory of interaction?



4

example(s) – *intimate media*

From the company that used the slogan "let's make things better", and now advertises its products with "sense and simplicity", there is the MIME[123] project, not to be confused with the multipart internet mail standard, which focusses on *Multiple Intimate Media Environments.*

As'can be read on their website: *Intimate media describes the things that people create and collect to store and share their personal memories, interests and loves.* And: *Intimate media is central to how people make sense of their world by representing roots, heritage and a sense of belonging, achievement and connection.*

In the MIME project seven core qualities are identified which *capture the essence of the intimate media experience*:

intimate media

- sensorial – experience is visual, audible, tactile, olfaric
- personalized – objects embody meaning and memories
- analogue – people relate to physical objects

[123] www.design.philips.com/about/design/section-13484

- enhancement – people already have extensive intimate media collections
- serendipity – it supports unstructured and flexible usage
- longevity – objects may exist over generations

As concepts embodying their ideas they propose, among other:

intimate media

1. *GlowTags* – a subtle way to trigger the person who has placed it or who sees it
2. *Living Scrap Book* – to capture and collect information and media digitally
3. *Picture Ball* – as an object of decoration and a focus for storytelling
4. *Lonely Planet Listener* – enabling people to listen to a real time connection to another place

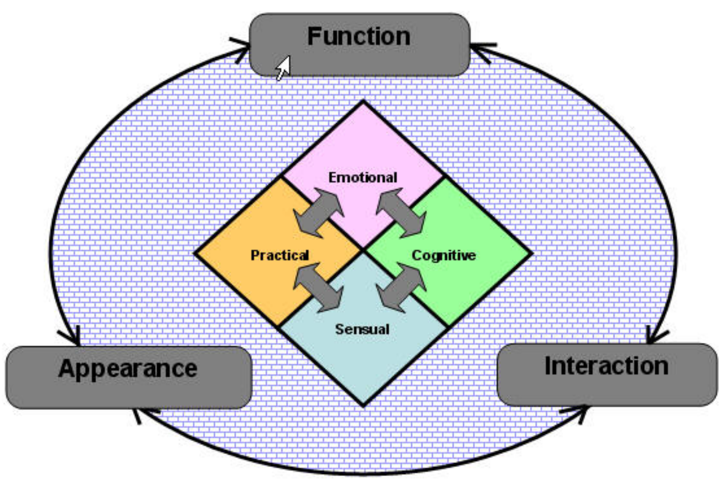
5

research directions– *experience as meaning*

framework set up by Dhaval Vyas Vyas and van der Veer (2006) .

experience as meaning

user's experience = meaning s/he construct



6

framework

- experience occurs during the interaction between the user(s) and the interactive system(s) in the lived environment
- designers convey meaning (consciously or unconsciously) through the appearance, interaction and function of the system

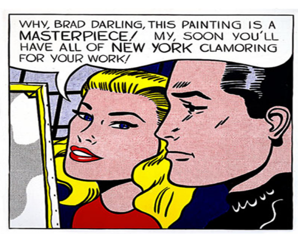
- user(s) construct a coherent whole that is a combination of sensual, cognitive, emotional and practical forms of experience

In other words, an *interactive system* is determined by *function*, *interaction* and *appearance*.

12.2 guidelines for narrative construction(s)

film as art

> By still being read, the little treatise seems to prove that in spite of all the changes that have taken place in their *form*, *content* and *function*, films are still most genuinly effective when they rely on the basic properties of the *visual medium*.



illusion

> Similarly, in film or theatre, so long as the essentials of any event are shown, the illusion takes place

patterns of light

> Thus we can perceive objects and events as living and at the same time imaginary, as *real objects* and as simple *patterns of light* on the projection screen, and it is this fact that makes film art possible.

frames of reference

> It is one of the most important formal qualities of film that every object that is reproduced appears simultaneously in two entirely different frames of reference, namely the two-dimensional and the three-dimensional, and that as one identical object it fulfills two different functions in the two contexts.

principles of montage

- cutting – unit length, whole scenes, cuts within scenes
- time relations – synchronized, before/after, neutral
- space relations – same place (different time), different place
- subject matter – similarity and/or contrast

film technique

- camera – position, focus, movement
- transitions – fading in/out, dissolving, stills

- arrangement – light/shade, color, sound

cinematographic motion

- movement of objects
- effect of perspective
- motion of camera
- montage of scenes

aesthetics of shock

aesthetics of shock

> It is within the realm of probability that the *shock*, which Walter Benjamin diagnosed as being film's aesthetic innovation, will undergo renewal and intensification with far more sophisticated means.

voyeurism

> The most obvious symptom of this loss of distance will be a voyeuristic, dissecting penetration of representations of objects and bodies.

the meaning of composition

narrative implications

- objects – the items in the image
- vectors – (imaginary) lines suggesting interaction
- gaze – inward (offer) / outward (demand)

transactional or non-transactional

composition

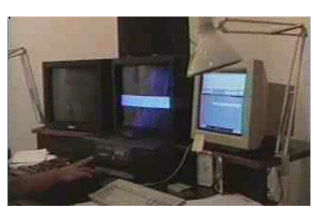
> Composition, then, relates the *representational* and *interactive* meanings of the picture to eachother, through three interrelated systems.

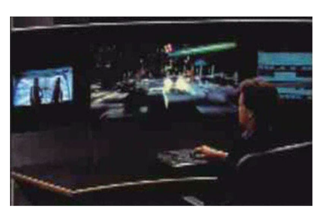
representations

- information value – left/right, top/bottom, centre/margin
- salience – foreground/background, relative size, contrast
- framing – connecting or dissolving lines

information value

- left/right – given versus new
- top/bottom – ideal versus real
- centre/margin – important versus marginal





7

example(s) – *edgecodes*

The *edgecodes*[124] documentary film by Phillip Daniels gives an inside account of film editing, a history of the evolution of editing conventions, as well as an account of the technological innovations of the late 20th century and their impact on film editing. It was shown at the documentary film festival IDFA[125] 2004, in Amsterdam. Movies were, as Daniels staes, the new artform of the 20th century, which distinguishes itself from other artforms by ... *editing*!

The film begins with the statement such as *the concept that a film is shot is entirely false, a film is not shot, its is built*, continuing with the statements that *the message of the movie medium is that of transition*, and that *a movie must have a beginning, middle and ending, but not necessarily in that order*.

The documentary is highly visual, after all it is an editor's movie, and contains many fragments from wellknown movies and interviews with famous directors, among which George Lucas, who introduced the *editoroid* in the eighties, an editing machine built with at the time modern computing technology. George Lucas, image left above, explained the introduction of his editing machine by saying that he wanted to have *a system, ... that is intuitive, obvious, ... and hihgly malleable, ... visual ...*. He wanted a machine that allowed him to use his moterskills, without the intervention of an engineer. But in the interview he admitted that they were *on the bleeding edge* in those days. Nowadays, real-time editing, with computer graphics (CG) support is (finally) feasible. See chapter 4.

research directions– *multimedia in context*

Course organized in 1998 with Lynda hardman (CWI) for PhD students: *multimedia in context*[126].

the scientific context Als gebied van onderwijs en onderzoek heeft multimedia een raakvlak met vele wetenschapsgebieden:

wetenschappelijke context

- mathematics – matrix algebra, transforms
- physics – game physic, particle systems
- computer science – technological infra-structure
- information theory – compression and delivery
- media theory – history of communication
- semiotics – theory of meaning

the societal context

maatschappelijke context

- cultural heritage – digital dissemination of art

124 www.edgecodes.com
125 www.idfa.nl
126 www.cs.vu.nl/∼eliens/online/courses/siks98

- education & communication – presentation of concepts and examples

the technological context

technological context

- modelling – objects, characters
- interaction – game programming
- architecture – game engine design
- rendering – programming the graphics hardware

the creative context

creative context

- visual design – style, models and attributes
- story telling – narrative structure

multimedia & game development

multimedia & game development

- game modelling and design
- game programming
- game engine architecture

12.3 the definition of meaning

From Bruner (1972):

learning/meaning

- actionary level – action and movements
- sensory/iconic – images and impressions
- symbolic – language and mathematics

learning by doing

8

basic geometrical shapes

The basic geometrical shapes have always been a source of fascination, even of religious awe. And our scientific age is no exception.

nervous system

(basic geometrical shapes) have been thought to have the power to directly affect our nervous system, for instance by the constructivist artist Gabo: "The emotional force of an absolute shape is unique and not replaceable by any other means ..."

9

semiotics – a theory of meaning

semiotics – a theory of meaning

- signifier – sign/symbol
- signified – what is referred to

meaning: relation between signifier and signified

style: ???

semiotic modes

... is the move from the verbal to the visual a loss, or a gain?

complexity

... it has to be handled visually, because the verbal is no longer adequate?

multimedia

The multi-modality of written texts has, by and large, been ignored, whether in educational contexts, in linguistic theorizing, or in popular common sense. Today, in the age of *multimedia*, it can suddenly be perceived again.

quotes

- *myth of transparency* – visual communication is always coded!

- *literacy* – standards for semiotic order
- *semiotic modes* – text, visual, auditive, ...
- *computer technology* – central to semiotic landscape
- *semiotic activities* – production, transformation, development

semiotic landscape

> The place of visual communication in a given society can only be understood in the context of, on the one hand, the range of forms or modes of public communication available in that society, and, on the other hand, their uses and valuations.

10

meaning

> What is the meaning of meaning when apparently meaningless media expressions, eg. *Sonic Acts*, are experienced as meaningful?

markers of veracity

From Kress and van Leeuwen (1996):

reliabilty

> One of the crucial issues in communication is the question of the *reliability of messages*. Is what we see or hear true, factual, real, or is it a lie, a fiction, something outside reality? To some extent the form of the message itself suggests the answer.

modality markers -> motivated signs -> transparancy

modality markers (1)

- *color saturation* – black/white
- *color differentiation* – monochrome
- *color modulation* – true to life

modality markers (2)

- *context* – background, frame
- *representation* – level of detail
- *depth* – perspective
- *illumination* – light or shade
- *brightness* – a matter of degree

modality

> Modality is realized by a complex interplay of visual cues. The same thing may be abstract in one or several markers and naturalistic in others.

Coding orientation, what counts as real.

coding orientation

- *technical/scientific* – effectiveness, blueprint
- *sensory* – pleasure principle is dominant
- *abstract* – used by socia-cultural elite
- *naturalistic* – dominant common sense paradigm of realism

TV

From Arnheim (1957):

TV

> For the first time in the history of man's striving for understanding, *simultaneity* can be experienced as such, not merely translated as a succession in time.

From Arnheim (1957):

sensory stimulation

> Although the new victory over time and space represents an impressive enrichment of the perceptual world, it also favors a *cult of sensory stimulation* which is characteristic of the cultural attitude of our time.

From Arnheim (1957):

direct experience

> Proud of our inventions – photography, film, radio, ... – we praise the educational virtues of *direct experience*.

From Arnheim (1957):

communication

> When communication can be achieved by pointing with the finger, however, the mouth grows silent, the writing hand stops and the mind shrinks.

11

new media

From Grau (2003):

channels

> Here the decisive questions remain: who controls the channels, who distributes right of access, and who exercises economic and political authority over the networks?

From Grau (2003), p. 281:

visions

> The history of technological visions is the history of our dreams, our vagaries and our errors. Media utopias fluctuate, often occurring in a magical or occult ambience.

panorama See www.cs.vu.nl/∼eliens/mma/panorama.html

research directions– *intelligent advice*

From the adapted version of I-GUARD[127] proposal, discussed in section 10.3.

Our aim is to arrive at a general framework for artist's digital dossiers, that provide intelligent guidance to both the expert user, responsible for the future re-installation of the work(s), and the interested layman, that wishes to get acquainted with a particular work or collection of works. In general, there are two techniques that we can apply to provide such guidance:

- filtering the information space according to the user's perspective, and
- intelligent agents, that (pro) actively aid the user in searching the information space.

[127] www.cs.vu.nl/∼eliens/research/i-guard.html

Filtering the information space may be used to restrict the concept graph that defines the navigation structure, by stating assumptions with respect to the relevance of particular categories from a user's perspective.

Intelligent agents is an approach stemming from artificial intelligence which allows for providing guidance in a variety of ways, possibly even in an embodied form using a face or humanoid figure to give suggestions to the user on what interactions to perform. In Empathic we have investigated the use of embodied agents in a digital dossier for the artist Marinus Boezem. In our current research, however, we will very likely not use embodied agents. Nevertheless, we will investigate to what extent we can use an agent model, possibly with learning capabilities as explored in Hildebrand et al. (2003), to provide guidance and support interaction.

Our goal is to arrive at an *advice function*, that offers the user at any navigation point a choice of continuations and/or a selection of guided tours, focussing on a topic of interest.

For selecting the items to be presented in a guided tour, the most obvious way is to pre-define a sequence based on user profiles. Very likely this can be done in a more flexible way in a rule-based manner, applied to a template tour. More interesting, however, is to investigate whether guided tours can be generated dynamically based on tracking actual user interaction of (expert) users, using techniques from prediction theory, as explained in section 6b.

To allow for meaningful interaction with 3D models, allowing to view for example information about the materials used or its installation procedure, we must find a way to connect that information to user actions in a generic way. In other words, there is an information representation problem, namely, how to relate contextual information in a generic fashion to elements of a 3D model representing an artwork. Although such interactions can be realized by embedding invisible) action/event objects in the model, a more generic way of representing such relations is desirable, to avoid the need for the time-consuming hand-crafting for which in practice there may not even be the necessary (human) resources.

regret function(s)

For the selection of items in guided tours and the generation of interesting sequences, we will explore the use of *prediction theory*. As explained in Cesa-Bianchi and Lucosi (2006), prediction theory uses a model of prediction based on *expert advice*. However, instead of the traditional *loss* function, used in a stochastic approach, prediction theory uses a *regret* function, which expresses the difference between an actual prediction and the advice of a *collection of experts*. An *expert*, in this context, is an abstract entity, that may be either embodied by an algorithm, a random selection, or an actual expert.

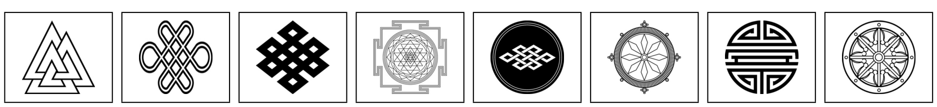
We will investigate, for the construction of guided tours, whether it is possible to generate interesting sequences by using a (sequence of) prediction(s) that minimizes the *regret* function, which respect to the navigation sequence(s) recorded from actual expert users.

In particular, we will strive for implementing the *advice function*, in a generic way, by means of a learning mechanism that extracts recommended continuations and guided tours from tracking expert user navigation.

12.4 development(s) – philosophy and beyond

phrase(s)

- to be aware what is there
- the rethorics of the material (Brancusi)
- Play Station Double Time
- art – select material
- technology – solving a problem
- scientist – establish a theory
- creative impulse will set you free (ad)
- design should serve us, rather than demand that we conform



12

questions

towards an aesthetics for interaction

1. (*) Discuss the factors affect interactive game playing, and indicate how they may contribute to the success of a game.

concepts

2. Describe the model underlying game playing.
3. Discuss how narrative(s) affect interaction in game playing.
4. Characterize the notion of meaning from a semiotics point of view. Explain why meaning is dependent on cultural context.

technology

5. How would you characterize the role of interaction in game playing?
6. Give at least two construction rules for cinematographic narrative, and explain their use by an example.
7. What is the difference between a signifier and a signified?

8. Explain the role signifiers play in the aesthetic appreciation of an application.

projects & further reading As a project, explore the ways narratives may be constructed from a collection of images. Deploy the various editing facilities for providing flashbacks, flashforwards, and other (temporal) relations within storytelling.

You may implement this using flash, VRML, or even try to embed such a narration facility in a game level developed with the Delta3D or the Source SDK.

For further reading I suggest you to take a look at more theoretical material from media theory, such as Bolter and Grusin (2000). Also there is a large collection of books from MIT Media Press that is of relevance for our new visual culture.

the artwork

1. einzelganger – *walking man* of Alberto Giacommeti, taken from an aanouncement of the Ives Ensemble, Amsterdam.
2. diagram MIME
3. diagram *experience as meaning*
4. Roy Lichtenstein, 1962, (a) Kiss II, (b) Masterpiece, (c) Forget it, forget me.
5. edgecodes – showing George Lucas and his *editoroid*.
6. El Lissitzky, suprematist works
7. El Lissitzky, suprematist works
8. Roy Lichtenstein, 1999, Still lifes with brushstrokes
9. Les Demoisselles dAvignon, Picasso, 1908, regarded as the start of Cubism, and Le Goutier, Jean Metzinger, 1911, often referred to as the Mona Lisa of Cubism.
10. signs – abstract, van Rooijen (2003), p. 228, 229.

The *walking man* is one of my favorate sculptures, for over a long time. It is also associated to the motto of part iv: *a journey of a thousand miles begins with the first step* . As an autobiographical note, the *walking man*, with *einzelganger* superposed (in translation *loner*), reflects the writing of *topical media*. In particular, the image put in sequence, reminds me of the repetitive complaints of my superior at the faculty, who over and over again told me that I was always *alone in my room*, *isolated*, *on an island*. I must admit there is a truth in this, as I felt that the disciplines of software engineering and multimedia are widely divergent, and in that sense I was on my own. This book has undergone many rewritings, due partly to a clash between the expectations of others and my own vision on multimedia. And with a superior who emphasizes that he is "the boss", but has no intellectual authority nor any inspirational leadership whatsoever, at least not in the area of multimedia and gane development, there is really no other way than to go your own way. So I did it my way, indeed, quoting Paul Anka's song, made 'unforgettable' by Frank Sinatra.

In other words, after this brief autobiographical digression, the visual theme of this chapter on the aesthetics of interactive systems is on individual judgement, as exemplified among others by the suprematist works of El Lissitzky, the amplification of cartoons as art by Roy Lichtenstein, and the pioneers of Cubism. After all, individual judgement is what you need, when you wish to be involved in multimedia and/or game development.

afterthought(s)

The world of multimedia may be looked at in many ways. In fact, the phrase *multimedia* is too generic to be meaningful in any way. Nevertheless, multimedia has become a subject of interest for academia. This book has been written from an academic perspective. Let me clarify this perspective, to provide you with some context that might help you in understanding this book and use it more effectively in either education, research, or even your artistic endeavors.

As a starting point, let's look (again) at the *media equation*, quoted in the *research directions* of section 9.1:

media equation(s) 1/4

> *We regularly exploit the media equation for enjoyment by the willing suspension of our critical faculties. Theatre is the projection of a story through the window of a stage, and typically the audience gets immersed in the story as if it was real.*

This suspension of our critical faculties seems opposed to what we are used to in academic practice. And, indeed, there is an often noted conflict between the arts and the sciences, a conflict that the introduction of multimedia in the academic curriculum cannot resolve.

If we try to delineate the 'meaning' of multimedia more precisely, we might come up with pseudo-equation such as the following.

multimedia equation(s) 2/4

$$\text{multimedia} = \text{presentation} + \text{context}$$

where *presentation* includes the sensory and aesthetic part and *context* everything else. Now, at the risk of getting too much involved in 'funny mathematics' we might define *context* by another series of pseudo-equations

multimedia equation(s) 3/4

- context = convergence + information + architecture

where

multimedia equation(s) 4/4

- convergence = data +platform + distribution
- information = storage and retrieval
- architecture = compression + components + connectivity

Clearly, and this is exactly what this exercise in funny mathematics intended to illustrate, this book is about the contextual aspects of multimedia. Contextual aspects that may be the subject of academic research.

Is there any hope to include the presentational or aesthetic aspects in the academic curriculum? Based on a thought experiment, that explored the possibility of algorithmic art and aesthetics, Eliens (1988), I would say no. And as a matter of fact, I strongly disagree with a recipe-based approach to developing multimedia presentations, as seems to be popular in a number of the academic multimedia courses.

There is another shade of meaning that may be attributed to the notion of *context*, namely context of application. Evidently, multimedia has become a natural ingredient of almost any application you can think of. In 1998, I organized a course on multimedia for Ph.D. students, entitled *Multimedia in Context.* This course dealt with some of the issues in distributed multimedia and multimedia information retrieval, as well as applications in the publishing industry, travel advertisement and medical diagnosis. To announce the course, I used an image from medieval alchemy, see part I, and a phrase characterizing 'perfect solutions'.

perfect solutions

> *Much more than the art of turning base metals into gold, alchemy is a system of cosmic symbolism. The alchemist learns how to create within a sealed vessel a* Model of the Universe *in which the opposing complementary forces of Male and Female, Earth and Air, Fire and Water attain the perfect synthesis of which gold is the emblem.*

Risking obscurity at this point, I wish to equate multimedia with alchemy, to emphasize that the engineering of multimedia is an art that takes a lifetime to master. Repeating the quote from section 9.1:

multimedia engineering

> *"engineering is the art of moulding materials we do not wholly understand ... in such a way that the community at large has no reason to suspect the extent of our ignorance."*

multimedia in context

Originally the book, that is chapters 1-7, were written for the *Multimedia and Culture* curriculum at the Vrije Universiteit, Amsterdam, that started in 2001. In particular, the book contains the course notes for the first year course *introduction multimedia*.

Then I constructed four follow-up courses:

multimedia courses[128]

- multimedia authoring – Web3D/VRML
- intelligent multimedia technology – Virtual Environments

[128] www.cs.vu.nl/~eliens/multimedia

- visual design – digital content creation
- multimedia casus – digital dossier(s)

The first of these courses deals with the technology for creating 3D scenes and worlds see appendix B, whereas the second is about providing intelligent services in virtual environments, as discussed in chapter 8 and appendix E. In addition, *Multimedia and Culture* students are required to take a course in *visual design* and to work on a *multimedia casus* to bring what they learned into practice. The structure and content of these courses are reflected in chapters 9 and 10.

Due to faculty politics, the *Multimedia and Culture* curriculum was reduced to a minor in Information Science, which made it less appealing, both for students and staff, including me. Over time, all the course mentioned above were re-purposed for the newly created specialisation *multimedia* in Computer Science, which attracts more technically oriented students, with better skills for actual multimedia and game application development. Although not technical in nature, chapters 11 and 12 were written with these students in mind. And very likely, or hopefully, the specialisation *multimedia* will soon become *multimedia and game development*.

Faculty politics is not a nice topic, but unfortunately has an effect on daily life, even to the extent that I sometimes regret that I gave up the, within an academic environment, relatively safe and simple discipline of software engineering and object-oriented software development. The truth of the matter is that, whatever the reasons, multimedia and game development does not fit in well in the standard academic context of computer science. Although liked by students, it is certainly not well accepted, and for that matter understood, by the senior staff. But although anecdotes about the many conflicts about research directions and scientific merit would be interesting for those who like gossip, the sad fact is that the multidisciplinary background of multimedia and game development would require an intellectually and artistically more rich environment than a department of computer science can offer.

explorative development

From the perspective of research, the situation is not much better. It is my strong belief, right or wrong, that relevant research in the area of multimedia and game development requires *explorative development*, that is the design and implementation of prototype applications that embody the realization of an idea, as with our research on the *digital dossier(s)*, an idea that includes technical as well as cultural and presentational aspects. But how hard it is to perform such multi-disciplinary research in an environment that is by tradition pre-dominantly mono-disciplinary.

Back to the book, apart from providing an introduction to a number of issues and research areas in the world of multimedia, this book also defines, in an implicit way, a research program that concerns the development and use of

virtual reality interfaces for multimedia information systems[129]

[129] www.cs.vu.nl/~eliens/research

All aspect covered in this book contribute, one way or another, to that (implicit) research program that may be classified under the heading of *intelligent multimedia*, of which a tentative definition is given in appendix E. And, admittedly, there are many aspects that are not covered, in particular those that are related to more advanced multimedia, virtual reality technology, and artificial inteligence.

the skill(s) of relevance

At this stage you may still wonder why I chose to name the book *topical media*. Let me explain. The phrase *topical*, as an adjective, has the following meanins:

topical

1. designed for involving local application (as an anesthetic),
2. relating to, or arranged by, topics,
3. referring to the topics of the day or place.

Although certainly not meant as an anesthetic, and even though it is arranged by, or at least refers to, topics, the intended meaning is due to the motivation to write a book that is relevant for the topics of (interest of) todays world. And, although it may not teach you the actual skills necessary to survive in todays world of multimedia and game development, it is certeainly meant to help you in acquiring the *skill of relevence*, see Bruner (1972), in this area, a skill that you will need to find your proper place and direction, anytime, anywhere.

Amsterdam, 9/8/2006
A. Eliëns

appendix

A. Web3D – VRML/X3D

Nowadays PCs allow for powerful 3D graphics. 3D graphics are, until now, mainly used by dedicated applications such as CAD/CAM and, not to forget, games. It is to be expected that 3D graphics will also manifest themselves in other types of applications, including web applications. In the Multimedia Authoring I course, students are required to develop such applications:

Multimedia Authoring I – Web3D/VRML

- *product demo* – with descriptive information and animation(s)
- *infotainment VR* – in the areas of Culture, Commerce or Entertainment

The latter assignment, the *infotainment VR*, may result in either a virtual museum, a game, or an extended product demo with a suitable environment and interaction facilities.

The purpose of the Web3D/VRML course is not so much the modeling of 3D objects *per s&eaigu;* but rather the organisation of 3D material (using the PROTO construct) and the development of suitable interaction mechanisms and guided tours (using sensors and scripts). This course, as well as the other multimedia authoring course is focussed on a programmatic approach to 3D. Hence, no advanced tools are used. Not because they are too expensive (which is also true), but because students should learn the basics first!

Why did we choose for Web3D, and more in particular VRML? Some argue that VRML is slow. Moreover, navigation in VRML is not altogether pleasant. Why not a (more native) format such as OpenGL? The answer is simply that VRML offers the right level of abstraction for modeling and programming 3D worlds. OpenGL does not. In the timespan of one month, VRML allows you to develop rather interesting and complex worlds, whereas with OpenGL (using C or C++) you would probably still be stuck with very simple scenes.

As concerns the focus on Web3D, I simply state that delivery of (rich media) 3D content is the way to go. The web is our global information repository, also for multimedia and 3D content. And we should be optimistic about performance issues. Already Web3D is of much better quality than the native 3D in the beginning of the 1990s.

What will be the future of Web3D and VRML? I don't know. As concerns VRML, the 3D modeling concepts and programming model underlying VRML are sufficiently established (as they are also part of X3D) that VRML will very likely survive in the future. The future of Web3D will depend on the success of the Web3D consortium of which a mission statement is given below.

www.web3d.org

> *The term Web3D describes any programming or descriptive language that can be used to deliver interactive 3D objects and worlds across the internet. This includes open languages such as Virtual Reality Modeling Language (VRML), Java3D and X3D (under development) - also any proprietary languages that have been developed for the same purpose come under the umbrella of Web3D. The Web3D Repository is an impartial, comprehensive, community resource for the dissemination of information relating to Web3D and is maintained by the Web3D Consortium.*

More in particular, the Web3D repository includes the X3D SDK to promote the adoption of X3D in industry and academia.

X3D SDK

> *This comprehensive suite of X3D and VRML software is available online at sdk.web3d.org and provides a huge range of viewers, content, tools, applications, and source code. The primary purpose of the SDK is to enable further development of X3D-aware applications and content.*

However, before downloading like crazy, you'd better get acquainted with the major concepts of VRML first. After all, VRML has been around for some time and VRML technology, although not perfect, seems to be rather stable.

Virtual Reality Modeling Language

VRML is a scengraph-based graphical format. A scenegraph is a tree-like structure that contains nodes in a hierarchical fashion. The scenegraph is a description of the static aspects of a 3D world or scene. The dynamic aspects of a scene are effected by routing events between nodes. When routing events, the hierarchical structure of the scenegraph is of no importance. Assuming compatible node types, event routing can occur between arbitrary nodes.

Below, an overview is given of the types of nodes supported by VRML as well as a number of browser-specific extensions introduced by *blaxxun*. The nodes that you might need for a first assignment are indicated by an asteriks. Additional information on the individual nodes is available in the online version.

abstraction and grouping

- *abstraction* – Inline Switch*
- *grouping* – Billboard, Collision Group, Transform*
- *scene* – Background LOD NavigationInfo Viewpoint* WorldInfo

geometry and appearance

- *geometry* – Box* Cone Coordinate Cylinder ElevationGrid Extrusion IndexedFaceSet IndexedLineSet Normal PointSet Shape* Sphere*

- *appearance* – Appearance* Color* Imagetexture* Material* MovieTexture PictureTexture TextureCoordinate TextureTransform
- *text* – FontStyle Text*

interaction and behavior

- *sensors* – Anchor CylinderSensor PlaneSensor ProximitySensor SphereSensor TimeSensor* TouchSensor* VisibilitySensor
- *behavior* – Script*
- *interpolators* – ColorInterpolator* CoordinateInterpolator NormalInterpolator OrientationInterpolator* PositionInterpolator* ScalarInterpolator

special effects

- *sound* – AudioClip Sound
- *light* – DirectionalLight Fog PointLight Spotlight

extensions

- *blaxxun* – Camera DeviceSensor Event KeySensor Layer2D Layer3D MouseSensor MultiTexture Particles TextureCoordGen

Not mentioned in this overview is the PROTO facility and the DEF/USE mechanism. The PROTO facility allows for defining nodes, by declaring an interface and a body implementing the node. Once a PROTO definition is given, instances of the PROTO can be created, in the same way as with built-in nodes. The DEF/USE maechanism may be applied for routing events as well as the reuse of fragments of code. Beware, however, that reuse using USE amounts to sharing parts of the scenegraph. As a consequence, one little change might be visible wherever that particular fragment is reused. In contrast, multiple instances of a PROTO are independent of eachother.

3D slides – the code

As you may have discovered, the material in this book is also available in the form of slides. Not Powerpoint slides but 3D slides, using VRML, with occasionally some graphic effects or 3D objects. At the Web3D Symposium 2002, I was asked *What is the secret of the slides?*. Well, there is no secret. Basically, it is just a collection of PROTOs for displaying text in VRML.[130]

protos

- *slideset* – container for slides
- *slide* – container for text and objects
- *slide* – container for lines of text
- *line* – container for text
- *break* – empty text

[130] The PROTOs were initially developed by Alex van Ballegooij, who also did the majority of the coding of an extended collection of PROTOs.

Note that for displaying 3D objects in a slide, we need no specific PROTO.

Before looking at the PROTO for a set of slides, let's look at the *slide* PROTO. It is surprisingly simple.

slide

```
PROTO slide [
   exposedField SFVec3f    translation 0 0 15
   exposedField SFRotation rotation    0 1 0 0
   exposedField SFVec3f    scale       1 1 1
   exposedField MFNode     children []
] {
   Transform {
      children    IS children
      translation IS translation
      rotation    IS rotation
      scale       IS scale
   }
}
```

The *slide* PROTO defines an interface which may be used to perform spatial transformations on the slide, like translation, rotation and scaling. The interface also includes a field to declare the content of the slide, that is text or (arbitrary) 3D objects.

The interface of the *slideset* PROTO allows for declaring which slides belong to the set of slides.

slideset

```
PROTO slideset [
   exposedField SFInt32    visible 0
   exposedField MFNode     slides [ ]
   eventIn SFInt32         next
] {
   DEF select Switch {
      choice      IS slides
      whichChoice IS visible
   }

   Script {
   ...
   }
}
```

Apart from the *visible* field, which may be used to start a presentation with another slide than the first one (zero being the first index in the array of slides), the *slideset* PROTO interface also contains a so-called *eventIn* named *next* to proceed to the next slide.

To select between the different slides a *Switch* node is used, which is controlled by a *Script*. The code of the script is given below.

script

```
Script {
directOutput TRUE
eventIn SFInt32 next IS next
field SFInt32 slide IS visible
field SFNode select USE select
field MFNode slides []
url "javascript:
function next(value) {
slides = select.choice;
Browser.print('=' + slide + ' ' + slides.length);
if (slide >= (slides.length-1)) slide = 0;
else slide += 1;
select.whichChoice = slide;
}"
}
```

In the interface of the script, we see both the use of *IS* and *USE* to connect the (local) script fields to the scenegraph. The function *next*, that implements the corresponding event, simply traverses through the slides, one step at a time, by assigning a value to the *whichChoice* field of the *Switch*.

example As an example of applying the *slide* PROTOs, look at the fragment below.

example

```
DEF slides slideset {
 slides [
    slide {
        children [
                text {
                        lines [
                          line { string ["What about the slide format?"] }
                          break { }
                          line { string ["yeh, what about it?"] }
                          break { }
                        ] # lines
                }
                Sphere { radius 0.5 }
        ] # children
    } # slide 1

    slide { # 2
        children [
                Sphere { radius 0.5 }
        ]
    } # slide 2
 ]  # slides
}
```

In the online version you may see how it works. (Not too good at this stage, though, since we have not included a proper background and viewpoint.)

For traversing between slides, we need a mechanism to send the *next* event to the *slideset* instance. In the current example, a timer has been used, defined by the code below.

timer

```
DEF time TimeSensor { loop TRUE cycleInterval 10 }
DEF script Script {
eventIn SFTime pulse
eventOut SFInt32 next
url "javascript: function pulse(value) { next = 1; }"
}
ROUTE time.cycleTime TO script.pulse
ROUTE script.next TO slides.next
```

Obviously, better interaction facilities are needed here, for example a simple button (which may be implemented using a *TouchSensor* and a *Sphere*) to proceed to the next slide. These extensions, as well as the inclusion of a background and viewpoint, are left as an exercise.

Naturally, the actual PROTOs used for the slides in this book are a bit more complex than the collection of PROTOs presented here. And, also the way slides themselves, that is the content, is different from what we have shown in the example. In appendix we will see how we can use XML to encode (the content) of slides. However, we will deploy the PROTOs defined here to get them to work.

B. XML-based multimedia

XML is becoming a standard for the encoding of multimedia data. An important advantage of XML-based encodings is that standard XML tools, such as XSLT stylesheet-based processing, are available. Another advantage is that the interchange of data becomes more easy. Examples of XML-based media formats include SMIL, X3D, Speech ML, Voice XML.

In fact, to my mind, we should have a course on XML-based multimedia. Zhisheng Huang, who developed the STEP language (and its XML-encoding) which is described in the next section, has compiled a list of topics that you should know about XML-based multimedia.

XML-based multimedia

- *introduction*: Extensible Markup Language (XML). Extensibility and profiling of web-based multimedia. Streaming. Model of timing and synchronization of web-based multimedia.
- *processing XML*: XSLT stylesheets, Java-based XML Processing, SAX, DOM, Java XSL object APIs
- *SMIL*: (Synchronized Multimedia Integration Language) SMIL modules: animation, content control, layout, linking, media object, metainformation, timing, and profiles.
- *X3D*: (XML-based VRML) Extensible 3D: architecture and based components, profile reference, translation between VRML and X3D. X3D examples: case studies.
- *VHML*: (Virtual Human Markup Language) Virtual Human Markup Language, Humanoid, H-anim specification, Speech Synthesis Markup Language Specification for the Speech Interface Framework (Speech ML), Voice Extensible Markup Language (VoiceXML). Text to Speech Technology.
- *STEP*: Scripting Technology for Embodied Persona and XSTEP, the XML-encoding of STEP and its processing tools. Embodied agents and multimedia presentation: theory, model, and practice.

The course should emphasize practice and experience. An example assignment is the development of an information system, including multimedia data in the form of images, 3D objects and audio recordings. The content should be organized according conceptual criteria, in an XML format to be designed by the student. Additional processing tools should then be written, using XSLT, to create a web site and to generate presentations in which the material is displayed from a particular perspective, for example a historic timeline, in one or more of the available presentation formats. See appendix **??** for a (more or less) concrete example.

As noted in the *research directions* of section **??**, XML comes with a set of related technologies. For processing XML we have XSLT, the transformation language which allows us to generate arbitrary text (including XML) from the information content of XML-encoded information. In the following, we will look at the use of XSLT to generate VRML-code from XML-encoded slides, using the collection of PROTOs developed in appendix .

3D slides in XML

To refresh your memory, a slide set is a collection of slides that may contain lines of text and possibly 3D objects. Writing slides in VRML would be rather tedious. Besides, slides written in VRML could not be used in, say, HTML pages.

So the solution I came up with is to isolate particular pieces in a text as slides and to process these slides to create a presentation. In effect, both dynamic HTML-based and VRML-based presentations are supported. As a notation, an XML-based encoding seems to be the most natural, since it very close already to HTML, thus reducing the amount of processing needed to convert text containing slides to HTML. Now, how should the conversion to VRML take place. The answer is, simply, by using XSLT.

Let's first look at the XML-encoding of the example slides of appendix .

slides in XML

```
<slideset>
<slide id="1">
<text>
<line>What about the slide format?</line>
<break/>
<line string="yeh, what about it">?</line>
</text>
<vrml>Sphere { radius 0.5 }</vrml>
</slide>
<slide id="2">
<vrml>Sphere { radius 0.5 }</vrml>
</slide>
</slideset>
```

One difference is that we introduced an *id* attribute in the *slide* tag, to allow for cross-referencing. These *id* attributes are, however, ignored in the conversion to VRML. Also, a *string* attribute has been introduced for the *line* tag. This is, however, just to illustrate how attributes are dealt with in processing XML files.

Before looking at the stylesheet used for the conversion to VRML, let me briefly say something about XSLT. The XSLT transformation language is a declarative language. It allows for processing an XML-encoded text by templates matching particular tags. In addition, the values of attributes of tags may be used when generating output.

The first part of our XSLT stylesheet looks as follows.

XSLT *stylesheet*

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="text"/>
```

Apart from the obligatory declaration that the stylesheet itself is written in XML, there is also the indication that the file is a stylesheet written according to the rules and conventions that can be found in a file dating from 1999, as given in the url. Since we do want to generate VRML (and not XML), we need to indicate that our output method is *text*, to avoid having an XML header at the start of the output.

Now we are ready to define our first template.

slideset

```
<xsl:template match="/slideset">

... load (extern) proto(s)

DEF slides slideset {
slides [
<xsl:apply-templates/>
] # slides
}

... include timer or user interface

</xsl:template>
```

Everything that is not part of a tag containing the *xsl* prefix is literally copied to output. In this fragment, I have not included the full PROTO declarations nor the timer or user interface needed to traverse the slides. In the middle of the fragment we see the *xsl* tag *apply-templates*. This results in further processing the content that is contained between the *slideset* begin and end tag, using the template definitions given below.

The template for the *slide* tag is simple.

slide

```
<xsl:template match="*/slide">
slide { children [
<xsl:apply-templates/>
] }
</xsl:template>
```

You will recognize the structure, which is in agreement with the way we encoded slides in VRML, as illustrated in appendix .

The template for the *text* is equally simple.

text

```
<xsl:template match="*/text">
text { lines [
<xsl:apply-templates/>
] }
</xsl:template>
```

For the *line* tag we need to do a bit more. Namely, we have to ask for the value of the *string* attribute, to obtain the complete result.

line

```
<xsl:template match="*/line">
line { string [ "<xsl:value-of select="@string"/>
<xsl:apply-templates/> " ] }
</xsl:template>
```

Note that, as mentioned above, the *string* attribute was just introduced to illustrate how to process attributes and is in itself superfluous. Actually, this way the *line* tag can be used as a closed tag, containing only the attribute and no contents, or an open tag with contents and possibly attributes.

Then, we are almost done.

etcetera

```
<xsl:template match="*/break">
line { string [ "<xsl:apply-templates/>" ] }
</xsl:template>

<xsl:template match="*/vrml">
<xsl:apply-templates/>
</xsl:template>

</xsl:stylesheet>
```

We need to define a template for the *break* tag and a template for the *vrml* tag, which does nothing but copy what is beteen the *vrml* begin and end tag.

And that's it. Check the online version for the resulting slides obtained by processing this specification with the XSLT stylesheet given above.

You may have wondered why no mention was made of a DTD or *schema*. Simply, because we do not need such a thing when processing an XML-file using XSLT stylesheets.

When you want to use XSLT to process your own XML-encoded information, you will probably want to know more about XSLT. That is a good idea. Consult Kay (2001) or one of the online tutorials.

D. a platform for intelligent multimedia

We have developed a platform for *intelligent multimedia*, based on distributed logic programming (DLP) and X3D/VRML. See Eliens et al. (2002). Now, before giving a more detailed description of the platform, let's try to provide a tentative definition of *intelligent multimedia*.

intelligent multimedia

> ... intelligent multimedia *provides a merge between technology from AI, in particular agent-technology, and multimedia ...*

However shallow this definition might be, it does indicate that we are in a multidisciplinary field of research that investigates how we may approach multimedia in a novel manner, using knowledge technology developed in Artificial Intelligence. More pragmatically, *intelligent multimedia* characterizes a programmatic approach to multimedia making use of high-level declarative languages, in opposition to low-level third generation and scripting languages, to reduce the programming effort involved in developing (intelligent) multimedia systems. Does this make the application themselves more intelligent? Not necessarily. In effect, nothing can be done that could not have been done using the available programmatic interfaces. However, we may argue that the availability of a suitable programming model makes the task (somewhat or significantly) easier.

In our Multimedia Authoring II course, students become familiar with our *intelligent multimedia* technology.

Multimedia Authoring II – virtual environments

- *intelligent services in virtual environments*

Knowledge of Web3D/VRML, as taught in Multimedia Authoring I, is a prerequisite. The course gives a brief introduction to logic programming in Prolog and DLP and then continues with building virtual environments using agent-technology to control the dynamic aspects of these environments.

distributed logic programming

The language DLP has a respectable history. It was developed at the end of the 1980s, Eliens (1992), and was implemented on top of Java at the end of the 1990s.

In retrospect, the language turned out to be an agent-programming language *avant la lettre*. What does it offer? In summary:

DLP

- *extension of Prolog*
- *(distributed) objects*
- *non-logical instance variables*
- *multiple inheritance*
- *multi-threaded objects*
- *communication by rendez-vous*
- *(synchronization) accept statements*
- *distributed backtracking*

Basically, the language is a distributed object-oriented extension of Prolog. It supports multiple inheritance, non-logical instance variables and multi-threaded objects (to allow for distributed backtracking). Object methods are collections of clauses. Method invocation is dealt with as communication by rendez-vous, for which synchronization conditions may be specified in so-called *accept* statements. As indicated above, the current implementation of DLP is built on top of Java. See Eliens (2000), appendix E for more details.

DLP+X3D platform

Our platform is the result of merging VRML with the distributed logic programming language DLP, using the VRML External Authoring Interface. This approach allows for a clear separation of concerns, modeling 3D content on the one hand and determining the dynamic behavior on the other hand. As a remark, recently we have adopted X3D as our 3D format. The VRML profile of X3D is an XML encoding of VRML97.

To effect an interaction between the 3D content and the behavioral component written in DLP, we need to deal with two issues:

- control points: *get/set* – position, rotation, viewpoint
- *event-handling* – asynchronous accept

We will explain each of these issues separately below. In addition, we will indicate how multi-user environments may be realized with our technology.

control points The control points are actually nodes in the VRML scenegraph that act as handles which may be used to manipulate the scenegraph. In effect, these handles are exactly the nodes that may act as the source or target of event-routing in the 3D scene. As an example, look at the code fragment below, which gives a DLP rule to determine whether a soccer player must shoot:

```
findHowToReact(Agent,Ball,Goal,shooting) :-
   get(Agent,position,sfvec3f(X,Y,Z)),
   get(Ball,position,sfvec3f(Xb,Yb,Zb)),
```

```
    get(Goal,position,sfvec3f(Xg,Yg,Zg)),
    distance(sfvec3f(X,Y,Z),sfvec3f(Xb,Yb,Zb),DistB),
    distance(sfvec3f(X,Y,Z),sfvec3f(Xg,Yg,Zg),DistG),
    DistB =< kickableDistance,
    DistG =< kickableGoalDistance.
```

This rule will only succeed when the actual distance of the player to the goal and to the ball satisfies particular conditions, see section **??**. In addition to observing the state of the 3D scene using the *get* predicate, changes to the scene may be effected using the *set* predicate.

event handling Our approach also allows for changes in the scene that are not a direct result of setting attributes from the logic component. Therefore we need some way to intercept events. In the example below, we have specified an observer object that has knowledge of, that is inherits from, an object that contains particular actions.

```
:- object observer : [actions].
var slide = anonymous, level = 0, projector = nil.

observer(X) :-
   projector := X,
   repeat,
     accept( id, level, update, touched),
   fail.

id(V) :-  slide := V.
level(V) :- level := V.
touched(V) :- projector←touched(V).
update(V) :- act(V,slide,level).
:- end_object observer.
```

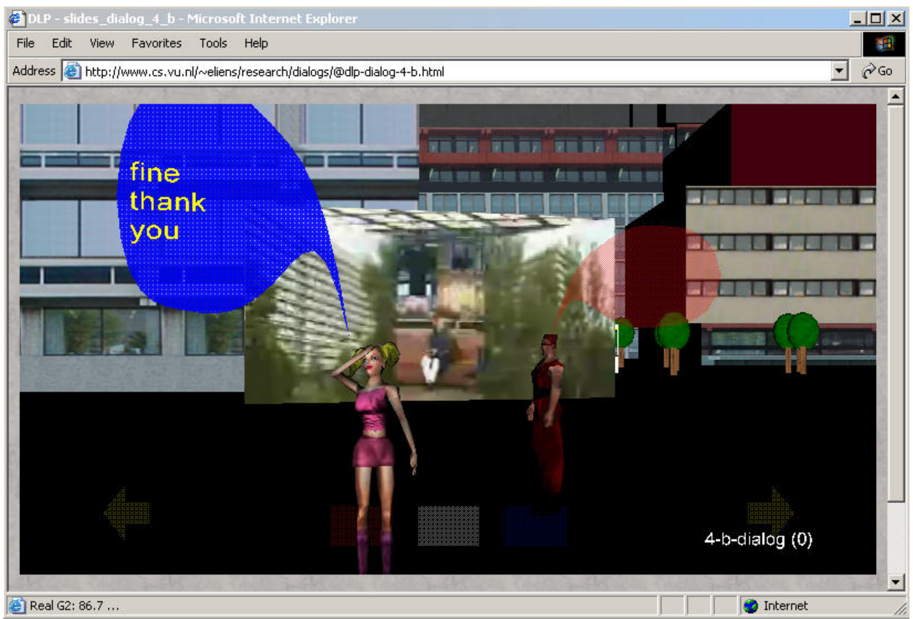
The constructor sets the non-logical variable *projector* and enters a repeat loop to accept any of the incoming events for respectively *id*, *level*, *update* and *touched*. Each event has a value, that is available as a parameter when the corresponding method is called on the acceptance of the event. To receive events, the *observer* object must be installed as the listener for these particular events.

The events come from the 3D scene. For example, the *touched* event results from mouse clicks on a particular object in the scene. On accepting an event, the corresponding method or clause is activated, resulting in either changing the value of a non-logical instance variable, invoking a method, or delegating the call to another object.

An observer of this kind is used in the system described below, to start a comment (dialog) on the occurrence of a particular slide.

case studies

To illustrate the potential of our DLP+X3D platform, we will briefly sketch two additional case studies deploying embodied agents, respectively the use of dialogs in VR presentations (fig. a), and a scripting language for specifying gestures and movements for humanoids (fig. b).



dialogs in virtual environments

Desktop VR is an excellent medium for presenting information, for example in class, in particular when rich media or 3D content is involved. At VU, I have been using *presentational VR* for quite some time, and recently I have included dialogs using balloons (and possibly avatars) to display the text commenting on a particular presentation. See figure (b) for an example displaying a virtual environment of the VU, a propaganda movie for attracting students, and two avatars commenting on the scene. The avatars and their text are programmed as annotations to a particular scene as described below.

Each presentation is organized as a sequence of slides, and dependent on the slides (or level within the slide) a dialog may be selected and displayed. See the *observer* fragment presented above.

Our annotation for dialog text in slides looks as follows:

```
<phrase right="how~are~you">
<phrase left="fine~thank~you"/>
<phrase right="what do~you think~of studying ..."/>
...
```

```
<phrase left="So,~what~are you?"/>
<phrase right="an ~agent" style="[a(e)=1]"/>
<phrase left="I always~wanted to be~an agent" style="[a(e)=1]"/>
```

In figure (b), you see the left avatar (named *cutie*) step forward and deliver her phrase. This dialog continues until *cutie* remarks that she *always wanted to be an agent*. The dialog is a somewhat ironic comment on the contents of the movie displayed, which is meant to introduce the VU to potential students.[131]

Furthermore, there are a number of style parameters to be dealt with to decide for example whether the avatars or persona are visible, where to place the dialogs balloons on the display, as well as the color and transparancy of the balloons. To this end, we have included a *style* attribute in the *phrase* tag, to allow for setting any of the style parameters.

Apart from phrases, we also allow for gestures, taken from the built-in repertoire of the avatars. Below we discuss how to extend the repertoire of gestures, using a gesture specification language.

Both phrases and gestures are compiled into DLP code and loaded when the annotated version of the presentation VR is started.

STEP – a scripting language for embodied agents

Given the use of humanoid avatars to comment on the contents of a presentation, we may wish to enrich the repertoire of gestures and movements to be able, for example, to include gestural comments or even instructions by gestures.

Recently, we have started working on a scripting language for humanoids based on dynamic logic. The STEP scripting language consists of basic actions, composite operators and interaction operators (to deal with the environment in which the movements and actions take place).

The basic actions of STEP consist of:

- *move* – `move(Agent,BodyPart,Direction,Duration)`
- *turn* – `turn(Agent,BodyPart,Direction,Duration)`

These basic actions are translated into operations on the control points as specified by the H-Anim 1.1 standard.

As composite operators we provide sequential and parallel composition, as well as *choice* and *repeat*. These composite operators take both basic actions and user-defined actions as parameters.

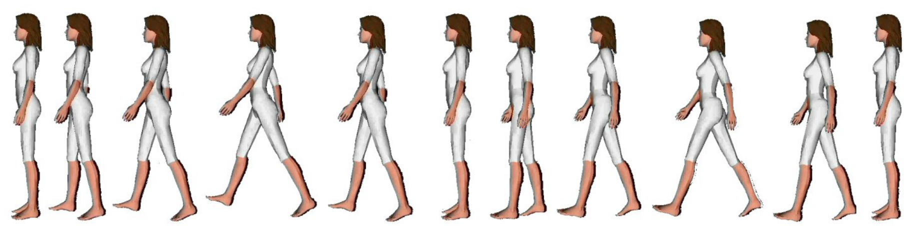
Each action is defined using the *script*, by specifying an action list containing the (possibly compound) actions of which that particular action consists. As an example, look at the definition of *walking* below.

```
script(walk(Agent), ActionList) :-
ActionList = [
  parallel([turn(Agent,r_shoulder,back_down2,fast),
        turn(Agent,r_hip,front_down2,fast),
```

[131] Clearly, our approach is reminiscent to the notorious *Agneta & Frida* characters developed in the Persona project. See the *research directions* of section .

```
              turn(Agent,l_shoulder,front_down2,fast),
              turn(Agent,l_hip,back_down2,fast)]),
       parallel([turn(Agent,l_shoulder,back_down2,fast),
              turn(Agent,l_hip,front_down2,fast),
              turn(Agent,r_shoulder,front_down2,fast),
              turn(Agent,r_hip,back_down2,fast)])
     ], !.
```

Notice that the *Agent* that is to perform the movement is given as a parameter. (Identifiers starting with a capital act as a logical parameter or variable in Prolog and DLP.)



Interaction operators are needed to conditionally perform actions or to effect changes within the environment by executing some command. Our interaction operators include: *test*, *execution*, *conditional* and *until*.

Potentially, an action may result in many parallel activities. To control the number of threads used for an action, we have created a scheduler that assigns activities to a thread from a thread pool consisting of a fixed number of threads.

As a demonstrator for STEP, we have created an instructional VR for *Tai Chi*, the Chinese art of movement.

XML encoding Since we do not wish to force the average user to learn DLP to be able to define scripts in STEP, we are also developing XSTEP, an XML encoding for STEP. We use *seq* and *par* tags as found in SMIL, as well as *gesture* tags with appropriate attributes for speed, direction and body parts involved. As an example, look at the XSTEP specification of the *walk* action.

```
<action type="walk(Agent)">
<seq>
 <par speed="fast">
 <gesture type="turn" actor="Agent" part="r_shoulder" dir="back_down2"/>
  ...
 </par>
 <par speed="fast">
  ...
 <gesture type="turn" actor="Agent" part="r_hip" dir="back_down2"/>
 </par>
</seq>
</action>
```

Similar as with the specification of dialog phrases, such a specification is translated into the corresponding DLP code, which is loaded with the scene it belongs to. For XSTEP we have developed an XSLT stylesheet, using the Saxon package, that transforms an XSTEP specification into DLP. We plan to incorporate XML-processing capabilities in DLP, so that such specifications can be loaded dynamically.

related work

There is an enormous amount of research dealing with virtual environments that are in one way or another inhabited by embodied agents. By way of comparison, we will discuss a limited number of related research projects.

As systems that have a comparable scope we may mention Broll (1996) and DIVE, that both have a client-server architecture for realizing virtual environments. Our DLP+X3D platform distinguishes itself from these by providing a uniform programmatic interface, uniform in the sense of being based on DLP throughout.

The Parlevink group at the Dutch University of Twente has done active research in applications of virtual environments with agents. Their focus is, however, more on language processing, whereas our focus may be characterized as providing innovative technology.

Both Tarau (1999) and Davison (2001) deal with incorporating logic programming within VRML-based scenes, the former using the External Authoring Interface, and the latter inline logic scripts. Whereas our platform is based on distributed objects, Jinni deploys a distributed blackboard to effect multi-user synchronisation.

Our scripting language may be compared to the scripting facilities offered by Alice, which are built on top of Python. Also, *Signing Avatar* has a powerful scripting language. However, we wish to state that our scripting language is based on dynamic logic, and has powerful abstraction capabilities and support for parallelism.

Finally, we seem to share a number of interests with the VHML community, which is developing a suite of markup languages for expressing humanoid behavior. We see this activity as complementary to ours, since our research proceeds from technical feasibility, that is how we can capture the semantics of humanoid gestures and movements within our dynamic logic, which is implemented on top of DLP.

future research

In summary, we may state that our DLP+X3D platform is a powerful, flexible and high-level platform for developing VR applications with embodied agents. It offers a clean separation of modeling and programming concerns. On the negative side, we should mention that this separation may also make development more complex and, of course, that there is a (small) performance penalty due to the overhead incurred by using the External Authoring Interface.

Where our system is currently lacking, clearly, is adequate computational models underlying humanoid behavior, including gestures, speech and emotive characteristics. The VHML effort seems to have a rich offering that we need to digest in order to improve our system in this respect.

Our choice to adopt open standards, such as XML-based X3D, seems to be benificial, in that it allows us to profit from the work that is being done in other communities, so that we can enrich our platform with the functionality needed to create convincing embodied agents in a meaningful context.

D. resources, tools and technology

What do you need to have to start working on your multimedia project? that depends, naturally, on what you want to do. In the following, I will give a brief overview of resources, tools and technolgies that you might find useful or that you might want to explore. This overview consists mainly of urls and a brief characterization and in some cases an indication of a price range.

This overview is definitely not meant to be complete, and is only included for your convenience, so that you don't have to *google*[132] it yourself. In the online version of the book more (online) resources are given, as well as a (clickable) list of all urls that appear (as a footnote) in the book.

resource(s)

This section contains a variety of itmes, including a selection of online tutorials and thesauri. Some examples are given of online museum tours and listings are included of the media art and cultural heritage institutes mentioned in the book. But we will start with introducing briefly with what you need for 3D authoring and rendering, since this is what we have primarily focused on in theis book.

3D authoring & conversion

- vrmlpad – www.parallelgraphics.com/products/vrmlpad
- polytrans – www.okino.com/products.htm
- maya – www.alias.com
- 3dsmax – www.discreet.com
- sketchup – sketchup.google.com/download.html
- flux studio – www.mediamachines.com/products.html

The *polytrans* tool from Okino has been included, since it allows you to convert almost any format into you format of choice, which is a great asset for (re) using models.

[132] www.google.com

3D rendering

- blaxxun – www.blaxxun.com/en/products/contact
- virtools – www.virtools.com
- flux web3d – sourceforge.net/projects/flux
- mediamachines flux – www.mediamachines.com/products.html

As concerns price, VRML-based solutions for authoring and rendering are clearly low-cost, whereas tools such as *Maya* and *Studio Max* require more investment, not only in money but also in learning time. Also *Virtools* is in the higher price range.

tutorials

- html – www.mcli.dist.maricopa.edu/tut
- javascript – www.javascriptkit.com
- php – www.php.net/docs.php
- rdf – www.w3.org/TR/rdf-primer
- vrml – web3d.vapourtech.com/tutorials/vrml97
- java – java.sun.com/docs/books/tutorial
- 3D modeling – www.raph.com/3dartists/tutorials/t-3dsmax.html
- games in VRML – www.3dezine.com/3DEZine/gamestory.html
- ria – www.macromedia.com/resources/business/rich_internet_apps/whitepapers.html

In many cases it is (more) convenient to have working examples at hand. Personally, I advice my students to learn using HTML, VRML, Javascript and the like from one of the online tutorials, which do provide such examples. The *php* documentation is not really a tutorial but does provide useful help and examples.

visual design

- collage – www.artlex.com/ArtLex/c/collage.html
- storyboard – www.thestoryboardartist.com/links.html
- drawing – www.thestoryboardartist.com/tutorial.html

For *visual design* it might be worthwhile to look at some examples, or even take a complete course in drawing.

museum

- van gogh – www.vangoghmuseum.nl
- rijksmuseum – www.rijksmuseum.nl
- canada – www.virtualmuseum.ca/English/index_flashFT.html
- zkm – www.zkm.de
- tate – www.tate.org.uk
- louvre – www.louvre.fr

More inspiration can perhaps be obtained from looking at what musea have to offer. It also gives you an opportunity to update your knowledge of the history of art.

media art

- montevideo – www.montevideo.nl
- V2 – www.v2.nl
- electronic arts intermix – www.eai.org/eai
- cinemanet – www.cinemaneteurope.com
- variable media – www.variablemedia.net
- net art – wwwwwwwww.jodi.org/100cc/index.html
- mediamatic – www.mediamatic.net

Listed above are institutions that play a role in the preservation and dissemination of contemporary media art. Not an institution, but an early pioneer of art on the internet, is *jodi* from *net art*.

virtual tours

- amsterdam – www.channels.nl
- panoramic amsterdam – www.panoramsterdam.nl
- rijksmuseum – www.rijksmuseum.nl/collectie/meesterwerken/?lang=en
- groningen – www.kalamiteit.nl/world/no_cache/museum/vrml/connect.html
- mondriaan – www.artmuseums.harvard.edu/mondrian

Many cities nowadays have virtual tours. And also many musea allow the (online) visitor to have a look at their collection.

cultural heritage

- incca – www.incca.org
- delos – www.delos.info
- echo – echo.mpiwg-berlin.mpg.de/home
- eu – www.iue.it/ECArchives
- cidoc – www.cidoc.icom.org
- collate – www.collate.de
- cimwos – www.xanthi.ilsp.gr/cimwos
- library of congress – www.loc.gov./
- scriptorium – sunsite.berkeley.edu/scriptorium
- tei – www.tei-c.org
- open archives – www.tei-c.org
- topia – topia.telin.nl

Above is a mixed collection of references to organizations and institutions that are in some way involved in cultural heritage projects, either related to traditional art or contemporary art.

thesaurus

- webopedia – www.webopedia.com
- visual – www.visualthesaurus.com
- 3D glossary – www.nvidia.com/page/pg_20010527107687.html
- art & architecture – www.getty.edu/research/conducting_research/vocabularies/aat/
- modern art – en.wikipedia.org/wiki/Modern_art
- (new) media art – en.wikipedia.org/wiki/New_Media_art
- art online – www.art-online.com
- multimedia – www.insead.fr/CALT/Encyclopedia/Media/multimedia.html
- virtual reality – www.insead.fr/CALT/Encyclopedia/ComputerSciences/VR
- gaming – www.insead.fr/CALT/Encyclopedia/ComputerSciences/Gaming
- mathematics – www.cs.brown.edu/people/scd/facts.html
- mpeg – www.m4if.org/mpeg4
- wikipedia – en.wikipedia.org/wiki/Multimedia

There is a wealth of online information sources, including glossaries and thesauri. Beware, not all of them are properly authorized. Nevertheless, it might be interesting to note that the online version of this book is referred to in the *wikipedia*, for the entry *multimedia*.

games

- gamasutra – www.gamesutra.com
- gamedev – www.gamedev.net
- developer – www.gdmag.com/resources.html
- and more – www.lostlogic.com/postnuke
- games at school – www.freewebs.com/schoolgamemaker
- gamemaker – www.gamemaker.nl/
- game learning – www.gamelearning.net
- scripting – htpp://www.lua.org
- open source – www.delta3d.org
- free source – www.thefreecountry.com/sourcecode/games.shtml

For games, there are several popular sites providing information about new upcoming games, as well as developer's resources, including software available for download.

A recommended open source game engine is *Delta3D*. This package contains a variety of open source software, well-integrated due to the efforts of a dedicated team at the Naval Postgraduate School in Monterey, CA/USA.

serious games

- play2learn – www.play2learn.nl
- nitrogenius – www.serc.nl/play2learn/products/nitrogenius
- at school – rla.oakland.edu/∼ist_699
- primary games – www.primarygames.com

- games at school – www.freewebs.com/schoolgamemaker
- arcade – www.educationarcade.org
- never winter – nwn.bioware.com

Serious games are a new brand of games. Not really new in terms of technology, but new with respect to focus and intent.

tool(s)

There is a great variety of tools, with huge differences in prize. Often, however, you can download a fully functional trial version that will last for a month, and thus may determine the length of your project. A number of tools, however, come with a free (such as Maya) or limited price (such as 3DSMax) student version.

imaging and graphics

- photoshop – www.adobe.com/products/photoshop
- illustrator – www.adobe.com/products/illustrator
- snagit – www.techsmith.com/products/snagit
- camtasia – www.techsmith.com/products/studio

Perhaps the most popular tools among designers are *photoshop* and *illustrator*. Both for capture and image catalogue maintenance I have benefited from *snagit* and *camtasia*, both from *techsmit*.

3D modeling

- vrmlpad – www.parallelgraphics.com/products/vrmlpad
- polytrans – www.okino.com/products.htm
- maya – www.alias.com
- 3dsmax – www.discreet.com
- houdini – www.sidefx.com
- bodystudio – www.reiss-studio.com
- poser – www.curious-labs.com

In addition to the modeling tools already mentioned before, there are many additonal tools and add-ons, such as *houdini* for procedural modeling, *bodystudio* for importing poser models in maya, 3dsmax and other tools, and *poser*, a somewhat outdated tool voor modeling humanoids, with a large collection of ready-made feature material.

Alias Wavefront Maya

- information – www.alias.com
- tutorials – www.alias.com/eng/community/tutorials
- community – www.alias.com/eng/community

A high end 3D modeling tool, with a respectable history and a large community of users. It is in the high end price range and requires significant effort to master.

Discreet 3D Studio Max

- information – www.discreet.com
- tutorials – www.pixel2life.com/tutorials/3dsmax.php?tut=16
- vrml – www.dform.com/inquiry/tutorials/3dsmax

Popular within the game community, *studio max* which includes *character studio* appears to be somewhat more straightforward than maya.

technology

Again, the technology overview is certainly not exhaustive. There are many commercial game engines that are well worth looking at when you engage in a real project. I have included a limited number of open source libraries and toolkits to provide you with a starting point for further exploration.

DirectX SDK 9

- information – www.microsoft.com/directx
- show + 3d – msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwmt/html/vmr_d3d.asp
- SDK – msdn.microsoft.com/library/default.asp?url=/library/en-us/directx9_c/directx/directx9cpp.asp
- frames – www.jkarlsson.com/Articles/loadframes.asp
- animation controller – www.jkarlsson.com/Articles/animation.asp

Direct X is an advanced, yet complicated multimedia platform. The managed code version is significantly less powerful than the C++ version. As indicated in section 4.2 there is a great many of books about DirectX. Some helpful online tutorials are listed above.

Wild Tangent

- information – www.wildtangent.com
- developers – www.wildtangent.com/developer

Wild Tangent is very appropriate for developing games. It provides conveniece layer around DirectX 7, and enables applications to be run via a Web browser, by deploying the COM interfaces for DirectX. It allows for authoring content and dynamics in Javascript and Java. However, also the original X meshes, the file format for DirectX, can be used.

Virtools Software Suite

- information – www.virtools.com

Virtools is announced to be a *comprehensive development platform, for games, virtual reality/simulations and marketing/multimedia applications.*

OpenML

- information – www.khronos.org/openml

OpenML might be the candidate platform for those that wish to develop platform-independent (read non Microsoft windows-tied) multimedia applications. It is *a royalty-free, cross-platform programming environment for capturing, transporting, processing, displaying, and synchronizing digital media - including 2D/3D graphics and audio/video streams. OpenML 1.0 defines professional-grade sample-level stream synchronization,* *OpenGL extensions for accelerated video processing, the MLdc professional display control API and the ML framework for asynchronous media streaming between applications and processing hardware.*

open source technology

- plib – plib.sourceforge.net
- OpenSceneGraph – www.openscenegraph.org
- OpenSound – www.cnmat.berkeley.edu/OpenSoundControl
- ARToolkit – artoolkit.sourceforge.net
- Mixed Reality Toolkit – www.cs.ucl.ac.uk/staff/rfreeman
- OpenNap – opennap.sourceforge.net
- ImageMagick – www.imagemagick.org
- *cygwin* – www.cygwin.com

There are many open source software toolkits and libraries. My experience with these is mixed. Anyway, when you start working with these make sure that you have sufficient programming skills, and patience. But then the results might be better than you could obtain with more expensive commercial technology. If you run Linux, then open source is probably the only way to go. For windows users, with a unix background, there is *cygwin*, which is a linux-like environment for windows.

XML

- XML Entities – tech.irt.org/articles/js212
- W3C – www.w3.org/Style/XSL
- resources – www.xml.org/xml/resources_cover.shtml
- saxon – saxon.sourceforge.net
- online tutorial – www.zvon.org/HTMLonly/XSLTutorial/Books/Book1/index.html
- Xeena XML editor – www.alphaworks.ibm.com/tech/xeena
- X3D Edit setup – sdk.web3d.org/spring2002disk2/tools/X3D-Edit/index.html

For XML there is a number of generic editors, such as Xeena, which has been adapted for X3D, see appendix B. There are also XSLT processing tools, such as *saxon*, which is the only one I have experience with.

Java

- information – http://www.javasoft.com
- art with Java – http://java.khm.de
- java media framework – http://java.sun.com/products/java-media/jmf/2.1.1/guide/JMFTOC.html
- slide show – http://developer.java.sun.com/developer/technicalArticles/Threads/applet/index.html
- basics – http://developer.java.sun.com/developer/onlineTraining/Programming/BasicJava1/compile.html
- tutorial – http://java.sun.com/docs/books/tutorial/index.html
- advanced – http://developer.java.sun.com/developer/onlineTraining/Programming/JDCBook/
- sound API – http://java.sun.com/products/java-media/sound/samples/JavaSoundDemo
- imaging – http://developer.java.sun.com/developer/technicalArticles/Media/AdvancedImage

Java is the programming language of choice for many computer science curricula. It is a well-documented, relatively easy to use language and the java framework provides a rich collection of libraries. There is also a version for mobile platforms.

student multimedia facilities

To conclude this overview of resources, tools and technologies, I have included a brief description of the student facilities we have for multimedia work at the Vrije Universiteit, spring 2005.

computers 14 fujitsu siemens scenico P320, AMD64 3400+ MHz, 1G memory, 80 GB serial ATA disk, 6 x USB, XFX Geforce 6600 GT 128 Mb AGP, dual display, 2 LCD monitors.

software

- Parallel Graphics VrmlPad – site license
- Alias Maya Complete (5.0 & 6.0) – 10 floating licenses
- 3D Studio Max 7 – 15 floating licenses
- DirectX9c SDK – http://www.microsoft.com/directx
- WildTangent WebDriver & SDK – http://www.wildtangent.com/developer
- CG Toolkit – http://developer.nvidia.com/page/tools.html
- RenderMonkey & SDK – http://www.ati.com/developer/rendermonkey
- Llustrator & Photoshop CS – http://www.adobe.com

There is no need to say that this is not the end of the story. More software will be installed, among which *virtools*, hopefully soon. And whenever the opportunity is there, we will no doubt upgrade to more powerful hardware as well!

E. write an essay!

Even when you prefer to do practical work, it might well pay off to take a step back, reflect on your approach and and study one aspect of multimedia in more detail. When you plan to work in an academic situation, it is very likely that at some point you must report about your work and provide some theoretical context to it. These few closing paragraphs are meant to give you some hints about how to approach writing a paper or report.

Independent of how you tackle the process of collecting material, organizing notes and writing it all down, keep in mind that the end result must consist of:

outline

title – *indicating the topic*
name – *to tell who you are*
abstract – *giving the 'message' of your efforts*
introduction – *clarifying the approach and structure*
background – *explaining the context of the subject*
sections – *to elaborate on the subject*
related work – *characterizing related approaches*
conclusion(s) – *summarizing the main point(s)*
references – *listing the literature you consulted*
appendices (optional) – *providing extra information*

It is surprising how often students forget, for example, an abstract or a proper introduction. Often the familiarity with the material, built up when working with it, seems to make them forget that for the reader these items are important and cannot be missed to grasp the point(s) of their efforts. Also, I wish to note that, although the discipline of giving references is in computer science much less strict than in, for example, philosophy, sufficiently clear references are necessary for the reader to check and verify your claims.

AS I already indicated I do not wish to elaborate on how to gather material, how to organize your collection of potentially useful notes, or how to convert these notes into readable text. Rather, I wish to to discuss the distinction, or tension, between form and content. Form, I would say, is determined by the perspective from which you approach the material and the goal you set yourself when writing the paper or report. Possible perspectives, or if you prefer forms, are:

perspective(s)

- review/background – *sketch perspectives, history, viewpoints*
- case study – *analyse assumptions, gather empirical data, and explain!*
- technical analysis – *technology-oriented, work out the details*

- formal study – *clarify in a formal manner, conceptualize and formalize*
- tutorial – *explain for the laymen, but do it very good*

To be clear, the phrase perspectives as used here is only vagely related to the use of perspectives when used to introduce the parts, where it meant to indicate the scientific discipline or point of view from which to look at a particular topic.

Content, as opposed to form, may be characterized as the collection of possible subjects, which in the area of multimedia include authoring, digital convergence, standards and information retrieval. Obviously, some subjects are better matched with particular forms or perspectives than others. For example, a formal study is suitable for discussing standards, but, to my mind, less so for explaining multimedia authoring. To get an idea of how I look at the problem of reconciling form and content when writing a paper about multimedia, consult the matrix:

	authoring	convergence	standards	retrieval
review/background	-	++	++	+
case study	+	+	+	+
technical analysis	-	++	++	++
formal study	-	-	++	-
tutorial	–	-	?	-

You may wonder why I don't think of tutorials as a suitable form for writing about multimedia. Well, in fact I do think that the form of a tutorial is an excellent way to write about multimedia technolgy, but it is not a very rewarding form for getting academic credits. When you want to be an acdemic, you'd better learn to write a technical analysis or case study. However, by that time perhaps the scientific paper generators[133] might have matured to the extent that writing has become a superfluous activity.

[133] www.pdos.lcs.mit.edu/scigen

references

Adams J. (2002), *Programming Role Playing Games with DirectX*, Premier Press

Adams J. (2003), *Advanced Animation with DirectX*, Premier Press

Alexander T., ed. (2005), *Massively Multiplayer Game Development 2*, Charles River Media

Anders P. (1999), *Envisioning Cyberspace – Designing 3D Electronic Spaces*, McGraw-Hill

Angel E. (1997), *Interactive Computer Graphics – A top-down approach with OpenGL*, Addison-Wesley

Arnheim R. (1957), *Film as Art*, The University of California Press

Astleneir H. (2000), Designing emotionally sound instruction: the FEASP approach, Instructional Science 28, pp. 169-198

Auster P. (2004), *City of Glass*, Faber and Faber, adaptation by Karasik P. and Mazzucchelli D.

Badler N., Bindiginavale R., Bourne J., Palmer M., Shi J., Schuler W. (1998), A Parameterized Action Representation for Virtual Human Agents, Workshop on Embodied Conversational Characters, WECC98, Lake Tahoe, CA,Oct 12-15, 1998.

Baeza-Yates R. and Ribeiro-Neto B. (1999), *Modern Information Retrieval*, Addison-Wesley, 1999

Ballegooij A. van and Eliens A. (2001), Navigation by Query in Virtual Worlds, Web3D 2001 Conference, Paderborn, Germany, 19-22 Feb 2001

Barcelo S.A., Forte M., Sanders D.H., eds. (2000), *Virtual Reality in Archeology*, Bar International Series 843, 2000

Barron T. (2003), *Strategy Game Programming with DirectX 9.0*, Wordware Publishing

Baxter W., Wendt J., and Lin M. (2004), IMPaSTo: A Realistic, Interactive Model for Paint, Proc. of NPAR 2004, The 3rd Int. Symp. on Non-Photorealistic Animation and Rendering. June 7-9 2004, Annecy, France. pp. 45–56.

Berners-Lee T., Hendler J., Lassila O. (2001), The semantic web, Scientific American, may 2001, pp. 28-37

Betsky A. with Eeuwens A. (2004), *False Flat – Why Dutch Design is so Good*, Phaidon

Bolter J.D and Grusin R. (2000), *Remediation – Understanding New Media*, MIT Press

Boncz P.A. and Kersten M.L. (1995), Monet – an impressionist sketch of an advanced database system, In Proc. BIWIT95

Bongers B. (2006), *Interactivation – Towards an e-cology of people, our environment, and the arts*, Ph.D. Thesis, Vrije Universiteit

Bordwell D. and Thomson K. (2003), *Film art, an introduction*, McGraw-Hill, 7th edn.

Briggs A. and Burke P. (2001), *A social history of the media – from Gutenberg to the Internet* , Polity Press

Broll W. (1996), VRML and the Web: A basis for Multi-user Virtual Environments on the Internet, In Proceedings of WebNet96, H. Maurer (ed.), AACE, Charlottesville, VA (1996), 51-56.

Broll W., Shäfer L., Höllerer T., Bowman D. (2001), Interface with Angels: the future of VR and AR interfaces, IEEE Computer Graphics, November/December 2001, pp. 14-17

Bruner J.S. (1972), *Relevance of Education*, Penguin Education

Bush V. (1945), As we may think, Atlantic Monthly, July 1945

CACM 44:3 (2001), The next 1000 years, Communications of the ACM, March 2001, 44:3

Capin T.K., Pandzic I.S., Magnenat-Thalmann N., Thalman D. (1999), *Avatars in Networked Virtual Environments*, Wiley

Carey R. and Bell G. (1997), *The Annotated VRML 2.0 Reference Manual*, Addison-Wesley

Cesa-Bianchi N. and Lucosi G. (2006), *Prediction, Learning, and Games*, Cambridge University Press

Chang S.C. and Costabile M.F. (1997), Visual Interfaces to Multimedia Databases, In [_Handbook]

Chapman N. and Chapman J. (2004a), *Digital Multimedia*, Wiley, 2nd edn.

Chapman N. and Chapman J. (2004b), *Digital Media Tools*, Wiley, 2nd edn.

Cheng F. (2006), *Five meditations on beauty (in french)*, Editions Albin Michel

Christel, M., Olligschlaeger, A., Huang, C. (2000), Interactive maps for digital video, IEEE Multimedia 7(1), pp. 60-67

Churchill E.F., Snowdon D.N. and Munro A.J., eds. (2001), *Collaborative Virtual Environments – Digital Places and Spaces for Interaction*, Springer

Conger D. (2004), *Physics Modeling*, Thomson

Conklin J. (1987), Hypertext: An Introduction and Survey, *IEEE Computer* 20(9), pp. 17-41

Danaher S. (2004), *Digital 3D Design*, Thomson

Davenport G.(2000), Your own virtual story world, Scientific American, november 2000, pp. 61-64

Davison A. (1993), A survey of Logic Programming-based Object-Oriented Languages, In Research Directions on Concurrent Object-Oriented programming, G. Agha, P. Wegner, A. Yonezawa (eds.), MIT Press (1993)

Davison A. (2001), Enhancing VRML97 Scripting, Euromedia2́001, Valencia, Spain, April 18-20. available from: http://fivedots.coe.psu.ac.th/∼ad

Davison A. (2001b), Logic Programming Languages for the Internet, Chapter in Computational Logic: From Logic Programming into the Future, Antonis Kakas, Fariba Sadri (eds.), Springer Verlag (2001). available from: http://fivedots.coe.psu.ac.th/∼ad

Diehl G. (1973), *Vasarely*, Crown Publishers Inc.

Diehl S. (2001), *Distributed Virtual Worlds – Foundations and Implementation Techniques using VRML, Java and Corba*, Springer

Dijkstra K., Zwaan R. Graesser A.and J. Magliano (1994), Character and reader emotions in literary texts, Poetics 23, pp. 139-157

Dix A., Finlay J., Abowd G., Beale R. (1998), *Human-Computer Interaction (1998)*, Prentice Hall, 2nd edn.

Dodge M. and Kitchin R. (2000), *Mapping Cyberspace*, Routledge

Dodge M. and Kitchin R. (2002), *Atlas of Cyberspace*, Addison-Wesley

Earnshaw R. and Vince J., eds. (2002), *Intelligent Agents for Mobile and Virtual Media*, Springer

Eberly D.H. (2001), *3D Game Engine Design*, Morgan Kaufmann Publishers

Eberly D.H. (2004), *Game Physics*, Morgan Kaufmann Publishers

Eco U. (1994), *Six walks in the fictional woods*, Harvard University Press

Eliens A. (1988), Computational Art, Leonardo, MIT Press

Eliens A. (1992), DLP – A language for Distributed Logic Programming, Wiley

Eliens A. (1996), workshop: Logic Programming and the Web, http://www.cs.vu.nl/∼eliens/online/workshops/www6

Eliens A., Welie M. van, Ossenbruggen J. van and Schonhage S.P.C (1997) , Jamming (on) the Web, In: Proceedings of the 6th International World Wide Web Conference — Everone, Everything Connected, OŔeilly and Associates, Inc. April 1997, pp. 419-426

Eliens A., Niessink F., Schonhage S.P.C., van Ossenbruggen J.R., Nash P. (1996), Support for BPR – simulation, hypermedia and the Web, Proceedings Euromedia9́6, Euromedia, London 1996

Eliens A. (2000), *Principles of Object-Oriented Software Development*, Addison-Wesley Longman, 2nd edn.

Eliens A. (2002), intelligent multimedia @ VU, http://www.cs.vu.nl/∼eliens/research

Eliens A., Huang Z., and Visser C. (2002), A platform for Embodied Conversational Agents based on Distributed Logic Programming, AAMAS Workshop – Embodied conversational agents - letś specify and evaluate them!, Bologna 17/7/2002

Eliens A., Dormann C., Huang Z. and Visser C. (2003), A framework for mixed media – emotive dialogs, rich media and virtual environments, Proc. TIDSE03, 1st Int. Conf. on Technologies for Interactive Digital Storytelling and Entertainment, Göobel S. Braun N.,n Spierling U., Dechau J. and Diener H. (eds>), Fraunhofer IRB Verlag, Darmstadt Germany, March 24-26, 2003

Eliens A., Huang Z., Visser C. (2005), Presentational VR – *What is the secret of the slides?*, in preparation

Eliens A., van Riel C., Wang Y. (2006), Navigating media-rich information spaces using concept graphs – the abramovic dossier, accepted for: International Conference on Multidisciplinay Information Sciences and Technologies (InSciT2006) October, 25-28th 2006, Merida, Spain www.instac.es/inscit2006

Eliens A. (2006b), Odyssee – explorations in mixed reality theatre, accepted for: GAME'ON-NA'2006, September 19-21, 2006 - Naval Postgraduate School, Monterey, USA

Eliens A., S.V. Bhikharie (2006), game @ VU – developing a masterclass for high-school students using the Half-life 2 SDK, GAME'ON-NA'2006, September 19-21, 2006 - Naval Postgraduate School, Monterey, USA

Engel W.F. (2003), *Beginning Direct3D Programming*, Premier Press

Engel W.F., ed. (2004a), *Shader X2, Introduction & Tutorials with DirectX 9*, Wordware Publishing

Engel W.F., ed. (2004b), *Shader X2 – Shader Programming, Tips & Tricks with DirectX 9*, Wordware Publishing

Engel W.F., ed. (2005), *Shader X3, Advanced Rendering with DirectX and OpenGL*, Charles River Media

Engelbart D. (1963), A Conceptual Framework for the Augmentation of Manś Intellect, Vistas in Information Handling 1(9)

England E. and Finney A. (1999), *Managing Multimedia – project Management for Interactive Media*, Addison-Wesley

Entabaclaz A. (2003), *Les metamorphoses dÚlysse – reecritures de lÓdyssee*, Editions Flammarion, Paris

Ericson C. (2005), *Real-time Collision Detection*, Morgan Kaufmann Publishers

Fan J., Ries E. and Tenitchi C. (1996), *Black Art if JAVA Game Programming*, Waite Group Press

Faulkner X. (2000), *Usability Engineering*, MacMillan Press

Fay T.M. with Selfon S. and Fay T.J. (2004), *DirectX Audio Exposed – Interactive Audio Development*, Wordware Publishing Inc.

Fernando R. and Kilgard M.J. (2003), *The Cg Tutorial – The Definitive Guide to Programming Real-Time Graphics*, Addison-Wesley

Fischetti M. (2000), The future of digital entertainment, Scientific American, november 2000, pp. 31-33

Fluckiger R. (1995), *Understanding networked multimedia – applications and technology*, Prentice Hall, 1995

Fogg B.J. (2003), *Persuasive Technology – Using Computers to Change What We Think and Do*, Morgan Kaufmann Publishers

Foley J.D. and van Dam A. (1982), *Fundamentals of Interactive Computer Graphics*, Addison-Wesley

Forman P. and Saint John R.W. (2000), Digital Convergence, Scientific American, november 2000, pp. 34-40

Fuhr, N., Gövert, N., Rölleke, Th. (1998), DOLORES: A System for Logic-Based Retrieval of Multimedia Objects, In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 257-265. ACM, New York

Gaver W. (2002), Designing for Homo Ludens, i3 Magazine

Gee J.P. (2003), *What video games have to teach us about learning and literacy*, Palgrave Macmillan

Gibson W. (1986), *Neuromancer*,

Gonzales R.C. and Wintz P. (1987), *Digital Image Processing*, Addison-Wesley, 2nd edn.

Grau O. (2003), *Virtual Art – From Illusion to Immersion*, The MIT Press

Grosky W., Jain R., Mehrotra R., eds. (1997), *The Handbook of Multimedia Information Management*, Prentice Hall, 1997

Hardman L., Bulterman D., van Rossum G. (1994), The Amsterdam Hypermedia Model: Adding Time and Context to the Dexter Model, CACM 37(2), pp. 50-62, febr 1994

Harel D. (1984), Dynamic Logic, In: Handbook of Philosophical Logic, Vol. II, D. Reidel Publishing Company, 1984, pp. 497-604

Hartman J. and Wernecke J. (1996), *The VRML 2.0 Handbook – Building Moving Worlds on the Web*, Addison-Wesley

Hawkins B. (2005), *Real-time cinematography for games*, Charles River Media

Hewlett W.B. and Selfridge-Field E., eds. (1998) , *Melodic Similarity – Concepts, Procedures and Applications*, MIT Press

Hildebrand M., Eliens A., Huang Z. and Visser C. (2003), Interactive Agents Learning their Environment, Proc. Intelligent Virtual Agents 2003, Irsee, September 15-17, 2003 J.G. Carbonell and J.Siekmann (eds.), LNAI 2792, Springer, pp. 13-17

Hix D. and Hartson R. (1993), *Developing User Interfaces*, Wiley

Hoggar S.G. (1992), *Mathematics for Computer Graphics*, Cambridge University Press

Hohl H. (1971), *Giacometti*, Abrams Publishers

Hoorn J., Eliens A., Huang Z., van Vugt H.C., Konijn E.A., Visser C.T., Agents with character: Evaluation of empathic agents in digital dossiers, Emphatic Agents, AAMAS 2004 New York 19 July - 23 July, 2004

Huang Z., Eliens A., van Ballegooij A., De Bra P. (2000), A Taxonomy of Web Agents, IEEE Proceedings of the First International Workshop on Web Agent Systems and Applications (WASA 2́000), 2000.

Huang Z., Eliens A., and De Bra P. (2001), An Architecture for Web Agents, Proceedings of the Conference EUROMEDIA 2001, 2001.

Huang Z., Eliens A., Visser C. (2001), Programmability of Intelligent Agent Avatars, Proceedings of the Agent0́1 Workshop on Embodied Agents, June 2001, Montreal, Canada

Huang Z., Eliens A., Visser C. (2002), 3D Agent-based Virtual Communities, In: Proc. Int. Web3D Symposium, Wagner W. and Beitler M.(eds), ACM Press, pp. 137-144

Huang Z., Eliens A., Visser C. (2005), *Intelligent Multimedia Technology: An Approach to Combine Agent Technologies with Multimedia*, in preparation

Huang, Z., Eliens, A., and Visser, C., *STEP: a Scripting Language for Embodied Agents*, in: Helmut Prendinger and Mitsuru Ishizuka (eds.), Life-like Characters, Tools, Affective Functions and Applications, Springer-Verlag, 2003.

Huang, Z., Eliens, A., and Visser, C. (2003d), "Is it within my reach?" – an agents perspective , Proc. Intelligent Virtual Agents 2003, Irsee, September 15-17, 2003, J.G. Carbonell and J.Siekmann (eds.), LNAI 2792, Springer, pp. 150-158

Huang Z., Eliens A., and Visser C. (2004), Facial Animation in STEP, AAMAS 2004 New York 19 July - 23 July, 2004

Huang, Z., Eliens, A., and Visser, C. (2003c), *STEP: a Scripting Language for Embodied Agents*, in: Helmut Prendinger and Mitsuru Ishizuka (eds.), Life-like Characters, Tools, Affective Functions and Applications, Springer-Verlag, (to appear).

Huang Z., Eliens A., Visser C. (2003b), XSTEP: A Markup Language for Embodied Agents, Proc. CASA03, The 16th Int. Conf. on Computer Animation and Social Agents

Huang Z., Eliens A., Visser C. (2003a), Implementation of a scripting language for VRML/X3D-based embodied agents, Proc. Web3D 2003 Symposium, Saint Malo France, S. Spencer (ed.) ACM Press, pp. 91-100

Huang Z., Eliens A., Visser C. (2003a), Implementation of a scripting language for VRML/X3D-based embodied agents, Proc. Web3D 2003 Symposium, Saint Malo France, S. Spencer (ed.) ACM Press, pp. 91-100

Huang Z., Eliens A., Visser C. (2002b), STEP – a scripting language for Embodied Agents, PRICAI-02 Workshop – Lifelike Animated Agents: Tools, Affective Functions, and Applications, Tokyo, 19/8/2002

Hughes B. (2000), *Dust or Magic – Secrets of Successful Multimedia Design*, Addison-Wesley, 2000

Hummelen IJ. and Sillé D. (1999), *Modern Art: Who cares?*, Foundation for the Conservation of Modern Art and the Netherlands Institute for Cultural Heritage

Huron D. (1997) , Humdrum and Kern: Selective Feature Encoding, In [_Beyond], pp. 375-400

Jain R. (2000), Digital Experience, [_1000], pp. 38-40

Javasoft (2001), Java3D API, http://java.sun.com/products/java-media/3D

Johnson A., Moher T., Cho Y-J., Lin Y.J., Haas D., and Kim J. (2002), Augmenting Elementary School Education with VR, IEEE Computer Graphics and Applications, March/April

Johnson B., Skibo C., Young M. (2003), *Inside Microsoft Visual Studio .NET*, Microsoft Press

Juul J. (2005), *Half Real – Video Games between Real Rules and Fictional Worlds*, MIT Press

Kalawsky R.S. (1993), *The Science of Virtual Reality and Virtual Reality Environments*, Addison-Wesley

Kay M. (2001), *XSLT Programmerś Reference*, Wrox Press

Kersten M. L.,Nes M., Windhouwer M.A. (1998), A feature database for multimedia objects, CWI Report INS-R9807, July 1998

Kientzle T. (1998), *A programmer Guide to Sound*, Addison-Wesley

Kimmel R. (2004), *Numerical Geometry of Images – Theory, Algorithms, and Applications*, *Springer*, Koegel Buford J.F., ed. (1994), *Multimedia Systems*, Addison-Wesley, 1994

Koenen R. (1999), MPEG-4 - Multimedia for our time, IEEE Spectrum, Vol. 36, No. 2,February 1999, pp. 26-33.

Koenen R. (ed.) (2000), Coding of moving pictures and audio, ISO/ITEC JTC1/SC29/WG11 N3747

Kress G. and van Leeuwen T. (1996), *Reading Images: The Grammar of Visual Design*, Routledge

Krikke J. (2003), Samurai Romanesque, and the battle for mobile cyberspace, IEEE Computer, Jan/Febr 2003

Kuipers J.B. (1999), *Quaternions and Rotation Sequences – A Primer with Applications to Orbits, Aerospace, and Virtual Reality*, Princeton University Press

Labrou L., Finin T. and Peng Y. (1999), The current landscape of Agent Communication Languages, IEEE Intelligent Systems, Vol. 14, no. 2, 1999

LaMothe A. (2002), *Tricks of the Windows Game Programming Gurus*, *Sams, 2nd edn.*,

Lengyel E. (2004), *Mathematics for 3D Game Programming & Computer Graphics* , Charles River Media

Li Z-N. and Drew M.S. (2004), *Fundamentals of Multimedia*, Prentice Hall, Pearson

Luna F. (2003), *Introduction to 3D Game Programming with DirectX 9.0*, Wordware Publishing Inc.

Manco T. (2004), *Street logos*, Thames & Hudson

McCool M. and Du Toit S. (2004), *Metaprogramming GPUs with Sh*, A K Peters, Ltd.

McCuskey M. (2002), *Special Effects Game Programming with DirectX*, Premier Press

McKnight C., Dillon A. and Richardson J. (1991), *Hypertext in Context*, Cambridge University Press

McNab R.J., Smith L.A., Bainbridge D. and Witten I.M. (1997), The New Zealand Digital Library MELody inDEX, D-Lib Magazine, May 1997

McQuail D. and Siune K. (1989), *Media Policy – Convergence, Concentration & Commerce*, Euromedia Research Group, Sage Publications

Messick P. (1998), *Maximum MIDI – Music Applications in C++*, Manning

Moortgat, M. (1996), Categorial Type Logics, Handbook of Logic and Language, J. van Benthem and A. ter Meulen (eds.), Elsevier Amsterdam (1996), pp. 93-177

Moortgat, M. (2002), Categorial Grammar and Formal Semantics, Encyclopedia of Cognitive Science, Nature Publishing Group, Macmillan Publishers Ltd.

Mongeau M. and Sankoff D. (1990), Comparison of musical sequences, Computers and the Humanities 24, pp. 161-175, 1990

Morrison M. (2005), *Beginning Mobile Phone Game Programming*, Sams Publishing

Mozart W.A. (1781), Twinkle,twinkle little star (Oh! vous dirai-je, mamam), K300, 1781-82

Munro A., Höök, K. and Benyon D.R. (1999), Footprints in the Snow, In: Social Navigation of Information Space, Springer

Murray J. (1997), *Hamlet on the Holodeck – The future of narrative in Cyberspace* , Free Press

Negroponte N. (1995), *Being Digital*, New Riders

Nelson T. (1980), *Literary Machines*, Mindfull Press

Nielsen J. and Mack R.L., eds. (1994) , *Usability inspection methods*, Wiley

Norman D.A. (2004), *Emotional Design*, Basic Books

Pachet F., Roy P. and Cazaly D. (1999), A combinatorial approach to content-based music selection, Proc. IEEE Int. Conf. on Multimedia Systems, Firenze

Ossenbruggen J. van and Eliens A. (1994), Music in Time-based Hypermedia, Proc. European Conference on Hypermedia Technology 1994, pp. 224-227

Ossenbruggen J. van (2001), *Processing Structured Hypermedia – A matter of style*, Ph.D. Thesis, Free University, 2001

Ossenbruggen J. van, Geurts J., Cornelissen F., Rutledge L., Hardman L. (2001), Towards Second and Third Generation Web-Based Multimedia, In Proc. of The Tenth International World Wide Web Conference pp. 479-488, May 1-5, 2001, Hong Kong

Patin S. (2006), *Regards sur les Nympheas (in french)*, Editions de la reunion des musees nationaux.

Pesce M. (2003), *Programming Microsoft DirectShow for digital video and television*, Microsoft Press

Picard R.W. (1998), *Affective Computing*, MIT Press

Preece J., Rogers Y., Sharp H., Benyon D., Holland S., Carey T. (1994), *Human-Computer Interaction*, Addison-Wesley

Purvis A.W. and Le Coutre M.F. (2003), *Graphic Design 20th Century*, BIS Publishers

Ratner P. (2003), *3-D Human Modeling and Animation*, Wiley, 2nd edn.

Riel C. van, Eliens A., Wang Y (2006), Exploration and guidance in media-rich information spaces: the implementation and realization of guided tours in digital dossiers, accepted for: International Conference on Multidisciplinay Information Sciences and Technologies (InSciT2006) October, 25-28th 2006, Merida, Spain www.instac.es/inscit2006

Riel C. van, Wang Y. and Eliens A. (2006b) , Concept map as visual interface in 3D Digital Dossiers: implementation and realization of nthe Music Dossier, CMC2006, Costa Rica, Sept 5-8 2006

Rooijen P. van, ed. (2003), *Signs & Symbols*, Pepin Press

Rosenblum L. and Macedonia M., eds. (2002), Projects in VR – From Urban Terrain Models to Visible Cities, IEEE Computer Graphics and Applications, 22:4, pp. 10-15

Rosenblum L. and Macedonia M., eds. (2005), Projects in VR – Tangible Photorealistic Virtual Museum, IEEE Computer Graphics and Applications, 25:1, pp. 15-17

Rutledge L., van Ballegooij A., Eliens A. (2000), Virtual Context - relating paintings to their subject, Culture Track of WWW9 in Amsterdam, The Netherlands, May 16th, 2000

Ruttkay M., Huang Z. and Eliens A. (2003), Reusable gestures for interactive web agents, Proc. Intelligent Virtual Agents 2003, Irsee, September 15-17, 2003 J.G. Carbonell and J.Siekmann (eds.), LNAI 2792, Springer, pp. 80-87

Ruttkay Z., Huang Z. and Eliens A., *The Conductor: Gestures for Embodied Agents with Logic Programming*, in: Recent Advances in Constraints K.R. Apt, F. Fages, F. Rossi, P. Szeredi and J. Vancza (eds.) LNAI 3010, Springer 2004

Sanchez-Crespo Dalmau D. (2004), *Core Techniques and Algorithms in Game Programming* , New Riders

Santiago D. (2005), *Creating 3D Effects for Film, TV and Games*, Thomson

Santos Lobao A. and Hatton E. (2003), *.NET Game Programming with DirectX 9.0*, The Authors Press

Schmidt, A.R. Windhouwer M.A., Kersten M.L. (1999), Indexing real-world data using semi-structured documents, CWI Report INS-R9902

Schneider P.J. and Eberly D.H. (2003), *Geometric Tools for Computer Graphics*, Morgan Kaufmann Publishers

Schonhage S.P.C., van Ballegooij A. and Eliens A. (2000), 3D Gadgets for Business Process Visualization: a case study, Proc. Int. Web3D/VRML Conference - 2000, Monterey CA, Febr 2000, ACM Press pp. 131-138

Schuytema P. (2007), *Game Design: A Practical Approach*, Charles River Media

Selfridge-Field E., ed. (1997), *Beyond MIDI – The Handbook of Musical Codes* , MIT Press 1997

Selfridge-Field E. (1998), Conceptual and Representational Issues in Melodic Comparison, In [_Similarity], pp. 3-64

Sherman W.R. and Craig A.B. (2003), *Understanding Virtual Reality – Interface, Application and Design*, Morgan Kaufmann Publishers

Sherrod A. (2006), *Ultimate Game Programming with DirectX*, Charles River Media

Shneidermann B. (1997), *Designing the user interface – strategies for effective human-computer interaction*, Addison-Wesley (3rd edn)

Shneiderman B (2003), *Leonardo's Laptop*, Mit Press

Singhal S. and Zyda M. (1999), *Networked Virtual Environments*, Addison-Wesley, 1999

Snook G. (2003), *Real-time 3D terrain engines using C++ and DirectX 9*, Charles Riven Media

Spalter A.M. (1999), *The Computer in the Visual Arts*, Addison-Wesley

Stephenson N. (1992), *Snowcrash*,

St-Laurent S. (2004), *Shaders for Game Programmers and Artists*, Thomson

Subrahmanian V.S. (1998), *Principles of Multimedia Databases*, Morgan Kaufmann

Sullivan A. (2005), 3-Deep, IEEE Spectrum 42-4, April 2005, pp. 22-27

Tarau P. (1999), Jinni: Intelligent Mobile Agent Programming at the Intersection of Java and Prolog, Proc. of PAAM9́9, London, UK, April, see also http://www.binnetcorp.com/Jinni

Temperley D. and Sleator D. (1999) , Modeling Meter and Harmony: A Preference-Rule Approach, Computer Music Journal 23(1), 1999, pp. 10-27

Trogemann G. & Viehoff J. (2004), *CodeArt – Eine elementare Einfhrung in die Programmierung als kunstlerische Praktik*, Springer

Vasudev B. and Li W. (1997), Memory management: Codecs, In [_Handbook], pp. 237-278

Vaughan T. (1998), *Multimedia – Making It Work*, Osborne/McGraw-Hill, 4th edn

Vince J. (1998), *Essential Virtual Reality*, Springer

Visser C. and Eliens A. (2000), A High-Level Symbolic Language for Distributed Web Programming, Internet Computing 2000, June 26-29, Las Vegas

Vyas, D., van der Veer, G. C. (2006), Experience as Meaning: Some Underlying Concepts and Implications for Design. , Proceedings of 13th European Conference on Cognitive Ergonomics. ACM Press: NY, (2006), 81-91.

Vyas, D., Chisalita, C., van der Veer, G. C. (2006), Affordance in Interaction. , Proc. of 13th European Conference on Cognitive Ergonomics. ACM Press: NY, (2006), 92-99.

W3C SMIL Working Group (2001), Synchronized Multimedia Integration Language (SMIL 2.0), W3C Recommendation 07 August 2001

Wagner W. and Beitler M. (eds), 2002, Proc. Int. Web3D Symposium, ACM Press

Walsh P. (2003), *Advanced 3D Programming with DirectX 9.0*, Wordware Publishing Inc.

Wang Y., Eliens A., van Riel C. (2006), Content-oriented presentation and personalized interface of cultural heritage in digital dossiers, accepted for: International Conference on Multidisciplinay Information Sciences and Technologies (InSciT2006) October, 25-28th 2006, Merida, Spain www.instac.es/inscit2006

Watt A. and Policarpo F. (2001), *3D Games – Real-time Rendering and Software Technology*, Addison-Wesley

Web3D Consortium (2001), VRML97 Standard, http://www.web3d.org/fs_specifications.htm

Weishar P. (1998), *Digital Space – Designing Virtual Environments*, McGraw-Hill

Welie M. and and Eliens A. (1996), Chatting on the Web, Proc. ERCIM W4G Workshop on CSCW and the Web, 7-9 febr 1996 GMD St Augustin, Germany, 1996

Welie M. avn, G.C. van der Veer G.C., and Eliens A. (1998) , Euterpe - Tool support for analyzing cooperative environments, Proceedings of the Ninth European Conference on Cognitive Ergonomics , August 24-26, 1998, University of Limerick, Ireland, pp. 25-30

Wiedermann J., ed. (2002), *Digital Beauties*, Taschen Verlag

Wiedermann J., ed. (2004), *Animation now*, Taschen Verlag

Wilson S. (2002), *Information Arts – Intersections of Art, Science and Technology*, The MIT Press

Yellowlees D. and Hargadon A. (2000), The pleasure principle: immersion, engagement, flow, Proc. of the 11th ACM Conf. on Hypertext and Hypermedia, May 2000,San Antonio, pp. 153-160

Zielinski S. (2006), *Deep Time of the Media – Towards an archaeology of Hearing and Seeing by Technical Means*, The MIT Press

Zimmermann D., Modeling Musical Structures, Aims, Limitations and the Artistś Involvement, Proc. Constraints techniques for artistic applications, Workshop at ECAI'98 25th August, 1998, Brighton, UK, ftp://ftp.csl.sony.fr/pub/pachet/workshopEcai

index

topical media & game development

This book provides a concise and comprehensive introduction to multimedia. It arose out of the need for material with a strong academic compontent, that is (simply) material related to scientific research. Indeed, studying multimedia is not (only) fun. Compare it with obtaining a driver license. Before you are allowed to drive on the higway, you have to take a theory exam. So why not take such an exam before entering the multimedia circus. Don't complain, and take the exam. After all it makes you aware of the rules governing the (broadband) digital highway. The book and accompanying material is available at http://www.cs.vu.nl/ eliens/media