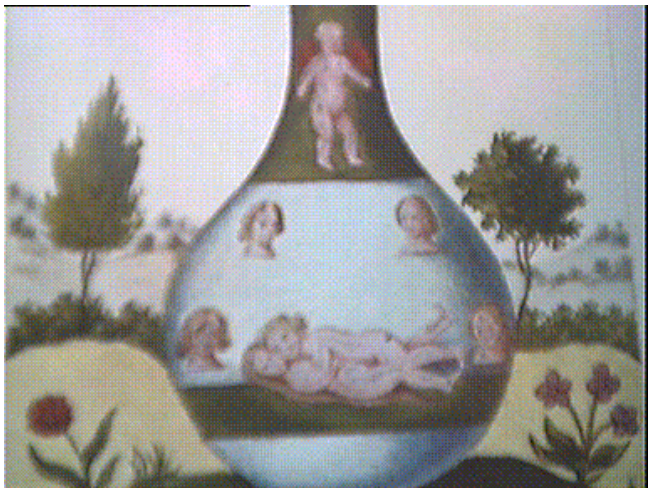
Æliens



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 (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 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. These notes 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?*

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.

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.

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 the relevance of multimedia.

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

I strongly believe that practical work is necessary, also for academics, to get a good grasp on multimedia. 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, for Macromedia Director or Flash. The online material contains a introductory manual for Director, so that you can start right away.

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 the theme underlying this book, that is the complementarity of authoring and retrieval:

multimedia

> *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 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 D. Nevertheless, the bulk of the (regular) material is relevent also for readers with a rather different opinion on what consitutes the *essence of multimedia*.

CDROM

The CDROM contains the full online version of the course notes. Open the file `index.html` in Netscape Navigator or Microsoft Internet Explorer, and click on *readme* for an explanation or *introduction multimedia* to access the material.

The online version provides you with both an HTML-based presentation format, as well as a VRML-based format, for presenting the lectures in class. The *blaxxun* Contact 3D VRML browser you need for this may be freely obtained from www.blaxxun.com.

The course notes are also available at http://www.cs.vu.nl/ eliens/media

how to use this book

The intended audience for this book is

- 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 Division of Mathematics and Computer Science of the Faculty of Sciences of the Free University of 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 presentation, including

- presentations for all chapters, including the preface in dynamic HTML and VRML slides
- a manual for Macromedia Director, also available in presentation format
- 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.

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 the first rough drafts of this material.

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 **??**. 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, 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 from ICN (Netherlands Institute for Cultural Heritage) for her contributions to the *multimedia casus*.

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) and Hughes (2000). 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 Eliëns (2000).
4. Subrahmanian (1998), Baeza-Yates and Ribeiro-Neto (1999).

5. Subrahmanian (1998), McNab et al. (1997), Kersten et al. (1998).
6. Subrahmanian (1998), Fluckiger (1995).
7. Fluckiger (1995), van Ballegooij and Eliëns (2001), Huang et al. (2002).

Only the material in sections 6.1, 6.3 and 7 reflects my own research efforts. The other material has all been diligently collected from (among others) the sources mentioned.

contents

1

digital convergence

Life is becoming digital for some time now, 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. 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 delivery of multimedia information, and we explore what impact the digital revolution may have from a commercial perspective.

1.1 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, it was observed that:

- digitizing everything audio and video will disrupt the entertainment industry's social order.
- the whole concept of holding a CD or movie in your hand will disappear once d-entertainment is widely available.

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

democracy

> *Creation of content will be democratized. Low cost digital movie cameras and PC video editors allow anyone with an eye to record and edit a movie for just a few thousend euro ...*

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 Jam makes the following observations:

> *The desire to share experiences will be the motivating factor in the development of exciting multimedia technology in the foreseeable future.*
>
> *Ramesh Jain, Digital Experience, CACM 44.3, pp.38-40*

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:

McLuhan (1976) – the medium is the message

> *..., the medium was the message when only one medium could be used to communicate messages.*
>
> *Now, the Internet allows the synthesis and rendering of information and experiences using whatever is the most appropriate media to convey the message*
>
> *(In other words) The message is just the message, and the medium is just the medium.*

Speculating on the future of multimedia communication and presentation technology, he states that:

presentation technology

- compelling experiences rely on carefully staged presentation
- in the coming years we'll see tremendous progress in presentation technology related to all our senses.
- enriched with (other) sensory information, virtual reality might approximate real reality ...

Clearly, from a technological perspective there seems to be no limit, except those imposed by our own phantasy.

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.*
>
> 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?

1.2 convergence

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:

history

- 1939 – New York World Fair – formal debut of television broadcast

They observe that

> *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 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 remark 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!

television

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.

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.

observations

- interactive television (1970) – people did not want to communicate back to the broadcaster
- enhanced television –

  - Disney – Who wants to be a millionaire?
  - Big Brother – ...

For example, 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.

Nevertheless, 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

As concerns digital television, we may come up with some immediate advantages:

digital television

- enhanced resolution
- multiplication of channels
- interactive television

Currently, there are some (competing) standards in development, that will enable the mass-scale adoption of digital television, notably:

standards

- US – 8-VSB (vestigial side-band) – not for antennas
- EU – COFOM (coded orthogonal frequency) – antennas, cable, satelite

When speaking about (digital) television, we must make a further distinction between:

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

In addition, we may mention the introduction of set-top boxes, such as

set top boxes

- BlueSky (UK), PrimeCon (D) – HTML, XML (X3D)

that, making use of what we may regard as standard web technology enable us to access the web through television.

As further discussed in chapter 3, we have (standard) codecs for d-TV, in particular

(standard) *codecs for d-TV*

- MPEG-2 – from Motion Picture Expert Group
- MPEG-4 – high quality streaming d-video on Internet

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.

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

The Entertainment TV

> *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:

problems

- thin clients (Sun/Java) vs fat clients (MS/Intel:Dell,Compaq)
- Internet (IP) is not robust – QoS
- proprietary architectures and codecs – RealVideo, QuickTime, Windows media

Do we opt for thin clients or fat clients? Will we be able to develop a more robust version of the Internet, that includes so-called *Quality of Service*, which gives you guaranteed bandwidth and delivery? And, will we be able to unify proprietary architectures and codecs into a common 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

ubiquitous computing

- smart houses,
- smart clothes, and even
- 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.

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 commercial impact

There is a large number of gadgets that could be classified as multimedia gadgets, a few of which are listed below, taken from the offerings of eluxury.com.

eluxury.com

> *Choose from today's state-of-the art entertainment, communication, and navigation products from industry leaders including Olympus, Motorola, and Panasonic:*
>
> *digisette*
> *pencam*
> *wristpc*
> *micropda*

More seriously, we are faced with the question what commercial impact multimedia, and in particular digital convergence, may have. Let's look at some news from business and comments from the popular press.

TV meets the Web

Monique van Dusseldorp & Partners operate on the European multimedia market as consultants. They promote, amongst others, the integration of TV and the Web.

http://www.tvmeetstheweb.com

Their mission:

> *helping clients in Europe position themselves for media convergence*

Their interests encompass the following types of content:

content

> *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:

European Broadband

> *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.

cable or (X)DSL?

- 1999 – 180 million conventional telephone lines

In contrast, there are only 15 million cable subscribers, giving DSL a large potential audience.

Matthijs Leendertse, co-author of the report, observes:

broadband landscape

> *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 of writing the battle is still going on!

new media

As may be read in all newspapers (in 2001), large investments are being made (by both cable and telephone companies) to improve the technological infrastructure for the new media. Not to forget, the companies operating on the mobile telephone market. Simultaneously, joint ventures arise 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 *TIVO* 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:

acceptance problems

- experiments (failed): videofoon, videotext, cd-i, DCC
- Canal+: information overload

Perhaps the reason for these failures is the *trial-and-error* (aka spaghetti) method that is being followed in developing new media.

Jan van Dijk (UTwente) The Network Society

- spaghetti method – plate against the wall, and see which will stick

Just throw it on the market and see what sticks. 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.

observations

- download & upload – (Sweden: upload!)
- video-on-demand – see webnoize.com

For example in Sweden, the upload of material far exceeded download, which is contrary to the assumptions underlying ADSL.

mobile multimedia

To conclude this chapter, let's look at another potential hype. In 2000, Webnoize published a report (by Matt Bailey), entitled *Wireless Entertainment: What Is It Worth?*:

wireless entertainment

> *The wireless web is being unleashed. Cellular providers around the globe are spending billions of dollars to bring faster connectivity to cell phones, personal digital assistants (PDAs) and other mobile devices. As new networks roll out, wirelessly streamed music will be a huge hit with commuters, while young media junkies will demand music videos and short animations. Tens of millions of consumers around the world will use wireless devices to gain ubiquitous access to content.*

The intent of the report is to investigate whether investments in the mobile entertainment are justified. Quoting again:

wireless or worthless?

> *Webnoize 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 are 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.

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.

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*.

2

information (hyper) spaces

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.

2.1 information spaces

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, 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.

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 the following

> definition: *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 definition 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 7.

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

> definition: *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).

(multimedia) information object

- or example, is a point in the (logical) information space

query

- is an arbitrary region in this information space

clue

- is 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

optimal query

> *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.

physical layer

- networked multimedia – client/server (6.3)

logical layer

- information hyper space – chunks and hyperlinks (2.2)

presentation layer

- virtual reality interface – physical location of student records (7.2)

Each of these improvements or extensions can be regarded as a technological or scientific adventure in it's own right. (See the sections indicated inbetween the brackets.)

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:

http://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

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 C we will explore how XML might be useful for your own multimedia application by treating some simple examples.

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)

Bush' (not the presidents') 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 *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 Bush (not the president) 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 you may ask. Just think about how you are taught to deal with information and how you actually go about with it. Speaking about Ted Nelson, quoting Hughes (2000):

> ... 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 Conklin (1987).

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 a brief characterization is given below.

Intermedia

- web = documents + links + maps

retrieval by attributes

- block – *to apply filters*
- link – *conditional traversal*

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.

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.

Dexter Hypertext Reference Model

- components, links and anchors

The Dexter model offers an abstract description of *hypertext*. It made a distinction between components, anchors within components and link 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*

compound

- children – *subcomponents*

The Dexter Hypertext Model has been criticised from the beginning. Among others, because *compound documents* 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.

authoring

- structure – *sequential and parallel composition*
- channels – *presentation*

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.

compound

- children – *(component, start-time)*
- synchronization – *(source, destination)*

Each component obtains a start-time, that may result from parallel or sequential composition and synchronisation arcs. Yet 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).

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 3.3 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.

visualization

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

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.

eliminating complexity

So now, in the new millenium, we are (sadder and wiser) in a position to approach the effective deployment of mutimedia 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 mutimedia 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. But read on, there is more

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 again

> *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 campain 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*. You may want to look at the site given below

> http://www.captology.org – Computers As Persuasive Technology

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:

- *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.

> *This is not like TV, only better – says Lenny Nero in Strange Days*

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:

> *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:

> *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.*

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.

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 7.3 and appendix D, we will present a framework that may be used as a testbed for developing programmed narrative structures with embodied agents as the main characters.

questions

information (hyper) 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*.

3

codecs and standards

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.

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 → source coder → channel coder (encoding)

signal ← source decoder ← 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).*

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 coding efficiency, the complexity of the coder and decoder, and the signal quality.

In practice this means that when we select a particular coder-decoder scheme we must consider whether we can guarantee

- resilience to transmission errors

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.

For particular applications, such as conferencing, we should be worried about

- the interplay of data modalities – in particular, audio & video.

And,with regard to the many existing codecs and the variety of platforms we may desire the possibility of

- transcoding to other formats – (interoperability),

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. Without going into details, we list a number of these standards below.

standard-based codecs

- JPEG – ISO/IEC 10918-1, ITU-T (T.81)
- MPEG
  - ISO 11172 (up to 1,5 Mbps) – MPEG-1
  - ISO 13818 ITU-T H.262 – MPEG-2
- H3.20 – for ISDN-like environments
- ITU-T H.261 – P x 64 standard (rate in kbs, p=1..30)
- H.324 – video conferencing for GSTN, 26kbps/sec

In the last decade of the previous millenium great progress has been made in finding efficient encodings for audio and video. I assume that most of you have heard of MP3 (the infamous 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 standards

- MPEG-1, MPEG-2, H3.20, H3.24

object-based codec(s)

- MPEG-4 – segmentation-based DFD (*Displaced Frame Difference*)

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.

MPEG-1

IBBPBBIBBPBBI...
IBBPBBPBBPBBI...

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

Finally, decoding takes place as outlined below.

decoding

- first I, then P, and finally B

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.

alternatives to MPEG-1

- MPEG-2 – higher pixel resolution and data rate

- MPEG-3 – to support HDTV
- MPEG-4 – object-based, ...
- MPEG-7 – content description

We will elaborate on MPEG-4 in the next section, and briefly discuss MPEG-7 at the end of this chapter.

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 7.3, and use the streaming video codec from Real Networks that is integrated as a rich media extension in the *blaxxun* Contact 3D VRML plugin.

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 four standards.

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.

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

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.

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.

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, we will further establish whatthe 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.

the press Now to conclude our discussion of MPEG-4, let's see what the press has to say about it.

http://www.eetimes.com/story/OEG20010220S0065

> *MPEG-4 is "a big standard," said Tim Schaaff, vice president of engineering for Apple Computer Inc.'s Interactive Media Group. "It's got tons of tools inside." Its success, he said, will depend on the industry's willingness to home in on a small subset, winnowing from a number of profiles and levels designed for streaming a slew of digital multimedia types – audio, several types of video, still images, and 2-D and 3-D graphics.*

Some may find it to ambitious.

unfocused ambition

> *"MPEG-4 is a very ambitious standard, but its biggest problem is that it wasn't focused on anything," said Didier LeGall, vice president for R&D and chief technology officer at chip house C-Cube Microsystems Inc. LeGall dismissed MPEG-4's vaunted object-based coding – one of the technologies that sets it apart from earlier MPEG spins – as "science fiction" and "nothing more than a gadget" at this point. "I haven't seen any content with objects that really makes sense," he said.*

But, then again, what it offers is clearly worthwhile.

> *MPEG-4's chief features include highly efficient compression, error resilience, bandwidth scalability ranging from 5 kbits to 20 Mbits/second, network and transport-protocol independence, content security and object-based interactivity, or the ability to pluck a lone image – say, the carrot Bugs Bunny is about to chomp – out of a video scene and move it around independently.*

And, not altogether unimportant, it may offer significant commercial benefits.

> *Broadband service providers, such as cable and DSL companies, are right behind wireless in sizing up MPEG-4, largely because its low bit rate could help them add channels in their broadband pipes while incorporating interactive features in the content. Possibilities include multiple video streams, clickable video, real-time 3-D animation and interactive advertising.*

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.
- 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!

RM3D

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.

http://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:

http://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!

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.

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.

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.

3.3 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 semantic web

which enhances the functionality of the current web by deploying knowledge representation and inference technology from Artificial Intelligence. 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? In Ossenbruggen et. al. (2001), we read:

multimedia

> *While text-based content on the Web is already rapidly approaching the third generation, multimedia content is still trying to catch up with second generation techniques.*

The reason for this is that processing multimedia is fundamentally different from processing text. As phrased in Ossenbruggen et. al. (2001):

processing requirements

> *Multimedia document processing has a number of fundamentally different requirements from text which make it more difficult to incorporate within the document processing chain.*

More specifically it is said that:

presentation abstractions

> *In particular, multimedia transformation uses different document and presentation abstractions, its formatting rules cannot be based on text-flow, it requires feedback from the formatting back-end and is hard to describe in the functional style of current style languages.*

Now this may well be true for specific categories of multimedia on the web. So, for example, rendering presentations written in SMIL is probably not an easy thing to do. But does this really prevent us from incorporating multimedia in the semantic web, or rather create a multimedia semantic web?

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 and RM3D content, such facilities are readily available. You may look at MPEG-7 to get an idea how this might be approached.

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.

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.

data model

> *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. Baeza-Yates and Ribeiro-Neto (1999).*

Now, how would you approach defining a unifying data model for the web? One project in this area that might be worthwhile to look at is the *OntoWeb* project, accessible through

http://www.ontoweb.org

that aims at producing the technology for ontology-based information exchange for both knowledge management and electronic commerce. Such technology allows for adding descriptive information and, equally important, to reason with such information. Moreover, it allows for dealing with information formulated in disparate terminologies by using so-called ontologies, which may be regarded as formalized perspectives or world views.

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, Eliëns (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 7.3, we will present a framework for doing such experiments.

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 Dormann and Eliëns (2002), 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.

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.

4

information retrieval

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.

4.1 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.

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 containg 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).

4.2 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.

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:

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

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.

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.

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 6, 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.

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 ...*

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 7 and appendix D for some initial ideas.

4.3 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?

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.

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.

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*.

5

content annotation

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.

5.1 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.

research directions – *musical similarity*

In this section on research directions for audio information retrieval, we will study how to provide content-based retrieval facilities based on *similarity* in the musical domain. This material comes from our previous research, part of which has been reported in Eliëns (2000). However, here we will look primarily at work that has been done in this field by others.

As concerns musical content, at least for most genres, it appears that we should focus primarily on *melody*, since, as phrased in Selfridge (1998):

> *"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 (1781). 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.

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 **??**. 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

$$\begin{array}{ll} d_{i0} = d_{i-1,0} + w(a_i, 0),\ i \geqslant 1 & \textit{deletion} \\ d_{0j} = d_{0,j-1} + w(0, b_i),\ j \geqslant 1 & \textit{insertion} \end{array}$$

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.

5.2 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.

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.

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.

Christel et al. (2000) 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.*

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!

5.3 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 $S \xrightarrow{*} ts$. 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.

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.

implementation The embedded logic component is part of the *hush* framework, Eliëns (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.

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*.

6

information system architecture

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.

6.1 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.*

6.2 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).)

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.

6.3 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

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.

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.

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 Eliëns (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:

http://java.sun.com/products/java-media

> *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 Eliëns (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 7.3 and described in more detail in appendix D, 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.

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*.

7

virtual environments

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!

7.1 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.

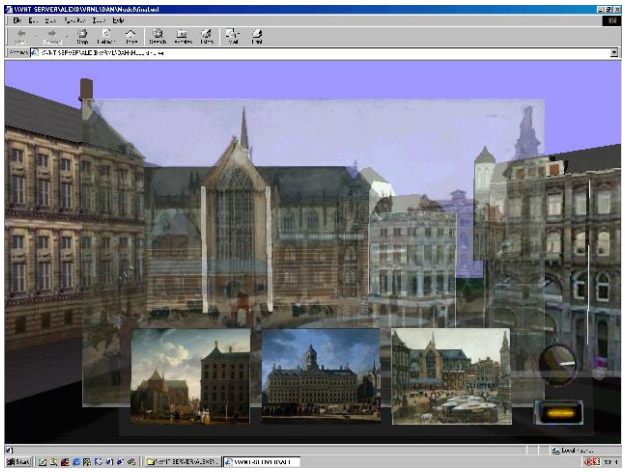
We can improve on the latter scenario I think. So let's explore the options. First of all, we may establish that the Dam square represents a rich information space. Well, the Dam Square is a 'real world' environment, with it has 700 years of (recorded) history. It has a fair amount of historical buildings, 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?

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.

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.

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 Eliëns (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.

7.2 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

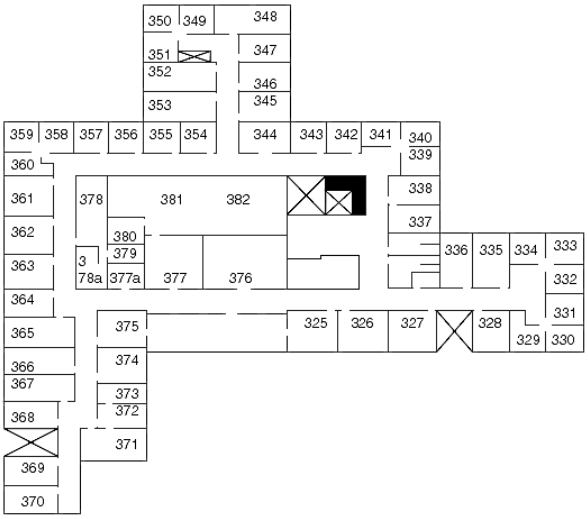
RIF

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.

case study – CWI

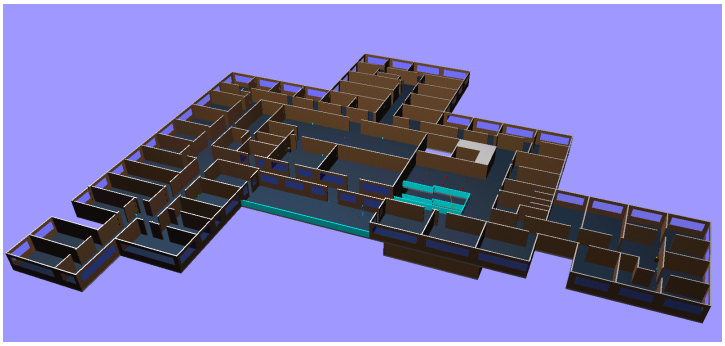
For our prototype, we took one of the worlds of our virtual environment, the third floor of the CWI. The reason for this is that we were (at the time) doing our research there, and so there were plenty locations of interest, such as the rooms of our collegues, the printer room, and not to forget, the coffee corner.

the map



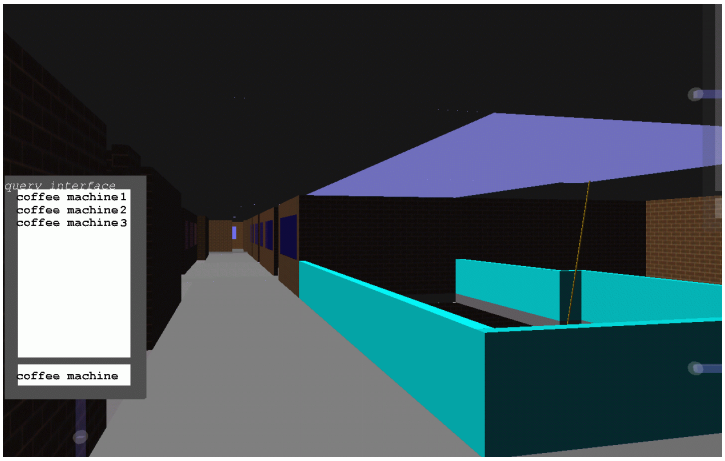
We started out by taking a map of the third floor, and developed a model of it, using a tool developed by a student, who needed such a tool for realizing his game *Out of the Dark*.

the model



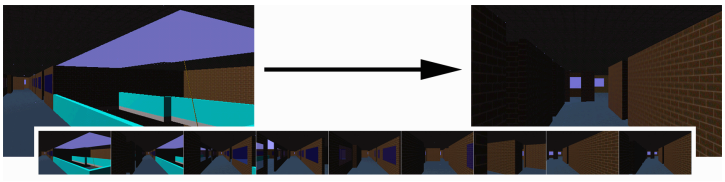
When dwelling around in (this part of) our virtual environment, the user may pose (arbitrary) queries, for example *where is the coffee machine*.

the query



Remind, that after a few hours of research, coffee might be needed to get fresh ideas!

navigation



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.

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.

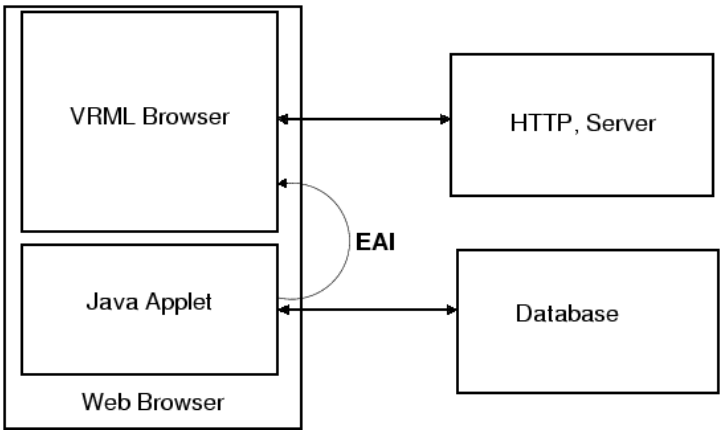
the prototype

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!

7.3 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.

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.

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.

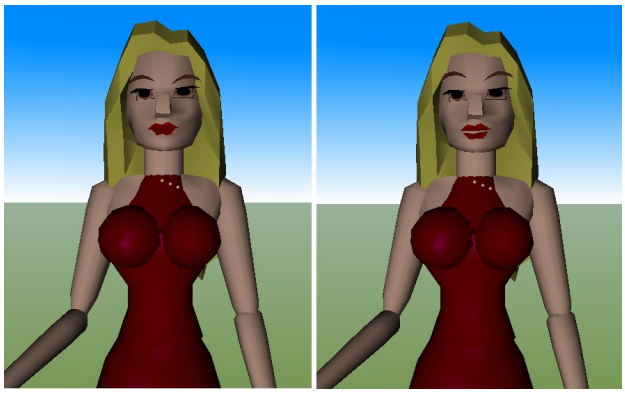
Our original demonstrator for the WASP project was an agent of the latter kind, with the nickname *PAMELA*, which is an acronym for:

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.

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.

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.

afterthoughts

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 section 2.3:

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.*

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.

the multimedia equation(s)

$$\text{multimedia} = \text{presentation} + \text{context}$$

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

- context = convergence + information + architecture

where

- 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, Eliëns (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 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 2.3:

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 and culture This book was written for the new *Multimedia and Culture* curriculum at the Vrije Universiteit, Amsterdam. In particular, the book contains the course notes for the first year course

- introduction multimedia

There are two follow-up courses:

- Multimedia Authoring I – Web3D/VRML
- Multimedia Authoring II – Virtual Environments

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 7 and appendix D). These courses

are also part of the master program *multimedia*, in which the focus is more on the technical aspects of multimedia.

In addition, *Multimedia and Culture* students are required to work on a *multimedia casus* to bring what they learned into practice, see appendix **??**.

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

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 D. And, admittedly, there are many aspects that are not covered, in particular those that are related to more advanced virtual reality technology.

Now, you may ask *what's Culture got to do with it*? I wouldn't know. My focus is on multimedia, that is the scientific issues and the engineering of multimedia information systems.

A. Eliëns

appendix

A

abbreviations and acronyms

ANMA Amsterdam New Media Association
http://www.anma.nl

AVID Agents and Virtual environments In DLP
http://www.cs.vu.nl/∼eliens/research/avid.html

Blendo Sony (RM3D) VRML extension
http://www.blendomedia.com

DL Digital Library
http://www.dlib.org

DLP Distributed Logic Programming
http://www.cs.vu.nl/∼eliens/dlp

ICN Netherlands Institute for Cultural Heritage
http://www.icn.nl

INCCA International Network for the preservation of Contemporary Art
http://www.incca.org

Informedia Digital Library Project
http://www.informedia.cs.cmu.edu

JPEG Joint Photographic Expert Group
http://www.jpeg.org

HyTime proposal for encoding hypermedia
http://www.cwi.nl/∼lloyd/HyTime

MPEG Motion Picture Expert Group
http://www.mpeg.org

MusicXML music interchange format
http://www.musicxml.org/xml.html

NDLTD Networked Digital Library of Theses and Dissertations
http://www.ndltd.org

PDF Portable Document Format
http://www.adobe.com

PERSONAS PERsonal and SOcial NAvigation through information space
http://www.sics.se/humle/projects/persona/web

RM3D Rich Media 3D
http://groups.yahoo.com/group/rm3d

RIF Retrieval of Information in Virtual Worlds using Feature Detectors
http://www.cs.vu.nl/∼eliens/research/rif.html

SGML Structured Generilized Markup Language
http://www.sgmlsource.com

SIGIR Special Interest Group Information Retrieval
http://sigir.org

SMDL Standard Music Description Language
http://xml.coverpages.org/gen-apps.html

SMIL Synchronized Multimedia Integration Language
http://www.w3.org/AudioVideo

SVG Scalable Vector Graphics
http://www.w3.org/Graphics/SVG/Overview.html

VRML Virtual Reality Modeling Language
http://www.web3d.org

WASP Web Agent Support Program
http://www.cs.vu.nl/∼eliens/research/wasp.html

WMF Wireless Multimedia Forum
http://www.wmmforum.com

W3C World Wide Web Consortium
http://www.w3.org

XML eXtensible Markup Language
http://www.xml.org

B

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.

http://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.[1]

protos

- *slideset* – container for slides
- *slide* – container for text and objects
- *slide* – container for lines of text
- *line* – container for text
- *break* – empty text

[1] 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 C 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.

C

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 Eliëns 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, Eliëns (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 Eliëns (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 7.3. 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.
```

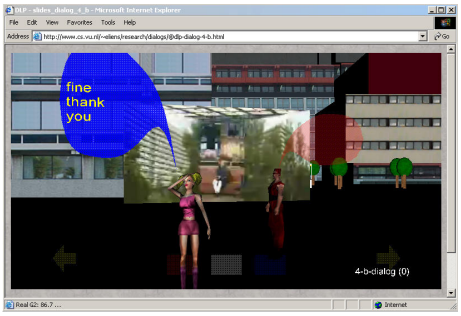
The constructor sets the non-logical variable *projector* and enters a repeat loop to accept any of the incoming events for respectively *id*, *level*, *update* and *touched*. Each event has a value, that is available as a parameter when the corresponding method is called on the acceptance of the event. 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).



(a) *dialog in context*

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.[1]

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.

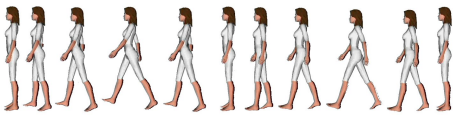
Each action is defined using the *script*, by specifying an action list containing the (possibly compound) actions of which that particular action consists. As an example, look at the definition of *walking* below.

```
script(walk(Agent), ActionList) :-
ActionList = [
  parallel([turn(Agent,r_shoulder,back_down2,fast),
         turn(Agent,r_hip,front_down2,fast),
         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),
```

[1] Clearly, our approach is reminiscent to the notorious *Agneta & Frida* characters developed in the Persona project. See the *research directions* of section 3.3.

```
            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.)



(b) *walking humanoid*

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.

E

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 appendix, 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 7.1 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.

project assignment

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

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 assests
- *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!

references

Baeza-Yates R. and Ribeiro-Neto B. (1999), *Modern Information Retrieval*, Addison-Wesley, 1999

Berners-Lee T., Hendler J., Lassila O. (2001), The semantic web, Scientific American, may 2001, pp. 28-37

Bolter J.D and Grusin R. (2000), *Remediation – Understanding New Media*, MIT Press

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

Bush V. (1945), As we may think, Atlantic Monthly, July 1945

Chang S.C. and Costabile M.F. (1997), Visual Interfaces to Multimedia Databases. In Grosky et al. (1997)

Christel, M., Olligschlaeger, A., Huang, C. (2000), Interactive maps for digital video, IEEE Multimedia 7(1), pp. 60-67

Conklin J. (1987), Hypertext: An Introduction and Survey, *IEEE Computer* 20(9), pp. 17-41

Davenport G.(2000), Your own virtual story world, Scientific American, november 2000, pp. 61-64

Davison A. (2001), Enhancing VRML97 Scripting, Euromedia'2001, Valencia, Spain, April 18-20. available from: http://fivedots.coe.psu.ac.th/∼ad

Dodge M. and Kitchin R. (2002), *Atlas of Cyberspace*, Addison-Wesley

Dormann C. and Eliëns A. (2002), Exploring the design space for emotive dialogues, submitted to: Third Int Conf on Emotion and Design

Eliëns A. (1988), Computational Art, Leonardo, MIT Press

Eliëns A. (1992), DLP – A language for Distributed Logic Programming, Wiley

Eliëns A. (2000), *Principles of Object-Oriented Software Development*, Addison-Wesley Longman, 2nd edn.

Eliëns A., Huang Z., and Visser C., A platform for Embodied Conversational Agents based on Distributed Logic Programming, AAMAS Workshop – Embodied conversational agents - let's specify and evaluate them!

Engelbart D. (1963), A Conceptual Framework for the Augmentation of Man's Intellect, Vistas in Information Handling 1(9)

Fluckiger R. (1995), *Understanding networked multimedia – applications and technology*, Prentice Hall, 1995

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

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

Hewlett W.B. and Selfridge-Field E., eds. (1998) , *Melodic Similarity – Concepts, Procedures and Applications*, MIT Press

Huang Z., Eliëns 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 '2000), 2000.

Huang Z., Eliëns 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

Hughes B. (2000), *Dust or Magic – Secrets of Successful Multimedia Design*, Addison-Wesley, 2000

Huron D. (1997) , Humdrum and Kern: Selective Feature Encoding. In Selfridge (1997), pp. 375-400

Jain R. (2000), Digital Experience, [_1000], pp. 38-40

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

Kay M. (2001), *XSLT Programmer's Reference*, Wrox Press

Kersten M. L.,Nes M., Windhouwer M.A. (1998), A feature database for multimedia objects, CWI Report INS-R9807, July 1998

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

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

Mongeau M. and Sankoff D. (1990), Comparison of musical sequences, Computers and the Humanities 24, pp. 161-175, 1990

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

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

Picard R.W. (1998), *Affective Computing*, MIT Press

Schmidt, A.R. Windhouwer M.A., Kersten M.L. (1999), Indexing real-world data using semi-structured documents, CWI Report INS-R9902

Schneidermann B. (1997), *Designing the user interface – strategies for effective human-computer interaction*, Addison-Wesley (3rd edn)

Selfridge-Field E. (1998), Conceptual and Representational Issues in Melodic Comparison. In Hewlett and Selfridge-Field (1998), pp. 3-64

Selfridge-Field E., ed. (1997), *Beyond MIDI – The Handbook of Musical Codes* , MIT Press 1997

Singhal S. and Zyda M. (1999), *Networked Virtual Environments*, Addison-Wesley, 1999

Subrahmanian V.S. (1998), *Principles of Multimedia Databases*, Morgan Kaufmann

Tarau P. (1999), Jinni: Intelligent Mobile Agent Programming at the Intersection of Java and Prolog, Proc. of PAAM'99, 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

Twinkle,twinkle little star (Oh! vous dirai-je, mamam), K300, 1781-82

van Ballegooij and Eliëns A. (2001), Navigation by Query in Virtual Worlds, Web3D 2001 Conference, Paderborn, Germany, 19-22 Feb 2001

Vasudev B. and Li W. (1997), Memory management: Codecs. In Grosky et al. (1997), pp. 237-278

Visser C. and Eliëns A. (2000), A High-Level Symbolic Language for Distributed Web Programming. Internet Computing 2000, June 26-29, Las Vegas

W3C SMIL Working Group (2001), Synchronized Multimedia Integration Language (SMIL 2.0), W3C Recommendation 07 August 2001

Zimmermann D., Modeling Musical Structures, Aims, Limitations and the Artist's 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

introduction *multimedia*

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