Interactive video in serious games

exploring the use of interactive video for Clima Futura

H.C. Huurdeman

hchuurde@few.vu.nl

November, 2007

Supervisor: dr. Anton Eliëns

Second reader: dr. Johan Hoorn (CAMeRA)

Prefinal version

Prefinal version notes:
-Title pages not finished (add cover)
-References need to be finalized
-Afterthoughts will be added

Abstract

The interest in serious games has increased rapidly in the past years. Interactive video in the context of (serious) gaming is largely neglected, however. New authoring tools provide fresh ways to achieve interactive video online. In this thesis we will demonstrate that interactive video is a viable technology for serious games. To this end, the developments in serious games and interactive video are analyzed. This is followed by a discussion of technology that enables interactive video. The pilot application for Clima Futura, a serious game about climate change, provides an example of how interactive video can be applied in the context of serious games. The narratives of interactive video will be discussed to provide some additional backgrounds. The final chapter is about the development and reengineering of the interactive video application.

Acknowledgements

I would like to express my sincere gratitude to:

Anton Eliens (for guidance and support)

Johan Hoorn (second evaluator)

Winoe Bhikharie (for helping to reengineer the Clima Futura interactive video application)

Wouter Vermeulen (co-creator of the Great Barrier Reef interactive video application)

the Clima Futura team (for support and great collaboration)

the Climate Centre and CAMeRA (for providing me the opportunity to work on Clima Futura)

and finally, my parents and family, for the omnipresent support

Table of contents

1. Introduction

In 1983, a sensation appeared in the arcade halls around the world: a game that did not use small sprites, like other games at the time. Instead, it featured fully animated graphics, created by a former Disney artist.

The game was called *Dragon's Lair*, one of the first games that used interactive video. The new LaserDisc technology facilitated random access to full-screen, full-motion video. The following years showed much interest in interactive video, but few applications. As the capabilities of computers increased in the early nineties, interactive video at home was possible and enjoyed some success.

But despite the name, interactive video was hardly interactive, resulting in declining interest. The term itself became synonymous with games that have an emphasis on cut scenes at the expense of real gameplay.

Right now, the new buzzword is 'serious games'. This is the use of games for educational purposes. On the other side, online video websites are thriving. YouTube is an immensely popular website, with a wealth of video material about all kinds of subjects. Still, a combination of both (interactive) video and serious games, is largely neglected.

This thesis should prove that interactive video is a viable technology for serious games. The developments in serious games and interactive video are discussed. The Flash-based prototype for the Clima Futura-project will also be described, including its technology and XML-annotation. In addition, based on a brief literature survey, a theoretical basis for the use of narratives in interactive video and games is made. The final chapter is about the development of the interactive video application, and the subsequent reengineering with the Flex technology.

1.1. Research Questions

The goal of this thesis is to explore the possibilities of interactive video in serious games, focusing mainly on Adobe Flash (and Flex) technology. Therefore the main research question is:

- *To what extent is it possible to use interactive video in Flash-based serious games?*

To find an answer to the main research question, it will be split in multiple sub questions.

Many definitions of interactive video exist, so it is necessary to define a suitable one for use in this thesis. This thesis is largely dedicated to Clima Futura, a serious game about climate change. As background information, the following questions are addressed:

- What is interactive video?
- What is a serious game?
- What is Clima Futura?

Adobe Flash is a suitable technology to use for interactive video, so an investigation of its features is needed:

- What are the possibilities of interactive video in Flash?

After investigating the features of Flash with respect to interactive video, the pilot application and its underlying XML-structure are discussed:

- How can interactive video be implemented in Flash?
- How can XML be used to define narratives for interactive video?

If interactive video is used for serious games, it could be valuable to broaden the scope, and investigate narratives and the rhetoric form in a game context:

- What are the possibilities for narratives and the rhetoric form in a game-context?

To integrate the Flash prototype with Clima Futura, it is necessary that it is reengineered with Adobe Flex 2:

- How can the implemented prototype in Flash be reengineered with Flex?

1.2. Thesis structure

Chapter 2: Terminology

The second chapter introduces the terminology that will be used in this thesis. It is mostly a theoretical approach, using reviewed literature.

Chapter 3: Background

In this chapter the backgrounds of interactive video, serious games and Clima Futura are discussed. Some Flash-based examples of interactive video are provided, to provide more insights into the subject.

Chapter 4: Interactive video in Flash

The fourth chapter is focused on the technology for achieving interactive video: Adobe Flash. The different possibilities are analyzed. In the second part of the chapter some technical examples will be provided.

Chapter 5: Interactive video in practice - Clima Futura pilot

This chapter contains the process of developing a prototype in Flash for an interactive video application; a pilot for the Clima Futura project.

Chapter 6: Interactive video annotation in XML

The narratives for interactive video can be specified in several ways. One way is via XML, which is used in the Clima Futura pilot. This chapter introduces XML and its contribution to the pilot application.

Chapter 7: Narratives in a game context

Important aspects of interactive video are the narrative and rhetoric form. This chapter provides a theoretical basis for the use of interactive narratives in games, mainly focused on interactive video.

Chapter 8: Development and reengineering of the Clima Futura pilot

This thesis ends with a chapter about both the development and the upcoming integration of interactive video in the Clima Futura game architecture, which will use the Flex 2 technology.

Chapter 9: Conclusion

The conclusion will wrap up the results of my research.

Reading guide

Different target audiences can be defined for this thesis:

- The general audience, interested in interactive video and serious games
- Students, interested in learning more about interactive video, Flash, Flex and XML
- Developers, who would like to develop their own interactive video application

Chapter 2: general audience
This is a basic guide to the terminology in this thesis. It is recommended to read this if you want to study this thesis thoroughly.

Chapter 3: general audience
The third chapter is meant for everyone remotely interested in interactive video, serious games and Clima Futura.

Chapter 4: students and developers
This is intended for students and developers interested in Flash, and the combination with interactive video.

Chapter 5: general audience
The first section of Chapter 5 is aimed at everyone that wants an overview of the interactive video application that is developed during the project. The second section of the chapter focuses on the video collection and editing process, and is somewhat more technical.

Chapter 6: students and developers
The first part of chapter 6 introduces XML. The second part shows the application the use of XML for the specification of interactive video narratives, and is more aimed at developers and people who want to define their own storylines.

Chapter 7: general audience
This chapter provides some more information about narratives and the rhetoric form.

Chapter 8: developers
The final chapter is meant for people interested in the development and integration of interactive video in the Clima Futura project, by means of reengineering it with Flex 2. This is more technical, so aimed at developers.

2. Terminology

In this chapter, the terminology is discussed, which serves as a (theoretic) basis for the remainder of this thesis. First, the technologies involved are discussed. The terminology regarding video and video editing follows. Finally, the basics of interactive video are described.

2.1. Technology

Adobe Flash
Adobe Flash (formerly Macromedia Flash) is an advanced authoring solution for creating interactive content for different platforms[1]. It is currently used for websites, advertisements, presentations and games. To view Adobe Flash content, the Adobe Flash Player plug-in is needed. It is installed on most Windows and Macintosh computers: according to a Millward Brown survey, it reaches 99.3% of all desktop Internet users.[2] The latest version of Adobe Flash is CS3.

Adobe Flex
Adobe Flex 2 is a cross-platform framework for developing rich internet applications[3]. While Flash is more suited to developing interactive content, Flex is preferable for the development of full applications. The Flex 2 Software Development Kit (SDK) is open-source, so you can develop and use Flex applications for free. The output of Flex projects is a Flash-compatible SWF-file, so applications developed with Flex 2 can be used on all computers with the Flash-plugin.

Flash Video
Flash video (FLV) is the video standard that is used in Adobe Flash 8 and higher. It has support for adding extra information (metadata) to the videos and alpha channels (transparent backgrounds).

ActionScript
ActionScript is the scripting language that is used by Flash and Flex, based on ECMAScript. Currently, most applications use ActionScript 2.0. The new Flash Professional CS3 and Flex support ActionScript 3.0. The performance is improved in this new version, and the programming model is more robust (making more complex web applications possible).

XML
XML is an acronym for *Extensible Markup Language*. It is an open standard for exchanging structured documents and data over the Internet. A growing number of applications make use of XML to exchange data. Adobe Flash and Flex have support for XML.

2.2. Video

Frame
A video consists of frames, still images, which are rapidly displayed after each other to create motion. A video frame is the atomic element of a video.

Scene
A scene is a sequential set of video frames. A scene in film is defined as “a segment in a narrative film that takes place in one time and space or that uses crosscutting to show two or more simultaneous actions.” [4]

[1] http://www.adobe.com/products/flash/
[2] http://www.adobe.com/products/player_census/flashplayer
[3] http://www.adobe.com/products/flex/
[4] http://www.personal.psu.edu/faculty/a/c/ach13/Asia/Glossary.htm#Film

Compression
This is a process that reduces data in a video frame (e.g. by taking into account similarities in an image). Higher compression usually results in lower video quality. The compression of a file can be set by altering the bitrate.

Keyframe
A complete (possibly compressed) image used as reference for subsequent images. Of the subsequent frames, only the differences with the keyframe are saved, thus needing less disk space. Many compression algorithms (like the Flash Video format) use keyframes.

Bitrate
The bitrate is the amount of data that is used for the storage of a video. Often the bitrate is separately indicated for the video and audio components of a movie, in kilobits per second (kbit).

Resolution
This is the amount of pixels (discrete picture elements) that together form an image. A higher resolution creates a more detailed image. The resolution is described as width times height. Standard definition television and DVD in our region use a resolution of 720x576 (PAL).

Aspect ratio
The aspect ratio of a video is measured by dividing the displayed width by its height. Common aspect ratios are 1.33:1 (or 4:3, the standard television aspect ratio) and 1.85:1 (widescreen, also called 16:9).

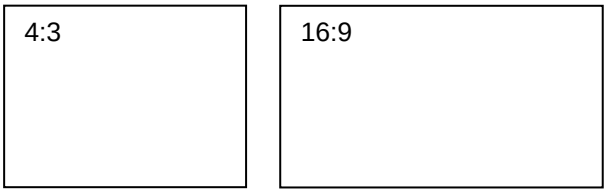


Figure 1: Video aspect ratios

Cue points
In Flash Video files, cue points are specific moments in the file that can generate an event; you add extra information (metadata) to a video. By specifying cue points at certain time points in a video file, actions can happen.

Alpha channel
A video with an alpha channel has transparent parts. You can achieve this for example by recording images using a blue screen and specifying this blue background as an alpha channel. Then this background can be made transparent (in software that supports alpha channels), and subsequently replaced by something else.

2.3. Interactive video

Interactive video
Normal video contains a predefined storyline. In interactive video the viewer can choose his or her own storyline. So you don not have to watch a predefined video sequence anymore, but you define your own story.
The author of an interactive video defines sequences of scenes in the narrative, and their structured relationship (containing link opportunities). However, the user chooses the final composition of the narrative sequence. (Sawhney, 1996)
The term *interactive movie* can be seen as a subclass of interactive video, and is often meant for interactive video that resembles or uses a movie.

Narrative sequence
A narrative sequence is a storyline. In interactive video it is a possible path through a set of linked video sequences, dynamically assembled based on user interaction.

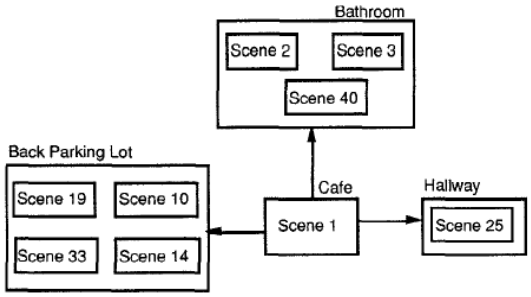


Figure 2: Spatial Map of Narrative Sequences (Sawhney, 1996)

Figure 2 contains a basic example of an interactive video. The user can choose in the first scene whether (s)he will go to the *back parking lot*, *bathroom* or *hallway*. All these choices result in a different path through the narrative. The various locations all contain scenes. These scenes in turn can be interlinked or linking to other locations. In this case, the author provides the links, but the user chooses his or her own path through the available material.

Hypermedia
Hypermedia is often used for describing interactive video related material. It is an extension of hypertext. Hypermedia consists of graphics, audio, video, text and hyperlinks. Together they create a generally non-lineair medium. Different models have been proposed for hypermedia, like the Dexter model (Halasz, 1990) and the Amsterdam Hypermedia model (Hardman, 1994).

Hypervideo
Hypervideo can be seen as a subset of hypermedia. Hypervideo is a combination of video and hypertext, providing interactive storylines. Sawhney describes a framework for hypervideo (Sawhney, 1996).
While the term interactive video is often used for game-related uses, hypervideo has been used more frequently for projects that are not game-related.
In an article from 1965, Nelson even mentions the ‘hyperfilm’ already, “a browsable or vari-sequenced movie”, basically one of the first mentions of interactive video (Nelson, 1965).

Hyperlinks (in video)
To navigate through hypervideo, you need to use hyperlinks. Sawhney (1996) and Chambel (2002) discuss various types of possible video hyperlinks:

- *Unconditional links*: these are always active, from anywhere in the video
- *Spatial links*: only dependent on space conditions; i.e. spatial regions of the video
- *Temporal links*: a time-based reference between different video scenes (a time in the source video triggers the playback of the destination scene)
- *Spatio-temporal links*: dependent on both time and space conditions, links can be established from regions of the video, for a certain period of time

Hotspot
A hotspot is a spatial location in a video to click on, also indicated above as a hyperlink (Hjelsvold, 2001). In this thesis, overlay is also used as a synonym for hotspot.

Levels of interactivity
As stated above, there can be different interaction forms for an interactive video; therefore the user can experience different levels of interactivity. The user interaction is the potential impact of a user on an application (Tannenbaum, 1998). As Cesar (2006) points out, there are also more concrete measures of the levels of interaction.

Aleem (1998) defines four possible levels of interaction:

- *passive*: the user has no control (e.g. watching video content)
- *reactive*: the control remains in the user interface (e.g. change volume)
- *proactive*: the user gains control and can for example navigate a path (e.g. interactive video)
- *directive*: user authoring (e.g. user-generated content on the internet)

Link awareness
Chambel (2002) also discusses types of link awareness. Link awareness is the challenge of how to represent link anchors in order to be perceived by the user. Important aspects are the link locations, time and duration. They identified these five parameters:

- *Where From*: where is the link anchor or hot spot?
- *Where To*: what is the destination of the link?
- *When*: when do we have link opportunities?
- *When To*: what is the destination sub-video?
- *For How Long*: for how long are the links active?

Link display
The user should also be aware of a link in a video, either by seeing it or by actively seeking it. There are two classes of required user action for link awareness: (Chambel, 2002)

- *automatic display*: the link is marked, without user action
- *user dependent display*: the user has to act in order to find the link

3. Background

In this chapter the basics of interactive video, serious games and the Clima Futura project are discussed. The final section of this chapter shows some Flash-based examples of interactive video applications.

3.1. Interactive video

3.1.1. Definition

In an interactive video (or hypervideo), the user makes choices that influence the presented material. Interactive video can be interpreted in multiple ways, though. For example, it is possible to influence the subject and flow of a video sequence by clicking *in* the video, or you could access extra information by interacting with the video; also called *detail-on-demand video* (Shipman et al, 2003). Even different points of view are possible, by clicking on characters in the video (Vardi, 1999).

But what makes a video truly 'interactive'? To define interactive video more precisely, a definition of interactive video from 1996 can be useful:

> A video application is interactive if the user affects the flow of the video and that influence, in turn, affects the user's future choices. (Eckert, 1996)

True interactive video should provide something more than standard navigation or information retrieval; it should adapt itself to the choices of the user. Therefore the main focus in this thesis is video that affects the user's future choices. Another aspect are the media types that the videos link to. The focus here will be videos that link to other videos, as opposed to hypermedia, which links to several other media, like images and websites.

3.1.2. History of interactive video in a game context

Interactive video has been used frequently in games, so this section contains a short history of interactive video in a game context.

Arcade games

The first commercial use of interactive video emerged in 1983: the arcade game Dragon's Lair. It was based on LaserDisc technology, a video format that was conceived in 1978. LaserDisc used large optical discs. The LaserDisc's random access to video chapters allowed the making of interactive movie games.[5]

Before Dragon's Lair, in 1982, another arcade game to feature LaserDisc technology had already been previewed at a game show. It was called 'Astron Belt', and it used live-action film footage, with a computer-generated spaceship on top of it.

Don Bluth, working on Dragon's Lair, saw this game and knew that he had to hurry if he wanted to be on the market first. Using a large team of animators, he succeeded. The final game features 800 decision points. The player needs to move the joystick or use the sword button to let hero Dirk make a move. If done wrong, a movie with a 'death sequence' is shown (and the player loses a life); otherwise, the movie continues. To prevent the game from being too repetitive, the game rooms (levels) were chosen randomly. Basically, the only possible gameplay was guessing which direction

[5] http://www.access-one.com/rjn/laser/ld_faq.pdf

the main character would make in the game, and hitting the 'action'-button from time to time. Despite this limitation, Dragon's Lair was an enormous success, mainly because of the quality of the sound and images.[6]

Following the success of Dragon's Lair, many sequels and clones were released. Games like Space Ace and Cliff Hanger reused the formula (animated graphics and simple controls). Some new features were added, like branches (using multiple paths). A problem for the production of these games were the huge costs: for Dragon's Lair, the animating process alone had cost 1.3 million dollars[7]. Therefore some companies even used existing (obscure) anime movies to make their own interactive movie. No real innovations were realized, only some minor changes, like simple graphic overlays that indicated which choice you should make. Mainly because of the limited interaction, the interest in the interactive movie games faded as quickly as it had arisen. Towards the end of 1984 there was little interest left in the genre.

In the beginning of the nineties, a company called 'American Laser Games' reused the formula, but with a light gun instead of a joystick[8]. The goal was to shoot the bad guys that appeared onscreen. The first game of this series, Mad Dog McCree, appeared in 1990, also spawning a range of sequels, that did not differ much in gameplay. This caused a brief revival of the interactive video genre in the arcades.

Home consoles

Following the 1983 arcade hype, there was much interest in interactive video for the game consoles at home. Companies like Atari talked about LaserDisc add-ons to their consoles. But in the end, no mainstream solutions were released.

But in the beginning of the nineties, the CD-ROM technology became feasible for home use. Consoles (like the Sega CD and 3DO), computers and CD-i could play video. For a while, people thought that titles like Night Trap and Sewer Shark would revolutionize gaming. But the same problems as plaguing the interactive video arcade games earlier persisted: limited interaction and minimal replay value.[9]
During 1995 and 1996 new game consoles were released, that could provide more realistic 3D graphics (with *un*limited interaction), like the Sony PlayStation and Nintendo 64. Also PCs were increasingly equipped with 3D accelerators (like 3dfx[10]). The interest in interactive video quickly declined. So once again, the genre had fallen into oblivion.

Online games

During the rest of the decade, there was not much interest in gaming combined with interactive video. The rapidly growing internet use on the other hand sparked some new ideas. The first online application was 'My Interactive TV', revealed in December 1998. It used filmed sequences, that adapted themselves to the choices of the users.
Years later, more examples were made, like 'Serial Writer' in 2004. It wasn't until 2005, however, that the genre got a new lease on life. Macromedia Flash 8 was released, and with it, new video possibilities. During the next years, an increasing number of applications for interactive video was made. Many of them were advertisements, like Nokia's 20 Lives[11] and Jeep's Way Beyond Trail[12]. Some are mainly focused on the video itself (like the Jeep example), and others have more game elements (like 20 Lives); more information about these examples is available in section 3.4. It seems

[6] http://www.thedoteaters.com/p2_stage6.php
[7] http://www.dragons-lair-project.com/games/pages/dl.asp
[8] http://www.dragons-lair-project.com/tech/pages/alg.asp
[9] http://www.videogamecritic.net/segacdinfo.htm
[10] http://www.x86-secret.com/articles/divers/v5-6000/v56kgb-2.htm
[11] http://www.prize-entry.com/nokia/webby/game/index_main.html
[12] https://www.patriotadventure.com/wbt/

like there is a new dawn of interactive video applications, and hopefully this time it will involve more innovation on the gameplay side.

Meanwhile, the legacy of Dragon's Lair continues: it has been released for almost every system available, now even on high-definition Blu-Ray and HD-DVD![13]

3.1.3. Interactive video as 'poor-man's immersion'

We can observe that there were multiple peaks in the usage of interactive video. It was popular in the arcades, on home systems and currently its popularity is rising on the internet. Interactive video has been called a 'poor-man's substitute for real 3D immersion' (Eliens et al, 2007). We can connect this observation to the state of the technology in the different eras.

In the arcades around 1983, no realistic graphics were possible at all. Examples of arcade hits during the time were Joust, Pole Position and Star Wars. They used very simple 2D, pseudo-3D and line-based vector graphics. Using movie-quality graphics in an interactive movie game was a real sensation. So you could call this a poor-man's substitute for real immersion, in the sense that 3D-technology was not yet ready (even though the recording of the interactive video was often very expensive).

When the first affordable CD-based consoles and PCs appeared around 1993, no realistic (immersive) environments in 3D were possible in the home environment yet. The sheer novelty value and immersion of interactive movie games made them very popular for a short time.

Right now, the internet is increasingly used for games that are using web-technologies like Adobe Flash. Though there are 3D-plugins available (like Papervision3D[14]), they are still in a development phase, and the 3D graphics they provide are not yet comparable to full 3D games. The collection of video material was a problem in the past, but not anymore, because of the rise of online video websites. So once again, interactive video can be called a 'poor-man's immersion', maybe even more than in the past.

[13] http://www.digitalleisure.com/contents/Blurayvideo_games.htm
[14] http://www.papervision3d.org

3.2. Serious games

Just a headline in October, 2007: *Dutch government invests 3 million Euro in serious games for the security sector* [15]. The market for serious games grows rapidly, and even governments take notice.

For many years games were mainly seen as toys, or worse, as having a bad influence on people. For example the game Night Trap[16] (see 3.1.2) resulted in US Senate hearings and banning of the game. Now, there are still complains about the bad influences of games, but the positive aspects are also becoming apparent.[17]

During the years, many games have been made for educational purposes. Often these games were no commercial success. But the last years, publicity around educational, or *serious* games increases.
Various goals can be achieved, like learning languages, learning ICT management skills (Eliens & Chang, 2007) and even providing training to soldiers (Nieborg, 2004).

An important aspect of a serious game is that it does not have consequences for real life. So be it a business simulation[18], or even a war simulation, no real harm is done. But lessons learned in the game *can* be put to practice in real life.

Many authors claim that gaming (and related efforts) provide a form of active learning (Gee, 2003; Vorderer & Bryant, 2006). This allows the gamer to experience the world(s) in a new way, to form new affiliations, and to prepare for future learning in similar or even new domains.

> More importantly, due to intense involvement and the need to analyze game challenges, according to Gee (2003), gaming even encourages critical learning, that is to think about the domain in a meta-level as a complex system of inter-related parts, and the conventions that govern a particular domain, which Gee characterizes as situated cognition in a semiotic domain.[19] (Eliens et al, 2007).

Serious games can be aimed at learning certain skills, but how they can also provide insights into political and social issues is being showed by PeaceMaker[20]. It is a game about the Israeli-Palestinian conflict, incorporating views of all parties involved. You need to act carefully in order to keep the balance. Real news footage increases the realistic nature of the game.
The popularity and press coverage of Peacemaker is steadily increasing, even a year after its release, and the game has won several awards. As one of Peacemaker's players puts it:

> I know more about the conflict than when I've read newspapers for 10 years. [21]

So games can also illustrate complex situations, provide players with new insights into a problem and create awareness. This is also the aim of Clima Futura (see the next section).

[15] http://automatiseringgids.sdu.nl/ag/nieuws/nieuws/toon_nieuwsbericht.jsp?di=363500
[16] http://www.defunctgames.com/shows.php?id=review-256
[17] http://www.spitsnet.nl/nieuws.php/21/2733/online/Serieuze_x_spelletjes.html
[18] http://www.theglobeandmail.com/servlet/story/RTGAM.20070502.wgtgames0502/BNStory/GlobeTQ/home
[19] Semiotic domain: a world of meaning that is due to social conventions and patterns of communication
[20] http://www.peacemakergame.com
[21] http://www.nytimes.com/2006/07/23/arts/23thom.html?_r=2&oref=slogin&oref=slogin

3.3. The Clima Futura project

Clima Futura[22] is a game about climate change. It is currently in development at VU University Amsterdam. Clima Futura is mainly aimed at a young audience: people of 12 to 26 years.
The player has the opportunity to tackle the problem of climate change, in a playful manner.

3.3.1. Origins

So there we stood, the Clima Futura team, at the presentation day of the Academische Jaarprijs. We just finished our presentation, and were very optimistic about the results. The jury's criticism was not as harsh to us as to the other 5 teams, and spectators were enthusiastic. We had the feeling that our plan, developing a game to raise climate awareness, was much more innovative than the other plans. We could win...

Clima Futura was the submission for the Dutch *Academische Jaarprijs* by VU University. The *Academische Jaarprijs* is the Dutch contest for the communication of science, where different universities compete[23]. Its main goal is to present scientific research to a broad audience. The winning team receives €100.000,- to fully develop their plan.

The Climate Centre[24], headed by prof. dr. ir. Pier Vellinga, decided to take up the challenge and to make a plan for a climate game. A multidisciplinary team was formed, to ensure that the climate-related, economic and multimedia aspects were dealt with. Between February and June 2007, this team worked on a suitable plan for the development of the game, which also had to be presented to the jury. My tasks in the process were (among other things) creating a promotional trailer[25], making designs for the screenshots of the game and to integrate interactive video into the game.

"So a game, that is shooting monsters, right?". As we were eagerly waiting for the jury to announce the two finalists on the 6th of June, we should have thought of this sentence. It was asked to us by the oldest member of the jury earlier that day. While we tried to convince him to think otherwise, and explained the value of serious games, we could not be sure if that helped. And when the two finalists were finally announced, we were not one of them, to our utter surprise. We wanted to present climate change in an innovative way, while the winners wanted to do the same thing, but by using more conventional educational material.

A new start

After a few weeks of contemplation, many of the team members strived to continue. In September 2007, a new team was formed, partly consisting of the old members. The goal is to develop a demo of Clima Futura before February 2008, and to attract investors. In hindsight, one of the mistakes we might have made for the Academische Jaarprijs was the lack of concrete material of the game, and even having a clear consensus about it in our own team.

Below some of the basic (technical) elements of Clima Futura are described.[26] It is partly based on the article *Clima Futura @ VU – Communicating (unconvenient) science*, that I co-wrote (for the full article, see Appendix A).

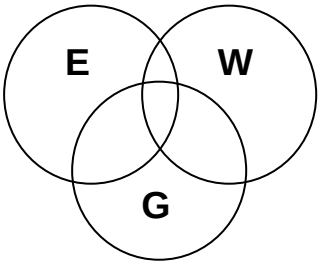
[22] http://www.climafutura.nl
[23] http://www.academischejaarprijs.nl
[24] http://www.climatecentre.vu.nl
[25] The trailer is described in Appendix X
[26] Note: the game is still in development at the time of writing, some elements might still be changed

3.3.2. Technical aspects

Platform

Different platforms were considered to develop the game. Adobe Flex was chosen, because of the flexibility and the high adoption rate. Together with the Flex 2 SDK, which recently became open source, Flex offers a Rich Internet Application (RIA) toolkit, that is sufficiently versatile to create (online) games. Games created with Flex require a comparatively moderate development effort.

Game elements



The Clima Futura game combines the following elements:

1. game cycle: turns, in subsequent rounds (G)
2. simulation(s): based on the climate model (W)
3. exploration: by means of interactive video (E)

Figure 3: Clima Futura elements

Each of the three elements is essentially cyclic in nature, and may give rise to game events. For example, game events may arise from taking turns after 5-year periods, due to alarming situations in the climate simulation, such as danger of flooding an urban area, or accidental access to confidential information in the exploration of video material. In addition, Clima Futura features mini-games, that may be selected on the occurrence of a game event, to acquire additional information, gain bonus points or just for entertainment.

Game architecture

To allow for component-wise development, a modular architecture was chosen, with four basic modules and three (variants) of integration modules, as indicated below.

Figure 4: Clima Futura architecture

1. climate model(s) - action script module(s)
2. game play interaction - event-handler per game event
3. video content module - video fragment(s) and interaction overlays
4. minigame(s) - Flash module(s) with ActionScript interface
5. Clima Futura - integration of modules 1-4, plus server-side ranking
6. adapted versions - educational, commercial
7. multi-user version - with server-side support

Without further specifying this architecture, it should offer us the flexibility to create a game that creates climate consciousness, and ultimately, that is *fun* to play.

3.4. Examples of (Flash) interactive video

Some (Flash-based) online interactive video applications have been tested for this thesis. The selection criteria were that the applications should feature sufficiently large videos[27] and are Flash-based. Also the definition from section 3.1.1 was used for the selection ('the user affects the flow of the video and that influence, in turn, affects the user's future choices')

title: Serial Writer
url: http://www.nobudget.org/serialwriter/en/
year of release: 2004 (Shockwave), 2006 (Flash)
goal: make the writer a serial killer or serial writer by making decisions

In Serial Writer you view a woman who cannot write. She observes people walking down the street. The player has to choose her thoughts to influence the story. The fragments of people walking are randomized. After a person walks by, you can choose the writer's thought. This is visualized simply by two boxes, and their text is read aloud when you move the mouse over them. A difference with most other examples provided is, that it (partly) uses a third-person perspective instead of first-person (you see the writer). No video controls are available. Compared to the sponsored interactive video applications it is somewhat less polished, i.e. the video quality and acting is not as good.

title: 20 lives
url: http://www.prize-entry.com/nokia/webby/game/index_main.html
year of release: 2005
goal: jump into one of the 20 lives and achieve the person's goals

In 20 lives you choose a person's life (for example a 'playboy') and you receive a mission. You often need to take decisions, which can vary from choosing a gift, a particular music mix, or deciding which question to ask in an interview. All these decisions determine if you succeed; sometimes you need to try different decisions in order to advance in the game. There is no possibility to save, and no pause-button (though decision-moments are not limited in time). New locations take a lot of time to load. There are transitions between video and photos at choice points. Sometimes different sounds are used for the same looped video images (e.g. when a person plays keyboard). At the end of your 'life' you see the results and a message if your life has succeeded or failed.

title: Stella Artois: L'Etranger
url: http://www.letranger-stellaartois.com/go.php
year of release: 2006
goal: tap a perfect glass of beer

Stella Artois: L'Etranger is technically the most impressive application of the examples. Even in menus, the video continues. If you need to make a choice or perform an action, this is also represented using interactive video (by clicking in the video or dragging the mouse). Everything moves along very smooth, and there are almost no noticeable loading times. In the end, you receive a rating for the quality of the beer you served. This is a simple game, which can be finished in 5 minutes (it takes some more time if you want to achieve the perfect score), but it contains some surprising moments.

[27] i.e. a video resolution of 320x240 or higher

title: Crimeface
url: http://www.crimeface.net
year of release: 2007
goal: finish the storyline while gathering enough information to solve a (crime) case

Crimeface is a media project that 'mixes the popular formats film, literature, music and gaming'. It makes use of multiple screens, and includes hyperlinks to both film and images (hypermedia). They open in the second video window. It is frequently possible to click on elements in the video. To complete the story, you need view 20 segments, which takes approximately 40-60 minutes. Unfortunately you cannot save or skip segments. The application has a lot of atmosphere, because of the music, grainy images and enigmatic storyline.

title: Jeep - Way Beyond Trail
url: https://www.patriotadventure.com/wbt/
year of release: 2007
goal: choose the right paths to reach the end (some paths reach a dead end)

Jeep – Way Beyond Trail is a sponsored interactive movie. It has very polished graphics, and professionally recorded video material. You watch video clips, and choose a path (branch) after a video ends in a separate menu. Unfortunately the transitions between the scenes are not fluent (i.e. there are loading times). There are 44 scenes, but you do not need to view them to reach the 'happy' ending. If you reach a dead end, there is always a possibility to try a different choice. The application offers the possibility to skip ahead and pause (but not to save). A map offers you the possibility to view the different locations where you have been.

Observations

When using the different interactive video applications, it became obvious that the professionally made, ad-driven applications had the smoothest visuals and better acting; though the independent applications had a lot of unique ideas.

If we compare these interactive movie games with their predecessors in the arcades and on the consoles, described in section 3.1.2, we see that these games still have many similarities. Some applications are not fundamentally different from Dragon's Lair, which was released 23 years ago. Of course, they have shorter loading times and are available online, but they share the same basic ideas. In both the Jeep and Serial Writer example you watch a clip, make a choice and watch another clip (there is no interaction during the clips). Serial Writer adds some audio cues to the choices you make. More innovation can be seen in Crimeface and l'Etranger. Crimeface offers the possibility to click in the video, to see extra information and unlock hidden material. However, it does not contain branching paths. l'Etranger has integrated the video to such an extent, that you are totally immersed. There are almost no loading times and even choices are made via video.

One thing is apparent with all interactive video examples, they do not pretend to be a full 'game', like the earlier versions in the arcades, on computers and consoles did. For example, there is usually no scoring mechanism or fixed amount of lives, while this could have made some of the games more challenging though.

All reviewed applications use Adobe Flash, and they show that interactive video is very well possible via the Flash technology; something which will be elaborated on in the next chapter.

4. Interactive video in Flash

This chapter is focused on Adobe Flash, a technology for the creation of multimedia web applications. First, its possibilities and limitations are discussed. The second part of this chapter is more technical, and provides some examples of the implementation of interactive video in Flash.

4.1. Possibilities of Flash

4.1.1. Introduction

Adobe Flash is an authoring tool that offers many possibilities for making different kinds of web content, described by Adobe as:

> simple animations, video content, complex presentations, applications, and everything in between.[28]

In Flash, these elements can be made using a *timeline*, which governs the flow of an application, and by using the *stage*, which contains the elements that appear onscreen at a certain point of the timeline.

ActionScript is the programming language that is used to create interactive functionality. It can either be included in the main project file, or stored externally (in .as files).

Finished Flash projects are exported as Flash movies, with the .swf file extension. This is an optimized version of the normal project file, reduced in file size. The Flash player plug-in shows SWF-files in a regular browser. This means that Flash files can be used on different operating systems and configurations.

4.1.2. Flash and interactive video

Flash also offers many possibilities for the creation of interactive video. Most of this can be achieved using the official authoring environment, which is currently Flash CS3 Professional. It contains some important features usable for interactive video, which are discussed below.

Flash Video format

Since Flash player 6, the Flash Video format is the proprietary standard for video files in Flash. It is possible to use separate FLV-files, which are loaded during playback of Flash movies. In 2005, Macromedia changed the compression format (codec) which is used for Flash Video. They moved from the Sorenson Spark codec to using the On2 VP6 codec. On2 VP6 yields much better compression and causes a big improvement of video quality. Soon afterwards, many large websites started using Flash Video instead of Windows Media Video and RealVideo. Also new uses emerged, like the video website YouTube[29]. It uses (low bitrate) Flash Video, and many comparable video websites, like Google Videos[30], use the same.

Currently, Flash Video is the only video format usable in Flash, but in the near future also the (open) H.264 video format will be supported.[31]

[28] http://download.macromedia.com/pub/documentation/en/flash/fl8/fl8_getting_started.pdf
[29] http://www.youtube.com
[30] http://video.google.com
[31] http://www.adobe.com/aboutadobe/pressroom/pressreleases/200708/082107FlashPlayer.html

The Flash Video format is suitable for both offline and streaming use over the internet. There are different options for the distribution, which influence the scalability and loading times. The following deployment methods are available:

Progressive download
In this case, a file is downloaded progressively from the web server. This means that you cannot skip ahead to parts that have not been downloaded yet. An advantage is that it is usable with all kinds of web servers.

Flash Video Streaming Service (FVSS)
With FVSS[32], you use selected content delivery network providers that deliver the content for you. An advantage is that the video immediately plays after it is started (no buffering time), but there is less flexibility, because you don't use your own web server.

Flash Media Server
The Flash Media Server[33] is a more advanced version of the FVSS service. It is suited to large applications, which integrate all kinds of content. An online video application that uses the Flash Media Server is Fabchannel[34], that provides an online archive of concerts (with advanced playback features).

The Flash Video format can be encoded using the Flash Video Encoder, which is in included with the Flash Professional authoring environment. Freeware tools can also be used to encode Flash Video files, like the Riva FLV encoder[35]. The official Flash Video Encoder can also encode cue points.

Cue points

Flash Video files can contain cue points (metadata at certain time points in a video). These cue points can trigger events. For example, you can generate a video overlay at a certain cue point, which is clickable. By doing this it is possible to create interactivity.

You can use two types of cue points:
- Embedded cue points: these cue points are integrated into the video itself
- ActionScript cue points: These cue points are added via ActionScript code

ActionScript cue points can be generated using XML-files. The advantage of using separate XML-files is that they are more flexible (easier to edit) than embedded cue points.

Video filters and effects

Various filters can be applied to videos in Flash. These are applied at runtime, so they do not change your source material, and you can even manipulate them during the playback of a video file.
Multiple filters can be applied at once, though they often demand much processing power.

The following video filters are available:
- drop shadow
- blur
- glow
- bevel
- gradient bevel
- color adjust

[32] http://www.adobe.com/products/flashmediaserver/fvss/
[33] http://www.adobe.com/products/flashmediaserver/
[34] http://www.fabchannel.com
[35] http://www.rivavx.com/?encoder

With these filters, you can dynamically alter the properties of a video, and let them be influenced by user interaction[36]. Also other effects are possible in Flash, for example the combination of video with (Flash) animations and still images.

Alpha channels

Flash has support for alpha channels in the videos. This means that you can make certain parts of an image transparent. The process of selecting areas of a video to mask for transparency is called keying. In many video applications you can select a key color, based upon which a mask for the video is created.

After specifying an alpha channel, you can for example stack multiple videos, or combine a video with a static background.

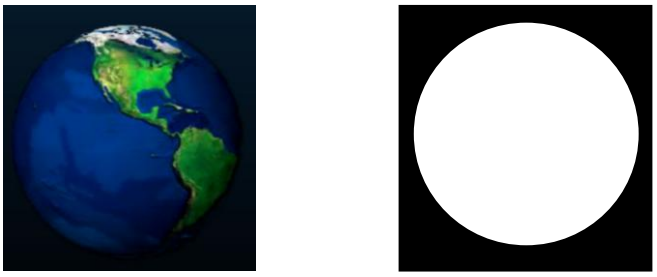


Figure 5: Alpha channel plus drop shadow & the black mask to create transparency

In Figure 5, a short animation of a rotating earth is used[37]. In Flash, a drop-shadow effect is added. The background of the image has been removed using a color key in Adobe Premiere Pro 2.0. The second figure indicates the mask that is encoded in the alpha channel, which determines the part of the image that is transparent.

Video overlays

Important elements of Flash are reusable objects, called *Symbols*. These are graphics, buttons or Movie Clips. A graphic symbol is the simplest object of the three. A button can be used for various user interactions. Movie Clips resemble buttons, but can contain other objects, and can even have their own timelines (they should not be confused with clips from a movie). So Movie Clips are very important in Flash, and can be used for all kinds of different functionality.
By specifying a Movie Clip to be clickable, and by altering its size, you can create an overlay on a video, to be able to make hyperlinks to create truly interactive video (i.e. if users click on a video, another video is loaded). These Movie Clips can either be visible, but also transparent or invisible; thus influencing the link awareness (see section 2.3).

XML

XML has become a standard structure for all kinds of different purposes. Flash has support for XML, which allows you to specify important assets of your project in XML. This creates versatility, because you can now edit these assets with a simple text editor[38]. Ideally, you will be able to share these XML-files with other people and programs. For more information about XML, see chapter 6.

The methods to process XML in Flash have changed considerably since Flash CS3 (it is now easier to access nodes; for more information see Appendix B).

[36] e.g.: http://www.flashcomguru.com/index.cfm/2005/9/29/Live-video-effects-in-Flash-8--a-small-demo-app
[37] By C.J. Hamilton: http://www.solarviews.com/cap/earth/vearth3.htm
[38] Note: some tools, like jEdit (jedit.org), also provide syntax highlighting, which eases the editing

4.2. Limitations of Flash

There are also some drawbacks to the use of Adobe Flash in an interactive video application, which are related to the Flash environment itself and the Flash Video format.

Flash & ActionScript

The Flash format itself underwent some major changes in the last few years, which caused the need for upgrade work in many web applications. These changes were described by some as 'growing pains'.[39] However, the arrival of ActionScript 3.0 will hopefully result in a more stable developing environment. It is available in Adobe Flash CS3 and higher; the previous versions (like Flash 8) use ActionScript 2.0 or even 1.0.

Flash Video format

The use of a proprietary codec for Flash video causes some problems. You cannot import your videos into your projects directly. Instead, you need to convert every file to the FLV format, which takes time.

Also, there are no advanced playback features yet, like playback speed variations or reverse playback. This means that it is harder to use e.g. slow-motion for a choice moment in interactive video. You would need to add this effect in another application before you encode the video to the Flash video format. It is not possible to fast forward or rewind the video, though jumping to chapters and points in the timeline is possible.

Finally, if you want transparency in your videos, you need to specify an alpha channel (i.e. colors or areas that need to be transparent). It is only possible to add an alpha channel using a professional video editing application (like Adobe Premiere or Adobe After Effects). You then also need to enable the alpha channel when encoding the file to the FLV-format. It is not yet possible to edit an alpha channel in the Flash application itself.

4.3. The workings of an interactive video application

To apply the theoretic elements of the previous section to a real situation, the technical structure of a possible interactive video application in Flash is discussed. Firstly, the elements of an interactive video application are indicated. Secondly, a (simplified) visualization, and thirdly, a more in depth description of the steps of the program is included.

4.3.1. Elements of an interactive video application

A Flash application for interactive video can consist of the following elements:

The FLVPlayback component
This is a Flash component to play Flash video files (FLV). It is automatically added to your project if you add a video in Flash Professional[40]

A (dynamic) playlist for videos
This can be done via ActionScript (possibly combined with XML). The playlist needs to be dynamic, i.e. adjusted if choices are made by the user. It is played via the FLVPlayback component.

[39] http://www.mediacollege.com/flash/video/pros-cons.html
[40] By using import > import video

Video files
These are the separately encoded FLV-files (using the Flash Video Encoder or freeware tools); or downloaded files from a Flash-based video website like YouTube.

A method to add clickable areas to the video
They can be encoded as cue points in the FLV-files, or be specified in an XML-file with cue points.

An event handler for cue points in Flash videos
The cue points in the FLV or XML-files must be read by Flash. Using that data, an image overlay can be placed on top of the video (to be able to click on a spot of a video)

Note that this is not the only way to achieve interactive video in Flash; other methods can also be used (by e.g. embedding the video into the Flash-project itself). An advantage of the method mentioned above, with separate movie files, is that it is flexible: you can add new videos and change the annotation afterwards.

4.3.2. Visualization

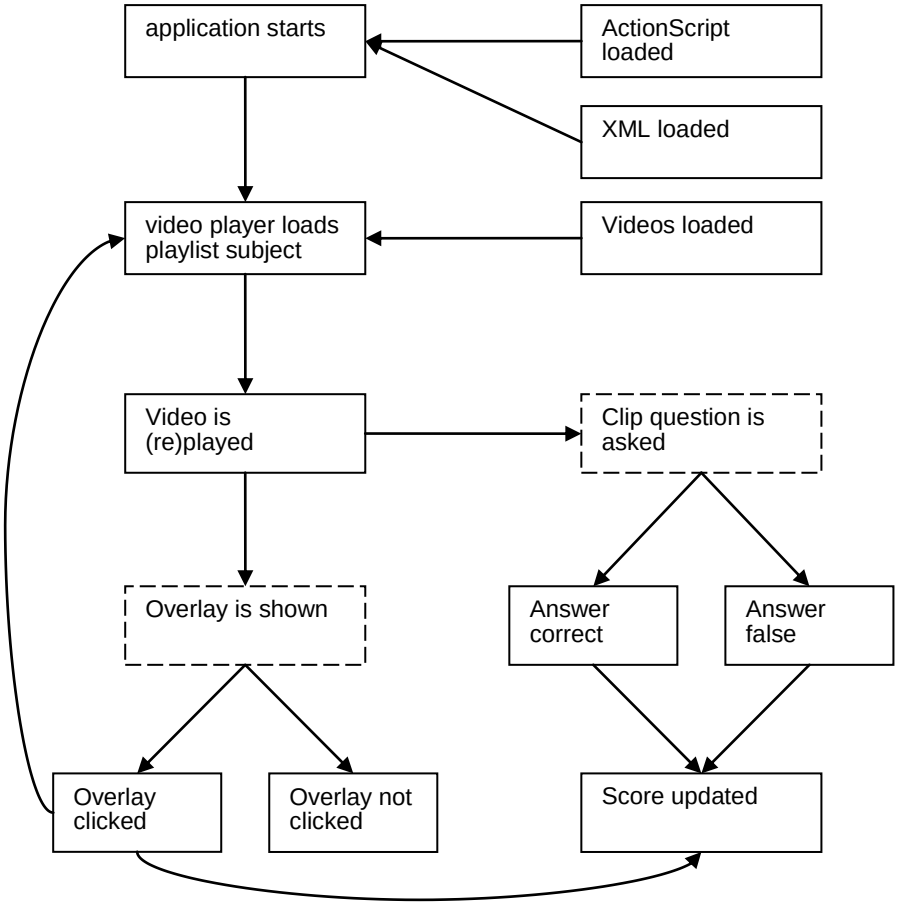


Figure 6: Simplified application structure

A (simplified) representation of an interactive video application's structure is shown in Figure 6. Basically, when the application is started, ActionScript files are loaded, and an XML-file with subject and video information is read (subjects contain playlists of videos).

After the video player is started, the videos in this playlist are loaded; and playback commences. Optional clip questions and overlays are shown during the videos. Overlays are the hyperlinks a user can click on, and clip questions are questions during a video. If an overlay is clicked, a new playlist is

loaded (corresponding to the clicked subject). The score is only updated if the user clicks an overlay, and if (s)he answers a clip question.

4.3.3. Description of program steps

To define the program structure of an interactive video application further, the specific steps that are taken by the Clima Futura pilot application are described below. Further information about the application is available in the next chapter.

After starting the application
- The program reads an ActionScript file, which defines the menu
- The program starts a video of a rotating earth in a *video component*

After clicking *next*
- The program reads all ActionScript files
- The program reads the XML-file with subjects, videos and overlays
- The subject information from the XML is converted to an array with *Subject* objects
- The video information from the XML is converted to an array with *Video* objects

After clicking *start movie*
- The program loads the subject information from the array with *Subjects*
- The program loads the videos of the first subject, that are located in the array with *Videos*
- The overlay information from the *Videos* array generates (ActionScript) *cue points* in the loaded videos
- The first video clip is started
- If there is a *clip question*, it is overlaid on top of the video

After detecting a *cue point* with overlay information (using an *event handler*)
- One or more overlays are shown, altered or removed; based on the cue point information
    - by altering the size of a *MovieClip* containing a half-transparent ellipse
- If there is a *branch question* specified, a question is shown above the overlay(s)

After a video clip ends
- The next video in the subject is displayed

If the last video clip of a subject ends
- It loads the next subject (if specified), or
- Repeats the last clip (if specified), or
- Displays the end screen (this is the end of the *narrative sequence*)

After answering a *clip question*
- The result of the answer is shown (correct / incorrect)
- Points are deducted from the total score if wrong, or
- Points are added to the total score

After clicking on an *overlay* (‘hotspot’)
- Based on the *label* information of the overlay, a new set of videos (*subject*) is loaded, corresponding with the label name
- The previously played subject is added to the *history* array
- A short transition clip is shown, corresponding with the *subject* name
- The score specified for the *hotspot* is added to the total score
- The first video of the subject is started

After reaching the end of the *narrative sequence*

- The end screen is displayed
- A message is generated, depending on the total score
- A graph with the followed storyline is shown (based on the *history* array)

Based on these program steps, a prototype for interactive video was created, which is described in the next chapter.

5. Interactive video in practice - Clima Futura pilot

The previous chapters focused on different aspects of interactive video, serious games and technical implementations. This chapter describes the implementation of an interactive video prototype that served as the Clima Futura pilot. Topics discussed are its development, video collection and the narratives.

5.1. Pilot results

The prototype is dedicated to the Clima Futura project (see chapter 3.3). This means that the video material of the application will include climate-related issues.

5.1.1. Prototype overview

By using the interactive video application, viewers do not have to watch a predefined video sequence anymore. Instead, they define their own story. By making decisions, discovering new storylines and answering questions, they will find their way through the material.

Choices can be made by clicking in the video itself, for example on the person the player agrees with. Hidden storylines are also clickable, so you can find information, conspiracies and more.

Points are awarded for decisions, discovered storylines and answered questions. So the player's performance is judged, thereby creating incentive to watch and play again.

5.1.2. Detailed features

The prototype is based on a set of 50 short climate-related videos. This large set of videos is divided into smaller *subjects* (containing a smaller selection of videos), which are played sequentially. The viewer can make choices at certain points to influence the narrative, by clicking on *hotspots* in the video. These choice points are called *branch points*. By making choices, the player earns points.
Sometimes, it is also possible for the player to click on *hidden hotspots*, less obvious spots in the video (providing extra bonus points). Furthermore, it is possible to answer questions that are asked during the videos (called *clip questions*). These questions also earn the player bonus points.

Hotspots

Hotspots are the video overlays that provide interaction with the video and represent hyperlinks to other videos. They are represented by grey ovals, which are partly transparent and can be altered in size and position. The overlays are automatically shown, without user interaction. They are highlighted if a user moves the mouse over them. Early user tests showed that the hotspots are very difficult to recognize if they aren't visualized clearly. When a user moves the mouse over a hotspot, the overlay is highlighted, and a textual description of the link is shown.

At a higher difficulty setting the links are not visible as clearly, to provide an extra challenge to the user.

Branch questions

Figure 7: A branch question

At a branch point, a user has to make a choice. This is indicated by a question above the video. It is possible to click on certain spots of the video, indicated by the overlays. By hovering over the overlays with the mouse, answers to the question above the video are shown (corresponding to the selected overlay).

An example question is: 'Are climate measures rubbish, or are they really needed'. By hovering over the heads of the (previously seen) experts, the answers 'Climate measures are good' and 'I don't want climate measures' are revealed. In the current prototype, up to three answers can be defined. If the user clicks, (s)he is redirected to one of these storylines. The video clip repeats itself until a choice is made.

The type of hyperlink at a branch-point is *spatial* (see 2.3), this means that it uses a spatial region of the video.

Clip questions

Figure 8: A clip question

During a video, statements can be shown to the viewer. The viewer has to respond quickly, because after a certain amount of time, the question will automatically disappear. This time is short, to provide a challenge to the viewer. The answers are simply 'true' or 'false', indicated by the green 'V' and red 'X' signs. If the user answers correctly, (s)he gets bonus points, otherwise points will be deducted from the score. Immediately after answering, it is shown whether the answer was correct or wrong. In Figure 8, the statement 'this is pathos' is shown. The user has to determine quickly if this is the case, and can gain 500 bonus points.

Hidden hotspots

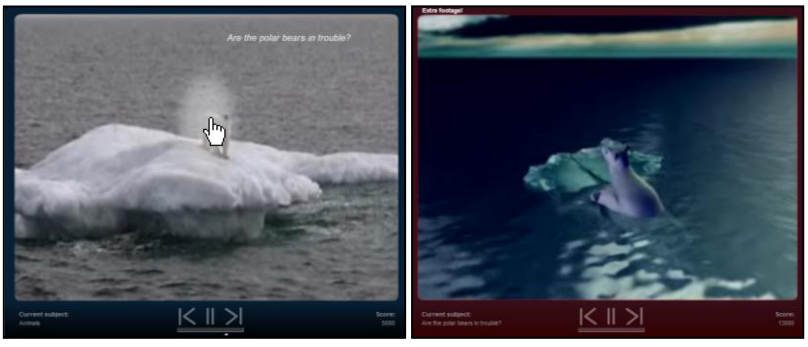


Figure 9: Polar bear hidden hotspot and the resulting animation

Sometimes there are hotspots that are less visible than the branch points. These hotspots are optional, and automatically disappear after a certain period of time. They can reveal a hidden storyline, for example providing additional insight in a specific problem or showing a conspiracy. Hidden hotspots gain the user bonus points.

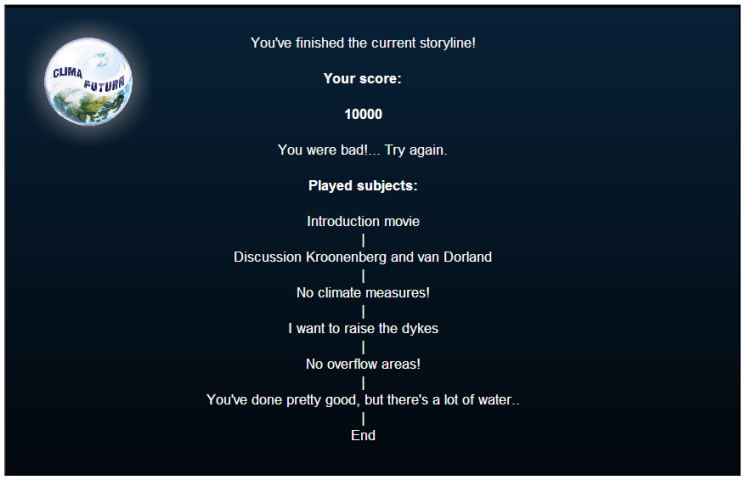
An example: if a user clicks on a hidden hotspot containing a polar bear, a bonus movie is shown about the problem of the decreasing habitat of the polar bears. The screen border temporarily changes color and a message indicates that the user has unlocked extra footage. If the viewer did not click on the hidden hotspot, a movie about humans is loaded automatically.

The link type of hidden hotspots is *Spatio-temporal*, this means that they are dependent on both time and space conditions (the polar bear hyperlink is only available for a certain period of time, at a certain spatial location of the video).

Video transitions

Between videos, visual transitions are shown to increase the fluidity of the material. In a subject, these are short fades. Between subjects (so after the user clicks on a hotspot), there is a short loading time. During this time, a short video is superimposed over the original video screen, and it gradually fades away. This video is visually connected to the subject it leads to (e.g. you see a polar bear if you click on 'Animals').

End screen & final score



When the player ends his storyline, the final screen is shown. It displays the final score, accompanied by comments, like 'you did great' or 'you were bad', along with some explanation. Furthermore, the list of played subjects in the narrative structure is shown. The player now has the option to try again (or to quit). The scoring mechanism should create incentive to try again, and to try to get the highest score (thus viewing more of the video material).

Figure 10: End screen

Difficulty settings

The application has three difficulty settings: *Easy*, *Medium* and *Hard*. These settings now influence the parameters indicated in table 1.

	Back-button	*Clip question time*	*Hotspot visibility*	*Score multiplier*
Easy	Yes	-	55%	900
Medium	Yes	8 seconds	45%	1000
Hard	No	7 seconds	25%	1100

Table 1: Difficulty settings

In the current application, the back-button is disabled at the *Hard* setting, while clip questions are disabled at the *Easy* setting. The hotspots are more difficult to see at the higher difficulty settings, but this is rewarded with a higher score multiplier. Every hotspot and clip question has an associated score value, which is multiplied by the score multiplier.

5.1.3. XML structure

The application should facilitate easy modifications. Therefore the different elements of the video narrative are all specified in XML. It is possible to encode FLV-files and add them to the different storylines. The questions, hotspots and scores to the videos are editable in XML.

More information about the XML structure can be read in the next chapter.

5.1.4. Visual design

The goal for the interface of the interactive video application was to redirect the viewer's attention to the video itself. Different interface colors and types have been tried, but in the end a minimalist interface seemed to work best. Sawhney et al (1996) used a similar approach in their *Hypercafe* application:

> By minimizing the traditional computer-based artifacts in the interface and retaining a filmic metaphor, we hope to provide the user with a greater immersion of the experience.

The first versions of the interface had no visual elements except a small *next* and *pause*-button. As interactive video is not really common these days, this might have been too simple. Therefore navigational aids were added, like a 'previous'-button, an indicator for the current position in the video and a description of the current subject.

Figure 12 shows a schematic visualization of the final interface, and a short description of its elements. In the final application, a dark blue interface is used, for both the menus and the video player itself.

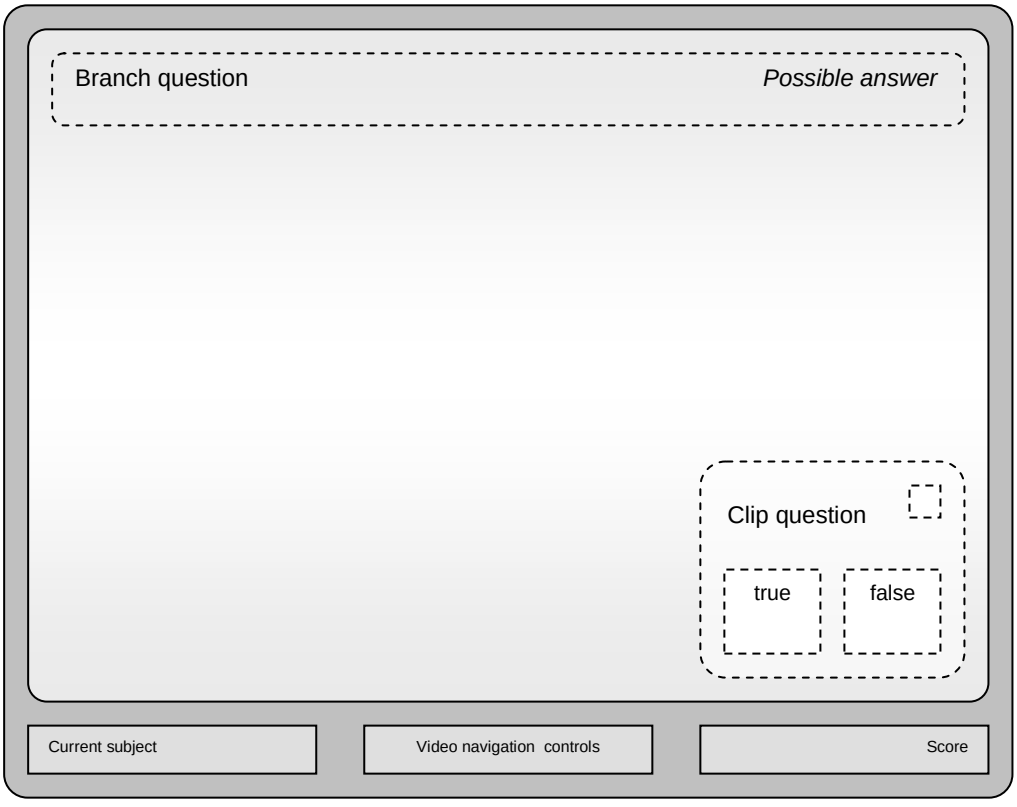


Figure 11: Schematic visualization of the interface

Persistent interface elements:

Available video navigation controls in the interface are a *previous*, *play/pause* and *next* button; and a pointer that indicates the current position in the video (which can be dragged for time-based navigation in the current video). The previous button is disabled if you choose the 'hard' difficulty setting.

Furthermore, the current subject name is indicated on the lower left corner, and the score on the lower right corner.

Temporary interface elements:

Branch questions are asked on the left side of a semi-transparent box on top of the video. The answer to the branch question is textually indicated on the right side of the box when a user hovers over a hotspot. If there is a hidden hotspot, only the 'answer' text is shown (without a box).

On the lower right there is an optional box for the clip questions. A statement is shown, and the true and false buttons can be clicked. A timer is located on the top right of this box.

5.2. Video collection and editing

5.2.1. Suitable video material

Essential to an interactive video application is, of course, appropriate video material. Most ideal would be to record video specifically for this pilot (like the examples shown in section 3.4). Unfortunately this is an expensive and time-consuming activity, so for this project video material needed to be collected differently.

Using already recorded material is a good option, though it is more difficult to construct suitable storylines from it. Video has been recorded for a specific purpose, which might not necessarily coincide with this project's purpose.

Then there is also the choice of genres to use in the narratives. Suitable genres could be news programs, news reports, interviews and debates. News is suitable because of the value to have current affairs in the application, while interviews and debates provide multiple opinions, which might provide choices to the users. Debates involve a statement (proposition), and multiple sides that defend or attack a given proposition[41]. So a debate creates an opportunity to show multiple sides arguing about a statement, and can offer a choice to the viewer. After all, choices are essential to truly interactive video.

5.2.2. Video sources

To add videos to the interactive video application, I started with the video collection I gathered for making the trailer of Clima Futura, consisting of television and DVD clips (for more information about the trailer, see Appendix F). Furthermore, I searched for climate-related material at the Dutch portal of Netherlands Public Broadcasting[42], called *Uitzending Gemist*. YouTube[43] offered the possibility to find a wealth of material, and Google Videos[44] to a lesser extent.

The basic strategy of finding suitable fragments was to search for clips with multiple persons, or discussions, without a lot of movement (which makes the annotation easier). Video clips were then edited using Adobe Premiere (and other tools), until short clips of 10-30 seconds remained.

[41] American Heritage Dictionary, 2006, via http://dictionary.reference.com/
[42] http://www.omroep.nl/uitzendinggemist
[43] http://www.youtube.com
[44] http://www.videos.google.com

5.2.3. Video quality

	Format	*Resolution*	*Sound*	*Bitrate (kbit/s)*
DVD	MPEG2	720x576	Stereo/surround	3000-9000
YouTube	Flash Video (FLV)	320x240	Mono	300
Google Videos (1)	Flash Video (FLV)	320x240	Mono	450
Google Videos (2)	Custom AVI[45]	480x360	Stereo	800
Google Videos (3)	MP4	320x240	Stereo	400
Uitzending Gemist	ASF/WMV	320x180	Stereo	500

Table 2: Video sources

Without getting into too much detail, it is obvious that there are large differences between the different sources for video. The source material from DVD has by far the highest resolution and bandwidth. The MPEG2 compression format provides the highest quality, though it is less efficient than the other formats used. Streaming formats, like YouTube, need to have a low bitrate because they need to be played via the internet. The Google Videos AVI format (downloadable via Google Videos) used to have a higher bitrate and resolution than the other formats, but the support for this format is discontinued since August 15, 2007 (when Google closed its video store)[46].

To obtain the material, different methods must be used. For DVD, ripping tools are needed to copy the material to the hard disk. For YouTube this strategy was slightly different. Clips needed to be downloaded using a YouTube video download tool[47], or retrieved from the browser's cache. The MP4-files on Google Videos are easy to download, by clicking on 'Download for iPod/Sony PSP'. Broadcasts from Uitzending Gemist are very difficult to download, so separate tools are needed.

Important point to note is that these 'lossy' formats degrade when a file is repeatedly re-encoded. So it is best to encode a file from the source format to the Flash Video format directly.
YouTube and Google Videos (1) are a different case, however, because they already are Flash Video. For these files, RichFLV[48] was used to remove unnecessary parts from clips. Disadvantage of using downloaded FLV-files is that they are already encoded, and cutting is only possible at certain positions of the video (keyframes).

During the process, also obtaining the video material via the Netherlands Institute for Sound and Vision[49] was also considered. They can provide your own selections of videos on DVD. Ultimately, this was not used, because of the large availability of videos on the internet, and because of the costs[50].

5.2.4. Final video format

In the prototype application, the video is encoded in the Flash Video format. The resolution of the videos can vary, as long as they have the same aspect ratio. The application is now suited to a 1.33 (also called 4:3) aspect ratio (e.g. 320x240 or 640x480). Using different aspect ratios sometimes results in problems with the size of the videos in the application, so the 4:3 images cannot be mixed with 16:9 (widescreen) images. The current application only uses the 320x240 and 640x480 resolution.

[45] Note: discontinued by Google
[46] http://www.techcrunch.com/2007/08/10/google-closes-video-marketplace-users-out-of-luck
[47] http://www.techcrunch.com/get-youtube-movie
[48] http://www.richapps.de/?p=48
[49] http://www.beeldengeluid.nl
[50] Costs: €7.25 / € 12.45 per video segment and €33.10 per DVD (with one segment of less than 60 minutes)

The bitrate of the videos can vary. To facilitate the use on the internet and decrease the memory usage, the current video bitrate is 400kbs, and the audio bitrate is 96kbs (stereo). Still, it would be possible to make a version with DVD-quality images, by altering the bitrates and resolution.

5.2.5. Video narrative example

One of the clips involved is a discussion between Salomon Kroonenberg, geologist, and Rob van Dorland, climatologist at the KNMI (Dutch weather service). This debate is selected from a broadcast of 1 Vandaag. They provide a lot of arguments, but in the end it comes down to Kroonenberg saying that we cannot influence the climate, and we should not take measures. On the other hand, van Dorland says that we should take climate measures while we still can.

Ultimately, one or two arguments per person provide enough information. To be able to make a choice, a shot of both persons is needed. An overlay can be used to indicate the possible choices. In this case I edited a sequence in Adobe Premiere, removed the sound and used slow-motion to increase the scene's length.

For a different part of the video narrative, a broadcast of 'Zembla' was used. The original broadcast has a duration of approximately 40 minutes. It is a documentary about the use of overflow areas (*overloopgebieden*) in certain areas of the Netherlands, which reduces the water pressure when there is a big risk of flooding.

For the interactive video, some very short clips were needed. Two persons in the documentary were striking: a farmer that would lose his land, and a civil servant. The farmer has a very emotional story (using *pathos*) and the civil servant has a logical, almost cold, monologue about the subject, emphasizing the necessity of the measures (which could be classified as *logos*).

At the end of the sequence of clips, the viewer has to make a choice between these points of view. A problem is though, that both persons are not visible in one shot in the documentary. The solution was to make a composition of both images. This could not be done with the supplied Flash Video encoder[51], so this was done using Adobe Premiere.

Figure 12: Rob van Dorland and Salomon Kroonenberg
Figure 13: Composition of a farmer and civil servant

[51] More information about the (dis)advantages of the Flash Video encoder is located in Appendix B

5.2.6. Clip length

Video can be segmented into clips, which are sequential sets of frames. In order to make a clip, you cut out a “piece” of a longer video (Bocconi, 2006).

In the video application, clips are used to build a story (in a subject). As Bocconi observes, the granularity of a clip is a key issue. Not all clips can be edited together:

> If a clip is too short, it cannot establish sufficient context for the viewer to understand it. The meaning of the clip is dependent on what comes before and after. (Bocconi, 2006)

The effect of a clip’s granularity is known as the Kuleshov-effect:

> The Kuleshov-effect is based on leaving out a scene’s establishing shot and leading the spectator to infer spatial or temporal continuity from the shots of separate elements (Thompson, Bordwell, 1994).

In the example of the Zembla broadcast in 5.2.4, a clip with the civil servant is not long enough to use on its own. If it was used in a different context, it would get a different meaning. In this case, an introduction video is played before this video, which explains the problems at stake. So the meaning of the clip ultimately becomes (roughly) the same as originally intended.

6. Interactive video and XML

When developing the interactive video prototype, different options were considered to define the storylines. At first, the application used embedded cue points to define the choices in the video. Later in the project, a text-based approach was preferable, so the annotations can be edited by hand. The most logical choice would be XML, so an XML structure needed to be defined. In this chapter. Therefore this chapter contains the basics of XML, a structure for interactive video and the actual application of the structure in the Clima Futura prototype.

6.1. XML basics

XML is the 'Extensible Markup Language', and it facilitates the creation of documents and data records that are platform-independent. The tag structure of XML resembles traditional HTML, but the word *Extensible* already indicates that you can add your own tags. XML is an open standard, and recommended by the World Wide Web consortium (W3C) [52].

XML contents

XML documents should conform to all of XML's syntax rules. A file that complies to this level of correctness is called *well-formed.* A basic XML-file looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>

<contact-information>
               <name>Hugo Huurdeman</name>
               <email>hchuurde@few.vu.nl</email>
               <initials=”HH” />
</contact-information>
```

The XML-example starts with the *XML declaration*, which indicates the XML version and (optionally) the character encoding. After that, there is one root tag (or container node), which contains all other nodes. The lower nodes in the hierarchy are called *child nodes*. For example `<name>` is the first child node of `<contact-information>`.

An XML-document contains both *markup* and *text*. The markup, like `<name>` in the example, describes the structure of the document, while the text is the actual information.[53]

The markup is specified using *XML-tags*, which include the < and > characters. An example of a s*tart tag* is `<email>`, while `</email>` is an *end tag*.

Attributes can also be specified in XML. They are single properties for an element, and use *attribute name* and *attribute value* pairs. The tag <initials=”HH” /> is an empty-element tag, and contains no text inside it, but does contain an attribute. Attributes should be quoted, and inside a node, an attribute name can only be used once.

XML Schema

The first part of this section covered the *well-formedness* of XML. Another aspect is the *validity*. An XML-document is valid if it conforms to certain semantic rules. Users can define these for themselves, or enforce an XML-schema. XML-files must then conform to this schema, and this can be checked automatically. Different languages for the specification of XML-schemas can be used.

[52] http://www.w3.org/TR/REC-xml
[53] http://www.xmlnews.org/docs/xml-basics.html

An XML-schema for the Clima Futura pilot can be found in Appendix D.

6.2. An XML structure for interactive video

For interactive video, an XML-structure had to be defined. This gradually evolved during the project. The application used multiple files at the beginning of the project, but at the end of the project, the video annotation had to be defined in one, compact, XML structure.

Basic requirements

To dynamically play videos based on user interaction, an XML-file for interactive video should include several elements. A playlist for videos is needed, that adapts itself to the choices of the user. Furthermore, the videos need to be linked to each other, to provide choices to the user. To allow the user to make choices, clickable hotspots need to be defined. Another important aspect is flexibility, the structure should allow the addition of future features.

Clima Futura XML elements

During the development of the interactive video application, several elements turned out to be important. These elements were subjects, videos and overlays:

- Subject: a (small) collection of videos, that is played sequentially (it could also be called a playlist).
- Video: a Flash Video file, indicated in XML by the filename
- Overlay: clickable hotspots in the video, that can respond to actions of the user; and indicate possible choices

Overlays are an important element of interactive video. If you need to specify these manually, though, this takes a lot of time. To reduce the time needed to add overlays, the Clima Futura pilot uses a 5x5 matrix to specify overlays. Of course, this is less precise than specifying exact coordinates, but the advantage is that only a few parameters need to be set to show the overlay. A more extensive explanation can be found in the next section.

6.3. Clima Futura pilot XML-structure

Below you find the XML-structure of the Clima Futura pilot application. The *subjects*, *videos* and *overlays* can be seen here, as they are the main contents of the XML-structure.

```
<?xml version="1.0" encoding="utf-8"?>
	<subjects>
		<subject name="Example">
			<longname>Example subject</longname>
			<score>5</score>
			<videos>
				<video file="001_Example_01">
					<question>A clip question is specified here</question>
					<rightanswer>true</rightanswer>
				</video>
				<video file="002_Example_02" repeat=”true”>
					<branchquestion>A branch question is specified
here</branchquestion>
					<overlays>
						<overlay time="0" position="01" size="12"
label="AnotherSubject" />
					</overlays>
				</video>
			</videos>
		</subject>
```

XML Nodes

Subject

A subject contains a collection of videos. The ‘name’ parameter is the (unique) name for the subject, that is also used for the video filenames (see Appendix B). The videos of a subject are automatically played after each other. Subjects have a ‘longname’, that is displayed in the interface during the video and when the mouse is hovering over a link. Furthermore a subject has a score, ranging from 1 to 5 (awarded when the player clicks on that subject). The score is multiplied by 1000 in the application itself to get a more ‘game-like’ value.

Subject		
Attribute:	name	String
Subnodes:	longname score videos	String Number

Table 3: Subject node

Video

A video is a sublevel of a subject. Per node, a video is specified using the ‘file’ attribute (corresponding with the filename of the video, without the extension). Optional parameters of a video are:

- *repeat=true*, when the video needs to be repeated (useful if a user *has* to make a choice)
- *leadsto=true*, when the video automatically leads to another subject.

For a video, there are *clip questions* and *branch questions*. Clip questions (between <question> tags) are the questions that are displayed during a video. The right answer is specified using the <rightanswer> node. A branch question is a question when there are multiple choices possible. The answers to the question are specified in the overlays (see below)

Video		
Attributes:	file repeat (opt) leadsto (opt)	String Boolean Boolean
Subnodes:	question (opt) answer (opt) branchquestion (opt) overlays (opt)	String String String

Table 4: Video node

Overlays

An overlay is a ‘hotspot’, a spot to click on in the video itself. As has briefly been described in the previous section, the application creates overlays by using a 5x5 overlay system.

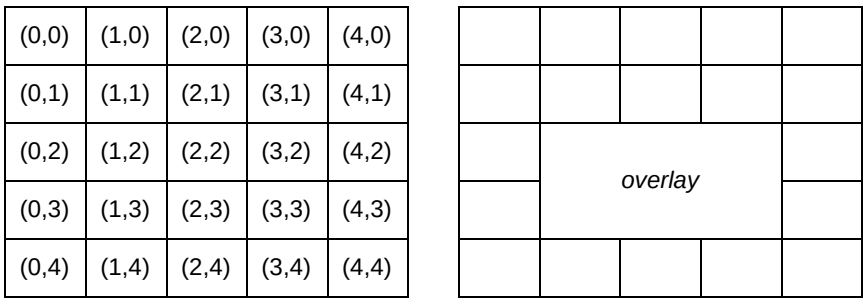


Figure 14: Coordinates of overlays & a sample overlay

Key elements of an overlay are the coordinates, width and height. The coordinates are located in figure 14 above. The width and height of an overlay can be 1 to 5 units. So an overlay at the position (1,2), which is 3 units wide and 2 units high, looks like the figure on the right.

Now these overlays need to be specified in XML. This is done via the (obligatory) *time*, *position*, *size*, and *label* parameters. To provide a short notation, the position and size parameters are indicated by a combination of the two single-digit numbers (for position x and y; for size the width and height).

The *time* is the moment when the overlay should be shown, measured in seconds from the beginning of a clip. The *position* can be 00 to 44, which are short notations for the coordinates (0,0) till (4,4). The *size* can be 00 to 55, short notations for (0,0) and (5,5). The label corresponds with the subject name it refers to, that also needs to be specified in the XML-file. So the overlay in 0, at the position (1,2), with a size of (3,2), is specified by the following tag:

```
<overlay time=”0” position=”12” size=”32” label=”Example” />
```

Up to 2 additional overlays can be specified using additional *position2*, *size2*, *label2* and *position3*, *size3*, *label3* parameters.
To remove one or more overlays from the screen, you can provide the overlay time without any parameters, like: `<overlay time=”4” />`

Overlay		
Attributes:	Time	Number
	position (opt)	Number
	size (opt)	Number
	label (opt)	String
	position2 (opt)	Number
	size2 (opt)	Number
	label2 (opt)	String
	position3 (opt)	Number
	size3 (opt)	Number
	label3 (opt)	String
Subnodes:	-	-

Table 5: Overlay node

6.4. Narrative structure visualization

To clarify the structure of the application, right before it will be specified in the XML-structure, the first four levels of the video tree for the Clima Futura interactive video application are visualized in Figure 15. It is an adapted version of the 'Spatial Map of Narrative Sequences' by Sawhney (see section 2.3, Figure 2:). The next section shows the corresponding XML-structure.

- The large boxes are subjects, that contain smaller videos (which are played sequentially)
- A *Q* indicates a video with a *clip question*
- *Branch points* are striped
- Scenes with a *hidden hotspot* have an striped oval in it and connected to the resulting subject
- Subjects with one connection automatically lead to another subject

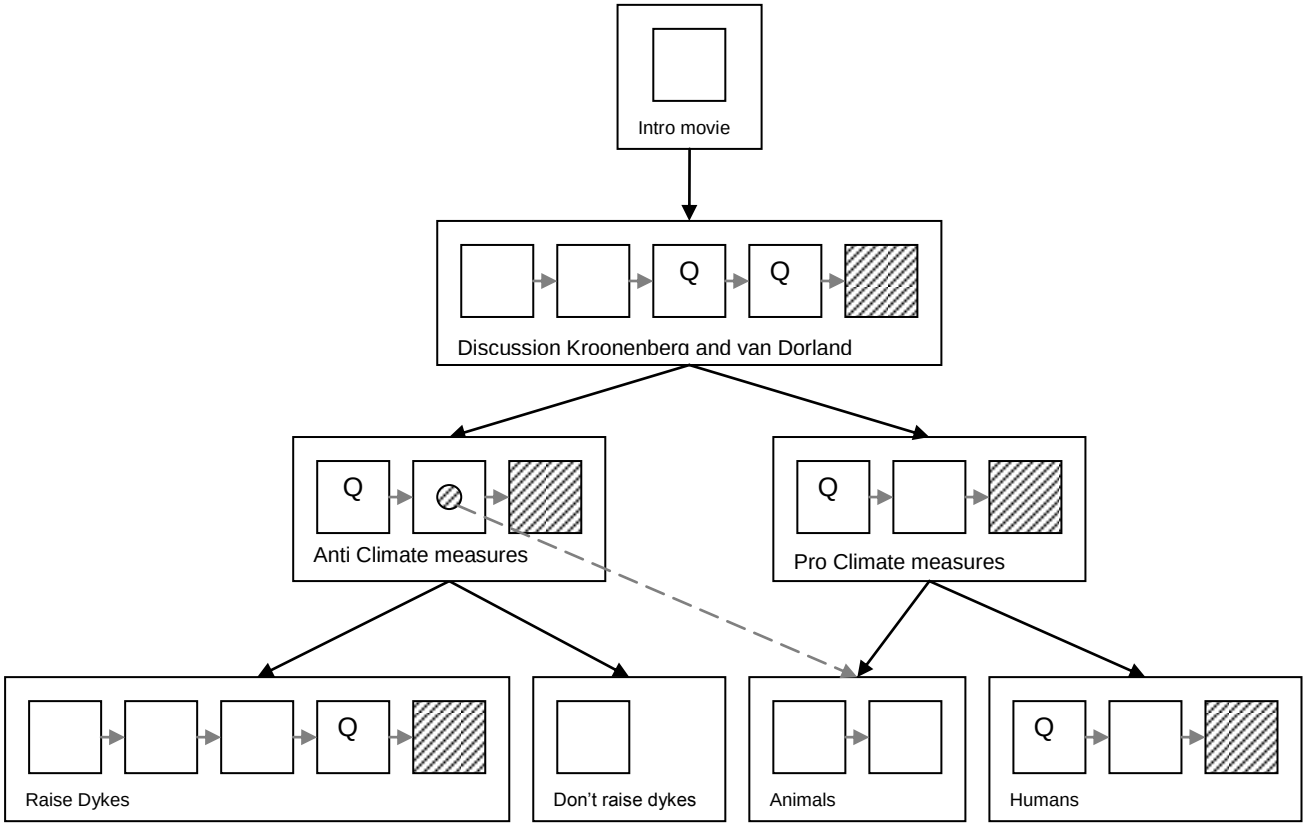


Figure 15: Video branching structure

Some theoretical background on narrative structures can be found in the next chapter.

6.5. Application of the XML structure in practice

Now we have seen part of the narrative structure of the Clima Futura pilot, we can apply this to the XML-structure. Three subjects are included here.

Subject 1: Introduction movie

The first clip involved is very straightforward. It is just a simple Clima Futura intro movie, that is played before the actual information about the climate starts.

```
<subject name="IntroMovie">
	<longname>Introduction movie</longname>
	<score>0</score>
	<videos>
		<video file="000_IntroMovie_01" leadsto="DorlandKroonenberg" />
	</videos>
</subject>
```

The longname is used to produce a more legible name, and there is no score used, because there are no choices yet. The *leadsto* attribute is meant to continue the narrative. It refers to the next subject, which is called *DorlandKroonenberg*.

Subject 2: Discussion Kroonenberg and van Dorland

Very early in the application, you see a discussion between Rob van Dorland, a climatologist and Salomon Kroonenberg, a geologist. This discussion uses five video files, the last of which indicates a branch point.

```
<subject name="DorlandKroonenberg">
	<longname>Discussion Kroonenberg and van Dorland</longname>
	<score>0</score>
	<videos>
		<video file="001_DorlandKroonenberg_01" />
		<video file="002_DorlandKroonenberg_02" />
		<video file="003_DorlandKroonenberg_03">
			<question>Van Dorland advocates mitigation</question>
			<rightanswer>true</rightanswer>
		</video>
		<video file="004_DorlandKroonenberg_04">
			<question>Kroonenberg says we should adapt to climate change</question>
			<rightanswer>true</rightanswer>
		</video>
		<video file="005_DorlandKroonenberg_05" repeat="true">
			<branchquestion>Are climate measures rubbish, or are they really
needed?</branchquestion>
			<overlays>
				<overlay time="0" position="01" size="12"
label="AntiClimateMeasures" position2="31" size2="22" label2="ProClimateMeasures" />
			</overlays>
		</video>
	</videos>
</subject>
```

All videos in this subject are automatically played sequentially. During two of the videos, short questions are asked (the right answers to both questions is *true*). The last video in this subject uses a branch question, and is repeated until the user makes a choice (***repeat="true"***).

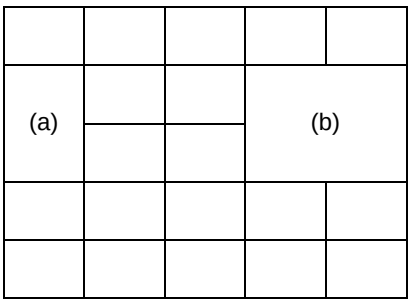


Figure 16: Overlays in the second subject

There is a choice between two options, like indicated in Figure 16. Because this is the first subject of the video, the viewer should get an overview of the subject, so there were no other overlays yet.

The overlays refer to the subjects *AntiClimateMeasures* and *ProClimateMeasures*.

Subject 3a: No climate measures

Overlay (a) leads to a branch with the effects of a negative viewpoint on climate measures. The user gets 3000 points for clicking on this link (represented by a *3* in XML).

```
<subject name="AntiClimateMeasures">
	<longname>No climate measures!</longname>
	<score>3</score>
	<videos>
		<video file="015_AntiClimateMeasures_01">
			<question>The Watersnoodramp happened in 1952</question>
			<rightanswer>false</rightanswer>
		</video>
		<video file="016_AntiClimateMeasures_02">
			<overlays>
				<overlay time="0" position="11" size="32" label="Animals" />
				<overlay time="2" position="02" size="52" label="Animals" />
				<overlay time="3.8" />
			</overlays>
		</video>
		<video file="017_AntiClimateMeasures_03" repeat="true">
		(...)
```

This subject has one clip question, and a 'hidden' hotspot. The viewer can click on this overlay, but this isn't obligatory; If the user clicks, (s)he is redirected to the Animals subject; otherwise, the next video is played automatically. The overlay is being moved during the video, because the position of the animals that form the link changes during the video. By altering the parameters, its position and size are changed. The last specification of the overlay time causes the overlay to disappear from the screen.

Subject 3b: Climate measures are good

If the user clicked on overlay (b) in the previous subject, (s)he sees clips about the positive value of climate measures.

```
<subject name="ProClimateMeasures">
	<longname>Climate measures are good!</longname>
	<score>4</score>
	<videos>
		<video file="008_ProClimateMeasures_01">
			<question>This is a protest against climate measures</question>
			<rightanswer>false</rightanswer>
		</video>
		<video file="009_ProClimateMeasures_02" />
		<video file="010_ProClimateMeasures_03" repeat="true">
		(...)
```

This subject, like 3a, has an associated score, that is awarded when the user clicks the subject (in this case with a value of '4', which is converted to 4000 in the game).

Ultimately, this subject also leads to a new branch (which is not included here).

7. Narratives in a game context

The narrative is an important aspect of interactive video. This chapter attempts to provide a theoretical basis for the use of (interactive) narratives in games, mainly focused on interactive video and games.

7.1. Narrative form

Narrative has been called a fundamental way that humans make sense of the world (Bordwell & Thompson, 2001). We read, learn, tell and even dream stories. So narratives are an important aspect of our lives.

Definition

Narrative, in film theory, is defined as:

> A chain of events in cause-effect relationship occurring in time and space (Bordwell & Thompson, 2001).

To make sense of a narrative, we identify its events and link them by cause and effect, time and space. We can also infer things that are not explicitly defined. In order to do these things, we can make a distinction of a story and a plot. The story contains both the inexplicit and explicit events, while the plot only includes the explicitly presented events, but also the nondiegetic[54] material (i.e. not part of the story world, like credits).

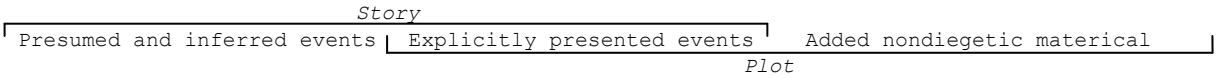


Figure 17: Story and plot visualisation (Bordwell & Thompson, 2001)

Cause-effect, time and space

The usual agents of cause and effect are the characters in a story, but also non-human agents can be used, for example the shark in the movie *Jaws* or the tornado in *Twister*. Spectators search for cause and effect:

> In general, the spectator actively seeks to connect events by means of cause and effect. Given an incident, we tend to hypothesize what might have caused it, or what it might in turn cause. (Bordwell & Thompson, 2001)

Another important aspect of the definition is the time. Important aspects are *temporal order, temporal duration and temporal frequency.* Many films present events out of chronological order. We are (usually) not confused, because the events are mentally reordered by ourselves - from the plot order we derive the story order. The temporal duration is the duration of certain stretches in the story. On the one hand, there are the story and plot duration, and on the other hand, the screen duration (the time to watch a movie). Some elements of a story can be repeated multiple times, e.g. to emphasize them or to view them from a different angle (the temporal frequency).

Finally, the space in narratives often plays an important role. It can either be visible in the video itself, but also in a different way, for example a person telling a story about his experiences (and our imagination fills in the locales that are never shown).

[54] Diegetic = everything that exists in the story world

A broader context

Bordwell & Thompson's definition is mainly focused on traditional forms of narrative. These days, it is possible to construct new forms of interactive narratives, which are usable in games.

As Pennefather observes, there is a difference between 'participative' and 'contemplative' interactivity. Participative interactivity is used in games, while contemplative interactivity is used in traditional cinema (Pennefather, 1994).

Weinbren also indicates that traditional cinema is very much different from interactive cinema:

> Much of cinema's power over us is our lack of power over it… It could be argued that the introduction of viewer impact on the representation is a destructive step for the cinema (…) To find interactive forms in which desire can be sustained will require the construction of a new cinematic grammar. (Weinbren, 1995, pp.19-20)

While maybe not a real 'destructive' step for the cinema, the introduction of viewer impact, be it in a game or interactive video, has a big impact on narratives

Narrative in gaming

In video games, different models can be used to describe the narrative. Majewski proposes three basic models of predesigned video game narrative:

- The *string of pearls* model, where the player essentially moves from one pre-designed event to the next, with a greater amount of freedom of action between the events.
- The *branching narrative* approach, where the player is occasionally able to affect the narrative by choosing from pre-designed narrative paths.
- The *amusement park* approach, where the player is located in a world with many possible narrative plots to tackle.

The most common approach might be the 'string of pearls' model. The branching narrative approach is somewhat less common. A problem with true branching narrative is that it requires more (cutscene) material, thus costs more.

> Wing Commander IV being the extreme example, at a cost of over US$12 million, and a 480 page script – of which, in a single game, the player would only see a certain portion. (Majewski, 2004)

Finally, the amusement park approach is often tied to the RPG-form, where the emphasis is on the evolution of the character's abilities. There can be a central plot, but because of the evolutionary approach a large world with many possibilities is needed. (Majewski, 2004)

Interactive video and narratives

If we would situate the pilot application's interactive video in the video game narrative model, it should be part of the branching narrative approach. After all, it contains different branches, that can be affected by the viewer using pre-designed narrative paths.

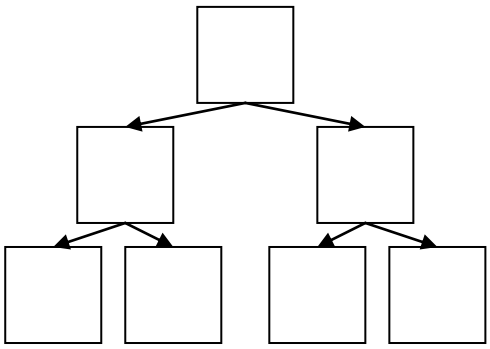


Figure 18: Branching paths

Still, the use of branching narratives can cause some problems in narratives. Aristotle once described a plot in his *Poetics* as "the arrangement of incidents", that should follow plausibly after each other. In a branching structure, it is possible that the viewer does not see every video, and it transitions between scenes could not be as logical as in a linear story (not ‘plausible’).

As Vardi observes, the classic description of plots and narratives might not be fully usable anymore:

> The classic description of plot (introduced by Aristotle in the Poetics) states that there is a determinate structure to narrative. The *dramatic arc* for instance, is a model that represents structure in which tension is built and released over time. On the other hand, interaction is generally constructed as the freedom to do anything at anytime without an implied direction. This distinction creates a potential for a conflict between narrative and interaction. (Vardi, 1999)

A way to circumvent this problem would be to let the branching paths in an interactive story converge at certain points, so some videos will always be seen (this could be scenes that are vital to building the story). So the interaction is somewhat more limited - a bit more akin to the ‘string of pearls’ model; with a greater amount of freedom between (obligatory) events.

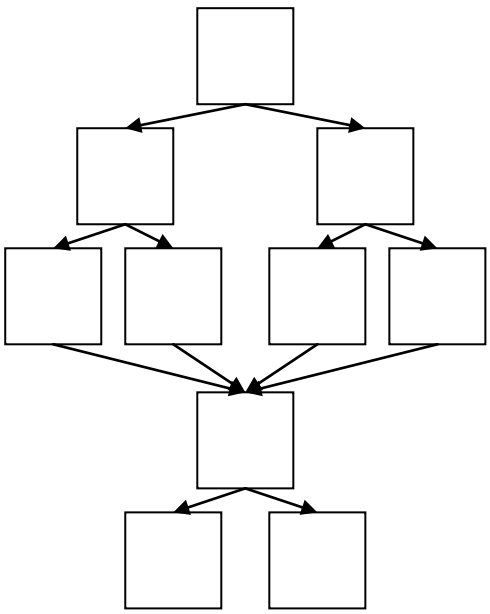


Figure 19: Branching example with converging paths

When building an interactive video (or game) that contains branching paths, it steadily grows more complex. The example above only contains 10 nodes, but it is already a lot of work to ensure that the narrative continues fluently, and doesn't stop at a 'dead end', as Sawhney indicates:

> Another concern is the possibility of "dead ends" to the continuously playing video narrative (what Terry Harpold considers the "moment of the non-narrative"). One of the aesthetic goals in HyperCafe[55] was to never permit a moment where the video would stop and break the cinematic experience of the user. Yet as the nodal structure of the videotext grows more complex, the video authors must painstakingly ensure that all sequences lead to other sequences, and thus all nodes are non-terminating. (Sawhney, 2003)

A (partial) solution to this problem is to use multiple sequences, that are played sequentially, without link opportunities:

> As the author defines the sequence of scenes in the narrative and their structured relationship to each other, he designates some of them as link opportunities for the user. (In general, the number of link opportunities will be substantially smaller than the number of links connecting scenes, as most scenes are fairly brief and are simply linked to the next scene of a narrative for continuity. (Sawhney, 2003)

This is also used in the pilot for Clima Futura and visualized in section 6.4. Every subject links to one or more other subjects (except the end nodes), but the subjects mostly contain multiple videos that are automatically played sequentially. This eases the building of large-scale narratives.
In addition, to prevent looking at a static image when a branch point is reached, the clips are repeated until the user makes a choice (also a way to prevent 'moments of non-narratives').

7.2. Rhetorical form

Interactive video, especially in the case of Clima Futura, is not only about narratives. In video, often opinions and arguments are expressed. If you want to convey a particular message or reach a certain goal, which we can loosely describe as 'creating climate awareness' in the case of Clima Futura, it is needed to consider the rhetoric form.

Basics

Documentaries form a basis for this section, because the goal of many documentaries is to persuade the audience to adopt an opinion about a subject (and this could also be valuable for Clima Futura).

Bordwell (2001, pp. 122) describes the four basic attributes of rhetorical form:

- It addresses the audience openly, trying to change its conviction, attitude or to take action
- Often, the subject of the documentary is not one of scientific truth, but of a matter of opinion
- If the conclusion cannot be proved beyond question, the maker often appeals to our emotions
- Often arguments are not presented as arguments, but as observations or factual conclusions

Argument types

To persuade the audience, different arguments can be used. These can be *arguments from source*, *subject-centered arguments and viewer-centered arguments* (Bordwell, 2001). Arguments from source try to convince the viewer that the film is a reliable source of information (for example by using a narrator with an authorative voice). Subject-centered arguments are about the film's subject matter, and often use common beliefs, example and logic. Viewer-centered arguments are arguments that appeal to the viewer's emotions.

[55] Note: HyperCafe is a hypervideo application which places the user in a virtual café, with multiple (fictional) conversations

As Bocconi (2006, pp. 13) states, these argument types can be traced back to Aristotle's categories of persuasion (in his book Rhetoric). Therefore we can distinguish:

- *Ethos*: appeal to the speaker's reputation
- *Logos*: appeal to logic or reason
- *Pathos*: appeal to the emotions of the audience

Rabiger (1998) also discusses different ways a documentary maker can behave. A *propagandist* shows only the evidence supporting predetermined conclusions. A *binary communicator* gives equal coverage to both sides in a controversy. Finally, a *mind-opener* aims at expanding the viewers mind, without manipulating, and presenting a subject in all its complexity.

In my pilot for Clima Futura, I tried to work as a *binary communicator*, and provide multiple views to the rather complicated subject of climate change. Also, the use of multiple views provides a form of dramatic conflict:

> A dramatic conflict is a structural tension that keeps the outcome (..) somewhat in doubt, and keeps the audience interested (Hampe, 1997).

A major difference between documentaries and interactive video is, of course, that in interactive video the viewer can choose the opinion that (s)he wants, and influence the story. However, in the pilot application, this choice cannot be made before opposing views of a subject have been watched.

Interactive video rhetoric issues

The fact that the user can choose his or her own paths in interactive video creates some issues, like: do you want the viewer to see all arguments, or only the line of thought (s)he chooses?

An example: test users of the pilot for Clima Futura showed that almost everyone chose the path in the video that advocates climate measures (whether this is a typical case of 'political correctness' or a real incentive is unclear). When they choose that path, they miss the branch that involves some of the trade-offs you need to make when applying climate measures (that is also briefly described in section 5.2.5), which could alter their opinion.

In the current interactive video application, multiple views on a subject are represented. Then the user makes a choice by clicking on the person (s)he agrees with. After a choice, there are multiple ways to continue with the video narrative. You can only provide videos that are corresponding to the user's choice or, to the contrary, you can show videos that are at odds with the choice. If you want the viewer to have a full understanding of the subject, you can even show both. This depends on what you want to achieve; so the author of an interactive video can also be a propagandist, binary communicator and mind-opener. So when designing the narratives, it is also important to consider the rhetoric.

While there is a lot of literature available for the rhetoric of documentaries, the rhetoric structure of interactive video still needs a lot of research. Serious games, like *Peacemaker*, already create awareness. Interactive video could even increase this awareness, by using appropriate rhetoric.

8. Development & reengineering of the Clima Futura pilot

In this chapter, the development history of the first interactive video project and the Clima Futura pilot is discussed. At the later stages of the development process of the Clima Futura pilot, it became apparent that Clima Futura will make use of the Flex framework. The pilot application has been programmed in Flash, however. To make the interactive video application a useful addition to the architecture of Clima Futura, it had to be reengineered with Adobe Flex.

8.1. Introduction

Before getting into details, the basic technical properties of the first Flash Video project, the Clima Futura pilot and the reengineered pilot are specified in table 6:

	Flash Video Project	*Clima Futura pilot*	*Reengineered pilot*
Development platform:	Flash 8 Professional	Flash 8 Professional	Adobe Flex 2
Scripting language:	ActionScript 2.0	ActionScript 2.0	ActionScript 3.0
Overlay grid:	3x3	5x5	5x5
Overlay shape:	Invisible	Oval	Square
Overlay display:	Embedded cue points	XML cue points	XML and timer
Skins / styles:	No	No	Yes

Table 6: Technical properties of the interactive video applications

8.2. Flash Video Project – the Great Barrier Reef

The first foundations for the interactive video pilot were laid out in the Flash Video Project[56] in 2006. This project was dedicated to proving that interactive video using Flash 8 was possible and feasible. It was done as a substitute project for the Multimedia Authoring II course, a third-year Bachelor course for the Multimedia & Culture study.[57]

The Great Barrier Reef was chosen as the subject of the project. This is the world's largest coral reef system, located near Australia. A Flash-application was made that allows you to view a documentary about the Great Barrier Reef in a non-linear way. If you click on certain elements in the video (for example a shark), you can view the clips of the documentary about that subject, accompanied by extra information. To achieve this, short sequences from the original documentary were used, which were grouped by subject. After seeing all video segments of a certain subject, you can return to the main movie again.

Figure 20: Flash Video Project screenshots

The sound was provided by additional MP3-files, that substituted the normal audio. This was done to glue the subjects together, and to provide suspense. Another reason is that if you view a documentary in a non-linear way, the audio commentary is mixed up (not providing a comprehensible story anymore).

[56] http://www.few.vu.nl/~flash-video

[57] Note: Wouter was focusing mostly on the design and documentation aspects, I did more of the coding and video editing.

To facilitate easier annotation, XML-files were used to specify the movie-files, information, photos and links. Cue points to generated (clickable) overlays, embedded in the Flash Video files. To make the annotation less difficult, a 3x3 grid was used for the overlays. Overlays could be 1-3 units wide and high, so the grid provided some flexibility. There was no visual indication of clickable objects (except for the mouse cursor that changed into a 'hand').

Application structure

Figure 21 shows the application's structure, which involved five XML-files, embedded cue points and MP3-audio.

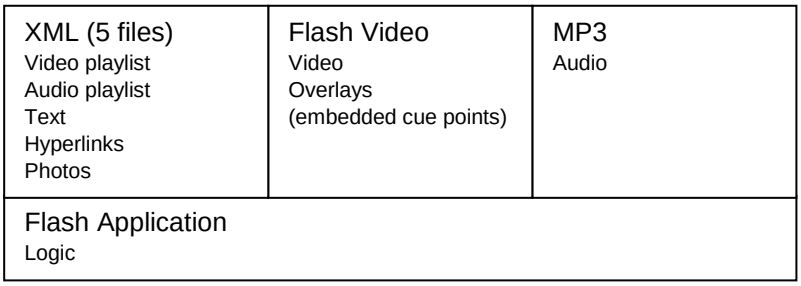


Figure 21: Flash Video Project – Great Barrier Reef structure

Issues

The final application had some limitations. The resolution of the 'grid' for the overlays was pretty low, and only one link could appear in a clip at once. The lack of normal audio also meant that you had less (auditory) information. Cue points were embedded in the movie files themselves. This was not without its problems, because if you wanted to change the annotation, you needed to re-encode the file. Furthermore, there were some small bugs and delays when loading a new set of videos (i.e. subject). Correcting these issues was not feasible for the Multimedia Authoring project.

8.3. Flash Video Project – Clima Futura

In February 2007 there was a possibility to join the Clima Futura project (see section 3.3). A quick version of the video application that allowed branching to more than one subject was made in the first month. In the next part of the project my task was to work on a (non-interactive) trailer for Clima Futura. After the presentation in front of the jury (in June 2007) the interactive video application was developed further, until the end of October 2007.

The main goals were to remove the limitations that the previous version of the application had, and more general, to prove that interactive video is a valuable combination with (serious) gaming.

Functionality

The starting point was the basic application that was made for the Multimedia Authoring project. Gradually new features were added to it, like multiple video overlays instead of one. Furthermore resolution of the overlay grid was increased (there are now 5x5 positions for an overlay). A history feature was added, so you can return to all previous videos you have played.

A challenge of this project was to combine interactive video with gaming elements. Therefore a score system was added. Another new element was the distinction between visible and hidden hotspots, so the player needs to pay extra attention. Also, extra questions during the video were asked to keep the viewer occupied (providing bonus points). To provide an extra challenge to experienced players, difficulty settings were possible (also influencing the score).

One of the problems of the first interactive video application was that there were loading times. To solve this, some performance tweaks were carried out (in the source code). Also, experiments with transitions were done. Transitions between subjects were used, to 'hide' the loading times. First, this was in the form of a simple fade. Later, by using photos, and ultimately, by using very short and small transition videos. The latter proved to be most effective.

Visuals

The first versions of the application had few interface elements, only a pause and next button. Hotspots were indicated by graphic overlays, and a small description was located above it. This proved to be too subtle for newcomers to interactive video. Subsequently, interface elements received predefined locations. Above and below the video, the current subject, chosen subject, information about the choice and the number of choices possible were indicated (mostly using textual information). The navigation controls were given a more prominent spot, in the middle below the video, and they now included a previous button and a progress bar for the video.

While the layout was changed, most of the interface elements remained in the final version of the application; At branch points, having a question (with a large font) above the video seemed more effective than below the video, and the answer to that question is indicated when you move over a hotspot.

Some test users complained about the visibility of overlays. Sometimes, especially with bright video clips, they were very difficult to see. Because of that, a transparent (black) overlay was added at obligatory choice points.
To improve the flow between the videos, which could be from different sources, transitions were added. This time not to hide loading times, as there are none in the playlist of a subject, but to improve the flow between different (short) clips.

Figure 22: Development of the application and interface

XML

The Great Barrier Reef project already made use of XML-files to store information for the application. An inheritance from that project was that the information was still scattered over multiple XML-files (one for the video and one for the subjects). Towards the end of the project, the fragmented information in different XML-files was combined into one file. Reading XML-files was somewhat difficult in Flash version 8, but a separate class proved helpful (see appendix B), and it replaced the old method of reading XML-files.

The limitation of having cue points embedded in the movie files was still creating some problems. Because of that, the cue point information was moved to the XML-file (though the application itself is still compatible to embedded cue points). Changing the program code was necessary to generate (ActionScript-based) cue points automatically.

Application structure

Compared to the structure of the previous application in Figure 21, the new version has only one XML-file (which also contains the cue points), and audio is located in the Flash Video files (instead of separate MP3-audio).

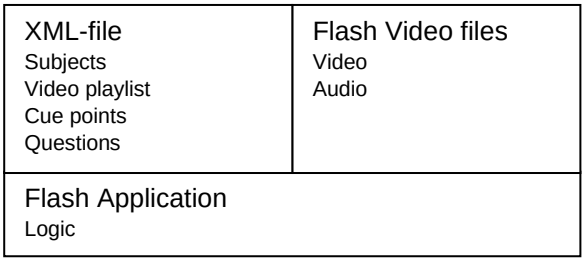


Figure 23: Interactive video pilot structure

Issues

The original prototype was made in Flash 8 Professional, this meant that it used ActionScript 2.0. During the course of the project, Macromedia released ActionScript 3.0. This version involved a lot of changes, and the current application could not be converted easily.
Also, it became obvious that the target technology for Clima Futura would be Adobe Flex 2. So a plan was made to reengineer the Flash-application to Adobe Flex and ActionScript 3.0.

8.4. Reengineering the Clima Futura pilot

When the reengineering of the Flash application to a Flex application was started, the goal was to make a new version of the interactive video application, that had cleaner source code and was implemented in Flex and ActionScript 3.0. Furthermore, it should reuse the XML-based annotation structure of the first interactive video application.

8.4.1. Technology

Adobe Flex 2

Adobe Flex 2 is a development solution for creating *Rich Internet Applications* (RIA). These are described by Adobe as:

> a new class of applications that combines the responsiveness and interactivity of desktop applications with the broad reach and ease of distribution of the web[58]

This already indicates that Flex is more ambitious than Flash, it aims more at business applications.

The Flex architecture contains the Flex Software Development Kit (SDK), an Eclipse-based development tool (Flex builder), data integration services (Flex Data Services) and a charting component. The Software Development Kit can be obtained free of charge, to compile Flex programs. The Flex Builder, the development tool that also allows easier visual editing and styling of an application, has to be purchased separately.

The output of a Flex project is a Flash-compatible SWF-file, so web applications developed in Flex 2 can be used on all computers with the Flash plug-in.

[58] http://www.adobe.com/products/flex/whitepapers/pdfs/flex2wp_technicaloverview.pdf

Differences between Flash and Flex

As has briefly been described in the terminology-chapter, Adobe Flash and Flex have some key differences.

Flash is more aimed at the development of interactive content, while Flex is better suited to application development. The source files of a Flash-project are stored in the (proprietary) FLA-format, only editable with the Flash authoring tool. Flex uses MXML-files (the markup language of Flex) and ActionScript-files. These are plain text files, so they can be edited with virtually every text editor.[59]

Adobe Flash was originally aimed at animation, and therefore contains elements like the stage, a timeline and frames. It is easier to make large applications in Flex, because the Flex coding environment uses more common coding and debugging standards (elements like the timeline are not used anymore).

Differences between ActionScript 2.0 and 3.0

Adobe Flex and the latest version of Flash use ActionScript 3. It uses a new 'Virtual Machine'. The changes in ActionScript 3 affect the performance, but also the robustness of the programming model. The performance has increased considerably, as some benchmarks indicate.[60] [61]

Major changes in ActionScript 3 are related to the strictness, syntax, data types, events and XML.[62]
The strictness refers to for example data types, which must now be specified. The syntax changes influence many built in functions (e.g. *object._height* is now *object.height*). There are new data types, providing more flexibility when working with numbers and graphics. Furthermore, the event-system has been changed (all events are now handled by the *EventDispatcher* class). Finally, the handling of XML is much more intuitive, making the use of separate classes to read XML redundant. More specific information regarding the changes can be found in Appendix E.

Porting the application versus starting from scratch

Before the process of reengineering started, two options were considered. The application could be ported to Flex, which means the existing code base was converted to both Flex and ActionScript 3. A different option was to start from scratch, and develop the application from the ground up again. Ultimately, the second option was chosen.

The reasons to start from scratch, as opposed to converting the current application to the Flex / ActionScript 3 environment, were:

- The large amount of code of the Flash-based interactive video prototype; which made it very difficult to maintain at the end of the project (adding features sometimes resulted in unexpected problems).
- Parts of the code were already used in multiple projects (the Flash Video Project and Clima Futura pilot project). This created redundant functions.
- In the previous section, the changes in ActionScript 3 were described. If the application was ported, the whole code needed to be checked.
- Also described in the previous section were the changes of Flex compared to Flash. Despite resulting in the same SWF-files, it is a very different platform.

[59] http://www.adobe.com/newsletters/edge/may2006/section2.html
[60] http://www.rockonflash.com/blog/?p=33
[61] http://blogs.adobe.com/aharui/2007/10/actionscript_readwrite_perform_1.html
[62] http://www.adobe.com/devnet/actionscript/articles/actionscript_tips.html

8.4.2. Results

After a few weeks of work, the first version of the application in Flex was ready. The reimplementation was done based on the program descriptions and XML structure, as included in this thesis, and the Flash prototype. The application contains all major features of the Flash application. The resulting application uses less resources, and is much faster than the original Flash prototype.

Table 7 shortly illustrates the differences, which will further be illustrated below.

	Interactive video pilot	*Reengineered interactive video pilot*
Technology	Flash (ActionScript 2.0)	Flex (ActionScript 3.0)
Lines of code	Approx. 800	Approx. 1150
Overlay display technique	Using Cue points	Using timer
Overlay shapes	Oval	Square
Loading times	Yes (~0.5 seconds)	No
Transition movies	Yes	No
Needs commercial authoring tool	Yes	No
Transition effects	Yes	No
Video smoothing	Yes	No
Different skins / styles usable	No	Yes

Table 7: Differences between the Flash and Flex interactive video pilot application

Functionality

As the existing Clima Futura pilot application was used as the basis for the development of the reengineered version, the functionality largely overlaps. It still plays videos and contains branch points, clip questions and hidden hotspots.

The overlay displaying system now uses the *Timer* class to show and alter overlays. In the Flash-version this was done via Cue points generated by ActionScript. Because it caused some problems in Flex, it was changed; but it has no influence on functionality.

Some smaller features were not yet implemented, because it was not clear if they would be usable in combination with the full Clima Futura game (for example the difficulty setting).

Visuals

Due to the different nature of Flex (compared to Flash), some visual elements were implemented slightly different. Since Flex is less aimed at the graphical side of developing web applications, it is more difficult to use for example animations. The overlays to indicate hotspot are now made of squares. This is due to a different system of visualizing the overlays, which are now made out of squares and were easier to use in Flex than the ovals in the Flex-version. Also, the transition effects between clips in a subject are not yet used, which were animated using Flash in the original version.

The interactive video pilot used transition movies to hide the fact that there was a short loading time between different subjects (though there were no loading times between clips *in* a subject). In the Flex-version, the increased speed of the application makes the use of transition movies redundant, since there are (almost) no loading times when loading a new subject.

A disadvantage of the Flash interactive video application was that it had a predefined graphical style. The interactive video narratives could be altered using XML, but not the style. An advantage of Flex is that allows the use of Cascaded Style Sheets (CSS) to define the application's style. On top of that, it is possible to create your own skins that can also be applied to it. So by using styles and skins you can externally define the look of your application. The Flex interactive video application makes use of styles and skins, and Figure 24 reveals the resulting looks of the application with different skins.

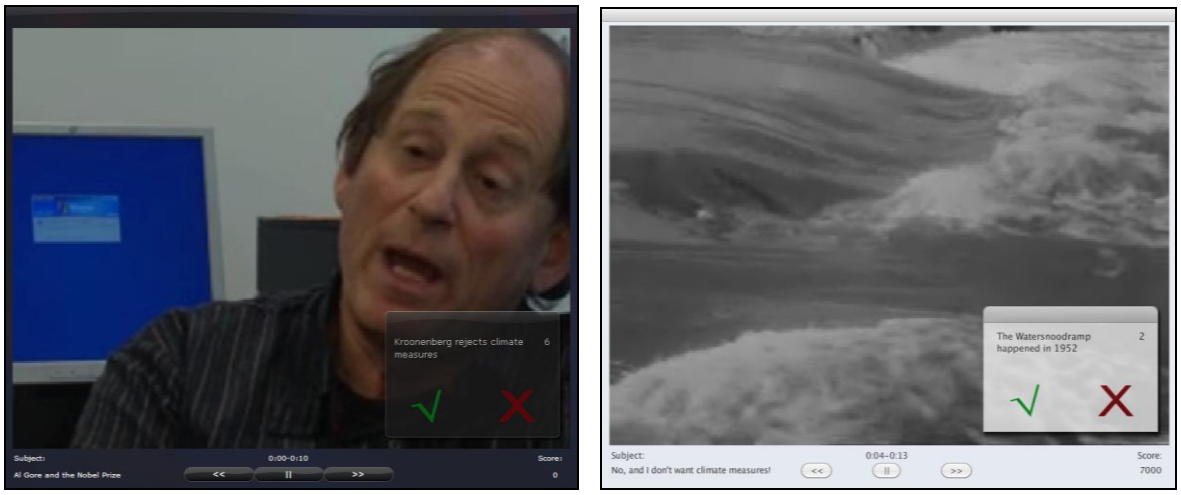


Figure 24: Flex video application using the *Obsidian* and *OSX* skin [63]

XML

The resulting Flex application makes use of the existing XML-structure for interactive video, as described in chapter 6. The Flex interactive video application reads the existing XML-file with videos and annotations, and makes use of the existing videos.
An extra settings file (also based on XML) is used to define the locations of the videos and the video annotation.

[63] Available at: http://www.scalenine.com

9. Conclusion

In this thesis, the basics of interactive video, (serious) games, technology, XML, narratives and rhetoric were discussed.

The third chapter started with the backgrounds of the subject, and indicated that user choices should influence future choices in interactive video. The history of interactive video in a gaming context was described, and the lessons that can be learned from it. Serious games were discussed, with a few examples. An important example of a serious game is Clima Futura, about climate issues. To take a first look on Adobe Flash's features, some examples of current interactive video applications closed the chapter.

The fourth chapter indicated the possibilities and limitations of Flash. The contents of a basic interactive video application were discussed and visualized. To apply this theory to practice, a pilot application was built.

The pilot application's features were described in chapter 5. It also pointed out the importance of video editing. In order to both acquire and use existing video material in a narrative, it is needed to use certain techniques.

As the video narrative should be editable, a proper data structure should be used. A prime example of a versatile data structure is XML. It was utilized in the Clima Futura pilot application. The sixth chapter covered the basics of XML and a structure for interactive video, which was subsequently applied to the interactive video pilot.

After the last, for the most part technical chapters, it was time to use the theory of narratives and rhetoric to gaming and interactive video. Theory from film studies and gaming was used for the various aspects of narratives. Also the rhetoric theory was considered, which can be used to convey a particular message. It was shown that rhetoric can be valuable for gaming, and particularly interactive video.

The final chapter covered the development process of the interactive video application and the final part of the Master's project: the reengineering of the pilot application for Clima Futura. The target technology was Adobe Flex, and this chapter pointed out some differences with Flash. A key difference in the new Flex video application is the improved speed. Currently, the Clima Futura team is working on the integration of interactive video into the actual game, as a reusable component.

Finally it is time to return to the research question again: *To what extent is it possible to use interactive video in serious games*. The realized prototype shows that it is feasible to use it in a more educational way, by using existing material and introducing game elements. Incorporating existing video material as a 'poor man's immersion' is a big difference with interactive video in the past, which relied on (expensive) filmed material for one purpose. However, it is still important to learn from past lessons and to provide sufficient ways of user interaction.

10. References

Aleem T. (1998), A Taxonomy of Multimedia Interactivity

Bordwell, Thompson (2001), Film Art: An Introduction, Sixth edition

Bordwell, Thompson (1994), Film History: An Introduction, First edition

Bocconi (2006), Vox Populi

Cesar P., Bulterman D., Jansen A. (2006), An Architecture for End-User TV Content Enrichment
Retrieved from: http://www.jvrb.org/3.2006/euroitv2006/759/

Chambel T., Guimaraes N. (2002), Context perception in video based hypermedia spaces

Cook P., Bernink M. (1999), The Cinema Book, British Film Institute, 1999, bfi publishing, isbn 0-85170-729-7

Eliëns A., van de Watering M, Huurdeman H., Bhikharie W. (2007), Clima Futura @ VU, communicating (unconvenient) science
Retrieved from: http://www.cs.vu.nl/~eliens/research/cv/research/projects/media/paper-climate.pdf

Eliëns A., Chang T. (2007), Let's be serious – ICT is not a (simple) game
Retrieved from: http://www.cs.vu.nl/~eliens/web3d/media/paper-serious.pdf

Eliëns A., Feldberg F., Konijn E., Compter E. (2007), VU @ Second Life – Creating a (virtual) community of learners
Retrieved from: http://www.cs.vu.nl/~eliens/media/paper-secondlife.pdf

Gee J. (2003), What video games have to teach us about learning and literacy

Halasz G., Schwartz M., The Dexter Hypertext Reference Model
Retrieved from: http://ei.cs.vt.edu/~mm/pdf/dexter.pdf

Hampe (1997), Making documentary films and reality videos: a practical guide to planning, filming, and editing documentaries of real events. Henry Holt and Company, 1997

Hardman L., Bulterman D., van Rossum G. (1994), The Amsterdam Hypermedia model

Nelson (1965), Complex information processing: a file structure for the complex, the changing and the indeterminate

Majewski (2004), Theorising Video Game Narrative
Retrieved from: http://www.home.umk.pl/~majewski/jakub/mt_cont.html

Nieborg D. (2004), America's Army: More than a game?
Retrieved from: http://www.gamespace.nl/content/ISAGA_Nieborg.PDF

Pennefather J. (1994), From cinema to virtual reality, Intermedia 22(5), October-November 1994

Rabiger M. (1998) Directing the Documentary. Focal Press, 1998.

Sawhney N, Balcom D., Smith I. (1996), HyperCafe: Narrative and Aesthetic properties of Hypervideo

Shipman F., Girgensohn A., Wilcox L. (2003), Generation of interactive multi-level video summaries

Stenzler M., Eckert R. (1996), Interactive video
Available at http://sigchi.org/bulletin/1996.2/eckert.html

Tannenbaum R.S. (1998), Theoretical Foundations of Multimedia, Computer Science Press, 1st Edition, New York (NY), isbn 0-7167-8321-5.

Vardi (1999), Navigation scheme for interactive movies with lineair narrative

Vorderer P., Bryant J. (2006), Playing computer games - Motives, responses, and consequences

Weinbren G. (1995), In the Ocean of Streams of Story, Millenium Film Journal No. 28 (Spring 1995): Interactivities
Retrieved from: http://www.mfj-online.org/journalPages/MFJ28/GWOCEAN.HTML

Appendix A

CLIMA FUTURA @ VU*– COMMUNICATING (UNCONVENIENT) SCIENCE

Anton Eliëns, Marek van de Watering, Hugo Huurdeman, Winoe Bhikharie
Intelligent Multimedia Group, VU University Amsterdam
eliens@cs.vu.nl rvdwate@few.vu.nl hchuurde@few.vu.nl svbhikha@few.vu.nl

Haroen Lemmers, Pier Vellinga†
Climate Centre, VU University, Amsterdam
lemh@geo.vu.nl p.vellinga@falw.vu.nl

KEYWORDS
climate change, science communication, model-based simulation, serious games

ABSTRACT

In this paper we introduce Clima Futura, a game about climate change. The primary aim of Clima Futura is to gain experience with the parameters affecting climate change and to give access to climate change related research in a playful manner. The concept for the game has been developed as a submission for the yearly Dutch contest for the communication of science. In this paper we will give an overview of the scientific background of the game, the overall design of the game, and our approach for realizing the game, deploying a modular architecture which allows for extending the game with minigames contributed by the community of players.

INTRODUCTION

Over the last couple of years, climate change has come into the focus of public attention. Moved by television images of dislocated people in far-away countries, ice bears threatened by the corruption of their native environment, tsunami waves flooding the third world, and hurricanes destroying urban areas, the general public is becoming worried by what Al Gore has so aptly characterized as *an inconvenient truth*: the climate is changing and human affluence may be the prime cause. In response to the *pathos* of the media, many civil groups do an appeal on the responsibility of individual citizens and start campains for an *ethos* of climate-correct behavior, by saving on energy-consumption or driving CO2-friendly cars. In the media, such campains are either advocated or critized by authorities from public government, and experts from a multitude of sciences, with conflicting opinions. As a result, the general audience, initially with genuine concern about the state of our world, gets confused and looses interest. And more worrisome, the adolescents, looking at the serious way adults express their confusion and ignorance, take distance and may decide that the *climate issue* is not of their concern.

At the Climate Centre of the VU University Amsterdam, we are not happy to observe that *pathos* and *ethos* overtake the public debate, and we actively wish to participate in the public debate bringing our multi-disciplinary scientific background into play. Moreover, since we *borrow the earth from our children*, as the old Indian saying goes, which Al Gore again brought to our attention, we feel that we must take an active interest in bringing the *climate issue* to the attention of the youth, in a form that is appropriate. From this background, we engaged in developing Clima Futura, a multi-disciplinary undertaking, bringing together climate experts from a variety of backgrounds with multimedia/game development researchers. The Clima Futura game addresses the issues of climate change, not altogether without *pathos* nor *ethos*, but nevertheless primarily focussed on bringing the *logos* of climate change into the foreground, in other words the scientific issues that are at play, and the science-based insights and uncertainties that may govern our decisions in the political debate. Given the state of our knowledge, the science of climate change itself may be characterized as a somewhat unconvenient science, and as such an interesting challenge to present by means of a game.

structure The structure of this paper is as follows. First, we will briefly discuss general issues of game design. Then we will describe the context, the science communication contest, the process of developing the concept for the game, and the actual design of Clima Futura. Before discussing the overall architecture of the game, we will characterize our *game event description format*, developed to allow for collaborative design, involving participants from a wide variety of disciplines. We will then outline the technical properties of our proposed game architecture, which accomodates extensions for special interest groups as well as contributions from the community of players.

*www.climafutura.nl
†also at University Wageningen

GAME PLAY, SIMULATION AND EXPLORATION

Games are increasingly becoming a vital instrument in achieving educational goals, ranging from language learning games, to games for learning ICT service management skills, based on actual business process simulations, Eliens & Chang (2007). In reflecting on the epistemological value of game playing, we may observe following Klabbers (2006), that the game player enters a *magic circle* akin to a complex social system, where *actors*, *rules*, and *resources* are combined in intricate (game) configurations:

game as social system

actors	**rule(s)**	**resource(s)**
players	events	game space
roles	evaluation	situation
goals	facilitator(s)	context

An often heard criticism on educational games is, unfortunately, that, despite the good intentions of the makers, they do not get the target audience involved, or put in other words, are quite boring. This criticism, as we will argue later, also holds for many of the climate games developed so far, and the question is how can we avoid this pitfall, and present the impact of climate change and the various ways we can mitigate or adapt to the potential threats of global warming in an entertaining way, that involves the player not only intellectually but also on a more emotional level? Put differently, what game elements can we offer to involve the player and still adequately represent the climate issue?

Looking at the games discussed in *Playing Games with the Climate*[1], we see primarily games that either focus on (overly simplified) climate prediction models (*logos*), or games that challenge the player how to become climate-correct (*ethos*). In our approach, we not only aim to include (well-founded) *logos* and *ethos* oriented game-playing, but also wish to promote an understanding of the *pathos* surrounding climate change, where we observe that the models taken as a reference are often gross simplifications and from a scientific perspective not adequate! To this end we will, as an extra ingredient, include interactive video as an essential element in game playing. This approach effectively combines a turn-based game-play loop, with a simulation-loop based on one or more climate reference models, with in addition exploratory cycles, activated by game events, which allow the player to explore the argumentative issues in the rethorics of climate change, facilitated by a large collection of interactive videos in combination with minigames.

[1] www.worldchanging.com/archives/003603.html

BUILDING THE TEAM

The *Academische Jaarprijs*[2] (yearly national Dutch prize for scientific communication) is a contest for bringing high-standing scientific research under the attention of the general public, including the younger generations! The VU University decided to submit their internationally well-renowned climate research[3], e.g. Kabat et al. (2005), as a candidate for the prize.

Looking for adequate means to communicate our scientific insights to the general audience, it took not long before the idea of a *game* came up. Both senior and junior staff of all relevant faculties were assembled to discuss the plan of a game, and an inventory was made of what games existed, followed by brainstorm sessions in which initial ideas were proposed.

Games we looked at included: *Planet Green*[4], offering ways to explore climate-correct behavior, the *ThinkQuest*[5] climate game, checking your knowledge for basic climate-related facts, the British *Climate Change Hero*[6] game, meant to improve the players knowledge about climate change factors, the German *Climate Simulator*[7], which allows for experimentation with climate change based on a simulation model, and the BBC game *Climate Challenge*[8], where the player must take decisions to tackle climate change and yet stay popular. But none of these games seemed to be satisfactory as a basis for our game, although each of them provided some inspiration, one way or another.

When we came accross a serious game in an altogether different domain, we nevertheless did find the inspiration we were looking for. In the ground-breaking *Peacemaker*[9] game, we found an example of how to translate a serious issue into a turn-based game, which covers both political and social issues, and with appealing visuals, not sacrificing the seriousness of the topic. By presenting real-time events using video and (short) text, Peacemaker offers a choice between the points of view of the various parties involved, as a means of creating the awareness needed for further political action. With Peacemaker as an example after which to model our climate game, we started working on the design of a turn-based game, allowing the player to manipulate parameters of climate change over a period of time, against the background of a climate simulation model, and offering the opportunity to explore climate-related issues and opinions, using interactive video or by playing minigames. Clima Futura was born!

[2] www.academischejaarprijs.nl
[3] www.climatecentre.vu.nl
[4] planetgreengame.com
[5] library.thinkquest.org/5721/climategame.html
[6] www.devon.gov.uk/index/environment/climatechange
[7] www.deutsches-museum.de/dmznt/climate/climategame
[8] www.bbc.co.uk/sn/hottopics/climatechange
[9] www.peacemakergame.com

CONCEPT – CLIMA FUTURA

The Clima Futura game is targeted at an audience in the age of 12-26. Primary goals are to create involvement with the *climate issue*, and to provide information by allowing the player to explore cause and effect relations, using models based on scientific research in a continuously evolving field of knowledge.
Clima Futura is a turn-based game, with 20 rounds spanning a 100-year period. In each turn, the player has the option to set parameters for the climate simulation model. The game is centered around the so-called *climate star*, which gives a subdivision of topics in climate research, as indicated below.

- climate strategies – (1) emission reduction, (2) adaptation
- climate systems – (3) feedback monitoring, (4) investment in research, (5) climate response
- energy and CO2 – (6) investment in efficiency, (7) investment in green technology, (8) governement rules
- regional development – (9) campain for awareness, (10) securing food and water
- adaptation measures – (11) public space, (12) water management, (13) use of natural resources
- international relations – (14) CO2 emission trade, (15) European negotiations, (16) international convenants

Of the topics mentioned, not all may immediately be represented in the simulation model underlying Clima Futura, but may only be addressed in exploratory interactive video. The *climate star* is actually used by the VU Climate centre as an organizational framework to bring together researchers from the various disciplines, and in the Clima Futura game it is in addition also used as a *toolkit* to present the options in manipulating the climate simulation model to the player.
The result parameters of the climate simulation model are for the player visible in the values for *People*, *Profit* and *Planet*, which may be characterized as:

- *People* – How is the policy judged by the people?
- *Profit* – What is the influence on the (national) economy?
- *Planet* – What are the effects for the environment?

A generally acknowledged uncertainty within climate research surrounds the notion of *climate sensitivity*, that is the extent to which the climate and climate change is actually dependent on human activity. In practice, the actual assessment of climate sensitivity may determine whether either a choice for mitigation or adaptation is more viable.
In the Clima Futura game we choose for using *climate sensitivity* as as a parameter for setting the level of difficulty of the game play, where difficulty increases with the value for climate sensitivity.

To give an example of game play, we let the player start in 2007, the year the IPCC[10] (Intergovernmental Panel on Climate Change) report was published. In each subsequent round, the player may choose to undertake action. For example, when the player decides to enforce restrictions on CO2 emissions, s/he may choose option (1) in the climate star, which can be reached through *climate strategies*. The result will then be visible, after some period of (game) time, in either one of the result parameters, *People*, *Profit*, and *Planet*.
The climate simulation model[11] underlying Clima Futura is primarily based on the Climber 2.0 model, which is used for scientific simulations of climate change, based on the division between land and sea, the density of vegetation, sea temperature, and the amount of CO2. Economic costs and benefits of climate policy options are calculated by means of an integrated assessment model coupled to the climate model. Additionally, an alternative model, the MERGE[12] model is used, which gives a flexible means to explore a wide range of contentious issues: costs of abatement, damages from climate change, valuation and discounting. MERGE contains submodels governing domestic and international economy, energy-related and non-energy related emissions of greenhouse gases, as well as market and non-market damages due to global climate change.
As an aside, the choice of models[13] is in itself a controversial scientific issue, as testified by J. D. Mahlman's article on the rethorics of climate change *science versus non-science*[14], discusssing *why climate models are imperfect and why they are crucial anyway.*

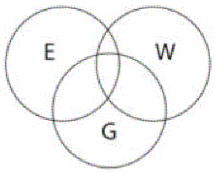


Fig 1. Game play, Simulation, Exploration

In summary, see fig. 1, the Clima Futura game combines the following elements:

1. game cycle – turns in subsequent rounds (G)
2. simulation(s) – based on climate model (W)
3. exploration – by means of interactive video (E)

Each of the three elements is essentially cyclic in nature, and may give rise to *game events*. For example, game events may arise from taking turns after 5-year periods, due to alarming situations in the climate simulation, such as danger of flooding an urban area, or accidental

[10] www.ipcc.ch
[11] en.wikipedia.org/wiki/Climate_model
[12] www.stanford.edu/group/MERGE
[13] www.grida.no/climate/ipcc_tar/wg1/308.htm
[14] www.gfdl.noaa.gov/~gth/web_page/article/aree_page1.html

access to confidential information in the exploration of video material. In addition, Clima Futura features *mini-games*, that may be selected on the occurrence of a game event, to acquire additional information, gain bonus points or just for entertainment. Examples of mini-games, are *negotiation with world leaders*, or a climate-related variant of Tetris. Clima Futura also features *advisors* that may be consulted, to gain information about any of the topics of the *climate star*.

GAME DESCRIPTION FORMS

Having decided on the general structure and elements of the Clima Futura game, a turn-based game loop, a climate-model driven simulation, exploratory video, and mini-games, the problem is how to connect these elements in a meaningfull way, and design a coherent collection of game events. This problem is further aggravated by the need to find a way to design in a collaborative fashion, necessitated by the sheer amount of disciplines and people involved.
To enable collaborative design we developed a game event description format, which standardizes the way game events are to be described, and for which we also developed an online form, structured as outlined below:

- name of event – give a meaningful name
- event-id – for administrators only
- type – (generic/specific) game/model/video
- cause – game play/simulation/exploration
- feedback/information – give a logical description
- player actions – indicate all (logical) player options
- description of visuals – for feedback, information and player options
- additional information – give a url with references to additional informatin and visuals
- relates to event(s) – give id's or descriptions of related events

Before enforcing the game event description format, our ideas about the design of Clima Futura were gathered in a collection of narratives and brief descriptions, in what we called the *Clima Futura Design Bible*. Using the standardized game event description format, we hope to arrive at a more uniform way of describing the narratives, the perspectives from which these narratives can be experienced, the challenges or problems a player must solve, the resources available to the player, such as capital, knowledge and political power, the rewards, possibly using bonus credits for succesfully playing a mini-game, as well as the visuals, which will where possible be derived from the collection of videos we have available.
For the elaboration of the design, we are developing storyboards, which characterize in a visual way the major (dramatic) elements of narratives, structured using a subdivision in:

scenario(s)

- context – general setting, situation
- problem – event(s) to occur, problem to solve
- S-R situation(s) – stimulus/response (one or more)
- climax – action must be taken
- resolution – find solution or result

Although the actual workflow that we will deploy during development is at the moment of writing not clear, we will strive for developing templates that allow for a quick realization of the designs captured by the game event and minigame description format(s), along with the storyboards for visual design.

A MODULAR ARCHITECTURE

In the beginning, we envisioned the realization of our climate game as a first-person perspective role-playing game in a 3D immersive environment as for example supported by the Half Life 2 SDK, with which we gained experience in creating a *search the hidden treasure*[15] game in a detailed 3D virtual replica of our faculty. However, we soon realized that the use of such a development platform, would require far too much work, given the complexity of our design. So, instead of totally giving up on immersion, we decided to use *flash* video[16], indeed as a poor-man's substitute for real 3D immersion, which, using *flash*[17] interactive animations, has as an additional benefit that it can be used to play games online, in a web browser. Together with the Flex 2 SDK[18], which recently became open source, *flash* offers a rich internet application (RIA) toolkit, that is sufficiently versatile for creating (online) games, that require, in relation to console games or highly realistic narrative games like Half Life, a comparatively moderate development effort. To allow for component-wise development, we choose for a modular architecture, with four basic modules and three (variants) of integration modules, as indicated below, in fig 5.

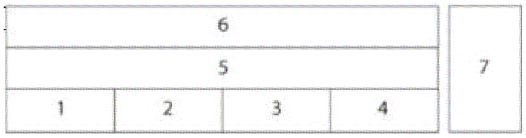


Fig 2. Clima Futura Architecture

1. climate model(s) - action script module(s)
2. game play interaction - event-handler per game event

[15] www.cs.vu.nl/∼eliens/game
[16] www.adobe.com/products/flash/video
[17] ww.adobe.com/devnet/flash
[18] www.adobe.com/products/flex/sdk

3. video content module - video fragment(s) and interaction overlays
4. minigame(s) - flash module(s) with actionscript interface
5. Clima Futura - integration of modules 1-4, plus server-side ranking
6. adapted versions – educational, commercial
7. multi-user version –with server-side support

In addition, we would like to develop a facility that allows players not only to submit their own video material, but also to build or modify their own minigames, which might then be included in the collection of minigames provided by Clima Futura.
For the actual production, we will use additional components, including *game physics*[19], a *relation browser*[20], and an *earth*[21] component. In particular, both physics and in-game building facilities seemed to have contributed to a great extent to the popularity of Second Life, Eliens et al. (2007). In creating *digital dossiers*[22] for contenporary art, we have deployed concept graphs, that is a relation browser, to give access to highly-related rich media information about art in an immersive manner. Finally, given the topic of Clima Futura, being able to visualize models of the surface of the earth seems to be more than appropriate. It is interesting to note that our technology also allows for the use of *flash* movies directly by invoking the *youtube* API[23] as a web service, which means that we could, in principle, build minigames around the evergrowing collection of *youtube*, or similar providers.
Providing flexible access to collections of video(s) to support arguments concerning controversial issues has been explored in, among others Vox Populi[24].
In Vox Populi, video fragments are annotated with meta-information to allow for searching relevant material, supporting or opposing a particular viewpoint. based on the users' preference, either a *propagandist* presentation can be chosen, expressing a single point of view (POV), a *binary commentator*, which shows arguments pro and con, or an *omniscient presenter* (mind opener), which displays all viewpoints. Although a research topic in itself, we would like to develop a *video content module* (3), that provides flexible access to the collection of video(s), and is media driven to the extent that video-material can be added later, with proper annotation. Together with in-game minigame building facilities, it would be in the spirit of a participatory culture, to provide annotation facilities to the player(s) of Clima Futura as well, to comment on the relevance and status of the video material,

[19] www.fisixengine.com
[20] http://der-mo.net/relationBrowser
[21] www.flashearth.com
[22] www.few.vu.nl/~dossier05
[23] www.youtube.com/dev
[24] homepages.cwi.nl/~media/demo/IWA/

CONCLUSIONS

To present the concept of Clima Futura, we decided to have three central presenters (anchors) and an expert-panel (choir), that may comment on detailed scientific or technical issues. The presentation, stressed the multi-disciplinary approach.
Although it too early to look back, we may on reflection ask attention for another potential pitfall, that endangers any educational game, once aptly expressed by Sartre in his criticism of *l'esprit de serieux*. Indeed, we may become too serious! In concluding our account of the design and development of Clima Futura, we may refer to an *ontology of humour*, Dormann et al. (2007), that may be taken as a guideline to avoid the common pitfall of *serious games*. In brief, Dormann et al. (2007) distinguishes between three theories of humour, that each denote a particular function of humour: *relief theory*, which explains humour as a reduction of stress, *superiority theory*, which asserts that humour has a social function, as a means to enforce the norm of a group or culture, and *incongruity theory*, which relates humour to the discovery of hidden meanings. We leave it to the imagination of the reader to establish in what way the various types of humour may be put to effect in the *climate issue*!

Acknowledgement(s) We thank all (other) members of the Clima Futura Team[25]: Frans Berkhout, Peter van Bodegom, Merlijn Draaisma, Alex Halsema, Thijs Louisse, Anne Martens, Karlien Meulenaars, Elia Orru, Frans-Jan Parmentier, Pieter Pauw, Rob Schuddeboom, Charlotte Spliethoff for their enhusiasm and effort, and Paulis Klerk (Harlequin), and Suzanne Waldau (Ex'Machina), for there suggestions and support in the realization of Clima Futura.

REFERENCES

Dormann C., Barr P. and Biddle R. (2007), Humour Theory and Videogames: Laughter in the Slaughter, In *Proc. of the 2006 ACM SIGGRAPH symposium on Videogames*

Eliens A. and Chang T. (2007), Let's be serious – ICT is not a (simple) game, In *Proc. FUBUTEC 2007*

Eliens A., Feldberg F., Konijn E., Compter E. (2007), VU @ Second Life – creating a (virtual) community of learners, In *Proc. EUROMEDIA 2007*

Kabat P., van Vierssen W., Veraart J. Vellinga P., Aerts J. (2005). Climate proofing the Netherlands, Nature 438, pp. 283-284

Klabbers J.H.G. (2006), *The Magic Circle: principles of Gaming and Simulation*, Sense Publishers

[25] www.climafutura.nl/team/

Appendix B

Classes to read XML

The Clima Futura pilot application uses a separate class, called XModel, to read the XML-files. This class eases the use of XML-files in Flash, because it automatically converts XML-data into an ActionScript object, whose properties are named after the elements and attributes of the XML.

An example:

```
<car manufacturer="Lamborgini" model="Testarossa" />
```

If you would read the XML-node via the standard Flash methods, it would be accessible via the following code: xmlObj.childNodes[0].attributes.manufacturer. With the XLEFF-class, you can just use attributes.manufacturer. With lower nodes in the hierarchy, the standard methods even get more difficult to use.

File naming protocol

When using a lot of video files, it is important to have a proper file naming scheme. In the case of the Clima Futura pilot it is the following:

[total video sequence number]_[subject]_[subject video sequence number].flv

For example:

001_Animals_01.flv
002_Animals_02.flv
005_Humans_01.flv
006_Humans_02.flv

etc.

So a filename contains a unique sequence number, the subject name and a sequence number for the subject itself. The application uses this to distinguish the different files, while it also helps to keep track of all video files that are used in a project.

Appendix C

Flash Video Encoder

(Dis)advantages of the Flash Video encoder

Many video formats can be imported directly into the Flash Video encoder, included with the authoring environment of Flash Professional. You can use AVI, Windows Media format, Quicktime, MPEG and MPEG4.

It is possible to choose a range of the video to encode. Also basic tools like crop and trim are available. So for many image formats, editing in advance is not really needed. The Flash Video Encoder has two drawbacks, however. Editing using the audio track isn't possible, because there is no audio preview. Furthermore, the editing controls have a fixed size (very small). So for very lengthy video files, it is difficult to cut out the right segment.

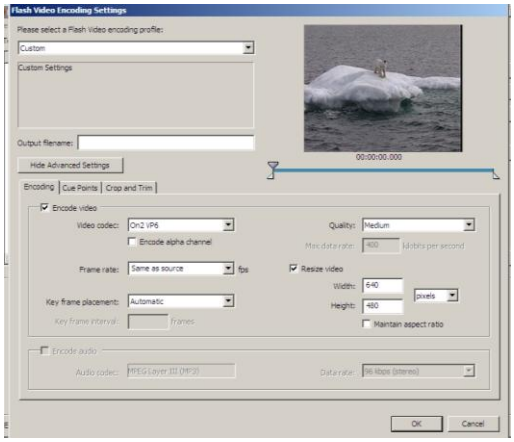


Figure 25: Flash Video encoder settings page

To overcome the drawbacks of the Flash Video Encoder, or to do advanced image compositing, other tools are needed. Which tool to choose is dependent on the image formats you want to use. For example, DVD-files need to be edited (and exported) with a DVD-compatible package. WMV-files can be edited using the Windows Movie Maker program (included with Windows XP and Vista). One of the better open source tools to edit AVI-files is VirtualDub. So there's a myriad of different tools.

To do advanced video compositing, you need an advanced video editing tool. This can be done with for example Adobe After Effects and Adobe Premiere[64].

[64] trial versions available at adobe.com

Appendix D

XML Schema for the Clima Futura pilot

This schema can be used to check the correctness of an XML-file with a narrative structure for the interactive video application.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
	<xs:element name="subjects">
		<xs:complexType>
			<xs:sequence>
				<xs:element name="subject" maxOccurs="unbounded">
					<xs:complexType>
						<xs:sequence>
							<xs:element name="longname" type="xs:string"
minOccurs="1" maxOccurs="1"/>
							<xs:element name="score" type="xs:integer"
minOccurs="1" maxOccurs="1"/>
							<xs:element name="videos" type="videostype"
minOccurs="1" maxOccurs="1"/>
						</xs:sequence>
						<xs:attribute name="name" type="xs:string"/>
					</xs:complexType>
				</xs:element>
			</xs:sequence>
		</xs:complexType>
	</xs:element>
	<xs:complexType name="videostype">
		<xs:sequence>
			<xs:element name="video" type="videotype" maxOccurs="unbounded"/>
		</xs:sequence>
	</xs:complexType>
	<xs:complexType name="videotype">
		<xs:all>
			<xs:element name="branchquestion" type="xs:string" minOccurs="0" maxOccurs="1"/>
			<xs:element name="question" type="xs:string" minOccurs="0" maxOccurs="1"/>
			<xs:element name="rightanswer" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
			<xs:element name="overlays" type="overlaystype" minOccurs="0" maxOccurs="1"/>
		</xs:all>
		<xs:attribute name="file" type="xs:string" use="required"/>
		<xs:attribute name="leadsto" type="xs:string"/><!--leadsto attr. has precedence over
repeat attr.-->
		<xs:attribute name="repeat" type="xs:boolean"/>
		<xs:attribute name="extra" type="xs:boolean"/>
	</xs:complexType>
	<xs:complexType name="overlaystype">
		<xs:sequence>
			<xs:element name="overlay" type="overlaytype" minOccurs="1"
maxOccurs="unbounded"/>
		</xs:sequence>
	</xs:complexType>
	<xs:complexType name="overlaytype">
		<xs:attribute name="time" type="xs:decimal"/>
		<xs:attribute name="position" type="xs:int"/>
		<xs:attribute name="size" type="xs:int"/>
		<xs:attribute name="label" type="xs:string"/>
		<xs:attribute name="position2" type="xs:int"/>
		<xs:attribute name="size2" type="xs:int"/>
		<xs:attribute name="label2" type="xs:string"/>
		<xs:attribute name="position3" type="xs:int"/>
		<xs:attribute name="size3" type="xs:int"/>
		<xs:attribute name="label3" type="xs:string"/>
	</xs:complexType>
</xs:schema>
```

Appendix E

Changes in ActionScript 3.0

In ActionScript 3.0, aspects regarding strictness, syntax, data types, events and XML have changed.

Strictness:

- Data Typing for variables, parameters and return types is now highly recommended; warnings will be shown if it this not done
- Null exceptions now return an error

Syntax changes:

- Native properties no longer use underscores (_height becomes height)
- Scale and alpha notations have changed (object._scale=80 has changed to object.scale=0.8)
- Some functions and classes now have different names

New data types:

- For numbers: int and uint
- For graphics: sprite and shape

Event-based architecture:

- All events in ActionScript 3.0 are now handled through the EventDispatcher class
- DisplayList instead of timeline

XML handling:

- E4X (ECMAScript for XML) is now the recommended means of manipulating XML in Flash. Earlier separate classes were necessary for easy handling of XML, see Appendix X.

Appendix F

Contents of the Clima Futura trailer

A large task for me in the first part of the Clima Futura project was the creation of a promotional trailer. This trailer was meant to announce the game and to invite our target audience (youngsters) to play the game. Like many computer games, we used a doom scenario to gain interest. The issue at stake had to be communicated clearly, using spectacular images.

The trailer starts with a descriptive part, which indicates that the world is changing. We live in luxury, but this luxury takes its toll: the climate is changing. Climate-related problems are emerging: disasters, economic threats and endangered life.

This announcement of doom is followed by an indication of political dilemmas. The VU climate star is then introduced as a means to gain scientific insight, accompanied by several images of research.

Despite science being a key to the solution of the problem, the world is divided. People in the media often communicate conflicting views on the topic (having different interests). Thus, the general audience is divided.

After the development of drama in the first part, the game itself is introduced. Viewers are asked if they can handle all the issues previously mentioned. (Provisionary) game images are shown in a rapid succession.

Two possible endings of the game indicate the influence of the player: will (s)he fail or succeed? The trailer ends with the game name, VU logo and web link.

Scene:	Sentence:	Contents:
1	It's the dawn of the 21st century The world is changing rapidly	▫ Earth animation (zoom in) ▫ Futuristic city (dawn) ▫ Power lines ▫ Power plant (polluting) ▫ Single luxurious SUV ▫ Multiple SUVs ▫ Traffic jam (polluting)
2	The climate is changing…	▫ Declining glaciers (with temperature graphic) ▫ Erosion
3	Disasters strike the earth	▫ Helicopter flying to multiple tornadoes ▫ Flooded city ▫ Melting ice
4	Life is in danger	▫ Polar bear on ice shelf ▫ People in trouble (flooding) ▫ Drowth, cow in trouble ▫ Dead cow ▫ Bones of dead animals
5	We all face the same problems	▫ Moving flags (1) ▫ Chinese president speech ▫ Moving flags (2) ▫ President Bush speech (mixed with pollution images)
6	Science provides the answers	▫ 4 clips of researchers in one image; VU climate star with VU logo

7	Yet our world is divided	▫ Al Gore; an opposing republican ▫ Pro-Kyoto protest (2x); anti-Kyoto protest; random protest; Ali G. shouting at protest
8	Can *you* handle this? Play Clima Futura!	Game images: ▫ World view (hurricane) ▫ Flooding rocks ▫ Local view (pollution)
9a	Will you waste your chances…	Game background with: ▫ Deserted building (outside) ▫ Deserted building (inside) ▫ Deserted playground ▫ Statue of liberty in the sea
9b	Or take control! using science *(fades in)*	Game background with: ▫ New sources of power ▫ Eco-bus ▫ Solar-power ▫ Wind-power ▫ Cycling people ▫ People signing contract ▫ Nature images (water) ▫ Nature images (snow)
10	*Final scene* Clima Futura The world is in your hands www.climafutura.nl	▫ Earth animation (rising sun)

Figure index

Table index