Faculty of Science
Information Sciences/Multimedia and Culture
Vrije Universiteit Amsterdam

Game Development Using Game Design Patterns

Serious Games and the Clima Futura Project

W. W. den Boer

November 2008

Supervisor: dr. Anton Eliëns

Second reader: dr. Jacco van Ossenbruggen

Abstract

Nowadays digital games are a very popular form of entertainment. Over the years the game industry has branched out into different areas. Serious gaming is a relatively new sector in the game industry and is rapidly gaining popularity. Serious games are games created for training, simulation, advertising, education, creating awareness, etc. Clima Futura, a game about climate change, is an example of a serious game. Several subjects related to Clima Futura, and therefore serious games, are discussed in this thesis. The aim of this thesis is to investigate to what extent it is possible to describe the Clima Futura game design in a formal description format that can be used for configuring and documenting the game. An important subject in this is the notion of game design patterns. The purpose of game design patterns is discussed as well as their relevance with regard to the creation of an adequate description format for Clima Futura. In addition the possibilities of interactive narratives in the context of Clima Futura and the acquisition of game material through a web service is explored. Next future work and how the findings in this thesis can possibly translate into a description format for serious games in general is discussed. Finally an overall conclusion is drawn.

Acknowledgements

I would like to thank:

Anton Eliëns for support and guidance.

Jacco van Ossenbruggen as the second evaluator of this thesis.

Winoe Bhikharie, Hugo Huurdeman, and Marek van de Watering for their help and valuable contributions.

Table of contents

1. Introduction

This thesis is about games. In particular it is about a game genre called serious games. Clima Futura, a game about climate change, is an example of a serious game. Serious games are software applications designed with the use of game technology and game design principles for purposes other than mere entertainment. Therefore the involvement of an interdisciplinary team is intrinsically related to the development process of a serious game. Since these different interdisciplinary groups will have to work on the development and deployment of a serious game, it is valuable to explore to what extent it is possible to create an abstract description format to help document design choices and configure specific game parameters.

Game design patterns offer a way to describe design choices that reoccur in many games. The collection of game design patterns provide semiformal interdependent descriptions of commonly reoccurring parts of the design of a game that concern gameplay. With respect to the development of a formal description format for a (serious) game the notion of game design patterns can offer valuable ideas and inspiration. To that end the prototype of Clima Futura will be described using the game design pattern collection.

When I started out working on the concept of a description format for Clima Futura, the game was still under active development. The idea was to draw inspiration from the game to develop a format and to incorporate the ideas back into the game. However development on the game gradually came to a stop. This gave me the opportunity to explore additional subjects related to Clima Futura and serious games in general. As a consequence the scope of the thesis has became quite wide, in turn somewhat limiting the exhaustiveness of the subjects but also providing a broad basis for future work.

1.1 Research questions

The primary goal of this thesis is to explore to what extent it is possible to describe the design of Clima Futura in a formal (read: XML) description format. More in particular it will be explored how the description format can be used as a way of configuring and documenting the Clima Futura game design. Therefore the research question is:

To what extent is it possible to describe the Clima Futura game design in a formal description format that can be used for configuring and documenting the actual game?

Next to the research question several sub questions can be defined. An interesting subject in game design are game design patterns. Game design patterns can possibly be an inspiration in designing the description format. Therefore a sub goal is to determine what role game design patterns can play in the Clima Futura description format:

What role can game design patterns play in the Clima Futura description format?

A substantial part of Clima Futura consist of interactive video through which the player experiences the game's narrative. It is therefore valuable to investigate the notion of interactive narratives in relation to Clima Futura:

How can interactive narratives be incorporated into Clima Futura in a flexible way?

In addition it is interesting to look into how Clima Futura can dynamically obtain game materials. Without a fixed set of game materials the game can be different each time it is played, possibly increasing the replay value of the game. The climate portal, a web portal developed by students from the VU Amsterdam, has the promise to be a suitable platform from where Clima Futura can obtain materials:

In what way can the climate portal be of use to Clima Futura?

For future we need to explore the characteristics and issues of serious games as this can be valuable in the design process of a serious game and a formal format to configure this game:

What are the issues to keep in mind while designing a serious game and an accompanying description format?

1.2 Thesis structure

Chapter 2 – Background

Game design patterns as well as serious games are discussed in this chapter. Moreover the difference between game design patterns and 'regular' design patterns is explained (semiformal interdependent descriptions v.s. patterns based on problem-solution pair templates).

Chapter 3 – Clima Futura

To gain understanding why a description format is useful, the Clima Futura project itself needs to be discussed. What is the current state of the project and what are the future goals?

Chapter 4 – The next step - Clima Futura description format

This chapter is focused on the description format of the Clima Futura game prototype. The important game design patterns are discussed as well as the possible configuration structure. The chapter is concluded with a short evaluation of the use of game design patterns in this context.

Chapter 5 – Interactive narratives

Here the possibilities of interactive narrative with regard to Clima Futura are discussed.

Chapter 6 – The climate portal – Organizing game related material

The relation between Clima Futura and the clima portal is explained. RSS feeds and webservices are discussed as well as how these can be useful for Clima Futura. In addition we look into several issues regarding the annotion of multimedia materials.

Chapter 7 – Towards a description format for serious games

In this chapter several charactaristics of serious games and games in general are explored. Important issues are defined that need to be kept in mind when designing a serious game and a formal description format. The game design patterns framework discussed in Chapter 2 might be a good starting point for developing a description format.

Chapter 8 – Conclusions

Here an attempt is made to draw conclusions from all that has been discussed in the previous chapters.

2. Background

In this chapter game design patterns and serious games are discussed. First is investigated what game design patterns are and how they work. This is the theoretical basis for the work in Chapter 4. Then serious games are discussed. What are serious games and what is their significance?

2.1 Game design patterns

"Game design patterns are semiformal interdependent descriptions of commonly reoccurring parts of the design of a game that concern gameplay." (Björk & Holopainen, 2005)

Game design patterns is a tool for understanding and creating games (Björk & Holopainen, 2005). This tool is essentially a collection of design choices possible in games. It focuses on, what the authors perceive as the most essential part of game design, gameplay. In this context, gameplay is defined as *"the structures of player interaction with the game and with other players in the game"*. In their book *Patterns in Game Design*, Björk & Holopainen present a large collection of game design patterns (about 300 patterns) which were harvested by analyzing existing games. Each pattern describes a part of the interaction possible in games. The patterns are related to each other and together they describe the possible gameplay of a game. These patterns can be used to analyze, describe, discuss or design games (Lundgrun, 2006). The notion of **design patterns** forms the basis for game design patterns. The background and origins of design patterns will be discussed later on, as well as the structural framework which offers concepts/terminology to talk about game design patterns – it is the 'medium' where the patterns occur in.

The reason for studying these game design patterns is to constitute a basis, or rather to create a starting point for the development of the Clima Futura description format. Important aspects, such as interaction and gameplay mechanics, that are represented in game design patterns should also be incorporated in the description format. The employment of game design patterns in this context serves two purposes. One is the constitution of different game concepts and design choices which will eventually result in the final game. This is the intended use of game design patterns – a tool to design and talk about games. The other purpose is the aforementioned development of a description format for Clima Futura based on game design patterns and its framework.

2.1.1 Design patterns

As mentioned before, the game design patterns are based on the notion of design patterns. The concept of design patterns was first introduced by the architect Alexander C. in the book *A Pattern Language: Towns, Buildings, Construction* (Alexander, 1977). Alexander asked himself the question whether it would be possible to objectively measure the quality of architectural design. In other words, the question was : *"Is quality objective?"* Alexander believed this was the case. He studied different architectural constructions that were designed to solve the same problem. This way he discovered similarities in the design of the constructions. He called these similarities, patterns. He defined a pattern as *"a solution to a problem in context"* (Shalloway & Trott 2004):

"Each pattern describes a problem which occurs over and over again in our environment and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice." (Alexander, 1977)

Alexander constituted the following four components of a pattern description:

- Pattern name
- Pattern purpose
- How to use (i.e. how to solve the problem)
- Constraints and forces to consider when using the pattern

He also mentioned that patterns can be related to each other and can be used together to solve problems. This notion and the above mentioned description also play an important role in game design patterns. However, Alexander's patterns are focused towards problem solving. Game design patterns are not based on these problem solution pair templates (Björk & Holopainen, 2005). They describe part of the interaction possible in a game and not so much a solution to a problem. According to Björk & Holopainen there are three reasons for the neutrality of game design patterns. It would not be a design tool for supporting creative work if game design patterns were defined from problems, because it would become a method for only removing unwanted effects. Also, patterns that are unwanted in one context can be desired in another context. Lastly, it would be an imprecise tool for removing problems since modifying or removing game design patterns can drastically affect the gameplay of a game.

In the late 1980's and early 1990's a couple of software developers started to explore the use of design patterns in the field of software design. Software design patterns have their origin in the

work of Cunningham and Beck who both pioneered software design patterns and outlined an adaption of the pattern language to object oriented programming (van Vliet, 2000) (Beck & Cunningham, 1987). Although their work is very important, the greatest influence on this topic was a book by '*The Gang of Four*' which contains a major collection of design patterns (Gamma et al., 1995). They described a structure within which to describe design patterns and also presented 23 patterns. These patterns were not created from scratch by the authors themselves but identified as already existing patterns. Again, the similarities between the above mentioned software design patterns and the game design pattern approach are evident.

2.1.2 Framework and game design pattern collection

The game design patterns framework is used to describe components of games. These components, in turn, can then be used to describe the game design patterns. It was built under the assumption that *"playing a game can be seen as making changes in quantitative game states."* (Lundgren, 2006)

This structural component framework consists of four categories (see Figure 1 and below). Each category represents a different view, or abstraction level, of the game components. It is worth mentioning that the framework and the game design patterns can be used independently (Björk et al., 2003). The framework is generic and can be used to describe any game. This property of game design patterns and the associated framework is one of the factors why they can be an

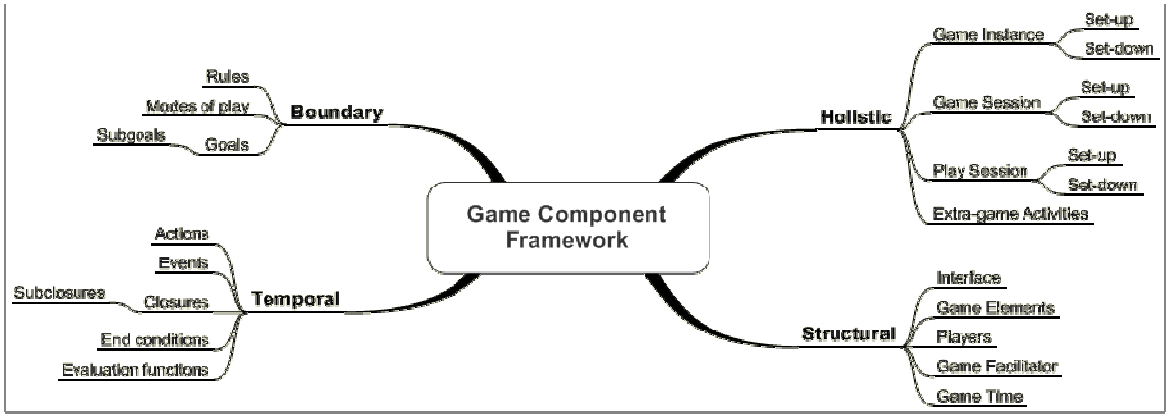

Figure 1 Game Component Framework. (Björk & Holopainen, 2005)

important inspiration for the development of a description format for Clima Futura of for serious games in general.

Holistic components

These components are on the highest level of abstraction. They cover the entire activity of playing a game. The holistic components are useful when exploring the relation between the activity of playing a game and other activities. The following holistic components are defined in the framework:

- Game instance (the whole lifetime of the game, from setup to end)
- Game session (one player's activity during one game)
- Play session (a game session can contain multiple play sessions)
- Extra-game activities (activities related to the game, e.g. uploading one's high score of a game to an online ranking system).

Boundary components

Boundary components define the game world and what can be done within the game world, i.e. they dictate how everything works. They limit the activities of players by restricting or allowing certain actions. These components are useful when defining the activities, rules and goals in a game. Hence the following components:

- Rules (how to play the game)
- Goals and sub goals (motivation for playing the game and how to win)
- Modes of play (the different phases of the game).

Temporal components

The Temporal components describe the flow of the game and the changes in the game state. These are useful to look into when one is trying to improve the flow and interaction in the game. Temporal components include:

- Actions (what players can do to change the game state)
- Events (changes in the game states initiated by the player, other players, or the system)
- Closures (meaningful game state changes, e.g. completing a goal)

- End conditions (determine a change of mode of play)
- Evaluation functions (evaluates the outcome of an end condition)

Structural components

Structural components is the least abstract category in the framework. These components are the basic parts of the game that can be manipulated by the players and the system. Important components are:

- Interface (provides information about the game state)
- Game elements (components that contain the game state and that can be manipulated by players and the system)
- Players

Pattern collection

The patterns in the collection were harvested and identified according to five iterative steps: recognize, analyze, describe, test and evaluate (Björk & Holopainen, 2005). A game design pattern has the following attributes:

- Name
- Description
    - Core definition
    - General description
    - Example(s)
- Using the pattern (*Which design choices have to be made?)*
- Consequences *(What are the consequences of using the pattern?)*
- Relations *(Relations to other patterns. Relation types are: instantiates, instantiated by, modulates, modulated by, conflicting with)*
- References *(E.g. games exemplifying the pattern)*

Table 1 contains an example of a design pattern (Score). Words in italic refer to other patterns from the collection.

Score

Score is the numerical representation of the player's success in the game, often not only representing the success but also defining it.

Having a *Score* value for each player allows the players to easily determine the leader or the winner of the game; the *Score* value also allows players to compare *Scores* between different games and to calculate outcomes of tournaments. Many games give players several different ways to gain *Score* points to promote different tactics.

Example: Pac-Man gives players three different possibilities to gain points: eating pills, capturing ghosts while under the effect of a power pill, or collecting the bonus object when it appears. The player's *Score* is shown in the upper part of the screen next to the current high *Score*.

Using the pattern

The two fundamental choices to be made when creating a *Score* system is first to determine which actions or goals give points and then to decide how large these amounts are. Examples of simple scoring goals include reaching a *Pick-Up* first (thus modulating a *Collecting* activity or a *Gain Ownership* goal) or successfully completing a *Dexterity-Based Action* or an action requiring *Timing*. More complex scoring goals can depend on the distribution of *Resources* between the players or completed *Collections*. In games with *Role Reversal*, different possibilities to *Score* may exist for different roles, but the use of *Score* can provide players with means to continue striving towards goals independent of their current role. The possibility of using *Tiebreakers*, commonly another form of *Score* systems, should also be considered when designing a *Score* system, as *Score* systems usually allow for *Tied Results* unless explicitly designed to make them impossible.

Besides affecting *Player Balance*, the choice of actions and goals that give points can be used to control the overall flow of the gameplay during a game session. This can be done by giving different points for the same achievements depending on when in the game it is achieved, which will encourage the players either to haste or wait; to reward certain types of actions and goals, e. g., *Eliminate* goals and *Player Elimination*; or to reward the use of actions requiring *Non-Renewable Resources* to limit the players' *Freedom of Choice* as the game progresses.

Allowing the players to gain *Score* points for different kinds of goals expands the players' *Freedom of Choice* and promotes *Varied Gameplay* and *Replayability*.

Another important consideration is to determine whether the *Scores* of the players are *Symmetric Information* that should, or should not, be available to all players. If the *Score* values themselves are known, they provide a *Game State Overview* and can give rise to *Dynamic Alliances* where the players actively work together against the leader, thereby achieving a *Balancing Effect*. However, this can cause *Analysis Paralysis* and prevents the *Tension* of not knowing who the leader is and the possible *Surprise* of an unpredicted winner.

One simple way of letting players of *Single-Player Games* compete against themselves or each other is to let them compare *Scores*. The typical way to explicitly support this in games is to use *High Score Lists*. As this requires that the game is played several times, such designs promote *Replayability*.

Consequences

Having *Scores* in games can be seen as an instantiation of an abstract *Race*. When outcomes are determined by who first reaches a certain *Score*, this creates a normal *Race*, but when outcomes are based on having the highest *Score*, the goal of achieving this becomes a *Continuous Goal*. The *Score* is used this way as both *Progress Indicator* and *Status Indicator*.

The use of a *Score* system promotes *Stimulated Planning* as players, given sufficient information, can calculate numerically optimal

tactics not only for themselves, but also for the other players. *Score* is typically the information from a game state that can be used as *Trans-Game Information* or provide *Extra-Game Consequences* in *Meta Games* such as *Tournaments* and *Highscore Lists*. It can also be used to give *Handicaps* by giving some players a bonus to their *Scores*. Some games use the *Score* as a *Resource* to perform actions, for example, to determine order in *Turn Taking*, while others base *Scores* on *Resources*, and completely linking *Score* to *Resources* is possible. This requires the players to see these kinds of actions as *Investments* and make *Risk/Reward* calculations, especially in games, which are parts of *Meta Games*.

The use of *Save-Load Cycles* and *Save Points* make the use of *Score* irrelevant to some degree, as players can repeat parts of the game until they reach a *Score* they are satisfied with.

Relations

Instantiates: *Continuous Goals, Meta Games, Status Indicators, Stimulated Planning, Trans-Game Information, Race, Investments, Dynamic Alliances, Balancing Effects, Tied Results, Replayability, Extra-Game Consequences, Progress Indicators, Resources, Game State Overview, Collecting*

Modulates: *Role Reversal, Player Elimination, Turn Taking, Gain Ownership, Single-Player Games, King of the Hill, Tournaments*

Instantiated by: *Resources*

Modulated by: *Pick-Ups, Handicaps, Tiebreakers*

Potentially conflicting with: *Save Points, Save-Load Cycles*

Table 1 Example of a game design pattern: *Score* pattern.

2.2 Serious games

Nowadays games are a multi-billion dollar industry (Squire, 2005). Hollywood's revenues and cultural impact are outshined by those of the game industry. But besides the glitter and glamour, high revenues, and fancy graphics, games can have a surplus value when deployed as a powerful learning environment. Games can provide situated experiences in which players can learn to solve complex problems. Serious games are all about learning through game play.

2.2.1 What are serious games?

There is not a single definition of serious games. It is usually associated with digital games created for training, simulation, advertising, or education. More generally one can think of a serious game as being a non-entertainment software application that is developed with the use of ideas, concepts, skills, and technology from the field of game design.

Ben Sawyer, co-founder and president of Digitalmill[1], argues that the opportunity to let serious games become a widely used and respected media lies in applying gaming technology and design outside of its entertainment market (Riley, 2007). Sawyer is, at the time of writing, in the progress of constructing a serious games taxonomy (Sawyer & Smith, 2008). The goal is to develop a broader and well defined look at the serious games field. This effort emphasizes there is not yet a consensus of what serious games exactly are or should be. Gee, of the University of Winsconsin-Madison, underlines this problem (Terdiman, 2006). He argues the serious games industry has yet to define itself and must do so in order to grow beyond its nascent state.

2.2.2 Educational games

Games that address serious issues are not new, but the last couple of years interest in them has increased. Egenfeldt-Nielsen gives an extensive overview of educational computer games that were released over the years (Egenfeldt-Nielsen, 2005). The 1970's can be marked as the early years of educational computer games, but it was not until the 1980's that the genre became popular and also researchers started to take notice. Throughout the 1990's a lot of earlier game concepts were being recycled and some questionable titles were released. Edutainment came to the forefront and pushed other type of educational games aside. With the rise of edutainment, educational computer games began to acquire a bad reputation. Even today this reputation is negatively affecting the break-through of serious games.

2.2.3 Skepticism

Despite the recent rise in recognition and acceptance of serious games there still is some skepticism (Jefferies, 2007). Suzanne Seggerman ascribes this to a number of possible reasons. It might be the terminology or the aforementioned reputation of edutainment software (Ochella, 2007). Another issue is the contrast in quality between mainstream commercial games and serious games. Serious games often do not have a financial return and that makes it harder to produce them with the same quality as commercial games. Apart from the skepticism about the quality of the games in terms of gameplay mechanics or graphics, a lot of people are also skeptical about serious games being an effective form of education. Their concern is the lack of

[1] Digitalmill consulting firm: http://www.dmill.com/

evidence proving serious games are better than traditional e-learning or simulation techniques (Jefferies, 2007).

2.2.4 Why serious games?

The advantage of serious games is the possibility to explore scenarios that would be much more expensive, dangerous, or even impossible to explore in the real world. Especially the military and health sectors are benefiting from using serious games in their training program since it allows them to explore high risk scenarios in a safe manner. An example is America's Army[2], a 3D military simulation game owned by the US government. It can be downloaded for free from the America's Army website and is given away at US recruiting offices. The **main aim** of America's Army is however not the training of the US soldiers, but rather recruiting players and changing attitudes about the army (Squire, 2005). The military actually has a long history of using games to simulate strategy on the battlefield. Now that serious games are gradually being accepted and recognized, other industrial braches are also catching on.

It seems that good games are motivating people to learn and enjoy learning. Maybe without even knowing it, designers of good games employ an array of very effective methods to get players to learn. Gee found out these methods share a lot of similarities with cutting-edge principles that are being discovered in research on human learning (Gee, 2003). Good games force people to exercise their learning abilities, often this goes unnoticed and the player enjoys it. Game designers could benefit from incorporating results of scientific studies about learning and cognition into their games but good game designers are already doing a good job at this point. If you want your game to be successful you will have to make games that people can learn in the first place, enjoy learning, and keep playing.

Nowadays companies want their employees to not only learn skills but also to adopt the goals and perspective of the company. With this goals and perspective in mind a company want their employees to think creatively in solving problems. Games are an excellent tool in training for adapting new roles and thinking creatively. The benefit of games is that they can immerse players in contexts where they have access to tools and resource and will have to figure out ways to use these to their advantage.

[2] http://www.americasarmy.com/

It is interesting to note that serious games might become the desired method of learning in the future. Since today's workers grew up with video games and video game logic is basically built into their neural pathways, the choice for serious games as educational tools becomes more obvious. On top of that, games are going to be the predominate form of entertainment and form of interaction in the future (Gee, 2004). From that point of view it seems smart to invest in research and development of serious games as it will fit in seamlessly with people's daily habits in the future.

3. Clima Futura

This thesis is for the most part about the work concerning the Clima Futura project. Therefore is necessary to discuss the project itself.

3.1 What is Clima Futura?

Clima Futura is a game about climate change. The primary aim of Clima Futura is to create involvement with the problem of climate change and letting the player explore cause and effect relations akin to climate issues (Eliëns et al., 2007). Clima Futura was the VU Amsterdam's submission for the Dutch Academische Jaarprijs[3] 2006/2007. The Dutch Academische Jaarprijs is a contest in which universities compete for the best proposition of communicating scientific research to the general public.

Despite all the hard work and against their expectations, the Clima Futura team was not elected as finalists by the jury of the Academische Jaarprijs. However, the Clima Futura team continued to pursue their goal of making a serious game about climate change.

Prior to the presentation of Clima Futura at the Academische Jaarprijs, a paper was written by Eliëns et al. communicating the idea behind Clima Futura as well as conceptual technical aspects of the actual game. The technical aspects described in the article are among other things are the platform for developing Clima Futura, game elements, the game architecture, interactive video, and the game event description format.

Clima Futura consists of three elements that together describe the basic game concept:

- The turn-based game cycle
- Simulation based on a climate model
- Exploration by means of interactive video

Each of these elements can give rise to game events, whether triggered by the player's actions or not. Mini-games are also featured in Clima Futura and can be started by game events.

[3] http://www.academischejaarprijs.nl/

To link the above mentioned elements and to enforce a uniform way of describing the game events, a game events description format was developed. This description format is discussed in more detail in 3.2.2.

3.2 Where are we?

In the autumn of 2007 new members were recruited to join the existing team in developing a demo of Clima Futura. After rethinking the game concept and game interface a new development iteration started that resulted in a rudimentary prototype of the game. The turn-based game cycle, interactive video exploration, and mini-games were present in this prototype. However in a nascent state, these elements showed the basic gameplay and flow of the intended game.

The climate simulation model element was however missing from the prototype. The presence of the model was faked by showing the player notifications of game events just like they were generated from the model's calculations. A more detailed description of the prototype can be found in 3.2.3.

The implementation of a climate model seemed to be a hard to take hurdle in the development process of Clima Futura. Being an integral of part of the game, the lack of such a model and the difficulties that came along with it where responsible for a declining interest in the development and completion of the intended game.

The FEW VU's multimedia team of Clima Futura decided to shift their focus to the further development of the interactive video platform instead. Eventually the work on this interactive video platform transformed into the XIMPEL[4] (eXtensible Interactive Media Player for Entertainment and Learning) project (Figure 2).

[4] http://www.ximpel.net/

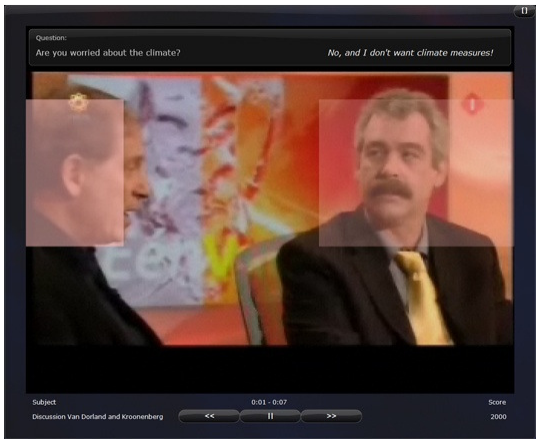

Figure 2 A screenshot from XIMPEL

At the time of writing XIMPEL is being used for the development of Clima Futura. Taking this into consideration, it is still relevant to explore the possibilities of integrating narratives and scenarios with XIMPEL.

3.2.1 Clima Futura interactive video XML format

Huurdeman (2007), as a member of the Clima Futura team, explored the possibilities of the use of interactive video in serious games in his master thesis. Among other things, he developed an interactive video XML structure to facilitate easy modification of interactive video narrative elements in Clima Futura. During the development of his pilot interactive video application he found the following three elements to be important for the XML structure:

- Subject – A sequential list of videos.
- Video – A flash video file.
- Overlay – Clickable hotspots which indicate possible user interaction.

Without going into the exact details, the structure allows for describing the three above mentioned elements. Videos that need to played sequentially can be defined in a list. The links between video can be specified in two ways. The first way is an automatic redirection to another video when the current video ends. The other way is a redirection via a so called branch question. A branch question is a question where there are multiple choices available. Each of these choices leads to another video. These questions can be answered by clicking on an overlay (or: hotspot) showed in the video. The XML format allows for the specification of these overlays.

The existing interactive video XML structure can be enhanced to support new future features of the XIMPEL platform. The three aforementioned elements are never the less essential concepts in an interactive video description and should be taken into account when redefining or enhancing the interactive video XML structure.

In a conversation with the author of the interactive video description format, he mentioned the addition of keywords to the format. With the introduction of keywords, videos can be annotated with short descriptive words to facilitate searching and possibly a more automated way of linking videos or presenting videos in the game depending on the game state. This concept will be discussed in more detail in Chapter 5.

3.2.2 Clima Futura game event description format

In the aforementioned paper where Clima Futura was introduced (Eliëns et al., 2007), a game event description format was presented to facilitate collaborative design and to standardize the way game events are to be described. The game event description format is displayed below in Table 2.

Name of event	Give a meaningful name
Event-id	For administrators only
Type	(Generic/specific) game/model/video
Cause	Gameplay/simulation/exploration
Feedback/information	Give a logical description
Player actions	Indicate all (logical) player actions
Description of visuals	For feedback, information and player options
Additional information	Give a URL with references to additional information and visuals
Relates to event(s)	Give id's or descriptions of related events

Table 2 The game event description format as described in Clima Futura @ VU – communicating (unconvenient) science (Eliëns et al., 2007).

Although this game event description format is suitable for communicating rough ideas, it lacks a more in-depth description of the actual game elements. The description format will need to redesigned so it will be useful as a way to configure the game, document the game, and to allow easy editing of game scenarios by means of an editor.

3.2.3 Prototype

In the winter of 2008 a prototype of Clima Futura was released (see Figure 3 for a screenshot of the game). It featured the implementation and interrelation of interactive video, mini-games, and a scenario.

The gameplay of the Clima Futura prototype can be described as follows. Each turn the player must make a decision to advance in the game. The available choices are multiple choice answers to question regarding the player's next move in the game scenario. In this case the scenario is about a climate related issue, namely the possible flooding of the Netherlands due to the rising sea level. The player can access a part of the scenario by selecting one of the available locations from the map. Next an interactive video, mini-game, or a question on how to proceed will appear. Game events will happen depending on the player's choices in the interactive video parts and question parts. During the game, the player's performance is represented with three status indicators: people, planet, and profit.

The process described above goes on until the scenario ends and the player is presented with statistics about his performance in the game. It is important to mention that most of the game elements and interaction in the prototype were faked and very limited. For instance, the available answers during questions were limited to one answer, most game events were scripted, and most mini-games could not actually be played.

Figure 3 A screenshot from the Clima Futura prototype (v06a)

3.3 Goals for Clima Futura – towards abstract descriptions

One of the goals for Clima Futura is to have a description format in which events, scenarios, and storylines can be formally described. Below these elements are briefly described.

3.3.1 Events

In the so called *Clima Futura Design Bible* several game events were described. The descriptions of these events very much reflected the game concept thought of at that time. In other words, the game events were specifically aimed at the particular game concept and scenario. The format of the events consists however of some useful element that can be reused. The events are described by a number of attributes which resemble the game event description format discussed in 3.2.2: name of the event, the cause of trigger of the event, player options, and visuals.

Ultimately the goal is to develop more abstract events that can be used for any Clima Futura game scenario, or ideally in every (serious) game in general for that matter. This is however a non-trivial challenge.

3.3.2 Scenarios

The following description of a scenario more or less reflects the scenario structure of the Clima Futura prototype. A scenario consists of one or more scenes. Each scenes consists of two or more actions the player can choose from in order to proceed. An action can lead to another scene, an interactive video subject, or a mini-game. See Figure 4 for a schematic overview.

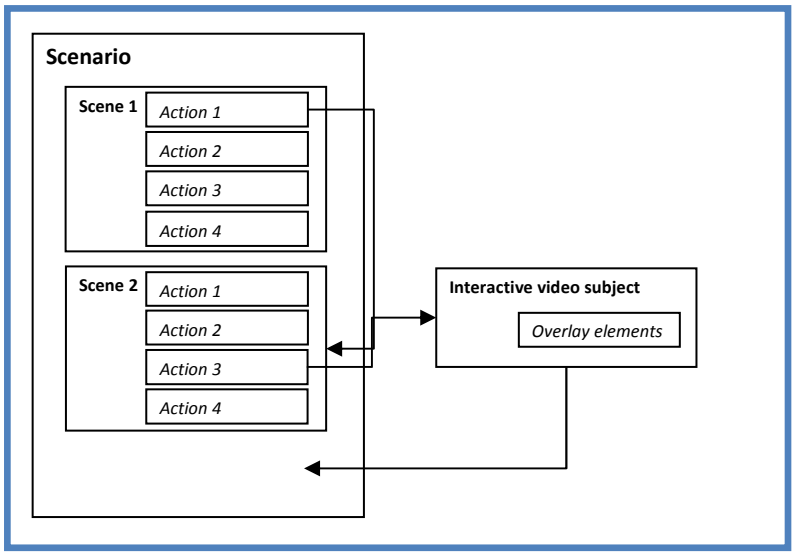

Figure 4 Scenario structure with interrelated interactive video subject

What is particularly interesting to look into is how to design the transitions between scenes, interactive video, and mini-games in such a way that it does not disrupt the scenario. There could be different transitions suitable for different corresponding contexts. Thus the type of transition would be a useful attribute of the description of a scenario.

3.3.3 Missions/Storylines

A mission based game can be characterized as a game in where the player has to accomplish a certain specific goal. Naturally this goal can consist of multiple sub goals which altogether constitute the storyline of the game. One way to accomplish this in the case of Clima Futura is to link multiple scenarios together.

4. The next step – Clima Futura description format

In this chapter the Clima Futura prototype as discussed in Chapter 3 is analyzed and described by using game design patterns. Next valuable elements from this description are transformed into an XML configuration file. Finally the prototype of the XML configuration editor is discussed and the use of game design patterns in this context is evaluated.

4.1 Limitations and presuppositions

Before actually describing the game in terms of game design patterns and deriving the configuration descriptions we need to state the limitations and presuppositions of the description format.

First off the Clima Futura game is in a very nascent state at the time of writing. The prototype gives a global idea of the game concept although it is very basic and lacks detail. Therefore a description format derived from the description of the game (prototype) will reflect this. In other words, the description format will by no means be a representation of an extensive game about climate change. That begin said, it can however act as a solid base for a more elaborate description format if work on the game continues in the future. In addition the description of the prototype in terms of game design patterns is an excellent start to flesh out the game concept in more detail since each game design pattern lists relations to other patterns making it easy to incorporate new ideas.

4.2 Clima Futura description format

4.2.1 Game design patterns and derived XML configuration file

As Lundgren (2006) points out, game design patterns can be used in a number of different ways; as a tool for – analysis, idea generation (both structured and unstructured), initial development, reparation, communication, and teaching. For describing the Clima Futura game concept and fleshing out this initial concept, game design patterns is best used as a tool for structured idea generation and initial development. In this case the structured part of the idea generation comes

from the fact the **prototype** of Clima Futura sets criteria to take into account when selecting appropriate game design patterns. These patterns are the starting patterns that can be built upon later in the structured idea generation – and development phases.

Game concept demands from the Clima Futura prototype (i.e. the core gameplay elements):

- Single player game
- Turn based game
- Access scenario from specific points of the map
- Decision making / choosing actions (from a limited set) to perform in each turn
- Gain/lose points for 'good'/'bad' decisions/actions
- Games within main game: mini-games
- Exploratory interactive video as an integral part of the game

Below the Clima Futura prototype is described in terms of game design patterns. The goal of this is to indentify patterns that reflect the prototype's concept so these can later be transformed into configuration files. The game design patterns are in italics to clearly distinguish them.

Although the ultimate purpose of Clima Futura is creating awareness and letting players explore the domain of climate change, the goal of the actual game is to traverse through the game scenario and get points for doing this. This goal can be regarded as collecting the different scenario scenes, i.e. completing the scenario by making choices in each scene of the scenario. This *collection* goal is a high-level goal requiring the completion of several sub-goals, which are in this case finishing the different parts of the game scenario. Not only is the completion of the collection the goal of the game but it is also the end condition (considering the game instance consists of one scenario).

The scenario is a description of the series of events that are happening in the game. In other words, it describes the story of the game in a *narrative structure*. In the case of Clima Futura the narrative structure will most likely be not fixed in the future, i.e. the player can achieve goals in different ways and the endings of the story might vary, but in the prototype the structure is fixed however. The narrative structure is the pivot on which everything hinges. From here the player experiences the story and chooses actions which in turn trigger events.

To proceed in the narrative structure, the player can choose the next step from a *limited set of actions*. In order for the story to unfold and have any meaning, some actions will be *irreversible actions*. The effect on the game state of these irreversible actions cannot be undone and therefore creates *tension*. Tension also comes from the narrative structure itself which is in turn

modulated by *imperfect information* and *clues*. The narrative offers the player clues on what decisions to make, but most of the time these clues will contain imperfect information creating uncertainty about the consequences of the player's actions and players will have to take *leaps of faith*.

Ultra-powerful events are events that cannot be altered by the player and thus will also be (indirectly) responsible for some of the tension the player will experience. These ultra-powerful events just occur at some point in time, either trigged or not triggered by the player's actions. Once the event has started it cannot be influenced by the player. An example is the eruption of a volcano in the game.

The player will be able to access different parts of the scenario by clicking one of the highlighted locations on the map. The map can be seen as a very basic *game world* and the highlighted locations as *goal points*, locations in the game world which the player can enter in order to complete a goal. Using these goal points can give the player a sense of control and *freedom of choice*.

Some actions from the limited set of actions trigger mini-games or interactive videos. Mini-games are *games within the game* and can be regarded as *quick games*; games with a single concrete goal and a few basic actions. These games are played completely within the framework of the main game. Mini-games offer the player *varied gameplay* and possibly more excitement since most mini-games will be *real-time games* and contain *dexterity-based actions*.

During a mini-game the player can receive *rewards* and *penalties*. When the end condition of a mini-game is met, the player returns to the main game and the obtained points from the mini-game are added to the player's *score*. The consequences of actions chosen in the scenario scenes are also translated to rewards and penalties. Ultimately all the player's actions are reflected in three *status indicators*: people, planet, and profit.

One thing that is missing from the game design pattern collection is the notion of interactive video. The pattern that best resembles the properties of interactive video is *cut scenes*. However, there are some discrepancies. Cut scenes are not interactive and even cause *downtime*, a period of time where the player cannot affect the outcome of the game. Therefore the *illusion of influence* is difficult to maintain during cut scenes. In comparison to cut scenes, the illusion of influence is a lot better in interactive video since the player can interact with elements in the video. The thought of being able to change the outcome of the game is often not an illusion, in

interactive videos the player can actually change the outcome or direction of the game. However, in most interactive video instances there will be periods of downtime. This are the moments where there are no elements to interact with and the player will have to watch the video without being able to directly influence the game. Interactive video will be an important and omnipresent part of Clima Futura and therefore it makes sense to create a custom game design pattern for interactive videos based on the cut scenes pattern. Such a pattern could also be useful for future serious games project containing interactive video.

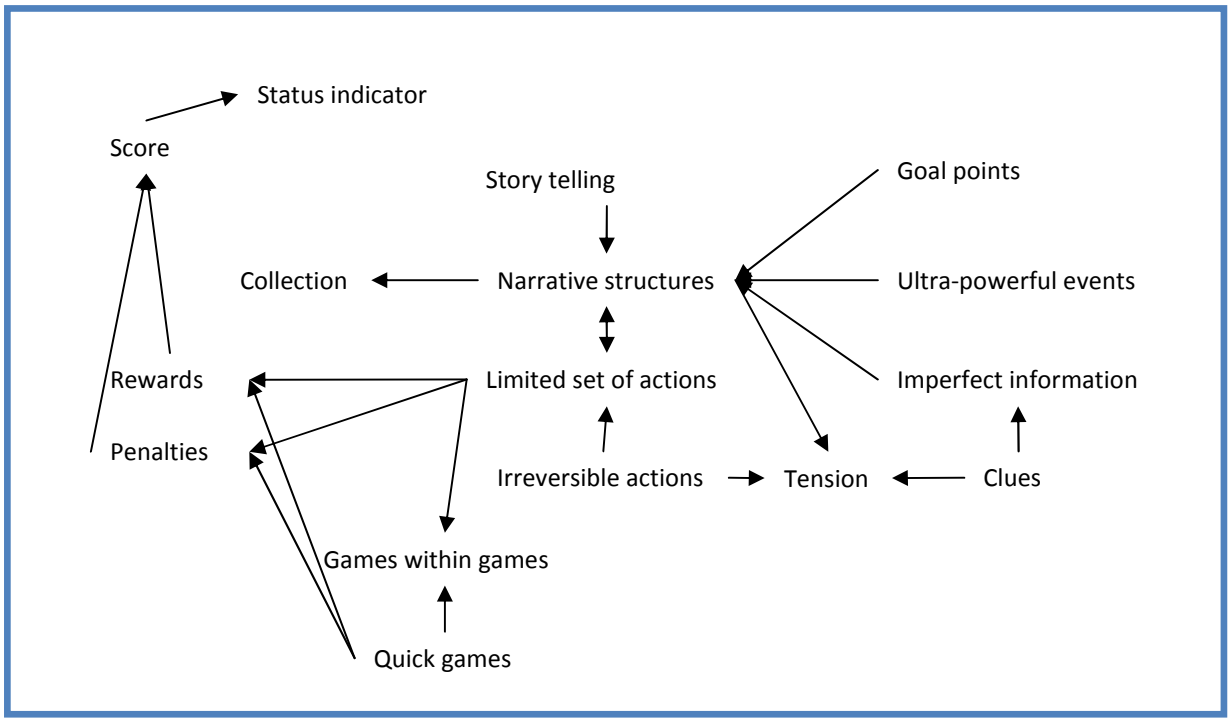

Figure 5 Pattern graph of the dominant game design patterns in the Clima Futura prototype

The core of most games only consist of ten to twenty dominant game design patterns (Lundgren, 2006). The sixteen dominant patterns and their relations in the Clima Futura prototype are depicted in Figure 5. In Table 3 the patterns are again listed, now with their definition. The pattern graph (Figure 5) shows the connections between the game design patterns. Presenting the patterns in a graph is a visual aid in understanding how the patterns are interrelated. The patterns in the original game design pattern collection are all connected to each other with three types of relations: instantiate, instantiated by, modulate, modulated by, or conflict with.

Depending on the context these relations can be explicitly mentioned in the pattern graph. In this graph the edges mean 'influence' of 'affect.' It is however not clear from the graph how much one pattern influences another pattern or how important one particular pattern is in the overall game concept. A pattern can also have an indirect effect on another pattern although there is no edge directly between the two patterns (only indirect through another pattern). As Lundgren (2006) mentions, it is difficult to foresee the precise effects of modulations theoretically and therefore hard to formalize in a graph or formal language. Play testing is a good method to find out which parts of the game have to be tweaked in order to balance the game.

	Pattern	**Definition**
Prototype core game design patterns	Clues	"Clues are game elements that give the players information about how the goals of the game can be reached."
	Score	"Score is the numerical representation of the player's success in the game, often not only representing the success but also defining it."
	Goal points	"Goal Points are locations in the game world which the players can enter in order to complete a goal."
	Imperfect information	"One aspect of information about the total game situation is not fully known to a player, either the information known is totally wrong or the accuracy of the information is limited."
	Status indicators	"Players are given information

	about a certain part of the game state or other players through other means than observing a game element.”
Irreversible actions	“Actions whose effect on the game state cannot be undone.”
Limited set of actions	“Players can only have a few actions to choose from.”
Penalties	“Players are inflicted with something perceived as negative or stripped of an advantage, due to failure to meet a requirement in the game.”
Rewards	“The player receives something perceived as positive, or is relieved of a negative effect, for completing goals in the game.”
Ultra powerful events	“Events that cannot be affected by player actions.”
Narrative structures	“The structures of the stories that are unfolded by playing the game.”
Story telling	“The act of telling stories within the game.”
Tension	“The feeling of caring about the outcome of actions or events in a game without having full control over them.”
Collection	“The completion of several

		goals that together form a coherent unit."
	Quick games	"Quick Games have a single concrete goal and few basic actions."
	Games within games	"A game which is played completely within another game."
Prototype additional game design patterns	Game world	"The environment in which the gameplay or parts of the gameplay takes place is determined by the spatial relationships of the game elements."
	Varied gameplay	"The game provides variety in gameplay, either within a single play session or between different play sessions."
	Dexterity based actions	"Actions whose success or failure depends on some form of dexterity, in most cases, eye-hand coordination."
	Downtime	"The player cannot directly affect the outcome of the game for a period of time."

Table 3 Clima Futura prototype core game design patterns. Definitions quoted from Björk & Holopainen (2005).

Besides the main patterns from the prototype described above, sub patterns or additional patterns can be selected to enrich the game concept and describe it in more detail (Table 4). The following features or gameplay elements can be added to enhance the gaming experience:

- Player can choose/set their goal(s)
- High score list

- Provide instructions on how to play the game
- Statistics after the game ends
- Difficulty levels

In the Clima Futura prototype the goal is to finish the game scenario and earn points. During the scenario the player is confronted with climate related problems and has to take actions in response. Although this approach lets the player explore cause and effect relations in the context of climate change, it lacks a defined learning goal which the player can hold onto while playing the game and can give a sense of purpose. Letting players define their own goals can encompass this. *Player defined goals* give the player control over what they want to achieve in the game. The number of parameters a player can configure depends on the rule system of the game. In the case of Clima Futura the goals could be defined in terms of the parameters people, planet, profit.

The scores of players can be saved in *high score lists*. Using high score lists improves *replayability* of the game as players want to outdo themselves or other players in a next game session. In the case of tied results a *tiebreaker* needs to be defined. A tiebreaker is a simple rule that will prevent tied results in the high score list. An example of a tiebreaker is older scores having priority over newer scores.

When playing the game for the first time, it may be unclear how the game system works and how to play the game. To help the player get started, *extra-game information* can be provided. Extra-game information is information that concerns objects outside the game world. One can think of instructions on how the interface works, i.e. how to play the game.

In order to balance the game and to give individual players the *right level of difficulty*, *handicaps* can be introduced. Handicaps can make actions in the game easier or more rewarding for certain players. More experienced players can get a more challenging gaming experience by choosing their right level of difficulty and thus lowering the level of handicaps.

Some effects of actions cannot be witnessed directly after the action takes place. An example of such an *delayed effect* is the player choosing a certain action and a couple of scenario scenes later being confronted with the effects of the decision.

To reflect on a finished game session an overview of achievements and a history of the actions can be presented. Such a *game state overview* can show the player the statistics of his performance. In the case of Clima Futura the traversed path in the interactive video can for instance be visualized.

	Pattern	Definition
Additional game design patterns	Extra-game information	"Information provided within the game that concerns subjects outside the Game World."
	High score lists	"High Score lists give players the chance to rank themselves against other players who have previously played the game."
	Game state overview	"Players are provided with information that extends beyond the observational abilities provided by game elements."
	Delayed effects	"The effects of actions and events in games do not occur directly after the actions or events have started."
	Player defined goals	"Goals and subgoals that players can create or customize within the game itself."
	Handicaps	"Making gameplay easier for certain players in order to make all players have the same chance to succeed."
	Right level of difficulty	"That the level of difficulty experienced by the player is the one intended by the game design."

Table 4 Clima Futura additional game design patterns. Definitions quoted from Björk & Holopainen (2005).

Now that the concept of the game is described in terms of game design patterns, useful elements from these patterns can be extracted and transformed into an XML configuration format. However, not all patterns that are 'true' for Clima Futura can be translated into meaningful configuration files. A lot of game design patterns describe gameplay elements that cannot be configured, they are either present in the game design or are not. Listing such elements as a single line of XML code would not be useful. The appropriateness of using game design patterns in the context of developing a description format is discussed in more detail in the evaluation part at the end of this chapter.

Here the derived XML configuration files are presented. The purpose of these simple configuration files is to configure the game to ones liking in a relatively easy way. When something is declared, an attribute is used (id="…"). When a tag links to something, an element with text is used (<id>…</id>).

Score system

```
<score_system>
	<high_score_list>
		<scores_saved>20</scores_saved>
		<tiebreakers>
			<older_first apply="true" />
		</tiebreakers>
	</high_score_list>
</score_system>
```

In the score system the number of scores saved in the list can be set as well as whether older scores have priority over newer scores in the case of a tied score or not.

Goal

```
<goal id="…">
	<scoring>
		<reward>...</reward>
		<penalty>...</penalty>
	</scoring>
</goal>
```

A goal can have a reward or a penalty. The reward or penalty element links to a reward or penalty id element which contains the actual amount of points.

Reward/penalty

```
<reward id="…">
	<scoring>
		<points>50</points>
		<handicaps>
			<difficulty level="easy" modifier="+10" />
			<difficulty level="normal" modifier="-10" />
			<difficulty level="hard" modifier="-20" />
		</handicaps>
	</scoring>
	<visualization active="true" />
</reward>
```

When designing a score system it is important two determine two things: which goals gives points and what are the amounts of points at a certain difficulty level. The modifier specifies the amount of points to add or subtract from the value of the points element. For each difficulty level another modifier can be set. A reward or penalty can also be visualized explicitly in the game to emphasize the effect of an action or the completing of a goal.

Scenario

```
<scenario id="...">
	<title>...</title>
	<description>...</description>
	<scene id="...">
		<title>...</title>
		<description>...</description>
		<clue>...</clue>
		<goal>...</goal>
		<actions>
			<action id="...">
				<title>...</title>
				<description>...</description>
				<follow_up_id>...</follow_up_id>
			</action>
		</actions>
	</scene>
</scenario>
```

This configuration element reflects the description from paragraph 3.3.2. A scenario can have multiple scenes and actions. In each scene a clue can be displayed that can help the player in making a decision. Also a goal can specified. Accessing a scene is a sub goal of the high-level collection goal (collecting scenario scenes to complete the scenario). In each action a follow up id

has to be defined. This is the link to another scene, mini-game, or interactive video. An action can also lead to a view of the map. In this case **multiple** follow up id's can be entered that link to goal points that have to be active on the map.

Clues

```
<clue id="...">
        <title>...</title>
        <type>...</type>
        <text>...</text>
        <disappear_from_level>...</disappear_from_level>
</clue>
```

Clues can be included in the scene part of a scenario configuration. This way a clue can be presented to the player before choosing an action. A clue can be of three different types: *advice*, *encouragement*, or *warning*. The visual representation of the clue could be changed depending on the type. To adjust clues to the difficulty level being chosen, it is possible to let clues not show up at certain difficulty levels from this designated level and above. Clues can also contain *extra-game information* about how to play the game for instance.

Goal point

```
<goal_point id="...">
        <title>...</title>
        <location x="..." y="..." />
        <follow_up_id>...</follow_up_id>
</goal_point>
```

Goal points are the locations on the map the player can click to start a part of the scenario. Important is the location of the goal point in the game world. In the follow up field an id of for instance a scenario scene can be entered.

4.2.2 XML configuration editor/generator prototype

The XML configuration editor is a tool to help setup and generate the XML game configuration units as described above. It was made with the Flex 3 framework[5]. The editor offers a form for

[5] http://www.adobe.com/products/flex/

the user to fill out the necessary fields and eventually generate the XML code. This code can then be copied and pasted in the final configuration file. The approach of copying and pasting is somewhat rudimentary but at the same time practical. In addition one keeps the freedom to edit the final configuration file manually when copying and pasting the different chunks of XML code.

Based on the prototype of the XML configuration editor S.V. Bhikharie made an application to setup and generate playlist for the XIMPEL platform. He constructed the application in a modular fashion so it can be extended relatively easy. See Appendix B for a more elaborate description of the XML configuration editor and the application based on it.

4.2.3 Evaluation of game design patterns

Having worked with the game design patterns collection for some time we can conclude that the game design patterns collection and method is not well suited as a basis for a game description format for the Clima Futura prototype.

The goal of game design patterns is to support the design process of game concepts. To this end, the patterns offer a knowledge base of useful parts of game concepts. Game designers can select patterns based on an initial idea of how the game should work or based on external requirements. The patterns describe relations to other patterns and thus the designer can follow these relations and choose the right sub-patterns to flesh out the game's concept depending on the specific context. This process can go on until there is an initial game design.

During the description of the Clima Futura prototype in terms of game design patterns I found the method of following the 'advice' the patterns offer and the description of possible relations to other patterns quite useful. When going through the pattern collection new game concept ideas arose, although they did not always fit in with the game (Clima Futura) at hand. This being said, we can state that game design patterns can be a helpful tool in the design process of a game concept.

In the context of developing a description format for Clima Futura, the game design pattern method did not prove to be an adequate tool however. The problem is the use of the game design pattern method in the wrong context. The intended description format requires a much more clearer and detailed picture of the game than a mere game concept designed with game design patterns can offer. Selecting a game design pattern from the collection in the design

process only means this certain game interaction element will be present in the future game design, it does not offer a ready to use object oriented programming solution like software design patterns do.

It is however true that certain game design patterns can offer inspiration for constructing a description format for Clima Futura as shown in the XML elements described above. Game design patterns that describe relatively low level concepts are suited best for this. The bulk of the patterns however describe concepts akin player's psychology and immersion which are notions that do not lend themselves very well to be transformed into a strict formal structure. Moreover, a description format that also serves as configuration structure is very much bound to the context. In other words, the different parameters of the configuration file depend largely on the concretized game concept and the actual implementation of the game at hand.

5. Interactive Narratives

Designing a description format which covers every aspect of a game is a challenging task. As it became clear in the previous chapters, drawing the line on the level of detail to include in the description format is hard. Especially in the case of Clima Futura, where there is not yet an actual clear game concept to work with. Because of this, a very detailed and complete description format was not even possible in the first place. At the other end, a too abstract format would be of little value since it would be too generic for this specific application.

A constant factor however in the evolving game concept of Clima Futura is the notion of a story through which the player is able to learn about the topic at hand. In Clima Futura the story is presented by means of interactive video. With the use of interactive video the course of the story can be changed by the viewer, to produce a more meaningful experience, as seen in the Clima Futura and XIMPEL demo applications (see Chapter 3). With the story being an omnipresent element in the Clima Futura concept, a shift in focus to the research of a story system and/or a story description format seems a valuable decision. That being said, one could write an entire book about the research and exploration of the field of interactive narratives. A lot of subjects related to interactive narrative go beyond the scope of this thesis and therefore we will only focus on subjects closely related to the use of interactive narrative in Clima Futura.

Before exploring interactive narratives in relation to Clima Futura it is needed to discuss some terminology. What exactly are narratives? David Bordwell and Kristin Thompson (2001) define a narrative as *"a chain of events in cause-effect relationship occurring in time and space"*. Gérard Genette (1980) defines **the content** of a narrative as the story (what is narrated) and the narrative as **the form** of the narrative discourse (how it is narrated). Interactivity can however change how a story is narrated. Each viewing has the potential to results in a narrative specific to that viewing and thus a different experience.

5.1 The Automatist Storytelling System

Micheal Murtaugh (1996) describes a specific kind of storytelling system in his master thesis *The Automatist Story Telling System*. The approach of the automatist storytelling system is different

from for example hypermedia story systems where connections between story material are ‘hard coded’. Murtaugh describes the interaction in hypermedia systems as limited to a pre-determined matrix as every possible action from the viewer has to be explicitly created by the author (much like the Clima Futura/XIMPEL interactive video branching structure). Storytelling systems have a different approach in where the material itself is described, instead of the sequencing or linking, to allow dynamic sequencing.

The specific model discussed in Murtaugh’s work has the prefix Automatist which stems from automatism, a surrealist movement. The essence of this movement is expression of the subconscious, art based on free association, and therefore the exclusion of a superimposed structure. The automatist storytelling system’s model has a similar approach as it is content driven and decentralized. Keywords are used to create possible associations between materials instead of the author explicit linking the material.

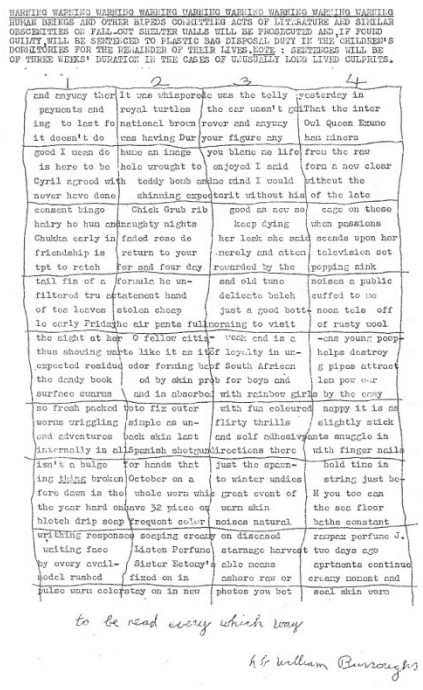

Figure 6 William S. Burroughs, To be read every which way, 1965.

This approach also shares similarities with the work of the situationist movement of the late 1950’s and, to a lesser extent, the work of William Burroughs (ten Berge, 1997). *Mémoires* (Paris, 1959) is a book by situationists Guy Debord and Asger Jorn that is composed of text fragments and images taken from their original context (*détourné*) and where the reader is not obliged to follow a predefined narrative (*dérive*). The result is a wandering between elements where

connections with materials come and go. In *Mémoires* the material is carefully selected by Debord to convey an ideological message. In contrast, William Burroughs used almost anything to construct his so called *cut-ups*; words and passages from newspaper and literature cut out and rearranged to be read every which way[6] (Figure 6). One of the goals was to free the reader from the structure imposed by the author and to let the reader decide in which order to read the words. According to Burroughs, the controlling role of the author had to be as insignificant as possible (Birmingham, 2008).

5.2 The keyword approach

Keywords play an important role in the decentralized approach described by Murtaugh. Content can be connected to one or more keywords, thereby creating the possible connections between materials that share the same keywords. A storytelling system can offer the viewer controls to activate these keywords which in turn select the available materials. One such a storytelling system that uses content described by keywords is *ConTour* (Murtaugh, 1996) (Figure 7). The following is a short description of the role of keywords in *ConTour*.

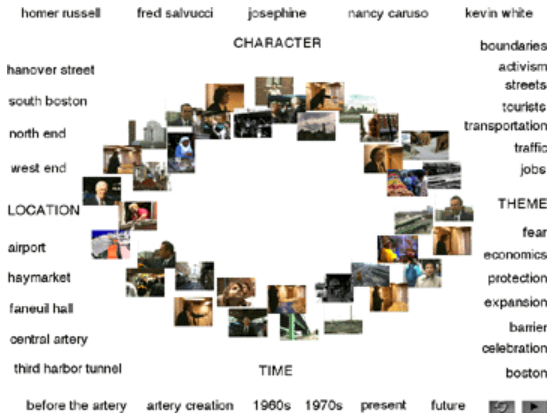

Figure 7 An initial ConTour screen.

Keywords and materials each have a activation value that gets raised whenever a keyword is activated or a material is presented. This activation value represents the material's relevance to the current state of the story. When a keyword is activated, activation is spread to materials

[6] For additional information about Burroughs's cut-ups you can consult the following excerpt from a documentary about Burroughs's work: http://www.youtube.com/watch?v=6NU3dIdqIBw. Burroughs also applied the cut-up technique to film. This resulted in an experimental short film that can be viewed online via http://video.google.com/videoplay?docid=-3151368650139493236.

described by the same keyword. When a material is presented, activation is spread to keywords describing the material. Notice how there is an interaction between these two events. Eventually the system presents the material with the highest activation value to the viewer. During the presentation the viewer is able to activate keywords. This way the presentation can be steered in a certain direction.

An interesting addition to the basic operation described above is the use of spread-weights. Category's of keywords can be weighted as negative, positive, or zero. When a category is set to negative, active material described by a keyword from this category suppresses other materials described by the same keyword. This allows for a presentation of a breadth of content described by this particular category. The inverse is true for positive weighted keyword category's. Combining both positive and negative weighted keyword category's can give interesting results. Murtaugh gives an example of a presentation about the viewpoints of **different characters** (**character** keyword category set to negative spread-weight) on a **particular theme** (**theme** keyword category set to positive spread-weight).

What is interesting about the keyword approach is the simplicity of it. The materials are not described by numerous slots with corresponding values, but with simple relations to autonomous keyword elements. Moreover, there is no need for extensive editing sessions before adding material to the content base. This makes it accessible for all kinds of authors to use (also groups of authors) and makes the content base extendible.

With an easily extendible content base, the opportunity to add material from online video databases arises. YouTube[7] is an online video hosting service with thousands of videos annotated by keywords. Since the videos are already described by keywords it would be a perfect source for extending the material of a keyword based storytelling system. There are however drawbacks to this idea. Each unit of content would ideally be coherent on its own, or as Murtaugh describes it: *"…each piece should form a kind of story phrase, complete enough to be coherent, yet not covering too large a range of ideas"*. This is a great dependency of the system and especially difficult to cope with when dynamically adding content from a service as YouTube. Depending on the goal of the author, material will still have be carefully selected or edited to form a so called story phrase.

[7] http://www.youtube.com

With YouTube being somewhat of a reliability problem with regard to the quality of the material, an intermediate form is desirable. The climate portal[8], developed by students from the VU Amsterdam, is a platform that collects and presents climate related material. Moreover, the portal can provide Clima Futura with appropriate material and thus offering the opportunity to author any raw material before using it in a narrative. The climate portal will be discussed in more detail in Chapter 6.

5.3 Degree of control

The role of the author is very limited in the automatist storytelling system approach; selecting appropriate materials and annotating it with keywords. As this role is inherent to a decentralized and content based system, the question is whether the author wants to give up the controlling role to have the advantages of such a system. I argue the answer greatly depends on the desired approach and intention of the author.

What exactly are the consequences of an interactive narrative in the first place in relation to the author's role? Regular documentaries and films are structured in a particular way and have a fixed length. The author often has to make a selection of the available material to conform to a certain theme and/or audience. This in turn limits the use of content and variety of meaning. An interactive form does not put a restricting on the use of content and can offer the viewer a more meaningful experience since one can form one's own narrative from the available material . Moreover, in the case of the automatist storytelling system the author does not have to explicitly link the material. The system can automatically link material based on the keyword descriptions. This approach makes the system scalable as new content can be easily added (as discussed above) and therefore saves time and effort.

Since the sequencing is not pre-coded in the automatist storytelling system, the viewer is able to steer the narrative in a certain direction. The result is a very *Mémoires*-like wandering between materials but possibly quite random, especially if each piece of material is not coherent enough to ultimately be part of a meaningful narrative. However, this approach seems ideal if the goal of the author is the to let the viewer explore the material her/himself and have a very personal

[8] http://www.climateportal.nl/

experience. But if the author wishes to have a more directorial role and have strict control over the form of the interactive narrative this approach is less appropriate.

For instance, an author may have a certain political or educational message to convey with the narrative and the structure of this narrative may be important to successfully getting the message across. In this case a branching approach as used in the Clima Futura/XIMPEL interactive video application is a better solution since it offers explicit control over the form and sequencing of the different elements. However, the drawback to this is the relatively large amount of time one must spend to create links between material, i.e. define branch points and connect material to them. To that end, such an approach is less scalable overall and less suitable to work with for multiple authors working on the same project. In other words, preserving the consistency of the experience when the amount of material grows becomes harder.

In Chapter 5.5 the possibilities of a hybrid between the two approaches to have the advantages of both will be explored.

5.4 Properties and classification of interactive narratives

The following is a summary of the description of five fundamental properties of interactive narrative as discussed by Murtaugh (1996). In the context of this thesis the description of these properties can be seen as a guide that can provide useful information when designing (or improving) an interactive narrative system.

Narrative intention refers to the goal of the author when telling a particular story. A storytelling system should give the author ways to communicate the story with the intention he or she desires.

Narrative immersion is about how well the narration works as a background element to the narrative. In the case of an interactive narrative it is important to consider how the interaction with the viewer is managed. Disrupting the narrative should be avoided unless the author emphatically wants this to be part of the interaction.

Although an interactive narrative is not a static object, a sense of structure is nevertheless important. **Narrative structure** can provide a sense of shape, pace, rhythm, and closure to the viewer. Over the course of the experience the story system should keep track of already shown

material to avoid repeating it and to articulate certain aspects. If the system is aware of the role or type of a certain material in the overall structure, it can dynamically present transitions between materials and/or generate appropriate titles for instance.

Narrative response relates to how the viewer can influence the narrative. Important to consider here is what type of controls the viewer is provided with. When the controls operate at the diegetic level (within the world of the narrative), the viewer has control over what happens in the story. Controls operational at the extradiegetic level (at the level of the narrating act) allow the viewer to control how subsequent material is selected and presented. Murtaugh argues narrative response at the diegetic level should best be avoided because it breaks the so called diegetic barrier and to that end works against narrative intention and immersion.

During an interactive experience the viewer can be guided through the narrative by the system. There may be different ways the viewer is able to interact at a particular point in time and **narrative guidance** can emphasize relevant choices depending on the current state of the narrative. The degree of interactivity could also be related to the narrative structure. In other words, some parts of the narrative may support a higher degree of interactivity than other parts representing respectively segments where the viewer is able to explore or where the viewer just sits back en enjoys.

Next to these properties of interactive narrative described by Murtaugh, Kuitenbrouwer (2005) wrote a continuum by which to classify ways in which story systems can be interactive. He used the **user's freedom** as criterion to classify the kinds of systems. The first kinds of systems are branching –tree types of environments in where the author has created all the possible paths (like the interactive video in Clima Futura and XIMPEL). In these multi linear environments the users can not change or add anything to the system; the author has already determined what material is available and how to explore it. These kinds of systems are the least interactive on Kuitenbrouwer's scale. Then there are the environments that consist of a fixed set of materials with paths where the user has influence upon. The links between materials are generated in response to the user's actions. In other words, the user has the ability to change the context of the materials. Next are environments that have a dynamic set of paths as well as a dynamic set of materials. Here a user can both influence paths and add or remove materials from the environment but within predefined restrictions. The next step on the scale are multi-user environments where users author content and influence the ways of interaction in the

environment. The last step is for systems that have open rules; only the medium is specified. Kuitenbrouwer notes that these kinds of systems will only thrive when self-organization emerges.

5.5 Towards a content-based storytelling system for Clima Futura

What if there was a way to combine the interactive overlays (hotspots) and clip questions from the Clima Futura and XIMPEL platforms with the content based approach of the automatist storytelling system? If this could be accomplished the viewer would be able to interact with elements on the screen but the author does not have to make explicit link between material.

How can such a system be constructed? Just like with the automatist story telling system, all materials have to be described by keywords. The author has to select appropriate material and link it to the keywords. Based on the keyword relations, material can be dynamically sequenced. This approach renders the branching structure that is now employed in the Clima Futura and XIMPEL platforms obsolete.

The overlays that function as hyperlinks to other videos in the Clima Futura and XIMPEL platforms can be used as controls for the viewer to steer the story in a certain direction (Figure 8). These hotspots can be connected to appropriate keywords and whenever a hotspot is clicked, the activation value of these keywords gets raised. Then the system immediately determines the keyword with the highest activation value and sequences the related material accordingly. Keywords that describe the material itself are also added into the mix. The author will still have to define overlays but the advantage is that the narrative, or sequence of the material, is bound to change whenever the content base grows. This way the viewer can have a different experience every time he/she uses the application if new materials were added.

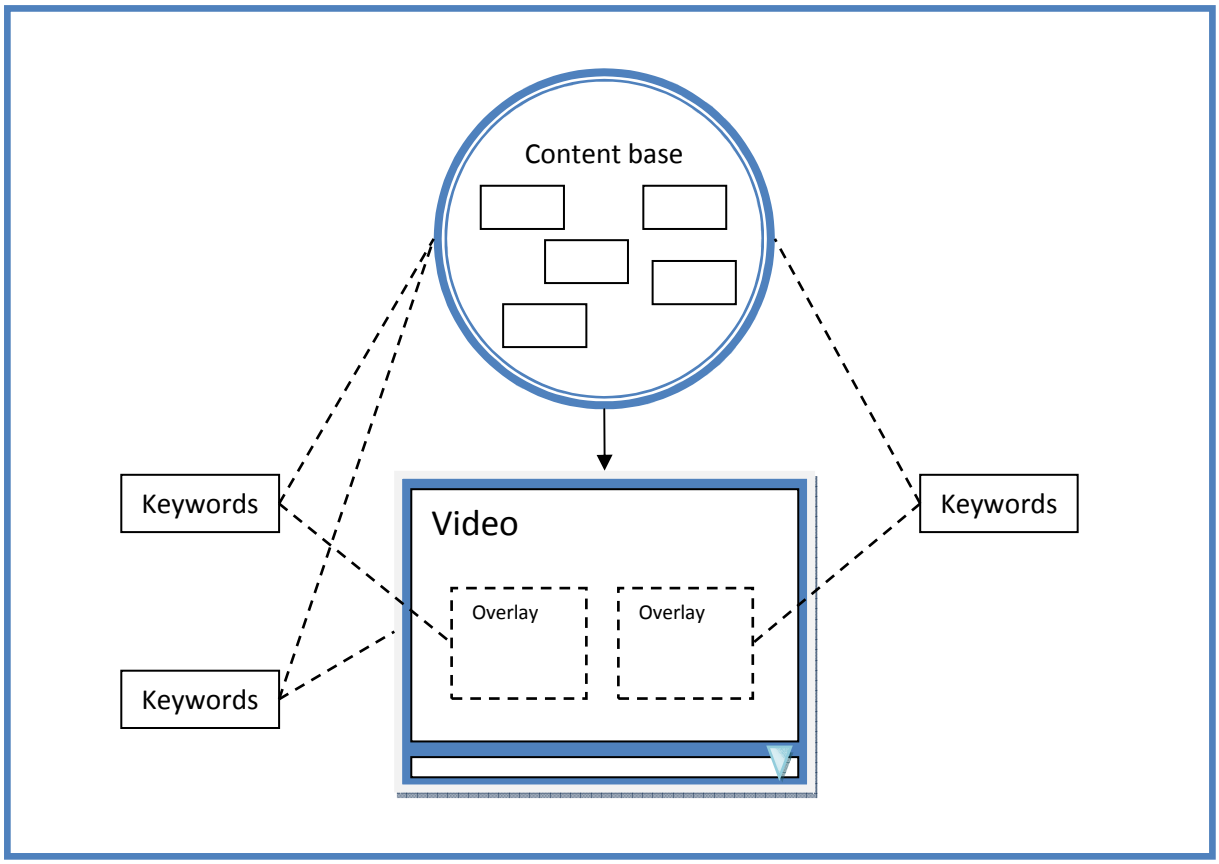

Figure 8 Content-based approach for Clima Futura

In the original Clima Futura approach the narrative response occurs at the diegetic level as the viewer is able to choose what happens in the story. With the above described proposal the narrative response is more difficult to determine.

However the viewer has the control over the mechanics of material selection rather than over what literally happens in the story, the controls are still mapped to actual elements in the story (hotspots). Using these controls will break the aforementioned diegetic barrier and thus disrupting the narrative. Moreover, when the story stops to wait for user input the narrative also gets disrupted.

The question is whether this disruption is really problematic. Murtaugh argues it is because *"…the technique shifts the viewer's attention from the narrative to its disruption"*. In the case of the content-based approach for Clima Futura described above this issue can be resolved, **if necessary**, by incorporating two additional changes:

- **Remove moments where the system waits for user input in order to proceed**. This means whenever a clip ends, a new one starts playing automatically based on the current highest activation value.
- **Place the viewer's controls outside the story world**. One way to support this is by letting the author link keywords to a certain portion of a video[9]. During playback these keywords can be displayed next to the video. The viewer is able to click these keywords only for the amount of time they are linked to a certain duration of the video. By clicking the keyword the viewer raises the activation value of the particular keyword and in turn improves the chance the next clip is related to this keyword.

Putting all the issues about disrupting the narrative aside, is it really worth sacrificing the pre-defined branching structure and overlays from the original Clima Futura platform to have a more extendible content base as described at the beginning of this paragraph? I think this depends on the author's intention. The choice is between an approach on one hand where the author has explicit control over the entire experience but with a tiresome authoring process and on the other hand an approach where the author is able to easily add material but does not have control over the exact sequence of the material. Ideally a system would offer the author the choice to use both methods in conjunction. This way whenever the author feels the need to explicitly link a particular material to an overlay or end of a clip he/she can do so. Otherwise keywords can be used. With regard to this choice, the wider the palette of tools a story telling system offers, the more detailed the author is able translate her/his ideas into a story. In other words, maybe this clear-cut distinction between one approach and the other is not needed and moreover not desirable.

[9] This approach resembles the stream based video representation as described by Marc Davis (1995). He argues a clip based representation imposes a fixed segmentation on the content of a video. In contrast, a stream based system supports multi-layered annotations which can generate multiple segmentations of a video.

6. The climate portal – Organizing game related material

The climate portal[10] is a web based application with the purpose of collecting and presenting climate related material. It was developed by students of the VU Amsterdam as a deliverable for the course multimedia casus 2007. The assignment of the course was to develop a media portal for climate related material that can both act as information source for browsing as well as a web service from which Clima Futura can obtain game related material. During this course the students developed the application as a set of interconnected components that gather, transform, store, and display information.

6.1 Approach

The climate portal retrieves data from a verity of different RSS feeds. RSS is an acronym for *Really Simple Syndication* and is specified in XML[11]. The RSS format allows web publishers to syndicate content quickly and automatically. An RSS document usually includes full or summarized text accompanied with metadata.

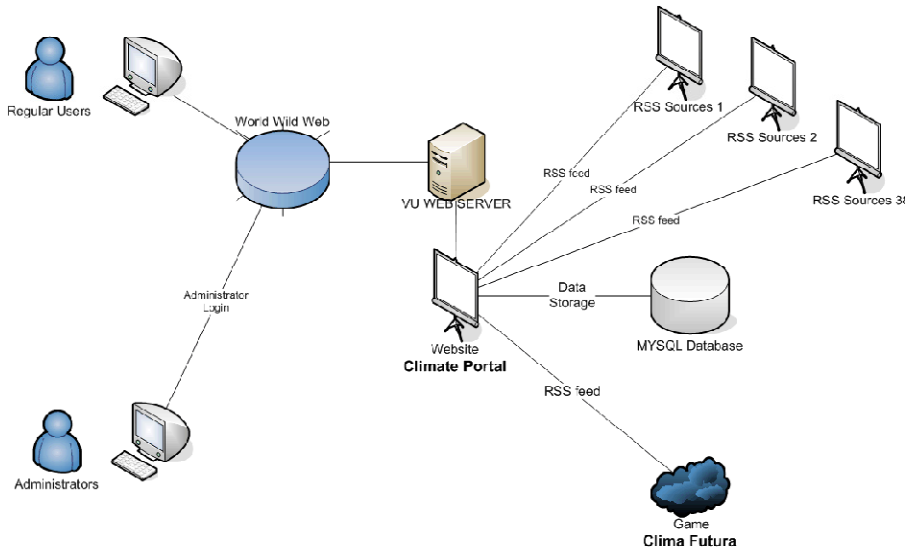

Figure 9 Deployment diagram of the Climate Portal (image taken from the Climate Portal technical document)

[10] http://www.climateportal.nl/
[11] http://www.rssboard.org/rss-specification

The climate portal aggregates the RSS feeds from climate related websites, determines the category of each item in the feed by analyzing the text, and stores the items in the database (Figure 9). The way content is added to the climate portal is to a certain extent automated since data is retrieved from a number of different RSS feed providers and subsequently automatically stored in the database. However, this aggregation process itself has to be started manually. Users are able to browse and search through articles, videos, images, and audio files by consulting the climate portal. In addition users can rate items in terms of quality and whether they agree with the content of an item.

6.2 Climate portal web service

6.2.1 Web services

One of the goals of the multimedia casus 2007 course was to develop a web service from which Clima Futura can obtain materials. What exactly is a web service? The W3C[12] defines a web service as:

"…a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards."[13]

This definition is already very specific as it states that the interaction between systems takes place using SOAP messages. There are however other methods available like REST or XML-RPC. Reinheimer (2006) gives a broader definition:

"Web services are a collection of protocols that are used to exchange data between disparate applications or systems. The essence of web services is the open standards on which they are built, by leveraging public and common protocols like HTTP, along with the XML document model."

[12] The World Wide Web Consortium. http://www.w3.org/
[13] World Wide Web Consortium glossary – Web service. http://www.w3.org/TR/ws-gloss/

Considering both definitions, the fundamental aspect of web services is the interoperation between different applications or systems. In other words, the main reason to use web services is because you require information from someone or you want to provide information to others.

Taking the broad definition of Reinheimer into consideration, web services include not only web APIs but also web feeds. The climate portal web service is at the time of writing implemented as an RSS web feed. This RSS feed is however not in particular aimed at providing Clima Futura with materials but more at providing a news feed for a general audience. The approach is nevertheless different from regular feeds which are relatively static. Based on a request by the user, the climate portal web service is able to return a custom RSS feed. Using a PHP script the service can select articles from the database by category, curtail the amount of articles by an arbitrary number, and return information about the articles as an RSS feed. Due to this interaction one could argue this service can actually be classified as a web API instead of a web feed. However in general web APIs are constituted in a more formal manner using for example the Web Services Description Language (WSDL) and a protocol like SOAP (Reinheimer, 2006).

Both WSDL and SOAP are W3C recommendations[14] [15] and are used to formally describe web services. WSDL is a XML based language for **describing** web services and how to access them. SOAP is a protocol for **accessing** web services, also XML based. It is beyond the scope of this thesis to go into the exact details of these specifications. More important is to determine what the climate portal web service should ideally offer to Clima Futura in terms of materials and related metadata.

The climate portal web service (read: RSS feed) currently deployed at the time of writing offers limited information about materials. The RSS feed consists of items which in turn consist of the title of an article, a link to a page with more details about an article, and a short description of the article. It goes without saying that a game or interactive video application has limited use for such information. A game or interactive video application would in general benefit more from information about images and videos because these are two types of assets inherent to game-like applications. That being said, information about articles is not completely irrelevant. It might for instance be interesting to incorporate certain texts, e.g. news headlines, into a game or interactive video application. What is however important to consider is the context of the to be

[14] W3C recommendation is a maturity level that represents an end state in the development process of a technical report. It means the specification has received the endorsement of the W3C members and the director. http://www.w3.org/2005/10/Process-20051014/tr.html#Recs
[15] http://www.w3.org/TR/#Recommendations

integrated materials. In other words, how is it possible to dynamically request appropriate materials to suite the game state/narrative at a particular point in time?

To accomplish this, a system that can determine what kind of materials need to be obtained at a particular point in time and in a certain context is needed. In addition, materials need to be sufficiently annotated to satisfy the system's requests, i.e. to allow repurposing of materials they need to be described in such a way that they can play a different role in each different context. However, in many cases this will work the other way around; based on the possible level of annotation of the materials the system is able to make requests to obtain the appropriate materials.

6.2.2 Annotation of multimedia assets and related problems

Annotation of media assets is a complex subject and there are many problems related to it as pointed out by Geurts et al. (2005) and van Ossenbruggen et al. (2003). Some of the problems discussed are more relevant than others with respect to the climate portal and the relation with Clima Futura, but all represent issues that have to be taken into account when setting up a system that involves media assets and accompanying metadata.

One of the problems is the time consuming process of annotating materials. A low level of annotation can be accomplished by using for instance automatic text analysis to extract metadata. This automatic extraction process occurs, to some degree, at the climate portal when RSS feed items are matched against certain keywords to determine the category of an item. However, in many cases a higher level of annotation is desired but this will often require manual annotation by humans. Moreover, annotation of image, video, and sound materials is even a more complex subject as metadata is more difficult to automatically extract from these types of assets when compared to the extraction of metadata from texts. Again the consequence will be annotation performed by humans and this in turn will result in highly subjective results.

Thus subjectivity is another problem related to metadata, especially the metadata of multimedia materials, as materials may be interpreted differently by different annotators. In the case of multimedia materials the problem stems from the numerous perceptional, cognitive, and cultural codes imbedded in the materials causing varying interpretations (van Ossenbruggen et al., 2003).

Going back to the climate portal and the relation with Clima Futura, the question arises what the climate portal should or can offer in terms of materials and how these are annotated. Here the distinction between the description of content and the description of technical aspects of a material has to be made. A system that presents different types of media will at least need some technical information about the media items to process them in a proper manner. One can think about specifications like the dimensions of an image, the duration of a video, etc. Also information like the name of the author, the creation date, etc. belong to this category. These type of descriptions are simpler to construct in a formal fashion than the description of the content of materials.

There are two main problems regarding content descriptions: complexity and size of the annotation, and granularity of the annotation (Geurts et al., 2005). Content descriptions of multimedia assets tend to get bulky due to the nature of multimedia itself; multimedia asset are very rich in the sense of (visual) information they contain, resulting in corresponding complex descriptions if one tries to annotate them in a formal way. Geurts et al. suggest to use a lightweight and extensible multimedia core vocabulary. The core consists of a number of often used properties sufficient for most applications but should be extensible to add more domain specific properties if needed. Using a small but extensible core can reduce the aforementioned bulkiness and complexity problems many existing vocabularies have. Then there is the issue of granularity. Video and audio are temporal types of data. In other words, the relation between space and time is important to the meaning of a particular audio or video fragment. Incorporating this relationship into metadata is a challenging task and subject of ongoing research[16].

6.2.3 Annotation of temporal data

Davis (1995) developed an iconic visual language for representing video that enables content-based annotation, retrieval, and repurposing of video called *Media Streams*. He took the granularity issue into account when developing the language by ignoring the use of keywords to represent video content and instead chose a stream based representation of video in which not only semantics can be better represented, but it also allows for dynamically re-segmenting, retrieving and re-sequencing of video. Davis lists a number of reasons why the keyword approach is inadequate for representing video content. This list might be useful to consult when one

[16] For an example of extensive research on video annotation see Li (2004)

considers to employ a keyword approach in a project (see Chapter 5.2) as it addresses several issues regarding keyword annotation of temporal data:

- Keywords do not describe the temporal structure of video information.
- Keywords are not a semantic representation; they do not support inheritance, similarity, or inference between descriptors.
- Keywords do not describe relations between descriptors.
- Keywords do not converge. A keyword might have a different meaning for different annotators. This can in turn result in subjective and non-convergent descriptions.
- Keywords do not scale. As the number of keywords grows, the possibility of matching a query to the annotation diminishes.

The goal of Davis' *Media Streams* was to create a durable and shareable representation of video content to ultimately support the use of a global media archiving system. The stream-based representation of the video is accomplished by segmenting the video data by means of applying multi-layered annotations with precise beginning and ending points in the video stream (Figure 10). The advantage of such an annotation system is that it leaves the video stream itself intact and in turn multiple segmentations can be generated from an annotated video stream.

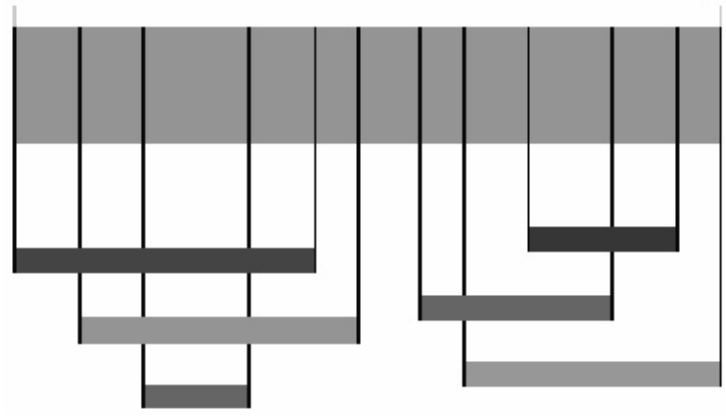

Figure 10 Video stream with 6 annotations (Davis, 1995)

The annotation in *Media Streams* is done by employing an iconic visual language. The iconic primitives of the language can be combined to form descriptors which are subsequently used to annotate the video content. These iconic primitives are grouped and structured to deal with the special semantic and syntactic properties of video data as described by Davis. What is interesting about *Media Streams* is that the annotation and corresponding segmentation of video really

supports the repurposing of video materials. Because of the stream-based representation of video content the system is able to compose new sequences out of different videos based on a query while at the same time taking temporal and semantic structure of video into account. On the other hand, annotation of video using *Media Streams*, or for that matter any other method to describe video content, is a time consuming task and most of it has to be done by humans. The approach of the climate portal at the time of writing is more automated with regard to the annotation of materials. However, this annotation is very limited and the materials' metadata lack the details a vocabulary or language like *Media Streams* could provide.

6.2.4 Pipes

Pipes[17] is an online tool to aggregate and manipulate content from around the web. Since the climate portal aggregates RSS feeds from a number of different sources, Pipes might be a helpful tool to strip out irrelevant or duplicate content from these feeds.

Just like Unix pipes, Pipes gives one the ability to string together a number of processes for a specific output. By using the visual editor it is possible to construct pipes by routing the output of a module to the input of another module. There are a number of different modules each with their own functionality. Usually a pipe starts with a data source module that grabs data from a particular source on the internet. Next it can be routed to another module that transforms or filters the data. Eventually the data can be outputted in RSS, JSON, or Atom formats.

In what way can pipes be useful for the climate portal? As a true portal application the climate portal present information from different sources separately. In contrast, so called 'mashup' applications take data from multiple sources and combine it to create something new that was not originally provided by either sources (Blake & Nowlan, 2008). With a tool like Pipes one can create 'mashups' in a relatively easy way[18]. The 'mashup' principle can be applied to the climate portal by taking for instance climate related news articles, looking up latitude and longitude coordinates of place names found in the news articles and ultimately presenting the news on a map.

An example of how to make a relevant pipe can be found in Appendix C.

[17] http://pipes.yahoo.com/
[18] There are numerous other 'mashup' editors available on the web. Microsoft Popfly (http://www.popfly.com/) and Google Mashup Editor (http://editor.googlemashups.com/) are some examples.

6.3 Summary and future work

The following is a short summary of this chapter that actually reflects the ways in which the climate portal can improve its potential to become a valuable service to Clima Futura or XIMPEL. The list is divided into short term improvements and improvement that are somewhat more difficult to accomplish and will therefore take more time.

Short term improvements:

- Setup a web service in a formal way, by using for instance SOAP and WDSL, in order to let other applications request data from the climate portal.
- Next to articles also offer images, videos, and audio from the climate portal to other applications.
- Consider using 'mashup' editors to aggregate, combine, and manipulate content from the web in a relatively easy way and incorporate this into the climate portal.

Long term improvements:

- Design Clima Futura in such a way that is aware of its current game state/narrative state to make it possible to dynamically request multimedia assets that fit in with this state.
- Annotate multimedia assets in such a way they can be dynamically requested by applications. Thereby taking into account the solutions regarding multimedia annotations as stated by Geurts et al (2005, page 7).

7. Towards a description format for serious games

In Chapter 2 the notion of serious games was discussed as background to the work regarding the Clima Futura project. Next a start was made on the construction of a description format based on the description of the Clima Futura prototype in terms of game design patterns. Obviously this effort was focused on one particular application – the Clima Futura prototype. This chapter however tries to investigate the characteristics of serious games in general instead of focusing on one application.

By exploring the characteristics and issues of serious games we can attempt to derive general topics to take into account when designing a serious game and an accompanying description format. In addition we can formulate interesting questions which can be the starting point for future research/work on serious games.

7.1 Learning game characteristics

In Chapter 2 we have seen there is currently not a singleton definition of serious games. In general serious games are associated with games for purposes other than mere entertainment. Often the goal of a serious game is to learn the player something and thereby including all aspects of education – teaching, training, and informing – in contrast with edutainment which seems to be limited to only teaching facts and rote memorization. In this regard it is interesting to look at the four key characteristics of a learning game as defined by Malone and Lepper (1987):

- **Challenge** – Having clear goals that are relevant to the player creates challenge. In addition challenge can be provided by incorporating uncertain outcomes due to variable difficulty levels, hidden information, and randomness. Frequent feedback can give the player clarification on how to succeed in the game and promote feelings of competence.
- **Curiosity** – There are two forms of curiosity. Sensory curiosity is triggered by audio and visual effects in (digital) games. Cognitive curiosity is aroused when players are surprised or intrigued by paradoxes or incomplete or inconsistence information.
- **Control** – When players have the ability to make choices that produce powerful effects it increases their sense of control.

- **Fantasy** – The context of the game includes some degree of fantasy, which engages the emotional needs of learners while providing relevant metaphors or analogies. Fantasies should also have an integral relationship to the content of the game.

These characteristics or principles can provide, next to valuable guidance, criteria by which the quality of a game as a learning method can be measured. Schaller (2005) investigates how these characteristics manifest themselves in two game examples. In the first game (a game about pest control) he notices that the convergent storyline of the game limits the game's potential. He argues a non-convergent storyline would increase the replayability and this in turn signals the control a game can gives its players. The second game (a game about the behavior of fish) has a more open-ended approach in which the player has more control but also more responsibility. In other words, the player has to make the connection between the gameplay and underlying concepts to be learned. By playing the game several times and evaluating it afterwards, students were able to understand some of the ecological concepts that were designed into the game. With this in mind Schaller suggests to add two additional criteria for an effective learning game next to Malone & Lepper's criteria:

- **Iteration** – Iteration can consist of replaying the entire game or of small iterations of a particular part of the game. It can support the learning process by encouraging experimentation, hypothesis testing, and synthesis. The content of a non-open ended game should be sufficiently engaging so it encourages replaying it.
- **Reflection** – Reflection during iterations can help players gradually develop an understanding of the subject at hand. This is however a difficult issue and the question is how reflection can successfully be built into the gameplay experience itself. An interesting concept in this regard is that of *Just in Time* information. This is about providing a user with only content necessary at the right time and place by anticipating the user's needs based on their immediate context within the overall experience[19].

Asgari & Kaufman (2004) state that good games are intrinsically motivating and engaging and that past research has shown that curiosity and fantasy (see Malone & Lepper's taxonomy above) are the features responsible for this. As mentioned before curiosity can be evoked by incorporating incomplete or inconsistence information. Fantasy can provide imaginary context, themes, or characters where players can experience situations they would not experience in real

[19] See Guerra & Cianchette (2006) for more about *Just in Time* information.

life. By satisfying an individual's needs a fantasy context can increase motivation. But as different individuals find different fantasies interesting it should be possible to create fantasies suited towards specific needs.

In a learning environment the fantasy context should be inherently related to the material to be learned. Malone & Lepper believe these so called endogenous fantasies are preferred to exogenous fantasies in where the relationship between the content and the fantasy is arbitrary. In endogenous fantasies the feedback to the learner is constructive as it reflects the problems related to the skill(s) learned in the activity.

7.2 Actors and game structure

What we have seen is that feedback to the player is important in achieving the learning goal. For teachers, trainers, administrators, or company management feedback about player's behavior could also be valuable. Therefore the game would have to track player behavior, assess their ability, and report the information. In cases where exact details are desirable the player's actions can be recorded and played back later for analysis. If we look further into the different roles related to serious games we can dissect four groups:

- **Developers** – Since serious games are developed and used for other purposes than mere entertainment, game developers will need to create software that employs game design principles and techniques but that is different from games they would normally develop. Serious games should provide features to tailor the game in order to apply to different uses, i.e. learning environments. Game developers should submit to the need of trainers – flexibility in scoring design, scenario design, challenges, and goal structures – and let them interpret the final outcomes and goals for players (Sawyer, 2002).
- **Domain experts** – Domain experts and learning designers should work closely with game developers. Game developers should accommodate domain experts with tools to transfer their knowledge into the game mechanics and game content. This is why an adequate description format is important.
- **Teacher/trainers/administrators** – As mentioned before this group should be able to adjust the game to the particular learning opportunity or simulation situation at hand. Modularity is important so that the game structure can be reused in different contexts.

From a financial point of view this is also interesting for the client as this modularity aspect maximizes the return on the investment.

- **Users/players** – The role of the player will often be limited to playing the actual game. However players might get the opportunity to configure small aspects of the game session; difficulty level, which part of the game they want to play, etc. In some cases player can be given the opportunity to add game content or alter certain aspects of the game. This activity will however need to be moderated.

By analyzing the general structure of a game the fundamental parts of a description format can be dissected as this format should be intrinsically related to the game itself. The game design patterns framework discussed in Chapter 2.1.2 provides some components to include. In addition Derryberry (2007) lists a number of common game attributes (although the hierarchy of attributes/components is not specifically mentioned):

- **Back story/story line/narrative** – The story is the rationale for the gameplay. It gives players the motivation for the existence of goals and challenges in the game and rewards for completing the goals by incorporating the consequences of players' actions into the unfolding narrative. Thus a structure to represent narratives is important in a description format.
- **Game mechanics** – These handle all the specific functions within a game and produce the actual gameplay. The interaction of mechanics in a game determines the complexity and level of player interaction. Parameters of game mechanics can be included in a description format for configuring the game to suit a specific context or target group.
- **Game Rules** – Limit the player actions and lay out the boundaries of the game. Rules can be implicit or explicit, they are either known to the player or hidden in the game system. Also game rules parameters can be included in a description so they can be tweaked or be made explicit/implicit.
- **Immersive environment** – The sensory representation of the game elements. This includes graphics, sound, and animation. A semantic approach is desirable if the material needs to be successfully repurposed in different contexts.
- **Interactivity** – What is the impact of the player's actions on the game world?
- **Challenge** – Challenge is important to keep the player interested and motivated. What are the goals the player tries to achieve?

- **Risks and consequences** – In striving to achieve the goals of the game a player faces both advantageous and disadvantageous effects of his/her actions. The variables to include in a description format are the probabilities of getting the reward or receiving the penalty. In essence this is actually part of the game mechanics and game rules.

This list may be somewhat arbitrary when compared to the well substantiated game design patterns framework. In is essence both describe how a game is constructed. From this the fundamental elements of a description format can be acquired. In this process and the overall game design the issues related to serious games that have been explored in this chapter have to be kept in mind:

- Provide appropriate metaphors and analogies for learning.
- Relate the fantasy to the content to be learned.
- Offer different fantasy context for different kind players (customization).
- Provide ways of giving reflection to players and feedback about player behavior to teachers/trainers.
- Incorporate iterations as this is vital to learning.
- Flexibility and modularity are important. Developers should construct systems in such a way that teacher/trainers and domain experts can configure certain aspects of the game.

The list above is not exhaustive as there are many other areas and aspects related to serious game to be studied. For example we have mainly discussed the learning aspect of serious games, but serious game are not always about learning and training (Sawyer, 2007). Sawyer notices that games based on goals other than education, for example health related therapies, public opinion research, and economic studies, have enjoyed successes. He argues games that act more like utilities offer a lot of promise for the future of serious games.

To return to the subject of constructing a description format for serious games, we stated that the layout of a game structure is a good starting point for this process. How the actual game design translates into a description format is for the most part depended on the specific implementation of the game itself as we have seen in Chapter 4.

8. Conclusions

Having discussed a varying range of topics, with Clima Futura and serious games as the overarching subjects, we can now draw conclusions from the different parts of this thesis.

First the notions of game design patterns and serious games have been discussed. It became clear that although game design patterns have their roots in the design patterns as described by Alexander back in the 1970's, the approach of game design patterns is distinctly different from that of 'regular' design patterns. Design patterns are solely focused towards problem solving as opposed to game design patterns which are not based on problem solution pair template but are a tool to analyze, describe, discuss or design games.

As Clima Futura is classified as a serious game it was necessary to explore this subject. What we have learned is that although serious games gradually are becoming more popular, there is still skepticism and not yet a consensus about what serious games exactly are. That being said, there are great opportunities for serious games to become an accepted educational tool in the future as research in human learning shares similarities with methods that are being used by game designers to get player to learn (in) their games.

In Chapter 4 the Clima Futura prototype is described in terms of game design patterns. As it turned out game design patterns as a tool to analyze and describe game design works quite well. Most of the element of the prototype game design could be described and along the way new ideas arose. We have attempted to create a description format for the Clima Futura prototype based on game concept elements offered by several game design patterns. Eventually the game design patterns could not provide a solid base for an exhaustive description format as the creation of such a format is in many ways different from the intended use of game design patterns. Some inspiration was drawn however from the patterns by which the Clima Futura prototype was described and a rudimentary description format was created. We have also briefly looked at the XML configuration editor prototype of which the goal was to show how to make a simple editor based on the cut-and-paste principle to support the construction of a configuration file.

Shifting to a new subject regarding Clima Futura, Chapter 5 introduces the notion of interactive narratives. More in particular we have explored ways how the existing branching structure

approach of Clima Futura can be altered towards a content-based approach in where the author does not explicitly links the material.

Having discussed several aspects related to the climate portal we have presented a short list of future improvements to the climate portal. This is done by discussing the various related subjects as web services, annotation of multimedia assets, aggregation of information on the web, and summarizing this into the aforementioned list of improvements.

In Chapter 7 we have looked into the characteristics of serious games and more in particular the characteristics of learning games. Challenge, curiosity, control, and fantasy represents essential features by which the effectiveness of a game can be measured. Another important aspect related to serious games is the distinction between developers, domain experts, and users and what their roles are regarding the use of description format. We have stated that the game designs patterns **framework** can play a valuable role in the construction of a description format for (serious) games as it describes the basic layout of a game.

During the time this thesis was written the work on Clima Futura as a game gradually shifted towards the work on the extensive interactive media player XIMPEL. It was therefore sometimes difficult to focus on Clima Futura as a game and explore it exhaustively. Despite being somewhat wide in scope, the results of this thesis can however offer valuable advice for future development of Clima Futura as a game or serious games in general for that matter.

References

Alexander C. (1977), A Pattern Language: Towns, Buildings, Construction. Oxford University Press.

Asgari M., Kaufman D. (2004), Relationships Among Computer Games, Fantasy, and Learning. Retrieved from http://www.ierg.net/confs/2004/Proceedings/Asgari_Kaufman.pdf

Birmingham J. (2008), Yay!: A moving time supplement (An In-Depth Examination of My Own Mag). Retrieved from http://realitystudio.org/bibliographic-bunker/my-own-mag/yay-a-moving-times-supplement-an-in-depth-examination-of-my-own-mag/

Beck K., Cunningham W. (1987), Using Pattern Languages for Object Oriented Programs. Submitted to the OOPSLA-87 workshop on the Specification and Design for Object-Oriented Programming.

Berge ten J. (1997), Droommachines: Nieuwe media als nieuwe roesverwekkers. Kunstlicht 18 2/3, pages 22-27.

Björk S. & Holopainen J. (2005), Patterns in Game Design. Charles River Media.

Björk S., Lundgren S., Holopainen J. (2003), Game Design Patterns.

Blake M.B., M.F. Nowlan (2008), Predicting Service Mashup Candidates Using Enhanced Syntactical Message Management.

Bordwell D., Thompson K. (2001), Film art: an introduction.

Davis M. (1995), Media Streams: An Iconic Visual Language for Video Representation.

Derryberry A. (2007), Serious games: Online Games for Learning. Retrieved from http://www.adobe.com/resources/elearning/pdfs/serious_games_wp.pdf

Egenfeldt-Nielsen S. (2005), Beyond Edutainment: Exploring the Educational Potential of Computer Games.

Eliëns A., van de Watering M., Huurdeman H., Bhikharie W., Lemmers H., Vellinga P. (2007), Clima Futura @ VU, communicating (unconvenient) science.

Gamma E., Helm R., Johnson R., Vlissides J. (1995), Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley.

Gee J.P. (2003), What Video Games Have to Teach Us About Learning and Literacy. New York: Palgrave Macmillan.

Gee J.P. (2004), Learning by Design: Games as Learning Machines. Retrieved from http://www.ub.edu/multimedia/iem/down/c8/Games_as_learning_machines.pdf

Genette G. (1980), Narrative Discourse. Cornell University Press.

Geurts J., van Ossenbruggen J., Hardman L. (2005), Requirements for practical multimedia annotation. Retrieved from http://homepages.cwi.nl/~media/publications/MMSemweb2005.pdf

Guerra J.A., Cianchette C. (2006), Just-in-Time Information: A Layered Approach to Information Design.

Huurdeman H.C. (2007), Interactive Video in Serious Games: Exploring the Use of Interactive Video for Clima Futura.

Jefferies D. (2007), Games which aren't fun but are useful. The Guardian, Thursday October 18 2007. Retrieved from http://www.guardian.co.uk/technology/2007/oct/18/news.games

Kuitenbrouwer K. (2005), Emergent narrative. Retrieved from http://www.mediamatic.net/page/9529/en

Li H. (2004), Hierarchical Video Semantic Annotation – The Vision and Techniques. Retrieved from http://www.ohiolink.edu/etd/send-pdf.cgi/Li%20Honglin.pdf?acc_num=osu1071863899

Lundgren S. (2006), Facets of Fun: On The Design of Computer Augmented Entertainment Artifacts.

Malone T.W., Lepper M.R. (1987), Making Learning Fun: A Taxonomy of Intrinsic Motivations for Learning.

Murtaugh M. (1996), The Automatist Storytelling System: Putting the Editor's Knowledge in Software. MIT Media Lab.

Ochella B. (2007), Who Says Video Games Have to be Fun? The Rise of Serious Games. Retrieved from

http://www.gamasutra.com/view/feature/1465/who_says_video_games_have_to_be_.php?page=1

Van Ossenbruggen J., Nack F., Hardman L., (2003), That Obscure Object of Desire: Multimedia Metadata on the Web (Part 1). Report INS-E0308.

Reiheimer P.M. (2006), Professional Web APIs with PHP: eBay, Google, Paypal, Amazon, FedEx, plus Web Feeds.

Riley S. (2007), Business, Government Use Video Games to Explore Serious Issues. Investor's Business Daily.

Sawyer B. (2002), Serious Games: Improving Public Policy through Game-Based Learning and Simulation. Retrieved from http://wwics.si.edu/subsites/game/Serious2.pdf

Sawyer B. (2007), Ten Myths About Serious Games. Retrieved from http://www.escapistmagazine.com/articles/view/issues/issue_121/2575-Ten-Myths-About-Serious-Games

Sawyer B., Smith P., (2008), Serious Games Taxonomy. Slides from the Serious Games Summit at the Game Developers Conference 2008.

Shalloway A., Trott R.J. (2004), Design Patterns Explained: A New Perspective on Object Oriented Design. Addison-Wesley.

Schaller D. (2005), What Makes a Learning Game? Retrieved from http://www.eduweb.com/schaller-games.pdf

Squire K. (2005), Game-Based Learning: Present and Future State of the Field.

Terdiman D. (2006), What's wrong with serious games? CNet News.com. Retrieved from http://www.cnet.com.au/games/0,239029232,240061367,00.htm

Van Vliet H. (2000), Software Engineering: Principles and Practices (Second Edition). John Wiley & Sons, Ltd.

Appendix A

Definitions

Atom – An XML based Web syndication format. *(See Web feed)*

Clima Futura – A game about climate change.

Design pattern – A formal way of documenting a solution to problem.

Edutainment – A form of entertainment designed to educate as well as amuse.

Game – A system in which players engage in artificial conflict, defined by rules, and resulting in a quantifiable outcome.

Game design patterns – Semiformal interdependent descriptions of commonly reoccurring parts of the design of a game that concern gameplay.

Gameplay – The structures of player interaction with the game and with other players in the game.

Granularity – The level of detail that characterizes an object or activity.

Graphical user interface – A user interface based on graphics which allows people to interact with an electronic device.

HTTP – Hypertext Transfer Protocol. A communication protocol used for retrieving inter-linked text documents.

Hypermedia system – A computer-based information retrieval system that enables a user to gain or provide access to texts, audio and video recordings, photographs, and computer graphics related to a particular subject. The links between the materials are fixed.

JSON – JavaScript Object Notation. A human-readable format for representing simple data structures.

Keyword – A term that captures the essence of a piece of information. Commonly used as *tags* on the World Wide Web.

Mashup – A web application that combines data from multiple sources to create a new and distinct web service. *(See Web service)*

Mini game – A small and often short game within a game.

Narrative – A chain of events in cause-effect relationship occurring in time and space.

PHP – General purpose scripting language especially suited for web development.

Prototype – An original type, form, or instance serving as a basis or standard for later stages.

Replayability – The extent to which it is valuable to play a game more than once.

Replay value – *See Replayability*

REST – Representational State Transfer. A collection of network architecture principles which outline how resources are defined and addressed.

RSS – Really Simple Syndication. The RSS format allows web publishers to syndicate content quickly and automatically.

Serious games – Software application designed with the use of game technology and game design principles for purposes other than mere entertainment.

SOAP – Simple Object Access Protocol. A lightweight protocol intended for exchanging structured information in a decentralized, distributed environment.

Web API – An application programming interface that can be accessed over a network and execute requested services.

Web feed – A way to distribute frequently updated content. Content distributors syndicate a web feed, thereby allowing users to subscribe to it.

Web portal – A website that presents information from different sources in a unified way.

Web service – Web services are a collection of protocols that are used to exchange data between disparate applications or systems. The essence of web services is the open standards on which they are built, by leveraging public and common protocols like HTTP, along with the XML document model.

Web syndication – *See Web feed*

XIMPEL – The eXtensible Interactive Media Player for Entertainment and Learning.

XML – Extensible Markup Language. It facilitates the creation of custom markup languages to structure documents.

XML-RPC – A remote procedure call protocol which uses XML to encode its calls and HTTP as a transport mechanism.

WSDL – Web Services Description Language. An XML format that provides a model for describing Web services.

Appendix B

XML configuration editor prototype

The prototype of the XML configuration editor helps users to generate the XML code of the different configuration elements as described in Chapter 4.3. The graphical user interface is divided into three parts (Figure 11). On the left side the list of configuration elements is displayed. In the middle the different fields are displayed for the user to fill out. On the right side the generated code is displayed whenever the 'insert data' button is clicked.

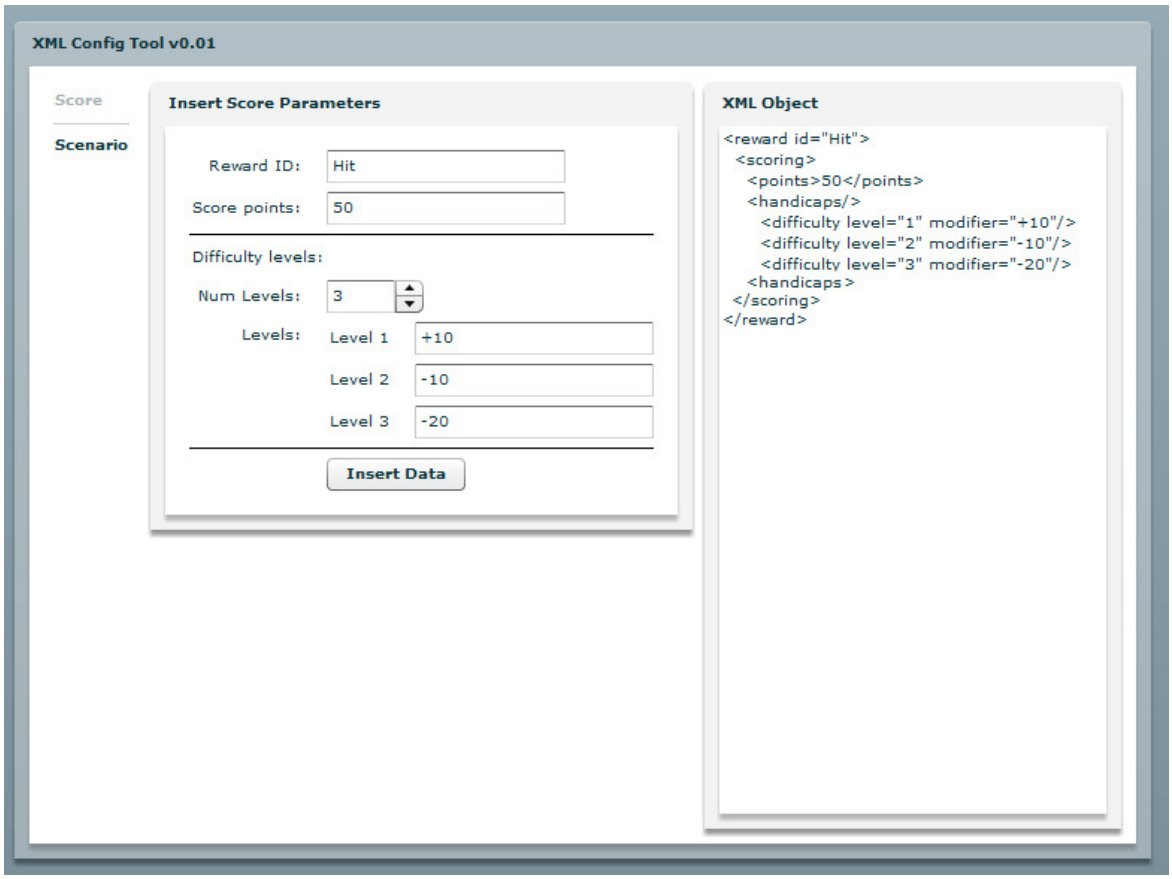

Figure 11 Graphical user interface of the XML configuration editor prototype.

After a user has selected an element from the list on the left and filled out the necessary fields he/she has to click 'insert data' to generate the XML code. This code can then be copied and pasted into the actual configuration file.

This prototype only contains two configuration elements, but more elements can be added in the future if this is desirable. The goal was to show how to make a simple editor/code generator that can support the construction of a configuration file.

Based on this prototype S.V. Bhikharie made an similar application to generate playlists for XIMPEL (Figure 12).

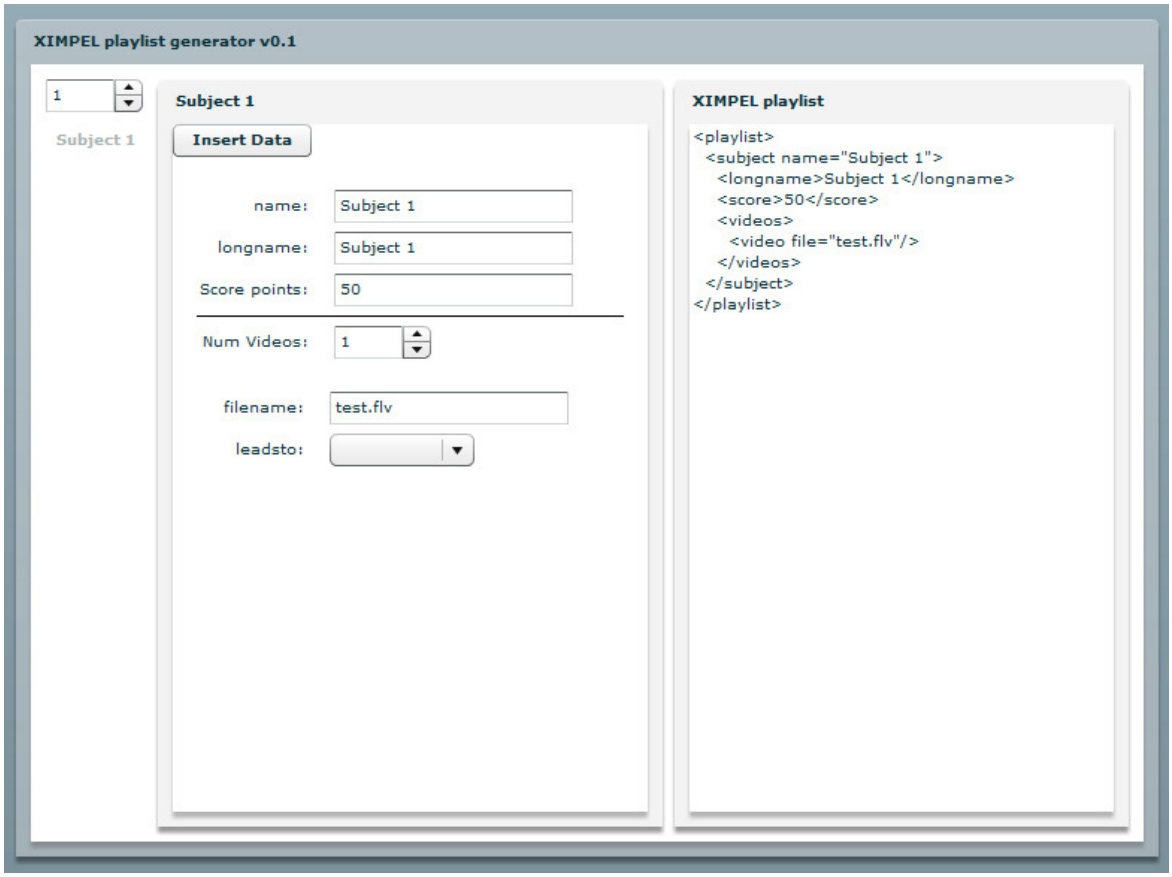

Figure 12 XIMPEL playlist generator v0.01

The overall approach of this application is the same as in the prototype mentioned before. The main difference is however that a user can create new subjects on the left side of the graphical user interface. These subjects can then be linked to each other in the 'leadsto' field of each subject. This way paths between videos can be constructed essentially constituting a simple narrative structure.

Appendix C

Pipes

Pipes is an online tool to aggregate and manipulate content from around the web. See Figure 13 for an example of a simple but effective pipe.

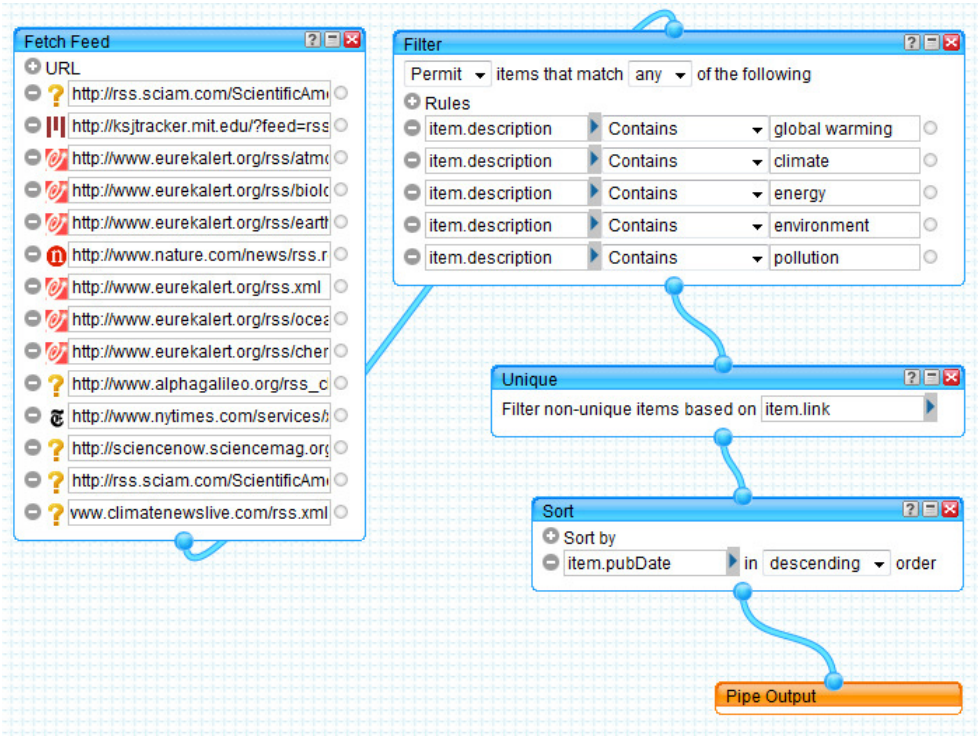

Figure 13 'Climate news': An example of a simple pipe

First all the feeds that are listed in the 'Fetch Feed' module are aggregated. Next the feed items are filtered. Only feed items with one (or more) of the specified words in their description field are let through. Then any duplicates are removed and the list of feed items is sorted by publishing date before it gets outputted as an RSS feed. As previously mentioned this is a very simple pipe. The following pipe is slightly more advanced as it combines multiple services to create something new. The RSS feed from the previous pipe (Figure 13) is used as the starting point of this new pipe (Figure 14). The RSS feed from Figure 13 is sent to the *'Geonames RSS to GeoRSS converter'* web service. This converter reads the entries of the RSS feed and searches the

Geonames[20] database to find a location for the entry text. If a location is found its latitude and longitude is added to the feed. The feed that returns from the *Geonames* web service is then fetched and sent to the location extractor module[21]. This module examines the input feed for any location data. If it finds geographic data, a 'y:location' element which contains the found data is added to the feed. The addition of this element makes it possible to display the results on a map later on.

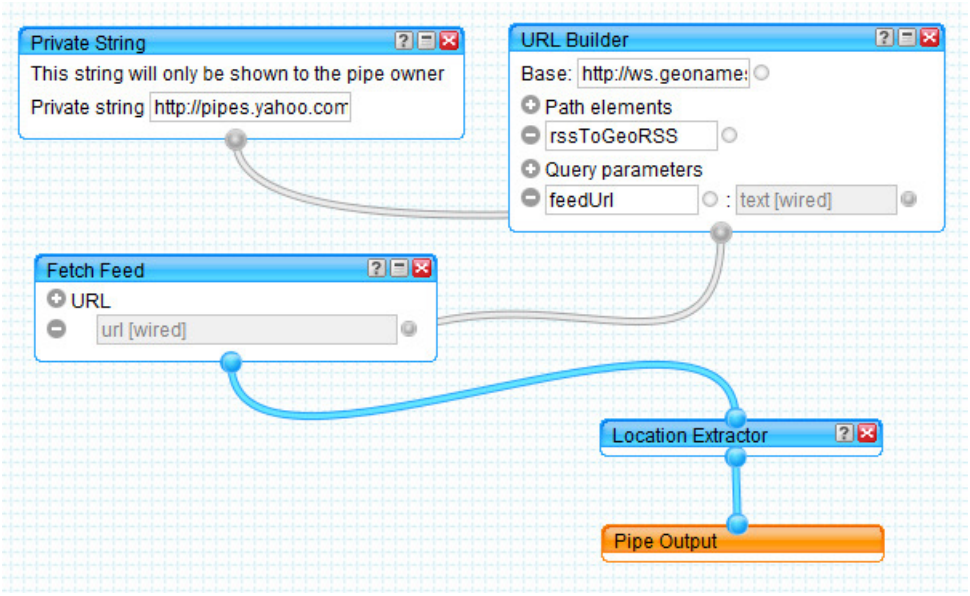

Figure 14 The pipe from Figure 12 combined with the *Geonames'* *'RSS to GeoRSS converter'* web service

The pipes' output can now be displayed on a map (Figure 15). This map could for instance be imbedded on the climate portal (see Chapter 6) to give users a sense of where news has happened.

[20] http://www.geonames.org
[21] For a complete list of available Pipe module see http://pipes.yahoo.com/pipes/docs?doc=modules

Figure 15 The map with climate related news