GDD as a communication medium
Design of the structure and communication

Sergio Gayoso Fernández

ii

KTH School of Computer Science and Communication
SE-100 44 Stockholm
SWEDEN

Abstract

Major gaming companies in Sweden are becoming reluctant to integrate the Game Design Document (GDD) into their new development methodologies. We suggest that this is due to the time it takes to address the GDD's shortcomings, in a fast-paced gaming industry. We aim to solve some of these shortcomings with the design of a new GDD medium. The first step is a structure specially designed to support the GDD and enhance the communication between different designers and developers involved in game development.

In order to improve the structure of the GDD, we formulated a set of requirements based on a review of published criticisms, post-mortems reports, and recent technological developments. An iterative design process was conducted based on these requirements and resulted in a tailored design structure for the GDD. Finally, the validity of this design was tested with a user study. This thesis outlines the results of these studies.

A mis padres

Acknowledgments

This masters thesis would not have been possible without the guidance and the help of Annika Waern. I wish to express my most sincere gratitude to Kim Nevelsteen for his guidance, patience and encouragement throughout the whole development. His advices have been helpful not only for the development of this project, but also for the development of my own life.

I thank Jon Back, Marie Denwardm, Mela Kocher for integrating me in the game studio and treating me as an equal. I want to express my gratitude to the people who have read this masters thesis providing valuable feed-back: Elena Marquez, Pauline Spengemann, Ramin Toussi and Bas Peschier. I want to thank Ruby Naomi Michel for her support during the last weeks of work, this project would have taken a couple extra months without your help.

I also wish to thank the people who have been working with me in our chilly room at Mobile Life: Magnus Linell, Syed Naseh and Elena Balan. Thank you for your help and inspiration. My gratitude to the people working in Mobile Life: Stina Andrén, Johanna Mercurio, Javier Jiménez and Morvarid Kashanipour, for all the good times that we have shared during this months.

I want to thank Thomas Andersson and Sebastian Armonioso, lead designers from DICE, without your generous help this investigation would not have been complete.

I want to thank the Mobile Life Center for giving me the chance to explore the academic community and KTH for providing me an excellent education.

A mi familia por su apoyo incondicional durante todos estos años.

I want to thank my friends in Madrid and Stockholm who has share with me great moments during all these years of studies. *A Jorge Peña por pasión y arrojo a la hora de apoyarme con la creación de mi propio proyecto. A Roger Moret (del Olmo) simplemente por estar ahí durante todo este tiempo, aun estando a miles de kilómetros.*

Finally, I would like to finish explaining one of my favourite games. The game is called *'I have the ball'* and it is a riddle. The players should guess the rule that describe who has the ball at the end of the game. The only rule of the game is that: it starts when someone says *'I have the ball'* and at the end the players will have to guess who has the ball. Let's make an example: I have the ball, *ok?...* I pass it to...

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

In the early stages of the game industry games were developed by small companies or individual programmers and hobbyists. Games were simple and game design and game development were two independent processes. A single monolithic document was enough to to develop ideas for a game and transmit them to designers and developers. That document was called Game Design Document (GDD) and for long time *was* considered the development paradigm of the game industry.

Companies in the industry grew and likewise the games they developed. As the games became more complex, the GDD became more extensive. At one point, the game design and development began to take place simultaneously and interdependently. The GDD, therefore, gained even more importance, because it became a bidirectional communication medium between designers and developers. Nevertheless, many problems arose due to the use of improper mediums to hold the GDD. The use of monolithic documents led to complex and inaccessible GDDs. Quick updates were badly supported. Due to these problems, designers and developers lost their faith in the GDD. The GDD lost its role as a communication medium and became a simple document that everybody edited but nobody ever read.

The game industry has grown enormously since these early stages and game development has become a complex process where large teams of designers and developers work together to create a good game. Nevertheless, the mediums which hold on the GDD have not evolved as much. New collaborative editing technologies have arisen. Some of them have been tested and used to hold the GDD, as for example blogs and wikis. However, there is not yet a medium which has been specifically designed for the GDD. The aim of this master thesis is to design a tailored structure to hold the GDD and enhance its role as a communication medium between designers and developers.

1.2   Goal

The aim of this master thesis is to design a new structure tailored to hold the GDD and allow a wide variety of communication operations.

The structure must 1)define characteristics of the GDD and ways in which designers make use of it 2) allow users to exchange and discuss ideas through the GDD, and 3) be simple and easy to use to avoid overloading the work of game designers.

1.3   Delimitation

This master thesis presents an analysis of GDD requirements and the design of a structure that will hold the GDD. The design outlines a structure and its operations from a functional perspective and does not deal with the user interface or the interaction model of the GDD. Although the design takes into account possible software limitations, the design is not described in detail from a software perspective. During the gathering of requirements, a wide range of technologies were analysed with the aim of integrating them into the GDD. The integration of multiple technologies was assumed possible.

1.4   Project context

This master thesis is developed within the context of a larger project, whose objective is to design a new medium to serve as a GDD and facilitate communication between designers and developers involved in a game creation. The process of design and development of this project will take place in three distinct phases:

1 Design a structure that will hold the GDD and allow communication operations;

2 design the user interface and interaction model to facilitate communication and visualization of information; and

3 build a prototype implementing the results of the previous phases.

This master thesis undertakes the first phase of the project.

1.5   Research methodology

An iterative design process was used to design the structure to support the GDD. Design workshops were carried out during each iteration to explore new functionalities and test the feasibility of a prototyped structure to perform them. Design workshop participants were encouraged to propose new features and organize the content of the GDD based on their personal preferences and habits. New features

were tested at that very moment immediately through the prototype. Due to the open nature of the design workshops and the prototype, they could not be analysed in a quantitative way. Therefore, a qualitative baseline was adopted based on an interview and analysis of the behaviours and motivations of the participants during the simulation of the game design process.

1.6 Approach

The first stage involved a review and analysis of published criticisms and post-mortems of the GDD and game development process. This identified a number of key weaknesses of the GDD. In addition, I reviewed technologies that have been used to hold the GDD and technologies which could potentially hold it or have interesting features that could be integrated. This review phase, resulted in a collection of desirable requirements that the future design should fulfill.

The second stage involved the design of the structure. An iterative design process was used to define the design. At each iteration, design workshops were conducted to test the validity of the current design and a paper sketch was made to simulate the functionalities of the structure and explore its strengths and weaknesses. At the end of each iteration, design improvements were made based on the results of the workshop. The experience gained during the workshop also served to refine the design requirements. Therefore, both the design requirements and structure of the GDD were updated at the end of each iteration.

The third stage involved a user study carried out to validate the design structure. The user study consisted of two sessions, each of them divided into three phases that tested in-depth the three main feature categories of the structure. The final conclusions from this study are presented in Chapter 8. The final stage involved validating the designed GDD structure with a major game company in Sweden with notable success.

1.7 Contribution

The results of the user study were positive and confirmed that the design of the structure successfully fulfilled the collection of requirements. The first contribution from this master thesis is precisely that collection of requirements. New designs can be developed for the same or similar purposes based on that collection. The second contribution is the design structure of the GDD. This design will now be used in the next stage of the project to design a totally customized medium for the GDD. In addition, the design structure can be used for the design of a medium that is focused on a different design process to game design. Finally, the last contribution is a specific feature of the design called *Views*. *Views* were the turning point in the design. This feature could be replicated in other collaborative editing technologies, as for example wikis, blogs, and Google Wave. *Views* are described in the following chapters in detail.

Finally, it is important to remark that the results from this master thesis have been published as a working progress paper at the Games and Innovation Research Seminar [Nevelsteen and Gayoso(2011)].

Chapter 2

Background

> "A **game** is a system in which players engage in an artificial conflict, defined by rules, that results in a quantifiable outcome."
> [Salen and Zimmerman(2003)]

2.1 Game industry

The game industry is the economic sector involved with the development, marketing and sale of games. The game industry has developed in parallel with the software industry. In the 1980's, at the early stages of the game industry, the game development costs were minimal and games could be quite profitable. Games were developed by a single programmer or by small teams and could potentially sell thousands of copies. The development process was fast and companies could release several titles each year. The risk of investment was low with short pay-back-periods, and therefore, publishers were generous with benefits. Many game publishers started during this golden age in the industry, such as Capcom, Sierra Entertainment, Activision and Electronic Arts.

As graphics power and computing performance increased, so too did the size of the development teams, as a wider range of specialist developers were needed to address the new programming and graphical complications. The game development process was lengthened up to several years several years with budgets reaching millions of dollars for a single game. However, unlike in the case of software development, investment did not increase to meet the needs of the changing game development industry. This possibly explains why the video game industry never succeeded in controlling it's own productions [Larsen(2002)]. The game industry is also focused on developing user experiences, unlike the software development industry which focuses more on the specification of functionalities. The game concept is defined through sketches, images, videos and different media types [Hagen(2010)]. This makes the design and development of games more abstract and complex. The game industry is often referred to as interactive entertainment

[Adams and Rollings(2006)], because of its similarities with other media entertainment industries as for example the music recording industry.

2.1.1 Why is the game industry so special?

The game industry is considered unique because it is located in between the software and media entertainment industries, without being part either of them. The game industry uses a mixture of techniques and concepts borrowed from software development and the film industry [Björk et al.(2003)Björk, Lundgren, and Holopainen].

- In comparison with the software development industry, the game development process is still not clearly defined due to its abstract nature. Games are focussed on developing user experiences. Game development has no formal methods to specify the game features, unlike software development. Game designers use many different media types to describe the intended feeling and experience that a game will create for a player. This makes game development subjective and therefore more complex. In addition, game development has become a long and expensive process with the resultant games having a relatively short life. As games are based on inner experience of the players, success cannot be guaranteed before a game is released. This forces the game industry to maintain a frenetic pace in order to develop and release games, a pace that is not as common in the software development industry.

- In comparison with the entertainment industries, such as the music recording industry and the film industry, the game industry is more focused on technical aspects. Games are complex software systems, which need software engineering methods to be planned and specified. The creative side of the game design process commonly demands innovative technical solutions. It is common to develop new technologies to fulfil the requirements of a game. These highly innovative technical challenges are not as common in other entertainment industries and in combination with the rapid pace of the industry, make game development a demanding and challenging process.

2.2 Game Design

Game design is part of game development process and focused on defining the content, rules, background, gameplay, environment, storyline, and characters for a game. The documentation that describes the design of the game is an important outcome of the process. Unfortunately, game design is not a well defined set of instructions and processes. There is not a perfect formula that can guarantee that the game will be a success, just characteristics that are common to the games that have been a success previously. Game design is a creative process that requires artistic as well as technical skills. Due to the variety of disciplines required in the design, the process is challenging. Programmers, artists and designers with totally

different backgrounds have to work together in harmony to achieve the goal. Salen and Zimmerman describe game design:

> "**Game design** is the process by which a game designer creates a game, to be encountered by a player, from which meaningful play emerges." [Salen and Zimmerman(2003)]

2.3 Game Design Document

The Game Design Document (GDD) is the documentation that describes the game design for a game. There is no consensus about how this documentation must be supported and structured. The form, medium and use of the GDD differs between development teams and is dependent on the structure of the company. The way GDD is shaped, supported and used influences the development of the game. The GDD has even been considered as a game development methodology by [Nevelsteen(2008)], due to this mutual dependence between practices to manage the GDD and the game development process.

2.3.1 Purposes of the GDD

The GDD has not always had the same purpose. Its use depends on the needs of the company and its Lead Designer. The Lead Designer is generally in charge of defining the use and purpose of the GDD at the beginning of game development. Depending on its purpose the GDD can be used during the whole game development process or only during the concept phase; it can be an extensive document or a short and brief description of the features of the game. The common purposes of the GDD are:

- **Contract**: The GDD can be used as a written contract between different parties outlining what shall be implemented. In this case, the GDD is rarely modified during the implementation of the game, due to its nature as a static agreement between the publisher and developer companies.

- **Communication**: This is the most common use of the GDD. The purpose of GDD is to collect ideas, concepts and features of the game and it is used during the whole game development process. At the beginning of game development the GDD content contains brief descriptions that are further expanded and detailed as development progresses. In this case, the GDD is mostly defined during game design, nevertheless it is constantly updated during the following phases, especially during the implementation. The GDD is therefore a communication medium between the designers and developers.

- **Final documentation**: The GDD can also be used as the final documentation of what has been implemented in the game. In this case, the GDD stores only the final description of the game. Therefore the content is not

modified or updated as frequently as when used as a communication medium. Game designers include content into the GDD only when it is sure that will be part of the final game. For example, it will not include any features that were rejected or any description as to why they were not included in the final game. The GDD becomes a way to document the game development process, but is not used as a communication medium.

2.3.2   Assumptions about the GDD

This masters thesis focuses on the use of the GDD as a communication medium. The following assumptions were made in order to assure that the GDD is used as a communication medium during the game development.

- The purpose of the GDD is the communication of design concepts and ideas between the game designers and developers.

- The GDD stores the ideas and concepts of the game during the whole game development process.

- The addition of content in the GDD is not necessarily approved by the Lead Designer and can be subject to debate and discussion. Therefore, the content of the GDD can be modified.

- The company is assumed to have large design and development teams that could potentially be in separate locations.

2.3.3   The GDD in the Game Design Process

The GDD is used during the whole game development process. According to [Hagen(2009)], the current game development process in Sweden has four phases as outlined below:

- **Concept phase**: The Lead Designer/s create the first version of the GDD with original ideas. It may be a conceptual and incomplete document describing ideas of the game in brief. This information is used to define a pitch for the game. The pitch is a document that describes briefly the essence of the game to find funds for the game.

- **Pre-production phase**: The pre-production phase commences when the project has been approved. During this phase, the GDD is expanded by the designers to a level where it can guide the development team.

- **Production phase**: During the production phase the game is developed and designed simultaneously. The GDD is continuously changed, revised and expanded as game development progresses and changes in scope and direction are explored. The GDD is often referred to as a *"living document"* [Cohen and Bustamante(2009)], because of this ever changing state

- **Post-production phase**: Finally, at the Post-production phase, the GDD is used as the final documentation of what has been implemented.

2.3.4 Game Design purpose and audience

The GDD does not conform to any single predefined structure or model. Successful templates of the GDD are sometimes used from other projects. However, more often the GDD is structured and used depending on each individual company and Lead Designers. The GDD can be a single monolithic document or hold on a collection of sub-documents within it, according to [Adams and Rollings(2006)]. However, regardless of how the GDD is structured either in chapters within a monolithic document or in sub-documents, each section has a different purpose and audience. [Adams and Rollings(2006)] gives some examples of GDD sections with specific purposes:

- High Concept: a brief description of the game (just a few sentences) intended to describe the essence of the game.

- Pitch Concept: a short abstract intended to present the game's selling points and detail why the game would be profitable to develop.

- Character Design

- Game Script

- World Design

- Levels

- Flowboard

- Story

Traditionally, the lead designer used to be in charge of managing and supporting the content of the GDD. However, the content of the GDD is now authored or consumed by teams of designers that take part in game design and development. These teams may include the following roles:

- Game mechanics designer,

- Level designer,

- Script writer,

- 2D and 3D artists,

- Programmers,

- Sound engineers,

- Testers

Professionals involved in the game development process come from a wide range of backgrounds which makes misinterpretations and misunderstandings are common It is essential that communication within the development teams is constant and effective. This is one of the main challenges in developing a game and the GDD is one of the communication mediums that make it possible.

> *"A complex project involves many talented people. Skilled programmers and artists tend to have minds of their own. While you intend to create a horse, the artist may be envisioning a unicorn and the programmer a highly efficient camel. A good document ensures that you are all planning to make the same thing."[Freeman(1997)]*

2.3.5 Weakness of the GDD

The GDD's main weakness is that it easily becomes bloated [Laramee(1999)]. The designers can produce a great quantity of documentation just about the smallest feature or element. The documentation can grow too rapidly and prolifically that can become unmanageable and inaccessible. The larger the GDD becomes, the more likely is that it will never be read again. This tendency makes the designers lose their faith in the documentation and become reluctant to keep it up to date. It is common to think about the GDD as the final documentation of the project instead of a medium to carry out the game design. Abandon the use of documentation foments the misinterpretations that can negatively affect the development of games.

2.4 A living document

The starting point of this master thesis was based on a description of a medium for the GDD as a *"living document"*, written by [Nevelsteen(2008)]. The primary purpose of the GDD is to make possible and foment the communication between the different designers and teams. Other identified characteristics that are beneficial for the GDD medium are:

- **Ease of authoring and accessing**: The GDD is a reference media for the designers. It is used during the whole game development process and it quickly becomes bloated and inaccessible due to its size. The designers need ready and efficient access to particular information in the GDD and to be able to effortlessly generate new documentation.

- **Linearity and linkable**: Visually the GDD must provide a classic linear narrative, but simultaneously it must be possible to jump to other relevant sections in the documentation via hyperlinks. The development teams should be able to comment and discuss the documentation through the same medium, so that comments are documented.

- **One voice**: The GDD must maintain coherency and narration. The Lead Designer is generally in charged of maintaining the GDD collection, revising and placing the final text in order to create the *illusion* that a single author is narrating it.

- **Linking and tagging**: Apart from the hyperlinking, the GDD should also allow access to external information through inline linking[1]. Other features such as auto-linking and/or tagging, could also be very useful in order to facilitate access to information related to a certain topic.

---

[1]Inline linking is the use of a linked object, often an image, from one site (for example, a web page) into a second site.

Chapter 3

Requirements

3.1 Requirements gathering

The purpose of this chapter is to study several sources related to game design and the use of GDD in order to elicit a collection of requirements for design of the GDD. This collection of requirements will be presented towards the end of the chapter. The gathering of ideas and inspiration to define the requirements was based on several sources, with Kim Nevesteen's description of the GDD as a *"living document"* (mentioned in the background) acting as a pivotal starting point. Other sources included:

- Post-mortem documents from successful game development projects,

- published criticisms about GDD's mediums, and

- the study of existing technologies that could be used as a GDD medium.

Finally, I also considered some requirements that were inspired by my personal experience in the field of documentation. Based on the information extracted from these sources, I formulated the requirements for my design.

3.2 Analysis of Post-mortems

The game industry does not share much information about the game development process, except for the post-mortem documentation. A post-mortem document is a brief description of what went right and what went wrong during the development of a game. The objective is to avoid repeating the same mistakes in future projects. In 2009 Fabio Petrillo published a survey of problems in the game development industry[Petrillo et al.(2009)Petrillo, Pimenta, Trindade, and Dietrich] that showed that 65% of the post-mortems studied had problems in the design phase and that most of the projects used GDDs to carry out the design. Other interesting data extracted from this survey are that 40% of the projects had problems due to the

lack of documentation and 35% had problems due to communication difficulties between the developers. In the post-mortem of the game 'Operation Flashpoint', Ondrej Spanel described problems caused by the lack of documentation:

> *"Lack of documentation is a common affliction among game developers, but some aspects of this problem were so severe in our case that they are worth mentioning. While we'd never believed too much in designing the game on paper, the real problem was that we never even had documentation of the things that we'd finished. This situation led to incredible problems in the final stages of development. Many tasks could only be done by one person on the whole team. In other cases, hours were spent trying to investigate how something had originally been meant to work."[Ondrej Spanel(2001)]*

The need for greater attention to be paid to game design documentation was supported by the majority of postmortems I reviewed. Austin Grossman, designer of *'Trespasser'*, assures that it is better to have out of date documentation than not have documentation at all:

> *"The biggest indication that 'Trespasser' had game design problems was the fact that it never had a proper design specification[. . .]. Our experiences on 'Trespasser' made it clear that it is worse not to have a design specification at all than to have one which becomes out of date and is frequently rewritten. 'Trespasser' started and finished weak in the game design, and this affected every other part of the project."[Grossman(2003)]*

The GDD should define the guidelines of what is going to be included in the game. However, it is assumed that these guidelines and ideas will be modified several times. The GDD must be able to handle these changes, to communicate them to the concerned designers and to provide them a good overview of the process. Brandom Reinhart mentions this when writing the post-mortem of the *'Unreal Tournament'*:

> *"If we develop a design document, we'll use it with the understanding that it can be modified at any time. That having been said, I think there is a definite positive argument for having some sort of central guide to everyone's ideas. Having the ability to sit down and look over the big picture is very valuable."[Reinhart(2000)]*

Some problems that are mentioned, are related to the lack of linearity of the documentation, the difficulty to access the information and to get a good overview of the design. Didier Malenfant wrote in the post-mortem of *'Wild 9'* that:

> *"Many times, titles are late because certain gaps in the original design were overlooked and the full design was never really laid down on paper before development started."[Malenfant(2000)]*

It is clear from reading the post-mortems, that the use of the GDD is dependent on the size and structure of the company. When the development team is small, they rely more on direct communication and use the GDD just as a simple documentation tool. An example of this approach can be seen at the post-mortem of *'Cell Damage'* with sixteen developers:

> *"Our approach to team communication went hand in hand with our approach to staffing. We found that weekly full-staff meetings, individual weekly lectures or presentations to the entire staff, and regular departmental reviews greatly improved all team members' understanding of how their co-workers contributed to the project."[Wu(2002)]*

R. Huebner explains in the post-mortem of *'Vampire: The Masquerade-Redemption'* how communication is improved by placing the whole team in a single room. The team consisted only of twelve developers.

> *"This type of environment doesn't just foster, but rather forces communication between all parts of the team. For instance, a programmer can overhear a discussion between two artists about how to proceed with something and be able to jump in with an answer that will save the project days or months of work. This sort of thing happens on a daily basis; artists correct missteps by the technology team before they are made, a level designer can immediately show a bug to a programmer, and so on. Each of these incidents represents hours or days of project time saved. In an office environment with walls and doors, most of these situations would go unnoticed or unaddressed."[Grossman(2003)]*

On the other hand, when the production is large and/or the company is distributed in large teams over different locations, the GDD becomes an indispensable communication medium. The objective is therefore to enhance the GDD's communication features to make them comparable to the communication that take place in a single room. For example, users should be able to take part in discussions that have been started by other users or provide ideas and suggestions in areas that are not strictly their own. Jamie Fristrom wrote about the communication problems during the development of *'Draconus'*:

> *"Communication between Microsoft and Turbine was also a major factor. The teams were separated by about 3,000 miles and three time zones. Although weekly conference calls were scheduled, they lacked the collaborative mentality necessary for maintaining a successful relationship. E-mail threads were either ignored or else escalated into tense phone calls [. . .] What's important to realize is that it is everyone's obligation to communicate expectations and problems before they escalate to the point of being a crisis."[Fristrom(2000)]*

3.3 Analysis of published criticisms

From the analysis of several publications and articles, I found new ideas that motivated some of the requirements. My aim was to figure out possible ways to express and share ideas between designers and developers. The traditional way to communicate the vision of the game was mostly based on textual descriptions. However, the use of textual descriptions leads to very long documents that nobody bothers to read. The tendency on the use of text to document the game design is to be as concise and expressive as possible. According to [Hagen(2010)], the designers try to sum up their ideas in *key areas of focus* to communicate the vision of the game. Or even to the point of using one single question to make sure the designers are on the right track.

Nowadays the game developers also use non-verbal ways to communicate the design. A common method is the use of concept art and story boards. Another common method is to use inspiring examples and anti-examples. This can include any kind of visual or audio source: images, animations, videos, songs and audio effects. The GDD used to include images from the concept art and the story boards, but its format generally limits integration of other media.

In order to avoid the common problems and limitations of using a single monolithic document to hold the GDD, many designers have experimented with new mediums. Analyses of pros and cons of using several mediums such as blogs, wikis, Google Docs, PowerPoint presentations and Word documents have been conducted by [Lang(2009), Ryan(2009)].

Blog

A blog is a type of website that is usually maintained by an individual. Blogs are structured in text entries that can also support material such as video and graphics. Blogs usually display the entries in a linear way with a chronological order (generally reverse-chronological). Entries can be commented on by the visitors and can be linked to other websites or entries. Blogs have several advantages that could be helpful in supporting the GDD during game development. As outlined by [Cook(2005)], these include:

+ Web-based: It is available anywhere and can be configured to be private.

+ One voice narration: The author of the entries is a single voice that maintains the coherency and narration of the documentation.

+ Ease of authoring: There is no need for formatting, just write plain text.

+ Searchable: It is possible to search for content in the entries.

+ Linkable: Entries are linkable and can also link to web-based content outside of the blog.

+ Comment Friendly: It is possible to make comments on the entries.

+ Support of images and videos

On the other hand the limitations of blogs are:

 - Lack of index: There is not a proper index in a blog, just a sequence of entries.

 - No version control: It is possible to edit entries but there is no way to control or check the history of changes.

 - No autolinking: It is possible to tag the entries, but there is not a feature that automatically makes links referring to a set of terms or articles.

 - Strict linear structure: Blogs are organized in reverse-chronological order.

 - Non support of external links: Blogs cannot link to external files, unless those files are web-based.

Wiki

A wiki is a type of website that allows multiple users to create and edit any number of interlinked web pages via a web browser using a simplified markup language. Wikis can serve many different purposes. For example, community websites, corporate intranets, knowledge management systems, and note services. The software is also commonly used for personal note taking. Wikis permit the user to set the level of control for different functions. For example, access control, editing rights and adding or removing material. Wikis' advantages in order to support GDD according to [Ryan(2009)] are:

+ Web-based: It is available anywhere and can be configured to be private.

+ Multiple user editing: It allows multiple editors to create and edit the content collaboratively.

+ Supports version control: Wikis can manage the changes to content.

+ Ease of updating: Wikis can be updated quickly.

+ Support of images and videos

+ Searchable: It is possible to search for content in the articles.

+ Linkable: Entries are linkable and can also link to web-based content outside of the wiki.

+ Unstructured: Wikis do not have any logical structure. The user is in charge of organizing the information himself. This can be seen as both a positive and negative property of wikis.

On the other hand the limitations of wikis are:

- Lack of index: Wikis are not necessarily linearly structured. They do not have an index or a printable version, unless the user defines it.

- Lack of a single voice: Wikis are not necessarily narrative as they can be edited by several users and can lack linearity.

- Unfriendly formatting: Wikis use a markup language for formatting that makes the editing of the content more complicated.

- Non support of external links: Wikis cannot link to external files, unless those files are web-based.

- Poor at convincing: blocks of linked text written by multiple users are not convincing when one must pitch the design [Cook(2005)].

During the analysis of the wikis, I become aware of two contradictions that appear among the requirements. The first of these paradoxes is: *"one voice"* -vs- *"multi-user editing"*. The designers agreed that it is positive to keep the narrative on the GDD but simultaneously it is necessary to allow multiple users to create and edit the content of the GDD. The future design should find a middle point solution that will not compromise the contradictory statements too much. The second paradox is: *"linear structure"* -vs- *"linking and unstructured"*. Despite its drawbacks wikis are frequently used to hold the GDD[1].

3.4   Survey of existing technologies as potential mediums

During the analysis of postmortems and publications I found several technologies that inspired the gathering of requirements. Some of the technologies could potentially hold the GDDs, but like all mediums, have some disadvantages. The study of technologies is mainly focus on specific features that should be considered for possible requirements of the future design. The technologies are:

- Mind map: Personal Brain,

- Google Wave, and

- Microsoft Pivot.

---

[1]Two open source games that use a wiki to hold their respective GDDs are *"BZFlag"* and *"The Battle for Wesnoth"*

Mind map

A mind map is a graphic representation of a set of ideas and concepts that are linked to each other often arranged around a central concept. They are diagrams used to visualize ideas. There are many mind map software tools, but I focused my study specifically on *'PersonalBrain'* [2]. It is characterised by having a customizable interface and by being extremely cross-platform.

In *'PersonalBrain'*, concepts are distributed in the space surrounding a selected concept. Concepts can have more than one parent. Therefore, a mind map is a graph where the concepts are nodes that are connected to each other through links. In **Figure 3.1** you can see the distribution of concepts around a selected concept, in this case the 'design notes'. Users can explore the concepts by selecting a new concept. Then, the mind map will automatically rearrange the concepts around the new selected concept. There are several possible visualizations that determine the way concepts are arranged around the selected concept and users can define a customized distribution. Mind maps are used to memorize information, solve problems and organize ideas [Buzan and Buzan(1995)]. *"PersonalBrain"* fulfils these objectives due to its dynamic and customizable interface and can integrate many kinds of external information into the mind map such as files, emails, web pages, and images.
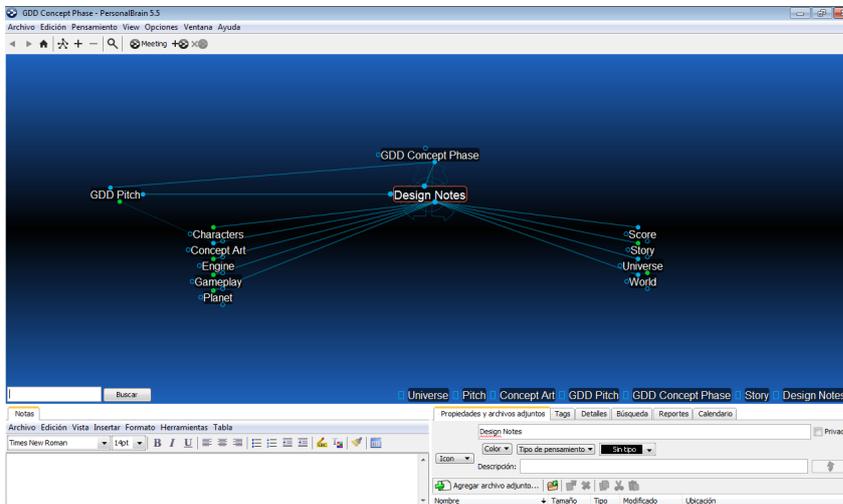

Figure 3.1. Personal Brain

---

[2]For more information about *"PersonalBrain"*: http://www.thebrain.com/

Google Wave

Google has been working on a number of technologies for enabling collaborative work. The most successful tool up to now is *‘Google Docs’* [3] that is a web-based word processor, spreadsheet, presentation and data storage. I have focused by analysis on *‘Google Wave’* [4] , another collaborative tool that has had much less success, but which was much more innovative and risky. *‘Google Wave’* is a web-based framework focusing on real-time collaborative editing. The main purpose of Google Wave is to merge the key features of media like e-mail, instant-messaging, wikis and social networking.

‘Google Wave’ is based on a tree structured editing platform where the simplest element is a wave. A wave is a document that can support text, images, video, pools and other multimedia types. Users can share waves to work collaboratively. Modifications to waves are reflected in real time into the other users' waves. *‘Google Wave's’* most interesting feature is that users are allowed to create a sub-wave at any point into another wave. This sub-wave is used to add a comment or note at the point where it has been created. Users can see where each sub-wave branch starts, because each sub-wave is bound directly to a location on the ‘general wave’. However, the texts stored into waves can become extremely confusing due to the ongoing creation of sub-waves. In my opinion, the ability to hide and reorder the sub-waves is missing in *‘Google Wave’*. It could be interesting for example, being able of gathering those sub-waves which are important conclusions.
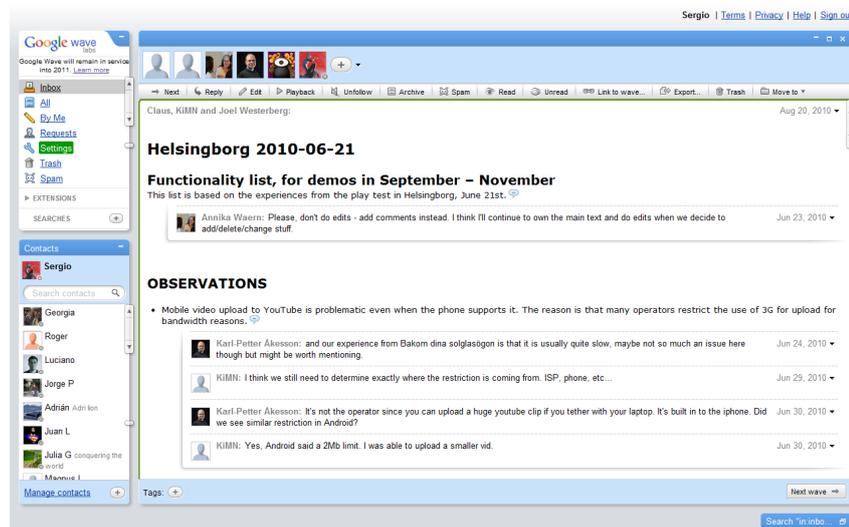
Figure 3.2. *‘Google Wave’*

---

[3]For more information about Google Docs: https://docs.google.com/

[4]For more information about Google Wave: http://wave.google.com/about.html

Pivot

'Pivot' is a software application from Microsoft Live Labs [5]. *'Pivot'* allows users to visualize, interact and sort large collections of data. The classification and redistribution of data is performed in a very dynamic and visual way, in order to help users understand the operations that are performed. *'Pivot'* can also be used to browse web pages and has been described as the way to view the web as a web rather than as isolated pages[Foley(2009)].

A key aspect to analyse is how *'Pivot'* makes access and interaction with large amounts of data easy. *'Pivot'* does not use any kind of basic structure or hierarchy, it just sorts data dynamically according to the user selected criteria. The transitions between the orderings are made visual, allowing users to see data literally fly away or change position on the screen to carry out a new ordering. In **Figure 3.3** it is possible to appreciate the visual effect of a transition in the application. The appearance of *'Pivot'* is of particular interest, because it addresses a limitation of *'Google Wave'*. It allows users to interact with large *unstructured* amounts of data.
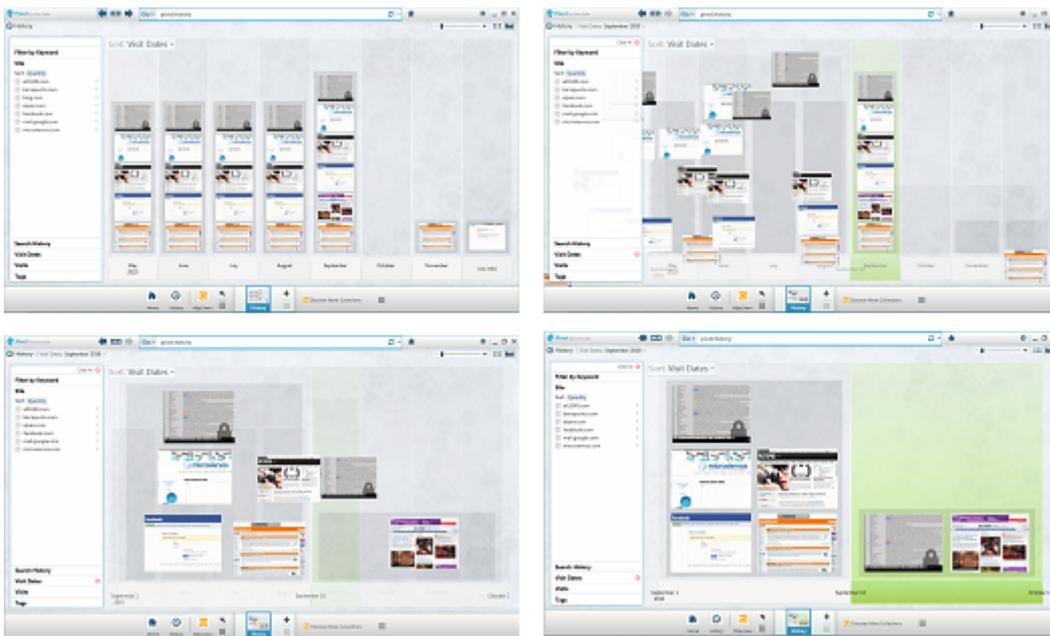
Figure 3.3. *'Pivot'*: Sorting information

---

[5]For more information: http://www.microsoft.com/silverlight/pivotviewer/

3.5   Design aspects derived from personal experience

One of the primary aims of this master thesis is to enhance communication between designers and developers through the use of the GDD. In order to achieve this, it is important to direct the design towards more collaborative and cooperative features. This collaborative development could potentially lead to bloated documentation, so it is necessary to break up the original monolithic structure of the GDD into smaller blocks of information. These blocks should be linkable in order to improve the access to relative information related with the content of each block. In addition, the information should be ordered and presented depending on the interest of the user, so that users can effortlessly access relevant information. Users should be able to organize and rearrange the information quickly in a manner which is most personally useful and accessible for them..

Figure 3.4. Napkin Designs. Source: http://www.frakfraco.com

The survey of technologies, highlighted a lack of GDD support for diverse media types. An example of these media types is the XLS sheet that is generally used, but not supported at all [Ryan(2009)]. Another good example of non-supported media types is the information that is generated spontaneously, like a 'napkin design'. Two designers are discussing matters related with the game development while they are taking a cup of coffee away from their office, suddenly one of them uses a napkin to illustrate his point of view. If the information on the napkin is relevant, it should be possible to quickly incorporated it into the GDD. The same should happen with white boards that are used to list tasks, brainstorm or draw sketches. The audio recording of a meeting could be useful too, or even a prototype of the game. In fact, any kind of media type that is used during the design process should be supported

as part of the GDD or at least be linked as an external source.

Figure 3.5. White Board

One way to avoid the loss of narrative voice is to assign a single person to collect the loose blocks of text and edit them into a presentable narrative. This solution, apart from being tedious work for the person in charge, is not compatible with the quick changes that take place during the rapid development of a game. Wikis allow editor roles to be assigned which give different levels of permission to different users. Using these permissions, it could be possible to require lead editors to sign-off on changes to the documentation before its final publication. Therefore, it should be possible to keep better track and control of the creation of new content and changes into it.

3.6 Design requirements

The following collection of requirements for the design of the GDD were the result of the review presented in this chapter. It is important to remind the reader that the primary objective of this master thesis is to design a data structure that store the GDD and the communication data generated. These requirements do not refer to the interface of the application. The study of the interface is part of on-going future work. That said, the requirements were defined to take into account the data that the interface will require for its proper operation.

- **Requirement 1:** The atomic storage unit is a small block of information that can be linked to other blocks.

- **Requirement 2:** The information can be easily accessed, organized and sorted.

- **Requirement 3:** It supports the storage of multiple media types.

- **Requirement 4:** Operations allow communication in and about the GDD.

- **Requirement 5:** Collaborative browsing and editing is supported.

- **Requirement 6:** Changes can be monitored and notified.

- **Requirement 7:** The information is under revision control tracking.

- **Requirement 8:** It is possible to define user permission levels for editing and revision.

  Note that **Requirement 7** and **Requirement 8** were not considered during the studies in the design process. The revision control tracking was excluded, because it has already been extensively studied and proven efficient [Tichy(1982)] and it would have added unnecessary complexity. The permissions levels were not taken into account because I considered them a non-central issue.

Chapter 4

Design methodology

4.1 Iterative design process

In order to validate and improve the functionality and quality of the design ideas, I utilized an iterative design process. This consisted of three iterations and ended with a final workshop. Iterations commenced after the design of an initial prototype, and progressed in the following sequence:

- Test of the prototype through a design workshop,

- analysis of the results and user experience during the design workshop, and

- refinement of the prototype for the next iteration

The design workshops were conducted at the Mobile Life Center and consisted of one hour sessions with two or three people participating. They were documented by audio recordings, pictures, and written notes, that were subsequently analysed to discover the strengths and weaknesses of the prototypes. The conclusions obtained from the analysis of the documentation were used to further refine the design and build a new prototype for the next iteration. The prototypes were built using sketching methods.

4.2 Sketches and paper prototypes

At the early stages of the research, I found out that the design scope was becoming very broad. The design could potentially end up incorporating features from a wide range of technologies: graph structured storage, communication mediums, support of media types and collaborative editing. Within the limited time frame available to conduct my research, it was clear that the construction of a prototype program that incorporated a representative set of all of these features was not realistic. However, it was clearly undesirable to reframe the prototype in a simplified manner, due to

a loss of critical functionality during the tests. For these reasons I decided that the best solution under the given time constraints was to sketch the design.

Sketches and prototypes are both valid ways to represent a design, but there are small differences between them. According to the book *'Sketching user experiences'* by [Buxton(2007)], sketches are used in the early stages of the design, while prototypes are used in the later stages. Sketches are concept representations of a design and are intended to quickly and inexpensively test new design ideas. They help to explore and suggest new design features rather than confirm and validate them. Some distinctive characteristics of sketches extracted from [Buxton(2007)] are that they:

- Suggest a low level of refinement,

- suggest and explore new ideas rather than confirm, and

- encourage ambiguity.

During the iterative design process I used sketches in order to test and improve the design in the design workshops. In the early stages, the sketches were minimal and highly susceptible to change, even during the design workshops. This was a great advantage as it was simple to make modifications and incorporate new ideas into the design. However, as the iterations progressed, the sketch became more complex and inflexible.

At the final user study, I used a paper prototype. It was based on the previous sketches, but it was more refined, elaborate and also rigid. The intention of the paper prototype was to validate the mature design, unlike the sketch which were also intended to suggest and explore new ideas. In the following chapters, I describe both in detail.

Chapter 5

Design

5.1 First version of the design structure

The first version of the design was a very simple structure that was intended to be improved and expanded during the design iterations. It was required to design a structure to store the GDD satisfying the requirements stated in Section 3.6. Initially, I selected a similar structure to the wiki structure with content stored in blocks of text that could be linked to other blocks. The first GDD design therefore was a collection of a number of blocks of text and their links. From this point forward, I refer to these linkable blocks of text as *nodes*. The nodes also stored also meta-information related to the creation and editing of the node. For example, author, editors, date of creation, last date of modification and the list of links and from other nodes. In order to interact with the nodes and links, a list of operations were defined. The basic collection of operations were the following:

- Define a new node

- Define a new link

- Edit a node

- Edit a link

- Delete a node

- Delete a link

- Split a node

In addition to these operations, a list of communication channels were defined to encourage the communication between the users during editing and interaction with the GDD. The basic collection of communication channels was the following:

- Chat

- Email

- VoIP

- Collaborative editing

5.2   Design Workshops

Three design workshops were carried out, each of them consisted of a one hour session with two or three people participating. The first objective of each design workshops was to test the validity of the structure to store the GDD and support its operations. The second objective was to test the designs ability to solve common problems situations that occur during game development. Some of these situations were initiated by the supervisors of the workshop and some others were problems that arose spontaneously. Users were only able to resolve these problems using the communication and documentation features of the GDD.

Participants were each introduced to the four stages of game design outlined by [Hagen(2009)] in section 2.3 and guided through different problematic situations that take place during this process. The workshop was intended to give participants the chance to figure out innovative ways to store and document the information using a paper sketch. During the workshops a paper sketch was used to simulate the design. The sketch was flexible enough to incorporate the spontaneous proposals of the participants during the design workshops. In the paper sketch, the nodes were represented by small pieces of paper notes with a template printed on them. Therefore, these *paper notes* represented nodes in the sketch. The template printed on the paper notes contained the names of the fields that the participant had to fulfil in order to define a new node. The fields that the template defined were:

- Node's title

- Sections of node content

- Parent of the node

- List of links to other nodes

Participants were provided with a list of operations as the basic interface to interact with the GDD and a list of communication channels. However, they were also encouraged to propose new and more complex operations and new communication methods. In the first iteration the following operations and communication methods were used:
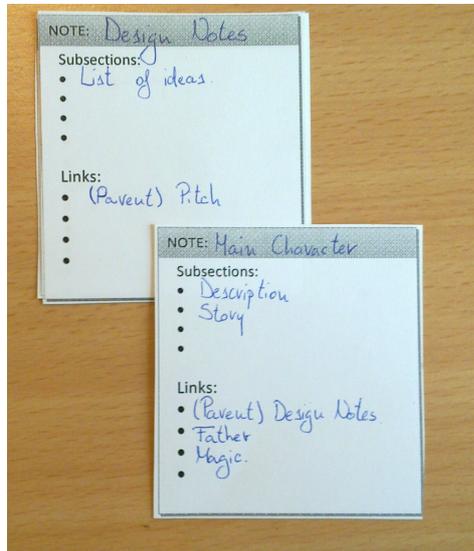
Figure 5.1. Detail of two notes used in the first iteration

- **Operations:**

  - Define a new node
  - Define a new link
  - Delete a node
  - Delete a link
  - Split a node

- **Communication:**

  - Chat
  - Email
  - VoIP
  - Collaborative editing

An empty desk represented the workspace where participants were allowed to physically arrange and organize the nodes. They were encouraged to interact with the GDD in the workspace locating the nodes in the way they felt most comfortable. The participants interacted with a personal copy of the GDD by turns, so in each turn they performed an operation or communication action. A supervisor guided the workshop and kept a global copy of the GDD mimicking the operations that the participants performed. The communication actions were simulated using paper notes if the communication was text based and through a direct conversation between the participants, if it was audio or video based.

5.2.1 Iteration 1

The results from the first workshop were quite favourable. The location of the nodes in the workspace worked well to facilitate participants understanding of the

content. The first task was to brainstorm ideas in order to generate a lot of nodes. The participants were forced to organize the nodes in the workspace as well as they could to keep track of the fast growth of the documentation. However, the participants felt confused when the number of nodes increased rapidly. It became too complicated to keep track of more than fifteen nodes simultaneously.

The participants appreciated the flexibility of the sketch to sort and order nodes. They took advantage of this flexibility to create new documentation based on previous content. This new documentation was usually based on the same GDD content but was filtered to fulfil the needs of different audiences. They communicated using all mediums but most frequently in real-time via VoIP and chat.

Figure 5.2. Notes distribution during the first iteration

During the improvement of the design I focused on facilitating the users to keep control of the the GDD during rapid growth phases. In order to control the number of nodes that the user had to deal with at any one time, I defined a new feature called a view. A *view* includes a subset of nodes defined personally by a user. Views are not related with the storage of the nodes and do not affect the previous GDD structure defined. They simply provide a user-specific way to visualize the nodes. The first objective of the view was to adapt to the way users wished to filter and arrange information. Users could select the relevant information for them and arrange it in a way they felt most comfortable. Then they could save their views to work with them later on, and share the views with other users. The second objective of the views was to limit the number of nodes that the user had to work with at any one time. In this way, users do not need to make an effort to deal with many nodes simultaneously and can stay focussed on their area of interest. Users can however work with several views simultaneously if they wish. The last improvement was to discard the parent link from the nodes. The sketch was modified in order to reflect the new features incorporated into the design. Some of the fields written on the paper note templates were removed to make node creation faster during the workshops.

5.2.2 Iteration 2

Views were used to represent typical sub-documents derived from the game development process (for example, the marketing pitch) through customized subsets of the GDD. During the workshop there was a specific section which tested the links and the relationships between the nodes. Nevertheless, the high number of links created some disorder in the workspace and led to user confusion. I consider links an important feature of the design, but due to the limitations of testing the feature via paper sketch I was not able to make any meaningful conclusions regarding them. This limitation is discussed in Section 5.3 The communication available through the GDD encouraged debate and discussion between users on game design features. Several conclusions that were important for the development of the game, were reached as a result of these debates.

Figure 5.3. Piles of empty notes

I thought that the discussions and conclusions that took place between workshop participants were very relevant for the development of the game design and it would be valuable to have them integrated into the GDD. I generalized GDD nodes to enable them to support simple text blocks as well as other forms of media. In addition, I incorporated audio and video-based communication into the GDD. Finally the sketch was updated, allowing audio and video communication methods that would be documented by the GDD in the form of audio and video notes.

5.2.3 Iteration 3

In the last workshop, I incorporated 'share views' into the list of operations. This operation allowed participants to exchange information through sharing their views. An example of a situation that required sharing a view was the 'share views' - a view specifically defined to introduce the game design and facilitate the integration of a fictitious new designer into the production team. During this iteration, several nodes without any links (orphaned nodes) were defined. Without context, these orphaned nodes caused some user confusion. Users were also unsure which nodes

could be important for them to add to their personal view. For this reason, nodes were required in future to have at least one link to another node to avoid the lack of the context i.e. no orphans were permitted. Some users added conversations, which took place during the workshop, to the GDD as audio nodes.

In the previous iteration, nodes were defined to allow the integration of audio and video and therefore it was important that users could operate on those nodes also. Either the GDD needs to include tools which enable manipulation of these nodes or users should be able to use third party applications to operate on them, with the result being incorporated into the GDD. With such tools in place, users should be able to split audio/video in to nodes. A notification system was added to views, in order for a user to better track changes to the nodes they are working with. Also, if any new nodes were created and linked to a users view, a notification was generated.

5.3   Limitations of the sketch and paper prototype

Some limitations of the sketch prototype arose during the iterative design process. It was not possible to properly simulate the links between the nodes in the defined workspace. The strings that represented the links made it slow and complicated to perform any operation in the structure. Because of this, the links were not represented physically in the paper sketch and the users had to imagine the links connecting the nodes. In my opinion, this feature should therefore be tested in a software prototype together with the user interface, where interactions and feedback can occur rapidly. An additional limitation was that node content modification was excluded during the workshops.

5.4   Design choices

The main design choices that were made during the design process are described and justified in this section. These design choices are the ones that had the greatest influence on the final design:

1. **Unstructured storage**: : As I stated in Section 2.3.4, the content of the GDD has different purposes and audiences. There are many requirements of the structure that stores the GDD content and they are often-times contradictory. For example, if designers want to define a document using the content of the GDD, the structure should be linear, however, if the same document needs to be discussed by other designers, the structure should be flexible enough to support several discussion branches. Because of this need for versatility, the storage structure selected in the design does not follow any of the traditional structures. The aim of the graph structure is to be flexible enough to be easily adapted by the designers for each specific purpose and audience.

2. **Definition of personal views**: The content of the GDD grows rapidly and extensively during game design and development. Designers need to focus during the process on their specific subjects in the GDD. The GDD content that the designers need within their personal workspace depends on their individual objectives. The Views were included in the design to fulfil the need for personal workspaces. Referring to this point, a Lead Designer from DICE [1] explained during an interview:

   > *"The best work actually happens when people are allowed to be creative within the boundaries that you have set up. Then you can sometimes see things that surprise you: "oh, that was a smart use of that thing", for instance. [. . .] I think it's good if as many as possible feel that they are a part of the process when developing a game, that they all are game developers." [Hagen(2010)]*

3. **Browse and share views**: The first design choice states that the GDD content does not have a predefined structure, however the designers need a semantic structure to meaningfully browse the GDD. The storage structure itself has little meaning, however, designers views are are more meaningful as they are created specifically for a purpose. The design defines that designers will browse the content in the GDD through views (which are defined by the designers themselves).

4. **Integration of communication mediums into the GDD**: During the study of post-mortems (Section 3.2) many sources highlighted the importance of communication during game design.

   > *"A programmer can overhear a discussion between two artists about how to proceed with something and be able to jump in with an answer that will save the project days or months of work."[Grossman(2003)]*

   Communication mediums, such as instant messaging and email were incorporated into the design to encourage the exchange of ideas and opinions between the designers and allow commentary and debate. VoIp and video conference mediums are not part of the design, however, the resultant audio and video from the sessions can be integrated. The design of the GDD defines a work environment where designers can exchange, comment, and discuss ideas that are relevant for them.

5. **Integration of media types**: The use of multiple media types during the game design process is well known[Hagen(2010)]. However, none of the technologies studied in Sections 3.4 and 3.5 fulfil the necessity to support and store these media types as part of the GDD. The design incorporates support for multiple media types, enabling them to be stored and manipulated as part of the GDD content.

---

[1] In the Chapter 7, I describe DICE and its relation with this masters thesis.

5.5   Final collection of requirements

The analysis of the design workshops highlighted some interesting new problems that can occur during game design. These new problems could only be revealed through the practical simulation of a real game design process. The requirements were updated in order to address the problems that the previous collection in Section 3.6 did not take into account. The updated list is outlined below:

- **Requirement 1:** The storage structure is a connected directed graph consisting of nodes that contain information and links between them.

- **Requirement 2:** The atomic storage unit is a small block of information.

- **Requirement 3:** The information can easily be accessed, organized and sorted.

- **Requirement 4:** Users can define *personal visualizations* of the information.

- **Requirement 5:** It is supported the integration and formatting of multiple media types.

- **Requirement 6:** Operations allow communication in and about the GDD.

- **Requirement 7:** Collaborative browsing and editing is supported.

- **Requirement 8:** Changes can be monitored and notified based on *personal visualizations* and based on the whole content.

- **Requirement 9:** The information is under revision control tracking.

- **Requirement 10:** It is possible to define user permission levels for editing and revision.

One of the significant changes is the specification of the structure as a connected directed graph at **Requirement 1**. This means that all the nodes should have 1)at least one link pointing to another node and 2) that the links are stored with a specification for both the origin node and destination node. This requirement was included to avoid the loss of information that occurs when a node is not related to other nodes. **Requirement 4** is a new requirement that guarantees the possibility to create customized visualizations of the content of the GDD. Finally **Requirement 8** was updated to specify that the notifications due to changes in the content of the GDD, can be based on the personal visualization that were defined at **Requirement 4**.

5.6   Final design structure

In this section, I will describe the final design structure that was the result of the design iterations. The design does not define the user permissions system or the revision control tracking system, as they were excluded during the study for reasons discussed in Section 3.6. However, for the proper comprehension of the design it is necessary to briefly describe the categories of users that would be involved. Users can be administrators or common users. Administrators have control over all the features of the GDD. Administrators are likely to include the lead designer and perhaps the leaders of the different development and design teams. Common users have access to a limited subset of the features of the GDD. This subset of features can be defined by the administrator using permissions. Common users of the GDD are teams of developers and designers involved in the game design. All the content of the GDD is under a revision control system, so are history of changes is logged and can be reverted to if necessary. The design is divided into three sections that group the features by its main purpose.

Storage structure

The basic elements of the structure are:

- **Nodes**: A node is the atomic storage unit of the GDD. The content of a node can be any kind of media type, including but not limited to text, audio, video, images, spreadsheets or presentations. Nodes are interconnected through links. All nodes have at least one link connecting them to the rest of the nodes. Therefore, the GDD structure consists of an interconnected graph structure with directional links. Nodes also contain tags to classify their content and a wide range of meta-information related to their creation and editing. For example, author, list of editors, creation date and last modification date.

- **Links**: A link relates two nodes in the structure. Links can point to specific node content information such as a specific paragraph or detail in an image. Links can optionally have a title or a tag related to them.

- **Views**: A view is a subset of nodes that are defined by an individual user or team. Views define a way to organize and visualize a subset of the nodes that form the GDD. Views can organize the nodes in different ways, for example as an index, a tree, a graph or a mind map. Views have a title, a brief textual description and optional tags. A view also contains all the information that refers to the way the user has arranged and sorted the nodes in the view. It is important to remark that the views describe ways to visualize the GDD, but this information does not affect the way the nodes are stored. Views are always related with a user or a team and can be explored by the users depending on their set level of permission.
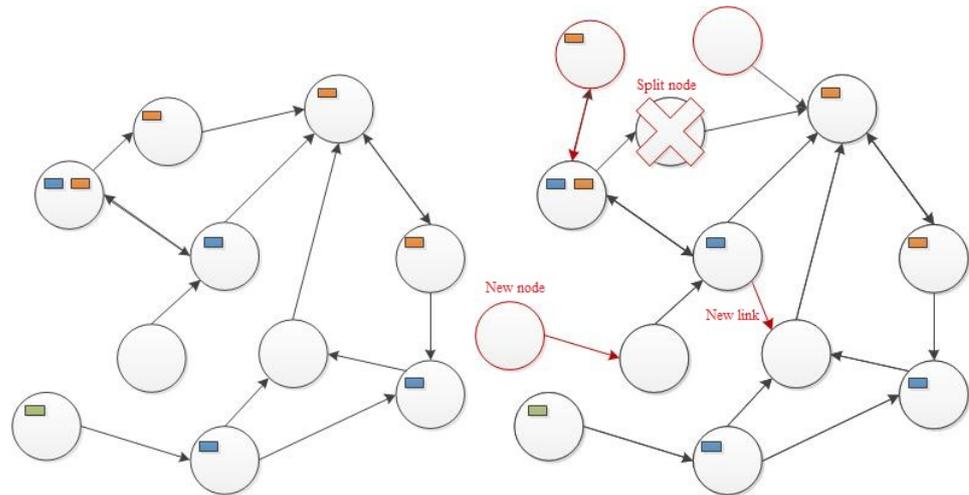
Figure 5.4. Example of the structure and the operations
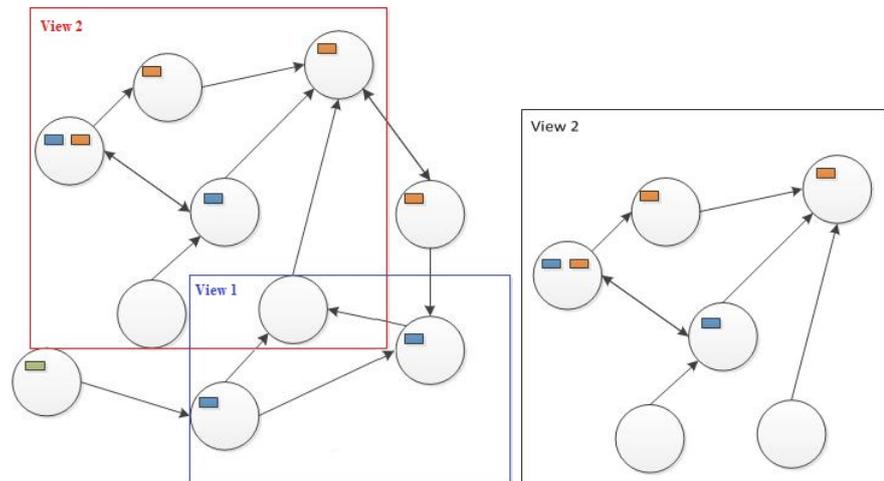
Figure 5.5. Example of Views in the structure and View 2

- **Tags**: A tag is a term assigned to a node, link or view to help classify its content or purpose. The administrator of the GDD can define new fields and sets of tags related to that field. The field can be defined as either mandatory or not to assign during the creation of nodes, links and views. For example, the administrator may define a new field that describes the level of of node content detail with three different optional information tags: low, medium

or high level of detail. Later on, the administrator could decide to make it mandatory for the designers to make a selection when they create a new node.

The basic operations that the structure supports are:

- Define a new node   - Delete a link

- Edit a node   - Define a new tag

- Delete a node   - Edit a tag

- Define a new link   - Delete a tag

- Edit a link   - Split a node

The GDD includes a search engine that makes it possible to find nodes and views that contain keywords. In addition, it allows users to make advanced searches to visualize subsets of the GDD that fulfil a set of characteristics defined by the user. The characteristics are based on meta-information, tags and node content. For example, a user could search a node for a particular keyword that was edited between two specific dates by a certain user.

The GDD also includes a notification system to help monitor and track changes to node content. The view that a user is accessing reflects the interest of the user. Therefore, the notification system is based upon the view that is being actively used. In this way, users are not overloaded with notifications that are not relevant to them at a particular moment. In addition, these notifications can be configured to be received automatically or by user request.

In brief, the structure supports two more features:

- Search and filtering engine

- Notification system

Communication mediums

The design includes a collection of communication mediums integrated into the GDD. Instant messaging and email are both included, with the only difference between them being that instant messaging takes place in real-time while email does not. However, both generate text information. This information is kept private to the users involved in the communication, but if the users think that the information is relevant, it can easily be published as a new node. The information can be filtered and split before publication to include relevant information only. Users can attach comments to nodes in a similar way. Each commentary is published as a new node with a link pointing to the node on which it is commenting.

Users can collaboratively edit notes in real time, so that they can live editing by other users and make immediate comments via instant messaging. In a similar
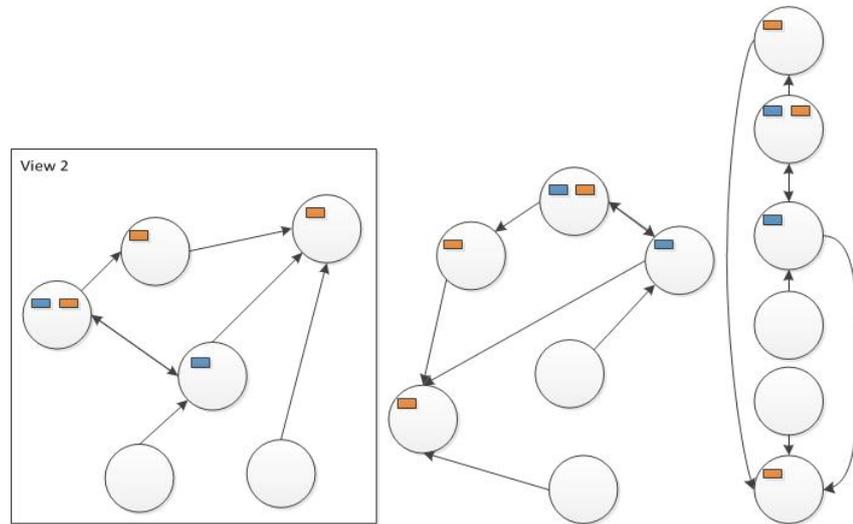
Figure 5.6. Two possible visualizations of the View 2

way, two or more users can browse the GDD collaboratively in real time. A single user must lead the browsing and can browse the activity of another user within the GDD and make comments via instant messaging. Using the collaborative browsing a user can make a presentation of specific content of the GDD. If the users are not working simultaneously, they can share previously defined views. Sharing views is one of the most important communication features, because it allows one user to define views for another user. One example of how sharing views can be used is the 'starter packet', described by [Hagen(2010)]. If a new person is incorporated into the game design mid-way through the process, an experienced designer in the project can define easily a 'starter packet' view with the information that is of most relevance to the new member. Another possible example is the search by the marketing team for interesting features to promote the game. Instead of laboriously going through all the documentation, the marketing team can ask designers from other teams to define a view with the features that they think are most interesting.

In brief, the design of the GDD supports the following communication mediums:

- Instant messaging

- Email

- Collaborative editing

- Collaborative browsing

- Share views

Video conference and VoIP communications are not integrated into the GDD, due to their complexity and cost. Instead, users can use third party tools[2] and integrate the media types generated through these tools at a later stage.

Integration of media types

The nodes of the GDD must support the majority of media types used during game design. Some examples of these media types are: audio, video, images, spreadsheets, presentations, prototypes and 3D models. Media types that cannot be integrated into the GDD are externally referenced by a link to the file system. Media types are stored in nodes and therefore have to fulfil the same requirements as the text based nodes: it must be possible to split the information in a node into several nodes, point out a specific section of the information via links and revise the content of the information via the revision control system. However, the practical use of these features depends on the specific media type that is being stored. While it could be useful to section a picture, it is not useful to segment a prototype or a 3D model. In the future work section, I discuss these problems further and suggest a starting point for resolving them.

In brief, the design of the GDD supports the integration of media types generated by communication mediums:

- Audio from VoIp calls

- Video from video conferences

In addition, it supports the integration of media types used to define concepts and ideas of the game. For example:

- Audio

- Video

- Images

- 2D and 3D models

- Spreadsheets

- Presentations

The design supports the following operations related with this media types:

- Segment the media type

- Link and point out details into the media type

Finally the media types that can not be supported into the GDD storage structure can be linked directly to the file system. Two possible examples are the prototypes and the 'vertical slice' of a game.

---

[2]One of this tools could be for example Skype.
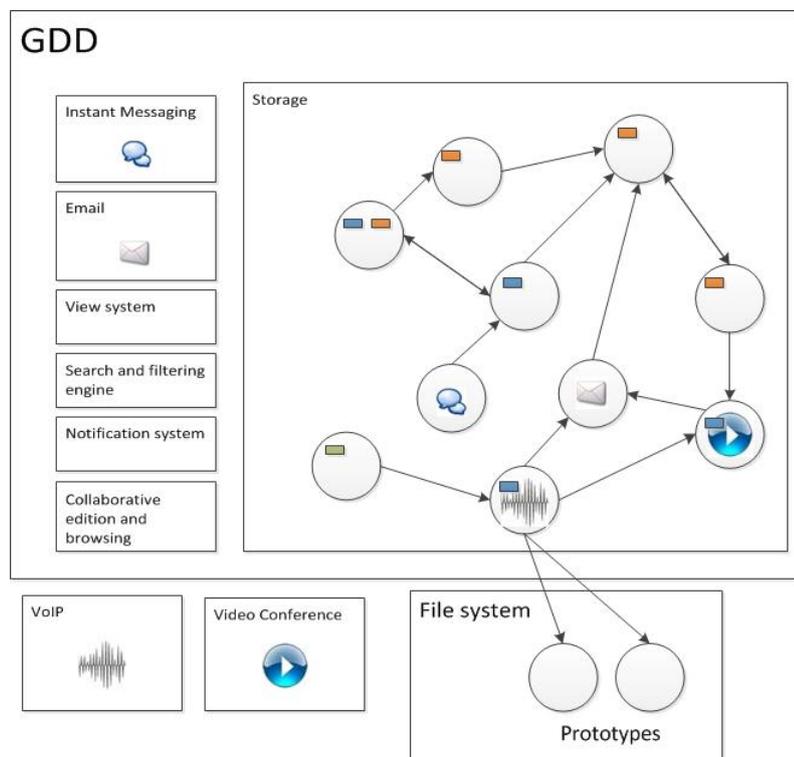
Figure 5.7. Overview of the system

Chapter 6

User study

6.1 Purpose

A final user study was carried out after completing the design process. The objective of the user study was to test in detail the validity of the final collection of GDD design features that the design proposed. The design was divided into three collections of features, each of them with a specific purpose. Correspondingly, the user study consisted of three phases each of them focusing on testing a collection of features.

- The first phase focussed on testing the structure of the nodes and views,

- the second phase was intended to validate the real-time and non-real-time communication mediums, and

- the third phase was intended to test the features relating to the integration of different media types.

6.2 Design of the user study

The user study was designed to simulate the working conditions present in a major game company of in the game industry. It was conducted in a workshop format, with the the main difference from the previous workshops being the involvement of three designer teams in two different rooms. Two teams were located in a single room to test real-time communication mediums between each other. The third team was located in a different room to test the non real-time communication mediums with the two teams in the other room. The separation of the third team prevented any direct communication or awareness of what the other two teams were doing. The communication actions between the two rooms were simulated using paper notes and messengers who physically delivered the paper notes to their destination. The communication in the room with two groups was simulated through passing

paper notes when it was text based communication and through direct conversation when it was audio or video based.

Figure 6.1. Design workshops

The collection of communication mediums that were simulated during the user study:

- Real time communication tools:

    - Chat
    - VoIp
    - Video Conference
    - Collaborative edition of notes
    - Collaborative browsing
    - Share views

- Non real-time communication tools:

    - Email
    - Comment and discuss notes
    - Approval of notes
    - Share views

The detailed documentation and script of the user study is presented in Appendix A.

6.3 Observations

The following observations were made during the design workshop:

- Participants were quick to learn the operations related to the GDD structure.

- Most of the participants had difficulty conceptualising the difference between views and links.

Figure 6.2. The paper prototype

- Participants really appreciated being able to personalise node organisation.

- Most of the behaviours related with the use of nodes, were related to the number of nodes that the participants had to deal with:

  - Participants generally arranged nodes in a linear way, when they had few nodes.

  - Participants generally arranged nodes in more creative ways as the number of nodes increased.

  - When dealing with many nodes, participants generally divided the nodes into groups depending on their content and separated the ones that were not relevant in the moment.

- Several times the participants asked to recover a view previously defined.

- If any of the participants in a team had a new idea to re-allocate the notes, the team was able to explore the new distribution quickly. They kept the distributions that had any special meaning for them.

- The structure operations were natural for the participants, except the ones related with links.

- The participants who were allowed to use all the communication mediums preferred real-time communication mediums such as video conference and VoIP.

- The participants who were allowed to use only non real-time communication mediums preferred share view communication.

- The participants who were allowed to use only non real-time communication mediums, were excluded from debates and decisions that the other participants did, due to the slowness of the process.

- The lead designer used the views and share it with the other team to get and keep them on the right track during the game design.

- Views where defined and shared for communication reasons.

- When two teams of designers had a conflict, they resolved it without leas any need for the lead designer to intervene.

- The participants never proposed to document the communication themselves. However, if the supervisor asked them if they thought the communication should be documented, they always answered affirmatively.

- All participants agreed that images, audio and video material were essential to communicate specific concepts and ideas about the game design.

- There was a single incidence of a debate about a possible feature through VoIP. They agreed to document the discussion, removing parts of the audio that were not relevant.

- The lead designers in both sessions asked for the teams to update their status with reports. The majority of the time they were answered with a view describing the situation. In one instance, a participant used video conference to report their status.

- The notifications were mostly ignored, but when the supervisor asked them directly about it, all agreed that they were essential to keep track of the changes as the number of nodes increased.

6.4   Results

The user study was conducted at the Mobile Life Center and consisted of two sessions of one hour and a half with six and five participants each. The sessions were documented by video recordings, photographs, and written notes. The participants were interviewed after the user study to attain direct and immediate feedback. All the documentation was subsequently analysed to ascertain the strengths and weaknesses of the prototype.

The study participants considered the new features of the GDD generally positive. The distribution of the nodes on the workspace helped them to understand and get an overview of the content of the nodes and their relations. The node structure was flexible enough to organize and rearrange the information quickly. Introducing

the concept of views was a turning point in the design. Views make it possible for designers to select the nodes that contain relevant information for them. Designers can define their own personal workspace without interfering with other designers. In addition, the possibility to share views creates a new communication method. Designers can define views with a specific purpose that is customized to the level of detail that is needed for other designers. Using views allows designers to keep track of the changes that are actually relevant for them. Monitoring subsections of the GDD is easy, however it is not entirely demonstrated that the monitoring of the whole content of the GDD is effective. The organization of the GDD relies on the personal organizational skills of the designers.

The wide variety of communication mediums allows designers to select the best communication medium to fit their purpose. Designers use direct communication mediums such as instant messaging or even VoIP for practical matters and concrete questions. In order to communicate abstract ideas or concepts, designers generally use the different media types that are supported by the GDD. The communication mediums and collaborative editing features incorporated into the GDD encourage dialogue and debate between the designers. That dialogue eventually leads to conclusions and decisions for the design. Finally, the possibility to document the communication where those decisions were taken improves the quality of the GDD.

Chapter 7

Validation in a major game company

7.1 Validation

The final step in the design process was the validation of the resultant design in a major company in the game industry. As I stated in the Section 2.3.2, the design was oriented towards large design and development teams that could potentially be in separate locations. I chose Digital Illusions Creative Entertainment, DICE,[1] in Stockholm (Sweden), because it fulfilled both characteristics. DICE is currently the largest gaming company in Sweden. In addition, DICE is owned by Electronic Arts, EA, in California (USA) and the teams involved in game development are located in different global offices. I scheduled a meeting to present my design to two of DICE's senior designers. Prior to my presentation the opinions of the two designers were opposing - one was for and the other against the documentation of the design through the GDD. The design of the structure and its features were presented and the senior designers affirmed the following:

- There is no need for a specific document editor role. Generally the Lead Designer was in charged of supervising the documentation collected in the GDD, however the Lead Designer is generally too busy to be solely responsible for editing the document. Moreover, there is not a defined role only that they are responsible for it.

- There is a common effort against documenting for documentations sake. Designers are aware of the problems that excessive documentation can cause and are generally reticent about documenting if it is not necessary.

- There is a need for layering the information according to its level of detail. Documentation can become too extensive due to the need for detailing every single feature into the game. The information becomes inaccessible for designers, who need to dig into the GDD in order to find specific content.

---

[1]http://www.dice.se/

Nevertheless, in specific sections of the game design a high level of detail is necessary.

- During the game design a lot of different materials and technologies are used. For example, images, pictures, 3D and 2D models, diagrams, audio, video and presentations. Some teams have defined their own repositories to store specific media types referring to their work, such as 3D graphics and modelling materials. Generally, the most influential design references are the materials hanging on the room walls, that describe the art concepts and 2D models or maps. They also remarked the 'vertical slice'[2] as an important reference during the design process.

The meeting was very constructive with many favourable comments directed toward my design. My final impression was that my design was on the right track, but that further value could be extracted by focussing greater attention on the support of multimedia types in the future.

7.2 Scenarios

During the meeting with DICE, the designers described several situations which take place often during the game design, but however are not properly supported by the GDD. In this Section, two of those situations are presented using the proposed design of the GDD.

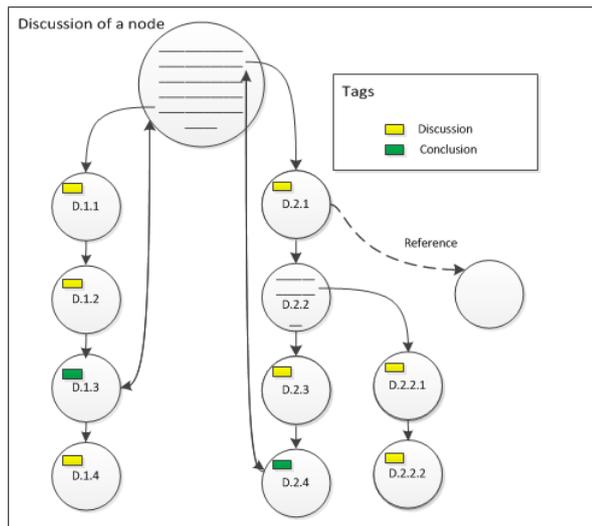### Discussion of the content

'The game of swords'

A game development company is in the concept phase of a fantasy role-playing game called *'The game of swords'*. The game is set in the Middle Ages and it takes place in a lost magical kingdom. The script writer and the world designer are working together using the GDD to describe this kingdom. They decide that it would be positive to get feed-back from other designers, so they send an email to three other designers with a link to the relevant description in the GDD. Two commentaries are then posted into two different sections of the document. A link is embedded in the exact paragraph highlighting the existence of the commentaries (D.1.1 and D.2.1 in Figure 7.1).

In the first commentary a designer asked, "Why are there magic creatures in the kingdom?". This question had not been addressed before and all three designers decided to contribute different ideas to answer the question (D.1.2, D.1.3 and D.1.4 in Figure 7.1). The world designer decides that the second idea is the best answer, so he changes the tag of the note to reflect his decision and makes a link to the section in the GDD where it is described.

---

[2]The vertical slice is a short playable version of a small section of the game, including all fundamental features in that section of the game.

The Lead Designer makes a second commentary (D.2.1 in Figure 7.1). He believes the kingdom should be surrounded by mountains and not by forests as mentioned in the description. In addition, he makes reference to a list of ideas from a meeting where this issue was previously discussed. The script writer explains in his answer that he chose the forest to justify the existence of trolls (that in his opinion can only live in forests). The character designer creates a sub-commentary about this last point, explaining that trolls can also live in the mountains. The Lead Designer agrees with him and describes a rock troll (D.2.2.1 and D.2.2.2 in the Figure 7.1). These commentaries are appreciated by the world designer and the script writer. They propose in the main discussion a middle solution where the kingdom is surrounded partly by mountains and partly by forests. The Lead Designer accepts this idea because it justifies having both kinds of trolls in the game and changes the tag of the note to mark that this is the conclusion (D.2.4 in Figure 7.1). The world designer adds a detailed description of the surroundings of the kingdom and makes a link from it to the conclusion.

Figure 7.1. Storage structure of the discussions

Designers and developers often consult each other regarding the description of the kingdom during the following stages of game development. During the pre-production stage, a level designer reads the kingdom description and gets surprised by the fact that it is also surrounded by forests as he took part in a design meeting where it was decided that the kingdom was surrounded by mountains. However, in the same kingdom description paragraph where he has read this fact, there is a link to a conclusion which describes the reason for the change. He decides to search the description of the kingdom to check if there are any other related conclusions

(Conclusions in Figure 7.2), but the only other conclusion he finds is not related. The level designer realizes that the conclusion was written by the Lead Designer and suspects that he might have forgotten the previous meeting, so he check the whole discussion (Discussion of a conclusion in Figure 7.2). In the discussion he finds a reference to the previous meeting and also the reasons why the Lead Designer has since changed his opinion.
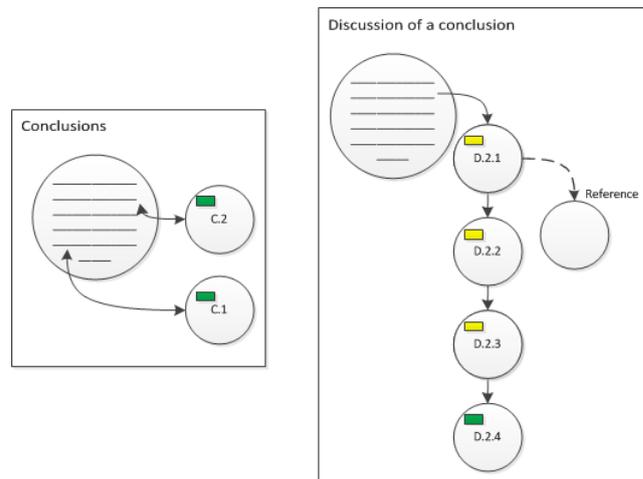
Figure 7.2. Examples of views related with the discussions

Collaboration to find detailed information

During the production of *'The game of swords'* a developer is in charge of implementing a system that enables the magic abilities of the characters depending on their location. The characters can only use magic when they are nearby a specific type of magic tree which is rare in the kingdom. This situation happens just a couple of times in the whole game, however, the developer needs to find out exactly which levels and under what conditions. The developer finds a view for each level that describes it in detail. He knows that checking all the levels individually will take an exceedingly long time, so he decides to send an email to a level designer asking where he can find the information. The level designer is much more familiar with the documentation for each level, so he easily creates a view displaying all the nodes where the magic trees appear and orders them according to the game time-line. In addition, he includes two nodes in the view that he thinks might be useful for the developer: a description of the magic trees and a description of the magical powers that they provide. When he finishes defining the view, he shares it with the developer. The developer receives an invitation to share the view and he

accepts. All the information that he was looking for is in the view, so he can start reading it and continue his work.

Chapter 8

Conclusions

Based on the analysis of published criticisms, post-mortems reports, and recent technologies that could contribute to improve the GDD medium, a set of requirements for the structure of the GDD were stated. According with this set of requirements, an iterative design process was conducted resulting in a tailored design structure for the GDD. Finally, a user study was carried out in order to test the validity of the resultant design. The analysis of the results obtained in the user study presented in Section 6.4 are discussed in this chapter followed by the final conclusions and suggestions for future work.

8.1 Discussion

The design structure was not testable with real GDD content, because it would be necessary to make a real game design process to generate enough GDD content to test it (see Section 5.3) . As a consequence, there are some unresolved questions remaining that concern the behaviour of the designed structure with real content. The first question refers to the identified weakness of the GDDs:

Will the design structure be able to support the GDD and prevent it from becoming bloated and inaccessible?

Relying on the skills of designers, the content of each node should be short and precise. Nodes should be related through links with one or more other nodes which contain relevant information for the node. Under these two basic assumptions, the information in the GDD should stay accessible through the views defined by the users.

At the beginning of game design, the content in the GDD should be enough to simply browse it globally. As the content of the GDD grows, browsing the whole content becomes more and more complicated. However, the same users who create the content can start to define views to keep track of it easily. Views can be defined by the user and for the user, so views keep browsing content simple. The content

would not grow faster than the views, because the users who are defining both the content and the views are the same. The views can be defined with many different purposes and audiences: views can create sub-documents in the GDD such as the marketing pitch; views can define the workspace of individual users, or specific teams; views can collect information about a specific feature and its relation to other features and views should define a 'starter package' for designers who are incorporated during the process. It is important to remark that views can be shared between the users, so designers of specific content can easily define a view based on their experience to help other teams understand their work. Theoretically, at the end of the game design, this ecosystem of views should organize the whole GDD content, customized for all purposes and audiences which have been needed throughout the process.

Access to GDD content should consequently be done through views. However, views are subsets of the whole content of the GDD and there is potential that if everybody accesses the content through views, nobody can get the general picture of the content of the GDD. This includes the lead designer and other team leaders who need to know that the design is in the right track. This leads me the next question:

Will the structure designed to support the GDD prevent it from becoming chaotic or incoherent?

The simplest way to resolve this problem should be to assign one individual the responsibility for editing and revising the GDD. However, this solution is impractical because the documentation can grow too rapidly for a single editor to control it. Furthermore, a single editor would not have sufficient expertise to know how to appropriately filter content from the wide range of disciplines involved in modern game design.

The designed GDD structure supports multiple ways to transmit design ideas. For example, through the integration of different media types. Moreover, the structure allows communication mediums such as instant messaging and email to be integrated into the GDD. The storage and communication of design concepts is also highly supported by the GDD structure. These design concepts describe the objectives of the game design for the users of the GDD structure to follow. Once the users know the objectives of the game design, the responsibility for maintaining the content on the right track should lie with all of them. They can collaboratively edit the GDD. The designed structure is focused on defining views in order to make the information more manageable and users can easily keep the coherence in the views that they use. In cases where views from different users intersect, the users can work collaboratively to keep the coherence between their views. Therefore, the conjunction of all the views should be coherent. The effectiveness of collaborative editing has been demonstrated in other collaborative projects, as for example

Wikipedia[1][Wilkinson and Huberman(2007)]. In conclusion, the collaborative editing of the GDD together with the integrated communication mediums will keep the content of the GDD coherent and focused towards its goals.

8.2 Conclusions

The thorough study and analysis of several sources related with the game design led me to formulate a first set of requirements. Following the workshops a new set of requirements was formulated including several improvements. These last improvements were taken into account thanks to the practical experience that only the workshops could have given me. The final set of requirements was used to design the GDD during and after the iterative design. This set of requirements can be used to extend the requirements of the GDD or as a reference to define similar mediums.

The design of the structure of the GDD itself is the next contribution of this master thesis. The designed structure has proven its validity in the final user study to effectively hold the content generated during the process of game design and fulfils the stated collection of requirements in Section 5.5. The structure is flexible enough to support the fast growth and constant change of the *"living document"* and incorporates a collection of communication mechanisms that assist in improving the overall quality of the GDD. In addition, the structure aims to support a wide variety of media type that are becoming essential to communicate design ideas in the modern game industry.

Finally, 'Views' is a specific feature formulated in the design to allow users to explore, work with and share large amounts of documentation. I consider this feature a significant contribution of this master thesis. In my opinion, the addition of the 'Views' to the structure was the turning point for the design. It has opened up a new range of possibilities that could be used to solve similar problems in other environments such as wikis.

8.3 Future Work

This master thesis is the first step in the design of a complete medium specially designed to hold the GDD. It opens a wide variety of possibilities for future works. The next step is to design a user interface and interaction model for the GDD design structure. The user interface should take into account the requirements outlined in this thesis for the GDD and the design structure. The future user interface should explore the environments where game development takes place including the integration of devices that are used in them. For example, projectors, multitouch screens, and big screens.

A exhaustive study from a software perspective needs to be conducted to check the feasibility of the design proposed. The integration of multiple communication

---

[1]www.wikipedia.org/

mediums into the GDD and the support of a wide range of media types, should be investigated in depth. The integration of media types into the GDD should be defined through standardization methods or through the design of APIs.

It would also be interesting to conduct a more thorough investigation of the game design and development process. Its major purpose should be to improve the GDD medium, but also to define the best way to integrate its use in the game design process.

Finally, the most interesting proposal for future research in my opinion is the generalization of the current design structure for similar development and documentation processes in other industries such as for example the film industry. Game design has many peculiarities which have prevented it from using general mediums for its documentation. However, because of this, the design of a medium specially customized for the requirements of game design, creates some interesting features. In my opinion, it would be worthwhile to investigate if this new structure could be used to document similar processes like the film production, or even generalize it to any documentation process.

Bibliography

[Adams and Rollings(2006)] Ernest Adams and Andrew Rollings. *Fundamentals of Game Design (Game Design and Development Series)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006. ISBN 0131687476.

[Björk et al.(2003)Björk, Lundgren, and Holopainen] Staffan Björk, Sus Lundgren, and Jussi Holopainen. Game design patterns. In *DIGRA Conf.*, 2003.

[Buxton(2007)] Bill Buxton. *Sketching user experiences: getting the design right and the right design*, volume 54. Morgan Kaufmann, 2007. ISBN 0123740371. URL `http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0123740371`.

[Buzan and Buzan(1995)] Tony Buzan and Barry Buzan. *The Mind Map Book*. BBC Books, London, 2 edition, 1995.

[Cohen and Bustamante(2009)] D. S. Cohen and Sergio A. Bustamante. *Producing Games: From Business and Budgets to Creativity and Design*. Focal Press, 2009. ISBN 0240810708, 9780240810706.

[Cook(2005)] Daniel Cook. Using a blog as a game design document, June 2 2005. URL `http://www.lostgarden.com/2005/06/using-blog-as-game-design-document.html`.

[Foley(2009)] Mary-Jo Foley. Pivot: Microsoft's experiment to 'view the web as a web', November 2009. URL `http://www.zdnet.com/blog/microsoft/pivot-microsofts-experiment-to-view-the-web-as-a-web/4612`.

[Freeman(1997)] Tzvi Freeman. Creating a great design document, September 1997. URL `http://www.gamasutra.com/view/feature/3224/creating_a_great_design_document.php`.

[Fristrom(2000)] Jamie Fristrom. Postmortem: Treyarch's draconus, August 2000. URL `http://www.gamasutra.com/view/feature/3138/postmortem_treyarchs_draconus.php`.

[Grossman(2003)] Austin Grossman. *Postmortems from Game Developer*. CMP Books, 2003. ISBN 1578202140.

[Hagen(2009)] Ulf Hagen. Where do game design ideas come from? invention and recycling in games developed in sweden. In Atkins Barry, Kennedy Helen, and Krzywinska Tanya, editors, *Breaking New Ground: Innovation in Games, Play, Practice*

and Theory: Proceedings of the 2009 Digital Games Research Association Conference, London, September 2009. Brunel University. URL `http://www.digra.org/dl/display_html?chid=09287.25072.pdf`.

[Hagen(2010)] Ulf Hagen. Designing for player experience: How professional game developers communicate design visions. In Lankoski Petri, Thorhauge Anne Mette, Verhagen Harko, and Waern Annika, editors, *Proceedings of DiGRA Nordic 2010: Experiencing Games: Games, Play, and Players*, Stockholm, January 2010. University of Stockholm. URL `http://www.digra.org/dl/display_html?chid=10343.03567.pdf`.

[Lang(2009)] Tim Lang. Four ways to write your design docs, May 2009. URL `http://www.gamecareerguide.com/features/737/four_ways_to_write_your_design_.php`.

[Laramee(1999)] Francois Dominic Laramee. The game design process, Nov 1999. URL `http://www.gamedev.net/reference/articles/article273.asp`.

[Larsen(2002)] Simon Larsen. Playing the game: Managing computer game development. Master's thesis, Business Academy of Information and Communication Technology, 2002. URL `http://blackwood.dk/PDF/PlayingTheGame-IE.pdf`.

[Malenfant(2000)] Didier Malenfant. Postmortem: Shiny entertainment's wild 9, January 2000. URL `http://www.gamasutra.com/view/feature/3417/postmortem_shiny_entertainments_.php`.

[Nevelsteen(2008)] Kim J L Nevelsteen. Gdd as development methodology. Master's thesis, Department of Information Technology, 2008.

[Nevelsteen and Gayoso(2011)] Kim J. L. Nevelsteen and Sergio Gayoso. Gdd as a communication medium. In *Games and Innovation Research Seminar*, page 7, 2011.

[Ondrej Spanel(2001)] Marek Spanel Ondrej Spanel. Postmortem: Bohemia interactive studios' operation flashpoint, December 2001. URL `http://www.gamasutra.com/view/feature/3018/postmortem_bohemia_interactive_.php`.

[Petrillo et al.(2009)Petrillo, Pimenta, Trindade, and Dietrich] Fábio Petrillo, Marcelo Pimenta, Francisco Trindade, and Carlos Dietrich. What went wrong? a survey of problems in game development. *Comput. Entertain.*, 7:13:1–13:22, February 2009. ISSN 1544-3574. doi: 10.1145/1486508.1486521. URL `http://portal.acm.org/citation.cfm?doid=1486508.1486521`.

[Reinhart(2000)] Brandon Reinhart. Postmortem: Epic games' unreal tournament, June 2000. URL `http://www.gamasutra.com/view/feature/3161/postmortem_epic_games_unreal_.php`.

[Ryan(2009)] Tim Ryan. Learning the ways of the game development wiki, July 2009. URL `http://www.gamasutra.com/view/feature/4094/learning_the_ways_of_the_game_.php`.

[Salen and Zimmerman(2003)] Katie Salen and Eric Zimmerman. *Rules of play : game design fundamentals*. MIT Press, October 2003. ISBN 0262240459.

[Tichy(1982)] Walter F. Tichy. Design, implementation, and evaluation of a revision control system. In *Proceedings of the 6th international conference on Software engineering*, ICSE '82, pages 58–67, Los Alamitos, CA, USA, 1982. IEEE Computer Society Press. URL `http://portal.acm.org/citation.cfm?id=800254.807748`.

[Wilkinson and Huberman(2007)] Dennis M. Wilkinson and Bernardo A. Huberman. Co-operation and quality in wikipedia. In *Proceedings of the 2007 international symposium on Wikis*, WikiSym '07, pages 157–164, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-861-9. doi: http://doi.acm.org/10.1145/1296951.1296968. URL `http://doi.acm.org/10.1145/1296951.1296968`.

[Wu(2002)] David Wu. Postmortem: Pseudo interactive's cel damage, February 2002. URL `http://www.gamasutra.com/view/feature/3007/postmortem_pseudo_interactives_.php`.

Appendix A

User Study: GDD as a communication medium

A.1 Overview

-> The whole appendix has to be double check and some parts re-written

The use of a Game Design Document (as well know as GDD) during the design and development of games is considered almost a challenge. There are a lot of inconveniences in the use of GDD: The documentation grows really fast (sometimes even out of control), it´s hard to keep track of the changes and evolution of the content and it becomes difficult to find specific information into it. However, the creation of documentation for the game design process is essential to be carried out successfully.

The objective of the study is to test the validity of a collection of new features to enhance the use of the GDD. The study will deal with the different stages of the development of games, introducing some problematic scenarios.

The main objective of this study is to prove that the communication through the GDD it´s possible and, at the same time, that it improves the documentation of the game design.

A.2 Objectives

The objective of the study is to test the validity of a collection of new features to enhance the use of the GDD. The purpose of these features is to enhance the communication of the designers through the use of the GDD and, at the

same time, improve the documentation of the game design. I divided the features in three categories that are aimed at resolving the weaknesses of GDD. The study will be divided in three phases that match the categories and will focus on test each of them. Next, I will the new features that are part of each category.

GDD's Structure

My proposal consist in a new GDD structure based on notes and customized views of the them. The notes are blocks of linked text and the views are indexes or graphs that describes the relations between the notes. The GDD will be a collection of notes and a collection of views that will not include necessarily all the notes. These views will be defined by the different teams or individual users involved in the design. The objective of the views is make simplified versions of the whole content of the GDD that will provide only the information needed by user. These views will not affect the storage structure of the GDD or the other views. The aim it's make possible the creation of customized versions of the GDD that will be more manageable and accessible by the users to work with and edit.

GDD's Communication

One of the main problems during the design of games is the miscommunication that take place between the designers from different teams. To avoid this problem, I propose a collection of communication tools that will be integrated into the GDD. The main reason why I propose to integrate the communication tools into the GDD is that the information generated during the communication of the team has a certain value. The discussion from two designers about a feature can be useful for a third designer to understand the feature, so it could be interesting to document it as a part of the GDD. Based on the structure previously presented, we generalize the notes from blocks of linked text, to any kind of digital files (audio files, chat conversations, video conferences...). The notes will support the documentation of the information generated during the communication between the designers through the GDD. This communication tool kit will support:

- Real time communication tools:
  - Chat
  - VoIp

- – Video Conference
- – Collaborative edition of notes
- – Collaborative browsing
- – Share views
- Non real time communication tools:
  - – Email
  - – Comment and discuss notes
  - – Approval of notes
  - – Share views

Integration of media types into the GDD

The content of the generalized notes is now any information that has been generated due to the edition of the GDD and the communication that takes places through it. That´s is the information that is generated while the user work actually in develop the GDD, but there is more information that could be generated during other situations that does not need to be directly related with the GDD. he next step is to integrate even more information from the design environment. Our objective it´s to facilitate the documentation of meetings, talks our any kind of spontaneous inspirational information that could improve the quality and usefulness of the GDD. The new formats that the notes support are:

- Images
- Audio
- Video
- Presentations
- Links to prototypes

A.3   Workshop set-up

Subjects

The study will need a minimum of five participants. The participants will be divided in three groups (if there are five participants one of the groups will be formed by a single person). The groups will represent the different design teams that work together during the design process (Lead designer,

development team, level designers, artists...). Depending on the phase of the study, the group will assigned to a design team or will have the chance of choose itself which design team wants to represent.

Space

The study will take place into two different rooms that must be situated close to each other. Two groups will be situated into a bigger room and a third group will be situated into a smaller one. The objective to divide the participants into groups is to force the use the communication tools that will be simulated during the study. The groups into the bigger room will be focus into the test of the real time communication tools and the group into the smaller room will be in charge of testing the non real time communication tools. The development of the GDD into the bigger room can be improvised and will be registered by the supervisors of the study. Into the smaller room the GDD will be arrange in advance by the supervisor, so the supervisor will know at every moment what it is approximately the content of the whole GDD.

Communication

One of the main objectives of the study is to test the communication between the participants through the GDD. The communication tools described previously will be all simulated. A few examples of how it will be simulated are explained next.

- Email, Chat, Comments: The participants will have notes to write the message on and they will fiscally bring it to the target.

- Collaborative edition, browsing and shared views: The participants from the groups involved will move together to the same board to "share" the view.

- VoIp, Video Conference: The participants will talk directly to each other, as a result of the VoIp or Video Conference there will be a audio or video file.

Documentation

The study will be documented mainly recording the process with several video cameras and a sound recorder. At the end of the study the participants will answer to a short questionnaire that will be recorded too. The

supervisors will be in charged of taking notes registering the time to access later on easily to the interesting points of the recordings.

- **Main room**

The following image describes the disposition of the supervisor (in red) and the participants (in blue) into the room. There will be two types of cameras, a professional video camera and two iPhones with video cameras. The supervisor will be situated between both groups and will control the progress of the groups.
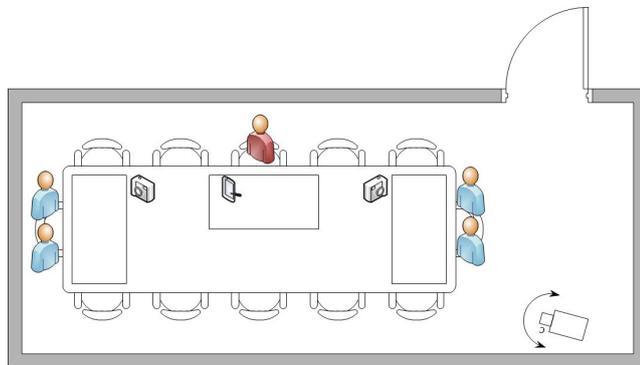
Figure A.1. Distribution into the big room

- **Small room**

The following image describes the disposition of the supervisor (in red) and the participants (in blue) into the room. There will be a professional video camera recording the process. The supervisor will be situated next to the group and will control the progress the study.
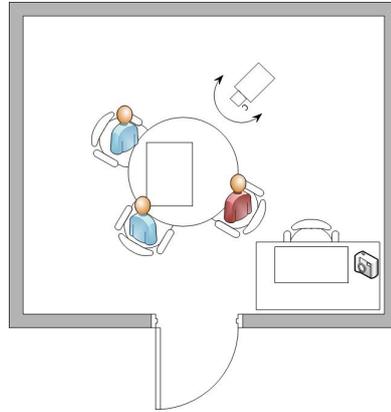
A.4 Procedure

The study will have four stages with different purposes. Next, I will explain each phase.

Introduction to the GDD

Material

- Sheets of paper with the operations available into the GDD.
- Cards representing the notes into the GDD

Figure A.2. Distribution into the small room

Main room (10 minutes)

This phase will tale place into the main room with all the participants of the workshop and it will be a short introduction to the study. It´s important that the participants get an idea of what is the GDD, a pitch, the notes and the views. They will be also introduced to the collection of operations that are available in the GDD.

GDD's Structure

Material

- Sheets of paper with the operations available into the GDD
- Cards representing the notes into the GDD
- Cards and sheets representing the notes of the levels, locations and enemies of a possible game
- Boards to share views between the groups, if necessary
- All kind of office material to let them improvise new solutions

Main room (25 minutes)

The two groups in the main room will work using turns. They will perform one action per turn alternately. The objective is to make the participants get familiar with the possibilities of the structure and the views. The process will simulate the first phases of the development of a GDD.

1. Generate a list of Design notes and try the possible operations with the note structure.

2. One group will get the role of the lead designer and will start on the creation of the pitch. The other group will get the role of the world designer team and will create more notes that will create a team view specific for his team.

3. The lead designer will contact the world designers to ask them about what should he include into the Pitch. The lead designer will contact the level designer to get an overview of the levels and the most interesting notes.

Small room (25 minutes)

The group in the small room will get the role of the level designer team. Their task will be to use the note view to try to make more understandable a table of content refer to the levels, the locations and the enemies into the game.

1. Use the note view to describe the content of the table that relates the levels into the game with the locations.

2. Use the note view to describe the content of the table that relates the levels into the game with the locations and the enemies.

3. Wait until a group from the other room contact them to ask for an overview of the levels and the most interesting points.

GDD's Communication tool kit

Material

- Sheets of paper with the operations available into the GDD
- Cards representing the notes into the GDD
- Cards and sheets representing the notes of the levels, locations and enemies of a possible game
- Boards to share views between the groups, if necessary
- All kind of office material to let them improvise new solutions

Main room (25 minutes)

The two groups in the main room will work using turns. They will perform one action per turn alternately. The objective is to generate more documentation through the use of the communication tools into the GDD. All

the groups will communicate with each other for different reasons and the documentation created by that communication should add some value to the GDD.

1. The lead designer should try to make an official version of the GDD. The lead designer will be notified at each time that a note is created and will integrate it into the GDD. At the same time he will try to keep track of what's going on between the other designers. The designer can decide to create new notes to complete the GDD. The world designers will create their personal view with the original design, the notes from the level designers and some new notes that they will create.

2. The world designers will communicate with the level designers to get the status of their project. Then they will notify/discuss with them the reasons why the Location 4 should be divided into two different locations. They will also send them the design of the map of the game. They will communicate using the non real time tools.

3. The lead design will conclude the discussion between the two other designers. The information that is considered useful by him will be stored into the GDD.

4. The lead designer will take a look to the view of the level designer and discuss with them some changes. The result of the communication should be documented.

Small room (25 minutes)

1. The level designer will create their personal view with the notes they have and some new notes that they will create.

2. The level designer will wait to be contacted by the world designers. They will discuss about the reason why the Location 4 should not be divided and finally decide to divide it.

3. They will get a map from the world designers and they should try to use it to improve their documentation.

4. The discussion about the division of the Location 4 will be concluded by the Lead designer.

5. If there is time, the documentation about the changes and the communication between the lead designer and the world designer should be checked by the level designers.

Interview

1. What is in your opinion the advantages of working with notes and views?

2. Which is the main disadvantage?

3. Do you think that is important to document the communication between the designers?

4. Which is in your opinion the most useful communication tool that should be documented?

5. Do you have any suggestion?

. . . so . . . Who has the ball?
I have the ball