Fakultät für Informatik
Technische Universität München

Bachelor's Thesis in Informatics: Games Engineering

Boss Engineering: Methods and Tools for Game Development

Clemens Fromm

Fakultät für Informatik
Technische Universität München

TUM

Bachelor's Thesis in Informatics: Games Engineering

Boss Engineering: Methods and Tools for Game Development

Boss Engineering: Methoden und Tools für die Spieleentwicklung

Author:	Clemens Fromm
Supervisor:	Prof. Gudrun Klinker, Ph.D.
Advisors:	Daniel Dyrda, M.Sc.
Submission Date:	November 15, 2021

Eidesstattliche Erklärung

Ich versichere hiermit, dass ich diese Bachelor's Thesis selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

Munich, November 15, 2021

CLEMENS FROMM

Abstract

This thesis explores the design concepts of bosses and boss fights and the process behind them. Its main focus is to create a prototype of the tool, Boss Board, in an attempt to support and assist the boss design process by creating a model of a boss and its fight. This tool is based on the foundation of a shared knowledge base, vocabulary and design language surrounding boss design. The creation of this foundation is attempted throughout the thesis as an answer to the problem statement of addressing a less utilized design space within game design. This is done by clarifying, categorizing and displaying concepts of boss design. The aforementioned prototype and its components as well as the intent and motivation behind it is elaborated on after looking at already existing ways on how to display boss fights. The results of this thesis and the prototype created are then discussed and further options on how to extend it are brought up.

Acknowledgements

Contents

1 Introduction

Games have employed the use of bosses since the early stages of the history of video games. Early examples of bosses as unique and powerful opponents at the end of a game already appear in the 1970s and 80s in the form of a golden dragon in Dnd (Whisenhunt & Wood, 1975) and the giant spaceship, Mothership, in Phoenix (TPN, 1980) and more examples have followed in the years after (Lee, 2015). At the time bosses may have not been addressed as such from the start but their core concepts as powerful enemies that test the player's abilities were already present. This concept transcends the domain of video games and is present in other forms of media in different contexts and shapes in for example movies and tabletop games (Grayson, 2021). While the precise origin of the term boss is unclear it has persisted as a common terminology for this type of enemy and challenge in games. Bosses and their fights are still a common sight in video games and some of the most famous and influential titles of the past decades still incorporate bosses in a variety of different ways. Bosses often appear at key points within a game and are often displayed as a highlight within them.

The creation of these bosses is an interdisciplinary endeavor within the game development process. A major part of this process relies on the game designer and the game design process but almost every aspect of a boss is created and collaborated on by different parts of the development team. This process is highly individualized and depends on how a development team works, what type of game it is, what genre the game is part of and what the intent behind the boss is.

The goal of this thesis is to address the process behind creating a boss and its corresponding fight as they are significant component of the games that do employ them. While there are already many existing tools in both software engineering and games development that assist the game design process, the process of boss design specifically is less utilized for a component of games that has persisted as long as it has. This thesis focuses on what a boss and its fight is and what type of components they include. After these concepts are clarified existing ways that could aid this design process will be explored. A major part of this thesis is the creation of a prototype for a tool that is intended to support the creation, visualization and evaluation of a boss and its fight. This tool is meant to be used as either part of the prototyping or pre-prototyping stage.

One of the primary problems addressed in this thesis is the concept of a foundation for boss design and communication about it. Parts of this thesis attempt to create a shared knowledge base and vocabulary for boss components as well as creating a design language to model bosses via the aforementioned tool.

The structure of this thesis is summarized in the following part of the introduction. The next chapter will take a look at existing research that relates to the process of boss design and its components. The three primary research topics are: research into bosses, game design and game design tools.

The following chapter, Chapter 3, clarifies the questions of what a boss and a boss fight are. The first two sections of this chapter concern themselves with describing and pre-

senting the role and purpose of a boss and the general structure of a fight as well as its components, principles and relevant design concepts. The two latter sections will then explore the aspects and categories of a boss and its fight as a foundation for later chapters by defining and labeling them.

Chapter 4 will then explore the modelling process of a boss. This will be done by first taking a look at existing ways on how to aid the process of creating a boss and its fight based on the established aspects and categories of the previous section. After that the purpose, motivation and early guidelines of the tool that is meant to assist the boss creation process are clarified. The last section of this chapter will then introduce and showcase the Boss Board, the prototype of a tool meant to assist with creation process. This includes the definition and visualization of core components and the tool's principles.

The next chapter, Chapter 5, discusses the results of the thesis concerning the boss design process while focusing on the created tool, the Boss Board. This will take the tool's usage, results and effect into consideration.

The last chapter, Chapter 6, summarizes and concludes the thesis while presenting additional ways on how to further proceed and add to what was researched and created.

2 Related Works

The related work that concerns the topic of this thesis can be grouped into the two major categories of game design and bosses. This section will take a look at research that has been done concerning game design and tools for game design and research that concerns itself with bosses.

2.1 Game Design

The research about game design that is related to the topic of this thesis primarily concerns itself with game design models and tools that assist in the process.

Hunicke, LeBlanc & Zubek (2004) propose the formal framework MDA (Mechanics, Dynamics, Aesthetics) that is meant to support the understanding and communication of games design in different areas surrounding games including development, consumption and research. They define the fundamentals of their framework as mechanics, dynamics and aesthetics and describe common models and vocabulary for their fundamentals. Their framework sets an emphasis on how each aspect of the framework influences other aspects based on changes made to it. This emphasis assists in creating a design process based on iteration over a game system. (Hunicke et al., 2004) MDA addresses a similar problem to one that is considered in this thesis which is building a framework or model that supports the design process and sets a foundation for shared communication about design. In the case of this thesis the design space of bosses is specifically taken into consideration as opposed to game design as a whole. The concept of iteration over the framework is also considered in the creation of the Boss Board described in Chapter 4.

In the PhD thesis "Engineering emergence: applied theory for game design" Joris Dormans (2012) presents a wide range of game design concepts and foundations as part of game design theories. Dormans' thesis also involves the creation of the two frameworks and tools that are meant to be applicable to the game design process. His frameworks focus on two specific areas of game design. The first framework presented by Dormans is called Machinations which focuses on game rules and dynamics that are displayed by using dynamic diagrams that simulate the internal economy of a game based on resources. The second framework he presents is called Mission/Space which focuses on level design and player progression by using two types of graphs, space and mission graphs, to display the two focus targets respectively. Dormans further explains that Machinations is meant to display the design concept of emergence while Mission/Space is meant to display the design concept of progression. He also describes the two tools created to use their respective frameworks, Machinations and Ludoscope (Dormans, 2012). The two concepts of emergence and progression are also mentioned in this thesis as part of the boss design and how it can be planned. Machinations is also further looked at in Chapter 4 as part of the potential ways on how to assist the boss design process and modeling thereof. The focus on a more graphical and visual notation for a framework and tool is also considered for the Boss Board described in the same chapter.

Katherine Neil (2015) addresses in her thesis if game design practice can be improved by game design tools. This is done by her through practice-led evaluation of design tools. She applies several tools to multiple game case studies and defines, reviews, presents and compares existing design tools. She also discusses the experiences of the creation of her own progression design tool Progressimo, a tool created to design and display player progression in a game. She summarizes how design tools might be used to support the game design process and how to further built on practice-led evaluation research (Neil, 2015). The research behind the extent of usefulness of game design tools helps with the understanding of how necessary a tool and its function are for the game design process. In the context of this thesis the functionality of a tool like the Boss Board needs to be taken into consideration for how to further develop and regard its use case regarding the benefits and difficulties that come with the use of tool-assisted design.

2.2 Bosses

Research concerning the topic of bosses primarily focuses on the game mechanics behind a boss or its design.

In his work Agriogianis (2018) describes the role of bosses, their history and their evolution over time while discussing their mechanics and impact on a game. He categorizes bosses into multiple roles, addresses mechanical shortcomings and changes to the common formula of bosses. His research culminates into his thesis game while collecting a knowledge base for future use (Agriogianis, 2018).This work primarily concerns itself with the design space behind a boss, the definition of a boss and what can be learned from a boss for future purposes similar up to a certain extent to what is discussed in the next chapter of this thesis.

Wood & Summerville (2019) research bosses by taking the 2D action game Cuphead (Studio MDHR, 2017) that is filled with a multitude of bosses as a case study for a quantitative analysis of bosses and boss attacks to create an ontology representing them. Their goals is to determine the responses and opinions that players have towards a boss by comparing collected data from a user study to the data collected throughout their analysis and creation of their ontology (Wood & Summerville, 2019). This research offers a foundation for how attacks could potentially be categorized from a mechanics standpoint while also taking the player's experience of the boss into consideration for design purposes. While the discovered information is not necessarily applicable to every problem of this thesis, attacks and parts of the player experience are addressed.

The work by Siu, Butler & Zook (2016) presents an approach to displaying and defining a boss by using a progamming model. They specifically concern themselves with boss experiences in 2D action games and their characteristics with the future intent of boss generation. Their model is based on primitive systems, states and finite state machines to define the behavior of a boss. The work done by Butler, Siu & Zook (2017) extends on the aforementioned model by generating bosses based on it. They define bosses as programs with constraints, design properties and behavior. Their aim is to generate bosses through procedural content generation by showcasing an algorithm that forms and generates programs (Butler et al., 2017). The progamming model created to display the boss experience and behavior focuses on a very specific type of bosses and on the logic and behavior surrounding them. The model that is created in this thesis also partially focuses on these aspects but follows a more graphical and visual notation to represent them.

3 Boss Design

This chapter will be giving an overview of what exactly a boss and a boss fight are while showcasing their core components and aspects. The process of creating a boss is intrinsically connected to game design itself. To design a boss one would first need to look at certain parts of game design and its process that are relevant for boss design. This will be done by introducing these parts in combination with parts of the boss design.

The beginning of this chapter starts off with defining what a boss is. This includes the role and the purpose of a boss within the game. The definition process is then extended to what a boss fight is by exploring the general structure and mechanics of a boss fight. Characteristics and principles of a boss fight are also explained and contextualized with game design concepts.

The next section of the chapter takes a look at the aspects of a boss fight that can be influenced. These aspects are then labeled and defined to build a further foundation for what a boss and a boss fight are. They include aspects that contextualize the boss and its mechanics and influence each other internally.

The final section explains three overarching categories that can define bosses given a game's genre. The individual categories are also further differentiated.

3.1 What is a Boss?

Bosses are a common sight in games that are often used at pivotal points within the narrative of a game or to test the player's skills that were learned throughout the sections leading up to a boss. Depending on the type of game they can be introduced in a variety of ways and are often tied into the story and presented through clues, dialogue or cutscenes. This section will explore the role and purpose of theses boss within a game.

3.1.1 Role and Purpose

The role and purpose of a boss varies depending on the game and genre but some of the core concepts are present in almost every game. Bosses tend to be difficult challenges or stronger enemies compared to the regular opponents faced in a game. They are often utilized at core points of the story of a game to raise the tension and test what the player is capable of. They often need to be defeated to accomplish a goal within the story, continue the progression or reward the player with additional tools and mechanics for the rest of the game. Bosses are often used to breakup the regular gameplay loop and introduce variations to it.

Mike Stout (2010) describes a boss as several different things at the same time. He regards bosses as a reward to the player as something new and different compared to the rest of the game to set them apart as a goal for the player to reach. He also presents them as a way for the player to display skills acquired in the game and a tension point that both

builds and releases tension through the built-up to the boss and the following victory over the boss. A boss is summarized as a test or milestone for the player and a story as part of the game itself as well as a story itself concerning the fight and its structure (Stout, 2010). Keren (2018) summarizes the purpose of a boss in a similar way by presenting the four categories: reward, dazzle, engage and challenge. He describes them each as a reward and a surprise to the player as a new experience, a way to move the story along and a skill test.

3.2   What is a Boss Fight?

The role and purpose of a boss can only be explained to a certain extent without detailing what a boss fight is. The boss fight describes the encounter between the player and the boss. Boss fights tend to be set up as combat encounters but the term boss fight has also been commonly used for difficult and unique challenges in games that do not employ the general concept of a boss. Besides the aforementioned exclusion a boss fight is often build in a similar way to one another. Mechanically speaking a boss fight is an interaction between the player and the boss in a space dedicated to the fight within the game. They interact with each other by using attacks and other actions to damage and hinder their respective opponent until one of them is defeated while traversing the dedicated space.

3.2.1   Structure: Goals and Beats

The structure of a boss fight often depends on the game and the genre the boss is part of. Stout (2010) generally describes the structure as story beats: Build-Up, Intro/Reveal, Business as Usual, Escalation, Midpoint, It's ON, Kill Sequence and Victory Sequence. He describes the Build Up as the process of incorporating the boss and hints towards the upcoming fight into the game world and story which is followed by the Intro/Reveal that then presents the boss with a unique way to introduce them to the player and start the fight. Business as Usual, Escalation and Midpoint and It's ON are detailed by Stout as the beats that primarily concern themselves with the process of the fight by first starting off with presenting the general mechanics of the fight, then introducing new mechanics which is followed up with raising the stakes off the fight and finally the entire set of mechanics is deployed against the player. Stout's two final beats are the Kill and Victory Sequence in which the player gets to enjoy his accomplishments by being presented the defeated boss while receiving a reward.

Birkhead (2012) alternatively presents the general goals of a boss fight from the perspective of a combat designer. Birkhead's goals are to present a conflict by depicting the boss in a unique manner, defy the expectations by offering unique designs and highlighting the differences between the boss and the rest of the game, overcome adversity through a bosses weaknesses, engender participation by promoting player interaction and skill and avoid complications by telegraphing core concepts of the boss before the fight starts (Birkhead, 2012).

For the purpose of later sections of this thesis both of these structural concepts can be united and modified into a general structure of a boss fight. This structure includes six parts which are an intro that promotes the boss before the fight, a display of the boss of the boss fights unique features and mechanics, a challenge that tests the player's skills and understanding of the game mechanics of the boss, a transition that increases the

tension, a test that deploys everything at the boss has at its disposal and finally a reward to let the player enjoy the accomplished achievement.

3.2.2 Boss Mechanics

A boss fight is partially defined by the actions available to a boss. Boss mechanics are primarily based around what a boss can do and what type of concepts they deploy or what is required to be used against them as part of their design. A bosses standard option for mechanics are attacks. Attacks are abilities deployed by a boss to either damage the player character or hinder their progress and access to tools to challenge the player. They are often telegraphed by auditory or visual cues to offer players a chance to evade. Depending on how threatening a fight is supposed to be their damage can vary. Sets of attacks are usually collected as patterns to be deployed given the context and phase of the fight. Part of the boss mechanics are also the boss arena or area which is designed to support the fight and potential actions that the boss can use. Boss mechanics also include specifications of the boss like health, resources and weaknesses. One of the core features in bosses is also the concept of phases. A boss can have multiple phases that are up to a certain extent different from one another. They introduce new mechanics and changes to the fight in general. Boss mechanics are further defined and elaborated on in the next major section of this chapter.

3.2.3 Principles

The principles of a boss fight describe the concepts that apply and surround boss fights. As previously described bosses and their surroundings require a unique design to lift them up over other parts of the game. This can both be done visually and mechanically. Keren (Keren, 2018) even suggests to break some of the rules of the game to make the experience more unique by offering new experiences. This should potentially be done with caution to not fully remove the boss from existing core gameplay mechanics of the game.

Challenge

One of the main principles that is seen throughout games that include bosses is that bosses appear as a challenge meant to test the player on their skills. Challenge is primarily connected to the difficulty of a game. As challenge and difficulty is often subjective measuring and designing for it proves to be difficult. The difficulty of certain game design choices can often only be tested through playtesting or prototyping these specific occurrences, this especially applies for bosses. The concept of flow by Csikszentmihalyi (Csikszentmihalyi, Abuhamdeh, & Nakamura, 2014) is often used to describe the concept of challenge. He defines flow as a state of focus between being overwhelmed and underwhelmed by an activity or challenge presented to oneself. A study concerning the role of challenge in gaming experience further suggests that challenge can not be considered as isolated part of games that can easily be designed around (Cox, Cairns, Shah, & Carroll, 2012). This generally sets an emphasis on the discussion surrounding difficulty and challenge in not only the context of game design but also boss design. Challenge as a principle for boss design thus needs to be taken into careful consideration on how it is handled.

Emergence & Progression

Another principle that influences the concept of a boss design are the two game design concepts of emergence and progression. Juul (2002) describes these two concepts as the two primary ways on how games are structured. He defines emergence as strcuture that is defined by a small number of rules that offer large numbers of variations of what can be accomplished in a game based on these rules. He also defines progression as a stucuture that requires the player to follow a set of actions and requirements predefined by the designers to accomplish the goals of the game (Juul, 2002).

Both of these structure appear in boss design. The gameplay within the fight and its stages is primarily emergent depending on how the actions of the player and the boss interact with each other based on set of rules defined for the fight. At the same time a boss fight usually follows a set of actions that are executed by the boss and a set of actions that is required by the player to defeat the boss. This follows a progression designed by the designer. How far a boss design leans into either structure is entirely dependent on the specific design of the boss.

3.3   Aspects

The following section describes a set of influenceable aspects of a boss fight that need to be taken into consideration for the boss design process. These aspects can be modified to fit the intention behind boss. At the same this section will attempt to define a common vocabulary for boss design by describing these aspects.

3.3.1   Time

The first aspect to be explored is time. In this case time not only refers to the duration or length of a fight but a manner of aspects of a boss fight that are dictated by time and its passage as well as the frequency of certain mechanics. This starts off with the outer limitations of a boss fight. These are set by the general length of a fight which is often defined by the amount of phases a boss has as well as the intended length of the fight by the designer. From a gameplay perspective this also influenced by the skill level of the player, the harder the fight is for the player the longer it may last. The aspect of time also concerns itself with the pacing of the fight as well as parts of the attacks a boss deploys. The frequency and amount of attacks a bosses in a given time fall under this category. Rest periods that are either integrated into the flow of attacks or the general progression of the fight also factors into this aspect, these are meant to give the player room to recover and replenish player resources.

3.3.2   Space

The aspect of space primarily explores the space of the area of the fight as well as distances and size differences between the boss and the player. The area of the fight describes any restrictions that are placed on it through obstacles and barriers, the line of sight and targetability of the boss and the player through actions and the intended distances between these two entities at a given time. This aspect also addresses boss mechanics by taking the size differences of the two entities into account, a boss usually be-

ing significantly larger than a player, forcing the movement of the player through attacks and dodging as well as forcing the player to move to certain locations to hit weakspots.

3.3.3 Player Resources

Player resources are defined all resources available to the player during a fight. This incorporates the statics and attributes a player character has during the game which could include health, mana and stamina which determine the players survivability and usage of abilities. This further extends into weapons, consumables to recover other resources, ammunition, tools and player ability choices.

3.3.4 Boss Resources

The aspect of boss resources include general resources similar to the ones of player resources like health and other resources at the disposal of the boss like additional enemies. They also include any type of boss mechanic like attack patterns, attacks and phases that define the nature of the boss. Boss resources are not only limited to attacks specifically but any type of action that can be taken by a boss.

3.3.5 Context and Challenge

The last aspect concerns itself with context and challenge. This aspect considers general game properties like the role and the story of the boss within the game, providing a context for the boss. It also includes the attempt at a difficulty given the context given within the game's progress. Clarifying the type of boss, these types include tutorial bosses, general bosses encountered in the middle, bosses at pivotal points int game, the final boss of the game and extra challenges for advanced bosses. Each of these types requires consideration for its different levels of difficulty. Additionally parts that fall under this aspect that could influence the challenge also include the amount of participants in a fight and the restriction and removal of player resources.

3.4 Categories

Bosses are used in a variety of different games showing up in a multitude of genres. They can primarily be defined by three categories. This includes patterns, the match-up and it's purpose. Similar to the previous section this section also attemps to further develop vocabulary for boss design.

3.4.1 Pattern

One of the main categories that needs to be considered for bosses is the pattern category. This category concerns itself with the pattern of attacks and actions a boss employs against the player. Patterns include all possible actions used by a boss. There are two primary types of patterns, fixed and random patterns.

Fixed Pattern

Bycer (2011) describes fixed pattern bosses as bosses that are designed around a pattern or a repeated set of patterns that are designed by designer to be deployed in a specific way which is used of unique enemies that require specific ways to beat them and with no exceptions to it. He argues that they are easy to design but lose their challenge as soon a they are understood.

Fixed pattern bosses follow predefined sequences the easier way to design them as described by Bycer corresponds with the concept of progression mentioned earlier in the chapter. This offers the the design process a clear way on how a boss is to be experienced by the player.

Random Pattern

Random patterns are also designed by game designer but are less rigid then fixed patterns. They follow more the concept of emergence mentioned earlier in the chapter. Bycer (2011) describes their core concept around randomized attacks that the boss chooses from instead of following a specific order which forces a more reactive fight that the player can be challenged with.

Employing random patterns adds more to the general difficulty and challenge of a fight as opposed to fixed patterns. Random patterns are more commonly used now then in the past at least concerning game from the action genre like Dark Souls (FromSoftware, 2011), a game famous for its use of bosses that all follow this concept of random patterns.

3.4.2 Match-up

The next category describes the match-up between bosses and players. This match-up follows a range of one boss to multiple bosses and one player to large groups of players within one fight. The most common match-up at the lower end of the spectrum is one boss against one player which is usual for singleplayer games. An extension of that is one boss against one player assisted by non-player characters within the game to assist them. This is then further built upon as one boss against a small group of player cooperating with each other. On the other end of the one boss spectrum are one boss against a large group of players, a setup predominately used by MMORPGs like World of Warcraft (Blizzard Entertainment, 2004) and Final Fantasy XIV (Square Enix, 2010) during the end game content in the form of raid bosses that require the coordination of a larger group of people. The spectrum of multiple bosses against a single player up to multiple bosses against large group of players appear throughout games as additional challenges. The match-up of boss fights heavily dictates the challenge, difficulty and what type of options can be used for boss design as multiple entities require additional considerations for how these bosses are designed.

3.4.3 Purpose

The last category describes the purpose of a boss in a boss fight. While two following types can be used exclusively, they can also appear in combination with each other.

Combatant

Combatants are bosses that predominantly fight concerning their game mechanics. The previous sections have mostly considered this type of a boss while considering aspects and categories. The design for this category of boss only revolves around the combat encounter between a player and a boss. The distinction is specifically made for cases where a boss does not only fight.

Boss as Puzzle or Level

Bosses can also serve as puzzles or levels or can include either of them as part of their fight. This is a less usual type compared to the previous one but needs to clarified for design purposes. These type of bosses are challenging encounters that force the player to solve a puzzle or traverse a level. They can be used in combination with the previous type which leads to a boss that resolves around a certain game mechanic or as part of a level. The game Shadow of the Colossus (Japan Studio, Team Ico, 2005) can be used as a close example for this, while you do fight the bosses within the game you do so by traversing across the large bodies until you find their weakspot instead of directly attacking them.

4 Boss Modeling

This chapter will explore the concept of creating a tool to assist with creating and analyzing a boss and the corresponding fight. This tool is meant to be applied at an early stage of the development process by offering a preliminary step for the prototyping process.

This will be done in the beginning of the section by looking at already existing ways on how to possibly aid in the process of creating a boss and its fight. The aspects and categories that were established in the previous section are used as a foundation and reference for this process. The three aids that are looked at for this are statecharts, UML and Machinations. The capabilities and options of each individual aid are looked at and analyzed on how a boss fight can already be created and to what extent. This will also take into consideration how core principles of each aid can then be built upon for the creation of the aforementioned tool to assist the boss design process.

The second section will then attempt to recognize the motivation and intent behind what the formerly described tool should accomplish. The purpose of the tool as well as early guidelines to its use will be defined.

The last section then introduces the prototype of a tool that is designed to assist with the creation process of a boss and its fight. A general description is given with its core principles and components which are then further developed within the remainder of this section. Components will be defined and visualized with examples of what they could look in the context of the tool.

4.1 Existing Options

In the following section an in-depth look is taken at three already existing ways that could support the creation process of a boss. This includes statecharts, UML and Machinations which are further elaborated on in their respective parts of the section. To see what can already be created with them and to what extent several of the aspects and categories established in the last two sections of chapter 3 will be taken into consideration.

4.1.1 Statecharts

Statecharts are an extension of state machines and diagrams created by Harel (1987) to turn them into a description language. Harel extends the original concept of state machines and diagrams by adding hierarchy, concurrency and communication to it. Statecharts describe the behavior of a system and attempts to confront the issue of working with and specifying reactive behaviour that is based on stimuli (Harel, 1987).

The behavior of a boss and by extension its fight can be displayed by using statecharts. This especially works when we take the three concepts introduced by statecharts into consideration. The general behavior of a boss can be described as a state that includes states that correspond to actions the boss can take for example moving, attacking and be-

ing idle. Besides the general behavior a boss has multiple parallel states charts, one for the phases and one for the boss area. The phases state includes multiple states representing each phase of the fight with each of them including multiple states representing an attack. Phases and attacks are are entered either by selection or condition entrance depending on what phase the fight is in as well on what criteria the attacks are selected. This can also be accomplished through statechart actions that function as intstructions within a state. The arena state could include states that represent the the position of the boss within the fight relative to the player. Whenever the attacking state is reached it sends a corresponding event to the phase state on entry to move to an attack pattern state that then triggers the event of attack state. Whenever the moving state is reached it sends an event to the arena state to trigger an event that switches to another state representing a new position of the boss in the fight. The general behavior states could potentially be triggered by events that originate from the arena state based on the states representing the relative location to the player within the arena. The statechart of the boss can be further expanded on by adding a state concerning the health of the boss as well as a state representing the player with its own states setup similarly to the states of the boss. The systems of the boss and the player could also be further subdivided to increase the complexity and detail on how the fight is represented at the loss of visual clarity and easy understanding.

Statecharts offer an in-depth and complex tool to describe the behavior of entities and components of a boss fight and their respective actions with its context. The use of concurrency and communication within a statechart and its subsidiary statecharts presents an opportunity to have all relevant parts visualized in one diagram while being to represent complex interactions between the different states. While the behavior of a boss and player can be presented comparatively easy, design choices and the intended progression of the fight are harder to display within the concept of a state chart.

4.1.2 UML

UML (Fowler, 2003) has been a staple tool for software engineering as a modeling language to design any type of software, offering a plethora of different types of graphical notations and ways to use it. Applying UML to game design has according to a review of game design methods and tools by Almeida & da Silva (2013) been suggested multiple times by authors for the game design process, the description of game elements as well as communicating within game projects. They also note that the general usage of UML for game design purposes would need to be further studied and modified to fit the specific requirements of game design (Almeida & da Silva, 2013). Sicart (2008) further describes the potential use of languages like UML to describe game systems and interactions within games.

As UML offers a variety of diagrams to choose from, several of them have to be taken into consideration for the boss design process. Different types of diagrams may be used for different aspects of boss design because not every diagram fits the concept of a boss as this both includes the boss fight as well as the boss itself. In the case of creating a boss and its fight the use of multiple UML diagrams in combination might be advantageous. The primary diagrams that could fit this process include class diagrams for the boss itself and sequence, communication, activity and state machine diagrams for the fight and certain aspects of the boss.

The boss and its components can be modeled with a class diagram. A boss is a class of its own with several attributes and operations. These can partially correspond to the influ-

enceable aspects. Attributes correspond with boss resources which can include a name, health, in what phase the boss currently is and a description of his difficulty. Its operations correspond with actions that it can take like moving and attacking. Other classes in this diagram are an opponent, an arena, an attack and a phase. The multiplicities between the associated classes depend on the concept of the boss. A boss can have one or more opponents, phases, attacks and arenas. An opponent which represents the player character has similar attributes as a boss which correspond to the player resources as well as similar operations like attacking and moving. Phases are reliant on a condition and a Boolean in the boss class to determine what phase the fight is in. Attacks are dependant on the attack pattern class that depends on the phase class. Attacks can also be assigned with attributes concerning damage and frequency. The arena can have attributes like size and shape, an arena can also have classes representing obstacles associated with them.

The boss fight has multiple interaction diagrams at its disposable that it can be displayed with. The options again include sequence, communication, activity and state machine diagrams. Sequence and communication diagrams represent similar concepts to one another as they are capable of portraying behavior through sequences of messages, sequence diagrams focus on the the sequence itself while communication diagrams focus on the links between the participants of a sequence (Fowler, 2003). They could portray some parts of a fight specifically the combat interaction between the player and the boss. Both can be displayed as classes and participants respectively. The sequence itself describes the calls that are made between the two for attacking, moving, getting hit and taking damage. Movement can also be shown by adding an arena class or participant that is sent calls for moving which then sends back a location as a response. Differences in the phase of a boss could further be portrayed in a sequence diagram through conditions tied to the amount of damage the boss received. These conditions could then include new calls representing the new actions a boss can take. The entire combat process in a sequence diagram can also be set inside a loop.

Activity diagrams in UML can be used to portray parallel behavior through action nodes, edges and their modifiers which assist in describing work flow and procedural or behavioral logic (Fowler, 2003). A boss fight could be portrayed with an activity diagram by representing the behavior of the boss as one activity made up of many actions and their subactivities. The actions a boss can take in its simplest form include attacking and moving. Attacking can be displayed with subactivities in form of a subsidiary activity diagram. An attack pattern for the attack is chosen as an action depending on a decision that is made by what phase it is as the input of the attacking action. The attack pattern then determines which specific attack action is picked to continue to the output of the attacking action. Parallel to the attack being chosen the boss can also move around the area. This in its core displays the basic activity a boss can perform during a fight. Additional complexity can be presented by using time signals to indicate delays in the behavior of the boss, creating further subsidiary activity diagram to represent more complex movement based on the position of the player or the sate of the arena and creating an additional activity diagram for the player character in similar fashion.

The state machine diagrams used in UML function similarly to the previously discussed statecharts based on them being an object-oriented variation of the statecharts created by Harel (Object Management Group (OMG), 2017). They thus fulfill similar requirements as the statecharts looked at in the previous section as they are capable of showcasing the advanced behavior of a boss in more detail.

Bosses and boss fights can be displayed as models through UML by using a state machine

or an activity diagram for the boss behavior, a class diagram to model a boss, its fight and components and a sequence or communication diagram for specific behavior of the boss in individual cases. This primarily helps with displaying an abstract model of a boss and its behavior throughout a fight with a multitude of available tools that can be transferred and adapted to the actual use case within the game. Logical and mechanical components are sufficiently explored with these options which fits the purpose of UML diagrams. Other aspects of boss fights that concern themselves with design choices and the context of a boss can only be displayed to a certain extent within UML diagrams but complexity can be added with additional advanced notations and more complex interactions depending on the type of diagram that is used. Some of this can also be addressed by having a huge variety of diagrams options and combination thereof to display a specfic boss and figth.

4.1.3 Machinations

The next tool that will be looked at is Machinations. It was originally created by Joris Dormans as part of his doctoral thesis and then further built upon and made accessible through Machinations S.àr.l. (Machinations, n.d.). Machinations is a formal framework created to display and simulate the economies of games, It is a visualization tool that uses diagrams to represent game mechanics and their connections within the game system (Dormans, 2012).

It was created by Dormans (2012) to both help the creation and the analysing process of game mechanics through drawing diagrams, collecting data about them and being able to simulate them without having to implement any of them. Dormans describes feedback as an essential part of Machinations as the feedback process during the simulation is the part where each component is updated based on the changes of other components.

As a visualization language Machinations follows a set of predefined rules and guidelines. The main components of a Machinations diagram are resources, connections and nodes. Resources are an abstract component that can represent any type of interactable part of the game mechanics that are actively used or influenced. Connections are transitions between nodes that either portray the movement of resources or changes applied to other components. Nodes display how resources are used and stored as part of the game mechanics. Connections and nodes additionally have labels, modifiers and secondary components that influence the flow of resources. The combination of these components and their additions make up a Machinations diagram which is then iterated over given a time step or trigger until a condition that ends this process is met.

A Machinations diagram in its most basic form showcases the flow of resources that originate from nodes, these resources then are transported by connections through other nodes until they end up in a specific node. These nodes and connections have certain properties applied to them that influence how and if resources move. This process is repeated for a defined amount of time after which the results of the process are displayed.

Machinations primary purpose of designing game mechanics can be used for a huge variety of different systems. Applying this tool on the concept of a boss and its fight offers a strong design method for the mechanical parts of the fight. A boss can both be visualized and simulated to a certain extent, taking the previously established influenceable aspects in to consideration. A simple version of a fight can already be displayed by giving each participant of the fight a node representing their health. This can be done by treating

the health of the boss and as an extension the health of the player character as resources that are placed in their respectively labeled nodes to be reduced. Any additional players or bosses can also be placed as an additonal node. Boss and player actions can also be described as nodes that send out resources as part of their actions. They are also given a source node for sending out resources to trigger any node that is supposed to represent an action the participant can take. This can then for example increase or reduce the resources of other nodes. In the case of a boss fight this can include an attack. The participants in this case attack each other until the health resources are depleted by sending resources originating from their attack nodes to the opponents health node, reducing the resources placed on it. More complexity can be added to this by adding a probability property to the node that represents an attack to simulate the concept of hitting or missing a target. A probability can also be added to the action nodes that influence which one is triggered to create a random pattern, a fixed pattern could be accomplished by only allowing certain nodes to be triggered before the other are triggered. Attacks can further be differentiated by the type and amount of resources that are used to portray the amount of damage, attack frequencies or status effects that attacks could apply. The different phases of a boss can also be visualized by using different sets of nodes that represent the additional actions provided. Each set of nodes could be moved to by a trigger attached to the health node of the boss. Other influencable aspects of a boss like player resources and boss resources can also be represented by additional nodes. The challenge of a fight can be adapted by increasing or decreasing the amount of resources that are used for attacks and health, differences in the arena can also be taken into consideration through modifying nodes or using another set of nodes depending on what components are influenced by the differences.

For the purpose of designing a boss, Machinations offers a strong suite of components to display the game mechanics of a boss fight. Almost every aspect and category that was previously established can be addressed successfully. A boss fight can not only be displayed but also be simulated and afterwards analyzed on how the resources flow throughout the diagram. This offers a way to see how each individual component affects the entire system designated to a boss fight. While each component is addressed mechanically the tool's capability of displaying each component and how it is supposed to look within the fight is dependent on the depth and complexity of the diagram. Game mechanics concerning a fight can successfully displayed by Machinations which aligns with the core focus that the tool was designed for. Corresponding to that the design of the boss including its context within the game and the specifics of how certain actions and the environment of the fight are supposed to function and look from a design versus mechanics standpoint are only addressed up to a certain extent.

4.2 Motivation and Intent

The motivation behind creating the Boss Board, a tool to assist the creation and design process of a boss, is to offer a model that creates a common foundation for boss design that can also support the communication about the process. This tool in combination with the aspects, categories and principles defined in Chapter 3 are meant to provide a shared knowledge base, vocabulary and understanding of a boss and its components. This intends to further offer a design language for bosses.

The use of models for systems is widely used in software engineering with examples like

UML. Applying models to the game design process is also not a novel idea. Creating a design language to create a model of a boss appears as an ample solution to assist the boss design process. To create the tool to represent such model key characteristics of model-driven development can be taken into consideration for the tools guidelines. Selic (2003) describes five characteristics for the quality of a model which are abstraction, understandability, accuracy, predictiveness and inexpensiveness.

The Boss Board is intended to be used during design process. The development cycle of games spans over multiple phases made up of the concept, pre-production, production, QA and maintenance phase (Fullerton, 2014). The intended deployment of this tool is meant to happen in the pre-production phase and parts of the production phase to assist the development of bosses as game features. Fullerton (2014) describes the pre-production phase as the phase that includes the prototype as well as design planning docs of a game project while the production phase concerns itself with implementing features. The tool that will be described in the next major section fulfills the role of a prototyping tool or preliminary stage to prototyping of features depending on what type is used. Prototyping is an essential part of the development cycle of games and is generally used to test out features and parts of the game before proceeding to fully implement them. Schell (2019) emphasises the use of iteration in the game design and development process which corresponds with the nature of prototyping. He describes how existing development processes already incorporate an iterative nature.

The Boss Board is supposed to offer a way of iterating over designs as part of designing a boss. As a prototyping tool it bridges the gap between a physical or paper prototype and a digital prototype. Paper prototypes can have variety of complexity reaching up to fully fledged tabletop games but are limited comparatively to digital prototypes (Ham, 2015) Ham (2015) describes their advantages as being easier and faster to create and iterate over to achieve quicker results but their limits include that they can only represent the feature that is prototyped up to a certain extent. He further describes digital prototypes as computer-based representations of a video game that provide insight into the usability or aesthetics of what the prototype is made of as it created in the same medium as the game. The Boss Board is intended to be a representation of a paper prototype by emulating a similar modular nature while offering the usability and visualization options of a digital prototype.

The iterative idea behind the Boss Board corresponds with the concept of fail fast. Modifying this concept described by Shore (2004) for debugging software as part of software engineering by adding parts to the code that help find errors and bugs more quickly. Transferring this idea to a tool like the Boss Board works in the way of offering the designer ways to view a boss and its components in one place and see their connections so issues can be found easier and quicker.

4.2.1 Unified Design Language

As previously mentioned part of the motivation and intent behind the creation of the Boss Board, and by extension the aspects, categories and principles that were defined in the previous chapter, is the creation of a design language for bosses. This is on one end done by the creation of a game design tool specifically created for boss design. On the other end is pushed by the creation of a shared vocabulary and knowledge base.

There are already approaches to create tools and frameworks to make the discussion

about game design easier and more precise like the MDA framework (Hunicke et al., 2004) and FADT (Church, 1999). Both of these frameworks create ways to make communication about game design easier by introducing common concepts and vocabulary concerning their respective frameworks. The FADT derives its concepts and vocabulary by analyzing existing games and reducing them down to these concepts and vocabulary (Church, 1999). MDA emphasises the impact of each of their aspects, mechanics, dynamics and aesthetics, on one another through iteration to create a design process (Hunicke et al., 2004).

The Boss Board attempts to follow similar objectives by defining fundamental sections and components that can make up a boss and its fight to represent any type of aspect and category relevant to them.

4.2.2 Visualization and Clarity

Another key intent behind the Boss Board is providing a method of visualizing a boss and its components while offering clarity and a clear overview of a boss fight within one tool. This provides the ability to both create and analyze a boss accordingly. The tool should be able to provide anything that is need to make conscious design choices to display intended design specifically.

For the creation of the Boss Board a visual overview and the fact that everything should be available to the user in one place were set as core principles. The board can be iterated over by replacing components. This corresponds to a type of game design document and process presented by Librande (2010) as One Page Designs. Librande describes a set of design documents that he created for different projects that included every component or as many as necessary and relevant to the game and its design in one large unified document which were physically placed in locations that were easily accessible by members of the development team. He designed them with visual representations of the game's design. The document was replaced with new version whenever the space of the document was no longer enough to fulfill the required specifications or changes were made to the design that significantly reshaped how components functioned (Librande, 2010). This type of document fills a similar space as the Boss Board by providing a shared overview of the design space while being modular and replaceable in nature.

4.3 Boss Board

The Boss Board is a modular board that is supposed to create a shareable overview of a boss fight and all of its part without actually having to implement said boss fight. It can be regarded as a preliminary stage to prototyping. The appeal of the Boss Board board is that everything is on one board that can be modified as well as having everything in one place or at least as much as is needed for the given purpose of the project. The board is split into multiple zones each representing a part of the boss fight as well as its core aspects. Sections are interchangeable and optional depending on the scope that is wanted for the fight. The following section will summarize and visualize different zones, elements and core principles of the Boss Board. Bosses, players and player characters are primarily referred to as single entities but the same guidelines also apply to multiple entities in the same fight. An example for a potential layout can be seen in Figure 4.1.
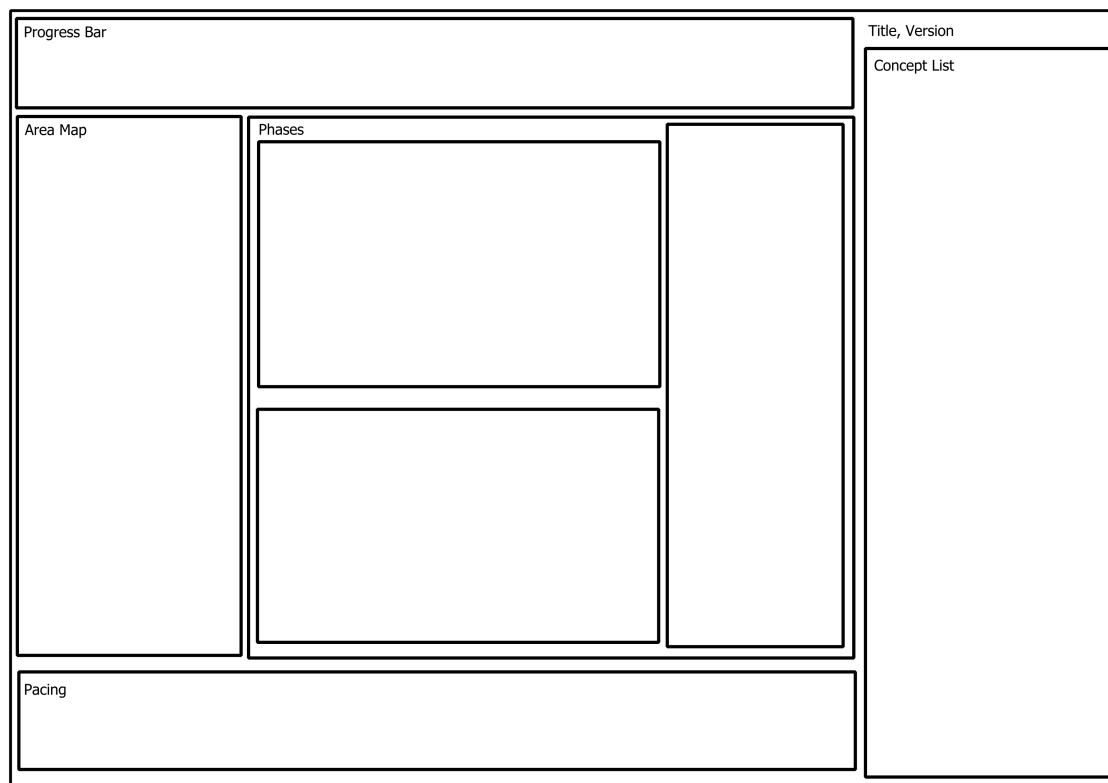
Progress Bar

Title, Version

Concept List

Area Map

Phases

Pacing

Figure 4.1: Example of a possible board layout

4.3.1 Core Principles

The Boss Board is primarily intended to be a tool that can aid in the process of creating a boss and its fight while offering ways to visualize and evaluate individual parts at the same time. The intention is to foster conscious and intended design by both supporting the creation process and the analyzing process of what was created. Zones and components offer specific spaces and visualization options to design the fight. Everything that is used is supposed to then be able to be viewed in one space in form of the entire board. This offers the ability to analyze and evaluate the entire boss fight by connecting each component with one another.

A core aspect of the board is its modular essence with a focus on choices and optional components. Components can be added and removed based on necessity and the scope of the project, without compromising the rest of the board. Several components and zones in the following section are vital for the understanding of the boss fight but their placement, specifications and details are primarily optional. It is entirely dependent on the user on how detailed each zone needs to be to display the intended design of the boss. The placement, size and shape of each individual zone is also not predefined but suggested. The board's modular nature also encourages the process of quickly changing parts of the board until reaching the desired result without having to revise the entire board.

The board also puts a focus on interconnection. Different zones and parts of the board can be marked with a multitude of components that refer and correspond to other parts and zones of the board to display any aspect of a boss fight that influences another one.

This is primarily intended to serve as visualization while keeping track of any potential chain of changes that might occur when a part of the board is changed.

4.3.2 General Components

Boards are primarily made up of multiple zones but before the zones are specifically explored several components need to be defined. These components are universal for all boards independent of what type of zones are used and to what extent. As previously mentioned even these components can be optional depending on the amount of detail that is required.

One of the first components besides zones is the board's naming structure. A boss is given a title or name as an identifier within the game as well as the design process. Additionally a number can be added to keep track of different versions across multiple boards if the boss fight goes through several major iterations that need to be kept track of.

Several parts of the board may require transitions between different elements especially within the zone. These transitions are made with arrows or lines to connect to elements with each other or pointing at an individual element. Any requirements for the transition are displayed on the transition component in the form of a condition or instruction. They can additionally be marked with a time frame or duration depending on the context of the transition. Events that are noted down on other parts of the board can also be attached to transitions by using event points.

Event points are markers that specifically refer to an event during the fight. They are either given a number, symbol or other type of identifier to connect them to the specific part of the board that they are connected to. An event point is placed at both the part that is referred to as well as the part that is referring to the other part. The number system is primarily used for each phase of a boss fight to keep track of any switches or changes to the phase or caused by the phase as seen in the phase zone section.

These markers as part of the board are referred to as tokens in the rest of the section. They serve as identifiers and referrals for and to other sections of the board and are mostly defined in their respective zones. Tokens that refer to a player and a boss are assigned a P or a B with the addition of a number depending on how many have to be displayed or are participating in the fight.

Another type of markings are icons and symbols. They are used to signify and describe player or boss resources and status effects that appear in the fight. Besides the symbol itself names and descriptions can also be applied. Icons and symbols that represent resources can have alternative versions that are crossed out to indicate their removal either permanently or temporarily. Any type of time related instances can also be marked with time symbols that either stand for certain time frames depending on context that is further elaborate in the relevant zones.

The last general component is zones. Zones represent the individual parts of the fight and the boss as well as its aspects, they can be added and removed. Their shape and size is dependent on the detail required. They include their own components that relevant to display their aspects. Each zone is distinct from one another and has its own rules and guidelines but zones and their components can interact with other zones and their respective components.

4.3.3 Area Zone

As previously shown a significant part of a boss fight is its area. Given the importance of the map for understanding the fight, it receives its own space on the board. The area zone is marked with the general dimension and perspective of the boss fight in either 2D, 2.5D or 3D. Additionally, any area map is also marked with perspective it is viewed from. An area consists of its space and any other type of obstacle. It is typically an open space that leaves room for the bosses' attacks as well as space for the player to traverse. The area itself is heavily dependent on the concept of the boss fight, maps will look drastically different depending on the genre of the game and the nature of the boss. Its shape is customized and often depends on other factors like the size of the boss, any obstacles or the level design. The area can be displayed in a variety of shapes. The edge of a map is marked with a double rim while obstacles are displayed with a diagonal line pattern inside the shape. Hazardous areas or a lack of ground are marked with a dotted pattern within the shape. The depiction of the area can also include the initial start positions of the boss and the player character relative to the environment as well as the direction they are facing. This is shown in Figure 4.2 Any changes to the map during the fight are shown in further iterations which are all displayed in the the area zone. This also applies to the transition or switch between entirely different areas during a fight.
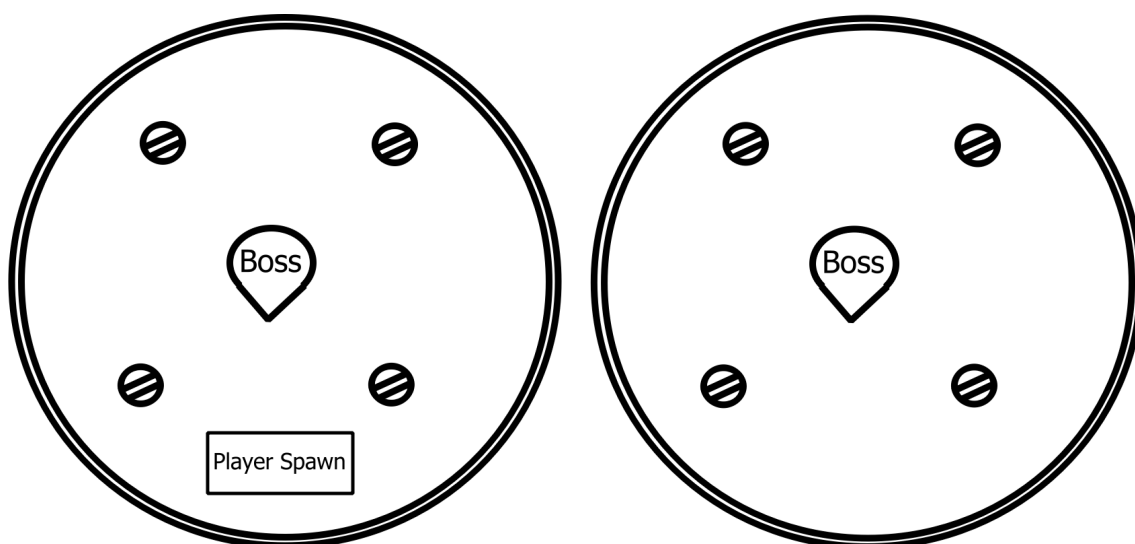

Figure 4.2: Example of a possible area setup

4.3.4 Phase Zone

One of the major zones of the Boss Board is the phase zone. The phase zone is the primary place for the gameplay mechanics the boss uses during the fight. Bosses tend to have multiple phases or stages with differing mechanics. Each phase is labeled with a number token corresponding to it. A phase entails several different attacks or patterns as well as general progression and behavior during said phase. Arrows indicate any relevant transitions between the stages within a phase and any transitions between phases. An additional zone within the phase zone is reserved for any information that is relevant across phases, this can for example include behavior or attacks.

Phases

A phase generally encompasses a significant portion of a boss fight, which is usually made up of multiple individual phases. The majority of the phase zone is thus made up of the individual phases and their inner components. As previously described a phase is labeled with a number token. This is not only limited to the phase itself but the number token can be used to connect them to other specific parts of the board that either need to be specified or put into relation with the relevant phase. This primarily serves as an indicator for the overall structure of the boss fight including when each phase is in progress, when a switch between phases is supposed to take place, when a certain event should happen or if there are any changes to a specific component like the map during each phase. Additionally a phase can also include conditions that are tied to other aspects of the board. These conditions can be placed adjacent to the identifying number token or in relation to any specific component in the the phase.

Any given phase in the phase zone is made up of an assortment of gameplay mechanics and behaviours that dictate the experience during the fight. This usually refers to any type of attack or action that is used by the boss against the player which in the following is referred to as boss actions.

Boss Actions

Boss actions make up the majority of the content of a phase. A boss action is given its own space in the shape of a rectangle that includes the specific action as well as any transitions, conditions, weights, threat levels or other relevant information for the action. The usage of all that information, any indicator or transitions is entirely dependent on the boss itself given its categories and aspects, each requiring a different combination. As shown in Figure 4.3 a boss with a strict pattern also follows a strict order of boss actions while a boss with a random pattern does not require the same. If a phase starts with a specific action it is marked with an arrow pointing at it.

More often than not a boss action is an attack against the player or the general environment. Besides that they for example include the creation or destruction of obstacles, the blocking of player resources, the regeneration of boss resources or the addition of boss resources like minions. To display these actions, especially attacks, a copy of the general area map is used as well as the position of the boss, usually in the center unless otherwise relevant. The effect of an action is indicated by its purpose and a shape that describes the area of effect or the hitbox the action takes up. Certain actions can be color-coded or be given a pattern within the shape to further contextualize aspects of the action in combination with any other relevant information within the boss action. The purpose or intended objective of a boss action is noted down with a descriptor or a further explanation within the boss action's rectangle if required. This could potentially apply for cases in which a boss action has an effect or role within the fight that is beyond what is shown in the depiction of the action. A threat level can be applied to an action to further specify the level of impact it is supposed to have on the recipient or boss on a scale from 0 to 3, going from non-threatening to highly threatening based on the effect that the action is applying to the progress of the fight. This includes modifications of resources on either side of the fight or attacks in general. A threat in this case is seen as an occurrence that can change the current development of the fight. A boss action is generally assumed to be indicative of the very instant the action takes place unless noted otherwise by a timer or duration.
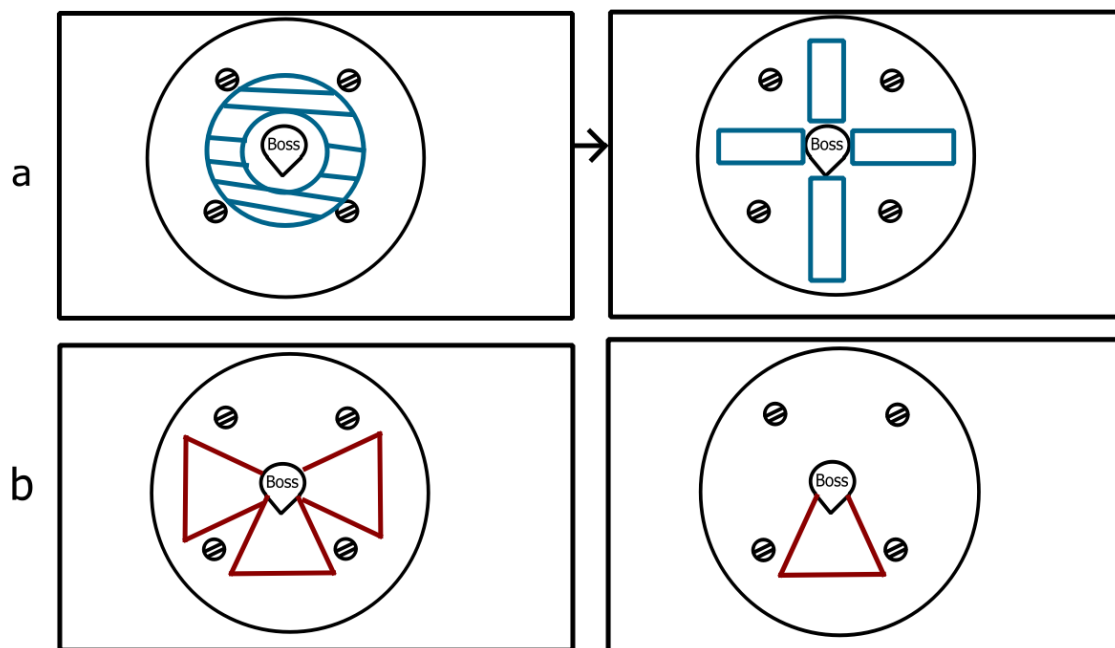
Figure 4.3: This figure shows multiple examples of boss actions while a. shows a fixed pattern and b. shows a random pattern of attacks

They can take place in relation to several factors like the position of the target, the direction the boss is facing or the direction the action is travelling in. This is shown by an arrow indicating the direction or a player token within the area. Any boss actions that are part of a direct sequence are connected with a transition that can include an event with or without a condition attached to it. Depending on the type of pattern the boss uses for its actions, either a fixed or a random pattern, weights can be applied to specific actions. This determines the probability of an action being picked out of a set of actions that are either all part of the same phase or a marked as set of actions. They can furthermore be affected by conditions. Weighted actions can be used for regular boss actions as well as attacks.

Attacks are a specific form of a boss action that appear in a variety of shapes. Direct attacks or continuous attacks are shown as shapes adjacent to or leading away from the boss. Any sort of projectile or progressively moving attack is displayed by multiple spaced-out shapes with an arrow attached to the shape pointing in the direction the attack is travelling. Attacks that cover certain areas for a duration are marked by shapes with a hatched pattern. Additionally, attacks have different indicators and descriptors that can be applied. A damage scale between zero and three is given as an option to describe how much damage an attack should deal in relation to the average player character at the time the boss is encountered. This scales from no damage to an instant death. An attack can also apply a certain effect besides damage, any type of effect can be set as a description within the boss action space.

4.3.5 Pacing Zone

The pacing zone is a unique zone as it can either be its own zone or be incorporated into the phases of the phase zone. The purpose of this zone is to showcase the general pacing

and intended development of the fight and its phases. This can be achieved by noting down the behavior and intended steps on how to fight is supposed to proceed. The amount of detail that is needed for the pacing can be adjusted by either only using the pacing zone as a mostly independent zone or by adding a pacing section to each phase within the phase zone. The pacing is made up of several individual parts that stand for the type of behavior, actions and transitions displayed during the fight. Transitions are breaks or switches during the fight that can happen when one phase switches to another. This also includes rest periods that are specifically made for a player to be able to recuperate and have a short window of time to rest. Specific events like phase transitions can also be marked with tokens corresponding to the phases and any other relevant zone of the board like the progress bar. The aforementioned parts can be displayed with different options ranging from text to symbols or shapes, they can also be used in combination with arrows and other markings.

4.3.6 Progress Bar

The next major zone is the progress bar as shown in Figure 4.4. The general purpose of the progress bar is to visualize the progression of the fight with its milestones, events and intended flow of key points. The progress bar can be made up of one or more relevant lines. A line within the progress bar is made up of the line itself, sections and their indicators, event points and tokens that correspond with other parts of the board. Event points are not only limited to one specific line, they are also capable of overlapping over multiple lines corresponding to shared instances concerning the lines and the event itself. In addition to the aforementioned components and their purposes, the progress bar is also used to place intended barriers. Barriers are highlights to indicate which part of the fight is supposed to be a challenging obstacle that could hinder the player's progress and thus potentially force them to have several attempts at beating the boss before they actually succeed.

Two core components of the progress bar are the timeline and the lifeline, both can either be used in combination or on their own. This choice entirely depends on the specific boss fight and its concept.

Timeline

The timeline can appear in multiple formats depending on how precisely time factors into the boss fight. The progress bar is not limited to only using one timeline within its zone but multiple ones with different levels of precision or varying tasks. It's main purpose is to describe the progression of the fight over time. The precision of the timeline depends on the context of the boss fight as well as the purpose that the time plays within the boss fight.

There are two primary ways of application for the timeline. One is a strict and more precise usage that applies in cases of the fight or certain aspects of the fight running on a set timer. As bosses are in most cases not limited by or set to a timer, it is often not necessary to have a precise time for the entire fight unless the fight does run on a timer. A precise time is more applicable for mechanics that happen after a given amount of time has passed or to set an indication for the intended design for how long the fight is supposed to go on average. Timed mechanics in this case both apply for the entire fight as well as individual phases which can be marked with the corresponding number

tokens set in the phase zone. The other application serves as a rough outline and is looser regarding its precision. Its main purpose is to give a guideline or an orientation for how the boss fight is supposed to play out and when something is supposed to happen during the fight. This also includes what is supposed to happen at those indicators by either referring to other lines, to barriers or other events, usually in combination with other zones.

Timelines can be paired with other lines, as previously mentioned multiple timelines can be applied simultaneously to display varying forms of time during the boss fight. They can also be used with the addition of a lifeline.

Lifeline

The lifeline represents the more common approach to displaying the current progress of a boss fight compared to the timeline. It depicts the health of the boss during the fight and can set certain milestones that are reached in the process. The milestones are triggered by the boss losing a certain amount of health in a not necessarily specified time frame. Milestones can be shown by using a variety of health display options depending on the context and nature of the boss. The various options include percentile of the full health, amount of health bars or set number values. A milestone is marked with an event that can refer to planned barriers or to a transition to another phase as phases are often tied to the health of a boss. These specific phase events are marked with the number token of the corresponding phase. Similar to timelines lifelines can also be used in combination with additional lifelines if a more detailed view of certain phases is required. This applies for situations where a phase could potentially be significantly different from the other ones and thus needs to be displayed separately.
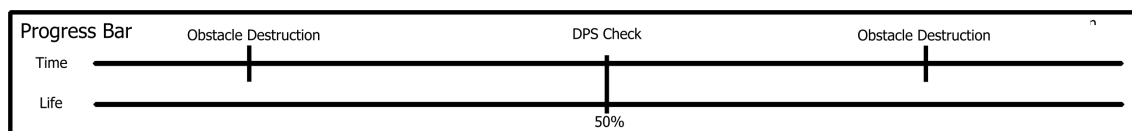

Figure 4.4: A progress bar with one timeline and one lifeline, the lifeline has an event at 50% that is shared with the timeline, towards the start and the end of the fight two more events are planned on the timeline

4.3.7 Concept Zone

The last major zone is the concept zone. It primarily describes the purpose and core concepts of the boss fight. This is done in the form of a checklist and additional descriptions. This sets up the core concept of the boss as well as its categories. Relevant themes or roles in the context of the game's story can also be marked down in this zone. The purpose of this zone is to display the intended design space of the boss so it can later be compared to the rest of the board to see if the specific zones still adhere to the original intention. Any deviation of the original intended design can then be adjusted or the intended design can be revised to fit what was created. The concept zone is made up of several different subsections.

Context

The first section of the concept zone is the context section, providing the context of the boss within the game. This encompasses the role of the boss in the game as well as any relevant themes or story elements in keywords. A boss can only be created in a vacuum up to a certain extent but as shown in previous sections bosses are intrinsically incorporated as a part of the game and can not function without its surroundings. The general design of a boss both mechanically, visually and generally within the game are shaped not only by what the boss can do or is supposed to do but also the story of the boss within the game as well as the themes surrounding the boss. These aspects of the context are noted down as short keywords or descriptions within the the context section to preliminarily set a tone and boundaries for the rest of the board.

The primary context for this section is set by where the boss is placed within the game as that majorly influences further parts of the concept zone. A boss can be placed in multiple different parts of the game and the requirements for a boss for each of these parts are drastically different. Bosses that appear for example in the tutorial of a game need to be treated differently than bosses that serve as the final boss of a game. The main way to differentiate between these parts is by categorizing the bosses in to the following type: Tutorial, General, Focal Point, Finale and Extra Challenge. Depending on what type is chosen for a boss, other sections of the concept zone are influenced.

Categories

The next section of the concept zone is the categories section. A boss can inhabit multiple different categories that define its context. The specifics of these categories were previously discussed as part of the categories of boss fights in the previous section about boss design. They include the nature of the boss, the amount of players versus the boss and what type of pattern is used for the boss actions and attacks.

The first category is the nature of the boss. This describes the core setup of the boss but this does not mean that the boss is only limited to that specific nature. The category distinguishes between a boss as a combatant and a boss as a puzzle or level. A boss can be either one of the two or even both of them at the same time depending on the fight itself, this is displayed by either marking both or adding one of them as secondary option. The influence of this category is mainly seen in the phase zone as well as the area zone, each boss requiring a different type of area and types of boss actions within the phase zone. This can also affect the progress bar given how a fight is supposed to be solved.

The second category is the amount of combatants during the fight. This ranges from one player to large groups of players and one boss to multiple bosses in one fight. Multiple bosses in this case are treated as a shared entity within the rest of the board. This category does not include additional boss resources that are added during the fight and only concerns itself with the initial setup. This is displayed by leaving option to describe how many characters participate in the fight.

The third category is the type of pattern which is used for the boss actions and attacks. This is either a random pattern or a fixed pattern. This category primarily affects the boss actions within a phase in the phase zone. Fixed patterns require the use of boss actions in sequences while random patterns require the use of weighted boss actions.

Challenge

The challenge section deals with the general difficulty of the fight as well as any type of obstacle or event that is meant to test the player or add an additional challenge to the fight. The difficulty is set by a difficulty level from 0 to 4, this is done in consideration of the type of boss that was chosen in the context section. Each type of boss requires a different level of challenge depending on its context, a boss of the tutorial type will require a lower level compared to a boss that is meant to be an extra challenge. In addition the a general difficulty level of the fight, the intended average amount of attempts a player is supposed to have can be set to further define the difficulty. Difficult parts of the fight, boss actions or changes during the fight, can be marked with a challenge token in the phase zone and progress bar at the relevant actions or events that could potentially trigger the end of an attempt for the player at the boss fight or create significant changes.

The use of challenge tokens is also part of the fail states. A fail state is any type of event that can potentially end the fight before the boss is defeated. This generally is the death of the player's character or the inability to fulfill a certain objective that an event sets. They include tests that are required to be passed before the fight can continue regularly or be terminated prematurely. Tests are meant to test the player's abilities, damage, understanding of new and old mechanics or any other characteristics. They are defined in the fail states by a keyword, short description and a token that can be set at relevant points on the board.

Characteristics

The last section of the concept zone is the characteristics section. It mainly encompasses any additional information that is relevant for the boss and the fight. It sets the outer boundaries of the fight that are not fully described within the other sections of the concept zone.

One of the primary characteristics is mechanics. Any specific or noteworthy mechanics that are used during the fight are given a keyword, a token and a short description. The tokens are then placed on the relevant parts of the board that they correspond to. This part also entails whether a mechanic in the boss fight was previously introduced during the game or if it is being introduced as a new mechanic during the fight.

The characteristics section also describes the external specifications of the fight. The first characteristic to be noted is in what dimension and perspective the fight takes place. This usually is the same as the rest of the game but can specifically be different for a boss fight. The specified dimensions are 2D, 2.5D and 3D. This is also marked on the area zone to establish the perspective of the fight. The intended length of the fight is set in this part of the board which is then also transferred to a timeline in the progress bar if needed. It also lists any kind of exposition for the boss and boss fight in form of cutscenes, texts or titles that are used surrounding the fight. They can additionally be sorted by when they appear which either happens before, during or after the fight. Additionally, any prerequisites the player needs to fulfil to initiate the fight as well as any rewards that the player receives after the fight can be noted down in this section as well.

5 Discussion

This section will now discuss the the results and work of this thesis. Chapter 3 attempts to create a common design language and vocabulary by defining and listing several principles, concepts, categories and aspects. While many of them are listed and elaborated on a more thorough and unified depiction could offer more precise results. Comparing this to the MDA (2004) and FDAT (1999) described within the thesis leaves room for more specific and less bloated groups of aspects. The groundwork is done to create a knowledge base for the vocabulary of boss design that be further elaborated on with more precise definitions and examples.

Chapter 4 builds upon the foundation provided by 3 applying the concept of a boss fight on existing tools. The tools chosen primarily offer solutions for representing the behavior of a boss fight and some of it components but are generally more oriented towards game mechanics over other design functions. These findings where then attempted to be incorporated and addressed throughout the motivation and intent as well as in the creation of the prototype of the Boss Board. The Boss Board approach leaves the question of its actual use as a tool for general boss design. While the tool addresses both game mechanics and design choices they are displayed on a more surface level compared to the depth of behavior and mechanics visualization that can be done with the tools that were looked at. This is not necessarily a negative aspect of the Boss Board, it just fulfills alternative purposes on top of some behavior and mechanics visualization. As the tools not been extensively tested for its application it unsure how much to tool can actually provide for game and boss design or if provides a more academical approach to the concepts of bosses that is not entirely feasible for the design process within a game. It also needs to be pointed out that while the tool was intended for the boss design process it might be too specific in its application and the process of learning it might not be as worth as learning other tools that can provide similar but less specific use cases.

The Boss Board delivers in its intent to create an easy overview that collects everything in one place. This could potentially foster easy communication as everything is also defined within the board. It offers a variety of options and alternatives on how to display each component while its iterative and modular nature can offer countless approaches to how a boss is designed. A danger within this lies in not having enough precise visualization and notation guidelines on how exactly to portray a boss and its fight which could lead to the opposite result of the Boss Board not being able to provide a shared common ground. While using the boss board one also has to taken into consideration that the portrayal of bosses heavily favors action based genres and their boss fights over more general and varied options. The argument can be made that it leaves enough options to develop any type of boss but with less precision over others, unless the concept is further developed on. The tool and knowledge base also walks the line of simplifying complex concepts while at the same time adding complexity to easy concepts by incorporating them all into one space. The tool sets limitations and guidelines for the general process while leaving enough room for the creative process of creating a boss that can be adjusted with more individualized visualization options based on modularity and the more open notation

approach.

Generally speaking a successful attempt was made in furthering the study of boss design in a different approach to the ones discussed in 2 by providing not only a tool capable of displaying the design process of a boss and fight but also starting a common vocabulary of boss design.

6 Conclusion

The goal of this thesis was to create a prototype of a tool that can assist the boss design process while also exploring the design concepts of bosses and boss fights by creating a shared knowledge base, vocabulary and design language by both categorizing and defining core components and concepts of bosses and their fights. The thesis displays a collection of aspects, categories and principles that are part of the boss design as well as the general structure of what a boss can look like. The attempt to define a common language was successful in that it has started the groundwork of creating this knowledge base by providing an overview of what exists and of what is possible.

A relatively simple tool, the Boss Board, was created with a visualization notation that offers enough guidelines to successfully use while leaving room for individual adjustments and further improvements. The tool was created with the knowledge base and a variety of game design concepts as considerations for its foundation.

Several suggestions can be made on how to further proceed with the results of this thesis. The Boss Board itself in its current stage is only a prototype of a tool. The possibility of expanding the tool into different spaces as a fully fledged tool is one way to further expand on this work. Its nature as a prototype tool and as previously mentioned a bridge between a paper prototype and a digital prototype can be elaborated on by creating both and editor and physical copies and components of the concept. This could lead to further studying and testing the tool for its actual use case. In addition, combining the tool with other ways of testing the produced model through processes like simulation and unit tests for specific scenarios concerning phases and attacks could also improve the Boss Boards capabilities.

The attempt at creating a design language, vocabulary and knowledge base can also be further elaborated on by incorporating and modifying already existing concepts of defining game designs. Game Design Patterns (Björk, Lundgren, & Holopainen, 2003) that takes game mechanics and game concepts and transforming them into basic patterns to create a unified vocabulary. This could be taken as a model and example for creating similar patterns applied to bosses. Specific types of bosses could be defined by their aspects and categories to create boss patterns. These boss patterns could then by extension be applied on the Boss Board to create templates of boss types.

Further attempts into properly conceptualizing the connection between challenge, flow and difficulty in the context of bosses could contribute to a better unified boss design language.

The core problem statements were addressed within the thesis. The suggestions made here can be further investigated to expand on what was found. This could potentially further develop the knowledge base of boss design.

List of Figures

Bibliography

Agriogianis, T. (2018). The roles, mechanics, and evolution of boss battles in video games.

Almeida, M. S. O., & da Silva, F. S. C. (2013). A systematic review of game design methods and tools. In J. C. Anacleto, E. W. G. Clua, F. S. C. da Silva, S. Fels, & H. S. Yang (Eds.), *Entertainment computing – icec 2013* (pp. 17–29). Berlin, Heidelberg: Springer Berlin Heidelberg.

Birkhead, M. (2012, January). *Tips from a combat designer: The art of boss design.* Retrieved 2021-09-03, from `https://www.gamedeveloper.com/design/tips-from-a-combat-designer-the-art-of-boss-design` (Section: design)

Björk, S., Lundgren, S., & Holopainen, J. (2003). Game design patterns.

Blizzard Entertainment. (2004). *World of Warcraft.* [PC]. Blizzard Entertainment.

Butler, E., Siu, K., & Zook, A. (2017). Program synthesis as a generative method. In *Proceedings of the 12th international conference on the foundations of digital games.* New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3102071.3102076` doi: 10.1145/3102071.3102076

Bycer, J. (2011, May). *The Evolution of Boss Designs in Video Games.* Retrieved 2021-08-18, from `https://www.gamedeveloper.com/design/the-evolution-of-boss-designs-in-video-games` (Section: design)

Church, D. (1999, July). *Formal Abstract Design Tools.* Retrieved 2021-09-03, from `https://www.gamedeveloper.com/design/formal-abstract-design-tools` (Section: design)

Cox, A., Cairns, P., Shah, P., & Carroll, M. (2012). Not doing but thinking: The role of challenge in the gaming experience. In *Proceedings of the sigchi conference on human factors in computing systems* (p. 79–88). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/2207676.2207689` doi: 10.1145/2207676.2207689

Csikszentmihalyi, M., Abuhamdeh, S., & Nakamura, J. (2014). Flow. In *Flow and the Foundations of Positive Psychology: The Collected Works of Mihaly Csikszentmihalyi* (pp. 227–238). Dordrecht: Springer Netherlands. Retrieved from `https://doi.org/10.1007/978-94-017-9088-8_15` doi: 10.1007/978-94-017-9088-8_15

Dormans, J. (2012). *Engineering emergence: applied theory for game design .* Universiteit van Amsterdam, Amsterdam: Creative Commons.

Fowler, M. (2003). *UML distilled* (3. ed. ed.). Boston [u.a.]: Addison-Wesley.

FromSoftware. (2011). *Dark Souls.* [PS3, Xbox 360]. Bandai Namco Entertainment.

Fullerton, T. (2014). *Game design workshop, 3rd edition* (3rd edition ed.). Boston, MA: A K Peters/CRC Press ; Safari.

Grayson, N. (2021, February). *Why Do We Call The Hardest Video Game Enemies 'Bosses,' Anyway?* Retrieved 2021-09-03, from `https://kotaku.com/why-do-we-call-the-hardest-video-game-enemies-bosses-a-1846301973`

Ham, E. (2015). *Tabletop game design for video game designers* (1st edition ed.). Boston, MA: Routledge ; Safari.

Harel, D. (1987). Statecharts: A visual formalism for complex systems. *Science of computer programming*, *8*(3), 231–274.

Hunicke, R., LeBlanc, M., & Zubek, R. (2004). Mda: A formal approach to game design and game research. In *Proceedings of the aaai workshop on challenges in game ai* (Vol. 4, p. 1722).

Japan Studio, Team Ico. (2005). *Shadow of the Colossus.* [PS2]. Sony Computer Entertainment.

Juul, J. (2002). The open and the closed: Games of emergence and games of progression. In *Cgdc conf.*

Keren, I. (2018). Boss Up: Boss Battle Design Fundamentals and Retrospective. In *Game developers conference 2018.* San Francisco, USA. Retrieved 2021-08-28, from `https://www.gdcvault.com/play/1024921/Boss-Up-Boss-Battle-Design`

Lee, T. (2015, September). *An annotated history of video game boss battles.* Retrieved 2021-09-03, from `https://www.polygon.com/features/2015/9/28/9333685/annotated-history-boss-battles`

Librande, S. (2010). One-Page Designs. In *Game developers conference 2010.* San Francisco, USA. Retrieved 2021-10-03, from `https://www.gdcvault.com/play/1012356/One-Page`

Machinations. (n.d.). *About.* Retrieved 2021-11-13, from `https://machinations.io/about/`

Neil, K. (2015). *Game Design Tools : Can They Improve Game Design Practice?* (Theses, Conservatoire national des arts et metiers - CNAM ; Flinders university of South Australia). Retrieved from `https://tel.archives-ouvertes.fr/tel-01344638`

Object Management Group (OMG). (2017). *Unified Modeling Language (UML) Specification, Version 2.5.1.* OMG File ID formal/17-12-05 (`https://www.omg.org/spec/UML/`).

Schell, J. (2019). *The Art of Game Design : A Book of Lenses, Third Edition.* Milton, UNITED KINGDOM: CRC Press LLC. Retrieved from `http://ebookcentral.proquest.com/lib/munchentech/detail.action?docID=5842983`

Selic, B. (2003, Sep.). The pragmatics of model-driven development. *IEEE Software*, *20*(5), 19-25. doi: 10.1109/MS.2003.1231146

Shore, J. (2004, Sep.). Fail fast [software debugging]. *IEEE Software*, *21*(5), 21-25. doi: 10.1109/MS.2004.1331296

Sicart, M. (2008). Defining game mechanics. *Game Studies*, *8*(2), 1–14.

Siu, K., Butler, E., & Zook, A. (2016). A programming model for boss encounters in 2d action games. In *Twelfth artificial intelligence and interactive digital entertainment conference.*

Square Enix. (2010). *Final Fantasy XIV.* [PC]. Square Enix.

Stout, M. (2010, September). *Boss Battle Design and Structure.* Retrieved 2021-08-09, from `https://www.gamedeveloper.com/design/boss-battle-design-and-structure` (Section: design)

Studio MDHR. (2017). *Cuphead.* [PC]. Studio MDHR.

TPN. (1980). *Phoenix.* [Arcade]. Taito.

Whisenhunt, G., & Wood, R. (1975). *Dnd.* [PLATO].

Wood, A., & Summerville, A. (2019). Understanding boss battles: A case study of cuphead.