

Exploring Product Line Concepts in Game Building

Diego Castro, Claudia Werner

Programa de Engenharia de Sistemas e Computação - COPPE

Universidade Federal do Rio de Janeiro

Rio de Janeiro, Brazil

email: {diegocbcastro, werner}@cos.ufrj.br

Abstract—The gaming industry is one of the most influential in the world and attracts a wide variety of audiences. However, game development can be a time-consuming effort. Therefore, it is necessary to reduce this development time. Software Reuse Techniques have demonstrated their capacity to reduce game development time and costs. One of the main approaches used in reuse is Software Product Line (SPL), which has been successfully adopted by different companies for the purpose of generating software on a large scale. In light of this, the purpose of this paper is to provide an overview of what it would be like to use SPL for support in game development.

Index Terms—Game, Mods, Derivation, Variations, Software Reuse.

I. INTRODUCTION

Games have emerged as a highly prevalent mode of entertainment, attracting fans of diverse preferences and generating substantial revenue in a period of several years [1]. The game development process, despite its huge fan community, has demonstrated an exhaustive and non-systematized process, resulting in prolonged release timelines for certain games [2].

The practice of Software Reuse has been widely employed in the industry as a means to mitigate development time and costs. Various ways have been adopted, ranging from opportunistic reuse to the implementation of a Software Product Line (SPL) procedure [3]. SPL is the focus of this work.

Given the aforementioned discussion, the objective of this study is to present a preliminary concept of utilizing SPL for game building, with the aim of mitigating the time and costs resources required in the existing procedure.

The rest of the paper is structured as follows. In Section 2, a brief contextualization of the problem related to the concept introduced in this study is provided. In Section 3, the suggestion for solving the situation at topic is presented, while Session 4 provides the final conclusions of this study.

II. CONTEXTUALIZATION

As previously mentioned, the gaming industry has experienced significant growth in recent years, accumulating a substantial and dedicated community. Considering the fact that the process of game production remains expensive, characterized by substantial costs and lengthy development periods, certain enthusiasts exhibit impatience towards the release of the games, encouraging them in the creation of their own

versions of those games. The phenomenon of game creation by the community is commonly referred to as modding. It can be conceptualized as a manifestation of players' artistic freedom in remaking and reinterpreting the original game, comparable to the opportunistic reuse process [4]. In this context, the original game serves as a structure upon which a new game is constructed.

The main goal of the SPL method is to create new products by using the “variation points” present in an SPL of the original product [3]. For instance, consider a software X with Y characteristics. These characteristics can be altered, eliminated, or expanded by introducing new features. Consequently, this process results in the creation of new products with Y+1 characteristics, imagining that we were adding a new feature to the software.

However, the process of building mods, like the process of building games, has some issues, with a focus on:

- The lack of specific tools for mods;
- The fact that building a mod can be time-consuming despite being faster than building a game;
- Making the source code available to be modified.

Specifically, the latter issue has already been addressed by major companies, who have made Software Development Kits (SDKs) accessible to users for modifying the original games [5, 6]. This enables users to make alterations to characters, maps, and incorporate minor modifications. Nevertheless, the availability of such services is limited to certain companies, and it should be noted that the scope of modifications that can be made is restricted, limited to certain mechanics and aesthetics of the game.

III. ENGAGESPL

This study proposes the creation of a platform that incorporates SPL concepts in order to address the issues identified in the development of mods and games, as previously examined. The primary objective of the platform is to facilitate the effective development of several versions of a game by streamlining the process and minimizing complications. This helps with large-scale game development.

The primary concept underlying the platform involves organizing the game's characteristics into a hierarchical structure, called feature tree. This arrangement simplifies the process of selecting, modifying, adding, or removing these aspects. The platform was designated as ENgine for GAME GEneration through Software Product Line (EngageSPL).

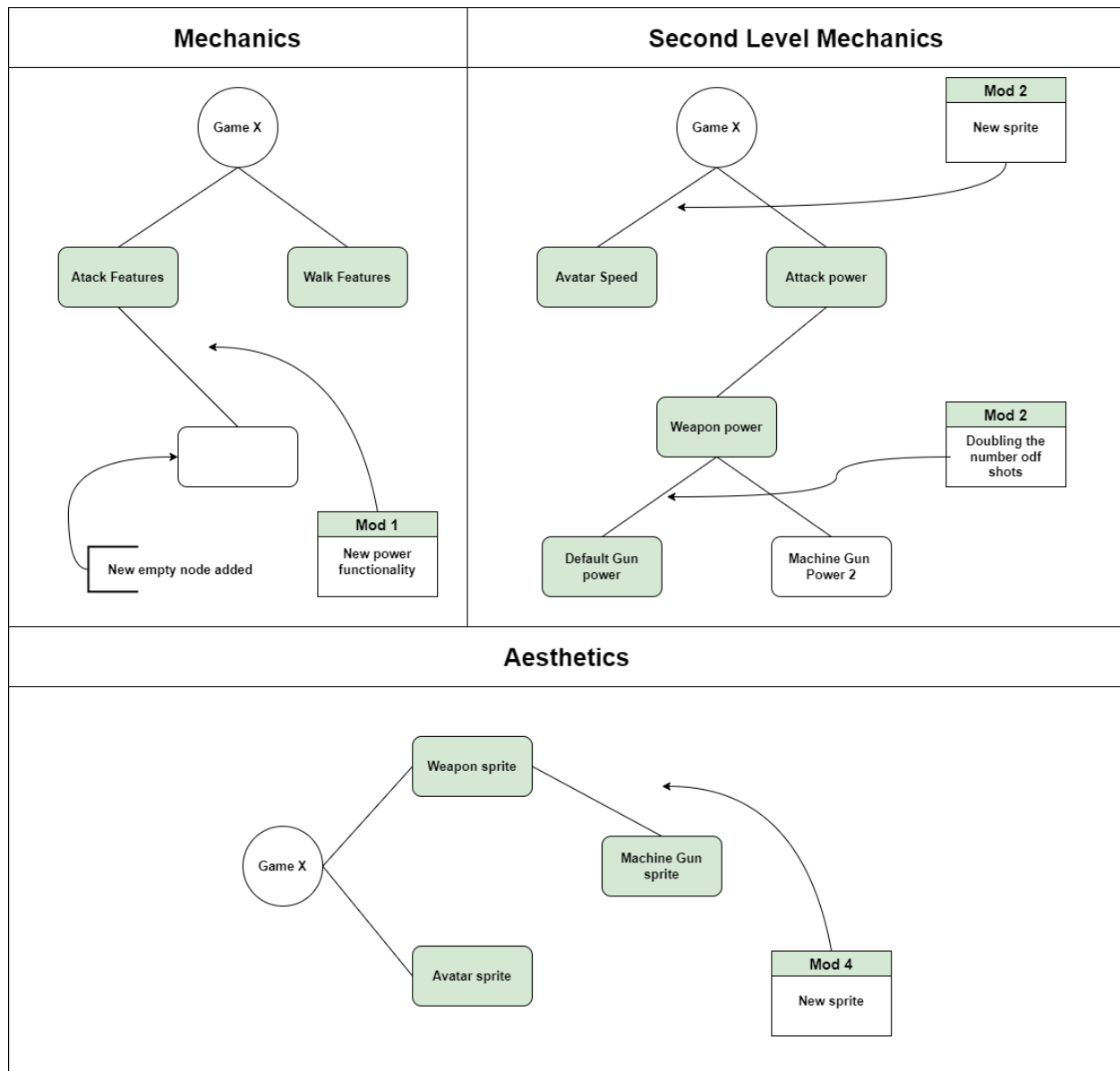


Figure 1. SPL of EngageSPL.

The primary objective of EngageSPL is to offer a comprehensive range of tools that facilitate the expansion of gaming. The platform is designed to enable the creation of a game with the ability to subsequently generate multiple video games with distinct characteristics by leveraging variation points within the functionality tree. Initially, it was thought that the platform could have 3 feature trees based on elements of tTrad [7] framework, each of which will be highlighted below.

The elemental tTrad is a widely recognized paradigm within the game industry. The properties of games are categorized into four distinct groups [8], however, in this work only 2 will be highlighted (Mechanics and Aesthetics).

- **Mechanics:** can be interpreted as the rules and activities that may occur during the course of the game.
- **Second level mechanics:** while the tTrad elements do

not include the second level mechanics, this study will incorporate them to elucidate the mechanics that arise from the combination of primary mechanics. The incorporation of this element will be implemented to augment the level of dynamism inside the proposed adjustments for the advancement of the games.

- **Aesthetics:** Can be defined as the graphical part of the game.

Consequently, there will be three trees: one for the mechanisms of the first level, one for the second, and a third tree for selecting the game’s aesthetics. Initially, the technology and story elements of the tTrad were eliminated from the work. The first is due to the ease of building for multiple platforms using current engines, and the second is due to the difficulty of implementing the game context to dynamically generate the

narrative.

Figure 1 depicts the three trees that correspond to each component of the previously specified elementary tTrad. This figure demonstrates the integration of a new mechanic into the hierarchical structure of the tree, with the alteration of some mechanisms at the second level. Ultimately, there exists the potential to visually represent the alteration of an aesthetic entity. Hence, the user has the ability to choose desired characteristics and determine the manner in which modifications are implemented to the existing features inside the game through the utilization of this tree structure.

IV. CONCLUSION

The gaming industry is seeing substantial growth, with gaming businesses expanding in scale and generating billions of dollars in annual revenue. These companies are consistently publishing a considerable number of titles across many genres, drawing a diverse range of fans. Nevertheless, as commented before, the process of game development can be a time-consuming endeavor that may span several years until its finalization. In contrast, the gaming community is experiencing continuous growth on a daily basis. Given the extensive user base, certain individuals within the community may express concerns or dissatisfaction regarding the prolonged waiting period for the commencement of a game.

REFERENCES

- [1] S. Pashkov, "Video game industry market analysis: Approaches that resulted in industry success and high demand," *Vaasan Ammattikorkeakoulu University of Applied Sciences - VAMK*, p. 40, 2021.
- [2] K. Bilińska, A. Dewalska-Opitek, and M. Hofman-Kohlmeyer, "To mod or not to mod—an empirical study on game modding as customer value co-creation," *Sustainability*, vol. 12, no. 21, p. 9014, 2020.
- [3] C. W. Krueger, "Software reuse," *ACM Computing Surveys (CSUR)*, vol. 24, no. 2, pp. 131–183, 1992.
- [4] D. Abbott, "Modding tabletop games for education," *Proceedings of International Conference on Games and Learning Alliance*, pp. 318–329, 2018.
- [5] S. Agarwal and P. Seetharaman, "Understanding game modding through phases of mod development," *Proceedings of ICEIS*, pp. 114–121, 2015.
- [6] W. Scacchi, "Modding as an open source approach to extending computer game systems," *Proceedings of IFIP International Conference on Open Source Systems*, pp. 62–74, 2011.
- [7] M. de Lemos Uliano and M. L. de Faria, "Benchmark analysis of board games with a narrative focus based on jesse schell's elementary tetrad," *Revista Poliedro*, vol. 4, no. 4, pp. 092–114, 2020. (In Portuguese).
- [8] J. Schell, *The Art of Game Design: A book of lenses*. CRC press, 2008.