UTRECHT UNIVERSITY

Department of Information and Computing Science

Game and Media Technology master thesis

Theory and practice of designing generative AI games: an autoethnographic case study

First examiner:

Dr. Julian Frommel

Second examiner:

Dr. Johannes Pfau

Candidate:

Ali Esat Özbay

In cooperation with:

Yvora Game Studio

September 18, 2024

Abstract

Generative AI (GenAI) technologies are opening up new possibilities in game design, potentially bridging the long-standing gap between tabletop role-playing games (TTRPGs) and digital games. While digital games have excelled in areas such as graphics and complex simulations, they have struggled to match the narrative flexibility and open-endedness of TTRPGs, largely due to the constraints of authorial burden and predetermined content. This thesis addresses the challenge of designing games that combine TTRPG-like narrative freedom with digital game mechanics using generative AI. Through an autoethnographic case study, I show both the practice of designing generative games in a case study at a game studio, as well as the development of a new theory and design framework for generative AI games. This framework, grounded in practical experience, introduces the MAS (Mechanics, Agents, Significs) pillars, the 4F (Function, Fiction, Form, Flow) model of outcomes, and the POV (Possibilities, Operation, Virtual world) model. These findings provide valuable insights into the unique considerations of designing generative games, including technical challenges, player experiences, and the integration of GenAI. By offering a detailed lens into the theory and practice of designing generative AI games, this research lays groundwork for this emerging field of game design.

Contents

Appendix

Part I

Foundations

1. Introduction

Few activities are so associated with humanity as *story-telling* and *play*. Whether you believe humans are story-telling animals [1] or Homo Ludens [2], it is hard to deny that these activities are fundamental parts of our lives.

Simply look at the historic relationship of story-telling with *technology*. Technological innovations have transformed our creative practices. While writing may have been originally invented to count goods [3], millions of people share the same religious beliefs in part due to the printing press. Conversely, our creative pursuits can lead to technological innovation. Cinema was birthed by the invention of the camera, and now cinema drives new innovations in recording and computer graphics technology.

Similarly, digital technologies have transformed the way we play with the advent of *digital games*, leading us to re-imagine what is possible in digital media. Whereas the contents of *recorded media* such books, films, and audio are fixed, digital games let us enter a *virtual world* that responds to our inputs. They excel at interaction, with one important exception.

In 1978, Space Invaders sounded the starting gun for arcade games. Since then, games have driven great technological innovation in computer hardware, graphics, simulation and so on. Though today's games are nearly unrecognizable compared to early games, both are limited in *story interactivity*.

Like Space Invaders, today's digital games have narratives that are predetermined. The player can only exert limited influence on the story, if at all, and only in ways that authors have been able to prepare for. This often puts the player's agency at odds with the narrative: the player is given great freedom in the game's simulation, but is frequently reminded that this does not extend to the story.

This is despite the efforts of academics and authors: the merits of more dynamic narrative do not go unrecognized. However, for the past decades, authors have hit the same fundamental limit, dubbed the authorial burden (Jones, 2022): the author cannot create enough predetermined content to cover the vast possibilities of a narrative that can unfold in any direction, or manage a system complex enough to deal with the combinatorial explosion of possible states that the player may cause.

Researchers have not sat idle in the face of this problem. Much effort has been expended in the pursuit of ever more dynamic narrative. Unfortunately, their research has not yielded any transformative change in the industry. Previous attempts have not adequately addressed the authorial burden: the tools, systems and techniques proposed require significant effort from non-expert authors, and the benefits seem not yet to be worth this cost. Put by Riedl &Bulitko, intelligent systems for interactive narrative "[. . .] are largely denizens of research labs [. . .]", and "[. . .] have required substantive knowledge engineering expertise." [4].

Yet this problem only exists for digital games, not analog games. In 1974, four prior to Space Invader, the original Dungeons and Dragons was published, sounding a different starting gun: the beginning of Tabletop Roleplaying Games (TTRPGs). It too caused us to reconsider what is possible for games, but for story-telling. In TTRPGs, players take on characters in an imagined, virtual world, which is explored through collaborative storytelling and moderated by a rule system. Through the collaborative improvisation of the story, a decades-old game can offer a degree story interaction that is *still* unheard of in digital games. For decades, there has been a gap between TTRPGs and digital games, because computers could not emulate the kind of creativity and intelligence required to make TTRPGs work.

Instead, almost everything in games must be predetermined: the way things sound, what characters say, mission objectives, and so on. This is usually understood as the game's content. What is not predetermined must be simulated. Simulation allows emergent states to arise from the game's systems, such as in the Sims. The game's simulation is sufficiently complex that, combined with the predetermined sounds, visuals and other structures of the game, the player can interpret it as "emergent narrative". *Predetermination* and *simulation* have been the two main paradigms that informed what is possible in digital games. This has recently changed with the introduction of a third paradigm: *generation*.

Generative AI (genAI), such as LLMs, exhibit unprecedented creative and intelligent behaviour accessible to non-experts via natural language prompting. Recent research into games and story generation has focused increasingly on the use of genAI [5][6]. While this technology can help at design-time in existing workflows for content creation, it is at run-time that genAI can create new possibilities.

I coin the term *generative AI games* (genAI games), alternatively generative games, to describe digital games that integrate genAI at run-time. GenAI games could be the breakthrough to bridge the gap between TTRPGs and traditional digital games. Beyond creating more dynamic narrative and obvious applications like NPC chatbots, genAI games could synthesize gameplay distinct from both traditional digital games and TTRPGs.

While genAI may be a solution to the TTRPG gap and and point to unexplored possibilities, numerous questions and issues about genAI games remain. First, there are few existing genAI games that can serve as case studies for both researchers and designers. This limits our understanding of what genAI games can be and how genAI can be used to make such games in practice. Furthermore, our existing design framework were not created with genAI games in mind. Such mental tools help designers approach and analyze game design [7]; after decades of the entrenched predetermination and simulation paradigms, we could use theory to help us imagine new types of games enabled by genAI.

In this autoethnographic study, I take a deep look at genAI games from my own perspective as a TTRPG-player and game developer. I explored these issues in a 12-week case study at Yvora Game Studio, where I aimed to integrate genAI in a series of novel prototypes, culminating in a commercial generative game. These efforts were directed at answering the following question.

> *What are the theoretical and practical considerations of designing games that combine the qualities of TTRPGs and digital games by integrating generative AI to create novel, open-ended narrative and gameplay?*

It is my hope that this study contributes to one more starting gun.

2. Related Work

This section provides an overview of the relevant literature and theoretical foundations for our study on generative games. We begin by establishing key terms and definitions, then explore the computer science perspective on narrative in games. We examine existing narrative techniques in the game industry, discuss relevant design frameworks, and finally, investigate the role of Tabletop Roleplaying Games (TTRPGs) in interactive narrative research. Throughout, we highlight how these areas connect to our research on generative games and the use of generative AI in creating novel, open-ended narrative experiences.

2.1 Theory, Terms, and Definitions

To ground our discussion of generative games, it's crucial to understand the distinction between non-interactive and interactive narrative, as well as the field of Interactive Digital Narrative (IDN).

Non-interactive narrative, often referred to as 'legacy narrative' [8], is the type found in traditional media such as novels. These narratives are authored once and remain fixed. In contrast, interactive narrative allows for user input to influence the story's progression. This distinction is fundamental to our research, as generative games aim to push the boundaries of interactive narrative even further.

Interactive Digital Narrative (IDN) is a field dedicated to exploring new expressive forms of narrative through digital means. Koenitz defines IDN as "a narrative expression in various forms, implemented as a multimodal computational system with optional analog elements and experienced through a participatory process in which interactors have a non-trivial influence on progress, perspective, content, and/or outcome" [8]. This definition aligns closely with my research goals in generative games, where we aim to create systems that allow for unprecedented levels of narrative interaction and emergence.

In the context of IDN, an author creates a system with which an interactor engages, resulting in an 'instantiated narrative'[8]. This concept is particularly relevant to generative games, where the game system, enhanced by generative AI, can produce a vast array of possible narrative instantiations.

It's worth noting that traditional literary theory and narratology, developed for non-interactive works, are often inadequate for describing these new narrative forms. Koenitz calls for the development of specific theory and frameworks for understanding interactive digital works, emphasizing the need for prescriptive rather than descriptive approaches[8]. This call aligns with my research goal of developing a framework for designing generative games that combine elements of TTRPGs and digital games.

2.2 Computer Science Perspective

From a computer science standpoint, interactive narrative has often been framed as a problem to solve. This perspective has led to various approaches, each with its own interpretation of what constitutes narrative and what problems need addressing.

One significant area of research has been the development of drama or experience managers [9], [4]. These systems aim to immerse the interactor in an unfolding narrative where they have agency while adhering to the author's intentions. Experience managers can accommodate or intervene when player actions might cause the plot to deviate from the intended path.

However, a recurring challenge in these systems is the authorial burden. As noted by Riedl & Bulitko, encoding an author's intent formally is not trivial, and the complexity increases for more intricate narratives [4]. The key word is *formally*. Through prompting in natural language, the author's intent can be expressed *informally*, as is the intent in this research. Leveraging generative AI may alleviate this authorial burden and enable more dynamic, open-ended narratives.

Another significant area is story generation, which can be split into formal approaches (associated with manual knowledge engineering and formal systems such as grammars) and open-domain approaches (associated with neural networks trained on large amounts of data). Recent advances in neural networks, particularly large language models (LLMs), have shifted the landscape. While previous systems struggled with generating complex story worlds [10], new technologies like ChatGPT excel at generating rich, detailed story worlds. However, they often struggle with maintaining coherence in longer narratives, likely due to limited context windows. However, we can expect these models to improve over time as major corporations are heavily investing in the technology: where versions of ChatGPT had 4k context window, Google's Gemini Advanced is widely available at the time of this writing and boasts a context window of 2 million.

The emergence of these powerful language models lead naturally to GenAI games. They offer the potential to create dynamic, responsive narrative experiences that were previously impossible due to the limitations of manual authoring and knowledge engineering.

2.3 Hypertext Theory, Storylets, and the Industry

While generative games represent a new frontier, it's important to understand existing techniques for creating interactive narratives in games. Many current approaches can be viewed as clever ways to present pre-authored content in different orders, drawing inspiration from hypertext theory and Choose-Your-Own-Adventure books.

Two key concepts from hypertext theory are particularly relevant: "calligraphic" and "sculptural" hypertext [11]. Calligraphic hypertext involves explicitly adding links between pieces of content, while sculptural hypertext starts with all pieces connected and removes undesirable links. The latter approach, which uses "guards" or preconditions to determine accessible content, is more adaptable and shares similarities with modern interactive narrative techniques in games.

In the game industry, a popular technique known as "storylets" [12] functions similarly to sculptural hypertext. Storylets are snippets of narrative content that are triggered based on the current game state. This approach has been used effectively in games like Wildermyth (2019) and Reigns, allowing for a degree of narrative responsiveness within a pre-authored framework.

While these techniques have proven effective, they still rely heavily on pre-authored content. Generative games aim to go beyond this limitation by using AI to dynamically generate narrative content in response to player actions and the game state. However, clever industry techniques like hypertext theory and storylets may still be valuable to consider. Rather than relying completely on GenAI to generate content, it may be more effective to design clever systems that combine GenAI with pre-written story snippets to balance authorial control with player freedom, and minimize poor quality content generation.

2.4 Design Frameworks

Several game design frameworks have been developed over the years, and understanding these is crucial for situating our work on generative games within the broader context of game design theory.

The Mechanics, Dynamics, and Aesthetics (MDA) framework [13] is widely accepted but has been criticized for excluding some design aspects, particularly narrative [14]. The Goals, Feedback and Interpretation (GFI) framework [15] attempts to address this by introducing a parallel structure modeling the player's intention, perception, and interpretation of the game's narrative side alongside the game's mechanics.

The Design, Dynamics and Experience (DDE) framework [14] revises MDA, presenting Design as everything a designer has full control over, split into Blueprint (conceptual aspects), Mechanics (implementation of rules), and Interface (what players directly perceive).

In the IDN field, the SPP (System, Process, Product) model [8] is particularly relevant to this work. It describes how a System created by the designer enables a Process of interaction, resulting in a Product (an instantiated narrative), and crucially emphasizes how the Systems's *Protostory* is are all the potential instantiations of the narrative. This idea finds purchase with GenAI games: by incorporating GenAI in the system's narrative, surely the *Protostory* changes to a much larger space of potential narratives.

While these frameworks provide valuable insights for traditional digital game design, they may need to be adapted, extended or replaced to to address specific aspects of generative games. The integration of generative AI into the game system introduces new dynamics that aren't fully accounted for in existing frameworks. For instance, how do we model the role of the AI in the design process and prompt engineering? How does the potential for truly emergent narrative affect our understanding of the relationship between mechanics and narrative?

This research aims to build upon or move beyond these existing frameworks, potentially developing new models that can account for the unique characteristics of generative games. This might involve reconceptualizing the relationship between designer, system, and player, or developing new ways to think about narrative emergence in the context of AI-driven systems.

2.5 Tabletop Roleplaying Games in Research

Tabletop Roleplaying Games (TTRPGs) have been a significant source of inspiration for interactive narrative research, and they are particularly relevant to our work on generative games. In TTRPGs, players engage in a shared, improvised narrative structured by the system's rules, often with one player (the Game Master or GM) managing the game world and non-player characters.

Many researchers have looked to GMs as ideal models for interactive narrative systems. For example, experience managers have been compared to GMs [4], and Peinado and Gervás describe GMs as "the best models we have found in real life for designing and directing interactive stories." [16]

Recent research has explored using Large Language Models (LLMs) to emulate or support GMs. Góngora et al. evaluated the capacity of LLMs to perform core GM skills [17], while Kelly et al. and Zhu et al. developed systems to support human GMs using LLMs [18], [19]. Triyason even experimented with using ChatGPT to fully replace the GM in Dungeons & Dragons sessions, with promising results [20].

While these approaches often focus on emulating or replacing the GM, this research takes a different angle. I propose looking to the TTRPG system designer, rather than the GM, as a model for designing generative games. TTRPG systems are designed not only for players but also to support the GM in creating the intended experience. The TTRPG designer should not expect the GM and the players to play the game will play as intended without research. Without playtesting, the designer does not discover how the TTRPG system might fail to guide the players and GMs, and thus cannot improve the system. Similarly, I expect that generative games should be designed to support and guide the GenAI in creating the desired experience, by learning the right guidance after playtesting, developing systemic support and by carefully observing the GenAI's biases, tendencies and failure modes to compensate for them.

This approach acknowledges that, like GMs, current AI systems have limitations and aren't universally capable. Just as a well-designed TTRPG system provides the necessary guidance, examples, and tools for a GM to create a compelling experience, a well-designed generative game should provide appropriate support and constraints for the AI to generate engaging narratives.

Moreover, digital systems can provide far more extensive support for an AI "GM" than traditional TTRPG books can for human GMs. This opens up possibilities for new types of narrative experiences that go beyond what's possible in traditional TTRPGs, combining the open-ended storytelling of TTRPGs with vast databanks and the computational power and rich audio-visual experiences of digital games.

2.6 Generative AI games

While the field of generative AI in games is still in its infancy, several pioneering projects have emerged, demonstrating the potential of this technology to revolutionize game design and player experiences. These early examples provide valuable insights into the possibilities and challenges of integrating generative AI into games, informing my research on generative games.

GameNGen is a world-first demonstration of GenAI not only integrated into a game, but generating a complex game (Doom) directly as a diffusion model. It shows clearly that GenAI can move beyond narrative generation and NPC conversatons, but generate the game's mechanical functioning, visuals, and input handling as well [21]. GameNGen was trained to generate Doom specifically, so it is unknown how generation can be obtained from more general, instructable models, without requiring vast data sets. It still serves as a proof-of-concept that we can go beyond narrative.

In "1001 Nights", in twist on the Arabian fable, sees the player take turns telling the story with GenAI, while the narrated world is visualized using Stable Diffusion. The player has a power that turns keywords uttered by GenAI controlled into weapons, which can be used in the climactic fight that takes place inside the narrated world.[22]

While AI Dungeon has gained popularity as an AI-driven text adventure, its reliance on a traditional chatbot format limits its innovation in game design. However, it has served as an important proof of concept for the integration of large language models in interactive narratives. Of greater interest from a game design perspective is "Infinite Craft." This game features an LLM-powered crafting system that allows players to combine elements, potentially discovering an infinite number of combinations. This approach demonstrates how generative AI can be used to create open-ended, emergent gameplay systems that extend far beyond predetermined content.[23]

2.7 Summary and Research Gaps

The current state of interactive narrative in games is characterized by a tension between the desire for dynamic, responsive storytelling and the limitations imposed by authorial burden and pre-authored content. Existing approaches, from experience managers to storylet systems, have made strides in creating more interactive narratives, but still struggle to achieve the level of narrative freedom and responsiveness found in TTRPGs.

The emerging trend of using generative AI in games and interactive narratives offers a potential solution to these challenges. Large language models and other AI technologies provide unprecedented capabilities for dynamic content generation, potentially alleviating the authorial burden and enabling more open-ended narrative experiences.

However, there is a significant research gap in understanding how to effectively design and implement "generative games" that fully leverage these AI capabilities to combine the narrative freedom of TTRPGs with the rich, systemic interactions of digital games. While there has been research on using AI to emulate or support GMs, there's a lack of comprehensive studies on how to design game systems that support and constrain AI to create cohesive, engaging narrative experiences, as well as lack of GenAI games to serve as case studies.

This research aims to address this gap by exploring the theoretical and practical considerations of designing such games, as well as create such games in the first place. By taking the TTRPG system designer as a model, I explore an approach to GenAI game design that considers not only the mechanics and player experience but also how to effectively support and guide the AI in generating compelling narratives.

This approach has the potential to push the boundaries of interactive storytelling, creating new forms of play that go beyond what is possible in either traditional digital games or TTRPGs. As we move forward with this research, we aim to develop new design frameworks and practical guidelines for creating generative games, contributing to both game design theory and the growing field of AI-assisted creative practices.

3. Method

This study employs an analytic autoethnography approach [24], combining elements of autoethnography with strategies from grounded theory [25] to systematically generate and analyze personal experiences. The research is grounded in the my subjective reflections and observations during a 12-week internship at Yvora Game Studio, where the primary focus was on designing and developing LLM-powered games.

3.1 Research Approach

The research approach is inductive, aiming to formulate new theory based on the data rather than testing pre-existing theories. This aligns with the grounded theory approach, where theory emerges from the data through a process of constant comparison and analysis [26].

3.2 Data Collection

I collected data primarily through free-form theorizing on paper. Throughout the case study and afterwards, I engaged in ongoing reflection, examining prototypes and designs, analyzing design decisions and my motivations for them, and documenting observations and theories.

Initially, I focused on theoretical notes and design ideas to establish a foundation for my practice. As prototype development began, theoretical notes became less frequent, typically arising from new design explorations and play test reflections. The middle stage primarily consisted of play test observations, feedback, and resulting insights. Post-case study, I focused on interrogating theoretical thoughts, reviewing written accounts, and reconstructing relevant memories to connect with the evolving framework.

My experiences encompassed various interactions and activities: discussions with Yvora Game Studio colleagues provided industry insights; silent observations of play tests with prototypes yielded valuable notes; informal conversations with play testers after sessions offered perspectives on LLM-powered games; and design ideation varied in depth and pursuit.

The internship involved a cyclical process of theorizing, designing, prototyping, playtesting, observing, and reflecting, allowing for continuous refinement of ideas and theories as the study progressed.

3.3 Data Analysis

The analysis of the autoethnographic data was conducted through inductive thematic analysis. I employed an iterative process to identify, refine, and connect themes and patterns within the data.

This process began with initial open coding to identify salient concepts and ideas, which I later developed as a glossary, followed by exploration of various theoretical angles to interpret the data. I then systematically examined accounts to discern recurring themes and patterns; I hypothesized relationships between these themes and patterns; and I iteratively refined the theoretical framework through comparison with the data. Multiple models and frameworks were developed and evaluated to ensure they stayed grounded in my autoethnographic accounts. This last point is important: I often discarded themes and models when they strayed away from my documented experience.

A flexible approach was chosen over a formulaic application of grounded theory methods, to prevent imposing a "pet theory" [25]. I developed the analytical method until I arrived at a suitable solution, for example by favoring a glossary of terms over a strict code.

Key to the analysis was the development of this glossary of terms to name the phenomena induced from my experience and notes. This glossary functioned as a coding system, with data points from the collected data either being described by existing terms, leading to refinement of their definitions, or prompting the development of new terms. In some cases, different names were initially developed for the same phenomenon, which were later merged when I realized their similarity.

The glossary terms were organized into themes and relationships. This process involved constant back-and-forth between interpreting the data, reflecting (creating new data and insights), working on the glossary, and forming themes and relationships. This approach amounted to a thematic analysis of the data.

3.4 Ethical Considerations

The Ethics and Privacy Quick Scan of the Utrecht University Research Institute of Information and Computing Sciences classified the research as low risk, with no further review required (see Annex A).

3.5 Qualifications

An autoethnographic study requires the researcher to meet certain criteria, which I discuss in "Personal background and Motivation" of the autoethnographic account in the next chapter.

3.6 Research Goal

The overarching goal of this research was to uncover the theoretical and practical considerations that informed my design of GenAI game design, so that it may serve as both a case study for researchers and designers. Through this methodology, the study aims to provide an early, insider perspective in this emerging practice, and an initial theory that is grounded in the experience of a TTRPG-player and game developer, in a real-world case study. So long as this practice is not yet mature, there is no conventional industry wisdom yet, nor a design consensus reached, nor experienced designers to interview. An autoethnographic methodology may be our best approach towards taking significant first steps towards understanding and establishing a GenAI design practice.

Part II

Results

4. Case Study

The results of this autoethnographic study on designing GenAI games reveal a wide range aspects, ranging from practical to theoretical. Through a series of experimental prototypes and the development of a commercial game, Yorecraft, this study uncovered key themes and concepts for GenAI game design, showcases how to integrate GenAI into games to create novel gameplay and documents early examples of GenAI games to serve as a case study for this emerging design practice.

The following sections present the main findings, organized into four primary categories: defining the gap between TTRPGs and digital games, an account of my experiments, an in-depth view of Yorecraft, and the results of a thematic analysis of the case study.

4.1 Gap between TTRPGs and digital games

> *What are the distinctive qualities of TTRPGs when compared to digital games?*

Before we move on to the results of my internship, let us first try understand the aforementioned "gap" between TTRPGs and digital games through the lens of their differences. To this end, I will answer this section's opening question from my own point of view as master student in game and media technology and TTRPG hobbyist, relying on my personal play experiences, conversations, readings and consulting of TTRPG materials[27],[28] and experiences from the case study that helped me revise or better express these differences.

The goal is not to derive an exhaustive or accurate taxonomy of TTRPG and digital game characteristics. Rather, the goal is to provide both myself and the reader context to help us understand how I believe digital games can be made more like TTRPGs. In doing so, I reveal the subjective underpinnings that inform my design practice.

4.1.1 Implementation and Automation

TTRPGs and digital games differ fundamentally in their implementation and level of automation. TTRPGs operate primarily through human interaction and conversation, with minimal physical components such as rulebooks and dice. This human-centric approach allows for a high degree of flexibility but lacks automated processes. In contrast, digital games rely on software running on specific hardware, enabling a high level of automation but requiring more rigid computation.

4.1.2 Rule Structures and Constraints

The nature rules in TTRPGs and digital games can differ, aligning with simulation vs generation paradigm in the introduction.. TTRPGs employ a combination of "hard" and "soft" rules [29]. Hard rules are unambiguous and can be computed (simulation), while soft rules are open to interpretation and contextual application, amounting to guidelines and subtle influences on human behavior. This mixture allows TTRPGs designers to maintain simplicity in their core simulation mechanics with simple hard rules while accommodating complex scenarios through human interpretation and improvisation guided by soft rules. Digital games, conversely, primarily utilize hard rules due to the constraints of computational systems. These rules are unambiguous and computable, allowing for far more complex rule systems and simulations, but lacking the flexibility of soft rules and improvisation.

4.1.3 Play Space and Possibilities

The concept of "possibility space" differs markedly between TTRPGs and digital games. TTRPGs offer a vast, open-ended possibility space by design, allowing players to attempt actions beyond explicitly stated rules. This openness is a core feature of TTRPG design philosophy. Digital games, while potentially complex, present a more constrained possibility space. Actions are limited to those programmed into the game, either explicitly or as emergent properties of complex systems. This constraint allows designers greater control over the player experience but can limit player agency and open-endedness.

4.1.4 Role of Intelligence

The application of intelligence in game execution differs significantly between the two formats. TTRPGs heavily rely on human intelligence for rule interpretation, scenario generation, and problem-solving. Game masters and players collaboratively fill in gaps in the game world and create detailed narratives. Digital games, while increasingly incorporating advanced AI systems, fundamentally rely on programmed logic. This requires explicit instructions for all possible scenarios, limiting the game's ability to adapt to unforeseen player actions but allowing for consistent and rapid execution of complex rule sets.

4.1.5 Interface and Interaction

The primary interface in TTRPGs is verbal communication, supplemented by physical aids such as character sheets, maps, or miniatures. This allows for rich, nuanced interactions limited primarily by players' imaginations and communication skills. Digital games mediate all interactions through programmed systems and hardware interfaces like screens and controllers. While this allows for immersive audiovisual experiences and complex input systems, it can limit the nuance of player expression compared to the open-ended and social nature of TTRPG interactions.

4.1.6 Flexibility and Improvisation

TTRPGs excel in flexibility and improvisation. Rules and narratives can be adapted in real-time by the game master and players, allowing for dynamic responses to unexpected situations or creative player choices. Digital games, bound by their programming, offer limited flexibility within predefined parameters. While modern games often include complex branching narratives or procedurally generated content, they cannot match the real-time adaptability of a human-moderated TTRPG.

4.1.7 Scale and Speed of Simulation

The simulation capabilities of TTRPGs are limited by human cognition and how fast we can communicate. This often results in simpler mechanical systems focusing on key dramatic moments rather than comprehensive world simulation. Digital games can handle vastly more complex simulations, processing intricate physics, AI behaviors, and game states in real-time. This allows for more detailed and responsive game worlds but can sometimes prioritize simulation fidelity over narrative flexibility.

4.1.8 Creation and Adaptation of Fiction

In TTRPGs, fiction is collaboratively created and adapted during play. The narrative emerges from the interactions between players and the game master, allowing for deeply personalized storytelling experiences. Digital games typically present fiction that is either authored or generated within set parameters. While player choices may influence the narrative, the range of possible stories is ultimately limited by what has been programmed into the game.

4.1.9 Conclusion

The distinctive qualities of TTRPGs – their reliance on human intelligence, flexible rule structures, open-ended play spaces, and collaborative storytelling – set them apart from digital games. Understanding these differences provides valuable insights into how GenAI can bring strengths of TTRPGs to digital games.

Aspect	TTRPG	Digital Game
Operation	Social situation; minimal physical components	Software-based; requires specific hardware
Automation	Manual; player-controlled	High; computer-controlled
Rules	Mix of hard and soft rules; flexible interpretation	Only hard rules; rigid
Complexity	Soft rules easy to understand; players expected to learn many rules	Complex system hard to reason about for designer; easy to play
Play space	Vast, open-ended possibilities; open to player creativity	Constrained by programmed options; closed by formal system
Intelligence	Broad reliance on player intelligence	Relies on automated logic
Interface	Shared imagination and physical aids; verbal interaction	High definition representation; hardware interaction
Flexibility	High; rules and fiction adaptable during play	Limited; constrained by programming
Simulation	Limited by human cognition; focus on key moments	Can handle complex, real-time simulations
Fiction	Improvised in collaboration, emergent storytelling	Pre-determined or procedurally generated within parameters
Control	Designer provides rules and guidance; GM and players in control	Designer has high control via software; player control limited
Host	GM acts as host	Software acts as host

Table 4.1: Distinctive Qualities of TTRPGs Compared to Digital Games

4.2 Experiments in GenAI game designs

In this autoethnographic account, I take you along on my journey as I navigate the exciting and largely unexplored world of designing GenAI games. Through my personal experiences as an intern at Yvora Game Studio, I share my reflections and the evolution of my understanding as I grapple with the unique challenges and possibilities that emerge at the intersection of game design, GenAI, and tabletop role-playing games (TTRPGs).

Analytic autoethnography involves analysing personal experiences to understand broader cultural phenomena. The author must be a full participant in the social world they are studying, critically reflect on their own experiences, make their personal experiences and emotions evident in the narrative, engage in conversations with others, and connect their personal experiences to broader theoretical concepts[24]. My background as decade-long TTRPG player, combined with my training in computer science and game technology, equips me to meet these criteria and offer a unique perspective on this emerging field. I discuss the theoretical foundations that inform my design practice, the practical challenges I encountered during development, and the insights I gained from play testing and iterating on various prototypes.

Furthermore, during my study I employ strategies from grounded theory, allowing theoretical understanding to emerge directly from the data, which in this case, is my subjective experience. Through constant comparison and analysis of my experiences, observations, and reflections, I generate and refine theoretical concepts related to LLM-powered game design. By sharing my personal narrative and the lessons I learned, I aim to contribute to a deeper understanding of the design space for LLM-powered games.

4.2.1 Personal background and Motivation

I've been playing tabletop roleplaying games (TTRPGs) for about a decade now, ever since a friend, let us call them T, introduced me to the hobby. Both T and I have a strong theoretical bent. Long before this study, we'd often theorize about TTRPGs. What makes them work? How can two different systems create such different experiences? What motivates people to play TTRPGs? Is there a TTRPG out there for everyone? We would also design, or at least talk about design. T would pitch me their ideas for their TTRPG designs, and we would have long discussions about design. I read up on the online conversation going on about TTRPG design: articles from TTRPG authors and such. T also encouraged me to try my hand at design, and I did. While TTRPG design was primarily T's passion, it rubbed off on me. My own passion was video game design and technology. Looking back, I can clearly see the impact of those TTRPG discussions, and I believe it's evident in this work as well.

At least, those experiences taught me to analyze TTRPG systems through a design lens. Over the years, I have played numerous TTRPG systems, participated in discussions, read extensively, and deeply contemplated their designs. I have even experimented with TTRPG design myself. On the digital games front, I have studied, designed, prototyped, and developed games as both a hobbyist and a student of computer science and game and media technology. I believe this combined experience with digital games, TTRPGs, and technology equips me to conduct an autoethnographic study on design practices at the intersection of game design theory, LLMs, digital games, and TTRPGs. Since LLMs are still recent, I believe that my own limited experience with LLM-powered game design is shared by others in this field.

4.2.2 Theory-Based Design Approach

For me, there's a close connection between theorizing and designing. A new design almost always sparks a theory - a theory about the game's mechanics and player experience interact, or a hypothesis about how a design tweak will impact the game. This theory-based design approach is the basis for my study. It seems particularly well-suited for TTRPG/LLM-based designs, where designers must hand out more control than in typical digital games. Since we can't control every aspect of the game and playtest them extensively, more of our design decisions are theoretical.

Starting my internship, I was brimming with ideas about what LLMs could bring to digital games. My focus wasn't on how LLMs could streamline existing game development processes, like content authoring. Instead, I was drawn to the untapped potential: what could LLMs enable in digital games that was previously impossible, things TTRPGs could already achieve? This led to a crucial distinction: LLMs used at design time versus runtime. I opted for the term "LLM-powered games" to signify that the LLM has an active role in gameplay.

My focus was on integrating LLMs directly into gameplay, not just as development tools. I wanted to explore beyond the obvious applications, like NPC chatbots, and discover new ways to leverage LLMs in games. I declared my intent to my colleagues, who met me with both enthusiasm and scepticism. When I presented my research proposal at the game studio, a colleague raised this point: aren't games about constraints? How do you constrain the game when LLMs are so open-ended? From the outset, I was intrigued by the contrast between computers and humans. Computers are inherently rigid, executing only precise instructions. They're incompatible with ambiguity, lacking the flexibility of human thought. While we've developed higher-level programming languages, they ultimately map to formal, low-level operations. At their core, computers operate on binary logic.

In contrast, humans and LLMs understand context and natural language. They're adaptable, capable of interpreting ambiguous instructions. They possess a broader context: history, biases, personality, knowledge. A computer program, however, is confined to its programming, understanding data solely through its model. It's "closed," its behaviour entirely determined by its code. This fundamental divide - open vs. closed, inflexible vs. flexible, computational vs. natural, or indeed, "constrained vs. open-ended" - was a recurring theme. This divide connected to critiques of the MDA framework. MDA, with its focus on Mechanics, overemphasizes rules and neglects aspects like style and narrative. MDA treats the game as a computational model. But where does a game's narrative come from? Not from the computational model; it's crafted by a developer. The visual style of objects and characters? That's the work of artists. And so forth.

Furthermore, "narrative" didn't seem to cover the full breath of non-mechanical elements of a game. Breath of the Wild" has minimal narrative but is rich in meaning beyond its mechanics. The experience of climbing a mountain, the beauty of the landscapes, the whimsical charm of Koroks and goblins - these all contribute to the game's significance beyond its rules. Chess is almost entirely mechanical, but still, the design of the pieces carry meaning that we intuitively understand. Their elegance, the heft of them in our hands, the connotations with battlefield tactics and medieval army warfare.

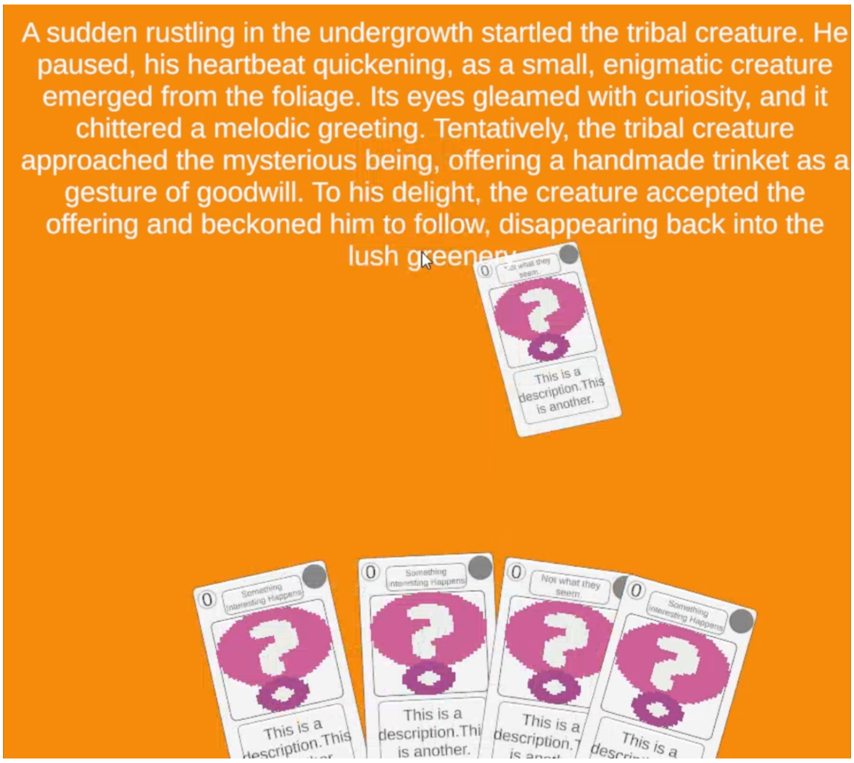
In response to this, I proposed the term "significs" as an umbrella term for the elements, rules and processes that govern the game's meaning or what the game signifies: its signs and our interpretation of them, the game's context, drawing on our own experiences and knowledge. Mechanics, meanwhile, describe the game's functional operation, its abstract model. We can't create an abstract model for meaning-making, as it's inherently subjective. The abstract model of mechanics, however, operates independently of the observer, its impact on the game unambiguous, computable.

An article on hard and soft design in TTRPGs further illuminated this. Hard design, like mechanics, enforces strict adherence to rules. If it's computer-implementable, it's hard design. Soft design guides and inspires players, its influence subtle rather than forceful. Discussing these ideas with a pragmatic colleague prompted me to consider their practical value. Do game developers use frameworks like MDA? Would "significs" resonate, let alone be useful? Was I theorizing in a vacuum, unconcerned with the realities of game development?

4.2.3 Design 1: Roguelike Deck Builder with Narrative Effects

It was time to shift from theory to practice. My initial prototype was a design I dubbed "Roguelike Deck builder with Narrative Effects." The core concept was a deck of cards, each with a mechanical effect and a narrative effect (e.g., "Gain +5 intrigue; A mysterious stranger leads the way"). Playing a card would impact both the game's mechanics (increasing an intrigue meter) and the narrative, with an LLM generating text based on the card's narrative effect and the current state of the story. For instance, a high "danger" meter would prompt the LLM to generate a more perilous narrative.

The primary goal was to demonstrate an LLM's ability to dynamically generate narratives that adapt to any card sequence. The sheer number of possible card combinations makes pre-authored narratives infeasible, but an LLM can generate them in real-time. This allows designers to create vast narrative possibilities beyond manual authoring. A secondary goal was to



A sudden rustling in the undergrowth startled the tribal creature. He paused, his heartbeat quickening, as a small, enigmatic creature emerged from the foliage. Its eyes gleamed with curiosity, and it chittered a melodic greeting. Tentatively, the tribal creature approached the mysterious being, offering a handmade trinket as a gesture of goodwill. To his delight, the creature accepted the offering and beckoned him to follow, disappearing back into the lush greenery.

Figure 4.1: Player can direct the story by playing different cards

intertwine narrative and mechanics. Players could shape the story through both mechanical choices (affecting meters) and by selecting cards with specific narrative effects.

The broader vision included expanding the card pool, introducing events that add cards to the player's deck, and creating cards that trigger story endings based on specific narrative conditions. This would challenge players to strategically manage their character and narrative choices to reach desired conclusions.

The initial prototype focused solely on narrative effects and a basic card deck. I let my daily supervisor play test the prototype shown in 4.2. I thought the mechanics were underdeveloped, but I was surprised they found the narrative control still exciting. I attribute this to the novelty of generated narrative; I think I had gotten so used to it that I couldn't realize it might impress someone else. But still, there was no way to win the game, and the point of the case study was to explore. There was no turn structure or economic strategy, typical elements of deck builders, so I started adding those mechanics.

To rapidly iterate on mechanics, I shifted to paper prototyping. I replaced story meters with player character meters (spirit, health, coins), introduced a wider range of mechanical card effects, and created distinct player and event card decks. Player cards represented character actions, while event cards were environmental encounters. A "danger" meter determined the number of event cards drawn each turn, adding a risk-reward dynamic to the game. The goals were to survive an increasing number of rounds and accumulate coins to upgrade the player's deck between adventures. Coins could purchase new player cards, and progressing to higher levels would introduce more challenging event cards.

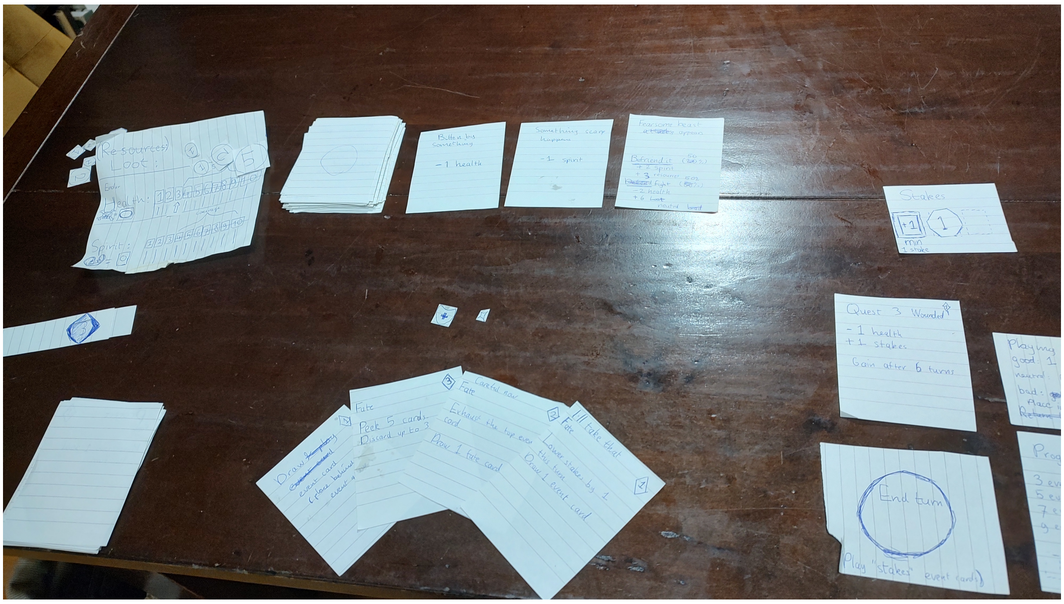


Figure 4.2: Paper prototyping of generative card game led to an overemphasis on mechanics

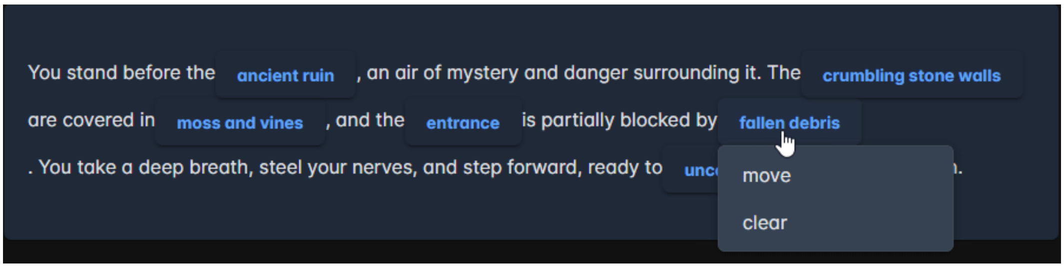
My collegue at Yvora Game Studio and I playtested the game, which revealed that mechanics overshadowed the narrative, reducing it to mere "flavour." We initially pretended to be the LLM, by improvising the story ourselves. However, we quickly abandoned this because we didn't *need* to. The game was not dependent on the LLM's output. The narrative reflected player choices but didn't influence them, lacking integration with gameplay. I came up with narrative integration: it means the narrative is an integral part of playing the game, consequential, shaping player decisions.

This is connected to an interesting point: how do you replace the LLM in a paper prototype? Although I tried to be a human stand-in as I mostly developed the paper prototype alone, this lacked the dynamic of expectation and surprise. This raises the question whether paper prototyping are useful for GenAI games. Rather, I think a better approach would be *social paper* prototypes. Rather than iterating alone, I would appoint one person the title "LLM", who would make sure to be an integral part of the developing paper prototype. Unfortunately, it is not as accessible as paper prototyping alone. As a result, the LLM's role became superficial in my paper prototype.

In an attempt to make the LLM more consequential, I experimented further with cards, like having the LLM generate them dynamically, but ultimately I decided to move on from cards.

4.2.4 Design 2: Generative Text Adventure

Returning to the drawing board, I next prototyped an LLM-generated text adventure, or what I call a "generative text adventure game". The player would provide a starting seed, and the LLM would generate the opening paragraph. My aim was to design a simple mechanic with strong narrative integration.



You stand before the ancient ruin , an air of mystery and danger surrounding it. The crumbling stone walls
are covered in moss and vines , and the entrance is partially blocked by fallen debris
. You take a deep breath, steel your nerves, and step forward, ready to unc

Figure 4.3: Generative text adventure prototype integrates an action menu with generated narrative

I tasked the LLM with identifying interactable elements within the narrative. Initially, I explored NLP tools for this, but they lacked the contextual understanding and ease of use offered by LLMs. With the LLM, I could simply describe my desired outcome. Through prompt engineering, I guided the LLM to format paragraphs using XML, tagging interactable elements and suggesting possible character actions. Ensuring valid XML output was crucial, as I hadd built a web interface to display the generated text, converting interactable elements into clickable buttons with dropdown action menus (see Figure 4.3).

This design featured an organic interaction between the LLM's output and the mechanics. Players clicked on elements of interest and chose actions based on their curiosity and goals, with constraints emerging naturally from the narrative context. The LLM then generated the next paragraph, reflecting the action's outcome and introducing new possibilities, creating an ongoing loop.

What struck me about this design was the process of integrating the LLM's narrative with the game system. The game system, the clickable buttons, dropdown lists and calls to the LLM, processing of the XML, displaying the next paragraph and disabling previous buttons, was attuned to the expected output of the LLM. Conversely, the LLM must be carefully prompted to format its data so that it the game system can use it. This crosstalk was missing from the roguelike deck builder.

This design showed that integrating LLM-generated narrative with the game system should be a priority. The interface elements - buttons, dropdowns, XML processing - were all tailored to the LLM's output format. Conversely, the LLM prompts were designed to produce compatible data. This kind of cros stalk was missing from the roguelike deckbuilder.

I explored additional ideas, such as an interactive avatar with clickable inventory items and character-specific actions, which required further refinement of the LLM prompt. However, the expanding narrative context of these hypertext adventures quickly pushed the limits of the LLM API. This cost consideration might be overcome by running the model locally, which I did for a while. Unfortunately, at that time, the local model did not perform well enough in my experiments. However, better, open-source models like LLama-3 8B have come out in the meantime, so local model may become more viable for cost-effective generative text adventure in the future.

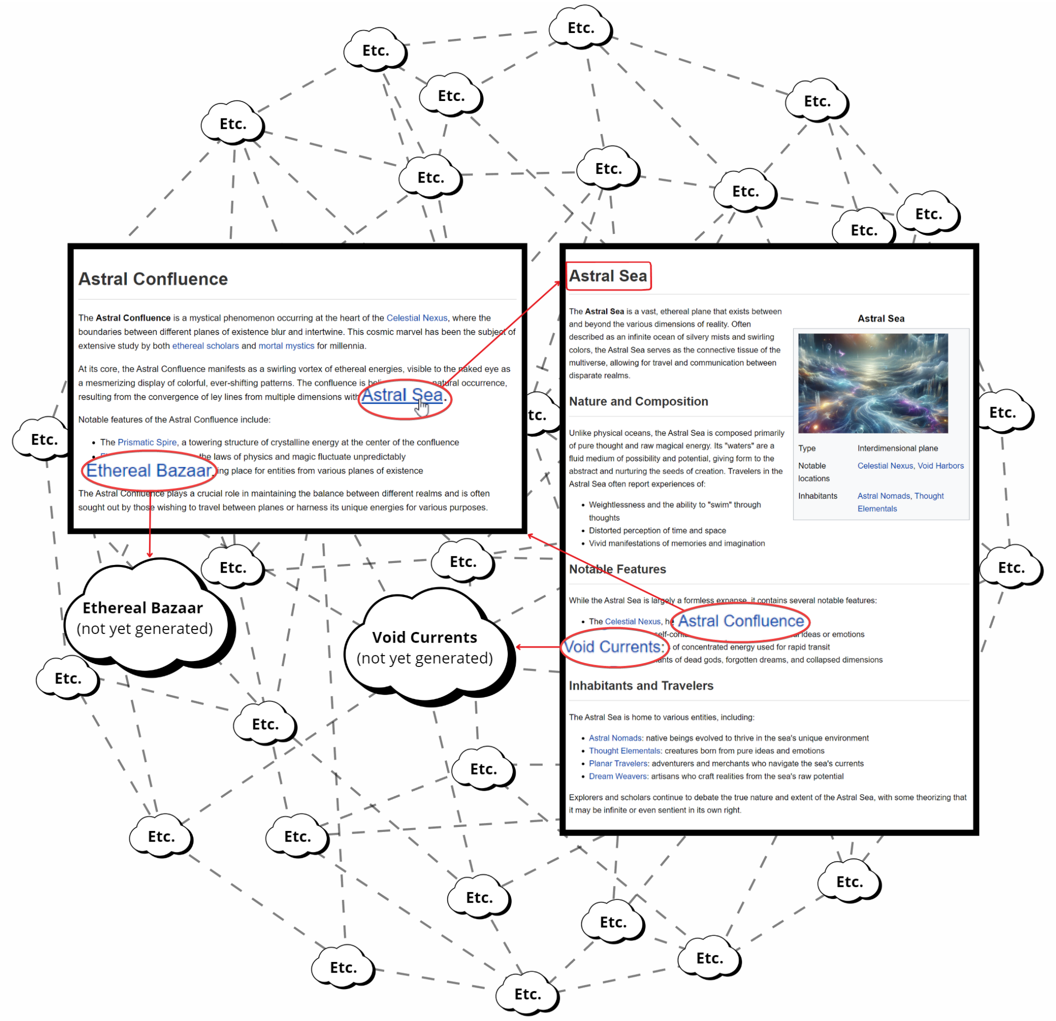
When a collegue played the prototype, I saw them form goals in response to the narrative, even aiming for fictional locations introduced by the LLM. This reminded me how players react to the open-endedness of TTRPGs, yet it was integrated into a simple button clicking mechanic. This emergent interaction was exciting, but LLM limitations were also apparent: repetitive or shallow generation, occasional formatting errors breaking the system, and inconsistencies over longer playthroughs. In the end the design proved unsustainable for long-form play due to API costs.

However, this experiment led me to identify four types of narrative integration: flavor (non-consequential), narrative context integration (options depend on narrative), narrative goal setting (narrative guides player goals), and player-directed narrative (player shapes the narrative). I suspect narrative integration is a spectrum, and that it applies not only to narrative.

4.2.5 Design 3: Generative Hypertext, caching and worlds

I made one last design in this text adventure theme. I never made a prototype: it is only a design on paper, briefly tested to guage how well GenAI performed. It demonstrated to me an important concept: different types of narrative possibility spaces. The design is inspired by online wikis like Wikipedia and the Wikipedia game. Figure 4.4 provides an overview. The pages were generated with Claude Sonnet 3.5, the prompt for the image was prompted in the same conversation with Claude, and the image prompt was given to Bing Image Creator, showing LLM support is there. This is unsurprising, as LLM hallucination is a feature in this case. The game consists of titled pages containing hyperlinks to other pages. The first page describes perhaps your starting location, contains hyperlinked points of interest for the player. If the player navigates to a non-existent page, we give the LLM the context of the current page and perhaps of other existing pages which also link to the non-existent page. We then task it to generate the page, complete with new links to existing or non-existent pages.

By caching the pages and sharing them in all playthroughs, players can navigate back and forth across known and unknown pages. The possibility space becomes unknown but theoretically fixed and shared, a single space all players explore. LLMs enable designers to create these spaces, which



Astral Confluence

The **Astral Confluence** is a mystical phenomenon occurring at the heart of the Celestial Nexus, where the boundaries between different planes of existence blur and intertwine. This cosmic marvel has been the subject of extensive study by both ethereal scholars and mortal mystics for millennia.

At its core, the Astral Confluence manifests as a swirling vortex of ethereal energies, visible to the naked eye as a mesmerizing display of colorful, ever-shifting patterns. The confluence is beli natural occurrence, resulting from the convergence of ley lines from multiple dimensions with Astral Sea.

Notable features of the Astral Confluence include:

- The Prismatic Spire, a towering structure of crystalline energy at the center of the confluence
- the laws of physics and magic fluctuate unpredictably
- Ethereal Bazaar, ing place for entities from various planes of existence

The Astral Confluence plays a crucial role in maintaining the balance between different realms and is often sought out by those wishing to travel between planes or harness its unique energies for various purposes.

Astral Sea

The **Astral Sea** is a vast, ethereal plane that exists between and beyond the various dimensions of reality. Often described as an infinite ocean of silvery mists and swirling colors, the Astral Sea serves as the connective tissue of the multiverse, allowing for travel and communication between disparate realms.

Astral Sea

Type	Interdimensional plane
Notable locations	Celestial Nexus, Void Harbors
Inhabitants	Astral Nomads, Thought Elementals

Nature and Composition

Unlike physical oceans, the Astral Sea is composed primarily of pure thought and raw magical energy. Its "waters" are a fluid medium of possibility and potential, giving form to the abstract and nurturing the seeds of creation. Travelers in the Astral Sea often report experiences of:

- Weightlessness and the ability to "swim" through thoughts
- Distorted perception of time and space
- Vivid manifestations of memories and imagination

Notable Features

While the Astral Sea is largely a formless expanse, it contains several notable features:

- The Celestial Nexus, h Astral Confluence
- self-co ideas or emotions
- Void Currents: of concentrated energy used for rapid transit
- nts of dead gods, forgotten dreams, and collapsed dimensions

Inhabitants and Travelers

The Astral Sea is home to various entities, including:

- Astral Nomads: native beings evolved to thrive in the sea's unique environment
- Thought Elementals: creatures born from pure ideas and emotions
- Planar Travelers: adventurers and merchants who navigate the sea's currents
- Dream Weavers: artisans who craft realities from the sea's raw potential

Explorers and scholars continue to debate the true nature and extent of the Astral Sea, with some theorizing that it may be infinite or even sentient in its own right.

Figure 4.4: A network of hyperlinked pages previously generated or generated on demand.

I categorized into three types based on world generation. The first is authored space, or known world: the narrative space is known at design time, as traditional digital game spaces are. Shared generative space: the narrative space is unknown to the designer as it will be generated at play. From the point of view of the players, it is as if there is a single, fixed world: I called this possible world Only, sections of the space or not generated until someone visits it, subsequently fixing it for all players. This is like in the generative wiki game. Finally, there is uncached generative space, or what I called possible worlds. The indeterministic nature of LLM leads to different worlds between playthroughs. The same paths may lead to entirely different outcomes.

Authored spaces gives designers full control but requires them to author everything. Possible world requires designers to constrain a single space, but must give up control to GenAI. Furthermore, designers can post-author generated content, letting players explore new content on demand, but lets designers revise it later if needed. Uncached generative space tasks designers to constrain a huge, high-dimensional space, but has the greatest potential to lead to unique, personal playthroughs that never repeat.

I came up with two related terms, which describe a related idea, but on a spectrum. Divergence describes the tendency of playthroughs to diverge from similar conditions to dissimilar conditions, while convergence describes the tendency of playthroughs to converge from dissimilar conditions to similar conditions. For example, in BOTW, players start in identical conditions, but go on to have entirely different playthroughs with different routes and experiences (divergence). However, in the end it is the fate of every player to confront the final boss (convergence). We can apply this to the narrative context of the game and the generation of an LLM. We can task it to tie narratives to common points that we pre-authored or provided material for, or let it improvise, taking similar conditions to entirely dissimilar conditions. We can imagine a sort of back and forth between diverging and converging, letting the designer balance control between designed constraints and open-endedness. We observe this in TTRPG as well. GMs can offer players a lot of freedom for exploration, leading to entirely unknown improvised conditions, but simultaneously lead them back with narrative hooks to prepared content or their “prep”.

4.2.6 Design 4: Floating Island meets Infinite Craft

After exploring generative hypertext adventures, I encountered "Infinite Craft."[23] It’s a game where you start with four basic elements and combine them to create new ones. The game uses a generative text model to produce these combinations, storing them in a database. The concept is simple but effective due to its seemingly endless possibilities. You can discover anything from "train", "unicorn" or even things like "were-lion.”

Inspired by "Infinite Craft’s" use of LLMs for content generation within mechanics, I prototyped my own version. To address the lack of explicit goals in the original, I framed the game as caring for villagers on a floating island. The player, acting as a god-like figure, must balance four meters (food, population, nature, happiness) to ensure the villagers’ survival and progress.

Starting with the four basic elements, the player crafts new ones and introduces them to the island, aiming to achieve a high score before the meters become unbalanced and the villagers perish. Introducing an element triggers an LLM-generated event describing its impact. I added variety by randomly assigning different writing styles to the LLM, such as haikus, scientific reports, or villager interviews. The LLM also determined the mechanical effects of each element (e.g. -2 nature, +1 population), ensuring these were hinted at and based on the event text. This encouraged players to learn and anticipate the consequences of their choices.

A large playtest conducted at a networking event yielded valuable insights. Despite my initial reservations, playtesters responded positively. I was surprised at how long some playtesters kept playing. I especially recall the playthrough of a games researcher who visited the event, who asked me about how I came up with the combinations, since the progression seemed so unusual for the genre (I believe he meant poorly designed). When I told them it was done by an LLM in real-time, the game seemed to gain another dimension for the researcher. It became an experiment intended to figure out the AI's logic, of gaming the AI. I call this effect AI awareness as part of Player-AI dynamics. In fact, I believe the researcher noticed a lack of procedural clarity. I saw this dynamic in other play testers as well, as they tested the AI with seemingly unrelated elements and received surprisingly fitting or baffling answers.

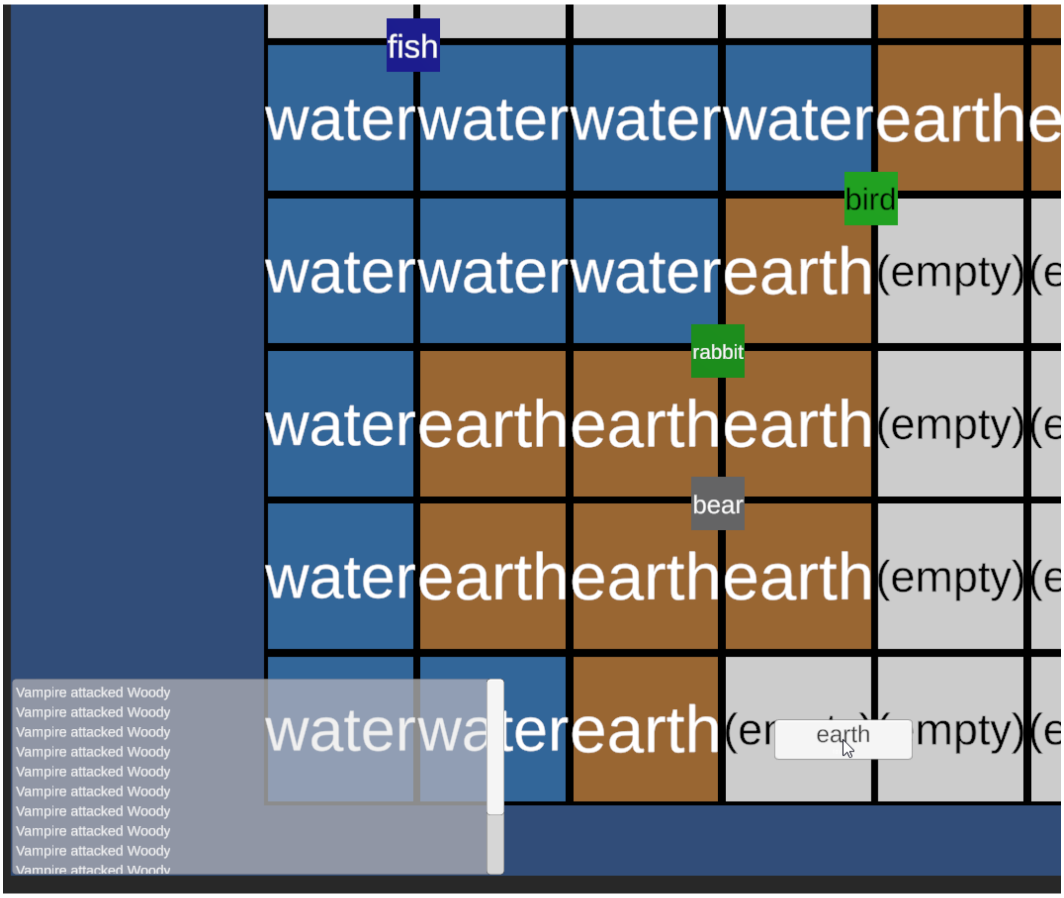
This connects to the concept of procedural clarity in hard design. Traditional game mechanics allow players to internalize the system: they learn how things work and can strategize based on predictable outcomes. LLM-based systems sacrifice this clarity, introducing unpredictability. However, they compensate by offering surprise and a sense of an external, almost magical, influence on the game world.

4.2.7 Design 5: LLM-powered real-time simulation

One more experiment beckoned. I felt something was missing from the previous designs: they were all reactive, waiting for player input. I wanted to explore combining LLM-powered crafting with real-time simulation, a core strength of digital games.

I was Inspired by games like Dwarf Fortress, which let player imagine rich narratives in its complex simulation represented with simple ASCII visuals and a textual event log. In this game, everything is represented in text: terrain, buildings, creatures, and other entities, and everything moves and interacts according to a rich, LLM-powered simulation (Figure 4.5). A cat looks like the word “Cat” wiggling around on a world map populated with entities like “tree”, “house”, “campfire” and “villager'. When the 'villager' bumps into the 'tree', the 'tree' might turn into 'wood'. Interactions show up in an event log, like “Villager chops down tree.” Players would have a crafting window with the familiar crafting mechanic, and could drag elements into the game world, turning them into simulated terrain or entities.

Interactions would be generated by an LLM: it would be prompted to create interactions in a subject-verb-object format, and to list entities to be removed (tree) and added (wood). If “wood” was spawned in the world (due



Vampire attacked Woody
Vampire attacked Woody
Vampire attacked Woody
Vampire attacked Woody
Vampire attacked Woody
Vampire attacked Woody
Vampire attacked Woody
Vampire attacked Woody
Vampire attacked Woody
Vampire attacked Woody

Figure 4.5: Prototype of LLM-powered real-time simulation. Tiles can be crafted by combining elements, entities move on their own and interactions are generated by an LLM

to an interaction or because the player crafted it) but was not yet defined in the simulation, the LLM would generate it. The LLM would be prompted to fill out a simulation-compatible data model, based on attributes linked to text descriptions. The LLM would determine the type and attributes of the element (terrain or entity), such as if it can move, its movement pattern, speed, size, whether it was a living creature, etc.

In Unity prototyping, the challenge was to simulate these attributes. Terrain was straightforward, but entity movement and interactions were complex. I opted for grid-based movement for simplicity and created a game loop to handle entity actions and let the game loop call arbitrary components linked to attributes. If an entity had the movement attribute, the game loop would call its movement component.

The vision was a self-sustaining world where entities interact and events unfold, with the player influencing it by crafting new elements and dragging them in. A thriving village could be brought to ruin by crafting “zombie” and dragging it in the world. The event log would read “zombie bites villager”, “villager turns into zombie”, starting the apocalypse.

Unfortunately, due to time constraints, it remains a basic prototype, and a playtest was never conducted.

4.2.8 Bonus experiment and rule coverage

While wrapping up my internship, I thought about next steps. Alongside a personal project, I had more time for TTRPGs, leading to insightful conversations. One person shared their theory on categorizing games based on their analytical versus social focus and their emphasis on story versus rules. Another suggested that the TTRPG system was secondary to the setting.

When I flipped through Ironsworn rulebook’s, I came across a diagram which defined the game’s flow explicitly. I saw parallels with computer program control flow. This sparked an idea: could I implement the "hard design" (mechanics) of a TTRPG, leaving the "soft" elements (narrative, interpretation) to an LLM? This raised questions about rule coverage - how much of the game could be handled by explicit rules versus LLM-driven interpretation? If too little of the game, especially the control flow, is covered by explicit rules, then an LLM would need to cover much larger parts of the game which they are most likely not ready for. Ironsworn, with its clear move structure and explicitly covered control flow, seemed well-suited, as did Apocalypse World, where the GM follows specific rules and chooses “GM moves” when engaging in a narrative conversation with players.

I designed a simple interface where the player describes their actions, and the system, using an LLM, determines if a move is triggered. If so, the system handles the mechanical aspects, while the LLM fills in the narrative gaps. The final output reflects the move’s outcome within the narrative. I began prototyping, designing move detection prompts. Since the LLM API costs were a limiting factor for the generative text adventures, I ran Llama 3 8B locally, which had just come out. The results were promising, with the LLM accurately identifying triggered moves. A full implementation could offer a TTRPG-like experience in a video game, something that I had secretly wanted. However, with my internship having ended already and a thesis to write, the experiment was over.

4.3 Commercial game

I was tasked with developing a commercial game, so I had to change focus from experimentation to a single feasible design. We discussed practical constraints for the final game. We could deploy on the web or build a mobile/PC app using a game engine. Local LLMs were off the table due to hardware requirements, so we needed to use a cloud-based LLM with an API. Hardcoding the API key was insecure, which meant we needed a relay server and client-server architecture. Since we needed to host a server in any case, I proposed to deploy the game on the web like Infinite Craft. This meant rapidly learning web technologies as my internship deadline loomed. The outcome of these efforts was Yorecraft.

Me cold. I want warm! Help me be warm!

Figure 4.6: Overview of Yorecraft

4.3.1 Gameplay overview

Yorecraft revolves around "elements," represented as nouns, each tied to a specific age and accompanied by an icon. The game features a progression system through eight distinct ages, from primal to future, with each age unlocking more advanced elements. The central mechanic is the familiar LLM-powered crafting: players combine two elements to discover a new one, potentially locked if they have not reached the corresponding age. While Infinite Craft used emojis, Yorecraft dynamically generates and caches unique icons for each new element using a generative image model.

Progression is driven by "requests" from the Eylander, an animated NPC who poses pre-authored challenges (e.g., "I'm freezing! Can you help me warm up?"). Players offer elements, and the Eylander responds via LLM, indicating success or failure. Success leads to the next request; failure prompts the player to try again, with potential hints from the Eylander, see Figure 4.9. The LLM-driven evaluation allows for surprising solutions. Freezing? How about a blanket, sauna or volcano? Each request is unique and age-specific. Fulfilling all requests within an age unlocks the next age's elements and requests, driving progression, see Figure 4.8.

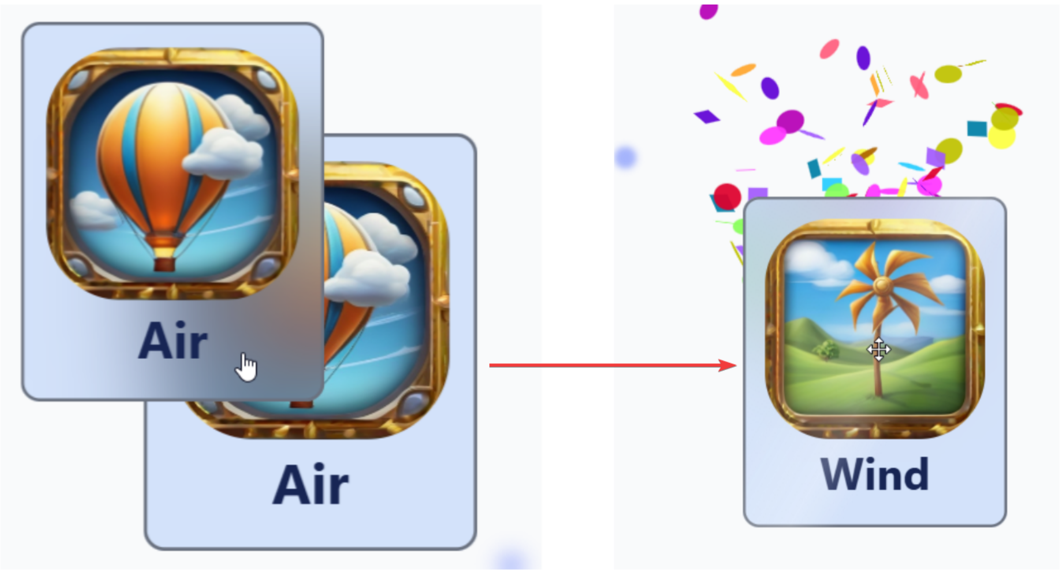


Figure 4.7: Combining 'air' and 'air' creates 'wind'

4.3.2 Architecture and implementation

The technical implementation of Yorecraft presented an interesting challenge. I modeled the elements, combinations, and requests for the database, and created validation logic to prevent invalid data from entering the database or being sent back to the client. I needed structured output from the LLM to comply with this model. Through prompt engineering and custom code, I processed LLM text into JSON, relying on the validation logic to catch errors. In case of invalid output, the LLM could be tried again due to its indeterministic nature.

The backend API was built with Python, FastAPI, and SQLAlchemy, handling client requests, generating and caching LLM responses, and icon storage and retrieval. The image model ran on a Comfy UI server providing generated icons on request. The language model was Claude Haiku, since it was cost-effective and very responsive. The image model used was SDXL Turbo, again to minimize generation times.

The client, implemented in Svelte and TypeScript, featured a searchable element list, an animated NPC, an interactive canvas for combining elements, and UI elements for progress tracking. Player data was saved locally using cookies.

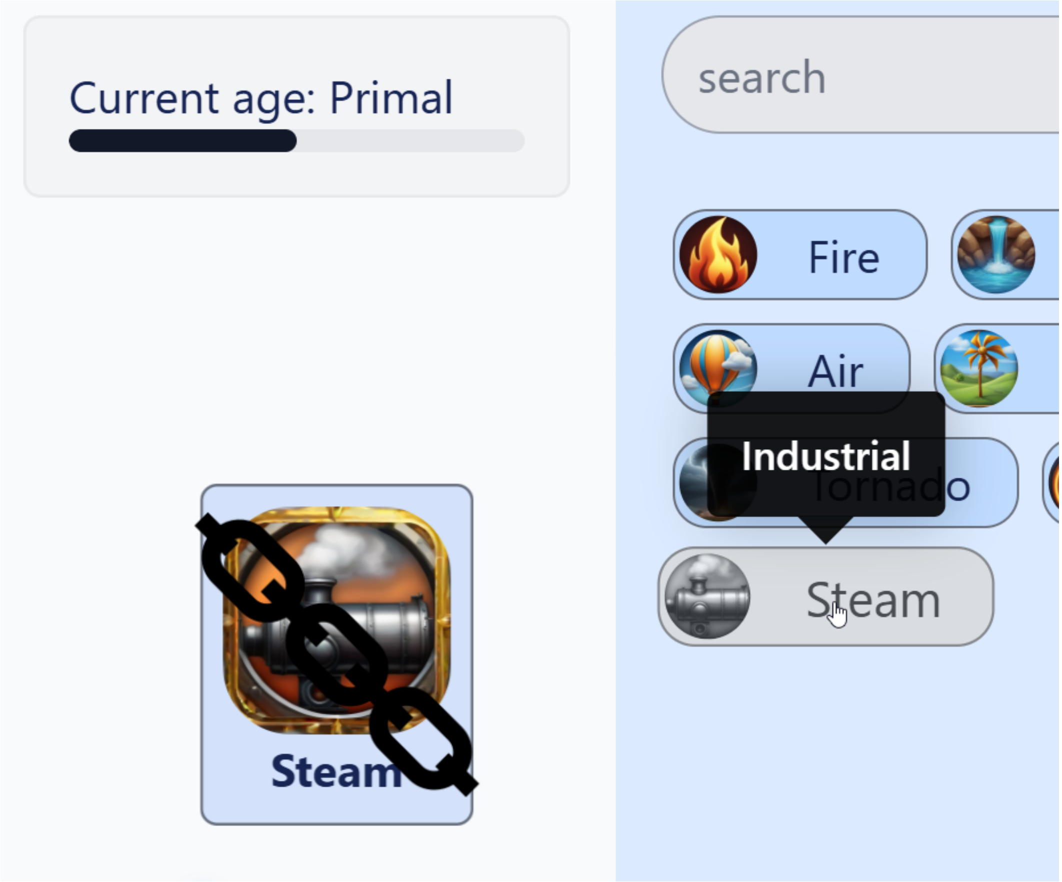


Figure 4.8: Elements may be locked behind ages that the player has not reached yet.

Us want to sail to island. We need place to sail our raft!

Fire? Eylanders no can sail on fire! Silly god, fire burn our raft and us! We need something to float on, not something to burn!

Us want to sail to island. We need place to sail our raft!

Ah, yes! Ocean be perfect for sailing our raft! We shall set sail at once and explore the vast blue waters. Many thanks, oh mighty god!

Figure 4.9: Generative AI decides what are accepted solutions and which are not; generates NPC response as well.

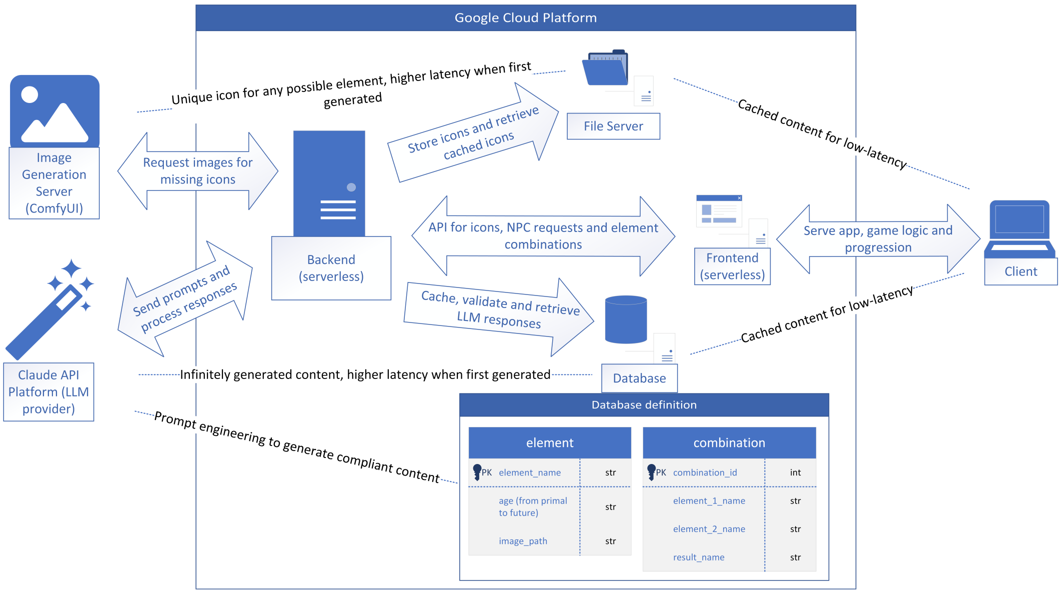


Figure 4.10: Architecture for Yorecraft. Arrows show how components communicate, while dotted lines show indirect relationships.

Both client and backend were deployed on Google Cloud Platform (GCP) using Docker containers, with CI/CD pipelines set up on GitHub for automatic deployment of updates. Icon hosting and the database were also implemented with GCP services.

4.3.3 Playtest

Playtesting highlighted areas for improvement. First, the LLM occasionally rejected seemingly suitable elements or gave non-sensical responses. Although we used Claude's smallest model for cost and speed, a larger model could enhance responses without significantly impacting performance or cost, thanks to caching. Secondly, Eylander's requests, compared to Infinite Craft's open-endedness, could lead to tunnel vision and a disappointing cycle of getting unhelpful elements. This becomes more pronounced the longer players were stuck on the same request. In Infinite Craft, combinations are driven by player curiosity, making the process inherently enjoyable.

A major challenge was preventing the LLM from falling into repetitive patterns, generating similar elements, and trapping players in loops. This was especially detrimental to the experience when players were stuck on the same request. To address this, I collaborated with colleagues and devised a solution. We manually seeded the initial database with various elements that were helpful for the requests. This ensured early-game variety, more reliable paths to helpful elements, and provided the LLM with fresh input, reducing the risk of repetitive loops.

4.3.4 Generative AI and game architecture

From a technical perspective, LLM-powered games present unique considerations and trade-offs. To optimize cost, accessibility, and performance, we deviated from the traditional model of running games entirely on the client. Instead, we adopted a client-server architecture, typically associated with multiplayer games, even for this single-player experience. This was necessary to enable caching of LLM responses, improving performance over time as common combinations are discovered. To minimize hardware requirements, we used a cloud provider for LLM generation.

This shift in architecture, from client-only to client-and-cloud, points to a practical distinction between LLM-powered games and traditional ones. Save for the smallest models, LLM generation is too slow on client hardware. I have seen at least three techniques: caching, offloading, or masking. Caching lets us reuse expensive generations and create a persistent world while minimizing client storage. Offloading lets us run intensive operations on more capable cloud hardware, preventing client slowdowns. Masking lets us hide intensive operations, by continuing the game loop while intensive operations run in the background. When their results are required, it seems like they were available immediately. For example, the system could generate narrative branches in advance, or generate new locations before the player visits them.

4.3.5 Design Reflections

The commercial game had different requirements than the experimental prototypes. Most importantly, it had to be feasible to develop to completion within a small time frame. In my design explorations, I found myself oscillating between wanting to find something entirely new and exciting, and finding something that was simply fun and commercially viable. I occasionally felt that I caught glimpses of the future for GenAI games. However, our understanding of GenAI is still very limited, and with so few examples to learn from, I lacked the knowledge for making a good game, as I had simply not yet even seen a GenAI game before.

An important turning point was playing Infinite Craft. It showed me that, while it is possible that GenAI will start something truly groundbreaking in games, it makes commercial sense to offer players polished, simple GenAI game mechanics that come with the novelty of an "infinite" or "improvised" label. Neither Infinite Craft or Yorecraft would be impossible experiences without GenAI. This novelty can be easily underappreciated as someone who interacts with GenAI daily, looking for something new and exciting.

I realized that although the first arcade games had a profound impact on gaming history, in retrospect, they seem very simple to us. I expect a similar evolution in GenAI games. The first groundbreaking GenAI games will likely be small in retrospect, like Pong, Space Invaders and Tetris, but will spark the imaginations of millions, marking the start of a fevered exploration of what is possible and, more importantly, what is *good*. -in the next chapter, I will look at the theory (in the practice part

5. Theory

This chapter presents a theory on Generative AI (GenAI) games and their design, based on a thematic analysis of an analytical autoethnographic case study. The theory is grounded in a developed glossary, which serves as a coding framework. Terms from this glossary were grouped thematically and integrated into conceptual models.

5.1 Pillars

The study identifies three pillars of the game: mechanics, significs, and agents. These pillars interact to create the overall game experience and provide a framework to understand how the game constrains and produces outcomes.

5.1.1 Mechanics

Mechanics refer to the functional elements, rules, processes, and structures that constrain and produce the game's functional outcomes. They are the backbone of how a game *function* and include:

- Rules of play (e.g., turn order, win conditions)
- Game systems (e.g., combat systems, economy systems)
- Player actions and their functional consequences
- Algorithms that drive game behavior

For example, in a chess game, mechanics include how each piece moves, the rule of check and checkmate, and pawn promotion. It also affects the possibility space and action Space: all possible states that can be reached, as well as all permitted actions. In a video game, mechanics might include physics simulations, resource management systems, or combat calculations. They are typically precise, unambiguous, and programmable. However, in my study I have seen another type of mechanics: generative mechanics. These mechanics are not completely programmable, and rely on an agent to fill in the gaps. For example, in the TTRPG Apocalypse World, moves are mechanics which cannot be programmed without GenAI. They are triggered depending on the story context and player actions. For example, the move "READ A SITCH" states "When you read charged situation, roll+sharp." If the situation is not charged, or if the player is not able to "read" it (e.g. they are blindfolded or not present), then they cannot perform this move. However, what constitutes a "charged situation"? What counts as "reading"? These questions cannot be resolved without interpreting the context, which is hardly computer-implementable without GenAI. This leads us to the next, complementary pillar.

5.1.2 Significs

Significs are the elements, rules, influences, processes, and structures in the game that constrain, determine and guide the game's meaningful or functional outcomes and their interpretation. This term encompasses:

- Narrative elements (story, dialogue, character development)
- Thematic content
- Artistic design (visual style, sound design)
- Symbolic representations within the game

Significs communicate meaning through signs and symbols, contributing to the game's atmosphere, story, and emotional impact. For instance, in a role-playing game, the backstory of a character, the visual design of a magical artifact, or the music that plays during a climactic battle are all part of the game's significs. Unlike mechanics, which can often be reduced to mathematical or logical constructs, significs deal with interpretation and context. They are what make a game more than just a system of rules, giving it with meaning and fictional significance. Let us return to the Apocalypse World example. The mechanics of the "READ A SITCH" include the trigger and rolling the dice. So, in strictly mechanical terms, the move is: When you [blank], roll the dice. The meaning of "read a charged situation" is part of the significs. Furthermore, when the player decides if the move is triggered by the current situation (e.g. Mercy's goons are trying to break into my house and I am looking around to weigh my options), that too is significs. So, to implement this move, we need to combine mechanics and significs. This drives home my point that these are interacting pillars. However, we need someone or something to do the actual interpreting.

5.1.3 Agents

Intelligent agents are entities that act within the framework of the game, influencing and constrained by both mechanics and significs. Agents are unique in that they can make decisions, adapt to circumstances, and potentially act in ways not explicitly prescribed by the game's rules. Players are the most obvious agents, making choices that drive the game forward. However, in modern games, AI-controlled NPCs can also exhibit complex, agent-like behaviour. Still, I make an important distinction between agents and such NPCs. This difference boils down to simulated behaviour and behaviour generated by intelligence. Historically, AI-controlled NPCs have always been completely simulated, or determined completely by rules or the mechanics. This means that, while appearing agent-like, these entities do not possess any authentic agency. Instead, they are automatons. True agents, on the other hand, are not simulated: they are capable of *generating* behaviour that is determined by themselves rather than automated rules. Agents are inherently capable of generation.

There are two types of agents: player agents and host agents. Player agents act on their own behalf, whereas host agents act on behalf of the game as facilitators. In Super Mario Bros., the movement of goomba's is simulated, part of the mechanics, while the shape of the clouds was preauthored and part of the game's significs. What is neither simulated or authored are the movement input of Mario, generated by the player.

Af for host agents, the game master in a TTRPG acts as one, facilitating the game by portraying the world, NPCs, adjudicating rules and so on. In Yorecraft, the LLM that runs on the server acts as a host agent, as it generates content based on its understanding of the mechanics and signficis (context) of the game.

Finally, we should consider agent capabilities. Text-based LLMs are capable of understanding and generating text in many languages, while multimodal LLMs can understand text, audio and images. These capabilities correspond to the many different forms of intelligence, such as creativity, problem solving, logic, language and so on. The introduction of advanced AI, particularly large language models (LLMs), is expanding the potential of non-player agents in games. These AI agents can potentially understand nuanced context, act and adapt accordingly, and generate novel behaviour and content. When designing instructions for a host agents, we must be familiar with its capabilities, as do TTRPG designers when they create their systems for GMs. For example, Ironsworn recognizes that preparing content takes time and energy, which may be wasted if never used. So it provides the GM and the players with run-time tools to generate everything on the spot. GenAI, on the other hand, may prepare thousands of potential story lines without getting fatigued, but may struggle to be as creative as human players.

These pillars may help us to understand GenAI game design: we design the mechanical rules, meaningful context and host agents that together shape the game for the player.

5.2 Outcomes

Outcomes are fundamental to games, whether they are sports, card games, digital games, or tabletop role-playing games (TTRPGs). They answer critical questions such as:

- Has a player scored a goal in a football match?
- Was an attack effective in a Pokémon battle?
- How does a failed dice roll affect the narrative in a TTRPG?

This theme emerged consistently throughout the case study. For instance, in Yorecraft, when combining elements like "Air" and "Water" their outcome must be determined. In the generative text adventure, outcomes dictate available actions and their impact on the story progression.

These generative games use GenAI to produce these outcomes, setting them apart from traditional digital games.

5.2.1 Methods for producing outcomes

The method of producing outcomes may explain the historical gap between digital games and TTRPGs. Digital games often rely on authored story outcomes, and often feature high-fidelity real-time simulations. For example, BeamNG.drive employs real-time soft-body physics to simulate vehicle handling and destruction. TTRPGs, on the other hand, generate story outcomes collaboratively among players. Other outcomes may involve simple simulations (e.g., resolving a vehicle collision by subtracting a die roll from a health pool), but this method is much less sophisticated than narrative generaton. Continuing the vehicle collision example: in addition to calculating the health pool, the game master might generate a new story situation: the side car door rips off, and an enemy pursuer grabs hold of a player, trying to pull them out of the car.

Generative AI introduces a third method for producing run-time outcomes: generation. This method, central to TTRPGs, has been largely absent from digital games until now. To fully explore the potential of GenAI games, designers must break away from the traditional paradigms of simulation and authoring.

5.2.2 Outcome types

To understand the differing strengths and weaknesses of authoring, simulation, and generation, I propose four outcome types, the 4F model:

- function: Outcomes resulting from mechanics, independent of interpretation, determining the game its functional behaviour.

- fiction: Outcomes resulting from significs, dependent on observer interpretation, determining the game's meaning and context.
- form: How the game is shaped in space.
- flow: Changes that occur over time.

Functional outcomes are analogous to real-world physical outcomes, like an apple falling from a tree. For example, in Super Mario Bros., Mario's falling speed and the reward for collecting 100 coins are functional outcomes.

Fictional outcomes require interpretation. Mario's iconic design affects the player's perception but not the game's core functionality. In Minecraft, objects shaped like apples spawn when cutting down a tree. If these objects were shaped like coins instead, the game would function identically, but the player's interpretation of the game's fiction would change.

Function and fiction manifest as form and flow, as shown in Figure 5.1. In Super Mario Bros. (Figure 5.2), block textures and sky details contribute to fiction but not function. Mario's pose changes during jumps affect fictional flow without functional impact. Figure 5.3 demonstrates how a game stripped of fiction is a functional prototype.

5.2.3 Methods analyzed

The 4F-model provides a framework for analyzing three primary methods of outcome creation in game design: authoring, simulation, and generation. Each method has distinct characteristics and applications in relation to functional and fictional outcomes.

Authoring affords designers and artists complete control over game outcomes. In the absence of run-time generation, it has been the predominant method for establishing fictional outcomes. Authored elements include cutscenes, dialogue options, and character designs, all of which are crafted during the design phase. However, this method is constrained by authorial burden—the amount of content that can be feasibly created prior to run-time.

Simulation is particularly suited for functional outcomes. While it has been applied to fictional outcomes, such as in systems utilizing the storylet approach, these applications are still limited by authorial burden, as they require pre-authored content. Fundamentally, simulation relies on formalized rules, which can be at odds with the contextual and interpretative nature of fiction. However, when applied to functional outcomes, simulation can yield highly emergent and dynamic results.

Generation offers flexibility comparable to authoring and is compatible with fictional outcomes, as it can incorporate context and meaning. We may generate fictional outcomes such as cut-scenes, dialogues, character designs and so on. Crucially, generation can operate at run-time, potentially revolutionizing the production of outcomes in games. This method extends the flexibility and fiction-compatibility of authoring into the run-time phase of the game.

It is worth noting that generation is not limited to fictional outcomes,as demonstrated by projects like GameNGen running Doom, which generates how the game functions, replicating Doom's mechanics. In my case study, in the crafting game with the island, generation of events also involved generating values for the status effects. In the generative text adventure game, the LLM decided which parts of the texts could be clicked on, affecting the game's function.

5.2.4 State and content

It may strike the reader as odd to call static elements such as texture *outcomes*. Mario's final score is certainly an outcome, but is a texture an outcome? One is the outcome of play, and the other is the outcome of *design*. I call one *content* and the other *state*. It is content because it is already in the game before play starts. A player cannot affect what content the game comes with, as it is not the outcome of any system they can interact with.

State is fleeting: it is the outcome of the game's systems at play, whereas content is persistent. It is the outcome of decisions that happened before play, at design time. As a consequence, all players have access to the same content. State, on the other hand, is not shared between players, but is spe-

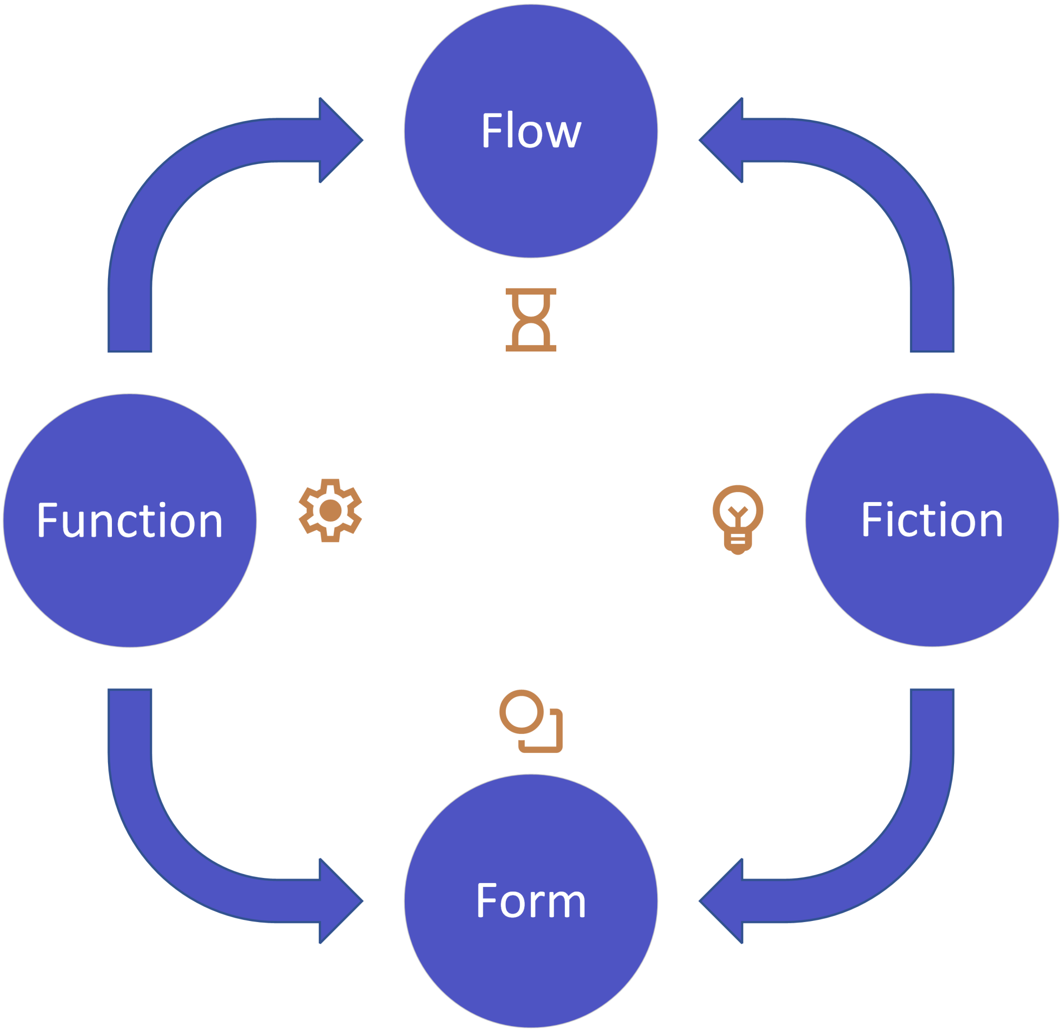


Figure 5.1: In the 4F-model, function and fiction manifests as form and flow.

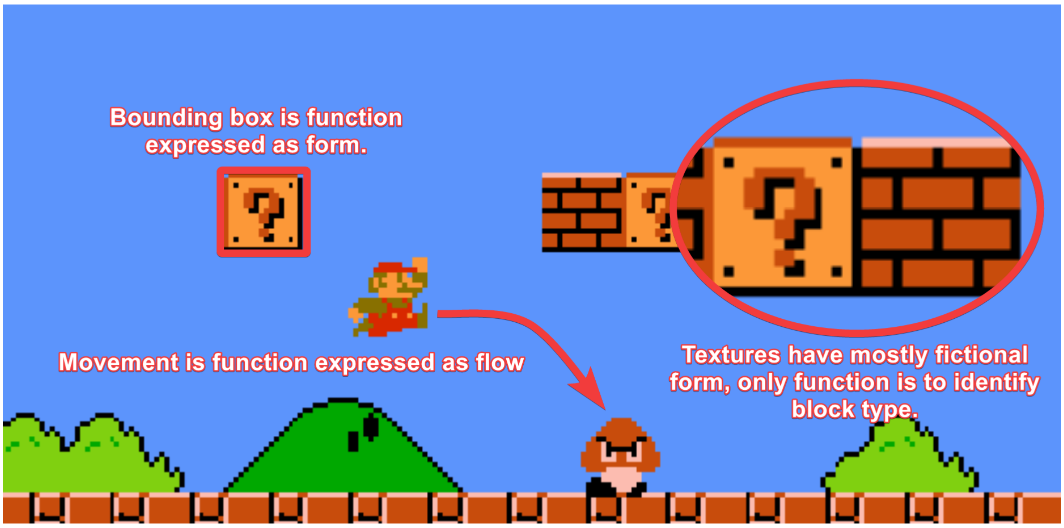


Figure 5.2: Outcome types visualized in Super Mario Bros.

Figure 5.3: Functional prototype of *Dead Spear* on the left and the full game on the right visualize how a game can be stripped of fiction without impeding its function.

cific to a single play through.

Both state and content can be authored, simulated or generated. We are familiar with authored content, as it is likely the only kind of content most players have encountered. Simulated content was produced not by hand, by through simulation. For example, in No Man's Sky, none of its trillions of planets were handcrafted. Rather, they were "procedurally generated", but under my definition this falls under simulation, because it is based on algorithms that do not use GenAI techniques.

Authored states are initial states, which are fleeting in the sense that as soon as play starts, the initial state disappears, but are similar to content in that they are shared across play throughs. This can be seen again and again in games: configurations that are hand-crafted but meant to be disturbed, like physics puzzles. Authored content on the other hand is ubiquitous, like environmental designs, music and so on.

Generated content *acts* like authored content, but can be used at run-time and cached to continually generate new content as the players explored the total generate-able space. This is the case in Yorecraft and the generative *hyper*text adventure. Most unlike traditional games, and most like TTRPGs,

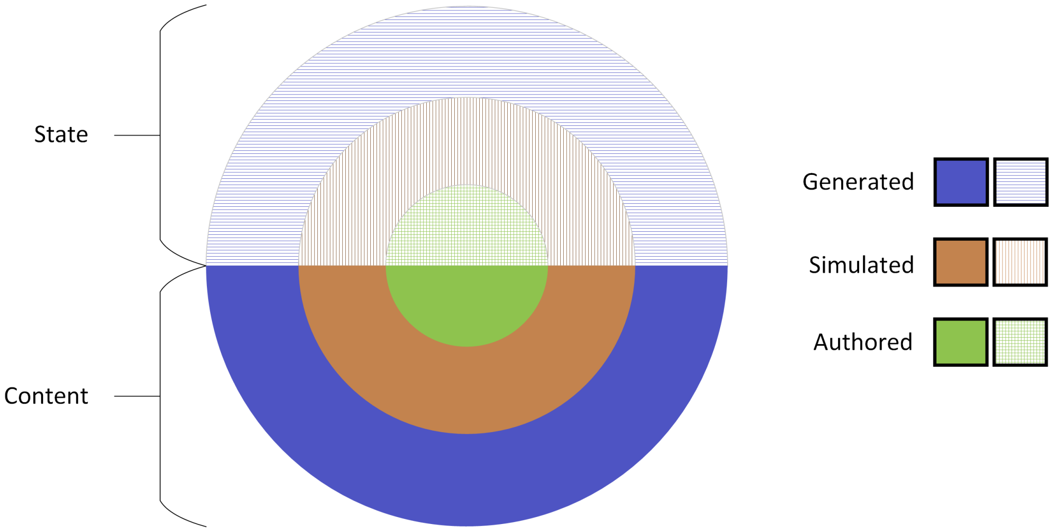


Figure 5.4: Content-state model.

is the generation of state: fleeting and unique to every playthrough. This can be seen in GameNGen's Doom, as the game's state is generated in real-time, and due to the unpredictable nature of GenAI, likely different every time someone plays it. It can also be seen in the generative *text* adventure and the card game with generative narratives. We can imagine how GenAI may affect flow, form, function and fiction as state in our designs, and attempt to create completely new types of games.

The 4F model, combined with the three methods of producing content and state outcomes, shows us how GenAI can integrate in game design to bridge the gap between TTRPGs and digital games. In particular, it shows us that GenAI can generate state as function, fiction, form and flow, something most unlike anything seen before in digital games.

5.3 Perspectives

A recurring theme in my study was that of perspectives. I propose the Possibilities, Operation and Virtual world or POV-model that encompasses three distinct perspectives: the game as a potential space, as a real-world operation, and as a virtual world. These perspectives offer a framework for designers to effectively constrain GenAI, implement it within game systems, and understand its impact on the game world.

5.3.1 Possibilities and constraints

When viewing the game as a potential space, we analyze how game mechanics and significs constrain both the game and its agents to a set of possibilities. This perspective contrasts with traditional linear media, emphasizing the exploratory nature of gameplay. The player may explore the game's content in many different ways: in different orders, states, contexts, some content may be skipped or new content may be generated entirely. This is stark contrast to traditional media like films or books, where the content can only be experienced in a fixed manner.

The designer that looks through this lens, considers how the game's design and rules leads to many different playthroughs: the potential space across playthroughs. For the player the game's possibilities are explored within a playthrough: rather than considering all possibilities for every player, the player considers the possibilities available to them in the context of their playthrough. Previous choices may preclude the player from some content, while their present circumstances may provide opportunities denied others. At this level, the player may formulate goals, plans and strategies, for example to win, explore new locations and so on.

In my study, I considered how generative AI could constrain the potential space distinctly from traditional digital game systems. Authoring gives the designer very explicit authorial control over the game's possibilities, while simulation lets the designers control the space more implicitly. The designer can not predict every emergent outcome of the simulation, but do to its closed, procedural nature, it is not as unpredictable as generation. This ties to procedural clarity and hard design: hard design, or simulation, lets players and designers reason about outcomes by internalizing the simulation rules. These outcomes can be put on an explicit-implicit spectrum.

Generation, on the other hand, must be guided with soft design, or guidelines, context and prompt engineering. Generation allows for a much larger, open possiblity space, and to ensure the generation is constrained to desirable generations, we must master soft design. However, this can create open-ended gameplay and this can surprise and delight players, as we have seen with AI awareness.

Through the possibility perspective, we observe divergence, the tendency for similar conditions to become dissimilar and convergence, the tendency of similar conditions to become dissimilar. When we author the space ourselves, when designing a branching narrative, for example, we have explicit control over divergence and convergence. However, when we allow GenAI to control the narrative, our control becomes implicit. The possibility space, then, also becomes implicit. Whereas in an authored branching narrative, we have seen the entire possibility space, in the case of generated narrative, we have not. Through playtesting, we can make inferences about how the LLM will behave, but considering the vast space of possibilities, we cannot know for certain for all cases. Achieving convergence and divergence as desired is therefore a challenge, but which may be overcome with a mix of hard and soft design and inserting preauthored content. In general, constraining GenAI in addition to the simulation is not trivial.

5.3.2 Operation

The game can also be viewed as a real-world process, as an operation. In a TTRPG, for example, the operation involves rolling dice, consulting tables, following rules, scribbling on character sheets, drawing maps and, above all, engaging in conversation. In a video game, the hardware must execute the software, performing physics calculations, rendering 3D objects and lighting, processing input and output and so on, while the player must interpret and manipulate the hardware interface. As the designer, we think about how we can implement the game's potential space. We need code, data, control structures and so on. We need to make sure the operation proceeds smoothly. In my study, I adopted this perspective as I designed the prompts to ensure valid outputs, so that I could integrate the LLMs output into the rest of the system. Latency masking, infrastructure, cost, API design were considerations of the game's operation.

The operation is an important perspective to take, as it concerns how the systems actually work, and how well they work. Can the vision for the game actually be implemented? We must also consider the player. The player exists in the real world, and thus needs to interface with the game. How is the interface implemented, and how can the player interact with it? What real-world actions are expected of the player, like talking or manipulating controllers.

5.3.3 Virtual world

Finally, we must view the game as a virtual world, which is constructed during the operation within the constraints of the potential space. The virtual world in a TTRPG is primarily constructed through conversation as shared imagination. In digital games, the virtual world is simulated and rendered in a digital interface: on screens and speakers.

The game's outcomes are ultimately understood as part of the virtual world. While content at the operational level can be considered in terms of data formats, data processing or contextual processing, it is in the virtual world all the game's separate elements are integrated into a single whole.

The 4F-model applies to the virtual world, and understanding how GenAI helps construct the virtual world for the player is crucial to understand the player's experience. In Yorecraft, the image model creates an icon for each element, significantly contributing to the game's ficitonal form. In the LLM-powered game with simulation, the LLM determines the functional attributes of new creatures, as well as fictional attributes such as size and color. Thus the LLM determines how new creatures function in the world. Mistaken generations may ascribe "snakes" the ability to fly, which could lead to flying snakes eating birds, or give creatures random colors. In the generative text adventure game, the game's flow, function, and fiction is determined by the LLM. How much time passes fictionally, the descriptions of locations and the words used in the story depend on GenAI.

To ensure we create the virtual world that we envision, we must learn to constrain the possibilities of GenAI, implement the systems that integrate them in the operation, make sure that the authored elements integrate well with the simulation systems, and the simulation systems integrate well with

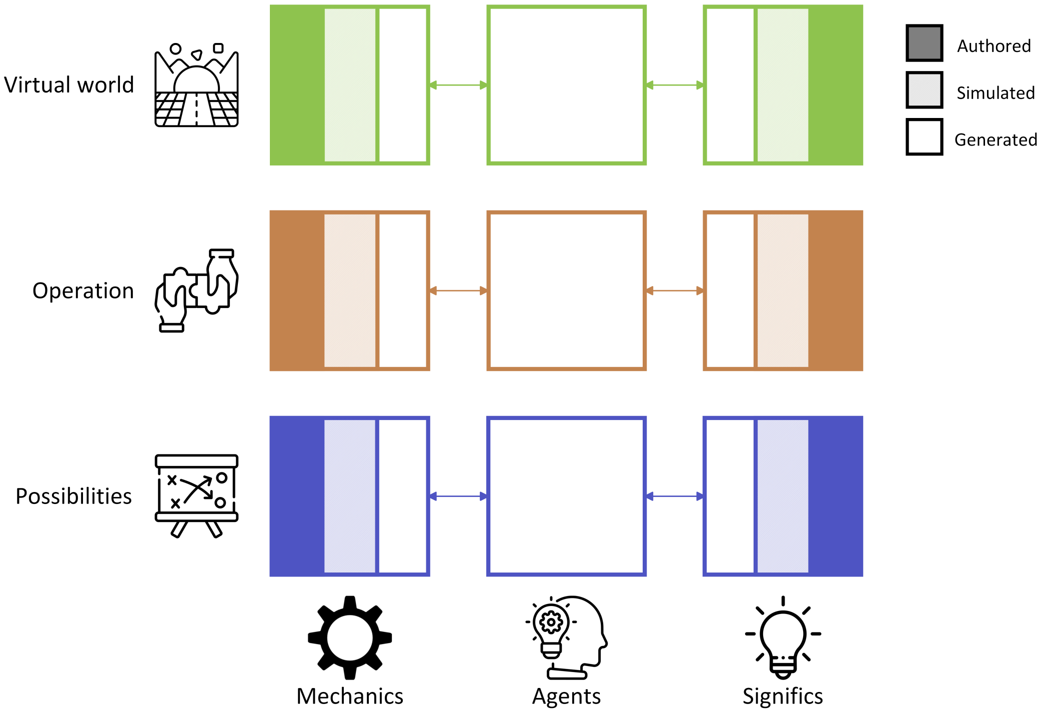


Figure 5.5: MAS-framework with POV-model: MAS constrain Possibilities; Operation implements MAS; Operation produces outcomes in virtual world. Content/state and 4F-model left out for clarity, but still apply.

the generative systems. Finally, we must consider players as agents too, as they generate as do GenAI agents, and we must integrate them into the system to ensure they have the intended agency. Figure 5.5 the proposed theory together.

5.4 Conclusion

This chapter has presented a comprehensive theory for understanding and designing Generative AI (GenAI) games. The theory is built upon three key components:

- The MAS (Mechanics, Agents, Significs) pillars, which provide a framework for understanding the core elements of GenAI games.
- The 4F (Function, Fiction, Form, Flow) model of outcomes, which helps categorize and analyze the various outputs of game systems.
- The POV (Possibilities, Operation, Virtual world) model, offering three distinct perspectives for game design and analysis.

These components, along with the distinction between content and state, offer a nuanced approach to conceptualizing how GenAI can be integrated into game design. This theory bridges the gap between traditional digital games and tabletop role-playing games, highlighting the unique potential of GenAI to generate both content and state across all outcome types. By providing a structured way to think about the constraints, implementation, and impact of GenAI in games, this theory aims to equip designers with the tools to create innovative and engaging experiences. As the field of GenAI games continues to evolve, this theoretical framework offers a foundation for further research, development, and creative exploration in game design.

Part III

Implications

6. Discussion

This study explored the emerging field of generative AI (GenAI) games through an autoethnographic case study, aiming to bridge the gap between tabletop roleplaying games (TTRPGs) and digital games. The research provides valuable insights into the theoretical and practical considerations of designing games that integrate generative AI to create novel, open-ended narrative and gameplay experiences. The study's findings highlight the unique potential of GenAI in game design, particularly in addressing the longstanding challenge of authorial burden in interactive narratives. By leveraging large language models (LLMs) for real-time content generation, GenAI games can potentially offer the flexibility and open-ended storytelling of TTRPGs within a digital format.

This study aimed to explore the design of generative AI (genAI) games, which integrate generative AI such as large language models (LLMs) at runtime. The focus was on how generative games can bridge the gap between tabletop role-playing games (TTRPGs) and video games. Through an autoethnographic approach, I documented my own practice of designing generative games during an internship at Yvora Game Studio. This process involved designing and prototyping a series of generative games, culminating in Yorecraft, a commercial game featuring crafting and request mechanics that integrate AI-generated content at runtime. Based on a thematic analysis of my documented experience, I propose a new theory of generative games.

One of the most significant findings of this study was the inadequacy of traditional game design frameworks in describing and analyzing generative games. Throughout the research process, I encountered substantial difficulties in articulating my experiences using existing frameworks, necessitating the creation of new terms and concepts. This challenge underscores the novelty of the generative game design space and suggests that we are not merely extending existing game design paradigms, but potentially entering a new territory that demands its own theoretical foundations.

The effectiveness of using the "gap between TTRPGs and digital games" as a lens for identifying new possibilities in digital game design was an unexpected but fruitful aspect of this study. Initially, this approach might have seemed to suggest a focus on emulating TTRPGs in digital formats. However, the study revealed that this perspective offers much broader insights. The development of the crafting mechanic in Yorecraft demonstrated that by combining the unique qualities of digital games with generative capabilities, we can create experiences that are not only distinct from traditional video games but also go beyond what is possible in TTRPGs.

This realization shifts our understanding of the relationship between TTRPGs and digital games. Rather than seeing them as separate categories with a gap to be bridged, we can view them as points on a shared spectrum of interactive experiences. From this perspective, TTRPGs serve not as a model to be emulated, but as a proof of concept, demonstrating that there are types of games and interactive experiences that cannot be created within traditional digital game paradigms.

The development of Yorecraft, a seemingly simple single-player game, revealed unexpected technical challenges that highlight the unique considerations in generative game design. These challenges centered around two main areas: the hardware requirements for running LLMs and the paradigm of a "shared cache" for generations. The computational demands of running LLMs presented a significant hurdle, requiring a balance between the quality of generated content and performance, especially when targeting a wide range of consumer hardware. The concept of a "shared cache" for generations introduced another layer of complexity, requiring careful management of generated content to maintain consistency across play sessions and between different players. These technical challenges underscore the need for new approaches in game architecture and optimization when working with generative AI, suggesting that the development of generative games may require a different set of skills and considerations compared to traditional game development.

Finally, my proposed theory points need not replace existing design frameworks. Notably, I have left out the three-stage order of design, dynamics and experience. This pattern is ubiquitous and nearly universally applicable, so repeating it in my theory does not offer a new perspective. Instead, the proposed theory offers more specificity, which can be applied to design, dynamics and experience.

6.1 Limitations

While this study offers valuable insights into the design of generative games, it's important to acknowledge its limitations to contextualize the findings and identify areas for future research.

6.1.1 Subjectivity and Perspective

The autoethnographic approach of this study, while providing rich, in-depth insights, is inherently limited by its reliance on a single perspective. This approach, while valuable for exploring new design spaces, may not capture the full range of experiences and challenges that other designers might encounter when working with generative AI in games.

6.1.2 Nature of the Proposed Theory

It's crucial to emphasize that the "theory" proposed in this study is a design theory rather than a scientific theory in the traditional sense. Design theories are inherently subjective and prescriptive, aiming to guide practice rather than describe universal truths. As such, the theory presented here may not resonate equally with all designers or researchers in the field.

6.1.3 Small scale

The prototypes developed during this study were small in scope and experimental in nature. Even the commercial game, Yorecraft, while more fully developed, is tiny when compared to AAA game productions. This limitation raises questions about how generative game design principles might change or evolve when applied to larger, more complex, and more expensive projects.

6.1.4 Limitations of Generative AI

While this study drew inspiration from TTRPGs, it's important to rec- ognize that generative AI, despite its capabilities, is not equivalent to human game masters or players. The risks, biases, and limitations inherent in current generative AI systems may significantly constrain the types of generative games that can and should be made.

6.1.5 Market Considerations

This study focused primarily on design and technical aspects of generative games. However, the commercial viability and market demand for such games remain open questions. While Yorecraft was deployed and is available online to play, we have no data for its market viability. More comprehensive market research would be necessary to understand the potential audience for generative games and how they might fit into the broader gaming landscape. Market for generative AI games can drive their growth, leading to a greater number and variety of games and new advancements and discoveries.

6.1.6 Technology constraints

The study was conducted using specific LLM models (e.g., Claude) and hardware setups. The rapid advancement of AI technology means that some findings may become outdated as more powerful or efficient models become available, and as consumer hardware becomes more capable of running them. However, it is worth noting that cloud-based architecture become more relevant as models seem to be getting ever larger.

6.2 Future Work

The findings of this study open up several exciting avenues for future research and development in the field of generative games:

6.2.1 Player Experience in Generative Games

Future studies could explore the relationship between player experience and generative games, including the identification and characterization of new player experiences unique to generative games, such as "AI awareness." This would help us understand not only how these games are ostensibly different from traditional digital games, but also what distinct experience they can offer to players.

6.2.2 Generative Game Design Patterns

As the field of generative game design matures, it will be valuable to identify and catalog recurring design patterns, developing a taxonomy of generative game mechanics and their effects on gameplay.

6.2.3 State of the Field Analysis

A comprehensive analysis of the current state of generative games would provide valuable context for future research and development, including interviews with generative game designers and analysis of existing generative games.

6.2.4 Practical Application and Validation of the Theory

To further develop and refine the theory proposed in this study, it would be valuable to put the theory into the hands of other designers and observe how they apply it in their own projects.

6.2.5 Technical Advancements

Future work should focus on addressing the technical challenges identified in this study, such as developing optimized LLM implementations for real-time game applications and creating tools and middleware specifically designed for generative game development.

6.2.6 Ethical Considerations and AI Limitations

Future research should take a deep look into the ethical implications and current limitations of using generative AI in games, including methods to mitigate bias and unsafe content in AI-generated game content. Other concerns exist as well, such as the rights of the creators of copyrighted works GenAI models are trained on, and excessive energy consumption in light of climate change.

6.3 Conclusion

This study offers a first-of-its-kind, deep look at generative games from an insider perspective, contributing new insights ranging from player experience and design approaches to technical challenges and a redefinition of what is possible in games. The proposed theory and design framework provide a foundation for understanding and creating generative games, bridging the gap between TTRPGs and traditional digital games. This research has made clear the need for design approaches specific to generative games, as traditional game design paradigms prove inadequate in fully capturing the unique characteristics of this emerging field.

This study serves as a valuable starting point for future research and development in generative game design. As the field evolves, addressing the identified technical challenges, exploring new design patterns, and investigating the ethical implications of AI in games will be crucial in realizing the full potential of generative games. If games are feats of imagination and technology, then let us imagine how generative AI can lead to a wholly new type of games: generative games.

A. Ethics and Privacy Quick Scan

Response Summary:

Section 1. Research projects involving human participants

P1. Does your project involve human participants? This includes for example use of observation, (online) surveys, interviews, tests, focus groups, and workshops where human participants provide information or data to inform the research. If you are only using existing data sets or publicly available data (e.g. from X, Reddit) without directly recruiting participants, please answer no.

- No

Section 2. Data protection, handling, and storage

The General Data Protection Regulation imposes several obligations for the use of **personal data** (defined as any information relating to an identified or identifiable living person) or including the use of personal data in research.

D1. Are you gathering or using personal data (defined as any information relating to an identified or identifiable living person)?

- No

Section 3. Research that may cause harm

Research may cause harm to participants, researchers, the university, or society. This includes when technology has dual-use, and you investigate an innocent use, but your results could be used by others in a harmful way. If you are unsure regarding possible harm to the university or society, please discuss your concerns with the Research Support Office.

H1. Does your project give rise to a realistic risk to the national security of any country?

- No

H2. Does your project give rise to a realistic risk of aiding human rights abuses in any country?

- No

H3. Does your project (and its data) give rise to a realistic risk of damaging the University's reputation? (E.g., bad press coverage, public protest.)

- No

H4. Does your project (and in particular its data) give rise to an increased risk of attack (cyber- or otherwise) against the University? (E.g., from pressure groups.)

- No

H5. Is the data likely to contain material that is indecent, offensive, defamatory, threatening, discriminatory, or extremist?

- No

H6. Does your project give rise to a realistic risk of harm to the researchers?

- No

H7. Is there a realistic risk of any participant experiencing physical or psychological harm or discomfort?

- No

H8. Is there a realistic risk of any participant experiencing a detriment to their interests as a result of participation?

- No

H9. Is there a realistic risk of other types of negative externalities?

- No

Section 4. Conflicts of interest

C1. Is there any potential conflict of interest (e.g. between research funder and researchers or participants and researchers) that may potentially affect the research outcome or the dissemination of research findings?

- No

C2. Is there a direct hierarchical relationship between researchers and participants?

- No

Section 5. Your information.

This last section collects data about you and your project so that we can register that you completed the Ethics and Privacy Quick Scan, sent you (and your supervisor/course coordinator) a summary of what you filled out, and follow up where a fuller ethics review and/or privacy assessment is needed. For details of our legal basis for using personal data and the rights you have over your data please see the University's privacy information. Please see the guidance on the ICS Ethics and Privacy website on what happens on submission.

Z0. Which is your main department?

- Information and Computing Science

Z1. Your full name:

Ali Esat Özbay

Z2. Your email address:

ozbay.ae@gmail.com

Z3. In what context will you conduct this research?

- As a student for my master thesis, supervised by::
  Dr. J. Frommel

Z5. Master programme for which you are doing the thesis

- Game and Media Technology

Z6. Email of the course coordinator or supervisor (so that we can inform them that you filled this out and provide them with a summary):

ozbay.ae@gmail.com

Z7. Email of the moderator (as provided by the coordinator of your thesis project):

ozbay.ae@gmail.com

Z8. Title of the research project/study for which you filled out this Quick Scan:

Informal and Formal Design for Interactive Systems

Z9. Summary of what you intend to investigate and how you will investigate this (200 words max):

I intend to investigate a new game design framework for creating narratives games using natural language agents such as large language models, inspired tabletop roleplaying games. This is in response to recent advances in AI and long standing complexity burdens on designers when designing dynamic, adaptive narrative in games. I intend to research this inductively based on other design frameworks and tabletop roleplaying games, based on my experiences in a case study for [removed], where I will create a novel type of game based on large language models: a roguelike deck building narrative card game.

Z10. In case you encountered warnings in the survey, does supervisor already have ethical approval for a research line that fully covers your project?

- Not applicable

Scoring

- Privacy: 0
- Ethics: 0

Glossary

4F model

Function, Fiction, Form, and Flow - the four lenses through which outcomes are observed in the framework. 1, 53

action Space

The set of possible actions or choices available to an agent within the game, shaped by mechanics and significs. 1, 49

agent

An entity within the game capable of independent action and decision-making, including players and GenAI-powered host agents.. 1, 49

AI awareness

The player's recognition and understanding that certain elements of the game, such as narrative or content, are generated or influenced by AI. This awareness can impact player behavior, strategies, and engagement. 1, 39, 60

authorial control

Degree to which designers control game outcomes, who have full control over authored content, control over the rules of the simulation, and least control over generation, via soft design and prompt engineering.. 1, 59

authoring

The method of pre-authoring outcomes of the game, such as narrative branches, character designs and level layouts, which are established before play begins.. 1, 53

automaton

A non-player character or game element whose behavior is entirely determined by pre-programmed rules or algorithms, lacking the capacity for generation. 1, 51

caching

Storing frequently accessed data locally to improve performance and reduce the need for repeated requests to external sources like LLMs, and turning the generation into delayed content creation in a single, shared world.. 1, 47

capabilities

The range of actions and interactions an agent can perform within the game world, encompassing both external (physical) and internal (cognitive) abilities. 1, 52

client-server architecture

A network architecture where clients (e.g., player's devices) request services or data from a central server. This can be necessary in LLM-powered games for caching and offloading computationally intensive tasks. 1, 42

closed Design

A design approach where the mechanics and significs strictly define the possible outcomes and experiences, leaving little room for generation.. 1

convergence

The tendency of playthroughs to converge from dissimilar conditions to similar conditions. 1, 37, 60

divergence

The tendency of playthroughs to diverge from similar conditions to dissimilar conditions. 1, 37, 60

dynamics

The interplay between mechanics, significs, and agents at the design, play, and experience levels, shaping the emergent properties and overall experience of the game. 1

emergent content

Elements of the game world that arise dynamically during play, not explicitly pre-defined by the designer, often resulting from the interaction of mechanics, significs, and agent actions. 1

explicit-implicit spectrum

Outcomes can be put on this spectrum, ranging from explicitly created by the designer, to implicit states emerging from deep dynamics and subtle influences to generation.. 1, 59

exploration

The process by which players discover and interact with the generated content and possibilities of the game world, often driven by curiosity and a desire to uncover emergent narrative and gameplay elements. 1

external capability

An agent's ability to interact with and manipulate the virtual world, including movement, combat, and object interaction. 1

fiction

The game's meaning and context conveyed to and interpreted by an observer.. 1, 54

flow

The dynamic unfolding of events and experiences within the game over time, encompassing pacing, rhythm, and the sense of progression. 1, 54

form

The visual and sensory representation of the virtual world, including graphics, sound, and user interface elements, shaped by both mechanics and significs. 1, 54

function

The game's function is determined independently of the game's fictional meaning as an outcome of the game's mechanics. Analogous to the real world's physics.. 1, 53

guidance

The design elements, such as rulebooks, flavor text, or examples of play, that provide players with direction and inspiration for navigating the game's possibility space and generating meaningful content. 1

hard design

Coercive control over play via hard, unambigious and computable rules.. 1, 59

hard mechanics

Unambiguous rules and procedures that govern the game's operation, leading to deterministic and predictable outcomes. 1

hard procedure

A process characterized by well-defined, computable steps that take well-defined input and produce an output completely determined by the input and the computation. 1

hard significs

Explicit narrative elements and guidelines that shape the game's fiction, leaving little room for player interpretation or ambiguity. 1

hard space

A discrete and constrained possibility space defined by hard mechanics and significs, limiting agent freedom and emphasizing predictable outcomes. 1

hard subsystem

A tool or component within a game that operates with well-defined input and output, following rigid rules and producing predictable results. It is often associated with traditional game mechanics and computational processes. 1

hard-to-soft mapping

The process of translating well-defined, structured data from the hard system (e.g., game state, numerical values) into expressive and meaningful representations within the soft system (e.g., visuals, narrative).. 1

host

An agent, often controlled by the game system or a designated player, responsible for facilitating play, enforcing rules, and managing the virtual world. 1

host agent

An agent acting on behalf of the game itself, often to facilitate play, manage the virtual world, or enforce rules. 1, 51

internal Capability

An agent's cognitive abilities within the game, including understanding, interpretation, decision-making, and memory. 1

known world

A type of narrative space where the entire narrative space is known at design time, giving designers full control but requiring them to author everything. 1, 37

LLM-powered game

Games that integrate Large Language Models directly into gameplay, not just as development tools. 1, 61

masking

Hiding the loading or processing time of intensive operations such as AI generation by allowing the game loop to continue while those operations run in the background. 1, 47

mechanics

The elements, rules, structures and processes that determine the game's function.. 1, 49

meta

Aspects of the game that exist outside the virtual world, such as player discussions, community interactions, and real-world consequences. 1

narrative context

The story elements and background information that shape player choices and understanding of the game world, often influencing available actions and potential outcomes. 1

narrative goal setting

The use of narrative elements to establish player objectives and motivations, guiding their actions and decision-making within the game. 1

narrative integration

The extent to which the narrative is an integral part of playing the game, influencing player decisions and shaping the overall experience. The thesis identifies four types of narrative integration: Flavor (non-consequential), Narrative context integration (options depend on the narrative), Narrative goal setting (narrative guides player goals), and Player-directed narrative (player shapes the narrative). 1, 32

offloading

Transferring computationally intensive tasks to more powerful hardware, such as cloud servers, to improve performance on less capable devices. 1, 47

open design

A design approach that prioritizes player agency and emergent gameplay by using flexible mechanics and significs, allowing for a wider range of possible outcomes and experiences. 1

open mechanics

Flexible and adaptable rules that provide players with greater influence over the game's simulation and outcomes. 1

open significs

Narrative elements and guidelines that encourage diverse interpretations, player-driven storytelling, and emergent meanings within the game world. 1

operation

The real-world perspective which implement the mechanics and significs, agent actions, system responses, and the construction of the virtual world. 1, 58

operational mechanics

The practical application and execution of game mechanics during play, influencing agent actions and the construction of the virtual world. 1

operational significs

The interpretation and application of the game's significs during play for the purpose of implementing the game's run-time. 1

outcome

Details that result from play that are applied to the game's virtual world, as opposed to the means of producing those details. Outcomes are functional or fictional, expressed over time as flow and expressed in space as form.. 1, 50, 52

outcome methods

Authored, Simulated, Generated - the three methods for producing outcomes that can occur in the virtual world. 1

paper prototyping

A design and testing method that uses low-fidelity materials (e.g., paper, cards, dice) to simulate game mechanics and interactions, often employed in the early stages of development to explore ideas and gather feedback. Seems hard to use for GenAI purposes, due to mechanical emphasis.. 1, 32

plan

The design-level blueprint that outlines the potential possibilities, constraints, and principles of the game, informing both its mechanics and significs. It also refers to the parts of the game that are pre-planned or predetermined. 1

player

An agent who participates in the game for their own enjoyment and goals, interacting with the virtual world and other agents. 1

player agent

An agent representing a human player within the game, acting on their own behalf and pursuing their own goals. 1

possibilities

Perspective that consider the potential outcomes of the game as constrained by (the rules of) mechanics and significs.. 1, 58

possibility space

The range of potential outcomes and experiences afforded by the game's mechanics, significs, and agent interactions. 1, 35

possible world

A type of narrative space where the possibilities are unknown but theoretically fixed to a single world, shared by all players. It is "uncovered" and then fixed in play. 1, 37

possible worlds

A type of narrative space where the indeterministic nature of LLMs leads to different worlds between playthroughs, with the same paths potentially leading to entirely different outcomes. 1, 37

pregenerated content

Pre-authored content, such as dialogue, descriptions, or narrative events, that exists within the game world before player interaction. 1

private model

An agent's internal understanding and representation of the game world, including their beliefs, goals, and strategies. 1

procedural clarity

The degree to which players can internalize the game's systems and predict outcomes based on their actions. LLM-based systems sacrifice procedural clarity for surprise and a sense of external influence. 1, 39, 59

rule coverage

The extent to which the game's mechanics and narrative are governed by explicit rules versus LLM-driven interpretation. If too little is covered by rules, the LLM might need to handle more than it's capable of. 1, 41

significs

The elements, rules, structures and processes that shape the game's fiction: its meaning and context. 1, 50

simulation

The computational processes that model the behavior and interactions of elements within the game world, often based on hard mechanics. 1, 53

soft design

Designers exert subtle influences over the players or generative agents via soft design, by changing the phrasing of prompts and guiding the imagination of players.. 1, 59

soft mechanics

Flexible rules and guidelines that influence the game's operation without strictly determining it, allowing for emergent behavior and player influence. 1

soft procedure

A process characterized by a natural language description of a task, where neither the input nor the output needs to be rigorously defined. The output is guided by the input and the description but is ultimately determined by an external system with its own capabilities and context. 1

soft significs

Open-ended narrative elements and guidelines that encourage player interpretation, creative expression, and emergent storytelling. 1

soft space

An open and flexible possibility space defined by soft mechanics and significs, emphasizing agent freedom and emergent outcomes. 1

soft subsystem

A tool or component within a game that operates with flexible input and output, capable of handling ambiguity and producing creative or unpredictable results. It is often associated with LLM-powered content generation and player interpretation. 1

soft-to-hard Mapping

The process of translating flexible and potentially ambiguous input from the soft system (e.g., player actions, natural language) into structured data that can be understood and processed by the hard system (e.g., game state updates, decision-making).. 1

spontaneous generation

The real-time, uncached generation of content by an LLM in response to player actions or prompts, leading to unique and unpredictable experiences. 1

statics

The unchanging elements of the game world, such as pre-defined maps, fixed storylines, or immutable rules, that provide a foundation for play. 1

Tabletop Roleplaying Game

collaborative storytelling game where players portray characters in a shared, imagined world through conversation moderated by a system of rules, where typically one player is the game master. 1, 6

uncached generation

The on-the-fly generation of content by an LLM without relying on pre-stored or cached responses, potentially leading to more diverse and unpredictable outcomes. 1

virtual world

The shared, imagined space where the game takes place, constructed through the interplay of mechanics, significs, and agent actions. 1, 58

Bibliography

[1] J. Gottschall, *The storytelling animal: How stories make us human*. Houghton Mifflin Harcourt, 2012.

[2] J. Huizinga, *Homo ludens ils 86*. Routledge, 2014.

[3] D. Schmandt-Besserat, *How writing came about*. University of Texas Press, 2010.

[4] M. O. Riedl and V. Bulitko, "Interactive narrative: An intelligent systems approach," *Ai Magazine*, vol. 34, no. 1, pp. 67–67, 2013.

[5] G. Trichopoulos, G. Alexandridis, and G. Caridakis, "A survey on computational and emergent digital storytelling," *Heritage*, vol. 6, no. 2, pp. 1227–1263, 2023.

[6] Y. Wang, J. Lin, Z. Yu, W. Hu, and B. F. Karlsson, "Open-world story generation with structured knowledge enhancement: A comprehensive survey," *Neurocomputing*, p. 126 792, 2023.

[7] Z. O'Shea and J. Freeman, "Game design frameworks: Where do we start?" In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, 2019, pp. 1–10.

[8] H. Koenitz, "Narrative in video games," in *Encyclopedia of computer graphics and games*, Springer, 2024, pp. 1230–1238.

[9] D. L. Roberts and C. L. Isbell, "A survey and qualitative analysis of recent advances in drama management," *International Transactions on Systems Science and Applications, Special Issue on Agent Based Systems for Human Learning*, vol. 4, no. 2, pp. 61–75, 2008.

[10] B. Kybartas and R. Bidarra, "A survey on story generation techniques for authoring computational narratives," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 9, no. 3, pp. 239–253, 2016.

[11] M. Bernstein, D. E. Millard, and M. J. Weal, "On writing sculptural hypertext," in *Proceedings of the thirteenth ACM conference on Hypertext and hypermedia*, 2002, pp. 65–66.

[12] A. E. Short, *Beyond Branching: Quality-Based, Salience-Based, and Waypoint Narrative Structures*, Read_Status: New Read_Status_Date: 2024-02-06T11:40:56.908Z, Apr. 2016. [Online]. Available: `https://emshort.blog/2016/04/12/beyond-branching-quality-based-and-salience-based-narrative-structures/` (visited on 01/02/2024).

[13] R. Hunicke, M. LeBlanc, R. Zubek, *et al.*, "Mda: A formal approach to game design and game research," in *Proceedings of the AAAI Workshop on Challenges in Game AI*, San Jose, CA, vol. 4, 2004, p. 1722.

[14] W. Walk, D. Görlich, and M. Barrett, "Design, dynamics, experience (dde): An advancement of the mda framework for game design,"

Game dynamics: Best practices in procedural and dynamic game content generation, pp. 27–45, 2017.

[15] R. E. Cardona-Rivera, J. P. Zagal, and M. S. Debus, "Gfi: A formal approach to narrative design and game research," in *Interactive Storytelling: 13th International Conference on Interactive Digital Storytelling, ICIDS 2020, Bournemouth, UK, November 3–6, 2020, Proceedings 13*, Springer, 2020, pp. 133–148.

[16] F. Peinado and P. Gervás, "Transferring game mastering laws to interactive digital storytelling," in *International Conference on Technologies for Interactive Digital Storytelling and Entertainment*, Springer, 2004, pp. 48–54.

[17] S. Góngora, L. Chiruzzo, G. Méndez, and P. Gervás, "Skill check: Some considerations on the evaluation of gamemastering models for role-playing games," in *International Conference on Games and Learning Alliance*, Springer, 2023, pp. 277–288.

[18] A. Zhu, L. Martin, A. Head, and C. Callison-Burch, "Calypso: Llms as dungeon master's assistants," in *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 19, 2023, pp. 380–390.

[19] J. Kelly, M. Mateas, and N. Wardrip-Fruin, "Towards computational support with language models for ttrpg game masters," in *Proceedings of the 18th International Conference on the Foundations of Digital Games*, 2023, pp. 1–4.

[20] T. Triyason, "Exploring the potential of chatgpt as a dungeon master in dungeons & dragons tabletop game," in *Proceedings of the 13th International Conference on Advances in Information Technology*, 2023, pp. 1–6.

[21] D. Valevski, Y. Leviathan, M. Arar, and S. Fruchter, "Diffusion models are real-time game engines," *arXiv preprint arXiv:2408.14837*, 2024.

[22] Y. Sun, Z. Li, K. Fang, C. H. Lee, and A. Asadipour, "Language as reality: A co-creative storytelling game experience in 1001 nights using generative ai," in *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 19, 2023, pp. 425–434.

[23] *Infinite Craft*, en. [Online]. Available: https://neal.fun/infinite-craft/ (visited on 09/16/2024).

[24] L. Anderson, "Analytic autoethnography," *Journal of contemporary ethnography*, vol. 35, no. 4, pp. 373–395, 2006.

[25] S. Pace, "Writing the self into research: Using grounded theory analytic strategies in autoethnography," 2012.

[26] P. Hodkinson, "Grounded theory and inductive research," in *Researching social life*, 5, Sage Publications Ltd, 2008, pp. 80–100.

[27] *Ironsworn - Tabletop Roleplaying Games*. [Online]. Available: https://www.ironswornrpg.com/ (visited on 09/16/2024).

[28] *Apocalypse World*. [Online]. Available: http://apocalypse-world.com/ (visited on 09/16/2024).

[29] *Hard Design and Soft Design in Roleplaying Games – Juhana Pettersson*, en-GB, Dec. 2019. [Online]. Available: https://www.juhanapettersson.com/hard-design-and-soft-design-in-roleplaying-games/ (visited on 02/27/2024).