Owen Scruggs

February 18, 2025

Prof. Macalintal

ENC1102

Annotated Bibliography- Conventions in RPG Maker VX Ace Scripting

Owen Scruggs

Cover Letter

As a brief introduction to this topic, RPG Maker VX Ace (RPGMA) is a video game engine marketed by Kadokawa Corporation used by new and experienced developers alike. RPGMA's scripting system enables more experienced developers to make advanced systems in Ruby Script, and for newer developers to learn Ruby through these core scripts or third-party scripts by more experienced developers.

The purpose of this annotated bibliography is to analyze and synthesize existing research on RPG Maker VX Ace scripting conventions to determine an ideal script structure as would be considered by a scripter. Through this process of organizing and evaluating scholarly sources, this work sets the foundation for my original research, positioning my study within a larger conversation of game development, digital literacy, and programming conventions. This annotated bibliography is designed to be accessible and informative, providing a clear synthesis of relevant studies while demonstrating how my research fills an existing gap. By concluding with specific research questions and outlining my methodological approach, this document ensures that my study is firmly grounded in academic discourse and positioned to contribute new insights to the field.

The process of developing this annotated bibliography involved selecting, analyzing, and synthesizing secondary research to create a cohesive academic conversation. I began by identifying sources that explored game engine limitations and community-driven software practices. As I worked through each source, I carefully evaluated their methodologies, credibility, and relevance to my research, making connections between studies where possible. Organizing the annotations made me carefully balance summary and analysis, where every entry must succinctly state the source's main contributions yet justify its relevance to my research. The hardest part was identifying the research gap, where different sources made references to scripting structures briefly, not directly addressing my research question. Nonetheless, by consolidating the results from several studies, I was able to identify the niche discussion regarding ideal script layouts for RPG Maker VX Ace.

Through consolidating and analyzing these sources, I have advanced key learning outcomes by generating genuine lines of inquiry and integrating diverse scholarly perspectives into a coherent research framework. This annotated bibliography not only demonstrates my ability to critically evaluate secondary sources based on criteria of relevance, credibility, and ethics—thereby strengthening my information literacy—but also showcases my skills in producing multimodal academic writing that navigates the constraints and possibilities of various research genres. In doing so, I have positioned my work to contribute new knowledge by drawing meaningful connections between primary evidence and established scholarship, and I have honed my ability to refine and revise my writing in response to constructive feedback, ensuring that my research remains dynamic and engaged with ongoing disciplinary conversations.

Introduction

Establishing the Research Territory

The field of game development is increasingly shaped by accessible game engines like RPG Maker VX Ace, which enable users to create games without requiring extensive programming knowledge. However, while these tools provide a structured scripting system, the conventions that define an ideal RPG Maker script layout remain largely community-driven rather than formally standardized. Researchers have examined how game-making tools facilitate creativity and learning (Fiadotau, 2019; Owens, 2011) and how developers engage in scripting practices that challenge or refine engine limitations (Fiadotau, 2016). Other studies focus on technical implementations of scripting (Schatten et al., 2020) or the role of code as a rhetorical medium (Brock, 2020). Despite these contributions, no research has yet established a universally accepted or ideal RPG Maker VX Ace script structure, leaving an open question as to what scripters prioritize when writing or evaluating scripts.

Reviewing the Existing Conversation

Fiadotau (2016) examines how developers challenge game engine limitations, arguing that pushing against constraints functions as metacommentary on game design and scripting conventions. Similarly, Owens (2011) explores how RPG Maker forums serve as learning spaces, showing that scripting conventions emerge organically through collaboration rather than formal instruction. These studies highlight how scripting evolves within communities but do not explicitly define which scripting elements are essential or ideal.

Other researchers examine scripting from a technical perspective. Schatten et al. (2020) analyze how APIs can extend game engine functionality, showing that modularity is key in scalable game development. Their findings suggest that scripting in RPG Maker could benefit from increased modularity, but the study does not assess how scripters themselves view script dependencies and self-containment. Meanwhile, Brock (2020) explores code as a rhetorical artifact, demonstrating that programming structures encode meaning and usability expectations. This perspective suggests that scripting layouts may not just

serve functional purposes but also communicate clarity, intent, and structure to future users.

Additionally, some studies demonstrate how RPG Maker's scripting system is adapted beyond traditional game-making. Clarke et al. (2018) discuss how RPG Maker can be repurposed for educational gamification, offering insight into how scripting layouts affect usability and learning. Their work suggests that clear structure and documentation are essential for usability, aligning with questions of whether scripting layouts should prioritize readability over compactness.

Identifying the Research Gap

While existing studies explore RPG Maker scripting from multiple angles, none specifically address what RPG Maker VX Ace scripters themselves consider essential, modular, or ideal in a script's structure. Prior research highlights community-driven development (Owens, 2011), constraints and subversion (Fiadotau, 2016), and modular scripting principles (Schatten et al., 2020), but no study synthesizes these perspectives to determine which scripting conventions scripters prefer and why. Given that RPG Maker relies heavily on third-party scripting for customization, understanding what elements define a well-structured script is crucial to both improving scripting standards and enhancing game engine usability.

Research Questions and Methods

My study seeks to answer the following research questions:

1. What features do RPG Maker VX Ace scripters believe to be part of an ideal script structure?
2. How do scripters evaluate a script's clarity, modularity, and usability?

To address these questions, I will conduct interviews with 3-5 RPG Maker VX Ace scripters of varying experience levels, ensuring perspectives from both novice and advanced users. Interviewees will evaluate 3-5 sample scripts, including intentionally flawed or incomplete scripts, and their feedback will be analyzed for common preferences and critiques. Responses will be coded on a scale from least to most detailed, assessing what script elements they find essential, nonessential, or missing. By synthesizing scripter perspectives with existing research, this study will contribute to establishing a clearer understanding

of scripting conventions within RPG Maker VX Ace. The studies and secondary research I've analyzed are as follows below:

Bibliography

Brock, Kevin.

"Rhetorical Code Studies: Discovering Arguments in and around Code." University of Michigan Press, 2020.

Brock examines code as a rhetorical artifact, arguing that software development involves implicit persuasion and meaning-making beyond its functional purpose. His work is positioned within critical code studies and technical communication, blending software studies with rhetorical analysis to explore how developers encode meaning into software structures. The book introduces key concepts from digital rhetoric, drawing from scholars like Hayles (2012) on technogenesis, Gillespie (2014) on algorithmic culture, and Lanham (2003) on digital persuasion. Brock employs rhetorical analysis of software projects, using case studies of Mozilla Firefox, OpenSSL, and Heartbleed vulnerabilities to illustrate how code can persuade, justify, or obfuscate certain decisions. The study emphasizes that programming languages are not neutral tools but instead contain inherent assumptions, biases, and intended user behaviors.

This book is highly relevant to my research because it explores how script structures convey meaning, offering a framework for analyzing how RPG Maker VX Ace scripts communicate function, usability, and modification potential. Brock's scholarly background in digital rhetoric ensures credibility, and his case study approach allows comparisons between different programming communities and how they construct readability norms. Compared to Fiadotau's studies, which focus on community-driven scripting evolution, Brock's work provides a theoretical perspective on why scripts are structured the way they are. His analysis of code persuasion complements Owens' discussion of forum-based learning, as both explore how developers interpret and negotiate meaning within digital tools.

Clarke, Samantha, et al.

"Gamifying the University Library: Using RPG Maker to Re-Design Library Induction and Online Services." Proceedings of the European Conference on e-Learning, 2018, pp. 721-725, Coventry

University.

Clarke and her co-authors examine how game-based learning, specifically through RPG Maker, can improve university library induction by making the experience more engaging and accessible. The authors address a common issue: low participation in traditional library inductions, particularly among distance and online learners. To address this, they propose a gamified alternative, designing an interactive RPG Maker experience that replicates the Coventry University Library space and guides students through key learning objectives, such as finding study areas and accessing digital resources. The study is positioned within educational gamification research, which examines how game mechanics and principles can improve learning outcomes. Clarke et al. cite scholars like Garris, Ahlers, and Driskell (2002) on game-based motivation and Arnab et al. (2013) on serious games in education. Using design-based research methodology, the authors develop a proof-of-concept RPG Maker game and analyze its effectiveness through student feedback and mixed-methods research. Their findings suggest that gamification enhances engagement and information retention while potentially reducing staff workload for university libraries.

This article is highly relevant to my research as it demonstrates how RPG Maker's scripting and event system can be adapted for non-traditional uses, offering insights into how modularity, clarity, and usability affect script effectiveness. While the focus is on educational applications rather than game development, the study provides a valuable case study on RPG Maker's flexibility and how structured scripting layouts contribute to usability. Compared to other sources, this study is more applied than theoretical, serving as a practical example of script structuring in a real-world, non-gaming context. It complements Fiadotau's work by showing how standard script conventions in RPG Maker can be repurposed for interactive storytelling and instructional design.

Fiadotau, Mikhail.

"Dezaemon, RPG Maker, NScripter: Exploring and Classifying Game 'Produsage' in 1990s Japan."

Journal of Gaming & Virtual Worlds, vol. 11, no. 3, 2019, pp. 215–230. Intellect Ltd.,

https://doi.org/10.1386/jgvw.11.3.215_1.

Fiadotau explores the concept of produsage, a blend of "production" and "usage" where players create their own content using game development tools such as RPG Maker, Dezaemon, and NScripter. The study focuses on 1990s Japan, a period before the modern indie game movement, and examines how these tools enabled hobbyist game development. Fiadotau argues that game-making software has historically balanced creative freedom with corporate-imposed restrictions, influencing how and what users could create. He employs a qualitative case study approach, analyzing technical features, licensing agreements, and user communities for each tool. He develops a two-axis framework, measuring game engines on a scale of expressive freedom (how much control users have over the game's content) and distribution control (how freely they can share their creations). The study highlights that RPG Maker offered a high level of customization but was limited in distribution due to licensing restrictions, which affected how developers shared their games.

This article is relevant to my research because it examines how scripting conventions evolved in RPG Maker from a historical perspective, providing insight into how early design decisions shaped the current structure of RPG Maker VX Ace scripts. Fiadotau's background in digital media and gaming culture lends credibility, and the peer-reviewed publication ensures scholarly rigor. Compared to his *Game Engine Conventions* study, which focuses on subversion of game engines, this article focuses more on historical development and community-driven adaptations. It complements sources like Owens' study on RPG Maker forums, as both discuss community-driven script evolution but from different angles—historical vs. contemporary.

Fiadotau, Mikhail.

"Game Engine Conventions and Games That Challenge Them: Subverting Conventions as Metacommentary." Replay 1, vol. 3, 2016, pp. 47-65. Tallinn University. http://dx.doi.org/10.18778/2391-8551.03.03.

Fiadotau explores how developers subvert game engine conventions to make a metacommentary—a critique or reflection on the design limitations and expectations of game-making tools like RPG Maker. The study examines how creative subversion challenges the notion of fixed software functionality, drawing from participatory culture and platform studies. Using autoethnography, a research method in which the author draws from personal experience as a developer, Fiadotau investigates how users push RPG Maker beyond its intended design. He highlights case studies of experimental game projects that deliberately break or repurpose RPG Maker's constraints, showing how users influence engine evolution and community standards. The article argues that game engines both enable and limit creativity, and that breaking these limitations fosters new forms of gameplay and game-making discourse.

This article is highly relevant to my research because it provides a theoretical framework for understanding why scripting conventions evolve, rather than just how they are used. Fiadotau's expertise in digital culture and game production lends credibility to his claims, and the peer-reviewed nature of the article ensures scholarly rigor. Compared to other sources, this study is distinct from Owens' analysis of RPG Maker forums, as it focuses on intentional acts of subversion rather than community-driven learning. It also complements Brock's rhetorical analysis of code, reinforcing the idea that scripting choices are not just functional, but also rhetorical and ideological. This study will help me assess whether certain scripting structures emerge from necessity or as a response to RPG Maker's built-in constraints.

Owens, Trevor.

"Social Videogame Creation: Lessons from RPG Maker." On the Horizon, vol. 19, no. 1, 2011, pp. 52-61.

Emerald Group Publishing.

Owens examines how RPG Maker communities function as digital learning environments, where users develop technical and creative skills through game creation. The study focuses on how web forums serve as informal learning spaces, with scripters, artists, and designers self-organizing into roles based on expertise. Owens positions the research within digital literacy and participatory culture, citing scholars like

Gee (2004, 2009) on situated learning and Ito et al. (2010) on youth engagement with digital media. Using a qualitative research approach, Owens conducts surveys, interviews, and discourse analysis of RPG Maker forums, collecting data from 80 participants to understand how they learn and apply scripting knowledge. The study concludes that forums provide structured yet flexible spaces for knowledge exchange, suggesting that educators should design learning platforms modeled after these organic digital communities.

This article is valuable to my research because it explores how scripters develop and refine their skills through community engagement, a key factor in determining script layout conventions. Owens' position as an Information Technology Specialist at the Library of Congress and the study's peer-reviewed nature adds credibility, ensuring that his analysis of digital communities is well-informed. Compared to Fiadotau's historical perspective, Owens provides a contemporary look at RPG Maker scripting as a literacy practice, making this source particularly useful for examining how best practices emerge from communal feedback. It complements sources like Clarke's study on gamification, as both explore non-traditional uses of RPG Maker in structured learning environments.

Schatten, Markus, Igor Tomičić, and Bogdan Okreša Đurić.

"Towards Application Programming Interfaces for Cloud Services Orchestration Platforms in Computer Games." Proceedings of the Central European Conference on Information and Intelligent Systems, Oct. 2020, pp. 9-14, Faculty of Organization and Informatics, University of Zagreb.

This article explores how cloud services orchestration can improve game server architectures, particularly in games that rely on real-time processing and large-scale multiplayer interactions. The authors address a gap in game development research by investigating how microservices orchestration—a system in which software components communicate via lightweight APIs—can be applied to game engines, including RPG Maker. Schatten et al. discuss the transition from monolithic game architecture to modular, cloud-based solutions, citing Fowler and Lewis (2014) on microservices, Khan (2017) on container orchestration, and Walker (2020) on game server scalability. Their methodology includes technical implementation and experimentation, developing four proof-of-concept API integrations for Godot, RPG

Maker, Ren'Py, and Blender Game Engine. The authors assess each engine's networking capabilities and modular structure, concluding that most game engines can integrate cloud-based services with proper API design.

This study is useful to my research because it highlights how RPG Maker scripts can be structured for external system integration, contributing to the discussion of modularity and script dependencies. The authors' affiliation with the University of Zagreb's Artificial Intelligence Laboratory ensures expertise in distributed computing and game engine optimization, adding credibility to their findings. Compared to sources like Brock's rhetorical analysis of code, this article provides a technical perspective on scripting as a modular and scalable system. It complements Fiadotau's discussion of engine constraints by offering a forward-looking perspective on how scripting limitations can be bypassed through external integrations.