Nostalgic Development Practices: Fantasy Consoles and DIY Game Engines

Logan Davis
Rensselaer Polytechnic Institute
110 8th Street
Troy, NY 12180
davisl6@rpi.edu

Keywords

Game engines, game development, interfaces, fantasy consoles, remediation, nostalgiaĺ

ABSTRACT (EXTENDED):

When *Animal Well* (Shared Memory, 2024) released to critical and commercial acclaim, Billy Basso – the sole developer of the game – did rounds of interviews with gaming journal outlets and YouTube personalities as is common with many indie games released today. However, many of these interviews posed a relatively uncommon question regarding *Animal Well's* production: why did Basso choose to develop a game engine from scratch for the project? Sam Machkovech, writer for the site Game Developer, went as far as to assert that the Basso's homebrew engine was "key to [the game's] success" (Machkovech, 2024). Basso represents one of a growing number of developers who synonymize their game with the act of creating bespoke tools and technologies as some essential aspect of the game's importance, like the developers of *Factorio* (Wube Software, 2020), *Teardown* (Tuxedo Labs, 2020), and *Sokoban* (Thekla Inc, unreleased). Developers use their custom engines as evidence of their technical acumen, intentional design, and proof of the uniqueness of their work.

In this paper I analyze the rising movement of game developers rejecting readymade game engines like Unity or Unreal by developing custom game engines using a mixture of archival and textual analysis as well as critical making. Drawing from scholarly work on digital sound production interfaces, I highlight how this practice of low-level "terminal aesthetics" (D'Errico, 2022) in game development is used as a communicative strategy. These developers play off of the masculine practice of exchanging technical knowledge for social acumen among fetishistic gaming discourse endemic to gear and tool culture (Bates and Bennett, 2025). Developers cite the "grain" (Nicoll and Keogh, 2019) of readymade engines as a limiter of creativity in which one is often fighting to get their game to work (Machkovech, 2024) yet simultaneously deriding users of engines as "developing in lala land" (Basso, 2024).

Proceedings of DiGRA 2026

Part of this strategy relies on emphasizing low level programming concerns like memory management and "asset packing" (defining how textures, sounds, and meshes are stored in a video game's executable) as some kind of deep and critical knowledge to interface with computers.

Video games within this community frequently use retro-nostalgic design and graphics as a "meta" about videogames themselves (Boluk and LeMieux, 2017) by critiquing, extending, revisiting, and reimagining the older games through the technological affordances of today. By remediating the immediacy of retro-nostalgic games while emphasizing the technical and mediated realities of the games we play, these developers attempt to make their games more real (Bolter and Grusin, 2000) than those of games that target "some imaginary system" (Basso, 2024). Imagining a system is not the same as using the defaults and tools of readymade software. In fact, a separate community of game developers imagine machines to cultivate the same lo-fi, remediative and minimalistic practices lauded by DIY engine developers.

The PICO-8, released by Lexaloffle Games, is a software based "fantasy console" that gives hobbyist and professional game developers a chunky, pixelated development environment for creating 2D games with limited color palettes and 4-channel audio for a 128x128 resolution display. The requirement to either host the game on a website or use a paid-for application to play games made on the system makes it difficult for developers to monetize their creations. However, games such as *Celeste* (Maddy Makes Games, 2018) which was originally designed on the PICO-8 before being rewritten for PCs and consoles shows that games developed by the fantasy console community can still be of mainstream appeal. By using the high-level scripting language Lua (which automates many of the lower-level duties of programmers like memory management), providing a complete set of development tools, and utilizing BBS forums for user communication, the PICO-8 cultivates a culture of anti-corporatized game development (Berge, 2025). Calling on retro interfaces, social forums, and systems constraints, fantasy consoles "remediate prior forms of play" and development as toys of production (Malazita, 2024). By moving the focus away from games and onto the making process, tools like the PICO-8 "encourage this kind of exploration through its own self-reflexive and self-effacing commentary, as well as its deconstruction and Frankenstein-like assemblage of games of the past" (Ivănescu, 2019). In other words: game developer tools like the PICO-8 make a game out of game development by using the constraints of retro game development as the rules of a game itself but makes that game accessible by decoupling itself from a real machine in interest of explorative modern technologies.

On both ends of the spectrum from accessible hacker fantasy consoles to handcrafted modern minimalisms, both communities of developers center a knowledge and intimacy with the machinery of computers as part of playing and making games. DIY developers share the voyeuristic details of a machine's inner workings as a testament to its intentionality; fantasy consoles make a game out of making games through a set of constraining-yet-explorative tools. While both react to the corporate nature of proprietary engines, the politics of these expressions are ambivalent: one covets exclusionary competence while the other creates a system of explorative play.

Bibliography:

Basso, Billy. 2024. “Billy Basso on Handmade Software (ANIMAL WELL).” Abner Coimbre. YouTube video, 12:30. URL: https://www.youtube.com/watch?v=KixgF8Ru1Mw

Bates, Eliot, and Samantha Bennett. 2025. *Gear: Cultures of Audio and Music Technologies*. The MIT Press.

Berge, PB. 2025. “Anti-Games, Fantasy Consoles, and the Rise of Speculative Game Development on Itch.Io.” *Proceedings of the 20th International Conference on the Foundations of Digital Games* (New York, NY, USA), FDG ’25, May 9, 1–12. https://doi.org/10.1145/3723498.3723739.

Bolter, J. David, and Richard Arthur Grusin. 2000. *Remediation: Understanding New Media*. MIT Press.

Boluk, Stephanie, and Patrick LeMieux. 2017. *Metagaming: Playing, Competing, Spectating, Cheating, Trading, Making, and Breaking Videogames*. Electronic Mediations 53. University of Minnesota Press.

D’Errico, Mike. 2022. *Push: Software Design and the Cultural Politics of Music Production*. 1. Oxford University Press.

Gustafsson, Dennis interviewed by Joel Couture. July, 2021. “Combining bombastic heists with a fully destructible voxel world in *Teardown.”* Game Developer (formerly Gamasutra). Accessed October, 2025. URL: https://www.gamedeveloper.com/design/combining-bombastic-heists-with-a-fully-destructible-voxel-world-in-i-teardown-i-

Ivănescu, Andra. 2019. *Popular Music in the Nostalgia Video Game: The Way It Never Sounded*. Palgrave Studies in Audio-Visual Culture. Springer International Publishing. https://doi.org/10.1007/978-3-030-04281-3.

Machkovech, Sam. November, 2024. “Why Animal Well’s home-brewed engine was key to its success.” Game Developer (formerly Gamasutra). Accessed October, 2025. URL: https://www.gamedeveloper.com/design/why-animal-well-s-home-brewed-engine-was-key-to-its-success

Maddy Makes Games. 2018. *Celeste.* PC. Maddy Makes Games

Malazita, James. 2024. *Enacting Platforms: Feminist Technoscience and the Unreal Engine*. Platform Studies. The MIT Press.

Nicoll, Benjamin, and Brendan Keogh. 2019. *The Unity Game Engine and the Circuits of Cultural Software*. Springer International Publishing. https://doi.org/10.1007/978-3-030-25012-6.

Shared Memory. 2024. *Animal Well.* PC. Bigmode.

Thekla, Inc. Unreleased. *Sokoban (Working Title).* PC.

Tuxedo Labs. 2022. *Teardown.* PC. Tuxedo Labs and Saber Interactive

Wube Software. 2020. *Factorio.* PC. Wube Software.