Game engines: Design, labour, and legality

Benjamin Nicoll
The University of Melbourne
Melbourne, Parkville 3010
bnicoll@unimelb.edu.au

Keywords
Game engines, Unity, Unreal, game design, game design education, intellectual property, indie, Australia

ABSTRACT
Game engines are code frameworks, software toolsets, and proprietary structures that enable videogame content to be produced and published on a variety of platforms. They tend to manage low-level computational routines such as rendering, physics, and artificial intelligence, thereby allowing game designers, artists, and programmers to streamline their development practices. It used to be that videogame companies would develop their own proprietary engines to optimize in-house development practices. Occasionally, companies would also license their engines to other companies, or make their toolsets freely available to modding communities. Since the mid-2000s, however, a small handful of third-party engines such as Unreal and Unity have come to monopolise videogame production across both professional and amateur contexts. These third-party engines tend to work on subscription-based models or, in the case of Unity, free-to-use (and rather opaque) platform-based models designed to monopolise network effects (see Srnicek, 2016).

This paper offers insight into the state of game engines today – that is, how game engines are being used and implemented in a variety of institutional settings and design practices – by drawing on interviews with Australian game designers (both professional and amateur) as well as tertiary game design students and educators. Interviews for this project are currently underway, though we have so far conducted 15 of approximately 17 planned interviews. The interviews are semi-structured, and typically last between 45—60 minutes. Participants are asked a range of questions about their thoughts and opinions on game engines, such as how differently engines influence the way they approach, learn about, or teach the process of game design. The interviews are then transcribed and thematically analysed through a process that involves coding the data based on common themes, responses, and concerns. We chose to interview Australian participants only so as to make geographically specific claims about the use of game engines in Australian contexts.

The project will contribute to an existing (though limited) body of research that has, for the most part, focused on the history of game engines. For example, Graeme Kirkpatrick (2013: 104) argues that game engines both streamline and standardise the craft of game

Extended Abstract Presented at DiGRA 2018

design. According to Kirkpatrick, game engines function to 'contain' potentially subversive or commercially undesirable uses of computer technologies. David Nieborg and Shenja van der Graaf (2008) make a similar observation, suggesting that open-source engines enable game corporations to co-opt the 'unofficial' practices of modders and hackers. Similarly, Ian Bogost (2006: 62) argues that game engines are essentially intellectual property, in that they bind certain games and genres to specific engines, both legally (as proprietary extensions of existing games) and materially (as software toolsets). John Banks (2013) provides perhaps the most comprehensive overview of game engines in his study of the SAGE engine, a short-lived engine developed by an Australian game studio in the mid-2000s. Describing SAGE as a "multiple object," he argues that game engines are "participants… in the making of co-creativity" (Banks, 2013: 53).

What is lacking, and what this paper aims to develop, is a more up-to-date perspective on the state of game engines today, especially in light of the recent widespread adoption of Unity as the 'engine of choice' for many game designers, students, and educators. In particular, we seek to answer the following key questions:

- What are the technical, proprietary, and economic functions of game engines; and how are they reshaping the industry and craft of game design?
- What are the limitations and opportunities for people from non-programming backgrounds (e.g. artists, designers, and hobbyists) to access and utilise game engines?
- How do game engines foster creative innovation in the Australian videogame industry?
- How could they be used to more effectively harness creativity and innovation, especially for non-videogame applications (e.g. VR software development)?

To this end, we focus on game engines through a three-pronged approach: as *technologies* that are reconfiguring the landscape of game design; as *intermediary platforms* that bring together different industry groups and imply certain labour practices; and as *economic* and *proprietary entities* operating in a wider platform ecosystem.

Our preliminary analysis of the interview data reveals several key themes: firstly, that game engines are incredibly complex objects that possess different meanings and functions depending on how they are used and who they are used by. Several interview participants seem to view engines less as contained toolsets and more as networks that intermediate between different groups of people. This is especially the case when it comes to Unity's 'asset store' – which allows the user to develop, sell, and purchase assets and plug-ins from other users – as well as its support network of online forums. Secondly, interview participants tend to agree that third-party engines such as Unity shape their development practices in subtle – and often quite imperceptible – ways. By presenting users with 'component-based' design interfaces, third-party engines such as Unity remove the need for deep, object-oriented programming. Interview participants view this as at once both liberating and stifling, to the extent that it speeds up and simplifies the process of game design, but also limits the amount of control and customization available to the user. Similarly, students and educators tend to describe engines as though they are technologies that equally 'participate' in the education process by shaping tastes, preferences, and development practices. Thirdly, game designers – and especially teams of designers – describe needing to spend time working out what an engine 'wants from them' in order to work with it effectively. Once again, this tends to be viewed as a fair trade off, as the alternative is to spend months or perhaps even years

developing in-house toolsets. The majority of interview participants also express significant confusion regarding the legal and proprietary functions of Unity in particular, which speaks to the platform's rapid growth and monopolisation of the market. Unity is constantly updating not only its terms of service but also its interface, which interview participants describe as a constant process of needing to play 'catch-up.' The analysis of the data is still underway, however, and will no doubt reveal further insights as the research progresses.

BIOGRAPHY

Benjamin Nicoll is currently working as a research associate and sessional lecturer at the University of Melbourne, Australia, where he also received his PhD. His research focuses mainly on the history and critical theory of videogames and videogame platforms, with a particular focus on notions of technological failure and marginality in game history. He is currently working on a project examining the use and implementation of 'game engines' in amateur and professional game design contexts.

ACKNOWLEDGMENTS

This research for this abstract is supported by a grant jointly funded by the University of Melbourne's Networked Society Institute (NSI), Intellectual Property Research Institute (IPRIA, and the Centre for Media and Communications Law (CMCL). The project website can be found here: https://networkedsociety.unimelb.edu.au/research/projects/active/game-engine-videogame-industry

BIBLIOGRAPHY

Banks J (2013) *Co-creating Videogames*. London: Bloomsbury.

Bogost I (2006) *Unit Operations: An Approach to Videogame Criticism*. Cambridge. MA: MIT Press.

Kirkpatrick G (2013) *Computer Games and the Social Imaginary*. Cambridge, MA: Polity.

Nieborg D and van der Graaf S (2008) The mod industries? The industrial logic of -non-market game production. *European Journal of cultural Studies* 11(2): 177-95.

Srnicek N (2016) *Platform Capitalism*. Malden, MA: Polity.