Summary Review: Cowboys, Ankle Sprains, and Keepers of Quality: How Is Video Game Development Different from Software Development?

Paul Wooseok Lee
University of Waterloo
Waterloo, Canada
w69lee@uwaterloo.ca

1 What is the Problem Being Solved?

Despite video games being a significant and growing sector in software engineering (at least in the year 2014, when the paper was published), there has been little empirical study on how game development differs from software engineering.

2 What is the New Idea they are proposing?

The paper proposes that video game development is distinct from traditional software engineering in several ways. It presents empirical evidence from interviews with 14 developers and a survey of 364 respondents, highlighting key differences. Some of the major differences include:

(1) Cowboy Coders - Unlike traditional software engineering, game development demands programmers who can swiftly translate creative concepts into functional code without strictly following structured methodologies. This is because game design is highly iterative, requiring rapid prototyping, and frequent changes.
(2) Software Testing - Automated testing is much less common in game development than traditional software, mainly because tests quickly become obsolete due to evolving design goals. Additionally, testing game mechanics and player experience is more subjective and difficult to automate.
(3) Creativity vs Technology - Game development is not purely an engineering discipline but a mix of creativity and technology. Game designers often modify requirements based on player experience, making formal requirements less stable than in other software fields.
(4) Management Practice - Game studios often have non-technical managers, making it difficult to convey engineering challenges effectively. Additionally, game teams are highly interdisciplinary, requiring developers to collaborate closely with artists, designers, and sound engineers.
(5) Code Reuse – Game developers uncommonly reuse code between projects, as each game requires unique mechanics and performance optimizations.

3 Class Discussion

During the class discussion, several passionate opinions were shared about the paper. One student expressed a particularly negative stance, rating the paper really low, and even suggesting it deserved a 0. The criticism stemmed from a belief that the authors lacked a deep understanding of how games or game development work. Some key points discussed include:

- The paper was written around the time mobile gaming and microtransactions were booming. One felt it did not adequately capture these industry changes.
- Poor planning in game development was highlighted, referencing Diablo's gold and auction house system as an example of mismanagement.
- The role of game testers was mentioned. Although they may not always have computer science degrees or any technical education at all, their skill and expertise in identifying game-breaking issues were considered extremely important, but not taken seriously in the industry.
- The comparison between companies embracing "patch culture" versus those taking a more meticulous approach to development, such as Supergiant Games with Hades, was discussed. The latter approach, focused on delivering a polished product from the start, was seen as a more sustainable model.
- The gap between developers and designers was explored. Designers focus on story and fun, but developers are often constrained by what is technically possible. This disconnect can lead to tension in development teams.
- The influence of managers was a recurring theme. Non-technical managers enforcing strict schedules without understanding the complexity of game development can lead to poor-quality products and low team morale.
- Finally, there was a discussion on whether the culture of game development has worsened since the publication of the article (2014). Some argued that work-from-home opportunities have increased, but overall, large developers seem to prioritize maintaining the status quo over innovation. However, indie game development remains a space where true innovation is happening, often driven by developers who previously worked in large companies.

4 Positives and Negatives

4.1 Positives

- I appreciated that the study didn't just focus on engineers but also included testers. They provided valuable insights into the differences and the challenges of game development compared to traditional software engineering.
- Even with a relatively small number of interviewees, the paper presented meaningful observations.

4.2 Negatives

- The paper focuses so heavily on how game development differs from traditional software engineering that it almost presents them as entirely separate fields. A more balanced discussion of similarities could have helped show where game developers could still benefit from general software engineering practices.
- The study mainly sampled developers from Microsoft, missing out on a broader industry perspective. It would have been more insightful to include developers from various companies, and game genres (e.g., single player, MMORPGs, indie games) to provide a more comprehensive view of game development practices.

5 Future Work

- The paper mentioned how game studios develop in-house tools for their needs. Future work could explore how these tools are developed and maintained, and whether standardization could benefit the industry.
- Research that focuses on improving code reuse in game development without sacrificing performance.
- Investigating sustainable development practices for healthier work environments.
- Enhancing collaboration between Engineers & Creatives in game development.
  - Understand the challenges and dynamics between the different disciplines.
  - Develop and test a structured framework or methodology to improve collaboration.

6 Rating (out of 5)

I would rate this paper out of 4.5. While it provides valuable insights into the differences between traditional software development and game development, it focuses so much on these differences that it almost treats them as entirely separate fields. A more balanced discussion of similarities, along with a broader range of interviewees from different companies and game genres, would have made the study even stronger.

7 Discussions

- What specialists can you think of that are required to make games? To elaborate on their Diablo III example, many games do require specialists not typically found in traditional software teams due to the game's specific domain. A great example is Sid Meier's Civilization series, which relies on accurate historical facts.
- When I was an undergraduate Computer Science student, there was a common stereotype that a certain (huge) game company exploited passionate gamers by hiring them at low wages and overworking them, knowing that their enthusiasm for games would keep them from resisting long hours and unpaid overtime. Is this true? Did you hear about this?
- Any potential methodology for collaboration between Creatives & Engineers?