


---

# Continuous Delivery of Games

Abhishek Tiwari 

Citation: *A. Tiwari*, "Continuous Delivery of Games", Abhishek Tiwari, 2021.  
[doi:10.59350/sjtfp-frr95](https://doi.org/10.59350/sjtfp-frr95)

Published on: January 08, 2021

What can we learn from Cyberpunk 2077's disastrous launch? Cyberpunk 2077 was extensively advertised yet highly buggy release. Sony has pulled Cyberpunk 2077 from its PlayStation store following widespread complaints about glitches. The game lost over 75% of players on Steam. The stock value of CD Projekt Red (CDPR) - Developer of Cyberpunk 2077 - is down by 50%. CDPR board and management are now defensive as they now face hostile staff (i.e. overworked software engineers) after the failed launch. One key learning for me is the lack of **continuous delivery** and modern software engineering practices in the gaming industry.

Traditional games are monolithic applications and traditionally developed as a boxed-product (pay-to-play). To deliver boxed-products gaming companies follow a waterfall project delivery model. New game projects typically run over 2-4 years and during this window, the underlying gaming console platform is always evolving. Besides, engineers lack ownership. Projects are mostly driven by game designers and marketing. Game designers continuously coming with new ideas so the scope is always evolving but the deadline is fixed. To make it worse teams don't use automated testing which means companies have to deploy a large number of manual testers mostly at the very end of the release cycle. Results delayed and buggy game launches, overworked software engineers, and disappointed gamers. Once games are released, development team moves on to a new project.

I found a really good presentation on the web about how continuous delivery is used in AAA Games *Sea of Thieves* developed by Rare. Unlike other boxed or pay-to-play games, *Sea of Thieves* game is offered as game as a service (recurring subscription). This presentation provides a good overview of underlying engineering problems in the gaming industry. It also covers approach taken by Rare,

- Developer are responsible for quality
- Game is always shapable
- Each build contains small number of changes
- Trunk-based development
- Static and dynamic feature toggles
- Continuously improve cycle time
- Tackle technical debt
- Continuous improvements

{{< youtube 8hXRvvFDztg >}}