

# **Is it Viable to use Continuous Integration to Reduce Development Time for an RPG**

**COMP130 - Software Engineering**

1604281

2018-03-19

## **Abstract**

## **1 Introduction**

At its core, continuous integration is an Extreme Programming concept where developers commit their code to the repository multiple times a day where it is automatically built and tested. [1] Although the frequency of builds can vary, in an ideal situation, the code will be built whenever a commit is made. This helps to ensure that new code can be integrated successfully.

## **2 Benefits of Continuous Integration**

One of the largest benefits of continuous integration is its ability to reduce merge errors when integrating code into the repository. If developers wait a long time before making committing code, resulting issues can be complex and hard to solve, often wasting valuable development time while they are fixed. Many small commits each day reduces

the likelihood of merge errors, but also means that errors that do occur are easy to solve. [?]

Continuous integration also encourages and enhances communication between developers because of the quantity of commits that are made. More commits means that it is easier to see what changes have been made by other developers and provides the platform to query or make notes on individual changes via commit messages and comments.

Many developers find that the benefits of continuous integration allow them to release more often and have more successful pull requests. Both of these factors help reduce development time and streamline the project lifecycle. [2]

### **3 Disadvantages of Continuous Integration**

One of the immediate concerns regarding continuous integration is the initial setup time and the associated costs. [3] Dedicated computers to build the code and run automated tests will be needed in any large development project, as running automated tests can take a long time, making it impractical to attempt to use developer computers. Additionally, installing and configuring a continuous integration platform may take some time.

If the project includes hardware components as well as software, integration tests could require changes to the physical components to be successful, necessitating time and potentially money being spent to update the hardware. If this takes a longer than the continuous integration process, delays could be caused, as integration tests cannot be completed until the hardware is changed. [4]

### **4 In the Games Industry**

The applicability of continuous integration changes somewhat in regards to the games industry. Games projects are often composed of large, complex development teams

working on many different aspects of the game at once. Because of this, merge conflicts can be particularly disastrous and may delay multiple development teams if there are dependencies between departments. Because of this, continuous integration can be extremely useful for games development teams to try and avoid such a situation, while also providing good communication and transparency between individual development teams via multiple commits each day.

On the other hand, games can quickly become very large as they usually include assets such as 3D models, shaders and textures. This can make automated builds very slow which could cause delays in the continuous integration system. Possible solutions include only doing a full build of the game when it's necessary - such as once per day - and use a minimal build at other times.

## **5 The Nature of RPG's**

The term RPG - or Role Playing Game - refers to a broad range of games that are difficult to define due to the open-endedness of the term and the genre overlap that is common in video games. However, traditional RPG's often include key elements such as a structured storyline and rules to restrict the player to that plot. [5] Mechanically, a leveling system, inventory and the use of quests to drive the game are also common.

Most relevant to this paper is the observation that RPG's tend to emphasise depth, and as such, combine many different systems within the game as a whole. For example, combat mechanics, xp and level-up systems, a world map, inventory and quest system may all be required for a relatively simple RPG. This breadth of content can make continuous integration especially relevant to the genre, as it is imperative that the game systems are compatible.

## 6 Conclusion

In conclusion, despite the initial cost - in both time and money - of setting up continuous integration, it has the potential to significantly decrease development time during an RPG project. The ability to identify merge conflicts early on in development is invaluable when working on a large project with many different components that may be worked on by individual teams. Additionally, if dedicated computers are used, the automated build functionality of continuous integration can help speed up the project by removing the need for developers to build huge projects on their local machines, which would take large amounts of valuable time. [6]

Despite this, the setup required may not be worth it for very small teams working on small-scope projects. In these cases, the low number of developers makes communication simpler and reduces the chances of merge conflicts. A small project is also likely to be less complex, which reduces the need to ensure compatibility when new commits are made.

## References

- [1] M. Leppänen, S. Mäkinen, M. Pagels, V.-P. Eloranta, J. Itkonen, M. V. Mäntylä, and T. Männistö, “The highways and country roads to continuous deployment,” *Ieee software*, vol. 32, no. 2, pp. 64–72, 2015.
- [2] M. Hilton, T. Tunnell, K. Huang, D. Marinov, and D. Dig, “Usage, costs, and benefits of continuous integration in open-source projects,” in *Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering*. ACM, 2016, pp. 426–437.
- [3] C. Amrit and Y. Meijberg, “Effectiveness of test driven development and continuous integration—a case study,” *IT professional*, 2017.

- [4] R. Kasauli, G. Liebel, E. Knauss, S. Gopakumar, and B. Kanagwa, “Requirements engineering challenges in large-scale agile system development,” in *Requirements Engineering Conference (RE), 2017 IEEE 25th International*. IEEE, 2017, pp. 352–361.
- [5] A. Primanita, R. Effendi, and W. Hidayat, “Comparison of a and iterative deepening a algorithms for non-player character in role playing game,” in *Electrical Engineering and Computer Science (ICECOS), 2017 International Conference on*. IEEE, 2017, pp. 202–205.
- [6] L. Winder. Opinion: Is it green yet? improving our ci process. [Online]. Available: [https://www.gamasutra.com/view/news/125977/Opinion\\_Is\\_It\\_Green\\_Yet\\_Improving\\_Our\\_CI\\_Process.php](https://www.gamasutra.com/view/news/125977/Opinion_Is_It_Green_Yet_Improving_Our_CI_Process.php)