

**Researching Expert Management Strategies For Crunch Time**

Liam G. Houston

Michigan Technological University

HF 4015 Introduction to cognitive task analysis

Dr. Beth Veinott

Dec 13, 2023

### Abstract

Crunch time is a major issue within the gaming industry today. Despite its relevance to a major industry, it continues to grow industry wide. Using the cognitive analysis methods of a knowledge audit and CDM interviews, an insight into how experts within the field address the issue can be uncovered. What management strategies they employ to restructure project timelines and team focus can help us develop training to mitigate the commonality of crunch time for developers everywhere.

## Researching Expert Management Strategies For Crunch Time

Crunch time is a prevailing issue within industries that require a high level of software development. The gaming industry is particularly affected by this issue due to many different factors. Despite this issue being common amongst game developers, There are some project managers that manage to accommodate for crunch time, or eliminate it from their project timelines entirely. Through various cognitive analysis methods, their knowledge can be harnessed to help solve the issue of crunch time across the industry.

### **Background Context**

In order to design a research plan, it is important to first understand what crunch time is in the context of video game development. It is also prudent to understand how prevalent the issue of crunch time is, what its major effects are, and what causes it to occur.

### **What is crunch time**

The phenomenon of crunch time can be defined as a period of time within a project where a large amount of extra time and effort is required in order for the project to meet its deadline. In the gaming industry, crunch time manifests itself as a period right before the release of a game where everyone working on the project must work effectively non-stop so that the game is in a complete state to be released.

Crunch time has numerous negative effects on developers' health and motivation. Both because of the sleep deprivation experienced during crunch, and the depersonalizing and fatigue caused by heightened levels of stress. The quality and enthusiasm toward their work is heavily influenced by this as well (Mendes & Queiros, 2022).

The cause of crunch time can be attributed to many different factors; poor project management, constant new requirements, poor communication between developers and publisher, large project

scopes and bug fixes. In most cases, there are multiple factors that play into any one instance of crunch time (Jozin, 2020).

Most of the above mentioned causes of crunch time directly relate back to management. The constant addition of new requirements and poor publisher communication are outliers related to a disconnect between development teams and their corporate parents. The rest can all be addressed by introducing better management strategies and project planning that better account for a project's foreseen hangups.

### **Prevalence of the issue**

Despite crunch time being a documented issue for several years at this point, the issue continues to persist, and is currently getting worse industry wide. The International Game Developer Association (IGDA) conducts a developer satisfaction survey roughly every 2 years, in which they ask developers about the state of the industry from their perspective. The 2021 survey reported a large increase in developers experiencing crunch time compared to the results of the 2019 survey. In 2019 only 35% of total respondents reported experiencing crunch time. In 2021, this metric was split into three groups—full time employees, freelancers and self employed developers—whose crunch time experience averaged 61.6% (IGDA, 2021). Showing that the issue of crunch time is only getting worse for developers.

Additionally, a new distribution model termed servitization has begun to increase the workload of developers in a way that will make crunch time occur more often. Under the servitization model, games are no longer released as a complete product. Rather, titles are released in an incomplete state with a dense pipeline of future content promised to the customer upfront. The strategy behind this model is to treat games like a service for customers to subscribe to through their purchase (Dubois & Weststar, 2022).

When it comes to the issue of crunch time, servitization actively makes the problem worse. Developers still have last minute crunch time prior to the launch of the game, and have a heightened

potential for continued crunch as they complete a tight timeline of content that needs to be routinely produced over the game's predicted lifetime.

### **What has already been done**

Most solutions currently put forward have a few common similarities. Many solutions involve an attempt to fix the two most common project management strategies in programming; AGILE and SCRUM. Others suggest introducing new development strategies or team structure changes. The third common solution is to reject the corporate culture that is most often the reason crunch time has yet to be solved on a large scale.

While all of these approaches to solving the problem are valid, it may be more productive to find what is already working in the industry and harness it, rather than introducing and testing new systems. Such an approach would not serve to solve the issue completely, but it will better equip developers to deal with crunch time when it presents itself.

### **Methodology**

The task of eliminating crunch time entirely is a complex and daunting task. However, there are ways to aid in alleviating the issue in some scenarios. I've recognised project management as a major point of weakness and potential within this issue. Since the quality of project management is a major cause of crunch time, improving that management should help to mitigate it as a source of crunch time. This is why looking into the developers that do not report experiencing crunch time could yield useful findings.

I propose a research study that involves two cognitive analysis methods that will endeavor to uncover what it is that expert project managers within the gaming industry do differently, that allows them to compensate for or eliminate the need for crunch time in their projects. This study will look at what they notice that allows them to prepare for crunch time, how they manage project timelines to

prevent crunch time, and what they have learned about through their experience that allows them to prepare for crunch time more effectively.

### **Participants**

The participants for this study should be professionals within the gaming industry with at least 10 years of experience leading projects as either project manager or director. The sample size of these experts will be 15. This number will yield statistically significant results while also saturating the outcome with many perspectives on the issue. Each participant will be chosen based on their experience in the industry and on the reporting of their subordinates that they are effective at mitigating crunch time.

### **Cognitive analysis methods**

To uncover what these experts know, two cognitive analysis methods will be conducted. The use of cognitive analysis methods has been chosen over other approaches because of their aptitude toward finding pertinent information that is already working for experts within the field that the research aims to aid. The two methods are as follows; a knowledge audit and a critical decision making (CDM) interview.

**Knowledge audit.** A knowledge audit will be conducted to gain a greater insight into the game development profession. This method will also be used to probe for scenarios that will be used to format the critical decisions interview.

The knowledge audit will follow a structured outline that focuses on probing experts on their view of the situation and what they consider as they make decisions (see appendix A). This outline asks participants to reveal how they view the big picture around the situation, how they think and work smarter in their position, and if they have the experience and ability to quickly understand complex situations.

**Critical decision methods (CDM).** A CDM interview should be conducted with each participant in addition to the knowledge audit in order to gain additional insight into how they think in regards to their job. This method has been selected for its ability to uncover the cognition behind each expert's actions and its ability to ask participants to speculate on what could be improved or done differently in future situations.

The scenario focused on in this method will either be derived from the results of the knowledge audit or developed as a new step. The interview will primarily follow an outline with ten questions designed to drill for additional understanding on how participants assess situations and make decisions. There are also questions that present alternative scenarios for participants to consider (see appendix B). The guide is expected to serve as a base and jumping off point for these interviews, it is not expected that it is followed exactly.

#### **Analyzing results.**

In order to devise new training systems within the industry, the results of both the knowledge audit and the CDM interview will be subjected to a lengthy analysis. This analysis will focus on identifying common factors—similar strategies, planning methods and cues—shared between the participants.

After the knowledge audit is completed, the results will be organized into profiles corresponding with the participants. Each profile will contain information like a participant's name, job title, and the personalized scenario for their CDM interview derived from the knowledge audit results. The CDM interviews will then be conducted.

The results of the CDM interviews will also be organized into their respective participant's profile. Once complete, these profiles will serve to tag information from each method with their source participant and scenario. Which may be useful in later analysis.

The results from both methods will be analyzed to identify similar cues and choices made by the participants. Several passes will be made to ensure all possible considerations are made. The data will

then be reorganized to group common cues together, and analyze what reactions to those cues occur. This data will then be condensed into a series of graphs that show common issues, the cues that allude to them, and the most productive actions to combat them. Training strategies can then be devised based on what the best solutions to the most common and threatening issues are.

### **Timeline**

This research endeavor is expected to take 6-9 months to complete. The first month will be dedicated to finding and reaching out to potential participants and making any necessary alterations to the research plan. The following month is when the knowledge audits will be conducted and cataloged into participant profiles. Within the next 2-3 months, the CDM interviews will be conducted and cataloged. The remaining 2-4 months will be dedicated to analyzing results and formulating potential training strategies based on the findings.

### **Expected results**

The results from this research could lead to a greater understanding of what leads to crunch time within a project, and how it can be avoided. Specific strategies on how a project's scope and timeline can be better managed to avoid the need for crunch time is the ideal and intended outcome. Any strategies that are devised from this study would be based on the cognitive strategies that experienced project managers have actually used within the industry to achieve success. Such strategies would also have potential applications in other industries that experience crunch time as a problem. Although they may need to be altered depending on how specific to the gaming industry they end up being. If the issue of crunch time can be efficiently mitigated to just cases of extreme emergencies, then the lives and health of developers will be greatly improved, and the quality of the work they produce will definitely follow suit.



## References

*Developer satisfaction survey: Summary report [2021]*. IGDA. (n.d.).

<https://igda.org/resources-archive/developer-satisfaction-survey-summary-report-2021/>

Jozin, R. (2020). Crunch time in software development: A theory.

[https://elib.uni-stuttgart.de/bitstream/11682/11110/1/Master%20Thesis\\_Ruzica%20Jozin.pdf](https://elib.uni-stuttgart.de/bitstream/11682/11110/1/Master%20Thesis_Ruzica%20Jozin.pdf)

Mendes, J., Queirós, C. (2022). It's Crunch Time: Burnout, Job Demands and Job Resources in Game Developers. In: Barbedo, I., Barroso, B., Legerén, B., Roque, L., Sousa, J.P. (eds) Videogame Sciences and Arts. VJ 2020. Communications in Computer and Information Science, vol 1531. Springer, Cham. [https://doi-org.services.lib.mtu.edu/10.1007/978-3-030-95305-8\\_4](https://doi-org.services.lib.mtu.edu/10.1007/978-3-030-95305-8_4)

Musil, J., Schweda, A., Winkler, D., Biffel, S. (2010). Improving Video Game Development: Facilitating Heterogeneous Team Collaboration through Flexible Software Processes. In: Riel, A., O'Connor, R., Tichkiewitch, S., Messnarz, R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2010. Communications in Computer and Information Science, vol 99. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-642-15666-3\\_8](https://doi.org/10.1007/978-3-642-15666-3_8)

Weststar, J., & Dubois, L.-É. (2023). From Crunch to Grind: Adopting Servitization in Project-Based Creative Work. *Work, Employment and Society*, 37(4), 972-990.

<https://doi-org.services.lib.mtu.edu/10.1177/09500170211061228>

Appendix A

**Knowledge Audit Outline**

**Complex Cognitive Task:**

Q: Tell me about a time when you altered the structure of a sprint or timeline to eliminate the need for crunch time?

**(1) Big picture probe:**

Q1: Can you describe an example of what is important about the big picture for this task?

Q2: What are the major elements you have to know and keep track of?

**(2) Job smarts probe:**

Q: Is there (regarding the task they outlined) a way to work smarter at this?

**(3) Past & future probe:**

Q: Is there a time when you walked into a situation and knew exactly how it got there and could predict what would happen next? Please describe this instance.

Appendix B

**CDM Interview Outline**

**SWEEP 1: Identifying incident**

Q: Tell me about a time when you altered the structure of a sprint or timeline to eliminate the need for crunch time?

*\*Same question from the knowledge audit, scenarios can be reused.*

**SWEEP 2: Timeline**

*\*Outline timeline of major decisions and the cues that lead to them.*

**SWEEP 3: Context questions**

- 1) What about the situation led you to your decisions?
- 2) What were you noticing about your team leading up to your decision?
- 3) Did you consider the potential impact (good or bad) that your actions could have had on future sprints?
- 4) Have you been trained to deal with these types of situations?
- 5) Were there any clear signs that the project was headed towards needing crunch time?
- 6) Were you reminded of any past projects that experienced a similar situation?
- 7) Were there any other courses of action you considered before making the choice you did? Why were they abandoned?
- 8) How would you describe the situation to someone outside of your field?
- 9) Did you consult any team members before making a decision?
- 10) How frequently are you able to use the same strategy?

**SWEEP 4: What if?**

- 1) What kind of training would aid in making productive decisions about a project's scope?
- 2) What mistakes are a less experienced project manager likely to make in your position?
- 3) What if your publisher comes to you with last minute additions?
- 4) What do you do if crunch time is largely unavoidable?
- 5) What if lengthy bug fixes and other issues push your timeline back?