Game Conceptualization and Development Processes in the Global Game Jam

Alexander Zook and Mark O. Riedl
School of Interactive Computing, College of Computing
Georgia Institute of Technology
Atlanta, Georgia, USA
{a.zook, riedl}@gatech.edu

ABSTRACT

The Global Game Jam provides a unique opportunity to study time-constrained game development at a massive scale. We administered a free-response survey to 2013 Global Game Jam participants about their game development process. Categorized responses show: (a) participants use diverse inspirations; (b) set goals for their personal benefit, the impact on game players, and structure of the game system; (c) rarely employ traditional prototyping; and (d) evolve their games by scoping down many ideas, grounding a vague idea through implementation, and iteratively expanding a simple core game. We discuss next steps to gain more in-depth information about design processes.

Categories and Subject Descriptors

K.8.0 [**Personal Computing**]: General—*Games*

General Terms

Human Factors, Measurement

Keywords

game design, game development, global game jam

1. INTRODUCTION

The Global Game Jam (GGJ) provides a unique opportunity to study the process of conceiving and building a game *de novo* within tight time constraints. Strict time limits allow studying the game design and development process at a level of detail normally not possible. Further, massive participation (16,705 registered participants) enables large-scale analysis. However, these opportunities come with methodological challenges for studying the design process. What are effective methods for understanding design practices that can balance the scale of the GGJ with rigorous, detailed analysis? How can the unique structure of the jam be accounted for to help generalize results from GGJ participants to broader game design practices and methods?

Foundations of Digital Games Workshop on the Global Game Jam 2013 Chania, Crete, Greece

In this paper we study the compressed development process of conceptualizing a game and realizing the game in a working product at the 2013 GGJ. Studying this process is challenging—building a rigorous theory of the time-limited development process requires understanding how designers choose ideas and develop their game ideas, and how this relates to the dynamics of group collaboration and code-level implementation. We propose a multi-step approach: (1) using survey instruments to first characterize the space of game design process and (2) following with more detailed studies of aspects of these processes. This paper describes the results of a free-response survey we administered to 2013 GGJ participants about their design inspirations and goals, process of implementing those ideas in a game over the course of the GGJ, how they refined their game, and pitfalls they encountered along the way. We find common trends in inspirations for game ideas, design goals for games, and process for implementing a design into a working game. We conclude with a discussion of ways to deepen this analysis through triangulation with other research methods and directions for further study.

2. BACKGROUND

We examine the process of conceptualizing and realizing a game's mechanics: inspirational sources, design goals, prototyping and developing the game, and the interplay between design concepts and game coding. The GGJ emphasizes values of experimentation and innovation; we seek to characterize the design goals and insirational sources GGJ participants have for their games and how these are managed through the development process.

The merit of different design goals have been extensively debated in the game design literature (e.g. consider prominent game design texts [5, 13–15]). Design goals range from teaching skills to players [8] to creating immersion and a sense of flow [13] to inducing social change [10]. Bogost [2] catalogs a plethora of uses for games—from inducing relaxation to drilling skills—using existing game examples. Despite this rich discussion, little empirical work has examined the space of design goals. We examine the range of design goals GGJ participants set for their designs.

Designers draw inspiration from a variety of sources. Models of game conceptualization suggest many entry points for starting a design, but are based on anecdotal experience and theoretical analyses rather than existing practices [7]. Example sources for game ideas include life experiences a game is inspired by [1, 16], metaphors a game is meant to convey [12], or models of systems [3]. We empirically exam-

ine GGJ participants' sources and their relation to the use of themes in the GGJ.

Developing an idea into game mechanics involves grounding the abstract game concept into a set mechanics through: (1) prototyping designs, (2) implementing running code, and (3) refining the game and/or design goals. Throughout this process there is potentially a feedback loop between the game artifact and conceptualization. Initial inspirational ideas must be grounded in particular game systems and designers vary in how they approach the problem [6, 9, 11]. Some approaches emphasize iterative playtesting [5,15] while others test a breadth of small ideas before settling on an idea [6]. Prototyping may leverage paper models [9], formal abstract models [4, 11], or simple code [6]. We examine the use of prototyping in the GGJ and approaches designers take to realizing their ideas in running game code.

Developers refine games by finding aspects to alter, selecting among those aspects, and choosing how to change them. Regardless of the final level of "polish," game designs typically go through some refinement of game systems to achieve the goals designers have set out [5, 15]. We study how GGJ games and ideas were altered.

3. METHODS

We provided a nine question open-response survey that was administered online as part of the post-GGJ extended survey (Appendix A). We gathered and manually coded responses to each of the questions into categories, allowing multiple possible codes for responses. Below we report on responses related to the main survey topics, combining information gained across specific survey questions. Note that responses were coded on a per-answer basis, allowing an individual response to have multiple codes. Respondents did not answer all questions. We report the number of responses of a given type and total number of responses to indicate comparative frequency, rather than serve as a rigorous quantitative analysis of magnitude. Due to the survey structure, demographic information was not available for most of the participants completing the extended post-GGJ survey, preventing comparisons to the full GGJ population.

4. RESULTS

Of 16,705 registered participants in the GGJ 419 responded to at least one question. Below we discuss broad categories of responses within each of the study topics: inspirations, design goals, prototyping processes, and the flow of realizing game ideas in code.

4.1 Inspirations

Participants drew from a breadth of sources for inspiration: other games, abstract concepts, emotions, life experiences, art styles, biological systems, books and poems, music, and films. Many mentioned explicit use of the 2013 GGJ theme—the sound of a heart beating—inspiring the use of rhythm in game mechanics, biological hearts as model systems, and life experiences of love.

The theme proved to be the most common inspiration, followed by mechanics and other games or game genres (Table 1). Game references included specific digital games (e.g. Super Mario Bros.), playground or field games (e.g. Simon Says or tag), tabletop games (e.g. Hive), or game genres (e.g. platformer, card games). Other games inspired mechanics, art styles, controls, "feel," and so on. Overall, game references targeted single-player games and action-oriented genres (side-scrolling runner, platformer, one-button games, etc.).

coding	frequency	percentage
theme	60	21.7
mechanic	40	14.5
other video game	39	14.1
game genre	34	12.3
life experience	17	6.2
movie	14	5.1
story	14	5.1
biological system	12	4.3
emotion	10	3.6
book	9	3.3
abstract	8	2.9
art style	6	2.2
board game	6	2.2
previously made game	5	1.8
music	2	0.7

Table 1: Inspiration sources for game concepts.

Life experiences included specific memories (e.g. watching a blind-friendly TV show) and general activities (e.g. holding a conversation). Biological systems—particularly the heart and associated diseases—were a common source for system-oriented designs, primarily due to the heart beat jam theme. Non-game media provided inspirational stories (e.g. Edgar Allan Poe's "The Telltale Heart") or characters and concepts (e.g. the Borg from "Star Trek"). Overall, these results show the breadth of topics addressed by GGJ participants is largely commensurate with industry and academic views, but scoped to meet the GGJ's time demands [2].

4.2 Design Goals

Three broad categories of goals drive GGJ participants: (1) personal goals, (2) player-oriented goals, and (3) system level goals. Personal goals focused on benefits to GGJ participant themselves. The single most common goal was to make and finish a game (Table 2). Other personal goals included learning skills, networking with others, building a portfolio, testing potential ideas for later expansion, enjoying the game creation process, or even to "win the competition." Participants see the jam as an opportunity to test out game development or seed their future projects. An emphasis on competition among some is particularly interesting given the GGJ site explicitly states the GGJ is not a competition.

Player-oriented goals emphasize the person(s) engaging with a game. GGJ participants referenced goals of players enjoying the game, impacting the player through learning about a new topic (e.g. bee colony collapse disorder), or engaging in critical thinking about a topic. Societal-level design goals aimed to raise awareness about world issues.

System-level goals emphasized creating a game of a certain type ("old school point and click adventures") or that meets certain design criteria ("multiplayer game with using [sic] physical mechanics"). Participants emphasized recreating other games, trying out new mechanics, having an original game, or attempting to convey a theme through the game structure. The GGJ theme and emphasis on innovation inspired some to set a goal for the final game system and strive to realize the conceived system in a concrete, running game.

Compared to the standard design mantra of focusing on

coding	type	frequency	percentage
make a game	personal	97	30.2
try out a mechanic	system	48	15.0
have player enjoy	player	44	13.7
learn skills	personal	33	10.3
recreate classic game	system	25	7.8
convey a theme	system	15	4.7
enjoy making a game	personal	15	4.7
be original	system	13	4.0
meet people	personal	8	2.5
impact the player	player	7	2.2
raise awareness of issue	player	6	1.9
test an idea	personal	4	1.2
win competition	personal	4	1.2
build portfolio	personal	2	0.6

Table 2: Goals for the game jam game design.

coding	frequency	percentage
programming	108	33.6
group members	25	7.8
art	25	7.8
time	24	7.5
bugs	20	6.2
none	18	5.6
scoping	18	5.6
express idea in code	16	5.0
balance	16	5.0
source control / collaboration tools	15	4.7
planning and integration	10	3.1
conceive idea	10	3.1
ability to execute	7	2.2
hardware	5	1.6
audio	4	1.2

Table 3: Problems encountered during development.

player experience, the GGJ encourages a broader range of goals for personal gain, social improvement, or innovation.

4.3 Prototyping and Development Processes

Prototyping and development varied in the tools employed and process for developing the game with those tools. Relatively few respondents reported any form of prototyping (127 of 241 response). Many noted that their either was no time to prototype or that they considered their final game a prototype in itself. Others described a process that began as prototyping, but ended up being the final game.

Prototyping processes broadly employed either paper prototyping (22) or engine prototyping (92). Paper prototyping used whiteboards, paper drawings, or various tokens and pieces to simulate game systems and mechanics before beginning to code the game. Relatively few participants mentioned the use of paper prototypes, possibly due to lack of experience, familiarity, or the limitations of the jam. Participants who did paper prototype described it as a beneficial practice: "Complete paper prototype, make a turn-based version of the game. It was critical to nail the design in an hour and get working."

Engine prototyping used game making software and engines (most commonly Unity) to test ideas or incrementally build up a core game. Mechanics, levels, characters, physics, controls, animations, movement, and user interfaces were all subject to this prototyping process. Participants often reported developing initial prototypes in game creation software intending to switch to a more complex development, only for the initial prototype to evolve into the final game. In these cases features were incrementally added to the initial game until the end of the jam.

Development processes either iteratively built on a simple prototype (67 of 93) or quickly tested multiple prototypes before moving into development (26). Iterative development approaches started from the core of the game before building upon it. Iterations would serially add new mechanics, add or improve art assets, or balance existing aspects of the game based on personal testing or outside playtest feedback. Iterative development aimed to realize a pre-conceived game and hone its execution starting from an in-engine prototype.

Test prototypes explored potential ideas to prove whether an idea was valuable or demonstrate a concept to others. Rarely (3 responses), this process would involve parallel creation of multiple ideas before selecting the design to use. Test prototypes sometimes became the final game, but conceptually differed from iterative development. Test prototypes checked if an idea was worth pursuing, rather than initiating the process of iteratively developing a chosen idea.

Development centered on learning tools to implement mechanics and removing game-breaking bugs. Respondents overwhelmingly indicated programming and acquiring and using game development tools as their most pressing challenges (108 of 280) (Table 3). Issues around coordinating a team and implementing code were most prominent, with conceptualization and game refinement as less frequent problems. GGJ participants struggled to program their envisioned ideas, with design-level issues and asset creation as secondary concerns.

4.4 Game Evolution

Converting a game concept into an implemented game typically involved changing both the intended game features and in-game mechanics. GGJ participants managed the set of game features and game artifact in three ways: (1) starting from many ideas and iteratively reducing scope; (2) starting from vague ideas and building up mechanics and ideas through implementation; and (3) starting from a core idea and building it up based on testing and feedback.

Ideas were changed by: adding or removing planned mechanics, swapping out one mechanic for an (often simpler) alternative, and fine-tuning and balancing a mechanic. Some participants included details on changes to the game objectives, story and theme, art assets and animations, or functionality of multiplayer interactions (Table 4).

Scope reduction started from a complex specification before cutting planned features, mechanics, story, or interface elements (138 of 278). Cutting features reduced the overall functionality of the game before implementation, typically because the magnitude of the task was unfeasible or time ran out. Participants removed systems within the game (e.g. attacks requiring combinations of buttons rather than single buttons) or reduced the total number of components used (e.g. fewer game levels or types of enemies). Swapping mechanics occurred when already implemented systems were buggy or dysfunctional or when playtesting (personally or with others) showed them to be overly complex or unintuitive. Iterative scope reduction was the most common way participants described their process and was typically due to development constraints.

Concept development started from a vague specification of the game and cyclically expanded the core systems while crystallizing the design concept (38). Rather than carefully

coding	category	frequency	percentage
feature cut	scope reduction	59	21.2
none	none	31	11.2
change details	idea expansion	28	10.1
mechanic cut	scope reduction	25	9.0
component cut	scope reduction	24	8.6
mechanic swap	scope reduction	20	7.2
feature add	idea expansion	17	6.1
genre change	concept development	17	6.1
visualization swap	idea expansion	13	4.7
story change	concept development	12	4.3
idea change	concept development	9	3.2
mechanic add	idea expansion	9	3.2
visualization cut	scope reduction	6	2.2
control change	idea expansion	4	1.4
story cut	scope reduction	4	1.4

Table 4: Changes to design ideas during jam.

plan out a full game, a vague inspiration would seed the process of solidifying ideas through incremental development: "Mostly we talked the idea out, and just got to jamming. We iterated on the first prototype, and went from there." Extreme cases involved scrapping an initial idea and restarting (9) or changing the genre of a game after starting implementation (17). Concept development approaches emphasized exploring alternative ideas over detailed pre-planning.

Idea expansion planned a small game and extended it through player feedback (71 of 278). Game changes aimed to improve usability through better feedback to players, controls, or changing mechanics to better align with design goals. Idea expansion emphasized changing features based on player testing to refine the planned game design. Unlike the concept development process of design exploration, idea expansion has a process of iterative design refinement.

Overall, GGJ participants tend to change game features, rather than compromise design ideas. Most development reduced the set of features implemented (132 of 278), refined initial ideas through feedback (77), or did not change the planned game at all (31). Openly exploring possible designs was comparatively rare (38). Features are more likely to change than the design ideas. Taken together, GGJ development is focused on delivering a planned idea, rather than experimenting with possible ideas.

5. METHODOLOGICAL IMPLICATIONS

Our voluntary survey methodology has important limitations in coverage of GGJ participants and the depth of response data gathered. Only 419 of 16,705 participants responded to these questions. Respondents were likely skewed toward successful projects and those more invested in the GGJ. Thus, we cannot easily examine similarities or differences in design processes between those who successfully complete the GGJ and those who do not. Future research will require methods to automate survey administration and collection or ensure randomized sampling from participants.

Survey responses are limited to the most salient aspects of an experience, preventing detailed processual information gathering. We cannot make strong conclusions about the cognitive or social processes involved in game development from this form of data. Retrospective protocol analysis—where participants are recorded and asked to then view this recording and narrate their thinking—is one means to gather more detailed data, although constrained to a smaller scale than we studied. Retrospective protocols are typically used for short sessions (up to hours). Modifications for longer duration events may review only key points in the process or to use a "fast-forward" viewing approach.

Semi-structured interviews allow an exploratory approach to collecting detailed data. Interviews are limited to subjective data, but require less time and detailed data than protocol analysis while yielding valuable qualitative insights. Using prompt materials gathered over the course of the GGJ—such as in-process game builds from source control, photos or video of onsite activity, and observer notes on the development process—may ameliorate participant memory biases.

Our survey did not have identifying information on participants. Thus, we could not study of how collaboration impacts the conceptualization and development process at the GGJ. Employing a retrospective protocol or semi-structured interview with individuals and then groups is one means to collect such information.

6. DISCUSSION

To date, game design processes have been examined using personal reflection [1, 7], small-scale individual interaction [11], or study of complete games [2]. We complement this work by with a large-scale analysis of time-constrained game design, examining development process trends. GGJ participants have diverse inspirations and set goals for their personal benefit, the impact on game players, and structure of the game system. Participants rarely employ traditional prototyping, instead evolving their games by scoping down complex ideas, grounding a vague idea through implementation, and iteratively expanding a simple core game.

Our results show great potential for fine-grained analysis of the relationships design between processes and development outcomes. Designers vary in scoping, grounding, or expanding out initial ideas. Our preliminary study opens a number of future research questions. How do final products of these methods differ? Which aspects of games are amenable to incremental addition and which must be present from the start? How do designers recognize dependencies among game systems and prioritize them? Relating design processes to outcomes is crucial for better structuring future jams, game development instruction, and tools.

Many designers struggled to implement their envisioned game systems and mechanics. How do they go about realizing the mapping of a mechanic concept to pieces of running code? Further, development was typically iterative. What kinds of feedback do designers use to guide development? How is feedback interpreted? What guides decisions to cut features as opposed to adjust the game code? Understanding the feedback loop between a running game and design concepts can inform methods for teaching designers and building better game development technologies.

Future work must develop methods for large-scale recording and analysis of design processes. Surveys provide useful large-scale qualitative results to develop theory, but require complementary small-scale studies of process-level details. Automated recording of game development processes and annotating these records are one means to enable quantitative analysis at the massive scale the GGJ provides. Future research should build on these results to examine fine-grained details of time-constrained game development and develop new methodologies to leverage the potential of massive development information from sources like the GGJ.

7. REFERENCES

[1] A. Anthropy. *Rise of the videogame zinesters: how freaks, normals, amateurs, artists, dreamers, drop-outs, queers, housewives, and people like you are taking back an art form*. Seven Stories Press, 2012.

[2] I. Bogost. *How to do things with videogames*. University of Minnesota Press, 2011.

[3] C. Crawford. *The art of computer game design*. Osborne/McGraw-Hill Berkeley, CA, 1984.

[4] J. Dormans. Simulating mechanics to study emergence in games. In *Workshop on Artificial Intelligence in the Game Design Process at the Seventh Conf. on Artificial Intelligence for Interactive Digital Entertainment*, 2011.

[5] T. Fullerton, C. Swain, and S. Hoffman. *Game design workshop: a playcentric approach to creating innovative games*. Morgan Kaufmann, 2008.

[6] K. Gabler, K. Gray, M. Kucic, and S. Shodhan. How to prototype a game in under 7 days: Tips and tricks from 4 grad students who made over 50 games in 1 semester. online, 2005.

[7] R. Hunicke, M. Leblanc, and R. Zubek. MDA: A formal approach to game design and game research. In *Proc. of the AAAI Workshop on Challenges in Game AI*, 2004.

[8] R. Koster. *A theory of fun in game design*. Paraglyph press, 2005.

[9] J. Manker. Game design prototyping. In *Games and Innovation Research Seminar 2011 Working Papers*, 2011.

[10] J. McGonigal. *Reality is broken: why games make us better and how they can change the world*. Penguin Press HC, 2011.

[11] M. J. Nelson and M. Mateas. A requirements analysis for videogame design support tools. In *Proc. of the 4th Int'l Conf. on Foundations of Digital Games*, pages 137–144. ACM, 2009.

[12] D. Rusch and M. Weise. Games about love and trust?: Harnessing the power of metaphors for experience design. In *Proc. of the 2008 ACM SIGGRAPH Symposium on Video Games*, pages 89–97. ACM, 2008.

[13] K. Salen and E. Zimmerman. *Rules of play: game design fundamentals*. MIT Press, Cambridge Mass., 2003.

[14] K. Salen and E. Zimmerman. *The game design reader: a rules of play anthology*. MIT Press, Cambridge Mass., 2006.

[15] J. Schell. *The art of game design: a book of lenses*. Elsevier/Morgan Kaufmann, 2008.

[16] M. Treanor, M. Mateas, and N. Wardrip-Fruin. Kaboom! is a many-splendored thing: An interpretation and design methodology for message-driven games using graphical logics. In *Proc of the Fifth Int'l Conf. on the Foundations of Digital Games*, pages 224–231. ACM, 2010.

APPENDIX

A. SURVEY QUESTIONS

- What was your initial goal for the game you made during the global game jam?
- What inspirations or initial ideas did you have for your game? What was the starting inspirational source or goal for the game?
- Why did you pick this particular idea for the game?
- What problems did you encounter in developing your game?
- What changes did you make to your initial idea as you worked on it during the game jam? Please describe the changes as small pieces of changes as possible.
- What game mechanics and/or gameplay systems did you use in your game?
- How did the mechanics or systems you made relate to the initial design ideas you had?
- How did these mechanics change as you worked on the game during the game jam?
- Did you prototype your game? If so, what kind of prototyping did you do and what did you learn from doing it?