Methods Used in Game Development to Foster FLOW

Isaac Ben Jeppsen – Producer

Breakaway Ltd

ijeppsen@breakawayltd.com

Abstract. Games designed for entertainment have a rich history of providing compelling experiences. From consoles to PCs, games have managed to present intuitive and effective interfaces for a wide range of game styles to successfully allow users to "walk-up-and-play". Once a user is hooked, successful games artfully present challenging experiences just within reach of a user's ability, weaving each task and achievement into a compelling and engaging experience. In this paper, engagement is discussed in terms of the psychological theory of Flow. I argue that engagement should be one of the primary goals when developing a serious game and I discuss the best practices and techniques that have emerged from traditional video game development which help foster the creation of engaging, high Flow experiences.

1. INTRODUCTION

Serious games and simulations are often focused on training or teaching a specific skill or procedure, and consequently there is a tendency to focus on the fidelity of the product at the expense of other salient factors. In contrast, the focus of a game developed purely for entertainment purposes, is to engage the user with fidelity taking a supporting role. An example of this shift in focus is the differences between usability analysis performed on traditional software and user experience analysis performed on games [1]. Usability as it applies to software is usually viewed from the perspective of business software; the goal is to provide a tool or tools to create or modify an artifact-- such as a document or picture. In general, this perspective is concerned with trying to create interfaces which are transparent to the user. Conversely, games provide a tool for users to experience an artifact; such as the experience of being a criminal, hero, doctor, etc. While straight forward, this shift in focus from transparent tool use to compelling experience has important implications for serious game development teams. It is not sufficient to make an easy-to-use tool; it most also engage the user in an optimal manner.

Most of us have experienced the gratification that arises from having sufficient skill to deal with a problem or challenge just outside of our comfort level. In other words, an experience characterized by the feeling of being so engrossed and focused on the task at hand that everything else is eclipsed. This is referred to in sports as "being in the zone". This state of optimal experience is what Dr. Csizenthmihalyi refers to as a state of Flow [2]. Creating optimal experiences is what games, both for serious and for entertainment purposes, are all about. Developers strive to create this optimal experience by immersing the user not only in an accurate environment, but also in one where the player is compelled to explore and to experiment. Fidelity alone is not enough to create immersion. Immersion arises out of a combination of both salient details and a unique perspective or role experienced by the user.

In what follows, I give a brief summary of the theory of Flow and how it relates to games. Then, I discuss how the practices and structures in game development have evolved to support the creation of optimal experience. Though originating from entertainment game development, the practices discussed should be generalizable to any immersive application.

2. FLOW AND USER EXPERIENCE

The theory of Flow was motivated by Dr. Csizenthmihalyi's desire to identify what happiness is, and the elements that contribute to positive emotions in general. From the experiments he conducted to investigate this psychological state, he found that naïve notions of happiness (e.g. care free, stress free, etc.) do not explain why people

performing demanding and complicated tasks report a sense of well being and enjoyment. To explain the phenomena of optimal experience, he developed a theory that he coined Flow. Flow is a set of eight heuristics that describe common features that contribute to an optimal experience. The eight components are [2]:

- Clearly defined goals
- Concentration on task at hand
- Merging of action and awareness
- An altered sense of time
- Clear and responsive feedback
- Balanced level of challenge and difficulty
- A sense of control over the task at hand
- A challenging task requiring skill to execute

These eight components serve as a good definition for engagement as it pertains to games. Through a process of evolution facilitated by the fiercely competitive commercial game market, successful game development teams have been deliberately leveraging some or all of these heuristics to deliver the compelling and immersive experience gamers have come to expect.

The most obvious example of the use of Flow in games is the game 'flOw' [3] created by Jenova Chen. While this game started as a thesis project, it has since been released commercially to both critical and public acclaim. In this game Chen uses the two dimensions of challenge and skill to design a game that dynamically adjusts the difficulty of game play through the user's own choices. This is in contrast to indirectly controlled methods that analyze game play and adjust difficulty by changing variables such as enemy spawn rates and weapon damage. An indirect method has several problems, including multiplayer balancing, feedback to the user and a tendency to create a less than optimal experience.

As can be seen in figure 1, the goal of the game 'flOw', and indeed any game, is to keep the player "in the zone".
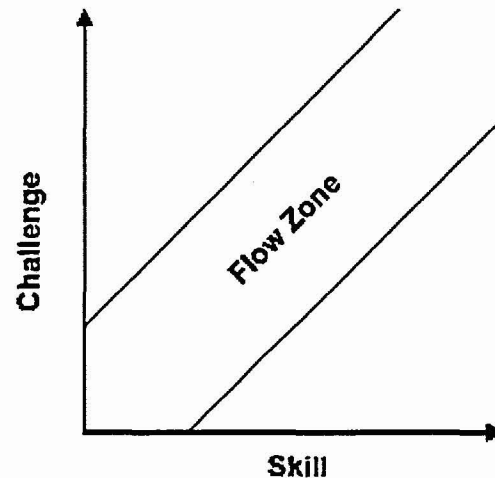

Figure 1 Showing the zone between challenge and skill where the user has an optimal experience

3. GAME DEVELOPMENT TEAMS – ANATOMY, STRUCTURE, PROCESS

Due to the fundamentally subjective experience of games and the wide range of people that play them, simply knowing the components of Flow does not imply this knowledge will translate to the implementation. As such, creating Flow in games remains more an art than science. Even efforts that go into much more detail than Flow, such as the heuristics of PLAY [4], acknowledge the fundamentally artistic nature of game development.

The artistic nature of games requires a different approach to the development process compared to other commercial software development. These differences can be seen in both the team structure and in the development process in general. For example most software development teams don't include visual artists throughout the processes. Having this resource throughout development allows for more sophisticated mockups to be created giving quicker and more accurate representations before significant resources are devoted. This is just one of many examples of the subtle to gross

242

differences in game development team dynamics compared to more traditional efforts.

Due to the competitive nature of commercial game development it is often hard to get a detailed picture of a particular development studio's structure and practices; however there has been a growing body of work by both academic researchers and game development professionals that allows for some broad generalizations to be made. The main high level components that directly contribute to the development of games that deliver a flow experience are:

- Rapid iteration
- Motivated Multidisciplinary teams
- Vision holder(s)
- Play testing

3.1 Rapid Iteration

As any movie or game executive can attest, ideas are abundant and playable products are scarce. Only the details that emerge through the production process allow for a given idea to be evaluated as being "good" or "bad". These details are even more important in game development due a game's interactive nature. For a game to be successful it has to be "fun," and at present, there is no way to determine the "fun" of a game without playing it. Game development takes an initial idea such as, "let's manage things," and through a series of successive iterations discovers "where the fun is". This processes starts by using mockups, concept documents, and playable prototypes and grows into the final polished product that (hopefully) gets shipped.

While rapid iterative development is beginning to get a lot of traction across the software development industry, successful game developers have been early and often trail-blazing practitioners. One of the reasons for the rapid adoption of Agile production methodologies such as Scrum [5] by game development studios is the emphasis on a "playable" product at the end of each short milestone or "sprint" (usually between 2 and 6 weeks). As illustrated in Figure 2, by taking small incremental steps, the team is able to make corrections to keep the game fun before it becomes infeasible to change.
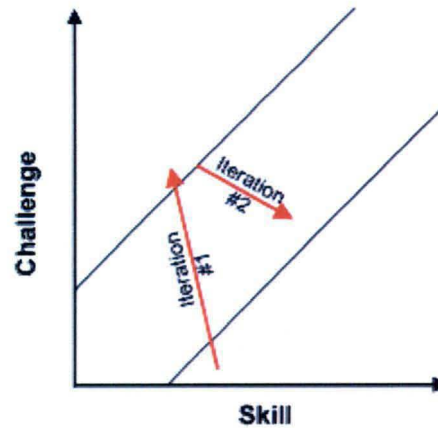

Figure 2 Rapid iteration used to evaluate "fun" Iteration #1 starts with the game being too easy and ends being a little too hard.

The importance of rapid iteration should not be underestimated. As games become more complex the ability to rapidly create, test and experience a game will only become more important.

3.2 Motivated Multidisciplinary teams

Modern day AAA games (an AAA game refers to a high budget, high production value game,) are incredibly complex endeavors often involving core team sizes in the hundreds. Smaller independent game titles can have teams ranging anywhere from one person to forty or fifty, the former being very rare for a game of any significant complexity. At a high level production, teams are composed of designers, software engineers, artists, sound engineers, quality assurance, subject matter experts and production staff. In contrast to development teams in other industries, most if not all of these positions are involved from the conception through the completion of the project. Involvement of all disciplines early in the project is due to the user centered design focus inherent in video game development. At the beginning, artists will create concept mockups that illustrate not only the functional elements of the user interface, but the aesthetics as well. In tandem, designers will be working out the logical flow of the interface from the user perspective, and engineers will be focusing on the logical flow as well as the supporting infrastructure from the software perspective. This parallel development helps to give a more complete picture of the product

early and at every stage of the production processes.

A maxim from usability experts states "know thy user, for they are not you" and while this largely holds true for game usability, game development teams tend to be more diverse and enthusiastic than developers in other markets of software development. For example, developers are almost universally consumers of the products they develop. If asked, a typical game developer, in any discipline, will likely categorize him/herself as a "gamer" (i.e., someone who plays games regularly if not obsessively). Game development teams in this respect, more closely reflect the movie industry in which professionals that create movies got into the career due to a passion for the medium as a consumer. Having developers that understand the "user" viewpoint helps to internalize a user-centered design approach. And while this doesn't guarantee that the end result will be widely appreciated, it does allow for larger leaps forward in creating and maintaining Flow-inducing games than a strictly formal approach could achieve.

3.3 Vision Holder

While the previous sections have focused on the team as a whole, a single authoritative vision holder is essential to insuring that the end product does not suffer from the "design by committee" effect. A vision holder also ensures that the overall game experience does not become confused and diluted due to a lack of cohesion. Again the film industry provides an excellent comparison, in the role of a director. While there is not always a title on a game development team that directly corresponds to that of a movie director, the role is often filled by either the producer or lead game designer.

The role of the vision holder is to internalize the details of the game and to integrate the contributions of all the team members into a cohesive whole. The complexity of modern games involves many different disciplines and specialties and requires that someone bring it all together to create something greater than the sum of its parts.

3.4 Play testing

As stated earlier, game development teams typically have a lot of "users" embedded in the form of developers; and while this helps to guide development, there is an inevitable feedback loop or echo chamber effect that can cause developers to lose touch with their larger audience. This is where user experience analysis or "play testing" [1] comes in. Play testing is very similar to usability testing in business software with a few key differences. In play testing, the goal is to measure not only the ease with which the user is able to interact with the game, but also the level of engagement of the user. Play testing in recent years has become more sophisticated with larger studios and publishers using a broad array of measurements and analysis tools to refine the user's experience. These tools include traditional methods such as surveys and focus groups, and more technologically driven approaches such as gaze tracking and EEG analysis.

While the costs of both the traditional and technological methods are continuing to decrease, they are currently prohibitive for smaller studios. Smaller studios may not be able to bring the same resources to bear; they can and do employ informal versions of the above with varying degrees of success.

The choice of play testing method should be driven by the development methods mentioned previously. Of these, the ability to integrate the play testing into a demanding and rapidly changing production schedule is of utmost importance. In order for play testing to be useful it must be done in a way that encourages rapid iteration. Formal play testing studies can take months if not years to organize and gather the data required, representing an unacceptable risk in terms of both time and resources required.

The emergence of brain imaging technologies combined with advances in neuroscience and cognitive models offer promising directions for a more automated and efficient approach to play testing. By combining cognitive models of various player profiles with psycho-physiological data collected as people play

games, it may become possible to create cheaper and more reliable methods to gauge the flow of the user. The automation and consequent decrease in time and resources that these approaches promise may make them excellent candidates for inclusion in future game development efforts.

4. SUMMARY

The number of problems that games can be applied to has just begun to be explored. A few projects have begun to explore using games to harness human's spatial and pattern matching abilities. For example "Folding@home" [6] uses a game to help determine how proteins fold. As the number, difficulty and complexity of problems that gaming is used to solve increases, it is imperative that development teams and those who manage them understand what motivates people to play games, and how to foster an environment that nurtures the creation of games people want to play.

5. REFERENCES

1. Thompson, Clive. "Halo 3: How Microsoft Labs Invented a New Science of Play". *Wired Magazine*. 2007, 15.09.

2. Csikszentmihalyi, Mihaly. The Psychology of Optimal experience. New York : Harper & Row, 1990.

3. Chen, Jenova. flOw. Welcome to Flow in Games. [Online] [Cited: August 27, 2009.] http://intihuatani.usc.edu/cloud/flowing/.

4. Game Usability Heuristics (PLAY) For Evaluating and Designing Better Games: The Next Iteration. Desurvire, Heather and Wiberg, Charlotte. San Diego : Springer, 2009. HCII 09.

5. Schawber, Ken and Beedle, Mike. Agile Software Development with Scrum. Upper Saddle River : Prentice Hall, 2001. 0-13-067634-9.

6. Folding@home - Main. Standford University. [Online] [Cited: August 27, 2009.] http://folding.stanford.edu/English/Main.

7. *Flow in Games (and Everything Else).* Chen, Jenova. April 2007, Communications of the ACM, pp. 31-34.