



**NAVAL
POSTGRADUATE
SCHOOL**

MONTEREY, CALIFORNIA

THESIS

**QUANTIFYING THE EFFECTIVENESS OF
CROWD-SOURCED SERIOUS GAMES**

by

Umit Tellioglu

September 2014

Thesis Advisor:
Second Reader:

Geoffrey G. Xie
Thomas J. Housel

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE September 2014	3. REPORT TYPE AND DATES COVERED Master's Thesis		
4. TITLE AND SUBTITLE QUANTIFYING THE EFFECTIVENESS OF CROWD-SOURCED SERIOUS GAMES			5. FUNDING NUMBERS	
6. AUTHOR(S) Umit Tellioglu				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) Crowd-sourced serious games (CSSGs) represent an emerging genre of games. Different from traditional games, the primary concern of the CSSGs is not player enjoyment, but contributing to difficult scientific problems or respectable social causes through incremental efforts embedded in parallel game plays by many non-specialists. CSSGs have a potential to support important tasks for humanity. Clearly, players' contributions and the effectiveness of CSSGs is crucial for success. Further, players may have different motivations to play CSSGs than traditional games. Some players (called whales) produce more than other players possibly due to a stronger motivation. In addition, those contributions and their effectiveness must be measured and evaluated to improve CSSGs. In this thesis, we propose a methodology to quantify the effectiveness of CSSGs by analyzing mainly two VeriGames produced for DARPA's Crowd Sourced Formal Verification project. The analyses show that low engagement rates (ERs) can be an obstacle to CSSGs and their ultimate purpose. The results also show this game genre to have a strong whale effect, and thus a strategy focusing on recruiting and retaining whales may be effective to counterbalance the low ERs.				
14. SUBJECT TERMS Crowd-Sourced Serious Games, Verigames, Crowd Source Formal Verification, Game Analytics, Whales, Engagement Rate			15. NUMBER OF PAGES 83	16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**QUANTIFYING THE EFFECTIVENESS OF CROWD-SOURCED SERIOUS
GAMES**

Umit Tellioglu
Captain, Turkish Army
B.S., Turkish Army Academy, 2005

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS AND TECHNOLOGY

from the

**NAVAL POSTGRADUATE SCHOOL
September 2014**

Author: Umit Tellioglu

Approved by: Geoffrey G. Xie
Thesis Advisor

Thomas J. Housel
Second Reader

Dan C. Boger, PhD
Chair, Department of Information Sciences

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Crowd-sourced serious games (CSSGs) represent an emerging genre of games. Different from traditional games, the primary concern of the CSSGs is not player enjoyment, but contributing to difficult scientific problems or respectable social causes through incremental efforts embedded in parallel game plays by many non-specialists. CSSGs have a potential to support important tasks for humanity. Clearly, players' contributions and the effectiveness of CSSGs is crucial for success. Further, players may have different motivations to play CSSGs than traditional games. Some players (called whales) produce more than other players possibly due to a stronger motivation. In addition, those contributions and their effectiveness must be measured and evaluated to improve CSSGs. In this thesis, we propose a methodology to quantify the effectiveness of CSSGs by analyzing mainly two VeriGames produced for DARPA's Crowd Sourced Formal Verification project. The analyses show that low engagement rates (ERs) can be an obstacle to CSSGs and their ultimate purpose. The results also show this game genre to have a strong whale effect, and thus a strategy focusing on recruiting and retaining whales may be effective to counterbalance the low ERs.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

1	Introduction	1
1.1	Background	1
1.2	Problem Statement.	2
1.3	Purpose Statement	2
1.4	Research Questions	2
1.5	Potential Benefits	2
1.6	Organization of the Thesis.	3
1.7	Scope and Limitations	3
1.8	Notification	3
2	Background and Game Analytics Tutorial	5
2.1	Introduction	5
2.2	Electronic Games	6
2.3	Formal Verification of Software	8
2.4	Crowdsourcing	11
2.5	Crowd-Sourced Serious Games.	14
2.6	Crowd Sourced Formal Verification	15
2.7	Basic of Game Analytics	17
2.8	Summary	24
3	Related Works	25
3.1	Introduction	25
3.2	Previous Researches	25
3.3	Summary	26
4	Methodology	27
4.1	Introduction	27
4.2	Data Collection	27
4.3	Post-Data Collection	28

4.4	Summary	37
5	Analysis and Evaluation	39
5.1	Introduction	39
5.2	Summary	48
6	Conclusion	49
6.1	Summary and Conclusion	49
6.2	Future Work and Limitations.	49
	Appendix: Line Fit Plots of Regression Analyses	51
	References	59
	Initial Distribution List	65

List of Figures

Figure 2.1	Cyclic Behavior of Game Analytics	18
Figure 4.1	A Simple Game Monetization Funnel Widely Used for Free-to-play Games	30
Figure 4.2	Whale Effect Graph (WEG)	33
Figure 5.1	Player profiles of VeriGames A and B	39
Figure 5.2	Session Time CDF of VeriGames	43
Figure 5.3	Session Count CDF of VeriGames	44
Figure 5.4	WEG of VeriGames	45
Figure 5.5	WEG of VeriGames Including Anonymous Players	46
Figure 5.6	WEG of VeriGames Including ST and SC	47
Figure 1	VeriGame A DAU LFP	51
Figure 2	VeriGame A DST LFP	52
Figure 3	VeriGame A DSC LFP	52
Figure 4	VeriGame A WAU LFP	53
Figure 5	VeriGame A WST LFP	53
Figure 6	VeriGame A WSC LFP	54
Figure 7	VeriGame B DAU LFP	54
Figure 8	VeriGame B DST LFP	55
Figure 9	VeriGame B DSC LFP	55
Figure 10	VeriGame B WAU LFP	56
Figure 11	VeriGame B WST LFP	56

Figure 12 VeriGame B WSC LFP 57

List of Tables

Table 2.1	Software Dependence of Military Aircrafts by Years	10
Table 2.2	LOCs of a Group of Software	12
Table 4.1	List of Traditional Games Used in this Thesis	28
Table 4.2	Summary of CSSG Data Used	28
Table 4.3	A List of All Metrics Used in the Thesis	37
Table 5.1	Average DAU and MAU for Selected Mobile, Social, and Online Games	40
Table 5.2	DAU of Sample VeriGames	41
Table 5.3	MAU of Sample VeriGames	41
Table 5.4	ER of Some Mobile, Social and Online Games and Developers . .	41
Table 5.5	ER of VeriGame A and B in the Monthly Average Basis	42
Table 5.6	R^2 Values for Regression Analysis (Daily Metrics)	47
Table 5.7	R^2 Values for Regression Analysis (Weekly Metrics)	48
Table 5.8	R^2 and p Values for Multiple Regression Analysis(Daily Metrics)	48
Table 5.9	R^2 and p Values for Multiple Regression Analysis (Weekly Metrics)	48

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

ARPU	average revenue per user
CSFV	Crowd Sourced Formal Verification
CR	conversion rate
CSSG	crowd-sourced serious games
DARPA	The Defense Advanced Research Projects Agency
DAU	daily active users
DSC	daily session count
DST	daily session time
EED	entry event distribution
ER	engagement rate
LTV	life time value
MAU	monthly active users
MSC	monthly session count
MST	monthly session time
WAU	weekly active users
WEG	whale effect graph
WSC	weekly session count
WST	weekly session time

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgements

First, I would like to express my gratitude to the Turkish Armed Forces for giving me the opportunity to study at the Naval Postgraduate School.

I thank the team members Charles Prince and Professor Justin Rohrer. I am also grateful to everybody who helped me during the study and shared their data. Special thanks to my advisor Professor Geoffrey Xie. His patient, endless support, and guidance helped me finish the study. I also appreciate Professor Thomas Housel's teaching of research design and being my second reader.

Of course the most special thanks go to my beautiful wife, Zehra, my lovely kids, Okan and Azra. I am very lucky for having such a wonderful family.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:

Introduction

1.1 Background

The increasingly pervasive Internet provides a platform for effective group communications on a global scale, even among strangers living in different continents. This transformation in communication has led people to envision crowdsourcing as a potentially cost-effective method for tackling tasks that previously could only be performed by domain experts. Two highly publicized executions of this vision are the Duolingo portal [1] and the EyeWire project [2]. The ultimate goal behind the free-of-charge Duolingo portal is to translate the web into all major languages, and the “crowd” is made of people who desire both to learn a foreign language and to support the cause of making useful web content universally accessible. Most of the exercises and exams completed via the Duolingo portal are in fact translating fragments of some real-world web pages from one language to another. The underlying purpose of the EyeWire project is to decipher the structure of the human brain at the neuron level. The researchers set up a web front-end in the form of a virtual I-spy game to recruit a crowd of volunteers to accelerate the process of mapping 2-D images of brain slices into 3-D neuron connectivity patterns.

More recently, the concept of crowdsourcing is also being explored in the highly specialized field of formal software verification [3]. A collection of puzzle-style games, called VeriGames, has been created and hosted publicly on the Internet. Each instance of a game level corresponds to an attempt to assert some properties about a code segment. A backend verification engine then combines the assertions produced from all related game instances and tries to obtain conditions that can rule out certain types of bugs in that code segment.

In this thesis, we broadly classify such crowdsourcing efforts into a new genre called crowd-sourced serious games (CSSGs) as their primary focus is to advance widely respected causes such as social equality (in the case of Duolingo) and science (in the cases of EyeWire and VeriGames).

1.2 Problem Statement

We observe that the general effectiveness of crowd-sourced serious games is largely unknown. The few performance analyses in current literature are limited to documenting experiences with individual systems. More importantly, existing game analytics approaches are designed for games that provide *personal* experience and entertainment. In contrast, CSSGs attract participants by evoking their sense of social responsibility and sympathy for others. Intuitively, social awareness and sympathy alone may not result in the same level of consistent participation as personal achievement or fun. Consequently, the success of a CSSG may be more tightly linked to the contributions of few highly dedicated players (commonly referred to as whales, a term borrowed from the gambling industry, in the current literature). Therefore, the problem is how to quantify the effectiveness of CSSGs.

1.3 Purpose Statement

The purpose of the thesis is to provide a systematic methodology to accurately characterize the performance of CSSGs. This is important because it will help game developers to identify the best practices for improving CSSGs as a genre.

1.4 Research Questions

The research questions are below:

- Player retention is more challenging for crowd-sourced serious games (CSSGs) than for traditional games (whether leisure or educational games).
- The difference in achievement levels between whales and typical players is bigger with CSSGs than the traditional games. In other words, it might be more critical for CSSGs to not just recruit new players, but retain highly-productive players, and at the same time incentivize existing players to increase their productivity.

1.5 Potential Benefits

The proposed methodology in this thesis is applicable to both VeriGames and other CSSGs. Game developers can use the methodology of the thesis to quantify the productivity distribution of all players and identify potential whales. Such an analysis helps them to improve

their games and to realize the overall purpose of the CSSGs.

1.6 Organization of the Thesis

This thesis is organized into the following chapters:

- Chapter I: Introduction
- Chapter II: Background and Game Analytics Tutorial
- Chapter III: Related Work
- Chapter IV: Methodology
- Chapter V: Analysis and Evaluation
- Chapter VI: Conclusion

1.7 Scope and Limitations

In this thesis, we used the raw data received from two VeriGames' developers to generate necessary metrics for the analysis. However, we do not have raw data to generate the same metrics for traditional games and other CSSGs. Therefore, in some places we used data from the Internet and previous researches for traditional games and other CSSGs.

1.8 Notification

Some parts of this thesis have been published in the proceedings of 19th International Conference on Computer Games as a paper entitled, "Whale of a Crowd: Quantifying the Effectiveness of Crowd-Sourced Serious Games," and some other parts will be published in the proceedings of 7th International Conference on Information Security and Cryptology as a paper entitled, "Call of Duty: Can Turkey Benefit from Crowd-Sourced Serious Games to Strengthen Its Cyber Security Capabilities?"

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:

Background and Game Analytics Tutorial

2.1 Introduction

This chapter covers the basic concepts in six topics that will help the readers to understand the thesis. The first topic is electronic games, which includes a definition, a brief history, and classification of electronic games. Next, we will give fundamental information about formal verification of software. This part will introduce a definition of formal software verification, why it is important, and why it is a difficult and expensive process. One of the core concepts in the thesis, crowdsourcing, is the third topic. Under this topic, we will provide a definition of the crowdsourcing concept by giving examples chronologically. In addition, we will give examples of modern crowdsourcing projects. Motivation factors that crowdsourcing projects rely on is the last part of this topic, which will help to understand why people contribute to crowdsourcing projects. The following topic is about crowdsourcing projects that use games to attract people. We will try to illustrate why electronic games are suitable tools to use in crowdsourcing projects.

In the fourth topic of this chapter, we will give examples of CSSGs that use electronic games to transform players' efforts into valuable outputs to solve difficult scientific problems, and how players can be incentivized to increase demand for the games. Next, we will mention the web portal, Verigames, that hosts the five CSSGs that DARPA has used in the Crowd Sourced Formal Verification (CSFV) project. The last topic is about the game analytics concept, which will help readers understand the methodology in the thesis. At first, we will illustrate basic information about game analytics. The questions we are posing include, what is game analytics, and why is it important? We will emphasize the cyclic behavior of game analytics, which we define as having three phases: decide (pre-data collection period), collect (data collection period), and analyze (post-data collection period).

2.2 Electronic Games

2.2.1 What are They?

Sabadello defines a game as a pursuit or activity with rules performed either alone or with others, for the purpose of entertainment and/or competition [4]. The definition of an electronic game is “a game in which electronics are used for establishing the game framework and enforcing game rules” [4]. Sabadello noted that there are several appliances used to play electronic games, including computers, stand-alone arcade consoles, consoles connected to TVs, game machines, and mobile devices [4].

2.2.2 History of Electronic Games

The history of electronic games dates to the middle of the twentieth century, and their popularity simultaneously grew with the affordability of electronic gaming devices. *OXO*, designed by A.S. Douglas in 1952, is one of the earliest examples of an electronic game with a graphical display [4]. Sabadello states that although the early examples of electronic games like *Tennis for Two* (1958) and *Spacewar* (1962) were not released to the public, entrepreneurs understood that making money from electronic games was possible. They came up with new ideas to benefit from the economic potential of the electronic games that initiated the era of electronic gaming.

The 1970s was an important decade for electronic gaming; the first arcade game, and the first home electronic game were developed in that era [5]. Electronic arcade gaming machines were very popular in those years. Moreover, Herman et al. observe that the three important companies for the electronic gaming industry Atari, Nintendo and Sega showed their potential in the period [5]. These companies created popular electronic games and dominated the electronic game market for several years.

In the beginning of the 1980s, Namco, a Japanese company, introduced *Pac-Man*, the most popular arcade game ever [5]. Herman et al. note that also in that period Commodore emerged with affordable computers as a rival to home game consoles. On the other hand, the home computers started to end the dominance of the arcade games and the home consoles in that period [4].

In the 1990s, the arcade games, consoles, and computers improved their games and de-

vices [4]. Later, Sony released the PlayStation to become an important player in the game console market [5]. The growing usage of the Internet and the networking-ability of the game playing devices introduced multiplayer games in the 1990s [4].

Herman et al., named the 2000s as the “The New Era” [5]. Competition in the electronic gaming market was growing. Microsoft entered the game console market with the Xbox, just after Sony released the PlayStation 2 [5]. In addition, the improvement of computer hardware and the Internet bandwidth fostered the emergence of the online and mobile games around 2000 [6].

Arcade Games

Arcade games are specially designed coin-operated machines that mostly exist in public areas [4]. The earliest arcade game was *Computer Space* (1971) [7]. After the success of the early arcade games, the manufacturers realized the potential of electronic gaming and designed arcade machines and electronic games for these devices [4]. After that production increase, arcade games reached their peak of popularity around 1980 [4].

Personal Computer Games (PC Games)

Electronic games that are designed to be played on a personal computer or laptop are called personal computer games [4]. Computers were initially produced for military and governmental organizations or for scientific purposes, and were expensive. In addition, while arcade games are designed for only gaming purposes, personal computers were not. The price reduction, mass production, and increasing usability of operating system graphical interfaces made computers more popular for home use [8]. This popularity created a market for PC games.

Console Games

Game consoles are the devices that are mainly designed to play electronic games. An early example was the Atari 2600 that needed a TV connection and joystick [4]. Modern examples of game consoles are the Sony PlayStation, the Microsoft Xbox, and the Nintendo Wii [4]. Today, games are usually developed for different platforms. In addition, small handheld versions of these consoles have emerged that have their own display and controls.

Online Games

Browser-based games can be played using a web browser [9]. These games have advantages over traditional computer games. They do not usually require a CD/DVD purchase or installation. Thus, launching the games is easy. Online games can also reach many people simultaneously which makes browser games a good platform for social interaction [9].

Mobile Games

Mobile device games are video games that can be downloaded as applications and played on mobile devices like smart phones, tablets, and so forth. Improvement of technology in mobile devices (smart phones and tablets) has increased their processing and storage capacity, visual and audio capabilities, all of which attracts many mobile game players to these devices [6].

Mobile devices such as smart phones and tablet computers are the preferred game platforms because they are affordable, they provide the same pleasure and performance qualities as other game platforms, and they are portable. In the Internet era, the mobility of electronic devices has increased according to a Gartner Report [10]. One billion smartphones were sold in 2013, up from 675 million in 2012. In addition, tablet sales increased from 116 million in 2012 to 197 million in 2013 [10]. As they become more affordable, these numbers are likely to increase. The sales increase of mobile devices positively affected the mobile gaming market. The market reached to \$2.8 billion in 2013, which was only \$900 million in 2012 [11]. This also shows that in the future the popularity of mobile device games likely will increase [11]. The evolution of this industry, in terms of the games themselves and the devices on which they can be played anywhere at any time, reflects users' eagerness to play games—sometimes with several players whom they do not even know—just for fun. Ideally, then, this “crowd” of gamers could put their energies and abilities to use to solve real problems using a game interface, and software verification presents such a problem.

2.3 Formal Verification of Software

2.3.1 Definition

According to Kroening and Sharygina, formal verification is a method used not only in the hardware design but also in software design to find the defects that cannot be found by testing based approaches [12]. They illustrated that there are several ways of making

formal verification using different algorithms and bases. Li defined the software formal verification as “an act of using formal methods to check the correctness of intended programs” [13]. The author specified that “The verification is done by providing a formal proof on an abstract mathematical model of the program, with respect to a certain formal specification or property” [13].

Software verification aims to guarantee some correctness properties when running a software program. This process makes sure that the software does only what it is designed for with no unintended tasks, which might be malicious [14]. In terms of cyber defense, the latter case (i.e., making sure that the software does not perform any unintended tasks) is crucial [15].

2.3.2 Importance of Software Formal Verification

The widely used open source software are operating systems. Linux is one of the best-known and widely used open-source operating systems and it has many derivatives with different distributions. According to the report shown in [16], five-computer science researchers examined 5.7 Million lines of *Linux* source code in four years. They concluded that the *Linux 6* kernel code was better and more secure than that of most proprietary software [16]. Throughout the study, the researchers worked on the 2.6 *Linux* production kernel, which was used by Red Hat, Novell, and other popular vendors and found 985 bugs in 5.7 million lines of code. On the other side, according to Carnegie Mellon University’s CyLab Sustainable Computing Consortium, typical commercial closed source software has 20 to 30 bugs for every 1,000 lines of code, which means that *Windows XP* with 40 million lines of code has a number of possible bugs between 114,000 and 171,000 [15], [16].

Software bugs can cause serious problems. At first, systems using such software may stop working or fail to achieve what it is designed for, including space shuttle crashes, financial loss to companies, fatal mistreatment of patients, power outages in cities, and more [15], [17] in the past. If a military software system fails, the results can be grave. For example, during the First Gulf War, the Patriot air defense system failed to prevent an incoming missile because of a bug in the software and caused the death of 28 soldiers [18]. Another example of a software bug resulted in the death of 29 people in a Chinook helicopter crash [15], [19].

Table 2.1: Software Dependence of Military Aircrafts by Years

Weapon	Year	% of Functions Performed in Software
F-4 Jet Fighter	1960	8
A-7	1964	10
F-111	1970	20
F-15	1975	35
F-16	1982	45
B-2 Bomber	1990	65
F-22	2000	80

Furthermore, vulnerabilities caused by bugs can be exploited by adversaries. Hackers mostly use software bugs and *zero day* bugs to exploit systems. One recent and very most dangerous example from a global perspective is the heart bleed bug [20]. Hackers reached encrypted data by using this bug, exploiting the *OpenSSL* cryptographic software library (i.e., the main security provider) which has had the bug for a while. Hacking of military systems and vehicles are also possible. Hacking of military systems and vehicles is also very likely (i.e., unmanned vehicles [UVs] have been hacked in Afghanistan) [21]. One crucial step to detect bugs in software is through formal verification [15].

With the improvement of technology, both military and civilian systems have become more software dependent, and the importance of formal verification of software has increased. The experiments related to software verification show that there are one to five bugs in every thousand lines of code [22]. Dean stressed that one of the solutions to the bugs is formal program verification, which is the only way of verifying that a piece of software does not contain certain bugs [22]. In particular, as Table 2.1 [23] shows clearly, the software dependence of military systems has increased over time, which makes formal verification even more urgent [15].

Considering the increase in the use of technology in daily life and the number of bugs in these technology systems, it is clear that it is important to improve the verification process to make these systems safer [15].

2.3.3 Difficulty of Software Verification

Although the formal verification technique is an approved method, it is not scalable for use in complex software written for advanced military systems [3]. Moreover, formal verifica-

tion of software is a very expensive process. According to [22], because computers cannot yet perform complete software verification, the total cost of the process can increase up to a hundred times because specially-trained engineers must perform the verification manually, which takes a long time.

To reduce the number of bugs in software, formal verification has to be performed in an improved and faster fashion. However, verification is a complex process that can be performed by limited number of experts and this leads to insufficient resources to verify many software products [24]. Also, while the lines of code produced in the world has been increasing rapidly, the number of experts qualified for verification phase has not followed in the same trend [24]. A study [25] shows interesting code production figures (i.e., there are 6 million software developers in the world and they produce 300 million lines of code weekly, and up to 15 billion lines of code yearly). Moreover, even if every verification expert in the United States worked only on the source code for *Windows 8* to verify and find 25 predefined vulnerabilities, they would not finish the process in more than 30 years, proving how time-consuming the process can be [24]. There are several types of software such as operating systems, commercial off-the-shelf (COTS) applications, and more. When we look at the Table 2.2 [15], [26], the size of the software that needs to be verified is incredible.

2.4 Crowdsourcing

2.4.1 Concept

In 2006, Yuen and Leung illustrated that Jeff Howe introduced the term crowdsourcing to the cyber world [27]. Although there are some historic examples, crowdsourcing is a contemporary term that emerged in the beginning of the new millennium and it has been increasingly effective being enabled by the Internet's ability to connecting people. Crowdsourcing, however, existed long before the Internet. The Longitude Contest in 1714 invited the general public to submit designs for a navigational gadget for sailors; the competition for the design of the Toyota logo in 1936, and the Sydney Opera House architecture project competition in 1955 are all examples of sourcing a problem to the crowd [28]. In all of these cases, the underlying assumption was that a large pool including professionals and non-professionals was more likely to produce an effective solution than a small number of

Table 2.2: LOCs of a Group of Software

Name	Lines of Code (LOC)
Windows XP	40 M
Windows 7	40 M
Linux 3.1	15 M
Mac OS X 10.4	86 M
Debian 5.0 (all software in package)	324 M
Android OS	12 M
Microsoft Office (2013)	45 M
F-22 Raptor Jet Fighter	1.7 M
F-35 Fighter	24 M
Patriot PAC-3 Missiles	Close to 2 M
US Army's Future Combat System	63.8 M
Hubble Space Telescope	2 M
Google Chrome 2011	5.4 M
Boeing 787 Dreamliner	6.1 M
Firefox	9.7 M
Chevrolet Volt (Electric Car)	10 M
Apache Open Office	23 M
MySQL	12.5 M
Software in typical new car, 2013	100 M
Healthcare.gov	500 M

subject matter experts. Furthermore, crowdsourcing is built on the premise that humans can be more useful than computers at solving problem. In 2003, Luis von Ahn and his companions were the first to use the term “Human Computation” when referring to humans performing computational jobs that are difficult for computers to process (i.e., image interpretations) [27]. Early crowdsourcing examples in the form of contests proved that a large group of non-professionals can be very effective at problem solving, sometimes even better than computers, and they enjoy problem solving when it is fun or competitive, when it might result in an award or cash prize, or when the participant might earn special recognition. Modern examples of crowdsourcing apply this concept and tap these motivations to solve ongoing problems.

2.4.2 Examples

Duolingo

In the research about Duolingo, a language-learning website and a crowdsourcing project, Garcia stated that machine translation is not good enough [1], [29]. Furthermore using professionals can be too expensive. At this point Duolingo comes up with a solution. They use the effort of language learners to translate websites into several languages, which results in much better translations than those done by machine [29].

Topcoder

Lakhani, Garvin, and Lonstein described the firm Topcoder as a software company that creates high-quality crowd code solutions so that programmers do not have to provide the code themselves [30]. They select their coders and codes from online competitions. All coders have to have a profile in the Topcoder system. Topcoder uses a type of ranking for coders who have created a profile. One of the incentives to register with Topcoder is money. Between 2001 and 2009, Topcoder paid more than 20 million dollars to its crowd-coders [30]. The incentive is directly related to the crowd-coding output. In addition to money, the coders assert that their Topcoder rating is very important to their career, because it reflects their knowledge, skills, and a potential promotion at future companies [30].

Amazon Mechanical Turk

Amazon Mechanical Turk is a crowdsourcing system that links employers with employees who are capable of providing simple coding solutions for complicated computer tasks [31]. On the website, the tasks their employees are capable of performing include: identifying objects in a photo or video, performing data re-duplication, transcribing audio recordings, or researching data details [31].

2.4.3 Motivation

Malone et. al indicated that it is important to understand how crowds can achieve difficult tasks and create high-quality results in an electronic environment, in the absence of any strongly-centralized control unit like Wikipedia, Linux and others [32]. They generally named this new type of electronic organizations as “collective intelligence.” To understand the crowd-coding example, they focused on the goal, staffing, structure/process, and incentives. Human motivation has been a research topic for centuries. In their study, they

selected three of the most high-level incentives for human motivation, including money, love, and glory [32]. The researchers claimed that in collective intelligence organizations, the main motivation emerges from love and glory, while money is the most powerful motivation source in traditional organizations [32].

2.5 Crowd-Sourced Serious Games

2.5.1 Concept

Electronic games can benefit humanity by transforming game players' fun efforts into solving important problems. Ahn stated that people from all around the world consume billions of hours playing computer games, which translates into potential solutions for many problems [33]. He came up with an idea that these efforts could be used productively to solve tasks that are difficult for computers but easy for humans [33]. One potential method would be a game with an algorithm that transforms effort into meaningful input for the process of problem solving [33].

2.5.2 Examples

Foldit

Foldit is a project that aims to understand the structure of the proteins to find the cure of protein-based diseases such as AIDS, cancer, and Alzheimer's [34]. According to the project's website, a protein can fold and create astronomical types of structures. Finding a cure for protein-based diseases requires identifying a protein's most stable structure. Players use their puzzle-solving abilities to reduce the trials, and thus, identify the most stable protein structures faster than computers [34], [35]. After playing *Foldit* for only three weeks, contributors were able to decipher an AIDS-related structure that had previously been unresolved for 15 years [35].

Eyewire

Eyewire is a neuroscience-based project that aims to gain the power of non-expert players for solving complex problems regarding the nervous system [36]. In the game, volunteer participants compete with one another by composing neurons in an area of the mouse eye to help scientists understand how the brain handles visual data [36]. Regarding this project,

Marx indicated that nearly 82,000 non-expert players in all ages, defined as citizen scientists, played the game. These players assisted in testing the artificial intelligence algorithms that allow computers to map neurons in the future [37].

Phylo

Phylo is a puzzle game designed to help scientists solve multiple sequence alignment (MSA) problems, thus assisting with genetic disorders that may be the cause of many diseases [38]. Having analyzed over 12,000 players of *Phylo*, researchers estimate that more than 350,000 issues were solved with high accuracy [38].

2.5.3 Motivation

Cooper et al., presented the view that incorporating with the efforts of non-experts in scientific discovery may be successful, but the consequence of the scientific discovery game is obscure for the experts and game developers as well [39]. This means that this type of games like *Foldit* do not have a specific end in the gaming process, which could be an incentive for players to see the outcome or to complete the game. So, in *Foldit* the developers tried to motivate gamers to discover the best possible protein structures which are also unknown by the developers [39]. For this reason, they decided to use the competition as a motivation source. In the game they created a scoreboard and announced the top scoring players. Additionally, they encouraged group scoring and let people to make groups and compete as a group [39]. Finally they asserted that unlike the rest of computer games in that type of scientific discovery the only goal that the developers think is not the entertainment. They try to make people, who have no specialty other than being a computer user participate in a scientific problem solving process. Finally, the design process of scientific games is different from the design of ordinary computer games in several aspects, one of which is incentivizing players [39].

2.6 Crowd Sourced Formal Verification

2.6.1 Verigames

Verigames is a web portal that is currently serving five online browser games that are being used by DARPA's Crowd Sourced Formal Verification (CSFV) project, and the purpose of the project is to use the game players' effort in the formal verification of military software [3]. The portal, operated by Topcoder Inc., combines the work of the game developers

from different organizations including universities, professional game developers and Top-coder [40]. Players have to confirm that they are older than 17 to play the games. However, the players have the option to play anonymously. Players can also terminate their membership whenever they want [40].

Circuitbot

Circuitbot is a strategic resource management game. In the game, players have to manage resources like energy, water, food, fuel, robots, etc., to colonize different planets or stars [40]. To achieve colonization and successful resource management players need to build new facilities producing some resources while consuming some others. To build each facility a different number of robots lands on the planets. Players have to activate the links between robots in logical order to gain points [40].

Stormbound

Stormbound is a puzzle game whose story is based on defeating a magical storm on a moon belonging to an artificial planet named Aeryth [40], [41]. In the game, players educate a semi-spiritual and semi-physical entity named Gola by defining the correct relationship of two given patterns [40]. This action charges Gola's power source and helps it to defeat the storms.

Xylem

Xylem is a game based on solving the code of the newly discovered plants in a recently found island of Miraflores [40]. In the game, the players solve some mathematical puzzles to define the new plants, and score based on the results of their solution [40]. The game is only playable on Apple iPads for now and is available at the application store.

Ghost map

According to its website *Ghost Map* is a puzzle game, in which players are trying to unlock a network [40]. The players operate *Ghost Map* and move forward in the game by solving the puzzle's structure [40].

Flow Jam

Flow Jam is a game that aims to remove jams in a rudimentary network design in order to expand electric flow on given links. Players advance by finding the correct relationship

between links and passages [40]. There are several levels in the game with different widgets, links, and jams. Players try to reach the limit of points to pass from one level to the next [40].

2.7 Basic of Game Analytics

Game analytics is the examination and interpretation of data that game makers collect during the electronic game playing process. The purpose of collecting this data is to increase revenue. Electronic game developers use game analytics because they have to learn more about their games and players [42]. This information is essential because new kinds of games, such as online social games and business models, such as free-to-play games, which do not require an initial payment but offer some purchase opportunities during the game, have emerged [42]. In these types of games, developers can collect real-time data, analyze the data. Based on the results of data analysis game developers can modify the weak and strong parts of the games to keep the players' demand constant or to increase the demand [43]. Furthermore, developers can also release new patches or alter the game code on the server, meaning they do not need to worry about the initial completeness of their game [43]. Overall, game analytics is a methodological tool for game developers to improve their games.

While competition in the electronic gaming market is a challenge, the modifiable nature of new online games is an opportunity for electronic game makers. According to Canossa, El-Nasr, and Drachen, creating a lucrative electronic game is difficult because there are many players in the market trying to attract customers and many games for all age, social, interest, and gender groups on different gaming platforms [44]. Other hand, Canossa et al. indicate that new types of games and business models depend on better understanding of the players' behavior to increase the revenue [44]. The researchers add that game analytics is the way of achieving that understanding. Moreover, regardless of complexity, each modern electronic game has to be tracked and needs to be modified based on results to increase revenue [45]. However, the problem is to decide which data is useful, whether it is worthy of analysis, and how it will be analyzed to benefit from analytics effectively [45].

El-Nasr et al. define analytics as “the process of discovering and communicating patterns in data towards solving problems in business or, conversely, to make predictions for support-

ing enterprise decision management, driving action, and/or improving performance” [42]. Game analytics uses analytics during game producing, and its goal is to help decision makers make the best choice [42]. These researchers note that game developers import different methods from other fields, such as analytics, to take a higher share of data. Consequently, game analytics benefit from many other fields, such as statistics, data mining, and business intelligence [42].

Collecting metrics from players, analyzing the metrics, and modifying the games based on the results, is a cyclic process, and that is a must-do activity for game developers to satisfy their customers [43].

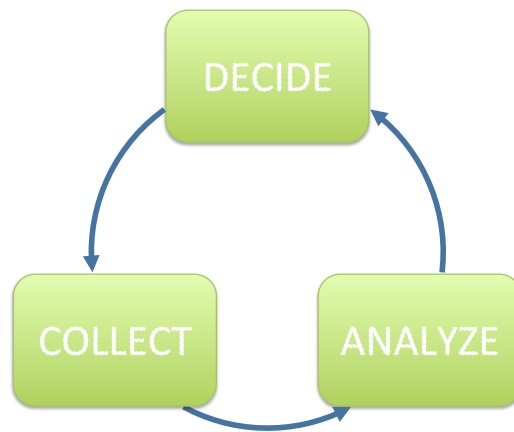


Figure 2.1: Cyclic Behavior of Game Analytics

2.7.1 Pre-Data Collection

The cooperation and coordination of different professionals with different interests (e.g., the project manager, designer, coder, customer researchers) is very important in the game development process [44]. Canossa et al. add that this cooperation between different professionals leads to distinct variables relating to their subprocess of interest [44]. For example, some of the stakeholders want to track the data about production, while some of them want to follow the data on players’ attributes, monetization of the game, and more [44]. Therefore, in the pre-data-collection phase creating a common language between stakeholders is important for success.

In the pre-data-collection phase, selecting the right metrics or data is important because

the complexity and cost of the analyses increases relative to the amount of data and metrics [46]. On the other hand, there are no firm rules on the type of metrics to be collected, but deciding some core ones after discussion with all stakeholders has a great impact on success [47]. Although it is very important to define metrics initially, it is possible to add or remove metrics during the data collection phase, which comes from the cyclic behavior (shown in Figure 2.1) of the game analytics [43].

2.7.2 Data Collection

According to Fields [43], the ability to collect data from online games lets developers follow the behavior of both games and players. In online games, players' high expectations as customers and the weak or strong parts of the games, can be measured remotely [43]. Game developers use game telemetry to measure some aspect from players spread all over the world [42]. Game telemetry is data collected from players at any remote place by using the Internet or any other network [42]. Drachen et al. indicate that the telemetry could be about anything in the gaming process, such as players' interaction with a game, payment system, or bug fix rates [42].

Taking the feedback from players is very important [43]. Canossa stated in an interview with game analysts from Junebud, a game developer company, that the company has recorded nearly all of the activities of its players, but monitors some of the most important ones, such as log in and log out times to know players' playing times and intervals [48]. Canossa named the process of taking data from remote players as telemetry and the categorized-data collected as game metrics [46]. Additionally, according to Fields game developers put a piece of code into the game that gathers and sends the game metrics data to the developers, which is called instrumentation [43]. On the other hand, Drachen et al., show that there are two ways to collect that data; one is to embed code into the game, and the other one is to get it directly from game servers [47].

2.7.3 Post-Data Collection and Analysis

In the post-data collection phase, analysts and developers work on the raw telemetry data collected in the previous phase. Game developers can use telemetry data in different ways to point out issues or triumphs in games [48]. For example, analysts from Junebud, found that the players could not progress in the game *Milmo* by tracking players' sessions, and

they modified the game. As a result, they have never seen the problem again. Moreover, the company is using the telemetry data not only to increase the amount of money gained from a single game, but also to conduct further researches on acquisition, customer services, and retention [48]. In one game, they tested the attractiveness of four different character selection screens in parallel; based on the metrics defined, they received two percent more returning users [48].

Once developers obtain telemetry data, it has to be processed [45]. According to Drachen et al. to do that, storing the data in a database is essential. Moreover, cleaning the data and organizing it should be a step before analyzing the data [45]. The authors claimed that after preparing the data for analysis, game developers select variables and metrics that they already discussed in the pre-data-collection phase [45].

Game metrics, derived from raw data, is a meaningful measure of anything about an electronic game. In a broader definition, they define game metrics as “a quantitative measure of one or more attributes of one or more objects that operate in the context of the game” [42]. Following are some of the most common game metrics.

Commonly Used Game Metrics in Analysis

- **Daily Active Users (DAU):** Fields indicates that daily active users (DAU) is the number of players, which can be calculated per unique user as well, logged on in one day [43]. Additionally, it can be calculated by counting all initiated playing activity, disregarding the identification of the player [43]. Fields also states that being active in a game might have a different meaning based on the type of the game. DAU might be a misleading metric because it does not count the time spent on the game, which means that a player spending one minute is the same as one spending an hour. On the other hand, it may be a good tool to measure the popularity of the game [43].
- **Monthly Active Users (MAU):** Monthly active users (MAU) is the number of players counted in one month [43]. Fields notes that this metric also can be collected for unique or non-unique users. Although MAU is a useful and important metric to show players' attraction to games, this metric does not show player engagement and is not solely enough [49].
- **Engagement Rate (ER):** If MAU is counted for non-unique users, its ratio with daily active unique users indicates the fraction of players who enjoy playing a game [43].

According to the Fields, the ER gives significant feedback about a game's initial success [43]. If a game reaches a high ratio, then it has achieved the hardest step, attracting the players. At that point, Fields suggests increasing the number of registered players through advertising [43].

$$ER = (DAU/MAU) * 100 \quad (2.1)$$

- Conversion Rate (CR): The conversion rate is the ratio of the players who spend money in free-to-play games, which do not require initial payment, but sell items, use virtual money, and offer gold-type items during the playing process [43].
- Average Revenue per User (ARPU): Average revenue per user (ARPU) is the ratio of total revenue to the number of players in a defined time interval such as weeks or months [43]. ARPU can show the expected revenue per player. In addition, if the cost of acquiring a new player is less than the ARPU, the advertisements and marketing techniques can be used to increase the number of players and the amount of profit [43].
- Life Time Value (LTV): Life time value (LTV) is the amount of money a player spent in a game [43]. Fields states that people typically play games for a period and give up playing. While the LTV for online and free-to-play games is the amount of money a player spent during the playing lifetime before giving up, in retail computer and console games, LTV is the price of the CD/DVD [43].
- Retention Rate: Retention rate is the ratio of returning players after first play [43]. According to Fields, it is a sign of a game's addictiveness. He adds that ER can roughly reflect the retention rate and can give a basic feedback about the rate. In addition, it can be improved by making the game more attractive and offering prizes for success [43].
- Entry Event Distribution (EED): Entry event distribution (EED) is the first action of a player after logging in or starting to play the game [43]. EED can reflect the motivating element of the game. For example, if most of the players are checking the leaderboards, then it demonstrates that competition is a strong energizer for the players [43].
- Exit Event Distribution (XED): Exit event distribution (XED) is the last action a player takes before logging out or leaving the game. XED is an important metric for

developers, because XED demonstrates the problem areas in the game [43].

Data Mining and Analysis

Big data is a problem for game developers because the amount of data may be enormous. Zynga, a game developer company, collects 15 terabytes of data each day [50]. El-Nasr and Canossa stated that the firm stores 1.4 petabytes of data, which requires an enormous data warehouse [50]. The amount of data gathered for several purposes for any reason has reached an immense volume given the increased technology in the information age [45]. Electronic game developers are collecting huge amounts of data. According to Drachen et al., existing games vary from simple to very complex. Selecting some core metrics to collect decreases the cost of analysis and makes analysis easier. For example, for the beta release of *Halo Reach* 2.7 million players played more than 16 million hours and created terabytes of data [46]. Moreover, game developers need enough resources to deal with big data [46]. As Canossa observes even a simple query in the database could take too much time [46].

Data mining is a way of obtaining the meaningful data [45]. Canossa states “Analyzing game-related data, at its core, is a process that involves being able to articulate knowledge and meaning from apparently meaningless data” [46]. The next step is the analyzing the useful data based on the purpose [45]. The authors name this process as “separating gold from rock in data mining results” [45]. Drachen et al. listed and defined eight of the most common data mining methods, which we list here [45]:

- Description shows the behavior of patterns mostly by using graphical methods such as bar charts. The authors state that before starting complex analysis, it would be beneficial to make a description.
- Characterization is obtaining data about a group by creating a characterization rule such as the players who passed one level in less than five hours.
- Discrimination is making a comparison. Comparing the most popular items purchased by two different age or gender groups is an example.
- Classification is creating groups with using common properties. Grouping players based on behavior to see if they would return as paying or non-paying players is an example.
- Estimation is making an estimation based on the current data obtained. It can be used

to estimate the purchasing behavior or time when a player will give up playing the game. Regression and correlation are the basic statistical methods to make an estimation. This method also shows the relation between two variables such as playtime and money spent.

- Prediction is a way of forecasting. There are many methods of doing prediction ranging from basic statistical methods to complex neural networks. The authors state that it is the most widely used analysis method.
- Clustering: Clustering is also a way of making classifications. Differently from classification itself, in clustering an algorithm groups the objects by gathering the data that are related under a group without using defined metrics.
- Association is finding a related attribute. Finding two players who take actions together can be an example.

Prediction Analysis

Prediction is very important for increasing the revenue for online games, because it gives developers a chance to modify their games before losing money or players [45]. The authors illustrate that regression analysis is the main statistical technique for prediction [45]. They also note that making many predictions by using different types of data, and interpreting the combination of predictions can increase the accuracy of the prediction [45].

Mahlman et al. made a predictive experiment on the *Tomb Raider: Underworld* game. The authors define the purpose of their analysis as “to investigate if it was possible to develop a model that could predict when a player would stop playing the game, based on their early play behavior” [51]. The hardest part of the experiment was dealing with the data of more than 200,000 players collected in two months [51]. The size of the data was almost 100GB [51]. They selected 10,000 players as the sample space for initial research. Next, they defined their metrics that they thought were most relevant, such as playing time, number of deaths, help-on-demand, causes of deaths, and the number of rewards collected [51]. Finally, they classified the players based on the levels the players had completed and ran the analysis using a data analysis tool. The authors concluded that it is possible to create a good model and predict player behavior using regression methods [51].

2.8 Summary

In this chapter, we presented the concepts of the thesis. Electronic games have been attracting people and evolving over time. This evolution merged with the idea of crowdsourcing and gave rise to CSSGs. We introduced a group of CSSGs including VeriGames in this chapter.

Additionally, we provided general information about game analytics, which is the combination of a series of cyclic processes. We classified these processes as pre-data-collection (decide), data collection (collect), and post-data-collection (analyze) phases. Before collecting data, there is a need to establish a common language between stakeholders of the game developing organization. In addition, having an agreement on the game metrics has a positive effect on effectiveness. The data-collecting phase is based on the concept of telemetry, which means collecting data from remote players all around the world connected to the Internet. Finally, we emphasized the data-analysis phase, which involves converting raw data into a meaningful output that will help all stakeholders to improve their games. In these phases, we listed some of the most common game metrics that can reflect valuable information about players' attitudes. In addition, different metrics can be generated by mining data. However, big data can be a problem for analysis.

CHAPTER 3: Related Works

3.1 Introduction

This chapter presents the previous researches on the effectiveness of CSSGs.

3.2 Previous Researches

CSSG developers have often provided the total number of registered participants as an indicator of game success. For example, in a November 2013 press release, *Duolingo* claimed 14 million registered users. *EyeWire* researchers stated in a recent paper [2] that more than 100,000 registered players from more than 130 countries had contributed to their experiment [49].

Other CSSG developers have used a measure of work performed to assess the contributions of their crowd toward the motivating cause. The creator of *Phylo*, a CSSG whose players solve puzzles to help find solutions to genetic disorders, reported obtaining a total of 254,485 completed puzzles (generated by $\sim 12,000$ registered players) in the first seven months of deployment [38]. *The Malaria Training Game* (MTG), created for advancing the concept of tele-diagnosis of diseases, was able to screen more than 1.5 million red blood cell images for malaria infection in less than four months, with the help of 2,150 people from 77 countries [52]. Comparative studies are also applicable in some cases. One such study concluded that *Duolingo* is more effective than *Rosetta Stone* or college classes in helping people to learn a foreign language [53].

Finally, the literature on CSSGs repeatedly describes the presence of and the key roles played by a few whales in the crowd. For example, according to one study [38], the top ten percent *Phylo* players (in terms of their skills at solving puzzles) participated in nearly 80 percent of the completed puzzles [49].

Common to all these studies is that their data and conclusions are specific to an individual game. The general effectiveness of CSSGs and the methodologies for applying the classic commercial game analytics to this new genre have not been examined. This observation is

not unexpected, given the relatively short history of CSSGs [49].

3.3 Summary

In this chapter, we focused on the previous researches on CSSGs effectiveness. We could not find much study on measuring the effectiveness of CSSGs, because CSSGs are an emerging genre. The researches we found are the ones written by game developing teams and mostly details the purpose of the CSSGs instead of measuring the effectiveness of CSSGs. In addition, we could not find any study on CSSGs, which used game analytics to evaluate the games.

CHAPTER 4:

Methodology

4.1 Introduction

This chapter proposes the methodology used in the thesis. Initially, we will show the datasets in three groups, which are belong to traditional games, other CSSGs, and VeriGames. Next, we will present the metrics used in the analysis. We will also show how we generate the metrics. This part is very important because metrics are the core of the thesis.

4.2 Data Collection

For this research, no data has been collected. Two VeriGames developers directly provided the data. Therefore, we did not have a control over which types of data were collected. Instead, we asked game developers for data containing players' identification and time stamps related to basic activities such as login and logout. Hence, the metrics of the analysis were derived.

4.2.1 Data Sets

In the thesis, datasets belong to two types of games. The first dataset, from `gamesbrief.com` [54], includes daily active users (DAU) and monthly active users (MAU), and engagement rate (ER) statistics for mobile and social online games (Tables 4.1, 5.1, and 5.4) which have been compiled from various resources [55], [56]. The data for each game shown in Table 4.1 includes averages calculated during several months within 2011 and 2012, and will be explicitly demonstrated in Chapter 5.

The second dataset consists of players' session and productivity data for two games from `verigames.com` (referred to as VeriGame A and VeriGame B in the rest of the paper). These data were obtained directly from game developers. VeriGame A data have information in four relational data tables. VeriGame A data table including session information has more than 30K entities. In contrast, VeriGame B's data are in a single table and in a different structure, which has more than 100K entities.

Table 4.1: List of Traditional Games Used in this Thesis

Zyanga*
Storm8*
Glu Mobile*
Angry Birds
Temple Run
Stardom
Deer Hunter
Junkies
Triple Town
Parallel Kingdom
DeNa*
GREE*

*Game developer/operator.

The DAU and MAU statistics for these two games are shown in Tables 5.2 and 5.3.

Table 4.2: Summary of CSSG Data Used

	Collection Period	Total Users
VeriGame A	1 Dec 2013 - 9 May 2014	1475 Reg. 8399 Anon.
VeriGame B	1 Dec 2013 - 17 Mar 2014	717 Reg. 7029 Anon.
EyeWire	Since Dec 2012	Over 100K
Foldit	Since May 2008	Over 500K
Phylo	Dec 2010 - Jun 2011	Over 12K
MTG	May 2012 - Aug 2012	Over 2K

Data of other CSSGs were gathered from the literature including *EyeWire* [2], *Phylo* [38], *Foldit* [34], and *The Malaria Training Game* (MTG) [52]. The sizes of these additional data sets are shown in Table 4.2, and the data will be referred to in Chapter 5, as well.

4.3 Post-Data Collection

Initially, the datasets were converted into an appropriate format for database import, because as mentioned before each game developers provided the data in different formats. While one set of data was in comma-separated values, the other one was in JavaScript ob-

ject notation. The table in JavaScript object notation was converted to comma-separated values. Second, the data were imported into a *MySQL* database. VeriGame A had data in four tables. Game developers used two tables for player identification and other tables to store player activities. VeriGame B developers used a single table for both players' identities and activities. During this process we contacted the game developers several times to understand the structure of the data. Actually, the hardest part of the research was understanding the data structure.

Before starting analysis the data were cleaned. Therefore, data recorded before 1 December 2013, which is three days prior to media release of VeriGames, were deleted. The data before media release most probably belongs to in-team game players or test accounts and may decrease the accuracy of analysis. Moreover, VeriGame A team sent an excluded player list which was deleted from the data, as well.

Listing 4.1: SQL Command to Delete Data Before 1 Dec 2013

```
DELETE FROM table
WHERE sessionStartTime < '2013-12-01' ;
```

Listing 4.2: SQL Command to Delete Excluded Players

```
DELETE FROM table
WHERE playerId IN ("playerId1", "playerId2", ....);
```

4.3.1 Metrics

It is relatively simple to measure productivity of retail electronic games: count DVDs/CDs sold, multiply with sell price, and compare with the cost of producing the game. Productivity in the commercial online gaming market (with a similar ecosystem to that of CSSGs) is a much more complex function of purchase price (\$0 in many cases) along with in-game purchasing and subscriptions. Theoretically, a player can spend zero to infinity dollars. In other words, while players traditionally spent a constant amount for a retail game, their spending can significantly exceed that amount for free-to-play games [57]. Due to these new pricing paradigms, not only maximizing the number of players, but also transforming free-players into paying-players are important issues for online games [49].

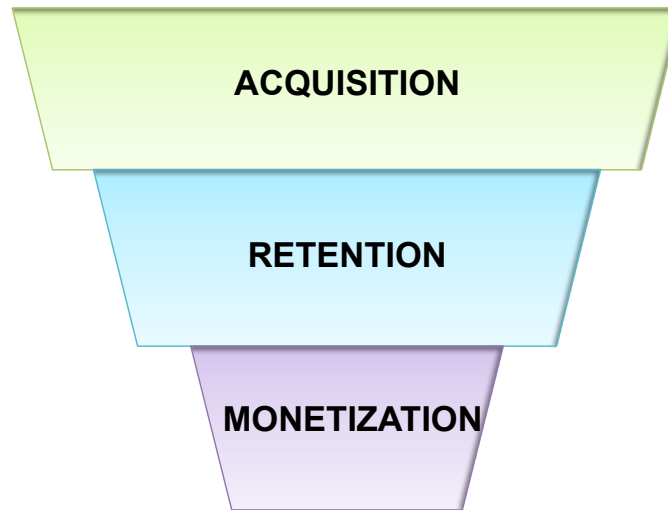


Figure 4.1: A Simple Game Monetization Funnel Widely Used for Free-to-play Games

Because of fluctuations in player spending over time, it is vital that game developers track players' attitudes towards particular games. Two of the most common metrics to measure players attitudes towards games are daily active users (DAU) and monthly active users (MAU) [58], [59]. According to Fields [58], DAU is the count of unique players in a day, and MAU records either unique or non-unique players in a calendar month. In our research we counted unique users for both DAU and MAU. In addition, we used weekly active users (WAU) to count unique users in a seven-day period [49].

DAU and MAU are important metrics that free-to-play game developers firmly follow. Obviously, players are vital for games and companies to make money from them. DAU and MAU show players' initial involvement in games. DAU and MAU are largely related with the first step, player acquisition, in Figure 4.1 [60]. Game developers essentially make the largest marketing expenditures on that phase to reach maximum DAU and MAU. In other words, each player has a cost, and DAU and MAU can be increased by spending money for marketing.

New attributes which show the number of days, weeks and months were added to data tables to find daily, weekly, and monthly metrics. This helps to simplify *SQL* commands. *MySQL* day and time functions, *DAYOFYEAR*, *WEEK*, and *MONTH* returned the proper information of used date time.

Listing 4.3: SQL Command to Fill Day Week and Month

```
UPDATE table
SET day = DAYOFYEAR(sessionStartTime);
# week = WEEK(sessionStartTime,2)
# month = MONTH(sessionStartTime)
```

Listing 4.4: SQL Command for DAU

```
SELECT day, COUNT(DISTINCT playerId)
FROM table
WHERE registered IS TRUE
GROUP BY day;
```

Listing 4.5: SQL Command for WAU

```
SELECT week, COUNT(DISTINCT playerId)
FROM table
WHERE registered IS TRUE
GROUP BY week;
```

Listing 4.6: SQL command for MAU

```
SELECT month, COUNT(DISTINCT playerId)
FROM table
WHERE registered IS TRUE
GROUP BY month;
```

Engagement Rate

Although DAU and MAU are very useful metrics, as independent values they are insufficient to represent a game's potential because they count all players, including non-returning one-time players, without capturing level of user engagement [58]. In the second phase of Figure 4.1, game developers expect returning players after the first interaction, because good games attract players and retain them. Accordingly, a metric is required to show player retention by games. By examining the relationship between DAU and MAU we are

able to quantify the ER of players [49]. If a game cannot attract players in early interactions, which means a low ER, it will lose players who were gained by marketing. Formally, we define ER as the DAU to MAU ratio:

$$ER = (DAU/MAU) * 100 \quad (4.1)$$

Once DAU and MAU metrics are exported to *MS Excel*, it is easy to find ER. ER was calculated for each day by dividing a single day DAU by the MAU of the calendar month for that day. Therefore, there is an ER for each day. The average of all days' ERs in a month gives the ER of that particular month.

This metric represents a game's "stickiness," which also roughly expresses the games' ability to retain players. In addition, ER may provide an indicator about the long term success of a game [58], [59]. If a game has low ER, the game mechanism should be changed to increase ER. It shows that the players do not enjoy game and give up playing. Expressly, marketing does not affect ER the way it does DAU and MAU [49], [58].

Once a good ER is achieved, the next step is monetization for free-to-play games. The non-paying players have to be transformed into paying players. However, CSSGs need different productivity metrics other than money. These productivity-related metrics are examined in the following subsections.

Whale Effect Graph

As was shown by Pareto's 80-20 rule, which basically claims there is an unbalanced situation between input and output, players' spending is not uniformly distributed in free-to-play online games [61]. A small subset of players called *whales* (a term borrowed from the casino gambling industry) far outspend average players. Jesse Divnich has defined whales as the top 5 percent of spenders [62]. He considers whales to be players who spend more than ten dollars monthly for online mobile games. While that does not sound very impressive, it constitutes a large percentage of the total revenue for online games. For example, a director of *Clash of Dragons* declared 40 percent of the in-game purchases were made by only 2 percent of players [63]. A recent report about monetization in mobile games also shows that 50 percent of revenue comes from 0.15 percent of players [64]. At this point a

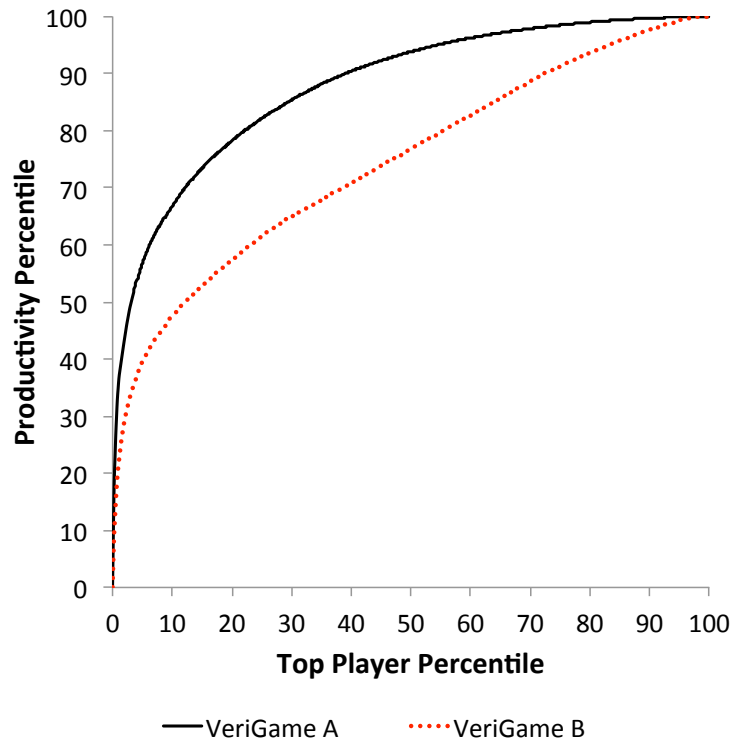


Figure 4.2: Whale Effect Graph (WEG)

standardized definition of a “whale” has not been established, and each game determines which players are whales based on a different standard [49].

To study the effects of whales on the VeriGames and CSSGs in general, a Whale Effect Graph (WEG) was proposed an example of which is shown in Figure 4.2. In this graph the x-axis shows the cumulative percentile of players sorted by productivity, and the y-axis shows the cumulative percentile of overall game productivity. In other words, any point on the curve shows the percentage of contribution to the overall productivity produced by the selected fraction of the most effective players. Therefore, in contrast to focusing on either an arbitrary fraction of top players or the cumulative distribution of players based on their productivity, a WEG provides a *complete view* of how players of different productivity levels contribute to the *overall productivity* of the game [49].

Listing 4.7: SQL Command for Player Productivity

```
SELECT playerId , SUM(productivitymetric) AS productivity
FROM table
WHERE registered IS TRUE
GROUP BY playerId
ORDER BY productivity DESC;
```

In the case of VeriGames, the goal is not monetization, so in order to measure productivity we were required to choose metrics other than money. Based on advice from the developers of the two games we chose to quantify productivity using the assertion count for VeriGame A, and the game score for VeriGame B. Since these two metrics are measured on different scales, the values were normalized, and the results are presented in Chapter 5 using percentile graphs [49].

Session Times and Counts

The ER metric, as defined earlier has a limitation in that it cannot capture the magnitude of total player activities. For example, $ER = 1$ even if only five players remain for a game, as long as they are active every day of the month. Therefore, we also use the aggregate session time (ST) and session count (SC) metrics to analyze CSSGs, as done in prior work [45]. ST is the amount of time a player interacts with a game until leaving. ST was counted as hours in this thesis. SC shows how many times a game is played. We measure ST and SC over different time intervals such as weekly (WST, WSC) and monthly (MST, MSC) [49].

These game-play metrics are closely related to the game productivity and whale effect. Recent research shows that while paying and non-paying players have an average WST of about four hours, whales typically spend close to twelve hours gaming each week [49], [62].

The data for both VeriGames had time stamps for players' login and logout activities. An attribute was added to each table that shows session time in hours, and this attribute was filled by finding the difference of logout and login times.

Listing 4.8: SQL Command to Fill Out Session Time Attribute as Hours

```
UPDATE table
SET sessionTime= TIME_TO_SEC
    (TIMEDIFF(loginTime , logoutTime))/3600;
```

Listing 4.9: SQL Command to Generate Session Time

```
SELECT playerId , SUM(sessionTime) AS session
FROM table
WHERE registered IS TRUE
GROUP BY playerId
ORDER BY session DESC;
```

Listing 4.10: SQL Command to Generate Session Count

```
SELECT playerId , COUNT (playerId) AS sessionCount
FROM table
WHERE registered IS TRUE
GROUP BY playerId
ORDER BY sessionCount DESC;
```

Listing 4.11: SQL Command to Generate Required Metrics for WEG in Player Productivity Order

```
SELECT playerId ,
    SUM(productivityMetric) AS productivity ,
    SUM(sessionTime) AS sTime ,
    COUNT (playerId) AS sCount
FROM table
WHERE registered IS TRUE
GROUP BY playerId
ORDER BY productivity DESC;
```

4.3.2 Prediction Analysis

As discussed in Chapter 2, prediction analysis is one of the game analytics methods. In Chapter 5, regression analysis will be used for prediction analysis. The dependent variables

will be the productivity metric. As mentioned before, it will be the number of assertions (NA) for VeriGame A and score (S) for VeriGame B. Independent variables for prediction analysis are Active Users, Session Time, and Session Count. In addition, productivity metrics will be calculated on a daily and weekly basis. A complete list of metrics shown in Table 4.3. These metrics were selected because they present valuable information about games' success. It is also easy to produce those metrics with simple SQL queries from a database.

Listing 4.12: SQL Command to Generate Daily Metrics for Prediction Analysis

```
SELECT day ,
       COUNT(DISTINCT playerId) AS dau ,
       SUM (sessionTime) AS dst ,
       COUNT(playerId) AS dsc ,
       SUM(productivityMetric) AS dna # or ds
                                     #for VeriGame B
FROM table
WHERE registered IS TRUE
GROUP BY day
ORDER BY day ASC;
```

Listing 4.13: SQL Command to Generate Weekly Metrics for Prediction Analysis

```
SELECT week ,
       COUNT(DISTINCT playerId) AS wau ,
       SUM (sessionTime) AS wst ,
       COUNT(playerId) AS wsc ,
       SUM(productivityMetric) AS dna # or ds for
                                     # VeriGame B
FROM table
WHERE registered IS TRUE
GROUP BY week
ORDER BY week ASC;
```

Table 4.3: A List of All Metrics Used in the Thesis

Daily Active Users (DAU)	The number of unique players in a day
Weekly Active Users (WAU)	The number of unique players in a week
Monthly Active Users (MAU)	The number of players in a calendar month
Engagement Rate (ER)	The ratio of DAU over MAU
Daily Session Time (DST)	The total time a player played the game in a day
Weekly Session Time (WST)	The total time a player played the game in a week
Monthly Session Time (MST)	The total time a player played the game in a month
Daily Session Count (DSC)	The duration between login and logout counted as a single session. Count of sessions for a player in a day
Weekly Session Count (WSC)	Count of sessions for a player in a week
Monthly Session Count(MSC)	Count of sessions for a player in a calendar month
Daily Number of Assertions (DNA)	Daily productivity metric for VeriGame A
Weekly Number of Assertions (WNA)	Weekly productivity metric for VeriGame A
Monthly Number of Assertions (MNA)	Monthly productivity metric for VeriGame A
Daily Score (DS)	Daily productivity metric for VeriGame B
Weekly Score (WS)	Weekly productivity metric for VeriGame B
Monthly Score (MS) Monthly	Productivity metric for VeriGame B

4.4 Summary

This chapter demonstrated the methodology of the thesis. Simply, we defined and described what we did and how we did it, step by step. The hardest part of the research was dealing with the data in two different forms and structures. We simplified the process by clearing the unnecessary data. The other thing we did for simplification was to decrease the number of data tables. If it is possible, one should work on a single table as it is the easiest method. We also kept SQL queries as simple as possible. The SQL command also provided sustained repeatability of the methodology.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5:

Analysis and Evaluation

5.1 Introduction

This chapter presents the analysis and results. The analysis is sorted by the types of metrics defined in Chapter 4. In addition, the games will be compared under the metrics if data is available for a particular game. Finally, prediction analysis will conclude the chapter.

5.1.1 Initial Analysis

After importing the data into the database, unnecessary and inconsistent parts of the data were eliminated. Then, early analysis was initiated. The first trend we noticed was the high drop-off rate of the players. Player fracture can be clearly seen in Figure 5.1. Both games allow anonymous playing until the end of the tutorials sections. After the tutorials players have to register to move forward. In addition, Figure 5.1 shows that both games could only transformed 10 to 15 percent of the players into registered players, and around 8 percent of all players into productive ones.

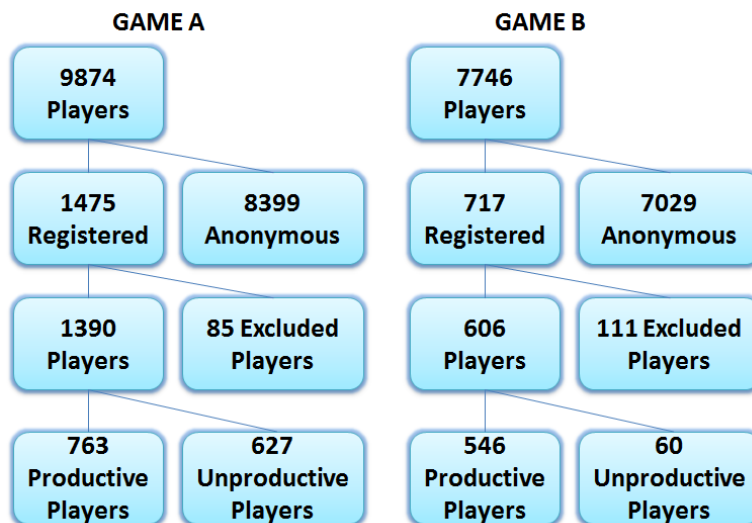


Figure 5.1: Player profiles of VeriGames A and B

5.1.2 DAU and MAU

Traditional Games

As discussed in previous chapters, players are vitally important for games. In the highly competitive gaming market, traditional games require as many players as possible to increase the revenue. DAUs and MAUs of a sample set of games is shown Table 5.1. This table illustrates that mobile, social, and online games and their developers can have as many as ten of thousands or millions of players.

Table 5.1: Average DAU and MAU for Selected Mobile, Social, and Online Games

	DAU	MAU
Zyanga*	11.1M	292M
Storm8*	4M	-
Glu Mobile*	3.4M	29M
Angry Birds	20M	200M
Temple Run	7M	-
Stardom	74K	-
Deer Hunter	271K	-
Junkies	114K	-
Triple Town	-	160K
Parallel Kingdom	-	50K
DeNa*	-	16.9M
GREE*	-	13.9M

*A collection of games from the named game developer/operator.

VeriGames

CSSGs may not have as big an audience as traditional games because the main purpose of CSSGs is not players' enjoyment, but solving scientific problems. In addition, CSSGs' developer teams may not have budgets as generous as those of gaming companies. Table 5.2 shows the statistical information for DAU, and Table 5.3 shows MAUs of VeriGames. Unlike traditional games, VeriGames have DAUs as low as one or two, and the highest DAUs are close to 900.

VeriGames MAU's are also comparatively low (Table 5.3). The first month's MAUs are the highest for both VeriGames, possibly because the highest marketing efforts were done in the first month after release. As a result, new players, specialist in the gaming industry or blog writers played the games after release. For the next few months, the games lost the

crowd. For instance, in January the games' MAUs decreased around 85 percent for both VeriGames.

Table 5.2: DAU of Sample VeriGames

DAU					
	Min	Max	Mean	Median	StDev
VeriGame A	2	872	71.5	23	158.2
VeriGame B	1	887	64.5	16	135.5

Table 5.3: MAU of Sample VeriGames

MAU					
Month	Dec	Jan	Feb	Mar	Apr
VeriGame A	7555	957	615	415	460
VeriGame B	5000	504	244	-	-

5.1.3 Engagement Rate

Traditional Games

As stated in Chapter 4, ER shows a game's stickiness. The games in Table 5.4 have an ER differentiation between 10 and 30 percent. Actually, it is difficult to define an ER threshold for a game's success. For example, *Angry Birds*, has the lowest ER but higher DAU and MAU than others, which shows that game is performing well. Consequently, although a higher ER is better assessing ER with DAU and MAU is essential.

Table 5.4: ER of Some Mobile, Social and Online Games and Developers

Average Engagement Rate (%)	
Angry Birds	10
Parallel Kingdoms	30
Glu Mobile*	11.7
Zynga*	22.5
Scrabble	30
Bejeweled Blitz	27
Pet Society	14

Other Crowd-Sourced Serious Games

Other CSSGs also have low ERs and high drop-off rates. For example, *Phylo*, which requires players to solve puzzles to assist in finding a solution for genetic disorders, had

around 12,000 registered players seven months after release, but only 23 percent of those players returned one more time to play the game [38]. Forty-two percent of acquired *Phylo* players gave up playing without completing a single puzzle [38]. *FoldIt* has more than 500,000 players on its soloist hall of fame leaderboard [34], but about 80 percent of those players have not scored any points, which also indicates a high drop-off rate and possible low ER [49].

VeriGames

As with other CSSGs, VeriGames developers have to primarily consider how games will transform players’ efforts into valuable inputs for science. Design of the game mechanism may cause less attractive games. While traditional games have around 10 to 30 percent ER, the VeriGames have less than 5 percent ERs. Notably, the ERs of the VeriGames are the lowest in the first month of deployment, although the number of players recorded (MAU) is the highest for that month. We attribute the high drop off rate of MAU primarily to having low ERs, caused by non-returning players. The ERs of the VeriGames tend to increase monthly while MAU is steadily decreasing over the first three months. This may show that the VeriGames obtained a core set of loyal people who keep playing [49].

Table 5.5: ER of VeriGame A and B in the Monthly Average Basis

Monthly Engagement Rate (%)						
Month	Dec	Jan	Feb	Mar	Apr	Avg
VeriGame A	3.41	3.85	4.36	4.10	3.59	3.86
VeriGame B	3.27	3.39	3.71	-	-	3.45

5.1.4 Session Time (ST) and Session Count (SC)

Traditional Games

ST and SC are two metrics related to the productivity of the players. According to a recent report players spend close to 100 minutes and an average session count is 3.29 for social, casual, and mobile games [65]. Another report says while other players consume around four hours, whales consume close to 12 hours for mobile games [62]. STs increase to almost nine for other players, and to almost 27 for whales, when console, PC and other games are included [62].

VeriGames

The cumulative distributions of the ST and SC metrics for the registered players of the two VeriGames are shown in Figure 5.2 and Figure 5.3. The registered players of VeriGame A spent 1236 hours in total, and the average is one hour per player. We observe that the order of the players by their STs is identical to the order of their productivity for the top ten players except one. For VeriGame B, the registered players spent 558 hours in total, and the average is again close to one hour per player. Eight of the ten most productive players are also in the top 20 in terms of session time. In addition, for both games, each of the top 20 most productive players played more than ten hours. In other words, the ratio of STs between whales and average players is about 10 to 1, much higher than the 3 to 1 ratio previously reported for social mobile games [49], [62].

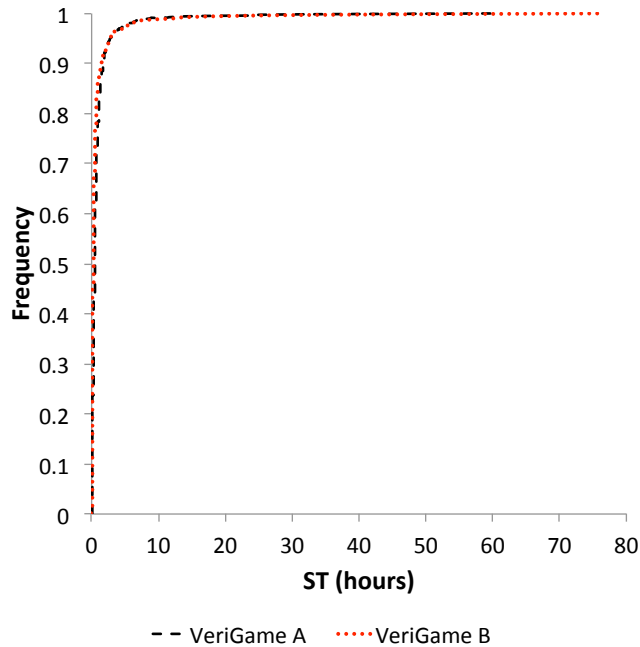


Figure 5.2: Session Time CDF of VeriGames

5.1.5 Whales

Other Crowd-Sourced Serious Games

Whales are important for other CSSGs, and the fractions of whales are low compared to commercial games. 90 percent of registered 12,000 *Phylo* players finished fewer than 25 puzzles while the top 10 percent of players participated in nearly 80 percent of all solutions

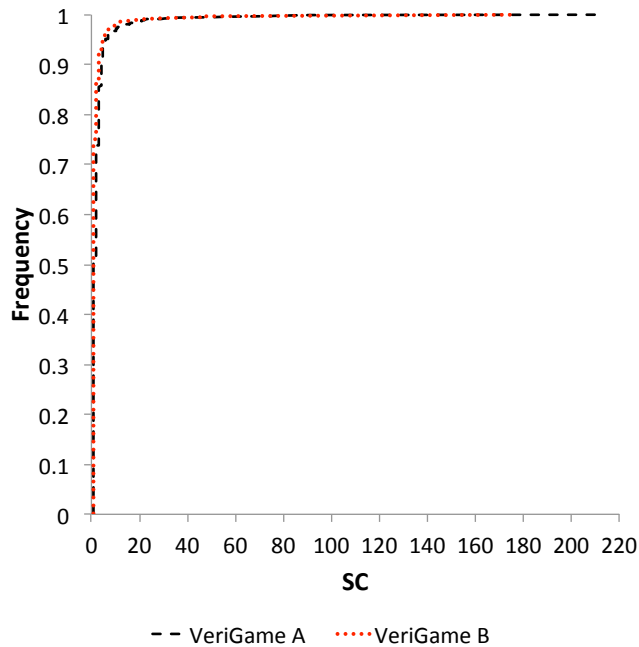


Figure 5.3: Session Count CDF of VeriGames

produced by registered players. The top 20 players solved more than 700 puzzles each [38]. On *FoldIt's* soloist hall of fame leaderboard, three players have more than 40,000 points each, eight players have between 30,000 to 40,000 each, 27 players fall between 20,000 and 30,000, and 64 players are between 10,000 and 20,000 points [34]. This indicates a similar WEG curve for *Foldit* players. *EyeWire* also relies heavily on whales [2]. Kim et al. has stated that more than 100,000 registered non-expert players from more than 130 countries have contributed to the experiment, however the 100 most productive players generated almost half of the production [49].

VeriGames

In VeriGames A and B a small group of whales is performing significantly better than the other players as well. Figure 5.4 shows the whale effect graph (WEG) for the registered users of these games. The rapid increase in productivity percentile over the first few percent of the players on the WEG shows the effectiveness of the whales. For VeriGame A, over 60 percent of the productivity is attributable to less than 10 percent of the players. For VeriGame B the top 10 percent of players produce more than 40 percent of the overall productivity. The steeper curve of VeriGame A clearly indicates that the whales of VeriGame

A are more productive than those of VeriGame B. In other words, VeriGame A relies more on whales than VeriGame B [49].

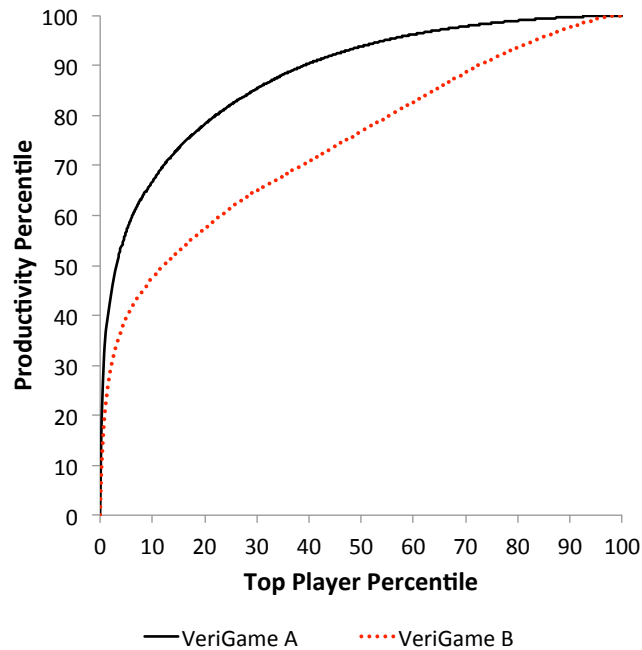


Figure 5.4: WEG of VeriGames

Figure 5.5 shows the WEG after including data from all players, including even those that do not choose to register. The WEG curve of VeriGame A has the same shape as before, while the slope of the curve for VeriGame B is more linear, possibly resulting from distinctive game mechanisms. In particular, VeriGame B allows non-registered players to accumulate scores while VeriGame A does not [49].

Figure 5.6 shows both the productivity and ST percentiles in one WEG. The ST curves of VeriGame A and B have similar slopes to those of the productivity curves, indicating that the whales of these games tend to spend more time playing than others. Furthermore, unlike the CDF plots, the WEG exposes a drastic difference between the two games. For VeriGame A, the ST curve is below the productivity curve, meaning that the whales for this game are more productive per unit of time than an average player. This is an expected outcome as a player's game skills should improve with more playing time. However, for VeriGame B, the situation is the opposite: the ST curve is *above* the productivity curve,

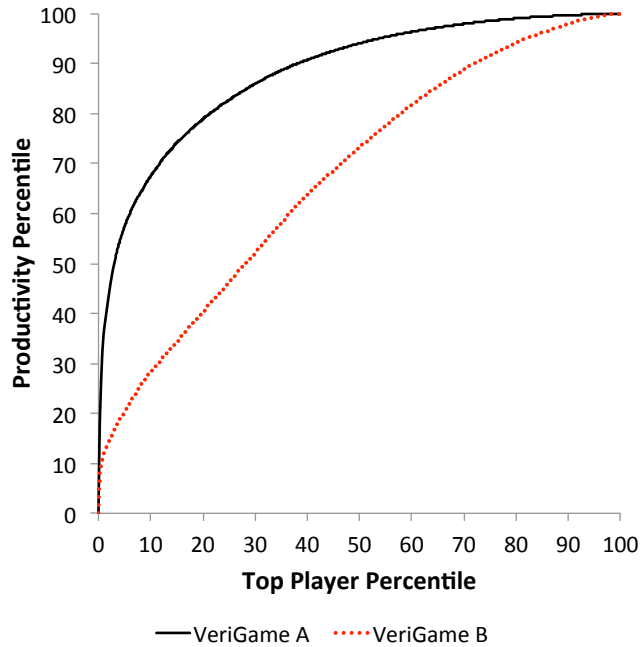


Figure 5.5: WEG of VeriGames Including Anonymous Players

meaning that a player produces less per unit of time when spending more time with the game. This indicates a potential deficiency in VeriGame B’s scoring system or game design.

5.1.6 Prediction Analysis

We perform additional analyses using the more detailed VeriGames datasets, seeking to further explain some of the results presented in the previous sections.

First, we analyze the player attrition pattern going through the registration and tutorial phases of each game for comparison with a prior study of Duolingo player attrition [53]. The results are presented in Figure 5.1. The patterns are very similar in both VeriGames. Most players did not maintain their interest after initially trying out the games. Only 10 to 15 percent of the players completed the registration process. After filtering out erroneous registrations, game development team members, and unproductive players (who completed the tutorials, but did not complete any game levels), one can conclude that fewer than ~8 percent of the total players recorded in our VeriGame datasets are productive players. Given such a low fraction of productive players to start with, the long-tail whale effect graphs presented in the last section are easily understood.

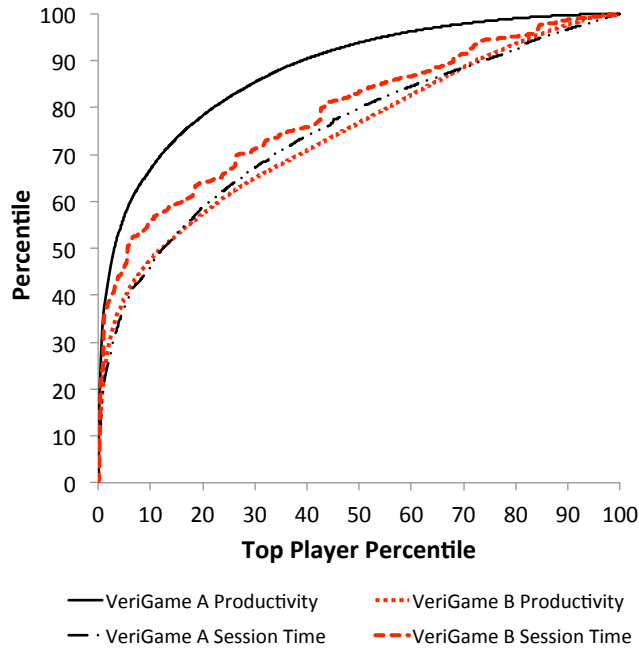


Figure 5.6: WEG of VeriGames Including ST and SC

Second, we perform a linear regression analysis with a 95 percent confidence interval to determine the best aggregate game play metric for predicting the total productivity over a period of time. Three game play metrics were evaluated: total active users, total session counts, and total session time. Each of the three metrics was evaluated over two different time intervals: per day and per week. The results are similar for the two time intervals. Weekly fitted regression line plots are shown in the Appendix. All three metrics are good indicators for game productivity.

Table 5.6: R^2 Values for Regression Analysis (Daily Metrics)

	VeriGame A	VeriGame B
	R^2 values	
DAU	0.788	0.818
DSC	0.799	0.768
DST	0.914	0.620

However, upon inspection of the p -values obtained (Table 5.8 and Table 5.9) when all three metrics are jointly considered in a multiple linear analysis, we conclude that the total session time is best for predicting the productivity of VeriGame A while the total active

Table 5.7: R^2 Values for Regression Analysis (Weekly Metrics)

	VeriGame A	VeriGame B
	R^2 values	
WAU	0.880	0.965
WSC	0.906	0.932
WST	0.961	0.896

users is best for VeriGame B. This result is consistent with the observation we made about Figure 5.6.

Table 5.8: R^2 and p Values for Multiple Regression Analysis(Daily Metrics)

	VeriGame A	VeriGame B
R^2	0.95	0.82
	p values	
DAU	0.071	5.827E-06
DSC	3.602E-14	0.259
DST	1.637E-53	0.816

Table 5.9: R^2 and p Values for Multiple Regression Analysis (Weekly Metrics)

	VeriGame A	VeriGame B
R^2	0.99	0.97
	p values	
WAU	0.077	0.004
WSC	0.047	0.540
WST	2.217E-10	0.173

5.2 Summary

In this chapter, we applied the methodology of the research. When compared with traditional games, other CSSGs and VeriGames have a lower number of players due to the obvious reason that CSSGs are not primarily designed for player enjoyment. Low ERs and high drop-off rates justify that. However, in CSSGs a small group of players (whales) perform significantly better than other players. Therefore, the contribution of whales can counterbalance the low ERs and player numbers.

CHAPTER 6:

Conclusion

6.1 Summary and Conclusion

From the data available to us, it appears that CSSGs have lower engagement rates than traditional games. Low ERs can be a significant obstacle in the path of CSSGs making a significant impact and accomplishing their ultimate purpose. CSSGs in general have not wielded a level of intrinsic attraction sufficient to attract and retain high numbers of long-term players. Given that situation, if the existing players only play the games occasionally, CSSGs face a serious productivity problem. Both VeriGames and other CSSGs examined in this paper have a high proportion of non-returning players and relatively low ERs. There may be several reasons for this problem such as CSSGs' purpose-driven game mechanisms which do not directly target players' personal entertainment, and relatively low game-development budgets.

All of this leads us to focus on the contribution that whales make to the productivity of CSSGs. Our analyses show that CSSGs benefit from whales as do commercial games. Vulnerability caused by low ERs and non-returning players can be partially mitigated by focusing on attracting new whales to CSSGs who are ideologically supportive of the games' underlying purpose. While the specific threshold for differentiating whales from other players varies from game to game, and will likely always do so, the Whale Effect Graph allows us to quickly evaluate the extent to which a particular game relies on whales' productivity, as well as qualitatively comparing their impact across multiple games. Unfortunately we do not have sufficient data from traditional games to create WEGs for them, which would allow us to state conclusively whether whales are more significant to CSSGs than to traditional games. This is an area for future research.

6.2 Future Work and Limitations

Here, we list the limitations of the current study and potential areas for future work:

- In this thesis we applied the methodology to half-year datasets from two VeriGames.

Clearly, additional analyses of more CSSGs as well as new datasets covering longer periods of time are required to confirm the generality of our methodology and strengthen or refine our conclusions.

- We did not have enough data to produce WEGs for traditional games. It will be a worthwhile pursuit to establish some ground truth about traditional games in this aspect.
- Studies should be performed to understand if and how the marketing and design of CSSGs may improve in order to recruit and retain whales more effectively.

APPENDIX: Line Fit Plots of Regression Analyses

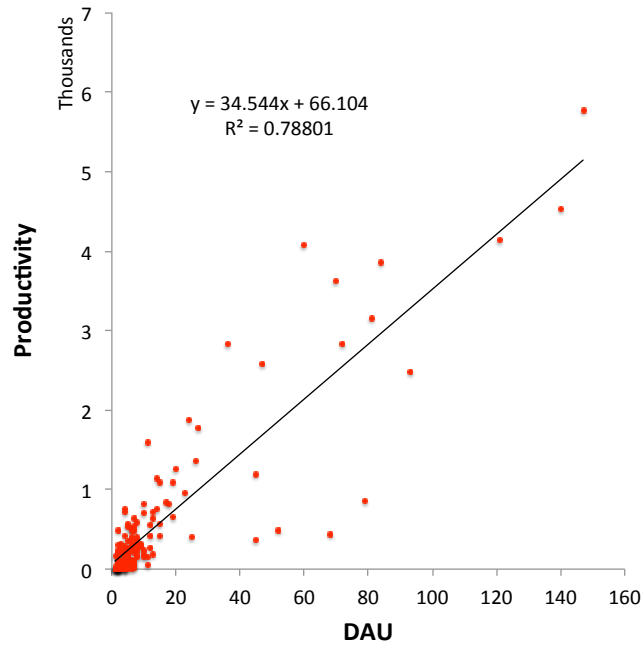


Figure 1: VeriGame A DAU LFP

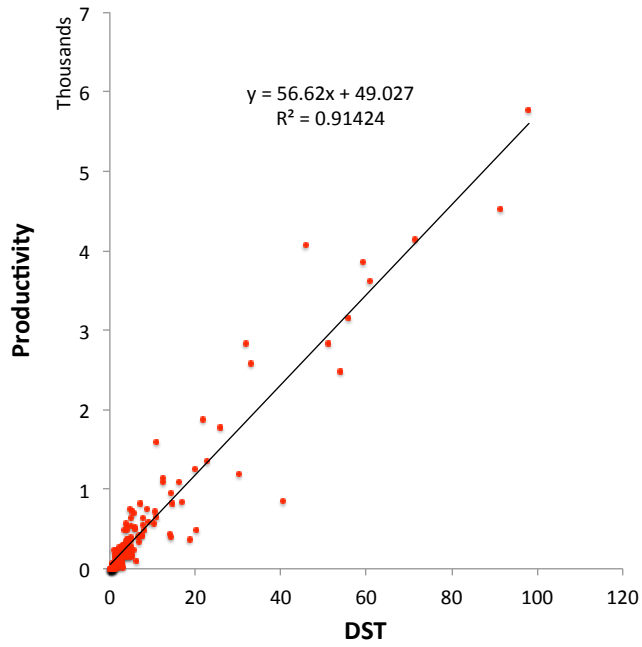


Figure 2: VeriGame A DST LFP

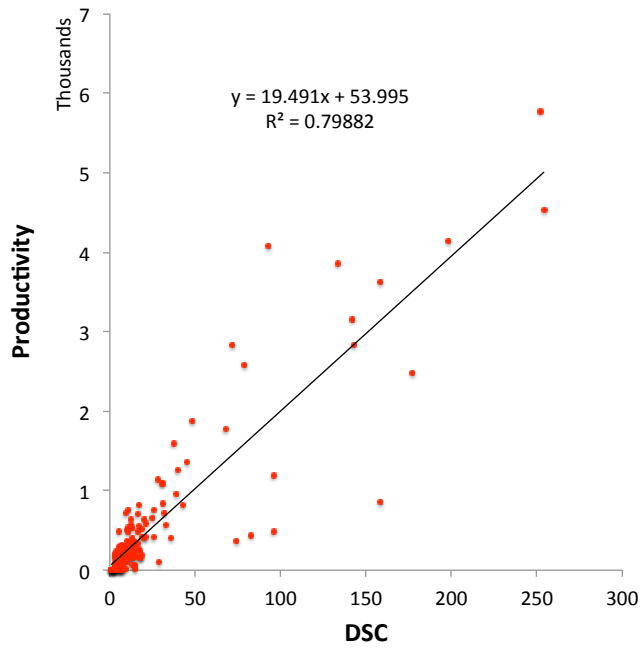


Figure 3: VeriGame A DSC LFP

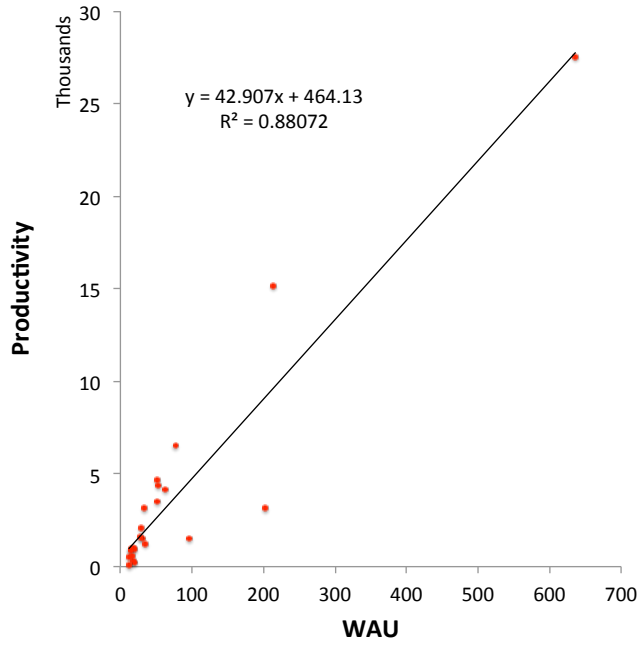


Figure 4: VeriGame A WAU LFP

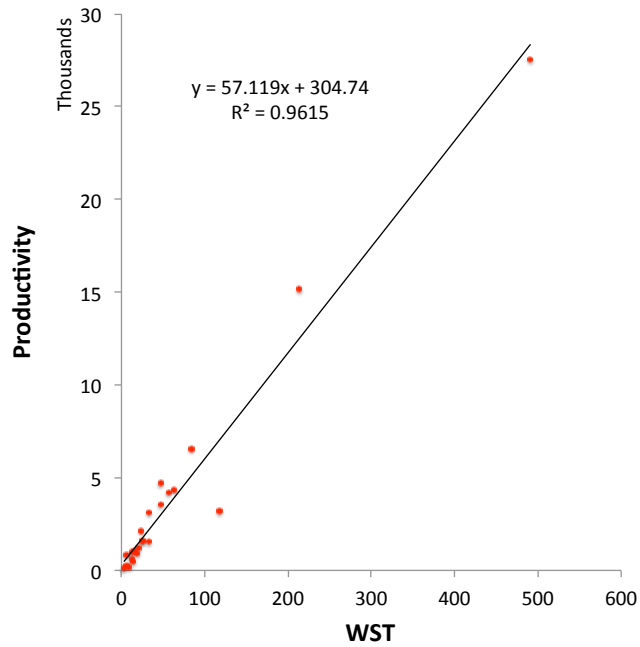


Figure 5: VeriGame A WST LFP

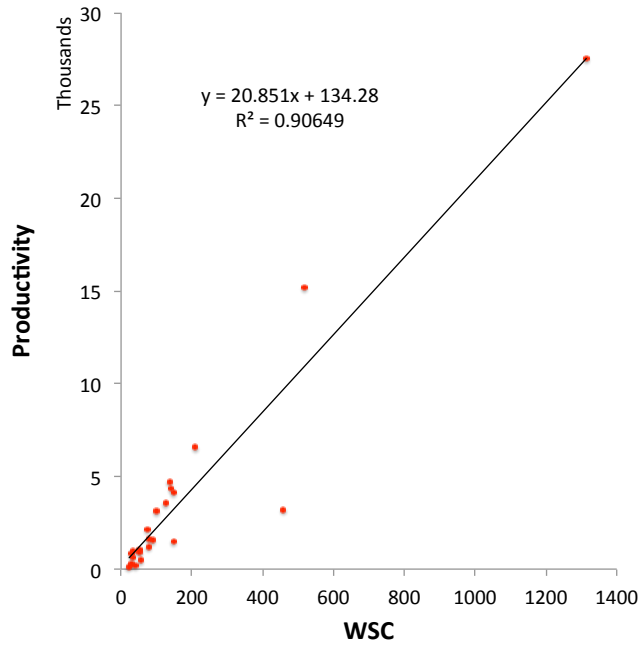


Figure 6: VeriGame A WSC LFP

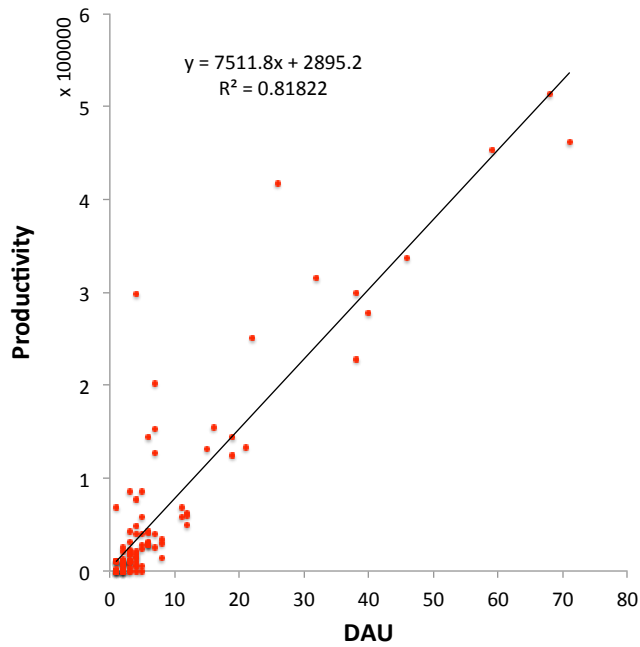


Figure 7: VeriGame B DAU LFP

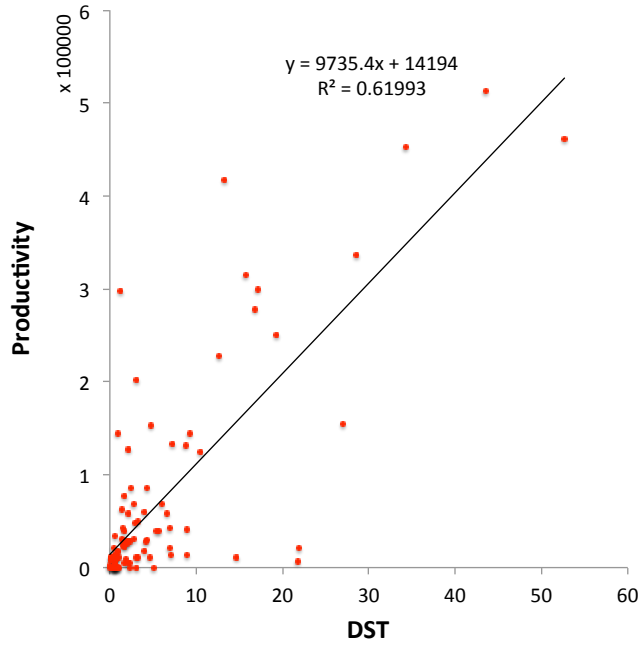


Figure 8: VeriGame B DST LFP

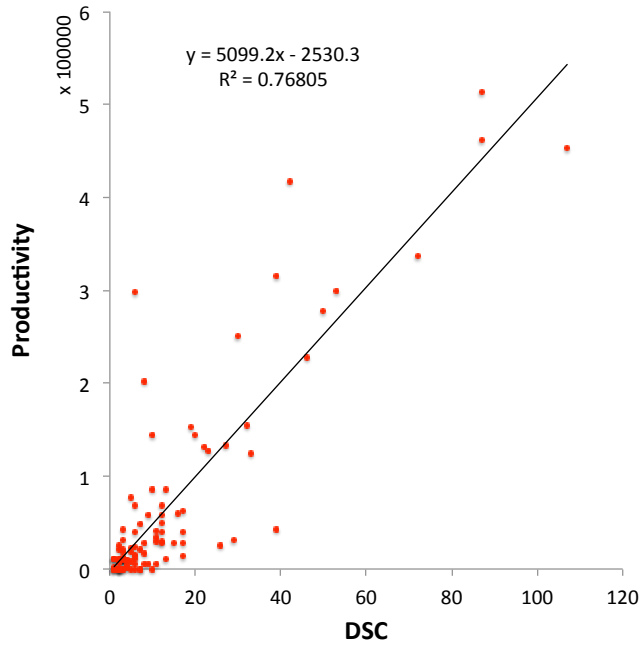


Figure 9: VeriGame B DSC LFP

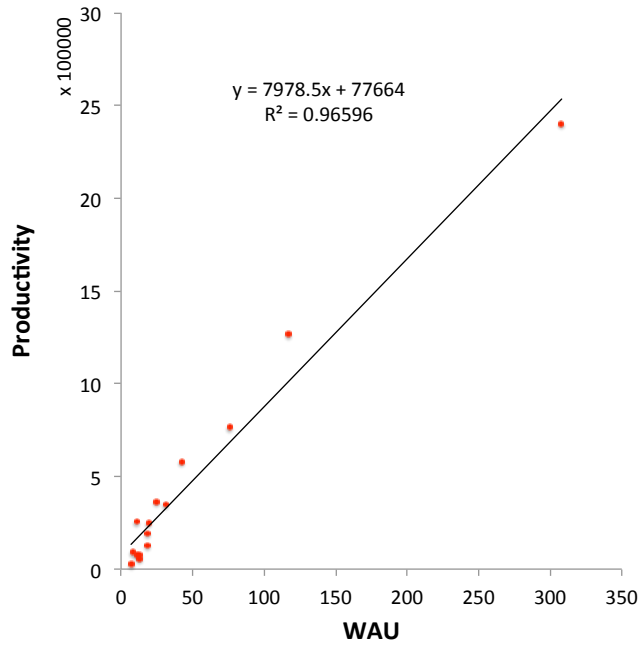


Figure 10: VeriGame B WAU LFP

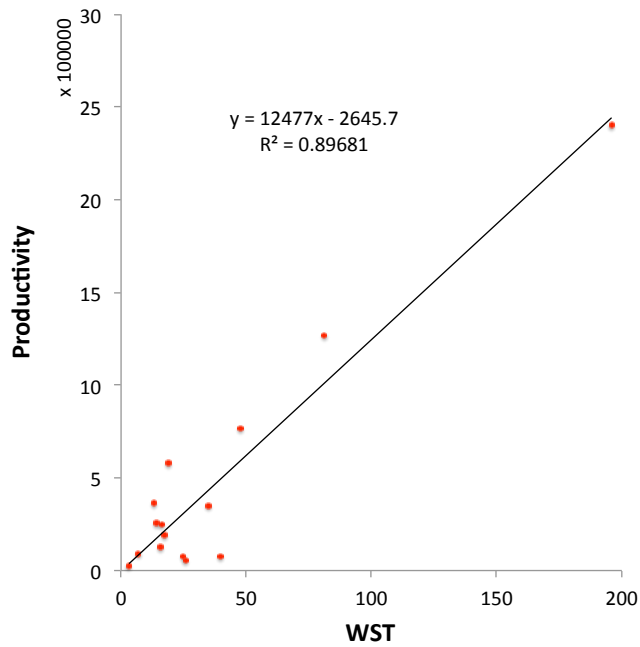


Figure 11: VeriGame B WST LFP

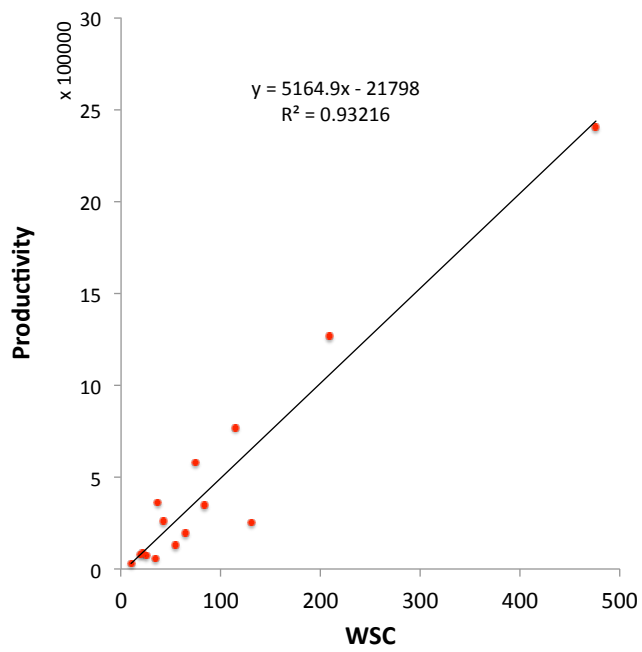


Figure 12: VeriGame B WSC LFP

THIS PAGE INTENTIONALLY LEFT BLANK

References

- [1] Duolingo web portal. [Online]. Available: <http://www.duolingo.com>
- [2] J. S. Kim, M. J. Greene, A. Zlateski, K. Lee, M. Richardson, S. C. Turaga, M. Purcaro, M. Balkam, A. Robinson, B. F. Behabadi *et al.*, “Space-time wiring specificity supports direction selectivity in the retina,” *Nature*, vol. 509, no. 7500, pp. 331–336, 2014.
- [3] “Crowd sourced formal verification (csfv).” [Online]. Available: [http://www.darpa.mil/Our_Work/I2O/Programs/Crowd_Sourced_Formal_Verification_\(CSFV\).aspx](http://www.darpa.mil/Our_Work/I2O/Programs/Crowd_Sourced_Formal_Verification_(CSFV).aspx)
- [4] M. Sabadello, “History of electronic games,” master’s thesis, Institute of Computer Graphics and Algorithms, Vienna Univ. of Techn., Oct. 2006. [Online]. Available: <http://www.cg.tuwien.ac.at/research/publications/2006/SABADELLO-2006-HEG/>
- [5] L. Herman, J. Horwitz, S. Kent, and S. Miller, “The history of video games,” *Gamespot*, vol. 7, 2002.
- [6] J. O. Soh and B. C. Tan, “Mobile gaming,” *Communications of the ACM*, vol. 51, no. 3, pp. 35–39, 2008.
- [7] N. Esposito, “A short and simple definition of what a videogame is,” Apr. 2005. [Online]. Available: <http://summit.sfu.ca/item/258>
- [8] “Personal computers.” [Online]. Available: <http://www.computerhistory.org/revolution/personal-computers/17>
- [9] J.-M. Vanhatupa, “Browser games: The new frontier of social gaming,” in *Recent Trends in Wireless and Mobile Networks*. Springer, 2010, pp. 349–355.
- [10] “Gartner says worldwide pc, tablet and mobile phone combined shipments to reach 2.4 billion units in 2013,” Apr. 2013. [Online]. Available: <http://www.gartner.com/newsroom/id/2408515>
- [11] “Obj: Mobile gaming’s big growth set to continue,” Jan. 2014. [Online]. Available: <http://www.businesswire.com/news/home/20140130005508/en/OBJE-Mobile-Gaming%E2%80%99s-Big-Growth-Set-Continue#.U72RwY1dUfK>
- [12] D. Kroening and N. Sharygina, “Formal verification of systemc by automatic hardware/software partitioning,” in *Proceedings of the 2nd ACM/IEEE Int. Conf. on Formal Methods and Models for Co-design*. IEEE Computer Society, 2005, pp. 101–110.

- [13] G. Li, “Formal verification of programs and their transformations,” Ph.D. dissertation, The Univ. of Utah, 2010.
- [14] D. R. Wallace and R. U. Fujii, “Software verification and validation: An overview,” *IEEE Software*, vol. 6, no. 3, pp. 10–17, 1989.
- [15] U. Tellioglu, B. Kartal, M. Simsek, and O. Eryilmaz, “Call of duty: Can turkey benefit from crowd-sourced serious games to strengthen its cyber security capabilities?” to be published in the proceedings of 7th Int. Conf. on Information Security and Cryptology.
- [16] M. Delio, “Linux: Fewer bugs than rivals,” Dec. 2004. [Online]. Available: <http://archive.wired.com/software/coolapps/news/2004/12/66022>
- [17] K. Poulsen, “Software bug contributed to blackout.” [Online]. Available: <http://www.securityfocus.com/news/8016>
- [18] A. Lum, “Verification software bug report patriot missile software problem.” [Online]. Available: http://sydney.edu.au/engineering/it/~alum/patriot_bug.html
- [19] S. Rogerson, “The chinook helicopter disaster,” *IMIS Journal*, vol. 12, no. 2, 2002.
- [20] “The heartbleed bug.” [Online]. Available: <http://heartbleed.com/>
- [21] D. Cenciotti, “German heron drone hacked and crashed by taliban in afghanistan,” Nov. 2013. [Online]. Available: <http://theaviationist.com/2013/11/13/heron-hacked-afghanistan/>
- [22] D. Dean, “Proceed and crowd-sourced formal verification,” DTIC Document, Tech. Rep., 2011.
- [23] M. Hansen and R. F. Nesbit, “Report of the defense task force on defense software.” [Online]. Available: <http://www.acq.osd.mil/dsb/reports/ADA385923.pdf>
- [24] D. E. Williams, “Gamers vs. hackers.” [Online]. Available: http://www.verigames.com/Geek-May_2014-LEVEL_UP.pdf
- [25] J. Johnson, “The new industrial revolution (infographic).” [Online]. Available: <http://insights.wired.com/profiles/blogs/the-new-industrial-revolution>
- [26] D. McCandless, “Codebases.” [Online]. Available: <http://www.informationisbeautiful.net/visualizations/million-lines-of-code/>

- [27] M.-C. Yuen, I. King, and K.-S. Leung, “A survey of crowdsourcing systems,” in *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*. IEEE, 2011, pp. 766–773.
- [28] M. F. Dumlupinar, “Deploying crowd-sourced formal verification systems in a dod network,” master’s thesis, Monterey, California: Naval Postgraduate School, Sep. 2013.
- [29] I. Garcia, “Learning a language for free while translating the web. does duolingo work?” *International Journal of English Linguistics*, vol. 3, no. 1, p. p19, 2013.
- [30] K. Lakhani, D. A. Garvin, and E. Lonstein, “Topcoder (a): Developing software through crowdsourcing,” *Harvard Business School General Management Unit Case*, no. 610-032, 2010.
- [31] “Amazon mechanical turk.” [Online]. Available: <https://www.mturk.com/mturk/help?helpPage=overview>
- [32] T. W. Malone, R. Laubacher, and C. Dellarocas, “Harnessing crowds: Mapping the genome of collective intelligence,” 2009. [Online]. Available: <http://18.7.29.232/handle/1721.1/66259>
- [33] L. Von Ahn, “Games with a purpose,” *Computer*, vol. 39, no. 6, pp. 92–94, 2006.
- [34] “Foldit web portal.” [Online]. Available: <http://fold.it/portal/info/about>
- [35] D. Praetorius, “Gamers decode aids protein that stumped researchers for 15 years in just 3 weeks,” Nov. 2011. [Online]. Available: http://www.huffingtonpost.com/2011/09/19/aids-protein-decoded-gamers_n_970113.html
- [36] V. Marx, “Neuroscience waves to the crowd,” *Nature methods*, vol. 10, no. 11, pp. 1069–1074, 2013.
- [37] A. Frank, “70,000+ have played ’eyewire’ game that trains computers to map the brain,” Aug. 2013. [Online]. Available: <http://www.forbes.com/sites/singularity/2013/08/19/70000-have-played-eyewire-game-that-trains-computers-to-map-the-brain/>
- [38] A. Kawrykow, G. Roumanis, A. Kam, D. Kwak, C. Leung, C. Wu, E. Zarour, L. Sarmanta, M. Blanchette, and J. Waldspühl, “Phylo: A citizen science approach for improving multiple sequence alignment,” *PLoS ONE*, vol. 7, no. 3, p. e31362, 2012.

- [39] S. Cooper, A. Treuille, J. Barbero, A. Leaver-Fay, K. Tuite, F. Khatib, A. C. Snyder, M. Beenen, D. Salesin, D. Baker *et al.*, “The challenge of designing scientific discovery games,” in *Proceedings of the fifth international conference on the foundations of digital games*. ACM, 2010, pp. 40–47.
- [40] “Verigames web portal.” [Online]. Available: <http://www.verigames.com>
- [41] “voidalpha’s stormbound challenges gamers to use puzzle solving skills to defeat real world hackers.” [Online]. Available: http://gamasutra.com/view/pressreleases/206399/voidALPHArsquos_StormBound_Challenges_Gamers_To_Use_PuzzleSolving_Skills_To_Defeate_Real_World_Hackers.php
- [42] M. S. El-Nasr, A. Drachen, and A. Canossa, *Game analytics: Maximizing the value of player data*. Springer, 2013.
- [43] T. V. Fields, “Game industry metrics terminology and analytics case study,” in *Game Analytics*. Springer, 2013, pp. 53–71.
- [44] A. Canossa, M. S. El-Nasr, and A. Drachen, “Benefits of game analytics: Stakeholders, contexts and domains,” in *Game Analytics*. Springer, 2013, pp. 41–52.
- [45] A. Drachen, C. Thureau, J. Togelius, G. N. Yannakakis, and C. Bauckhage, “Game data mining,” in *Game Analytics*. Springer, 2013, pp. 205–253.
- [46] A. Canossa, “Meaning in gameplay: Filtering variables, defining metrics, extracting features and creating models for gameplay analysis,” in *Game Analytics*. Springer, 2013, pp. 255–283.
- [47] A. Drachen, A. Canossa, and J. R. M. Sørensen, “Gameplay metrics in game user research: Examples from the trenches,” in *Game Analytics*. Springer, 2013, pp. 285–319.
- [48] A. Canossa, “Interview with ola holmdahl and ivan garde from junebud,” in *Game Analytics*. Springer, 2013, pp. 689–693.
- [49] U. Tellioglu, G. G. Xie, J. P. Rohrer, and P. Charles, “Whale of a crowd: Quantifying the effectiveness of crowd-sourced serious games,” to be published in the proceedings of 19th International Conference on Computer Games.
- [50] M. S. El-Nasr and A. Canossa, “Interview with jim baer and daniel mccaffrey from zynga,” in *Game Analytics*. Springer, 2013, pp. 73–82.
- [51] T. Mahlmann, A. Drachen, J. Togelius, A. Canossa, and G. N. Yannakakis, “Predicting player behavior in tomb raider: Underworld,” in *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on*. IEEE, 2010, pp. 178–185.

- [52] S. Mavandadi, S. Feng, F. Yu, S. Dimitrov, R. Yu, and A. Ozcan, “Biogames: A platform for crowd-sourced biomedical image analysis and telediagnosis,” *GAMES FOR HEALTH: Research, Development, and Clinical Applications*, vol. 1, no. 5, pp. 373–376, 2012.
- [53] R. Vesselinov and J. Grego, “Duolingo effectiveness study: Final report,” Dec. 2012. [Online]. Available: http://static.duolingo.com/s3/DuolingoReport_Final.pdf
- [54] N. Lovell, “Dau mau engagement.” [Online]. Available: <http://www.gamesbrief.com/2011/10/daumau-engagement/>
- [55] N. Lovell, “Daily active users.” [Online]. Available: <http://www.gamesbrief.com/2011/10/daily-active-users-daus/>
- [56] N. Lovell, “Monthly active users.” [Online]. Available: <http://www.gamesbrief.com/2011/10/monthly-active-users/>
- [57] M. Rose, “Chasing the whale: Examining the ethics of free-to-play games,” Jul. 2013. [Online]. Available: <http://www.gamasutra.com/view/feature/195806/>
- [58] T. V. Fields, “Game industry metrics terminology and analytics case study,” in *Game Analytics*. Springer, 2013, pp. 53–71.
- [59] M. Carpenter, “How to measure social games success,” Jan. 2011. [Online]. Available: <http://www.gamesindustry.biz/articles/2011-01-24-how-pocap-measures-social-games-success-editorial>
- [60] “Breaking into social gaming-a must-read guide to entering the facebook game space,” Aug. 2011. [Online]. Available: http://socialtimes.com/breaking-into-social-gaming-a-must-read-guide-to-entering-the-facebook-game-space_b75872
- [61] A. W. Hafner, “Pareto’s principle: The 80-20 rule,” Mar. 2001. [Online]. Available: <http://www.bsu.edu/libraries/ahafner/awh-th-math-pareto.html>
- [62] E. Kain, “Mobile gaming’s whales overwhelmingly male, spend big on all types of video games,” Aug. 2013. [Online]. Available: <http://www.forbes.com/sites/erikkain/archive/2013/08/>
- [63] S. Carmichael, “What it means to be a whale and why social gamers are just gamers,” Mar. 2013. [Online]. Available: <http://venturebeat.com/2013/03/14/whales-and-why-social-gamers-are-just-gamers>

- [64] D. Takahashi, "Only 0.15 percent of mobile gamers account for 50 percent of all in-game revenue (exclusive)," Feb. 2014. [Online]. Available: <http://venturebeat.com/2014/02/26/>
- [65] "Playnomics player engagement study q3 2013," Dec. 2013. [Online]. Available: <http://www.playnomics.com/wp-content/uploads/2013/12/Playnomics-Player-Engagement-Q3-2013.pdf>

Initial Distribution List

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California