

5-1-2014

Hermes, Technical Communicator of the Gods: The Theory, Design, and Creation of a Persuasive Game for Technical Communication

Eric Walsh

University of South Florida, walshericc@gmail.com

Follow this and additional works at: <https://scholarcommons.usf.edu/etd>

 Part of the [Education Commons](#), [Other Communication Commons](#), and the [Rhetoric Commons](#)

Scholar Commons Citation

Walsh, Eric, "Hermes, Technical Communicator of the Gods: The Theory, Design, and Creation of a Persuasive Game for Technical Communication" (2014). *Graduate Theses and Dissertations*.
<https://scholarcommons.usf.edu/etd/5147>

This Thesis is brought to you for free and open access by the Graduate School at Scholar Commons. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Scholar Commons. For more information, please contact scholarcommons@usf.edu.

Hermes, Technical Communicator of the Gods: The Theory, Design, and Creation of a
Persuasive Game for Technical Communication

by

Eric Walsh

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Arts
Department of English
College of Arts and Sciences
University of South Florida

Major Professor: Marc C. Santos, Ph.D.
Meredith Zoetewey, Ph.D.
Julie Staggers, Ph.D.

Date of Approval:
March 25, 2013

Keywords: Design, Education, Persuasive Games, Procedural Rhetoric

Copyright © 2014, Eric Walsh

Acknowledgments

I would like to thank Drs. Santos, Zoetewey, and Stagers for agreeing to this somewhat untraditional thesis, and for allowing me the creative freedom to follow my intellectual passions, even if it leads into material with which they are not particularly familiar. I would also like to thank them for always being available when I have a question, for helping me further refine my ideas in search of a better topic, and for striving to accommodate some rather fast turnaround times as I work towards the completion of my thesis.

Table of Contents

| | |
|---|-----|
| List of Figures | ii |
| Abstract | iii |
| Introduction..... | 1 |
| Idea: Why Games Are Good for Education | 4 |
| Games and Learning | 4 |
| Existing Research in the Field | 6 |
| My Game Idea..... | 8 |
| Development: Procedural Rhetoric and Design..... | 10 |
| Procedural Rhetoric | 10 |
| Crafting My Argument | 14 |
| Attention | 15 |
| Projects/Tasks | 15 |
| Success/Time | 15 |
| Game Design: Aesthetics and Fun..... | 17 |
| Designing My Game to Be Aesthetically Pleasing and Fun..... | 18 |
| Practice: Hermes, Technical Communicator of the Gods..... | 20 |
| Code | 20 |
| Procedural Rhetoric | 21 |
| Interface | 22 |
| Sound | 24 |
| Visuals..... | 24 |
| Narrative | 25 |
| Menus..... | 27 |
| Projects..... | 28 |
| Tasks | 31 |
| Gods | 32 |
| Mythos | 33 |
| Conclusion | 35 |
| References | 38 |

List of Figures

| | |
|--------------------------------------|----|
| Figure 1. Game Interface | 23 |
| Figure 2. Tooltip | 26 |
| Figure 3. Chat Box | 26 |
| Figure 4. Start Menu | 28 |
| Figure 5. How to Play Sub-Menu | 29 |

Abstract

For my thesis, I have undertaken the creation of a persuasive game to advance a particular argument of the way that work is performed in the field of technical communication. Designed using procedural rhetoric, with an attention to aesthetics, fun, and the qualities that make games viable pedagogical tools, my game has been programmed using HTML5 and JavaScript, and made freely available online at RhetoricalGamer.com. This written document is meant to serve as a supplement to the game, providing a rationale for the use of games in education and in technical communication; a definition of procedural rhetoric and the necessary qualities of game design to ensure that the rhetoric operates correctly; and a detailed breakdown of the final elements and mechanics in place within my game. It is my hope that this work will serve as an exemplar for others interested in pursuing the creation of persuasive games, as a case study for the application of procedural rhetoric to education, and as a means of advancing technical communication's study of games and their relationship with such emerging technologies.

Introduction

When roughly 58% of Americans (Entertainment Software Association, 2013, p. 2) and 97% of American youth (McGonigal, 2011, p. 11) play video games, it is inevitable that games would come to have an immense impact on modern society. It is thus equally inevitable that efforts would be made to better understand this pervasive new medium, and to adapt it for specific purposes. One such purpose is the use of games to enhance or replace traditional pedagogies. Research related to games and education tends to follow one of two tracks: gamification or serious gaming.

Gamification is defined as “the use of game elements and game-design techniques in non-game contexts” (Werbach & Hunter, 2012, p. 26). Such work operates under the premise that games are optimally designed to educate their players in a variety of ways in which more traditional classroom environments are not, and that, by designing classrooms to better emulate games, teachers can enhance the experience students have while learning. Sheldon (2012), for example, structures his classes to include cooperative “boss fights” in place of tests, and points and levels in place of grades, in order to increase student engagement. While gamification is noble in its aspirations, scholars such as Bogost (2011) and Layne (2011) have worried that gamification is often a purely cosmetic change: rather than truly adjusting the manner in which students are expected to learn and engage with material, it tends to focus simply on the addition of gaming tropes (such as points) that might make the same old memorization feel slightly less tedious, without actually adjusting the learning process in any meaningful way. Indeed, whether students are completing a test or a boss fight, earning a level or earning a grade, the task itself ultimately remains the same.

Other research has instead sought to introduce games themselves into the classroom as powerful new learning tools for students. Some of this work has appropriated popular titles in innovative ways. Squire (2011), for example, successfully utilized the game *Civilization III*, in conjunction with Q/A sessions and traditional lectures, to help teach a more systemic view of history. Related work has involved the creation of specially designed *serious games*, which are intended from the moment of their inception for use beyond entertainment. Examples of serious games include the *Reader Rabbit* and *Math Blaster* series. Efforts have already been made to introduce both types of games into the rhetoric and composition classroom (Colby & Colby, 2008; deWinter & Vie, 2008; King, 2008; Lacasa, Mendez, & Martinez, 2008; Sheridan & Hart-Davidson, 2008). Similarly, there has been a push in technical communication to pay more attention to the ways games and gamers engage in the effective transmittal of information (Eyman, 2008; Mason, 2013; Schmid, 2008; Williams, 2008). Serious games, however, suffer from many of the same criticisms levelled against gamification. Most serious games focus on advocating the particular goals of an institution, rather than encouraging critical thought (Bogost, 2010). They also differ little from traditional pedagogy in their emphasis on content; like gamification, serious games are often used merely as a way to dress up “boring” content, hoping to increase learning by making that learning feel less tedious and more fun.

In place of serious games, Bogost (2010) advocates for a third use of games in the classroom, which he terms *persuasive games*. Persuasive games are games that effectively mount arguments, rather than simply advancing a particular point of view. In this way, they are able to encourage critical thought and debate, not rote memorization of facts. Persuasive games operate through their *procedural rhetoric*, which uses carefully crafted game mechanics to advance an argument about how a particular system works (Bogost, 2010). For example, while a serious

game against the death penalty might present statistics and other facts about its harmful impacts, a persuasive game against the death penalty might instead require the player to attempt to utilize the death penalty, but have the rules of the game rigged in such a way that it will only cause more harm than good. Thus, unlike gamification and serious games, both of which are more concerned with the effective presentation of information, persuasive games shift the focus away from game content, and instead try to critically engage players via the underlying systems of play.

For my thesis, I elected to further explore the capabilities of persuasive games for technical communication by actually designing and creating one of my own. In so doing, my research question is twofold:

1. Why/how are games able to advance a particular view of technical communication?
2. What would/should/could a game for technical communication look like?

In order to answer these two questions, I first discuss the theoretical foundations of games in education, stemming from scholars such as Gee (2007), Squire (2011), and McGonigal (2011). As part of this discussion, I include examples taken from the aforementioned work in rhetoric and composition and technical communication, leading up to my initial idea to create my own game. Then, I explore one potential way my idea might be put into practice, relying primarily on Bogost's (2010) procedural rhetoric. I couple this with other concerns necessary for good game design, such as aesthetics and fun. Finally, I describe my actual experience creating a game, including various changes that occurred in my original plans. By offering up my final game, in conjunction with this descriptive account of its design, I hope to advance our knowledge of games and education, provide a case study of procedural rhetoric and persuasive games in action, and make the potential of games for technical communication a more concrete reality.

Idea: Why Games Are Good for Education

Before I discuss the design and implementation of a persuasive game for technical communication, I find it prudent to first establish a brief rationale for the use of games in education generally, and in technical communication specifically. In the last several years, there has been a recognized need for pedagogical adaptation, based on new technologies and the changing ways we utilize them to interact with the world around us. For example, the Obama Administration's 2010 education plan urges the adoption of "a strategy of innovation, careful implementation, and continuous improvement" in order to address emerging challenges (U.S. Department of Education, 2010). Similarly, when discussing 21st century literacies, the NCTE advocates the necessity of "the continued evolution of curriculum, assessment, and teaching practice itself" (NCTE Executive Committee, 2013). Games are one potential response to this perceived need for innovation, allowing for the exploration of new literacies in engaging ways.

Games and Learning

Gee (2007) claims that certain games are designed to encourage learning that "is not only active, it is increasingly critical" (p. 34). Players are forced to think through a particular problem, engaging with the given design space to discover a solution. This type of experience aligns with our goals to encourage critical thought in the classroom: much as we want students to be able to reason through a problem and offer a solution that matches the situation, so too must *World of Warcraft* players be able to examine a difficult encounter, consider their abilities and how those abilities might interact with the encounter, and forge a strategy for progression. As Sheldon (2012) notes, "Learning through play is not a new concept. It is the fundamental way young mammals acquire knowledge of the world around them" (p. 13). Gee (2007) frames this in terms

of semiotic domains: players learn how to overcome certain types of obstacles based on a conflux of conditions, and are then able to reapply this knowledge later in other situations. Since “meaning is material, situated, and embodied if and when it is useful,” players remains invested in what is going on, and knowledgeable of the contexts in which certain actions should take place (Gee, 2007, p. 87). This is hardly different from our desire as teachers to give students the critical skills they need to succeed at life outside of the classroom: rather than being told what they need to know, players learn through doing.

Much of this active, critical learning in games arises from the way they are designed from the bottom up to be as engaging as possible: while not all classrooms are explicitly designed to keep students engaged as they learn, every successful video game must be. McGonigal (2011), for example, notes that games “are teaching and inspiring and engaging us in ways that reality is not” (p. 4). By giving us more satisfying work with clear goals and constant feedback; the experience/hope of being successful; social connections to others via shared experiences; and a greater meaning arising from being part of something larger than ourselves, games can help engage us “deeply with the world around us – with our environment, with other people, and with causes and projects bigger than ourselves” (McGonigal, 2011, p. 50). Through such strategies, it is possible for games to engage players with themselves, creating an ideal space for eliminating the transparency of players’ existing values or beliefs; to engage players with work, inspiring them to persevere despite their failures; and to engage players with others, leading them to interact with particular affinity groups and learn to navigate different communities (Gee, 2007; McGonigal, 2011).

In other words, “good video games build into their very designs good learning principles,” making them powerful tools for education, and appropriate inspiration for our own

classrooms (Gee, 2007, p. 215). Squire (2011) notes how games are specifically designed with regards to progression, so that they increase in difficulty at just the right pace. This helps to create *flow*, an “intense, optimistic engagement with the world around us” which results from operating on the absolute edge of our abilities (McGonigal, 2011, p. 36). Not only does this ensure players’ continued interest, it also keeps players constantly adapting and learning new strategies as they encounter increasingly difficult challenges. Bogost (2010) credits the interactivity of games with their driving appeal and power. In tandem with what Gee (2007) claims about critical learning, Bogost (2010) argues that games allow players to act, experiencing something rather than simply being told about it: it is much easier to be engaged with something when you feel a sense of autonomy and control. Even more so than other games, persuasive games embody this notion by explicitly designing the very possibility space of the game in such a way as to advance the chosen argument (Bogost, 2010).

Existing Research in the Field

Given how games are already leveraging their design principles and interactive natures to engage players in a kind of learning, it makes perfect sense that it might be possible to adapt them for traditional education. In fact, efforts have already been made to do so in rhetoric and composition. Some of these efforts have involved the appropriation of existing titles. deWinter and Vie (2008), for example, found that the creation of in-game avatars in *Second Life* can be a powerful tool for demonstrating the fluidness and existence of power structures, both in the game world and without. Another study by Lacasa, et al. (2008) determined that combining simulated experiences from games like *The Sims* with exercises based on Vygotsky and Bakhtin helped open students up to increased levels of participation and new ways of thinking. Yet another study by Colby and Colby (2008) leveraged *World of Warcraft* to create an established discourse

community for which students could write. All of these efforts focused around utilizing game environments to help encourage a deeper, more critical understanding of the real world. Some scholars have even gone one step further, independently developing their own games for use in the composition classroom. For example, Sheridan and Hart-Davidson (2008) created a game called *Ink*, while King (2008) led the design of a game called *Rhetorical Peaks*. Both of these games encourage students to engage more with their writing by placing them in an imagined rhetorical situation and thus providing them with a structured goal.

While there have not been as many published efforts at integrating games into the technical communication classroom, scholars have nevertheless begun to explore the potential of games for the field. Because of their interactive natures, games are “complex rhetorical spaces where both players and designers engage in the solving of rhetorical problems” (Eyman, 2008, p. 244). This, coupled with their rising popularity and increased use, has made them perfect sites for technical communication scholarship. On the one hand, the gaming industry could benefit greatly from the expertise technical communication has to offer. Schmid (2008), for example, postulates that virtual worlds still rely heavily on text, despite their visual and procedural natures, to convey information to the player. He claims that “we can bring our expertise to these worlds and make them more usable. The newness of the field represents a groundfloor opportunity for us to contribute our sound principles and authorial expertise” (Schmid, 2008, p. 283). In addition, as this quote suggests, technical communicators can translate their expertise into new jobs by expanding their purview to include games (Williams, 2008). Finally, studying the ways communication occurs in and around games may help unlock new understandings of common topics like documentation or interface design by looking at them in a different setting. Such findings might then be translated to work in the real world or work with other forms of media.

Mason (2013) goes so far as to claim that “video games enculturate players into literacies that extend far beyond the game itself,” transforming players into fledgling technical communicators (p. 226). This is in line with Bogost (2010), who notes that “videogames are particularly useful tools for visualizing the logics that make up a worldview (following Gramsci), the ideological distortions in political situations (following Zizek), or the state of such situations (following Badiou)” (p. 74-75). Games can help us explore hidden boundaries, presenting simplified systems of complex issues in the real world.

My Game Idea

Stemming from my interest in games and their potential for education, I decided to undertake the creation of a game for technical communication as my thesis project. In this project, I wanted to explore what it means to be a technical communicator, and how the discipline is defined. I selected the book *Solving Problems in Technical Communication*, a collection of 19 short pieces by prominent scholars in the field, edited by Johnson-Eilola and Selber (2013), for this task. Given its purpose to provide “a coherent approach to understanding and solving problems and developing strategies that work in different types of communication situations” for “students who are learning about the field of technical communication,” as well as its variety of topics and recent publication date, it seemed a good choice for the basis of my project (Johnson-Eilola & Selber, 2013, p. 1). My goal, then, was to leverage the interactivity and engagement prevalent in games to help better convey the information provided in this book to students. It was not intended to serve as a replacement for more traditional instruction, nor even for reading the book; instead, I wanted to provide a supplement that could work in tandem with such methods, allowing students a preliminary engagement with the provided information, and/or enhancing their existing understanding of the field’s different components. It was also not

intended to convey the actual content of the book to players through any kind of narrative, but rather to condense what I had identified as the book's underlying arguments into the game mechanics of the final game system via procedural rhetoric (a process which I discuss below).

Based on my desire to make this information as engaging as possible to as many people as possible, I decided to make it more relatable to novices in the field by situating it within Greek mythology. This would allow me to select well-known Greek gods to serve as figures in the game, and to draw upon various myths for my narrative backdrop. It also led to my selection of the title *Hermes, Technical Communicator of the Gods*, since, in Greek mythology, Hermes was the messenger responsible for conveying information. Further, in order to make my game as widely available as possible, I decided to program it for the web, making it freely accessible online to anyone with an internet connection. This meant that my game would be made using a combination of HTML5, CSS, and JavaScript. The current incarnation of my game can be found online at RhetoricalGamer.com.

Development: Procedural Rhetoric and Design

Having determined that games can be viable tools for education, my next step was to decide on a methodology for my own game. Much has been written about the design of serious games and/or games for education: in addition to the previously discussed work by Gee (2007), Squire (2011), and McGonigal (2011), other theorists, particularly in psychology and education, have also taken up the mantle, elaborating on how game design can best be initiated to engage students with new material. However, I elected not to follow such strategies, and instead adopted procedural rhetoric (pioneered by Bogost (2010)) as the main design principle for my game. This decision was twofold. First, as mentioned in the introduction, Bogost (2010) distinguishes between serious games, which present content to be learned, and persuasive games, which present rules to be interrogated. In serious games, the point is for players to absorb the material, but in persuasive games, players are expected to engage with the game mechanics, question how they operate, and then decide for themselves what that means. I wanted to critically engage students with the field of technical communication, rather than encourage them to memorize certain facts about it. Second, I found the idea of procedural rhetoric intriguing and underutilized, and wanted to further explore its strengths and weaknesses as a viable methodology: thus far, surprisingly little work has been done regarding persuasive games. Designing my own game seemed like the perfect opportunity to begin such an exploration.

Procedural Rhetoric

Procedural rhetoric, or “the art of persuasion through rule-based representations and interactions rather than the spoken word, writing, images, or moving pictures,” utilizes processes

to embody a particular logic within the rule set of a game (Bogost, 2010, p. ix). Bogost relies on three primary concepts to define how this process occurs: unit operations, procedural enthymemes, and simulation fever. Unit operations are the fundamental building blocks of a game. Each unit operation is a rule combined with its visual representation and meaning. Bogost uses the example of a health bar: in many games, when your character is damaged, you lose a portion of your health. This symbolizes a fairly simplistic rule: if you are hit, you take damage; if you take enough damage, then you die. The unit operation for this rule includes the visual representation of a health bar that informs the player of the rule, as well as the meaning the player takes away from it: don't get hit. Every unit operation makes "a claim about how part of the system it represents does, should, or could function" (Bogost, 2010, p. 36). Most games have many different unit operations, all of which work relationally to make the game run in a meaningful way.

A procedural enthymeme is the argument created by a game through the interaction of the player with a given unit operation. Traditional enthymemes are informal syllogisms, with one premise necessary for the argument left out because it is assumed to be true. For example, the common enthymeme "I think, therefore I am" leaves out the premise that the presence of thought implies existence. Procedural enthymemes operate in a similar way: through the act of play, the player literally completes the logical argument represented by the rules within the game, filling in the premise that would normally be left assumed. For example, the unit operation of a health bar represents the rule that taking damage reduces your health, which will eventually kill you. However, until the player engages with that unit operation, nothing happens; the player needs to actually take damage for the rule to apply. It is only through the addition of play – what Bogost (2010) defines as "the free space of movement within a more rigid structure" – that a unit

operation becomes meaningful: that the procedural enthymeme is completed by the player (p. 42). Just as there are many different unit operations in a given game, so are there a plethora of procedural enthymemes that are integral to its success. Every game “could be seen as a system of nested enthymemes, individual procedural claims that the player literally complete[s] through interaction” (Bogost, 2010, p. 43).

Simulation fever is the crisis created in a player when he/she recognizes the gap between their current understanding of a system and the procedural representation of that system in the game. In other words, simulation fever occurs when the player completes a procedural enthymeme, only to find that the unit operation does not do what he/she thought it should, based on his/her experiences outside of that game (Bogost, 2010). For example, it is common in first-person shooters to advocate the interests of the American military: thus, players typically control an American soldier, fighting to preserve American values. Such actions do little to stimulate simulation fever, as this is what popular culture and conventions of the genre have led players to expect. If, however, players are instead forced to play as a foreign soldier, or face Americans as the enemy, then this might cause those players to pause and reflect on the change, noting the break in the expected pattern. It is through the creation of simulation fever that procedural rhetoric mounts an argument: “players are persuaded when they enter a crisis in relation to this logic,” and are motivated to “address the logic of a situation in general, and the point at which it breaks down and gives way to a new situation in particular” (Bogost, 2010, p. 333).

Taken together, these three components give us the terminology we need to understand how procedural rhetoric works: developers construct unit operations through the programming of specific rules. Players engage with these unit operations and enact procedural enthymemes through their play. If the logic of an enthymeme is familiar to the player, then they will accept

that rule and move on; if it is strange or unfamiliar, then it will create simulation fever in the player, causing them to stop and think about it in an attempt to resolve this conflict. The creation of simulation fever alone, however, is not enough to guarantee a convincing argument. Bogost (2010) outlines two additional requirements for the creation of effective procedural rhetoric: high process intensity and high interactivity.

Process intensity, a term Bogost (2010) borrowed from Chris Crawford, is the extent to which processes are emphasized over data. Since games are composed entirely of processes, their process intensity tends to be relatively high. Higher process intensity leads to greater vividness, and an increased likelihood of meaningful expression (Bogost, 2010). Interactivity (also taken from Chris Crawford) focuses on the extent to which the game and player are effectively able to communicate. Whenever the player performs an action in the game, the game “listens” to that input, “thinks” about how to respond by checking for the appropriate rule in its code, and then “speaks” to the player by enacting the rule. If there is a breakdown or weak component in this process, then meaning will be lost. In other words, the procedural rhetoric of my game will only be effective if it does something meaningful *and* the player is able to understand that meaning.

The main criticisms of procedural rhetoric arise out of its lack of attention to other, essential characteristics of playing a game beyond the rules of the system. Sicart (2011), for example, notes a lack of attention to the role of players and their actual, embodied experience within the game. Different players may engage with games in different ways; it is highly unlikely that every player’s experience will be exactly as the game designer intended, just as it is highly unlikely that every reader of a novel will develop the same critical interpretation. Klabbers (2011) expands on this lack of attention to players by noting the social aspects of many games, as

well as the presence of user production possibilities such as add-ons or mods. Procedural rhetoric as defined by Bogost (2010) only focuses on the interaction of the developer to the player, not vice versa. Additionally, all games are played at a certain time in a certain place; this makes it necessary to examine not only the rules of the game and the experience of players, but also the wider social and cultural contexts influencing that relationship (Voorhees, 2009). In order to fully understand how games are played, it is necessary to look at the relations between the rules of play, the actual experience of play, and how the player construes that experience (Voorhees, 2009). Procedural rhetoric also does not consider “what is typically cast in the shadow: the material agency or effectivity of nonhuman or not-quite-human things” (Bennett, 2010, p. ix). In other words, it doesn’t take into account the limitations of code and chosen technologies on what can and cannot be done in the design and creation of a given game, and thus, on what the procedural rhetoric can accomplish. Rein (2010) goes so far as to advocate the creation of a “post-procedural rhetoric” that investigates the broader social and cultural assemblages in which games operate.

Crafting My Argument

Based on Bogost’s definition of procedural rhetoric, I sought to reframe my goal into an argument: that Johnson-Eilola and Selber’s (2013) book presents a particular view of technical communication as a field, and that adopting this view will lead to increased professional success. I then had to decide how best to represent Johnson-Eilola and Selber’s (2013) ideas procedurally. In order to do so, I began by crafting several key unit operations to be at the core of my game: *Attention*, *Projects/Tasks*, and *Success/Time*. While these three unit operations by no means represent the entirety of the mechanics that occur in my game, I believe that it is from these three that the remaining unit operations are derived.

Attention.

Technical communication is primarily about project management, and requires “information design, user advocacy, and content and community management” (Johnson-Eilola & Selber, 2013, p. 51). It is no longer simply about writing and editing: instead, it requires that the technical communicator effectively divide his/her attention across a number of different tasks and concerns. In order to capture this, I decided to design my game as a resource management game, similar in style to a game like *Civilization*. In such a game, the player is given a limited amount of resources, and must correctly use those limited resources to advance. For my game, I decided to make the primary resource the player must manage “Attention.” Players must choose when/how to allocate their limited supply of Attention in order to play the game. As they assign Attention to different items, they will earn progress towards completing those items.

Projects/Tasks.

While technical communication is primarily about completing various types of work projects, the sheer depth of scholarship in the Johnson-Eilola and Selber (2013) collection shows that it is not that simple; in addition to simply completing projects, a number of other concerns are presented, such as multiculturalism, ethics, new media, rhetoric, and work tools. Each of these elements represents a vital concern for the technical communicator, and is critical to his/her success. Thus, for my game, players are able to assign their Attention to projects as their primary goal, attempting to complete them. However, players are also able to allocate Attention towards the completion of secondary Tasks, each of which will indirectly help them progress.

Success/Time.

Technical communication isn't exactly something you can lose, so instead, I wanted to make a game that would last for a certain frame of time, allowing players ample opportunity to

assign their Attention and try out different strategies to figure out what works and what doesn't. Thus, I elected to make the game last 100 Turns: enough time for experimentation, but still short enough to allow for multiple play sessions. During that time period, the goal of players is to earn as much Success as possible by completing Projects and other Tasks. After 100 Turns, the player's Success is totaled, and his/her score recorded. At that point, players may always play again if they want, in order to try and earn a higher score.

These three unit operations, working together, comprise the core elements of my game system: in the course of their work, technical communicators must spread their Attention across multiple Projects and Tasks in order to try and earn as much Success as possible over a period of Time. To ensure high process intensity, I knew that I needed to carefully consider each and every mechanic present in the game, allowing players a variety of different gameplay options. In order to ensure high interactivity, I knew that I needed to provide regular and consistent feedback on exactly what players' actions had accomplished. I also did what I could to address the previously mentioned concerns with the efficacy of this method. Sicart's (2011) concerns over lack of attention to the player were largely addressed by my efforts to allow for a variety of different play styles, all still tied to my central argument. Because the game would be single player, Klabbers' (2011) concerns over social interaction were largely inapplicable, though I did decide to include High Score functionality to allow players to compare their efforts. I addressed Voorhees' (2009) concerns over context by focusing my argument primarily towards technical communication students, and in particular, those working with the Johnson-Eilola and Selber (2013) collection. Finally, in order to address Bennett's (2010) concerns over materiality, I knew that I would have to consider the material constraints of the game as I programmed it, and adjust my procedural rhetoric accordingly.

Game Design: Aesthetics and Fun

Squire (2011) defines three primary concerns for educational game design: the art design; a focus on the experience over content; and the making of explicit connections to existing learning theory (p. 105). While not perhaps what Squire (2011) envisioned, in this case, my procedural rhetoric represents my adopted learning theory. This leaves me in need of addressing the art design, which I define here as *aesthetics*, as well as a focus on the experience, which I define here as *fun*.

According to Squire (2011), educational researchers spend far too little time paying attention to the aesthetic qualities of their games, putting in only the minimal effort required to reach passable levels of ugliness (p. 86). This is sometimes a result of the thinking that strong visuals may actually detract from the learning experience; Squire (2011), however, has found that good game aesthetics actually lead to an increased sense of enjoyment from the game. There is simply something more engaging about a game that looks nice. Beyond that, a certain level of aesthetics is required to make a game feel like a legitimate artifact, worthy of being played. Bogost (2010) frames this logic well: “The coupling of abstract processes to particular topics produces particular meanings that represent particular positions” (p. 243). In other words, it is impossible to look at the rules of a system disconnected from other factors: the ways in which information is dressed up and presented to the player matters a great deal in how they perceive it.

Games are also typically expected to be fun: “with games, learning is the drug,” but only so long as it remains enjoyable (Koster, 2005, p. 40). For Koster (2005), games stop being fun when they are too easy or too hard to understand, move too quickly or too slowly, or become meaningless or have been mastered (p. 44). A good game is “one that teaches everything it has to offer before the player stops playing” due to any one of those preceding factors (Koster, 2005,

p. 46). If a game isn't fun enough to hold a player's attention long enough for the player to grapple with the procedural rhetoric, then it doesn't matter how great the argument is: players will miss it entirely. For this reason, it is crucial that games are appropriately designed to maximize engagement via the creation of fun. Koster (2005) defines the elements for a successful game as a sense of space, a solid core mechanic, a range of different challenges, a range of available abilities, and some amount of required skill (p. 120). Further, he claims that a successful learning game requires a variable feedback system, an appeal to players of various skill levels, and a cost to failure (Koster, 2005, p. 122). All games will eventually become boring to anyone, but the adequate design of these elements can help keep players enthralled long enough for them to learn everything there is to learn in a given game.

Designing My Game to Be Aesthetically Pleasing and Fun

In order to ensure the aesthetic quality of my game, I decided to incorporate both sound and images, with background music to set an ambience, sound effects to indicate when certain events occur, and images to capture the player's imagination and play into the intended mythological narrative. Since these all needed to be of high quality (higher than I might be able to produce on my own), I opted to enlist the help of my more artistic friends in this endeavor. In order to maximize the fun, I focused on addressing each of the elements listed by Koster (2005). I wanted to create a sense of space by setting the game in Mt. Olympus, providing a backdrop to that effect. I trusted in my primary procedural rhetoric to serve as a solid core mechanic that held the rest of the game together through intuitive controls (left click to add Attention, right click to remove it). Via my combination of Projects with other Tasks, all of which required unique strategies, I hoped to allow for a range of different challenges, increasing progressively in difficulty as the game continued. This would also allow for a variety of different available skills

and a certain degree of customizability. For feedback, I would provide constant messages informing players of the effects of their decisions. By making the game impossible to lose, but contingent on the amount of Success earned, I hoped to allow for the play of both novices and experts, with novices struggling to learn the different mechanics, while experts strategically navigate them as they attempt to trump their old scores. Incorporating multiple ways to play, as well as a plethora of distinct projects, would ensure that there was adequate replayability. Finally, costs for failing projects would ensure the player is able to receive feedback about how well they did, with motivation to improve and do better next time.

Practice: *Hermes, Technical Communicator of the Gods*

As I began to actually create my game, I quickly learned that, even with the theory defined and a general plan in place, the entire process was often much messier than I expected and more driven by material constraints than I would have liked. Despite this (and perhaps because of it), I believe that my experience might allow others in the field to glean useful insights beyond the outlined theoretical foundations of educational games and procedural rhetoric. In order to better explain this experience, and the design and rhetoric found in my final game artifact, the following sections provide specific details about each element of the game. In each case, rather than simply relate the final product, I have tried to capture something of the process that went into that element, discussing my thoughts and aspirations, as well as my failings. These elements (in order) constitute the following: Code, Procedural Rhetoric, Interface, Sound, Visuals, Narrative, Menus, Projects, Tasks, Gods, and Mythos.

Code

I would like to start by offering a word of warning to others interested in following in my footsteps: coding is hard. It takes a lot of time, and is rife with constant problems, many of which have no clear solution. For example, I once altered the image files in my code, which broke the sound for about an hour, before it fixed itself for no apparent reason. I relied heavily on the website CodeAcademy, coupled with other independent reading online, to learn the basics of the programming languages I intended to use. As mentioned previously, I built my game in HTML5, utilizing the Canvas feature, which is essentially a box displayed on the page that contains all of the necessary information. I used a bit of CSS to build the website in which the game is housed,

and a bit of PHP to provide an option for feedback in that site. I used JavaScript to actually make the game. I opted to avoid jQuery, as I am not as familiar with it, and because, from an educational standpoint, I was more interested in learning the more basic JavaScript first. I did, however, find it necessary to rely on the CreateJS JavaScript libraries to supplement my own work. From this, I used EaselJS to more easily make discrete shapes in Canvas, SoundJS to activate the music and sound effects, and LoadJS to ensure the sounds were all loaded properly before being played. The site itself is currently hosted on a friend's server.

During the coding process, I found it useful to comment on anything and everything to help me remember what certain things did. Relying on Google searches helped me solve problems I otherwise had no idea how to tackle. As game design is an iterative process, I began the practice of habitually saving old copies of my game, just in case I needed to revert back to one. I also found it helpful to create a separate test page, in addition to the standard game page, in which I could try out various things on the site before making the changes live. Despite the difficulties I encountered, and a lot of backbreaking work, this experience nevertheless did help me learn how to code.

Procedural Rhetoric

Determining the exact method of instantiating the three unit operations I had developed (Attention, Projects/Tasks, and Success/Time) took a lot of time, and is the underlying cause of many of the design decisions discussed in the following sections. I began by intending to translate each of the 19 chapters in Johnson-Eilola & Selber's (2013) collection into a specific rule within the game. Thus, for example, "New Media" might be a type of project, "Multiculturalism" might be a special resource, and "Studying Rhetoric" might be a supplementary Task the player can perform to help him/her complete projects. However, this

kept getting in the way of the design: no matter how I tried to divide them, the resulting game always felt clunky, with either too few or too little options. Ultimately, what realigned my efforts was a quote from Bogost (2010): “Vividness comes not from immersion, but from abstraction” (p. 45). Bogost (2010) discusses how persuasive games are not about mimicking reality, but about best capturing what you want to argue is the essence of a particular reality. Koster (2005) recognizes this quality of games as well, stating that games are “iconic depictions of patterns in the world” (p. 34). In other words, it was not necessary for me to translate each of the discrete elements that comprise the field of technical communication into the game; instead, I needed to focus on reading between the lines in order to capture their essences. In the remaining sections, I discuss specifically how each of the elements helped to create the “iconic depictions” I desired; however, I believe that this breakthrough (a focus on the overall system over its specific parts) is what saved my game from mediocrity and a failed argument.

Interface

As part of my attention to aesthetics, I wanted to keep the interface as clean and streamlined as possible. Although I considered incorporating extra selection menus, I eventually decided that doing so would increase the game’s complexity too greatly. This self-imposed constraint (only include elements that can fit onto the initial game screen) forced me to occasionally become creative when adding new effects, and to preserve every inch of available space. It also played a surprisingly large role in dictating which abilities I could add, and which I couldn’t. For example, at one point I considered adding a dynamic skill tree, in which the player could unlock different abilities based on their preferred play style; however, since this would require a new window, the idea never actually made it into the game. The final interface (see Figure 1) remains sparse, while still managing to convey a surprising amount of information.

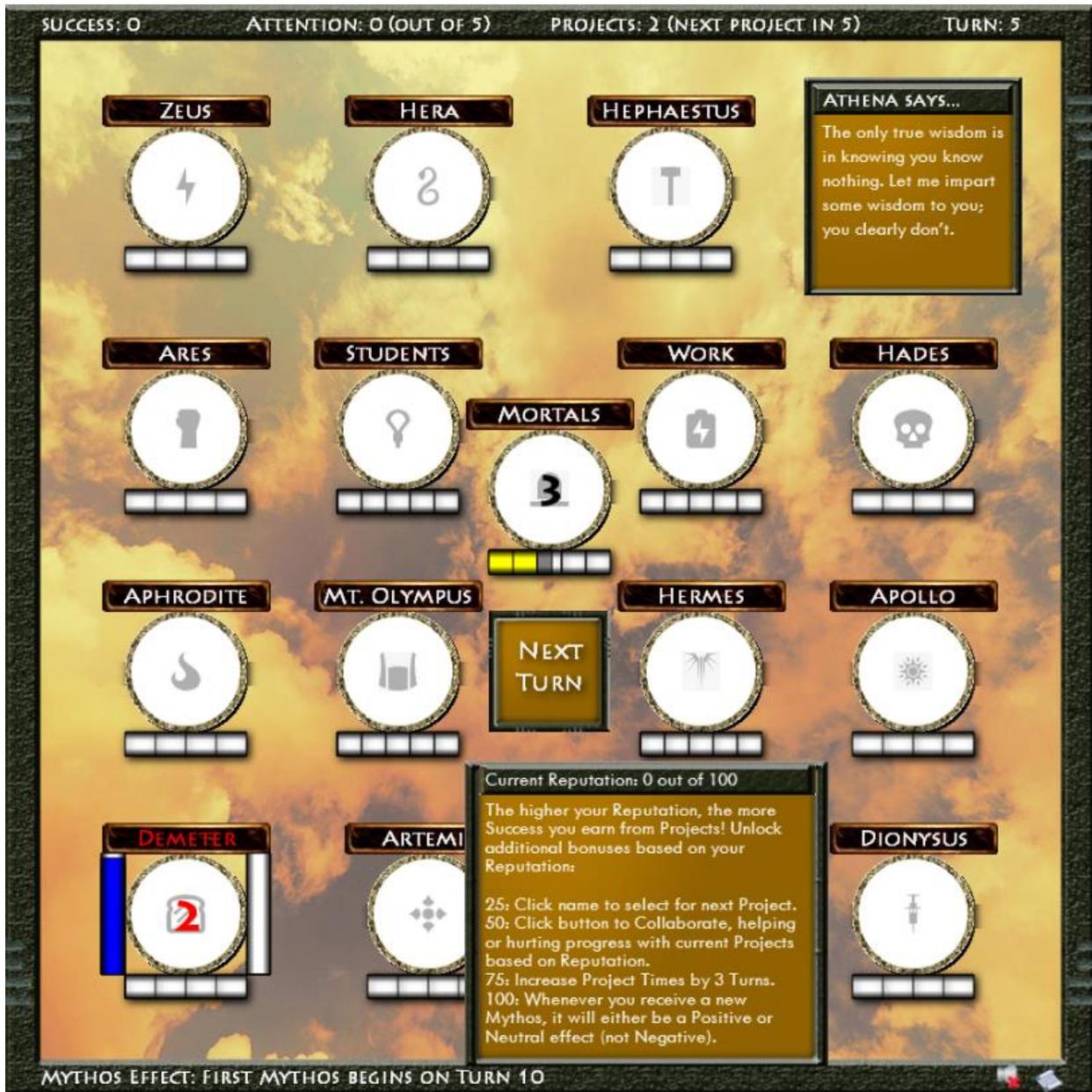


Figure 1. Game Interface

The top and bottom of the screen are used to convey the current state of the game, including the Turn number and the player's current score. They adjust dynamically as the game progresses. Along the outer edge of the square game screen are twelve circles, each corresponding to one of the Twelve Olympians (Hermes excluded). The inner set of five circles corresponds to five supplementary Tasks. The display of each circle changes, depending on progress made and its

current state. This can be seen on Demeter, who currently has an active Project, in the bottom left corner. The “Next Turn” button is situated in the middle of the screen to make it obvious and easily accessible. Because of the limited nature of the interface, detailed tooltips were added to almost every object within the game to help better convey that object’s purpose and use. These tooltips help ensure the kind of interactivity demanded by procedural rhetoric: without this information, it would be difficult for players to understand all of the complex rules underlying their play (Bogost, 2010).

Sound

Sound was one of the last things I added to the game, and required little effort to incorporate. I convinced one of my friends to compose the sound effects and music. The music plays on a loop in the background, and is intended to provide a classical ambience to fit with the intended aesthetics. At first, I relied on sound effects taken from Android devices, available under a Creative Commons license; however, these were simply stand-ins for the sound effects that have since been included. Almost every effect within the game, such as starting a project, failing a project, and clicking, is accompanied by a unique sound, similar to but distinct from the other sounds. Each sound is short, so as to not interrupt the play experience, and serves as immediate feedback to the player that their actions in the game world did something. One thing I quickly learned was necessary when working with sound is a Mute button, which has since been included in the bottom right corner of the game screen (see Figure 1).

Visuals

Unfortunately, I was unable to realize my aspirations for custom artwork: I lacked the time and skills to do it myself, as well as the funds to hire an outside contractor. Instead, I did the best I could using images I was able to acquire online (see Figure 1). For the background, I

utilized a cloudy sky, turned orange by the setting sun. While it is not exactly the glorious depiction of Mt. Olympus I'd imagined, I feel it still captures its essence. For the Gods, Tasks, and other in-game buttons, I utilized various symbols which, like the backdrop, I felt captured its essence, if not its exact identity. With the help of a friend familiar with Adobe Photoshop, I was then able to introduce limited design into the game. I used white text in place of black for its pleasing contrast with the orange of the clouds in the background. Also for contrast with the clouds, and to create a sense of solidness, I used a stony gray for the game screen border, as well as the nameplates, tooltip borders, and chat borders. Some detailing added to these elements helped give them the appearance of stone, and made them more aesthetically pleasing. To match the orange clouds, I utilized orange as the background color for most other pop-ups and menus within the game. Both for the aesthetic effect, and for the immediate feedback provided, I tried when possible to include dynamically updating visuals. For example, as can be seen in Figure 1, Demeter's text changed to red to indicate that she has a Project. Similarly, there are red shadows added to buttons when they are moused over, and the value bars of each God and Task change from red, to yellow, to green as they advance. For fonts, I utilized Lithos Pro for titles, headers, and larger display effects, due to its general correlation with traditionally-depicted Greek lettering. I used Metallophile Sp8 for the longer blocks of text, due to its legibility, even at smaller sizes. I discuss some of the more specific visual design choices of each element in that element's corresponding section.

Narrative

The narrative of my final game is mostly just a story hook: as Hermes, you are the technical communicator of the Gods. For the next 100 years, they expect you to complete projects for them to earn Success. After 100 years are up, your performance will be reviewed so

that you might reflect and improve. This narrative is introduced in the “How to Play” menu option, and is also concluded in the end game screen. Otherwise, the only narrative elements are embodied by the Mythos Effects, which help dictate current contexts for actions, and the Tasks and Gods, each of which has a tooltip providing general background (see Figure 2).

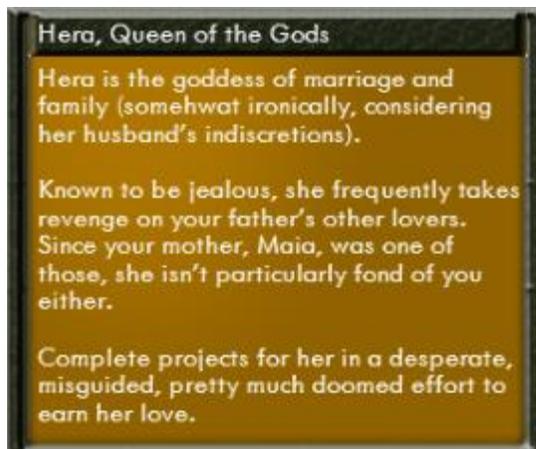


Figure 2. Tooltip



Figure 3. Chat Box

Such efforts, while providing only a limited description of the object, help to build the particular game world in which the player is engaging in their activities. Not only does this give the player a greater sense of purpose and immediacy, it also provides them with a partial story they can latch onto as they progress. Chat pop-ups serve a similar function: every Project is accompanied by a chat box when that Project begins and another when a Project ends (see Figure 3). Again, such effects, while not required for gameplay, help to better situate the player’s actions in a particular way (as well as introduce a bit of humor).

Originally, it was my intention to create unique chat pop-ups for each Project for each God, based on specific myths related to that God. This, however, quickly began to prove too time consuming for what I felt was too little reward. I next considered adding a smaller pool of more general responses for each god, one of which would randomly be pulled whenever a Project was

started, failed, or completed for that God. This cut the amount of work roughly in half, but still left a significant amount of dialogue to compose, with even less of a payoff. What finally worked well for me was to have each chat box dynamically update based on events that had occurred within the game. For the Gods, this meant creating different Project Start messages depending on the player's Reputation with each God. Thus, even from the chat, players receive feedback on their progress within the game. For Project End messages, as well as for Tasks, I opted to include references to resources gained or lost by that event. For example, an end chat for completing a Project states the Success and Reputation earned from that Project. In this way, the chat is able to serve an aesthetic, world-building purpose, while also enhancing the interactivity required for successful procedural rhetoric.

Menus

For my game, I provided both a Start Menu and an In-Game Menu. The Start Menu (see Figure 4) follows the same general aesthetic used elsewhere in my game. It provides the player with the ability to start the game, view information about how to play, view high scores, and view information about the game itself. The In-Game Menu instead includes options to resume the game, restart the game, and return to the Start Menu. Each sub-menu is designed to be easily scanned for information, and provides a way to return to the appropriate main menu (see Figure 5). Originally, I envisioned the How to Play sub-menu (Figure 5) as a set of detailed descriptions for each game mechanic, along with some general strategies for success. This proved to be difficult for players to navigate, resulting in most players skipping the information entirely. It also resulted in a kind of information overload. In its current incarnation, I instead opted to go for a sparser, cleaner approach, like that I used for the interface. This serves the dual purpose of not assaulting the player with too much information, and of remaining vague in order to force the



Figure 4. Start Menu

player to explore and experiment within the game. Since I have provided a plethora of detailed tooltips, I feel as though all the necessary information is already able to be found in game. The High Scores sub-menu stores each player's top scores, updating dynamically as the player acquires new ones. It is intended to help increase fun by increasing replayability, encouraging players to attempt to top their last performance. It also provides players with yet another continuous source of feedback on their progress. While I would like to provide the ability for players to compare their scores in-game, doing so is currently beyond my technical capabilities. Instead, only each player's local scores are recorded, although they are, of course, still able to then communicate those scores to other players. Finally, the About sub-menu provides a short blurb on my thesis, as well as credits and upcoming fixes.

Projects

Based on my interpretation of Johnson-Eilola & Selber's (2013) work, the main task of technical communicators is the completion of various projects. An active project can be viewed in Figure 1, granted by Demeter. Each project possesses four different elements: Time, Attention

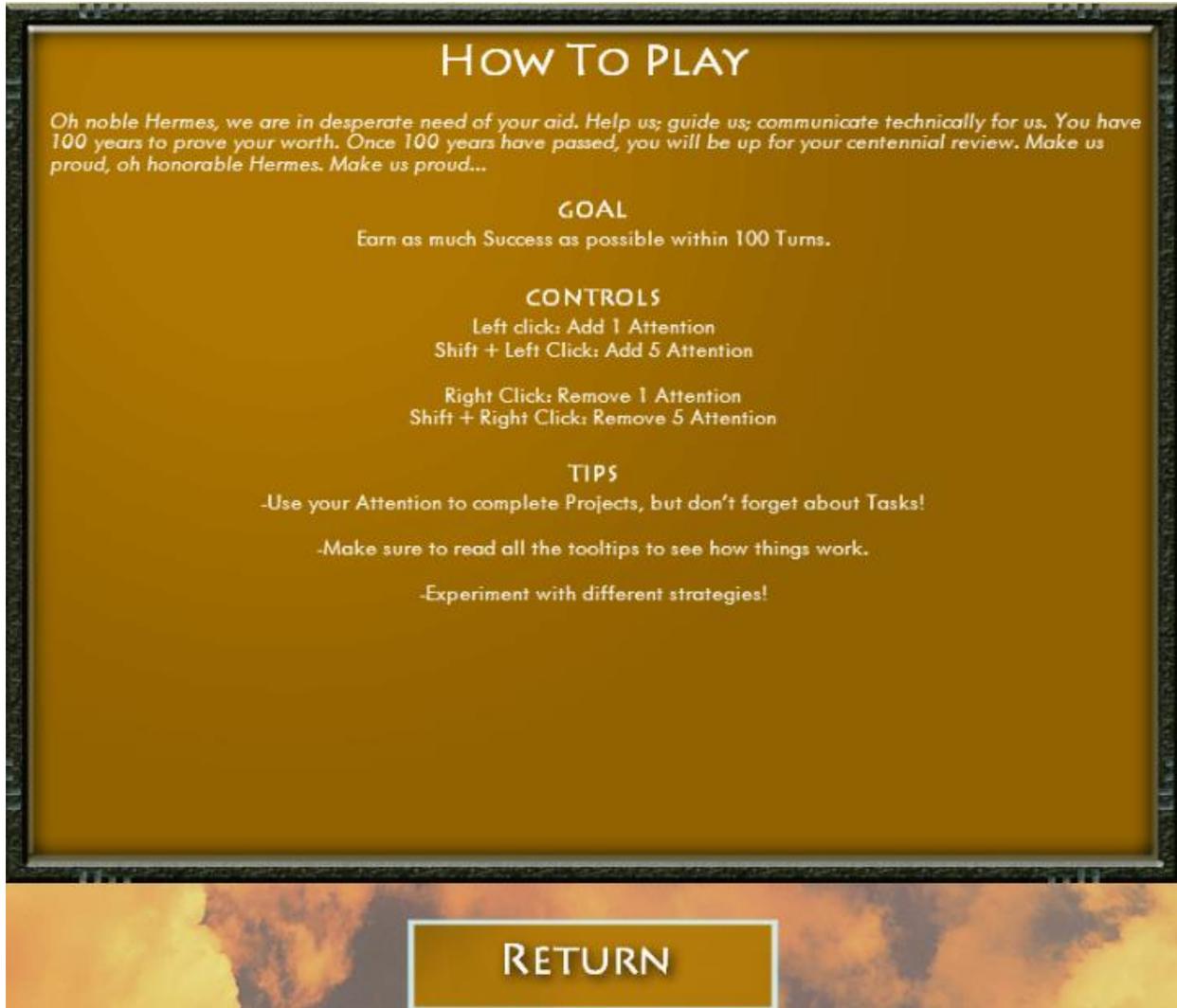


Figure 5. How to Play Sub-Menu

Required, Success, and Reputation. Players have until the specified amount of Time has passed in order to complete a given Project. To complete a Project, players must accumulate the Attention Required before they run out of Time. If they succeed, they gain the specified Success added to their score, as well as the specified Reputation with that God. If they fail, they instead lose double the specified Success from their score, as well as the specified Reputation. Because it can be difficult to judge how much Attention a Project will take before beginning, players are unable to see the Attention Required until they've already earned at least one Attention towards

that Project. Visual displays are provided for both the Time remaining, and the Attention accumulated (see Demeter in Figure 1). Initially, I had planned to have each God possess a unique collection of Projects, which required different special actions to unlock and complete. However, as already mentioned, this proved too unwieldy. It was also originally my intention to frame each Project as a particular kind of work possible in technical communication, such as a report, a speech, or a compilation of information. This similarly proved too time-consuming. Additionally, the more I worked on my game, the more I realized that it was less about the types of work technical communicators do, and more about the particular way technical communicators do it.

My next solution was to simply make all Projects random: rather than storing pre-made Projects in arrays for each God, I had Projects be generated with random values within a predefined range. This fix somewhat worked, reducing the amount of work to manageable levels, but it had the unintended side effect of making Projects bland, dramatically reducing replayability. It also eliminated any meaningful distinction between Gods. Both from a procedural rhetoric standpoint and from a game design standpoint, this seemed unacceptable. I eventually solved this dilemma by making it so that each God had a set Time for their Projects, with a corresponding Attention Required range. This helped restore distinctions between each God, since having to complete a 10 Turn project with Hephaestus felt very different from having to complete a 1 Turn project with Ares. I also set the Success and Reputation values to remain static across all Gods, for the sake of balance. Finally, to ensure ramping difficulty for continued player engagement, I programmed the Projects to automatically increase the Attention Required as more Projects are completed.

Tasks

As mentioned previously, I had initially envisioned a good dozen Tasks, each corresponding to a chapter from Johnson-Eilola & Selber's (2013) collection. However, once I had my procedural rhetorical epiphany, I realized that this kind of burdensome design was unwarranted. Instead, I then sought to deduce which main points were being stressed as primary focus points for technical communication across all chapters. Ultimately, I decided on the following five Tasks for my final game: Students, Mortals, Hermes, Mt. Olympus, and Work. The first three all represent different stakeholders in technical communication, beyond the boss who assigns a project. The training of students was an underlying theme throughout the entire collection, suggesting that it should remain a priority. There was also a standard concern for users – those who will be directly affected by the work – who were referred to as mortals in this instance to fit the intended narrative. Hermes represents the player, who is expected to continue improving and acquiring new skills. Mt. Olympus represents the workplace, within which technical communicators should strive to establish their place. Finally, work quality itself is integral to success, and performing better work can help lead to increased job opportunities.

The design of these Tasks has perhaps been the single aspect of my game that has changed the most over time. At first, the Tasks operated as passive buffs to all Projects, increasing based on how much Attention was earned in them, out of a maximum of 100. This worked to indicate their importance at first, but these passive bonuses offered only minimal feedback, resulting in reduced interactivity. This made it difficult to note their actual impact on Projects, so they felt largely ignorable. I decided to supplement the existing design by adding an additional bonus, specific to each Task, which was unlocked for every 25 Attention earned. While this made the Tasks feel more unique and powerful, the cap at 100 made it seem as though

the Tasks were only important up to a point, after which they had all been mastered, and were no longer a priority. Players tended to max them out early, then never touch them again. The bonuses also felt a bit clunky, with a wide variance in strength and some confusion over how exactly they functioned.

My ultimate solution was to scrap that system altogether, and to instead make each Task out of 25. For every 25 Attention earned with a Task, the player receives a different bonus based on the Task and its relation to technical communication. Completing Students simply grants a large amount of bonus success, supplementary to that earned from projects. Completing Mortals increases the Success earned from all future projects. Completing Hermes increases the player's Attention (the primary resource within the game). Completing Mt. Olympus increases the reputation rewarded from all future projects. Finally, completing Work causes an extra project to start. Once the player has completed a Task, the Attention in it is reset, allowing it to be completed again for additional benefit. In this way, players never lose the ability to work on Tasks, and remain encouraged to do so throughout the game. The widely varying bonuses granted by different Tasks also helps provide players with multiple strategies for success (a key requirement for keeping the game fun). At first, I made it so that any excess Attention assigned when a Task was completed carried over to the next use of that Task. I have since eliminated that feature, however, in order to encourage players to work with more than one Task at a time, and to more strategically consider where/how to best allocate their Attention.

Gods

The role of the twelve Gods is that of bosses assigning players Projects. This has remained consistent throughout all iterations of my game, as has each God's general, in-game personality. Players are able to earn Reputation (up to a maximum of 100) with each God.

Reputation is earned both by completing Projects, and by independently working with that God outside of Projects. Higher Reputation increases the Success earned by completing Projects with that God. At first, this was the only benefit to earning Reputation. However, more recently, I updated the Gods so that, for every 25 Reputation earned, a new special ability is unlocked. The first three of these abilities are the same for each God: the ability to select that God for future Projects, rather than having the Project assignment be random; the ability to collaborate with that God to try and gain bonus Attention towards active Projects; and an increase by 3 in all Time Requirements with that God. The final unlocked ability for each God, however, is a unique passive boost. Some of these boosts increase the viability of certain Tasks, while others simply provide a general bonus, such as increased Time on all Projects. The specific power granted by each God is based roughly on that God's characteristics, in keeping with the crafted narrative. They are also designed to add additional options for players, providing emerging complexity to the game as the player begins to choose with whom they wish to complete Projects based on the bonuses they most desire. This particular decision was made more for the enhancement of gameplay and the game design, as opposed to the procedural rhetoric.

Mythos

Mythos effects, like the God powers, are one of the more recent additions to the game, and are similarly an effort to extend the replayability of the game and to allow for emergent gameplay. Squire (2011), for example, discusses the power of unexpected occurrences in maintaining player interest by adding an element of luck and chance. The Mythos effects occur once every ten turns (beginning Turn 10), and introduce 1 of 30 possible effects into the game. Each effect lasts for 10 turns, ending once the next effect begins. While some of these effects offer only benefits, others are more neutral, or may even serve solely as a detriment to the player.

All of them are designed to significantly impact the gameplay, encouraging the player to consider the ongoing effect as they are deciding how to relegate their Attention. In addition to simply adding an extra element of challenging randomness to the game, these effects also serve the rhetorical purpose of encouraging players to consider the specific context in which they are working. Every Mythos effect is framed as a different condition, environment, or event. For example, one is framed as the Trojan War, while another is framed as an angered boss, and still another is framed as overwhelming work making the player exhausted. Examples of these Mythos effects on gameplay include bonuses granted to Projects completed for certain Gods, the inability to reassign Attention, and a timer reducing each Turn to only 15 seconds.

Conclusion

Despite all the difficulties I encountered, I believe that my decision to create a game for education, utilizing procedural rhetoric coupled with good game design, has been worthwhile. Personally, throughout this process I was able to learn a great deal about coding and the application of procedural rhetoric, which I believe will be invaluable to my future technical and theoretical work. Indeed, acquiring these skills was one of the original driving forces behind my desire to complete this project. Professionally, I hope that my experience can help further our understanding of both games and procedural rhetoric via this illustration by example. As should be evident to anyone who has played my game, persuasive games operate very differently from the more commonly discussed serious games. In many respects, they can be harder and more frustrating to play. They may also be more difficult to understand and study, especially in comparison with the original content on which the game is based. This is undoubtedly the case in my game: readers familiar with Johnson-Eilola and Selber (2013) may find it nigh impossible to discern exactly how each chapter and its major points are reflected within the game mechanics.

I do not believe that this is a failing of my game in particular, nor of persuasive games in general. As I have discussed above, it is not the goal of a persuasive game to teach content to students; instead, it is the goal of a persuasive game to engage students with a set of rules, force them to grapple with the implications of those rules via their play, and encourage them to derive their own content based on their experiences. In this way, persuasive games are capable of encouraging the kind of critical engagement for which we often advocate and, unlike serious games or gamification, may be able to offer a truly innovative pedagogical tool, rather than a

new coat of paint on an old one. Unfortunately, utilizing persuasive games effectively is not as easy as one might hope. This is because, even as the unique capabilities of procedural rhetoric expand the possibilities for critical exploration, they also result in serious design constraints. As I discussed above, the desired procedural rhetoric will often come into direct conflict with other aspects of the game design. When implementing a game mechanic, it can be a difficult decision to choose between crafting that mechanic to fit your desired argument, and tweaking your desired argument to make a better, more fun game. In these situations, it was my experience that even the best laid plans required constant, iterative adjustment, and that sometimes, the perfect procedural argument needed to be changed to better fit the game format. After all, changing one aspect of an argument can still preserve its central claims, but clinging to a pristine argument and presenting it as a game that has been rendered unplayable defeats the entire point.

There remain many future opportunities for research in this area. My own study is one of the only examples of procedural rhetoric in action I have seen framed as such, beyond Bogost's (2010) own, and is similarly one of the few examples I have seen of a game created for technical communication, or of procedural rhetoric utilized for education. Our knowledge could be greatly extended via similar efforts to mine, in which a practical example is used to learn more about how procedural rhetoric works. Future research might also examine different genres of games utilized for different purposes within technical communication, or in other contexts: after all, my resource management game to advance an argument of the field's organization is but one potential use of procedural rhetoric. One might just as easily imagine a role-playing game created to place students within an imagined workplace to better learn how day-to-day operations progress, or a puzzle game based around solving different problems which might arise as a technical communicator in order to better learn about the different types of work performed.

Research might also further explore the viability of procedural rhetoric through more detailed testing of new/existing persuasive games, as compared to more traditional methods. While my experience has been that it is entirely possible to create an educational game utilizing procedural rhetoric, as long as care is taken for its limitations and for other design characteristics, it remains to be seen how effective such games actually are in comparison to other pedagogical tools. I had originally intended to perform preliminary testing on my own game, but time constraints made this impossible. Such efforts might help us better refine what works and what doesn't in practice, and further address the criticisms leveraged against procedural rhetoric. Finally, work might be done to continue exploring the distinction Bogost (2010) draws between serious and persuasive games to see if there truly is one that might impact how we, as educators, should seek to utilize games in our classrooms.

It is my sincere hope that others will be able to make use of the game I have created. Even after my thesis is completed, I plan to continue working on the game to update its features and refine the procedural rhetoric based on the feedback I receive. As other scholars have noted, it is up to technical communication as a field to continue moving forward by investigating the exciting potential of games. As should be evident by this point, procedural rhetoric offers one such way the power of games might be harnessed.

References

- Bennett, J. (2010) *Vibrant matter: A political ecology of things*. Durham, NC: Duke University Press.
- Bogost, I. (2010). *Persuasive games: The expressive power of videogames*. Cambridge, MA: The MIT Press.
- Bogost, I. (2011). Gamification is bullshit. *Ian Bogost: Videogame theory, criticism, design*. Retrieved from http://www.bogost.com/blog/gamification_is_bullshit.shtml
- Colby, R., & Colby, R. S. (2008). A pedagogy of play: Integrating computer games into the writing classroom. *Computers and Composition*, 25(3), 300-312.
<http://dx.doi.org.ezproxy.lib.usf.edu/10.1016/j.compcom.2008.04.005>
- deWinter, J., & Vie, S. (2008). Press enter to 'say': Using *Second Life* to teach critical media literacy. *Computers and Composition*, 25(3), 313-322.
<http://dx.doi.org.ezproxy.lib.usf.edu/10.1016/j.compcom.2008.04.003>
- Entertainment Software Association. (2013). *Essential facts about the computer and video game industry*. Retrieved from http://www.theesa.com/facts/pdfs/ESA_EF_2013.pdf
- Eyman, D. (2008). Computer gaming and technical communication: An ecological framework. *Technical Communication* 55(3), 242-250.
- Gee, J. P. (2007). *What video games have to teach us about learning and literacy*. New York, NY: Palgrave Macmillan.
- Johnson-Eilola, J., & Selber, S. A. (2013). *Solving problems in technical communication*. Chicago, IL: The University of Chicago Press.

- King, M. (2008). Remapping Rhetorical Peaks: A video game for first year writing. *Computers and Composition Online*. Retrieved from http://www.bgsu.edu/departments/english/cconline/gaming_issue_2008/King_Rhetorical_peaks/index.html
- Klabbers, J. H. G. (2011). Tensions between meaning construction and persuasion in games. *Game Studies: the international journal of computer game research*, 11(2). Retrieved from http://gamestudies.org/1102/articles/klabbers_book_review
- Koster, R. (2005). *A Theory of Fun for Game Design*. Scottsdale, AZ: Paraglyph Press, Inc.
- Lacasa, P., Méndez, L., & Martínez, R. (2008). Bringing commercial games into the classroom. *Computers and Composition*, 25(3), 341-358. <http://dx.doi.org.ezproxy.lib.usf.edu/10.1016/j.compcom.2008.04.009>
- Layne, A. (2011). Gaming the system in a system of gaming: The inherent nature of games in pedagogy. *Not Your Mama's Gamer: I press X Therefore I am*. Retrieved from <http://www.samanthablackmon.net/notyourmamasgamer/?p=368>
- Mason, J. (2013). Video games as technical communication ecology. *Technical Communication Quarterly* 22, 219-236.
- McGonigal, J. (2011). *Reality is broken: Why games make us better and how they can change the world*. New York, NY: The Penguin Press.
- NCTE Executive Committee. (2013). *NCTE Framework for 21st Century Curriculum and Assessment*. Retrieved from <http://www.ncte.org/positions/statements/21stcentframework>
- Rein, A. (2010). Post-procedural rhetoric and serious games. *Digital Digs: An Archeology of the future*. Retrieved from <http://alex-reid.net/2010/03/postprocedural-rhetoric-and-serious-games.html>

- Schmid, (2008) Real text in virtual worlds. *Technical Communication* 55(3), 277-284.
- Sheldon, L. (2012). *The multiplayer classroom: Designing coursework as a game*. Boston, MA: Course Technology.
- Sheridan, D. M., & Hart-Davidson, W. (2008). Just for fun: Writing and literacy learning as forms of play. *Computers and Composition*, 25(3), 323-340.
<http://dx.doi.org.ezproxy.lib.usf.edu/10.1016/j.compcom.2008.04.008>
- Sicart, M. (2011). Against procedurality. *Game Studies* 11(3). Retrieved from http://gamestudies.org/1103/articles/sicart_ap
- Squire, K. (2011). *Video games and learning: Teaching and participatory culture in the digital age*. New York, NY: Teachers College Press.
- U.S. Department of Education. (2010). Introduction. In *National Education Technology Plan 2010*. Retrieved from <http://www.ed.gov/technology/netp-2010/introduction>
- Voorhees, G. (2009). The character of difference: Procedurality, rhetoric, and roleplaying games. *Game Studies* 9(2). Retrieved from <http://gamestudies.org/0902/articles/voorhees>.
- Werbach, K., & Hunter, D. (2012). *For the win: How game thinking can revolutionize your business*. Philadelphia, PA: Wharton Digital Press.
- Williams, S. (2008). 3D virtual worlds and technical communication: One more tool in the kit. *Technical Communication* 55(3), 239-241.