

---

# Placeholder Content in Game Development: Benefits and Challenges

**José P. Zagal**

Entertainment Arts & Engineering  
University of Utah  
50 S Central Campus Dr, Rm 3190  
Salt Lake City, UT 84112  
jose.zagal@utah.edu

**Roger Altizer**

Therapeutic Games and Apps Lab  
University of Utah  
50 S Central Campus Dr, Rm 3190  
Salt Lake City, UT 84112  
roger.altizer@utah.edu

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.

Copyright is held by the owner/author(s).

*CHI PLAY 2015*, October 03-07, 2015, London, United Kingdom

ACM 978-1-4503-3466-2/15/10.

<http://dx.doi.org/10.1145/2793107.2810319>

**Abstract**

Literature on game design and development recommends using placeholder content and assets as a technique for, amongst other things, working around production bottlenecks (e.g. needs music to work but music is not ready) and streamlining development (e.g. focus on the “fun”, without getting distracted by the art). However, this commonly accepted practice can cause problems that are rarely discussed or presented. We unpack this practice by discussing some of its issues. As a practice that is central to game development, it is important to better understand the interactions of social, technical, and cultural factors that result in the creation of sophisticated software artifacts, i.e. computer games. We discuss some of these issues and sketch out a research agenda for their further exploration.

**Author Keywords**

placeholder; placeholder assets; game design; game development; video games; temporary assets; dummy text; lorem ipsum; white box

## ACM Classification Keywords

K.6.3. Software development; K.8.3. Games

### Introduction

Game development is a multidisciplinary process requiring a wide-range of skills and abilities: art, design, programming, and more. It has been described as a particularly challenging form of “creative, collaborative and inter-disciplinary work” [8]. One of the many challenging aspects of game development has to do with production dependencies: coordinating the work of multiple people in such a way that no one is idly waiting for someone else’s work-product. For instance, in order for a programmer to write code allowing in-game enemies to throw tridents at the player an artist would, arguably, first need to create a 3D model of the trident. A common technique used to reduce these kinds of development bottlenecks is to use placeholder content, assets, and computer code.

Using placeholders is a technique that is integral to game development [11] and is widely used and recommended. The existence (or lack of) placeholder assets is also used as signifier for how along in the development process a particular game is [6]. Examples of placeholder use in game development include programmers recording saying the word ‘boom’ as a placeholder for a sound effect that is not yet available [10], designers using art assets from earlier projects [9], level designers using “white box” 3D models to test gameplay before final models and textures are incorporated [3], and interface designers using placeholder assets to test process flow and usability [1].

This work exists in the context of developing a deeper understanding of developer practices and experiences. We feel that by focusing on practices that are central to game development, yet often invisible or ignored, we can better understand the complex interactions of social, technical, and cultural factors that result in the creation of sophisticated software artifacts, i.e. computer games. Our goal is to argue that placeholder asset use often has a complicated role in game development and sketch out a research agenda for further exploring this role.

### Placeholders

What do we mean by placeholder? For the purposes of this work, Vaillancourt and Egli’s definition is suitable: “Placeholder assets are temporary resources used during the development of a game in place of final resources that haven’t been created yet” [11]. Placeholder assets are distinct from other byproducts of the game development process in that they occupy the place of final assets. They are distinct from transitory assets (e.g. works-in-progress) or other byproducts of the game development process. For example, sketches aren’t replaced by final art, they are used to help develop final art. So, a sketch is (usually) not used as a placeholder. Similarly, a draft of a game’s script will be refined into the final script, it is not used as a stand-in for the final script. Placeholders are not iterated upon, they don’t evolve, they hold a place in the production pipeline until the real assets show up.

#### *General Benefits of Using of Placeholders*

There are plenty of reasons for the using placeholders in game development. The main one is that they help reduce dependencies and bottlenecks. If a member of a development team needs an asset in order to

#### **General Reasons for Using Placeholder Assets**

##### **Address Production**

**Bottlenecks:** If there is a need for an asset that is not ready, use a placeholder asset until the final one is ready.

**Minimize Distractions:** If you want an audience to focus on a specific element use a placeholder asset to prevent distracting the audience.

**Quality Control:** Use placeholder assets as replacements for content that is not acceptable in the current version/build of the game to highlight what is not yet ready/finished.

accomplish her goals she can use a placeholder to avoid waiting for the final asset to be created. Placeholders are used for all kinds of media including text, animations, images, models, and sound. Similarly, production dependencies are often complicated and multi-directional – i.e. it is not the case that programmers are the only ones who require placeholder assets.

Placeholder assets are also used to minimize distractions and focus attention on specific elements of a game. An artist might use placeholder text (e.g. “lorem ipsum”) as a way to emphasize the graphical aspects being developed. It is known that people are unable to avoid reading a text when they see it and this can shift attention from the graphical elements to the meaning of the text [2]. Placeholder text that only looks readable can offset this problem.

Placeholders can serve as a mechanism for quality control. The build process, wherein source code and assets are converted into a standalone game program, can be configured such that if an art asset cannot be processed (due to an error) it is automatically replaced with a placeholder asset instead [7]. Reducing the impediments to produce a build, especially when they are not critical, is usually a high priority [8].

The above benefits are the ones most commonly referred to in the game design and development professional and academic communities. However, there does not seem to be much public discussion or reporting on the potential drawbacks and challenges that using placeholders entail. This is in contrast to our professional development experience, informal conversations we have had with game industry

colleagues, and our practice teaching game design and development to university students. In the following sections we use narratives from our experience to provide a preliminary overview of what some of the challenges of using placeholder assets are. As preliminary work, we hope to take the initial steps towards identifying more formal research questions that need to be explored in this area as well as outlining a more formal research agenda to pursue.

### **Challenges in Using Placeholder Assets**

The following narratives are based on our professional work as well as experiences relayed to us directly by game industry professionals. We have changed details for reasons of privacy while preserving the “lessons” learned from these experiences. Methodologically, this work is situated in the context of qualitative inquiry for the study of the development of media artifacts [e.g. 5, 12]

#### *Challenge 1: Lack of Placeholder Signaling*

A few years ago we collaborated with local partners in the development of a small-scale handheld game for young children. During the initial design meetings it was decided that the game should include 3D modeled animal characters. To facilitate future design meetings and remove production bottlenecks the team purchased a 3D model of a dog from an online assets vendor while the team’s artist designed the character to be used in the game.

The local partners fell in love with the placeholder dog.

From the moment they first saw the dog they “knew” that was their main character and all attempts to dissuade them from using a placeholder model as a

#### **Potential Issues Created by Placeholder Use**

**Non-Placeholder:** The placeholder is not perceived or understood as such leading to problems.

**Over-obvious Placeholder:** The placeholder is clearly perceived as such, but this causes problems.

**Placeholder Quality:** The placeholder’s quality exceeds that of non-placeholder assets leading to problems.

**Placeholder as Final:** A placeholder is never replaced with a final asset, leading to problems.

final asset failed. While this caused several issues specific to this project (e.g. the placeholder dog was being used in other projects, it was inconsistent with the style guide that was created) we quickly realized that we had stumbled upon a more general issue: we had failed to adequately signal to our partners that our placeholder was, in fact, a placeholder.

Because the placeholder was representational of the final asset (an animal), it became a distraction that derailed the design and development process. This was something that was largely uncontrollable – it does not seem to make much difference whether you tell people that an asset is a placeholder or not, they simply cannot avoid seeing it as an asset and are influenced and affected by whatever it represents. In other words, a placeholder that does not look like a placeholder can easily cause problems.

#### *Challenge 2: Overboard Placeholder Signaling*

For a recent project we had to develop a text-rich game for a partner institution. We decided to develop placeholder text that met some of our design requirements (e.g. approximate word length and count) while the final text was developed by our partners. Because we were concerned that our placeholder text would be confused with non-placeholder text, we made an effort to ensure that our placeholder text stood out and was clearly inappropriate by, for example, using outlandish (and often funny) words. The idea was that anyone reading it would immediately understand that it was neither a final or draft version of the text.

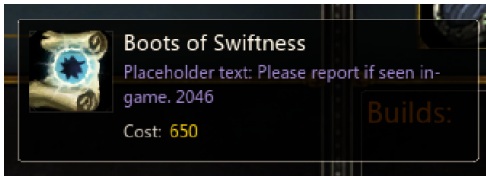
This effort of going overboard in signaling the placeholder status of the text was successful within the team and even had unexpected positive side effects.

Team members enjoyed reading and writing the placeholder text which increased morale and team bonding as they incorporated additional placeholder text that began to include “insider-jokes”. This is similar to Chandler’s recommendation for improving morale by using team members as voice actors for recording placeholder dialogue [4].

There were also unexpected side-effects. The placeholder texts’ brazen and purposeful signaling began to attract attention within the team. For example, pitches for features and content began to service the placeholder content and it became increasingly challenging to maintain focus on the game’s goals and the audience it was being created for. More generally, developing placeholders started to become a significant task, with the team spending too much time and effort designing them.

The second problem was slightly embarrassing. During a build-presentation for our partners, the joke placeholder content had been partially, but not fully, replaced with more serious placeholder content. Meaning, first, placeholder text was written twice, which was inefficient, and second, as our partners were not “in” on the jokes, they were confused by text in the build we were presenting on the screen.

Our efforts to signal the placeholder nature of an asset backfired because the “signaling” we used was the wrong method. It caused problems both internally as well as externally and it highlighted the challenge of creating placeholders that solve problems but at the same time are thematically “neutral”. In other words, a placeholder that is too different from the content it is standing in for can cause problems.



### Placeholder bug from League of Legends (circa 2011)

Image source:  
<http://forums.na.leagueoflegends.com/board/showthread.php?t=1148761>

#### Challenge 3: Placeholder Quality

A few game industry colleagues related an experience they had with a large-scale AAA title they were developing. Members of the engineering team needed high quality/definition assets in order to test their in-development game engine. At that stage in their production cycle the final specs (and limitations) for art assets had not been determined (e.g. size of 3D models) so the artists provided content that was high quality, but probably over-specified (e.g. higher resolution) for the final game. Unfortunately, other members of the team were not aware of the placeholder nature of the assets and, when the game was shown to the press, the assets drew considerable favorable attention. At this point the team was faced with a quandary – the game had now been seen by the public in a state that was “better” than what the original plans for the game were and the engineering team was unsure that achieving this new (and higher) quality would be technically possible in their current schedule. There was also uncertainty regarding whether or not it mattered if the game eventually shipped with art assets that were “worse” than those that had been initially shown. This uncertainty arose from game fan controversies surrounding some recent high-profile releases that had been extensively scrutinized with early screenshots compared to the released product.

As in the earlier examples we have discussed, the problem hinged on the fact that a certain audience was unaware of the placeholder nature of certain assets. This misunderstanding led to issues, but notably in this case, the problem was that the placeholders, while helping with a development problem (stress-testing a graphics engine), were of higher quality than the final

assets. In other words, the placeholder caused problems because it was too good.

#### Challenge 4: Placeholder Bugs

Perhaps the most commonly related story we have heard from industry colleagues has to do with the problems incurred by forgetting to replace a placeholder with a final asset. There are even community websites dedicated to documenting and tracking issues like these (e.g. The Cutting Room Floor, <http://tcrf.net>). Modern game titles are large, complex, and easily require thousands of distinct assets. Given the size, it is easy for some placeholder assets to “slip through the cracks” and appear in a shipped game title. When this happens this can cause embarrassment to the developers, but also affect the player experience and perceptions of quality surrounding the game (and the company that produced). In these cases it might have been better for the placeholder to not be clearly identified as such. A “draft” version of the final asset may have been better since it would have drawn less attention. In other words, the placeholder caused problems because it was clearly a placeholder.

#### Discussion

We do not argue that these examples are representative in a statistically meaningful way. However, we have hopefully argued successfully that there is an issue worth exploring. More broadly, we feel there is a lack of discussion surrounding the role that content (assets) can have as they are created and used in game development. Also, there is a need for a deeper understanding of how placeholder assets influence and affect the social, technical, and cultural processes of designing and developing games.

The term “placeholder” implies impermanence, and perhaps more generally, a degree of invisibility within the game development process. This invisibility belies the important role that placeholders have in facilitating game development as well as their potential to derail it. As noted by O’Donnell [8], there are a many processes and artefacts that, while critical to game development, are rarely perceived or understood from “the outside”. Placeholders and their use are one example of this and as such worthy of further study. Thus, we highlight some future avenues of inquiry we should pursue:

- 1) What best practices could we articulate for game developers in their use of placeholder assets?**

### **Acknowledgements**

We thank our students, industry colleagues, and partners for their stories, experiences, and insights.

### **References**

- [1] Adams, E. *Fundamentals of Game Design, Second Edition*. New Riders, Berkeley, CA, 2010.
- [2] Boytchev, P. Lorem Ipsum Logos. In *Proceedings of the Balkan Conference in Informatics (BCI 2007)*. Bulgaria, 2007, 107–118.
- [3] Byrne, E. *Game Level Design*. Charles River Media, Hingham, MA, 2005.
- [4] Chandler, H. *The Game Production Handbook*. Jones and Bartlett, Sudbury, MA, 2010.
- [5] Creswell, J.W. *Qualitative Inquiry & Research Design: Choosing Among Five Approaches*. SAGE Publications, Thousand Oaks, CA, 2007.

- 2) Some guidelines for placeholder assets have been developed in other areas, in what ways do these guidelines transfer (or not) to game development?**
- 3) What role does the type/media of a placeholder asset play in how it is used/misused?**

The challenges we have identified seem contradictory, perhaps even counter-intuitive. An obvious placeholder can cause problems just as much as one that is not. This suggests that more work is needed to better understand the subtleties and nuances involved in placeholder use. We look forward to further development of these ideas in a more systematic and thorough way.

- [6] Fisher, C. *Designing Games for Children: Developmental, Usability, and Design*. 2014.
- [7] Navarro, F. Taming the Beast: Managing Complexity in Game Build Pipelines. In M. Ansari, ed., *Game Development Tools*. CRC Press, 2011.
- [8] O’Donnell, C. *Developer’s Dilemma: The Secret World of Videogame Creators*. MIT Press, 2014.
- [9] Rouse III, R. *Game Design: Theory & Practice*. Wordware Publishing Inc., Plano, TX, 2001.
- [10] Szymczyk, M. and LaMothe, A. *Mac Game Programming*. Premier Press, 2001.
- [11] Vaillancourt, O. and Egli, R. Placeholders beyond Static Art Replacement. In E. Lengyel, ed., *Game Engine Gems 2*. CRC Press, 2011, 279–306.
- [12] Wimmer, R.D. and Domick, J.R. *Mass Media Research: An Introduction*. Wadsworth Publishing Company, 1991.