# Gender-Inclusive Quest Design in Massively Multiplayer Online Role-Playing Games

Anne Sullivan
Expressive Intelligence Studio
University of California, Santa Cruz
anne@soe.ucsc.edu

## Introduction

Studying the genealogy of Massively Multiplayer Online Role-Playing Games (MMORPG), it is easy to trace their lineage back to table-top RPG systems such as Dungeons & Dragons. Because these table-top games were built around a discrete system of rules and calculations, it was very simple to transfer these systems to their computer-based counterparts. Unfortunately, the majority of these calculations focused primarily on combat, leaving the dungeon master (DM) to integrate the players' non-combat role-playing efforts such as story-arc and character development into the game. While the math-based rules of the table-top RPGs were easily integrated into the computer role-playing games, it was not possible to convert the DM into a computer system [11]. With the loss of the DM, these games also lost their depth; typically becoming combat-centric systems with a thin story veneer and a shallow quest system focused on combat.

Even in some of the most popular MMORPG titles, such as *World of Warcraft* (*WoW),* the majority of the quests and experience received are for activities that revolve around killing. A warrior embarks on the same quests as a priest – the role-playing aspects of these games are relegated to strictly social activities such as chat.

To illustrate this further, let us examine *World of Warcraft*, which was released in 2004. The second expansion of *WoW – Wrath of the Lich King –* was released in late November 2008, four years later. One of the first regions (called zones) that a player could choose to adventure in is called Howling Fjord. In this zone, there are 133 quests available to an Alliance player. Breaking down these quests into commonly used categories [5, 7, 12] and then weighting the categories based on the experience received for doing each quest illustrates the continued combat-centric game design.

Of the 133 available quests, 55% of the quest experience comes from straight kill-based tasks of the type "Kill X number of enemies", "Kill a specific named enemy", or "Get X number of items from killing enemies (drops)." Furthermore, 22% of the experience is received from collection-style quests, most of which require collecting items from areas that are infested with enemies. Additionally, the only ways to gain experience in *World of Warcraft* are to complete quests, kill enemies, or discover a new

area. The experience received from discovering new areas only accounts for a minute portion of the overall experience received by a player.

Finally, none of the quests available in Howling Fjord are dependent on your class. In the entire game, 7954 quests are available (as of December 2008), but only 393 are class-specific – with at most 59 being available for a particular class [14]. This shows just how far MMORPGs have moved away from the core idea of role-playing within these games.

The lack of depth in these games is not the only issue. While many contemporary AAA titles involve combat-centric game motifs, the casual game genre has captured a different market. Casual games are defined by the IGDA Casual Games Whitepaper as "Games that generally involve less complicated game controls and overall complexity in terms of gameplay or investment required to get through game." These games cater to a demographic that is both older and largely women, with puzzle games topping the list of favorite genres, and none of the favorites including combat-style play [13]. The games industry has an obvious interest in increasing the number of women playing their games, yet they still create games that focus on combat-centric play, targeted at the 13-25 year old male gamer [9].

If women are interested in more puzzle-based games, as well as other games that fall under the casual game definition, it is a natural hypothesis that this style of play increases a game's gender inclusiveness. The MMORPG genre is a likely place to test this hypothesis, since the genre already attracts a broader player demographic than most games [15], and tends to have a built-in quest system.

Additionally, MMORPGs typically offer players a choice of class to play. Currently, player class has little impact on the game other than solo or group-specific combat dynamics. Integrating the different classes with role-playing opportunities is a natural hook for adding different modes of play. For instance, the warrior could still be a combat-centric character, rogues could have sneak-based quests, and priests could have heal-based puzzle tasks. Thus, exploring ways to include play styles into MMORPGs has the potential to simultaneously increase the gender-inclusiveness of such games, and to integrate more genuine role-playing (ala tabletop RPGs) into the actual *game mechanics* of MMORPGs.

The heavy authorial burden of implementing non-combat tasks forces game designers to implement combat-based systems out of necessity. Game engines already have standard technical solutions for solving combat-related problems (pathing, combat statistics, etc.) built into them. Therefore, combat-oriented quests can be represented in a data-driven way, with a minimum amount of quest-specific scripting. At an even higher level, games such

as *Diablo* are able to generate their dungeon levels and place enemies that are the appropriate level and difficulty for the player.

The necessity of keeping the authorial burden light-weight is demonstrated best by Bioware's in-progress MMORPG, *Star Wars: The Old Republic.* Bioware is very interested in creating a story-centric experience, with each class having its own set of quests that show a different subset of the overall story arc. However, they are accomplishing this by hiring over a dozen writers for years of full-time writing before the game entered pre-production [10]. This is not remotely feasible for many game developers. Therefore, technical research must be performed that significantly lowers the authorial burden of non-combat oriented quests.

## Current Work: QuestGen

QuestGen was created to help ease the authorial burden associated with creating non-combat based quests. It is the initial stage of a full quest authoring support tool and automatic quest generation system. While quest generation has been studied in the past [2, 8] these systems were aimed at creating quests for specific game engines. In contrast, QuestGen is a brainstorming tool for quest designers wanting to create *environmental puzzle* quests and are not designed for a specific engine. Here I use the term *environmental puzzle* to refer to a style of puzzle that requires finding a logical object in the environment to accomplish a goal.

These types of puzzles are very popular in hidden object and adventure style casual games. For instance, in the hidden-object casual game *Azada* (one of four games nominated for Downloadable Game of the Year in 2008 by the Academy of Interactive Arts & Sciences [1]), the first puzzle the player must solve is to find something to burn a piece of paper. As the player examines the room, they will find a book of matches in a trashcan.

This style of quest has two points of high authorial burden. The first is that a quest designer must hand-craft each of these logic-based environmental puzzles. Secondly, scripts or code must be written to integrate the puzzle into the game world. QuestGen was created to address these issues.

To create an environmental puzzle using QuestGen, the designer first inputs the location for the quest, which results in a list of concepts related to that location. Using these concepts, the designer then enters an objective for the quest. QuestGen uses the location and the objective to find possible solutions for the quest.

As an example, the designer may choose to create a quest that is located in a church, forest, cave, alley, or city. The designer can be as specific or general as they would like. Next, QuestGen creates a list of concepts associated with the location specified. If the designer decides they would like to create a quest for a cave, QuestGen will generate a list of concepts including *underground*, *dark*, and *mine* among others.

Seeing the property *dark*, the designer may then decide that they would like to generate quests where the objective is to light the cave. With QuestGen, once given the location, *cave*, and the objective, *light,* it will generate a list of possibilities of lighting the cave based on things a player might find there. For instance, it will return solutions such as fireflies, effervescent fungus, and glowing rocks, as well as the more typical solutions.

In unassisted quest authoring, the designer has to imagine properties of a location that the player might change (such as dark), and imagine ways to light a cave, typically generating lists such as flashlight, matches, or torch. Using a brainstorming tool such as QuestGen, designers discover interesting and unusual quest possibilities, such as finding pieces of the moon to light a dark area (a quest actually suggested by QuestGen), lowering the creativity burden of creating a large number of unique quests.

## Mechanics of QuestGen

QuestGen functions by leveraging the common-sense database ConceptNet [6] (along with WordNet [4]) to find links between quest-related ideas. ConceptNet structures its data by storing concepts as nodes which are connected to other nodes based on their relationships. There are currently twenty discrete relationship types, such as *PartOf*, *ConceptuallyRelatedTo*, *UsedFor*, *CapableOfReceivingAction*, and *LocationOf*.

ConceptNet was mined from a crowdsourced corpus of natural language sentences about the world, and consequently contains a fair amount of noise. Through experimentation it was determined that it was necessary to limit which link types we used, eliminating the most general link types, such as *ConceptuallyRelatedTo* and k-line, in order to not swamp generally useful quest suggestions in a sea of bizarre results.

Further, the author is able to define constraints on the relationship type connecting the last node. In this way, the author has more control over the paths that are returned. Without these constraints, QuestGen can return results such as this path between *underground* and *light*:

*Underground (LocationOf)➔ Sewer (CapableOf)➔ Carry (CapableOf)➔ Light*

However, if the author constrains the last relationship to the type *UsedFor,* they can narrow down the results to more easily find useful ideas such as the following result:

*Underground (LocationOf)➔ Rock (PartOf) ➔ Moon (UsedFor)➔ Light*

These constraints are not always necessary, and often the default relationship links return usable quests. For instance, choosing *church* as the location and *heal* as the quest objective. While many ideas may come immediately to mind, QuestGen generates this as one of its solutions:

*Church (LocationOf)➔ Music (CapableOf)➔ Heal*

The power of QuestGen is easily seen from these examples, where interesting examples of quest solutions are dynamically generated for the designer. With the help of the brainstorming tool, designers are able to create interesting environmental puzzles in a fraction of the time that is normally required for such brainstorming sessions.

## Proposed Work

QuestGen is still in progress with a number of extensions planned in the near future. Currently, the tool is only available for brainstorming a specific type of quest – the environmental puzzle. Before this is considered complete, additions will need to be made to QuestGen. The output of QuestGen currently shows nodes connected by relationship links. The author will be able to right

click nodes and select *deny* or *require* and then regenerate quests. If a node is *required* then all generated quest ideas must include that node. If the concept is marked deny, then it is disallowed in future generated runs.

In addition to the brainstorming capabilities of QuestGen, it is the goal to more fully support the quest design process. Once a designer creates a quest that they would like to save, QuestGen will offer an export function which will create an XML file with the quest information.

After the UI is completed, it is important to evaluate the usefulness of QuestGen. This will be accomplished by introducing the system to designers and requesting anonymous feedback. We will also interview designers and MMORPG designers to create future tools that will have a perceived benefit.

Other extensions planned for QuestGen include adding support for other quest types (logic puzzles, hidden objects, currently used quest types), and constraints based on class (generate quests that are applicable to healers, warriors, or rogues). It is likely that additional extensions will come from designer interviews and feedback.

Once the designer side of the tool is created, the next step is to integrate ConceptNet's common sense knowledge into a game world. This opens up the world to quests with multiple options for completion with low overhead on the designer's part. It also alleviates the issue many adventure games suffer, where the player must find the exact answer the designer was considering when other options may be more obvious or interesting to the player. Adding this flexibility makes the game world much more interactive and interesting, with quests becoming less task-based and more puzzle-based, something even the most popular MMORPGs suffer from [3].

With this addition, this also increases the feeling of co-authorship, something sadly missing from computer role-playing games [11]. The player is no longer following a pre-chosen, directed path through the world, but rather creating their own solutions and choices within the story they are creating. Increasing the choices increases the gender-inclusiveness of the game, since combat is no longer the dictated mode of play, but rather a choice of the player.

Furthering the idea of player co-authorship, data will be collected from a player during their sessions; what choices they made during quests, who they grouped with, or what they chatted about. We are already in conversations with a couple of major studios about gaining access to several terabytes of server-side gameplay data to begin experimentation with this kind of data mining. Using this data, it is possible to create a player history and use this to inform future quests. Not only will it be possible to extract preferred modes of play (what quests they prefer to complete) but data about their social nature will be available as well.

Creating a dynamic social network will allow for generated quests that are based on player preferences: who they group with, who they chat with, or how often they group. For instance, if a player often groups with another player, the system can generate a quest that requires communicating with that player, or perhaps doing a specific task with that person based on where they've been and what they've chatted about. With this in mind, the player will be able to feel like the main character of a story they are creating, instead of being forced into playing a pre-ordained story line.

## Expected Contributions

Overall, this work helps further the gender-inclusiveness of the MMORPG genre by allowing players the freedom within the game world to choose how they wish to play. This has the added benefit of increasing the perceived ability to role-play and feelings of co-authorship. Choices of class, race, and even gender may now have an affect on gameplay, making these choices something more than play mechanics during combat.

While gender differences have been studied in the past, this work has not been incorporated into game design. With this project, I hope to show that allowing choices beyond combat-centric modes of play increases gender-inclusiveness in a popular game genre without an insurmountable authorial burden. Additionally, this research will explore quest design and structure, and its role in increasing player co-authorship in stories.

## References

[1] AIAS 11th Annual Awards. 2008. http://www.interactive.org/awards/annual_awards.asp?idAward=2008 (December 18, 2008).

[2] Ashmore, C., Nitsche, M. 2007. "The Quest in a Generated World." In Proceedings of DiGRA 2007 (Tokyo, Japan, September 2007). 503 – 509.

[3] Corneliussen, H.G., Rettberg, J. W. 2008. "Quests in *World of Warcraft*: Deferral and Repetition." 167 – 184 in Digital Culture, Play, and Identity. Cambridge: The MIT Press.

[4] Fellbaum, C, (Ed). 1998. WordNet: An Electronic Lexical Database. MIT Press.

[5] Karlsen, F. 2008. "Quests in Context: A Comparative Analysis of Discworld and World of Warcraft." Game Studies Journal, Vol. 8, Issue 1.

[6] Liu, H., Singh, P. 2004 "ConceptNet: A Practical Commonsense Reasoning Toolkit". BT Technology Journal, v.22 n.4, p211-226, October 2004.

[7] Oh, G., Kim, J. 2005. "Effective Quest Design in MMORPG Environment", Game Developers Conference 2005, March 11, 2005.

[8] Onuczko, C, Cutumisu, M., Szafron, D., Schaeffer, J., McNaughton, M., Roy, T., Waugh, K., Carbonaro, M., Siegel, J. 2005. "A Pattern Catalog for Computer Role Playing Games." In Proceedings of GameOn North America (Montreal, Canada, August 2005). 30 – 38.

[9] Ray, S. G. 2004. Gender Inclusive Design. Expanding the Market. Hingham, Mass., Charles River Media, 2004.

[10] Remo, C. 2008. "A New Galaxy: Daniel Erickson On Writing The Old Republic." http://www.gamasutra.com/view/feature/3835/a_new_galaxy_daniel_erickson_on_.php (December 18, 2008).

[11] Tosca, S. 2003. "The Quest Problem in Computer Games." Presented at Technologies for Interactive Digital Storytelling and Entertainment (TIDSE) conference, in Darmstadt (2003).

[12] Walker, J. 2007. "A Network of Quests in World of Warcraft." In Wardrip-Fruin, N., Harrigan, P. (Eds.) Second Person: Role-Playing and Story in Games and Playable Media. Cambridge, Massachusetts and London, England, The MIT Press.

[13] Wallace, M., Robbins, B. (Eds.). 2006. "2006 Casual Games White Paper". IDGA.

[14] Wowhead.com. *World of Warcraft* database and tools. http://www.wowhead.com/ (December 18, 2008).

[15] Yee, N. 2006. "The Demographics, Motivations and Derived Experiences of Users of Massively-Multiuser Online Graphical Environments." PRESENCE: Teleoperators and Virtual Environments, 15, 309-329.