IRC Quest: Using the Commons Dilemma to Support a Single-Screen Game for Hundreds of Players

Dan John Moran
Queen's University
Kingston, ON K7L 3N6 Canada
9djm13@queensu.ca

Carey Metcalfe
Queen's University
Kingston, ON K7L 3N6 Canada
8caom@queensu.ca

T.C. Nicholas Graham
Queen's University
Kingston, ON K7L 3N6 Canada
nicholas.graham@queensu.ca

CHI PLAY'14, October 19–21, 2014, Toronto, ON, Canada.
ACM 978-1-4503-3014-5/14/10.
http://dx.doi.org/10.1145/2658537.2662991

Abstract
In this paper we describe the challenges of creating a game that can be played by large groups on a single display. Our solutions include the use of smart phones as game controllers using the standard IRC protocol, voting-based turn interaction, and automatically customized avatars allowing hundreds of players to appear on the display simultaneously. To provide meaningful gameplay for large numbers of people, the game is designed around a series of commons dilemmas.

Author Keywords
Game Design; Commons Dilemma; Massively Multiplayer Game

ACM Classification Keywords
K.8.0. [Personal Computing]: General - Games

Introduction
Since the earliest generations of home game consoles, players have enjoyed multiplayer games on a shared screen. Many of these games feature split screen, a graphics technique that divides a single screen into multiple rectangles to simulate multiple displays on a single screen. Coupled with the need for a game console controller in each player's hand, the prevalent

techniques for multiplayer games on a shared screen greatly limit how many players can play simultaneously. In IRC Quest, we demonstrate how literally hundreds of players can enjoy a game on a shared display. Players control an avatar by issuing commands from a smartphone or tablet that they likely already own. All players' progress through the game as a group to avoid having to divide up the screen. To coordinate players' actions, the gameplay is turn-based, based on a simple voting mechanism. To provide meaningful gameplay for large numbers of players, the game is based on a series of *commons dilemmas,* which put each player's best interests at odds with the group as a whole. This clash of individual strategy versus group strategy creates a unique gameplay experience.

Many Players on a Single Display

Supporting groups larger than the two to four players permitted in traditional games introduces several challenges. Most important is the control scheme. The early days of home game consoles had players stretching controller wires across their living room floor to support four players. Even with the recent transition to wireless controllers, requiring users to purchase a controller for each additional player limits how many players can play simultaneously. To counter this, we enabled any smartphone or tablet to become a controller, thus allowing people to play with a device that is likely already in their pockets. The game is controlled through Internet Relay Chat (IRC). Players join the game by adding themselves to the appropriate IRC channel. When a user joins, an avatar will be created for them and subsequent commands will control that avatar. Once the game has started, all of the avatars move together as a group.

Figure 1. The group moving together across the world

The group comes across choice nodes as they move. A binary choice is presented by game visuals and a text prompt (Figure 2). Players have thirty seconds at each choice node to make a decision by entering a keyword prefixed by a number sign in IRC. Players' choices are visualized by a bar graph that displays the ratio of one choice versus another.

Figure 2. The screen that appears at each choice node

For large groups, it can be difficult for an individual to keep track of their own avatar. To help players find the avatar they control, each avatar has floating text above

its head with the corresponding player's name as it appears in IRC. Avatars are also given a random assortment of equipment so that they vary in appearance. Once a player has identified their own avatar, they can recognize it later by its appearance or by its floating name tag.

The Commons Dilemma

Engaging players through binary decisions and text input is challenging. To create more consequential decisions we give the group a shared goal of defeating a dragon while promoting individual success for those who survive the encounter. An individual cannot succeed without the group, but helping the group may also lead to a player's demise. Example: The players come across a choice node where they must either fight a powerful enemy or attempt to sneak past. A few players can successfully sneak past while the rest of the group confronts the enemy, but if too many choose to sneak and the enemy isn't defeated, all players will be penalized.

Figure 3. Several players attempt to sneak past while the rest face a dangerous enemy

Such choices are labeled as a Commons Dilemma [1], in which short-term interests of the individual contradict the long-term interests of the collective. During the timed choice nodes, implicit negotiation takes place between players as they decide who will serve the group and who will help themselves. Players see a visualization of each other's choices, updated as players make or change their choice. This visualization provides meaningful communication for large groups that otherwise could not effectively communicate in a chat window. Creating friction between members of the group is what makes IRC Quest's gameplay compelling even with many players sharing a screen and limited to simple text inputs.

Implementation

IRC Quest has several major components: the IRC Bot, the Command Parser, the Avatar Manager, and finally the game world itself. The IRC Bot is a simulated user that connects to the IRC Channel just as any other user would. The bot idles in chat and reads all messages posted to the channel and passes them along to the Command Parser. The parser interprets players' messages and propagates valid commands to the Avatar Manager. This is a centralized manager of all player avatars that uses actions from the Command Parser to control player avatars. The avatars then move through the game world interacting with other game elements such as a way point system and enemies.

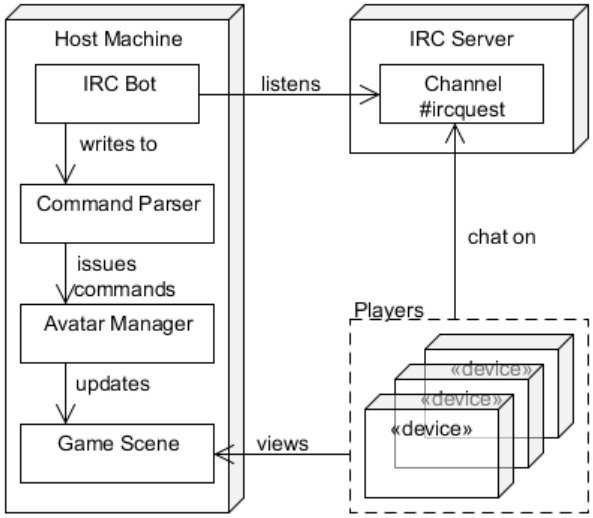

Figure 4. Architecture diagram of IRC Quest

The game itself was created using the Unity Engine and irc4net, an IRC library for the .NET framework. The IRC Bot and Command Parser were designed modularly, to allow use in any game implementing a multiplayer IRC based control scheme.

Experience

IRC Quest has been tested with as many as one hundred simulated users. This tested the amount of traffic that the IRC Bot and Command Parser can handle. Both components performed well under this load, providing feedback with no observable latency. Even with 100 users, individual players' avatars could be distinguished. We also trialed the game with several groups of up to ten real users. For these groups, we played on a large television, and players joined the IRC channel from their own smartphones, tablets and laptop computers. Players were enthused, engaging in vigorous discussion around decision-making, and laughing and cheering at the results. This experience provides early evidence of the success of the techniques we used to support large groups of players. Players reported that the randomized avatar appearance was useful in recognizing their own avatar, more so than the floating name text.

During the game's development, we considered the possibility of playing the game over a wide area network. To accomplish this, the host machine would stream the visual output to a popular media streaming service. Users could join the stream to view the game and participate by joining the associated IRC channel. We tested this using Twitch.tv: a streaming service primarily for games that includes an IRC chatroom with each stream. Unfortunately this service exhibited latency as high as fifteen seconds, making it impractical for real-time networked play.

In summary, IRC Quest demonstrates a new paradigm for multiplayer games. The success of the techniques we used encourages us to continue developing multiplayer games that don't conform to the definition outlined by the early generations of home video games.

Reference

[1] Cass, Robert C., and Julian J. Edney. "The commons dilemma: A simulation testing the effects of resource visibility and territorial division." *Human Ecology* 6.4 (1978): 371-386.