

# Usability Heuristics for Networked Multiplayer Games

David Pinelle<sup>1</sup>, Nelson Wong<sup>2</sup>, Tadeusz Stach<sup>3</sup>, Carl Gutwin<sup>2</sup>

<sup>1</sup>National Research Council  
46 Dineen Drive  
Fredericton, NB, Canada E3B 9W4

<sup>2</sup>University of Saskatchewan  
110 Science Place  
Saskatoon, SK, Canada S7N 5C9

<sup>3</sup>Queen's University  
25 Union Street, Goodwin Hall  
Kingston, ON, Canada K7L 3N6

david.pinelle@nrc.gc.ca, nelson.wong@usask.ca, tstach@cs.queensu.ca, gutwin@cs.usask.ca

## ABSTRACT

Networked multiplayer games must support a much wider variety of interactions than single-player games because networked games involve communication and coordination between players. This means that designers must consider additional usability issues that relate to group play – but there are currently no usability engineering methods that are specifically oriented towards the needs of multiplayer games. To address this problem, we developed a new set of usability heuristics, called Networked Game Heuristics (NGH), which can be used in the design and evaluation of networked multiplayer games. The new heuristics were identified by analyzing problem reports from 382 reviews of networked PC games, covering six main genres. We aggregated problem reports into ten problem categories (covering issues from session management to cheating to training for novice players) and developed heuristics that describe how these usability problems can be avoided. We tested the new heuristics by having evaluators use them and an existing set to assess the usability of two networked games. Evaluators found more usability problems with NGH, and stated that the new heuristics were better for evaluating multiplayer game usability. Our research is the first to present networked game heuristics that are derived from real problem reports, and the first to evaluate the heuristics' effectiveness in a realistic usability test.

## Categories and Subject Descriptors

H5.2 [User Interfaces]: Evaluation/methodology

## General Terms

Design, Human Factors.

## Keywords

Usability, networked games, multiplayer games, game usability, heuristic evaluation, Networked Game Heuristics, NGH.

## 1. INTRODUCTION

Most of the commercial video games that have been released in recent years support networked play, in which groups of people can play together from different locations. Current networked

games are extremely varied and cover several genres including first-person shooters, strategy games, sports simulations, and role-playing games. Many of the games support several different styles of cooperative and competitive play over the network, including free-for-all where players compete against each other, and team-based play where people work together toward a common goal (e.g., capture the flag).

Networked games involve interactions between players as well as interaction with the game environment itself, which means that these systems must provide support for communication, coordination, awareness, and social interactions between people. As a result, there is a much wider variety of usability concerns that must be addressed for these systems than in single-player games.

These additional usability issues that affect group gaming can collectively be called multiplayer game usability, which we define as the degree to which a player is able to successfully interact with other players and understand their actions. Multiplayer usability is not the only important issue in a networked game – but it is distinct from other aspects of game design that have been studied previously. First, a game's entertainment value is a well-known but separate concern, and involves issues such as engagement, challenge, fun, and storyline. Second, multiplayer usability is different from single-user usability issues that are important for all games, such as input mappings, control sensitivity, visual representations, and training support [35].

Evaluating multiplayer usability is an important issue, since poor usability can greatly detract from the overall play experience, even if other aspects of the game are well designed. However, there are currently very few methods for evaluating multiplayer usability. Playtesting is one of the most common ways to uncover design problems [12], but this method needs playable prototypes that usually only exist in the later stages of the development process. Few methods exist to allow designers to carry out less expensive usability inspections of games, and to also evaluate early, non-functional prototypes. Recent usability inspection techniques look at single-user usability issues in games [35,11,7], but there are currently no established methods for evaluating multiplayer usability issues in networked games.

In this paper, we address this problem by presenting a new set of heuristics for evaluating multiplayer game usability. The heuristics, called Networked Game Heuristics (NGH), can be used during heuristic evaluations, where evaluators explore an interface while looking for usability problems [31,32]. The heuristic approach has been successfully used in previous game evaluations because it does not make assumptions about task structure, and it is flexible enough to be adapted to specialized domains [32,10]. We developed our heuristics using a

Permission to make digital or hard copies of all or part 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. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GROUP '09, May 10–13, 2009, Sanibel Island, Florida, USA.

Copyright 2009 ACM 978-1-60558-500-0/09/05...\$5.00.

methodology introduced by Pinelle and colleagues [35], in which heuristics are derived from actual experience reports – in this case, public reviews of networked PC games. From these reviews, we derived a set of usability heuristics for assessing multiplayer game usability.

Our new heuristics, called the Networked Game Heuristics (NGH), are the first design principles to focus exclusively on networked game usability, and are the first to be based on a structured analysis of reported usability problems from a large number of games and game genres. A shortened form of the heuristics is shown in Table 1, and they are shown in more detail in Table 4. In the following sections, we describe the development of the new heuristics, and then report on a study in which ten evaluators used two different methods (NGH and Baker’s groupware heuristics [5]) to evaluate two networked PC games. Evaluators were able to find more multiplayer usability problems with NGH, and also reported that the principles were well-suited to the goals of the evaluation, were easy to learn, and were straightforward to apply.

**Table 1. Short summary of the Network Game Heuristics.**

- |   |
|---|
| <ol style="list-style-type: none"><li>1. Provide simple session management.</li><li>2. Provide flexible matchmaking.</li><li>3. Provide appropriate communication tools.</li><li>4. Support coordination.</li><li>5. Provide meaningful awareness information.</li><li>6. Give players identifiable avatars.</li><li>7. Provide protected training for beginners.</li><li>8. Support social interaction.</li><li>9. Reduce game-based delays.</li><li>10. Manage bad behaviour.</li></ol> |
|---|

## 2. USABILITY INSPECTION AND NETWORKED GAMES

Usability inspection techniques can play an important role in the design of networked video games. Most game design processes use playtesting to identify problems, but this is only possible during later stages of design and development when a functional prototype exists. Usability inspection techniques, where skilled evaluators inspect the prototype to find problems, are a potentially valuable addition to the game design process. They are inexpensive, and they can be used to evaluate prototypes at all stages of development.

Most current usability inspection techniques are not appropriate for games, since most current methods assume a different set of design goals than what are typically seen in games [35]. For example, many games are relatively unstructured compared with productivity applications, and promote exploration and varied playing styles rather than the completion of specific tasks. Further, one of the main goals of game designers is to engage and entertain the user – video games emphasize music, voice acting, storyline, and artwork, so they do not follow the conventions seen in other applications. This means that evaluation techniques focusing on concepts such as task sequences (e.g., [4]) and desktop design (e.g., [31]) are often not appropriate for games.

Literature on groupware evaluation provides insight into how usability inspection techniques can be developed for networked games. Groupware applications support multiple users, usually

while carrying out cooperative activities. Groupware applications are incredibly varied, but there are several persistent issues that designers have to address in most application areas. These include support for communication and coordination, and support for helping team members to maintain an awareness of others’ actions, locations, activity levels, etc. [16,37]. Gutwin and Greenberg [15] call these common interaction primitives the “mechanics of collaboration.”

Gutwin and Greenberg [15] state that evaluations of multi-user applications must address both taskwork, which covers support for single user actions, and teamwork, which covers the “work of working together.” Most techniques for evaluating the usability of groupware systems have addressed the teamwork aspects of design, and have left the evaluation of taskwork to single user evaluation techniques. Several techniques have been proposed, many of which have been based on the mechanics of collaboration. These include a set of usability heuristics for groupware [5], a task analysis technique for multi-user scenarios [37], and a multi-user walkthrough technique [36]. However, most of these techniques are based on the notion of structured task sequences and focus on cooperation between users, so do not map well to games.

The principles that we describe in this paper address “teamwork” (i.e. multiuser) issues in networked games and rely on the heuristic evaluation technique. Heuristic evaluation has been favored in recent work on game evaluation [e.g. 7,11,35] because it is flexible—it does not make assumptions about task structure, and it can be used in either formal or informal inspections. In heuristic evaluation, evaluators explore the interface while looking for problems based on a set of usability principles called heuristics [31]. They look for instances where there is a mismatch between the principles and the design, and record mismatches when they are real usability problems.

In the next section, we describe design principles (i.e. heuristics) that have been developed for evaluating networked video games.

## 3. DESIGN PRINCIPLES FOR NETWORKED GAMES

In recent years, several researchers have developed sets of heuristics that can be used to improve the design of video games. Some researchers have focused on single-user aspects of design, and have not addressed multiplayer issues in detail. For example, both Federoff [11] and Desurvire et al. [7] developed heuristics that include coverage of several design issues, including playability, game mechanics, and usability. More recently, Pinelle et al. [35] developed a heuristic set that provides detailed coverage of usability issues seen in single player games, and that addresses a range of issues including control sensitivity, input mappings, and training and help.

Many other researchers have investigated issues related to the design of multiplayer games. Much of this work has addressed technical issues that are important to the performance of networked games, including approaches for handling network traffic [39] and for designing distributed architectures [8]. Other researchers have covered issues related to making games competitive and engaging [3,17], and some have investigated how games can be designed to foster online communities [9,21].

Few researchers have considered how usability issues should be addressed when designing multiplayer networked games.

Researchers that have studied usability have focused on design for specific platforms or specific types of games, such as mobile games and massively multiplayer games. For example, Cornett [6] performed an exploratory study to identify usability issues seen in MMORPGs. He developed a list of 17 usability problems that were found while observing users playing games, and he presents potential solutions for each problem. The problem list covers many issues that deal with single-player usability, and others are specific to the role playing genre (e.g. “Attempts to buy/sell items were often unsuccessful”).

Korhonen and Koivisto point out that designing games for mobile platforms poses a unique set of challenges not seen elsewhere [23], and they developed a set of heuristics that address how networked multiplayer games can be designed for mobile devices [22]. Even though their heuristics focus on mobile games, many of them cover usability issues that are relevant to other types of networked games, including the need to support communication, the need to control address spoofing, and the need to help people find other players. Korhonen and Koivisto also address technical issues related to managing lag and network connections [22].

In summary, current literature does not address the problem of evaluating the usability of general networked games. In the next section, we describe our solution to this problem – a new set of heuristics specifically for multiplayer usability.

#### 4. HEURISTIC EVALUATION FOR NETWORKED GAMES

We created a set of heuristics that can be used to carry out usability inspections of networked multiplayer games. We developed the heuristics using an approach that is very similar to the one used by Pinelle et al. [35] to develop heuristics for single-player games. We created the heuristics by identifying usability problems reported in 382 reviews of networked multiplayer games, and then developing statements that encapsulate how the problems can be avoided by designers.

Using actual problem reports (from the game reviews) as the basis for the heuristics is vital, as it ensures that the heuristics will reflect real usability concerns. Games are incredibly varied, and significant design differences are seen between games and between game genres. Therefore, we felt that it was important that the heuristics represent the range of problems found in current networked games, and that that they provide both breadth and depth coverage of the design space. We addressed this by analyzing problems found in reviews of a large number of commercial games, covering six major networked game genres.

The new heuristics focus on multi-user aspects of the design, and are not intended to provide complete coverage of all usability problems found in networked games. They can be paired with heuristics that cover single-user aspects of design (e.g., visual representations, control sensitivity, and input mappings) or evaluations that deal with fun, playability, and entertainment value. The next sections outline the steps we used to develop the heuristics.

##### 4.1 Step 1: Identify Problems in Reviews

Our first activity was to identify usability problems that were described in reviews of networked games. The reviews were obtained from two popular gaming websites – GameSpy and GameSpot – that review a large number of games and that

maintain archives of game reviews. Reviews on both sites are written by professional editors and describe the quality of games, enumerating their strengths and limitations. In many cases, they cover usability problems (both single-user and multiplayer), providing a useful source of information on design problems that are found in games.

We analyzed all reviews of networked PC games that were posted on GameSpot in 2006 and 2007 and on GameSpy from 2005 to 2007. To guarantee that broad coverage of multiplayer usability problems, we included a large number of reviews with coverage of several major genres. The reviews in the set were written by 42 different authors. We included 382 reviews: 29 of the reviews (8%) covered persistent online games (including massively multiplayer online games), and the remaining reviews covered transient games (where players set up and control the game servers themselves). The distribution of genres in the game reviews were: Strategy 34%, Shooter 23%, RPG 17%, Sports 16%, Simulation 5%, Action 3%, and Other 2% (see Figure 1).

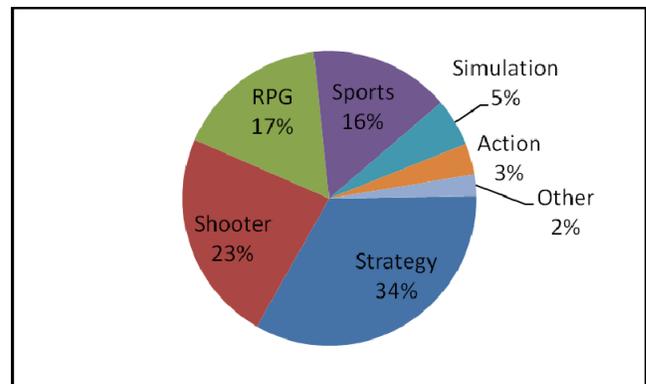


Figure 1. Distribution of reviews by genre.

Reviewers reported problems about multiuser aspects of the game in 213 of the 382 reviews (56%). From a first pass through this data, we further subdivided problems into those that are specifically concerned with aspects of the game’s design, and those that are caused at least partially by external factors such as network lag or server responsiveness. After this division, we had 112 reviews (29% of the total) that discussed problems clearly related to multiplayer game usability.

A typical example of a usability problem report can be seen in Madigan’s review of the game *Cold War Conflicts*. In discussing the matchmaking features, the reviewer states:

CWC does include a multiplayer option, but once you take a closer look you see that the developers are just taunting you. The matchmaking mechanism is all but nonexistent, requiring you to either be on a LAN or enter a computer’s IP address to connect. There’s supposedly a rudimentary search system, but it didn’t work [30].

Further analysis of these 213 reviews was carried out by three of the authors who have substantial experience with playing games and with conducting usability evaluations. Each person worked separately on the review sets, so that they could independently draw conclusions about the problem classes that were found. In this analysis, we identified the multiplayer usability problems in each review, and also coded other problems related to multiplayer features. We excluded problems related to other aspects of the

game (i.e., single-user usability, and fun and engagement). Each researcher devised informal categories based on the problems found, and individually coded problems from the reviews using these initial categories.

## 4.2 Step 2: Develop Problem Categories

After all reviews were coded, we discussed the problem classifications to develop a common classification scheme. There were initially 61 problem categories, but 18 were removed during the discussions. We aggregated the remaining categories based on similarities between problem descriptions, and identified ten new problem categories. We also created five categories for the common problems that were related to networked play but that were not true multiplayer usability issues. We then recoded the 213 reviews using the new problem categories, and identified 147 multiplayer usability problems and 187 related problems.

The usability problem categories are shown in Table 2, which briefly describes the problem covered by each category and lists a set of key issues that provide more detail on how the problem occurred in games from the review set. For example, the first category deals with session management, and the key issues involve “problems with finding, joining, or starting games.” The problem example listed in the last section was classified as a session management problem since it deals with difficulties encountered when trying to search for and join game servers.

Table 3 shows the additional problem areas that were frequently mentioned, but that are not directly related to usability. We include these because the problems were common, because they can have a serious impact on playability, and because they are related to multiplayer aspects of the games. The problem areas cover issues that have been previously discussed in CSCW, such as network latency, critical mass, and compatibility issues. We separated them from our main set because the problems are caused by elements such as the quality of a network connection, rather than the game design itself.

The distribution of game genres seen in our review set was skewed, with four genres accounting for 90% of the reviews. Problems from almost all categories were reported for each of the four most common genres:

- *Strategy* (35% of reviews): problems found in all ten categories, and in all five secondary areas.
- *Shooters* (23% of reviews): problems found in nine of ten categories, and in all five secondary areas.
- *Role playing* (17% of reviews): problems found in nine of ten categories, and in all five secondary areas.
- *Sports* (15% of reviews): problems found in seven of ten categories, and in four of five secondary areas.

There were fewer problems found for some problem categories than for others. For example, three problem categories (6, 9, and 10) have fewer than ten problems. These categories were included because they were found in multiple genres, and because they had a substantial impact on the usability of the games.

## 4.3 Step 3: Develop Heuristics

We used the problem categories and problem descriptions to develop ten usability heuristics for networked games. The heuristics, called Networked Game Heuristics (NGH), describe design principles that are intended to help designers avoid

common problems seen in games. Each heuristic is based on the problem categories listed in Table 2. For example, problem 7, which deals with inadequate support for training, became the heuristic “provide training opportunities where novice players are not subject to pressures from experts.” The heuristics are listed in Table 4. We did not create heuristics for the problems of Table 3. Each heuristic contains a paragraph that provides more detail on ways that related usability problems can be avoided. The paragraphs are grounded in details found in the game reviews. They elaborate on the basic concept stated in the heuristic itself, and were included to help designers map the heuristics to features found in game interfaces.

Our heuristics have some similarities to other heuristic sets presented in previous work (e.g., [5,22]); however, as described below, there are several differences, and it is clear that deriving the heuristics directly from problem reports in game reviews provides considerable specificity to the domain.

Baker et al. [5] presented a set of groupware heuristics that are based on patterns of collaborative interaction seen in the real world. The heuristics cover three main areas: support for verbal and gestural communication, support for maintaining awareness, and support for group coordination. Our heuristics address many of these same issues, but are more specific to games, and make the mapping between these issues and the game interface more explicit. Further, our heuristics address several issues that are not covered by Baker’s heuristics (such as matchmaking or cheating).

Several sets of game-specific heuristics have also been presented in earlier literature. Our heuristics have few similarities with the heuristics created by Desurvire et al. [7] and Federoff [11]. Our heuristics emphasize multiplayer usability, whereas previous heuristics primarily focus on playability and emphasize single player aspects of design.

There are similarities between some of our heuristics and those created by Korhonen and Koivisto [22]. Their heuristics address playability issues related to multiplayer mobile games, so although the focus is somewhat different, there is some overlap in four of the heuristics related to communication support, matchmaking, awareness, and deviant behavior. There is some divergence in the details of the heuristic descriptions, however, since their principles focus on games for mobile platforms; for example, their heuristics deal with design constraints related to mobile phones, such as text entry limitations.

We developed NGH using a method that was similar to the one used by Pinelle et al. [35]. They derived their heuristics by first analyzing usability problems found in public game reviews, but they included a different type of game (primarily single-player) and covered a different type of usability problem. Pinelle and colleagues’ heuristics address fundamental usability issues seen in video games, including consistency, customizability, input mapping, and control sensitivity. Our heuristics focus exclusively on multiplayer issues, so there is very little overlap between the two heuristic sets.

Several of our heuristics are completely new and are not covered in any other heuristic sets. For example, we define heuristics that address avatar design, protected training for beginners, and the need to minimize interaction delays. These issues have not, to our knowledge, been covered in any other game evaluation methodologies.

**Table 2. Usability problem categories. Categories are listed with key issues, example problems, and total problems found.**

Problem category	Key issues	Example	Total
1. Session management	Problems with finding, joining, or starting games	“Perhaps I'm not as bright as the folks who designed it, but all I want to do is to see a list of servers to join and I can't even accomplish that.” [29]	34
2. Matchmaking	Problems with finding friends, or with finding appropriate people to play with	“Even worse, pick-up groups only want to do the high experience value dungeons, which means that many players won't even be able to experience half the content that is there because they can't get a proper party together.” [14]	19
3. Communication	Poor support for text or voice chat; poor communication interface; not enough breaks in game play to use supported communication channel	“Another problem is the chat box: a flaw makes it impossible to type more than a short sentence into it, solved only by resizing the box to a larger size.” [18]	13
4. Coordination	Difficult to coordinate actions with other players	“Players who can form up into organized teams...might have better luck with actually following some sort of naval attack plan (maintaining a line of battle, etc.), but the game doesn't lend itself to organizing such tactics.” [25]	16
5. Awareness	Does not provide enough information on others' actions, locations, statuses, or availabilities	“Sometimes crazy things are happening to you and you don't know what's happening... It's as if you're playing Magic: The Gathering in real-time with no chance to read the cards as they're being played.” [26]	12
6. Embodiments	Avatars do not provide an adequate sense of presence, or are too similar and are easy to confuse	“You can race both point-to-point and multi-lap races, though your competitors' cars are ghosts that you can drive right through, so it doesn't much feel like you're racing anyone.” [34]	8
7. Training	Elite players make play difficult for new players; no offline training; no servers for beginners	“Matters get worse online where you'll play against tracks designed by TMU masters who have created mind-boggling challenges that will break the will of any gamer who isn't an elite TMU player.” [27]	15
8. Social opportunities	Size of game world is too small or too large for desired interactions; limited social opportunities	“...the tiny Tube stations are just too small to display more than a fraction of the player base, reducing the opportunity for casual meetings and making new friends.” [38]	15
9. Interaction timeframe	Slow online play (not related to latency); players forced to take turns or wait for other players	“Playing a turn-based game via multiplayer mode is always a tough sell...In Heroes V you have ‘Ghost Mode,’ which in a roundabout way allows you to move a ghost hero around and do stuff, but it's not really the same as simultaneous turns.” [1]	7
10. Cheating and unsavory behavior	Inadequate features to discourage cheating or negative game play	“Only about half of the servers online are running the Punkbuster anti-cheating measures, and players seem to have discovered an unlimited nuke cheat for the non-Punkbuster servers.” [24]	8
Combined total			147

**Table 3. Other common problems found in networked multiplayer games.**

Problem category	Key issues	Example	Total
1. Network latency	Network lag, where response times interfere with game play	“Latency was pretty high for an online game (the lowest ping I ever got was 200), and there were consistent issues with ghosting, where an enemy would simply vanish right before you double-tapped him.” [20]	49
2. Network connections	Problems establishing and maintaining connections, firewall problems	“Once I reached the server list, I repeatedly got cryptic error messages trying to connect to servers that seemingly had players fragging away, and had similar hiccups trying to use the built-in QuickMatch system.” [2]	35
3. Critical mass	Not enough players or servers online; game does not allow players to create bots to fill out teams	“You can have up to 28 players at a time but I could hardly even find 28 servers. Most of the servers I found were empty and the biggest race I managed to join only had six people playing.” [28]	71
4. Game play balance	Some teams, weapons, or units are too powerful; maps give unfair advantages to some players	“My biggest concerns, though, rest with how multiplayer is balanced. The heroes are simply far too powerful; in one game, three heroes utterly destroyed an entire enemy army.” [19]	15
5. Bugs and compatibility	The application crashes; compatibility problems where people cannot play together because of differences in game version	“Our only major issue with the PC version is that there's a known and nasty bug that causes the entire game to crash if you attempt to search for a game using the default settings.” [13]	17
Combined total			187

**Table 4. Networked game heuristics. Each heuristic is listed with a paragraph describing how common problems can be avoided.**

- 1. Simple session management: provide session management support that allows players to start new games, and that allows them to find and join appropriate games.** Non-persistent games require session management support so that players can start and publicize new games, and so that they can filter and join appropriate games. Session management features should allow players to broadcast information about the game they are starting, including the type of player they are interested in (e.g. expert or novice) and the type of game play they are supporting (e.g. cooperative, competitive, capture-the-flag, etc.). When players are searching for games to join, they should be able to selectively filter games listings so that they can find one that is suitable for them, and they should be able to obtain additional relevant information about game servers (e.g. ping, number of players online).
- 2. Flexible matchmaking: provide matchmaking features to help people find players with similar interests.** Games can provide two types of matchmaking support: they can help people find others inside large persistent game worlds, or they can help people find compatible people prior to starting a new game (usually in game lobbies). Designs should support the type of matchmaking that is appropriate for the game (for persistent or transient games, respectively). In both cases, support should be provided for discovering whether a player's friends are online and for helping to locate them. Games should also provide communication support and other features that help players find strangers who have similar game play goals or experience so that they can put together an ad-hoc group or game.
- 3. Appropriate communication tools: provide communication features that accommodate the demands of game play.** Most multiplayer games allow player to communicate using text or voice channels. The choice of channel and the design of the communication interface should be appropriate for the input demands of the game. In games that require constant interaction, it is often difficult to access communication features, so text communication tools that require mode switching can be difficult to use. Possible ways to reduce communication overhead include shortcut keys that trigger commonly used text phrases, open voice channels, and well-designed interfaces that do not take the user's attention away from events in the game.
- 4. Support coordination: provide features that allow players to coordinate their actions during cooperative game play.** When games support cooperative play, players usually must coordinate their actions to succeed. Designers should provide coordination features that allow players to seamlessly conduct shared activities in the game. Common examples of coordination features include interfaces for: distributing treasure among teammates, planning and communicating intentions, and managing command structures in teams.
- 5. Meaningful awareness information: provide meaningful information about players, including information about action, location, and status.** Games should supply players with enough information to allow them to understand the activities of others, and the information should be presented so that it is easy to access and understand. Awareness information can be presented either asynchronously or synchronously. Asynchronous awareness information includes information about players' statistics (e.g. wins, losses, points scored, etc.) or about their ranking. Synchronous awareness information helps people understand what is currently happening in a game, and examples include features to inspect other characters' inventories and features to help track teammates (e.g. mini-maps and other visual indicators).
- 6. Identifiable avatars: use noticeable and distinct avatars that have intuitive information mappings.** Many games use avatars to convey information about player presence, location, and movement. Avatars usually have several visual indicators that encode information about each player (e.g. avatar color can be used to represent team, and size can be used to represent the character's strength), and selected mappings should be simple and easy to interpret. Avatars should also be noticeable, providing a strong sense of presence so that players know when others are nearby, and they should be visually distinct so that players do not confuse team membership and player identity.
- 7. Training for beginners: provide training opportunities where novice players are not subject to pressures from experts.** In cooperative multiplayer games, players usually expect their teammates to actively contribute to the success of the team. When players are new to a game, they may not have the expertise needed to succeed, causing them to be badly outmatched and opening them up to criticism from more experienced players. Games should be designed to provide safe training opportunities for new players, allowing them to develop game skills either offline or on servers for beginners, so that they can avoid criticism and hostility from expert players.
- 8. Support social interaction: provide support for planned and opportunistic social interactions.** Several factors help to promote social interactions in games, including the use of space in game worlds and the presence of features that encourage cooperation. Game worlds should be designed so that they are not too small or too large for the expected number of players, making it difficult to track conversations or to find other players, respectively. Games should also provide features that encourage conversation and cooperation between players. For example, features that allow people to auction their items can promote trading between players, and features that allow people to broadcast messages to other players in the vicinity can help to encourage teamwork between strangers.
- 9. Reduce game-based delays: minimize interaction delays by reducing temporal dependencies between players.** Multiplayer games can introduce temporal dependencies between players in two main ways. First, they force turn-taking, where one player must wait for another to complete their turn, and second, they decrease performance for all players when one player has a slow network connection. When temporal dependencies interfere with the flow of the game, designers should adopt strategies to minimize them. Possible ways to deal with turn-taking delays are to use simultaneous turns or to limit the amount of time players can spend on a turn. Possible ways to improve network dependencies are to use client side prediction, or to only penalize the player with network problems, not the entire set of users that are logged into the game.
- 10. Manage bad behaviour: provide technical and social solutions for managing cheating and unsavory behavior.** Online game play often suffers from problematic behavior, such as when players modify code on the local client to gain an unfair advantage, work against teammates, violate game play conventions (e.g. spawn killing), or use inappropriate language. Technical solutions are often implemented to prevent cheating or to filter obscene language in text-based communication, and they should be used to reduce occurrences of these problems. Games should also provide social solutions, by helping people recognize cheating and other unacceptable behavior, and by providing mechanisms so that they can remove or sanction offending players.

## 5. Evaluation

We conducted an evaluation where ten people used NGH to evaluate two networked multiplayer games. Our goals were to determine whether knowledgeable evaluators could understand and operationalize the heuristics and whether the heuristics would be useful at identifying multiplayer usability problems. We also wanted to evaluate the areas of coverage seen in the set to see whether the heuristics were effective at specializing the evaluation process for networked games. Last, we wanted to determine whether the heuristics improved on current evaluation approaches by comparing them to an existing set.

We compared our heuristics to Baker et. al's [5] groupware usability heuristics. Baker's heuristics are based on the 'mechanics of collaboration' [15] and address common issues in shared tasks, including the need for verbal and gestural communication, and the need to coordinate cooperative actions. We chose Baker's heuristics as a comparison case since they address generic collaboration issues that are important in most multi-user applications.

### 5.1 Method

We recruited ten participants who had previous experience with usability evaluations, and experience with multiplayer networked games. The participants evaluated two networked video games. Participants were divided into two groups of five. One group used our heuristics to evaluate the first game and then Baker's heuristics to evaluate the second. The other group used the heuristic sets in reverse order. We balanced the groups according to their experience with usability evaluation, with an equal number of expert and novice evaluators in each group.

Participants evaluated two multiplayer games: BZFlag ([www.bzflag.org](http://www.bzflag.org)), a 3D tank game, and Tee Wars ([www.teeworlds.com](http://www.teeworlds.com)), a 2D side-scrolling shooter. We used open source games in the evaluation because they were not professionally developed and were continually under revision by members of the development community. This means that they had design flaws that were similar to those seen in functional prototypes, making them well suited to evaluation with usability inspection techniques.

Prior to each evaluation, we trained participants on the heuristic sets that they would use during that session, and on classifying problem severity using Nielsen's scale [33]. Participants were given printed copies of the heuristics that they would use to evaluate the game, and they were given forms for recording usability problems.

During each session, evaluators were asked to spend 15 minutes playing and familiarizing themselves with the game, using any of the public servers available through matchmaking features. We then set up a testing server so that participants would be able to assess both competitive and cooperative aspects of play, and we asked them to log into our server for the remainder of the session.

We asked participants to individually inspect the game interface using the heuristics. When they found a mismatch between the heuristics and the game interface, they were asked to record the mismatch if they felt that it was a usability problem. They were instructed to describe the problem, record the heuristic that they used to find the problem, and give each problem a severity rating

using Nielsen's severity scale [33]: 1-Cosmetic problem, 2-Minor problem, 3-Major problem, and 4-Usability catastrophe.

The study lasted approximately 3½ hours, and participants had roughly 90 minutes to evaluate each game. At the end of each session, participants completed a questionnaire that asked them to describe the strengths and weaknesses of the heuristics that they used. Once participants completed both evaluations, they were given a final questionnaire that asked them whether they could identify problems more effectively with one heuristic set or the other, and they were asked which set was easiest to use.

### 5.2 Results

We analyzed the problem descriptions, removing problems that were repeated by the same evaluator, or that did not describe real multi-player usability problems (e.g., commentary on gameplay). We counted the number of problems found using each heuristic, and we tracked the severity ratings that were assigned to each problem.

Participants found a combined total of 49 usability problems using Baker's heuristics (mean of 4.9 problems per evaluator), and found 67 problems with NGH (mean 6.7). The severity rating of problems for both sets was similar, with an average severity for Baker's heuristics of 2.39, and 2.30 for the NGH set.

#### 5.2.1 Summary of problems found

Evaluators found a large number of problems in both games, and it is clear even from our simple study that multiplayer usability evaluations could have a major impact on the success of a networked game. Although it is not our goal to report on the specific evaluations of the games themselves, we briefly summarize the main problems found before analyzing the differences between the heuristic sets.

*BZFlag.* BZFlag is a 3D tank game where people maneuver through the game world while shooting other players. Participants found a variety of problems, but their greatest frustration came from the absence of offline training and protected servers, since expert players logged into the evaluation server and dominated game play. People also found problems related to communication support, and had particular difficulty using the text chat due to the real-time demands of the game. Other problems were found with the game's support for coordinating strategies with team members, understanding what is happening in the game, tracking teammates, and interpreting embodiments in the mini-map.

*Tee Wars.* Tee Wars is a 2D game where players move their character through the game world, attacking other players with an assortment of weapons. People found a wide range of problems, including issues related to finding appropriate servers and players with similar expertise levels, and limited opportunities to train offline. As in BZFlag, people found it difficult to find time to use text chat. They found a range of problems relating to awareness and coordination support, including problems with identifying teammates, the absence of a mini-map, and an inability to coordinate strategy with teammates.

#### 5.2.2 Differences in the problems found with the two heuristic sets

The problems found by the evaluators differed according to which heuristic set they used. Problems found using Baker's heuristics reflected the emphasis placed on real-world collaboration,

including issues such as verbal and gestural communication, and this set appears to have constrained the range of issues that people considered. Eleven problems were found using the verbal communication heuristic, but they missed problems with the text chat entirely; instead, they focused on the need for voice chat support: “Cannot engage in voice chat. This would be good for this game so you can easily play and talk simultaneously.” People also described eight problems related to the gestural communication heuristic, although the described problems related to the appearance of players’ embodiments. Participants described eight problems related to the coordination heuristics, for example, “no way to make an alternate strategy or reorganize efforts.” Problems were found for all other heuristics in the set, but these problems were less frequent (five or fewer).

Problems were found using nine of our ten heuristics (see Table 5). No problems were found for heuristic 9 (interaction timeframe), but this heuristic was not very relevant to the real-time games included in the study. As with Baker’s heuristics, evaluators found a large number of problems related to communication, awareness, and coordination. In some of the areas of overlap between the two heuristic sets, the focus of the problem descriptions was different. Additionally, some substantive problems that were found using our heuristics were not found using Baker’s (training, matchmaking, unsavory behavior, session management). There were some problems found using Baker’s heuristics that were not found using ours (a total of 8); however, we inspected each of these and it was clear that the problems could have been found with NGH. For example, a participant found a problem related to mini-map design using Baker’s consequential communication heuristic: “The direction a tank is facing cannot be determined on the mini map.” Coverage for this problem is provided in our set by the awareness heuristic (H5), which explicitly addresses mini-map design. We believe that these differences were due to normal variation among evaluators.

**Table 5. Summary of results for NGH. Shows heuristic, total problems, and mean severity.**

Heuristic	Total	Mean severity
1. Session management	7	2.14
2. Matchmaking	8	2.50
3. Communication	8	2.38
4. Coordination	12	2.18
5. Awareness	8	2.38
6. Embodiments	5	2.00
7. Training	10	2.40
8. Social opportunities	4	1.67
9. Interaction timeframe	0	-
10. Cheating, unsavory behavior	6	2.20

Evaluators found twelve problems using our coordination heuristic (heuristic 4), and the problems are similar to those found when Baker’s heuristics were used. People also found ten problems related to training for novices (H7); for example: “Since games seem to be open to anybody it is hard to keep out expert players.” There were eight problems related to awareness issues (H5), and there was significant overlap with those found with Baker’s heuristics. Participants identified eight matchmaking problems (H2), for example: “no way to search for a particular player or a particular expertise level.” People also found eight

communication problems (H3), and the problem descriptions covered a broader range of issues than those seen in Baker’s heuristics. For example, they dealt with both chat problems and verbal communication issues: “It is hard to differentiate between status messages and chat” and “Since the game is quite fast, voice chat would be awesome. I died on a few occasions when texting.”

### 5.2.3 Subjective differences: benefits, disadvantages

After each session, we gave participants questionnaires and asked them to describe the benefits and disadvantages of using each heuristic set. For Baker’s heuristics, three people indicated that there were no advantages. The other participants indicated that they helped to focus the evaluation on collaboration issues, with one person writing that they “emphasized coordination and communication which this game lacked” and another stating that they “focus on communication and interaction among players.” All people listed disadvantages, and focused on difficulties that they encountered when mapping the heuristics to the games: “too task oriented whereas the game is very realtime/action oriented”, “the groupware issues only loosely relate to gaming”, and “did not really fit the domain.” We also asked whether people were able to find problems using the heuristics that they would have missed otherwise. Only three people agreed, and indicated that they found problems related to support for informal encounters, and communication support.

Participants listed two main benefits for using the NGH set. First, six people indicated that the heuristics provide a clear path to follow during the evaluation: “lets you focus on the important issues. Provides guidance.” Second, five people wrote that the heuristics help to focus the evaluation on issues that are important in games: “Clearly targeted at game specific issues. Surprisingly effective at focusing the evaluation on salient issues.” Six people listed disadvantages, and these fell into two main categories. First, four people wrote that all heuristics may not apply to all game: “a few heuristics are game specific and would not apply to all multiplayer games.” Second, two people wrote that using heuristics constrains the problems that they can find in games, and that they could miss problems not covered by the heuristics. Eight people indicated that they found problems with the heuristics that they would have missed otherwise, and these cover a wide range of issues including communication, cheating, cooperative strategies, session-finding, and matchmaking.

### 5.2.4 Subjective preferences

At the end of the study, we gave participants a final questionnaire and asked them to compare the effectiveness and ease of use of the heuristic sets. Most people (9/10) found that our heuristics were more effective at discovering problems than Baker’s; one person felt that there was no difference. Most people stated that NGH provided better coverage of multiplayer usability problems. For example, one person wrote: “NGH totally trumped Baker’s. NGH specifically identified multiplayer game-sensitive problems and Baker’s didn’t come close to identifying them.” Another participant wrote: “NGH by a wide margin because it does look at specific things like session management, training, cheating, and things that you wouldn’t have with more general heuristics.”

Most people also felt that our heuristics were easier to use than Baker’s (9/10), again with one person indicating no difference. People indicated that they found that Baker’s heuristics to be abstract and difficult to map to the game interface: e.g., “NGH

was easier and simpler because it was more specific and more easily mapped to the game elements. Baker's was a little abstract" Another wrote: "NGH was better adapted to the task that I'm judging. With Baker's, it was just too hard to try to figure out how to apply a heuristic in this particular context."

Finally, we asked participants which heuristic set they preferred for evaluating multiplayer games. Almost all of them (9/10) preferred NGH, but one person indicated that they found both sets to be useful.

## 6. DISCUSSION

### 6.1 Summary of Results

The goal of the evaluation was to determine whether people could operationalize the heuristics and find real usability problems when using them to evaluate networked games. We also wanted to assess whether the heuristics provided adequate coverage of the problem space seen in networked video games. Further, we wanted to determine whether the heuristics improved on current evaluation approaches.

The evaluation shows that the new heuristics were effective at specializing the evaluation process for multiplayer networked games. Almost all participants found that they were a good match for evaluating the games used in the study, and they preferred using NGH instead of Baker's heuristics when carrying out game evaluations. People indicated that it covered game specific problems that were not addressed by Baker's heuristics, and that they were easier to apply because the mappings between features in the game interface were easier to establish.

People found more problems using our heuristics than they did using Baker's heuristics. The area of coverage seen in problems found with Baker's set shows some of the limitations of applying generic heuristics to games. The heuristics emphasize verbal communication, which largely led people to overlook problems with chat interfaces that were found using our heuristics. Further, there were many areas of coverage found using our heuristics that were not found using Baker's, including training, matchmaking, unsavory behavior, and session management.

### 6.2 Scope and Generalizability

Our heuristics were developed from problem descriptions found in reviews of commercial video games. We considered a large number of games, from several major genres, and we believe that the heuristics will generalize to other similar games. However, it is possible that they will not generalize to games that are significantly different than the ones covered in the review set. For example, some of the heuristics may not be applicable to simple online games where user interaction is very limited, such as online card games and other Web-based games.

We based the heuristic set on PC games rather than console games; however, we believe that the heuristics will transfer to most networked console games since many commercial games are released concurrently for both platforms. We do not believe that the heuristics will transfer well to single-display games where several players use the same console. Single display games do not need the same level of interaction support that is seen in networked games since people can communicate face to face. Therefore, while some of our heuristics may be relevant, it is possible that a different set is needed for single-display settings.

Our heuristics focus exclusively on the multiplayer usability issues seen in networked games. They do not address single player usability, and need to be paired with single user heuristics, such as those described by Pinelle et al. [35], to provide full coverage. Further, the heuristics do not address other aspects of design that are important to the success of games, such as the challenge level and entertainment value. Other heuristics, such as those described by Federoff [11] and Desurvire et al. [7], address these issues, and they could potentially be used in conjunction with our set.

NGH, like all heuristic sets, must be viewed as guidelines rather than rules; and it is possible that designers will intentionally violate various heuristics in support of different types of game play. For example, in a spy game designers may want friends and enemies to be difficult to distinguish, which would violate our heuristic that emphasizes the need for identifiable avatars. The point of the heuristics is to provide designers with a set of principles that they can use to clearly decide whether a challenge or difficulty in the game is an intentional part of the gameplay, or whether it is an unnecessary usability problem.

It is also likely that expert evaluators can identify many multiplayer usability problems without the need for a heuristic set. However, using the heuristics provides a structure to help them consider each major design dimension in turn, and to prevent them from becoming distracted by other aspects of the design, such as single user or entertainment issues, which could cause them to miss important multiplayer problems. We also note that the heuristic set was compiled from problems that were found in commercial games that were released to the market, meaning that many of the issues covered in the set were not addressed by designers prior to the release.

Our heuristics are the first general set that covers general multiplayer problems found in networked games. We believe the reviews we used provide thorough coverage of major usability problems seen in networked games. However, the reviews were not written by usability professionals, and did not focus on usability issues as the main evaluation criteria. Therefore, it is possible that our heuristic set missed subtle usability problems, or problems that occur in early formative stages. We expect the heuristics to continue to grow and evolve in the future as we gain more experience and insights from using them to evaluate more games.

## 7. CONCLUSION

There are few methods for evaluating the usability of multiplayer networked games. We developed a new set of heuristics (NGH) to address multiplayer issues during network game design; the heuristics can be used to evaluate both early mockups and functional prototypes. Our heuristics are the first set to focus exclusively on usability problems found in multiplayer features of network games. The heuristics are based on problem reports from a large number of games and provide broad coverage of major game genres. Evaluation results show that evaluators are able to use the heuristics after a brief training session, that they are able to find real usability problems related to multiplayer features, that they provide broader coverage of problems seen in games than Baker's groupware heuristics [5], and that people preferred them over Baker's heuristics. The new heuristics contribute a valuable new tool for assessing and improving multiplayer usability. In

addition, they also show the value of domain-specific heuristic sets, and show the potential for using reviews as a source of usability data.

## 8. REFERENCES

- [1] Abner, W. Review of Heroes of Might and Magic V, [pc.gamespy.com/pc/heroes-of-might-magic-5/710168p3.html](http://pc.gamespy.com/pc/heroes-of-might-magic-5/710168p3.html), 2006.
- [2] Accardo, S. Review of Medal of Honor: Airborne, [pc.gamespy.com/pc/medal-of-honor-airborne/820348p1.html](http://pc.gamespy.com/pc/medal-of-honor-airborne/820348p1.html), 2007.
- [3] Alexander, T. *Massively Multiplayer Game Development*. Charles River Media, 2003.
- [4] Annett, J. and Duncan, K.D. Task analysis and training design. *Occupational Psychology* 12, 1967, 211-221.
- [5] Baker, K., Greenberg, S., Gutwin, C. Empirical development of a heuristic evaluation methodology for shared workspace groupware. *Proceedings CSCW 2002*, 96-105.
- [6] Cornett, S. The usability of massively multiplayer online roleplaying games: designing for new users. *Proceedings CHI 2004*. 703-710.
- [7] Desurvire, H., Caplan, M., Toth, J. A. Using heuristics to evaluate the playability of games. *Proc. CHI 2004*, 1509-12.
- [8] Diot, C., Gautie, L. A Distributed Architecture for Multiplayer Interactive Applications on the Internet. *IEEE Network*, 13(4), 6-15, 1999.
- [9] Ducheneaut, N., Moore, R. J., & Nickell, E. Designing for sociability in Massively Multiplayer Games: An examination of the "third places" of SWG. *Proc. of the Other Players Conference*, 2004.
- [10] Dykstra, D.J. *A Comparison of Heuristic Evaluation and Usability Testing: The Efficacy of a Domain-Specific Heuristic Checklist*. Ph.D. dissertation, Department of Industrial Engineering, Texas A&M University, 1993.
- [11] Federoff, M. *Heuristics and Usability Guidelines for the Creation and Evaluation of Fun in Video Games*. Indiana University Master of Science Thesis, 2002.
- [12] Fullerton, T., Swain, C., Hoffman, S. *Game Design Workshop: Designing, Prototyping, and Playtesting Games*, CMP Books, 2004.
- [13] Gamespot staff. Review of Tom Clancy's Rainbow Six Vegas, [www.gamespot.com/pc/action/rainbowsixvegas/review.html?sid=6163307](http://www.gamespot.com/pc/action/rainbowsixvegas/review.html?sid=6163307), 2006.
- [14] GameSpy staff. Review of Dungeons & Dragons Online: Stormreach, [pc.gamespy.com/pc/dungeons-dragons-online/696856p4.html](http://pc.gamespy.com/pc/dungeons-dragons-online/696856p4.html), 2006.
- [15] Gutwin, C., Greenberg, S. The mechanics of collaboration: developing low cost usability evaluation methods for shared workspaces. *Proceedings WETICE 2000*.
- [16] Gutwin, C. and Greenberg, S. The effects of workspace awareness support on the usability of real-time distributed groupware. *TOCHI*, 6(3), 1999, 243-281.
- [17] Habgood, J., Overmars, M. *Game Maker's Apprentice: Game Development for Beginners*. Apress, 2006.
- [18] Harker, C. Review of RYL: Path of the Emperor, [pc.gamespy.com/pc/risk-your-life-2-path-of-the-emperor/635953p1.html](http://pc.gamespy.com/pc/risk-your-life-2-path-of-the-emperor/635953p1.html), 2005.
- [19] Harms, W. Review of The Lord of the Rings: The Battle for Middle-earth II, [pc.gamespy.com/pc/the-lord-of-the-rings-the-battle-for-middle-earth-2/693176p3](http://pc.gamespy.com/pc/the-lord-of-the-rings-the-battle-for-middle-earth-2/693176p3), 2006.
- [20] Harms, W. Review of Rainbow Six: Lockdown, [pc.gamespy.com/pc/rainbow-six-4-tentative-title/690399p2.html](http://pc.gamespy.com/pc/rainbow-six-4-tentative-title/690399p2.html), 2006.
- [21] Koivisto, E.M. Supporting Communities in Massively Multiplayer Online Role-Playing Games by Game Design. *Proc. Level Up*, 2003.
- [22] Korhonen, H. Koivisto, E. M. Playability heuristics for mobile multi-player games. *Proc. DIMEA 2007*, 28-35.
- [23] Korhonen, H. and Koivisto, E. M. Playability heuristics for mobile games. *Proc. Mobile HCI 2006*, 9-16.
- [24] Kosak, D. Review of Crysis, [pc.gamespy.com/pc/ea-crytek-title-untitled-project/839785p1.html](http://pc.gamespy.com/pc/ea-crytek-title-untitled-project/839785p1.html), 2007.
- [25] Kosak, D. Review of Age of Pirates: Caribbean Tales, [pc.gamespy.com/pc/age-of-pirates/737252p2](http://pc.gamespy.com/pc/age-of-pirates/737252p2), 2006.
- [26] Kosak, D. and Lopez, M. Review of Guild Wars, [pc.gamespy.com/pc/guild-wars/613203p1.html](http://pc.gamespy.com/pc/guild-wars/613203p1.html), 2005.
- [27] Kuo, L.C. Review of Trackmania United, [pc.gamespy.com/pc/trackmania-united/799496p1.html](http://pc.gamespy.com/pc/trackmania-united/799496p1.html), 2007.
- [28] Kuo, L.C. Review of GTR 2, [pc.gamespy.com/pc/gtr-2/751848p1.html](http://pc.gamespy.com/pc/gtr-2/751848p1.html), 2006.
- [29] Kuo, L.C. Review of Tom Clancy's Splinter Cell Double Agent, [pc.gamespy.com/pc/tom-clancys-splinter-cell-4/751217p1.html](http://pc.gamespy.com/pc/tom-clancys-splinter-cell-4/751217p1.html), 2006.
- [30] Madigan, J. Review of Cold War Conflicts: Days in the Field 1950-1973, <http://pc.gamespy.com/pc/cold-war-conflict/626650p1.html>, 2005.
- [31] Nielsen, J., and Mack, R.L. (Eds.), *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.
- [32] Nielsen, J. How to Conduct a Heuristic Evaluation. [www.useit.com/papers/heuristic/](http://www.useit.com/papers/heuristic/), 2007.
- [33] Nielsen, J. Severity Ratings for Usability Problems. [www.useit.com/papers/heuristic/severityrating](http://www.useit.com/papers/heuristic/severityrating), 2007.
- [34] Osborne, S. Review of TrackMania Sunrise, [pc.gamespy.com/pc/trackmania-sunrise/620071p1](http://pc.gamespy.com/pc/trackmania-sunrise/620071p1), 2005.
- [35] Pinelle, D., Wong, N, Stach, T. Heuristic Evaluation for Games: Usability Principles for Video Game Design. *Proceedings CHI 2008*, 1453-1462.
- [36] Pinelle, D. and Gutwin, C. Groupware Walkthrough: Adding Context to Groupware Usability Evaluation. *Proceedings CHI 2002*, 455- 462.
- [37] Pinelle, D., Gutwin, C., Greenberg, S. Task Analysis for Groupware Usability Evaluation: Modeling Shared-Workspace Tasks with the Mechanics of Collaboration. *TOCHI*, 10(4), 2003, 281-311.
- [38] Rausch, A. Review of Hellgate London, [pc.gamespy.com/pc/hellgate-london/832252p3.html](http://pc.gamespy.com/pc/hellgate-london/832252p3.html), 2007.
- [39] Smed, J., Kaukoranta, T., and Hakonen, H. Aspects of Networking in Multiplayer Computer Games. *The Electronic Library*, 20(2):87-97, 2002.