

# Overcoming the Challenges of Game Networking with Quazal

## Overview

Computer gaming has evolved from its roots as a solitary pursuit to become a leading form of online social interaction. No longer are gamers content with playing an imaginary opponent, they now seek to challenge their skills against other players in vast online worlds, potentially involving thousands of opponents. The rise of online gaming has been predicated on technological advances ranging from the widespread availability of powerful PCs with Intel® architecture processors to the rise of the Internet. However, the increasing popularity of online gaming also brings with it greater challenges for developers as they push the technological limits while trying to satisfy gamers' escalating demands.

One of the greatest challenges developers face is how to simultaneously connect thousands of players to one virtual world given the Internet's inherent unreliability, high latencies, bandwidth limitations, and resource constraints. Faced with these daunting technical challenges, game developers are increasingly turning to specialized providers of gaming technology for off-the-shelf solutions to the unique infrastructure problems of multiplayer (MP) and massively multiplayer (MMP) games, which are all neatly addressed by the Quazal Net-Z\* 3.1 and Eterna\* products (respectively).

## The Challenges of Multiplayer Connectivity

It seems easy at first. Just have one player send messages to another. However, as the number of players increases this relatively simple task can take on nightmarish proportions; the network is flooded with messages and quality of play begins to suffer in ways that are hard to correct. Say you have your gun sight locked on your opponent, you fire, you see the ammo hit him cleanly, however, to your surprise his character doesn't fall. Game-play may have been affected by a high-latency Internet connection saturated with messages attempting to keep all the players aware of each others activities.

At the heart of online game-play is a communications layer that can either make or break the user experience. Developing a game is hard enough without worrying about the issues of latency, resource management, security, and scalability. The nature of the problem is clearly expressed by Mike Drummelsmith, Developer Relations Manager at Quazal, "Most online MP and MMP game implementations are initially flawed because of issues with the application's communications layer. Developers are constantly reinventing the wheel; they are writing code at the socket level to deal directly with the TCP/IP and UDP protocols. Quazal provides a high-level solution that abstracts away network and messaging issues so developers can focus on doing what they do best: building an addictive game."

Another challenge game developers encounter is the need for code portability. Although offline MP gaming traces its roots back to the dedicated game console, it has now evolved to its current online form and occupies a central role in the expanding universe of PC-based entertainment.



All this means that developers must support the online gaming experience on platforms ranging from the Intel processor powered Microsoft Xbox\* game console to traditional desktop PCs (and even Pocket PCs enabled with 802.11b wireless connections). The need to support these diverse platforms provides yet another reason for game developers to consider the Quazal Net-Z and Eterna products.

### **The Quazal Solution**

---

In the world of gaming, everything is an object. Some objects represent players (commonly called characters) which hold their current location and status. Others are used to hold the state of non-player characters (often referred to as NPCs) such as a monster controlled by the game's artificial intelligence engine. In the typical online MP game, individual characters are hosted on the machine of their respective owner and non-player characters are hosted on the machine of the player who started the online gaming session. This all sounds simple enough until one realizes that this architecture has numerous pitfalls that present serious technical problems for the game developer.

In the traditional client/server game architecture, player objects communicate with each other by passing state change messages through the machine of the player who started the gaming session (called the game server). Non-player character objects must broadcast similar messages to all players participating in the game. In addition to stressing communications bandwidth, this model lacks fault tolerance and the ability to effectively balance consumption of machine resources. Although many machines are participating in game-play, not all are contributing evenly to the scalability of the game, the game server is unfairly burdened with controlling all the non-player characters. And, if the server goes offline for any reason, the game is effectively over as all messages will cease to be routed.

### **Safety in Numbers**

---

Quazal changes the rules of building online MP and MMP games with its Duplicated Objects\* technology.

In the past a developer had to worry about the mechanics of keeping all players up to date, now Quazal automates this process and provides an easy-to-use API that keeps the focus on the object, not the messaging layer. Duplicated Objects also takes full advantage of distributed computing by allowing the game to effectively use every player's Internet bandwidth and processor resources for the greater good.

A Duplicated Object can be either a duplication master, which is the controlling instance of the object, or a duplica, which is a complete copy of the master object. When a Duplicated Object is created, its duplication master is created on a player's machine. The duplication master then publishes itself on all other machines that require the object. If a player joins a session that is already in progress, duplicas of the existing Duplicated Objects that the new player requires will be automatically created on that player's machine. To maintain a coherent object state between the duplication master and its duplicas, the duplication master (controlling) object sends updates to all of its duplicas. Quazal also provides performance optimizations such as dead reckoning and data compression to minimize the volume and size of messaging traffic between Duplicated Objects.

With Duplicated Objects the game can scale to support many more players than is possible with a traditional architecture. Say you've decided to host an online gaming session with 50 of your closet friends. You've yet to upgrade to the latest Intel processor and are still running an older 1.0GHz Intel Pentium® III machine. Luckily, some of your gamer friends have recently revamped their setups and are now running the latest 3.2GHz Pentium 4 processor with 512KB caches. Duplicated Objects can take advantage of relative disparities in processing power by migrating duplication master objects, and the responsibility to execute their artificial intelligence routines, to faster machines.

Another plus of using Duplicated Objects is support for built-in fault tolerance. As each player's machine contains a complete set of duplica objects, it is a simple matter for the game to detect that a duplication master object is no longer available and designate one of the duplica objects as the new master (Eterna supports subsets of duplicas for better scaling of MMP games with even larger numbers of objects).

With this approach the failure of any player's machine results in the duplication master objects transferring 'ownership' to another station, causing an almost imperceptible hiccup in the game, as opposed to a critical failure.

The ability to migrate master objects between player machines provides a clever way to combat the problem of high latency connections. In a traditional architecture, non-player objects always reside on the machine of the player who started the gaming session-no matter what the latency between the game server and a player. Say you're battling a monster in a Quake type game. Unfortunately your experience is compromised by the high latency connection between your machine and the game server-in other words; you get killed because you couldn't react in time. Since you're the one fighting this monster it makes sense for that object to reside on your machine. With Duplicated Objects this migration takes place, allowing the object to become local, completely eliminating the latency and you survive the battle because of faster response times.

### ***Faster Messaging***

---

Even though Net-Z and Eterna take great pains to reduce the volume and size of messages between player machines, there are still many messages to send. Much of the work involved in sending these messages can be classified as pure administrative overhead-the compression, encryption, bundling and transport of messages between player machines is a necessary evil that cannot be avoided. It can, however, be made more efficient. To reduce the impact of the message transport layer Quazal turned to Intel Hyper-Threading Technology. This technology enables threaded applications to see improved performance due to additional concurrency and better utilization of on-chip execution resources-each physical processor is viewed as two logical processors each with its own state.

To maximize the benefit of Intel Hyper-Threading Technology, the Quazal development team implemented a multi-threaded approach to the transport layer where the compression, encryption, bundling, and transmission of messages now occurs with higher levels of concurrency.

According to Drummelsmith, the threaded transport layer is 50-100% faster than the original non-threaded code. Threading support can be turned off for backwards compatibility with some older game developer toolkits.

### ***Summary***

---

Today's gamers demand a lot from a game. Not only must it be more fun to play and look better than the last game they bought, but it must also perform well and be reliable over a wide range of network conditions. Quazal's innovative Net-Z and Eterna products help meet these demands by providing a robust multiplayer connectivity platform that allows developers to focus on creating groundbreaking online games instead of worrying about troublesome networking and messaging issues.

<http://www.quazal.com>

# QUAZAL

MULTIPLAYER CONNECTIVITY

## Pointers from the Pros

---

### ■ **Latency is a killer:**

The effect of latency on the gaming experience cannot be understated—even on a broadband connection. Developers must construct games that are appealing on networks with latencies that average 50 to 120ms. One common option for dealing with latency is hiding it. Net-Z and Eterna make this easy by means of a configurable loopback control that builds a fixed latency into the game. The more desirable solution is architecting the game to reduce latencies across the board—a goal that Duplicated Objects\* makes possible through object migration.

### ■ **Thwart the cheaters:**

It's no fun playing a game if there are cheaters abound. Security has become a huge issue as online game developers strive to create a level playing field. Even more importantly, corporate sponsored online tournaments are now coming with real-world cash prizes. Games constructed using Net-Z and Eterna are particularly amenable to post action auditing and administrators have access to the full message stream and object states so it is easier to find cheaters.

### ■ **Resource management is critical:**

Network bandwidth, system memory, and processor cycles are finite resources. Use them wisely by minimizing network traffic and supporting load distribution across player machines. Pay close attention to the design of your game objects. Do you really need an 8-byte double to represent the color of a player's body armor? Quazal also helps to reduce network traffic with support for data compression. Take advantage of the built-in load balancing that Duplicated Objects offers.

### ■ **Design for scalability:**

Building a game for 16 players is a very different experience than building one for 16,000. Developers that attempt to address scalability requirements as an afterthought usually have a disastrous experience. Leveraging the work of others in building a scalable gaming infrastructure can save years of effort.

## About the Author

---

Allan Mc Naughton is the principal analyst at Triangle Technology Advisors LLC (<http://www.triangletechadvisors.com>), a firm specializing in the composition of high-technology white papers. He can be reached at [allan@triangletechadvisors.com](mailto:allan@triangletechadvisors.com).

## Additional Resources

---

Intel® Developer Services provides online resources to help software developers accelerate the development and delivery of their applications, tools and solutions.

Find out more at the Intel Developer Services website at: <http://www.intel.com/ids>



Hyper-Threading Technology requires a computer system with an Intel® Pentium® 4 processor at 3.06 GHz or higher, a chipset and BIOS that utilize this technology, and an operating system that includes optimizations for this technology. Performance will vary depending on the specific hardware and software you use. See [www.intel.com/info/hyperthreading](http://www.intel.com/info/hyperthreading) for information.

Information in this document is provided in connection with Intel® products. Except as provided in Intel's terms and conditions of sale for such products, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY RELATING TO SALE AND/OR USE OF INTEL PRODUCTS, INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT, OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life-saving, life-sustaining, critical control or safety systems, or in nuclear facility applications. Intel may make changes to specifications, product descriptions, and plans at any time, without notice.

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, reference [www.intel.com/procs/perf/limits.htm](http://www.intel.com/procs/perf/limits.htm), or call (U.S.) 1-800-628-8686 or 1-916-356-3104.

Intel, the Intel logo, Pentium and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

\*Other names and brands may be claimed as the property of others.