UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

CARLOS EDUARDO BENEVIDES BEZERRA

<PEGAR DO PLANODOC NO NOTE>

Dissertação apresentada como requisito parcial
para a obtenção do grau de
Mestre em Ciência da Computação

Prof. Dr. Cláudio Fernando Resin Geyer
Orientador

Porto Alegre, janeiro de 2009

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Bezerra, Carlos Eduardo Benevides

<PEGAR DO PLANODOC NO NOTE> / Carlos Eduardo Benevides Bezerra. – Porto Alegre: PPGC da UFRGS, 2009.

16 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2009. Orientador: Cláudio Fernando Resin Geyer.

1. Jogos online maciçamente multijogador. 2. MMOG. 3. Simulação interativa distribuída. I. Geyer, Cláudio Fernando Resin. II. Título.

AGRADECIMENTOS

<TO-DO>

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

ADI	*Área de Interesse*
API	*Application Programming Interface*
CVE	*Collaborative Virtual Environment*
CPU	*Central Processing Unit*
DHT	*Distributed Hash Table*
DIS	*Distributed Interactive Simulation*
FPS	*First-Person Shooter*
IP	*Internet Protocol*
J2SE	*Java 2 Standard Edition*
MMG	*Massively Multiplayer Game*
MMOFPS	*Massively Multiplayer Online First-Person Shooter*
MMORPG	*Massively Multiplayer Online Role-Playing Game*
MMORTS	*Massively Multiplayer Online Real-Time Strategy*
NVE	*Networked Virtual Environment*
RTP	*Real-time Transport Protocol*
RTS	*Real-Time Strategy*
TCP	*Transmission Control Protocol*
TSS	*Trailing State Synchronization*
UDP	*Unreliable Datagram Protocol*
URL	*Uniform Resource Locator*

LISTA DE FIGURAS

RESUMO

<lalalalalalalalá ... fazer resumo lalalalalalá....>

A popularização das tecnologias de acesso à Internet por "banda larga" suportam a crescente proliferação de aplicações do tipo "par-a-par" (peer-to-peer), onde o usuário doméstico, tipicamente consumidor de informação, passa também a atuar como provedor. De forma simultânea, há uma popularização crescente dos jogos em rede, especialmente dos "jogos maciçamente multijogador" (MMG ou *massively multiplayer game*) onde milhares de jogadores interagem, em tempo real, em mundos virtuais de estado persistente. Os MMGs disponíveis atualmente, como *EverQuest* e *Ultima Online*, são implementados como sistemas centralizados, que realizam toda a simulação do jogo no "lado servidor". Este modelo propicia controle de acesso ao jogo pelo servidor, além de ser muito resistente a jogadores trapaceiros. Porém, a abordagem cliente-servidor não é suficientemente escalável, especialmente para pequenas empresas ou projetos de pesquisa que não podem pagar os altos custos de processamento e comunicação dos servidores de MMGs centralizados. Este trabalho propõe o FreeMMG, um modelo híbrido, cliente-servidor e par-a-par, de suporte a jogos maciçamente multijogador de estratégia em tempo real (MMORTS ou *massively multiplayer online real-time strategy*). O servidor FreeMMG é escalável pois delega a maior parte da tarefa de simulação do jogo para uma rede par-a-par, formada pelos clientes. É demonstrado que o FreeMMG é resistente a certos tipos de trapaças, pois cada segmento da simulação distribuída é replicado em vários clientes. Como protótipo do modelo, foi implementado o jogo *FreeMMG Wizards*, que foi utilizado para gerar testes de escalabilidade com até 300 clientes simulados e conectados simultaneamente no mesmo servidor. Os resultados de escalabilidade obtidos são promissores, pois mostram que o tráfego gerado em uma rede FreeMMG, entre servidor e clientes, é significativamente menor se comparado com uma alternativa puramente cliente-servidor, especialmente se for considerado o suporte a jogos maciçamente multijogador de estratégia em tempo real.

Palavras-chave: Jogos online maciçamente multijogador, MMOG, simulação interativa distribuída.

FreeMMG: A Client-Server and Peer-to-Peer Support Architecture for Massively Distributed Games

RESUMO

The increasing adoption of "broadband" Internet access technology fosters the increasing development of networked "peer-to-peer" applications, where the end-user, usually the consumer of information, begins to act also as a producer of information. Simultaneously, there is a growing popularity of networked games, especially "massively multiplayer games" (MMGs) where thousands of players interact, in real time, in persistent-state virtual worlds. State-of-the-art massively multiplayer games such as EverQuest and Ultima Online are currently implemented as centralized, client-server systems, which run the entire game simulation on the "server-side". Although this approach allows the development of commercially viable MMG services, with game access control and player cheating prevention, the processing and communications costs associated with running a MMG server are often too high for small companies or research projects. This dissertation proposes FreeMMG, a mixed peer-to-peer and client-server approach to the distribution aspect of MMGs. It is argued that the FreeMMG model supports scalable, cheat-resistant, massively multiplayer real-time strategy games (MMORTS games) using a lightweight server that delegates the bulk of the game simulation to a peer-to-peer network of game clients. It is shown that FreeMMG is resistant to certain kinds of cheating attempts because each segment of the distributed simulation is replicated by several clients. A working prototype game called FreeMMG Wizards is presented, together with scalability test results featuring up to 300 simulated game clients connected to a FreeMMG server. The obtained results are promising, by showing that the generated server traffic in a FreeMMG network, between the server and the clients, is substantially lower if compared with purely client-server alternatives, especially if the support for massively multiplayer real-time strategy games is considered.

Palavras-chave: massively multiplayer games, distributed interactive simulation, peer-to-peer systems, real-time strategy games.

1 INTRODUÇÃO

2 CONTEXTO, ESTADO DA ARTE E MOTIVAÇÃO

3 MODELO BASE

4 A^3: UM ALGORITMO DE OTIMIZAÇÃO POR ÁREA DE INTERESSE

[TODO:cap/lista de termos]

5 BALANCEAMENTO DE CARGA

A principal característica dos jogos maciçamente multijogador é a grande quantidade de jogadores, chegando a ter dezenas ou centenas de milhares de participantes simultaneamente [TODO:ref]. Essa grande quantidade de jogadores interagindo entre si gera um tráfego na rede de suporte que tem crescimento quadrático em relação ao número de jogadores, no pior caso [TODO:referenciar/provar].

Quando se utiliza uma arquitetura cliente-servidor, é necessário que o servidor intermedie a comunicação entre cada par de jogadores – supondo que se pretende prover ao jogo garantias de consistência e resistência a trapaça. Obviamente, esse servidor terá uma grande carga de comunicação e, conseqüentemente, deverá ter recursos (largura de banda disponível) proporcional à demanda do jogo.

[TODO:largura de banda X CPU]

A questão posta aqui é que, quando se faz uso de um servidor distribuído, principalmente com recursos escassos, que é a proposta deste trabalho, é necessário otimizar o uso destes recursos, atribuindo a cada servidor uma carga que ele seja capaz de suportar. Dessa forma, não importando a qual servidor cada jogador estiver conectado, sua experiência de jogo será semelhante, no que diz respeito ao tempo de resposta para suas ações e o tempo que leva para ser notificado de ações de outros jogadores, assim como de mudanças de estado no ambiente virtual do jogo.

[aqui já fala do n ao quadrado, por cima]

Uma idéia inicial poderia ser a de distribuir os jogadores entre servidores, de maneira que o número de jogadores em cada servidor fosse proporcional à largura de banda daquele servidor. No entanto, essa distribuição não funcionaria, pelo fato de que a carga causada pelos jogadores depende também do quanto os jogadores estão interagindo entre si. Por exemplo, se os avatares de dois jogadores estiverem muito distantes um do outro, provavelmente não haverá interação entre eles e, portanto, o servidor precisará apenas atualizar cada um a respeito de suas próprias ações. No entanto, se estes avatares estiverem próximos, cada jogador deverá ser atualizado não apenas a respeito de suas próprias ações, como também das ações do outro jogador.

Percebe-se, então, que quando os avatares estão distantes uns dos outros, o tráfego cresce linearmente com o número de jogadores. Porém, se eles estão próximos uns dos outros, o tráfego cresce quadraticamente. Por fim, ambas as funções de crescimento do número de mensagens podem estar presentes no mesmo jogo se, em alguns lugares do ambiente virtual, os avatares estiverem próximos e, em outros lugares, eles estiverem distantes.

A grande maioria dos jogos multijogador apresenta a característica de localidade, no que diz respeito à distribuição dos jogadores no ambiente virtual. Existem alguma exceções, como GuildWars [TODO:ref], em que apenas grupos com um número limitado

de jogadores podem iniciar uma partida. Este tipo de jogo é baseado no modelo de instâncias, onde todos os avatares dos jogadores se encontram em um espaço social, de interação limitada, menos dinâmica e, portanto, com tráfego de rede algumas ordens de grandeza menor [TODO:ref/prove]. Quando pretende-se iniciar uma partida "‘real"’, os jogadores requisitam ao servidor que seja criado um grupo de ação. Dessa forma, impede-se que um número teoricamente ilimitado de jogadores interajam entre si, sobrecarregando o servidor.

Normalmente, porém, os jogadores podem mover seus avatares livremente através do mundo do jogo. Isso torna possível a formação de pontos de interesse – também conhecidos como *hotspots* [TODO:ref] – ao redor dos quais os jogadores se concentram mais do que em outras regiões do ambiente virtual. Aliás, muitos jogos de RPG online maciçamente multijogador não só permitem como também estimulam, até certo ponto, a formação destes pontos de interesse. Nestes mundos dos MMORPGs, existem cidades inteiras, onde os jogadores se encontram para conversar, trocar mercadorias virtuais do jogo e/ou duelar, assim como existem também zonas desérticas, sem muitos atrativos para os jogadores, e onde o número de avatares presentes é relativamente pequeno, se comparado com os outros lugares no jogo.

[TODO:screenshot(s)]

Por esta razão, não é suficiente simplesmente dividir os jogadores entre os servidores, mesmo que proporcionalmente aos recursos de cada um destes. Primeiro, em alguns lugares o consumo de largura de banda do servidor é quadrático ao número de jogadores, enquanto é linear em outros. Essa razão por si só já é suficiente para buscar outro critério para o balanceamento de carga. Além disso, surge outra questão importante: a existência de pontos de interesse. Esta última característica motiva à criação de um esquema de balanceamento de carga para jogos que impeça que a presença de hotspots degrade a qualidade do jogo além do tolerável.

Como foi dito, quando existe um número considerável de avatares em um mesmo ponto de interesse, é gerado um tráfego proporcional ao quadrado do número de avatares ali presentes. Também foi mostrado que os servidores recebem as ações enviadas pelos jogadores, calculam seu resultado e o enviam para todos os jogadores interessados, que são, geralmente, aqueles cujos avatares estiverem próximos do avatar do primeiro jogador. Se esses jogadores forem divididos entre diferentes servidores, cada um destes precisará não apenas enviar o estado do mundo resultante das ações para os jogadores controlados por ele, como também deverá enviá-lo para o servidor ao qual os outros jogadores estão conectados. Este, por sua vez, encaminhará este resultado para seus jogadores.

[por figura do envio de estados através de servidores diferentes. duas figuras: no mesmo servidor, e através de diferentes servidores.]

Percebe-se, então, que cada estado deverá ser enviado duas vezes, para cada par de jogadores que se comunicam através de servidores diferentes. Esse overhead não apenas causa o desperdício de recursos dos servidores, como também aumenta o atraso para atualização de estado das réplicas do jogo nas máquinas dos jogadores. Isto faz com que o tempo entre o envio de uma ação por um jogador conectado a um servidor e o recebimento do estado resultante por outro jogador, conectado a outro servidor, seja maior, prejudicando a interação entre eles.

Assim sendo, jogadores que estão interagindo entre si devem, idealmente, estar conectados ao mesmo servidor. Contudo, é possível que todos os jogadores estejam ligados entre si através de relações de interação. Por exemplo, dois avatares, de dois jogadores diferentes, podem estar distantes, porém ambos interagindo com um terceiro avatar, en-

tre os dois. Esse tipo de situação faz com que todos os jogadores estejam relacionados de alguma forma. Portanto, é necessário um critério para decidir quando dois jogadores estarão conectados ao mesmo servidor, e quando não estarão.

O balanceamento de carga entre servidores de jogos online maciçamente multijogador é fortemente dependente da distribuição dos avatares dos jogadores através do ambiente virtual. Além disso, a depender da concentração de avatares, pode-se alternar entre uma função de crescimento de tráfego linear e uma função quadrática. Sendo assim, é necessário lidar com a localidade dos avatares, de maneira a otimizar o uso de largura de banda do sistema servidor, minimizando, tanto quanto possível, o overhead causado pela comunicação entre jogadores ligados a servidores diferentes.

A distribuição ideal seria então aquela que agrupasse jogadores que estivessem próximos uns dos outros

Uma idéia proposta por [TODO:ref] foi a de

6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS