

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

CARLOS EDUARDO BENEVIDES BEZERRA

**Um modelo de suporte distribuído para
jogos MMOG em cenários com recursos
computacionais limitados
(Título Provisório)**

Plano de Curso para Doutorado

Prof. Dr. Cláudio Fernando Resin Geyer
Orientador

Porto Alegre, novembro de 2008

SUMÁRIO

RESUMO	3
1 INTRODUÇÃO	4
1.1 Jogos Multijogador	4
1.2 Jogos Online Maciçamente Multijogador (MMOGs)	5
2 MOTIVAÇÃO E ESTADO DA ARTE	6
2.1 Requisitos dos jogos MMOG	6
2.2 Trabalhos relacionados	7
2.3 Questões em aberto	9
3 OBJETIVO	11
3.1 Objetivo geral	11
3.2 Objetivos específicos	11
4 PLANO DE TRABALHO	12
4.1 Metodologia	12
4.2 Cronograma	12
4.3 Planejamento dos créditos	13
5 INFORMAÇÕES DO CANDIDATO	14
5.1 Dados de identificação	14
5.2 Experiência com pesquisa	14
REFERÊNCIAS	15
ASSINATURAS	17

RESUMO

Este plano de curso de doutorado propõe um modelo de suporte a jogos online maciçamente multijogador fazendo uso de sistemas de compartilhamento em larga escala de recursos computacionais, tais como redes P2P. Seguindo este modelo, sobre uma rede P2P seria formado um sistema que atuaria como servidor de um jogo MMOG. A cada nodo deste sistema seria designado uma porção da carga de comunicação e processamento do jogo – mais especificamente, seria designada a cada nodo uma região do ambiente virtual do jogo. Tal sistema seria percebido como um servidor central pelos jogadores, que conectar-se-iam a ele através de um *Gateway*, sendo cada um redirecionado para o nodo servidor adequado. Para que esse sistema servidor funcione como esperado, será necessário tratar algumas questões, tais como manter consistente o andamento do jogo nos diferentes nodos servidores, de forma que cada jogador tenha a mesma visão; gerenciamento da carga atribuída a cada servidor, sendo que maior parte dela recai sobre a rede e, por se tratar de um jogo, é bastante variável; o que fazer quando um nodo servidor sai da rede ou entra em colapso e, por fim, como difundir e armazenar os estados dos jogadores de maneira distribuída, de forma que sempre se possa recuperar o estado mais recente.

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

1 INTRODUÇÃO

Jogos online maciçamente multijogador – ou MMOGs, *massively multiplayer online games* – são o tema central deste plano de doutorado. Geralmente, utiliza-se uma infraestrutura cliente-servidor para dar suporte a este tipo de jogo. Neste plano será proposto um modelo de distribuição do lado servidor, de maneira a reduzir ou eliminar seu custo de manutenção. Nas próximas seções será apresentado o contexto referente a jogos multijogador e MMOGs.

1.1 Jogos Multijogador

Nas últimas décadas, desde o surgimento do computador pessoal, jogos de computador tem se popularizado de maneira crescente. Mais ainda, após o surgimento da Internet aliado ao fato de as conexões domésticas estarem cada vez mais baratas e mais rápidas, jogos online multijogador têm merecido atenção especial. Estes jogos se caracterizam por haver vários participantes interagindo uns com os outros, utilizando uma conexão através da Internet entre seus dispositivos computacionais, que são geralmente computadores pessoais ou consoles (dispositivos computacionais específicos para jogos). Este tipo de jogo pode ser dividido em vários gêneros, como por exemplo:

- FPS: *first-person shooting*, ou jogos de tiro com visão em primeira-pessoa, em que cada jogador controla um personagem que dispõe de diferentes armas, com as quais enfrenta monstros ou personagens de outros jogadores. São caracterizados por ação rápida, incluindo atirar, mover-se, esquivar-se etc.;
- RTS: *real-time strategy*, ou jogos de estratégia em tempo-real, onde cada jogador controla um exército, mas suas ações são executadas simultaneamente às ações dos outros jogadores, ao invés de serem separadas em turnos. A ação é mais lenta que em jogos FPS;
- RPG: *role-playing game*, ou jogo de interpretação de papéis, no qual cada jogador tem um personagem que evolui com o tempo, adquirindo mais poder e acumulando tesouros. Jogos deste tipo são caracterizados por não terem um início ou fim de partida definidos, constituindo um mundo virtual de estado persistente. Os jogadores então podem evoluir seus personagens, desconectarem-se e reconectarem-se mais tarde para continuarem jogando a partir do ponto onde pararam, pois seu estado estará armazenado.

Como exemplos de jogos multijogador online conhecidos, podem ser citados *Quake* (?), um jogo do gênero FPS lançado em meados da década de 90; *Starcraft: Brood War*

(BLIZZARD, 1998), da mesma época, porém consistindo de um jogo RTS, Counter-Strike (VALVE, 2000), FPS lançado em 2000 e Diablo II (?), RPG lançado no mesmo ano.

1.2 Jogos Online Maciçamente Multijogador (MMOGs)

Da mesma forma que jogos de computador evoluíram para jogos multijogador on-line após a popularização da Internet, evoluiu-se em seguida para jogos online maciçamente multijogador (ou MMOG, *massively multiplayer online games*), graças ao barateamento das conexões à Internet e o aumento da sua velocidade. Jogos deste tipo têm se popularizado bastante nos últimos tempos. Além de poderem ser jogados on-line, permitem a interação simultânea de um grande número de participantes. Nos casos de maior sucesso, como World of Warcraft (BLIZZARD, 2004) e Lineage II (NCSOFT, 2003), por exemplo, é dado suporte a uma base de dezenas a centenas de milhares de jogadores simultâneos (CHEN; MUNTZ, 2006). Estes jogadores podem, então, interagir entre si e com o ambiente virtual do jogo.

Cada jogador controla uma entidade chamada de *avatar*, que nada mais é que sua representação no ambiente virtual - um personagem que executa as ações determinadas pelo jogador, que dessa forma passa a interferir na história e no encaminhamento do jogo. Para que a partida se mantenha consistente para os diferentes participantes, cada ação executada por cada avatar deve ser processada pelo servidor, que calculará o estado resultante dessa ação no jogo. Esse novo estado é então difundido para os outros jogadores. Outras alterações no mundo do jogo, como destruição ou criação de objetos no ambiente, e eventos, como ações de personagens controlados pelo servidor, devem também ser transmitidos aos jogadores envolvidos. É então através deste envio de comandos do jogador e recebimento do novo estado do servidor que os jogadores conseguem interagir entre si e com o mundo simulado no jogo.

No entanto, percebe-se que tal mecanismo pode facilmente saturar a banda e sobrecarregar o(s) processador(es) do servidor. Geralmente o que se faz é montar uma infraestrutura que dispõe de um grande servidor central, geralmente um cluster com seus nodos ligados por uma rede local de alta velocidade e baixo atraso, com conexão à Internet de algumas centenas ou milhares de MBps. O maior peso reside na manutenção desta conexão, que visa a enviar para cada jogador o estado mais recente do jogo sempre que este mudar, dentro de um limite estabelecido de atraso, que irá depender do gênero do jogo.

2 MOTIVAÇÃO E ESTADO DA ARTE

Já foi dito que MMOGs se caracterizam por uma grande quantidade de jogadores. Para dar suporte a essa grande quantidade de participantes, geralmente é usado o paradigma cliente-servidor, sendo que toda interação é feita através da máquina servidora. Neste paradigma, o computador servidor possui a cópia oficial do estado do jogo e é responsável pela computação sobre a mesma, enquanto o cliente atua na apresentação do estado para o usuário. O servidor é responsável, também, por atualizar os clientes, em tempo real, sobre as alterações que ocorrem no ambiente virtual do jogo. A centralização nesse computador servidor, devido ao grande número de participantes, faz com que seu custo de manutenção seja bastante elevado. Geralmente, é necessária uma capacidade de comunicação com a Internet de alguns GBps e poder de processamento suficiente para executar toda a simulação. Torna-se inviável, por exemplo, a manutenção de jogos online maciçamente multijogador por empresas pequenas ou grupos independentes com orçamento limitado.

Em razão deste custo elevado, tem-se buscado soluções alternativas, distribuindo-se de alguma maneira as cargas de processamento e de comunicação decorrentes do jogo. Geralmente, é proposto o uso de redes P2P (ref) ou, em alguns casos GRIDs. Porém, um MMOG é uma simulação de um ambiente virtual dinâmico, com limite máximos de atraso entre o envio das ações do jogador e recebimento do estado resultante. Além disso, possui outros requisitos, que precisam ser observados com atenção de maneira que os jogadores tenham uma experiência adequada de jogo. Distribuir toda esta estrutura entre diferentes máquinas ligadas através de conexões através da Internet – como no caso das redes P2P – e ao mesmo tempo satisfazer os requisitos de um MMOG torna-se uma tarefa desafiadora, que tem motivado diversos trabalhos de pesquisa nos últimos anos.

Nas seções a seguir, serão apresentados alguns requisitos dos jogos MMOG, além do estado da arte no que se refere a distribuição da infra-estrutura de suporte a este tipo de jogo. Por fim, serão apresentadas questões de pesquisa que ainda estão em aberto, e que podem ser tratadas em um trabalho de doutorado.

2.1 Requisitos dos jogos MMOG

Durante o desenvolvimento de trabalhos anteriores (BEZERRA; CECIN; GEYER, 2008; VILANOVA et al., 2008) e deste plano de curso de doutorado, além de consulta à literatura da área (DARLAGIANNIS; HECKMANN; STEINMETZ, 2006; SCHIELE et al., 2007) foram identificados alguns requisitos dos jogos MMOG, que são os mais importantes do ponto de vista da infra-estrutura de suporte:

- **Persistência** - Jogos MMOG, em geral, consistem em um mundo virtual onde os

jogadores interagem entre si e com o mundo virtual, alterando o estado do mesmo e de seus avatares (representação virtual do jogador no jogo). Espera-se que cada jogador possa deixar o jogo e voltar algum tempo depois sem que essas mudanças tenham sido desfeitas. Por este motivo, é necessário suporte a persistência dos estados de cada avatar e do mundo virtual como um todo;

- **Disponibilidade** - Em geral, jogos MMOG são pagos - tais como World of Warcraft (BLIZZARD, 2004) - e, por causa disso, seus usuários costumam ser exigentes quanto ao quesito de disponibilidade. Assim, deseja-se que, a qualquer momento, o sistema esteja disponível;
- **Consistência** - Jogos multijogador consistem de um ambiente virtual compartilhado entre os jogadores, que disputam a mesma partida. Sendo assim, é necessário que aquele ambiente seja representado da mesma maneira nas máquinas dos diferentes jogadores para que estes possam interagir entre si e com o mundo de maneira adequada;
- **Segurança** - Em um sistema onde há dezenas de milhares de participantes, não é razoável confiar em todos eles, o que torna necessário algum mecanismo que proveja segurança. Segurança aqui se refere não somente a autenticação de usuários cadastrados, como também verificar se os estados/ações que os diferentes jogadores enviam são válidos. Jogadores trapaceiros, por exemplo, podem alterar o software executado em suas máquinas de forma que sejam enviados estados/ações que lhes dêem alguma vantagem no jogo;
- **Escalabilidade** - Levando-se em conta o número de participantes em um jogo MMOG, é necessário que o sistema que lhe dá suporte seja capaz de manter a qualidade do serviço a despeito da carga crescente de processamento e comunicação exigida para isso.

2.2 Trabalhos relacionados

A **arquitetura cliente-servidor** supre diversos aspectos necessários para a execução satisfatória de jogos do tipo MMOG, o que inclui um alto nível de controle sobre o sistema como um todo, o que facilita autenticação, persistência e segurança. Porém isso custa caro, como já foi dito, além de ser um possível gargalo. Objetivando minimizar este problema, foram propostas algumas alternativas. Uma delas é a de usar computação agregada, onde um cluster, ao invés de um computador único, faz o papel de servidor. Tal abordagem tem um ganho expressivo de poder de processamento, mas não resolve todos os problemas dos jogos online maciçamente multijogador. Deve-se prover também a largura de banda necessária para dar suporte ao tráfego intenso entre o servidor e os jogadores.

Outra abordagem possível é a **arquitetura par-a-par**, ou P2P, onde se divide a simulação entre os computadores envolvidos. Pode-se ter um sistema sem qualquer servidor, onde os pares (antes clientes), que são as máquinas dos jogadores, entram em algum tipo de acordo para os diversos passos da simulação. No que se refere à escalabilidade, tal abordagem não é ótima, pois garantir esse "acordo" é custoso em termos de troca de mensagens (LAMPORT; SHOSTAK; PEASE, 1982). Ainda que seja eleito um dos pares para decidir o andamento da simulação, ainda haverá o problema de que todos os pares precisarão trocar mensagens com todos. Tendo-se n pares, há uma complexidade de $O(n^2)$

trocas de mensagem para cada passo da simulação. É evidente que tal abordagem não é tão escalável quanto se possa querer para um sistema onde se pretende executar um jogo online maciçamente multijogador. Além disso, seria necessário prover armazenamento distribuído e recuperação dos estados do jogo.

Alguns trabalhos já foram realizados no sentido de tornar jogos em redes P2P mais escaláveis, como (SCHIELE et al., 2007; ?; ?). Para reduzir o tráfego entre os pares, cada par pode enviar atualizações de estado apenas àqueles que tiverem interesse nas mesmas. Para atingir este objetivo, (SCHIELE et al., 2007) sugere que o ambiente virtual seja dividido em regiões. O objetivo disto é que cada região funcione como se fosse um jogo independente, de menor escala. Os jogadores cujos avatares estivessem em determinada região poderiam formar uma pequena rede par-a-par, e decidirem entre eles o andamento do jogo naquela área do mundo do jogo. Quando um avatar se movesse de uma região para outra, deixaria o grupo da primeira e se uniria ao grupo da sua nova região. Para que o ambiente seja contíguo, ou seja, para que avatares possam mover-se livremente entre regiões adjacentes, alguns dos pares mantêm conexões com pares das regiões vizinhas, e é através destes pares que se consegue as informações necessárias para que haja essa transferência de grupo.

Ainda na proposta de (SCHIELE et al., 2007), cada região tem um coordenador. O coordenador é eleito entre os pares ali presentes e se encarrega, apenas, de decidir para quem cada atualização de estado interessa - ele não intermedia as trocas de mensagens entre os pares, que se comunicam diretamente entre si. No entanto, tal abordagem se baseia no fato de que o coordenador é confiável, o que não pode ser garantido, já que o software utilizado pelo jogador pode ter sido alterado de forma a se comportar de maneira incorreta a fim de beneficiá-lo. Outra abordagem seria a de eleger múltiplos coordenadores por região, mas isso implicaria na implementação de algum mecanismo de votação, além de depender da disponibilidade de pares para gerenciarem. Além disso, não seria eliminada a necessidade de cada par enviar atualizações de estado a diversos outros pares. Outros trabalhos que têm proposta bastante semelhante são (EL RHALIBI; MERABTI, 2005) e (IIMURA; HAZEYAMA; KADOBAYASHI, 2004).

Existem também as propostas de **arquiteturas híbridas**, que utilizam servidores ao mesmo tempo em que fazem uso de infra-estrutura P2P. É proposta, por exemplo, uma arquitetura onde pares e servidor dividem a simulação do jogo (CHEN; MUNTZ, 2006). O ambiente virtual é dividido em regiões, cada uma gerenciada por um dos pares, que atua como um sub-servidor. É para ele que cada um dos outros participantes naquela região envia cada atualização de estado, que de lá é encaminhada aos outros pares interessados na mesma. Isto pode gerar problemas de segurança e disponibilidade, já que não há garantias de que aquele par é confiável para servir aquela região, ou que ele irá permanecer no sistema enquanto for necessário. Para prover tolerância a falhas, os autores sugerem o uso dos agregados de pares - que consistem no gerenciador de região, mais pares "reserva" - eles recebem do gerenciador da região as ações recebidas de jogadores, que são processadas por estes "reservas", que mantêm uma réplica do estado do jogo contido no gerenciador daquela região. Além de prover robustez ao sistema, no caso do gerenciador sofrer colapso, estes nodos reserva podem detectar falhas na simulação e executar procedimentos de recuperação. Porém, tal abordagem não resolve o problema do par eleito para ser o gerenciadores da região não ser confiável, pois não há qualquer tipo de acordo entre ele e seus reservas em relação ao andamento do jogo. Por fim, se um nodo gerenciador estiver sobrecarregado, ele simplesmente devolve ao servidor parte da carga que lhe foi atribuída, mantendo a dependência de uma infra-estrutura centralizada.

Outra arquitetura híbrida é a do FreeMMG (CECIN et al., 2004). Nela, os pares se organizam de maneira P2P em cada região do ambiente virtual e o servidor intermedia a comunicação entre diferentes regiões. Novamente, tem-se o problema da segurança: os pares dentro de uma região controlam a simulação que ocorre ali, podendo subvertê-la. É feita uma abordagem probabilística, utilizando um par selecionado aleatoriamente de outro ponto do ambiente para verificar a simulação naquela região. Espera-se que, caso os pares ali desejem entrar em conluio para subverter o jogo, pelo menos o nodo inserido ali detecte as ações inválidas e reporte ao servidor. No entanto, além de não garantir completamente segurança, persiste o problema de poder haver muitos pares se comunicando com muitos, o que pode comprometer a qualidade do jogo.

Por fim, já foram propostas alguns modelos que sugerem o uso de arquitetura cliente-servidor, porém com a distribuição do lado servidor. Em (ASSIOTIS; TZANOV, 2006), por exemplo, é proposta a divisão do mundo virtual em regiões menores, cada uma atribuída a um diferente servidor. Os autores tratam a questão da consistência, através de um mecanismo de travas – a cada entidade presente no jogo é associada uma trava, que precisa ser obtida antes de algum dos servidores fazer quaisquer alterações. Além disso, tentou-se atacar o problema dos *hotspots* (aglomerados de jogadores em uma área relativamente pequena) através do particionamento recursivo da área sobrecarregada, até que esta sobrecarga seja eliminada ou não houver mais servidores disponíveis. No entanto, há um limite para este reparticionamento, pois áreas muito pequenas implicam em avatares migrando entre regiões com maior frequência e, conseqüentemente, em mais tráfego entre servidores. Ademais, os autores deste trabalho, assim como em (NG et al., 2002; CHERTOV; FAHMY, 2006; LEE; LEE, 2003), consideram que os nodos servidores estão conectados através de uma rede de alta velocidade e pouco atraso, sendo que suas soluções são apenas parcialmente aplicáveis em um cenário com recursos altamente dinâmicos e voláteis, como são as redes P2P montadas com recursos de voluntários. Tais redes têm como problemas inerentes: baixa disponibilidade e dependabilidade, largura de banda escassa e baixo poder de processamento, quando comparada com servidores dedicados mantidos por empresas fabricantes de MMOGs.

2.3 Questões em aberto

O que se pretende investigar é o uso de um sistema servidor composto de "super-nodos" distribuídos geograficamente. Cada super-nodo teria uma capacidade de processamento e link de comunicação mínimos exigidos para fazer parte do sistema. No entanto, como a simulação seria distribuída entre eles, não seria necessário que cada um destes super-nodos tivesse a mesma capacidade de processamento ou comunicação possuída pelos grandes servidores centrais usados comumente para jogos MMOG. O mundo do jogo poderia então ser dividido em regiões, cada uma gerenciada por um destes super-nodos. Dessa forma, os jogadores que estivessem em uma dada região apenas enviariam atualizações de estado àquele nodo servidor, que as processaria, armazenaria e enviaria para a quem fossem relevantes.

Utilizando esta abordagem, seria preciso apenas garantir a confiabilidade dos nodos servidores, que, então, verificariam se cada ação dos jogadores é válida. Assim, torna-se mais simples a tarefa de garantir segurança no sistema, pois pode-se assumir que cada super-nodo é confiável. Outro aspecto - o da disponibilidade - seria melhor satisfeito, pois cada região do mundo virtual estaria sendo servida por um nodo que estaria dedicado a esta tarefa.

Seguindo esta última vertente, existem alguns problemas que, ou estão por resolver, ou estão sendo pesquisados atualmente, tais como:

- Encontrar uma maneira de particionar o espaço virtual, mapeando cada região a um nodo servidor, de forma que nodos com boa comunicação entre si estejam próximos na topologia lógica da rede;
- Balancear dinamicamente a carga da simulação entre os nodos servidores, de forma que seja mantida uma qualidade mínima de serviço em cada uma das diferentes regiões do ambiente virtual;
- Manter consistente a simulação nas diferentes regiões, entre jogadores da mesma região, e de jogadores em regiões adjacentes;
- Coordenar a entrada, a saída e a detecção de colapso de nodos do sistema;
- Distribuir o armazenamento persistente do estado dos jogadores, de forma a poder recuperá-lo quando necessário;
- Otimizar a comunicação entre clientes e servidores, de forma a economizar o máximo possível de largura de banda, sem que isso seja perceptível pelos jogadores;
- Como cada nodo tem recursos mais limitados que os de um grande servidor central, deve-se buscar algoritmos adaptativos que tornem a simulação (atualizações de estado e sincronização da simulação entre os clientes) coerente com os recursos disponíveis no momento;

Foram feitos trabalhos que buscam tratar alguns destes problemas, como balanceamento de carga (LEE; LEE, 2003), assim como foram feitas algumas propostas de arquiteturas de suporte a MMOG (NG et al., 2002) e (ASSIOTIS; TZANOV, 2006). No entanto, tais propostas levam em consideração que os servidores são de grande porte e ligados através de links de comunicação poderosos, havendo muito pouco atraso entre eles. Cabe, então, fazer a análise da aplicabilidade das técnicas já pesquisadas em um contexto de recursos mais escassos e propor, tendo isso em vista, aperfeiçoamento das mesmas e uma arquitetura que as integre, além de efetuar implementação e testes de um protótipo que as utilize.

3 OBJETIVO

3.1 Objetivo geral

O objetivo é criar um modelo arquitetural de suporte a jogos online maciçamente multijogador. Seguindo este modelo, deverá ser possível formar um sistema geograficamente distribuído de nodos servidores, utilizando a Internet como meio de ligação entre os mesmos. Além de distribuir as cargas computacionais (comunicação e processamento) impostas pelo grande número de jogadores, este modelo deverá incluir técnicas com o fim de reduzir o tráfego entre os componentes deste sistema (nodos servidores e máquinas clientes dos jogadores), economizando largura de banda. Tais técnicas são o foco deste trabalho, tendo em vista que se considera um cenário com recursos voláteis e limitados, já que a ligação entre os nodos será feita através da Internet.

Existem outros aspectos relevantes, além da distribuição do servidor e da economia de largura de banda, que poderão ser acrescentados ao modelo, dependendo da disponibilidade de tempo para realizar tal tarefa.

3.2 Objetivos específicos

4 PLANO DE TRABALHO

4.1 Metodologia

Para atingir o objetivo definido na seção anterior, deverão ser cumpridas algumas etapas, que são as seguintes:

1. Revisar as técnicas utilizadas para dar suporte distribuído a jogos MMOG;
Será dada continuidade à investigação bibliográfica que já foi iniciada para elaboração deste plano de estudos e pesquisa, além de buscar separá-las em categorias segundo: contexto considerado e problemas que buscam solucionar.
2. Analisar a aplicabilidade destas técnicas em cenários em que cada nodo do sistema tem recursos mais limitados;
Técnicas utilizadas para dar suporte distribuído a jogos MMOG costumam se basear em cenários onde há grande disponibilidade de recursos, com grandes servidores ligados através de rede local com capacidade de comunicação super-dimensionada. Será analisada a viabilidade de serem utilizadas em um contexto onde os nodos têm recursos mais limitados.
3. Propor integração das mesmas, com otimizações focadas no contexto definido (disponibilidade de super-nodos, ao invés de servidores poderosos);
Buscar-se-á fazer aperfeiçoamento de algumas destas técnicas, tendo como ênfase o contexto de super-nodos distribuídos geograficamente e, posteriormente, será proposta uma integração das mesmas.
4. Implementar um protótipo que integre as técnicas estudadas e/ou desenvolvidas nas etapas anteriores;
5. Efetuar simulações que demonstrem o resultado obtido do trabalho e avaliar seus resultados.

4.2 Cronograma

A Tabela 4.1 apresenta o cronograma de atividades a serem desenvolvidas, conforme citadas abaixo

1. Levantamento bibliográfico do estado-da-arte das técnicas de distribuição do suporte a MMOG;

2. Levantamento bibliográfico de técnicas de distribuição em geral, que possam ser aplicadas a jogos MMOG;
3. Análise e categorização destas técnicas;
4. Elaboração de um modelo arquitetural de suporte distribuído a jogos MMOG;
5. Integração das técnicas no modelo proposto;
6. Implementação de um protótipo com o fim de testes;
7. Levantamento dos simuladores mais indicados para efetuar tais testes;
8. Execução dos testes em um dos simuladores pesquisados;
9. Análise dos resultados;
10. Apresentação do seminário de andamento na semana acadêmica;
11. Escrita da dissertação;
12. Escrita de artigos;
13. Entrega da dissertação;
14. Defesa da dissertação;

Tabela 4.1: Cronograma

Etapa	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez	Jan	Fev
1	X	X	X											
2	X	X	X											
3		X	X	X										
4			X	X	X									
5				X	X									
6					X	X	X	X						
7							X	X						
8									X					
9										X	X			
10					X									
11					X	X	X	X	X	X	X	X		
12									X	X	X	X	X	
13												X		
14														X

4.3 Planejamento dos créditos

5 INFORMAÇÕES DO CANDIDATO

5.1 Dados de identificação

5.2 Experiência com pesquisa

REFERÊNCIAS

ASSIOTIS, M.; TZANOV, V. A distributed architecture for MMORPG. **Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games**, [S.l.], 2006.

BEZERRA, C. E. B.; CECIN, F. R.; GEYER, C. F. R. A3: a novel interest management algorithm for distributed simulations of mmogs. **Proceedings of the 12th IEEE International Symposium on Distributed Simulation and Real Time Applications, 2008, Vancouver, BC, Canada**, [S.l.], p.35–42, 2008.

BLIZZARD. **Starcraft**: brood war. <http://www.blizzard.com/broodwar/>.

BLIZZARD. **World of Warcraft**. <http://www.worldofwarcraft.com/>.

CECIN, F.; REAL, R.; OLIVEIRA JANNONE, R. de; GEYER, C.; MARTINS, M.; BARBOSA, J. FreeMMG: a scalable and cheat-resistant distribution model for internet games. **IEEE Int. Sym. on Distributed Simulation and Real-Time Applications**, [S.l.], p.83–90, 2004.

CHEN, A.; MUNTZ, R. Peer clustering: a hybrid approach to distributed virtual environments. **Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games**, [S.l.], 2006.

CHERTOV, R.; FAHMY, S. Optimistic Load Balancing in a Distributed Virtual Environment. **Proceedings of the 16th ACM International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)**, [S.l.], 2006.

DARLAGIANNIS, V.; HECKMANN, O.; STEINMETZ, R. Peer-to-peer applications beyond file sharing: overlay network requirements and solutions. **e & i Elektrotechnik und Informationstechnik**, [S.l.], v.123, n.6, p.242–250, 2006.

EL RHALIBI, A.; MERABTI, M. Agents-based modeling for a peer-to-peer MMOG architecture. **Computers in Entertainment (CIE)**, [S.l.], v.3, n.2, p.3–3, 2005.

IIMURA, T.; HAZEYAMA, H.; KADOBAYASHI, Y. Zoned federation of game servers: a peer-to-peer approach to scalable multi-player online games. **Proceedings of ACM SIGCOMM 2004 workshops on NetGames' 04: Network and system support for games**, [S.l.], p.116–120, 2004.

LAMPORT, L.; SHOSTAK, R.; PEASE, M. The Byzantine Generals Problem. **ACM Transactions on Programming Languages and Systems (TOPLAS)**, [S.l.], v.4, n.3, p.382–401, 1982.

LEE, K.; LEE, D. A scalable dynamic load distribution scheme for multi-server distributed virtual environment systems with highly-skewed user distribution. **Proceedings of the ACM symposium on Virtual reality software and technology**, [S.l.], p.160–168, 2003.

NCSOFT. **Lineage II**. <http://www.lineage2.com/>.

NG, B.; SI, A.; LAU, R.; LI, F. A multi-server architecture for distributed virtual walk-through. **Proceedings of the ACM symposium on Virtual reality software and technology**, [S.l.], p.163–170, 2002.

SCHIELE, G.; SUSELBECK, R.; WACKER, A.; HAHNER, J.; BECKER, C.; WEIS, T. Requirements of Peer-to-Peer-based Massively Multiplayer Online Gaming. **Proceedings of the Seventh IEEE International Symposium on Cluster Computing and the Grid**, [S.l.], p.773–782, 2007.

VALVE. **Counter-Strike**. <http://www.counter-strike.net/>.

VILANOVA, F. J.; BEZERRA, C. E. B.; CRIPPA, M. R.; CECIN, F. R.; GEYER, C. F. R. P2PSE - A Peer-to-Peer Support for Multiplayer Games. **VII Brazilian Symposium on Computer Games and Digital Entertainment - Computing Track, 2008, Belo Horizonte, MG, Brazil**, [S.l.], p.47–53, 2008.

ASSINATURAS

Carlos Eduardo Benevides Bezerra

Prof. Dr. Cláudio Fernando Resin Geyer