

Protótipo de motor de servidor de jogos *online* em massa

Autor: Daniel Presser

Orientador: Francisco Adell Péricas

Roteiro da Apresentação

- Introdução
- Objetivos do trabalho
- Fundamentação teórica
- Desenvolvimento do trabalho
- Conclusão
- Extensões

Introdução

- *MMOG, Massive Multiplayer Online Games*
 - Jogados exclusivamente *online*, em mundos virtuais persistentes
 - Sociedades virtuais
 - Normalmente no estilo RPG (*Role Playing Game*)
- Problemas
 - Crescimento acelerado
 - Pouco tempo de pesquisa
 - Custos elevados para manutenção do ambiente

Objetivos do trabalho

- Desenvolver um protótipo de motor genérico para servidores de jogos *online* em massa que sigam estilo RPG
- Permitir extensão do jogo através de *scripts*
- Utilizar arquiteturas e técnicas que visem a otimização do processamento no servidor
 - Arquitetura de camadas
 - Modelo publicador/assinante
 - IOCP
 - Linguagem de *Scripts* Lua

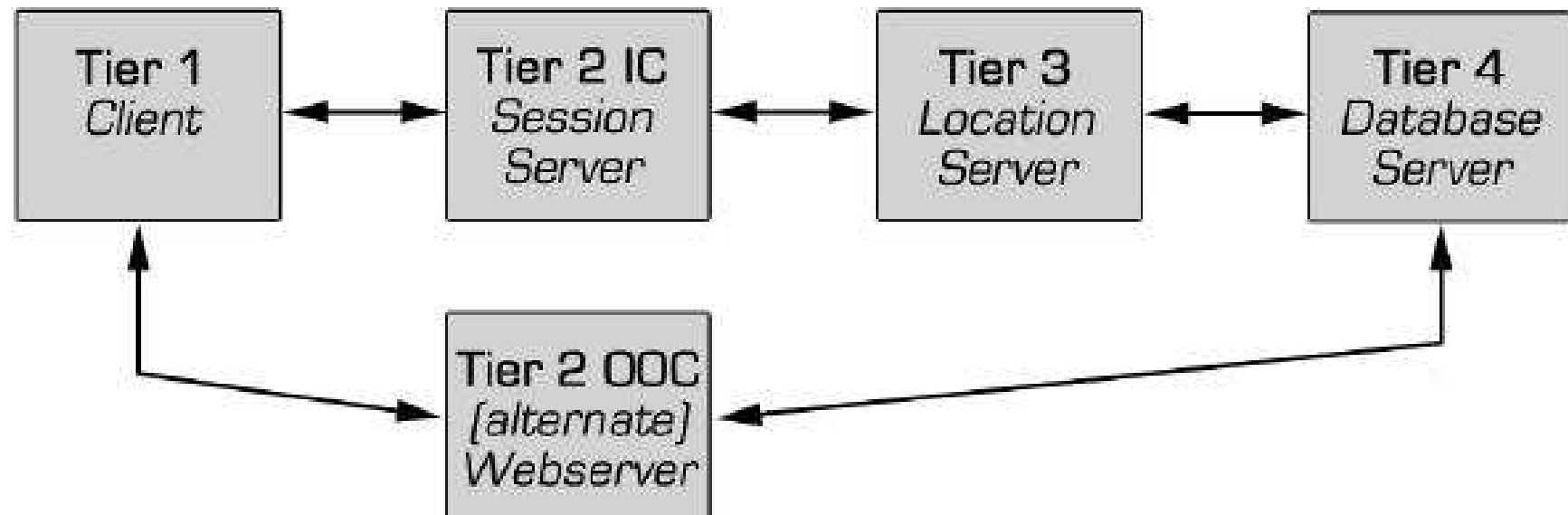
Jogos *online* em massa e RPG

- Jogos *online* em massa
 - Caracterizados pelo grande número de jogadores num único ambiente virtual persistente
- MMORPG - RPG *online* em massa
 - Grandes mundos virtuais persistentes
 - Jogadores interpretam papéis através de personagens do mundo virtual
 - Existência de classes para os personagens
 - Evolução dos personagens conforme enredo

Arquitetura de Camadas

- Visa divisão do servidor em camadas conforme suas responsabilidades
 - Camada de sessão: controle da conexão e persistência dos dados dos jogadores
 - Camada de localização: controle do ambiente do mundo virtual
 - Camada cliente: *interface* do jogo com o usuário
 - Camada de dados: armazenagem dos dados dos jogadores
 - Camada alternativa: manter a aplicação da camada cliente atualizada

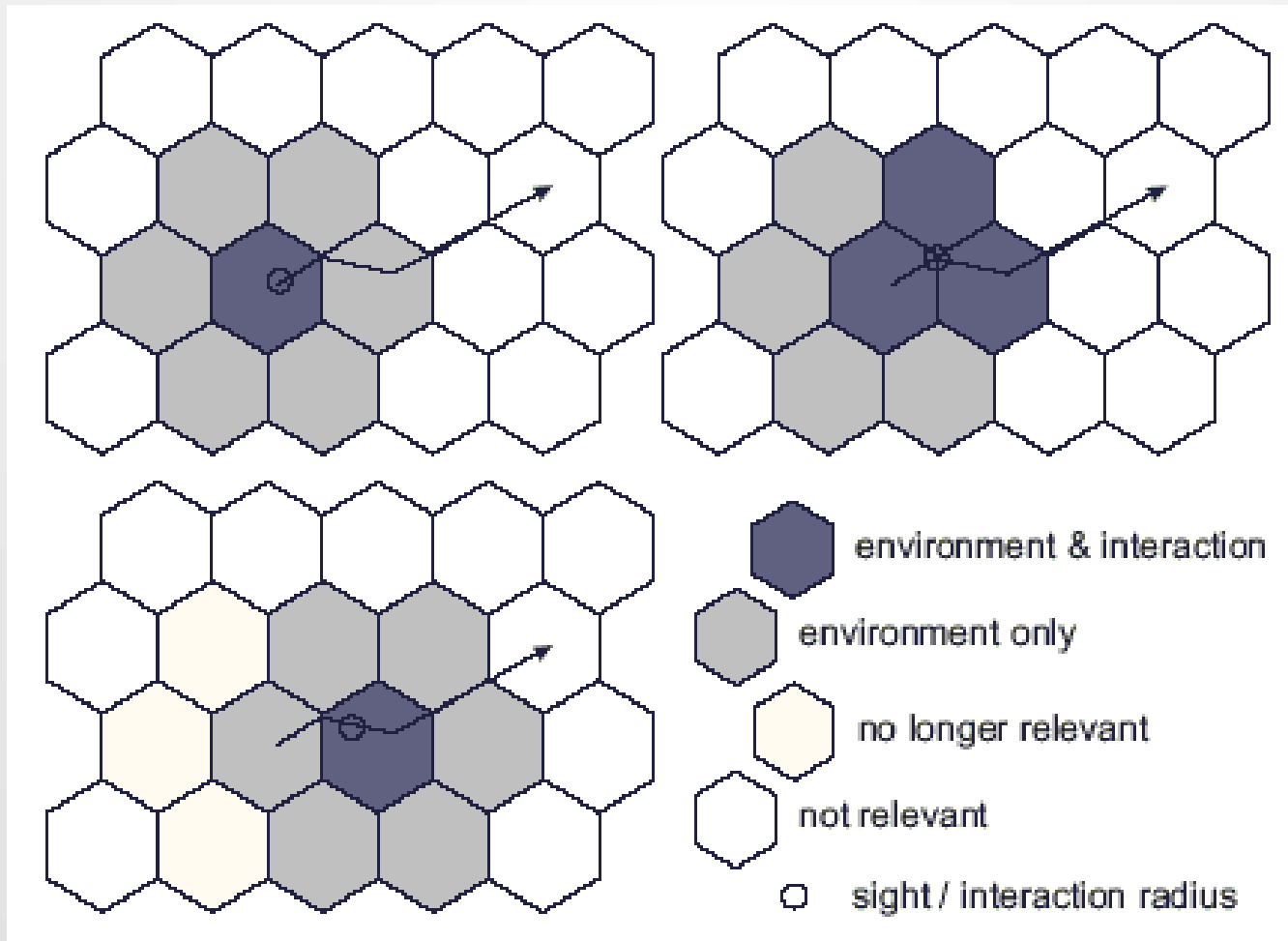
Arquitetura de Camadas - Visão geral



Modelo Publicador/Assinante

- Visa divisão do mapa do mundo virtual em partes menores, para dividir o processamento no servidor
- Cada parte de mapa publica dois canais
 - Interação: mensagens de interação do personagem com o ambiente e outros personagens (próximos)
 - Ambiente: mensagens de mudanças no ambiente do jogo ao redor do personagem
- Clientes assinam os canais que os interessam, filtrando o conteúdo que desejam receber

Modelo Publicador/Assinante - Visão geral



IOCP - *Input/Output Completion Port*

- Gerenciamento eficiente de grandes volumes de conexões
 - Através de uma fila, controla os eventos nas conexões
 - Aplicação associa conexões à fila
 - Aplicação aguarda eventos através de *threads* de trabalho
 - Sistema operacional notifica aplicação sobre eventos nas conexões
 - Aplicação acessa diretamente as conexões com eventos pendentes
 - Exclusivo do Windows (a partir do NT)

Linguagem de *Scripts* Lua

- Interpretador e linguagem
- Implementada utilizando C
- Simples, leve, rápida e nacional
- Utilizada por grandes empresas (LucasArts, BioWare, etc)
- Sintaxe com características de C e Pascal
- Variáveis e funções definidas em qualquer parte do código
- Variáveis não tipadas
- Comunicação através de funções disponibilizadas pela aplicação e registradas no ambiente Lua

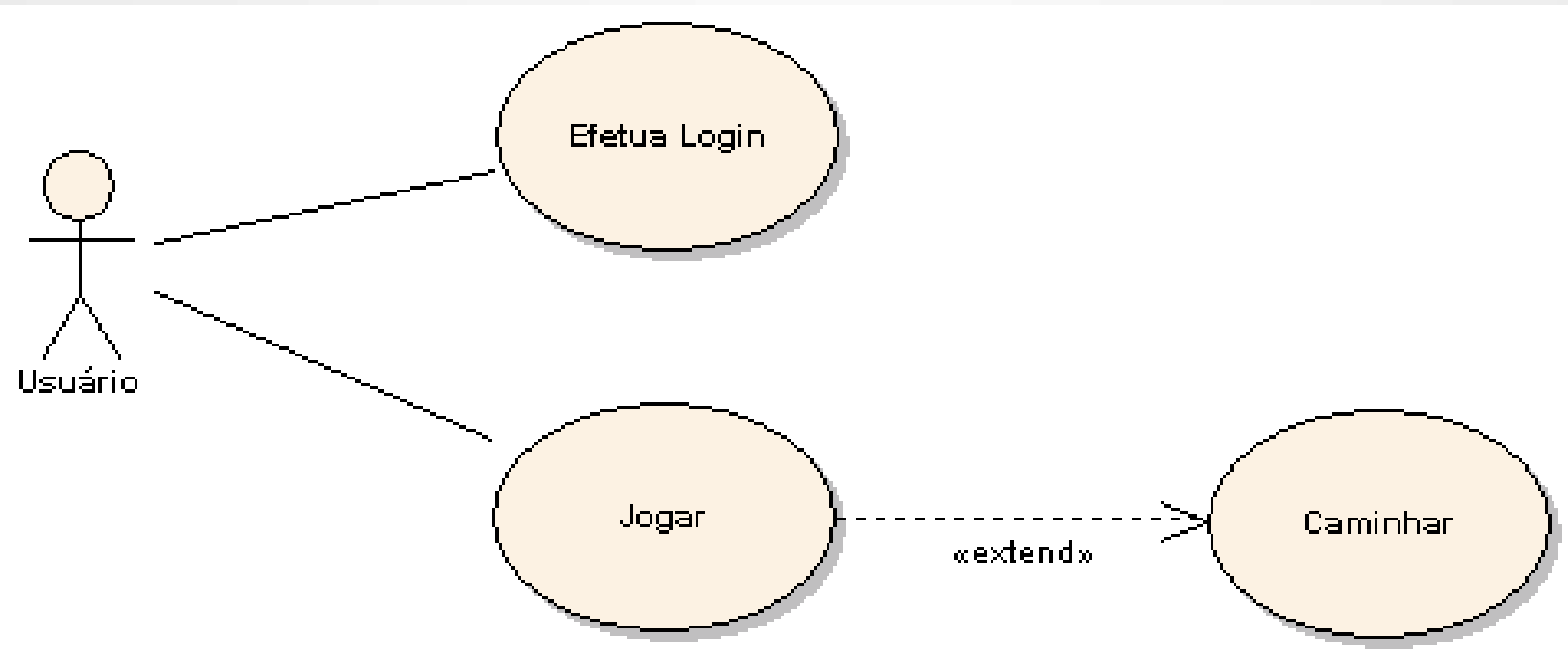
Principais requisitos

- Gerenciar conexões de usuários (RF)
- Gerenciar o mundo virtual configurável (RF)
- Disparar eventos conforme ações do usuários ou do ambiente virtual (RF)
- Permitir criação de *scripts* para os eventos disparados (RF)
- Permitir ao usuário locomover-se no mundo virtual e conversar com usuários próximos a ele (RF)
- Implementado em C++, utilizando banco MySQL e linguagem de *scripts* Lua (RNF)

Especificação

- UML (*Unified Modeling Language*)
 - Diagramas de casos de uso
 - Diagramas de atividades
 - Diagramas de classes

Casos de uso - Usuário



Casos de Uso - Engenheiro de jogo

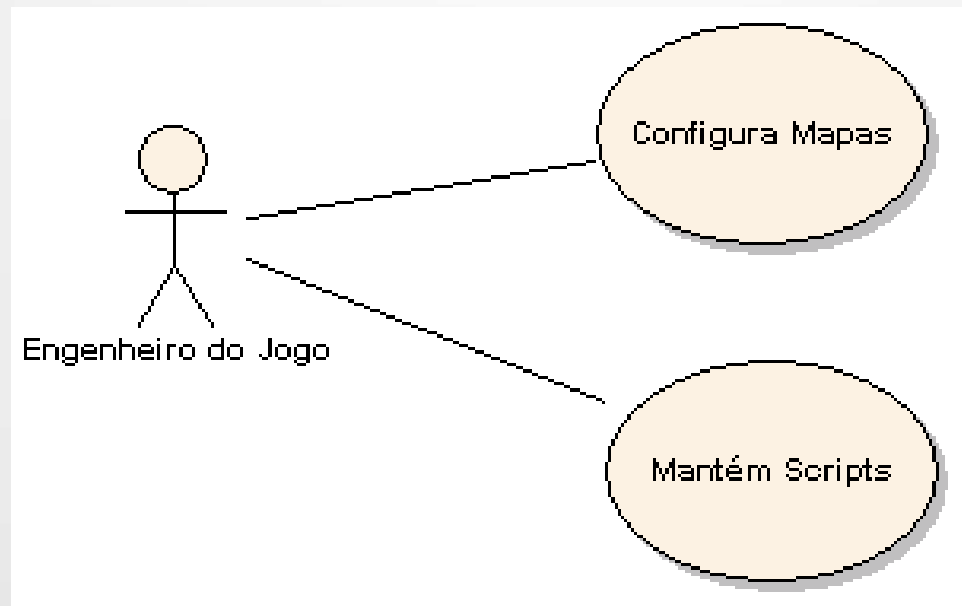


Diagrama de Atividades - Visão geral da aplicação

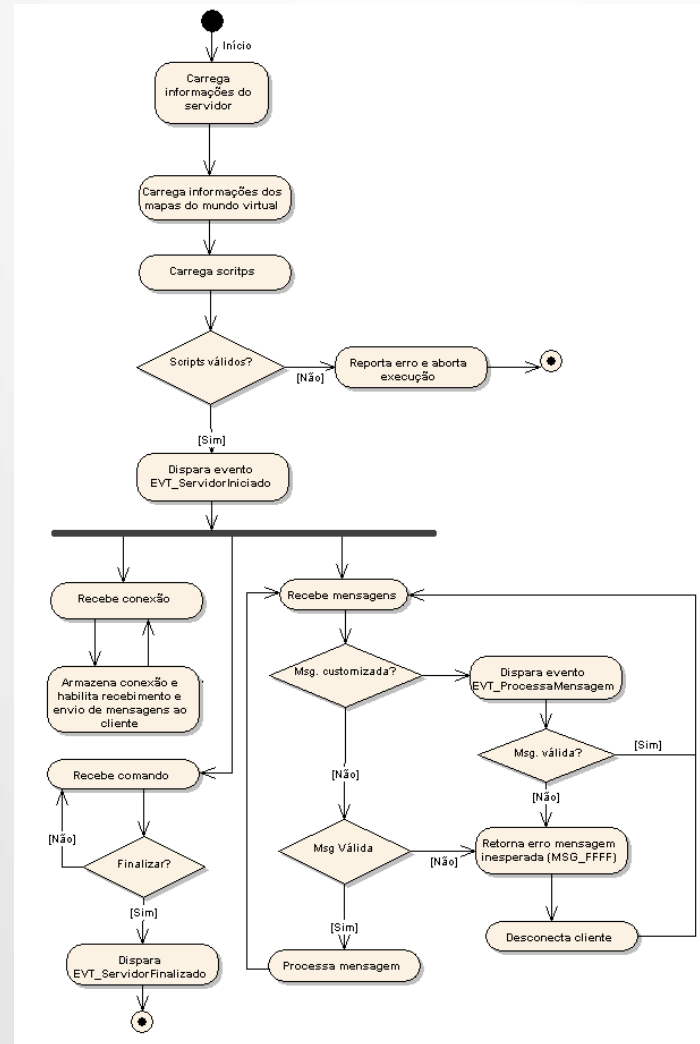
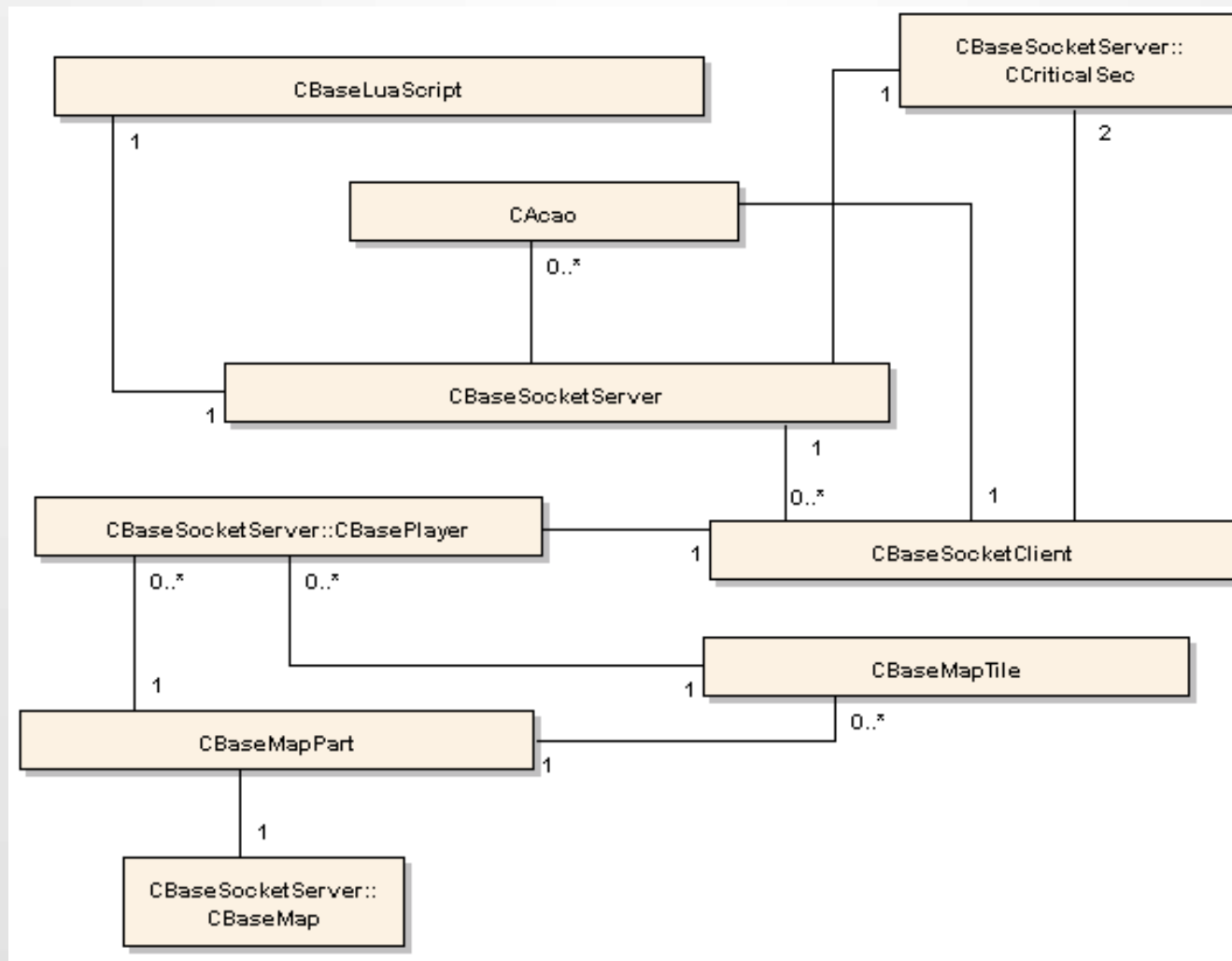


Diagrama de classes

Classes das camadas do servidor



Implementação

Técnicas e ferramentas utilizadas

- Arquitetura de camadas simplificada em 4 camadas (cliente, sessão, localização e dados)
- Modelo publicador/assinante
 - Publicadores: camada de localização
 - Assinantes: camada de sessão
- IOCP
- Linguagem e interpretador de *scripts* Lua
- Visual Studio .Net 2003 e MySQL.
- Cliente de testes implementado no Borland Delphi 7

Implementação Interface das camadas

- Localização

```
C:\WINDOWS\System32\cmd.exe
E:\_Daniel\TCC\testes>LocationServer.exe 1
Iniciando...
*** SERVIDOR DE LOCALIZACAO ***

Iniciando scripts...
Scripts iniciados!
Iniciando WINSOCK...
WINSOCK iniciado.
Iniciando IOCP...
IOCP iniciado.
Iniciando conexões...
Conexões iniciadas!

**SERVIDOR INICIADO**

Pressione qualquer tecla para finalizar._
```

- Sessão

```
C:\WINDOWS\System32\cmd.exe
E:\_Daniel\TCC\testes>SessionServer.exe 1
Iniciando...
*** SERVIDOR DE SESSAO ***

Iniciando scripts...
Scripts iniciados!
Iniciando WINSOCK...
WINSOCK iniciado.
Iniciando IOCP...
IOCP iniciado.
Iniciando conexões...
Conexões iniciadas!

**SERVIDOR INICIADO**

Pressione qualquer tecla para finalizar._
```


Resultados e discussão

- De maneira geral, os resultados esperados foram obtidos
- Técnicas, ferramentas e modelos utilizados apresentaram resultados satisfatórios
- Extensibilidade comprovada pela definição dos *scripts* de ataque a outros personagens
- Intervalos entre mensagens e desempenho geral satisfatórios
 - Não foi possível executar testes em grande escala

Conclusões

- Protótipo implementado atingiu os objetivos
- As técnicas e modelos utilizadas mostraram-se viáveis ao desenvolvimento de MMOGs
- Principais limitações do protótipo:
 - Inimigos são apenas outros jogadores
 - Não há ferramentas para as configurações de mapas e scripts
 - Não há figura do administrador do jogo
 - Gerenciamento de memória e sincronização dos processos não são eficientes

Extensões

- Criação de inimigos que não sejam outros usuários, como monstros
- Criação de ferramentas para o gerenciamento do servidor
- Alteração do esquema de mapas, para suporte de mapas mais complexos
- Otimização dos diversos processos
- Criação de versões para outras plataformas
 - Utilizar tecnologias equivalentes ao IOCP

FIM