

Uso do Middleware CORBA na Distribuição de Ambientes Distribuídos para suportar Jogos Multi-Jogadores. Estudo de Caso “Projeto LUDOS TOP”

Everton S. De Souza, Luciano F. Silva, Marlene Roque, Alexandre Cardoso e Edgard Lamounier Jr.

Departamento de Engenharia Elétrica, Laboratório de Computação Gráfica –
Universidade Federal de Uberlândia (UFU)

Caixa Postal 38.400 – 34.3239-4148 – Uberlândia – MG – Brasil

eevesou@ieee.org, {fsluciano,marlene_roque}@hotmail.com,
{lamounier,alexandre}@ufu.br

***Abstract.** This work presents the project “LUDOS TOP”, a virtual educational environment associated with games. It has been utilized as model to implement distributed architecture through the Common Object Request Broker Architecture (CORBA) specification in order to support communication and distribution techniques. This solution shows new models of architecture to integration virtual environments with distributed systems, providing contribution and improvements for scalability, heterogeneity and distributed interaction.*

***Resumo.** Este trabalho apresenta o projeto “LUDOS TOP”, um ambiente virtual educacional associado com a área de jogos. Este sistema foi utilizado como modelo para implementação da arquitetura de distribuição através da especificação Common Object Request Broker Architecture (CORBA) para comunicação e técnicas de distribuição. Esta solução demonstra o novo modelo de arquitetura para integração de mundos virtuais com sistemas distribuídos, fornecendo contribuição e melhorias para escalabilidade, heterogeneidade e interação distribuída.*

1. Introdução

Desde 1958, quando Willian A. Hinginbotham usou um osciloscópio para simular um jogo virtual de tennis, a tecnologia computacional tem feito avanços em tamanhos e sofisticação. A alta velocidade dos dados numa rede internacional são partes da modernização, que tem se traduzido em evolução das tecnologias computacionais [Grenville, A., Mark, C. e Philip, B. (2006)].

Os jogos multi-jogadores de forma crescente é usado para treinamento militar e industrial, engenharia, projetos colaborativos e aplicações comerciais, incluindo shopping virtual com showrooms, conferências, suporte remoto ao cliente e educação à distância.

Atualmente, em jogos multi-jogadores, não raro jogadores estão fisicamente separados, comunicando-se através da máquina, consoles, palm-top estão conectados na rede.

A motivação para jogos multi-jogador está no uso da Internet e das redes convencionais permitindo que múltiplos usuários possam interagir em tempo real, mesmo que cada um desses usuários esteja localizado ao redor do mundo.

O objetivo do trabalho é explorar um novo modelo de arquitetura para ambientes virtuais distribuídos, juntamente com o CORBA aplicar novas técnicas de distribuição entre os ambientes virtuais, visando melhorias em escalabilidade, heterogeneidade e interação distribuída.

2. Trabalhos Relacionados

A seguir, são relacionados três ambientes multi-usuários correlatos ao trabalho proposto, procedendo-se assim a uma breve investigação sobre o nível de desenvolvimento a que tem chegado à utilização de jogos multi-jogadores:

- **Plato:** Foi desenvolvido no início de 1960 pela Universidade de Illinois pelo Professor Don Bitzer, como um pequeno sistema com simples terminais, para aprendizado baseado no computador. Foi o primeiro sistema que possuía arquitetura cliente/servidor, com comunicação por meio de um programa Telnet ao servidor mainframe usando o protocolo TCP [Postel, J e Reynolds, J. (1983)]. O jogo tinha a velocidade de 1200 bps (bytes por segundos), suportando mais de 1000 usuários e com boa resolução gráfica. Juntamente com a proposta especial de uma linguagem de programação chamada TUTOR, usada para escrever software educacional.

- **Peloton:** É um ambiente virtual distribuído de simulação ciclística, onde vários usuários conectados em diferentes computadores sobre a Internet são capazes de pedalar simultaneamente em tempo real usando diferentes dispositivos de entrada e saída, e foi desenvolvido pelo Bell Laboratories, Instituto de pesquisas da Lucent. [Gianpaolo, U., Mauricio, C., John, T. e Robert, E.]. Sua arquitetura demonstra integração entre vários sistema e comunicação via sockets. A aplicação cliente tem um núcleo de gerenciamento que controla as sessão, simulação e dispositivos com EAI e LiveConnect. Enquanto a aplicação servidora consiste de servidor de gerenciamento de sessão, repositório de cenas e Web Server. Esta solução não é completamente satisfatória devido limitações de conexões e performance devido a quantidade de sistemas.

- **Doom:** No final de 1993, a id Software lançou o Doom, um jogo de tiro em primeira pessoa. Embora outros jogos do gênero terem sido lançados antes, o Doom forneceu um poderoso engine de rápida comunicação e maior nível de realismo [Grenville, A., Mark, C. e Philip, B.]. Todos os jogadores estavam em independentes “peers” com uma réplica do jogo e comunicando diretamente com outro peer, todo 1/35 de um segundo. Doom tem uma topologia de rede peer-to-peer, onde permitia quatro jogadores para jogar cooperativamente usando o protocolo IPX (Internet Packet Exchange) combinado com a pilha SPX (Sequence Packet Exchange) da Novel em rede, funcionalmente equivalente com a pilha TCP/IP em que a Internet atualmente é baseada. [Doom Engine].

3. Arquitetura do Sistema

Esta seção apresenta a arquitetura do sistema proposto - Figura 1.

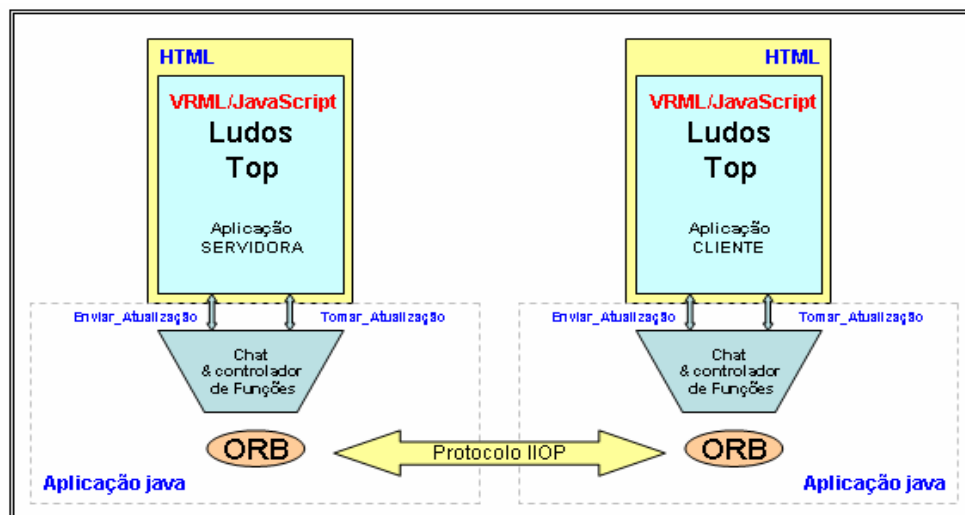


Figura 1. Arquitetura proposta para o sistema.

Resumidamente, verifica-se que:

- O jogo virtual é executado separadamente através de eventos, ou seja, primeiramente o servidor é iniciado na rede para disponibilizar seus serviços.
- Em seguida, os clientes serão iniciados, por meio do seu controlador de funções e vão se registrar com servidor enviando uma requisição a ORB que transportará pela protocolo IIOP ao servidor, onde o servidor será responsável por atualizar todos os clientes registrados;

3.1. Common Object Request Broker Architecture

CORBA é um padrão que foi definido em 1991 pela OMG (Object Management Group). O padrão CORBA é um sistema que permite que aplicações distribuídas em uma rede comuniquem-se entre si e troquem informações completamente distribuídas sendo executado em qualquer parte da rede ou Internet. [Object Management Group (2004)]. Como principais vantagens são: transparência de acesso aos dados, independência de linguagem e neutralidade de plataforma.

3.2. Object Request Broker

A ORB constitui como componente da arquitetura de comunicação, denotado como “objeto de transporte”, que possui a tarefa de estabelecer a comunicação entre cliente/servidor por meio do relacionamento entre objetos.

3.3. Técnica de Distribuição

Dentre as técnicas em sistemas distribuídos a técnica de callbacks distribuídos que possui mais padrões como: “Observer [Gamma, E., Helm, R., Johnson, R., Vlissides, J. (1995)]”, “Subject-Observer”, “Observer-Observable” ou “Publish-Subscribe”. A principal meta desses padrões é definir um-para-muitos, ou seja, dependência entre objetos distribuídos, quando um objeto muda de estado, todos seus dependentes são notificados e atualizados automaticamente [Mowbray, T. e Malveau, R. (1997)].

3.4. Quadro comparativo entre CORBA e Sockets

Nesta secção, apresenta-se uma tabela comparativa geral das tecnologias analisadas, de acordo com os diversos aspectos tomados como critério nesse trabalho.

Todos os resultados obtidos nos testes e análises estão sumarizados na Tabela 1.

Tabela 1. Comparativo do CORBA e Sockets

	Corba	Sockets
Implementação o concorrentes	Não	Sim
Transparência	Sim	Não
Técnicas de distribuição	Sim	Não

4. Estudo de Caso

O estudo de caso proposto baseia-se no jogo conhecido como “Quatro”, o qual foi desenvolvido pelo francês Blaise Müller em 1985. Neste jogo, utilizam-se 16 peças e um tabuleiro de 16 casas. A estratégia para vencer é alinhar, sejam em linha, coluna ou diagonal, peças que possuem “pelo menos um atributo” em comum (cor, tamanho, furo ou não, etc) [Freitas, M., Silva, L., Goulart, R., Lamounier, E. e Cardoso, A., 2007].

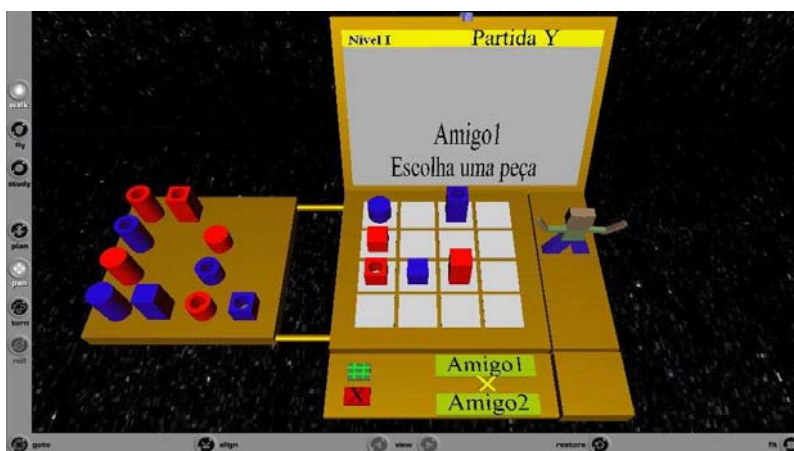


Figura 2. Projeto Ludos TOP.

O jogo 3D foi construído com as linguagens Virtual Reality Modeling Language (VRML) e JavaScript. O acesso ao sistema é obtido por meio de um navegador de Internet onde é realizado por um plug-in Cortona, que é executado sob supervisão do programa navegador.

5. Desenvolvimento e Funcionamento do Sistema

Para o desenvolvimento da distribuição do estudo de caso “Ludos Top” foram utilizadas as linguagens Java, IDL e CORBA. A linguagem Java foi usada para as interfaces de controle dos objetos virtuais que compõem o cenário tridimensional e Chat textual.

Na construção do sistema proposto, desenvolveu-se primeira uma interface de autenticação para ambiente virtual, tendo uma opção de escolha se o usuário gostaria de jogar on-line ou sozinho. Caso optar pela opção “Jogar On-line”, o jogador irá para a lista de jogadores para escolher o adversário.

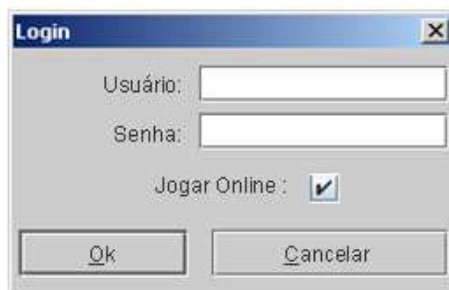


Figura 3. Autenticação no jogo multi-jogador.

Após a autenticação será conectado automaticamente ao Chat de textual e será incluído na lista de usuários on-line, portanto será apresentada a tela inicial do jogo – Figura 2.

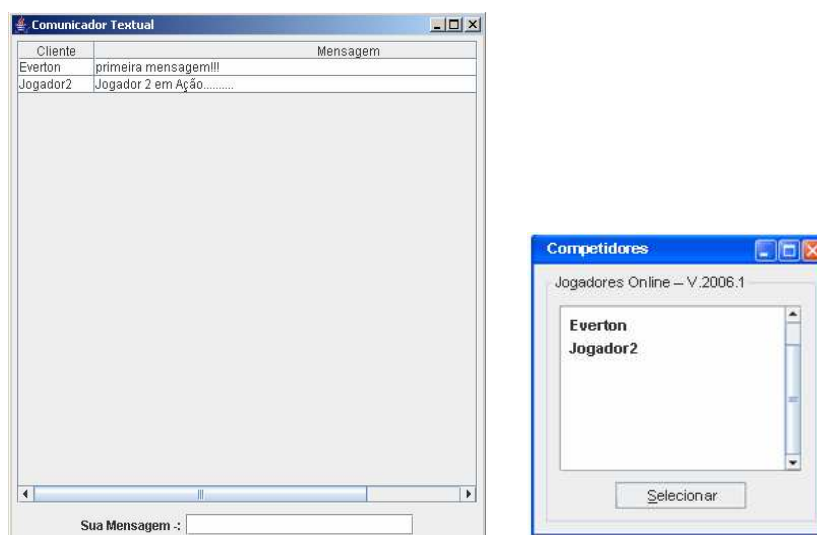


Figura 4. – Chat textual e lista de usuários conectados.

Na lista de jogadores on-line, o jogador pode selecionar com quem deseja jogar assim iniciará a competição.

6. Considerações Finais e Trabalhos Futuros

Este artigo teve como objetivo apresentar o desenvolvimento/distribuição de um jogo multi-jogador, cuja finalidade é permitir a utilização de uma plataforma distribuída usando o padrão Common Object Request Broker Architecture e mostrando suas vantagens, por meio de aplicações de técnicas de distribuição em relação a utilização de sockets.

A arquitetura CORBA indica eficiência como camada intermediária (*middleware*) na comunicação dos objetos, pois possibilitando a transparência para os programadores facilitando a implementação e apresentando o jogo como um modelo para aplicação de Educação a Distância multi-usuário.

Como trabalho futuro pretende avaliar escalabilidade junto ao CORBA e para futuras implementações sugere-se a utilização de Chat de voz em tempo real e/ou operações de ferramentas multi-usuário através da Internet, assim aumentar a interatividade do jogo com os jogadores.

7. Referências Bibliográficas

- Gianpaolo, U., Mauricio, C., John, T. e Robert, E. (1998) The Peloton Bicycling Simulator. In Proceedings of 3rd Symposium on the Virtual Reality Language VRML 98, ACM SIGGRAPH.
- Postel, J. e Reynolds, J. (1983). Telnet Protocol Specification. RFC 854.
- Doom Engine. (2007). http://doom.wikicities.com/wiki/Doom_networking_engine, acessado em Jan, 2007.
- Grenville, A., Mark, C. e Philip, B. (2006) Networking and Online Games: Understand and Engineering Multiplayer Internet Games, páginas 12 – 15, Editora Wiley.
- Freitas, M., Silva, L., Goulart, R., Lamounier, E. e Cardoso, A. (2006). “Proposta de Integração de Jogos e Realidade Virtual no Ensino/Aprendizagem de Matemática – Projeto LUDOS TOP”. ICECE – Conferência Internacional em Educação em Engenharia e Computação., UNISANTOS, 2007.
- DIEHL, S., (2001). Distributed Virtual Worlds. páginas 10 – 51. Editora Spring, 2001
- Aleksy, M., Korthaus, A., e Schader, M., (2005) Implementing Distributed Systems with Java and CORBA, Berlin, páginas 15-24, Editora Spring.
- Gamma, E., Helm, R., Johnson, R., Vlissides, J. (1995) Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Longman.
- RIBEIRO, M., Lamounier, E. e Cardoso, A. (2004) Uso de Corba na Distribuição de Ambientes Virtuais para Suportar Multidisciplinaridade no Processo de Educação”. IV Seminário de Realidade Virtual. SVR'2004.
- Object Management Group (2004). The Common Object Request Broker Architecture. Version 3.0.3, OMG Technical Document Number format/04-03-12, URL:<ftp://ftp.omg.org/pub/docs/formal/04-03-12.pdf>.
- Mowbray, T. e Malveau, R. (1997) CORBA Design Pattern. In: John Wiley & Sons, Chichester.