

# Sistemas Cooperativos de Realidade Virtual usando Java e VRML

Verison José Vicentin<sup>1</sup>  
Mario Massakuni Kubo<sup>1</sup>  
Wagner José Dizero<sup>1</sup>  
Claudio Kirner<sup>1,2</sup>

{verison, mkubo, wagner, claudio}@dc.ufscar.br

<sup>1</sup>Grupo de Realidade Virtual – GRV  
Departamento da Computação – DC  
Universidade Federal de São Carlos – UFSCar  
Caixa Postal 676 CEP 13.565–905  
São Carlos – SP – Brasil

<sup>2</sup>Grupo de Realidade Virtual – GRV  
Faculdade de Informática – FIM  
Fundação Eurípedes Soares da Rocha – FESR  
Caixa Postal 2041 CEP 17.525–901  
Marília – SP – Brasil

## Resumo

Este trabalho apresenta um sistema cooperativo de Realidade Virtual que foi desenvolvido através das linguagens Java e VRML. O sistema habilita vários usuários visualizarem, manipularem e compartilharem o mesmo ambiente virtual. Além disso, este artigo explora de vários recursos para o desenvolvimento de interfaces tridimensionais altamente interativas, que permite estabelecer os mecanismos de comunicação entre os usuários na Internet.

**Palavras-Chaves:** Realidade Virtual, CSCW, VRML, Java.

## Abstract

This work presents a Virtual Reality cooperative system, developed with Java and VRML languages. The system enables several users to view, manipulate and share the same virtual environment. Moreover, this paper explores of several resources to the development of highly interactive 3D interfaces, which allows to establish the communication mechanisms among Internet users.

**Keywords:** Virtual Reality, CSCW, VRML, Java.

## 1. Introdução

Atualmente, com o desenvolvimento alcançado pelas redes de computadores, a Internet surge como uma nova área de aplicação para a Realidade Virtual (RV). Aliando-se o poder da rede mundial de computadores como meio de comunicação, ao alto grau de interatividade e poder de visualização proporcionado pela RV, é possível a criação de sistemas de comunicação altamente estimulantes e com uma interface que permita um relacionamento mais natural entre os usuários. Nesse contexto, está cada vez mais marcante a presença na Internet, o que denominamos de ambientes virtuais cooperativos.

Um ambiente virtual cooperativo, como o próprio nome sugere, consiste em um ambiente virtual (interface gráfica tridimensional) compartilhado por vários usuários, na maioria das vezes distantes geograficamente, cada um interligado ao sistema por intermédio de uma rede de computadores. Nesse contexto, os usuários são representados dentro do ambiente virtual por um avatar, ou seja, um objeto (normalmente na forma humana) através do qual suas ações são refletidas.

O desenvolvimento de um sistema cooperativo de RV é um processo complexo. Além de requerer uma análise profunda dos requisitos do sistema, devem ser observados também uma série de outros aspectos relacionados à criação da interface tridimensional (modelagem do ambiente virtual), ao modelo de comunicação em rede utilizado, e às ferramentas e mecanismos de interação necessários ao relacionamento remoto entre os participantes do sistema.

Nesse artigo apresenta-se um estudo sobre sistemas cooperativos de RV, sendo inicialmente discutidos conceitos fundamentais sobre RV, sistemas cooperativos e o uso de interfaces de RV em tais sistemas. Em seguida, é feito um comentário sobre o uso das linguagens Java e VRML na implementação desse tipo de sistema. No final é analisado um sistema desenvolvido utilizando as tecnologias apresentadas.

## 2. Visão Geral

### 2.1 Realidade Virtual

Quando ouve-se falar em ambientes virtuais ou sistema baseados em RV, há uma tendência em se pensar no uso de equipamentos especiais, tais como capacetes, luvas, etc. Este ponto de vista é somente um exemplo da grande variedade de sistemas de RV e que busca focalizar o método de interação entre o usuário e o sistema [1]. Porém, a RV deve ser considerada em um contexto mais amplo, envolvendo também sistemas que não necessitem de equipamentos especiais e ainda preservem idéias básicas da RV: imersão, interação e envolvimento [2].

Com o passar do tempo, houve uma crescente evolução computacional, que introduziu dispositivos de RV mais sofisticados a um custo comercialmente menor. Além disso, um fator fundamental para o crescimento da RV foi o aumento do poder de processamento dos computadores pessoais, o que incentivou o uso da RV não Imersiva, que tratam as imagens geradas diretamente no monitor.

A interface de RV envolve um controle tridimensional altamente interativo de processos computacionais. Ao entrar no ambiente virtual (interface 3D) das aplicações, o usuário visualiza, manipula e explora os dados da aplicação em tempo real, usando seus sentidos, particularmente os movimentos naturais do corpo. A grande vantagem desse tipo de interface é que o conhecimento intuitivo do usuário a respeito do mundo físico pode ser transferido para manipular o mundo virtual [3].

Para finalizar, devemos analisar a RV sobre três pontos de vista:

- *Tecnológico* - Compreende uma nova área da computação envolvendo outras áreas, como Engenharia de Software, Inteligência Artificial, Arquitetura de Computadores, etc.;

- *Social* - Representa, sobretudo, uma interface mais natural, intuitiva e simples, o que possibilita o aumento do uso do computador na sociedade;
- *Cultural* - Com a crescente utilização do computador como meio de comunicação, surge o que alguns autores chamam de cultura cibernética. A RV surge com forte impacto nessa cultura, principalmente por representar uma forma diferente de relacionamento entre os usuários.

## 2.2 Trabalho Cooperativo Suportado por Computador

Computer Supported Cooperative Work (CSCW) é uma área de pesquisa que utiliza-se de tecnologias da computação para descobrir caminhos para aperfeiçoamento do trabalho em grupo. Uma aplicação baseada em CSCW consiste em um conjunto de ferramentas que compõem uma interface comum (compartilhada) entre vários usuários, fornecendo assim um suporte para troca de informações. Tais aplicações são geralmente classificadas como aplicações de Groupware. Entre as áreas da computação envolvidas, inclui-se Interação Homem-Computador, Redes de Computadores, Multimídia, Engenharia de Software, Inteligência Artificial e, mais recentemente, RV.

Existem dois fatores principais que delimitam o domínio do CSCW [4]:

- *Tempo*
  - Síncrono (Tempo Real): Quando a comunicação entre os participantes ocorre ao mesmo tempo. Exemplo: Vídeo e áudio conferência, chat, entre outros;
  - Assíncrono: A comunicação ocorre em tempos diferentes. Exemplo: e-mail;
- *Espaço*
  - Mesmo local: Os usuários se reúnem em uma mesma sala;
  - Locais diferentes: Os participantes da reunião estão distantes geograficamente.

A seguir apresenta-se alguns conceitos e tecnologias que auxiliam aplicações CSCW atingir seu objetivo [5]:

*WYSIWIS*: abreviação da expressão 'What You See Is What I See' ou 'o que você vê é o que eu vejo'. Representa uma abstração da técnica de replicação da interface do sistema entre seus usuários. Essa abordagem é criticada por dificultar a interação, pois os usuários são impedidos de configurar e controlar as interfaces;

*Multimedia*: A comunicação entre os participantes pode ser enriquecida com a utilização de vídeo, imagens gráficas e som;

*Integração de diferentes ferramentas*: Integrando-se componentes com diferentes funções, diminui-se o conflito na utilização de componentes entre diferentes aplicações;

*Time Management*: Aplicações de groupware que fornecem agendamento de projetos e calendário diário do grupo ajudam os usuários a melhor administrar o tempo das tarefas, aumentando a produtividade.

## 2.3 Realidade Virtual e CSCW

Entre as dificuldades encontradas na elaboração de sistemas cooperativos convencionais, talvez o mais desafiador é o desenvolvimento de interfaces que suportem o relacionamento entre seus usuários. Atualmente, as tecnologias utilizadas para tal relacionamento variam desde comunicação assíncrona, tais como correio eletrônico e listas de discussões, até sistemas mais sofisticados que utilizam comunicação sincronizada, envolvendo áudio e vídeo conferência, chats textuais (MUD/MOO, IRC), além de recursos de multimídia principalmente através da World Wide Web (WWW). Apesar

dos bons resultados já obtidos, o principal obstáculo é fornecer interatividade a estes sistemas. O uso de ambientes virtuais multi-usuários em sistemas cooperativos permite aos participantes interagirem com o mais alto grau de naturalidade, pois através do ambiente é possível a comunicação por meio de imagem, vídeo, texto e áudio.

Utilizando-se então a teoria de ambientes virtuais multi-usuários, podem ser criados sistemas cooperativos através da modelagem de um ambiente gráfico tridimensional representando o ambiente que será compartilhado entre os usuários, como uma sala de reuniões por exemplo, onde seus participantes relacionam por intermédio de seus avatares.

Ambientes virtuais oferecem características peculiares quando utilizados como interface em sistemas cooperativos, como:

- A habilidade de apresentar grande quantidade de informação;
- Visualização natural da informação;
- Suporte para várias modalidades sensoriais;
- Interação natural entre os usuários;
- Facilidade para observar as atividades dos outros cooperadores;
- As diferentes ferramentas que compõem o sistema podem ser representadas no ambiente virtual e usadas concorrentemente ou alternadamente pelos participantes.

Dentre as aplicações nesse campo de ambientes virtuais multi-usuários, temos:

*Teleconferência:* A utilização de interfaces de RV prove uma das principais deficiências dos sistemas de teleconferência convencionais, que é a falta de noções de espaço. Um ambiente virtual pode oferecer:

- Suporte a espacialização do som (som 3D);
- Posicionamento da visão, como direcionamento da atenção;
- Presença e atividade espacial dos participantes.

*Educação à Distância:* Associando-se o alto grau de interação e o poder de visualização da RV com a potencialidade de comunicação proporcionada pela Internet, é possível o desenvolvimento de sistemas cooperativos de educação à distância altamente interativos, onde professores e alunos relacionam-se remotamente por intermédio de ambientes virtuais explorando, construindo e alterando esses ambientes.

*Editoração e Projeto Compartilhados:* Aplicações que fornecem suporte para que usuários remotos trabalhem cooperativamente na criação de um produto, o qual pode ser representado como um objeto dentro do ambiente. Como exemplo, podemos ter vários engenheiros desenvolvendo a planta de uma construção.

## 2.4 Interação Remota em Ambientes Virtuais

Para definir o modo como os usuários se relacionam através do ambiente virtual multi-usuário, devemos analisá-los sob três pontos de vista:

- *Cognitivo:* Do ponto de vista cognitivo, o projetista deve observar como os usuários pensam, aprendem e quais suas habilidades para aquisição de conhecimento;
- *Físico:* Analisa-se aqui como os usuários são 'feitos' (altura, idade, sexo) e quais habilidades sensoriais e motoras eles possuem. A partir dessa análise podem ser definidos os dispositivos e sensores convenientes ao sistema;

- *Social*: Observar o relacionamento do usuário em seu meio social, incluindo estrutura e dinâmica de grupo, poder, políticas, e quais habilidades sociais eles possuem. Com isso cria-se uma hierarquia de controle do ambiente, ou seja, quais os tipos de direitos devem existir e como serão distribuídos entre os usuários;

### 3. Linguagem Java e VRML

#### 3.1 VRML

VRML (*Virtual Reality Modeling Language*) [6] é uma linguagem de modelagem que permite a criação de objetos tridimensionais e a combinação destes objetos para montagem das cenas de um ambiente virtual (mundo virtual). Com VRML podemos criar simulações interativas que incorporam animação, noções de leis físicas, e suporte a vários usuários através de uma rede, em especial através da Internet.

A especificação da primeira versão da linguagem VRML foi apresentada em 1994, na 1ª Conferência Anual sobre a WWW. Desde sua criação, esta linguagem vem se desenvolvendo rapidamente devido à uma série de fatores, entre os quais podemos citar [7]:

- Grande crescimento da WWW nos últimos anos;
- Foi desenvolvida com o objetivo de criar uma linguagem padrão que não somente funcionasse, mas também fosse acessível à qualquer pessoa e não fosse dominada ou controlada por alguma companhia ou pessoa;
- É uma tecnologia avançada o bastante para despertar o interesse e imaginação de desenvolvedores, e ainda é simples o bastante para ser prática e acessível.

Ambientes virtuais modelados em VRML possuem características em comum:

- Proporcionam imersão: O usuário explora o ambiente tridimensional utilizando seus sentidos como se estivesse explorando o mundo real;
- O usuário é que possui controle sobre o ambiente, não o computador: O browser (software que gera o ambiente virtual a partir do arquivo VRML) permite ao usuário definir a rota de exploração do mundo. O ambiente pode oferecer apenas alguma forma de guiar o usuário;
- Um ambiente virtual VRML pode combinar objetos 2D e 3D, animação, efeitos multimídia, etc.;
- Interatividade: Objetos do ambiente podem responder à ações (eventos) de outros objetos, ou provocadas pelo usuário.

A modelagem de um ambiente virtual em VRML a principio não exige softwares especiais, bastando apenas um editor de texto padrão. Porém, na medida em que o ambiente ganha complexidade, com a inserção de animações por exemplo, torna-se necessário uma ferramenta de apoio à modelagem. Existem várias dessas ferramentas no mercado atualmente, as quais oferecem suporte desde a criação dos objetos até a junção destes na montagem do ambiente virtual.

#### 3.2 Linguagem Java

Desenvolvida pela empresa SUN-Microsystems [8], a linguagem Java foi lançada oficialmente em 1994 e vem conquistando cada dia mais espaço entre desenvolvedores de software. Por ser uma linguagem voltada à criação de aplicações para a Internet, seu crescimento é impulsionado pela grande

expansão da rede mundial de computadores. Além disso, existe uma série de características presentes na linguagem que permite ao programador simplicidade no desenvolvimento de aplicações robustas e seguras.

Entre as várias características da linguagem, destaca-se a seguir as que possuem maior influência na sua utilização em sistemas cooperativos:

*Portabilidade:* Java é uma linguagem multiplataforma, ou seja, uma mesma aplicação pode ser executada em diferentes tipos de plataformas sem a necessidade de adaptação de código, o que a torna portátil. Um sistema cooperativo é na maioria das vezes utilizado em uma rede heterogênea (como a Internet por exemplo), tornando-se então indispensável uma boa portabilidade;

*Suporte à Comunicação:* Uma das vantagens do Java é fornecer uma boa funcionalidade, ou seja, muitos detalhes de programação são encapsulados em objetos já prontos. Nesse contexto, a linguagem oferece um conjunto de classes para programação em rede, o que agiliza a implementação de sistemas cooperativos. Tais classes são desenvolvidas para suportar tecnologias avançadas de comunicação, como protocolos TCP/IP, HTTP, FTP, além de Sockets, Datagramas e até mesmo Multicast.

Um ponto relevante na escolha da linguagem Java para desenvolvimento de sistemas cooperativos de RV é a possibilidade de sua comunicação com a linguagem VRML. A junção entre essas duas linguagens já é utilizada a algum tempo por desenvolvedores, resultando em alguns sistemas que estão sendo utilizados na Internet como Community Place [9] e Molecular Arts [10].

### 3.3 Integração da Linguagem Java e VRML

Como foi apresentado anteriormente, várias são as características que levam ao uso das linguagens Java e VRML em sistemas de comunicação. Porém a característica comum a elas e que determinaram a sua integração é principalmente pelo fato de terem sido desenvolvidas especialmente para a Internet.

#### Extern Authoring Interface

Muitas vezes é necessário que o ambiente virtual seja alterado a partir de dados processados em uma aplicação executada em conjunto.

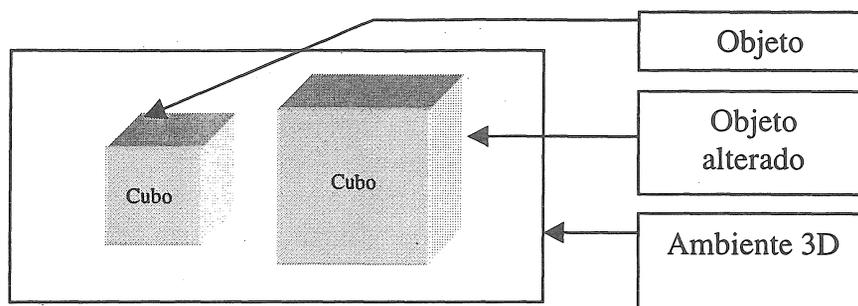
Por exemplo, suponha um ambiente para simulação de lançamento de um projétil. No momento que o usuário entra com as coordenadas do alvo, o ângulo de lançamento deve ser calculado. Surge aqui o problema: Onde fazer esse cálculo, já que a linguagem VRML não possui operadores? Dessa maneira, há a necessidade de uma aplicação que execute esse cálculo e envie o resultado ao ambiente. Para que essa comunicação entre a aplicação e o ambiente ocorra, necessita-se de uma interface externa entre eles.

A *External Authoring Interface* (EAI) [11] é uma interface externa que possibilita a comunicação entre uma aplicação Java e um ambiente virtual VRML.

#### Funcionamento da Extern Authoring Interface

EAI é constituída por um conjunto de classes Java (*package*) que pode ser utilizado tanto em um *Applet* como em um *Script* dessa linguagem.

A base de seu funcionamento é simples. Inicialmente obtém-se uma referência a um dado nó (objeto) do ambiente. A partir dessa referência, o programador tem acesso aos campos (atributos) desse nó, podendo assim alterar dinamicamente suas propriedades, conforme a Figura 1 e Figura 2.



Figuras 1 – Alteração Dinâmica do Objeto

```
// parte do algoritmo que duplica o tamanho do objeto 'Cubo'
:
cubo = obterNo("Cubo");
dimensoesCubo = cubo.obterCampo("dimensoes");
dimensao_atual = dimensoesCubo.ObterValor();
nova_dimensao = 2 * dimensao_atual;
dimensoesCubo.estabelecerValor(nova_dimensao);
.
//obs. palavras em negrito correspondem a métodos implementados na EAI
```

Figura 2 – Representação do Algoritmo da Propriedades da EAI

Como citado anteriormente, podemos utilizar a EAI em um *Applet* ou em um *Script Java*. O primeiro caso é utilizado quando há necessidade de uma interface com o usuário para a entrada de dados. No exemplo do nosso lançador de projétil, poderíamos desenvolver a interface conforme a Figura 3.

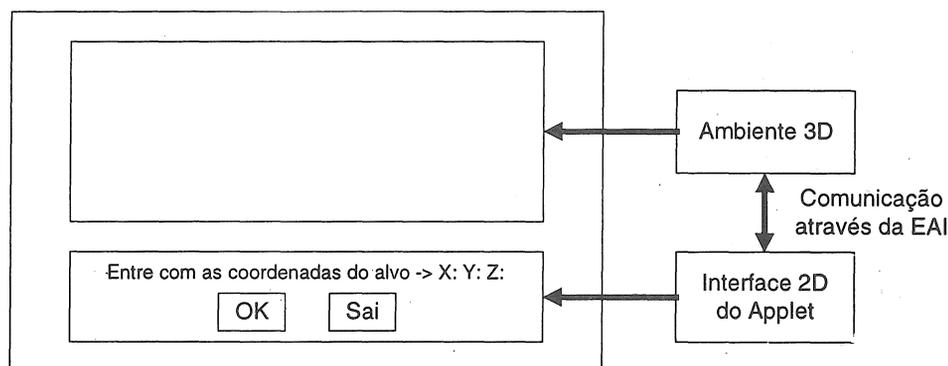


Figura 3 – Interface Applet com Ambiente 3D

*Scripts* são pequenos aplicativos que tem o objetivo único de auxiliar a linguagem VRML no processamento dos dados, permanecendo praticamente oculto ao usuário durante a exploração do ambiente. Utilizando ainda nosso exemplo do lançador, poderíamos ter um outro tipo de interface composta somente do ambiente 3D, onde o usuário escolhe o alvo apontando diretamente com o mouse. Nesse instante suas coordenadas são enviadas ao *script*, que calcula o ângulo de lançamento e retorna esse resultado ao ambiente, tudo transparente para o usuário.

## 4. Sistema Cooperativo de Realidade Virtual

### 4.1 Descrição

O sistema descreve uma sala que permite múltiplos usuários trocarem informações, propriedades de objetos tridimensionais uns com os outros.

Na Figura 4, observa-se a presença de vários usuários na sala, onde os usuários poderão discutir, manipular e/ou controlar vários objetos do ambiente virtual. Cada usuário é representado por um *avatar*, que tem a liberdade de movimentos dentro do ambiente virtual. O usuário poderá utilizar um capacete de RV para a visualização da sala, ou simplesmente um monitor, onde sua movimentação é realizada por funções do teclado ou mouse. O usuário pode movimentar seu ponto de visualização de acordo com os movimentos normais de sua cabeça (rotação em torno dos três eixos espaciais) e não está restrito quanto ao seu posicionamento: ele poderá levantar-se da cadeira e movimentar-se livremente pela sala (translação na sala).

A comunicação entre os usuários podem ser de duas maneiras: através de uma janela de "chat" ou pela transmissão de voz em tempo real. A janela de "chat" foi implementada para facilitar a transmissão de dados para os demais participantes, tais como fórmulas, símbolos, valores, etc. Para a discussão normal entre os participantes, processos de comunicação via áudio estão sendo desenvolvidas para serem acoplados ao ambiente virtual.

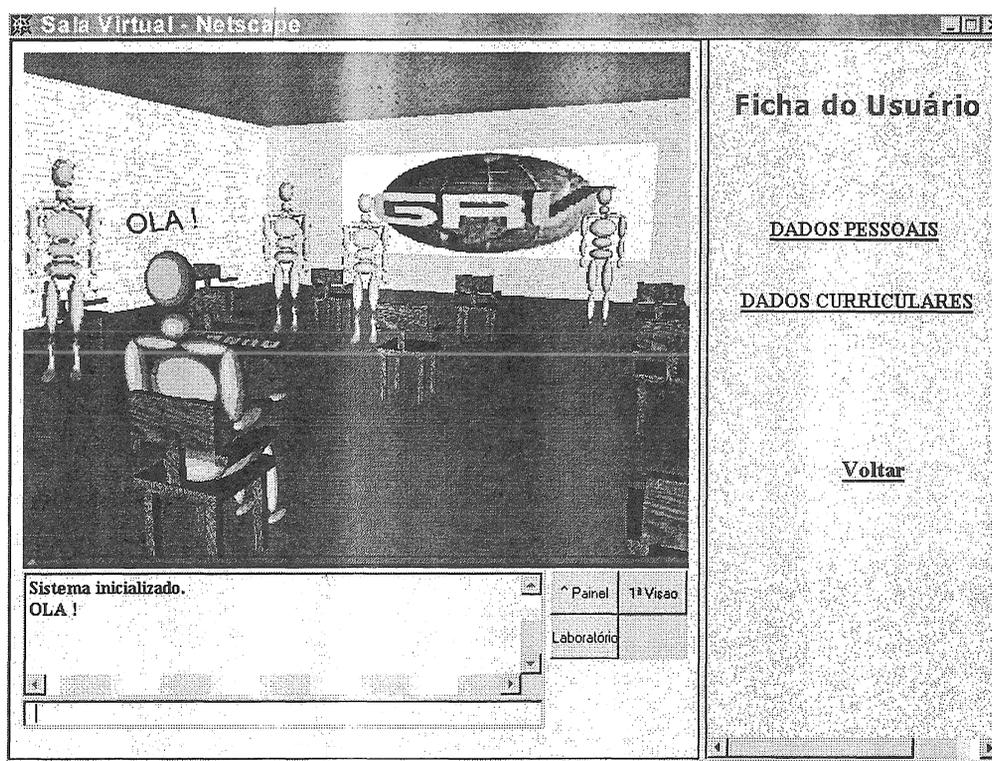


Figura 4 – Apresentação do Sistema de Realidade Virtual Cooperativo

No processo de desenvolvimento do sistema, utilizou-se a metodologia orientada à objetos Fusion [12] para especificar as interfaces de comunicação entre os usuários. A Figura 5 mostra através do modelo de objetos, o relacionamento das seguintes Classes: Usuário, Textual, Gráfica, Módulo de Comunicação e Meio Físico de Comunicação.

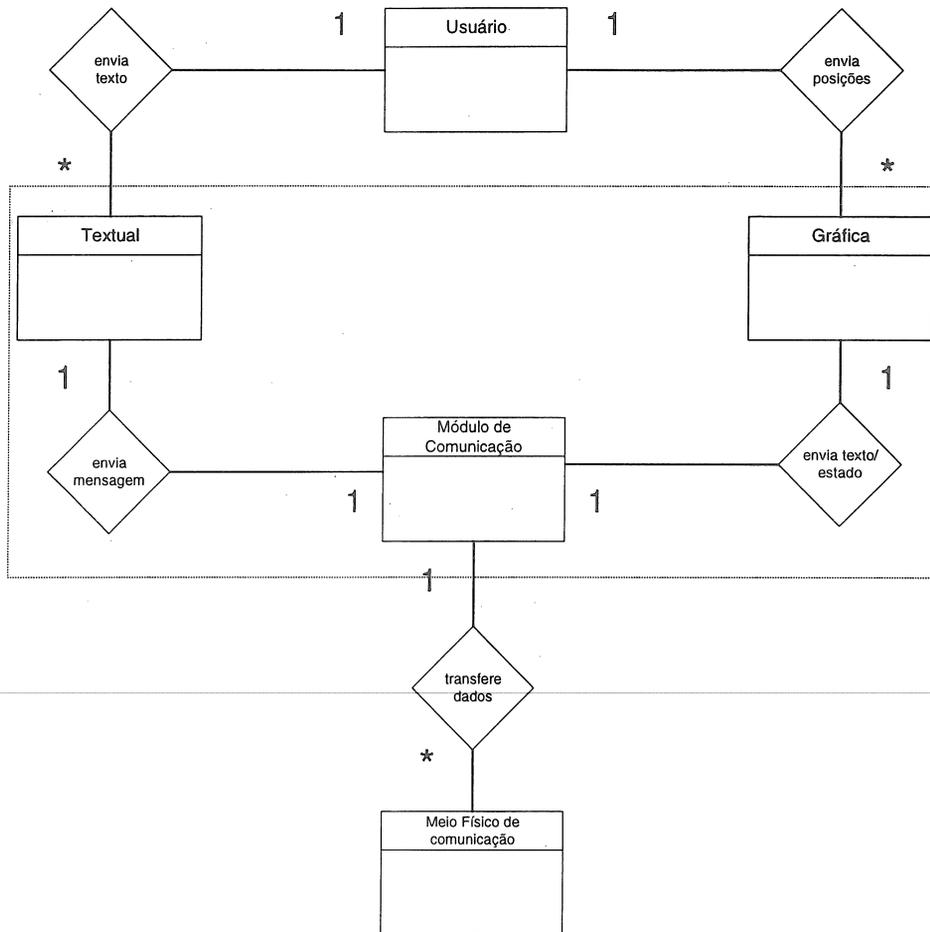


Figura 5 – Modelo de Objetos da Interface de Comunicação entre os Usuários

#### 4.2 Infra-estrutura do Sistema

A infra-estrutura do sistema foi desenvolvida a partir das linguagens Java e VRML. O sistema de RV possui uma interface tridimensional, que consiste dos seguintes módulos (Figura 6):

*Barra de Ferramentas:* constituída por um conjunto de botões através dos quais o usuário poderá controlar seu avatar, tendo opções de expressar ações, tais como: acenar, dançar, ficar feliz ou descontente, entre outros. Esse componente possuirá também outros botões para alterar a visão do usuário, entre primeira ou terceira pessoa, dentro do ambiente. Em primeira pessoa, o usuário terá a mesma visão de seu avatar, ou seja, como se o usuário fosse o avatar; já em terceira pessoa, o usuário terá uma visão externa de seu avatar, podendo então observá-lo à distância;

*Visualizador Gráfico:* consiste no componente onde o ambiente virtual será exibido. Esse ambiente é formado por avatares (objetos tridimensionais que representam os usuários e refletem suas ações dentro do ambiente) e por outros objetos tridimensionais estáticos (árvores, casas, ruas e outros).

*Visualizador Textual:* nele serão exibidas as mensagens enviadas e as recebidas pelo usuário, ou seja, o texto do diálogo;

*Visualizador Textual:* consiste no componente em que as todas as mensagens do ambiente virtual serão visualizadas, ou seja um texto de dialogo;

*Entrada de Texto*: componente que permite ao usuário entrar com a mensagem, a qual será enviada para o usuário remoto no momento em que a tecla ENTER for pressionada. Ao mesmo tempo esta mensagem será também enviada ao Visualizador Gráfico (mais especificamente, ao balão de diálogo do avatar local).

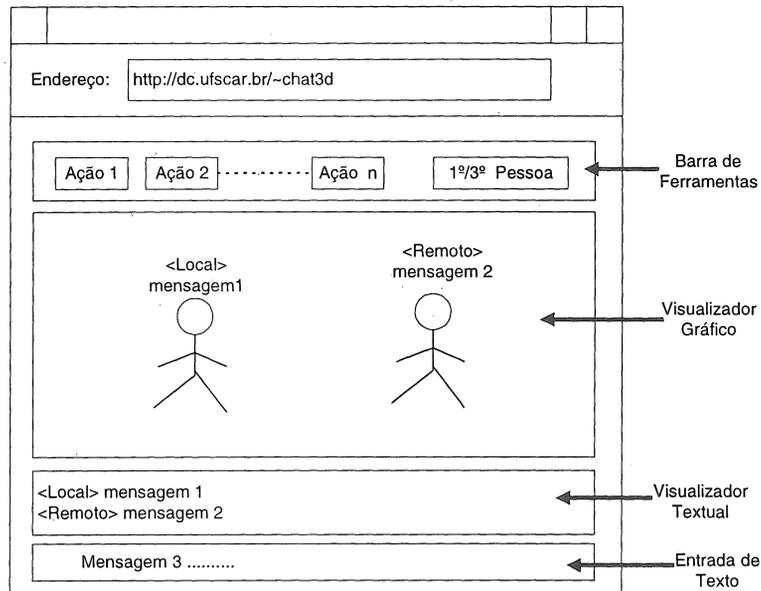


Figura 6 - Desenho Esquemático da Interface do Sistema

### 4.3 Arquitetura de Comunicação

A interface de programação do sistema utiliza o pacote `java.net`. O pacote `java.net` possui classes que são divididas em duas categorias: interfaces de programação baseada em sockets e classes para manipulação de URLs. A comunicação realizada no sistema ficou restrita à comunicação baseada em sockets (classes `Socket` e `DatagramSocket`), devido ao fato de possuírem interfaces de programação de baixo nível para comunicação em rede, em que envio de dados entre as aplicações ocorrem através de streams.

Sockets são difíceis de serem utilizados pelo programador, principalmente por envolverem a manipulação de diferentes protocolos de transporte de dados. Porém, a linguagem Java oferece uma interface simples para utilização de sockets, utilizando-se princípios de orientação à objetos para encapsular toda sua complexidade.

Existem duas classes Java para programação com sockets: `Socket` e `DatagramSocket`. Essas classes trabalham sobre os dois tipos de protocolos de transporte mais utilizados na Internet, o TCP e o UDP, respectivamente.

O sistema utiliza como modelo de comunicação Cliente/Servidor (Figura 7). A idéia é simples: de um lado temos o sistema do usuário funcionando como o cliente, a qual faz requisições de serviços à uma outra aplicação denominada servidor. Normalmente temos um único servidor atendendo à vários clientes.

O usuário inicializa o browser do sistema e envia um pedido de abertura de conexão a um servidor que executa em determinado host. O pedido é respondido, somente se a conexão foi estabelecida e o cliente faz a requisição da informação (posição, orientação, mensagem do chat, etc.). O servidor responde ao pedido enviando o documento.

## A Classe Socket

O uso dessa classe implica na implementação do modelo cliente servidor, como esquematizado na Figura 7.

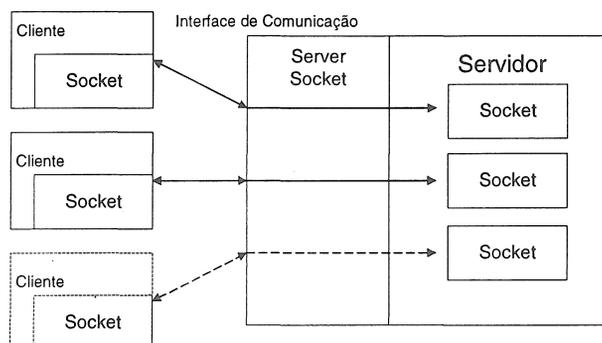


Figura 7 – Modelo Cliente/Servidor: Mecanismo de Conexão através de Sockets

- Estabelecimento de Conexão

No esquema acima, observa-se que a classe `java.net.Socket` aparece tanto no cliente quanto no servidor, formando as extremidades de um canal de comunicação. O estabelecimento de uma nova conexão, é utilizado pela instância da classe `ServerSocket` no servidor, cuja função é esperar o pedido de abertura de conexão dos novos clientes. Quando uma conexão é aceita, seu método `accept` retorna um objeto `Socket`, o qual representará o canal de comunicação do servidor com esse novo cliente.

A outra função do objeto `ServerSocket` é fornecer uma porta de identificação para o servidor, já que em um mesmo host podem haver vários servidores rodando ao mesmo tempo. No lado do cliente, a criação de uma conexão com o servidor é simples, bastando instanciar um objeto da classe `Socket` como a seguir:

```
Socket sock = new Socket( <"nome_host">, <porta_servidor> );
```

- *Leitura e escrita via Socket*

Após criar a conexão, através dos métodos `getInputStream` e `getOutputStream` obtém-se, respectivamente, as streams de entrada e saída do socket. Tais streams são do tipo `InputStream` e `OutputStream`, que podem então ser utilizados para instanciação de objetos de subclasses `Stream`;

- *Servidor Multi-usuário*

O servidor é capaz de atender a um único cliente por vez. Porém existem casos onde vários clientes necessitem acessar o servidor simultaneamente. Nesse caso devemos ter vários processos sendo executados simultaneamente no servidor.

A maneira mais comum de resolver esse problema é a implementação de servidores com threads. No momento que um cliente estabelece conexão com o servidor, um novo thread é criado para atender esse cliente.

## 5. Conclusão

A RV pode ser aplicada nas mais variadas áreas do conhecimento e de maneira bastante diversificada. A todo momento surgem novas aplicações, em função da demanda e da capacidade criativa das pessoas. Em muitos casos, a RV vem revolucionando a forma de interação das pessoas com sistemas complexos tratados com o uso de computadores, propiciando maior desempenho e economizando custos.

Neste contexto, este artigo apresentou o desenvolvimento de um sistema de RV cooperativo capaz de suprir as necessidades dos usuários da Internet. O sistema de RV cooperativo propõe

mecanismos de interface e interação multi-usuários, através de gráficos tridimensionais que representam o mundo real.

Para a implementação do sistema de RV cooperativo foram utilizados as linguagens Java e VRML. Java possui diversas características Orientadas a Objetos, que permite habilitar a comunicação de rede do sistema de RV cooperativa. VRML por sua vez, permite criar interfaces tridimensionais. A integração das linguagens Java e VRML proporciona a construção de sistemas na Internet.

Este trabalho representa o trabalho contínuo do Projeto AVVIC do Grupo de Realidade Virtual (GRV) do Departamento de Computação da Universidade Federal de São Carlos (DC-UFSCar).

## 6. Agradecimentos

Este trabalho foi apoiado pelo CNPq - PROTEM-CC-AVVIC (Nº 680063/95-1) e pela Fapesp (processo 95/6560-8). Verison José Vincentin é apoiado pela Fapesp (processo 98/05542-4), Mario Massakuni Kubo é apoiado pela Fapesp (processo 98/01940-5) e Wagner José Dizero é apoiado pela Capes.

## 7. Referências Bibliográficas

- [1] Kalawsky, R.S., "Exploring Virtual Reality Techniques in Education and Training: Technological Issues", Advanced VR Research Centre, Lough-borough University of Technology, 1997.
- [2] Kirner, C., Pinho, M., "Introdução à Realidade Virtual", Minicurso JAI/SBC, Recife, PE, 1996.
- [3] Kirner, C., Araújo, R. B., Ipolito, J., "Sistemas de Realidade Virtual – Aspectos, Distribuição e Programação na Internet ", UFSCar, 1996.
- [4] Borges, M. R. S., Cavalcanti, M. C. R., Campos, M. L. M., "Suporte por Computador ao Trabalho Cooperativo", XIV Jornada de Atualização em Informática, Canela-RS, Brasil, 1995.
- [5] Grudin, J., "Computer-Supported Cooperative Work: History and Focus", IEEE Computer, May, 1994.
- [6] Matsuda, K., Miyashita, K. Lea, R., "Java for 3D and VRML World", New Riders Publishing; 1996.
- [7] Hartman, J., Wernecke J., "The VRML 2.0 Handbook", Addison-Wesley; 1996.
- [8] Sun Microsystems, URL:<http://www.javasoft.com/javastore/index.html>, 1998.
- [9] Sony Corporation, Community Place, URL:<http://vs.spiw.com.vs/>, 1998.
- [10] Molecular Arts Corporation, URL:<http://www.molecules.com/>, 1998.
- [11] Marrin, C., "External Authoring Interface Reference", Silicon Graphics, Inc., Jan 21, 1997.
- [12] Coleman D. et al., "Object Oriented Development: The Fusion Method", Prentice Hall, 1994.