

## ARQUITETURA DE UM AMBIENTE VIRTUAL DISTRIBUÍDO PARA APLICAÇÕES DE LARGA ESCALA

**REGINA BORGES DE ARAUJO<sup>1</sup>**

Grupo de Sist. Distr. e Redes - GSDR  
Departamento de Computação  
Universidade Federal de São Carlos  
Cx. Postal 676  
13565-905 São Carlos - SP  
[drea@power.ufscar.br](mailto:drea@power.ufscar.br)

**DR. CLÁUDIO KIRNER<sup>2</sup>**

Departamento de Computação  
Universidade Federal de São Carlos  
Cx. Postal 676  
13565-905 São Carlos - SP  
[ckirner@blanca.uccs.edu](mailto:ckirner@blanca.uccs.edu)

### RESUMO

Este artigo caracteriza sistemas de ambiente virtual distribuído de larga escala, propondo uma arquitetura distribuída para o suporte às aplicações que envolvem a participação de milhares de usuários compartilhando um ambiente sintético que pode variar de poucos a centenas de quilômetros. É discutido também, um modelo de comunicação entre os objetos que populam este ambiente virtual.

### ABSTRACT

This paper characterizes large distributed virtual environment system, proposing an architecture to support applications involving the participation of thousands of users who share a synthetic environment ranging from a few to thousands of miles. It is also discussed a communication model among the objects that populate a virtual environment.

### 1. INTRODUÇÃO

A Realidade Virtual tem surgido como algo inovador na interação homem-máquina e promete uma revolução em áreas que vão do entretenimento à educação e medicina. A Realidade Virtual, também referenciada na literatura como realidade artificial, *cyberspace*, mundo virtual e ambiente virtual, é definida aqui como uma forma mais natural e poderosa de comunicação entre um usuário e o seu computador. O termo ambiente virtual é o adotado pelos autores para definir uma interface que simula um ambiente de tal forma que o usuário se sinta imerso neste ambiente e interaja com o mesmo de forma natural e direta.

---

<sup>1</sup> Doutoranda do LSI - EPUSP (bolsista PICD-CAPES).

<sup>2</sup> Professor Visitante do Departamento de Computação, Universidade do Colorado em Colorado Springs - UCCS, Colorado Springs, CO, USA, com bolsa do CNPq, Processo 201443/93-7.

Vários sistemas de ambiente virtual - AV têm sido construídos e avaliados para aplicações que suportam tipicamente um pequeno número de usuários, como as aplicações de trabalho cooperativo suportado por computador. Este artigo trata de uma classe especial de aplicações: Sistemas de Ambientes Virtuais Multi-usuários Distribuídos de Larga Escala. Estes sistemas são caracterizados pela participação de milhares de usuários em ambientes com extensas áreas geográficas como um campo de batalha para aplicações de treinamento militar, uma cidade virtual para fins educacionais ou de entretenimento, jogos do tipo guerra nas estrelas etc. Estas características tornam um sistema de AV difícil de ser representado em cada nó do sistema, tornando-o inerentemente distribuído. Este artigo propõe a arquitetura de um sistema de AV de larga escala, discutindo um modelo de comunicação entre objetos para a arquitetura proposta.

A seção 2 mostra características comuns de AVs de larga escala. A seção 3 descreve o modelo básico de um sistema de AV, apresentando os objetos que populam um mundo virtual e propondo uma arquitetura distribuída para um sistema de AV de larga escala. As questões de comunicação em sistemas de AV existentes são discutidas na seção 4, seguida da apresentação de um modelo de comunicação entre objetos, para a arquitetura proposta, na seção 5. O estado atual de um projeto sendo desenvolvido para validar a arquitetura proposta é descrito na seção 6.

## 2. CARACTERIZAÇÃO DE UM AMBIENTE VIRTUAL DE LARGA ESCALA

Sheperd [SHEP93] classifica os sistemas de AV em duas grandes classes: *telepresença*, onde um ambiente sintético comum é compartilhado entre várias pessoas, como uma extensão ao conceito de trabalho cooperativo suportado por computador e *teleoperação*, onde robôs agem sobre um elemento, seja ele um corpo humano, um produto sendo manufaturado etc. Nos sistemas de telepresença, a manutenção da ilusão de que o usuário está presente no ambiente sintético é essencial. Para que esta ilusão seja mantida, o sistema tem que oferecer: tempo de resposta dentro de um limiar aceitável; as ações dos participantes têm que ser refletidas de forma imediata e simultânea a todos os outros participantes que compartilham o mesmo ambiente, e os ambientes têm que ser realistas.

Isto implica nas seguintes demandas feitas sobre um sistema de AV, conforme sumarizado por Van Dam [VanD93]: taxas rápidas de atualização e atraso mínimo; tratamento de múltiplos dispositivos de entrada e simulação de um número grande de objetos com comportamento complexo.

Estes requisitos demandam não apenas ferramentas poderosas para se construir mundos virtuais e suas representações gráficas, mas também processadores gráficos de alto desempenho, dispositivos de entrada e saída que sejam precisos, confortáveis e naturais, novas arquiteturas de sistemas distribuídos, sistemas operacionais adequados, redes de comunicação que possam garantir a entrega da informação em tempo, entre outros.

Os requisitos acima são ainda mais rigorosos quando consideramos um grupo especial de aplicações dentro da classe de telepresença: *ambientes virtuais distribuídos multi-usuários de larga escala*. Estas aplicações apresentam as seguintes características: existência de um

grande número de participantes e de objetos simulados; extensas áreas geográficas que vão de poucos a milhares de quilômetros; os participantes são geograficamente distribuídos; os ambientes são multi-dimensionais e a interação entre objetos e participantes é feita em tempo real. Além disto, para que um grande número de participantes possa interagir com o AV, é interessante que o sistema suporte recursos computacionais e de rede heterogêneos.

Poucos sistemas de AV em larga escala foram implementados. Os mais conhecidos são SIMNET [CALV93] and NPSNET [ZYDA92] que continuam a ser desenvolvidos objetivando uma escala de milhares de participantes. A construção destes tipos de sistemas é bastante complexa, demandando a integração de várias áreas tecnológicas. Este artigo trata da proposição de uma arquitetura distribuída para o suporte de AVs de larga escala. Uma das aplicações almeçadas é a de uma cidade virtual onde um grande número de participantes possa compartilhar um mesmo ambiente com fins educacionais, de turismo, lazer etc.

### **3. ARQUITETURA DE UM AMBIENTE VIRTUAL DISTRIBUÍDO PARA APLICAÇÕES DE LARGA ESCALA**

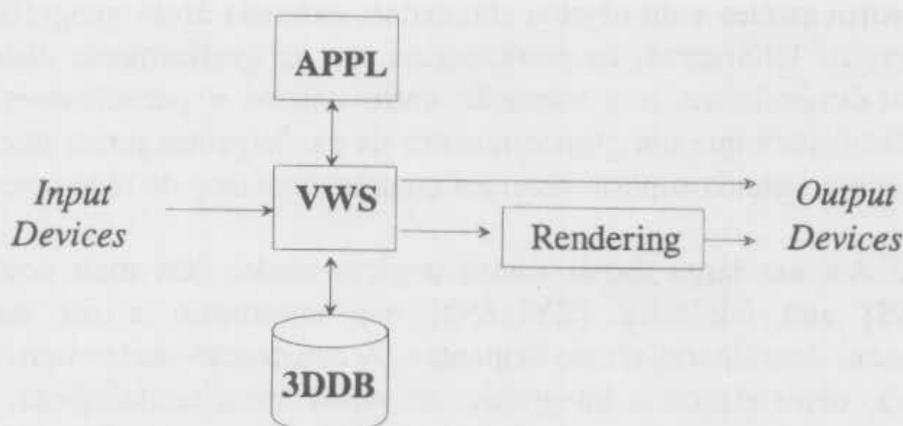
#### **3.1 A POPULAÇÃO DO MUNDO**

O mundo virtual é habitado por objetos que são classificados aqui como estáticos, dinâmicos dirigidos pelo usuário e dinâmicos dirigidos por simulação. Grande parte da população do mundo é composta de objetos estáticos, ou seja, objetos que são fixos como prédios, montanhas, árvores, parques, mobília etc. Objetos dirigidos por simulação vão desde aqueles programados a agirem de acordo com padrões pré-definidos (*scripts*) até objetos altamente complexos dirigidos por sistemas especialistas, como humanóides, robôs etc. Estes objetos mais complexos podem ser desenvolvidos por diferentes empresas e plugados no ambiente virtual. As ações dos usuários são refletidas no AV através dos objetos dirigidos pelo usuário. Estes podem ser representados como elementos abstratos (formas geométricas), partes do corpo humano (mãos, pés etc.), veículos móveis, luvas etc.

#### **3.2 O MODELO BÁSICO**

O modelo básico de um sistema de AV compreende os seguintes módulos: aplicação do usuário (APPL), simulador do mundo virtual (VWS), base de dados de objetos 3D (3DDB), *rendering* em 3D (visual, acústico, tátil etc.), e dispositivos de E/S, conforme mostra a figura 1.

O módulo APPL é definido pelo projetista da aplicação que decide sobre as características dos objetos, o comportamento dos objetos dirigidos a simulação, como os objetos interagem, se o ambiente sintético é para ser compartilhado entre os usuários ou se cada usuário vai ter sua própria visão do AV [ISDA93] etc. Um usuário pode adaptar a aplicação a seu gosto, através da adição dinâmica de novos objetos, ou da alteração das características e/ou do comportamento de objetos existentes.



*Fig. 1 O Modelo de um Sistema de AV [ARAU95a]*

A simulação (VWS) suporta a aplicação e pode ser feita em dois níveis: simulação de ambiente e simulação de objeto [ARAU95a]. A simulação de ambiente roda sem parar e é responsável pelo posicionamento de todos os objetos na área sob seu controle bem como detecção de colisão, translação de objetos etc. A simulação de objeto é responsável pelo controle dos detalhes de animação de cada objeto dinâmico. Para se diminuir a comunicação entre os dois níveis de simulação descritos acima, o conceito de simulação inercial é utilizado, onde todos os objetos dinâmicos seguem leis físicas ou de movimentos previsíveis, como o conceito de *dead-reckoning* usado no SIMNet [CALV93]. Neste caso, o simulador de ambiente não precisa enviar informações de reposicionamento de objetos a menos que um comportamento imprevisível ocorra por parte de um objeto. Em um carro dirigido pelo usuário, por exemplo, o usuário pode manter a velocidade e orientação, de forma que o reposicionamento do carro não precisa ser refletido para suas réplicas<sup>3</sup> a menos que o usuário altere quaisquer destes parâmetros. Para aquelas aplicações que não seguem as leis da física, como alguns jogos de entretenimento por exemplo, vai haver um aumento no tráfego já que cada ação efetuada pelos objetos dinâmicos terá que ser comunicada para suas réplicas.

As ações dos usuários são capturadas por sensores ligados a eles (luvas, interceptadores de movimentos de cabeça, corpo etc.) ou dispositivos apontados para eles (câmeras). O VWS interpreta estas informações transformando-as em comandos gráficos e outros que são enviados aos *renderers* 3D para o processo de reposicionamento. O *renderer* gráfico gera a imagem final a ser apresentada ao usuário e o *renderer* de áudio reproduz o áudio que coincide com a posição dos objetos gráficos. As ações dos usuários, lidas de sensores, podem engatilhar ações que são refletidas sobre outros usuários através, por exemplo, de atuadores. Estas ações dos usuários são refletidas no AV através dos objetos dirigidos a usuários.

### 3.3 PROPOSIÇÃO DE UM MODELO DISTRIBUÍDO

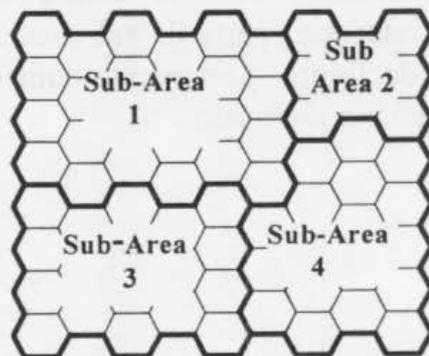
Alguns dos sistemas multi-usuários de AV mais conhecidos, como SIMNET, NPSNET [MACE94a] e DIVE [CARL93], mantêm uma base de dados homogênea distribuída do mundo virtual. Sistemas de menor escala são baseados em bases de dados centralizadas,

<sup>3</sup> Cópias do objeto distribuídas pelos hosts participantes.

outros em bases de dados particionadas que são acessadas através de servidores replicados [SING94].

Em um sistema onde milhares de usuários podem tomar parte na aplicação, e onde extensas áreas estão envolvidas, a distribuição é inerente. A informação geométrica de uma área muito extensa, como uma cidade, pode precisar ser dividida em sub-áreas que são distribuídas pela rede sob demanda. Em uma cidade virtual que sintetize, por exemplo, a cidade de Mônaco, que apesar de pequena em extensão, é bastante densa, ambientes multi-dimensionais como o Cassino de Monte Carlo, o Palácio Real, o percurso da cidade, provavelmente terão que ser distribuídos em uma ou mais bases de dados. O conjunto destas bases de dados formam o conjunto de dados do mundo.

O mundo é dividido em células de hexágonos [MACE94b]. Uma ou mais células compõem uma sub-área controlada por um simulador principal. A figura 2 mostra como as células são associadas a sub-áreas.



*Fig.2 Células associadas a sub-áreas em um AV*

Um servidor de ambiente é responsável pela simulação e armazenamento de uma sub-área e seu respectivo conjunto de dados. Como um AV pode compreender várias sub-áreas, todo o AV pode ser distribuído em vários servidores de ambiente. Para efeito de visualização, as células são armazenadas nos servidores de ambiente em vários graus de degradação.

Para se diminuir a complexidade da simulação de todos os elementos envolvidos na aplicação, o simulador principal é estruturado como um conjunto de simuladores responsáveis por uma célula em particular ou um conjunto de células e seus objetos. Organizado desta maneira, um carregamento parcial da sub-área pode ser realizado numa máquina cliente. A máquina cliente vai conter simuladores responsáveis pela simulação inercial dos objetos dinâmicos, bem como o controle de objetos dirigidos a simulação, menos complexos, como aqueles contendo ações descritas na forma de *scripts*.

Quando um usuário seleciona uma área de interesse, esta área é carregada de uma sub-área em um servidor de ambiente para uma máquina cliente, que pode ter um ou mais simuladores simples, responsáveis apenas pela área de interesse do usuário. O carregamento de uma área de interesse do usuário vai depender de pelo menos dois fatores: o volume de visão do usuário e a velocidade dos objetos dirigidos a usuário. O volume de visão é aquela parte do mundo virtual visível ao usuário a partir de seu ponto de vista atual. A posição e orientação do usuário são os fatores mais importantes na determinação do volume

de visão [PRAT93]. Entretanto, pode não ser suficiente carregar apenas as células que compõem o volume de visão. Se o usuário vira sua cabeça, é interessante ter as células ao seu redor já carregadas na máquina cliente, de forma que o novo volume de visão possa ser calculado.

A previsão de carregamento de novas células é uma questão extremamente importante que vai contribuir para a melhoria de desempenho do sistema [WLOK94]. A velocidade dos objetos dirigidos a usuários (carro dirigido pelo usuário, avião pilotado pelo usuário, um corpo andando e sendo controlado pelo usuário etc.) vai ditar o grau de antecipação no carregamento das células. Quanto mais rápido for o objeto dirigido pelo usuário, maior tem que ser a antecipação no carregamento de células. Quando um usuário se posiciona, por exemplo, entre as sub-áreas 1 e 2 de um ambiente virtual, as células que estão dentro do volume de visão do usuário, de ambas as sub-áreas, são carregadas na máquina do usuário nos respectivos graus de degradação.

A figura 3 mostra o sistema distribuído descrito acima onde dois usuários estão compartilhando uma mesma área de interesse, parte da sub-área 3. Esta área de interesse é mostrada em *zoom*, do lado direito da figura, juntamente com o volume de visão de um usuário, representado como um triângulo sombreado.

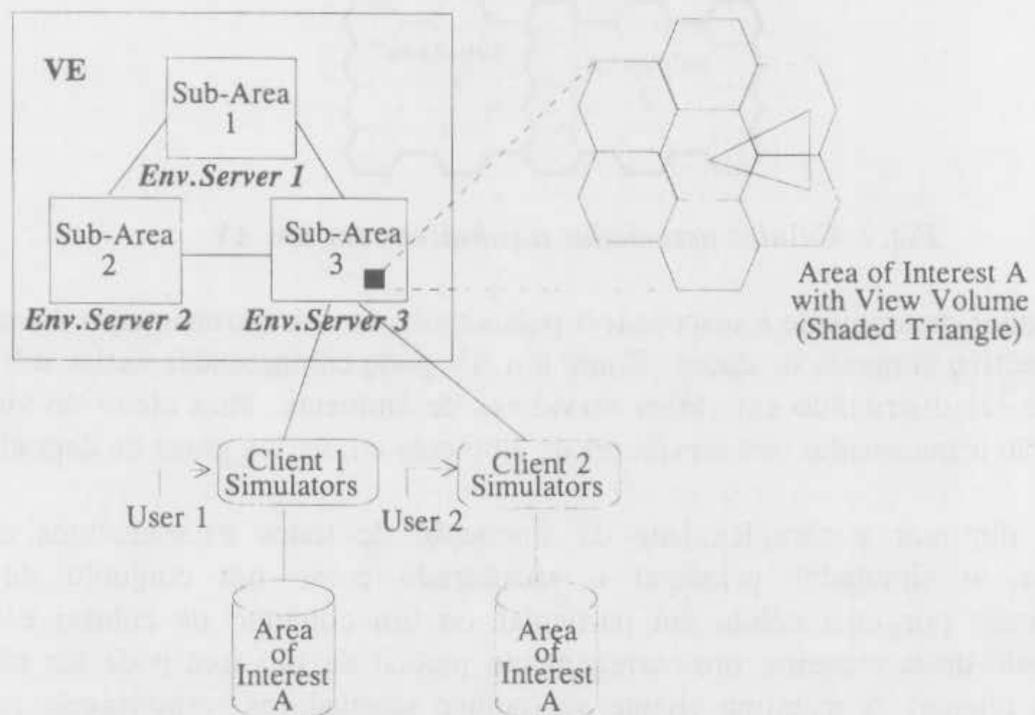


Fig. 3 O Modelo de um Sistema de AV Distribuído [ARAU95a]

#### 4. A COMUNICAÇÃO EM SISTEMAS DE AV EXISTENTES

O sistema DIVE [CARL93] utiliza uma arquitetura de comunicação *peer-to-peer* para a interação entre os objetos e o mundo virtual, através do sistema distribuído ISIS. BrickNet [SING94] é implementado como o modelo cliente-servidor onde os objetos são mantidos em

diferentes clientes e são acessados e atualizados através de servidores que juntos mantêm informação sobre todos os objetos do sistema. No sistema SIMNET [CALV93], cada sistema participante possui uma cópia de todo o mundo virtual. Os tipos de mensagens trocadas entre os participantes são basicamente mensagens de alteração de estado de objetos e ações de cunho militar do tipo detonar míssil, fogo etc. A comunicação aqui é feita por difusão de eventos através do protocolo DIS<sup>4</sup>. As Unidades de Dados deste Protocolo (PDUs), referentes ao estado dos objetos, são transmitidas periodicamente mesmo quando não há qualquer alteração no estado destes objetos. Isto ocorre porque não há informação centralizada sobre o estado dos objetos, levando a uma perda na largura de banda.

O sistema NPSNET mantém a arquitetura de dados do SIMNET, fazendo uso do protocolo DIS mas sobre a rede IP *multicasting* - Mbone [MACE94a]. O NPSNET associa objetos, denominados *entidades*, relacionados de forma espacial, temporal e funcional, a grupos *multicasting* [MACE94b]. Aqui, cada célula que compõe uma área de interesse de um participante é ligada a um grupo *multicasting*. Desta forma, se a área de interesse de uma entidade compreender vinte células, esta entidade vai fazer parte de vinte grupos. Conforme esta entidade se movimenta em direção às bordas de sua área de interesse, novas células vão sendo carregadas e, portanto, a entidade passa a fazer parte de novos grupos *multicasting*, ao mesmo tempo que deixa de participar de células que saíram de sua área de interesse. Esta solução é interessante do ponto de vista de economia de largura de banda, já que utiliza o conceito de *multicasting*, onde uma mensagem apenas é enviada a um grupo de  $n$  destinatários, ao contrário do modelo *unicasting*, onde, para um grupo de  $n$  destinatários, é necessário que  $n$  mensagens sejam enviadas. Entretanto, em áreas contendo objetos de muita mobilidade dirigidos por simulação, como carros em grande velocidade, estes objetos sairiam de um grupo e entrariam em um novo grupo de forma altamente dinâmica. Esta dinamicidade pode gerar inconsistência na manutenção dos membros dos grupos em esquemas de endereçamento *multicasting*. Araujo [ARAU94b] aponta outras dificuldades em relação à implementação do esquema *multicasting*. A próxima seção discute a comunicação entre objetos no sistema aqui sendo proposto.

## 5. A COMUNICAÇÃO NO SISTEMA DISTRIBUÍDO PROPOSTO

Quando a área de interesse de um participante é carregada para uma máquina cliente, réplicas dos objetos dinâmicos vivendo naquela área também são carregados para a máquina cliente. Quando existir mais que um participante para uma mesma área de interesse, réplicas daquela área são carregadas nas máquinas clientes ligadas aos respectivos participantes. Desta forma, existe aqui o conceito de réplicas para dois tipos de elementos: réplicas de uma área de interesse e réplicas de objetos dinâmicos.

Uma ação engatilhada por um participante tem que ter uma resposta visual rápida, além desta ação ter que ser difundida a todas as suas réplicas em outras máquinas clientes contendo a mesma área de interesse. Isto demanda, em termos locais, uma atualização

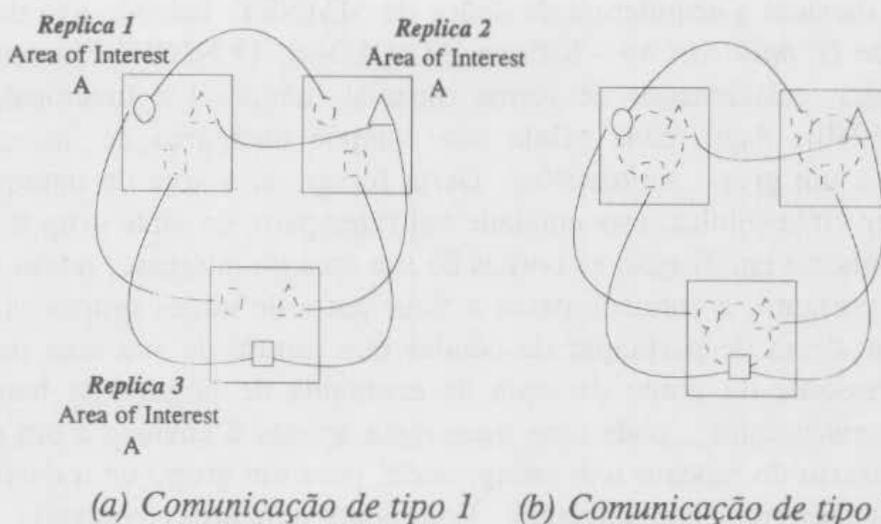
---

<sup>4</sup> DIS - Distributed Interactive Simulation é um grupo de padrões sendo desenvolvido pelo Departamento Americano de Defesa e da Industria, que permite que simulações diferentes, rodando em máquinas diferentes, compartilhem o mesmo ambiente virtual.

gráfica do ambiente virtual que o participante está inserido, e em termos remotos, uma atualização em rede da representação deste participante nas outras máquinas clientes.

A comunicação entre objetos em um sistema de AV envolve basicamente a comunicação entre um objeto e suas réplicas (chamada aqui de comunicação de tipo 1) e entre objetos dinâmicos, sejam estas representações ligadas diretamente a um participante, ou objetos dirigidos por simulação (denominada de comunicação de tipo 2).

A figura 4 abaixo mostra o fluxo de informação nos dois tipos de comunicação descritos:



**Fig. 4 Fluxo de Informação na Comunicação Entre Objetos de um AV [ARAU94a]**

Na figura acima, os quadrados representam três réplicas de uma mesma área de interesse. As formas geométricas sólidas, dentro de cada quadrado, representam ou um objeto dirigido pelo usuário, cujo usuário (participante) reside na máquina cliente que mantém a área de interesse replicada, ou a cópia principal de um objeto dirigido por simulação. As linhas sólidas refletem o fluxo de mensagens de um objeto dinâmico para suas réplicas, que são representadas como formas geométricas pontilhadas. Considere por exemplo, o círculo sólido como um objeto dinâmico e os círculos pontilhados como representações modeladas dinamicamente daquele objeto em outras duas máquinas ([BLAU92] chama o objeto dinâmico de *Player* e suas representações modeladas, de *Ghosts*). Qualquer movimento executado pelo objeto representado como o círculo sólido, que não puder ser previsto pelo código rodando na representação modelada (simulação inercial), mostrada em linhas pontilhadas, é enviado para suas réplicas através de mensagens de reposicionamento.

Para a comunicação entre os objetos dirigidos pelo usuário (comunicação de tipo 2), por exemplo, entre dois participantes compartilhando uma mesma área de interesse, o conceito de auras é adotado. Benford [BENF94] introduz um modelo de interação espacial baseado no conceito de aura, focus, nimbus, e níveis de conscientização de presença entre objetos. Qualquer interação entre objetos ocorre através de algum medium (visual, áudio, textual, tátil etc.), onde a combinação da media a ser usada na comunicação, e os níveis de consciência entre objetos, baseados em suas distâncias relativas, são negociados pelos

objetos. O modelo de comunicação entre objetos adotado pela arquitetura aqui proposta é uma simplificação do modelo desenvolvido por Benford e é descrito a seguir.

A todo objeto dirigido pelo usuário é atribuído uma aura por medium. Uma aura é uma área que envolve um objeto e cujo alcance varia de acordo com a área de alcance do medium que ela representa - a área de alcance da aura de áudio pode ser diferente da área de alcance das auras de vídeo, de cheiro etc. Quando um participante emite um som, por exemplo, uma mensagem é gerada com aquele som como dado, mais um valor referente ao alcance desejado pelo objeto gerador do som. Se o objeto não especificar o alcance, um valor *default* é colocado. Esta mensagem é então enviada ao(s) grupo(s) associado(s) a uma mesma área de interesse, na qual aquele participante está inserido.

Quando esta mensagem chega em cada um dos membros daquele(s) grupo(s), cada máquina cliente calcula os objetos que estão dentro da área de alcance especificada na mensagem. Um filtro é então utilizado para que, dentre os objetos que estão dentro da área de alcance, apenas aquele que representar o participante ligado àquela máquina cliente recebe a mensagem de som. O mesmo acontece em cada máquina cliente membro do(s) grupo(s) associados. Os outros objetos, que porventura estiverem dentro da aura mas são réplicas de outros objetos, ou seja, os participantes ligados a eles não residem naquela máquina cliente, simplesmente não recebem a mensagem filtrada. Já o objeto representando um usuário, recebe a mensagem que é então reproduzida em sua respectiva máquina cliente. A figura 4b reflete a comunicação entre objetos dirigidos pelo usuário. As linhas pontilhadas mostram que a comunicação entre os objetos é feita de forma indireta. As linhas sólidas refletem o fluxo real de mensagens para que a comunicação entre os objetos seja efetuada.

Assim como no sistema NPSNET, no modelo de comunicação descrito acima, os clientes de uma mesma área de interesse (composta de uma ou mais células), podem fazer parte de vários grupos, onde cada grupo é associado a uma célula. Desta forma, uma mensagem de atualização de estado de um objeto é enviada para o(s) grupo(s) de clientes que mantêm réplicas de sua área de interesse, inclusive para o servidor de ambiente que é quem mantêm informações atualizadas sobre todos os objetos da sub-área sob sua responsabilidade, através de *multicasting*. Este modelo de comunicação, apesar de trazer um pouco mais de complexidade ao código da máquina cliente, que tem que saber o endereço *multicasting* do(s) grupo(s) de clientes de sua área de interesse, evita que todo tipo de comunicação tenha que ser feita através do servidor de ambiente. Neste modelo de comunicação, os objetos dinâmicos se comunicam com suas réplicas de forma *peer-to-peer*, ou seja, de máquina cliente para máquina cliente, ao contrário de uma comunicação apenas através do servidor de ambiente, evitando eventuais gargalos.

A comunicação entre os servidores de ambiente e seus clientes se constitui basicamente na troca de mensagens de sincronização entre os objetos que estão presentes no servidor e em seus clientes, movimentação dos objetos de uma célula para outra, troca de mensagens no carregamento inicial da área de interesse do servidor de ambiente para a máquina cliente, resultados de disputas de concorrência etc.

A vantagem deste tipo de comunicação é que não se precisa alocar endereços de grupos dinamicamente, já que cada endereço *multicasting* está associado a uma célula de um AV.

Portanto, não se corre o risco de não ter endereços *multicasting* suficientes em tempo real. O raio de ação de qualquer medium pode ser determinado pelo objeto transmissor e pode variar de acordo com o desejado pelo usuário. Este modelo permite também que diferentes tipos de mídia sejam representados da mesma forma. No sistema NPSNET, cada medium vai demandar o estabelecimento de grupos diferentes de objetos. As desvantagens do modelo proposto são que cada máquina cliente tem que calcular, em tempo real, quais objetos estão dentro de sua área de alcance, além de verificar qual objeto está ligado diretamente àquela máquina cliente e processar a mensagem para o objeto destino. Entretanto, em se tratando de um sistema de AV de grande porte, a otimização da comunicação é um aspecto relevante a ser considerado.

## 6. PROJETO DE UM AMBIENTE VIRTUAL DISTRIBUÍDO DE LARGA ESCALA

A arquitetura do AV distribuído e o modelo de comunicação entre objetos propostos neste artigo fazem parte do projeto de um sistema de ambiente virtual distribuído envolvendo uma rede de PCs e estações de trabalho. O ambiente compreende um número de servidores de ambiente responsáveis pelo armazenamento e controle das sub-áreas do AV. Outras estações e PCs conectados à rede podem ser usados como pontos de acesso ao AV para participação e visualização. Conforme o esquema aqui proposto, sub-áreas do AV e respectivos simuladores são replicados e portanto se tornam cópias redundantes de sub-áreas sendo utilizadas. Isto faz com que o sistema seja inerentemente tolerante à falhas.

A primeira versão deste sistema está considerando o uso de recursos simples de computação gráfica, já que a idéia é suportar equipamentos heterogêneos de usuários finais, com ênfase na estrutura distribuída e problemas de rede. Versões futuras vão tratar da qualidade gráfica.

## 7. CONCLUSÕES

Os Ambientes Virtuais de larga escala são caracterizados por aspectos que demandam a integração de várias áreas tecnológicas. Este artigo identificou e analisou alguns destes aspectos, em particular os relacionados à distribuição e comunicação entre objetos em um AV, através da proposição da arquitetura de um sistema distribuído de suporte às aplicações de realidade virtual de larga escala bem como um modelo de comunicação entre os objetos que habitam o sistema. Para validar estas idéias, um projeto de ambiente virtual distribuído está sendo desenvolvido com ênfase nos aspectos de distribuição e comunicação. Os resultados a serem obtidos vão fornecer meios para uma reavaliação das atuais propostas e convergência para possíveis soluções melhores.

## REFERÊNCIAS

- [ARAU94a] Araujo, R. B. and Kirner, C. - Network Requirements for Large Scale Virtual Environment, paper presented at the IFIP TC 5 WG 5.10

Workshop on Virtual Environments, October 24-25, Coimbra, Portugal, 1994.

- [ARAU94b] Araujo, R. B. and Kirner, C. - Distributed Large Scale Virtual Environments: Discussion, Proposals and Design, paper not published.
- [ARAU95a] Araujo, R.B., Kirner, C. and Chow, E. - The Characteristics of Large Distributed Virtual Environments, Technical Report being edited to be published at the UCCS - Colorado.
- [ARAU95b] Araujo, R.B. and Kirner, C. - A Communication Architecture to Support Large Distributed Virtual Environments, document being prepared.
- [BENF94] Benford, S. et. alii. - Managing Mutual Awareness in Collaborative Virtual Environments, 1994 (electronic copy from the authors).
- [CARL93] Carlsson, C. and Hagsand, O. - DIVE - a Multiuser Virtual Reality System, in IEEE Virtual Reality Annual International Symposium, September 18-22, 1993, Seattle, WA, pp. 394-400.
- [CALV93] Calvin, J. et. alii. - The SIMNET Virtual World Architecture, IEEE Virtual Reality Annual International Symposium, September 18-22, Seattle, WA, 1993, pp. 450-455.
- [ISDA93] Isdale, J. - What is Virtual Reality? A Homebrew Introduction, electronic document in URL=<ftp://sune.uwaterloo.ca/pub/vr>.
- [MACE94a] Macedonia, M. et. alii. - NPSNET: A Network Software Architecture for Large Scale Virtual Environments, Presence Magazine, Vol.3 No.4, 1994.
- [MACE94b] Macedonia, M. et. alii. - Exploiting Reality with Multicast Groups: A Network Architecture for Large Scale Virtual Environments, Submitted to the 1995 VRAIS Conference.
- [PRAT93] Pratt, D. - A Software Architecture for the Construction and Management of Real-Time Virtual Worlds, PhD Dissertation, Naval PostGraduate School, Monterey, CA, June 1993.
- [SHEP93] Sheperd, B. J. - Rationale and Strategy for VR Standards in IEEE Virtual Reality Annual International Symposium, September 18-22, Seattle, WA, 1993.
- [SING94] Singh, G. - BrickNet: A Software Toolkit for Network-Based Virtual Worlds, in Presence Magazine, Vol.3 No.1, 1994.
- [VanD93] van Dam, A. - VR as a Forcing Function: Software Implications of a new Paradigm in IEEE 1993 Symposium on Research Frontiers in Virtual Reality, October 25-26, San Jose, CA, 1993.
- [WLOK94] Wloka, M. - Graphics Interaction in Virtual Environments, IFIP TC WG 5.10 Workshop on Virtual Environments, October, 24-25, Coimbra, Portugal, 1994.
- [ZYDA92] Zyda, M. J. et. alii. - NPSNET: Constructing a 3D Virtual World, Proceedings of the 1992 Symposium on Interactive 3D Graphics, Cambridge, MA, pp. 147-156.