

Um Ambiente Virtual Colaborativo e Telecomandável Baseado em X3D

Bruno Rafael de Araújo Sales, Liliane dos Santos Machado
LabTEVE – Universidade Federal da Paraíba, João Pessoa/PB, Brasil
brunorasales@gmail.com, liliane@di.ufpb.br

Resumo

Este trabalho trata de um sistema de realidade virtual (RV) colaborativo baseado no novo padrão Extensible 3D (X3D). Uma estrutura é definida a fim de demonstrar como é possível o desenvolvimento de ambientes desse tipo utilizando o X3D. Deste modo, é apresentada a união de uma aplicação colaborativa com um sistema de armazenamento de dados que pode dar suporte à interligação do meio virtual com o real a fim de possibilitar experimentos sejam realizados por usuários de forma conjunta utilizando dispositivos localizados remotamente. Assim, o sistema pode vir a auxiliar professores e alunos em suas atividades laboratoriais visto que estes podem compartilhar e colaborar num mesmo ambiente.

1. Introdução

A capacidade de unir pessoas geograficamente distribuídas oferecida pela Internet serve como base para o desenvolvimento de aplicações e serviços colaborativos em diversos campos de atuação. Neste contexto estão inseridos os Ambientes Virtuais Colaborativos (AVC), de maneira a possibilitar seus participantes a interagirem através da simulação de um mundo real ou imaginário ou até mesmo da manipulação de objetos virtuais no mundo real [3].

O que diferencia um AVC de um Ambiente Virtual (AV) de uso local é que no primeiro caso existe a possibilidade de interação e cooperação entre usuários distantes no decorrer da execução de uma determinada tarefa [3]. Um dos problemas encontrados quando utilizados esses ambientes é a garantia de consistência das informações, isto é, garantir que todos os usuários estão presentes em um mundo com mesmas configurações independente do momento em que a conexão foi estabelecida.

A interligação de um AVC com um sistema de armazenamento de dados permite que variáveis de status dos componentes da cena sejam guardadas com suas configurações atuais. Deste modo, é possível ser

feito um trabalho contínuo dentro do AVC, ou seja, mesmo após desconectar o usuário terá a garantia de que a cena irá permanecer da maneira que foi deixada. Isso possibilita o usuário a dar continuidade a tarefas inacabadas em um momento futuro, partindo do ponto onde parou.

Além disso, o armazenamento das configurações atuais da cena pode dar suporte à interligação de AVs com ambientes reais com intuito de permitir o acesso remoto a recursos e dispositivos fisicamente existentes [10,11]. Por meio desse acesso, pessoas situadas em locais distantes podem manter controle sobre um ambiente real somente manipulando o AV relacionado. Dessa maneira, professores e estudantes podem fazer uso de estruturas laboratoriais nos chamados laboratórios remotos [8].

Este artigo pretende apresentar uma estrutura usada no desenvolvimento de um AVC que permite a colaboração entre usuários na execução de tarefas e realização de experimentos. Também é apresentada a possibilidade de interligação do AVC a um ambiente real a fim de gerenciar dispositivos fisicamente existentes por meio de telecomandos.

2. Ambientes Virtuais Colaborativos

Um AVC possui como característica fundamental o fato de englobar além da visualização e navegação em uma cena 3D, a possibilidade de usuários poderem colaborar na execução de tarefas. O aspecto colaborativo é inserido nos AV a fim de permitir que usuários possam estar presentes em um mesmo mundo, realizar tarefas em conjunto, entre outras coisas.

Em ambientes deste tipo a sensação de presença é bem maior do que em aplicações que utilizam tecnologias 2D. O sentimento de “estar junto” ou “trabalhar junto” com outros membros aparece com maior afinco em ambientes 3D pois permite às pessoas identificarem-se visualmente tornando a interação remota e a colaboração mais natural [3].

Entretanto, o que realmente promove a colaboração e o envolvimento em um ambiente 3D é a seqüência de atividades que requer participação e interação entre

estudantes de diferentes localidades. A competição amigável, discussões sobre temas propostos e o conhecimento prévio de cada estudante, faz com que a colaboração traga benefícios no aprendizado de todos [9]. Cada participante do ambiente contribui com os seus conhecimentos, acrescentando novas idéias e informações a outros estudantes [5,6].

A inserção do fator colaborativo em um AV acarreta a análise de alguns pontos antes não observados. Aspectos como a quantidade de usuários, a complexidade da cena, a capacidade da rede utilizada e como será feita a interação entre usuário e AV e usuários entre si, devem ser observados no desenvolvimento de um AVC [3]. Um número elevado de usuários exige mais processamento computacional. Observando também o número de tarefas que poderão ser realizadas por cada usuário. A complexidade do mundo está relacionada ao número de objetos presentes na cena e também do nível de detalhamento do ambiente. A capacidade da rede influencia o funcionamento de um AVC, pois a comunicação e a troca de informações entre os usuários é intensa. Redes de baixa velocidade podem comprometer a sensação de tempo real da aplicação.

Sob o ponto de vista da interação, os usuários podem interagir por meio de ações ou mensagens de texto ou voz, sendo que uma abordagem muito comum é utilização de avatares (representações do usuário através de entidades virtuais) para fazer a representação visual dos usuários no AV.

3. Estrutura do AVC

Esta sessão visa expor uma estrutura proposta para desenvolvimento de Ambientes Virtuais Colaborativos e Telecomandáveis. A estrutura a ser apresentada está dividida em camadas referentes às diversas necessidades que um ambiente desse tipo possui. Como se trata de um sistema cliente/servidor algumas camadas da estrutura do cliente diferem da estrutura do servidor.

O servidor está subdividido em: Gerenciamento, Transporte e Armazenamento. O cliente, por sua vez, é composto por: Gerenciamento, Cena 3D e Transporte. A camada de Gerenciamento do servidor trata de receber informações sobre alterações na cena feitas por clientes e decidir o que fazer com essas informações (a quem retransmitir, enviar para armazenamento, etc). A camada de Armazenamento é responsável por receber dados da camada de gerenciamento e adicioná-los ou alterá-los no banco de dados. Bem como recuperar do banco de dados, dados solicitados pela camada de gerenciamento.

No cliente, a camada de Gerenciamento tem a tarefa de receber dados da camada de transporte (provenientes do servidor) e enviá-los a cena em forma de eventos. Cena 3D diz respeito ao ambiente virtual no qual o cliente está inserido. As camadas de transporte do cliente e do servidor se assemelham. Ambas possuem os mesmos objetivos: promover a troca de informações entre as estações. A Figura 1 ilustra as estruturas de cliente e servidor.

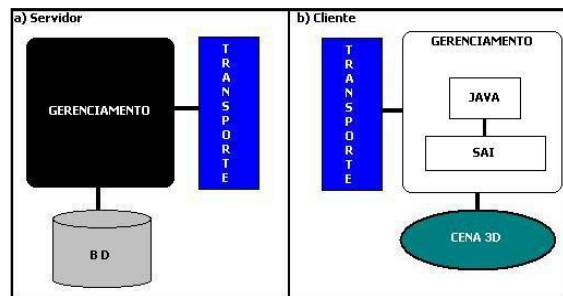


Figura 1: Estrutura de camadas do Servidor e do Cliente.

4. X3D e Scene Access Interface (SAI)

Desenvolvido pelo grupo Web3D, o X3D é “*um padrão aberto para distribuir conteúdos de realidade virtual em 3D, em especial pela Internet.*” [1]. Um arquivo X3D contém não só dados geométricos, mas também descrições de comportamentos dos objetos e da cena. O X3D surgiu com o intuito de substituir o padrão existente (VRML – *Virtual Reality Modeling Language*), corrigindo alguns aspectos indesejáveis que este contém e incorporando avanços de dispositivos e técnicas. O X3D oferece suporte a uma série de dispositivos, métodos e técnicas, sendo possível adicionar funcionalidades às suas aplicações. Essa extensibilidade (adição de funcionalidades) e o conjunto de serviços (funcionalidades já disponíveis) do X3D são definidos por *Profiles* e *components* [1,7].

Apesar de suas diversas características, o X3D não é uma linguagem de propósito geral. Por esta razão uma API chamada SAI (*Scene Access Interface*) está sendo desenvolvida com o objetivo de permitir integrar cenas X3D e aplicações desenvolvidas em outras linguagens, como Java, C ou linguagens de *script*. A SAI visa possibilitar o acesso a uma cena X3D através de uma aplicação externa. Assim, passa a ser possível inserir objetos, removê-los, notificar eventos e realizar modificações que afetam a cena e, conseqüentemente, elementos relacionados e externos a ela. Ou seja, a cena pode ser totalmente controlada de forma indireta através de um programa escrito em linguagens de programação ou *script* [4,7].

5. Desenvolvimento

O desenvolvimento do AVC baseou-se na estrutura definida para cliente e servidor. Foram abordadas questões sobre consistências dos mundos, colaboração efetiva dos usuários, armazenamento do status da cena, telemanipulação, sensação de presença e tempo real. Através desta estrutura, os objetos virtuais passam a poder representar equipamentos reais localizados remotamente de modo que seja viabilizada a cooperação entre estudantes situados em locais distantes na realização de experimentos em laboratórios remotos. Uma ilustração de como as ações executadas no ambiente virtual chegam ao ambiente real é mostrada na Figura 2.

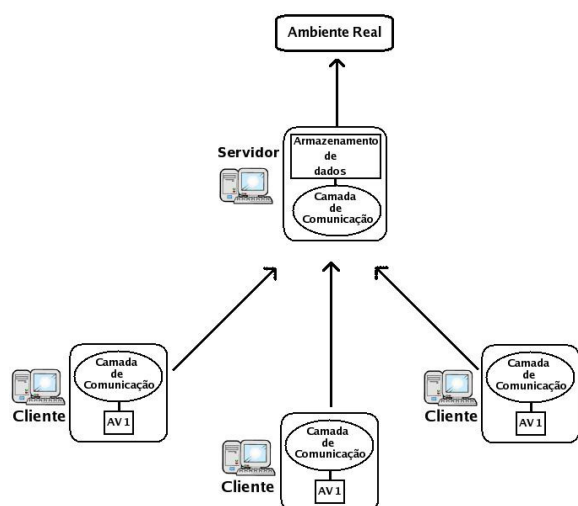


Figura 2. Esquema de como ações executadas na aplicação do cliente chegam na aplicação do servidor são interpretadas e repassadas para o ambiente real.

O X3D se inseriu no desenvolvimento do sistema proposto como padrão para modelagem do AV. As vantagens do X3D sobre o VRML influenciaram essa decisão, visto que haverá necessidade de integrar dispositivos específicos ao AV e permitir seus controles. Neste caso, a SAI permite esta integração no X3D.

O modelo de comunicação adotado foi o cliente/servidor. Nesse modelo os dados são armazenados em computadores denominados servidores. Além disso, toda informação trocada por clientes passa antes pelo servidor. Este gerencia para quem irá reenviar as mensagens que chegaram a ele. O protocolo de comunicação utilizado foi o UDP (*User Datagram Protocol*), devido à sua maior rapidez na

entrega dos dados e não necessidade de confirmação de mensagens [2].

O sistema permite ainda que usuários estejam conectados a mundos diferentes ou até mesmo partes diferentes do mesmo mundo virtual. Assim, as mensagens que chegam ao servidor provenientes de um determinado cliente, só são repassadas para os demais que se interessarem por essa mensagem, ou seja, os que estiverem inseridos no mesmo ambiente. Isto possibilita o carregamento gradativo das cenas, isto é, ao invés de toda a cena ser armazenada localmente, apenas a parte na qual o usuário está navegando é requisitada. Tal abordagem evita a sobrecarga de memória principal da máquina cliente, além de diminuir o tempo de espera do cliente para iniciar a navegação. Desse modo, o cliente recebe do servidor o ambiente em que deseja se inserir e também as configurações atuais deste ambiente, ou seja, o status de componentes da cena. Essas configurações ficam armazenadas no banco de dados. Para cada cena estão armazenadas as configurações de componentes que podem sofrer algum tipo de alteração. Por exemplo, o status de uma lâmpada informando se ela se encontra ligada ou desligada. Com estas configurações armazenadas, garante-se que os usuários que escolherem uma mesma cena para navegar estejam inseridos em um mesmo contexto, com o status dos componentes iguais.

6. Resultados

Os resultados obtidos consolidaram o funcionamento do sistema e foi possível verificar a colaboração entre os usuários em tempo real. A aplicação utilizada para testes tratou um ambiente bastante simples, representando uma sala de aula composta por cadeiras que podem ser rotacionadas e um interruptor que apaga ou acende a luz. Neste caso, o servidor não possui uma cena gráfica e apenas escuta conexões de clientes, gerencia as informações recebidas e se comunica com o banco de dados.

Assim que a cena é exibida, o cliente pode navegar livremente além de fazer alterações desejadas e visualizar alterações feitas por outros usuários presentes na mesma cena. A Figura 3 exhibe a cena utilizada para testes mostrando as que modificações podem ser feitas nas suas configurações. Nesta cena, o usuário é capaz de acender ou apagar uma luz clicando em um suposto interruptor e também rotacionar as cadeiras.

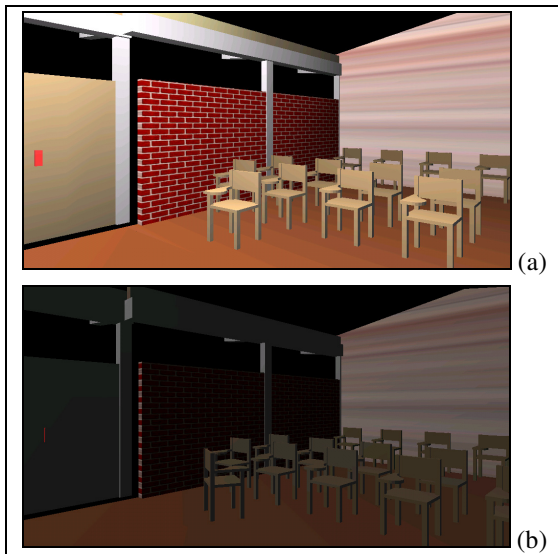


Figura 3: a) Cena original não modificada e b) Cena com a luz apagada e uma cadeira rotacionada.

Foram realizados testes com cinco clientes, localizados geograficamente distantes, comunicando-se através da Internet. Os clientes possuíam uma conexão com velocidade de 300kbps. Os atrasos não foram perceptíveis para os usuários e as alterações feitas por um cliente foram observadas quase que de imediato pelos demais. Adicionalmente foi realizado um teste composto por quatro clientes, dois conectados a um ambiente e dois conectados a outro. As modificações individuais de cada cliente não afetaram os clientes conectados a outro ambiente e apenas foram reenviadas para os que estavam inseridos na mesma cena do cliente que realizou modificações, confirmando assim o funcionamento correto do sistema.

7. Conclusão

A colaboração em AV é de fato algo que pode dar suporte ao processo de ensino e aprendizado de estudantes e professores. Além do mais, fazer com que as tarefas ou ações feitas sejam armazenadas em um banco de dados por meio de variáveis de status dá ao usuário a sensação de dinamismo e tempo contínuo. Desta maneira, tarefas inacabadas podem ser concluídas em um momento futuro. Considerando ainda a possibilidade de interligação do AV com um ambiente real, conclui-se que fazer uso desses AVCs direcionados a experimentos laboratoriais beneficia ainda mais o processo de construção do conhecimento devido à possibilidade de realização de experimentos através dos laboratórios remotos.

Como trabalho futuro sugere-se o melhor tratamento no caso em que dois clientes tentam modificar um mesmo objeto simultaneamente. Além disso, também devem ser feitos testes de rede a fim de medir com precisão os atrasos das mensagens, mesmo tendo sido observado que tais atrasos não comprometeram o bom funcionamento do sistema.

8. Agradecimentos

Este trabalho tem apoio do CNPq através do processo 485437/2007-4.

9. Referências

- [1] A. Cardoso, C. Kirner, E. Lamounier, J. Kelner, "Tecnologias para o Desenvolvimento de Sistemas de Realidade Virtual e Aumentada". Ed.UFPE, Recife. 2007.
- [2] A. S. Tanenbaum, "Redes de Computadores". Editora Campus, 4ª ed. 2003.
- [3] L. C. A. Rinaldi, et al. "Fundamentos e Tecnologia de Realidade Virtual e Aumentada", cap 5. SBC. 2006.
- [4] Web3D, "SAI Tutorial". Online: www.xj3d.org/tutorials/general_sai.html. Acesso em: outubro/2007.
- [5] N. D. Blas, C. Poggi, "European virtual classrooms: building effective "virtual" educational experiences". *Virtual Reality*, v. 11, n.2-3, pp: 129-143. 2007.
- [6] B. Lok, et al. "Applying virtual reality in medical communication education". *Virtual Reality*, v. 10, n.3, pp: 185-195. 2006.
- [7] Web3D, "X3D Public Specifications". Online: www.web3d.org. Acesso em: outubro/2007.
- [8] L. S. Machado, et al. "Improving Interaction in Remote Labs Using Haptic Devices". Em: Proc. of REV International Conference, Porto/Portugal. 2007.
- [9] Youngblut, C., "Educational Uses of Virtual Reality Technology". Technical Report IDA Document D-2128, Institute for Defense Analyses, Alexandria, VA, 1998.
- [10] Monferrer, A., Bonyuet, D., "Cooperative Robot Teleoperation through VR Interfaces". *Proc. Sixth Int. Conf.on Information Visualization (IV'02)*, 2002.
- [11] Müller, D., "MARVEL- Mechatronics Training in Real and Vritual Environments. Concepts, Pratices, and Recommendations", Marvel. 2005.