

Servidor Multimídia RIO em Ensino a Distância

Bernardo Calil Machado Netto*, Jorge Allyson Azevedo†,
Edmundo A. Souza e Silva, Rosa M. M. Leão

¹UFRJ - COPPE/Sistemas
LAND Prédio do Centro de Tecnologia
Bloco I-2000 - Sala I-246
Caixa Postal 68511 21941-972 Rio de Janeiro, RJ

{bnetto,allyson,edmundo,rosam}@land.ufrj.br

Abstract. *RIO is an universal multimedia storage system with random allocation and block replication. Its visualization client, riommclient, has several features like VCR buttons to control the video, slide synchronization and an index to quick access to specific points of the media. These features makes RIO an useful tool for distance learning. Currently RIO is being used in a undergraduate program.*

Resumo. *O Servidor RIO é um sistema de armazenamento multimídia universal que usa alocação aleatória e replicação de blocos. O RIO possui um cliente para visualizar seus objetos, o riommclient. Esse cliente possui diversas funcionalidades, como os botões de um videocassete para controlar a exibição de vídeo, sincronização de transparências e um índice para acesso rápido a partes específicas da mídia. Essas características tornaram o RIO uma ferramenta útil para ensino a distância e, por isso, esse sistema já está sendo usado em um curso superior.*

1. Introdução

O Servidor RIO é um sistema de armazenamento multimídia universal que usa alocação aleatória e replicação de blocos. Sendo um servidor universal, o RIO suporta vários tipos de mídias: vídeo, áudio, texto, imagem, além de ser capaz de gerenciar aplicações com ou sem restrição de tempo. Aplicações como visualização de imagens e textos são exemplos de aplicações sem restrição de tempo. Por outro lado a exibição de um vídeo sob demanda é um exemplo de aplicação de tempo real. O acesso aos objetos armazenados no servidor é feito através dos clientes riommclient e riosh.

O RIO foi originalmente desenvolvido pelo laboratório de multimídia da UCLA (University of California at Los Angeles). A partir de 1999, o processo de desenvolvimento foi assumido pelo grupo de pesquisas LAND (Laboratório de ANálise, modelagem e Desenvolvimento de redes e sistemas de computação) [LAND, 2005] que desenvolveu um outro cliente para visualização de vídeos, o riommclient, e um cliente gráfico para administração dos objetos do servidor, o riosh. O desenvolvimento desse novo cliente criou novas possibilidades, pois ele possui sincronização com transparências e estas permitem uma melhor interação entre o usuário e o sistema.

As novas funcionalidades do RIO o tornaram uma ferramenta adequada para o ensino a distância. Entre os diversos projetos desse segmento existe o consórcio CEDERJ

*Financiado por CEDERJ.

†Financiado por CEDERJ e CNPq

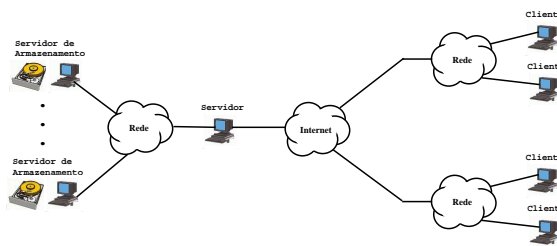


Figura 1: Arquitetura RIO

[CEDERJ, 2005] que tem como objetivo oferecer cursos superiores a distância utilizando novas tecnologias. O CEDERJ procura utilizar, preferencialmente, software livre em seus laboratórios de informática, como é o caso do curso de Tecnologia em Sistemas de Computação, que tem como principal diferencial a utilização do servidor RIO para prover acesso às aulas de suas disciplinas. O objetivo do RIO é fazer o armazenamento, a transmissão e a exibição das aulas incluindo suas respectivas transparências. É importante ressaltar que o RIO não é um sistema de gerenciamento de usuários e disciplinas de cursos on-line como por exemplo Moodle [Moodle, 2005] e [TelEduc, 2005]. Porém, o RIO pode ser usado em conjunto com estes sistemas, como é o caso do CEDERJ que tem o seu próprio sistema de gerenciamento de conteúdo.

Esse artigo está organizado da seguinte forma. A seção 2 discute as principais características do servidor RIO. Uma breve descrição dos clientes é apresentada na seção 3. A seção 4 traz um exemplo de aplicação do RIO.

2. Servidor Multimídia RIO

Para o servidor, todos os tipos de mídias são chamados de objetos e são armazenados da mesma forma. Para melhorar o desempenho do servidor, os objetos são divididos em blocos de dados e estes são armazenados nos discos, onde é feita uma alocação aleatória dos blocos [Santos et al., 2000]. Nesta técnica, é escolhido um disco aleatório e uma posição aleatória dentro dele para o armazenamento. O tamanho do bloco de dados é definido na configuração do servidor.

A Figura 1 mostra que a arquitetura do RIO é composta por um único nó servidor, um ou mais nós de armazenamento e os clientes. A comunicação entre o cliente e o servidor é feita usando os protocolos TCP (*Transmission Control Protocol*) e UDP (*User Datagram Protocol*), sendo o primeiro para comunicação de controle, como pedidos de blocos de dados, e o segundo para a transmissão dos blocos. Os blocos de dados são transmitidos diretamente do servidor de armazenamento para o cliente, não sobrecarregando assim o servidor principal.

3. Clientes

3.1. riommclient

O riommclient é o cliente de visualização de objetos do RIO. A principal função desse cliente é gerenciar a comunicação com o servidor bem como a exibição e sincronização das transparências e interatividade do usuário. A Figura 2 mostra os componentes do cliente. Como *player* podem ser utilizados o MPlayer [MPlayer Project, 2005] ou o RIOxine, sendo este último desenvolvido no LAND, a partir das bibliotecas do Xine [Xine project, 2005]. As transparências podem ser exibidas por um navegador, como

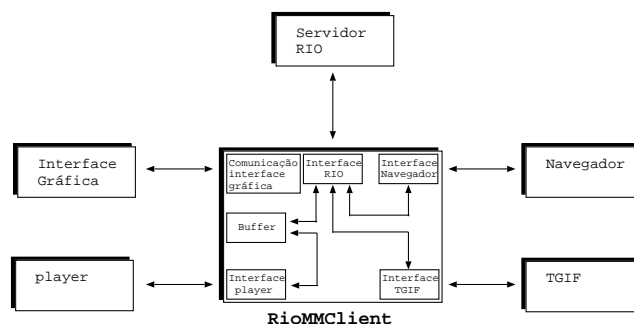


Figura 2: Estrutura do riommclient

o Mozilla Firefox, ou pelo TGIF [William C. Cheng, 2005], um software livre para desenho vetorial 2D, com diversas funcionalidades. A interface do riommclient oferece as principais funções de um videocassete, como tocar, pausar, retroceder, avançar e parar.

Em [de Quevedo Cardozo, 2002], foram desenvolvidas novas funcionalidades para o RIO como a implementação, no servidor, de *buffers* para permitir o gerenciamento dos dados solicitados pelo cliente de forma a minimizar o *jitter* decorrente da leitura nos discos e da transmissão pela rede. Outra característica adicionada foi um novo algoritmo detalhado para a admissão de novos usuários no sistema, além de um módulo para retirada de diversas medidas de desempenho, como tempo médio de leitura de um bloco de dados nos discos.

Em [Netto, 2004], foi desenvolvida uma nova técnica, chamada *Patching* Interativo, para compartilhamento dos fluxos de dados transmitidos aos clientes com o objetivo de reduzir a carga total gerada na rede. Esta técnica foi implementada como um módulo do servidor RIO e foram necessárias várias mudanças no riommclient.

Em [Botelho, 2005], são apresentadas novas funcionalidades que foram adicionadas ao riommclient para facilitar a interação em aplicações de ensino a distância. Foi criado um índice com todos os tópicos da aula, permitindo que o aluno tenha acesso direto aos principais pontos discutidos naquela aula. Além disso, o riommclient registra o comportamento do usuário em um arquivo de *log* que, ao término de uma sessão, é armazenado no servidor. Esses *logs* serão usados posteriormente para criação de modelos de comportamento dos usuários, que são base para projetos de pesquisa no LAND.

3.2. riosh

O riosh (*RIO shell*) é um cliente de administração dos objetos armazenados no servidor RIO. Com ele é possível executar as principais funções de um gerenciador de arquivos, tanto em modo texto quanto em modo gráfico. Dentre essas funções estão a criação/cópia/remoção de objetos/diretórios e alteração de nomes, além de edição e visualização de objetos que estão em formato texto. É possível administrar simultaneamente diversos servidores em modo gráfico, como pode ser visualizado na Figura 3. Nesta figura, podem ser identificadas três janelas: *localhost*, onde estão os arquivos do sistema local e *trindade* e *forte* que exibem o conteúdo de dois servidores RIO diferentes.

4. Aplicação em Ensino a Distância

Uma das aplicações mais importantes do Servidor Multimídia RIO é o ensino a distância. Utilizando a sincronização de transparências com o vídeo, é possível montar aulas interativas, onde o professor pode pedir ao aluno, por exemplo, que execute uma animação na própria transparência. Dessa forma, a experiência de assistir uma aula se torna mais interessante, conseguindo assim atrair a atenção dos alunos. Essa funcionalidade do RIO pode

informação será utilizada para criar um modelo do comportamento dos usuários, o que pode levar a criação de novas funcionalidades no RIO, além, é claro, da otimização dos mecanismos já existentes.

Para facilitar ainda mais o acesso a essa tecnologia, foi criado um DVD-ROM, contendo um sistema operacional GNU/Linux com o RIO e os principais softwares de uso geral, para que os alunos possam assistir às aulas e realizar os exercícios em casa. Com esse DVD-ROM, o aluno tem apenas que inicializar o computador para ter acesso ao ambiente, sem precisar instalar qualquer outro software.

5. Conclusão

O sistema multimídia RIO se adequa perfeitamente ao ensino a distância pois provê ao aluno a interatividade desejada, como Tocar (*Play*), Pausar (*Pause*), Parar (*Stop*), Avançar (*FastForward*) e Retorceder (*FastRewind*), além de possibilitar a navegação por meio de um índice, onde o aluno pode acessar diretamente os principais tópicos da aula que está assistindo. O RIO pode ser integrado a um sistema de gerenciamento de usuários e disciplinas de cursos on-line, melhorando o ambiente de aula dos alunos.

Referências

- Botelho, E. R. (2005). Análise da interatividade dos alunos acessando um servidor multimídia em um ambiente de ensino a distância. Trabalho de Conclusão de Curso, DCC/UFRJ.
- CEDERJ (2005). Fundação Centro de Ciências e Educação Superior a Distância do Estado do Rio de Janeiro. URL <http://www.cederj.rj.gov.br>. Último acesso: 09/05/2005.
- de Quevedo Cardozo, A. (2002). Mecanismos para Garantir Qualidade de Serviço de Aplicações de Vídeos sob Demanda. Tese de Mestrado, COPPE/UFRJ.
- LAND (2005). Laboratory for Modeling, Analysis and Development of Networks and Computing Systems. URL <http://www.land.ufrj.br>. Último acesso: 09/05/2005.
- Moodle (2005). Moodle - A Free, Open Source Course Management System fo Online Learning. URL <http://moodle.org>. Último acesso: 09/05/2005.
- MPlayer Project (2005). MPlayer: Movie Player for Linux. URL <http://www.mplayerhq.hu>. Último acesso: 09/05/2005.
- Netto, B. C. M. (2004). Patching Interativo: Um Novo Método de Compartilhamento de Recursos para Transmissão de Vídeo com Alta Interatividade. Tese de Mestrado, COPPE/UFRJ.
- Santos, J., Muntz, R., and Ribeiro-Neto, B. (2000). Comparing Random Data Allocation and Data Striping in Multimedia Storage Servers. In *Proceedings of ACM SIGMETRICS*, pages 44–55.
- TelEduc (2005). TelEduc: Ambiente de Ensino a Distância . URL <http://teleduc.nied.unicamp.br>. Último acesso: 09/05/2005.
- William C. Cheng (2005). TGIF: TANGRAM Graphic Interface Facility. URL <http://bourbon.usc.edu:8001/tgif/index.html>. Último acesso: 09/05/2005.
- Xine project (2005). Xine: a free video player. URL <http://xinehq.de>. Último acesso: 09/05/2005.