

Giga-Colab: Uma Arquitetura de Ambiente Virtual Colaborativo com suporte a Serviços Multimídia*

Raoni Kulesza¹, Luiz G. P. Alves¹, Fernando L. Almeida¹, Marcos F. Jardini¹,
Marco A. V. M. Vasconcelos², Guido L. S. Filho²,
Regina M. Silveira¹, Graça Bressan¹

¹Laboratório de Arquitetura e Redes de Computadores (LARC) – Escola Politécnica da Universidade de São Paulo (EPUSP) – São Paulo – SP – Brasil

²Laboratório de Aplicações de Vídeo Digital (LAVID) – Departamento de Informática – Universidade Federal da Paraíba (UFPB) – João Pessoa – PB - Brasil

{raoni, luizgpa, falmeida, mjardini, regina, gressan}@larc.usp.br}
{marcovas, guido}@avid.ufpb.br}

Abstract. *The development of collaborative and multimedia systems is a complex task and one of the key challenges is to promote the reuse and integration of those two software categories in the same environment. This paper presents an extensible collaborative virtual environment architecture based on client-server model, taking advantage of a service oriented approach and open multimedia standards (ISMA, H.323 and SIP). To validate this proposal, a prototype of a collaborative virtual environment was developed, supporting two multimedia applications: video on demand and videoconference. The project had as target the Giga Network (RNP/CPqD) and had as its main requirement the support of high definition video, resulting in some challenging issues to overcome.*

Resumo. *Desenvolver sistemas colaborativos e multimídia é uma tarefa bastante complexa e um dos principais desafios é promover a reutilização e integração dessas duas categorias de software num mesmo ambiente. Neste contexto, apresentamos uma arquitetura de um ambiente virtual colaborativo extensível baseada no modelo cliente-servidor, mas que utiliza uma abordagem orientada a serviços e padrões abertos da área multimídia (ISMA, H.323 e SIP). Para validar esta proposta, foi desenvolvido um protótipo de um ambiente virtual colaborativo com suporte a duas aplicações multimídia: vídeo sob demanda e videoconferência. O projeto é focado na Rede Giga (RNP/CPqD) e tem como principal requisito o suporte a vídeos de alta definição, resultando em alguns problemas desafiadores a serem superados.*

1. Introdução

Sistemas de suporte ao trabalho cooperativo (do inglês CSCW, *Computer Supported Cooperative Work*), ou sistemas colaborativos, incluem um ambiente compartilhado com funcionalidades relacionadas à cooperação, coordenação e comunicação para suportar grupos de usuários engajados em uma atividade comum (ou objetivo comum).

* Trabalho Financiado pela RNP com recursos FINEP/FUNTEL (Sub-Projeto Giga 2472)

Devido ao avanço nas áreas de computação e redes comunicação, cada vez mais esses sistemas utilizam aplicações multimídia com o objetivo de melhorar as funcionalidades e experiência dos usuários no processo de colaboração [Steinmetz e Nahrstedt 2004]. Com isso, o desenvolvimento de sistemas colaborativos, que já era considerado complexo [Ellis et al. 1991], torna-se um desafio ainda maior, uma vez que as aplicações multimídia são caracterizadas pelo processamento de grande volume de dados e tempo de reprodução contínuo. Adicionalmente, devido à natureza dinâmica das funcionalidades colaborativas, também é exigido que os sistemas colaborativos sejam flexíveis o suficiente para se adequarem a novos requisitos do ambiente e usuários [Kahler et al. 2000].

Para tratar o aspecto da complexidade na construção de sistemas colaborativos esta proposta se baseia em um Ambiente Virtual Colaborativo extensível (do inglês CVE, *Collaborative Virtual Environment*) desenvolvido por [Harada 2001], que fornece uma infra-estrutura para o desenvolvimento de novas aplicações colaborativas: o jVE (*Java Virtual Environment*). Entretanto, de forma a facilitar a integração de novas funcionalidades e interoperabilidade com aplicações já existentes, a arquitetura do ambiente foi estendida para suportar tecnologias de serviços Web e protocolos abertos na área de sistemas multimídia. Particularmente, foram utilizados SOAP (*Simple Object Access Protocol*) e WSDL (*Web Service Definition Language*) [Alonso et al. 2004] para acesso e configuração de serviços, os padrões ISMA (*Internet Streaming Media Alliance*) [Fuchs e Farber 2005] para distribuição de mídias, e H.323 [H.323 1999] e SIP (*Session Initiation Protocol*) [Rosenberg et al. 2002] para controle de sessões de conferência.

Como principal resultado deste trabalho, uma solução de um ambiente virtual colaborativo está sendo desenvolvida no contexto do projeto GIGA-RNP, de forma a facilitar a interação e o trabalho cooperativo entre pessoas situadas remotamente através de conferência multimídia e transmissão e recepção de elevado volume dados. Uma grande vantagem desta proposta é a reutilização de aplicações multimídia já desenvolvidas anteriormente em outros projetos de nossos grupos e que têm capacidades elevadas para a transmissão e recepção de vídeo.

Além desta seção de introdução, este artigo está estruturado em mais 4 seções. A próxima seção apresenta os trabalhos relacionados. Na seção 3 é apresentada a proposta, através de uma análise, visão geral da arquitetura e detalhamento dos principais módulos. A seção 4 descreve a implementação de um protótipo e resultados preliminares da implantação na rede do projeto GIGA-RNP. Na seção 5 são discutidas as conclusões e trabalhos futuros e em andamento.

2. Trabalhos Correlatos

Atualmente, existem vários sistemas colaborativos na comunidade científica que fornecem suporte a aplicações colaborativas e multimídia conjuntamente. Destacam-se os projetos Agora [Agora], do “Centre for e-Science” da Universidade de Lancaster, AccessGrid [AccessGrid], do “Argonne National Laboratory” de Chicago, o ConferenceXP [Beavers et al. 2004], da Microsoft Research e o Global Multimedia Collaboration System (Global MMCS) [Wu et al. 2004] do “Communit Grids Lab” da Universidade de Indiana.

O *AccessGrid*, baseia-se numa infra-estrutura de servidores espalhados, chamados de nós, interconectados formando uma grade. O acesso à grade e os requisitos para se estabelecer uma sessão colaborativa seguem um protocolo próprio (o *AccessGrid* ou *AG*). O *AG* define um mecanismo de comunicação entre todas as entidades do sistema. É através dele que se inicia uma nova sessão para o ambiente colaborativo, chamada de “local do evento ou fonte de vídeo” (do inglês, *Venue*) e definem-se as características e recursos utilizados pela mesma, como portas no servidor e as capacidades de áudio e vídeo.

O *ConferenceXP* é semelhante ao *AccessGrid*. A principal diferença entre ambos está no emprego dos protocolos de comunicação, enquanto o segundo baseia-se na utilização restrita de aplicações multimídia existentes (VIC e RAT)¹ sobre vários protocolos (RTSP, RTP, XMPP, AG) e uma rede ponto-a-ponto de acesso restrito (os servidores de *Venue*), o primeiro utiliza somente RTP para todas as comunicações e o conceito de rede ponto-a-ponto com acesso através da Internet, ou seja, não restrito, para tratar as interligações entre os participantes. Adicionalmente, o *ConferenceXP* é desenvolvido totalmente sobre a plataforma .NET, limitando a portabilidade do sistema. Ressalta-se a utilização de um mecanismo específico e simplificado de comunicação, que define um pacote genérico de transporte chamado *RTOBJECTS*. Estes são objetos (do inglês, *Objects*) usados para fluxo de dados e troca de mensagens e que, apesar de serem encapsulados em pacotes RTP, são diferenciados dos fluxos de áudio e vídeo.

O Agora é uma ferramenta baseada na plataforma Java, que procura criar um ambiente colaborativo completo, abrangendo as mais diversas formas de comunicação e colaboração. Ele é desenvolvido sobre o ambiente SAKAI, seguindo especificações *Java Enterprise Edition*² (*servlets* e *portlets*), e gerencia todo o processo de uma sessão de colaboração, oferecendo aplicações colaborativas. O Agora engloba as funções do SAKAI acrescentando uma interface cliente personalizável com suporte conferência multimídia utilizando o *Java Media Framework*³ (JMF), área de trabalho compartilhada, *whiteboard* e vídeo sob demanda.

Seguindo o conceito de descentralização que as redes ponto-a-ponto oferecem como no *ConferenceXP*, o *Global MMCS* oferece um sistema distribuído, baseado em serviços Web, para gerenciamento de sessões de colaboração e com suporte a multimídia. Como infra-estrutura de distribuição é utilizada uma rede sobreposta, que é capaz de registrar e buscar informações dos participantes de forma interoperável. O principal objetivo é tratar de forma transparente a heterogeneidade de sistemas existentes, possibilitando o uso dos protocolos SIP, H.323 e AG para sessões de conferência. Assim como o Agora, o *Global MMCS* oferece um cliente JMF com suporte de áudio e vídeo.

A principal diferença da arquitetura Giga-Colab, que será apresentada neste trabalho, em relação aos demais trabalhos com escopo similar, é a utilização de uma arquitetura híbrida, cliente-servidor para as aplicações de colaboração e baseada em serviços (que pode se tornar ponto-a-ponto) para as aplicações multimídia. Outro

¹ Soluções de software-livre descontinuadas e limitadas no suporte a áudio e vídeo.

² Solução da *Sun Microsystems*TM para desenvolvimento de aplicações corporativas avançadas.

³ Solução da *Sun Microsystems*TM para desenvolvimento de aplicações multimídia.

aspecto é a capacidade de tratamento de vídeo. No Giga-Colab as aplicações multimídia são baseadas em soluções desenvolvidas em código nativo integrado com Java e suporte a padrões recentes de codificação, o que oferece um melhor desempenho e capacidade na exibição de vídeos para conferência e/ou vídeo sob demanda. A Tabela 1 apresenta um resumo comparativo das características de cada trabalho.

Tabela 1. Comparação de características entre ambientes colaborativos

	AccessGrid	ConferenceXP	Agora	Global MMCS	Giga-Colab
Tipo de Arquitetura para Gerenciamento de Sessão Colaborativa	Ponto-a-ponto (para servidores de Venue)	Ponto-a-ponto	Cliente-Servidor	Ponto-a-ponto	Cliente-Servidor
Bibliotecas para áudio e vídeo	Nativas (VIC e RAT)	DirectShow e Windows Media Player	Java Media Framework	Java Media Framework	Java2D + Nativas (pwlib e ffmpeglib)
Supporte a ferramentas de colaboração	Através de desenvolvimento de terceiros	Integração com qualquer produto .NET	WhiteBoard, chat de texto, área de trabalho compartilhada. Capacidade de integrar serviços web	Em desenvolvimento Capacidade de integrar serviços web	WhiteBoard e chat de texto. Integração com qualquer aplicação Java (através do jVE) ou serviços web
Capacidade de Gravação de Vídeo	Sim	Sim	Sim	Sim	Sim
Resoluções de vídeo suportadas	176x144, 320x240, 352x288	Informação não fornecida	176x144, 320x240, 352x288	176x144, 320x240, 352x288	176x144, 352x288, 720x480, 1280x720
Multiplataforma	Sim	Não, apenas Windows	Sim, mas com limitações no Linux para vídeo.	Sim, mas com limitações no Linux para vídeo.	Sim
Número de participantes em áudio e vídeo⁴	Ilimitado	4	Ilimitado ³	Ilimitado ³	16
Suporte a protocolos abertos na área de aplicações multimídia	RTSP e RTP	RTP	RTP	SIP, H.323, AG	HTTP, RTSP, SDP, RTP, SIP e H.323

3. Arquitetura Giga-Colab

Nesta seção serão apresentadas inicialmente (subseção 3.1) as características (requisitos) do domínio de aplicações de CSCW que o Giga-Colab deverá atender incluindo características comuns aos projetos analisados. As demais subseções (3.2 e 3.3) descrevem uma proposta de arquitetura para sistemas colaborativos, apresentando seus elementos e as principais interfaces de comunicação entre eles.

⁴ Quando ilimitado, o número de participantes depende da largura de banda disponível e capacidade de processamento das máquinas dos clientes.

3.1. Características

As características comuns de sistemas colaborativos foram identificadas através da descrição de dimensões de análise propostas por [Licea 2006] e do modelo 3C de Colaboração apresentado por [Gerosa 2006].

De acordo com [Licea 2006] há sete dimensões de análise obtidas do estudo de sistemas colaborativos, de modo que cada dimensão representa um requisito ou decisão de projeto para a aplicação. As dimensões são: interação, participação, visualização, notificação, coordenação, distribuição e suporte. Já [Gerosa 2006] faz uma análise do modelo 3C de Colaboração proposto originalmente por [Ellis et al. 1991] para suporte computacional à colaboração e que se baseia em três principais características: comunicação, coordenação e cooperação. A percepção (do inglês, *awareness*), que permeia os 3C's, também é um requisito essencial presente em sistemas colaborativos. A partir do estudo dos trabalhos citados, foram destacados os requisitos para compor a arquitetura Giga-Colab:

- Existência de um ambiente virtual que representa um espaço compartilhado no qual há operação conjunta dos usuários para atender a cooperação;
- Definição de uma arquitetura de comunicação, que neste caso é caracterizada como cliente-servidor, uma vez que o jVE é baseado na relação entre clientes e um ambiente virtual centralizado;
- As informações do ambiente virtual devem ser compartilhadas através de interação síncrona ou assíncrona, e explícita ou implicitamente. A troca de informações é dita explícita quando um cliente inicia explicitamente o envio da informação e implícita quando o sistema dissemina uma informação sem que um cliente precise solicitar;
- Definição de um modelo para coordenação do ambiente virtual: uma Sala (metáfora de uma sala real voltada para um ambiente virtual) pode conter uma ou mais atividades associadas. Cada atividade é composta por usuários com suas respectivas permissões, um repositório compartilhado de conteúdo (arquivos de vários tipos de mídias, representando documentos, imagens, áudio e vídeo) e aplicações colaborativas (ver seção 4). A coordenação também abrange aspectos de controle de concorrência no acesso e, isto é realizado no nível de objetos do ambiente virtual;
- Existência de um mecanismo de persistência da informação, armazenando o estado dos objetos de uma Sala. Adicionalmente, é necessário também um mecanismo de *cache* desses objetos, de forma a melhorar a recuperação dessa informação;
- Suporte a um mecanismo de notificação e percepção das atividades do grupo, onde o sistema comunica aos usuários mudanças no contexto de uma Sala (entrada/saída de usuário, inclusão de novo arquivo no repositório, estado atual da tarefa realizada etc.);
- Ser extensível para suportar a integração com ferramentas e serviços externos;
- Independência de interface de usuário.

De posse destes requisitos elencados acima será detalhado na próxima subseção a visão geral da arquitetura proposta.

3.2. Visão Geral

A estrutura geral da arquitetura proposta neste trabalho (Figura 1) é baseada em um modelo híbrido, onde existe uma entidade central, denominada *Servidor Giga-Colab* (responsável pelo gerenciamento da colaboração e dos serviços) e vários outras entidades, denominadas *Serviços*, que podem estar distribuídas pela rede e complementam as funcionalidades de colaboração, provendo serviços tanto para o servidor central, como diretamente para os clientes (*Cliente Giga-Colab*).

A arquitetura concebida permitiu a reutilização de módulos do jVE desenvolvidos por [Harada 2001] que se beneficiam do modelo cliente-servidor e ao mesmo tempo, através do uso de tecnologias de serviços Web, provê flexibilidade e escalabilidade para adicionar novas funcionalidades, por exemplo, para as aplicações multimídia servidoras. Nas subseções seguintes serão detalhados cada subsistema da arquitetura e seus respectivos módulos.

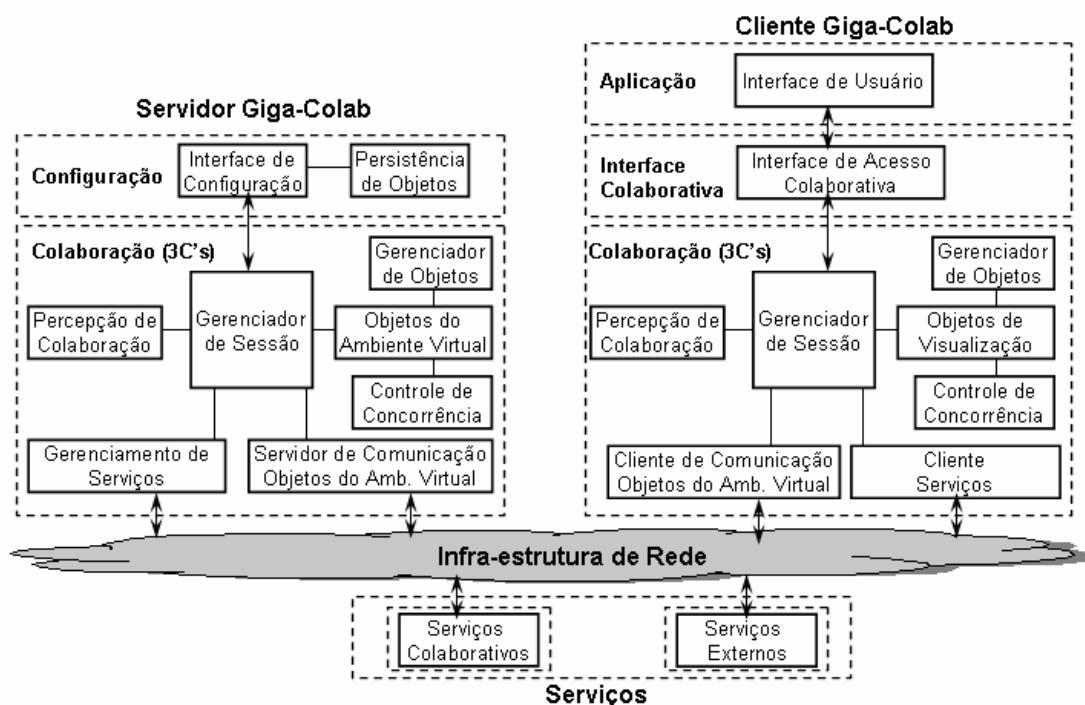


Figura 1. Arquitetura Geral da Solução

3.2.1. Servidor Giga-Colab (GC)

O Servidor GC representa a entidade capaz de fornecer um conjunto de funcionalidades de colaboração (subsistemas *Colaboração* e *Configuração*) que permite o acesso remoto do Cliente GC ao ambiente virtual. Abaixo uma descrição de cada módulo:

- Interface de Configuração – define uma interface de acesso para administração de todas as entidades do ambiente virtual colaborativo, definindo um modelo de dados e mecanismos de acesso para o mesmo. Este módulo é responsável pela pré-articulação das tarefas, ou seja, pelas ações necessárias para preparar a colaboração, normalmente concluídas antes do trabalho colaborativo se iniciar.

- Persistência de Objetos – representa a base de dados para armazenamento do estado dos objetos do ambiente virtual colaborativo e também das entidades relacionadas a estes objetos. Consiste na pós-articulação das tarefas ocorrida ao seu término.
- Gerenciador de Sessão – provê funcionalidades que permitem que o servidor GC gerencie dinamicamente os participantes de uma Sala, bem como a troca de dados e operações entre as entidades associadas. Para isso cria uma sessão colaborativa para cada Sala e controla todas as suas associações (Atividades, Usuários, Ferramentas, Repositórios e Serviços). Este módulo é responsável também por realizar o controle de acesso de cada participante através de um mecanismo de autenticação e autorização.
- Controle de Concorrência – podem ocorrer problemas relacionados à consistência, pois os estados dos objetos são replicados e passíveis de alterações de várias entidades de forma concorrente. A estrutura fornecida para o controle de concorrência é feita no nível de objetos através do mecanismo de bloqueios. Este controle é implementado na raiz da hierarquia de classes do ambiente virtual, sendo assim herdado por todas as outras classes. O controle por bloqueio funciona como um semáforo, pois para poder realizar uma alteração deve-se possuir o bloqueio da propriedade sendo alterada, ou seja, quando um usuário desejar, por exemplo, utilizar uma aplicação, ele deve inicialmente fazer o pedido de bloqueio de uma determinada propriedade para em seguida realizar o trabalho colaborativo, caso ele tenha permissão de efetuar tal operação. O servidor GC também prevê um outro controle de concorrência que utiliza o conceito de transformadas operacionais [Harada 2001]. Com este mecanismo é possível, por exemplo, que duas pessoas editem simultaneamente o mesmo texto, tornando-se útil quando o tempo de bloqueio é intolerável.
- Objetos do Ambiente Virtual – são entidades que habitam o ambiente virtual e possuem: (i) comportamento – associado à capacidade de gerar e responder eventos que resultam na execução de alguma tarefa e; (ii) propriedades – são responsáveis por armazenar o estado do objeto (identificação, localização, conteúdo).
- Gerenciador de Objetos – define as funcionalidades para os participantes salvarem os objetos no repositório e recuperá-los em uma próxima sessão. Este elemento é responsável por manter o ambiente virtual ativo, pois o Gerenciador de Objetos mantém um buffer contendo todos os objetos do ambiente virtual em uso no servidor (mantidos na memória). Quando um objeto do ambiente virtual é requisitado, este módulo é encarregado de localizar a classe que o implementa, instanciá-lo garantindo que não existam réplicas do objeto e recuperar os dados persistentes do objeto (dados que foram armazenados quando o objeto foi descarregado pela última vez).
- Percepção de Colaboração – permite que os participantes estejam cientes da presença de outros participantes na sessão e, além disso, ter a noção da atividade do grupo e do estado atual da tarefa. Os participantes podem ser representados por meio de uma lista, ícones ou então barra de status.
- Servidor de Comunicação Objetos do Ambiente Virtual – inclui funcionalidades responsáveis pelo tratamento da conexão e de eventos recebidos dos participantes. Consiste da implementação do protocolo de comunicação da colaboração executada entre os participantes conectados no ambiente virtual. Pode-se dizer que corresponde à execução do servidor colaborativo propriamente dito. Tem a função de enviar a

replicação das informações compartilhadas do ambiente virtual. A comunicação dá-se, basicamente, de duas formas: (i) entre objetos do ambiente virtual; (ii) entre objetos do ambiente virtual e os objetos de visualização correspondentes que estão localizados nos clientes. Todas as mensagens enviadas de e para os clientes são assíncronas. As mensagens enviadas de um objeto do ambiente virtual para um objeto de visualização normalmente transportam atualizações da representação para todos aqueles que estão no mesmo contexto de tarefa. Já as mensagens enviadas de um objeto de visualização para um objeto do ambiente virtual geralmente correspondem a uma ação do usuário.

- Gerenciamento de Serviços – este módulo possui uma interface de comunicação para gerenciar os serviços registrados no ambiente virtual colaborativo utilizando WSDL e SOAP. Pode realizar operações para sincronizar o serviço remoto de acordo com mudanças de configuração ou estado do ambiente virtual, assim como requisição de dados para objetos do ambiente virtual.

3.2.2. Cliente Giga-Colab (GC)

Representa os usuários finais do ambiente virtual colaborativo. O cliente Giga-Colab possui três subsistemas, sendo que o subsistema Colaboração é semelhante ao do servidor, mas executando funcionalidades relacionadas a um cliente e com representação dos objetos do ambiente virtual (módulo Objetos de Visualização), uma vez que uma instância de um objeto existe apenas no servidor e é replicado para os clientes.

O subsistema Interface Colaborativa tem o propósito de fornecer uma API (*Application Program Interface*) para possibilitar a integração de aplicações Java no ambiente colaborativo empregando o paradigma WYSIWIS (*What-You-See-Is-What-I-See*) [Borghoff e Schlichter 2000]. Adicionalmente, possui mecanismos específicos para notificação de eventos e mensagens do ambiente de colaboração que são úteis para o desenvolvimento de interfaces gráficas especializadas, destinadas exclusivamente à colaboração como uma janela de inspeção, por exemplo. E, por fim, o subsistema Aplicação comprehende a aplicação colaborativa a ser acoplada no cliente Giga-Colab.

3.2.3. Serviços

No contexto da arquitetura proposta, um serviço representa uma aplicação que fornece uma interface de acesso remoto utilizando tecnologias de serviços *Web*. Esta interface pode oferecer tanto acesso às funcionalidades do serviço (por exemplo, acesso a dados), como também acesso a funcionalidades de gerenciamento (por exemplo, envio de comando). Existem dois tipos de serviços utilizados na arquitetura: Colaborativo e Externo. O primeiro representa uma aplicação que disponibiliza dados para o *Servidor GC* complementar as funcionalidades do ambiente virtual colaborativo e, como exemplo, pode-se citar uma aplicação científica que fornece dados de forma transparente para uma ferramenta exibir gráficos estatísticos no ambiente colaborativo. É interessante notar que o *Servidor GC* não conhece os aspectos de implementação da aplicação científica citada no exemplo anterior, ele apenas acessa a interface de serviço provida, o mesmo acontece com aplicações de outra natureza. O segundo não tem ligação direta com as funcionalidades do ambiente colaborativo sendo que o *Cliente GC*, através de parâmetros de configuração das entidades, acessa-o de forma independente. Uma ilustração desse tipo de serviço é a requisição de um arquivo de

vídeo (representado por uma entidade do repositório) em um Cliente de Serviço do Cliente GC para um Serviço de vídeo sob demanda.

3.3. Funcionamento

Com o objetivo de explicar a integração dos serviços com o ambiente virtual, as figuras 2a, 2b e 2c representam alguns cenários de funcionamento da arquitetura proposta.

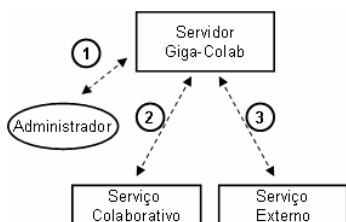


Figura 2a. Cenário 1

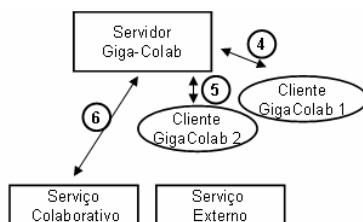


Figura 2b. Cenário 2

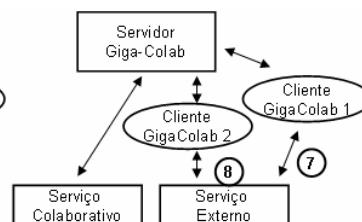


Figura 2c. Cenário 3

No Cenário 1 (Figura 2a) um usuário Administrador cadastra um Serviço Colaborativo e um Serviço Externo (seta 1). A partir dessa configuração o Servidor GC pode realizar operações de gerenciamento (por exemplo, sincronizar configurações ou passar parâmetros de acesso para os Clientes GC) ou acessar dados dos serviços (setas 2 e 3). Neste cenário todas as mensagens utilizam SOAP. No Cenário 2 (Figura 2b) é descrito o acesso de dois Clientes GC aos dados de um Serviço Colaborativo (setas 4 e 5), através do Servidor GC (seta 6) e as configurações realizadas pelo administrador no cenário anterior. Neste caso, para melhorar o desempenho da transmissão de dados, é possível utilizar qualquer protocolo de transporte, é necessário apenas o suporte do mesmo nas entidades que irão receber os dados. No último cenário (Figura 2c) é ilustrado o acesso dos Clientes GC a um Serviço Externo (setas 7 e 8), é possível notar que não é necessária uma comunicação com o Servidor GC, uma vez que não existe colaboração e os parâmetros de acesso aos Serviços Externos já foram repassados para os Clientes GC no momento da conexão com o Servidor GC (Cenário 2).

4. Protótipo do Ambiente Virtual Colaborativo

Com o objetivo de validar a arquitetura descrita na seção anterior, está sendo implementado um protótipo do Ambiente Virtual Colaborativo no contexto do Projeto GIGA-RNP. Até o momento, foram desenvolvidos os seguintes subsistemas: (i) Portal Web de Administração (configuração e administração), (ii) Servidor GC com principais módulos, (iii) Serviço Multimídia de Vídeo sob demanda, (iv) Serviço Multimídia de Videoconferência e (iv) Cliente GC com seus principais módulos. Nas próximas subseções será apresentada uma breve descrição de alguns desses subsistemas.

Em relação às opções de tecnologias foram escolhidas soluções de código aberto escritas em C++ para os módulos de transmissão e recepção multimídia e a plataforma de desenvolvimento Java para as demais funcionalidades. Entretanto, para o caso do subsistema do Cliente GC foi realizada uma integração entre as funcionalidades de C++ e Java através da tecnologia *Java Native Interface* (JNI).

4.1 Portal de Administração

O Portal de Administração permite ao administrador do sistema configurar informações pertinentes ao contexto do sistema de colaboração (pré-articulação das tarefas). Através

de um *browser* o administrador realiza o gerenciamento (cadastro, edição, remoção etc.) das principais entidades do sistema (salas, atividades, repositórios, ferramentas e serviços), seus relacionamentos e o controle de acesso aos recursos.

Como principais funcionalidades, podem-se descrever o gerenciamento de salas e atividades, que corresponde à configuração do sistema através do cadastro dessas duas entidades e associação dos usuários, repositórios de conteúdo, ferramentas e serviços relacionados a uma determinada atividade, além dos repositórios disponíveis no sistema colaborativo. Uma atividade corresponde a uma tarefa específica que estará associada a uma única sala. A interface Administração de Atividades permite ao administrador visualizar e administrar informações referentes às atividades associadas em uma determinada sala, por exemplo, nome da atividade, sala associada, data e hora de início e fim, descrição, participantes desta atividade, ferramentas utilizadas e repositórios disponíveis. As interfaces do Portal de Administração que permitem a manipulação de informações das Salas e Atividades podem ser visualizadas na Figura 3.

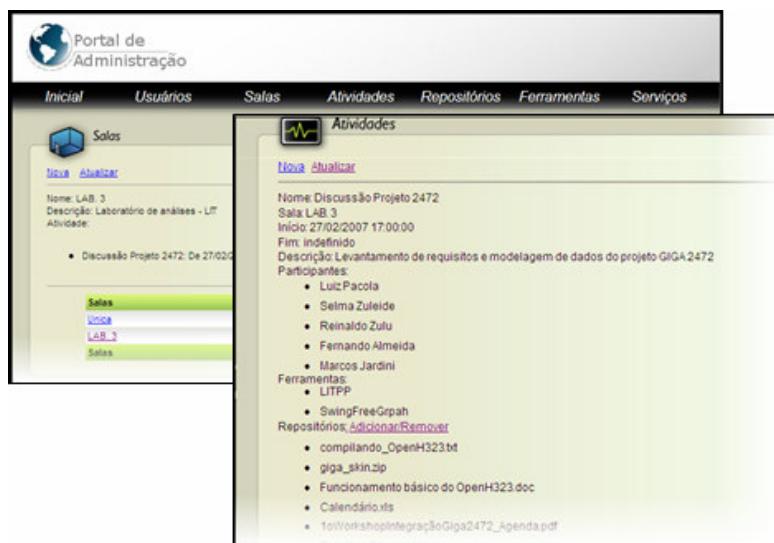


Figura 3. Interfaces Portal de Administração – Salas e Atividades

4.2 Serviços Multimídia

Os serviços multimídia desenvolvidos no protótipo utilizam duas aplicações desenvolvidas em trabalhos anteriores do grupo⁵ e permitem que usuários do ambiente virtual colaborativo visualizem sob demanda arquivos multimídia e realizem comunicação síncrona através de canais de vídeo, áudio e texto. Uma visão geral desses subsistemas é ilustrada na Figura 4 e a seguir são descritos resumidamente cada módulo:

- Cliente JSC – representa o módulo responsável pela recepção e exibição de arquivos multimídia disponibilizados pelo Servidor JSC, ou seja, um cliente de vídeo sob demanda. Executa em plataforma *Windows* e *Linux*, utiliza a biblioteca *ffmpeg* [FFmpeg] para tratamento multimídia na recepção (transporte, demultiplexação,

⁵ Java Stream Control (JSC) para vídeo sob demanda e Dynavideo Conference System (DCS) [Vasconcelos e Filho 2004].

decodificação e exibição), a *API JSA*⁶ [JSA 2004] para estabelecimento e controle das sessões de recepção (através de HTTP e RTSP) e bibliotecas padrão do J2SE para implementar a interface gráfica.

- Servidor JSC – módulo que implementa o armazenamento, recuperação e transmissão de arquivos multimídia representando um servidor de vídeo sob demanda. Executa apenas em ambiente *Linux* e utiliza a biblioteca *liveMedia* [LiveMedia] para tratamento multimídia na transmissão (leitura, multiplexação e transporte), a API JSA para gerenciamento das sessões de transmissão com os clientes e bibliotecas da Apache Axis para configuração remota com tecnologias de serviços Web [Axis].
- Cliente DCS – módulo que representa um ponto de comunicação em um sistema de conferência, permitindo a transmissão e recepção de vídeo, áudio e texto com os demais usuários. Sua implementação possui uma interface gráfica e de controle em Java e é baseada no projeto *Opal* [VoxGratia]⁷, que implementa o protocolo H.323 e SIP em C++ e com a utilização da biblioteca *PwLib* [VoxGratia] é portável para plataformas *Linux* e *Windows*.
- Servidor DCS – módulo que implementa um servidor para vídeo conferência centralizada baseada no protocolo H.323 e com suporte a serviços de um gerenciador de chamadas (*Gatekeeper* ou GK) e uma unidade de controle multiponto (*Multipoint Control Unit* ou MCU). O servidor executa em ambiente *Linux* utilizando aplicações (*GnuGk* e *OpenMCU*, respectivamente para o GK e a MCU) e bibliotecas (*OpenH323* e *PwLib*) desenvolvidas em C++ no projeto *VoxGratia*. Adicionalmente, foram desenvolvidas aplicações para controle remoto dos dois serviços com Apache Axis.

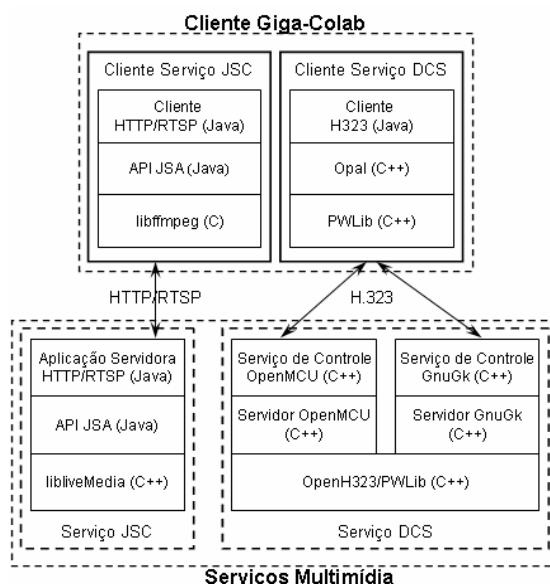


Figura 4 - Implementação de Serviços Multimídia

4.3. Interface Gráfica do Cliente Giga-Colab

⁶ A JSA é tem como objetivo fornecer para aplicações multimídia em rede (particularmente, no lado do servidor) uma interface de software unificada e padronizada para gerenciamento de fluxos de mídias.

⁷ VoxGratia é um projeto de código-fonte aberto com implementações dos padrões SIP e H.323.

O Cliente GC do protótipo apresenta uma interface gráfica em Java para os usuários finais do sistema, permitindo interação entre eles, visualização de informações, bem como realizar tarefas em comum. Esta interface é composta por três módulos básicos (Figura 5): Janela de Inspeção, Interface de Colaboração e Interface de Comunicação.

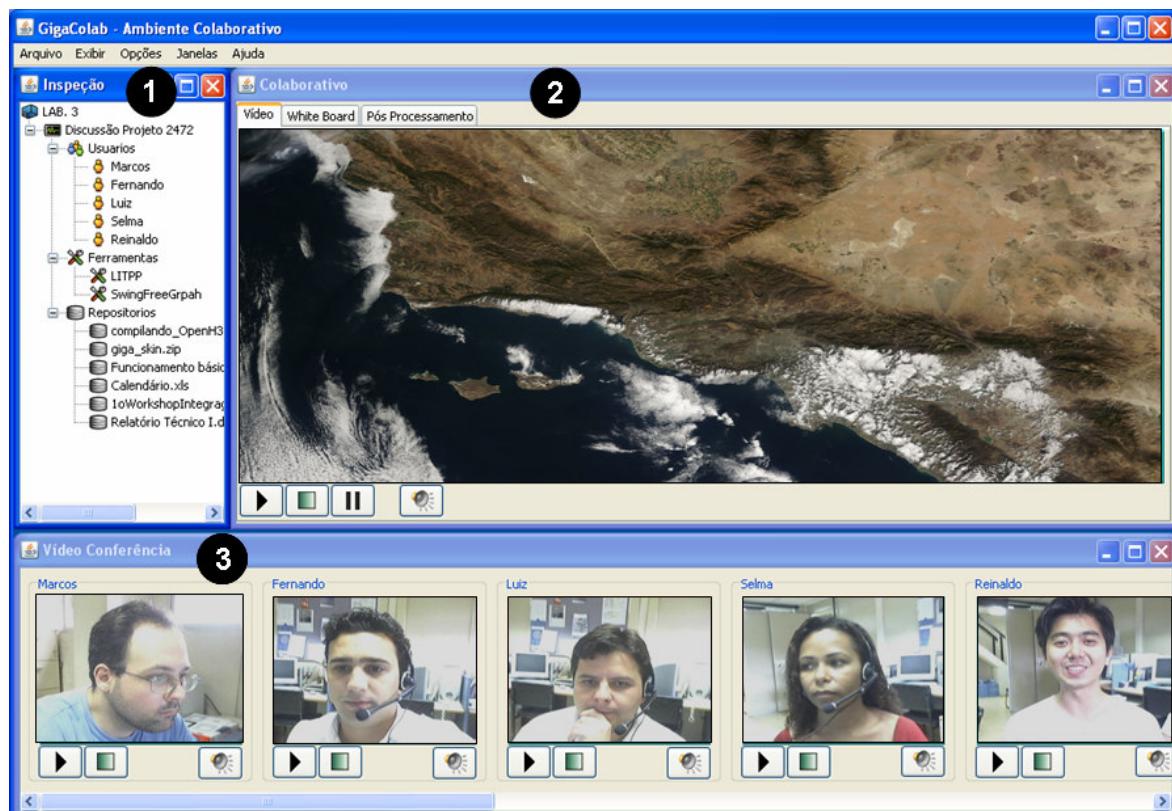


Figura 5 – Interface Gráfica do Cliente Giga-Colab

A Janela de Inspeção (Indicação 1) permite aos usuários do sistema visualizar informações e estado de entidades referentes a uma determinada Sala. Dentre essas informações destacam-se: nome da atividade, usuários, ferramentas associadas, bem como o repositório de conteúdo disponível para colaboração. A Janela de Inspeção também provê mecanismos de notificação de eventos, como por exemplo, mudança de estado (presente, ausente, etc.) de um determinado usuário da Atividade, ou até mesmo a indicação se um usuário está utilizando uma determinada aplicação colaborativa através do pedido de bloqueio.

A Interface de Colaboração (Indicação 2) dispõe de informações relevantes ao contexto geral do sistema, pois é através desta interface que os participantes apresentam seus documentos, vídeos e discussões de forma totalmente colaborativa. O usuário pode, por exemplo, visualizar vídeos sob demanda (Cliente JSC), utilizar o whiteboard ou uma aplicação de forma colaborativa com o uso do controle de concorrência. Tais informações são apresentadas em suas respectivas abas.

A Interface de Comunicação (Indicação 3) possibilita a configuração e visualização das funcionalidades de comunicação (áudio, vídeo e texto) de cada participante da colaboração (Cliente DCS). Atualmente, esta interface suporta até 16 (dezesseis) participantes simultâneos.

5. Considerações Finais

Este artigo apresentou a Giga-Colab, uma proposta de arquitetura de um ambiente virtual colaborativo. O objetivo foi reutilizar uma solução já existente (jVE) e permitir a integração dinâmica de aplicações multimídia através da utilização de tecnologias de serviços *Web* e padrões abertos. Como forma de validação, a arquitetura foi instanciada para um protótipo que está em desenvolvimento no contexto do Projeto GIGA-RNP. Duas aplicações multimídia – vídeo sob demanda e videoconferência – foram utilizadas no sistema colaborativo e demonstraram a funcionalidade de integração.

O projeto da arquitetura trouxe os seguintes benefícios: (i) reutilização de uma infra-estrutura que facilite o desenvolvimento de aplicações colaborativas; (ii) fraco acoplamento na integração de aplicações externas ao ambiente colaborativo; e (iii) possibilidade de interoperabilidade com aplicações multimídia já existentes. Adicionalmente, também se pode citar como contribuição o emprego de funcionalidades avançadas de transmissão e recepção de vídeo para apoiar atividades colaborativas.

No contexto do projeto Giga-RNP, estão em andamento os seguintes trabalhos: (i) realização de experimentos para análise da solução na rede Giga; (ii) refinamento e adição de novas funcionalidades para o protótipo (por exemplo, vídeo ao vivo) e; (iii) desenvolvimento de serviços para o LIT/INPE (Laboratório de Integração e Testes do Instituto Nacional de Pesquisas Espaciais), de forma a possibilitar o uso de ferramentas colaborativas para acompanhamento de atividades de testes na construção de satélites.

Como trabalhos futuros, pode-se citar a possibilidade de uso de um modelo de política de colaboração para definir a semântica de integração entre os serviços [Gomes et al. 2006] e evolução da arquitetura para adição de novas características de um modelo orientado a serviços [Jorstad et al. 2005].

Referências

- AccessGrid, “Access Grid”, Disponível em: <http://www.accessgrid.org>.
- Agora, “Agora – The Online Meeting Tool”, Lancaster University, Disponível em: <http://agora.lancs.ac.uk>.
- Alonso, G. et al. (2004), “Web Services: Concepts, Architectures and Applications”, Springer.
- Axis, A. Disponível em: <http://ws.apache.org/axis>
- Beavers, J. et al. (2004), “The Learning Experience Project: Enabling Collaborative Learning with ConferenceXP”, Technical Report MSR-TR-2004-42.
- Borghoff, U. M. e Schlichter, J. H. (2000), “Computer-Supported Cooperative Work: Introduction to Distributed Applications”, Springer, USA.
- Ellis, C. A., Gibbs, S. J. e Rein, G. L. (1991), “Groupware: some issues and experiences”, In *Communications of the ACM*, v.34, n., p.35-58.
- FFmpeg, Disponível em: <http://ffmpeg.mplayerhq.hu>.

- Fuchs, H. e Farber, N. (2005), "ISMA Interoperability and Conformance," In *IEEE Multimedia*, 2(2), p. 96-102.
- Gerosa, M. A. (2006), "Desenvolvimento de Groupware Componentizado com Base no Modelo 3C de Colaboração", Tese de Doutorado, Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio).
- Gomes, R. L., Hoyos-Rivera, G. J. H. e Courtiat, J. P. (2006), "Um Ambiente para Integração de Aplicações Colaborativas". In: Anais do Simpósio Brasileiro em Sistemas Colaborativos.
- Harada, C. Y. (2001), "Arquitetura para suporte a aplicações colaborativas", 317p. Dissertação de Mestrado, Escola Politécnica, Universidade de São Paulo, São Paulo.
- H.323, ITU Recommendation (1999), "Packet-base multimedia communications systems".
- Java Stream Assembly (2004), "API Programmer's Guide", Sun Microsystems Inc.
- Jorstad, I., Dustdar, S., Thanh, D. V. (2005) "A Service Oriented Architecture Framework for Collaborative Services", In: Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise, pp. 121-125.
- Kahler, H., Mørch, A., Stiemerling, O., Wulf, V. (2000) "Introduction to the Special Issue on Special Issue on Tailorable Systems and Cooperative Work" Computer Supported Cooperative Work: The Journal of Collaborative Computing, v.9, n.1. pp.1-4.
- Steinmetz, R. e Nahrstedt, K. (2004) "Multimedia Systems" X.Media.publishing, Springer.
- Licea, G. (2006), "Supporting reusability in fixed and mobile groupware applications", In *International Journal of Computers and Applications*, V.28(2), p. 99-111. ACTA Press.
- LiveMedia Library, Disponível em: <http://www.live555.com/liveMedia>
- Rosenberg, J. et al. (2002), "SIP: Session Initiation Protocol", RFC 3261, Internet Engineering Task Force, Disponível em: <http://www.ietf.org/rfc/rfc3261.txt>.
- Vasconcelos, M. A. V. M. e Filho, G. L. S. (2004) "Dynavideo Conference System, Um Sistema de Videoconferência H.323". In Anais do WebMedia/LA-Web Joint Conference.
- Voxgratia Open-Source VoIP, Disponível em: <http://www.voxgratia.org>
- Wu, W. et al. (2004) "Design and Implementation of a Collaboration Web-services system", In *Journal of Neural, Parallel & Scientific Computations*, Volume 12.