

UNIVERSIDADE FEDERAL DE SANTA CATARINA

**PROGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA ELÉTRICA**

**UM ESTUDO SOBRE
AMBIENTES COLABORATIVOS MULTIMÍDIA
COM CONTROLE DE PALAVRA**

Dissertação submetida à
Universidade Federal de Santa Catarina
como parte dos requisitos para a
obtenção do grau de Mestre em Engenharia Elétrica.

FABIANO BACHMANN

Florianópolis, abril de 2004.

**UM ESTUDO SOBRE
AMBIENTES COLABORATIVOS MULTIMÍDIA
COM CONTROLE DE PALAVRA**

Fabiano Bachmann

‘Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Engenharia Elétrica, Área de Concentração em *Automação e Sistemas*, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.’

Jean-Marie Farines, Dr.

Orientador

Coordenador do Programa de Pós-graduação em Engenharia Elétrica

Banca Examinadora:

Jean-Marie Farines, Dr.

Presidente

Carlos Montez, Dr.

Roberto Willrich, Dr.

Não basta adquirir a sabedoria, é preciso também usá-la.

(Cícero, escritor romano – 106-43 a.C.)

*Aos meus pais,
Vitorino e Lorita Bachmann.*

Agradecimentos

Inicialmente agradeço a Deus que me concedeu saúde, paz, amor e todas condições para realizar este trabalho.

Agradeço minha mãe Lorita, exemplo de força, garra e perseverança, pelas sábias palavras de incentivo e apoio, proferidas, sempre, nas horas certas.

Pelo amor, compreensão e apoio durante os anos que está ao meu lado, agradeço minha querida esposa, Ediane.

Agradeço meu orientador, Prof. Jean-Marie Farines, pelos ensinamentos, incentivo e orientação ao longo deste trabalho.

Sou grato aos membros da banca examinadora, que contribuíram com opiniões e sugestões para o trabalho.

Finalmente, agradeço ao amigo, colega de trabalho e professor, Evande João da Silva, pelos conselhos, ensinamentos e palavras de incentivo proferidas durante estes últimos dois anos.

Muito obrigado a todos que contribuíram para este trabalho.

Resumo da Dissertação apresentada à UFSC como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica

UM ESTUDO SOBRE AMBIENTES COLABORATIVOS MULTIMÍDIA COM CONTROLE DE PALAVRA

Fabiano Bachmann

Abril/2004

Orientador: Jean-Marie Farines, Dr.

Área de Concentração : Automação e Sistemas.

Palavras-chave: Coordenação, Sistemas Multimídia, Controle de Palavra, Ambientes Colaborativos, Multicast.

Número de Páginas: 141

Os ambientes colaborativos multimídia (ACM) encontram-se na confluência de várias áreas: sistemas computacionais, trabalho colaborativo, multimídia, protocolos e comunicação de redes, comunicação interpessoal, entre outros, tornando-os, dessa forma, bastante específicos a cada cenário e o seu desenvolvimento uma tarefa extremamente complexa. Por isso, tais ambientes deixam de ser compostos por grandes ferramentas, proprietárias, monolíticas e limitadas ao tipo de mídia e tamanho da sessão, passando a serem formados por aplicações e ferramentas modulares, levando em conta os diferentes estilos e cenários de colaboração. Neste contexto, foram realizadas pesquisas, estudos e testes das principais ferramentas existentes para compor um ACM para o trabalho colaborativo multimídia síncrono. Por causa das especificidades de cada cenário e da variedade de estilos de interação existente foi adotado o modelo LWS (*Light-weight session*) para modelar e conceber um ACM, que passa a ser visto como um conjunto de aplicativos modulares que atuam de forma coordenada e controlada para fornecer as funcionalidades requeridas e adequando-se facilmente às especificidades de cada cenário. Por outro lado, o acesso a recursos compartilhados nestes ambientes causa competição e conflitos, tornando necessárias políticas e mecanismos que implementem controle de palavra a fim de permitir a colaboração e utilização racional e otimizada destes recursos. Foi implementado protótipo para validar o modelo de ACM proposto com suporte a múltiplas políticas de controle de palavra tendo sido também, para tal, definido um conjunto de diretivas de controle, requisitos mínimos para infra-estrutura de comunicação e três cenários característicos para a aplicação do protótipo e análise das diferentes políticas de controle de palavra.

Abstract of Dissertation presented to UFSC as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering.

A STUDY ABOUT MULTIMEDIA COLLABORATIVE ENVIRONMENTS WITH FLOOR CONTROL

Fabiano Bachmann

April/2004

Advisor: Jean-Marie Farines, Dr.

Area of Concentration: Automation and Systems.

Keywords: Coordination, Multimedia Systems, Floor Control, Collaborative Environment, Multicast.

Number of pages: 141

Multimedia collaborative environments (MCE) are in the confluence of several areas: computing systems, collaborative work, multimedia, interpersonal communication, networking protocols and communication. This characteristic turns them quite specific to each context, and its development turns a complex task. Therefore, such environments, instead of composed by monolithic applications and limited to the media type and size of the session, becoming modular applications and tools, taking into account the different styles and scenarios of collaboration. In this work, researches, studies and tests of the main existent tools were accomplished to compose an MCE for synchronous collaborative multimedia work. Because of the particular characteristic of each scene and of the variety of styles of existent interaction, the LWS (Light-weight session) model was adopted to make a representation an MCE, becoming a group of modular application, acting in a coordinated and controlled way to supply the requested functionalities and being adapted easily to the particular characteristic of each scene. In addition, the race condition caused by the shared access to resources in these environments, raise the importance of policies and mechanisms to control this resources. It is necessary to implement floor control in order to allow the collaboration and optimized use of these resources. In this work, a prototype was implemented to validate the proposed model. In addition, was defined: a group of control directives, minimum requirements for communication infrastructure and three characteristic scenarios for the application, test and analysis of the different floor control policies.

SUMÁRIO

1		1
INTRODUÇÃO		1
1.1	OBJETIVOS GERAIS DO TRABALHO	3
1.2	OBJETIVOS ESPECÍFICOS DO TRABALHO	4
1.3	ORGANIZAÇÃO DO TEXTO	4
2		6
AMBIENTES COLABORATIVOS MULTIMÍDIA		6
2.1	GROUPWARE E CSCW	8
2.2	DEFINIÇÃO DE AMBIENTE COLABORATIVO MULTIMÍDIA	9
2.3	A EVOLUÇÃO DOS AMBIENTES COLABORATIVOS MULTIMÍDIA	10
2.4	DIFICULDADES EM AMBIENTES COLABORATIVOS MULTIMÍDIA	11
2.5	CONCLUSÕES DO CAPÍTULO	15
3		17
INFRA-ESTRUTURA DE COMUNICAÇÃO PARA AMBIENTES COLABORATIVOS MULTIMÍDIA		17
3.1	O PROBLEMA DA TRANSMISSÃO MULTIMÍDIA E DE SUA QUALIDADE	18
3.2	CARACTERÍSTICAS DO TRÁFEGO MULTIMÍDIA	20
3.3	REQUISITOS PARA INFRA-ESTRUTURA DE COMUNICAÇÃO	24
3.4	ENDEREÇAMENTO DE GRUPO	27
3.5	COORDENAÇÃO EM ACMS E INFRA-ESTRUTURA DE COMUNICAÇÃO	40
3.6	CONCLUSÕES DO CAPÍTULO	46
4		48
FERRAMENTAS E APLICAÇÕES PARA AMBIENTES COLABORATIVOS MULTIMÍDIA		48
4.1	TESTES E AVALIAÇÃO DE FERRAMENTAS E APLICATIVOS	50
4.2	PLATAFORMAS COMPLETAS	51
4.3	APLICATIVOS DE MÍDIA	54
4.4	FERRAMENTAS DE APOIO	55
4.5	CONCLUSÕES DO CAPÍTULO	58
5		61
COORDENAÇÃO E CONTROLE DE PALAVRA EM AMBIENTES COLABORATIVOS MULTIMÍDIA		61
5.1	COORDENAÇÃO EM CONFERÊNCIAS MULTIMÍDIA	63
5.2	O MODELO LWS (LIGHT-WEIGHT SESSION)	65
5.3	CONTROLE DE PALAVRA	70
5.4	POLÍTICAS E MECANISMOS PARA CONTROLE DE PALAVRA	81
5.5	CARACTERÍSTICAS E REQUISITOS DO CONTROLE DE PALAVRA	75
5.6	MODELO DE CONTROLE DE PALAVRA	72
5.7	CONCLUSÕES DO CAPÍTULO	85
6		88
PROTÓTIPO DE CONTROLE DE PALAVRA PARA AMBIENTES COLABORATIVOS MULTIMÍDIA		88
6.1	MODELANDO UM AMBIENTE COLABORATIVO MULTIMÍDIA	90
6.2	CONTROLE DE PALAVRA PARA AMBIENTE COLABORATIVO MULTIMÍDIA	97
6.3	PROTÓTIPO PARA CONTROLE DE PALAVRA	98
6.4	CONCLUSÕES DO CAPÍTULO	130
7		132
CONCLUSÕES		132

LISTA DE FIGURAS

<i>Figura 1: Classificação das interações quanto ao tempo e espaço em que ocorrem</i>	9
<i>Figura 2: Transmissão síncrona de mídias contínuas</i>	19
<i>Figura 3: Largura de banda utilizada por diferentes tipos de aplicações</i>	25
<i>Figura 4: Pilha de protocolos da Internet e Classes de Endereços IP</i>	29
<i>Figura 5: Comunicação unicast</i>	30
<i>Figura 6: Comunicação multicast</i>	30
<i>Figura 7: Roteamento Multicast</i>	36
<i>Figura 8: Formato do pacote IGMP</i>	37
<i>Figura 9: Túneis Multicast</i>	39
<i>Figura 10: Arquitetura do protocolo RTP. O Protocolo de tempo real é composto por um conjunto interdependente de especificações arranjadas em uma classe hierárquica para suportar o modelo ALF - application-level framing</i>	68
<i>Figura 11: Facilidade de integração, adaptação e evolução de um ACM</i>	93
<i>Figura 12: Ambiente Colaborativo Multimídia</i>	96
<i>Figura 13: Diagrama de Classes UML do Protótipo</i>	108
<i>Figura 14: Classe ControleDePalavra que implementa a interface LrmpEventHandler</i>	109
<i>Figura 15: Implementação do método enviaMensagemControleDePalavra da classe ControleDePalavra, responsável pelo envio de dados</i>	111
<i>Figura 16: Implementação do método processData</i>	111
<i>Figura 17: Definição do Perfil para o LRMP</i>	112
<i>Figura 18: Criação de um objeto Lrmp e o início da recepção de dados</i>	113
<i>Figura 19: Função main do protótipo</i>	113
<i>Figura 20: Informações carregadas pela primitiva RequisitaPalavra</i>	114
<i>Figura 21: Transmissão da Primitiva de Controle DefinePolítica</i>	117
<i>Figura 22: Recepção das informações carregadas pela Primitiva de Controle DefinePolítica nos sítios 2 e 3</i>	117
<i>Figura 23: Transmissão da Primitiva de Controle CriaPalavra a partir do sítio 1 (moderador)</i>	118
<i>Figura 24: Recepção da Primitiva de Controle CriaPalavra pelos sítios 2 e 3</i>	118
<i>Figura 25: Transmissão da Primitiva de Controle AlteraPalavra a partir do sítio 1</i>	119
<i>Figura 26: Recepção da Primitiva de Controle AlteraPalavra nos sítios 2 e 3</i>	119
<i>Figura 27: Transmissão da Primitiva de Controle RequisitaPalavra a partir do sítio 2</i>	120
<i>Figura 28: Recepção da Primitiva de Controle RequisitaPalavra nos sítios 1 e 3</i>	120
<i>Figura 29: Transmissão da Primitiva de Controle ConcedePalavra a partir do sítio 1</i>	120
<i>Figura 30: Recepção da Primitiva de Controle ConcedePalavra nos sítios 2 e 3</i>	121
<i>Figura 31: Transmissão da Primitiva de Controle RequisitaPalavra a partir no sítio 3</i>	121
<i>Figura 32: Recepção da Primitiva de Controle RequisitaPalavra nos sítios 1 e 2</i>	122
<i>Figura 33: Transmissão da Primitiva de Controle LiberaPalavra a partir no sítio 2</i>	122
<i>Figura 34: Recepção da Primitiva de Controle LiberaPalavra nos sítios 1 e 3</i>	122
<i>Figura 35: Transmissão da Primitiva de Controle ConcedePalavra a partir no sítio 1 (controlador)</i>	123
<i>Figura 36: Recepção da Primitiva de Controle ConcedePalavra nos sítios 2 e 3</i>	123
<i>Figura 37: Controle Realizado Localmente no Sítio do Receptor</i>	126

LISTA DE TABELAS

<i>Tabela 1 :Largura de banda utilizada por alguns padrões de mídias típicos</i>	23
<i>Tabela 2: Grupos multicast permanentes</i>	33
<i>Tabela 3: Níveis de conformidade dos grupos multicast</i>	34
<i>Tabela 4: Tipos de recursos e características de manipulação</i>	78
<i>Tabela 5: Variáveis e valores utilizados na troca de mensagens ASCII entre os agentes de controle e coordenação de uma sessão</i>	105

Capítulo 1

INTRODUÇÃO

Avanços tecnológicos vêm permitindo a evolução e desenvolvimento de aplicações distribuídas cada vez mais sofisticadas. Outrossim, mecanismos e tecnologias dominadas tornam-se, rapidamente, insuficientes para atender a demanda por tais aplicações.

Ambientes colaborativos multimídia (ACM) vêm sendo utilizados nas mais diversas áreas, para reuniões, educação à distância, simulações, desenvolvimento colaborativo de projetos, telemedicina, entre outros.

Assim, observa-se que os ambientes colaborativos multimídia encontram-se na confluência de várias áreas: sistemas computacionais, trabalho colaborativo, multimídia, protocolos e comunicação de redes, comunicação interpessoal, entre outros, tornando-os, dessa forma, bastante específicos a cada cenário e o seu desenvolvimento uma tarefa extremamente complexa.

Vários modelos, mecanismos e abordagens vêm sendo desenvolvidos, apresentados e adotados para permitir os ambientes

colaborativos multimídia mais sofisticados que, por sua vez, exigem um grande conjunto de funcionalidades disponíveis como, interfaces gráficas amigáveis, CODECs (codificadores e decodificadores) de áudio e vídeo próprios para cada situação, aplicações para compartilhamento de dados, módulos para gerir preferências de cada usuário, resolução de nomes, acesso a bases de dados e coordenação para colaboração síncrona e assíncrona, para tornar o ambiente cada vez mais próximo de um ambiente colaborativo presencial.

A difusão de ACMs vem mostrando-se, cada vez mais, uma tendência, superando dificuldades de espaço e tempo, compartilhando recursos e tornando possíveis colaborações com a qualidade e funcionalidades próximas das de um encontro presencial. Enquanto no início os ACM eram compostos por aplicações proprietárias, monolíticas e limitadas ao tipo de mídia e tamanho da sessão, hoje, as novas aplicações suportam colaborações entre múltiplas partes e multimodo em sessões pequenas ou grandes, públicas ou privadas, atendendo especificidades de interação e colaboração de cada cenário.

É evidente que limitações dos recursos compartilhados disponíveis para os usuários de ambientes colaborativos multimídia, muito comuns em ACMs, causam competição, disputa e conflitos, tornando necessárias políticas e mecanismos que implementem a coordenação e controle, a fim de permitir a colaboração e utilização racional e otimizada dos recursos.

Entretanto, comparados aos avanços e trabalhos realizados em tecnologias como roteamento de grupo, comunicação confiável de grupo, algoritmos de compressão de mídias e tecnologias de acesso, observa-se que não existem trabalhos na mesma proporção acerca de assuntos relacionados à coordenação de grupos para tais ambientes colaborativos multimídia. Além disso, com o crescimento da demanda por tais

ambientes, a lacuna existente na área de controle e coordenação torna-se um fator limitante para a evolução e desenvolvimento destes ambientes.

1.1 Objetivos Gerais do Trabalho

Após levantamento, estudo, análise e testes das principais ferramentas existentes que poderiam vir a contribuir para o desenvolvimento de um ACM concluiu-se que a maior limitação para o desenvolvimento de tal ambiente, para uma grande variedade de cenários, atendendo as especificidades que lhes são inerentes, está no suporte à coordenação de grupo.

Observa-se também que, dentre todos os aspectos necessários e fundamentais para o desenvolvimento de um ACM, a coordenação e controle dos recursos compartilhados num grupo têm especial importância, pois irão reger todo conjunto de recursos disponíveis e como serão utilizados, tendo impacto direto sobre como a sessão ou conferência irá transcorrer e na utilização eficaz dos recursos.

Neste sentido, adotar-se-á um modelo para comceber e desenvolver ACMs flexíveis que se adaptem e contemplem a multiplicidade de cenários e suas especificidades, suportando as várias classes de interações e colaboração exigidas para cada contexto. Este modelo e infra-estrutura de comunicação são requisitos mínimos para a implementação de um protótipo que permitirá demonstrar e avaliar diferentes políticas de controle de palavra em ACMs para o trabalho colaborativo multimídia síncrono.

Para validar este modelo e infra-estrutura básica de comunicação o protótipo implementado demonstrará a troca de mensagens contendo as primitivas de controle de palavra sobre a infra-estrutura do serviço de

controle flexível apresentada. Este protótipo deverá integrar-se facilmente a aplicações e ferramentas multimídia já existentes, podendo ser empregado em sessões com diferentes estilos de interação e suportar múltiplas políticas de coordenação.

1.2 Objetivos Específicos do Trabalho

Como objetivos específicos figuram a pesquisa, estudo, teste, análise e classificação de aplicações e ferramentas existentes de áudio e videoconferência, quadro branco, votação, gerenciamento de grupos, compartilhamento de dados, etc., que possam contribuir para compor um ACM; identificação das principais necessidades e carências de tais aplicações e ferramentas já existentes para comporem ambientes para serem utilizados em cenários específicos; definição de características gerais e específicas necessárias e mais importantes para as ferramentas e aplicações modulares que comporão um ACM; demonstração e aplicação de diferentes políticas e mecanismos de controle de palavra em ACMs em diferentes cenários (como reuniões de pequenos e grandes grupos, ambientes de ensino a distância e desenvolvimento colaborativo); e verificação da facilidade de integração do protótipo e infra-estrutura para prover o serviço de controle de palavra às ferramentas e aplicações multimídia já existentes para o trabalho colaborativo síncrono.

1.3 Organização do Texto

O presente trabalho está organizado em 7 capítulos. O capítulo 1 apresenta uma introdução ao problema e objetivos do trabalho.

No capítulo 2 são apresentados conceitos e considerações relacionadas a ambientes colaborativos multimídia.

No capítulo 3 são abordados aspectos importantes para o provimento de uma infra-estrutura de comunicação para os ambientes colaborativos multimídia.

No capítulo 4 são apresentados resultados de pesquisas, análises e testes das ferramentas e aplicações multimídia gratuitas consideradas para compor um ACM que suporte trabalho colaborativo síncrono. Ao final deste capítulo são apresentados resultados da avaliação e identificadas as principais lacunas existentes.

No capítulo 5 é abordado o problema da coordenação e o controle de palavra em ambientes colaborativos multimídia. O foco deste capítulo recai sobre o controle de palavra para tais ambientes.

No capítulo 6 são apresentados um modelo para o desenvolvimento de ACMs e infra-estrutura básica para prover serviço flexível de controle de palavra em sessões de trabalho colaborativo síncrono.

Finalmente, no capítulo 7 são apresentadas as conclusões acerca do trabalho e perspectivas para trabalhos futuros.

Capítulo 2

AMBIENTES COLABORATIVOS MULTIMÍDIA

Encontros presenciais são baseados no compartilhamento de recursos, num mesmo local e horário, onde as mensagens visuais, auditivas e gestuais são expressas e transmitidas de acordo com políticas definidas e recursos disponíveis.

O trabalho colaborativo suportado por computador (CSCW – *Computer Supported Collaborative Work*) se propõe a apresentar soluções e alternativas para as limitações de espaço e tempo dos encontros presenciais, permitindo que informações sejam compartilhadas a partir de locais geograficamente distribuídos ao mesmo tempo (síncrono) ou tempos diferentes (assíncrono).

Para permitir conferências remotas, compartilhamento de recursos, e colaboração entre os usuários é necessário lançar mão de vários aplicativos e recursos e tornar estes cenários o mais próximo dos presenciais.

Considerando tais ambientes para CSCW, as maiores preocupações em relação a plataformas de hardware e software até o momento têm sido relacionadas à interoperabilidade, compatibilidade e portabilidade. Porém, aplicando-se o escopo do uso dos softwares em direção à “cooperação” em tais ambientes distribuídos introduz-se o conceito de “colaborabilidade”. O objetivo de qualquer software com capacidade de colaboração, além de suportar o conceito de “*What You See Is What I See*” (WYSIWIS), é compartilhar as informações, de forma eficaz, durante o processo de colaboração.

Os encontros presenciais permitem conversações e atividades não verbais como expressões do olhar e da face e gestos. Já num encontro virtual, o espaço de cooperação é criado de forma eletrônica, onde os canais de comunicação implícita muitas vezes não são suportados devido à visão limitada do espaço e eventos remotos, ou seja, falta conhecimento mútuo das atividades individuais e do grupo.

O CSCW, especialmente a telecolaboração multimídia, busca viabilizar formas mais poderosas de cooperação e coordenação, quando comparadas às formas rudimentares de colaboração da primeira geração *groupware*. Do ponto de vista do usuário, uma aplicação colaborativa deve atender as necessidades deste, oferecendo funcionalidades e qualidade mínimas para garantir a colaborabilidade e satisfação do mesmo. Do ponto de vista de um sistema distribuído, as interações entre usuários e destes com os recursos distribuídos, como, por exemplo, uma base de dados, são baseadas em políticas de Acordo, Concorrência e Recuperação,

que implicam propriedades de atomicidade, consistência, isolamento e durabilidade. Já do ponto de vista da rede, muitos esforços têm sido despendidos para oferecer mecanismos de comunicação de grupo mais eficientes, porém, sem muita consideração com a interatividade do usuário final.

2.1 Groupware e CSCW

O uso do computador para suportar trabalho colaborativo tem muito em comum com o uso de computadores para suportar trabalho individual. Historicamente, os sistemas e aplicações computacionais multi-usuário vêm tentando isolar os usuários e seus trabalhos para prover a ilusão de que cada sistema apenas possui um único usuário [01]. Entretanto, o trabalho colaborativo faz os usuários, cada vez mais, defrontar-se com situações onde são imprescindíveis a comunicação, interação e colaboração direta, não apenas com o computador, mas com outros usuários ao mesmo tempo. A noção de *groupware* e a área de CSCW buscam prover respostas para este cenário.

A definição de *groupware*, considerada neste trabalho é a apresentada em [01]:

“Sistemas baseados em computadores que suportam grupos de pessoas engajadas em uma tarefa comum (ou objetivo) e que oferecem uma interface para um ambiente compartilhado”.

Assim, a dimensão do “ambiente” na definição acima pode ser decomposta em tempo e espaço. Mesmo nas interações do dia-a-dia (interações sociais, por exemplo) podem ser classificadas segundo estas dimensões, mostradas na Figura 1 [01]:

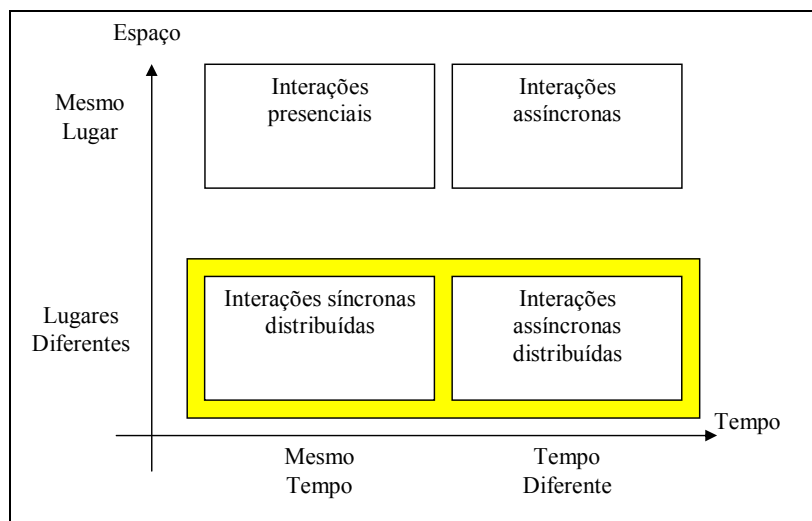


Figura 1: Classificação das interações quanto ao tempo e espaço em que ocorrem

A preocupação do *groupware* está em oferecer uma interface para o ambiente compartilhado, já que “o mesmo lugar”, destacado na Figura 1, não é possível para os cenários onde os participantes estão em locais geograficamente distribuídos. Obviamente, isso implica conhecimento mútuo entre os participantes e de seus engajamentos com as tarefas e ambiente [01].

Assim, *groupware* síncrono se refere a sistemas baseados em computadores através dos quais grupos de pessoas interagem “ao mesmo tempo” para realizar uma tarefa comum.

2.2 Definição de Ambiente Colaborativo Multimídia

A comunicação em encontros presenciais é baseada em compartilhamento do mesmo espaço e intervalo de tempo, permitindo a comunicação através de mensagens gestuais, auditivas e ilustradas. [02]

Entretanto, o trabalho colaborativo suportado por computador pode ajudar a superar limitações de espaço e tempo existentes nos encontros não presenciais e permitir que pessoas interajam e compartilhem informações a partir de lugares geograficamente distribuídos, no mesmo ou em tempos diferentes, através de um ambiente que permita este compartilhamento e colaboração. Outrossim, o objetivo de qualquer *groupware* é integrar as comunicações visual, auditiva e textual em um ambiente de software logicamente consistente, integrando informações locais e remotas, públicas e privadas [02].

Neste sentido, um ambiente colaborativo multimídia (ACM) pode ser entendido como um sistema que permite a colaboração de grupos geograficamente distribuídos com suporte multimídia [03], ou seja, uma interpretação virtual de um ambiente de colaboração “real” formado através de uma rede de computadores, onde um grupo de participantes executa as mesmas aplicações [02]. Um grupo, por sua vez, é uma associação de pessoas em torno de um objetivo em comum.

2.3 A Evolução dos Ambientes Colaborativos Multimídia

A difusão de ambientes colaborativos multimídia vem mostrando-se, cada vez mais, uma tendência, superando dificuldades de espaço e tempo, compartilhando recursos e tornando possíveis colaborações com a qualidade próxima da de um encontro presencial. Enquanto no início estes ambientes eram proprietários, monolíticos e limitados ao tipo de mídia e tamanho da sessão, hoje, as novas aplicações e ferramentas dão origem a ambientes que suportam colaborações, desde sessões pequenas e fechadas, onde o acesso é restrito a participantes autenticados ou convidados, até grandes, geralmente abertas, com acesso irrestrito ou com

condições relaxadas, podendo, todas elas, suportarem múltiplos recursos, ou seja, mídias discretas e contínuas e diferentes estilos de interações.

Contudo, ainda que atualmente exista uma gama muito grande de aplicações para trabalho colaborativo disponíveis, sobretudo síncrono, as formas de compartilhamento de recursos são muitas vezes limitadas [02].

2.4 Dificuldades em Ambientes Colaborativos Multimídia

Para resumir os problemas encontrados no desenvolvimento e utilização de *groupware*, pode-se considerar a sugestão citada em [01]:

“É improvável... que os sistemas colaborativos causarão um grande impacto no mercado nos próximos anos. Além deles serem difíceis de serem desenvolvidos tecnicamente, os pré-requisitos para projeto são deficientes – nós realmente conhecemos muito pouco sobre como as pessoas trabalham em conjunto.”

Paradoxalmente, embora a colaboração *on-line* tem o objetivo de aumentar a produtividade e o escopo da comunicação das pessoas, em direção a tipos de tarefas mais complexas, as interações mediadas por computador criam novos gargalos. A largura de banda para comunicação das pessoas é limitada por suas ferramentas colaborativas, que para si próprias ou para o grupo permitem apenas certos padrões de interação, causando despersonalização, distância psicológica, menor comprometimento e encontros mais formais [06].

Enquanto encontros presenciais permitem revezamento nas conversações e nas atividades via mensagens sugestivas e dicas não verbais, como olhares, gestos, ou mudança na intensidade e entonação da voz, em espaços de colaboração virtuais a comunicação implícita baseada em convenções culturais, sociais e pragmáticas não é completamente suportada. Isto é devido a uma visão contextual limitada dos eventos e áreas de trabalho remotas, ou seja, falta conhecimento mútuo [02] das atividades individuais e coletivas.

2.4.1 Arquiteturas de Colaboração Centralizada e Distribuída

O *groupware* síncrono cria alguns requisitos a serem considerados. Um deles é a noção de WYSIWIS. A abstração WYSIWIS em *groupware* síncrono pode ser alcançada através de uma arquitetura centralizada, que apresenta a mesma saída de uma única instância da aplicação para todos os participantes. Já na arquitetura distribuída cada usuário executa instâncias separadas da aplicação de maneira sincronizada para, com isso, apresentarem sempre a mesma saída. A dificuldade de ambas as arquiteturas é, basicamente, manipular as entradas vindas de cada usuário, e não as saídas de cada aplicação.

Na arquitetura centralizada a dificuldade é manter as informações de entrada vindas de um usuário em particular separadas das que vêm de outro. Um caminho para se conseguir isso é permitir que apenas um usuário tenha acesso por vez à aplicação: é, então introduzido o problema do controle de palavra, já que é necessário ser definido que usuário irá controlar a aplicação, como, e quando irá controlar.

Já na arquitetura distribuída o problema a ser resolvido é a sincronização de todas as entradas vindas de cada uma das instâncias replicadas. Este problema pode ser tratado de diferentes formas:

1. Como nos sistemas centralizados, a aplicação pode permitir que apenas um usuário de cada vez introduza suas alterações, necessitando igualmente de políticas e mecanismos para implementar o controle de palavra;
2. Outra abordagem possível é que cada instância publique suas alterações para as demais (*broadcast* ou *multicast*) de forma a manter todas sincronizadas, necessitando, obviamente, de um suporte de comunicação com garantias de confiabilidade e ordenação.

2.4.2 Visibilidade em *Groupware* Síncrono

A noção de visibilidade em *groupware* síncrono gera a noção de conhecimento mútuo (*collaboration awareness*). Poder-se-ia afirmar que o controle de palavra simplesmente está preocupado em manter a visibilidade adequada, e com isso permitir que os participantes tomem decisões razoáveis a partir do que vêem. Entretanto, o controle de palavra pode ser tanto diretivo quanto restritivo, e decisões são difíceis sem uma visão clara de toda a situação [01].

Assim, comparando a comunicação presencial com a mediada por computador, pode-se observar uma dificuldade inerente para se conseguir um suporte tecnológico efetivo para o controle de palavra, ou seja, o meio tecnológico limita, ou até mesmo elimina a visibilidade, limitando políticas eficientes para controlar a palavra.

A forma mais comum para fazer os usuários de um grupo tomarem conhecimento uns dos outros é o tele-apontador. Para a maioria dos casos, o tele-apontador é um “bastão” compartilhado, controlado por um

dos participantes e visível para um ou mais dos demais participantes em um ambiente compartilhado.

2.4.3 Granulosidade do Controle em *Groupware* Síncrono

Uma forma elementar de controle de palavra em *groupware* colaborativo síncrono é o suporte a uma única palavra global. Entretanto, outras formas mais elaboradas para controle podem ser utilizadas.

Um distanciamento da abordagem WYSIWIS pode ser buscado para relaxar os requisitos de projeto, implementação e uso. Este afastamento basicamente pode permitir perspectivas mais individualizadas em uma aplicação de um ambiente compartilhado. Isto não é desejável apenas para participantes com diferentes papéis para conseguir diferentes visões do ambiente compartilhado, mas, de forma mais geral, isso é desejável para permitir que os participantes sejam capazes de focar sua atenção em partes mais específicas do ambiente quando estas forem mais apropriadas. Assim, uma granulosidade mais fina para o controle de palavra pode ser bastante útil.

2.4.4 Controle de Concorrência e Acesso

O conceito de controle de acesso e de concorrência em *groupware* síncrono é ampliado para o conceito de controle de palavra. A palavra é uma permissão para acesso e manipulação temporária a um determinado recurso. O controle de palavra, assim como as dificuldades para suas implementação, serão tratados num capítulo específico.

2.5 Conclusões do Capítulo

Atualmente, as novas aplicações e ferramentas estão dando origem a ambientes que suportam colaborações cada vez mais sofisticadas com múltiplas mídias. Para atender estas aplicações, grandes avanços têm sido realizados em áreas como roteamento *multicast*, *multicast* confiável, algoritmos de compressão de mídias e tecnologias de acesso. Entretanto, o suporte à coordenação de grupos, fundamental para tais aplicações, não tem sido abordado por uma quantidade tão grande de trabalhos, configurando uma lacuna na área de CSCW e prejudicando o desempenho geral de ambientes colaborativos multimídia.

Enquanto a falta da presença física pode impor certos limites na qualidade de uma atividade em grupo, a tele-presença pode ser empregada para auxiliar na coordenação do trabalho colaborativo através de ambientes colaborativos multimídia.

Embora exista um número razoável de softwares e sistemas *groupware* disponíveis para o trabalho colaborativo, a limitação na colaboração continua presente e comprometendo o desempenho destes sistemas. Isto se deve a dificuldades técnicas e também ao fato de que processos cooperativos e colaborativos entre humanos, que envolvem questões de conflito, considerações, preferências e interesses pessoais, confiança e privacidade, não são completamente entendidos, especialmente quando computadores são utilizados para suportar tais processos. [02]

As dificuldades técnicas que também podem impor limites aos ambientes colaborativos multimídia (que suportam, sobretudo, o *groupware* síncrono) são motivadas por requisitos mais rígidos impostos por tais ambientes. A implementação de arquitetura de sistema distribuída, visibilidade adequada de cada sítio remoto, controle de acesso

e concorrência conforme natureza das interações e implementação da granulosidade do controle adequada a cada cenário podem, por conta de dificuldades técnicas de implementação, limitarem a colaboração, sobretudo síncrona. Esta limitação pode, então, permitir apenas certos padrões elementares de interação em tais ACMs, causando despersonalização, distância psicológica, menor comprometimento e encontros virtuais menos dinâmicos e com menos interações, tornando os ACMs, conseqüentemente, menos eficientes e com menor colaboração.

Capítulo 3

INFRA-ESTRUTURA DE COMUNICAÇÃO PARA AMBIENTES COLABORATIVOS MULTIMÍDIA

Disponibilizar ambientes colaborativos multimídia a fim de suportar aplicações como videoconferências, ensino e treinamento à distância, teleseminários e aplicativos de entretenimento via *Internet* nem sempre é tarefa trivial, visto que a *Internet* não é, de fato, adequada à transmissão de informação em tempo real. Para viabilizar tais sistemas tornam-se necessários recursos adequados em diversas áreas, como por exemplo, algoritmos de compressão, protocolos de rede, sistemas operacionais, armazenamento digital e uma infra-estrutura de comunicação com rígidos requisitos e parâmetros de desempenho.

Os requisitos exigidos da infra-estrutura de comunicação para tais aplicações e ambientes diferem significativamente dos exigidos para as

tradicionais aplicações Web (texto e imagens), e-mail e transferência de arquivos. Em particular, aplicações multimídia interativas, como as supracitadas, possuem as características que diferem das demais aplicações tradicionais de rede e que, conseqüentemente, leva a diferentes requisitos de comunicação como, por exemplo, requisitos temporais e de confiabilidade.

Neste capítulo serão apresentados os principais aspectos que caracterizam o tráfego de dados multimídia e os requisitos básicos para a infra-estrutura de comunicação suportar tal tráfego. O endereçamento de grupo será abordado e suas características serão detalhadas para que seja possível entender as vantagens e implicações de sua utilização. Na seqüência são apresentados e comentados requisitos e tópicos da infra-estrutura de comunicação necessários e relacionados com a coordenação de grupo para aplicações e ambientes colaborativos multimídia.

3.1 O Problema da Transmissão Multimídia e de sua Qualidade

O processo de transmissão de informações multimídia, em particular para aplicações tempo real (mídias contínuas, como áudio e vídeo, por exemplo), via *Internet*, geralmente não é simples, embora fique transparente para os usuários finais destas aplicações.

Como pode ser visto na Figura 2, existem basicamente dois pontos críticos diferentes neste processo, que são o meio de comunicação e os sistemas finais utilizados nas pontas do processo [07].

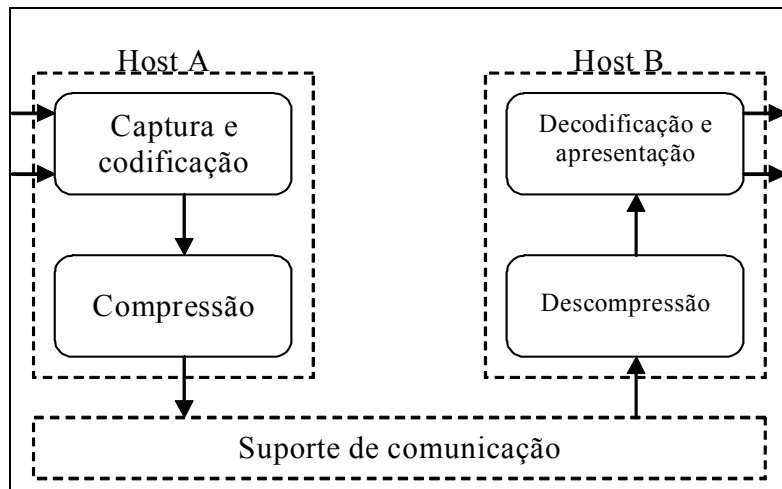


Figura 2: Transmissão síncrona de mídias contínuas

Aplicações multimídia síncronas (também chamadas de interativas ou tempo real) podem ser altamente sensíveis a atrasos (*delay sensitive*) e às variações que podem ocorrer nesses atrasos (*interarrival jitter*) ao passo que não necessitam da confiabilidade e serviço orientados à conexão oferecidos pela camada de transporte do padrão TCP/IP [08] e utilizados pelas aplicações tradicionais da *Internet*. Estas aplicações podem, ainda, necessitar de largura de banda relativamente grande para funcionarem adequadamente.

Assim, por causa das limitações atualmente encontradas na *Internet*, relativas a requisitos como largura de banda, atrasos e tolerância à perda, as aplicações multimídia síncronas e ambientes colaborativos multimídia atualmente utilizados através da *Internet* podem apresentar baixo desempenho ou nem mesmo serem viáveis.

Por outro lado, as redes locais, corporativas e metropolitanas, que utilizam novas tecnologias de comunicação para interconexão de seus sistemas finais, tanto para *backbone* quanto para acesso (última milha) buscam prover o suporte de comunicação com maior largura de banda e

diferentes classes de serviços, a fim de garantir melhor desempenho e mais recursos para tais aplicações. Outrossim, inúmeros esforços vêm sendo realizados para permitir que a arquitetura *Internet* possa oferecer suporte aos serviços exigidos por esta nova classe de aplicações [09].

Entretanto, é necessário que a evolução aconteça simultaneamente no dois principais pontos de congestionamento identificados na Figura 2, nos sistemas finais e no suporte de comunicação, caso contrário, a limitação de um ou de outro prevalecerá.

3.2 Características do Tráfego Multimídia

Antes de detalhar a infra-estrutura de comunicação para aplicações multimídia, convém caracterizar o tráfego de dados multimídia, que será gerado pelas aplicações que irão compor um ACM.

O tráfego multimídia é composto por informações provenientes de, basicamente, cinco categorias de aplicações: áudio, vídeo, texto, imagem *bitmap* e gráfico. Neste trabalho a preocupação maior permanece acerca das fontes transmissoras de informações multimídia contínuas de áudio e vídeo, por estas imporem requisitos mais rígidos aos sistemas de armazenamento e de comunicação.

3.2.1 Categorias de Transferência e Interatividade

A transmissão de mídias contínuas (áudio e vídeo, por exemplo) pode se dar de duas formas: síncrona e assíncrona.

Na transmissão assíncrona a informação é totalmente transferida e armazenada no receptor para, então, ser apresentada. Esta categoria é

viável para seqüências com tamanho reduzido, sob pena de se ter um atraso muito grande para o início da apresentação de seqüências muito grandes. A transmissão assíncrona, para transferência de volumes de dados muito grande, também impõe requisitos de armazenamento no destino.

Já na transmissão síncrona, também chamada tempo real, a informação é capturada e transferida em tempo real pela rede, sendo apresentada continuamente pelo receptor. Para esta categoria de transferência de dados os requisitos de armazenamento no receptor são reduzidos. Por outro lado, são impostos requisitos à infra-estrutura de comunicação.

Neste capítulo, a partir daqui, quando for citado “tráfego multimídia” estar-se-á tratando e considerando transferência síncrona de mídias contínuas, pois esta categoria e estas mídias, que são as que impõem restrições mais rígidas à infra-estrutura de comunicação, são utilizadas em ACMs e estão no contexto deste trabalho.

3.2.2 Vazão x Tempo

O tráfego multimídia pode ser caracterizado também quanto a sua vazão ao longo do tempo, podendo ser considerado VBR (*Variable Bit Rate*) ou CBR (*Constant Bit Rate*).

Várias aplicações geram tráfego multimídia a taxas de bits constantes (CBR) como, por exemplo, áudio PCM. Nestes casos, considerando transferência síncrona, é necessário que a rede também suporte uma vazão de dados constante, sob pena de tornar necessário o armazenamento local (*buffer*) relativamente grande nos sistemas finais para evitar interrupções na apresentação da mídia.

Já o tráfego VBR tem sua taxa de bits variável ao longo do tempo, podendo gerar rajadas na transferência de dados na rede, bem como períodos de relativa inatividade. Um exemplo deste tipo de tráfego é áudio e vídeo compactados. Uma forma de caracterizar este tipo de mídia do ponto de vista da rede é através de uma taxa média e uma taxa de pico durante as rajadas.

3.2.3 Dependência Temporal ou Interatividade

Quando o tráfego multimídia é parte de um sistema interativo, o atraso total fim-a-fim deve ficar abaixo de um nível de tolerância para que não prejudique a interatividade.

Tomando como exemplo uma videoconferência entre duas ou mais pessoas, experiências mostram que atrasos abaixo de 150ms fazem com que os participantes não percebam seus efeitos [10].

3.2.4 Volume de Dados

O tráfego multimídia síncrono pode configurar grandes quantidades de dados a serem transferidas pela infra-estrutura de comunicação disponível. A largura de banda disponível vem sendo um grande problema para tais aplicações. Assim, a compressão de dados, especialmente de áudio e vídeo, é necessária para otimizar a utilização da largura de banda existente nas redes atuais e limitar a demanda por maior capacidade de armazenamento e transferência.

A necessidade de aplicar algoritmos de compressão às mídias difundidas torna-se evidente quando se analisa a Tabela 1, onde são mostrados alguns valores típicos de taxas sem compressão.

Tabela 1 :Largura de banda utilizada por alguns padrões de mídias típicos

Áudio c/ qualidade CD	16 bits, 2 canais a 44.1KHz	1.4Mbps
Vídeo NTSC	640x480 pixels/frame - 24 bits/pixel	27Mbps
Vídeo HDTV	1280x720 pixels/frame - 24 bits/pixel	81Mbps

As técnicas de compressão seguem basicamente dois modos: a compressão sem perda, onde a informação é recuperada sem qualquer alteração depois da descompressão; e a compressão com perda ou irreversível, onde a informação depois da descompressão é diferente da informação original. Este modo permite maior taxa de compressão e por isso é o mais conveniente e utilizado para mídias contínuas, como áudio e vídeo [11].

3.2.5 Continuidade Temporal

Quando o tráfego multimídia é formado por mídias contínuas, como áudio e vídeo, embora a compressão reduza o volume de dados a ser transmitido, as amostragens do áudio e os quadros do vídeo devem ser apresentados em intervalos regulares [10] para não comprometer a qualidade da apresentação. Esta propriedade é chamada de isocronia.

3.2.6 Sincronização Temporal

Outra característica dos ACMs é a necessidade da existência de sincronismo entre diferentes mídias recebidas. A recepção do áudio, por exemplo, deve estar sincronizada com a recepção do vídeo para evitar apresentações simultâneas das mídias nos sítios receptores, defasadas entre si [10]: esta defasagem pode facilmente ser percebida, quando a

infra-estrutura de comunicação não provê este serviço de sincronização temporal, observando atentamente o movimento labial e o áudio recebido.

3.3 Requisitos para Infra-estrutura de Comunicação

Não é difícil imaginar que para uma infra-estrutura de comunicação suportar aplicações colaborativas multimídia síncronas, esta deva ter uma largura de banda razoável para suportar as várias mídias utilizadas, inclusive áudio e vídeo. De fato, a largura de banda é um requisito crucial para tais aplicações. Entretanto, o tráfego multimídia – caracterizado na seção 3.2, impõe rígidos requisitos a vários parâmetros de rede, que vão além da largura de banda da rede, como atraso, vazão, jitter e confiabilidade.

A *Internet* é a infra-estrutura de comunicação empregada pelos ACMs em grande parte dos cenários que estão sendo considerados neste trabalho. Assim, é importante observar seus modelos de serviço e transmissão: o tipo de serviço oferecido pelo protocolo IP, em sua forma nativa não fornece suporte à qualidade de serviço (*Quality of Service – QoS*) satisfatório [12]. Algumas aplicações podem não funcionar adequadamente quando se tem atrasos em filas ocasionados por roteadores e comutadores ou pontos de congestionamento na rede. Outrossim, a *Internet*, desde sua concepção, sempre teve como característica o modelo de serviço de melhor esforço (*best effort*). Dessa forma, para atender determinados requisitos de rede para ACMs e cenários específicos, pode se tornar necessário adotar serviços de rede diferenciados como, por exemplo, os que forneçam largura de banda mínima garantida entre sítios e baixos atraso e variação do atraso [12].

3.3.1 Largura de Banda Ocupada

Como pode ser observado na Figura 3, a largura de banda demandada pode variar desde poucos Kbps até mais de 100Mbps, dependendo da natureza da aplicação [13]. Dependendo da largura de banda disponível e serviço de entrega adotado [12], pode ser necessário lançar mão de artifícios técnicos para tornar possível a aplicação desejada sobre o suporte de comunicação disponível. Uma aplicação de videoconferência, por exemplo, pode requisitar mais de 1Mbps de largura de banda. Entretanto, adotando algoritmos de compressão adequados e redução na qualidade e tamanho do vídeo transmitido – por conta de limitações da percepção da visão humana, pode-se viabilizar ambientes de videoconferência com largura de banda disponível a partir de 64Kbps.

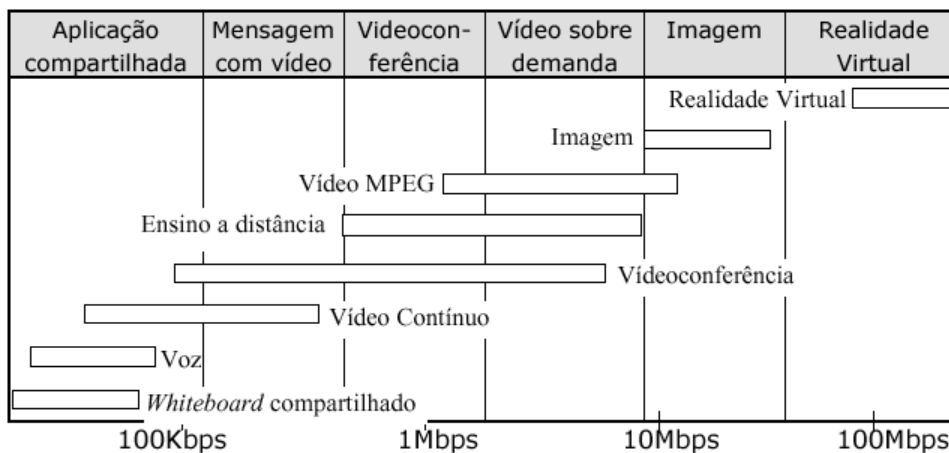


Figura 3: Largura de banda utilizada por diferentes tipos de aplicações

Como boa parte das aplicações multimídia síncronas manipulam grandes quantidades de dados, além de algoritmos de compressão, torna-se necessário um esquema de endereçamento adequado para permitir a difusão seletiva e eficaz das informações, como o endereçamento *multicast*. Esse endereçamento, devido sua importância em ACMS, sobretudo nos

ambientes implementados através da *Internet*, será apresentado com detalhes na seção 3.4.

3.3.2 Atraso

O tráfego multimídia síncrono impõe à infra-estrutura de comunicação rígidos requisitos temporais.

Os pacotes carregando informações de áudio e vídeo transmitido de maneira síncrona, por exemplo, têm utilidade para o receptor apenas se forem entregues dentro de um período específico. Se forem entregues após este período não terão utilidade e serão descartados pelo receptor.

A latência e o *jitter* [13], provocados pelos equipamentos de rede entre o transmissor e receptor, são os principais fenômenos que contribuem para o atraso entre transmissor e receptor de mídias síncronas através da *Internet*.

A latência da rede é composta e depende de vários fatores [14], como o tempo de propagação, tempo de transmissão (serialização), tempo de armazenamento e envio (dispositivos *store and forward*) e tempo de processamento para o encaminhamento dos pacotes.

O *Jitter*, por sua vez, ocorre quando a latência da rede varia com o passar do tempo, prejudicando a reprodução das mídias (como áudio e vídeo) nos sítios receptores. Como o *Jitter* é uma característica inerente à *Internet* (por causa do serviço *best effort* adotado desde sua concepção), outros artificios devem ser usados para eliminar ou minimizar os efeitos do *jitter*. A utilização de *buffers* nos receptores, juntamente com informações de marcação de tempos feitas no sítio transmissor e

carregadas através da rede junto com a informação multimídia é uma técnica comumente utilizada [15].

3.3.3 Taxa de Erro e Confiabilidade

Aplicações tradicionais funcionam, através da *Internet*, utilizando serviços confiáveis e orientados a conexão (providos pela camada de transporte do modelo de referência OSI através do protocolo TCP – *Transport Control Protocol*), como FTP (*File Transfer Protocol*), HTTP (*Hiper Text Transfer Protocol*) e SMTP (*Simple Mail Transfer Protocol*).

Ambientes de trabalho colaborativo, além deste tipo de serviço orientado a conexão e confiável, toleram, para determinadas aplicações, serviço de transporte com certo grau de perda ou erros, mas que, por outro lado, necessitam de tempos de atraso muito menores entre transmissor e receptor, impossíveis de serem obtidos com a utilização do protocolo TCP.

A fim de criar condições para lidar com estes requisitos das mídias contínuas síncronas, podem-se utilizar protocolos apropriados como UDP [16] e RTP [17], adequados para transmissão de mídias contínuas e com requisitos temporais mais rígidos que as aplicações tradicionais.

3.4 Endereçamento de Grupo

A maioria das aplicações tradicionais da *Internet* opera entre um emissor e um receptor. Com os recentes avanços tecnológicos e o aparecimento de aplicações cada vez mais sofisticadas surge a necessidade de transmitir informações para um grupo selecionado de

participantes distribuídos pelo mundo, como por exemplo, transmissão de mensagens corporativas, áudio e videoconferência para encontros remotos, trabalho cooperativo, ensino e treinamento à distância, monitoramento de ambientes e processos remotos e entretenimento.

Paralelamente, a necessidade de transferir volumes de informações cada vez maiores por tais aplicações emergentes torna evidente a necessidade de melhoramentos na infra-estrutura da rede atual e também, muitas vezes considerado mais viável, a utilização de mecanismos de comunicação que melhorem o desempenho de tais aplicações.

Assim, é necessário que os mecanismos de comunicação de grupo permitam o uso de endereçamentos de grupo, livrando o desenvolvedor de detalhes de endereçamento dos processos individuais que compõem o grupo e simultaneamente utilizem a largura de banda disponível na rede de forma mais eficiente.

Na *Internet*, que segue o padrão TCP/IP [16], um endereço IP é um par composto do identificador de rede e do identificador de *host*. O cabeçalho do pacote contém informação utilizada para o roteamento, tais como os endereços de rede dos *hosts* emissor e receptor. Baseados nestes identificadores, os endereços IP são classificados em 5 classes: A, B, C, D e E (reservado e ainda não utilizado). A Figura 4 mostra o formato das quatro primeiras classes.

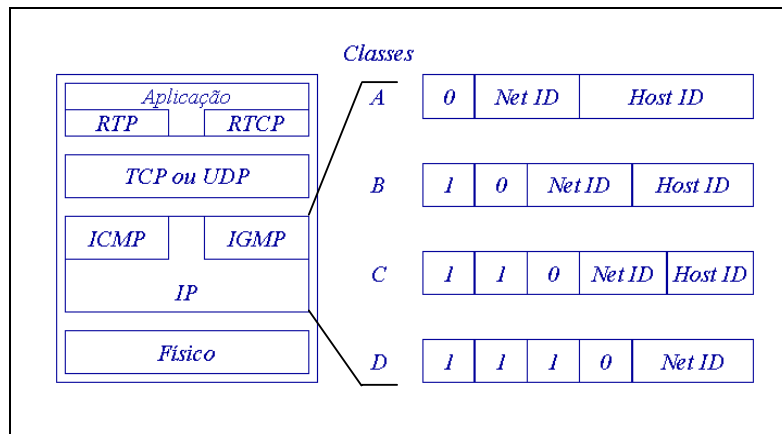


Figura 4: Pilha de protocolos da *Internet* e Classes de Endereços IP

A classe D identifica os pacotes destinados a grupos *multicast*, foco desta seção.

O conceito de endereçamento *multicast* para comunicação de grupo na *Internet* foi proposto por *Steve Deering* em sua tese de doutorado na *Universidade de Stanford*, e mais tarde desenvolvido na *XeroxParc* [18].

A comunicação *multicast* emprega um modelo em que um *host* emissor envia uma mensagem a um grupo de *hosts* de destino. Muito embora isso possa ser feito mandando-se diferentes mensagens do tipo *unicast* (um-para-um), conforme Figura 5, existem várias razões pelas quais a comunicação *multicast* é desejável.

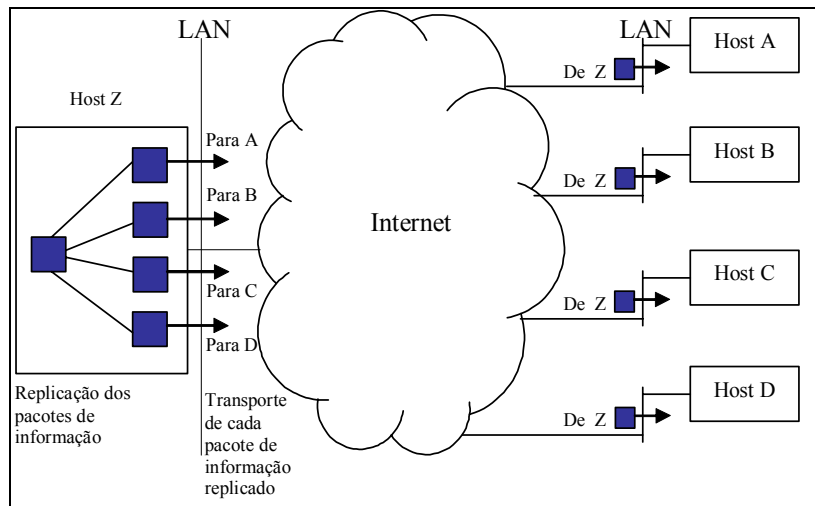


Figura 5: Comunicação *unicast*

A principal vantagem é que o endereçamento *multicast* diminui a carga de tráfego na rede, já que o transmissor envia somente um pacote da mensagem, como pode ser visto na Figura 6, sendo que o pacote é replicado apenas se necessário, não consumindo largura de banda da rede com pacotes replicados para participantes de uma mesma rede ou sub-rede.

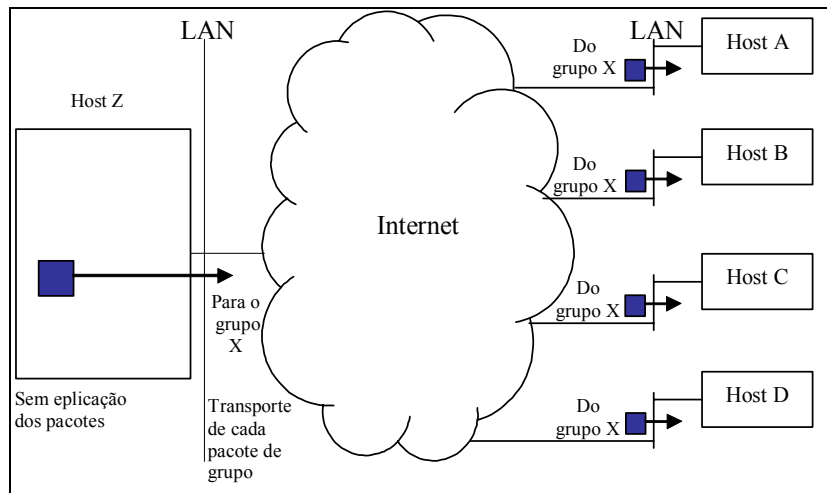


Figura 6: Comunicação *multicast*

Cada vez mais aplicações com suporte a áudio e vídeo vêm usando a comunicação *multicast*. Receptores, ao invés de usar a conexão ponto-a-

ponto entre os *hosts* do grupo, utilizam o endereçamento *multicast* para a distribuição dos dados multimídia entre os *hosts* destino. A flexibilidade para se integrar e se retirar de determinado grupo, fornecida pelo *multicast*, torna a manipulação dos integrantes do grupo muito fácil [18].

O endereçamento *multicast* oferece suporte básico de comunicação para colaborações síncronas para, desde pequenos grupos com políticas de acesso e participação mais rígidas e formais até grandes grupos (seminários via *Internet*, por exemplo) com políticas de acesso, participação mais relaxadas, através de redes LAN (*Local Area Network*), MAN (*Metropolitan Area Network*) ou WAN (*Wide Area Network*), onde vários recursos podem ser compartilhados de forma controlada e coordenada entre os vários participantes das sessões.

Para transmissões *multicast*, *hosts* emissores e receptores e a infraestrutura de rede entre eles (incluindo os roteadores intermediários) devem suportar o protocolo *multicast* como, por exemplo, *IP MULTICAST*. Os requisitos para os emissores e receptores suportarem *IP Multicast* são:

- Suportar transmissão e recepção *IP Multicast* na pilha do protocolo TCP/IP;
- Software deve suportar IGMP para comunicar requisições para se juntar a grupos *multicast* e receber o tráfego;
- Placas e *drivers* de rede que eficientemente filtram para uma LAN, os dados do endereço da camada mapeados da camada de rede *IP Multicast*;
- Softwares de aplicação *IP Multicast*, como os de videoconferência.

Já para expandir o tráfego *multicast* para uma WAN é necessário que todos os roteadores intermediários entre o emissor e o receptor suportem *IP Multicast*. Outra alternativa é a utilização de túneis e *firewalls* reconfigurados para permitir tráfego *IP Multicast* [18].

3.4.1 Grupos *Multicast*

Unicast (endereços IP classes A, B e C) é o serviço clássico com o qual a maioria dos usuários de rede está acostumada. Trata-se de um tipo de transmissão ponto-a-ponto onde o pacote é transmitido do *host* origem ao *host* destino.

O serviço *multicast* (Classe D) é uma técnica usada para enviar cópias de um mesmo pacote a um subconjunto de todos os possíveis destinatários. Se esse subconjunto é o conjunto de todos os possíveis destinatários a técnica é chamada de *broadcast*. Neste caso o pacote é transmitido a todos os *hosts* conectados à rede. Este serviço é freqüentemente indesejável, porque ele requer que as máquinas conectadas à rede executem algum tipo de processamento para determinar se estão ou não interessados nos dados que estão sendo transmitidos. A transmissão em *broadcast*, principalmente em redes de grande porte, é indesejável pelo fato de que estas trabalham freqüentemente no limite da largura de banda.

A comunicação *multicast* é a mais adequada para transmitir dados a um grupo potencialmente amplo na rede *Internet* ou mesmo em redes LAN e MAN: somente aqueles *hosts* que estão interessados em uma sessão *multicast* em particular, terão os dados roteados até eles. O conjunto destes *hosts* é chamado de grupo *multicast*, sendo-lhe associado um endereço IP Classe D específico.

Um *host* que desejar receber o tráfego *multicast* destinado a um grupo, deve juntar-se ao endereço *multicast* correspondente.

A comunicação *multicast* não é orientada a conexão, ou seja, um pacote *multicast* é liberado aos membros do grupo de destino com a

mesma confiabilidade e modo de transmissão "*best effort*" que um pacote IP *unicast* padrão. Isto significa que não é garantido que um pacote *multicast* alcance todos os membros do grupo ou chegue na mesma ordem em que foi transmitido em relação a outros pacotes.

Hosts individuais são livres para juntar-se ou abandonar grupos *multicast* a qualquer momento, não existindo restrições na localização física ou no número de membros de um grupo *multicast*. Cada *host* pode também participar de mais de um grupo.

Um grupo *multicast* pode ser permanente ou transiente. Grupos permanentes são aqueles que possuem um endereço conhecido e fixo. Apenas o endereço é permanente. Os membros deste grupo não são permanentes e num dado momento um grupo permanente pode ter qualquer número de membros, até mesmo, nenhum.

Os endereços *multicast* que não são reservados para nenhum grupo permanente estão disponíveis para atribuição dinâmica de grupos temporários e existirão somente enquanto possuírem membros. Estes grupos são criados quando necessário, e descartados quando o número de membros atinge zero ou seu tempo de vida termina. A Tabela 2 mostra os grupos *multicast* permanentes.

Tabela 2: Grupos *multicast* permanentes

Endereço	Utilização
224.0.0.1	Todos os <i>hosts</i> em uma LAN
224.0.0.2	Todos roteadores em uma LAN
224.0.0.5	Todos roteadores OSPF em uma LAN
224.0.0.6	Todos roteadores OSPF designados em uma LAN

Considerando a relação entre o emissor e o receptor num grupo, podem-se caracterizar três tipos de relações de comunicação:

Um-para-Muitos: um único *host* envia mensagens para dois ou mais *hosts* como, por exemplo, a distribuição de áudio e vídeo num seminário via *Internet* ou sistemas de monitoramento.

Muitos-para-Um: um número qualquer de receptores manda dados de volta, via *unicast* ou *multicast* como, por exemplo, a localização de recursos numa rede.

Muitos-para-Muitos: um número qualquer de *hosts* enviando ou recebendo dados de um grupo *multicast* como, por exemplo, conferências multimídia e ensino a distância.

3.4.1.1 Níveis de Conformidade

Uma estação participa de grupos IP *multicast* em conformidade com um dos três níveis existentes [18] apresentados na Tabela 3:

Nível	Significado
0	<i>Host não envia nem recebe IP multicast</i>
1	<i>Host pode enviar, mas não receber IP multicast</i>
2	<i>Host pode enviar e receber IP multicast</i>

Tabela 3: Níveis de conformidade dos grupos *multicast*

Atualmente, não existem exigências de que todas as implementações IP suportem *multicast*. Os *hosts* pertencentes ao nível 0 são aqueles que, de modo geral, não são afetados pela atividade *multicast*. A exceção acontece em alguns tipos de redes locais, onde a presença de máquinas de nível 1 ou 2 causam a entrega errônea de pacotes *multicast* para máquinas de nível 0. Estes pacotes podem ser facilmente identificados pela presença do endereço IP classe D em seu *campo destination address*, e devem ser descartados pelas máquinas de nível 0.

No nível 1, as máquinas têm a possibilidade de participar de alguns serviços baseados em *multicast*, como a localização de um recurso ou a informação de um estado de alguma variável, mas não possuem as condições necessárias para participar de nenhum grupo.

Por fim, no nível 2, um *host* pode entrar e sair de um grupo *multicast*, assim como pode enviar pacotes para grupos. Este nível requer a implementação do *Internet Group Management Protocol*, o IGMP [18].

3.4.1.2 Protocolo de Gerenciamento de Grupo – IGMP

Para um *host* participar de um grupo *multicast* através de várias redes, ele deve informar o roteador *multicast* local. Os roteadores locais contatam outros roteadores *multicast*, passando a informação de associação a um grupo e estabelecendo rotas. A idéia é similar à propagação de rotas entre *gateways* convencionais.

O IGMP é utilizado pelos *hosts* que suportam *multicast* e roteadores *multicast* para comunicarem informações de associação em grupos. O IGMP é análogo ao protocolo ICMP (*Internet Control Management Protocol*): ele usa pacotes IP para transmitir suas mensagens e fornece um serviço utilizado pelo protocolo IP.

Pacotes *multicast* de fontes remotas devem ser tratados por roteadores, os quais devem somente transmiti-los, se existir um *host* membro na rede local, que possa receber o pacote, conforme Figura 7.

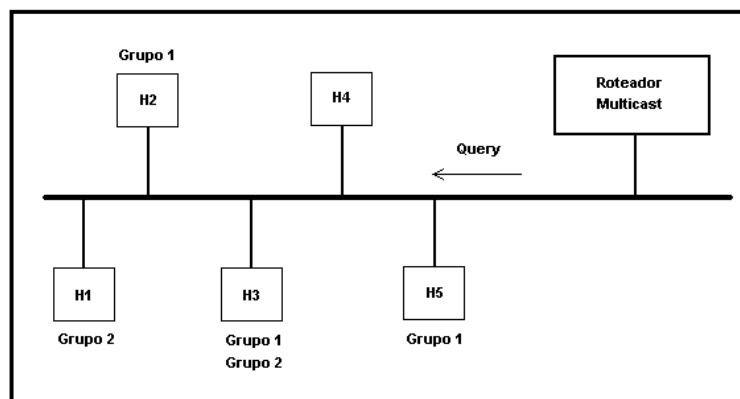


Figura 7: Roteamento *Multicast*

O IGMP é usado por roteadores *multicast*, para especificar que existem *hosts* pertencentes a um determinado grupo conectados à rede. Isto é feito enviando questionamentos, *IGMP Query* - Figura 7 [18], tendo uma resposta de todos os *hosts* e de qual grupo fazem parte.

Um *host* pode expressar seu interesse em ser membro de um determinado grupo *multicast*, enviando uma mensagem IGMP para juntar-se a tal grupo, podendo, a partir daí, enviar e receber informações endereçadas a este grupo.

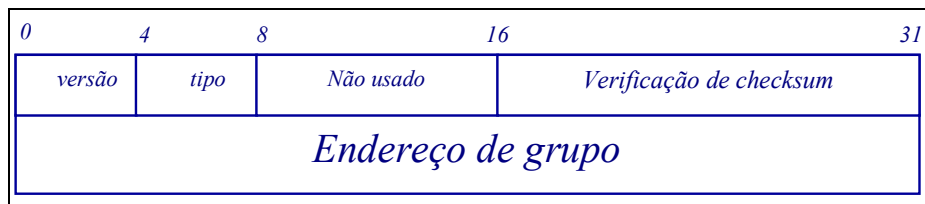


Figura 8: Formato do pacote IGMP

O IGMP, cujo formato da mensagem é apresentado na Figura 8, usa principalmente três tipos de mensagens:

- Tipo 1: Questionamento dos membros do grupo (*membership query*).
- Tipo 2: Relatórios dos membros do grupo (*host membership report*)
- Tipo 3: Atualizações de roteamento de protocolo DVMRP.

As mensagens do Tipo 1 são geradas por roteadores *multicast* (*mrouter*) numa sub-rede com capacidade *multicast*, e enviadas ao endereço de grupo 224.0.0.1 para determinar quais grupos de *host* têm membros nas redes diretamente conectadas a ele, como mostra a Figura 7.

Quando um *host* recebe uma mensagem do tipo "*query*", ele responde com uma mensagem do tipo 2, "*host membership report*", para cada um dos grupos de *hosts* a que ele pertence. Para evitar um grande fluxo de respostas, cada *host* escolhe atrasos de tempo aleatórios para gerar os dados de cada um dos grupos a que ele pertence. Durante o período em que o *host* está aguardando para gerar o relatório, ele também fica "ouvindo" a rede e, se existir outro relatório para o mesmo grupo, o *host* cancela o seu, caso contrário transmite o relatório para o grupo fazendo com que todos os outros membros do grupo cancelem seus relatórios.

Com base na informação de quais são os membros do grupo, os roteadores são capazes de determinar qual tráfego *multicast* necessita ser transmitido, para cada uma das redes ou sub-redes. Os roteadores *multicast* usam esta informação juntamente com o protocolo de roteamento *multicast* para suportar IP *multicast* através da *Internet*.

3.4.2 Roteamento *Multicast*

A comunicação *multicast* não é suportada por todos os roteadores na *Internet* e, dessa forma, a topologia *multicast* difere da topologia da *Internet*. Com isso, os roteadores *multicast* precisam executar um protocolo para descobrir sua própria topologia e identificar para onde enviar os datagramas *multicast*.

O protocolo de roteamento *multicast* mais utilizado é o *Distance Vector Multicasting Routing* - DVMRP. Também existem outros protocolos de roteamento *multicast*, menos difundidos como, por exemplo, o *Multicast Open Shortest Path First* (MOSPF) e o *Protocol Independent Multicasting* (PIM), que podem interoperar com os roteadores DVMRP [18].

Cada pacote *multicast* usa o campo TTL (*Time to live*) do cabeçalho do pacote IP como um parâmetro de limitação de escopo. O campo TTL controla o número de saltos (*hops*) permitidos para cada pacote *multicast*, ou seja, toda vez que um pacote passa por um roteador, o TTL é decrementado e, quando o TTL de um pacote se encontra em zero, ele é abandonado sem nenhuma notificação de erro ao emissor. Este mecanismo evita a transmissão desnecessária de mensagens para regiões da *Internet*, que se situam além das sub-redes contendo os membros de grupos *multicast*.

Em uma rede local, um pacote *multicast* (com TTL definido em 1) alcança todos os membros imediatamente vizinhos em um grupo de *hosts* destino. Se um pacote *multicast* possui um TTL maior que 1, os roteadores *multicast* ligados à rede local, assumem a responsabilidade de fazer a interconexão com outras redes ou sub-redes. Assim, os pacotes são transmitidos para outras redes que possuem membros de um grupo *multicast*, desde que alcançáveis pelo TTL do pacote.

O valor limite do TTL em roteadores *multicast* evita que, pacotes com valores menores de TTL percorra sub-redes a mais. Este pode ser um mecanismo conveniente, para confinar o tráfego *multicast* dentro de uma área determinada.

3.4.3 MBone e Mecanismos de Tunelamento

O MBone - *Multicast Backbone*, é uma rede virtual que existe através da *Internet* e que possui topologia própria e distinta da topologia da *Internet* por causa das limitações e configurações dos dispositivos de encaminhamento que não suportam IP *Multicast*.

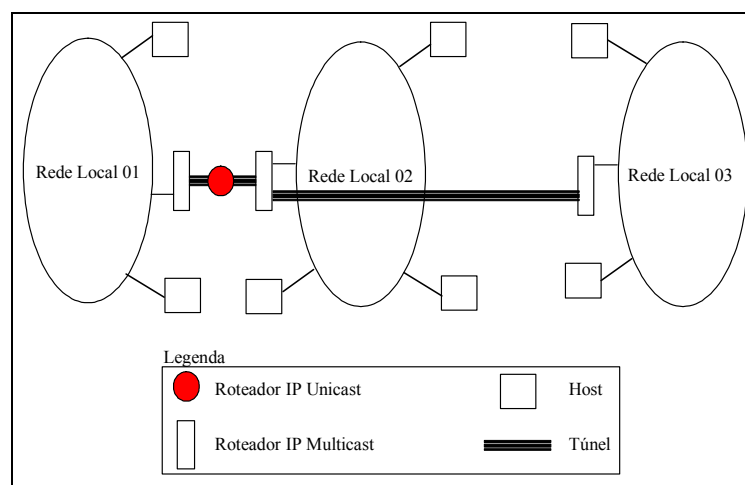


Figura 9: Túneis *Multicast*

Assim, o *MBone* é composto por sub-redes que suportam IP *Multicast*, denominadas ilhas *multicast*, conectadas umas às outras através de enlaces virtuais ponto-a-ponto, chamados túneis.

O tunelamento é um esquema utilizado através de roteadores que não suportam o roteamento *multicast*. Os túneis precisam ser criados entre dois roteadores que suportam tal roteamento, tornando transparente para os roteadores intermediários o conteúdo dos pacotes *multicast*, conforme Figura 9. Esta foi a maneira encontrada para contornar a existência de roteadores sem suporte à comunicação *multicast* entre sub-redes diferentes: o encapsulamento de pacotes *multicast* dentro de pacotes *unicast* regulares para que estes roteadores pudessem comportar-se como *mrouters*.

3.5 Coordenação em ACMS e Infra-estrutura de comunicação

O serviço de coordenação e controle de ambientes colaborativos multimídia tem requisitos específicos em relação à infra-estrutura de comunicação. As informações de coordenação e controle do ambiente devem alcançar todos os sítios que colaboram numa determinada sessão e, mensagens que falham, ao contrário de eventuais perdas de pacotes toleradas em fluxos de áudio ou vídeo, devem ser retransmitidas. A ordenação das mensagens para este tipo de serviço também pode ser importante devendo, portanto, ser suportada pela infra-estrutura de comunicação.

Diante destes requisitos, a comunicação *multicast* não parece adequada, principalmente pelo motivo de utilizar serviço de rede e de

transporte não orientado a conexão, sem garantia de entrega, por meio dos protocolos IP e UDP.

Por outro lado, a utilização de protocolos orientados a conexão, como o TCP, para suportar o serviço de coordenação em ambientes colaborativos multimídia, também pode não ser apropriada, já que se perdem características importantes e desejáveis para estes ambientes e que são garantidas pela comunicação *multicast* como, por exemplo, a escalabilidade do ambiente e a flexibilidade na entrada e saída dos participantes de determinado grupo.

Dessa forma, torna-se necessário que se empregue algum protocolo que permita que mensagens sejam roteadas através de redes *multicast* e entregues de forma ordenada e confiável.

Na seqüência são apresentados sucintamente os principais protocolos que buscam prover confiabilidade aos pacotes que trafegam em grupos *multicast*.

3.5.1 Reliable Multicast Protocol

O RMP *Reliable Multicast Protocol* [19] é baseado num esquema de *token* e que garante ordenação total das mensagens.

A desvantagem é que, em redes de longa distância, as mensagens de controle baseadas em *token* introduzem alta latência [19].

3.5.2 *Reliable Multicast Transport Protocol*

O RMTP (*Reliable Multicast Transport Protocol*) [20] é um protocolo de transporte *multicast* confiável para a *Internet*, fornecendo seqüência, entrega sem perda do fluxo de dados de um transmissor para um grupo de receptores.

O RMTP é baseado num modelo hierárquico multicamada, no qual os receptores são agrupados numa hierarquia de regiões locais, com um receptor designado (DR) em cada região. Os receptores de cada região, periodicamente enviam ACKs para os seus DRs correspondentes, e estes enviam os ACKs para os DRs do nível acima, até que os DRs do mais alto nível enviem os ACKs para o transmissor, evitando assim o problema da explosão do número de ACKs. Atualmente o RMTP suporta fluxos contínuos de pequenas mensagens (de 50 a 100 bytes) com rigorosos requisitos de latência.

Um problema que este protocolo apresenta é a escolha dos DRs, não adequada para o modelo de comunicação *multicast*, que é feita através de arquivo de configuração [21].

3.5.3 *Scalable Reliable Multicast*

O SRM (*Scalable Reliable Multicast*) [15] foi desenvolvido para sessões que seguem o modelo LWS [21], que será detalhado no capítulo 5. Este protocolo emprega mecanismos de recuperação local e usa um esquema de delay aleatório antes de enviar uma requisição ou correção de pacote que é uma função da distância daquele participante, em segundos, a partir do nó que disparou a requisição ou correção.

Figura como desvantagem deste protocolo a dificuldade de configuração dos temporizadores para redes altamente dinâmicas com variações rápidas na carga e alterações freqüentes na estrutura da rede [21].

3.5.4 Light-weight Reliable Multicast Protocol

O LRMP (*Light-weight Reliable Multicast Protocol*) [21] tem o objetivo de oferecer um serviço de envio de dados seguro e com capacidade de ordenação para grupos de comunicação. Ele foi desenvolvido, como o SRM, para ser utilizado em ambientes de rede heterogêneos e suportar múltiplos transmissores de dados.

O LRMP também possui suporte a controle de congestionamento e fluxo, permitindo que a largura de banda da rede possa ser compartilhada com outros fluxos de dados. A escalabilidade e o controle de fluxo são duas questões mal resolvidas pelas implementações de protocolos *multicast* confiáveis e que motivou o desenvolvimento do LRMP. Ao contrário de outros protocolos, o LRMP foi concebido para ser leve no que diz respeito à sobrecarga do protocolo (*overhead*) e simples no que se refere aos mecanismos de controle.

Atualmente existe uma implementação em Java, livre para uso comercial e acadêmico, e usado num número expressivo de projetos e produtos, inclusive adotado pela SUN no JSST (*Java Shared Data Toolkit*) [22].

Suas características, sua adoção em diversos sistemas e sua implementação disponível na forma de um pacote de bibliotecas reutilizáveis Java [23] mostram que o LRMP é uma excelente alternativa para auxiliar no provimento de uma infra-estrutura de comunicação para

suportar e atender os requisitos para a infra-estrutura de comunicação impostos pela coordenação e controle em ambientes colaborativos multimídia. Nesse sentido, o LRMP, suas características e mecanismos serão mais bem detalhados a seguir.

3.5.4.1 Objetivos do LRMP

Os objetivos principais do LRMP [21] são:

- Prover confiabilidade razoável sobre a infra-estrutura atual da *Internet*, devendo, para tal, garantir a transmissão confiável quando a rede estiver em condições normais, sem necessitar do suporte de algum roteador intermediário para prover tal confiabilidade;
- Prover infra-estrutura que se estenda facilmente para grandes grupos e que faça uso racional dos recursos de rede;
- Prover mecanismos que permitam que o protocolo continue funcionando para o caso de receptores serem extremamente lentos, se comportarem de forma anormal ou, até mesmo, se parte da rede perder a comunicação.

Os objetivos secundários do LRMP são imparcialidade no controle de congestionamento, bom desempenho e facilidade de evolução.

Frente a estes objetivos, o LRMP foi desenvolvido, sobretudo, para grupos fracamente acoplados [21] onde, geralmente, não existe um ponto de início para todos os participantes: cada participante pode entrar ou deixar o grupo a qualquer instante e não se tem conhecimento exato e registro dos participantes do grupo num momento exato.

3.5.4.2 Principais Características do LRMP

A principais características do LRMP são: recuperação local de erros, controle de fluxo e congestionamento, confiabilidade seletiva (por pacote), relatório de receptores seletivo e recuperação de erros em avanço (FEC – *forward error recovery*).

Enquanto em conexões TCP a retransmissão ocorre a partir do sítio transmissor, em grupos *multicast* a retransmissão pode ocorrer a partir de um terceiro sítio, localizado mais próximo do sítio requisitante (comunicação de erros utilizando NACKs) [21], e que tenha recebido os pacotes solicitados sem problemas. Este esquema, adotado pelo LRMP, pode evitar boa parte do tráfego gerado pelos pedidos de retransmissão. O escopo local e a distância para o sítio requisitante é definido com o auxílio da informação do campo TTL no pacote IP.

O controle de fluxo e de congestionamento busca ajustar a taxa de transmissão de uma aplicação à capacidade de processamento de informações dos demais sítios e à capacidade de vazão da infra-estrutura de comunicação. Para múltiplos fluxos a imparcialidade no controle deve ser mantida.

Através da confiabilidade seletiva é possível que os dados de uma aplicação sejam transmitidos com ou sem confiabilidade. Isto permite que uma aplicação possa fazer uso dos dois serviços, transmissão com e sem confiabilidade, utilizando o mesmo protocolo de transporte, o LRMP.

O mecanismo de *feedback* seletivo permite monitorar a qualidade de serviço nos receptores. Estas informações provenientes dos receptores são úteis para estimar os participantes de uma sessão, taxa de perda, número de pacotes recebidos e auxiliar nos ajustes de parâmetros da sessão e

transmissão, bem como na detecção e localização de falhas na infraestrutura de comunicação.

O mecanismo de FEC consiste em adicionar alguns pacotes redundantes no fluxo de dados a serem transmitidos para criar condições de recuperar o fluxo original sem a necessidade de retransmissão. Esta técnica pode evitar a geração de pacotes NACK e se torna uma vantagem em redes com enlaces assimétricos.

3.6 Conclusões do Capítulo

As limitações que restringem o desempenho de um ACM podem, basicamente, ser classificadas em duas categorias: limitações da infraestrutura da comunicação e dos sistemas finais que hospedam os ambientes multimídia em cada sítio.

Em relação à infraestrutura de comunicação para ambientes colaborativos multimídia, devido às características do tráfego gerado por ambientes colaborativos multimídia (grandes quantidade de dados a serem transmitidos, dependência temporal, necessidade de continuidade temporal e sincronização intra e intermídia), os requisitos vão além de uma maior largura de banda disponível.

Assim, para a implantação de ACMs, devem ser incorporados, também, requisitos mais rígidos para latência e *jitter* da rede para garantir qualidade exigida para estes ambientes, sobretudo os que empregam ferramentas colaborativas multimídia síncronas. Por outro lado, para utilizar tais ACMs através da *Internet*, que oferece o serviço de transmissão de melhor esforço (*best effort*) [12] pode se tornar necessário lançar mão de mecanismos para garantir níveis mínimos de qualidade de serviço.

Fica evidente a necessidade da infra-estrutura de comunicação herdar as características e vantagens, como escalabilidade e flexibilidade para gerenciamento de grupos, do endereçamento *multicast*.

Além disso, é necessário que a rede também disponha de modo de transmissão confiável para suportar a troca de mensagens entre aplicações da coordenação entre ambientes multimídia que compõem um ACM. Esta confiabilidade, sobre canais de comunicação *multicast*, pode ser obtida através do emprego de um protocolo de *multicast* confiável que implementa mecanismos para ordenação, detecção e correção de erros, já que o modelo de comunicação *multicast* não provê transporte confiável nativo para os dados.

Assim, o modelo de comunicação baseado no endereçamento *multicast*, podendo fazer uso de mecanismos adequados para prover comunicação entre sub-redes *multicast* através de esquemas de tunelamento e viabilizar transferência de dados confiável e ordenados, mostra-se adequado para compor a infra-estrutura de comunicação que suporte os ambientes colaborativos multimídia através da *Internet*. De uma forma geral, este modelo mostra-se adequado tanto para comunicação multimídia quanto para troca de informações de controle e coordenação em ACMs, através de redes comutadas por pacotes.

Para implementar a confiabilidade na comunicação *multicast* deve-se empregar um protocolo que forneça tal serviço. Dentre os protocolos estudados, acredita-se que o LRMP tenha as características adequadas e desejadas para prover a infra-estrutura de comunicação confiável necessária para a coordenação nos ambientes colaborativos multimídia pretendidos. Sua implementação está disponível sob forma de uma biblioteca reutilizável Java para utilização em aplicações comerciais e acadêmicas.

Capítulo 4

FERRAMENTAS E APLICAÇÕES PARA AMBIENTES COLABORATIVOS MULTIMÍDIA

Com os avanços recentes tanto na infra-estrutura de comunicação quanto nos sistemas computacionais de *desktop*, os ambientes colaborativos multimídia vêm sendo requisitados nos mais diversos cenários.

Neste sentido, busca-se ambientes colaborativos multimídia flexíveis e com suporte a um conjunto básico de mídias, capazes de atenderem necessidades em cenários como reuniões de pequenos grupos, aulas virtuais e seminários via *Internet* para grandes platéias.

Entende-se que o conjunto básico de mídias suportado pelos ACMs a serem utilizados nos cenários acima citados deva compreender,

basicamente, áudio, vídeo, texto, quadro branco e área de trabalho compartilhados. Estas mídias, no contexto de cada cenário, podem ser disponibilizadas, conforme necessidade e propósito, de forma individual ou combinações de mídias.

Por exemplo, para uma reunião informal entre um grupo de alunos para tratar de um assunto qualquer, um canal de áudio e um de vídeo podem compor o conjunto de mídias suportadas, enquanto que para uma aula virtual outras seriam necessárias. Sobre estas mídias, o ACM deve oferecer um conjunto de funcionalidades que dependem da natureza e objetivos de cada ACM a ser usado em determinado cenário. Assim, o ACM que se pretende deve ser flexível para atender as necessidades geradas pelas especificidades de cada cenário onde se pretende utilizá-lo.

Assim, foram realizados pesquisas, estudos e testes de ferramentas e aplicativos (desenvolvidos e em desenvolvimento) que pudessem compor um ACM para vários cenários, sobretudo para os supracitados. As pesquisas, e testes concentraram-se em ferramentas e aplicações gratuitas e que pudessem se integrar, compor ou servir de base para o ambiente pretendido.

Por causa da grande quantidade de ferramentas (existente e em desenvolvimento) pesquisada e testada [46], estas foram agrupadas em três classes distintas: Plataformas completas, aplicativos de mídias, e ferramentas de apoio à comunicação.

Neste capítulo são apresentados metodologia e resultados das pesquisas, estudos e testes, critérios para agrupamento das ferramentas e critérios empregados para avaliação dos resultados.

Ao final são apresentadas as conclusões, destacando-se as características desejáveis encontradas nas ferramentas e aplicativos existentes, bem como suas principais deficiências.

4.1 Testes e Avaliação de Ferramentas e Aplicativos

Devido à complexidade de um ambiente colaborativo multimídia adequado aos cenários pretendidos, estes podem ser compostos por ferramentas e aplicativos com funções bastante específicas.

Assim, para permitir avaliação mais clara e objetiva dos resultados dos estudos, pesquisas e testes, foram definidos e utilizados alguns critérios considerados fundamentais para as ferramentas e aplicativos e, conseqüentemente, para o ACM pretendido. Dentre os critérios adotados destacam-se os que seguem [07]: suporte a transmissão *multicast* sem necessidade de servidor, sistema gratuito e com código-fonte disponível, interoperabilidade e utilização de padrões, suporte a diferentes políticas de coordenação, portabilidade, escalabilidade e estabilidade do sistema.

Observando cada ferramenta e aplicativo segundo o conjunto de critérios acima se pôde ter uma idéia da relevância deste para o ACM pretendido.

Para facilitar a avaliação, as ferramentas testadas e avaliadas foram agrupadas em três classes distintas: *plataformas completas*, *aplicativos de mídias*, e *ferramentas de apoio à comunicação*.

As *plataformas completas* são aquelas que apresentam suporte a um conjunto de mídias e funcionalidades para um conjunto de cenários. Um exemplo de *plataforma completa* é o ReLaTe [24] que apresenta uma

interface com suporte a áudio, vídeo, edição de texto compartilhado e quadro branco.

Os *aplicativos de mídias* são aquelas que suportam a comunicação de um determinado tipo de mídia, podendo ser parte integrante de uma *plataforma completa*. Um exemplo de *aplicativo de mídia* é o RAT [25] destinado à comunicação síncrona de áudio e que pode ser integrado a vários sistemas de conferência (classificados como *plataformas completas*) como, por exemplo, o ReLaTe.

As *ferramentas de apoio à comunicação* são aquelas que não são utilizadas exatamente para comunicação entre participantes através de uma mídia específica, mas que servem para apoio a esta comunicação. Dois exemplos de *ferramentas de apoio à comunicação* são o *WBimport* que permite importar diferentes figuras no aplicativo WB [26], e o *Refletor RAT* que permite que sítios sem acesso ao *MBone* tenham acesso a sessões de áudio difundidas pelo *MBone* utilizando a ferramenta RAT.

4.2 Plataformas Completas

Foram testadas várias plataformas completas que suportam o trabalho colaborativo multimídia síncrono. As principais são apresentadas na Tabela 4 e os resultados mais relevantes são comentados a seguir.

<i>Classificação das Principais Ferramentas e Aplicativos Avaliados</i>							
Legenda:	Critérios Considerados						
<i>S – sim</i> <i>N – não</i> <i>I – informações insuficientes para determinado critério</i>	Multicast	Coordenação e controle de palavra	Código aberto	Segue padrões	Interoperabilidade	Portabilidade	Estabilidade

Mídia Suportada	Ferramenta							
Áudio e vídeo	TangoInteractive	S	N	N	S	S	S	I
Quadro branco e compartilhamento de área de trabalho	TeamWare	N	S	N	I	I	S	I
Salas de trabalho, áudio, compartilhamento de área de trabalho e quadro branco	MushRoom	S	N	N	I	S	S	I
Áudio, vídeo e apresentações	WindowsMedia	S	N	N	N	N	N	S
Áudio, vídeo, compartilhamento, quadro branco e chat	VRVS	S	N	N	S	S	S	N
Áudio e vídeo	CuSeeMe	S	N	N	S	S	N	N
Áudio, vídeo, quadro branco, transferência de arquivos e chat	Netmeeting	N	N	N	S	S	N	S
Áudio, vídeo, chat, transferência de arquivos e compartilhamento	Sunforum	N	N	S	S	S	N	S
Áudio, vídeo, chat, transferência de arquivos e compartilhamento	DC-Share	S	N	N	S	S	N	S
Áudio, vídeo, chat, transferência de arquivos, voice-mail	Video VoxPhone	S	S	N	S	S	N	S
Áudio, vídeo, chat, texto e quadro branco	Relate	S	N	S	N	N	N	S
Áudio e vídeo	RV	S	N	N	S	S	N	N
Áudio e vídeo	IVS	S	N	N	S	S	N	N
Áudio, vídeo, votação, controle de palavra, reserva de recursos, anúncio de sessões.	MinT	S	S	S	S	N	N	N
Áudio, vídeo, quadro branco, Chat, anúncio de sessões.	OpenMASH	S	S	S	S	S	N	N
Wb, votação, chat	Mdesk	S	N	N	S	N	S	S

Tabela 4: Critérios e classificação das principais plataformas completas avaliadas [07]

Dentre as testadas vale destacar algumas que possuem suporte a várias mídias, inclusive áudio e vídeo síncrono: ReLaTe, VRVS [27], DC-Share [28], VoxPhone [29], SunForum [30], Mint [31].

A plataforma VRVS apresenta capacidade para comunicação através de aplicativos de áudio, vídeo, mensagens instantâneas de texto e quadro branco, podendo fazer uso de aplicativos específicos existentes, como VIC (*Videoconference Tool*) e RAT (*Robust Audio Tool*). A plataforma suporta a comunicação multiponto através do uso de refletores MCU (*Multi Control Unit*) H.323 e, através de refletores, torna possível integrar aplicativos MBone e H.323, como o NetMeeting [32] e RAT (verificada durante os testes). Como limitações desta plataforma pode-se citar a falta do serviço de coordenação para o acesso aos recursos compartilhados, os problemas de estabilidade encontrados nos aplicativos de mídia executados em sistemas operacionais MS Windows e a necessidade de se utilizar MCU para a realização de conferências multiponto.

O DC-Share é uma plataforma para conferência multiponto para computadores pessoais com sistema operacional Unix desenvolvido a partir do Sunforum. Oferece serviços de comunicação multiponto de áudio e vídeo sobre TCP/IP e UDP/IP, compartilhamento de software e de *whiteboard*, transferência de arquivos e comunicação via *chat*. Como limitações desta plataforma figuram o fato de não ser gratuita, não ter o código-fonte acessível, não implementar políticas de controle de palavra e necessitar de MCU para permitir a comunicação multiponto.

Nesta mesma linha está a plataforma Vídeo VoxPhone, que dispõe de comunicação através de canais de áudio e vídeo, *chat*, *voice* e-mail, lista de usuários *on-line*, possibilidade de seleção de CODEC (codificador – decodificador) visando melhorar o desempenho, transferência de arquivos, e compatibilidade com outros programas que sigam o padrão H.323. Entretanto, também não é uma ferramenta gratuita e não aborda a coordenação de forma adequada.

A plataforma MinT, por sua vez, é um conjunto de ferramentas multimídia que permitem estabelecer e controlar sessões multimídia através da *Internet*, sendo que sua arquitetura é completamente distribuída sem a existência de componentes centrais. Esta plataforma oferece suporte a áudio, vídeo e outras mídias. Permite também interface com ferramenta SDR [25] e implementa ferramenta para controle de palavra. Embora esta plataforma tenha apresentado características desejáveis como integração com ferramentas tradicionais do *MBone*, código-fonte disponível, plataforma gratuita, implementação de coordenação e interoperabilidade, não foi possível testar a plataforma completamente devido a problemas encontrados com a ferramenta de controle de palavra.

Finalmente, a plataforma ReLaTe (*Remote Language Teaching*) combina os componentes de vídeo conferência, a partir dos aplicativos tradicionais do *MBone* em uma única interface de usuário. Dentre os aplicativos existentes, foram utilizados VIC, RAT, WBD (*Whiteboard*), NTE (*Network Text Editor*) e SDR (*Session Directory*) apresentando resultados bastante razoáveis quanto à estabilidade. Em relação aos demais critérios considerados, a limitação desta plataforma fica por conta da falta de uma ferramenta que implemente os serviços de coordenação, principalmente o controle de palavra.

4.3 Aplicativos de Mídia

Foram testados vários aplicativos de mídia que suportam o trabalho colaborativo multimídia síncrono. A avaliação dos principais aplicativos segundo os critérios utilizados é apresentada na Tabela 5.

Classificação das Principais Ferramentas e Aplicativos Avaliados		Critérios Considerados						
Legenda:								
<i>S – sim</i> <i>N – não</i> <i>I – informações insuficientes para determinado critério</i>								
Mídia Suportada	Ferramenta	Multicast	Coordenação	Código aberto	Segue padrões	Interoperabilidade	Portabilidade	Estabilidade
Video	VIC	S	N	S	S	S	N	N
Áudio	RAT	S	N	S	S	S	N	N
Áudio	VAT	S	N	S	S	N	N	N
Wb	WB	S	N	S	N	N	N	S
wb	WBD	S	N	S	N	N	N	S
Texto	NT	S	N	S	N	N	N	S
Texto	NTE	S	N	S	N	N	N	S
MP3 multicast	LiveCaster	S	N	N	S	S	N	S
Browser compartilhado	XwebTeach	S	N	S	S	N	N	S
Receptor MP3	PlayRTPMPEG	S	N	N	S	N	N	S
Video	NV	S	N	S	S	S	N	N
Áudio	FreePhone	S	N	N	S	S	N	N
Votação	Mpoll	S	N	S	N	N	N	S
Browser compartilhado	MMosaic	S	N	S	S	N	S	S

Browser compartilhado	Shared mosaic	S	N	S	S	N	S	S
Quadro branco	Teledraw	S	N	S	N	N	S	S
Importa FIGs no quadro branco	Wbimport	S	N	S	S	S	N	N
Compartilhar HTMLs	Webcanal	S	N	N	S	N	N	N
Bibliotecas e aplicativos Java para áudio e vídeo	JMF	S	N	S	S	S	S	S

Tabela 5: Critérios e classificação dos principais aplicativos de mídia avaliados [07]

Dentre os aplicativos testados destacam-se os aplicativos *MBone* tradicionais [24] e aplicativos desenvolvidos em Java, sobre a API (*Application Program Interface*) JMF (*Java Media Framework*) [33], que possuem suporte às mídias pretendidas citadas no início deste capítulo. Destaca-se o JMF pelo fato de apresentar características importantes para o desenvolvimento dos ambientes colaborativos multimídia para os cenários pretendidos, como pode ser visto na Tabela 5.

Os aplicativos tradicionais do *MBone*, como os para transmissão e recepção de áudio e vídeo, e para compartilhamento de editor de texto e quadro branco, apresentam como características desejáveis o suporte à transmissão *multicast* e *unicast*, o fato de serem gratuitos e terem seus códigos-fonte disponíveis e a interoperabilidade. Suas limitações se concentram principalmente na estabilidade, sobretudo para as implementações para sistema operacional MS Windows.

Já as ferramentas para transmissão de áudio e vídeo desenvolvidas a partir do JMF, disponíveis em [33], apresentaram boa estabilidade, e todas as demais vantagens das ferramentas tradicionais do *MBone* descritas acima. Por serem desenvolvidas sobre a plataforma Java, elas também apresentam boa portabilidade.

4.4 Ferramentas de Apoio

Durante este trabalho foram testadas várias ferramentas de apoio à comunicação e colaboração multimídia síncrona. A avaliação das

principais ferramentas de apoio segundo os critérios utilizados é apresentada na Tabela 6.

Classificação das Principais Ferramentas e Aplicativos Avaliados								
Legenda:		Critérios Considerados						
<i>S – sim</i> <i>N – não</i> <i>I – informações insuficientes para determinado critério</i>		Multicast	Coordenação	Código aberto	Segue padrões	Interoperabilidade	Portabilidade	Estabilidade
Anúncio e lista de sessões	SDR	S	N	S	N	N	N	S
Anúncio e lista de sessões	SD	S	N	S	N	N	N	S
Anúncio e lista de sessões	Multikit	S	N	N	S	S	N	S
Floor control	QB	S	S	S	S	N	N	N
Tunelamento multicast	Mtunnel	S	N	S	S	S	S	N
Tunelamento multicast	LiveGate	S	N	N	N	N	N	S
Tunelamento multicast	Mrouted	S	N	S	S	S	N	S
Acesso ao Mbone	Refletor Multi- Uni	S	N	S	S	S	S	N
Acesso Mbone (áudio)	Refletor RAT	S	N	S	S	S	N	N
Grava e reproduz sessões	NMod	S	N	S	S	S	S	I
Ajuste da qualidade de áudio e vídeo	Quass	S	N	S	S	S	N	I
Anúncio e controle de sessões	SPAR	S	N	S	S	S	S	S
Grava e reproduz sessões	MMRC	S	N	S	S	S	N	I
Grava A/V multicast	Mcontrol	S	N	S	S	S	N	I
Gerenciamento e controle de conferências	ConfMan	S	N	S	S	S	N	N
Grava e reproduz sessões multicast	MMOD	S	N	S	S	I	S	I
Grava e reproduz sessões multicast	MvoD	S	N	S	S	I	S	I
Grava e reproduz sessões RTP	WrtP	S	N	S	S	I	S	I
Monitora RTP	Msessmon	S	N	S	S	N	N	I
Monitora RTP	MultiMON	S	N	S	S	N	N	I

Tabela 6: Critérios e classificação das principais ferramentas de apoio avaliadas [07]

Dentre as ferramentas testadas destacam-se o SDR (*Session Directory*) [25], *ConfMan* (*Conference Manager*) [34], *Mrouted* [35], *MTunnel* [36] e *QB* (*Question Board*) [37].

As ferramentas SDR e *ConfMan* realizam o controle de sessão, permitindo anúncio, entrada e saída dinâmica de sessões, controlar acesso, convidar e remover participantes em sessões, e atuar na

privacidade e segurança das informações que trafegam entre os participantes da sessão. Ambas ferramentas testadas apresentaram comportamento estável, podendo gerenciar sessões com suporte a diferentes mídias, além de facilmente também poderem ser integradas a diferentes aplicativos. Por exemplo, pode ser definida o aplicativo desejado para um canal de vídeo de uma determinada sessão. Também foi observado que ambas são gratuitas, suportam o endereçamento *multicast* e apresentam implementação em diversos sistemas operacionais, incluindo, Windows, Linux, Irix e Solaris.

As ferramentas Mouted e Mtunnel, por sua vez, permitem que sítios sem acesso ao *MBone* o façam através de túneis *unicast*. Os túneis podem ser feitos de maneira simples e apresentaram bons resultados interconectando ilhas *multicast* que não se comunicavam. O Mouted apresenta versão disponível apenas para sistema Unix, boa estabilidade e é gratuito. O Mtunnel é desenvolvido em Java (sendo, portanto, portátil para outros sistemas operacionais), é gratuito e apresenta interface Web nas pontas dos túneis (onde disponibiliza informações sobre as sessões *multicast* acessíveis) e gerencia padrões de mídia (por exemplo, traduzir áudio PCM para outro e mixar o áudio de várias fontes).

A ferramenta QB é utilizada para introduzir controle de palavra (*Floor Control*) em sessões *multicast*, através de uma abstração utilizada pelas ferramentas do *MBone*, como o VIC e VAT, conhecido como *Coordination Bus* [38]. O QB permite que os participantes de uma sessão demonstrem interesse em enviar áudio e vídeo. Este pedido é enviado ao sítio do moderador que concede o direito de palavra aos participantes da conferência. O moderador controla a conferência e tem mais privilégios que os usuários comuns. Esta ferramenta é gratuita, tem seu código-fonte disponível, se integra aos aplicativos VIC e VAT e viabiliza a coordenação em sessões para trabalho colaborativo síncrono (áudio e vídeo).

Entretanto, a ferramenta não interopera com outros aplicativos do *MBone*, não se mostrou estável nos testes e não suporta outras políticas de coordenação. A portabilidade da ferramenta também fica comprometida por conta da linguagem de programação utilizada.

4.5 Conclusões do Capítulo

Diante do objetivo de compor ambientes colaborativos multimídia flexíveis e com suporte a um conjunto básico de mídias, capazes de atenderem necessidades em cenários como reuniões de pequenos grupos, aulas virtuais e seminários via *Internet* para grandes platéias, foram realizados pesquisas e testes de plataformas, aplicativos e ferramentas já existentes e em desenvolvimento. Embora tenham sido testadas várias ferramentas, plataformas e aplicativos, apenas as consideradas mais relevantes foram apresentadas e sua avaliação comentada ao longo deste capítulo.

Dentre os resultados dos testes [46] [47] pode-se destacar que nenhuma plataforma completa atendeu plenamente a necessidade em relação aos serviços de coordenação pretendidos, principalmente controle de palavra.

Dentre os aplicativos de mídia e ferramentas de apoio testadas foram destacadas neste capítulo as consideradas relevantes para comporem os ambientes e suas funcionalidades para cada cenário pretendido. Entretanto, a única ferramenta testada que busca prover suporte para controle de palavra apresentou problemas de integração com os aplicativos de áudio (VAT) e vídeo (VIC), não atendendo as demandas dos cenários apresentados.

Além disso, os aplicativos de mídia desenvolvidos a partir da API Java JMF e disponíveis em [33], se mostraram bastante estáveis, funcionando sem problemas em sistemas operacionais Windows e Linux por muitas horas consecutivas. Os mesmos são portáteis, interoperam com outras ferramentas através do uso de algoritmos de codificação padronizados e largamente utilizados, suportam modos de comunicação *unicast* e *multicast*, e tem seu código disponível na *Internet*.

Assim, chama-se a atenção para uma necessidade identificada nos testes e avaliação dos resultados sintetizados ao longo deste capítulo: o controle de palavra para coordenar o acesso a recursos compartilhados numa sessão de trabalho colaborativo multimídia síncrono.

Embora outras lacunas e necessidades também tenham sido identificadas durante os testes (como, por exemplo, integração a Web, facilidade de instalação, configuração e utilização), propõe-se neste trabalho abordar apenas o problema do controle de palavra. Esta proposição baseia-se nos resultados dos testes das plataformas, aplicativos e ferramentas, realizados nos cenários considerados: reuniões de pequenos grupos, aulas virtuais e seminários via *Internet* para grandes platéias.

Durante os testes verificou-se que a colaboração síncrona fica bastante prejudicada, senão inviabilizada, quando não se utiliza uma ferramenta para prover políticas de controle de palavra adequadas a cada um dos cenários considerados.

Para atender ao quesito flexibilidade e permitir que o ambiente possa se adequar a cada um dos cenários e, eventualmente, integrar ferramentas e aplicativos já existentes, propõe-se que um ACM seja formado por ambientes multimídia distribuídos em cada sítio que participa de uma sessão. Cada ambiente multimídia, num determinado

sítio, é uma coleção de aplicativos e ferramentas multimídia que funcionam de forma coordenada e controlada através de um barramento local [38], que será detalhado no capítulo 5.

Analisando a tendência e quantidade de aplicativos e ferramentas já desenvolvidas, bem como visando tornar os ambientes pretendidos portáteis entende-se que o protótipo deste trabalho deve ser desenvolvido utilizando linguagem de programação Java [23].

Capítulo 5

COORDENAÇÃO E CONTROLE DE PALAVRA EM AMBIENTES COLABORATIVOS MULTIMÍDIA

Quando se pretende estender uma comunicação, a partir da máquina-máquina e homem-máquina para uma comunicação homem-homem, através de um sistema de comunicação eletrônica, precisa-se levar em conta as diferentes formas e estilos de comunicação humana.

Não se pode simplificar o problema da comunicação a ponto de incorrer no erro de considerar as duas formas básicas de conferências como sendo pequena (controlada e privada) e grande (não controlada e pública) [39].

Decompondo o problema da comunicação homem-homem, através de um sistema de comunicação eletrônica, utilizando aplicações multimídia colaborativas, segundo quatro diretrizes básicas [39], que são Criação, Seleção de participantes, Comunicação e Coordenação, pode-se dar atenção a cada diretriz, independente das outras três. Neste capítulo estará sendo abordada a diretriz Coordenação, aplicada aos ambientes colaborativos multimídia.

Durante as pesquisas, estudos, análises e testes de ferramentas para trabalho colaborativo, realizadas e apresentadas no capítulo anterior, observou-se que, de fato, a coordenação, sobretudo o controle de palavra, em conferências multimídia, além de ser de fundamental importância para a eficácia da colaboração, deve suportar diferentes estilos de comunicação e cooperação entre os participantes. Foi constatado também que a coordenação figura como uma das lacunas existentes nas ferramentas gratuitas existentes para compor um ACM, inexistindo na maioria das aplicações e ferramentas para *groupware* síncrono testadas.

Portanto, neste capítulo será apresentado um estudo sobre a coordenação em ambientes colaborativos multimídia. Neste contexto, focar-se-á o trabalho especificamente no controle de palavra para os recursos compartilhados nestes ambientes. Também serão abordados outros paradigmas de controle, mecanismos e políticas de controle de palavra bem como as principais características e requisitos para realizar o controle de palavra num ACM.

5.1 Coordenação em Conferências Multimídia

A coordenação, como definida em [03], é um processo de planejamento interativo entre dois ou mais usuários, formando um grupo para alcançar um objetivo comum.

A coordenação está relacionada com o objetivo de cooperação entre os sítios que compõem um ambiente colaborativo. Esta cooperação, por sua vez, é entendida como a ação conjunta dos usuários para o benefício mútuo. Assim, coordenação e cooperação entre usuários de sistemas multimídia ligados em rede mantêm o processo de colaboração multimídia [03].

Outra definição apresentada em [03], diz que coordenação é o ato de gerir interdependências entre atividades predefinidas para alcançar um objetivo, observando os atores (pessoas) e agentes (procedimentos computacionais), identificando objetivos, mapeando estes para atividades, e gerenciando a interdependência entre atores e atividades para se alcançar com sucesso o trabalho em grupo.

Assim, a coordenação busca resolver, numa conferência multimídia, problemas relacionados ao controle de palavra, sincronização de mídias e estados de aplicações, interdependência de atividades e sessões, podendo também incorporar funções de gerenciamento de sessões.

5.1.1 Coordenação Centralizada e Distribuída

A coordenação pode se dar de maneira centralizada e permitir o acesso aos recursos e a sessão de forma determinista. O conjunto de protocolos mais difundido para controle de conferências com estas características é o H.323. O H.323 é uma recomendação da ITU (*União Internacional de Telecomunicações*) que fornece um arcabouço de

comunicação para áudio, vídeo e dados através de redes comutadas a pacotes [10]. Associados à recomendação H.323 estão vários outros protocolos com funções específicas.

Por exemplo, o padrão H. 245 é responsável pelo controle de mídia, enquanto o padrão H.261 especifica a codificação de vídeo a taxas maiores que 64Kbps e o H.263 a codificação de vídeo a taxas menores que 64Kbps. Uma limitação que pode ser citada é a necessidade do uso de MCU – *Multipoint Control Unit* para implementar, por exemplo, os serviços especificados pelo padrão H.243.

Na arquitetura baseada em MCU, pequenas conferências são geralmente realizadas com configuração manual com um único MCU no centro da topologia em estrela [40], não apresentando boa escalabilidade, robustez e flexibilidade. Dessa forma, esta arquitetura se torna imprópria para sessões grandes, com grupos de participantes dinâmicos, que necessitem de múltiplos estilos de interação ou que são realizadas em redes heterogêneas.

Assim, para a realizar a coordenação de ambientes colaborativos multimídia, sobretudo *groupware* síncrono, que contemplem desde pequenos até grandes grupos, com entradas e saídas dinâmicas de participantes, sobre uma infra-estrutura de comunicação bastante heterogênea, como a *Internet* ou redes metropolitanas ou corporativas, comutada a pacotes com serviço de entrega não confiável e que apresenta atrasos e variações no tempo de entrega dos pacotes, esta arquitetura centralizada baseada em MCU não é a mais adequada.

Para atender os requisitos e demandas destes cenários torna-se necessária uma arquitetura de coordenação distribuída, que deve ser viabilizada por um modelo de conferências adequado.

5.2 O Modelo LWS (*Light-Weight Session*)

O modelo “*Light-Weight Sessions*”, introduzido por [39] foi concebido para criar condições para conferências e trabalho colaborativo através da *Internet* sem os problemas comuns às abordagens orientadas a conexão ou com nós centrais responsáveis pelo controle das conferências como, no casos dos MCUs.

O modelo de comunicação para aplicações colaborativas *Light-Weight Sessions*, referenciado a partir daqui simplesmente como LWS, não sobrecarrega os usuários, tem toda a escalabilidade e robustez do IP, se ajusta à conectividade intermitente e apresenta um bom desempenho mesmo quando a largura de banda dos usuários é bastante diferente.

5.2.1 Partes que Compõem Conferências Segundo o Modelo LWS

As principais partes que estruturam uma conferência LWS são IP (com a entrega *best-effort*, sem garantia), IP *Multicast*, recuperação da temporização através de adaptação no receptor (compensação da variação dos atrasos dos pacotes de áudio), e uma camada de transporte.

Dessa forma, o modelo LWS herda a simplicidade do modelo de transmissão do IP *Multicast*, onde os receptores anunciam interesse numa sessão, os transmissores apenas enviam e os roteadores cooperam para enviar os dados do transmissor apenas aos receptores interessados.

Assim, simplifica-se a criação de uma conferência para os usuários: associa-se uma conferência a um endereço *multicast* e se permite que o usuário possa entrar e sair da conferência sem enumerar ou conhecer todos os demais participantes (anunciando interesse em receber dados

dos transmissores de determinado endereço) a qualquer momento. A conferência, segundo este modelo, tem uma identidade visível à camada de rede, o que permite que tarefas como gerenciamento da sessão e distribuição sejam realizadas pela pilha de protocolos de rede.

Outra característica suportada pelo modelo LWS é a recuperação da transmissão através da adaptação do receptor, que se torna necessária por conta dos requisitos de transmissão existentes para os encontros síncronos (*real-time meeting*).

Ao contrário do tráfego de dados tradicional, a estrutura de tempo em transmissões em tempo real tem importância e, como a rede, via de regra, é compartilhada, existe competição, congestionamento e atraso na recepção, podendo ocorrer a distorção na seqüência temporal.

Para resolver esta questão da distorção na seqüência temporal pode-se atuar na rede, tornando a dedicada ou utilizar um esquema de compartilhamento determinista, o que restringiria bastante os cenários para aplicação do modelo. Para não atuar diretamente na rede, a solução pode ser a marcação dos pacotes de dados na origem, informando dados temporais. Com isso, permite-se a reconstrução da mídia no receptor antes de apresentá-la ao usuário.

Para transportar os dados em tempo real, com seus diferentes requisitos de comunicação, uma camada de transporte eficiente é necessária para este modelo.

A idéia é que a camada de transporte suporte a semântica da aplicação para otimizar seu desempenho através da rede e nos sistemas finais.

Muitas vezes, do ponto de vista da aplicação não é interessante separar a aplicação do protocolo de transporte. Integrar funções do

protocolo tradicional de transporte como confiabilidade e controle de fluxo requer que a aplicação controle o tamanho do pacote enviado à rede. Uma alternativa a este problema é o ALF (*Application Level Framming*) [41] que propõe um tamanho de pacotes fixo, sem a definição de algoritmos para controle de fluxo, ou retransmissão. Estes aspectos são definidos pela aplicação, permitindo maior desempenho na transmissão dos dados.

Por exemplo, o RTP (*Real Time Protocol*) [17] não está disposto como uma camada na forma do tradicional modelo de comunicação OSI [16] mas sim como uma árvore hierárquica de protocolos interdependentes, como apresentado na Figura 10, permitindo criar uma camada muito fina para o transporte de dados sem sobrecarregar o projeto da aplicação.

As folhas da árvore hierárquica representam a especificação do formato do “*payload*” do protocolo RTP e definem parâmetros detalhados específicos da aplicação, como a sintaxe de codificação.

Por exemplo, as especificações do formato do “*payload*” definem como os fluxos de bits do H.261, MJPEG e MPEG são enquadrados em fluxos de pacotes RTP. Além disso, o formato do “*payload*” pode ser decomposto, conforme visto na Figura 10, para formatos de vídeo como o LDCT [42].

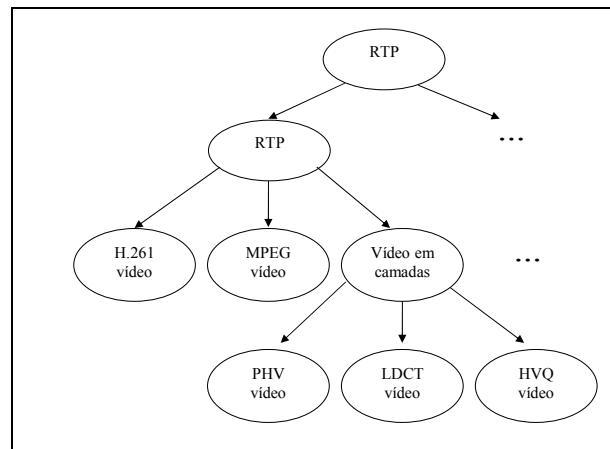


Figura 10: Arquitetura do protocolo RTP. O Protocolo de tempo real é composto por um conjunto interdependente de especificações arranjadas em uma classe hierárquica para suportar o modelo ALF - *application-level framing*

5.2.2 Sessões Multimídia e o Modelo LWS

No modelo LWS uma sessão é caracterizada por um endereço *multicast* e uma porta, através da qual os dados são transmitidos por um ou mais transmissores até todos os receptores que anunciaram interesse em receber os dados desta sessão.

Todos os sítios que participam de uma sessão, quase periodicamente, transmitem pacotes de sessão contendo identidade, informações de recepção e de sincronização, caracterizando um protocolo *soft-state* (que é baseado num modelo de “anunciar/ouvir” com envio de mensagens *multicast* periódicas para o grupo) e um modelo de comunicação distribuído. Isto faz com que o modelo LWS obtenha boa robustez em relação a falhas na rede.

5.2.3 Considerações Finais em Relação ao Modelo LWS

O modelo LWS, por estar apoiado em um modelo de comunicação leve, *soft-state* e descentralizado, suportar endereçamento de grupo,

contar com protocolo eficiente para transporte dos dados, e por suportar correções temporais na recepção, mostra-se bastante adequado para conferências e trabalho colaborativo através de redes heterogêneas, complexas, com diversos modos de falha e múltiplos níveis de confiabilidade, como a *Internet* [43]. Este modelo é útil e adequado tanto para conferências pequenas, quanto para grandes com grupos dinâmicos, com até centenas de participantes mantendo, contudo, boas características de robustez, estabilidade, flexibilidade e baixa sobrecarga no transporte dos dados.

As ferramentas para conferência e trabalho colaborativo, foco de pesquisas, estudos, análises e testes realizados seguem, em sua grande maioria, o modelo LWS. Dessa forma foi possível verificar as vantagens deste modelo de conferência em relação ao primeiro – com o controle realizado de forma centralizada e múltiplas conexões ponto-a-ponto (como por exemplo, as ferramentas que utilizam MCUs [48]) para multiplexar e filtrar os fluxos de mídia, no centro de uma topologia em estrela para possibilitar a comunicação ponto a ponto.

Entretanto, várias das aplicações de trabalho colaborativo multimídia testadas e analisadas, na sua grande maioria gratuitas, e que fazem uso das vantagens e características do modelo LWS, não apresentam serviços de coordenação, prejudicando e, muitas vezes, inviabilizando a colaboração.

Testes e colaborações realizadas a partir de pequenos grupos, utilizando, em sua maioria, ferramentas desenvolvidas segundo o modelo LWS [39], com objetivos comuns de complexidade relativamente baixa, como socializar experiências de pesquisa e discutir resultados de simulações feitas em laboratório, mostraram-se de pouca eficiência

quando o ambiente colaborativo era composto apenas por ferramentas para comunicação de mídias, sem ferramenta para o controle de palavra.

Contudo, se o controle de palavra existir, deve suportar mecanismos para permitir diferentes estilos de comunicação e cooperação, empregando políticas de coordenação que se adequem e atendam as necessidades e especificidades [39] de cada cenário. Como exemplos, podem ser citadas reuniões entre professor e um grupo de alunos para viabilizar uma orientação para atividades acadêmicas; e reuniões com necessidade de controle mais rígido como, por exemplo, a reunião de gerentes de filiais de uma empresa para discutir, com a equipe de planejamento estratégico da matriz, um novo plano e estratégias de produção a serem implantados.

5.3 Controle de Palavra

O presente trabalho busca focar-se num aspecto específico da coordenação de conferências multimídia: no controle de palavra de ferramentas para *groupware* síncrono [01].

O problema está, basicamente, em como, quando e porquê os participantes interagem num ambiente colaborativo multimídia enquanto trabalham de forma colaborativa, ao mesmo tempo, em tarefas comuns.

Uma definição não formal de controle de palavra é citada por [01]:

Controlar ou exercer influência sobre os direitos de discurso e recepção; ou, um dispositivo ou mecanismo usado para regular ou guiar a atividade coletiva dos membros de uma reunião.

Em outras palavras, controle de palavra é uma tecnologia para lidar com conflitos em ambientes colaborativos, que ajuda a coordenar atividades conjuntas e concorrentes entre pessoas e seus processos computacionais que interagem, como um revezamento ordenado em conversações ou atualizações de escrita em arquivos compartilhados, preservando a coerência local e remota dos dados [02].

Assim, palavras, por sua vez, são acessos e permissões de manipulação temporários, capazes de gerenciar acesso a dados multimídia.

O controle de palavra abranda condições de competição em sessões permitindo ou negando o envio, recepção ou manipulação de dados compartilhados ao longo da sessão. O controle de palavra desloca o paradigma de controle de concorrência sobre dados para o domínio da cooperação entre pessoas e agente inteligentes de sistemas multimídia em rede.

A implementação do controle de palavra pode ser realizada através de duas estratégias: o *Controle de Palavra Otimista* que permite conflitos, mas provê meios para resolvê-los; e o *Controle de Palavra Pessimista* que segue rigorosamente a premissa de evitar o conflito. O conflito ocorre quando mais de um usuário tenta acessar um determinado recurso compartilhado num mesmo momento.

Notadamente, o controle de palavra é um conceito centrado no usuário e suas implementações necessitam estarem alinhadas com as expectativas do usuário e também com as restrições impostas pelo sistema e rede. Além de garantir a manipulação segura dos recursos compartilhados, o objetivo maior do trabalho colaborativo com controle de palavra é permitir a máxima sinergia entre os membros da sessão [02].

5.4 Modelo de Controle de Palavra

Para dar se ter uma visão comum de cada uma das entidades que compõem um ACM e de como estas se relacionam, é apresentado um modelo formal e taxonomia empregada em [04] e [05], onde um ambiente colaborativo \mathbf{E} é definido como sendo uma tupla

$$\mathbf{E} = \langle \mathbf{S}, \mathbf{R}, \mathbf{U}, \mathbf{F} \rangle$$

onde \mathbf{S} é um conjunto de sessões \mathbf{s} , \mathbf{U} é um conjunto de usuários (*hosts*, processos, agentes, participantes), \mathbf{R} é um conjunto de recursos compartilhados, e \mathbf{F} é um conjunto de “palavras” que controlam os recursos.

Já uma sessão $\mathbf{s} \in \mathbf{S}$, que compõe um ambiente colaborativo \mathbf{E} , e que oferece a infra-estrutura para a coordenação, cooperação e colaboração, é definida como sendo uma tupla

$$\mathbf{s} = \langle \mathbf{S}_{id}, \mathbf{T}_i, \mathbf{T}_e, \mathbf{A}_s, \mathbf{L} \rangle$$

onde \mathbf{S}_{id} é um identificador único em \mathbf{E} , \mathbf{T}_i é o tempo de início ou anúncio, \mathbf{T}_e é tempo de encerramento, e \mathbf{A}_s é uma lista de atributos que caracteriza a sessão \mathbf{s} no nível \mathbf{L} [4]. Uma conferência é um conjunto de sessões \mathbf{s}_i onde $i \geq 1$.

Uma sessão \mathbf{s} pode ser:

- a. *aberta*, permitindo a escolha de recursos, momento de encerramento de interações, e a adição ou remoção de partes ou mídias, ou

- b. *fechada*, composta por uma lista predeterminada de participantes convidados.

Sessões colaborativas podem variar bastante com relação ao estilo, como aulas, encontros de negócios, discussões de painéis, ou de audiência, causando impacto no tamanho, organização, fluxo de dados e modelo de interação.

Os recursos $r \in R$ podem estar localizados em um nó particular, distribuídos em seus componentes ao longo do conjunto de nós, ou replicados em todos os nós na sessão. Um recurso com características espaciais ou temporais pode ser uma aplicação, um *host*, uma entidade de rede compartilhada na colaboração, tais como um componente de interface com usuário (tele-apontador), um dispositivo de hardware ou um fluxo de vídeo ou áudio. Qualquer usuário ou recurso pode atuar como uma fonte de informações no ambiente de trabalho colaborativo.

Uma palavra $f \in F$, por sua vez, é definida como sendo uma tupla

$$f = \langle F_{id}, R_{id}, U_{id}, T_i, T_d, A_f \rangle$$

onde F_{id} representa um identificador único de uma palavra no ambiente E associado a um determinado recurso, identificado por R_{id} . A palavra f é atribuída a um usuário u identificado por U_{id} , no instante T_i e desativada no instante T_d . O parâmetro A_f denota uma lista de atributos da palavra.

Um determinado recurso R_{id} pode ter associado múltiplas palavras identificadas por F_{id} . Entretanto, cada palavra f controla exatamente um recurso. Assim, os pares (F_{id}, R_{id}) definem a granularidade do controle: uma palavra pode controlar toda a conferência, uma aplicação ou um

único objeto compartilhado. Assim, é possível refinar a granulosidade introduzindo sub-palavras para componentes específicos de um recurso que precisa ser controlado individualmente, como uma operação específica sobre um dispositivo ou o posicionamento de uma câmera remota.

Usuários $u \in U$ podem ser humanos ou agentes do sistema [04], e podem ocupar, neste modelo, um ou mais dos seguintes papéis de controle:

- a. Criador de Palavra (*Floor Originator – FO*) de um recurso $r \in R$ é o nó que introduz o recurso r na sessão e inicia o controle de palavra para r . Se o *FO* deixa a sessão s , o recurso r desaparecerá, a menos que ele tenha sido replicado;
- b. Coordenador de Palavra (*Floor Coordinator – FC*) é um árbitro para um recurso r , ou um moderador para a sessão que garante ou nega a palavra durante o tempo de uma sessão para o Detentor da Palavra (*Floor Holder – FH*);
- c. Detentor da Palavra (*Floor Holder – FH*), que mantém a palavra exclusivamente sobre r durante um tempo finito.

Estes papéis, de acordo com o modelo de sessão e propósitos, podem ser atribuídos estaticamente no início da sessão, ou perambular entre as estações dos usuários durante o tempo da sessão.

Os atributos de uma palavra f representados pelo parâmetro A_f compreendem estado, instanciação, regras de passagem e estratégias de acesso à palavra. Por exemplo, para certos recursos, como canais de áudio, o *FC* pode manter uma fila de requisições, que armazena as requisições de palavras e as serve de acordo com a política de acesso prevista.

Por simplicidade, neste trabalho, assumir-se-á que existe um usuário u por nó e uma palavra f alocada por recurso r .

Variações, como uma única palavra controlando múltiplos recursos, detenção simultânea de uma palavra por vários usuários, ou múltiplas palavras controlando um único recurso podem existir, mas não estarão sendo consideradas neste trabalho.

5.5 Características e Requisitos do Controle de Palavra

5.5.1 Controle de Palavra e Outros Paradigmas de Controle

É importante, além de descrever as tarefas de controle de palavra, distingui-lo de outros modelos para lidar com competição em ACM.

A primeira diferenciação a ser feita é em relação ao controle de sessão. Enquanto o controle de palavra controla o acesso a recursos compartilhados, o controle de sessão fornece informações sobre participantes, início e fim dos caminhos de comunicação, e problemas de compatibilidade acerca de uma mídia específica.

Já o controle de conferência complementa mecanismos de coordenação [03], como o controle de palavra, permitindo também o gerenciamento de usuários, serviços de diretórios, anúncio, convite e finalização. Em outras palavras, serviços integrados de controle de palavra e de sessão são chamados de controle de conferência.

Prover gerenciamento de sessão sem controle de palavra é o mesmo que uma infra-estrutura de transporte sem controle de tráfego [03], pois qualquer arquitetura que permita diferentes entidades cooperar, necessita de protocolos para mediar condições de disputa e evitar ou resolver conflitos e suas implicações.

Vale diferenciar o controle de palavra de outros esforços para lidar com conflitos em trabalho colaborativo [02]: o controle de concorrência e o controle de acesso; mesmo que a base comum para todos estes métodos de controle, no contexto de trabalho colaborativo remoto, é a *exclusão mútua distribuída*.

Alguns tipos de ACM, como os que permitem o trabalho colaborativo através de compartilhamento de gráficos e documentos de texto, são baseados na noção de base de dados de um espaço de trabalho conjunto, aos quais se aplica o modelo de controle de concorrência, cujo objetivo é a serialização das atividades. Entretanto, para fluxos de mídias contínuas, a exclusão otimista não é possível, e um esquema de controle deve evitar o conflito de uma forma “pessimista” [02], em vez de tentar resolvê-lo.

Já o controle de acesso é usado em sistemas de arquivos estáticos e as permissões sobre os recursos são permanentes. Por outro lado, as permissões do controle de palavra, durante as sessões que são dinâmicas, são efêmeras [02], podendo depender de diversos fatores, como, por exemplo, papel do usuário na sessão, fase atual da sessão, objetivos e interesses em cada momento da sessão.

5.5.2 Requisitos de Projeto

As opções de projeto para oferecer controle de palavra são inúmeras. Entretanto, as necessidades de utilização dos diferentes recursos podem ser úteis para restringir o número de caminhos de implementação.

Uma premissa do projeto é emular o mundo real do dia-a-dia num ambiente virtual colaborativo. Enquanto certas qualidades do ambiente real são intangíveis, novas características dos ambientes virtuais podem ser exploradas como, por exemplo, a combinação de várias tarefas numa

única sessão de trabalho. Atualmente, a maioria dos softwares colaborativos [02] é específica e dedicada a determinados tipos de tarefas e sem um controle de palavra muito sofisticado.

Os sistemas colaborativos vêm tornando-se cada vez mais polimorfos em relação à variedade de recursos que podem ser usados ao mesmo tempo. A Tabela 7, baseada em [02], classifica as aplicações típicas de acordo com os tipos genéricos de dados e tarefas e mostra a importância de requisitos de QoS e Controle de palavra. Os campos marcados com “x” indicam que o critério correspondente que se aplica:

- Trabalha com *recursos temporários* (T) ou se resulta em dados persistentes;
- É possível aplicar à *colaboração síncrona* (S);
- São recursos que necessitam de envio de dados *tempo real* (RT) e são sensíveis a atrasos;
- Toleram *perdas* (L) ou são sujeitos ao *jitter* (J);
- Requerem *alta* (H), *média* (M) ou *baixa* (L) *largura de banda* (BW).

A última coluna indica a dimensão da palavra (FD), ou seja, o tipo de conflito no domínio *temporal* (t), *espacial* (s), ou *funcional* (f) que o controle de palavra normalmente precisa resolver no trabalho colaborativo para um recurso específico [02]:

- Compartilhamento temporal indica possibilidade de conflitos causais ou temporais, por exemplo, o revezamento de dois participantes numa conversa;
- Conflitos de compartilhamento espacial surgem do uso da mesma área de apresentação ou espaço de armazenamento de uma área de trabalho compartilhada;

- Compartilhamento funcional se concentra em torno do uso das mesmas funções da aplicação que alternam o *status* ou conteúdo de um recurso compartilhado, por exemplo, a função para apagar arquivos de texto.

Tipo	T	S	RT	L	J	BW	FD
Texto							
Editor		X				L	<i>t,s,f</i>
Chat	X	X				L	<i>t</i>
E-mail						L	<i>t,f</i>
Agenda		X				L	<i>t,s</i>
Áudio	X	X	X	X	X	M	<i>t,f</i>
Imagens e Vídeo							
Estático				X		M,H	<i>s,f</i>
Em movimento	X	X	X	X	X	H	<i>t,f</i>
Gráficos							
Estático		X				M	<i>t,s,f</i>
Em movimento	X	X	X			M,H	<i>t,s,f</i>
WWW		X			X	L-H	<i>t,s,f</i>
Realidade Virtual	X	X	X		X	H	<i>t,s,f</i>

Tabela 7: Tipos de recursos e características de manipulação

Do ponto de vista da rede, pode-se identificar, basicamente, quatro classes de tráfego multimídia [02], as quais impõem diferentes requisitos de QoS ao ACM:

- Pacotes de controle (*control packets*) para controle de palavra e de sessão que, geralmente, consistem em pequenas quantidades de dados a serem transmitidos, mas que necessitam transmissão confiável;
- Pacotes tempo real (*real-time data*) para as mídias com requisitos temporais, que possivelmente suportam determinada taxa de perdas;

- Pacotes de dados não tempo real (*non real-time data*) para aplicações que possuem requisitos temporais não rígidos, e;
- Transferências de volumes de dados (*bulky data-transfers*) que normalmente acontecem nos momentos de replicação, re-sincronização que geralmente ocorrem no início, recuperação e finalização de uma sessão. Esta classe de transferências necessita de alta largura de banda e transmissão confiável, mas toleram certo nível de atraso.

Assim, devido à magnitude dos tipos de recursos, dos cenários de sessões e condições de redes, é necessário que o controle de palavra seja realizado através de uma arquitetura de coordenação que contemple os diferentes tipos de serviços necessários [02]. Então, para realizar o melhor serviço possível num ACM, considerando estes aspectos, deve-se favorecer decisões de projeto que permitam requisitos fundamentais [03] para a arquitetura de coordenação, e conseqüentemente para o controle de palavra, como:

- Simplicidade de implementação e manutenção;
- Escalabilidade em termos do número de usuários e *hosts*;
- Segurança em relação à troca de dados e informações de coordenação;
- Extensibilidade para novos recursos e modelos de sessões;
- Eficiência da coordenação em relação à baixa latência e *overhead* do protocolo;
- Confiabilidade em relação à falhas em *hosts*, recursos e elementos de rede;

- Persistência das informações da coordenação nos *hosts* não obstante a natureza efêmera dos sinais de coordenação, e;
- Interoperabilidade entre plataformas heterogêneas.

Em trabalho colaborativo, não faz sentido separar a capacidade de distribuição do sistema, da necessidade do usuário relativo a estas interações. Assim, considerando os requisitos fundamentais, dentre as decisões de projeto [02] da arquitetura de coordenação devem estar:

1. Distribuição das informações de estado armazenadas para permitir a escalabilidade, melhor desempenho a respeito de concessão de palavra via conexões multipontos, e resiliência. Esta permite sessões tolerantes à falhas para o caso de sítios ou enlaces falharem e seccionarem a rede. O modelo com um servidor de palavra centralizado poderia ser um gargalo para o desempenho e configurar um ponto único de falha;
2. Assimetria das interações para permitir que cada sítio proceda independentemente com seu trabalho local e entre na sessão com seu conjunto de recursos que lhe são peculiares, quando desejar. Os recursos podem, por isso, ser heterogêneos, e comutar suavemente entre trabalho colaborativo síncrono e assíncrono durante interações em uma sessão ao vivo;
3. Hierarquia no gerenciamento de sessão reflete a dinâmica dos encontros presenciais. Ela permite a reconfiguração da sessão criando subgrupos sem a necessidade de se separar ou sair da sessão principal.

4. Adaptação com relação a recursos, *firmeza (tighness)* no controle do espaço compartilhado, e atendimento das requisições, baseado num QoS desejado x possível, permitem que as aplicações se ajustem às expectativas do usuário e a configuração do ACM. A Adaptação também considera os requisitos de planejamento de vários recursos, por exemplo, *deadlines* do tipo *hard* ou *soft* para mídias dependentes de requisitos temporais.

5.6 Políticas e Mecanismos para Controle de Palavra

O controle de palavra em ACM deverá desempenhar as seguintes tarefas: exclusão mútua para objetos compartilhados durante acessos concorrentes; concessão ou negação da palavra para as requisições de acordo com a política em questão; rastreamento e manutenção das informações sobre os recursos compartilhados em todos os sítios conectados; revezamento da palavra entre os diferentes sítios; gerenciamento do acesso temporal a dados; autorização ou proibição do uso de acordo com as informações do controle de sessão; e difusão dos estados do controle de palavra para os participantes.

Todas estas tarefas do controle de palavra são viabilizadas, do ponto de vista do sistema, através de mecanismos específicos ditos de controle de palavra, e, do ponto de vista do usuário, através de políticas de concessão e de uso da palavra. Os mecanismos de controle de palavra é que controlam o fluxo de dados e sincronização de eventos através de todos os sítios enquanto as políticas, implementadas por meio destes mecanismos, determinam como as palavras podem ser requisitadas e garantidas, permitindo reordenação, preempção, e mudança dos serviços

prescritos [02]. Deve-se notar que alguns mecanismos são compatíveis apenas com determinadas políticas.

São exemplos de mecanismos de controle de palavra:

- A negociação;
- A passagem de bastão em seqüência predefinida;
- A passagem de bastão a partir de requisição ao detentor atual;
- A sincronização com mecanismo de temporização global;
- O bloqueio que segue seqüência FCFS (*First Come First Served*);
- A percepção de atividade remota;
- A reserva, e;
- A detecção de dependências.

As políticas de controle podem ser classificadas [02] segundo diferentes critérios com priorização de aspectos específicos. A seguir são apresentadas algumas classificações das políticas de controle para facilitar o entendimento do comportamento das políticas e mecanismos utilizados para viabilizá-las.

5.6.1 Política Explícita x Implícita

As políticas podem prever a execução das primitivas de controle para requisição e concessão de palavra de forma explícita ou implícita.

Numa requisição explícita, por exemplo, o participante deve, explicitamente, demonstrar interesse em obter a palavra associada ao recurso desejado.

Já uma obtenção implícita pode ser executada, por exemplo, no contexto de uma sessão orientada à agenda, onde, num determinado horário ou no início de determinada atividade a palavra é garantida implicitamente a determinado participante.

5.6.2 Política Autônoma x Não Autônoma

É possível classificar as políticas de controle segundo seu grau de autonomia em executar o controle sobre os recursos compartilhado sem a intervenção externa de, por exemplo, um moderador.

Já o controle não autônomo faz uso da figura de um moderador ou árbitro que pode decidir sobre questões relacionadas ao controle de palavra, podendo alterar política, alterar ordem de concessão, cancelar, revogar, garantir ou negar a palavra aos usuários.

O controle autônomo também pode realizar todas estas atividades, entretanto, o faz sem a intervenção externa de um moderador ou outro participante.

5.6.3 Política Preemptiva x Não Preemptiva

As políticas preemptivas permitem a apropriação antecipada de um recurso, ou mesmo da palavra, em relação a um planejamento como, por exemplo, a revogação da palavra concedida a um usuário. As políticas não preemptivas não permitem esta possibilidade.

5.6.4 Política Dependente de Tarefa x Não Dependente de Tarefa

Nas políticas dependentes de tarefa, ao contrário da segunda, a concessão, ou liberação da palavra pode estar condicionada ao início ou à conclusão de tarefas, permitindo o planejamento e definição de agenda para a concessão e liberação da palavra.

5.6.5 Política Mutuamente Exclusiva x Mutuamente Seletiva

Uma política exclusiva garante acesso exclusivo ao recurso evitando, com isso, o conflito, enquanto as políticas seletivas realizam um controle relaxado, concedendo palavras não exclusivas aos participantes.

5.6.6 Exemplos de Políticas de Controle de Palavra

São exemplos de políticas de controle de palavra [02]:

- Chair-guidance, onde um usuário eleito é o árbitro sobre o uso de palavras específicas;
- Agenda-orientation, onde a concessão da palavra obedece a uma determinada ordem, definida antes ou durante a sessão;
- Time-orientation, onde a requisição e uso das palavras têm tempos limites definidos por eventos ou condições de sistema;
- Predefined ordering, onde as palavras são oferecidas ou requisitadas segundo uma seqüência ou um caminho numa

determinada topologia de sítios. Um exemplo é o *round-robin*, relativo à passagem de bastão;

- *Ad hoc reordering*, que coloca as requisições de cada recurso numa fila e os serve de acordo com a oportunidade, prioridade e critério de QoS. Exemplos são os serviços orientados por demanda, *least-recently-served*, *weighted-fair-queueing*, *early-deadline-first*, *etc.*;
- *Election*, que permite que os usuários votem no próximo detentor da palavra;
- *Lottery scheduling*, que usa cartões de loteria num esquema probabilístico para a concessão de palavra.

Outras políticas podem ser criadas ou derivadas das apresentadas acima. Outrossim, dentre estas políticas, algumas podem ser mais adequadas a recursos, e sessões de propósito específico, devendo também empregar mecanismos específicos. Assim, o controle de palavra deve implementar políticas adequadas, levando em conta a natureza e peculiaridades de cada cenário e das interações entre os participantes a fim de alinhar as expectativas e necessidades dos usuários com as características e conseqüente eficácia do ACM.

5.7 Conclusões do Capítulo

A coordenação em ACMs se propõe a resolver problemas de alocação de recursos em termos de sua disponibilidade e planejamento de utilização para permitir o máximo de sinergia entre os participantes,

podendo contemplar controle de palavra, sincronização entre aplicações e segurança.

Neste sentido, o controle de palavra é uma tecnologia de coordenação para lidar com conflitos em ambientes colaborativos.

Palavras, por sua vez, são permissões temporárias para acessos e manipulação de algum recurso compartilhado em um ACM.

Do ponto de vista do usuário, o controle de palavra é realizado através de políticas, que indicam como as palavras podem ser requisitadas e garantidas, permitindo reordenação, preempção, e a mudança dos serviços prescritos. Estas políticas são implementadas, do ponto de vista do sistema, por mecanismos de controle de palavra adequados à política selecionada, controlando o fluxo de dados e sincronizando eventos através de todos os sítios de uma sessão.

Por outro lado, os experimentos realizados com as ferramentas no contexto deste trabalho mostraram que a colaboração fica prejudicada ou é até mesmo inviabilizada à medida que os conflitos no acesso a recursos compartilhados não são evitados ou resolvidos.

Contudo, a grande maioria das ferramentas gratuitas disponíveis e utilizadas nas avaliações não suporta nenhuma política de controle de palavra.

As ferramentas e cenários avaliados também mostraram que o controle de palavra, quando existir, deve contemplar os diversos estilos de interação humana nos mais diversos cenários onde um ACM se faz necessário.

Conclui-se, entretanto, que, para atender a grande variedade de cenários que demandam ACMs com suporte a vários estilos de

comunicação e cooperação e peculiaridades que lhes são inerentes, um “sistema único para conferências” não é adequado.

Neste contexto, propõe-se a decomposição do problema da comunicação homem-homem através de sistemas de comunicação eletrônicos, inclusive em ACM, segundo as diretrizes apresentadas em [39] e se concentrar na diretriz “Coordenação”, independente das demais.

Para abordar a diretriz “Coordenação” adequadamente utilizar-se-á o modelo ACM e de Controle de Palavra, bem como taxonomia apresentada neste capítulo. Os mesmos poderão ser utilizados para validação de protótipos de ACMs e seus componentes.

Capítulo 6

PROTÓTIPO DE CONTROLE DE PALAVRA PARA AMBIENTES COLABORATIVOS MULTIMÍDIA

O controle de palavra tem se mostrado fundamental para a eficácia das ferramentas para *groupware* síncrono. Por outro lado, conforme identificado nos testes de ferramentas e aplicações colaborativas, o controle de palavra é bastante dependente da natureza da sessão: por exemplo, numa reunião entre profissionais de classe, com objetivo de

discutir um problema de interesse comum e com necessidade de interações freqüentes demanda uma política de controle bastante diferente da que se faz necessária para uma aula à distância.

Assim, o que se pretende, é propor uma infra-estrutura básica para fornecer o serviço de controle de palavra em ferramentas para *groupware* síncrono, considerando o modelo LWS, descrito anteriormente, para conferências e trabalho colaborativo. Sobre esta infra-estrutura básica demonstrar-se-á a comunicação entre agentes de coordenação para viabilizar a política de controle de palavra adequada a cada cenário. Os agentes de coordenação são os sistemas de software presentes em cada um dos sítios participantes de uma conferência que implementam todas as funcionalidades relacionadas à coordenação de um ACM.

Neste sentido, o foco deste capítulo é o controle de palavra: políticas, mecanismos e infra-estrutura de comunicação. Ao final, pretende-se disponibilizar um protótipo que permita demonstrar a comunicação entre os vários agentes de coordenação dentro de uma sessão de *groupware* síncrono em diferentes cenários e, a partir daí, demonstrar diferentes políticas e mecanismos de controle de palavra.

Embora exista uma interdependência entre o gerenciamento de sessão e o controle de palavra num ACM, neste trabalho não serão abordados aspectos relacionados ao gerenciamento de sessão: considerar-se-á que os participantes já têm acesso às sessões, com seus papéis definidos, as fases das sessões definidas e identificadas, bem como às outras definições necessárias para que os mesmos estejam aptos a acessar os recursos compartilhados disponíveis na sessão conforme a política selecionada.

Mesmo sendo desejável a adoção de um protocolo de controle de palavra padronizado e que já seja utilizado em outros sistemas, isto não

foi feito neste trabalho: limitou-se a definir um conjunto básico de primitivas de controle e informações a serem trocadas entre os agentes presentes em todos os sítios integrantes de uma sessão. A seleção das primitivas e informações trocadas, bem como o conjunto de regras que regem o serviço de controle de palavra foi definido com vistas a três cenários bastante característicos e comuns: reunião de pequeno grupo de pessoas, seminário para grande número de participantes e sala de aula virtual que permita atividades em grupos de alunos.

6.1 Modelando um Ambiente Colaborativo Multimídia

Embora os ACM vêm se apresentando bastante polimorfos, existem algumas características de ferramentas e aplicações identificadas durante este trabalho que irão garantir facilidades desejadas em tais ambientes. Entende-se que estas características devam ser comuns a todos os ACM e consideradas durante a fase de projeto destes ambientes. Estas características foram classificadas em gerais e específicas. As gerais, não menos importantes que as específicas, dizem respeito ao que é comum à maioria dos sistemas de software. Já as específicas são as que se referem às ferramentas colaborativas e para *groupware*, sobretudo o síncrono, que está no contexto deste trabalho.

6.1.1 Características Gerais

Foram consideradas características gerais importantes para um ACM que segue o modelo LWS: a facilidade de evolução, integração com ferramentas existentes, modularidade, interoperabilidade, portabilidade e uso racional de recursos, sobretudo os de comunicação.

A facilidade de evolução de um ACM está intimamente ligada a características como a integração com ferramentas já existentes e modularidade, que permite o desenvolvimento de novos módulos para serem integrados ao sistema atual, tornando possível adaptação a uma variedade maior de requisitos específicos aos diferentes cenários.

A interoperabilidade das ferramentas que compõem determinado ACM pode ser alcançada através da adoção de padrões amplamente aceitos e utilizados no desenvolvimento destas ferramentas. Este trabalho não se preocupou com este aspecto, por considerar que esta característica o ACM herdará das ferramentas e aplicações que compõem o ACM, e que não estão no contexto do trabalho, pois, durante os testes destas observou-se grande quantidade de ferramentas e aplicações de diferentes fabricantes que interoperam através de padrões abertos.

A portabilidade do sistema pode ser obtida com o uso de uma plataforma de desenvolvimento portátil. Observando o grande número de ferramentas e aplicações que vêm sendo desenvolvidas sobre a plataforma Java, conclui-se que esta pode ser auxiliar no desenvolvimento de ACMs portáteis. Utilizando a plataforma de desenvolvimento Java, o ACM herda, ainda, outras características desejáveis, como a modularidade e reutilização de componentes, contribuindo também para a flexibilidade e facilidade de evolução de tais ambientes.

Características gerais não apresentadas até aqui, como facilidade de instalação, configuração e uso, interface amigável e integração com Web, fogem do escopo deste trabalho.

6.1.2 Características Específicas

Dentre as características específicas desejáveis para os ACMs destaca-se o suporte a conjunto de funcionalidades adequado a cada cenário e suporte a políticas de controle de palavra adequadas a cada cenário.

Devido ao fato dos ACMs atenderem a demandas bastante específicas nas mais diversas áreas, o suporte a conjunto de funcionalidades específicas a cada um destes cenários esta característica está ligada à modularidade do sistema. Esta característica pode ser alcançada evitando-se aplicações monolíticas, e adotando-se um modelo composto por um conjunto de aplicações e ferramentas independentes e modulares que podem funcionar em conjunto, de forma coordenada, em cada sítio, como pode ser visto na Figura 11. Assim, consegue-se adaptar o ACM facilmente a cada cenário, incluindo, retirando ou alterando cada um dos módulos, conforme a necessidade. Por exemplo, para uma sessão entre três participantes que queiram discutir um problema acerca de um projeto, bastariam ferramentas para compartilhamento de áudio, vídeo e uma ferramenta para compartilhamento de ambiente de trabalho, podendo-se abrir mão das demais ferramentas possíveis de serem usadas, mas não necessárias para este contexto. O controle de palavra, para este caso, por exemplo, pode ser realizado por meio de regras sociais conhecidas pelos participantes. Já para uma aula virtual para uma turma de trinta alunos, estes mesmos módulos e política de coordenação e controle de palavra podem não ser os mais adequados.

Outra característica muito importante e considerada neste trabalho é a necessidade do ACM ter boa escalabilidade. É fundamental permitir que os ambientes contemplem desde sessões pequenas até sessões com centenas de participantes. A escalabilidade é uma característica presente

no modelo LWS e é obtido, principalmente devido à infra-estrutura de comunicação ser baseada no endereçamento *multicast*.

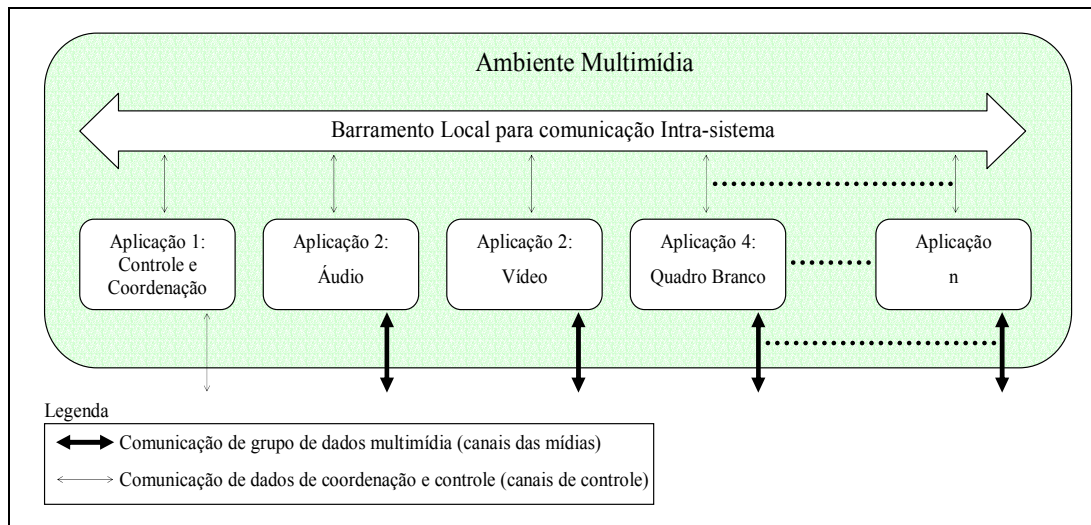


Figura 11: Facilidade de integração, adaptação e evolução de um ACM

O uso do endereçamento *multicast*, que viabiliza a comunicação entre os vários sítios participantes de uma sessão, também vem ao encontro de outra necessidade para tais ambientes: o uso racional dos recursos de comunicação. Com o uso do endereçamento *multicast* consegue-se otimizar a utilização da largura de banda, importante no contexto deste trabalho, já que várias aplicações para trabalho colaborativo síncrono manipulam grandes quantidades de dados, por exemplo, áudio e vídeo.

O suporte a políticas de controle de palavra adequadas a cada cenário está no cerne deste trabalho e também está intimamente ligado a outras características aqui consideradas: ele se torna importante ao passo que a alocação de recursos compartilhados numa sessão de trabalho colaborativo síncrono gera competição e conflitos, interferindo no desempenho da sessão. Outrossim, as políticas para prover o acesso a

determinado recurso mostram-se bastante particulares à natureza de cada sessão.

Características específicas não detalhadas até aqui, como as relacionadas à facilidade de gerenciamento de usuários numa sessão, alocação de endereços para a sessão e agenda de sessões, padronização para comunicação entre os participantes, mecanismos eficientes para prover qualidade de serviço e suporte a múltiplos padrões de codificação e compressão de mídias, embora também sejam importantes, não são abordadas neste trabalho.

6.1.3 O Ambiente Multimídia de Cada Sítio

Considera-se que cada sítio participa de uma sessão de trabalho colaborativo por meio de um ambiente multimídia, conforme apresentado na Figura 11.

Nesta figura, as aplicações, com funcionalidades e requisitos de sistema e comunicação específicos, podem funcionar independente ou em conjunto com outras, integradas, compondo o ambiente multimídia de um determinado sítio.

As ferramentas utilizadas para suportarem transmissão e recepção de mídias ou compartilhamento de recursos específicos são chamadas de aplicações de mídias, e utilizam canais de dados para a comunicação com outros participantes.

Todas estas ferramentas, quando operam em conjunto, compondo o ambiente multimídia do sítio, têm seu comportamento controlado e coordenado pelo agente de coordenação que, baseadas nas políticas de

gerenciamento e coordenação da sessão se comunicam através de mensagens com cada uma das aplicações.

A comunicação entre o agente de coordenação – aplicação 1 (que implementa, inclusive o controle de palavra e, denominados, a partir daqui, de *agente de controle*) e as demais aplicações de mídia, chamada de “comunicação intra-sistema”, não será implementada no protótipo, mas pode facilmente ser viabilizada através de mecanismos como o Mbus [44], JMS [45], ou ainda através de bibliotecas reutilizáveis Java com protocolo simplificado, que faz uso de um grupo *multicast* local, ou seja, com TTL dos pacotes IP definido em zero em endereço e porta conhecida [37].

6.1.4 O Ambiente Colaborativo Multimídia Proposto

A partir dos ambientes multimídia presentes em cada um dos sítios, compõe-se o ambiente colaborativo multimídia, conforme Figura 12. Esta figura apresenta as aplicações de mídia que compõem o ACM se comunicando através dos canais de mídia. Estas comunicações são controladas e coordenadas pelos agentes de controle, que se comunicam com os seus pares em cada um dos sítios através dos canais de controle.

Esta comunicação entre as aplicações de diferentes sítios, chamada de comunicação intersistema é feita através de uma rede UDP/IP com suporte *multicast*.

Como se pretende que o ACM possa atender desde pequenos até grandes grupos, o suporte de comunicação precisa ter boa escalabilidade. Isto é possível fazendo com que cada participante seja responsável por sua recepção. Não é desejável que o controle da recepção seja feito pelo transmissor, pois este fato pode criar gargalos na sessão [37], contrariando o modelo LWS.

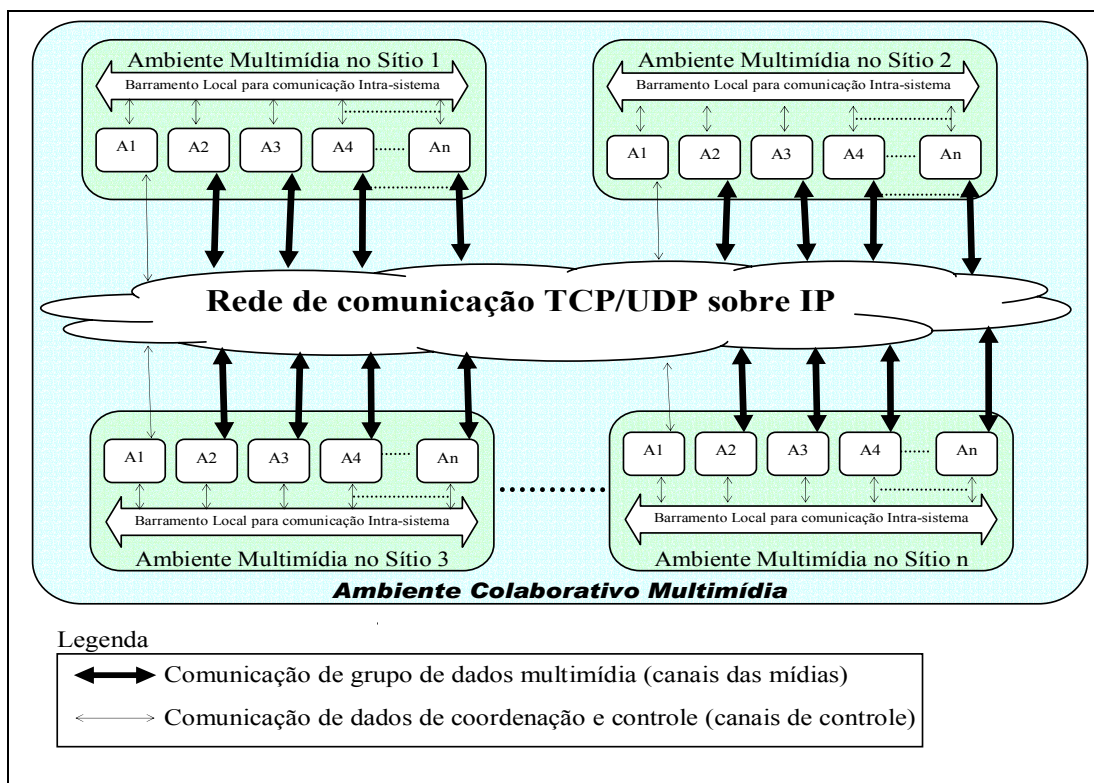


Figura 12: Ambiente Colaborativo Multimídia

Chama-se atenção ainda para o fato de o ambiente apresentado na Figura 12 ter diferentes requisitos de comunicação para cada uma das aplicações que compõem o ambiente multimídia em cada sítio. Um exemplo é a necessidade de confiabilidade na infra-estrutura de comunicação para os agentes de controle.

6.2 Controle de Palavra para Ambiente Colaborativo Multimídia

Quando se inicia um projeto para controlar a palavra em trabalho colaborativo síncrono, deve se ter em mente claramente a natureza do trabalho ou das interações que serão suportadas.

Neste capítulo consideraremos três cenários para demonstrar a comunicação das diretivas de controle entre os agentes de controle:

Cenário 1 – Reunião de um pequeno grupo de pessoas, cerca de cinco participantes geograficamente distribuídos, que discutem os resultados de um projeto que está sendo desenvolvido de forma colaborativa. Os recursos compartilhados são canais de áudio, vídeo, quadro branco e um tele-apontador sobre o ambiente gráfico local;

Cenário 2 – Seminário para um grande número de pessoas com diferentes papéis atribuídos (palestrante, professor, aluno, empresário, visitante). A interação com o palestrante é permitida utilizando áudio e vídeo apenas para professor, empresário ou outro palestrante. Alunos poderão interagir via *chat*, e visitantes serão participantes passivos, não sendo permitida a interação com os demais. Considera-se que o gerenciamento dos usuários é contemplado por uma ferramenta de controle de sessão, fora do escopo deste trabalho;

Cenário 3 – Sala de aula virtual onde com dois papéis definidos (aluno e professor). Inicialmente o professor ministra uma aula para todos e, a partir daí são definidos pequenos grupos que passam a trabalhar em atividades paralelas. O professor participa de todos os grupos (subsessões) avaliando e orientando os alunos. O professor mantém privilégios especiais para intervir quando necessário em cada um dos grupos.

Diante destes cenários e baseando-se nas características descritas acima, serão favorecidas as seguintes decisões de projeto: simplicidade de implementação e manutenção, escalabilidade, extensibilidade para novos recursos e modelos de sessões, confiabilidade em relação à falhas em *hosts*, recursos e elementos de rede [03].

6.3 Protótipo para Controle de Palavra

Com a apresentação de um protótipo com funcionalidades básicas para provimento de serviço de controle de palavra espera-se validar o modelo de ACM apresentado na Figura 11 e Figura 12, bem como disponibilizar a base para a implementação futura de um protocolo completo sobre este protótipo, e que este possa ser integrado e reutilizado para novas implementações de políticas e mecanismos de controle de palavra.

O protótipo permitirá visualizar a troca de diretivas de controle entre os agentes de controle presentes em cada um dos sítios participantes da sessão. A comunicação entre o agente de controle e as demais aplicações de mídias e de gerenciamento de sessão não é implementada neste trabalho.

6.3.1 Modelos de Controle de Palavra

Na implementação do protótipo serão considerados modelos de controle centralizado e distribuído.

O controle de palavra centralizado apresenta como grandes vantagens a sua simplicidade de implementação e de rastreamento de

palavras. A desvantagem está na possibilidade de sobrecarga e falha no sítio que realiza o controle.

O modelo distribuído, no qual os estados das palavras são calculados através do rastreamento de eventos locais e remotos a cada sítio que ocorrem numa sessão, tem a vantagem de não oferecer ponto único de falha ou gargalo no sítio que realiza o controle. Por outro lado, sua implementação é mais complexa e apresenta replicação de processamento nos sítios participantes da sessão.

Pretende-se que o protótipo suporte ambos os modelos de controle (centralizado e distribuído): conforme política selecionada, as informações deverão, eventualmente, serem obtidas e fornecidas para o nó central de controle. Contudo, outra política de controle pode permitir que cada agente de controle colete informações e analise eventos remotos e, a partir daí, decida sobre ações futuras para evitar ou abrandar conflitos e garantir a colaboração. Esta última política, certamente, apenas será possível com arquitetura de coordenação distribuída.

Embora o protótipo atualmente não faça o tratamento das informações trocadas entre os diversos sítios, implementações de controle centralizado e distribuído podem ser realizadas futuramente sobre este protótipo aqui apresentado.

6.3.2 Políticas para Controle de Palavra

A política de controle de palavra define como as palavras podem ser requisitadas e garantidas [02]. Para o protótipo, buscando contemplar os três cenários anteriormente descritos, serão simuladas as seguintes

políticas de controle: orientada a moderador, orientada a agenda, orientada a tempo e reordenação “*ad hoc*”¹.

Entretanto, analisando cada um dos cenários, observa-se que muitas vezes pode ser necessário, em consideração às atividades a serem desenvolvidas a cada momento, utilizar mais de uma política num determinado cenário. Assim, por exemplo, o cenário três, pode necessitar política de controle orientada, inicialmente, por agenda, mas em determinada fase pode vir a ser necessário um moderador que distribua as palavras aos que requisitarem, por um tempo determinado. Assim, esta característica também deve ser suportada pelo ACM.

6.3.3 Mecanismos para Controle de Palavra

Os mecanismos de controle de palavra regulam os fluxos de controle de baixo nível e sincronizam os eventos dos sítios participantes de uma sessão.

Os mecanismos de controle de palavra podem ser derivados de métodos de exclusão mútua e controle de concorrência. Para viabilizar as políticas previstas para o protótipo podem ser implementados mecanismos de passagem de bastão, solicitação de bastão e percepção de atividade remota.

¹ Do latim, significa «para o efeito», «com esse fim», «para o caso específico». A expressão qualifica um evento ou ação que tenha sido criada para uma finalidade especial.

6.3.4 Suporte de Comunicação

Para permitir a comunicação entre os vários sistemas multimídia e compor o ACM apresentado na Figura 12 deve se ter uma infra-estrutura de comunicação adequada e capaz de lidar com os diversos requisitos impostos pelo conjunto de aplicações e ferramentas.

O grande volume de dados gerado pelas aplicações de mídias contínuas, o requisito temporal e a necessidade de sincronização entre algumas mídias impõem à rede de comunicação requisitos de largura de banda, atraso e *jitter*.

Por outro lado, existem aplicações, como as de controle e coordenação, apresentado na Figura 11, que impõem requisitos especiais a rede, como a necessidade de canais confiáveis entre os sítios participantes de uma sessão. Para estes casos onde são necessários canais de comunicação confiáveis, a dificuldade está em garantir todas as características desejáveis ao ambiente, do ponto de vista da rede, como a boa escalabilidade, que é obtida através do endereçamento *multicast* e que não possui a confiabilidade implementada de maneira nativa. O suporte a endereçamento *multicast*, parte integrante do modelo LWS é um dos requisitos da rede para que um ACM possa lidar de maneira eficaz com o grande volume de dados, sem comprometer o desempenho, qualidade e escalabilidade do ambiente.

Para garantir a confiabilidade, com detecção e recuperação de erros e ordenação de mensagens, sem comprometer as características de escalabilidade e desempenho do modelo de comunicação, no protótipo será utilizado uma biblioteca reutilizável Java do LRMP (*Light-weight Reliable Multicast Protocol*) [21], que teve suas características e funcionamento detalhado no capítulo 3 deste trabalho.

O LRMP tem o objetivo de oferecer um serviço de envio de dados seguro e com capacidade de ordenação para grupos de comunicação. O LRMP também tem suporte a controle de congestionamento e fluxo, permitindo que a largura de banda da rede possa ser compartilhada com outros fluxos de dados. Ele foi desenvolvido para ser utilizado em ambientes de rede heterogêneos, e suporta múltiplos transmissores de dados, por isso mostra-se adequado aos cenários apresentados neste capítulo.

O LRMP foi implementado em plataforma Java (*Java Development Kit* - JDK 1.1 e 1.2), livre para uso comercial e acadêmico, sendo usado num número expressivo de projetos e produtos. O LRMP ainda está em desenvolvimento sendo, atualmente, usado e distribuído com o pacote de bibliotecas JSST (*Java Shared Data Toolkit*) [22] e, com isso, tendo importantes funcionalidades e melhorias sendo integradas.

6.3.5 Regras Básicas de Comunicação

Embora não seja utilizado nenhum protocolo existente ou padrão para comunicação entre os agentes de coordenação e controle localizados em cada sítio do ACM da Figura 12, um conjunto de regras básicas e conhecidas é necessário para poder oferecer o controle de palavra pretendido para os cenários apresentados.

Aqui será definido o formato básico para a troca de mensagem que atenda às necessidades para os mecanismos e políticas pretendidas para cada cenário.

A comunicação entre os agentes de controle dar-se-á através de troca de mensagens ASCII através de canais de comunicação de grupo (*multicast*) confiável. As mensagens ASCII serão formadas por uma lista de

variáveis associadas aos seus respectivos valores. A lista de variáveis sugeridas e seus possíveis valores, baseada em [02], [46] e [47], está em consonância com o modelo e taxonomia apresentados no item 5.4.

Considerando a tupla apresentada no item 5.4 para caracterizar e definir um ambiente colaborativo, os pares *variável:valor* a serem trocadas entre os agentes de controle são agrupados segundo quatro categorias, às quais estes pares se relacionam:

- Sessão;
- Recurso;
- Usuário;
- Palavra.

Considerando uma palavra $f \in \mathbf{F}$, que é definida como sendo uma tupla $f = \langle \mathbf{F}_{id}, \mathbf{R}_{id}, \mathbf{U}_{id}, \mathbf{T}_i, \mathbf{T}_d, \mathbf{A}_f \rangle$, na Tabela 8 são apresentados os pares “*variável:valor*” associados à categoria *Palavra*. A coluna “*Parâmetro associado*” faz um mapeamento dos pares “*variável:valor*” (colunas “*Variável*” e “*valores possíveis*”, respectivamente) com o modelo apresentado no item 5.4

Parâmetro associado	Variável	Valores possíveis ²	Descrição da variável e seus respectivos valores
A_f	PrimitivaDeControle	<i>DefinePolitica</i>	Define política de controle de palavra a ser utilizada numa sessão
		<i>AlteraPolitica</i>	Altera a política de controle a ser utilizada numa sessão
		<i>CriaPalavra</i>	Cria uma palavra e a associa a um determinador recursos compartilhado, atribuindo-lhes as características básicas
		<i>RemovePalavra</i>	Remove uma determinada palavra de uma sessão
		<i>RequisitaPalavra</i>	Requisita a palavra sobre determinado recurso

² O texto em itálico representa exatamente os valores passados pelas mensagens e os valores entre os sinais < > indicam exemplos de possíveis valores, separados por vírgula, que podem ser atribuídos às variáveis.

		<i>LiberaPalavra</i>	Libera a palavra para o controlador, o detentor anterior, ou simplesmente informa que a palavra está Livre, dependendo da política adotada
		<i>RevogaPalavra</i>	Revoga a concessão da palavra para determinado usuário sobre determinado recurso
		<i>ConcedePalavra</i>	Concede palavra sobre determinado recurso para um usuário
		<i>RequisitaControle</i>	Requisita o controle da palavra numa determinada sessão
		<i>LiberaControle</i>	Libera o Controle de determinada sessão, repassando-o explicitamente para alguém
		<i>ConcedeControle</i>	Concede o controle de determinada sessão
		<i>AlteraPalavra</i>	Altera o estado da palavra (permitido apenas ao controlador de palavras da sessão)
		<i>TestaServico</i>	São mensagens periódicas enviadas pelos sites participantes para informarem uns aos outros de suas respectivas existências na sessão.
		<i>TestaPalavra</i>	Mensagens enviadas pelo controlador quando ele percebe períodos grandes de inatividade de algum detentor de alguma das palavras, permitindo prover mecanismo que evite que a saída de um nó que detenha uma palavra ou que esteja na fila de requisições interfira no fluxo das atividades.
A_f	PoliticaDeControle	<i>OrientadaModerador</i>	Define a política atual como sendo orientada por moderador
		<i>OrientadaAgenda</i>	Define a política atual como sendo orientada por agenda com seqüência e tempos definidos para a concessão da palavra
		<i>OrientadaTempo</i>	Define a política atual como tendo tempos definidos para cada concessão de palavra
		<i>ReordenadaAdHoc</i>	Define a política atual como Ad Hoc, serializando as requisições e reordenando-as conforme prioridade ou outros critérios que podem ser definidos.
A_f	EstadoAtualDaPalavra	<i>LIVRE</i>	Informa que uma palavra está livre para ser requisitada
		<i>INATIVA</i>	Informa que uma palavra está inativa
		<i>REQUISITADA</i>	Informa que uma palavra está sendo requisitada por um ou mais usuários
		<i>USADALOCALMENTE</i>	Informa que uma palavra está sendo usada localmente
		<i>USADAREMOTAMENTE</i>	Informa que uma palavra está sendo usada remotamente
		<i>BLOQUEADA</i>	Informa que a palavra está bloqueada
A_f	NovoEstadoDaPalavra	<i>LIVRE</i>	Informa que uma palavra estará livre para ser requisitada a partir da presente primitiva
		<i>INATIVA</i>	Informa que uma palavra estará inativa a partir da presente primitiva
		<i>REQUISITADA</i>	Informa que uma palavra estará sendo requisitada por um ou mais usuários a partir da presente primitiva
		<i>USADALOCALMENTE</i>	Informa que uma palavra estará sendo usada localmente a partir da presente primitiva
		<i>USADAREMOTAMENTE</i>	Informa que uma palavra estará sendo usada remotamente a partir da presente primitiva
		<i>BLOQUEADA</i>	Informa que a palavra estará bloqueada a

			partir da presente primitiva
F_{id}	IdDaPalavra	<Palavra Global>	Define um identificador único da palavra na sessão
R_{id}	RecursoAssociado	<VIDEO> , <VIDEO, AUDIO> , <TODOS>	Define os recursos associados a uma palavra
A_f	TipoDePalavra	<1> , <2> , <INFINITO>	Define quantos usuários podem, simultaneamente, fazer uso da palavra.
U_{id}	NovoDetentor	<vc@200.135.48.104>	Define o sítio que irá deter a palavra a partir desta primitiva
T_i	TempoDeObtencao	<08.32.20>	Indica o tempo em que a palavra foi obtida
T_i	TempoConcedido	<00.01.30> , INFINITO	Indica por quanto tempo a palavra é concedida
A_f	PapeisPermitidos	<professor;aluno>	Indica são os papeis permitidos para requisitar determinada palavra. Estes papéis são obtidos do gerenciamento de sessão.
A_f	SequenciaAgenda	<200.135.48.104, 00.02.30; 200.135.48.101, 00.02.30; 200.135.48.105, INFINITO>	Define a seqüência de usuários e tempos de concessão da palavra para política orientada por agenda
A_f	PapelnoControle	CONROLADOR	Papel que o usuário tem atribuído no contexto do controle de palavra: Controlador da palavra
		DETENTOR	Papel que o usuário tem atribuído no contexto do controle de palavra: Detentor da palavra
		PARTICIPANTE	Papel que o usuário tem atribuído no contexto do controle de palavra: apenas participante, sem ser detentor ou controlador da palavra.
A_f	Mensagem	<" Olá, tenho uma dúvida..">	Mensagem do usuário transmissor que acompanha a primitiva e indica motivo de, por exemplo, uma requisição de palavra

Tabela 8: Variáveis e valores utilizados na troca de mensagens ASCII entre os agentes de controle associados à categoria *Palavra*

Considerando uma sessão $s \in \mathbf{S}$, que compõe um ambiente colaborativo \mathbf{E} , como sendo uma tupla $\mathbf{s} = \langle \mathbf{S}_{id}, \mathbf{T}_i, \mathbf{T}_e, \mathbf{A}_s, \mathbf{L} \rangle$, na Tabela 9 são apresentados os pares “*variável:valor*” associados à categoria *Sessão*. A coluna “*Parâmetro associado*” faz um mapeamento dos pares “*variável:valor*” (colunas “*Variável*” e “*valores possíveis*”, respectivamente) com o modelo apresentado no item 5.4.

Parâmetro associado	Variável	Valores possíveis ³	Descrição da variável e seus respectivos valores
S_{id}	IdDaSessao	<"Aula Virtual">	Define um identificador único para uma sessão dentro de uma conferência
A_s, L	IdDaSubSessao	<AtividadeGrupo1>	Define um identificador único para uma sub-sessão dentro de uma sessão

³ O texto em itálico representa exatamente os valores passados pelas mensagens e os valores entre os sinais < > indicam exemplos de possíveis valores, separados por vírgula, que podem ser atribuídos às variáveis.

T_i	InicioDaSessao	<08.30.34>	Informa a hora de início de uma sessão
T_e	FimDaSessao	<09.30.00>	Informa a hora de fim de uma sessão
A_s, L	Papelnasessao	<ALUNO>, <PROFESSOR>	Papel que o usuário tem atribuído numa sessão

Tabela 9: Variáveis e valores utilizados na troca de mensagens ASCII entre os agentes de controle associados à categoria *Sessão*

Outras variáveis associadas às categorias *Usuário* e *Recurso* também são necessárias para compor um conjunto básico de informações para suportar as políticas. Entretanto, como o principal objetivo é apresentar o protótipo trocando, principalmente, diretivas de controle de palavra entre os diferentes sítios participantes de uma sessão, estas outras variáveis não são aqui apresentadas. Outrossim, é preciso, para implementar outras políticas e mecanismos mais sofisticados, outras informações associadas às categorias *Palavra* e *Sessão*, bem como a definição de pares “*variável:valor*” adicionais.

O tratamento que cada agente controle dará às informações recebidas e enviadas é que caracterizará os mecanismos de controle. Os mecanismos a serem utilizados para determinada política viabilizarão diferentes modelos de controle de palavra, como centralizado, distribuído ou até mesmo híbrido (mantendo parte das informações de controle distribuída e parte num nó de controle centralizado). Por exemplo, para um mesmo cenário, todas as informações sobre eventos, histórico e estado de recursos, usuários e palavras podem ficar armazenadas no sítio que controla as palavras, simplificando a implementação. Entretanto, dependendo da natureza da sessão e de suas interações, pode ser mais adequado cada participante manter um conjunto básico de informações sobre histórico, eventos e estado dos recursos, usuários e palavras em seu sítio local, permitindo a tomada de decisões baseada nestas informações e viabilizando, até mesmo, o controle de palavra completamente autônomo [02].

As informações carregadas pelas mensagens trocadas entre os agentes de controle do protótipo estão alinhadas com o modelo de sistema e taxonomia apresentado em 5.4, como pode ser observado na Tabela 8 e Tabela 9, que apresentam as informações suportadas pelo protótipo que foi implementado.

6.3.6 Plataforma de desenvolvimento

Para se alcançar as características apresentadas no início deste capítulo e, feitas as considerações de projeto iniciais, apresentadas na seção 6.2, definiu-se a plataforma J2SE (*Java 2 Standard Edition*) para o desenvolvimento do protótipo.

Utilizando esta plataforma espera-se poder disponibilizar uma infraestrutura mínima para o serviço de controle de palavra para os três cenários apresentados e facilitar a continuação dos trabalhos sobre este protótipo, integrá-lo a aplicações e facilitar a evolução e portabilidade do sistema.

6.3.7 A Implementação do Protótipo

Para demonstrar a troca de mensagens de controle de palavra entre os agentes de controle, localizados em cada um dos sítios de uma sessão, foi desenvolvido um pequeno protótipo que faz uso do LRMP implementado sob forma de um pacote de bibliotecas reutilizáveis Java. Em cada sítio são executadas duas linhas de execução: uma para a transmissão das informações e outra para a recepção das informações recebidas no âmbito do grupo.

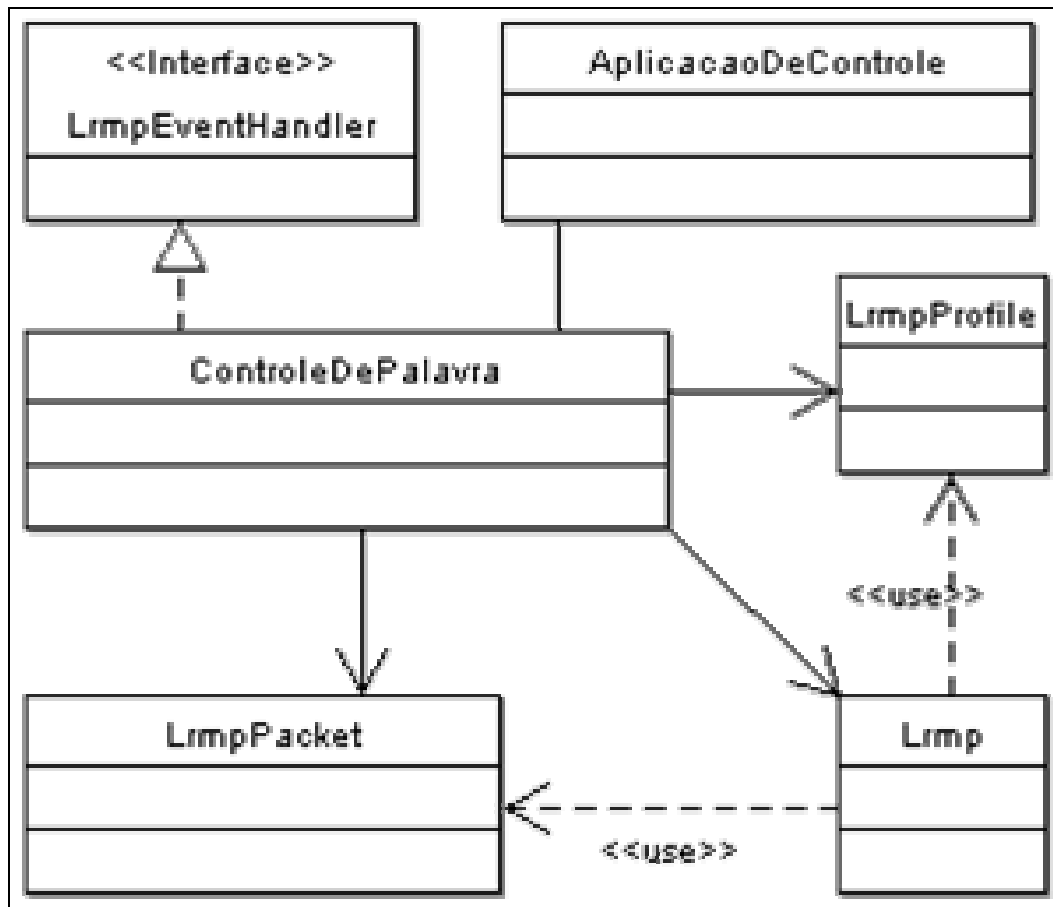


Figura 13: Diagrama de Classes UML do Protótipo

A Figura 13 apresenta um diagrama de classes UML (*Unified Modeling Language*) do: neste diagrama são mostradas apenas as principais classes, ocultando as demais. A seguir são detalhadas as principais classes do protótipo.

6.3.7.1 Classe *ControleDePalavra*

A Figura 14 apresenta o código onde a classe *ControleDePalavra* implementa a interface *LrmpEventHandler*. Ainda pode ser observado na mesma figura os parâmetros *group*, *port* e *ttl* definidos no construtor da classe *ControleDePalavra*, representando, respectivamente, o grupo, a

porta e o TTL (*Time To Live*) a serem utilizados para a troca de mensagens de controle de palavra.

```

...
import java.io.*;
import java.util.*;
import inria.net.lrmp.*;

public class ControleDePalavra implements LrmpEventHandler {
    Lrmp lrmp;
    LrmpEntity sender = null;
    String PrimitivaDeControleRecebida = null;
    OutputStream out = null;

    public ControleDePalavra(String group, int port, int ttl) {
...

```

Figura 14: Classe *ControleDePalavra* que implementa a interface *LrmpEventHandler*

O método *enviaMensagemControleDePalavra* da classe *ControleDePalavra* é responsável pelo envio dos dados. A sua implementação pode ser vista na Figura 15.

É passado para este método uma primitiva de controle que, por sua vez carrega consigo uma série de informações (pares formados por variáveis e valores), conforme convenção estabelecida na Tabela 8 e Tabela 9.

Neste método é criado um objeto *LrmpPacket* que irá carregar os dados através da rede. Basicamente, o que este método faz é criar um objeto *LrmpPacket*, preenchê-lo adequadamente, e enviá-lo através da rede, chamando o método *send* do objeto *lrmp*, criado a partir da classe *Lrmp*, na Figura 18.

Vale a pena chamar atenção para o fato de que a primitiva de controle é identificada a partir do nome de um arquivo que contém as demais informações (pares *variável:valor*). Estes dados não são obtidos de

forma dinâmica ou integrada a outras aplicações como, por exemplo, ferramentas de mídias ou de gerenciamento de grupo, e sim de arquivos estáticos definidos para simular cada uma das etapas das trocas de mensagens de controle entre os sítios de uma sessão através de canais de *multicast* confiáveis.

Com a implementação da Figura 15, o primeiro pacote carrega apenas a primitiva de controle, sendo que a partir do segundo pacote são transferidos os demais dados associados à primitiva.

```

...
public void enviaMensagemControleDePalavra(String PrimitivaDeControle) {
    InputStream in;
    try {
        in = new FileInputStream(PrimitivaDeControle);
    } catch (FileNotFoundException e) {
        imprima("Falhou a abertura da PrimitivaDeControle: " + e);
        System.exit(1);
        return;
    }
    try {
        int headerLen = 1;
        LrmpPacket MensagemDeControle = new LrmpPacket();
        int offset = MensagemDeControle.getOffset();
        byte buffer[] = MensagemDeControle.getDataBuffer();
        buffer[offset] = 0;
        offset += headerLen;
        System.arraycopy(PrimitivaDeControle.getBytes(), 0,
            buffer, offset,
            PrimitivaDeControle.length());
        MensagemDeControle.setDataLength(PrimitivaDeControle.length() + headerLen);
        lrmp.send(MensagemDeControle);
        int pos = 0;
        while (in.available() > 0) {

            MensagemDeControle = new LrmpPacket();
            offset = MensagemDeControle.getOffset();
            buffer = MensagemDeControle.getDataBuffer();

            int maxlen = MensagemDeControle.getMaxDataLength() - headerLen;

            int len = in.read(buffer,
                offset + headerLen,
                maxlen);
            MensagemDeControle.setDataLength(len + headerLen);
            if (in.available() > 0)
                buffer[offset] = 1;
            else
                buffer[offset] = 2;
            lrmp.send(MensagemDeControle);
            pos += len;
        }
        in.close();
    } catch (IOException e) {
        imprima("Erro - Nao foi possivel ler Primitiva de Controle: " + e);
        return;
    } catch (LrmpException e) {
        imprima("Erro - Nao foi possivel enviar pacote LRMP: " + e);
        return;
    }
}
...

```


Figura 15: Implementação do método *enviaMensagemControleDePalavra* da classe *ControleDePalavra*, responsável pelo envio de dados

Sempre que uma seqüência de dados é recebida pelo objeto *Lrmp*, o método *processData*, definido na interface *LrmpEventHandler*, é chamado. A implementação deste método é apresentada na Figura 16. Na mesma figura ainda pode ser observado que após os dados serem recebidos com sucesso, é chamada a função *trataVariaveisDeControle*, para dar o devido tratamento aos dados recebidos. Na implementação atual, esta função simplesmente apresenta os pares *variável:valor* na tela, não atuando sobre aplicações e ferramentas que comporão o ambiente multimídia em cada sítio.

```

...
public void processData(LrmpPacket MensagemDeControle) {
    byte buffer[] = MensagemDeControle.getDataBuffer();
    int offset = MensagemDeControle.getOffset();
    int length = MensagemDeControle.getDataLength();

    switch (buffer[offset]) {

    case 0:
        PrimitivaDeControleRecebida = new String(buffer, offset + 1, length - 1);
        imprima("Recebida Primitiva de Controle: " + PrimitivaDeControleRecebida);
        out = System.out;
        break;

    case 1:
        if (out != null) {
            try {
                Properties variaveisDeControle = new Properties();
                variaveisDeControle.load(new ByteArrayInputStream(buffer, offset + 1, length - 1));
                trataVariaveisDeControle(variaveisDeControle);
            } catch (IOException e) {
                imprima("Erro - Recepcão de Primitiva de Controle: " + e);
            }
            out = null;
        }
        break;

    case 2:
        if (out != null) {
            try {
                Properties variaveisDeControle = new Properties();
                variaveisDeControle.load(new ByteArrayInputStream(buffer, offset + 1, length - 1));
                trataVariaveisDeControle(variaveisDeControle);
            } catch (IOException e) {
                imprima("Erro - Recepcão de Primitiva de Controle: " + e);
            }
            out = null;
        }
        break;

    }
}
...

```

Figura 16: Implementação do método *processData*

6.3.7.2 Classe *LrmpProfile*

A Figura 17 apresenta a criação de um objeto chamado *PerfilControleDePalavra*, a partir da classe *LrmpProfile*, que permite definir o perfil desejado para a transmissão das mensagens. Podem ainda ser observados na mesma figura: a definição de parâmetros como confiabilidade, ordenação das mensagens, taxas de transferência mínima e máxima, e; tamanho da janela (*buffer* de transmissão).

```

...

LrmpProfile PerfilControleDePalavra = new LrmpProfile();

PerfilControleDePalavra.setEventHandler(this);

PerfilControleDePalavra.reliability = LrmpProfile.NoLoss;

PerfilControleDePalavra.ordered = true;

PerfilControleDePalavra.throughput = LrmpProfile.AdaptedThroughput;

PerfilControleDePalavra.minRate = 8;
PerfilControleDePalavra.maxRate = 64;

PerfilControleDePalavra.sendWindowSize = 64;
PerfilControleDePalavra.rcvWindowSize = 64;

...

```

Figura 17: Definição do Perfil para o LRMP

A criação de um objeto *Lrmp* e o início da recepção de dados ocorrem por meio da implementação apresentada na Figura 18. Durante o processo de criação do objeto *Lrmp* são informados, sob forma de argumentos, o grupo, porta, TTL e perfil desejado para o objeto. A partir do momento da execução do método *start*, a aplicação está apta a receber dados conforme argumentos passados para o construtor do objeto *Lrmp*.

```

...

    try {
        lrmp = new Lrmp(group, port, ttl, PerfilControleDePalavra);
    } catch (LrmpException e) {
        imprima("Falha na criacao do objeto LRMP - " + e);
        System.exit(1);
    }

    lrmp.start();

...

```

Figura 18: Criação de um objeto *Lrmp* e o início da recepção de dados

6.3.7.3 O Programa Principal

A Figura 19 apresenta a função do programa principal. As primitivas de controle da instância do programa que transmite os dados a partir de um sítio são passadas ao programa através da linha de comando. Caso o programa não receba nenhum argumento contendo primitivas de controle, o programa criará um novo objeto *ControleDePalavra*, permanecendo pronto para receber dados através deste objeto. Caso o programa receba alguma primitiva de controle como argumento através da linha de comando, o programa executa o método *enviaMensagemControleDePalavra* do objeto *ControleDePalavra* criado.

```

...
public static void main(String args[]) {
    String PrimitivaDeControle = null;
    String group = null;
    int port = -1;
    int ttl = 1;

    ...

    ControleDePalavra obj = new ControleDePalavra(group, port, ttl);
    LrmpPacket MensagemDeControle = new LrmpPacket();
    if (PrimitivaDeControle != null)
        obj.enviaMensagemControleDePalavra(PrimitivaDeControle);
    else {
        imprima("pronto para receber Primitivas de Controle!");
    }
}
...

```

Figura 19: Função *main* do protótipo

6.3.7.4 As Mensagens Trocadas

As informações carregadas pelas mensagens trocadas pela infraestrutura de serviço para controle de palavra são em formato ASCII e

organizado na forma de uma lista de duplas: cada “*variável*” tem associada a ela um “*valor*”, separados por um caractere delimitador definido “:”. Cada dupla é separada da anterior e seguinte por um caractere separador “;”.

Para os casos em que os valores contém mais de uma informação e onde pode ser útil o tratamento individual destas informações, utiliza-se um caractere organizador “,”.

As variáveis e valores utilizados são os apresentados na Tabela 8 e Tabela 9. Para facilitar o tratamento das informações, as duplas (*variável:valor*) são agrupadas de acordo com a natureza da informação que carregam (categorias do item 6.3.5) e em consonância com o modelo e taxonomia do item 5.4.

O exemplo de mensagem da Figura 20 esclarece esta organização apresentando três categorias de informações considerados para o caso de transmissão da primitiva de controle “*RequisitaPalavra*”.

```
USUARIO:USUARIO;
IdDoUsuario:maria@200.135.48.103;
Papelnasessao:participante;
PapelnosControle:PARTICIPANTE;

SESSAO:SESSAO;
IdDasessao:"Teste - Reuniao Virtual";
InicioDasessao:08.00.00;FimDasessao:09.30.00;

PALAVRA:PALAVRA;
IdDaPalavra:"Palavra Global";
TipoDaPalavra:1;
EstadoAtualDaPalavra:USADAREMOTAMENTE;
ControladorDePalavra:fabiano@200.135.48.101;

PRIMITIVA:PRIMITIVA;
NovoEstadoDaPalavra:REQUISITADA;
Mensagem:"Nao concordo com o Jose. Gostaria de expor meus argumentos."
```

Figura 20: Informações carregadas pela primitiva *RequisitaPalavra*

6.3.8 Simulação do ACM e serviço de controle de palavra para diferentes Cenários

A coordenação, especialmente a política empregada para o controle de palavra sobre recursos compartilhados (que é específica a cada contexto), é bastante difícil de realizar.

A dificuldade na realização do controle de palavra emana da falta de conhecimento sobre processos entre pessoas, envolvendo conflitos, interesses pessoais, confiança e privacidade. Aliado a estas dificuldades está a especificidade de cada contexto. Estas questões levam a se planejar um serviço de controle de palavra flexível que suporte múltiplas políticas de controle e também a implementação de novas.

Para testar o protótipo em diferentes contextos, será apresentada a troca de diretivas de controle de palavra em três cenários distintos, demandando diferentes políticas de controle de palavra.

Aspectos relativos ao controle da sessão [02] não serão contemplados neste protótipo: será considerado que as informações necessárias do controle de sessão, como o gerenciamento de usuários, conectividade, invocação de ferramentas e informações sobre o estado da sessão, já estão disponíveis em cada sítio para serem utilizados pelo agente de controle.

6.3.8.1 Simulação do Cenário 1 – Reunião de Pequeno Grupo

O Cenário 1, é caracterizado pela demanda de um ACM para realizar reuniões de pequenos grupos de pessoas, com cerca de cinco participantes geograficamente distribuídos, que se reúnem para discutirem resultados de um projeto que está sendo desenvolvido de forma colaborativa

compartilhando, nestes momentos os seguintes recursos: canais de áudio, vídeo, quadro branco e um tele-apontador sobre o ambiente gráfico local.

Para este cenário, o modo de controle selecionado é o controle remoto no transmissor [39]. Isso significa que as diretivas de controle trocadas entre os agentes de controle levam à realização do controle sobre cada aplicação de mídia (*mute áudio*, *mute vídeo*, *unmute whiteboard*, por exemplo) no sítio do transmissor.

Baseado em resultados de reuniões virtuais realizadas (no contexto dos estudos e análises apresentados no capítulo 4), e que seguem o modelo LWS [39], onde se utilizavam políticas sociais para chegar ao consenso na concessão da palavra, acredita-se que a política de controle mais adequada seja a política com moderador (*Chair-Guidance*), com modelo de controle centralizado.

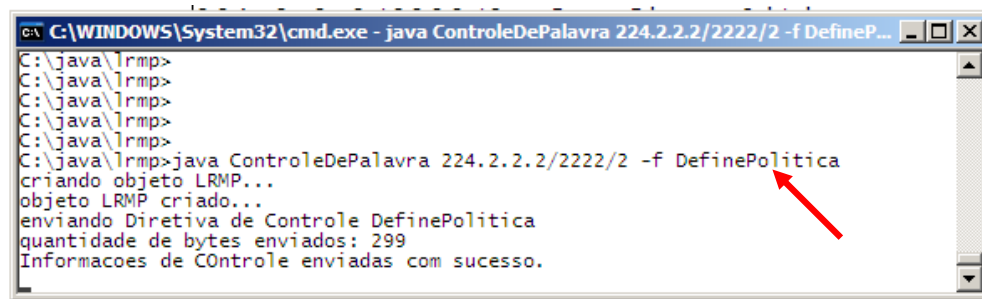
Embora seja possível alcançar o consenso através de políticas de controle de palavra sociais, acredita-se que o mais adequado seja realizar controle pessimista, evitando o conflito.

A política escolhida possui a requisição e liberação de palavra explícita, sendo que quando ocorrem várias requisições para o mesmo recurso, estas são colocadas numa fila FIFO (*First In First Out*).

Para este cenário, as requisições para obtenção de palavras deverão ser feitas diretamente ao Controlador de Palavras da sessão (*Floor Controller*) que, explicitamente concede a palavra por um tempo determinado. Se o detentor da palavra (*Floor Holder*) não liberar a palavra, excedendo demasiadamente o tempo estimado, o controlador da palavra pode revogar explicitamente a palavra.

Exemplo de Troca de Mensagens Entre os Agentes de Controle de Palavra

A seguir são apresentadas as trocas de mensagem ocorridas entre três participantes considerados para os três sítios da sessão: sítio 1 (moderador); sítio 2 (participante); sítio 3 (participante).

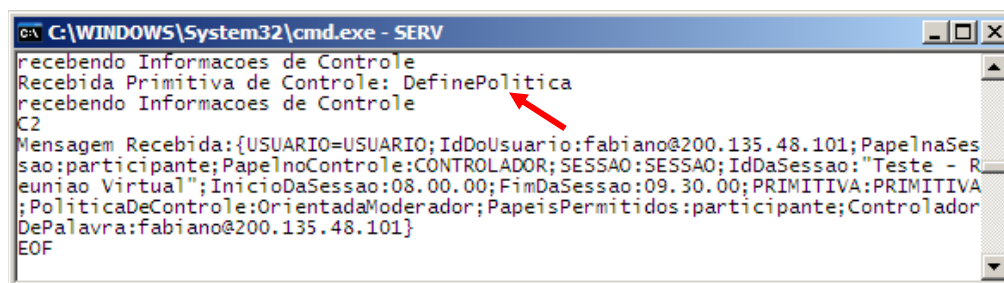


```

C:\WINDOWS\System32\cmd.exe - java ControleDePalavra 224.2.2.2/2222/2 -f DefineP...
C:\java\lrmp>
C:\java\lrmp>
C:\java\lrmp>
C:\java\lrmp>
C:\java\lrmp>
C:\java\lrmp>java ControleDePalavra 224.2.2.2/2222/2 -f DefinePolitica
criando objeto LRMP...
objeto LRMP criado...
enviando Diretiva de Controle DefinePolitica
quantidade de bytes enviados: 299
Informacoes de Controle enviadas com sucesso.
  
```

Figura 21: Transmissão da Primitiva de Controle *DefinePolitica*

Inicialmente, o moderador define a política e transmite para os demais participantes da sessão. A transmissão, a partir do sítio do moderador, pode ser vista na Figura 21, enquanto a recepção nos dois outros sítios é apresentada na Figura 22. Ambas as figuras apresentam também as informações carregadas pela primitiva de controle transmitida.



```

C:\WINDOWS\System32\cmd.exe - SERV
recebendo Informacoes de Controle
Recebida Primitiva de Controle: DefinePolitica
recebendo Informacoes de Controle
C2
Mensagem Recebida:{USUARIO=USUARIO;IdDoUsuario:fabiano@200.135.48.101;PapelnaSes
sao:participante;PapelnoControle:CONTROLADOR;SESSAO:SESSAO;IdDaSessao:"Teste - R
euniao Virtual";InicioDaSessao:08.00.00;FimDaSessao:09.30.00;PRIMITIVA:PRIMITIVA
;PoliticaDeControle:OrientadaModerador;PapeisPermitidos:participante;Controlador
DePalavra:fabiano@200.135.48.101}
EOF
  
```

Figura 22: Recepção das informações carregadas pela Primitiva de Controle *DefinePolitica* nos sítios

Analisando detalhadamente a mensagem enviada pelo moderador e recebida pelos participantes da sessão (Figura 22), pode-se obter as seguintes informações:

Mensagem recebida: Primitiva para definir política de controle
 Política de controle selecionada: OrientadaModerador
 Usuário que enviou a mensagem: Fabiano@200.135.48.101
 Papel do usuário que enviou a mensagem: CONTROLADOR
 Identificador da Sessão: Teste – Reunião Virtual
 Horário de início da sessão: 08:00:00h
 Horário de término da sessão: 09:30:00h

Com a política definida e publicada, o próximo passo é a criação de uma palavra. A Figura 23 e Figura 24 apresentam a transmissão a partir do sítio 1 e a recepção nos sítios 2 e 3 da primitiva *CriaPalavra* que cria uma palavra global e a coloca no estado INATIVO, ou seja, o acesso aos recursos compartilhados ainda não está sendo controlado.

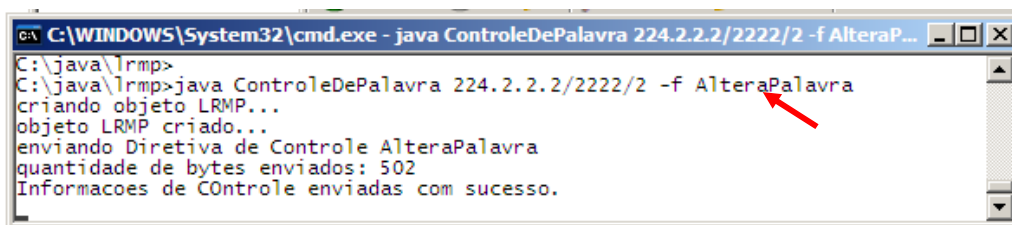
```
C:\WINDOWS\System32\cmd.exe - java ControleDePalavra 224.2.2.2/2222/2 -f CriaPalavra...
C:\java\lmp>java ControleDePalavra 224.2.2.2/2222/2 -f CriaPalavra
criando objeto LRMP...
objeto LRMP criado...
enviando Diretiva de Controle CriaPalavra
quantidade de bytes enviados: 342
Informacoes de Controle enviadas com sucesso.
```

Figura 23: Transmissão da Primitiva de Controle *CriaPalavra* a partir do sítio 1 (moderador)

```
C:\WINDOWS\System32\cmd.exe - SERV
recebendo Informacoes de Controle
Recebida Primitiva de Controle: CriaPalavra
recebendo Informacoes de Controle
C2
Mensagem Recebida:{USUARIO=USUARIO; IdDoUsuario:fabiano@200.135.48.101; PapelnaSessao:participante; PapelnoControle:CONTROLADOR; SESSAO:SESSAO; IdDaSessao:Teste - Reunião Virtual; InicioDaSessao:08.00.00; FimDaSessao:09.30.00; PRIMITIVA:PRIMITIVA; IdDaPalavra:Palavra Global; TipoDaPalavra:1; ControladorDePalavra:fabiano@200.135.48.101; NovoEstadoDaPalavra:INATIVA}
EOF
```

Figura 24: Recepção da Primitiva de Controle *CriaPalavra* pelos sítios 2 e 3

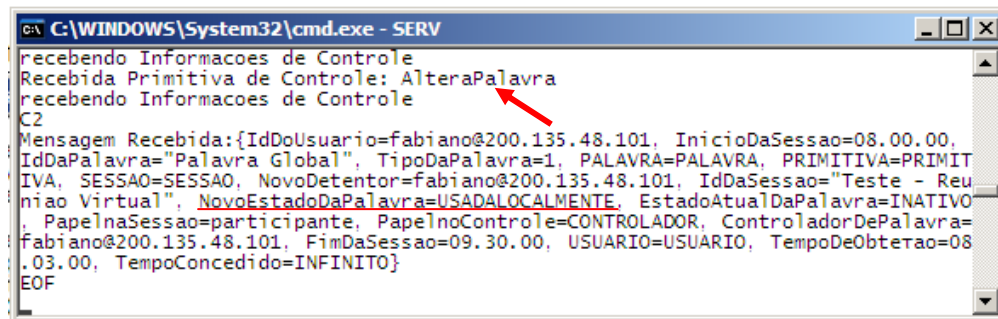
O controle sobre os recursos passa a existir a partir do momento em que a palavra sai do estado INATIVO. A Figura 25 e a Figura 26 mostram a transmissão a partir do sítio 1 e recepção nos sítios 2 e 3 da primitiva *AlterarPalavra*. Neste momento a palavra passa a estar ativa e o detentor da palavra passa a ser o próprio moderador que, efetivamente, inicia a reunião.



```

C:\WINDOWS\System32\cmd.exe - java ControleDePalavra 224.2.2.2/2222/2 -f AlteraP...
C:\java\lrmp>
C:\java\lrmp>java ControleDePalavra 224.2.2.2/2222/2 -f AlteraPalavra
criando objeto LRMP...
objeto LRMP criado...
enviando Diretiva de Controle AlteraPalavra
quantidade de bytes enviados: 502
Informacoes de COntrole enviadas com sucesso.
  
```

Figura 25: Transmissão da Primitiva de Controle *AlterarPalavra* a partir do sítio 1



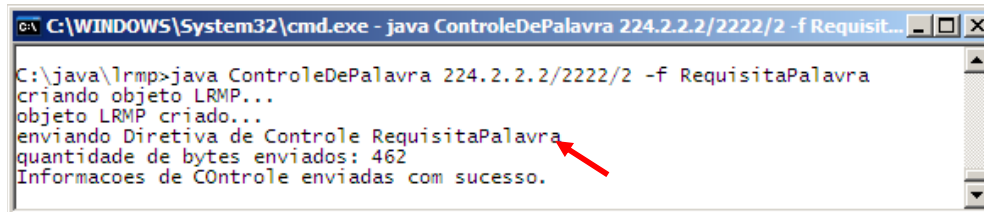
```

C:\WINDOWS\System32\cmd.exe - SERV
recebendo Informacoes de Controle
Recebida Primitiva de Controle: AlteraPalavra
recebendo Informacoes de Controle
C2
Mensagem Recebida:{IdDoUsuario=fabiano@200.135.48.101, InicioDaSessao=08.00.00,
IdDaPalavra="Palavra Global", TipoDaPalavra=1, PALAVRA=PALAVRA, PRIMITIVA=PRIMITIVA,
SESSAO=SESSAO, NovoDetentor=fabiano@200.135.48.101, IdDaSessao="Teste - Reunião Virtual",
NovoEstadoDaPalavra=USADALOCALMENTE, EstadoAtualDaPalavra=INATIVO,
PapelnaSessao=participante, PapelnoControle=CONTROLADOR, ControladorDePalavra=fabiano@200.135.48.101,
FimDaSessao=09.30.00, USUARIO=USUARIO, TempoDeObterao=08.03.00, TempoConcedido=INFINITO}
EOF
  
```

Figura 26: Recepção da Primitiva de Controle *AlterarPalavra* no sítios 2 e 3

Neste momento, o moderador, participante no sítio 1, inicia o controle sobre os recursos compartilhados através de uma palavra global, ou seja, todos os recursos estão acoplados [02]. Em seguida, o participante no sítio 2 requisita a palavra (Figura 27). Esta primitiva é recebida por todos os sítios participantes mas, devido ao modelo e política

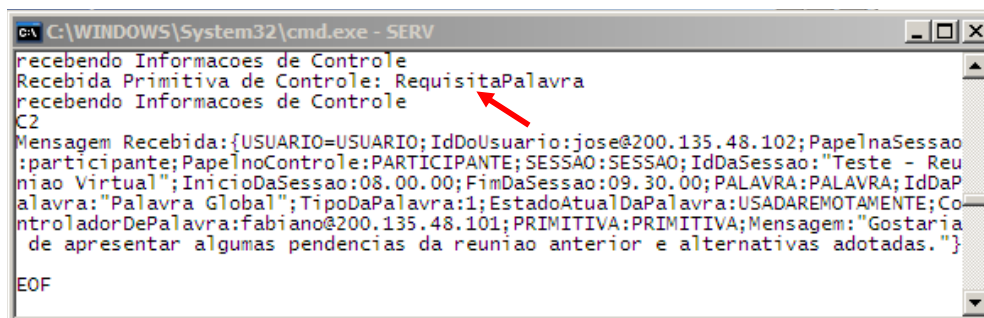
de controle escolhidos, o moderador coloca a requisição numa fila, baseado nas informações recebidas (Figura 28), (cabendo a ele conceder explicitamente a palavra) e os demais participantes descartam e não tratam a informação sobre este evento remoto.



```

C:\WINDOWS\System32\cmd.exe - java ControleDePalavra 224.2.2.2/2222/2 -f Requisit...
C:\java\lrmp>java ControleDePalavra 224.2.2.2/2222/2 -f RequisitaPalavra
criando objeto LRMP...
objeto LRMP criado...
enviando Diretiva de Controle RequisitaPalavra
quantidade de bytes enviados: 462
Informacoes de Controle enviadas com sucesso.
  
```

Figura 27: Transmissão da Primitiva de Controle *RequisitaPalavra* a partir do sitio 2

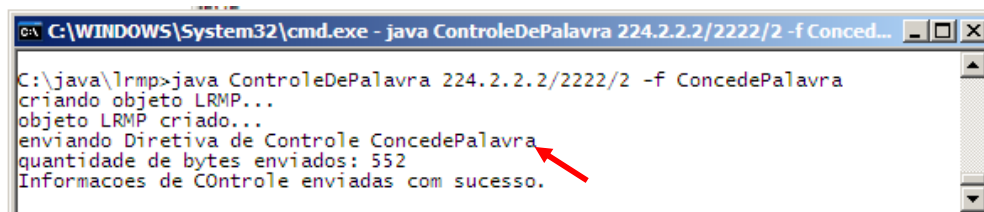


```

C:\WINDOWS\System32\cmd.exe - SERV
recebendo Informacoes de Controle
Recebida Primitiva de Controle: RequisitaPalavra
recebendo Informacoes de Controle
C2
Mensagem Recebida:{USUARIO=USUARIO;IdDoUsuario:jose@200.135.48.102;PapelnaSessao
:participante;PapelnoControle:PARTICIPANTE;SESSAO:SESSAO;IdDaSessao:"Teste - Reu
niao Virtual";InicioDaSessao:08.00.00;FimDaSessao:09.30.00;PALAVRA:PALAVRA;IdDaP
alavra:"Palavra Global";TipoDaPalavra:1;EstadoAtualDaPalavra:USADAREMOTAMENTE;Co
ntroladorDePalavra:fabiano@200.135.48.101;PRIMITIVA:PRIMITIVA;Mensagem:"Gostaria
de apresentar algumas pendencias da reuniao anterior e alternativas adotadas."}
EOF
  
```

Figura 28: Recepção da Primitiva de Controle *RequisitaPalavra* nos sitios 1 e 3

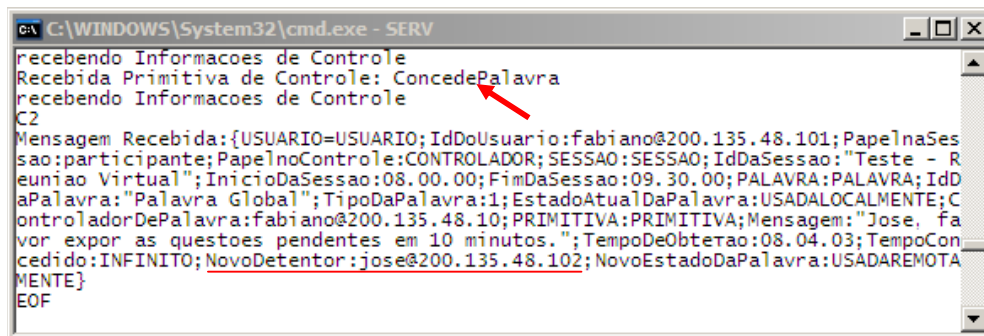
Na seqüência o moderador concede a palavra ao sitio 2, que a requisitou (Figura 29 e Figura 30).



```

C:\WINDOWS\System32\cmd.exe - java ControleDePalavra 224.2.2.2/2222/2 -f Conced...
C:\java\lrmp>java ControleDePalavra 224.2.2.2/2222/2 -f ConcedePalavra
criando objeto LRMP...
objeto LRMP criado...
enviando Diretiva de Controle ConcedePalavra
quantidade de bytes enviados: 552
Informacoes de Controle enviadas com sucesso.
  
```

Figura 29: Transmissão da Primitiva de Controle *ConcedePalavra* a partir do sitio 1



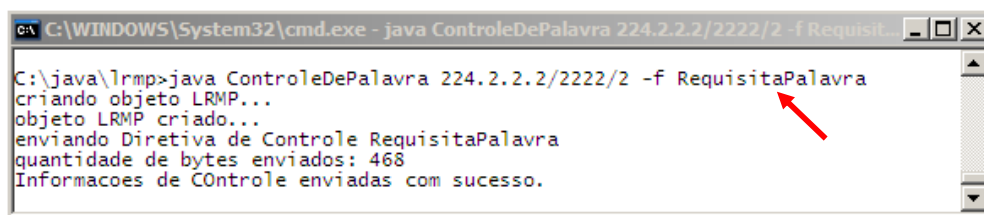
```

C:\WINDOWS\System32\cmd.exe - SERV
recebendo Informacoes de Controle
Recebida Primitiva de Controle: ConcedePalavra
recebendo Informacoes de Controle
C2
Mensagem Recebida:{USUARIO=USUARIO;IdDoUsuario:fabiano@200.135.48.101;PapelnaSes
sao:participante;PapelnoControle:CONTROLADOR;SESSAO:SESSAO;IdDaSessao:"Teste - R
euniao Virtual";InicioDaSessao:08.00.00;FimDaSessao:09.30.00;PALAVRA:PALAVRA;IdD
aPalavra:"Palavra Global";TipoDaPalavra:1;EstadoAtualDaPalavra:USADALOCALMENTE;C
ontroladorDePalavra:fabiano@200.135.48.10;PRIMITIVA:PRIMITIVA;Mensagem:"Jose, fa
vor expor as questoes pendentes em 10 minutos.";TempoDeObterao:08.04.03;TempoCon
cedido:INFINITO;NovoDetentor:jose@200.135.48.102;NovoEstadoDaPalavra:USADAREMOTA
MENTE}
EOF

```

Figura 30: Recepção da Primitiva de Controle *ConcedePalavra* nos sítios 2 e 3

Considera-se, por exemplo, que esta vez, o participante no sítio 3, não concordando com o exposto pelo detentor da palavra, requisita a palavra (Figura 31) ao moderador para apresentar um contraponto. A requisição é recebida (Figura 32) por todos colocada na fila das requisições não atendidas no sítio do moderador. O estado da palavra muda para REQUISITADA nos sítios que mantêm uma tabela de estados para as palavras da sessão.



```

C:\WINDOWS\System32\cmd.exe - java ControleDePalavra 224.2.2.2/2222/2 -f Requisit
C:\java\lrmp>java ControleDePalavra 224.2.2.2/2222/2 -f RequisitaPalavra
criando objeto LRMP...
objeto LRMP criado...
enviando Diretiva de Controle RequisitaPalavra
quantidade de bytes enviados: 468
Informacoes de Controle enviadas com sucesso.

```

Figura 31: Transmissão da Primitiva de Controle *RequisitaPalavra* a partir no sítio 3

```

C:\WINDOWS\System32\cmd.exe - SERV
EOF
recebendo Informacoes de Controle
Recebida Primitiva de Controle: RequisitaPalavra
recebendo Informacoes de Controle
C2
Mensagem Recebida:{USUARIO=USUARIO;IdDoUsuario:maria@200.135.48.103;PapelnaSessao:participante;PapelnoControle:PARTICIPANTE;SESSAO:SESSAO;IdDaSessao:"Teste - Reuniao Virtual";InicioDaSessao:08.00.00;FimDaSessao:09.30.00;PALAVRA:PALAVRA;IdDaPalavra:"Palavra Global";TipoDaPalavra:1;EstadoAtualDaPalavra:USADAREMOTAMENTE;ControladorDePalavra:fabiano@200.135.48.101;PRIMITIVA:PRIMITIVA;NovoEstadoDaPalavra:REQUISITADA;Mensagem:"Nao concordo com o Jose. Gostaria de expor meus argumentos."}
EOF

```

Figura 32: Recepção da Primitiva de Controle *RequisitaPalavra* nos sítios 1 e 2

O participante do sítio 2, atual detentor da palavra, percebe a intenção da participante do sítio 3 e voluntariamente libera a palavra (Figura 33). A liberação é transmitida a todos e o estado da palavra passa a ser LIVRE (Figura 34).

```

C:\WINDOWS\System32\cmd.exe
criando objeto LRMP...
objeto LRMP criado...
enviando Diretiva de Controle LiberaPalavra
quantidade de bytes enviados: 406
Informacoes de Controle enviadas com sucesso.

C:\java\lrmv>
C:\java\lrmv>java ControleDePalavra 224.2.2.2/2222/2 -f LiberaPalavra

```

Figura 33: Transmissão da Primitiva de Controle *LiberaPalavra* a partir no sítio 2

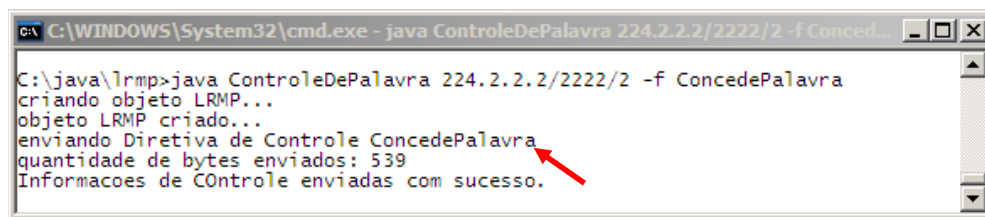
```

C:\WINDOWS\System32\cmd.exe - SERV
EOF
recebendo Informacoes de Controle
Recebida Primitiva de Controle: LiberaPalavra
recebendo Informacoes de Controle
C2
Mensagem Recebida:{USUARIO=USUARIO;IdDoUsuario:jose@200.135.48.102;PapelnaSessao:participante;PapelnoControle:DETENTOR;SESSAO:SESSAO;IdDaSessao:"Teste - Reuniao Virtual";InicioDaSessao:08.00.00;FimDaSessao:09.30.00;PALAVRA:PALAVRA;IdDaPalavra:"Palavra Global";TipoDaPalavra:1;EstadoAtualDaPalavra:USADALOCALMENTE;ControladorDePalavra:fabiano@200.135.48.101;PRIMITIVA:PRIMITIVA;Mensagem:"Terminei.";NovoEstadoDaPalavra:LIVRE}
EOF

```

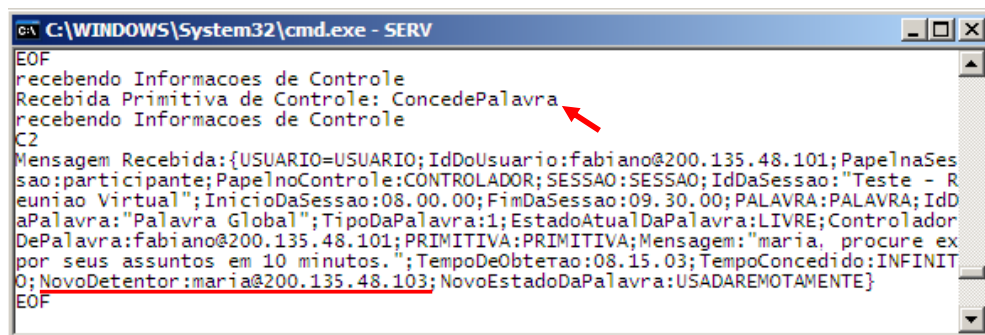
Figura 34: Recepção da Primitiva de Controle *LiberaPalavra* nos sítio 1 e 3

A partir daí o moderador envia primitiva concedendo a palavra (Figura 35) ao primeiro requisitante da fila de requisições não atendidas, sítio 3. Todos que recebem a primitiva (Figura 36) atualizam suas informações locais e a participante do sítio 3 passa a deter o direito de utilizar exclusivamente os recursos por tempo indeterminado.



```
C:\WINDOWS\System32\cmd.exe - java ControleDePalavra 224.2.2.2/2222/2 -f Concede...
C:\java\lrmp>java ControleDePalavra 224.2.2.2/2222/2 -f ConcedePalavra
criando objeto LRMP...
objeto LRMP criado...
enviando Diretiva de Controle ConcedePalavra
quantidade de bytes enviados: 539
Informacoes de Controle enviadas com sucesso.
```

Figura 35: Transmissão da Primitiva de Controle *ConcedePalavra* a partir no sítio 1 (controlador)



```
C:\WINDOWS\System32\cmd.exe - SERV
EOF
recebendo Informacoes de Controle
Recebida Primitiva de Controle: ConcedePalavra
recebendo Informacoes de Controle
C2
Mensagem Recebida:{USUARIO=USUARIO;IdDoUsuario:fabiano@200.135.48.101;PapelnaSes
sao:participante;PapelnoControle:CONTROLADOR;SESSAO:SESSAO;IdDaSessao:"Teste - R
euniao Virtual";InicioDaSessao:08.00.00;FimDaSessao:09.30.00;PALAVRA:PALAVRA;IdD
aPalavra:"Palavra Global";TipoDaPalavra:1;EstadoAtualDaPalavra:LIVRE;Controlador
DePalavra:fabiano@200.135.48.101;PRIMITIVA:PRIMITIVA;Mensagem:"maria. procure ex
por seus assuntos em 10 minutos.";TempoDeObterao:08.15.03;TempoConcedido:INFINIT
O;NovoDetentor:maria@200.135.48.103;NovoEstadoDaPalavra:USADAREMOTAMENTE}
EOF
```

Figura 36: Recepção da Primitiva de Controle *ConcedePalavra* nos sítios 2 e 3

Comentários sobre o Controle de Palavra

A troca das mensagens irá depender de vários fatores, como: eventos ao longo da sessão, política de controle utilizada, mecanismos de controle empregados, modelo de controle (centralizado, híbrido ou distribuído) utilizado. Estes fatores, além de influenciarem a periodicidade, momento da troca, e informações carregadas pelas

mensagens, influenciam também a forma de tratamento dada às informações recebidas. Por exemplo, para um modelo de controle distribuído, os eventos remotos e informações trocadas entre os diversos sítios são fonte de informação para conhecer o estado e criar histórico da utilização de recursos, palavras e eventos locais e remotos de uma sessão.

Para este primeiro cenário foram simuladas trocas de mensagens disparadas por eventos típicos de uma sessão desta natureza, como definição de política, criação de uma palavra global (dependendo do tratamento dado, todos os recursos podem estar acoplados e associados a mesma palavra que circula entre os sítios participantes da sessão) e requisição, concessão e liberação da palavra. Outros eventos poderiam disparar outras mensagens que, aqui, foram omitidas.

6.3.8.2 Simulação do Cenário 2 – Seminário para Grande Grupo

O Cenário 2 é caracterizado pela demanda de um ACM adequado para seminários para grande número de pessoas com diferentes papéis atribuídos (palestrante, professor, aluno, empresário, visitante), sendo a interação com o palestrante permitida através de áudio e vídeo apenas para professor, empresário ou outro palestrante. O participante Aluno poderá interagir via *chat*, e participante Visitante não poderá interagir com os demais.

Para este cenário, como se trata de uma sessão aberta, deve-se prevenir contra participantes que não agem de acordo com as regras (por exemplo, participantes que ignoram a política de controle selecionada). Para lidar com esta situação, o controle de palavra pode ser realizado no sítio do receptor [39].

Enquanto no cenário 1 tinha-se um cenário com certo grau de garantia de que todos os participantes da sessão estavam utilizando o agente de controle em cada um dos sítios participantes, no cenário 2 não se tem esta garantia. Assim, é possível que um participante ignore as diretivas de controle trocadas entre os sítios (e que localmente enviam mensagens para cada uma das aplicações de mídia) e passe a acessar um recurso compartilhado sem autorização como, por exemplo, canal de áudio ou vídeo. Para evitar esta situação, pode-se dar outro tratamento às informações recebidas (por meio das diretivas de controle) pelo agente de controle em cada sítio: cada agente passa, neste cenário a controlar a recepção das mídias, e não mais a transmissão, como acontecia no cenário 1.

A Figura 37 apresenta um exemplo em que o agente de controle recebe informações do moderador de que apenas o sítio 2 estará transmitindo áudio, vídeo e utilizando a ferramenta de quadro branco. A partir daí o agente envia diretivas de controle local, através de um barramento local, informando as aplicações para ignorar a recepção de dados provenientes de qualquer sítio, exceto do sítio 2. A partir daí, qualquer participante malicioso terá sua interferência (áudio e vídeo, por exemplo) bloqueados na recepção.

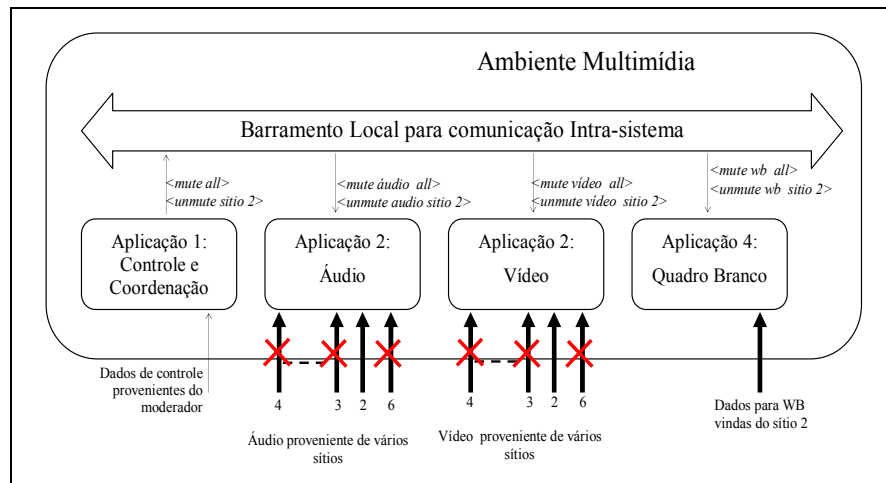


Figura 37: Controle Realizado Localmente no Sítio do Receptor

Entretanto, se todos os participantes concordarem e seguirem as regras, ambos os modelos de controle (no receptor ou do transmissor) são equivalentes.

Da mesma forma como no Cenário 1, aqui se entende que a política de controle mais adequada seja moderada (*Chair-Guidance*), com modelo de controle centralizado e controle pessimista, evitando o conflito.

A política escolhida deve suportar a requisição e liberação de palavra explícita, sendo que quando ocorrem várias requisições para o mesmo recurso, estas são colocadas numa fila FIFO (*First In First Out*). As requisições para obtenção de palavras também deverão ser feitas diretamente ao Controlador de Palavras da sessão (*Floor Controller*) que, explicitamente concede a palavra por um tempo determinado. Se ao final do tempo concedido o detentor da palavra (*Floor Holder*) não liberar a palavra, ocorre a liberação implícita, ou revogação explícita, e a palavra volta para o Controlador de palavras da sessão. Para este cenário as palavras associadas aos recursos de áudio e vídeo são permitidas apenas para *Palestrante*, *Professor* e *Empresário*. O participante *Aluno* juntamente

com *Palestrante*, *Professor* e *Empresário*, têm acesso à palavra do recurso *chat*.

Troca de Mensagens Entre os Agentes de Controle de Palavra

Para este cenário a troca de mensagens será muito semelhante à apresentada no cenário 1. A diferença entre os dois cenários está no tempo de concessão que passa a ser determinado antes da concessão da palavra e, conseqüentemente, esgotando-se o tempo, ocorre uma liberação implícita da palavra. Esta diferença implica apenas na alteração dos valores das variáveis relacionadas ao tempo de concessão da palavra e no mecanismo de monitoramento do tempo utilizado no sítio do detentor da palavra.

6.3.8.3 Simulação do Cenário 3 – Sala de Aula Virtual

O cenário 3 é caracterizado pela demanda de um ACM adequado para sala de aula virtual onde se tem dois papéis definidos (aluno e professor). Inicialmente o professor ministra uma aula para todos e, a partir daí são definidos pequenos grupos que passam a trabalhar em atividades paralelas.

O professor, que desempenha o papel de controlador das palavras (*Floor Controler*), participa de todos os grupos (sub-sessões) avaliando e orientando os alunos. Ao definir as sub-sessões, para cada uma delas é definido um detentor (*Floor Holder*) das palavras associadas aos recursos utilizados. O professor, como controlador das palavras, mantém privilégios especiais para intervir quando necessário em cada um dos grupos.

Para este cenário, considerando a natureza das interações e objetivo da colaboração considera-se adequado o controle de palavra com as seguintes características:

- Política autônoma, ou seja, sem a presença de um moderador: Após o controlador definir política, criar palavras e atribuir as palavras a um participante de cada um dos grupos de alunos, o gerenciamento das palavras será feito sem a presença de moderador em cada um dos grupos;
- Requisição explícita da palavra: o interessado deverá explicitamente demonstrar interesse em obter a palavra;
- Preempção: uma palavra pode ser revogada implicitamente quando o tempo de concessão expirar.
- Exclusão mútua: garante acesso exclusivo ao recurso, evitando o conflito;
- Controle Distribuído: os agentes tomam as decisões (como as de concessão e revogação de uma palavra) baseado em informações mantidas distribuídas em cada um dos sítios. Estas informações são obtidas a partir da detecção de atividades nos sítios remotos, que chegam aos sítios participantes por meio das diretivas de controle trocadas entre os agentes de controle dos sítios. Exemplos de informações são: número de participantes que requisitam determinado recurso, tempo que falta para revogação de determinada palavra, detentor de cada palavra e estado de cada palavra.

Para viabilizar a política com as características acima se acredita que os mecanismos mais adequados e facilmente implementáveis são a passagem de bastão do detentor da palavra para o primeiro da fila de participantes requisitantes (informação que está distribuída em cada sítio) e percepção de atividades remotas a partir das diretivas de controle (para montar base de informações distribuída em cada sítio).

As informações referentes à fila de requisições são mantidas distribuídas em cada um dos sítios a partir das informações de eventos remotos coletados da rede, de forma que o novo detentor da palavra sabe exatamente para quem irá repassar a palavra quando liberá-la.

Caso a palavra seja liberada e não existam requisitantes, é difundida uma informação de estado “LIVRE” para a palavra. Assim, neste cenário existe a possibilidade de conflitos que deverão ser resolvidos, ao passo que mais de um participante pode tentar obter a palavra “LIVRE” ao mesmo tempo. A palavra apenas é efetivamente obtida quando todos os sítios de uma sessão atualizarem a informação local do estado da referida palavra.

Troca de Mensagens Entre os Agentes de Controle de Palavra

Para este cenário, da mesma forma que para o cenário 2, embora com características muito diferentes do cenário 1, não apresentará diferenças significativas na troca das mensagens de controle.

O que efetivamente irá alterar de um cenário para outro é o tratamento dado às informações a medida que elas forem sendo recebidas pelos agentes de controle a partir de eventos remotos e locais. O tratamento destas informações para os cenários não foi realizado neste trabalho.

6.4 Conclusões do Capítulo

A decomposição de um ACM em aplicações de mídia, de controle de sessão, de controle de palavra mostrou-se bastante adequado para abordar o assunto. Com o modelo de ACM apresentado na Figura 12, pode-se concentrar e atacar especificamente o problema relacionado ao controle de palavra, que tem sido identificado como uma lacuna para os ambientes criados a partir de ferramentas e aplicações para conferências através da *Internet*.

O modelo apresentado também permite que novas funcionalidades, tanto para as aplicações de mídias quanto para os agentes de controle e coordenação (Figura 11) sejam facilmente integradas e melhoradas. Um exemplo é a implementação de novas políticas de controle de palavra quando estas se fizerem necessárias para atender as especificidades de determinado cenário.

Seguindo este modelo, foi implementado e apresentado um protótipo, utilizando linguagem de programação Java e pacote de bibliotecas reutilizáveis do protocolo LRMP [21]. Dessa forma pôde-se apresentar uma infra-estrutura básica para um serviço de controle de palavra para ACMs utilizando endereçamento *multicast*, com características desejáveis como portabilidade, escalabilidade, suporte a várias políticas e mecanismos para controle de palavra e flexibilidade para atender especificidades de cada cenário. O modelo baseado em troca de mensagens também incrementa o grau de interoperabilidade do protótipo.

O protótipo e a troca de mensagens de controle de palavra foram apresentadas em três cenários bastante distintos e freqüentemente utilizados. Assim foi possível demonstrar que as especificidades de cada cenário podem ser absorvidas por bibliotecas de tratamento das

informações em cada sítio, podendo o modelo e a infra-estrutura básica para troca de mensagens de controle de palavra ser utilizados de forma eficaz numa gama grande de cenários e contextos.

A manutenção das características da infra-estrutura de controle e das mensagens trocadas para cada um dos cenários, por mais que estes sejam diferentes e específicos, mostra que a infra-estrutura para o serviço de controle de palavra demonstrado pode se adaptar e abranger a grande variedade de cenários e contextos que demandam o emprego de ACMs.

Para viabilizar a demonstração do protótipo, foi definido um conjunto de mensagens e seus respectivos valores e semântica das mensagens a serem trocadas entre os agentes de controle em cada um dos sítios. Dentre as possíveis atividades para o futuro está a redefinição deste protocolo básico para troca de diretivas de controle ou adoção de outro protocolo para controle de palavra já existente.

O tratamento das informações de controle trocadas entre os agentes de controle em cada sítio de uma sessão ainda não foi realizado e está entre as atividades futuras, na direção de uma implementação completa de serviço de controle de palavra para ACM através de redes IP (redes locais, corporativas e *Internet*), baseado em aplicações e ferramentas gratuitas.

Capítulo 7

CONCLUSÕES

Novas aplicações e ferramentas continuam dando origem a ambientes que suportam colaborações com múltiplas mídias e vários estilos de interação. Para atender estas aplicações, avanços têm sido realizados em tecnologias como roteamento *multicast*, *multicast* confiável, algoritmos de compressão de mídias e tecnologias de acesso.

Embora exista um número razoável de softwares para *groupware* síncrono, a limitação na colaboração, sobretudo nos softwares gratuitos, continua presente e comprometendo o desempenho destes.

Esta limitação na colaboração é devida ao fato de que processos cooperativos e colaborativos entre humanos, que envolvem questões de conflito, considerações pessoais, interesses pessoais, confiança e

privacidade, não são completamente entendidos, especialmente quando computadores são utilizados para suportar tais processos.

Esta limitação na colaboração também foi identificada durante os testes e avaliação das plataformas, aplicativos e ferramentas para *groupware* síncrono. Também foram, durante os testes, identificadas limitações na colaboração, originadas por limites tecnológicos. Estes podem ser classificados, basicamente, em duas categorias: limites tecnológicos de infra-estrutura da comunicação e limites tecnológicos nos sistemas finais que hospedam os ambientes multimídia em cada sítio.

Em relação à infra-estrutura de comunicação para ambientes colaborativos multimídia, devido às características destes no tocante à quantidade de dados transmitidos, dependência temporal, necessidade de continuidade temporal e sincronização intra e intermídia, os requisitos vão além de uma largura de banda disponível maior para a implantação de tais ambientes. Além da largura de banda, figuram como necessidades, também, requisitos mais rígidos para latência e *jitter* da rede para garantir qualidade exigida para os ambientes que empregam ferramentas colaborativas multimídia síncronas.

Além destes requisitos, fica evidente a necessidade de se dispor de uma infra-estrutura de comunicação que herde as características e vantagens, como escalabilidade e flexibilidade para gerenciamento de grupos, do endereçamento *multicast*, mas que também disponha de modo de transmissão confiável para suportar a troca de mensagens entre aplicações da coordenação entre ambientes multimídia que compõem um ACM. Esta confiabilidade sobre canais de comunicação *multicast* pode ser obtida através do emprego de um protocolo de *multicast* confiável, que implementa mecanismos para ordenação, detecção e correção de erros, já

que o modelo de comunicação *multicast* não provê transporte confiável nativo para os dados.

Assim, o modelo de comunicação baseado no endereçamento *multicast*, que pode fazer uso de mecanismos adequados para prover comunicação entre sub-redes *multicast* através de esquemas de tunelamento e para viabilizar transferência de dados confiável e ordenados, mostra-se adequado para compor a infra-estrutura de comunicação que suporte os ambientes colaborativos multimídia através da *Internet* ou, de uma forma geral, através de redes comutadas por pacotes, tanto para comunicação multimídia quanto para troca de informações de controle.

Os testes e avaliações de plataformas, aplicativos e ferramentas para *groupware* síncrono, realizados no contexto deste trabalho, tomaram como base três cenários nos os quais se pretendia empregar um ambiente para trabalho colaborativo: reuniões de pequenos grupos, aulas virtuais e seminários via *Internet* para grandes platéias.

Pelo fato de cada um dos cenários considerados ter necessidades específicas pôde-se avaliar a flexibilidade das plataformas, ferramentas e aplicativos testados. Com os resultados dos testes pode-se destacar que nenhuma plataforma completa atendeu plenamente a necessidade em relação aos serviços de coordenação pretendidos, sobretudo ao controle de palavra.

Dentre os aplicativos de mídia e ferramentas de apoio testadas destaca-se que a única ferramenta testada que busca prover suporte para controle de palavra, o QB, apresentou problemas de integração com os aplicativos de áudio (VAT) e vídeo (VIC), não atendendo as demandas dos cenários apresentados. É importante destacar também que os aplicativos de mídia desenvolvidos a partir da API Java JMF se mostraram: bastante

estáveis, funcionando sem problemas em sistemas operacionais Windows e Linux por muitas horas consecutivas; portáteis; interoperáveis, interagindo com outras ferramentas através do uso de algoritmos de codificação padronizados e largamente utilizados; flexíveis, suportando modos de comunicação *unicast* e *multicast*; e tendo seu código disponível na *Internet*.

Conclui-se, entretanto, que para atender a grande variedade de cenários que demandam ACMs com suporte a vários estilos de comunicação e cooperação e peculiaridades que lhes são inerentes, um “sistema único para conferências” não é adequado. Neste contexto, propõe-se a decomposição do problema da comunicação homem-homem através de sistemas de comunicação eletrônicos segundo as diretrizes apresentadas em [39] e se concentrar na diretriz “Coordenação”, independente das demais, mais especificamente no controle de palavra.

A coordenação em ACMs se propõe a resolver problemas de alocação de recursos em termos de sua disponibilidade e planejamento de utilização para permitir o máximo de sinergia entre os participantes, podendo contemplar controle de palavra, sincronização entre aplicações e segurança. Neste sentido, o controle de palavra é uma tecnologia de coordenação para lidar com conflitos em ambientes colaborativos. Palavras, por sua vez, são permissões temporárias para acessos e manipulação de algum recurso compartilhado em um ACM.

A decomposição de um ACM em aplicações de mídia, de controle de sessão, de controle de palavra mostrou-se bastante adequada para abordar o assunto. Com o modelo de ACM apresentado na Figura 12, pode-se concentrar e atacar especificamente o problema relacionado ao controle de palavra, que tem sido identificado como uma lacuna para os ambientes criados a partir de ferramentas e aplicações gratuitas para

conferências através da *Internet*, testados e avaliados durante este trabalho.

O modelo adotado permite que novas funcionalidades, tanto para as aplicações de mídias quanto para as aplicações de controle e coordenação (Figura 11) sejam facilmente integradas e melhoradas.

Seguindo este modelo, foi apresentada uma infra-estrutura básica para um serviço de controle de palavra para ACMs utilizando endereçamento *multicast*, com características desejáveis como portabilidade, escalabilidade, suporte a várias políticas e mecanismos para controle de palavra e flexibilidade para atender especificidades de cada cenário. A infra-estrutura é baseada numa implementação de *multicast* confiável, o LRMP, permitindo boa escalabilidade para a ferramenta.

Foi possível verificar que o protótipo de controle de palavra, baseado na troca de mensagens ASCII, pode ser utilizado em diversos cenários, trocando um conjunto conhecido de primitivas de controle e parâmetros.

Do ponto de vista do usuário, as demandas específicas de cada cenário devem ser contempladas no diferente tratamento das informações obtidas em cada sítio através das mensagens de controle trocadas.

O tratamento das informações de controle trocadas entre os agentes de controle em cada sítio de uma sessão ainda não foi realizado e está entre as atividades futuras, na direção de uma implementação completa de serviço de controle de palavra para ACM através de redes IP (redes locais, corporativas e *Internet*), baseado em aplicações e ferramentas gratuitas.

Outro trabalho a ser desenvolvido é a comunicação entre o agente de coordenação e as demais aplicações de mídia, chamada de

“comunicação intra-sistema”, por meio de mecanismos como o Mbus [44], JMS [45], ou ainda através de bibliotecas reutilizáveis Java com protocolo simplificado, que faz uso de um grupo *multicast* local, ou seja, com TTL dos pacotes IP definido em zero em endereço e porta conhecida [37].

REFERÊNCIAS

- [01] BOYD Jr, J. A.; **FLOOR CONTROL IN SYNCHRONOUS GROUPWARE;** Dissertation. Pag. 20-25, n. 02. 1996. Ohio.
- [02] DOMMEL, H.-P.; J. J. GARCIA-LUNA-ACEVES. **Floor control For Multimedia Conferencing and Collaboration.** Baskin Center for Computer Engineering & Information Sciences University of California, Santa Cruz, CA, USA.
- [03] DOMMEL, H.-P.; J. J. GARCIA-LUNA-ACEVES; **A Coordination Architecture for Internet Groupwork.** Proc 26th EUROMICRO Conference – Informatics: Inventing the future, Whorkshop on

Multimedia and Telecommunications, Maastricht, Netherlands, Set 2000. IEEE.

- [04] DOMMEL, H.-P.; J. J. GARCIA-LUNA-ACEVES. **Network Support for Group Coordination**. Santa Clara University. Santa Clara, CA, USA.
- [05] DOMMEL, H.-P.; J. J. GARCIA-LUNA-ACEVES. **Efficacy of Floor Control Protocols in Distrubuted Multimedia Collaboration**. Cluster Computing J, 2(1):17-33, 1999.
- [06] SELLEN, A. J.. **Remote Conversations: The effects of mediating talk with technology**. Human Computer Interaction, 10, pp 401-444.
- [07] BACHMANN, F; **Endereçamento Multicast e Aplicações Multimídia Distribuídas**; Revista de Automação e Tecnologia da Informação. Pag. 20-25, n. 02. Vol. 01. Jul-dez. 2002. Florianópolis. SC.
- [08] BACHMANN, F., FARINES, J-M, MARTINS, I. L.; **Endereçamento Multicast e Aplicações Multimídia Distribuídas na RMAV-FLN**; II Workshop RNP2; Belo Horizonte - MG, 2000.
- [09] ROSS, Keith W. e KUROSE, James F. **Computer Networking: a top-down approach featuring the Internet**. Boston: Addison Wesley, Inc., 2001.
- [10] FLUCKIGER, François. **Undersanding Networked Multimedia: Applications and Technology**. Prentice Hall International (UK) Limited, 1995.
- [11] BACHMANN, F., SARI, S., MARTINS, I. L.; **Avaliação do TV Multicast da RMAV-FLN**; Documento Interno. NURCAD; Florianópolis - SC, 2000.
- [12] OLIVEIRA, R. D. V.; **Serviços Diferenciados em Redes IP: Medições e Testes para Aplicações envolvendo Mídias Contínuas**; Dissertação. PGEEL - DAS; Florianópolis - SC, 2001.
- [13] CISCO SYSTEMS. **Designing Internetworks for Multimedia**; Disponível em <http://www.cisco.com/univercd/cc/td/doc/cisintwk>. Acessado em 2002.
- [14] CISCO SYSTEMS. **Cisco - Building Consistent Quality of Service Into the Network**; Disponível em <http://www.cisco.com/warp/public/674/6.html>. Acessado em 2002.

- [15] JACOBSON, V. **Multimedia Conferencing on the Internet Tutorial**, Technical Report, Lawrence Berkeley Laboratory, nov. 1994.
- [16] TANENBAUM, Andrew S. **Redes de computadores**. Rio de Janeiro: Campus, 1996.
- [17] LIU, Chunlei. **Multimedia Over IP: RSVP, RTP, RTCP, RTSP**. Ohio, 2000. Disponível em: <http://www.cis.ohio-state.edu/~jain/cis788-97/ip_multimedia/index.htm>. Acesso em out. 2001.
- [18] V. KUMAR; "**MBone: Interactive Multimedia on The Internet**", NewRiders Publishing, 1995.
- [19] **Reliable Multicast Protocol (RMP)**.
- [20] S. PAUL; K. K. SABNANI; J. C. LIN. **Reliable Multicast Transport Protocol (RMTP)**. IEEE Journal on Selected Areas in Communications, Vol. 15 No. 3, Abr. 1997, Pags. 407-421.
- [21] LIAO, T. **Light-weight Reliable Multicast Protocol**. INRIA, França.
- [22] SUN Microsystems. **JSDT – Java Shared Data Toolkit**. <http://java.sun.com/products/java-media/jsdt>. Acessado em 2003.
- [23] SUN Microsystems. **Java Technology**. <http://java.sun.com>. Acessado em 2003.
- [24] UCL. **ReLaTe**. <http://www-mice.cs.ucl.ac.uk/multimedia/software/relate>. Acessado em 2001.
- [25] UCL. **UCL MBone Tools**. <http://www-mice.cs.ucl.ac.uk/multimedia/software>. Acessado em 2001.
- [26] UCL. **UCL WB: White Board Tool**. <http://www-mice.cs.ucl.ac.uk/multimedia/software/>. Acessado em 2001.

- [27] **VRVS - Virtual Room Videoconferencing System.**
<http://www.vrvs.org/>. Acessado em 2001.
- [28] **DCSHARE.** <http://www.dataconnection.com/conf/DCshare.htm>. Acessado em 2001.
- [29] **VOX-PHONE** <http://www.voxphone.com>. Acessado em 2001.
- [30] SUN Microsystems. **SUNFORUM**
<http://www.sun.com/desktop/products/software/sunforum>. Acessado em 2001.
- [31] **MINT – Multimedia Internet Terminal.**
<http://www.fokus.gmd.de/research/cc/glone/products/mint>. Acessado em 2002
- [32] Microsoft. **NETMEETING.**
<http://www.microsoft.com/windows/netmeeting>. Acessado em 2002.
- [33] SUN Microsystems. **JMF – Java Media Framework.**
<http://java.sun.com/products/java-media/jmf>. Acessado em 2003.
- [34] **CONFMAN Conference Manager.** <http://www.rvs.uni-hannover.de/products/confman/v2.0>. Acessado em 2002
- [35] **MROUTED.** <ftp://playground.sun.com> <ftp://parcftp.xerox.com/pub/net-research/ipmulti/>. Acessado em 2002.
- [36] **MTUNNEL.** <http://www.cdt.luth.se/~peppar/progs/mTunnel/>. Acessado em 2002.
- [37] MALPANI, R.; ROWE, L. A. **Floor Control for large-scale Mbone Seminars.** ACM Multimedia 1997 – Eletronic Proceedings.
- [38] MCCANNE, S.; JACOBSON, V.; **vic: A Flexible Framework for Packet Video.** ACM 1995.

- [39] JACOBSON, V; MCCANNE, S.; FLOYD, S. **A Conferencing Architecture for Light-Weight Sessions**, Technical Report, Lawrence Berkeley Laboratory, nov. 1993.
- [40] MCCANNE, S.; **Scalable Multimedia Communication with IP Multicast, Lightweight Sessions and Mbone**. University of California, Berkeley. 1999.
- [41] FLOYD, S., JACOBSON, V., LIU, C., MCCANNE, S., and ZHANG, L., **A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing**, IEEE/ACM Transactions on Networking, December 1997, Volume 5, Number 6, pp. 784-803.
- [42] Amir, E, MCCANNE, S., and VETTERLI, M., **A Layered DCT Coder for Internet Video**. Proc IEEE Int'l Conf. Image Processing. Set. 1996.
- [43] MCCANNE, S.; **Scalable Multimedia Communication: Using IP Multicast and Lightweight Sessions**. IEEE *Internet Computing*, mar-abr 1999.
- [44] KUTSCHER, D.; OTT, J. **The Message Bus: A Communication & Integration Infrastructure for Component-based System**. White paper, jan. 1999.
- [45] HANNA, S.; KADANSKY; M, ROSENZWEIG, P.. **The Java Reliable Multicast Service: A reliable Multicast Library**. Palo Alto - CA. Set. 1998.
- [46] DOMMEL, H.-P.; J. J. GARCIA-LUNA-ACEVES. **Comparison of Floor Control Protocols for Collaboration multimedia Environments**.
- [47] BACHMANN, F.; **Levantamento De Aplicações, Ferramentas E Projetos Voltados Para Videoconferência E Trabalho Colaborativo Mediado Por Computador**; NURCAD/UFSC; Fpolis, 2001.
- [48] BACHMANN, F.; **Classificação De Aplicações E Ferramentas Voltadas Para Videoconferência E Trabalho Colaborativo Mediado Por Computador**; NURCAD/UFSC; Fpolis, 2002.