



UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ

CAMPUS LUIZ MENEGHEL



JAILTON DA PAZ

**Análise das Tecnologias para a implantação de uma Rádio
Web utilizando Software Livre**

Bandeirantes

2009

JAILTON DA PAZ

**Análise das Tecnologias para a implantação de uma Radio
Web utilizando Software Livre**

Trabalho de Conclusão de Curso
submetido à Universidade Estadual do
Norte do Paraná – Campus Luiz
Meneghel, como requisito parcial para a
obtenção do grau de Bacharel em
Sistemas de Informação.

Orientador: Prof. Luiz Fernando Legore do
Nascimento.

Bandeirantes

2009

JAILTON DA PAZ

**Análise das Tecnologias para a implantação de uma Radio
Web utilizando Software Livre**

Trabalho de Conclusão de Curso
submetido à Universidade Estadual do
Norte do Paraná – Campus Luiz
Meneghel, como requisito parcial para a
obtenção do grau de Bacharel em
Sistemas de Informação.

COMISSÃO EXAMINADORA

Prof. Luiz Fernando L. Nascimento.
Orientador

Prof. Ms. Ricardo Gonçalves Coelho
Membro da banca examinadora

Prof. Ms. Ederson Marcos Sgarbi
Membro da banca examinadora

Bandeirantes, __ de _____ de 2009

A meus pais e a todas as pessoas que contribuíram de alguma forma para a minha formação acadêmica.

AGRADECIMENTOS

Agradeço primeiramente a Deus pelo que representa pra mim. Ao meu orientador Prof. Luiz Fernando L. Nascimento, pelo apoio durante o desenvolvimento deste trabalho. Aos meus pais pelo apoio e aconselhamento. A minha esposa e filho pela paciência ao longo deste projeto. E por fim, agradeço a todos os meus amigos e colegas que contribuíram de alguma forma para este momento.

RESUMO

A distribuição de conteúdo é uma aplicação típica de redes de computadores. Para uma quantidade grande de usuários, a concentração do tráfego no ponto fornecedor do conteúdo pode tornar-se um fator limitante para a funcionalidade do sistema. Diversos sistemas utilizam redes *peer-to-peer* para distribuição de arquivos na Internet, evitando a concentração do tráfego em um único ponto e melhorando assim a escalabilidade do sistema. Este trabalho propõe um sistema que utiliza redes *peer-to-peer* para auxiliar a distribuição do áudio de uma rádio web. Os diversos usuários estabelecem periodicamente acordos de encaminhamento de partes do fluxo entre si, fazendo assim que o áudio chegue ao maior número de pessoas possíveis.

Palavras-chave: Rádio Web, *peer-to-peer*.

ABSTRACT

The distribution of content is typical application of computer networks. For a large number of users, the concentration of traffic in the point provider of content can become a limiting factor for the functionality of the system. Several systems using peer-to-peer for the distribution of files on the internet, avoiding the concentration of traffic on a unique point and improving the scalability of the system of the system. This work proposes a system that uses peer-to-peer to aid the distribution of audio from a radio website. The several agreements regularly provide users of transferring parts of the flow between them, making than that the audio arrives in a bigger number of people of possible

Key-words: Radio web, Peer-to-peer;

LISTA DE FIGURAS

Figura 1 – Funcionamento do <i>Streaming</i>	18
Figura 2 – <i>Streaming Unicast</i>	20
Figura 3 – <i>Streaming Multicast</i>	21
Figura 4 – Funcionamento de uma rede P2P.....	22
Figura 5 – Funcionamento do servidor de <i>streaming</i>	24
Figura 6 – <i>Icecast</i> em Funcionamento	26
Figura 7 – Tela de configuração do <i>Peercast</i>	28
Figura 8 – Fluxo dos dados.....	31
Figura 9 – Site da rádio	36
Figura 10 – Esquema de funcionamento da Radio Web	38

LISTA DE SIGLAS

HTML	Hypertext Transfer Protocol
KBPS	KiloBits Por Segundo
MP3	MPEG-1/2 Audio Layer 3
P2P	Peer-to-Peer
XML	Extensible Markup Language
VOIP	Voice Over Internet Protocol

SUMÁRIO

1. INTRODUÇÃO	11
1.1 Objetivos.....	12
1.1.1 Objetivo Geral.....	12
1.1.2 Objetivos Específicos	12
1.2 Justificativas	13
1.3 Organização do Trabalho.....	14
2. FUNDAMENTAÇÃO TEÓRICA.....	15
2.1 Radio Web	15
2.1.1 Historia da Radio Web	15
2.1.2 História da Radio Web no Brasil	16
2.2 <i>Streaming</i>	17
2.2.1 Funcionamento do <i>Streaming</i>	18
2.2.2 <i>Streaming</i> de Áudio ao Vivo.....	19
2.2.2.1 <i>Streaming Unicast</i>	19
2.2.2.2 <i>Streaming Multicast</i>	20
2.2.3 <i>Streaming On Demand</i>	21
2.2.4 <i>Streaming P2P</i>	22
2.2.5 Servidores de <i>Streaming</i>	23
2.2.6 Ferramentas a serem utilizadas para o <i>Streaming</i>	23
3. MONTAGEM DO SERVIDOR DE <i>STREAMING</i>	25
3.1 Funcionamento	25
3.2 <i>Ices2</i>	26
3.4 <i>PeerCast</i>	28
3.4.1 Descrição do funcionamento <i>PeerCast</i>	30
3.4.1.1 Servidor	31
3.4.1.2 Cliente.....	33
3.5 Web Site.....	36
3.6 Vantagens da utilização de um servidor <i>PeerCast</i> versus um servidor de streaming por broadcast.....	37
4. METODOLOGIA	38
4.1 Estrutura da Radio Comunitária.....	38

5. CONCLUSÃO	40
REFERÊNCIAS.....	41

1. INTRODUÇÃO

O número de usuários que utilizam a Internet tem crescido ao longo dos anos. O relatório inicial do NIC.br (Núcleo de Informação e Coordenação do Ponto.br) demonstra que tem aumentado o número de usuários que utilizam a Internet em casa, em *lan house*, em cafés e em escolas. A maioria destes usuários tem utilizado a rede mundial de computadores para fins pessoais/privativos e educacionais (CETIC.br, 2006).

Este crescimento da utilização da Internet tem promovido o desenvolvimento e o aprimoramento de diversos serviços, como transações, pagamentos de faturas pela *internet banking*, serviços de telefonia com o VOIP.

Outro serviço que teve um grande crescimento devido à melhora nos serviços de internet é o *Streaming*. O *Streaming* é uma forma de distribuir informação multimídia numa rede através de pacotes. Em *Streaming*, as informações multimídia são reproduzidas na medida em que chega ao cliente não havendo a necessidade de aguardar essas informações serem transmitidas por inteiro, pois até pouco tempo atrás, quando se falava de áudio pela internet logo se pensava em muita lentidão, havendo a necessidade em se fazer o *download* deste arquivo para em seguida reproduzi-lo.

Com o surgimento do *streaming*, outra inovação que vem sendo muito utilizada é a rádio web, que nada mais é do que a transmissão do áudio de uma rádio convencional via internet.

ZAREMBA (1999:13) adverte sobre o rádio na Internet, afirmando ser necessário inicialmente conhecer seus modelos, conceitos, linguagem, para que possamos usufruir desse novo instrumental, sendo que a ameaça de morte do rádio é simplesmente seu desconhecimento.

1.1 Objetivos

1.1.1 Objetivo Geral

O trabalho a ser realizado tem como objetivo a montagem de um servidor de *streaming* para uma rádio comunitária fazendo o uso da tecnologia *peer-to-peer*, identificando os fatores que constituem obstáculos no processo de criação e distribuição de conteúdos de áudio. E juntamente com o estudo desenvolver uma rádio web.

1.1.2 Objetivos Específicos

- Estudo das ferramentas envolvidas para implantação de um servidor de *streaming*.
- Elaboração do website para a difusão do áudio.
- Montagem de um servidor de *streaming*.

1.2 Justificativas

A Rádio Comunitária de Santa Amélia - PR é o meio de comunicação responsável pelo processamento, pela edição e disseminação das informações relativas ao município. Essa rádio além de um ótimo veículo de entretenimento é ainda responsável pela cobertura jornalística, divulgação de utilidade pública e dos eventos que ocorrem na cidade.

Por essa razão, visivelmente tangível, uma Rádio comunitária apresenta-se como um excelente meio de divulgação. Operando atualmente em frequência modulada (FM), pretende-se ainda ampliar a área de cobertura através de uma rádio web.

O trabalho desenvolvido poderá servir de base para pessoas que tiverem o interesse em fazer a montagem de uma rádio web tendo um orçamento relativamente baixo.

1.3 Organização do Trabalho

Este trabalho apresenta uma descrição relacionada à rádio web e o streaming sendo vista no capítulo 2. O capítulo 3 apresenta a montagem do servidor de *streaming* além do estudo das ferramentas envolvidas no projeto. Já no capítulo 4 são apresentadas as ferramentas e metodologias empregadas no desenvolvimento deste do trabalho. E por fim no capítulo 5 são apresentados os resultados e as conclusões do trabalho.

2. FUNDAMENTAÇÃO TEÓRICA

2.1 Radio Web

Rádios Web são emissoras de rádio transmitidas via internet gerando áudio em tempo real, bem como disponibiliza entrevistas e outros entretenimentos anteriormente confeccionados pela rádio. Hoje emissoras comerciais usam essa tecnologia para emitir sua programação também pela Internet.

O que possibilitou a entrada do rádio na internet foi justamente à criação de uma nova tecnologia que possibilita os usuários poder reproduzir o áudio sem a necessidade do download completo desse arquivo, sendo essa tecnologia denominada *Streaming*.

2.1.1 Historia da Radio Web

A rádio é um meio de comunicação muito rico, um meio que ao longo da sua história soube adaptar de forma rápida as mudanças, aos novos desenvolvimentos tecnológicos.

A origem das transmissões áudio via Internet remonta a março de 1993, quando Carl Malamud que liderava a *Columbus Internet Engineering Talk Force* criou um *talk show* semanal chamado Internet Talk Rádio. O modelo utilizado por Carl aproxima-se do que hoje denominamos de *podcasting*, um programa que o ouvinte tinha de descarregar todo o ficheiro para o seu computador para o poder reproduzir. Mas só a partir do ano 1996 que este conceito ganhou um desenvolvimento significativo.

Segundo MOREIRA (2002), duas companhias americanas reclamam os direitos e reconhecimento histórico de serem as emissoras pioneiras na transmissão de suas programações via Internet, a AudioNet, que em setembro de 1995 teria disponibilizado pela Internet a rádio KLIF, de Dallas, no Texas, e a *Broadcast.com*, que seria uma das primeiras a operar via Internet através de um portal específico de áudio e vídeo, tendo realizado a primeira transmissão de rádio também em setembro de 1995.

Também na segunda metade da década de 90, algumas rádios

inglesas já experimentavam as novas possibilidades com a transmissão instantânea de áudio via internet. Uma das primeiras emissoras inglesas a operar via internet foi a InterFace, disponibilizando uma versão online da programação musical muito comum às rádios piratas.

Ainda de acordo com MOREIRA (2002), essas emissoras pioneiras na transmissão via internet estavam equipadas com computadores Pentium 100 MHz, com 32 MB de memória RAM no provedor, que codificava toda a programação para depois distribuí-la aos internautas. A transmissão de áudio evoluiu mais rapidamente na internet por exigir menos recursos técnicos, e ter sido beneficiada com a rápida criação de programas voltados para a reprodução de áudio na rede mundial de computadores.

2.1.2 História da Radio Web no Brasil

Segundo CUNHA (2003), no Brasil, a rádio iniciou sua penetração, no novo meio de transmissão, somente dois anos após as primeiras experiências das emissoras norte-americanas, ou seja, apenas a partir de 1997.

Ainda de acordo com CUNHA (2003), A primeira rádio a operar via internet, no Brasil, foi a radio Manguetronic, que em abril de 1996 veiculava pela primeira vez um programa exclusivamente pela web. O programa foi inicialmente criado por dois integrantes do movimento pernambucano denominado “Mangue Beat”: José Carlos Arco Verde, mais conhecido como HB Mabuse e Renato Lins. Os dois, então, somaram seus conhecimentos para a criação de um site que divulgasse todas as informações do movimento. Em maio de 1995 eles começaram as primeiras experiências para criação do site www.manguetronic.com.br . Mas, devido a deficiências tecnológicas do momento, somente em abril de 1996, eles conseguiram desenvolver o programa de rádio especialmente destinado para o acesso de usuários da grande rede.

2.2 Streaming

Segundo AVILA (2008), *streaming* é uma forma de distribuir informação multimídia numa rede através de pacotes. Ela é frequentemente utilizada para distribuir conteúdo multimídia através da Internet. Em *streaming*, as informações da mídia não são usualmente arquivadas pelo usuário que está recebendo a *stream* (a não ser a arquivação temporária no cache do sistema ou que o usuário ativamente faça a gravação dos dados) - a mídia geralmente é constantemente reproduzida à medida que chega ao usuário se a sua banda for suficiente para reproduzir a mídia em tempo real. Isso permite que um usuário reproduza mídia protegida por direitos autorais na Internet sem a violação dos direitos, similar ao rádio ou televisão aberta.

A tecnologia *Streaming* foi lançada pela Real Áudio em 1995. A primeira transmissão digital via *streaming* ocorreu no ano de 1988 utilizando a tecnologia *Shoutcast* com base no MP3 comprimido à 128kbps (qualidade de CD), porém, a internet ainda não era capaz de suportar transmissões de elevada velocidade e o projeto foi engavetado (LIEB, 2001).

Nesta técnica o sinal de áudio é transmitido ao cliente e sua apresentação inicia-se após uma momentânea espera para armazenamento dos dados em um *buffer*. Nesta forma de transmitir áudio não é preciso fazer o download prévio do arquivo, o micro vai recebendo as informações continuamente enquanto mostra ao usuário. Esta técnica reduz o tempo de início da exibição e também elimina a necessidade de armazenamento local do arquivo. Com isto, existe a sensação de que o áudio é transmitido ao vivo, dando a impressão que está realmente o rádio ligado. Quando se fala em sensação, está se referindo a alguma latência de alguns milissegundos ou segundos que pode haver entre a rede que está o servidor até chegar a sua rede onde está o seu computador.

2.2.1 Funcionamento do *Streaming*

Streaming é a tecnologia que permite tocar conteúdos multimídia no momento em que se inicia o recebimento dos dados. Este conceito difere do *download* onde é preciso esperar o arquivo ser transferido para o computador para que então possa ser acessado. Meios de entretenimento e informação utilizam uma forma de *streaming* como televisão e rádio (TOPIC, 2002).

AVILA (2008) afirma que o *streaming* funciona da seguinte maneira, primeiro computador (o cliente) conecta-se com o servidor, e este começa a lhe enviar o arquivo dividido em pacotes. O cliente começa a recebê-los e constrói um *buffer* onde começa a salvar a informação. Quando se enche o *buffer* com uma pequena parte do arquivo, o cliente começa a mostrar e ao mesmo tempo continua o *download*. O sistema está sincronizado para que o arquivo possa ser visto enquanto o restante do arquivo é transferido, de modo que quando o arquivo acaba de ser transferido, também acaba de ser visualizado. Se em algum momento a conexão sofre decréscimos de velocidade se utiliza a informação que existe no *buffer*, de modo que se pode agüentar um pouco esse decréscimo. Se a comunicação se corta durante muito tempo, o *buffer* se esvazia e a execução do arquivo se cortaria também até que se restaurasse o sinal.

A figura 1 ilustra o funcionamento do *streaming*, quando o mesmo os envia para os clientes.

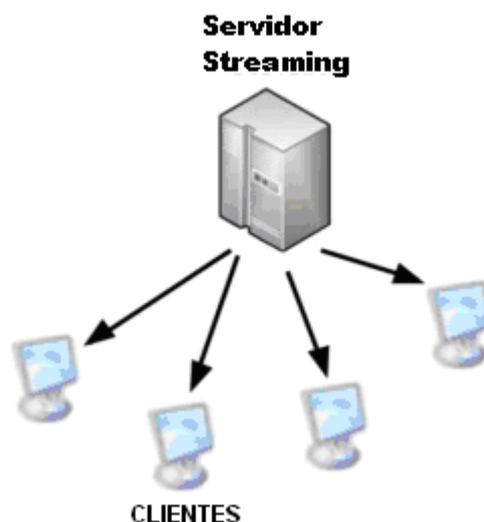


Figura 1 – (Funcionamento do *Streaming*)

2.2.2 Streaming de Áudio ao Vivo

Este tipo de aplicação é conhecida como *broadcast*, onde o cliente assume uma posição passiva e não controla quando o stream começa ou termina. A única diferença está no fato de está transmissão ser feita através da Internet. Tais aplicações permitem ao usuário receber um sinal de rádio ou televisão ao vivo que foi emitido de qualquer parte do mundo. Neste caso, como o *streaming* não é armazenado em um servidor (KUROSE, 2004).

Segundo ESTACIO (2000), neste tipo de transmissão podem existir muitos clientes recebendo o mesmo conteúdo simultaneamente com a sua distribuição podendo ocorrer de duas formas: *Streaming Unicast* ou *Streaming Multicast*.

2.2.2.1 Streaming Unicast

Unicast: é uma conexão ponto-a-ponto entre o cliente e o servidor, em que cada cliente recebe seu próprio *stream* do servidor. Dessa forma, cada usuário conectado ao *stream* tem sua própria conexão e os dados vêm diretamente do servidor (KUMAR, 1997).

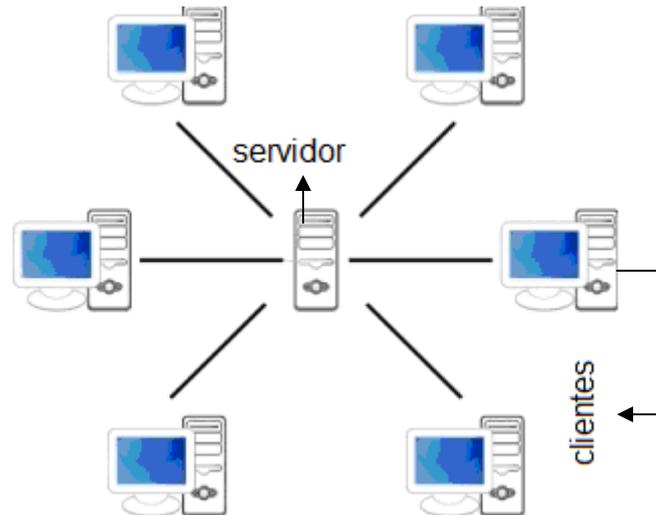


Figura 2 – *Streaming Unicast*

A figura 2 ilustra o funcionamento de um *Streaming Unicast*, onde o cliente requisita dados do servidor e o mesmo envia um *stream* para cada cliente.

2.2.2.2 Streaming Multicast

Multicast: ocorre quando o conteúdo é transmitido sobre uma rede com suporte à *multicast*, em que todos os clientes na rede compartilham o mesmo *stream*. Assim, preserva-se largura de banda, podendo ser extremamente útil para redes locais com baixa largura de banda (ESTACIO, 2000).

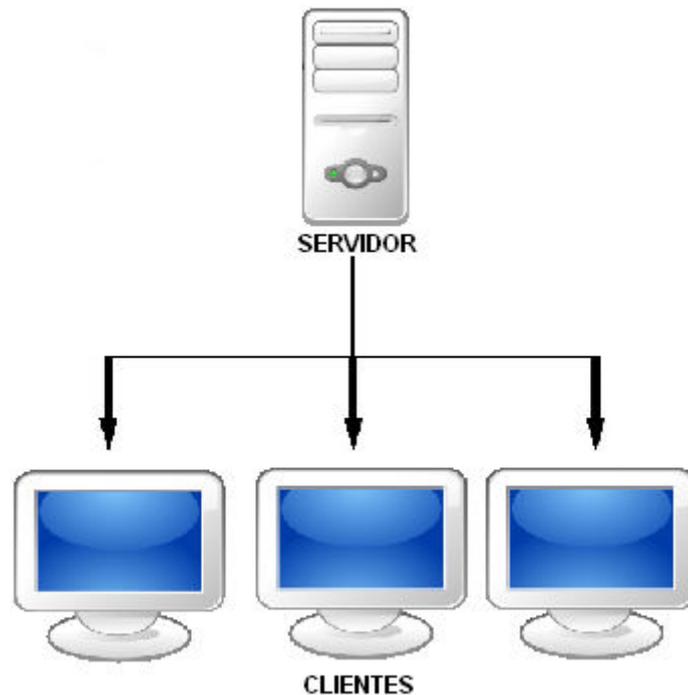


Figura 3 – *Streaming Multicast*

A figura 3 ilustra o funcionamento de uma rede *multicast*, onde o servidor envia um *streaming* que é compartilhado entre todos os clientes.

2.2.3 Streaming On Demand

Streaming on demand trata-se da tecnologia que o usuário utiliza para fazer o *download* de um arquivo digital (fala, música, vinhetas) para executá-lo no microcomputador (Bufarah, 2002).

As transmissões *on demand* proporcionam uma interatividade que só passou a existir com a chegada da Internet. Estes são arquivos gravados que o usuário pode acessar, via *streaming*, a hora que quiser e quantas vezes quiserem desde que esteja disponível no site desejado.

2.2.4 Streaming P2P

Segundo Castro (2008) P2P é um tipo de rede distribuída ou rede não hierárquica é uma topologia de redes caracterizada pela descentralização das funções na rede, onde cada terminal realiza tanto funções de servidor quanto de cliente.

Em um *streaming* P2P, o conteúdo é compartilhado em tempo real entre os vários clientes que passam a ser ao mesmo tempo clientes e servidores dentro da rede P2P. Desta forma, o conteúdo de áudio ou vídeo é transferido entre os clientes, tornando possível expandir a rede a um número praticamente ilimitado de clientes.

As aplicações que mais se beneficiam da estrutura que utiliza redes *peer-to-peer* são aquelas em que o conteúdo representa um grande volume de dados, especialmente a distribuição de grandes arquivos e fluxos multimídia, como transmissão de áudio e vídeo.

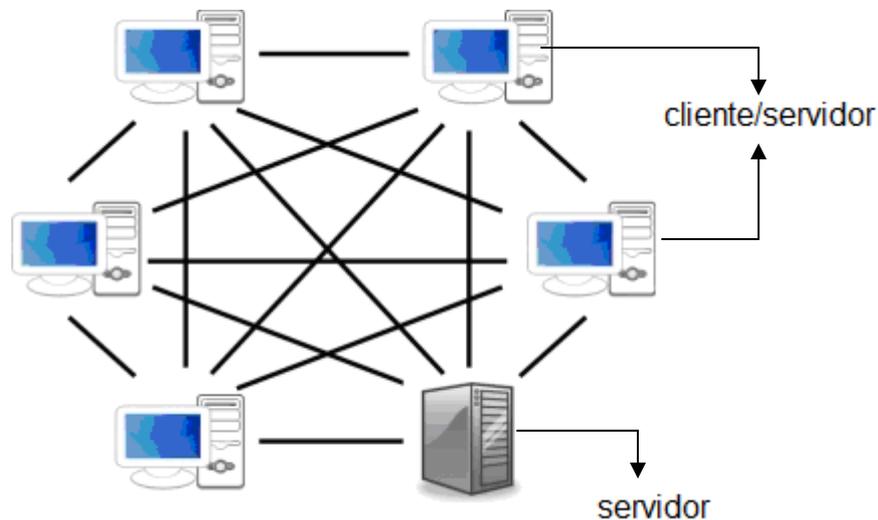


Figura 4 – Funcionamento de uma rede P2P

A figura 4 ilustra o funcionamento de um *streaming* P2P onde o servidor transmite os dados para alguns clientes, esses clientes fazem o papel de

servidor enviando os dados que já foram recebidos outros clientes se tornando ao mesmo tempo cliente e servidor.

2.2.5 Servidores de *Streaming*

Segundo AVILA (2008) a princípio não é necessário contar com um servidor especial para colocar arquivos de áudio ou vídeo com *streaming* em nossas webs. Qualquer servidor normal pode enviar a informação é o cliente quem se encarrega de processá-la para poder mostrá-la na medida em que for recebendo. Entretanto, existem servidores especiais preparados para transmitir *streaming*. Embora em muitas ocasiões não seja necessário utilizá-los, estes servidores podem nos oferecer importantes prestações, como enviar um arquivo de maior ou menor qualidade dependendo da velocidade de nossa linha.

Em determinados casos, que será imprescindível contar com um servidor de *streaming* ao que enviar o sinal os transmitira a todos os clientes à medida que os recebe.

2.2.6 Ferramentas a serem utilizadas para o *Streaming*.

A seguir são descritas algumas ferramentas que auxiliam no processo de transmissão do áudio via internet. A função de cada uma delas varia de acordo com o seu funcionamento.

- *Icecast*: Sistema *multicast* responsável por receber o áudio e o distribuir para os seus receptores.

- *Ices2*: Sistema responsável pela captação do áudio lançado pela placa de som e por enviá-lo ao sistema que está fazendo o *streaming*.
- *Peercast*: Sistema que utiliza a tecnologia P2P responsável pela transmissão do áudio gerado pelo *Streaming*.

3. MONTAGEM DO SERVIDOR DE *STREAMING*

3.1 Funcionamento

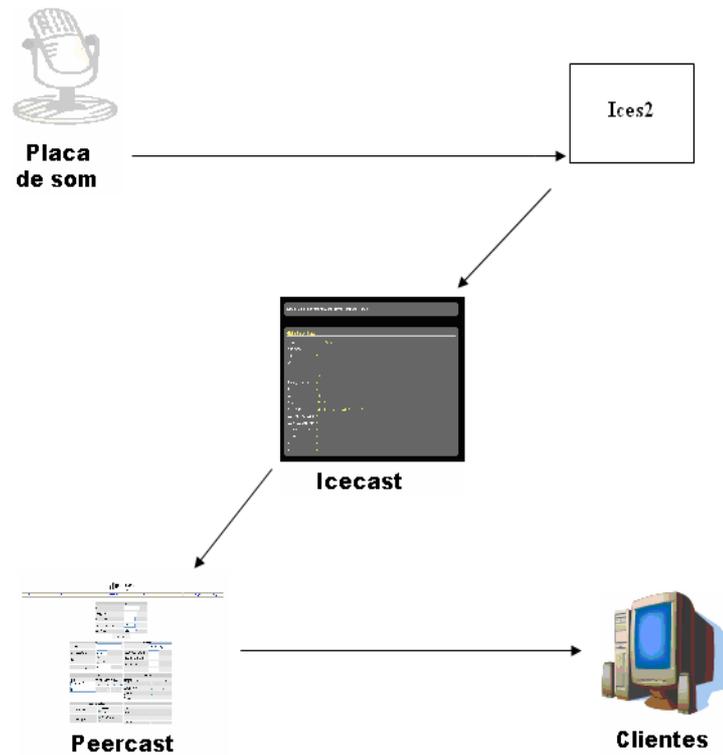


Figura 5 – Funcionamento do Servidor de *Streaming*

A figura 5 ilustra o funcionamento do servidor sendo que o mesmo utiliza sistema operacional Ubuntu 9.04 com kernel 2.6.28-11, sendo necessário o uso *Ices2*, o qual seu dever é capturar o áudio da placa de som e o enviar ao *icecast*, este por sua vez tem o papel de fazer o *streaming* do áudio. E por fim o *peercast* ira capturar este áudio enviar aos clientes.

3.2 Ices2

Mantido pela Xiph.org, o Ices2 trabalha com a transmissão de áudio no formato livre Ogg Vorbis. (ICECAST, 2008).

O objetivo do Ices2 é proporcionar um fluxo de áudio para um servidor de *streaming* de tal ordem que um ou mais ouvintes podem acessar o fluxo.

Sua função é capturar o áudio da placa de som e o enviar para o software responsável pelo *streaming*, neste caso o Icecast. Toda sua configuração é feita através de um arquivo XML este arquivo tem como padrão a porta 8000 para a transmissão, onde se deve alterar para a porta 7145 do icecast, pois esta é a porta usada pelo peercast para capturar esse áudio e assim fazer o streaming.

3.3 Icecast

Mantido pela Xiph.org, Icecast foi inicialmente utilizado para construir estações de rádio realizando *streaming* somente de áudio nos formatos proprietário MP3 e livre Ogg Vorbis.

Este servidor de *streaming* atualmente, na sua versão 2.3.2, é distribuído debaixo da licença GNU GPL, versão 2, possui em sua instalação o requerimento das bibliotecas libxml2 (XML *parser library*), libxslt (XML *transformation library*), (ICECAST, 2008).

O icecast deverá transmitir como *localhost*, ou seja, para o próprio servidor, como pode ser visto na figura 6:

The screenshot shows the 'Global Server Stats' page of the Icecast web interface. At the top, there is a navigation bar with links: 'Admin Home', 'List Mountpoints', 'Move Listeners', and 'Index'. Below this, the 'Global Server Stats' section displays a list of server metrics and their current values.

Stat	Value
admin	icemaster@localhost
client_connections	11
clients	0
connections	11
file_connections	6
host	localhost
listener_connections	0
listeners	0
location	Earth
server_id	Icecast 2.3.2
server_start	Thu, 28 May 2009 03:23:55 Hora oficial do Brasil
source_client_connections	0
source_relay_connections	0
source_total_connections	0
sources	0
stats	0
stats_connections	0

Figura 6 – Icecast em funcionamento

Toda configuração do icecast é feita através de um arquivo XML localizada dentro da pasta onde foi este foi instalado, e os principais configurações são:

- *source-password* – mesma senha a ser definida no Ices2;
- *admin-password* – senha para acesso a características;
- *hostname* e *port* – qual endereço e porta irá escutar;
- *logdir* – diretório dos arquivos *log* (*/icecast/log*);
- *webroot* – diretório dos arquivos a serem transmitidos (*/icecast/web*);
- *adminroot* – diretório contendo arquivos administrativos;

O *icecast* tem em sua configuração padrão a porta 8000 para transmissão, assim como no *Ices2* deve-se alterar a porta de transmissão para 7145 fazendo assim com que o *peercast* capture esse áudio e o transmita a seus clientes.

3.4 Peercast

Peercast (2006), este é um software *open source* que utiliza a tecnologia *peer-to-peer*, uma única árvore *multicast* para distribuir conteúdo entre nós conectados. A abordagem utilizada para determinar o posicionamento de cada nó na árvore é bastante simples. Quando um novo nó deseja integrar-se à árvore, ele envia uma solicitação de inclusão para algum nó que, possuindo disponibilidade de recursos para atendê-lo, adiciona-o como filho na árvore. Caso o nó não possua recursos disponíveis, ele redireciona a solicitação para um de seus filhos. Esta abordagem pode levar a construção de árvores desbalanceadas.

O servidor é responsável por produzir o fluxo de dados que será transmitido para os clientes e por ajudar os clientes a encontrarem uns aos outros, além de transmitir uma quantidade suficiente de cópias do fluxo para os clientes de forma que estes possam trocar as partes recebidas entre si e obter o fluxo completo.

O cliente deve instalar um *plugin* do *peercast* para assim receber o áudio e também enviar para outros clientes. Quando um cliente inicia, ele conecta-se ao servidor. Ao receber a conexão, o servidor adiciona o novo cliente a uma lista de clientes ativos e envia para o cliente dados sobre o fluxo que é transmitido. Estes dados são usados pelo cliente para configurar seu módulo consumidor de mídia e incluem, por exemplo, a taxa em que as fatias são geradas e o tamanho de cada fatia.

Ao instalar o *peercast*, todas as suas configurações são feitas através de uma página HTML, acessada pelo endereço “<http://localhost:7144/html/en/connections.html>”, como pode ser visto na figura 7:

Basic Settings	
Port	7144
Password	
Max. Relays	2
Max. Direct streams	1200
Language	English
Save Settings	

Server	Network
DJ Message	YP Address: yp.pearcast.org
ICY MetaInterval: 8192	Max. Output (Kbits/s): 0
Mode: <input checked="" type="radio"/> Normal <input type="radio"/> Root	Max. Relays Per Channel: 0
Refresh HTML (sec): 5	Max. Controls In: 3
	Max. Connections In: 50

Filters	Security																														
<table border="1"> <thead> <tr> <th>IP Mask</th> <th>Network</th> <th>Direct</th> <th>Private</th> <th>Ban</th> </tr> </thead> <tbody> <tr> <td>255.255.255.255</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>0.0.0.0</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> </tbody> </table>	IP Mask	Network	Direct	Private	Ban	255.255.255.255	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0.0.0.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<table border="1"> <thead> <tr> <th>Allow on port:</th> <th>7144</th> <th>7145</th> </tr> </thead> <tbody> <tr> <td>HTML</td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>Broadcasting</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>Network</td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> <tr> <td>Direct</td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> </tbody> </table>	Allow on port:	7144	7145	HTML	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Broadcasting	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Network	<input checked="" type="checkbox"/>		Direct	<input checked="" type="checkbox"/>	
IP Mask	Network	Direct	Private	Ban																											
255.255.255.255	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																											
0.0.0.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																											
Allow on port:	7144	7145																													
HTML	<input checked="" type="checkbox"/>	<input type="checkbox"/>																													
Broadcasting	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																													
Network	<input checked="" type="checkbox"/>																														
Direct	<input checked="" type="checkbox"/>																														

Authentication	Log
HTML Authentication: <input checked="" type="radio"/> Cookies <input type="radio"/> Basic HTTP	Debug: <input type="checkbox"/>
Cookies Expire: <input checked="" type="radio"/> End of session <input type="radio"/> Never	Errors: <input type="checkbox"/>
	Network: <input type="checkbox"/>
	Channels: <input type="checkbox"/>

Figura 7 – Tela de configuração do *Peercast*

A figura 7 apresenta a tela de configuração do *peercast*, onde os campos que devem ser configurados são:

- Port: indica a porta de transmissão que o *peercast* está usando, este vem como padrão a porta 7144.
- Password: indica a senha para a configuração do *peercast*, vem como padrão em branco.
- Max. Direct streams: indica quantidade de usuários que poderão se conectar a rádio.

3.4.1 Descrição do funcionamento *Peer*cast

O *peer*cast é composto por um servidor e um conjunto de clientes. O servidor é responsável por produzir o fluxo de dados que será transmitido para os clientes e por ajudar os clientes a encontrarem uns aos outros.

Segundo Mello (2003), os clientes são responsáveis por obter e executar o fluxo multimídia para o usuário e servirem partes do fluxo para outros clientes, formando uma rede *peer-to-peer*. Cada cliente conecta-se a um grupo de outros clientes para tentar obter partes do fluxo e fornecer partes que possua. Caso o cliente não consiga, por algum motivo, obter alguma parte do fluxo de outros nós o servidor fornecerá essa parte, evitando que o usuário perceba interrupções na execução do fluxo multimídia.

Ainda de acordo com Mello (2003), o fluxo multimídia é dividido em fatias de tamanho fixo e numerado sequencialmente. As fatias são produzidas e consumidas a uma taxa constante. Após produzir cada fatia do fluxo, o servidor divide esta em *blocos* de mesmo tamanho que são enviados para um determinado número de clientes. Cada bloco da fatia também é numerado através de um índice que indica sua posição dentro da fatia.

Os clientes encaminham os blocos recebidos para outros nós cumprindo acordos de encaminhamento estabelecidos. Cada acordo de encaminhamento é referente a blocos com um mesmo número de bloco e é válido por uma determinada quantidade de fatias do fluxo, sendo renegociados após este período. Todos os acordos de encaminhamento estabelecidos no sistema possuem a mesma duração e iniciam em fatias cujo número sequencial é múltiplo da duração. Caso algum cliente não consiga, por qualquer motivo, estabelecer acordos para receber alguma parte do fluxo, ele pode estabelecer acordos diretamente com o servidor (PEERCAST, 2006).

A figura 8 mostra o fluxo dos dados para a transmissão e uma fatia do fluxo. No servidor, o módulo produtor de mídia cria o fluxo e divide-o em fatias. Cada fatia recebe uma numeração sequencial e é então dividida em certa quantidade de blocos. No exemplo da figura, cada fatia é dividida em quatro blocos. Os blocos são transmitidos pela rede até chegarem ao cliente. Esta transmissão pode ocorrer diretamente entre o servidor e o cliente ou utilizando a rede *peer-to-peer* formada pelos demais clientes. O cliente ordena os blocos recebidos

reconstruindo a fatia do fluxo que é então executada para o usuário pelo módulo consumidor de mídia.

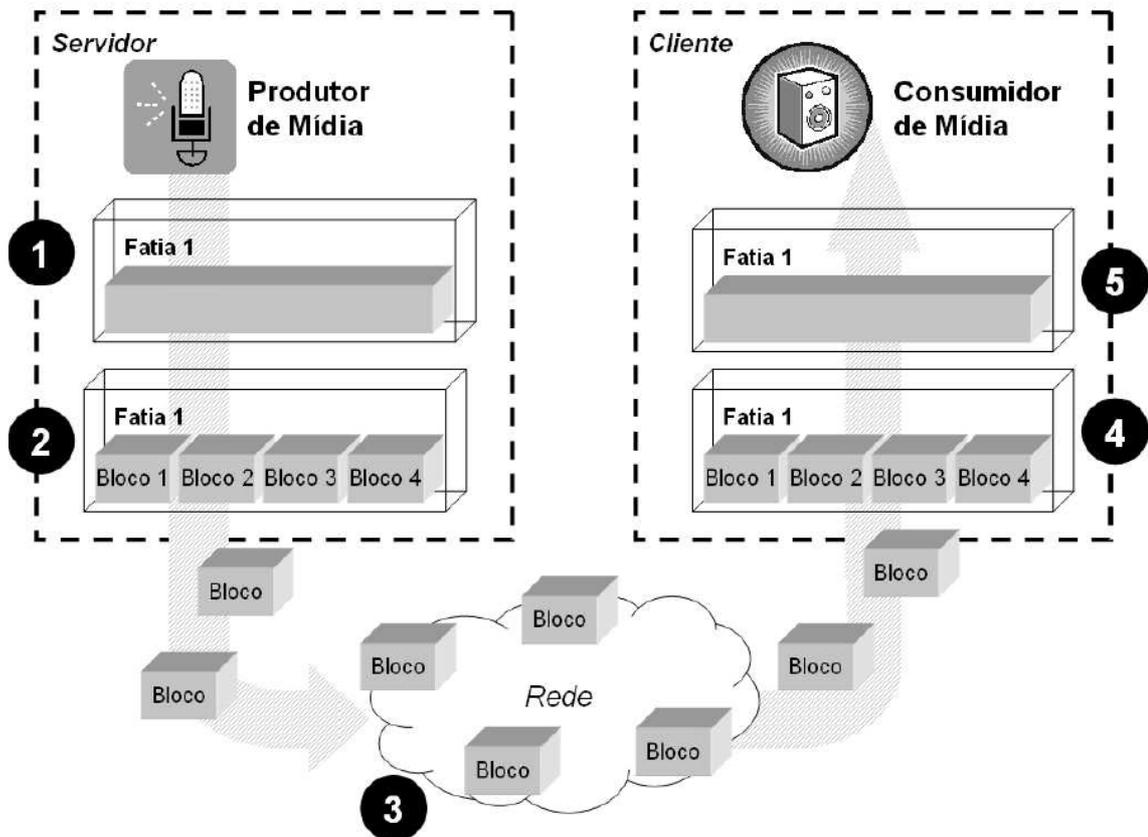


Figura 8 – fluxo dos dados

Os mecanismos de cada componente do sistema serão descritos com detalhes a seguir.

3.4.1.1 Servidor

De acordo com Mello (2007), o servidor realiza tarefas de produção de mídia, gerenciamento de clientes e encaminhamento inicial dos blocos para alguns clientes. O módulo produtor de mídia cria as fatias do fluxo a serem enviadas para os clientes. As fatias criadas pelo produtor de mídia são divididas em blocos que são enviados aos clientes e em seguida armazenados em um buffer para atender eventuais requisições de reenvio de blocos recebidas dos clientes. Este buffer mantém as cópias dos blocos enviados por, pelo menos, a duração de um acordo de encaminhamento.

Segundo Peercast (2006), quando um cliente inicia, ele conecta-se ao servidor. Ao receber a conexão, o servidor adiciona o novo cliente a uma lista de clientes ativos e envia para os clientes dados sobre o fluxo que é transmitido. Estes dados são usados pelo cliente para configurar seu módulo consumidor de mídia e incluem, por exemplo, a taxa em que as fatias são geradas e o tamanho de cada fatia.

A lista de clientes conectados é usada para ajudar os clientes a encontrarem uns aos outros. Os clientes enviam para o servidor requisições de informações sobre outros clientes na rede e o servidor responde com um determinado número de clientes da lista de clientes conectados escolhidos aleatoriamente (ORAM, 2001).

O servidor mantém uma lista de acordos de encaminhamento de blocos. Após a criação de uma nova fatia do fluxo, o servidor encaminha os blocos para os clientes nesta lista. Os acordos são criados atendendo as solicitações dos clientes ou espontaneamente pelo servidor. Em ambos os casos os clientes recebem notificações de que os acordos foram criados.

Segundo Oram (2001), os clientes requisitam acordos de encaminhamento com o servidor quando não conseguem estabelecer acordos com nenhum outro cliente. O servidor cria acordos de encaminhamento espontaneamente em dois casos. O primeiro é quando um cliente acaba de conectar-se e ainda precisará de algum tempo para conseguir conectar-se a outros clientes e estabelecer acordos de encaminhamento. Neste caso, o servidor espontaneamente envia o fluxo para o cliente durante o período de um acordo.

Ainda segundo Oram (2001), o outro caso está relacionado com a disponibilidade de blocos na rede, em que o servidor precisa enviar cada fatia do fluxo para a rede ao menos uma vez para que os clientes sejam capazes reconstruir a fatia completa para execução. Quanto mais vezes o servidor transmitir espontaneamente cada fatia para a rede, mais fácil será para os clientes encontrarem outro cliente que possua acordo para receber os blocos desejados e que possa encaminhar. Desta forma, o servidor escolhe uma determinada fração dos clientes conectados para receber estes acordos espontâneos. Esta fração e a quantidade de cópias da fatia completa que são enviadas para estes clientes. Os clientes são escolhidos de acordo com a ordem crescente do tempo de ir-e-vir (*RTT*, *Round Trip Time*). Após criar espontaneamente acordos para estes nós o servidor

notifica todos os nós conectados de que acordos para uma nova série de fatias já estão disponíveis na rede e os clientes podem começar a estabelecer acordos entre si.

3.4.1.2 Cliente

O cliente é o componente do sistema responsável por obter o fluxo da rede, apresentá-lo ao usuário e servi-lo para outros clientes.

Segundo Mello (2007), após iniciar, cada cliente registra-se no servidor e torna-se disponível para receber conexões de outros clientes, atuando como um *peer* da rede de distribuição. Como resposta ao registro do cliente o servidor envia informações sobre o fluxo transmitido, como a taxa em que as fatias devem ser consumidas, o tamanho de cada bloco e a quantidade de blocos que forma cada fatia.

O cliente usa estas informações para configurar o módulo consumidor de mídia. Este módulo é responsável por ordenar os blocos que foram recebidos da rede, reconstruindo as fatias do fluxo para que sejam executadas para o usuário. No início do processo o módulo consumidor permanece ocioso por um período de tempo para que um conjunto inicial de dados seja recebido da rede.

Algum tempo antes de consumir cada fatia, o módulo consumidor verifica se todos os blocos desta fatia já foram recebidos da rede. Caso algum bloco ainda não tenha sido recebido, o cliente envia uma requisição emergencial para o servidor solicitando apenas os blocos faltantes. O servidor usará o buffer de reenvio de blocos para atender esta requisição. Este é um procedimento emergencial que não deve ser executado frequentemente (MELLO, 2007).

Os clientes mantém listas dos acordos estabelecidos para recebimento e encaminhamento de blocos. Todos os acordos estabelecidos no sistema possuem uma mesma duração, medida em número de fatias.. A fatia em que cada acordo inicia é chamada *fatia-base* do acordo. Cada acordo refere-se ao encaminhamento de todos os blocos com mesmo índice de bloco, de fatias com numeração sequencial entre a fatia-base do acordo e o próximo múltiplo da duração dos acordos.

De acordo com Mello (2007), num fluxo em que a duração dos

acordos é de 10 fatias o primeiro acordo possui fatia-base 0 e refere-se as fatias de 0 a 9, o segundo possui fatia-base 10 e refere-se as fatias 10 a 19 e assim consecutivamente.

A lista de acordos de recebimento indica os acordos onde outros nós encaminham blocos para o cliente. Quando o cliente recebe uma solicitação de acordo para encaminhamento referente a um determinado índice de bloco e fatia base, ele primeiro verifica na lista de acordos de recebimento se já possui um acordo estabelecido para receber esses dados. Os clientes só se comprometem a encaminhar dados para os quais já possuam acordo de recebimento.

Para cada acordo são armazenadas informações sobre o outro nó do acordo, que pode ser outro cliente ou o servidor, o índice de bloco, a fatia base a que o acordo se refere, uma lista com os clientes por onde os dados passam até chegar nó destino e a estimativa do tempo necessário para transferir um bloco desde o nó anterior na lista. Com os dados dessa lista é possível ter uma estimativa de quanto tempo decorre entre a disponibilização de um novo bloco pelo servidor e o recebimento pelo cliente (PEERCAST, 2007).

Cada cliente mantém também uma lista de outros clientes conhecidos.

Logo após o registro no servidor o cliente solicita informações sobre outros clientes. Periodicamente todos os outros clientes são monitorados e durante esse monitoramento é medido o tempo necessário para transmitir um bloco e trocadas informações sobre o tempo estimado para recebimento de cada bloco.

Esta lista de outros clientes é usada durante a negociação dos acordos.

Segundo D. A e K. A (2003), após o servidor enviar espontaneamente acordos para um conjunto de clientes, todos os clientes recebem uma notificações para iniciarem a negociação de acordos para a nova série de fatias iniciada na próxima fatia-base. Ao receber esta notificação o cliente cria uma lista de clientes para cada índice de bloco que precisa negociar acordo, ordenados pelo tempo estimado de recebimento dos blocos com aquele índice nos acordos anteriores. Além disso, é iniciado um temporizador para finalizar a negociação dos acordos.

O cliente inicia a negociação de acordo para cada bloco enviando uma solicitação para o primeiro cliente da lista construída para aquele bloco. Após

enviar a requisição de acordo para o cliente, ele é removido do início da lista e inserido novamente no final desta. Caso o outro cliente aceite encaminhar os blocos solicitados é encerrada a negociação para estes blocos e uma nova entrada é inserida na lista de acordos de recebimento. Se o outro cliente rejeitar a solicitação de acordo, é enviada uma requisição para o próximo cliente da lista. Os nós para os quais já se tentou estabelecer acordo são inseridos novamente no final da lista para que, caso não se consiga firmar acordo com nenhum outro cliente conhecido, sejam tentados novamente. Como os nós somente aceitam encaminhar blocos para os quais possuam acordo de recebimento, este tempo entre as tentativas pode ser suficiente para que o outro cliente consiga estabelecer um acordo para o bloco desejado (MELLO, 2007).

Este processo se repete até que o temporizador para finalizar a negociação expire. Neste momento o cliente verifica se foram estabelecidos acordos para todos os blocos. Caso não tenha sido possível estabelecer acordo para algum bloco uma requisição de acordo é enviada para o servidor.

Peercast (2007), além de estabelecer o acordo com o servidor, o cliente também solicita informações sobre outros clientes, já que com os clientes aos quais se está conectado não foi possível receber todas as partes do fluxo. Assim, nas próximas negociações existirá um número maior de outros clientes para se tentar estabelecer acordo, aumentando a probabilidade de sucesso. Esta medida também beneficia indiretamente os outros clientes que estão conectados a este porque terão maior probabilidade de sucesso no estabelecimento de acordos para receber dados a partir deste cliente.

Além da lista de acordos de recebimento, o cliente mantém também uma lista de acordos de encaminhamento, que indica para quais outros clientes cada bloco recebido deve ser encaminhado.

Segundo Mello (2007), ao receber uma requisição de acordo, o cliente verifica inicialmente se possui acordo de recebimento para os dados solicitados. Caso possua, o tamanho da lista de acordos de encaminhamento é verificado. Cada cliente deve manter pelo menos um determinado número de acordos de encaminhamento. Se a lista de acordos de encaminhamento for menor do que este número o novo acordo é aceito. Caso contrário é rejeitado.

3.5 Web Site

Para a conclusão do projeto, foi criado site e HTML para divulgação do áudio através do *peercast*, que se encontra no endereço <http://www.santaameliafm.xpg.com.br>. O site encontra-se hospedado no servidor de páginas web gratuito chamado xpg. O site conta ainda com o *link* para acesso ao áudio da radio e também o *link* para *download* do *plugin peercast* para que se possa ter acesso a esse áudio, como se pode observar na figura 8:



Figura 9 – Site da rádio

3.6 Vantagens da utilização de um servidor *Peercast* versus um servidor de *streaming* por *broadcast*

Em caso de transmissões de rádio web em *streaming* por *broadcast*, e para um número grande de usuários, será imprescindível o uso de um serviço de *streaming* que tem um custo relativamente alto. Este tipo de transmissão passa necessariamente pelo servidor, o que causa a concentração do custo de *upload* e é um possível fator limitante de desempenho da banda de internet.

A principal vantagem da utilização do *peercast* em relação ao *streaming* por *broadcast* é a transmissão do áudio para um número ilimitado de clientes, sendo que o custo de um serviço de *streaming* a uma taxa de 64 kbps em um *streaming* por *broadcast* é em média de R\$ 150,00 para um número máximo de 100 clientes, além de comprometer o desempenho do servidor, pois todos os clientes estarão conectados ao servidor. O *peercast* por sua vez não necessita apenas de uma conexão de internet com taxa de *upload* acima de 64 kbps, e também não compromete o desempenho da banda de internet disponível, pois apenas poucos clientes estarão conectados a ele.

4. METODOLOGIA

Através de uma parceria com a Rádio Comunitária de Santa Amélia que atua em frequência modulada (FM), a qual necessita desse serviço, mas devido a altos custos para a montagem de uma radio web não o tinha feito. Foi desenvolvido um estudo e se chegou ao parecer que a tecnologia P2P era a melhor alternativa para essa rádio comunitária.

4.1 Estrutura da Radio Comunitária

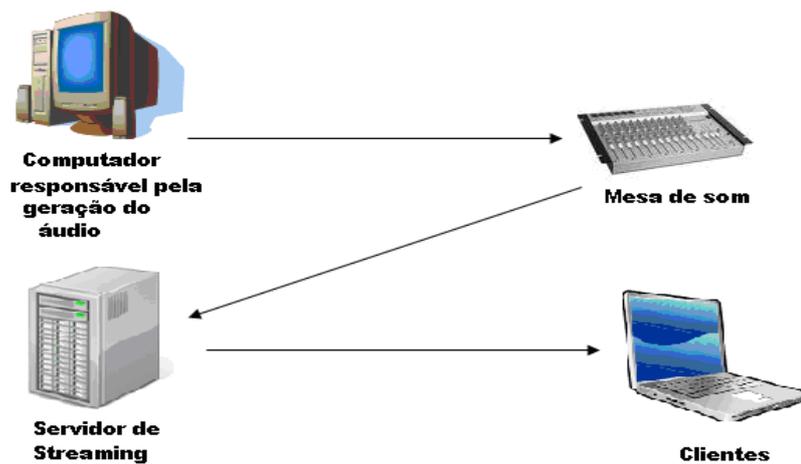


Figura 10 – Estrutura da Radio Web

A figura 9 descreve a estrutura física que radio comunitária é composta, o primeiro computador é responsável pela geração do som e por envia-lo a mesa de som, logo em seguida a mesa de som envia este áudio para o servidor de *streaming*, este fica responsável por envia-lo aos clientes.

O servidor de *streaming* é composto pelo seguinte *hardware*, processador Pentium 4 3.0GHz, 512 MB de memória e HD de 80 GB. Este composto pelas seguintes ferramentas para o funcionamento do *streaming*:

- Sistema operacional Ubuntu 9.04 com *kernel* 2.6.28-11. O Ubuntu é um sistema operacional *open source* baseado em linux, plataforma na qual foi desenvolvida a aplicação;
- *Icecast*: Sistema *multicast* responsável por receber o áudio e o distribuir para os seus receptores;
- *Ices2*: Sistema responsável pela captação do áudio lançado pela placa de som e por enviá-lo ao sistema que está fazendo o *streaming*;
- *Peercast*: Sistema que utiliza a tecnologia P2P responsável pela transmissão do áudio gerado pelo *Streaming*;

5. CONCLUSÃO

Conclui-se que a montagem de um servidor de *streaming* baseada na tecnologia P2P, é uma boa forma de transmissão de áudio principalmente levando em conta os custos para montagem de um servidor usando o serviço de *broadcast*, além de não influenciar no desempenho da banda de internet do servidor, pois poucos clientes estarão ligados diretamente ao servidor. A qualidade do áudio destinado ao cliente é a mesma comparada com o uso do *streaming* por *broadcast*.

Financeiramente o custo utilizando um serviço *peercast* chega a ser mais 50% menor em comparação ao *streaming* por *broadcast*

Este trabalho apresentou ainda um sistema de distribuição arquivos multimídia auxiliado por redes *peer-to-peer*. Nele, os usuários que estão recebendo o conteúdo estabelecem acordos para o encaminhamento de partes do fluxo, evitando a concentração do tráfego na origem do fluxo.

REFERÊNCIAS

AVILA, Renato Nogueira Perez. **Streaming: Aprenda a Criar e Instalar sua Rádio ou TV na Internet**. 1ª edição. Rio de Janeiro: Editora Ciência Moderna, 2008.

Braz, N. C. **Projeto Radio escola em Vargem Grande Paulista-SP**. Secretária de Educação, 2000. Disponível em <http://www.educacional.com.br/projetos/ef1a4/bancoprojeto1a4/radio/vargem.asp>. Acessado em: 12 de janeiro de 2009.

Bufarah, A. **Rádio na Internet: Convergência de Potencialidades**. Dissertação de Mestrado em Comunicação e Mercado apresentada a Fundação Cásper Líbero, São Paulo, 2002.

CASTRO, Inácio Rodrigo. **O que é P2P** 2008. Disponível na Internet via www. url: <http://www.inaciorodrigodecastro.com.br/o-que-e-p2p-peer-peer>, acessado em 11 de maio de 2009

CETIC.br. TIC DOMICÍLIOS e USUÁRIOS 2006: **Pesquisa sobre o uso das tecnologias da informação e da comunicação no Brasil**. Disponível na internet via www. url: <http://www.cetic.br/usuarios/tic/2006/index.htm>. Acessado em 12 de janeiro de 2009;

CUNHA, Magda Rodrigues e HAUSSEN, Doris Fagundes (orgs). **O pioneirismo do rádio levado à internet brasileira**. IN: Rádio brasileiro: Episódios e personagens. Porto alegre: Coleção Comunicação. EDIPUCRS, 2003.

D. A. Tran, K. A. Hua, T. Do, ZIGZAG: **An Efficient Peer-to-Peer Scheme for Media Streaming**. *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*, 2003.

Estácio, A. Rádio PC TV. São Paulo: **Revista What's In?** .Eletro Equip, Eletro Equip. Telecomunicações: dezembro de 2000, ano 5, número 13, p. 14.

ICECAST. *Icecast 2 Documentation Table of Contents*. 2008. Disponível na internet via www. url: <http://www.icecast.org/docs/icecast-2.3.2/>. Arquivo capturado em 4 de maio de 2009.

KUMAR, Vinay. **Real-time multimedia Broadcasts with the Internet Multicast Backbone**. Disponível na internet via www. Url:

<http://www.microsoft.com/mind/0297/mbone/mbone.asp>. Arquivo capturado em 17 de maio de 2009.

KUROSE, James F.; ROSS, Keith W..; **Redes de Computadores e a Internet**, pg. 1-513, 2004;

LIEB, T. **Sound on the Web**. Disponível em (<http://www.towson.edu/~lieb/editing/sound.html>). Acesso em 07 de maio de 2009.

MELLO, Samuel Lucas Vaz de. **Uma abordagem para distribuição de fluxos de conteúdo multimídia auxiliado por redes peer-to-peer**. Dissertação de Mestrado em Informática apresentada a Universidade Federal do Paraná, Curitiba, 2007.

MOREIRA, Sonia Virginia. **Novos espaços para o áudio**. In: Rádio em transição: tecnologias e leis nos Estados Unidos e no Brasil. MOREIRA, Sonia Virginia. Rio de Janeiro: Mil Palavras, 2002. p. 122-161.

ORAM, **Peer-to-Peer: O Poder Transformador das Redes Ponto a Ponto**. Editora Berkeley, 2001.

PEERCAT, 2006. Disponível em [www](http://www.peercast.org). url: <http://www.peercast.org>, Acesso em 3 de maio de 2009.

TOPIC, Michael. **Streaming Media Demystified**. 1.ed. EUA: McGraw-Hill Professional, 2002.

UFPA. NAS ONDAS DO RADIO: **Pesquisa sobre a história da radio web**. Disponível na internet via [www](http://www.oparanasondasdoradio.ufpa.br). url: <http://www.oparanasondasdoradio.ufpa.br> acessado em: 16 de janeiro de 2009;

ZAREMBA, Lílian. **Idéia de rádio – Entre olhos e ouvidos**. Rio de Janeiro: Universidade Fluminense. Revista Eletrônica Permanente. Mestrado de Comunicação, Imagem e Informação, número 2, 1999.