

Módulo para Segmentação de Vídeo em um Servidor de Contexto

Carlos Felipe Dias da Mata
Universidade federal de Juiz de Fora
Juiz de Fora
Minas Gerais - Brazil
+55-32-2102-3311
carlosfelipedias@gmail.com

Eduardo Barrére
Universidade federal de Juiz de Fora
Juiz de Fora
Minas Gerais - Brazil
+55-32-2102-3311
eduardo.barrere@ice.ufjf.br

ABSTRACT

This work is inserted into a context-server where we extract the context whether it is mobile devices, user, physical or time of any activity performed by the user of the system. In this paper is presented the module responsible for handling any video submitted to the context-server, this module extracts the meta-data required for processing the video, then the video will be processed to meet the specifications of each type of hardware that can access it also the video is segmented into several parts with a fixed size as it is transmitted using HTTP Streaming.

Categories and Subject Descriptors

H.5.1 Multimedia Information Systems

General Terms

Performance, Human Factors.

Keywords

Segmentação de vídeo, multimídia, contexto.

1. INTRODUÇÃO

Aplicações cientes de contexto têm como característica poder se adaptar com base no ambiente que estão sendo executadas e nas preferências do usuário. Para ampliar e potencializar uma aplicação dessa natureza é interessante um servidor de contextos bem estruturado com diferentes funcionalidades como a captura e persistência de contextos (usuário, dispositivo, temporal, espacial, etc.) e possibilidade de recuperação desses contextos, conforme a necessidade e características apresentadas pelas aplicações.

Uma forma interessante é reunir informações de contexto sobre o dispositivo com informações sobre o QoS da rede. Desta forma é possível potencializar a adaptação de vídeo [1]. A utilização de mídias adaptativas é um assunto que começou a ser tratado com maior ênfase no começo da década. Essa área oferece um rico conjunto de técnicas para tentar solucionar questões desafiadoras em aplicações que envolvam a mídia vídeo [2], visando transformar a mídia de entrada em uma saída em forma de vídeo ou Multimídia modificada pela utilização de manipulações em vários níveis (sinal, estrutural), a fim de satisfazer as limitações de

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

WebMedia '13, November 5-8, 2013, Salvador, Bahia, Brazil.
Copyright 2013 ACM 1-58113-000-0/00/0010 ...\$15.00.

recursos diversificados e as preferências do usuário [3]. Desta forma, os usuários podem acessar e interagir com o conteúdo em diferentes tipos de redes e em diferentes dispositivos. Portanto, as novas aplicações têm a necessidade de lidar com a enorme variação das limitações dos recursos como largura de banda, capacidade de exibição, a velocidade de CPU, alimentação, entre outras.

Uma ferramenta de adaptação de vídeo deve tratar de um ou vários arquivos multimídia atendendo suas necessidades, ou melhor, personalizações. A Figura 1 ilustra o esquema básico para adaptação de vídeo em ambientes de mídia pervasiva.

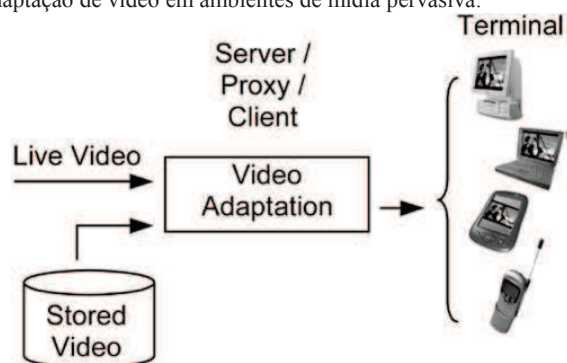


Figura 1 - adaptação de vídeo difundida em ambientes de mídia para apoiar os terminais e redes heterogêneas [2]

Este trabalho aborda uma solução adotada para a segmentação, dentro do projeto que envolve o desenvolvimento de um servidor de contexto e aplicações que consumam os conteúdos (contextos e mídias) por ele providos.

2. SEGMENTAÇÃO DE VIDEO

Existem diversas configurações para um arquivo de vídeo, como formato, resolução, codec, quadros por segundo, etc. Essas configurações influenciam na qualidade, tamanho e processamento da mídia. Elas também são fundamentais para que o vídeo possa ser exibido em diferentes tipos de hardware, por exemplo, a resolução do Iphone é 320x480 já a da TV-HD é 1920x1080. Para fornecer o vídeo para os dois tipos de equipamentos citados, temos como possibilidade:

- Enviar o mesmo vídeo para os dois. Desta forma, ou o vídeo terá uma resolução muito grande para o dispositivo ou muito inferior à resolução suportada pelo dispositivo.
- Adaptar, em tempo real, a resolução, conforme a resolução do display. Isso gera diversos problemas, entre eles o grande processamento necessário na conversão e a ausência de escalabilidade no provimento do serviço.

- C. Gerar versões diferentes da mídia, conforme os possíveis displays previstos que consumirão o vídeo. Esta solução demanda pré-processamento do vídeo e grande espaço de armazenamento.

As soluções B e C envolvem a adaptação de mídia. É importante frisar que adaptação de vídeo e codificação de vídeo são duas coisas diferentes. Transcodificação pode ser de dois tipos: homogeneia que modifica um conjunto de parâmetros do vídeo sem modificar o seu padrão de compressão, ou seja, realiza conversão da taxa de bits, da resolução espacial, da resolução temporal e mudanças da codificação VBR e heterogênea, onde se realiza a conversão de padrões de compressão ou a conversão entre os formatos entrelaçado e progressivo [1].

Também existe outra questão: como o vídeo será transmitido? Mesmo tendo uma boa largura de banda um arquivo de vídeo por ser geralmente grande pode demorar muito tempo para ter a transmissão concluída, se o usuário quiser ver o vídeo imediatamente ele vai ter que esperar, mas isso não é uma situação interessante. Pode-se adotar como solução:

- A. Dividir o arquivo original em pequenos arquivos de tamanho fixo que serão enviados ao cliente, como utilizado no Youtube e no Sistema RIO [4].
- B. Utilizar alguma técnica de streaming (fluxo) de vídeo [5].
- C. Segmentar o vídeo em pedaços baseados em grandezas do próprio vídeo, como o tempo, por exemplo. Segundo [6], o tamanho ideal dos segmentos é de 6 segundos para conexões não persistentes e 2 segundos para conexões persistentes.

Como o presente projeto tem como objetivo prover o vídeo para aplicações multimídia e estas tradicionalmente utilizam o tempo como elemento de sincronização do vídeo com as demais mídias, optou-se por utilizar a segmentação baseada em tempo para a implementação do projeto. Outra opção adotada foi a adaptação baseada na resolução espacial (dividiu-se em grupos de dispositivos) e na taxa de bits, não adotando a adaptação na resolução temporal (retirar quadros do vídeo), pois a mesma pode gerar perdas significativas na qualidade do vídeo e no seu processo de compactação [1]. Devemos observar que é inviável prover vídeos 100% adaptados a todos os tipos de hardware, pois para cada mídia recebida deveríamos ter que gerar dezenas ou até centenas de mídias adaptadas, isso consumiria muito espaço em disco além de gastar um tempo muito maior de processamento. O que decidimos fazer foi adaptar para os tipos mais comuns de hardware que são Computador, TV-digital, móbil. Para computadores é mais fácil, pois ele geralmente consegue reproduzir qualquer tipo de vídeo, basta ter os codecs instalados. Para TV-digital estamos levando em conta que elas são HD, sabemos que as configurações de vídeo como formato, resolução, etc. são padronizadas vamos utilizar os dados obtidos da TV Samsung [8] que aceita vários tipos de configuração, mas certamente não vamos utilizar as configurações de maior qualidade devido às velocidades de conexão. Já no caso dos dispositivos móveis é mais complicado, pois as configurações dos aparelhos mudam bastante e como dito anteriormente é inviável começar a criar mídias para todos os hardwares. O que esta sendo feito é analisar os tipos mais gerais de configuração, a principio usaremos a especificação do Andróide [9] para o codec h.264 que segundo [10] um codec extremamente eficiente, pois tem qualidade igual ao mpeg2 porem comprime o vídeo duas vezes mais. O Andróide esta sendo usado como referência, pois a maior parte dos fabricantes tem adotado esse sistema operacional.

Optou-se também em utilizar segmentos de 4 segundos, pois eles atendem de forma satisfatória os dois tipos de conexão, conforme mostrado na Figura 2. Vale a pena observar que a largura de banda utilizada é fora da realidade da maioria das cidades brasileiras, ainda mais se tratando de conexões em aparelhos moveis, então possivelmente esse tamanho pode ser alterado após a realização de testes. Segundo [6], seguimentos de tamanho menor podem causar uma maior sobrecarga em termos de pedidos, por outro lado seguimentos maiores podem dificultar a transmissão em clientes com conexão mais fraca.

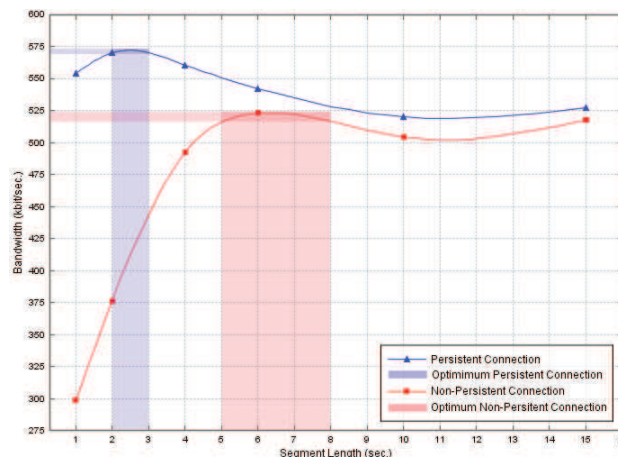


Figura 2 - Largura de banda x Tamanho dos segmentos [6]

3. MÓDULO DE VIDEO

A solução adotada para a transmissão do vídeo por [3] é o DASH (*Dynamic Adaptive Streaming over HTTP*) que é uma solução aceita pelo W3C. A Figura 3 descreve a arquitetura geral de uma aplicação que implementa o DASH, o MPD (*Media Presentation Dataset*) é um arquivo que descreve os segmentos e seus tempos.

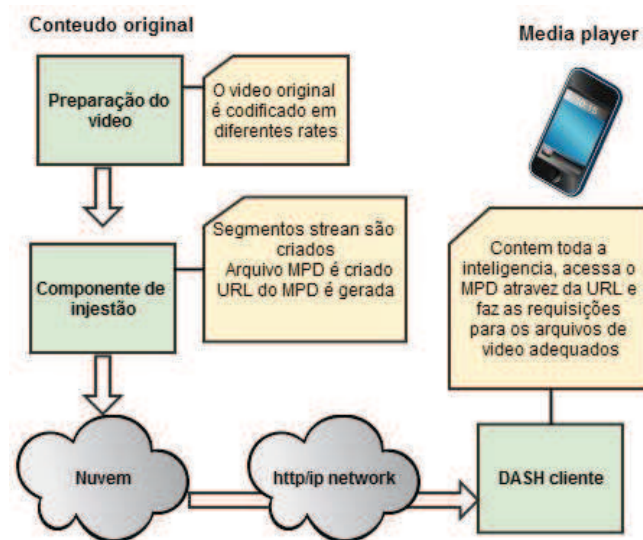


Figura 3 - Arquitetura do DASH

No presente trabalho foi adotada a arquitetura do DASH como solução. O vídeo é convertido gerando diferentes segmentos de mídia, cujas propriedades como resolução, codec e formato sejam

mais adequadas para cada tipo de dispositivo. Essas mídias são segmentadas de modo que, para cada dispositivo, grupos de segmentos com diferentes bitrates são gerados. A Figura 4 ilustra esse processo.

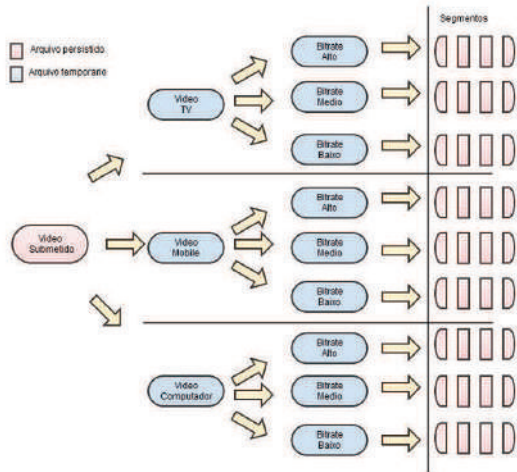


Figura 4 - Processo de criação dos segmentos

Dessa forma, cada vídeo original será convertido para até três resoluções diferentes e para cada resolução serão gerados três grupos de segmentos, cada um com uma taxa de bits diferentes. Assim é possível disponibilizar o vídeo mais adequado para o contexto de display do equipamento e ainda disponibilizar os segmentos conforme o QoS da rede (taxas de bits distintas).

4. SERVIDOR DE CONTEXTO

Permite que vários clientes acessem de maneira concorrente as informações dos dispositivos. É similar a arquitetura cliente-servidor, onde o servidor terá o seu acesso os dispositivos e as informações geradas e o cliente irá requisitar essas informações. Problemas como protocolo de comunicação, desempenho de rede e qualidade de serviço devem ser levados em conta ao se usar um servidor de contexto. A figura a seguir mostra a arquitetura do servidor de contexto no qual este projeto esta inserido.

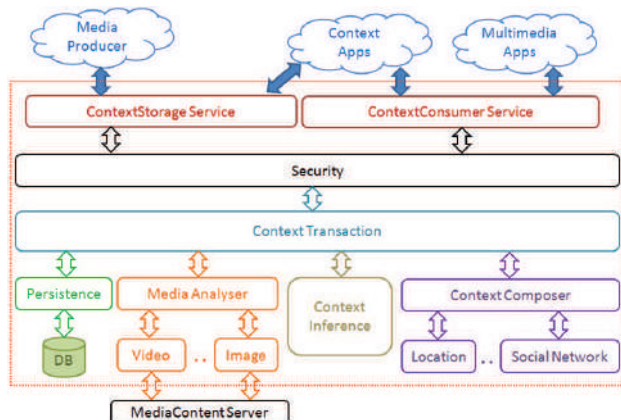


Figura 5: arquitetura do servidor de contexto

É importante frisar que o Módulo de vídeo está inserido no servidor de contexto, mas isso não é o ideal, futuramente haverá um servidor de vídeo dedicado que será acionado pelo “Media

Analysers” do servidor de contexto quando o “Media Producer” que pode ser um usuário do sistema submeter uma media cujo contexto é vídeo. Para o “Media Analyser” essa mudança será transparente, pois ele envia a requisição para a API do módulo de vídeo, não importando para ele como a API vai tratar essa requisição.

Detalhar o servidor de contexto explicando suas funcionalidades e implementação não faz parte do escopo desse trabalho

5. IMPLEMENTAÇÃO MÓDULO VÍDEO

O “Media Analyser” é responsável por receber uma mídia e repassar para o módulo responsável pelo seu tratamento. Este módulo está inicialmente implementado junto ao servidor de contexto, mas em etapa posterior será incorporado ao servidor de mídias, também em desenvolvimento no projeto conforme explicado anteriormente.

A arquitetura do módulo responsável pela segmentação de vídeos é apresentada na Figura 6. Esse módulo espera que o “Media Analyser” faça uma requisição de uma nova mídia vídeo (Receive Request). Nesta requisição são passados como parâmetros o XML e o ID correspondentes à mídia, um dos elementos contidos no XML é a URL do vídeo.

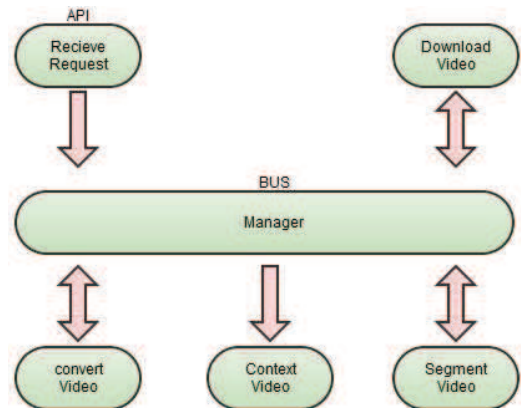


Figura 6 – Arquitetura do Módulo de Segmentação de Vídeos

O arquivo de vídeo pode estar armazenado localmente (no próprio servidor), em outra máquina da rede ou na internet. Caso não esteja armazenado localmente, é realizado o download do vídeo para o servidor (Download Vídeo) e, em paralelo, é verificada a necessidade de obter informações referentes ao contexto do vídeo (pessoas, locais, etc.). Se o contexto for relevante é disparada (Context Vídeo) uma requisição para o “Media Analyser”, que por sua vez repassará essa requisição ao módulo que faz a análise/busca de contexto nas diversas mídias.

Em seguida é iniciada a conversão do vídeo (Convert Vídeo) para diferentes tipos de display. Depois de finalizada a conversão, é iniciada a fase da segmentação (Segment Vídeo), onde cada formato de vídeo é dividido em segmentos com duração de 4 segundos, esse processamento é feito utilizando a ferramenta FFmpeg (www.ffmpeg.org). O passo final é a geração de um XML, contendo um cabeçalho com informações de contexto, algumas informações da mídia e uma estrutura contendo metadados sobre os segmentos de vídeo. A importância deste arquivo XML é que aplicações cliente irão acessar esses vídeos e dependendo do contexto de hardware e de conexão do cliente ele segundo [3] tem a “inteligência” de escolher os segmentos

adequados, para isso ele precisa das informações sobre eles que esta contida no XML.

Todo o processamento presente no módulo de vídeo é coordenado por um BUS que executa os submódulos correspondentes às requisições e eventos que acontecem.

6. CONCLUSÃO

Mesmo nos dias atuais a disponibilização de vídeos é um desafio, ainda mais com o surgimento de novos dispositivos computacionais, com processamento gráfico, tamanho do display e acesso à internet tão distintos. Dentro deste universo de possibilidade e requisitos diferentes, a adaptação da mídia a diferentes contextos é um fator importante para se garantir a disponibilidade com qualidade e agilidade satisfatória.

A solução apresenta boa resposta às variações de QoS durante o fornecimento do vídeo, entregando o próximo segmento com outra taxa de bits, mas estudos e simulações se fazem necessárias para uma avaliação mais precisa desta adaptabilidade.

O ponto negativo da presente solução é o grande pré-processamento e necessidade de espaço de armazenamento, mas processamento e armazenamento são conseguidos com certa facilidade nos dias atuais. A partir de futuros testes de desempenho, pretende-se comprovar que o esforço é recompensado pelo desempenho gerado, podendo atender computadores, Televisores, dispositivos móveis, principalmente quando utilizados como segunda tela [7].

7. REFERÊNCIAS

- [1] Regis, C. D. M. Anjos, G.C.B., Oliveira, J.F.F., Rocha, J.S., Farias, M.C.Q. and Alencar, M.S. 2008. Transcodificação de vídeo digital para receptores portáteis. IV SEGeT – Simpósio de Excelência em Gestão e Tecnologia.
- [2] Chang, S. and Vetro, A. 2005. Video Adaptation: Concepts, Technologies and Open Issues. Mitsubishi Electric Research Laboratories. In: <http://www.merl.com/>.
- [3] Stockhammer, T. 2011. Dynamic adaptive streaming over HTTP: standards and design principles. ACM-MMSys '11. Proceedings of the second annual ACM conference on Multimedia systems. Pages 133-144.
- [4] Silva, E. S. Leão, R. M. M., Santos, A. D., Azevedo, J. A. and Netto, B. C. M. 2006. Multimedia Supporting Tools for the CEDERJ Distance Learning Initiative applied to the Computer Systems Course. 22th ICDE World Conference on Distance Education. pages 1-11.
- [5] Jim Summers, J., Brecht, T., Eager, D. and Wong, B.. 2012. To Chunk or Not to Chunk: Implications for HTTP Streaming Video Server Performance. Network and Operating System Support for Digital Audio and Video - ACM-NOSSDAV'12.
- [6] Lederer, S. Müller, C. and Timmerer, C.. 2012. Dynamic Adaptive Streaming over HTTP Dataset. ACM-MMSys '12 - 3rd Multimedia Systems Conference. Pages 89-94. Finger, C. and Souza, F. C.. 2012. Uma nova forma de ver TV no sofá ou em qualquer lugar. Revista FAMECOS: mídia, cultura e tecnologia, Vol. 19, No 2 – PUCRS.
- [7] Samsung LED TV – AllShare supported video format <http://inglele.wordpress.com/2011/01/03/samsung-led-tv-allshare-supported-video-format/>. Visitado em Julho de 2013.
- [8] Supported Media Formats. <http://developer.android.com/guide/appendix/media-formats.html>. Visitado em Julho de 2013.
- [9] Ricardo L. De Queiroz 2005. Elementos básicos de transcodificação de vídeo para o sistema Brasileiro de Televisão Digital(SBTVD), Ricardo L de Queiroz, Universidade de Brasília