

VOD を利用した学習支援環境の構築 (Ⅲ)

3 P-7

小林 健一 飯倉 道雄 吉岡 亨 伊原 征治郎

日本工業大学工学部

1. はじめに

コンピュータが一般家庭に普及するにしたがって、デジタル・データを利用した教材も広く利用されるようになってきた。

しかし、現在普及しているマルチメディア教材は、そのほとんどが、CD-ROM 等のスタンドアローンでの使用を前提とした媒体に記録されている。

そこで現在、コンピュータ・ネットワークを利用して教材を管理、提供する方法についての研究がさかんに行われている。

本研究では、コンピュータ・ネットワーク上でマルチメディア教材を管理し、利用するための方法について検討する。

2. 前回からの変更点

JDK(Java Development Kit)1.1 を使用し、ビデオ・データを jar アーカイブ・ファイルの形でまとめることで、ビデオ・データの管理をしやすくした。

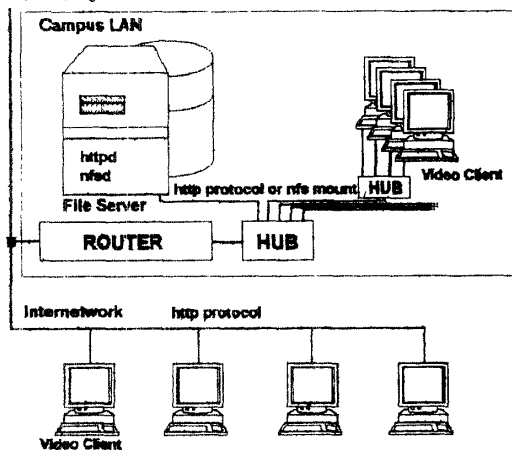


図1 VOD システム構成図

Development of a Learning Environment for Computer Engineering with a Video On Demand System(Ⅲ)

Kenichi Kobayashi, Michio Iikura, Toru Yoshioka, Seijirou Ihara

Nippon Institute of Technology

4-1 Gakuendai, Miyashiro, Saitama 345, Japan

暫定的ではあるが、オーディオ・データファイルを画像の表示間隔に合わせて分割し、これを再生する方法をとった。

転送速度の変化にある程度対処するため、最初の jar ファイルの読み込み時間をもとに、ビデオ・データ再生を開始する時間を遅らせる処理を追加した。

3. 実験環境

実験に使用したハードウェアは、前回の実験と同じである。ファイル・サーバ、ビデオ・クライアントともに、IBM-PC 互換機を使用している。

ファイル・サーバは、jar アーカイバによって圧縮された画像ファイルと音声ファイルを格納する。ビデオ・クライアントは、10Base-T でスター型に接続されている。

ビデオ・クライアント側の OS 上では、JDK1.1 が動作していなければならない。今回は、サーバ、クライアントともに、OS として、Linux kernel version 2.0.xx を使用した。

インターネット上でも動作可能な VOD システムを構築するため、使用する機器、OS に比較的依存しない動作環境として、Java を使用した。

4. 実験方法

再生プログラム

ビデオ・データの圧縮、非圧縮の両格納方式に対応した再生プログラムを作成した。

ビデオ・データ

ビデオ・データは、前回の実験で使用したものと同じ、GIF 形式の画像ファイルに加えて、Java 上で使用できる AU 形式のオーディオ・ファイルを使用して構成した。

画像データは、1 秒間に 1 枚表示するため、音声データも 1 秒ごとに分割した。

初回再生開始までの時間測定

再生プログラムの実行には、JDK1.1 に添付き

れている、appletviewer を使用する。X-Window 上の kterm から appletviewer を実行した時の、ビデオ・データ再生までにかかる時間を測定した。

5. 実験結果

再生プログラムとビデオ・データ

再生プログラムが起動した状態を図2に示す。今回の再生プログラムでは、ビデオ再生を途中で止めても、再実行時には止めた位置から、画像、音声共に再生される。

圧縮データと非圧縮データの両方について、一括読み込みと、読み込みと並行して再生するプログラムを作成した。

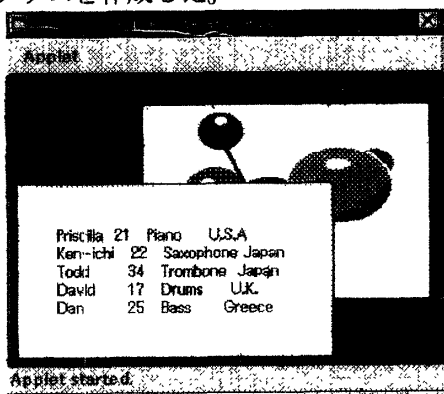


図2 再生プログラム起動時の画面

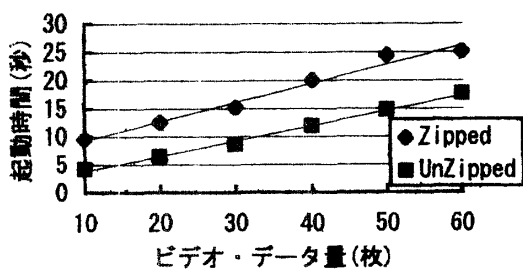


図3 ファイル枚数と起動時間
(一括読み込み時)

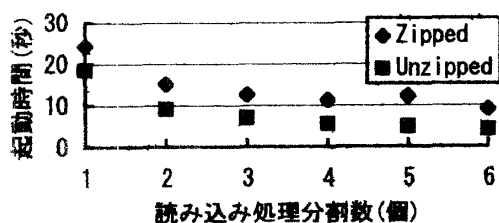


図4 ファイル枚数と起動時間
(読み込み、再生平行処理時)

ビデオ・データ

音声データを分割してしまったため、音のブツ切れ現象などが発生することが考えられたが、音声ははっきりと連続して聞き取れる。ただし、1秒ごとに低音の雑音が入ったり、時々、前後の音声为重なるといった不都合も生じる。

初回再生開始までの時間測定

読み込み方式の違いによる、初回再生開始までの時間の測定結果を図3、図4に示す。

全て読み込んでから再生する方法では、非圧縮データを読み込むよりも、圧縮データを読み込む方が起動時間がかかっている。これは、今回の実験が、LAN 上で行われたためである。

読み込みと再生を並行して行う方法では、ビデオ・データの解凍を再生プログラムの中で行っている。このため、全て読み込む方法と比べても、起動時間が早くなる。

6. 考察

転送すべきデータ量は、jar ファイルの圧縮効率にかかっている。そこで jar アーカイバを使用することで、どの程度ビデオ・データが圧縮されるのかを調査した。その結果、AU ファイルは、非圧縮のファイルであるため、約半分まで圧縮されることが分かった。それに対し、GIF ファイルはほとんど圧縮されず、場合によっては、元のファイル・サイズよりも大きくなることが分かった。

音声のランダム・アクセスは、一応可能だが、今回の方法では不完全である。雑音を消す方法が望まれる。

7. おわりに

Java が本格的に音声や動画像に対応するのは、時間の問題ではないかと思われる。しかし、そのような拡張を行わなくても、現在の Java の機能だけでも、学習支援環境用には十分な規模の動画再生が行えることが分かった。

参考文献

飯倉道雄、吉岡亨：情報教育を支援するマルチメディア環境の開発、情報処理学会 51 回全国大会講演論文集(1995.1)