ＡＴＭネットワークを用いたクラスタ型ビデオサーバシステム

金田 悟　　菊地 芳秀
NEC 機能エレクトロニクス研究所

ＡＴＭネットワークを用いたクラスタ型ビデオサーバの概要と評価結果について報告する。クラスタを構成する各ディスクエレメントへのアクセスが集中すると、配送が遅れて滑らかな動画再生に支障をきたす。これを避けるため、本ビデオサーバでは各エレメントを通信により互いに同期させ、同期したタイムスロットテーブルにもとづいて動画データをクライアントに配送する。タイムスロット管理により個々のエレメントが持つスループットのほぼ最大値近くまでストリームを供給することができた。また、エレメント間通信をほとんど行わないことにより、サーバ台数に比例したスケーラビリティを実現できた。

1.はじめに

近年のディジタル情報の記憶、処理、およびネットワークシステムの高性能化によって、連続的な大規模データである音声や動画をオンデマンドで配送するビデオサーバシステムが可能となり、実用化実験が進められてきた[1]。

また同時に、より高帯域なビデオデータのサービスや、より多くの利用者へのサービスのため、スケーラブルに規模を拡張できるビデオサーバシステムが望まれている。ビデオサーバの処理の大半は、記憶装置からの連続的な読み出しとネットワークへの配送であり、単純な計算機を ATM などによる高速ネットワークで結んだクラスタシステムとして実現する研究がなされてきている[2]。

筆者らもこれまでに、クラスタ型 VOD サーバのアーキテクチャの検討と提案を行なってきた[3]。本稿では、そのアーキテクチャの実現方式と性能評価について報告する。

Clustered Video Server System on ATM Network,
Satoru KANEDA,　Yoshihide KIKUCHI, Functional Devices Laboratories, NEC
{kaneda,kiku}@mech.cl.nec.co.jp

2.クラスタ型ビデオサーバの検討

2.1.クラスタ型ビデオサーバの分類

クラスタの各エレメント間でコンテンツを共有するタイプのクラスタ型ビデオサーバを分類すると、次の３つの方式に分類することができる。

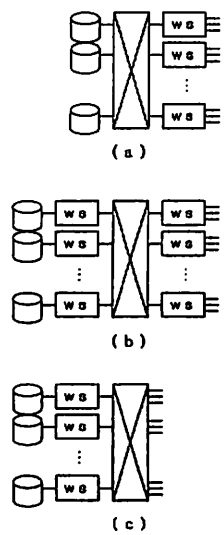


図１　クラスタ型ＶＯＤの各種方式

(a)は再構成エレメントがネットワークを通してディスクを共有するディスク共有型である。

(b)はディスク非共有型でかつ再構成エレメントを持つタイプである。左側のファイルエレメントでディスクからデータを読み出し、右側の再構成エレメントでデータを並べ変えてユーザに配信する。

(c)は、再構成エレメントを持たず、ファイルエレメントそのものがユーザへ順番に映像を送り出す方式である。

次に、これらの方式を選択する上で、クラスタ型ビデオサーバに生じる２つの課題について取り上げる。

2.2.クラスター型サーバの課題と解決方法

2.2.1.ストライピング格納

コンテンツをクラスタ型サーバのエレメント間で共有する場合も、従来のビデオサーバと同様、エレメント間でストライピング格納することで、負荷を分散することができる。

しかし、単純なストライピング格納では、同じファイルエレメントに過度の利用者が同時にアクセスすると問題が生じる。図２のように、格納順序を各周期で同じ順序とすると、このアクセスの集中は、同じ利用者の組で連続して発生するので、動画再生品質を著しく悪化させる。

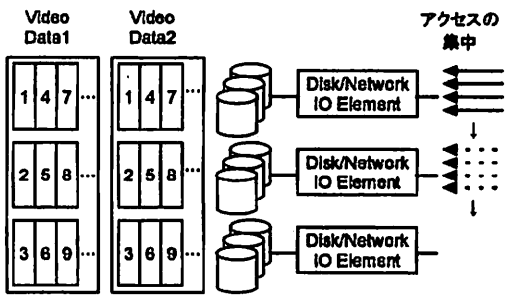


図2 アクセスの集中の発生

この集中を避ける為に、従来の単一サーバではタイムスロット管理が行われてきたが、疎結合されているクラスタサーバでのタイムスロット管理は難しく、あまり例を見ない。特に図１(b)のように要求がディスクへ到着するまでにネットワークとワークステーションが間に入る構成ではほとんど採用されていない。このタイプでは要求が集中する確率を計算し、フレーム落ちが許容できる程度にユーザ数を制限する方法が採られるが、この方式では、個々のワークステーションが持つ最大スループット近くまで利用することが難しい。

一方、タイムスロット管理を採用すると、ストライピング周期が長くなるので、混雑時にタイムスロットの空きを待つ待ち時間の最悪値が大きいという欠点がある。この対策としては、格納順序を周期毎に変化させ、連続的にはアクセスが集中しないようにして、見かけのタイムスロットを増やす方法などである程度解決を図ることが可能である。

2.2.2.プロトコル

サーバと STB 間のプロトコルとしては、一般に、NFS のように STB が主体となってブロックの配送をサーバに要求する方式と、サーバが主体となって STB にブロックを送り付ける方式がある。STB が主体となってサーバからデータを要求する方式では、NFS 等のプロトコルを STB にも持たせることになり、STB の軽量化が図りづらい。一方、STB とサーバ間のプロトコルを軽量にしようとすると、最も柔軟性のある方式を採れば図１(b)の方式となり、サーバの低価格化がしづらい。

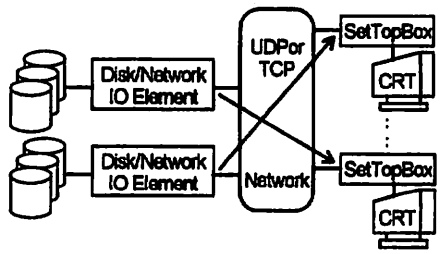


図3 データ配送プロトコル

2.3.本方式での取り組み

筆者らは、クラスタ型ビデオサーバの開発にあたって、次の点に重点を置いた。

1) 低価格性
2) スループットを最大限に引き出せる（保証できる）
3) スケーラビリティが良い

4) STB を簡略化できる

低価格性を重視することから、図1 (b)のタイプは避けた。

さらに、スループットを最大限に引き出せればストリームあたりの単価も安くなるため、「2.2.1 ストライピング」で検討したように全エレメントで共通なタイムスロット管理を取り入れることが有効と考えた。

また、タイムスロット方式を採用すると、サーバ主導でストリームを配信できる。これは、「2.2.2 プロトコル」で検討したように、STBの機能を簡略化することが可能となる。

ただし、タイムスロット管理を用いた時の問題点は、タイムスロットのタイミングを1箇所からブロードキャストする方式を採ると構成エレメントの台数が増えるとオーバヘッドが無視できなくなるほど大きくなることである。このため、それぞれのエレメントがゆるい同期制御を行いながら独立して動作することとした。

これらの検討の結果、図1 (a)と(c)が候補に残ったが、(a)のディスク共有環境が現状では構築しづらいため、(c)のアーキテクチャを採用した。本ビデオサーバは、その上に「タイムスロット管理」と「各エレメントの独立動作機構」を開発してクラスタ型サーバを構築している。

3.システムの構成

システムのモジュール構成を下図に示す。

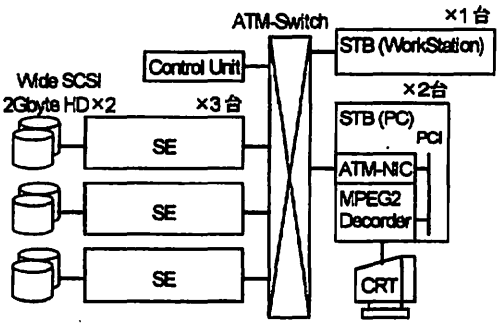


図4 実験システムの構成

クラスタ型サーバ PVS は、1台のコントロールユニットと3台のストリームエレメント(SE)からなり、CPU としては UNIX ワークステーションを用いている。また、個々の SE には WideSCSI を2台づつ接続している。

クライアント側の STB には、受信のみ行う UNIX ワークステーションと、MPEG のデコードも行う Windows PC で構成した。

クラスタ型ビデオサーバとクライアント間は、155Mbps の ATM で接続されている。接続は、現状では PVC を用いている。

個々の要素の詳細は表1に示す通りである。

表1 実験システムの構成要素

SE	CPU, Memory	R4400(200MHz), 160Mbyte
	OS	UX4800 R11.5:UNIX SVR4(MP)
	Disk I/F	Wide SCSI / PCI board
	Network I/F	ATM-NIC / PCI board
STB (WS)	CPU, Memory	R4400(200MHz), 160Mbyte
	OS	UX4800 R11.5:UNIX SVR4(MP)
	Network I/F	ATM-NIC / PCI board
STB (PC)	CPU, Memory	Pentium(130MHz), 48Mbyte
	OS	Windows95
	Network I/F	ATM-NIC board (PCI bus)
	Decoder	MPEG2 Decoder board (PCI bus)
ATM Switching hub		155 Mbps, 7 port

WS…Workstation

4.システムの動作

4.1.全体の動作

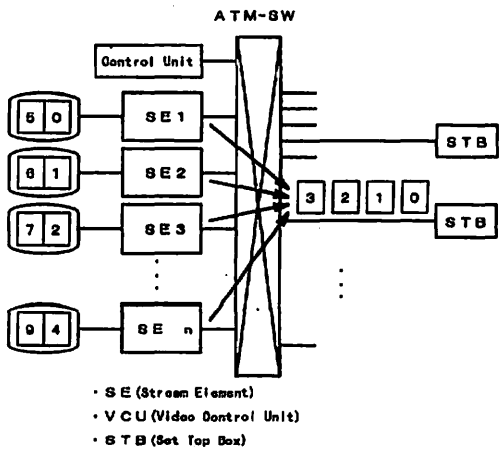


図5 動作概要

映像は、各 SE 間にストライピングして格納される。このため、各 SE は他の SE と同期を取り、STB に順番に映像が届くように映像を送り出す。

ユーザ側(STB)では、SE 側のどこにデータがあるかは意識しない。順番に送られてくるデータを単に受取るという動作を繰り返していく。

4.2.映像再生プロトコルの概略

利用者のビデオ操作に関する指令(再生開始、停止、ジャンプ等)は、ユーザ側の端末であるセットトップボックス(STB)からビデオサーバを管理するビデオコントロールユニット(VCU)に一旦送られ、そこからビデオデータの読み出しと配送を行うストリームエレメント(SE)に分配される。各 SE は、再生開始指令からファイルが終了するか、停止指令が与えられるまで、連続的にファイルを読み出して、STB に配送し、STB はこれを表示する。

利用者のビデオ操作の指令から、映像データの配送までのプロトコルの概略を図 6 に示す。

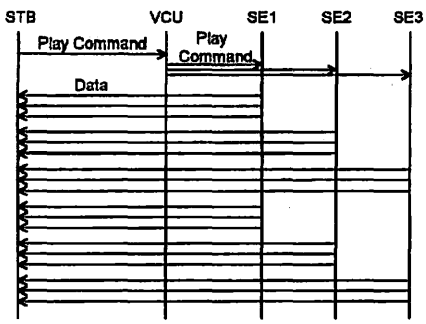


図6　映像再生までプロトコルの概略

SE は、ディスク上のビデオデータを、ブロックを単位として読み出し、バッファに格納しておき、送信タイミングに達したら、1 パケットづつ STB に送信する。

4.3.SE のデータ送信と時間同期

データの配送プロトコルは、送信負荷の軽い UDP 用い、ロス率を抑えるため、SE は各パケットをできるだけ等時間間隔で STB へ送信する。また、ある STB が複数の SE から同時にパケットを受信することもロスの原因となるので、これを防ぐために、SE どうしを Network Time Protocol (NTP)[4]で同期させるとともに、SE の切り替わりの一定時間前は、データを送信しないマージン時間ができるように、データを配送する。

NTP はおよそ 15 分ごとの同期でも、およそ 10ms までの時刻同期精度を持つ。基本的には、各 SE が持つクロックを基に動作し、時刻のずれが生じるとその差を徐々に縮めるような補正情報がそれぞれに通知される。今回の実験システムでも、補正間隔は１５分に１度程度なので、通信や CPU の負荷はほとんど生じない。

4.4.SE 内のストリーム管理

各 SE では、ストリームを管理するための同一のテーブルを持っている。SE は、VCU からストリームの再生要求を受けると、該当するストリームを各自のテーブルに登録する。

ストリーム管理プロセスは、テーブルから処理すべきブロックを抽出し、キューにバッファリングする。それぞれのブロックはディスクへのアクセスタイミングと ATM への送信タイミングを持っており、その時間になるとディスクへの要求と ATM への送信が独立に処理される。ディスクの管理と ATM への送信管理が独立に行われるため、それぞれの資源を効率良く管理できる。

また、CPU の負荷を軽くするために、これらの動作は単一プロセス内で行われる。

4.5.SE から STB 間の配信

SE から STB へは、UDP プロトコルを使用した。各 SE から STB へは時間順に送られてくるため、STB 内での並べ換えは基本的には行わなわない。

ただし、SE 間の時間同期の乱れが起き得ることを考慮して、切り替え付近にマージンを用い、さらに STB にダブルバッファを設けた。

現状では、SE と STB 間は、ATM の PVC で接続されている。将来的には SVC で接続し、拡張性を持たせる予定である。

5.基本性能の評価

前述のビデオサーバの基本性能がスケーラブルに得られるかどうかを、実際にストリームを流して評価した。

サーバ側の SE を 1 台から最大 3 台まで変化させ、クライアント側の利用者数も 1 人から 40 人まで変化させて、ビデオサーバの基本性能であるストリーム数、パケット間隔の変動、フレームのロス率を測定した。

評価は、UNIX ワークステーションでスケーラビリティの評価を行い、Windows PC で実際の画像を送り、見ながら動作の確認を行った。

5.1.最大ストリーム数

配送に乱れを生じない利用者数（すなわち最大ストリーム数）を調べるため、ユーザ数を変化させた時の配送レートを測定した結果を図 7 に示す。ただし、実験では全てのストリームを受信するだけの STB を用意できなかったため、1 台の UNIX ワークステーションで受け、残りは ATM スイッチ内で破棄するように設定した。

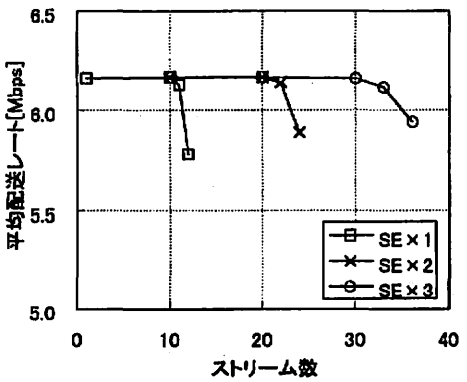


図7　ストリーム数と平均配送レート

データとして MPEG2 を想定すると、各ストリームは約 6Mbps／1 ストリームの配送レートが必要である。サーバ側では、平均配送レートが 6Mbps になるように送り出しているが、サーバの能力を越えると転送が間に合わなくなってくる。この時点のストリーム数がサーバの供給できる最大ストリーム数となる。

測定結果から、この配送レートで配送可能できるストリーム数は、サーバ 1 台当たり、10 ストリームであり、この値はサーバ台数が 1 台から 3 台の間で、変化がなく、良いスケーラビリティが得られていることが分かる。この理由は、動作の項でも説明したが、主に、1)SE 間あるいは SE-VCU 間の通信が非常に少ないこと、2)SE 内のタイムテーブル処理が非常に軽いことによると考えられる。

なお、ストリームはタイムスロット管理されているため、アクセスの集中による突発的なフレームロスは起きない。

5.2.応答時間

再生開始コマンドからビデオデータの配送が始まるまでの応答時間(図 8)は、STB から VCU へ再生命令を出してから最初の画像が STB へ到着するまでの応答時間は、ストリーム数が 1 の場合では、平均 0.6s である。

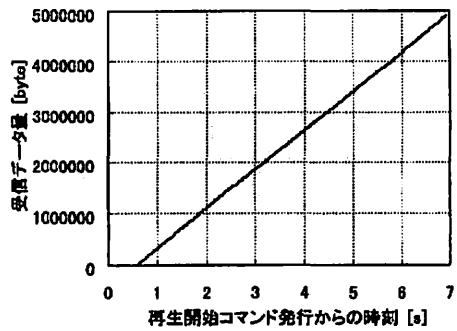


図8　応答時間

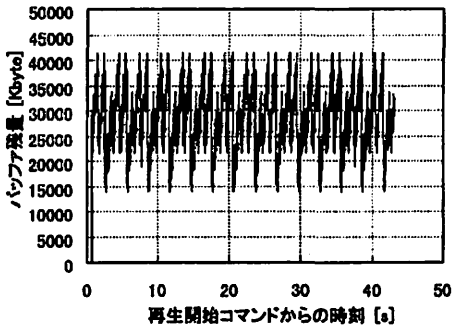
ただし、タイムスロット方式を採っているので、最大ストリーム付近で応答時間は長くなる。

5.3.パケット間隔の変動

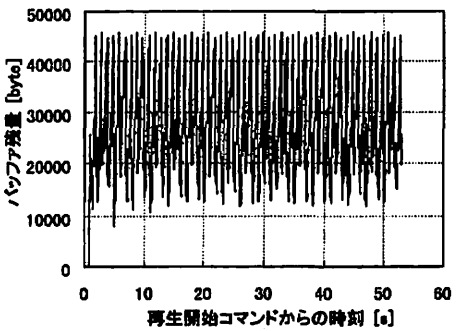
サーバが送信するデータの配送レートの変動が少なければ、STB に必要なバッファ量とロス率および再生開始までの遅延時間を共に抑えることができる。

まず、実際の STB に必要なバッファ量を評価するため、仮想的なバッファのデータ残量の変動を測定した結果を図 9 に示す。これは、受信データを蓄えているバッファから、MPEG デコーダが一定の速度で読み出すものとして求めた。

図では、ストリーム数が 1 と 30 の場合を示す。どちらも最初のデータの受信からデコードの開始までの遅延時間は、0.03 秒とした。図に示すように、バッファ残量の変動は、非常に少なく、遅延時間とバッファを小さく抑えることが可能である。また、ストリーム数の影響もほとんどないことが分かる。



(a) 1 ストリームの場合



(b) 30 ストリームの場合

図9 バッファ内データ残量の変動

5.4. 瞬間最大配送レート

一方、瞬間的に配送レートが高くなると、STB のドライバが ATM—NIC からの受信処理が間に合わずに、セルロスが発生する。

10 パケット毎の受信時間から、瞬間最大配送レートを求めた結果を表 3 に示す。この結果から、ストリーム数によらず、瞬間最大配送レートにはほとんど変化はないことが分かる。

表2 ストリーム数とパケット数の変動

ストリーム数	瞬間最大配送レート[M bps]
1	7.28
30	7.28

6.おわりに

クラスタのエレメント間でデータを共有する形式のクラスタ型ビデオサーバシステムの開発をすると共に、その性能評価を行った。

本クラスタ型ビデオサーバでは、各エレメントが時間同期しながらも独立して STB に映像を配信するアーキテクチャを提案している。このことにより、サーバ全体としてタイムスロット方式を適用することができ、個々のエレメントが持つスループットの最大値近くまでサーバのスループットを得られる。

上記方式により SE 間、あるいは VCU と SE との間の通信を極めて少なくすることができた。また、SE 内のタイムスロット管理ではイベント駆動型プログラムを 1 プロセスで実現したことにより、オーバヘッドを非常に低くすることができた。これらのことから、1 台で 10 ストリーム(60Mbps)のスループット、3 台でもほぼ 30 ストリーム(180Mbps)を得ることができ、高いスケーラビリティが実現できた。

今後は、クライアント数の上限近くにおける応答時間の改善と、障害対策について検討を進める予定である。

<参考文献>

[1]小特集：ビデオオンデマンド，テレビジョン学会誌，Vol.49, No.5, pp.593-624 (1995).

[2]中村，他：分散 RAID 方式ビデオサーバー，情処研報，マルチメディア通信と分散処理，73-22, pp123-128 (1995).

[3]菊地,金田：パラレルビデオサーバ(PVS)の検討，情処第 5 2 回全国大会，pp.3-199-200 (1996).

[4]Mills, "Network Time Protocol (Version 3) specification", RFC-1305 (1992). (http://www.eecis.udel.edu/~ntp.html)