# ネットワーク型分散メディアストレージの開発

阿部哲士† 村山公保† 小林和真†

近年，MPEG-2やDV，D1，HDTVなどのストリームデータをネットワークで配信するシステムが注目されている．また，これらのストリームデータを受信者側で安価に記録させたいというニーズも高まってきている．

そこで本稿では，広帯域ストリームデータを，ネットワーク上で複数のストレージノードに分散させることで，ストレージにかかる負荷を大幅に削減し，安価にかつリアルタイムにストリームデータを記録することができるネットワーク型の分散メディアストレージシステムを提案する．また，プロトタイプシステムを実装し，評価を行った結果，既存の高価なシステムと比較して遜色のない性能を発揮できることを示す．

# Development of Netowrk Distributed Media Strage System

SATOSHI ABE ,† YUKIO MURAYAMA † and KAZUMASA KOBAYASHI†

Recently, there are several system which transmits the stream data such as MPEG-2, DV,D1, and HDTV by using the Internet. There is demand by which these stream data are stored cheaply on the local side. A very high-speed storage system is necessary for storing broadband network stream. But, the high-end storage system is very expensive.

In this paper, we propose "the Network Distribute Media Strage System". The system transmits to two or more storage nodes by splitting the broadband stream. The load of each storage node has been considerably reduced. As a result, the broadband network stream can be recorded by not using an expensive high-end storage system. We implement prototype system on the proposed model and evaluated it.

## 1. はじめに

近年，ケーブルテレビやxDSL(Digital Subscriber Line)などを利用したラストマイルの高速通信環境が整備されつつある．また，通信環境の高速化に比例し，ブロードバンドネットワークを必要とするDV[1]やMPEG-2[2]，D1，HDTVなどの映像ストリームをネットワーク配信するシステム[3)4)]が出現し，注目されてきている．これに伴い，これらの広帯域ストリームを受信者側で安価に記録したいというニーズが高まってきている．

広帯域ストリームデータをリアルタイムにディスクへ記録するには，高速なI/O性能を持つシステムが必要となる．しかし，そのようなストレージシステムは高価であり，現時点☆では安価なシステムで記録するのは難しい．

そこで本稿では，広帯域ストリームデータを，ネットワーク上で複数のストレージサーバに分散させることで，安価にかつリアルタイムに記録することができるネットワーク型の分散メディアストレージシステムを提案する．そのプロトタイプとして，実装したシステムはDVストリームをネットワークを用いて配信する「DVTS(Digital Video Transport System)」[3)4)]のストレージに対応した．本稿では，提案するネットワーク型分散メディアストレージの設計，実装とその評価について述べる．

## 2. ネットワーク型分散メディアストレージシステム

本章では，広帯域のネットワークストリームを記録するシステムを作成する際の課題を指摘し，その課題を克服したシステムを提案する．

### 2.1 メディアストレージ構築上の課題

**記録媒体に関する課題:**

ネットワークストリームを，欠落なくリアルタイムストレージするためには，ストリームの帯域に応じたディスク記憶装置が必要になる．Bonnie[5]などのベンチマークで測定したローカルディスクのI/O性能値とパケットを欠落させず，ネットワークストリームを記録できる実I/Oの帯域は異

† 倉敷芸術科学大学
Kurashiki University of Science and the Arts

☆ 西暦2001年2月

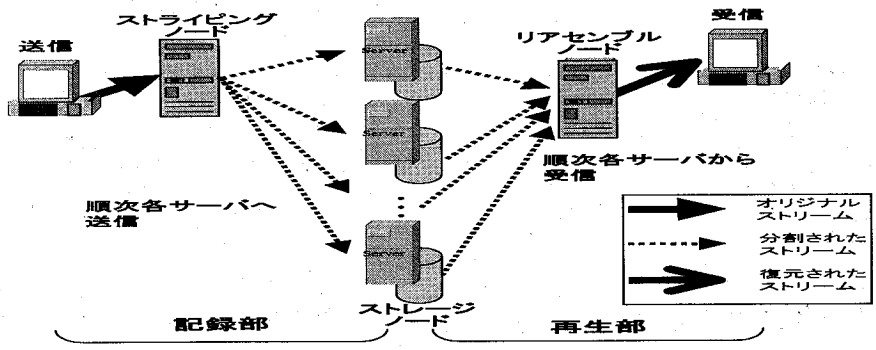


図1　設計したシステム

表1　ディスクのI/O性能と記録可能なネットワーク帯域

| ローカルディスク性能 | ネットワーク帯域 |
|---|---|
| (IDE) 20Mbit/s | 2Mbit/s |
| (ATA-33) 55.2Mbit/s | 6Mbit/s |
| (SCSI-2 RAID5) 96Mbit/s | 10Mbit/s |
| (U-SCSI RAID0) 192Mbit/s | 30Mbit/s |
| (ATA-100 RAID0) 248Mbit/s | 50Mbit/s |

なる．実験を行った結果表1に示すように，ディスク記憶装置はネットワーク帯域の約5倍から10倍のI/O性能が必要となる．

**ネットワークの再送に関する課題:**

TCP/IPネットワークにおいて，映像などのリアルタイム性が重要なストリームを配信する場合，UDPやRTP[6]などのプロトコルを用いるのが一般的である．UDPやRTPはアプリケーションにて再送処理を実装しない限り，データの再送が行われないため，受信に失敗するとそのデータは欠落する．

**ネットワークの帯域，遅延，ジッタに関する課題:**

ストレージしたデータを再送信する際には，ネットワークにデータを送信する帯域，遅延やジッタをオリジナルストリームのものとあわせる必要がある．これらをあわせなければ，受信側のバッファがあふれパケットが欠落したり，映像であれば早送りやスローになりかねず，帯域，遅延やジッタのコントロールを再送信する側で十分に行う必要がある．

## 2.2　システムの設計と実装

提案システムは，ストリームデータを分散させる「ストライピングノード」と，分散された個別のストリームを記録する複数の「ストレージノード」，分散されたデータをオリジナルストリームに復元するために結合させる「リアセンブルノード」から構成される．(図1) ネットワーク上を流れるオリジナルストリームを，異なる複数のストレージノードに一定の順序で分散送信することで，ストレージノードあたりのI/O処理の負荷を軽減し，一台の高性能サーバと同等のI/O性能を複数のサーバで実現させる．

この方式によって，たとえ安価でネットワークストリームの帯域よりも低いI/O性能のディスクやコンピュータであっても，ストレージノードの数を増やすことで，広帯域ストリームを記録できる．

以下に各ノードの実装と構成するモジュールについて述べる．(図2)

**ストライピングノード**

このノードでは，データリンク[7]からBPF[8]などを用いて受信（以下タップと呼ぶ）したオリジナルストリームをパケット毎に異なるストレージノードへ転送する．そうすることで各ストレージノードにかかる帯域は オリジナルストリームの帯域をストレージノードの数 で割った値となり，ストレージノードにかかる負荷を大幅に削減できる．

本ノードがストリームをタップするのは，ストリームを保存するだけでなく，同時に映像などを再生したい場合に対応するためである．

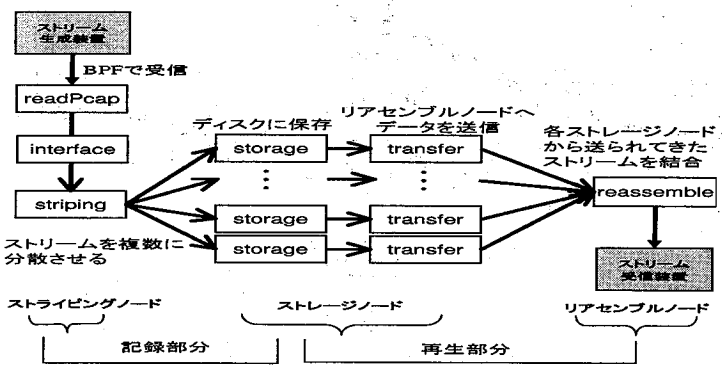


図2　モジュール図

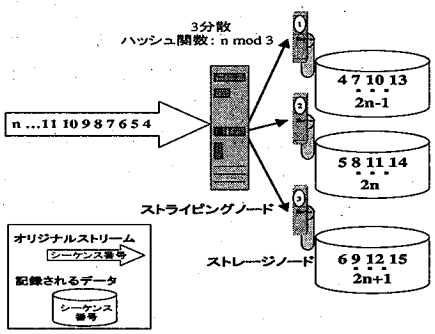


図3　ネットワークストライピング

ストリームデータを3分散する場合，ハッシュ関数を n mod 3 (n:シーケンス番号) としたとき，図3のように分散される．

ストライピングノードはreadPcap，interface，stripingの3モジュールで構成される．

**readPcapモジュール:** readPcapモジュールは，BPF(pcap ライブラリ[9]) を用いてデータリンクから直接パケットを受信している．

受信したデータはデータリンクヘッダ（Ethernet Header など)，IPヘッダ，UDPヘッダを含むため，必要なデータ部分を取り出すためにinterfaceモジュールを介す．そのときRTPヘッダは取り除かない．また，マルチキャスト[10]されたストリームにも対応できる．

また，通常 PROMISCUOUS ModeでBPFを用いると全てのパケット情報がユーザ領域へあがってくるが，BPFのKernel Filter機能を用いることで，ユーザ領域で処理しなければならないデータを最小限に抑えている．

**interfaceモジュール:** interfaceモジュールは，readPcapモジュールによって受信したデータリンクフレームから必要となるデータ部分を取り出すためのモジュールである．RTPヘッダが取り出したデータに付属していなければ，シーケンス番号やタイムスタンプなどの情報を計算し付加する．

**stripingモジュール:** このモジュールはインタフェースモジュールから渡されたデータを各ストレージノードへ順次送信する．順次送信する相手はハッシュ関数によって決める．ハッシュ関数を変えることによって性能の高いストレージノードと性能の低いストレージノードが混在する場合にそれぞれのノードに送信する割合を変えることができる．例えば，3ストリームに分散する場合に，ハッシュ関数に1,2,1,3と返すものを使用すれば，ストレージノード2や3と比べ1の割合を増やすことができる．ただしリアセンブルノードでも同一のハッシュ関数を使わなければならない．

ストレージノードとの間の通信にはRTPを用いる．RTPを用いることにより，データ送信の待ち時間を最小に抑え，オリジナルストリームのデータ欠落をできる限り抑える．

最初の実装では，この間の通信をTCPにより行っていた．TCPを用いるとストレージノードとの間は再送処理が行われる．また，ストレージノードでディスクにストリームを書き込み終わらない場合，フロー制御がかかり，送信処理のスルー

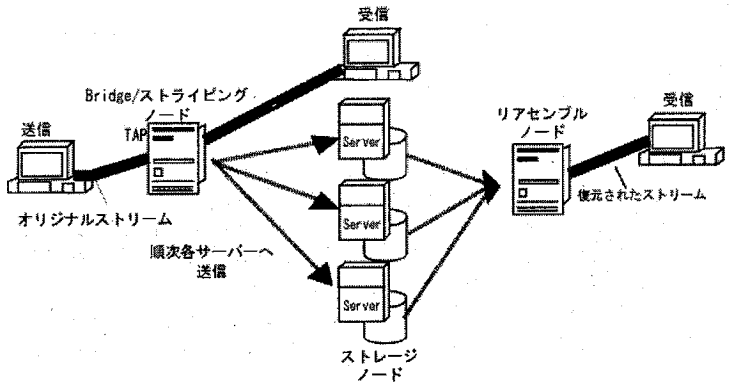


図4　ブリッジを用いたときのシステム構成図

プットが低下する．送信するデータよりも受信するデータが多くなると，いずれ受信バッファがあふれ，データの欠落が発生する．DVTSではオリジナルストリームはRTPを用いており，データの再送などの処理はおこわなれない．そこで，ストレージノードとの間をTCPではなく，送信処理に関してフロー制御やデータ再送などの処理を行わないRTPを用いることで送信処理の負荷を抑え，オリジナルストリームの受信を最優先する．また，ストレージノードとの間の再送処理は行わず，パリティによるパケット修復や，ミラーリングによる冗長構成にすることで対応する．

実装したプロトタイプシステムは，ディスクストレージでは一般的な，RAID0(ストライピング)，RAID1(ミラーリング)，RAID4(パリティによるデータ修復)の手法と同様な機能を実現した．パリティによるデータ修復は，パケット単位で行う．本ノードはFreeBSD[11]をインストールしたPCで利用できるように実装した．

ストレージノード

このノードは，ストライピングノードにより分散されたストリームを受信し，順次自らのディスクへ記録する．ディスクのI/O性能は分散されたストリームをストレージするのに十分な能力を持っていない場合にはデータが欠落する．

このノードはstorage,transferの2モジュールからなる．

**storageモジュール:** このモジュールはストライピングノードにより分散されたストリームを受信し，自分のディスクに記録する．ストライピングノードとの通信はRTPを用い，書き込みに失敗したり，受信できなかったデータは欠落する．

RTPヘッダのシーケンス番号により，パケットの欠落を検知し，ユーザに通知する．

**transferモジュール:** このモジュールは各ストレージサーバに保存されているデータをリアセンブルノード(reassembleモジュール)へ送信する．ストレージノードとリアセンブルノード間の通信はTCPを用いる．TCPのスループットを向上させるため，TCPの最大ウィンドウサイズを大きく設定した．

本ノードはPOSIX準拠のUNIX WorkStation，またはPCで構成できるように実装した．なお，異なるOSやハードウェアが混在していても問題ない．

リアセンブルノード

全てのストレージノード（transferモジュール）から送信されたデータを受信し，分散したときと同じハッシュ関数を用い，同順序になるように結合する．

このノードはreassemble,recoveryの2モジュールより構成される．

**reassembleモジュール:**

ストレージノードとの間の通信にはTCPを用い，各ストレージノードから順番にデータを受信して，結合する．そのとき，RTPヘッダのタイムスタンプ情報からオリジナルストリームの帯域，遅延，ジッタを計算する．計算された情報を元に，FreeBSDのDUMMYNET機能を用いて，帯域，遅延，ジッタ値をコントロールし，ダミーストリームをローカルホストへクロックとして送信する．reassembleモジュールはクロックデータを受け取るまでデータの送信処理をブロックする．クロックを受けると，受信したデータを送信する．これでオリジナルストリームと同じ帯域，遅延やジッタで送信できる．

表2　システムの評価に利用したノード構成

| | CPU | HDD I/O | NIC |
|---|---|---|---|
| ストライピングノード | PentiumIII (500MHz) | ******** | 300Mbps |
| ストレージノードA | PentiumIII (600MHz) | (ATA-100 RAID0) 248Mbit/s | 95Mbps |
| ストレージノードB | PentiumIII (600MHz) | (U-SCSI RAID0) 192Mbit/s | 97Mbps |
| ストレージノードC | Alpha (500MHz) | (SCSI-2 RAID5) 96Mbit/s | 200Mbps |
| ストレージノードD | Pentium MMX (200MHz) | (IDE) 20Mbit/s | 97Mbps |
| ストレージノードE | Pentium MMX (233MHz) | (IDE) 10Mbit/s | 88Mbps |
| リアセンブルノード | PentiumIII (500MHz) | ******** | 200Mbps |

ストレージノードとの間の通信にTCPを用いる理由は，フロー制御が重要になるからである．クロックによる帯域制御を行うため，クロックデータが到達するまで送信処理をブロックする．このとき，フロー制御が行われない場合，ストレージノードからデータを受け続けなければならず，リアセンブルノードのバッファがあふれ，データの欠落が生じる．また，DVTSは送信前のネゴシエーションなどは一切行わないため，ストレージノードに記録されているデータストリームを結合し，DVTSの受信側の待機しているポート番号にそのままストリームデータを送信する．

**recoveryモジュール:**　本モジュールは結合時にシーケンス番号をチェックし，欠落したデータがあると，ミラーリングストリームやパリティストリームを用いて修復を試みる．ミラーリングストリームにより修復する場合は，ミラーリングされたストリームに欠落データが含まれていれば修復される．パリティストリームによる修復では，複数に分散したストリームのうち，1ストリームのみ欠落した場合だけ修復される．

本ノードはFreeBSDをインストールしたPCで動作するように実装した．

### 2.3　システムの運用

本システムは，オリジナルストリームを受信する際，ネットワークをタップする．このため，レイヤ2スイッチなどによってフレーム伝搬が抑制されると，ストリームの受信ができなくなる．その場合には，スイッチのポートミラーリング機能を用いる方法や，スイッチをハブに変える方法，スイッチと映像受信するコンピュータの間にストライピングノードを置き，図4のように，ストライピングノードをブリッジとして使う方法がある．

ストライピングノードをブリッジとして使用すると，ストライピングによる処理の他に，IP-Forwardingによる処理が必要となり，ストライピングノードにかかる負荷が増え，CPUやNICに余裕がなければデータの欠落などが発生する可能性がある．

表3　分散数により必要となるディスクの性能

| 分散数＼帯域 | 90Mbps | 60Mbps | 30Mbps |
|---|---|---|---|
| 分散なし | ******* | ******* | 192Mbit/s |
| 3分散 | 192Mbit/s | 192Mbit/s | 96Mbit/s |
| 6分散 | 192Mbit/s | 96Mbit/s | 55.2Mbit/s |
| 9分散 | 96Mbit/s | 96Mbit/s | 55.2Mbit/s |
| 12分散 | 96Mbit/s | 55.2Mbit/s | 55.2Mbit/s |
| 15分散 | 55.2Mbit/s | 55.2Mbit/s | 20Mbit/s |
| 100分散 | 20Mbit/s | 10Mbit/s | 10Mbit/s |

表4　分散数と帯域によるストライピング処理の失敗率

| 分散数＼帯域 | 120Mbps | 100Mbps | 50Mbps | 30Mbps |
|---|---|---|---|---|
| 分散なし | 2.943 | 0.0032 | 0 | 0 |
| 3分散 | 2.642 | 0.0031 | 0 | 0 |
| 6分散 | 3.232 | 0.0032 | 0 | 0 |
| 9分散 | 2.146 | 0.0033 | 0 | 0 |
| 12分散 | 2.598 | 0.003 | 0 | 0 |
| 15分散 | 3.313 | 0.013 | 0 | 0 |
| 100分散 | 2.244 | 0.0131 | 0 | 0 |

## 3.　システムの評価

この章では，本研究にて開発したシステムを用いて行った様々な実験や性能評価などの結果を示す．

本システムの評価は表2に示す構成で行った．

### 3.1　ストリーム分散性能

本章ではストライピングノードのストリーム分散性能を測定した結果を示す．分散相手を決めるハッシュ関数はストレージノードをラウンドロビンで順番に選択するものを利用した．

様々な帯域のオリジナルストリームをストライピングし，ストレージノードA～Eを用いて記録した．その結果表3に示したように，分散数を上げる毎に最低限必要となるディスクの性能が下がっていることが確認できる．

また，ストライピング処理によるデータ欠落の割合(分散前帯域/(分散数×分散後帯域))×100「単位%」を表4に示す．評価したシステムの性能で，100Mbps

表5　ストリーム結合能力

| 分散数＼帯域 | 100Mbps | 60Mbps | 30Mbps | 10Mbps |
|---|---|---|---|---|
| 分散なし | × | △ | ○ | ○ |
| 3分散 | △ | ○ | ○ | ○ |
| 6分散 | △ | ○ | ○ | ○ |
| 9分散 | △ | ○ | ○ | ○ |
| 12分散 | △ | ○ | ○ | ○ |
| 15分散 | △ | △ | ○ | ○ |

○:十分な帯域に達した
△:-10%の範囲
×:-10%以上下回った

程度のストリームを分散できる．また，分散数を変更しても，データ欠落率が大きく変わらないことから，分散数はシステムの負荷となる大きな要因ではないことが分かる．

### 3.2　ストリーム結合能力

様々な帯域のストリームを1分散から15分散し，リアセンブルノードの能力を測定した．その結果を表5に示す．約60Mbpsのストリームであれば結合後の帯域や遅延，ジッタ が正常に復元できることが分かった．また，ストレージノードのディスクのI/O性能，バスの速度などが原因で，求められる帯域に達しないことがある．1ノードでも求められる帯域に達しないストレージノードが存在すると，そのノードがボトルネックとなり，システム全体のパフォーマンスが低下する．

評価実験の結果を見る限り，結合後の帯域が高ければ高いほど復元に失敗することが分かる．

### 3.3　総合評価

提案したシステムを評価した結果，1ストレージノードあたりの負荷の軽減を確認し，複数のストレージノードにてオリジナルストリームを分散ストレージすることができた．また，複数に分散されたストリームをリアセンブルし，復元できた．

本システムを用いて，DVTSによりネットワーク配信された，DVストリームを記録した．その結果，約30Mbpsのストリームを5ストリームに分散し，正常に記録，ミラーリング，再生を行うことができた．また，そのときの欠落したデータの割合はシステム全体で0.0002%程度であった．

これらのことから，本システムは既存の高性能ストレージサーバと同等のI/O性能を実現させていると言える．

## 4. おわりに

本稿では，ネットワークを用いたDVやMPEG2配信などの広帯域ストリームをストレージするシステムを作成する際の課題を指摘し，それを解決するシステムとして，ネットワーク上を流れるオリジナルストリームデータを，異なる複数のストレージノードへ一定の順序で分散送信することで，ストレージノードあたりのI/O処理の負荷を軽減し，1台の高性能サーバと同等のI/O性能を実現させるシステムを提案した．

また，本システムのプロトタイプを実装し，評価した結果本システムは実用的であり，十分なネットワーク帯域が確保されるLANなどの環境であれば，I/O性能，価格，冗長性などにおいて，既存のシステムと比較して遜色ない性能を発揮することが可能であることが実証された．

謝辞　本研究に関して共同研究の機会をくださった電通国際情報サービスの熊谷氏に感謝いたします．

## 参考文献

1) “Specifications Consumer-Use Digital VCR's using 6.3mm magnetic tape”, HD Digital VCR Conference, 1994 society, 1995
2) “MPEG Home Page”, http://drogo.cselt.stet.it/mpeg 1999
3) “DV Stream on IEEE1394 Encapsulated into IP” http://www.sfc.wide.ad.jp/DVTS/, 小川 晃通 ,1999
4) “Design and Implementation of DV Stream over Internet”, IWS99, A.Ogawa K.Kobayashi K.Sugiura O.Nakamura J.Murai, 1999
5) “Bonnie home page”, http://www.textuality.com/bonnie/ , 2000
6) “RTP: A Transport Protocol for Real-Time Applications” “RFC 1889”, H. Schulzrinne,S. Casner,R. Frederick,V. Jacobson , 1996
7) “TCP/IP Illustrated, Volume2”, Gray R. Wright, W. Richard Stevens , 1997
8) “UNIXネットワークプログラミング第2版 Vol.1”, W.リチャード・スティーブンス著, 篠田陽一訳, 1998
9) “pcap manual”, Van Jacobson, Craig Leres ,Steven McCanne, 1998
10) “INTERNETWORKING WITH TCP/IP VOLUME II”, Douglas E. COMER, David L. Stevens, 1991
11) “FreeBSD Project (Japan)”, http://www.jp.freebsd.org , 2000