

## クライアント主導型によるインターネット映像配信技術: SoftwareVision

秦泉寺 浩史 高田 久靖 笠原 久嗣

{jinzen, tkd, kasahara}@nttvdt.hil.ntt.co.jp (<http://www.hil.ntt.co.jp/SoftwareVision/>)

NTT ヒューマンインタフェース研究所  
〒239 神奈川県横須賀市光の丘 1-1

あらまし: SoftwareVision は、インターネットならびにイントラネットにおいて実時間映像配信を実現するインターネット映像配信システムである。PC ならびに WS にソフトウェアをインストールすることにより、ライブ映像の配信やビデオオンデマンドシステムを構築することができる。SoftwareVision は、インターネット実時間配信プロトコル Real-time Stream Transport Protocol (RSTP)を用いて実現されている。RSTP は、サーバ、ネットワーク、クライアントの状況に応じて転送レートを制御し、映像のスケラブル配信を実現するクライアント主導型のプロトコルである。SoftwareVision は、ITU-T H.323 や Mbone ツールをエンコーダとして使い、そのストリームを配信することも可能である。

## SoftwareVision: A Scalable Video Distribution Technique for the Internet

Hiroshi Jinzenji, Hisayasu Takada and Hisashi Kasahara

{jinzen, tkd, kasahara}@nttvdt.hil.ntt.co.jp (<http://www.hil.ntt.co.jp/SoftwareVision/>)

NTT Human Interface Laboratories  
1-1 Hikarinooka, Yokosuka-Shi, Kanagawa 239 Japan

**ABSTRACT:** SoftwareVision is a technology that makes real-time video distribution possible, either over the Internet or over an intranet. Simply by installing software on PCs and workstations, it enables users to obtain real-time video distribution as well as VOD services. SoftwareVision is currently implemented on the Real-time Stream Transport Protocol (RSTP). RSTP can change data stream rate dynamically, depending on the situations of server, network and clients. RSTP achieves a scaleable video transmission. Mbone tools, such as vic and vat, and ITU-T H.323 terminals are used for the SoftwareVision encoder.

## 1. はじめに

我々は、クライアントにソフトウェアデコーダを用いたインターネット対応 VOD システムの検討を進めてきた<sup>[1][2]</sup>。SoftwareVision は、インターネット上で映像や音声などの実時間メディアをサーバ、ネットワーク、クライアントの状況に応じて適応的に配信することのできる 1 対 n のインターネット映像配信システムである。現在、各種イベント中継<sup>[3]</sup>や、蓄積映像配信サービスに広く利用されている。本稿では、SoftwareVision が利用し、スケラブル映像配信を実現するインターネット実時間配信プロトコル Real-time Stream Transport Protocol (RSTP) の制御方式について述べるとともに、SoftwareVision と他の映像通信システムとのインターネット、SoftwareVision のインターネットにおける性能評価について報告する。

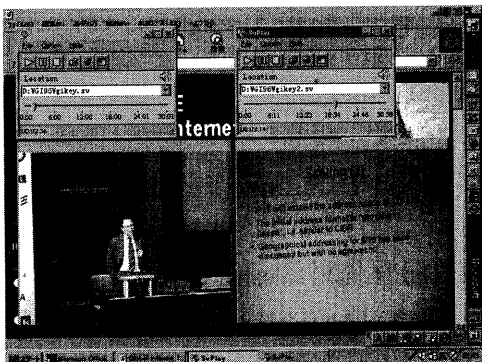


写真 SoftwareVision のクライアント動作状況

## 2. インターネット実時間配信プロトコル

インターネットにおける実時間配信プロトコルを考えるために、問題点の整理ならびに方式検討を行い、SoftwareVision で用いている配信プロトコル Real-Time Stream Transport Protocol の詳細について述べる。

### 2.1 インターネット実時間配信の問題点

インターネット上で映像音声の実時間配信システムを構築する際の問題点は、実時間ストリームを配信するサーバ、ストリームが配信されるネッ

トワーク、配信されたストリームの受信ならびにユーザへの提示を行うクライアントのシステムを構成する3つの構成要素に分けて考えなければならない。サーバならびにクライアントは、多くの場合インターネットに接続される PC や WS のプログラムとして実現されるため、個々のマシンの処理能力や同時に動いている他のプログラムの動作状況によって処理能力が影響を受けることも考慮しなければならない。各3つの構成要素毎に、解決すべき問題点を整理する。

#### ●サーバ

サーバ側は、ネットワークを介してクライアントに実時間ストリームを絶え間なく供給しなければならない。サーバ側での問題は、多数のクライアントに配信しなければならない状況下において、サーバが過負荷の状態になり、クライアントに対しての配信が滞ることである。

#### ●ネットワーク

インターネットは、複数のネットワークが相互に接続し互いに経路を共有しているため、End-to-End の帯域確保は現状極めて困難である。実時間ストリームは、伝送中ほぼ一定の帯域を必要とし、伝送遅延に対する許容値が小さいために、経路上のトラフィックの変化によって影響を被り易い。またクライアントが直接接続されるネットワークが、モデムや ISDN を用いたダイアルアップ型から、LAN 経由の場合と様々な状況があり、それぞれの状況に対応しなければならない。

#### ●クライアント

インターネットで実時間映像音声配信サービスを行う場合、クライアントは多くの環境で動作させるためにソフトウェアで実現されることが一般的である。クライアントに関する問題は、クライアントが動作するマシンによって処理能力が異なることや、同じマシン内で動作している他のプログラムからクライアントが影響を受けることから生じる。サーバからのデータの取りこぼしや処理能力不足による復号遅延や復号エラーが発生する。

それぞれの構成要素における問題点を示したが、いずれの問題点もその影響がシステムの内部だけでなく周辺の状況によって時々刻々と変化する。インターネット上における実時間配信プロトコルを考える上で、これらの状況に動的に対応できるプロトコルであることが求められる。以降、実時間配信プロトコルについて、サーバとクライアント間の制御方式から見たシステム構成の検討と、ネットワークの帯域を多く消費する映像の符号化方式に関する検討を行う。

### 2.1.1 システム構成から見た検討

実時間ストリームを配信するサーバ・クライアント型システムは、サーバからクライアントへ映像音声の実時間再生に必要なデータを継続的に供給しなければならない。よって、長時間ネットワークの帯域を占有し、データ転送を行う形となる。前項の考察から、実時間配信プロトコルは、システムを構成する3つの構成要素が被る周囲の影響に対して、適応的にかつ安定した転送を行わなければならない。ここでは、配信するストリームの転送制御に着目し、サーバ主導型とクライアント主導型について検討する。

#### ●サーバ主導型

配信するストリームの転送レートは、サーバ側が制御を行う。図1に示すような形態で、MPEG2等を利用した広帯域で遅延が許されないようなビデオオンデマンドシステムに用いられている。サーバ側においてネットワークならびにクライアントの状況をモニタすることは困難なので、サーバ・クライアント間の接続が確立後、ネゴシエーションにおいて配信に利用するビットレートを決定し、その範囲内で実時間ストリームの配信を行う。制御自身は単純であるが、配信中の状況の変化には適応しづらい。また、サーバにストリームのレート制御が集中するため、クライアント数が増加した場合、サーバの負荷が高くなりサーバの動作が不安定になる可能性がある。通常は、サーバに同時接続するクライアントの台数を制限することで問題を回避する。一般的に、サーバ・クライアント間の

ネットワーク上で利用可能な帯域の変動が少なく、サーバならびにクライアントの能力が均質な場合に適していると考えられる。

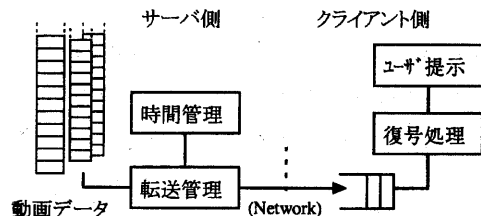


図1 サーバ主導型システム

#### ●クライアント主導型

配信ストリームのレート制御は、クライアント側から行う形態である。図2に示すように、サーバ主導型と異なりクライアント側からストリーム配信中にサーバに対して逐次指示を出すことによりレート制御を行う。クライアントは、要求したストリームの到着状況を確認することによってサーバならびにネットワークの状況を把握することができる。クライアント自身の処理状況も確認できることから、クライアントはシステムの状況を一元的に把握することができる。クライアント主導型は、状況に応じて適応的に制御を行うことができる反面、制御の内容は複雑になる。また、クライアントからの制御は状況確認後にネットワークを介してサーバに伝えられるので、広帯域サービスのような遅延に厳しいシステムの場合は適さない。

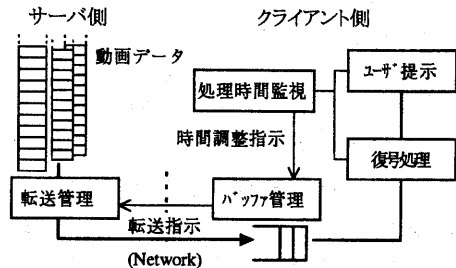


図2 クライアント主導型システム

### 2.1.2 映像符号化方式からの検討

ネットワークで配信される映像音声は、通常はなんらかの圧縮符号化が施された後に配信される。音

声は、通常固定レートで出力される符号化方式を用いる。圧縮率は、符号化レートが 10kbps 以下というモデムによる送受が十分可能な方式から利用できる。また、音声メディアの特性として、データの欠落は同時に情報の欠落となり、音飛びや無音区間が発生するが、その後の復号処理に大きな影響を与えることは少ない。よって、インターネットで音声配信を行う場合、安定してサーバからクライアントまで配信できることが可能な帯域を想定し、その帯域以下のレートの符号化方式を利用する。

映像の場合、もともとデータ量が大きいため、インターネットで配信するためにはかなりの高圧縮符号化方式を用いなければならない。現状のインターネットにおいて、映像配信に割り当てられる現実的なビットレートは、コンディションの良い場合でも 100kbps 程度が上限と考えられる。また、多くのユーザがモデムや ISDN を利用したダイヤルアップ接続を利用していることから、利用できるビットレートはさらに低くなる。一方、映像情報の特徴として、音声データと異なりデータの一部を欠落させても情報の意味を大きく損なわない場合がある。状況に応じてサーバからの映像データの供給量を制限し、表示フレームレートを下げることができれば、映像情報としての意味を大きく損なわずに転送レートを抑える効果がある。また同時に単一のストリームを用いて高いレートから低いレートまでのスケーラブル性を実現することもできる。以下、映像の高圧縮符号化方式の中で広く使われているフレーム間差分画像符号化方式を用いたスケーラブル制御について検討する。

フレーム間差分画像符号化方式は、ITU-T H.261、ITU-T H.263、MPEG 等のインターネット映像配信でもよく利用される符号化方式のベースとなる符号化方式である。まず、図3に示すように符号化ストリームをパケット化し、信頼性のないUDPトランスポート等で配信を行なった場合を考える。フレーム間差分符号化方式によって生成された符号化データは、図中の I で示されているフレーム内符号化データと、その他のフレーム間

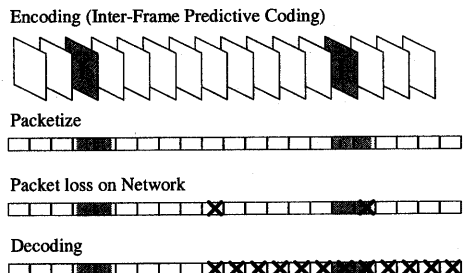


図3 制御のないインターネット映像配信の例

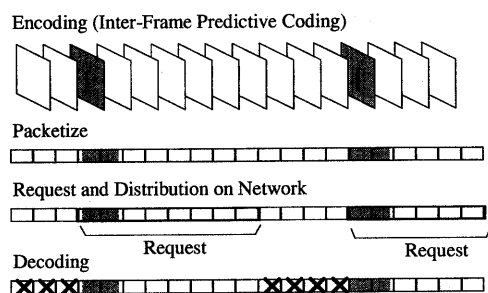


図4 制御下のインターネット映像配信の例

差分符号化データからなる。フレーム内符号化データは、他のフレームの情報なしに復号できる反面、フレーム間差分符号化データと比較して情報量が大きくなり圧縮効率を高められないため、あまり多く挿入することはできない。フレーム間差分符号化データは、必ず関連するフレームの情報が復号の際に必要となるが、情報量は小さくなる。インターネットで配信を行なう場合、パケット単位という比較的大きなデータをパケット廃棄で失うことがあるため、少しのパケット落ちが大きなデコードエラーを引き起こす。符号化データが欠落すると、それ以降のフレーム間差分符号化データは復号できなくなり、次に現れるフレーム内符号化データまですべてのデータが無効データとなる。

インターネットでフレーム間差分符号化方式のデータを配信する際に重要なことは、フレーム内符号化データから引き続きフレーム間差分符号化データを連続した形でパケット廃棄無しに配信することである。図4に、その例を示す。サーバからクライアントに配信するデータは、必ずフレーム内符号化方式データを先頭とし、それに引き続きフ

フレーム間差分符号化データをまとめたものを信頼性のあるトランスポートを用いて配信する。サーバ、ネットワーク、ならびにクライアントの状況が悪い場合は、2つのフレーム内符号化データに挟まれたフレーム間符号化データの配信量を減らし、状況が良い場合には、すべてのデータを受信するようにする。この方式によって、簡易な単一符号化ストリームからスケーラブル制御施した映像配信を実現することができる。この方式を実現する際に、符号化データストリームの中に周期的にフレーム内差分符号化データが含まれていると効果的に機能する。

## 2.2 Real-Time Stream Transport Protocol

SoftwareVision は、インターネット実時間配信プロトコルとして RSTP(Real-Time Stream Transport Protocol)を利用している。RSTP は、前項による検討に基づき、実時間性よりも信頼性を重視し、以下のような特徴を有する。

- クライアント主導型の配信プロトコル
  - スケーラブル性を有する映像配信プロトコル
  - 状況に応じた適応的な制御方式
  - 原則的に信頼性のあるトランスポートの使用
- 以下、その詳細を示す。

### 2.2.1 RSTP の概要

RSTP は、映像音声を共に扱う実時間配信プロトコルであるが、特にスケーラブル性の期待できる映像配信制御に関してその威力を発揮する。RSTP の基本的な考え方は、配信するデータの量をクライアントがその時点の状況に応じて決定し、処理可能な最小限のデータを確実に配信することである。図5に、RSTP で用いる優先度付き

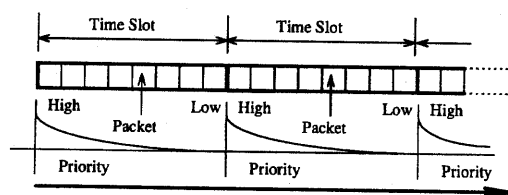


図5 優先度付きストリームフォーマット

ストリームフォーマットを示す。この優先度付きストリームフォーマットは、以下の4つの条件を満たしている。

1. 優先度付きストリームデータは、図6に示すようにあらかじめ決められた周期のタイムスロットで区切られている。
2. タイムスロットに区切られた優先度付きストリームデータは、配信するパケット単位に分割されている。
3. タイムスロット内のパケットは、優先度付けがなされ、タイムスロットの先頭から優先度が高い順に並べられている。
4. 分割されたパケットは、それぞれ処理されるべき時間を表すタイムスタンプが記録される。

クライアントからの要求は、まず処理を開始したいタイムスタンプ上の時間で最も近いタイムスロットを指定し、その先頭に配置された最も優先度の高いパケットから順に連続するパケットを個数によってサーバに指示する。クライアントは、サーバから配信されたパケットを処理していく際に、パケット毎に記録されているタイムスタンプを確認し、サーバから配信される際のネットワーク遅延、ならびにクライアントの復号時における処理遅延を算出する。クライアントは、あらかじめ許容される遅延の値が決められており、その範囲内であれば引き続き前に要求したパケットに連続するパケットの配信を要求する。処理中に許容される遅延の値を上回った場合、遅延を補正した正しい時間以降に出現する最も優先度の高いパケットから要求する。一度に要求する個数や要求を出す間隔等については、クライアントの実装に任せるものとしてRSTP においては詳細を規定しない。

具体的な利用例として、SoftwareVision が用いているフレーム間差分符号化方式 ITU-T H.261 の場合を示す。H.261 の場合、仕様の中にフレーム内符号化データに関する規定がないため、SoftwareVision ではフレーム内符号化データとして1フレーム内に存在すべきすべてのマクロブロック(MB)が該当フレームの符号化デー

タ内に存在し、すべてのMBがイントラモードで符号化されている場合としている。符号化装置には、このようなデータを周期的に生成できる機能を有したものを利用している。タイムスロットは通常5秒としており、タイムスロットの先頭パケットにはフレーム内符号化データの先頭が必ず入っている。H.261 を利用する場合、タイムスロット内のパケットは、号化された時間の順序で並んでおり、パケットの並べ替えは行なわない。優先度は、タイムスロットの中でパケットの並ぶ順に低くなっている。クライアントで許容される処理遅延は3秒である。

## 2.2.2 RSTP の詳細

RSTP は、1つのTCPコネクション上でサーバクライアント間のネゴシエーション、クライアントからのパケット要求、サーバからのデータパケット配信をおこなうことを基本としている。例外として、モデム等の低速リンクでクライアントが接続されている場合、サーバからの音声データの配信にUDPトランスポートを利用する。これは、音声データの場合、データの復号時にパケット間の相関がなく、信頼性のある配信よりも実時間性が重視されるからである。またRSTPのマルチキャスト配信においては、IPマルチキャストを利用するためUDPトランスポートに準じた形となる。RSTPの動作シーケンスは、クライアントからサーバに配信してもらうコンテンツを指定するネゴシエーションモードと、実際にストリームの配信を行なうデータ転送モードに分けられる。

### 1) ネゴシエーションモード

ネゴシエーションモードは、電子メールのSMTPやPOP、ならびにWWWのHTTPと同様に、テキストモードでコマンドのやり取りを行う。これは、コンテンツ確認を簡易に行うだけでなく、RSTPで定義されないコマンドを用いてサーバ管理を行うためである。ネゴシエーションモードで使われるクライアント-サーバ間の基本的なコマンドを示す。

- サーバ接続確認メッセージ  
+OK RSTP/X.X/  
\*X.Xはバージョン番号

- コンテンツ指定コマンド  
REQ RSTP://{host}:{port}/{content}  
host: サーバのアドレス  
port: TCPポート番号  
content: サーバ内のコンテンツ名
- サーバからのACKメッセージ  
+OK {付加メッセージ}
- サーバからのNACKメッセージ  
-NG {エラー状況メッセージ}
- データ転送モード遷移コマンド  
BIN
- UDP使用データ転送モード遷移コマンド  
UDP {port}
- サーバからのデータ転送モード遷移ACK  
BIN

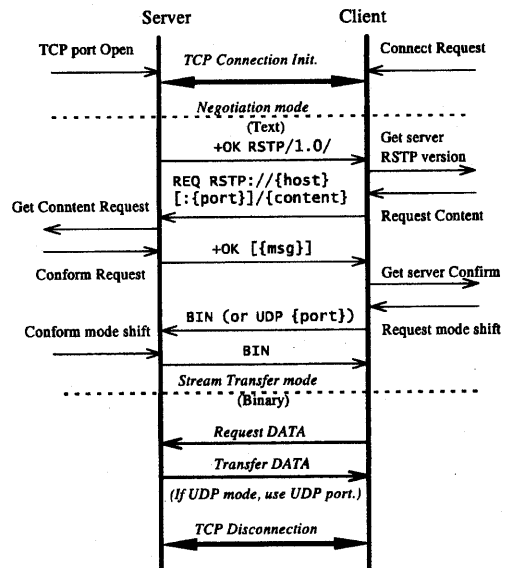


図6 RSTP動作シーケンス

### 2) データ転送モード

データ転送モードは、ネゴシエーションモードと異なり、バイナリモードでコマンドならびにデータ転送を行う。これは、ネゴシエーションモードと異なり、長時間にわたりデータ転送が継続するため、処理速度ならびに転送効率が重要となるからである。データ転送モードで使われるコマンドを以

下に示す。

- 時間指定高優先度パケット要求 (6 bytes)  
パラメータ: 要求時間、パケット数
- 時間指定パケット要求 (6 bytes)  
パラメータ: 要求時間、パケット数  
(音声等の優先度が無い場合)
- 前回要求からの連続パケット要求 (2 bytes)  
パラメータ: パケット数
- データ転送の一時停止要求 (1 byte)
- データ転送の終了要求 (1 byte)  
(クライアントならびにサーバの双方)

サーバからクライアントに配信されるデータパケットには、符号化ストリームのデータ以外に 9 バイトのヘッダが付けられる。ヘッダは、コマンド ID(1 byte)、データサイズ(2 bytes)、タイムスタンプ(4 bytes)、メディア ID(1 byte)、CODEC パラメータ(1 byte) から構成されている。

### 3. SoftwareVision

SoftwareVision は、RSTP を用いるインターネット映像配信システムである。

#### 3.1 システム概要

SoftwareVision のシステム構成を、図7に示す。実時間コンテンツを中心に見ていくと、発信すべきコンテンツを持つサーバを、発信サーバ(Originating Server)とする。サーバへのアクセスを分散し、同一経路上での重複伝送を避けるためインターネット上に配置するサーバを中継サーバ(Relay Server)とする。クライアントから配信要求を待ち受けるサーバを、配信サーバ(Distribution Server)とする。各サーバ間の配

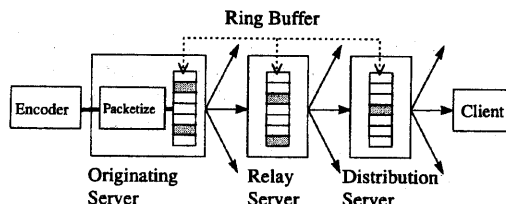


図7 SoftwareVision のシステム構成

信プロトコルは RSTP を用い、機能的には発信サーバが発信コンテンツを有することを除けば、発信サーバ、中継サーバ、配信サーバはクライアントから見て同一に振る舞う。各サーバは、各配信ストリームに対して 30 秒程度に相当するパケットリングバッファを確保し、伝送ならびに処理遅延による配信時の時間的揺れを吸収する。蓄積コンテンツは、どのサーバにも配置することができる。

映像のエンコーディングには、LAN 用映像通信装置<sup>4)</sup>を用いている。LAN 用映像通信装置は、周期的にフレーム内符号化データを生成することが可能である。音声のエンコーディング、各サーバ、ならびにクライアントは、PC と WS 上で動作するソフトウェアによって実現されている。音声の符号化方式は、複数の方式の中から選択できるが GSM(13.2 kbps) が主に利用されている。

#### 3.2 他システムとのインターワーク

SoftwareVision システムは、広く普及している他のインターネット用映像音声配信ツールとインターワークする機能を有している。図8に示すように、LAN 用映像通信装置の他に、ITU-T H.323 端末ならびに Mbone ツールの符号化ストリームを SoftwareVision のストリームとして扱うことができる。それぞれの実時間配信プロトコルには RTP<sup>5)</sup> が用いられており、簡単に RSTP 形式に変換できるので、そのままの形式か簡単なトランスコーディングによって利用可能である。

##### ● ITU-T H.323 端末とのインターワーク

H.323 は、現在広く普及しつつあるインターネットテレビ電話の国際標準プロトコルである。SoftwareVision のサーバは、H.323 プロトコルを終端する機能を有し、H.323 端末からの接続を待ち受ける。H.323 端末と接続後、受信したストリームを RSTP 形式に変換する。

##### ● Mbone Tools とのインターワーク

Mbone<sup>6)</sup> は、インターネット上で IP マルチキャストの実験を行なっている仮想ネットワークである。

MBone では、IP マルチキャストを用いた多地点実時間会議ソフトが広く普及しており、セッション管理プログラム *sdr*、映像通信プログラム *vic*、音声通信プログラム *vat* が広く利用されている。SoftwareVision のサーバは、指定された端末からの IP マルチキャストストリームを受信し、RSTP 形式に変換する。

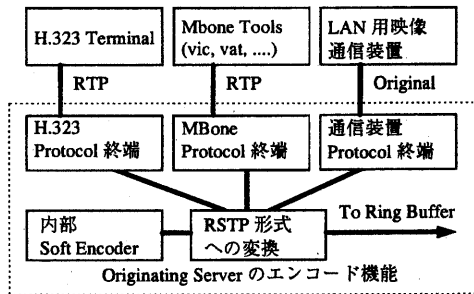


図 8 SoftwareVision と他システムとの  
インターワーク

#### 4. インターネット上の性能評価

SoftwareVision のパフォーマンスを、現実のインターネット上で評価した。評価は、同一コンテンツをサーバと同じ LAN、ISDN 64kbps 同期接続ならびに 28.8 kbps モデムの PPP 接続 の3つの環境下でクライアントを動作させ行った。H.261 の符号化には、LAN 用映像通信装置を用い、5秒に1度フレーム内符号化データが挿入されるようにした。クライアントは、Windows95 が動作する Pentium166MHz のマシンを用いた。評価に用いたストリームを以下に示す。

- Video1: 100k bps CIF H.261 + GSM audio
- Video2: 50k bps QCIF H.261+ GSM audio  
(Video1&2 は動きの激しい祭りのビデオ)
- Video3: 100k bps CIF H.261 + GSM audio  
(Video3 は動きの少ない学術講演のビデオ)

表1に評価の結果を示す。RSTP を用いることによって、スケーラブル制御が効果的に働き、モデム経由の場合でも、音声に半分近い帯域を占有されながら、タイムスロットあたり CIF の場合は1フレーム以上、QCIF の場合は 2.5 フレーム近く

表示された。表示されたフレームは、高いレートで符号化されたもので、細かい部分までクリアな輪郭のはっきりした画像であった。

表1 SoftwareVision の性能評価

	フレーム/秒 (fps)	フレーム/TS (TS = 5sec)
<b>Video 1</b>		
LAN	2.425	12.123
ISDN 64kbps	0.743	3.716
PSTN 28.8kbps	0.201	1.007
<b>Video 2</b>		
LAN	4.243	20.527
ISDN 64kbps	1.881	9.407
PSTN 28.8kbps	0.484	2.422
<b>Video 3</b>		
LAN	5.380	26.898
ISDN 64kbps	3.135	15.673
PSTN 28.8kbps	0.255	1.275

#### 5. まとめ

SoftwareVision は、クライアント主導型のインターネット実時間配信プロトコル RTSP を用いてスケーラブル制御を有した映像配信を実現した。単一のストリームを用いて、LAN 接続からモデムによるダイヤルアップ接続まで状況に応じて適応的に動作する。また、インターネットの汎用映像通信ツールとインターワークし、エンコーダとして利用する機能も有する。

#### 参考文献

- [1] 秦泉寺, 田尻: "分散適応型VODシステムの一検討", 1995年秋 信学全大 D-81
- [2] 秦泉寺, 田尻: "分散適応型VODシステムの性能評価", 1996年春 信学全大 D-324
- [3] H. Jinzenji and K. Hagishima: "Real-Time Audio and Video Transmission of IEEE GLOBECOM'96 over the Internet Using New Software", IEEE Communications Mag., April 1997, pp34-38
- [4] H. Jinzenji, S. Azegami and T. Tajiri, "Audio and Video Communication System for the Internet": F2, Packet Video 94, Portland, Sep 1994.
- [5] H. Schulzrinne et al.: "RTP: A Transport Protocol for Real-Time Applications", 1889 RFC, Jan. 25 1996
- [6] Steve Casner: "Frequency Asked Questions on the Multicast Backbone (MBONE)", <http://www-mice.cs.ucl.ac.uk/mice/faq.html>