

## 画面レイアウト共有可能な複数映像受信システムの設計

安部 光一<sup>†1</sup> 前田 香織<sup>†1</sup> 井上 博之<sup>†1</sup>  
近堂 徹<sup>†2</sup> 鈴木 徹<sup>†1</sup>

IP 通信網では、複数の視点からのマルチアングルストリームの複数伝送や視聴者の属性や嗜好などに応じた推奨映像を配信するなど映像配信の高度化が進んでいるが、映像の視聴画面は固定的なものが多く、依然受動的な視聴が多い。そこで、本論文では映像を受信する視聴者側の視点から視聴画面や機能の高度化について焦点をあてる。具体的には映像画面の自由なレイアウト、他の視聴者の視聴画面を自身の視聴画面のレイアウトと同期させるスクリーンシェア機能、視聴者自身の映像や音声の送受信機構を備えて、視聴者間のインタラクティブなコミュニケーションを可能とし、視聴体験を共有するパブリックビューイング機能などを持つ映像受信システムの設計とプロトタイプシステムの実装について述べる。加えて、複数画面の同時再生やスクリーンシェア機能のオーバヘッドに関する実験や理論的考察から、採用した機能の実用性を示す。

### A Design of a Receiving System of Multiple Streams Sharing Layout of Video Windows

KOICHI ABE,<sup>†1</sup> KAORI MAEDA,<sup>†1</sup> HIROYUKI INOUE,<sup>†1</sup>  
TOHRU KONDO<sup>†2</sup> and TOHRU SUZUKI<sup>†1</sup>

In recent years, a rich video streaming service that allows multi-angle videos and personalized videos based on user's preference design has become increased in IP network. Nevertheless, in many cases a receiver can not arrange video windows of received streams freely because of a static layout design. To address this problem, we propose a receiving system which is sophisticated from the viewpoint of a receiver. It enables following functions: a receiver-driven flexible layout function of video windows, a screen sharing function which is mutually synchronized among receivers, and a public viewing function with interactive communication between receivers. In this paper, we describe the system design and its implementation, and we show the effectiveness of the proposed functions

through the performance experimentations and considerations.

#### 1. はじめに

インターネットの広帯域化にともない、映像配信サービスの開発が進み、真に通信と放送の融合サービスが展開されるようになってきた。すでに、地上デジタル放送を IP で再送信する IPTV サービスも提供されている<sup>1)</sup>。インターネット網を用いた映像配信は、ビデオオンデマンド (VOD)、多チャンネル放送、映像品質の選択など多様なサービスの可能性を持つ。しかし、いまだその可能性を模索する時期にある IPTV サービスはさらなる機能の高度化が求められており、たとえば、多くの映像コンテンツから視聴者の属性や嗜好によって薦めるパーソナライズ型の映像配信に関する技術開発が進められている<sup>2)</sup>。

一方、マルチソース形式の映像配信サービス、マルチアングル放送が商用サービス化されている<sup>3)</sup>。マルチアングル放送とは、複数のカメラで撮影した異なるアングルの映像をそれぞれ配信することで、視聴者は見たい映像を任意に選択したり、同時に複数アングルの映像を視聴したりできる。たとえば、野球中継の場合、視聴者はバッターに注目したいときは、つねにバッターを追う映像を視聴し、投手に注目したいときは投手の映像を視聴するといったように、好きなタイミングで任意の映像を選択して視聴することができる。文献 4) では、与えられたキーワードにより複数映像から任意の映像を自動で切り替え、1つのストリームを生成することで各個人にカスタマイズされた映像の配信を可能にしている。

このようにマルチソース形式の映像配信は視聴者の映像コンテンツの選択肢を大幅に増やすことができ、視聴者固有の要望や属性にあったパーソナライズ型の映像配信とともに普及が期待される。しかし、現在のパーソナライズ型の映像配信は映像コンテンツの選択にのみ焦点があたっているため、視聴画面のレイアウトは固定的なものが多い。また、視聴者自身が映像ソースとして、他の視聴者の視聴画面に登場するなど、視聴者がインタラクティブに参加できる視聴環境が提供されているものもほとんどない。

そこで、本論文では視聴者自身が視聴環境を創るための技術について焦点をあて、自由な

<sup>†1</sup> 広島市立大学大学院情報科学研究科  
Graduate School of Information Sciences, Hiroshima City University

<sup>†2</sup> 広島大学情報メディア教育研究センター  
Information Media Center, Hiroshima University

画面レイアウト作成や視聴者間のインタラクティブなコミュニケーションを実現するような視聴環境をカスタマイズできるシステムを提案する。これにより、たとえば、スポーツ中継やライブコンサートなどの映像配信イベントにおいて、自分以外の視聴者の歓声や声援を視聴し、視聴体験の共有による相乗的な効果を得ることで臨場感が高まるなど、今後のサイバー社会に新たな視聴環境を提供することを目指す。

以下、2章では、視聴環境の高度化の要素について述べる。3章では提案システムの前提条件、システムの概要、システム内での管理情報と各機能の処理フローについて、4章では試作したプロトタイプシステムの実装と動作検証による評価について、5章ではシステム設計に関する考察を述べる。最後に6章でまとめと今後の課題について述べる。

## 2. 視聴環境の高度化

すでに提案されたり、サービスとして実用化されたりしているパーソナライズ型の映像配信は、マルチソースの映像コンテンツを用意し、個人の嗜好や属性に基づいて、希望する映像コンテンツを提供する機能を有している。この型の配信では推奨するコンテンツを選択したり、複数の推奨コンテンツを動的に結合し、1本のコンテンツとして配信したりするなど、コンテンツの選択が配信側の機能にて提供されている。

これに対して、選択された映像コンテンツをどのように視聴するかを視聴者自身が視聴環境をカスタマイズすることにより、映像の受信側で視聴環境を高度化していくことができる。そのために、以下の要素を視聴環境に取り入れることとする。次章でこれらの要素の一部を実現するシステムの設計とプロトタイプシステムの実装について述べる。

### (1) 視聴画面のカスタマイズ

デジタル放送の開始とともにテレビ画面は細かく区分され、個々の画面に様々なコンテンツが配置されるようになった。また、パーソナライズ型の映像配信によって多数のコンテンツが選択できるようになったが、依然画面構成（視聴インタフェース）は定型的なものが多い。文献 3)、4) の提案システムでも映像配信元によってあらかじめ決められた数のソースを固定の画面レイアウトで視聴する。視聴環境のカスタマイズ要素として、画面レイアウトを各視聴者ごとに自由にカスタマイズできることをあげる。現在の画面では視聴映像の形は四角に限られているが、今後はその形状を丸や楕円にするような画面も考えられる。このように、視聴者の属性や嗜好によって任意の位置に任意の形で表現することで、視聴者の意志を自由に表現でき、臨場感や没入感の向上が期待できる。

また、自由に画面レイアウトをカスタマイズすることで、視聴映像の大きさや形により

再生する映像に求められる映像品質が変わってくる。また、ネットワークデバイスの多様化により、視聴者が再生に用いるディスプレイ環境も多種多様なものとなっている。そのため、視聴環境のカスタマイズ要素として、映像品質そのものについても視聴者が要求する品質の映像を提供することが求められる。

### (2) 複数視聴者の視聴空間の共有とインタラクティブコミュニケーション

放送網とIP通信網との違いの1つに視聴者自らが映像を送信することができる点がある。映像配信においてはIPTVも含め、視聴画面には配信映像コンテンツのみが表示されるが、IP通信においては視聴空間に視聴者から送信した映像を同時にレイアウトすることもできる。これにより、視聴者間で映像のやりとりが可能となり、さらにその視聴者間で視聴画面を同期させることにより、視聴体験を共有することができる。このような視聴空間の共有も視聴環境のカスタマイズ要素の1つとする。これは、“同じものを視聴する”というサイバー空間における一体感をより向上させるものであり、より円滑なインタラクティブコミュニケーションを促進することができる。

本研究で対象とするインタラクティブコミュニケーションとは異なるが、動画共有サイトの1つである「ニコニコ動画」<sup>5)</sup>では動画再生中にコメントを投稿すると、そのコメントは他の閲覧者のコメントとともにその動画の再生時に同じタイミングで表示される。これにより、擬似的に時間共有して動画の閲覧に関する感想を述べあい、閲覧者同士の一体感を獲得している。コメントだけでなく、動画や音声による視聴空間の共有という要素を視聴環境に入れることで、自分以外の視聴者の歓声や声援を視聴でき、臨場感の向上や視聴体験の共有が実現できる。まさにサイバー空間において、お茶の間で一緒に同じテレビ画面を見ながら、スポーツ観戦をしているかのようなコミュニケーションが可能となる。

## 3. 提案システム

本章では、初めに提案システムの設計方針を述べ、システム概要と機能について詳細を述べる。

### 3.1 設計方針

#### 3.1.1 前提条件

##### (1) 利用規模

本研究では、数十人規模の視聴グループが多数存在し、各グループ内で視聴画面を共有しながらインタラクティブコミュニケーションを行うモデルを想定する。この場合、グループ内の受信ホストの視聴制御（シグナリング）を行う必要があるため、これを管理す

るサーバを用意することとする。なお、5.2 節における評価の指標として、本研究では、20 人規模の視聴グループが 50 グループ程度存在すると想定して、シグナリングのための管理サーバ 1 台が処理する規模は最大視聴者数の 1,000 人程度で考える。

### (2) 対象ストリーム形式

提案システムは、数百 kbps ~ 数 Mbps 程度の複数ビットレートのストリームデータを扱うこととする。これは、モバイル端末で用いられる QVGA 品質映像 (320 画素 × 240 ライン) から HD 品質映像 (1,440 画素 × 1,080 ライン) を、MPEG2-TS (Transport Stream) や H.264/AVC にエンコードしたビットレートを想定したものであり、近年の IPTV で一般的に利用されているビットレートである。

### (3) ストリーム配送方式

提案システムにて用いるストリーム配送方式について検討する。一般的なストリームの配送方式として、サーバ・クライアント方式、IP マルチキャスト方式、ALM (Application Layer Multicast) 方式<sup>(6),7)</sup>がある。これらの方式は、視聴者数、ストリームの安定的な配送の要求度など使用場面によって有効性は変わる。提案システムでは、複数映像を受信ホストで受信しながら視聴環境を柔軟にカスタマイズすることを目的としている。ゆえに、1,000 人規模の視聴者という大規模な配信に対しても、複数の映像と音声安定した品質で提供することができる IP マルチキャストを用いる。

IP マルチキャストは、文献 8) で示されているように、広域環境での利用も進められており実現の可能性は徐々に高まりつつあるため、本研究では、このような IP マルチキャスト網を活用するアプリケーションの 1 つとして位置づけている。また、IP マルチキャストでは、ストリーム切替え時におけるスムーズで高速な画面表示が要求されるが、本研究で示す受信システムの設計は、映像・音声ストリームのマルチキャスト配信フレームワークとは独立であるため、高速切替え手法については既存手法<sup>9)</sup>などが適用可能である。

ネットワーク環境としては、前述したビットレートのストリームを多数送受信できるだけの十分な帯域が確保されており、かつ IP マルチキャストの到達性が確保されていることを前提とする。

また、複数の映像・音声ストリームを受信ホストにて再生する際に、遅延やジッタに対する対策や受信者間同期などの要求も生じる可能性がある。本論文では、レイアウト共有を目的とした映像受信システムに焦点をあてているため、ネットワークの外的要因に対する対応策についての議論は行わないが、ストリーム伝送と本研究で考える受信システムは独立して設計できるため、グループ内におけるホスト間の伝送遅延を考慮した同期手

法<sup>10)</sup>などの既存提案手法が適用できると考えられる。

### 3.1.2 実現する機能

#### (1) 画面レイアウトのカスタマイズ

画面レイアウトのカスタマイズを検討するうえでの前提条件として、視聴インタフェースは PC を用いるとする。

提案システムでは、視聴者の画面のレイアウトのカスタマイズ方法として、マウスによる直感的な操作によるインタフェースを採用する。複数の画面の合成は、あたかも OS のデスクトップ画面に複数のウィンドウで複数プログラムを実行するかのようになり、クライアントノードをマルチウィンドウのアプリケーションとして設計する。個々の受信した映像ごとにウィンドウを割り当て、それぞれのウィンドウの位置、サイズを変更することで、画面レイアウトが簡単にカスタマイズ可能となる。

複数映像の合成方法には、提案システムのようにエンドノードであるクライアントで画面合成を行う方法と、MCU (Multipoint Control Unit) などのネットワーク上の中間ノードでの映像合成処理による合成方法がある。MCU の場合、受信ホストでは、画面合成されたストリームを表示するのみであるため、選択的なレイアウト構造や MCU における単一障害の問題を含んでいる一方で、合成負荷は考慮する必要がない。しかしながら、提案システムでは、受信ホストで複数のストリームを受信し、マウスを利用して自由度の高い画面合成処理を行うため、その処理負荷が増大することが可能性として考えられる。これまでこのような受信ホストでの画面合成手法における負荷の定量的な評価は行われていないため、これに関する検証については 4.2.2 項で述べる。

#### (2) 映像品質のカスタマイズ

提案システムでは、映像のビデオコーデックやオーディオコーデック、および、そのビットレートの組合せを映像品質として取り扱う。MCU などのネットワーク上の中継装置で映像合成をすることで、映像品質のカスタマイズを行うことができるが、MCU では複数映像を中間ノードで合成した後にトランスコードなどを行うため、合成前の複数の送信元映像に対しての品質調整はできない。そこで、複数の品質の映像を個々に提供する機構が必要となる。複数の品質の映像を提供する方式として、映像の配信元において複数の映像品質へ変換したストリームをあらかじめ用意しておく方法<sup>11)</sup>と、映像品質の変換を要求されたときに、映像配信システム内のノードにおいて任意の映像品質のストリームを実時間での変換によって生成する方法<sup>12)-14)</sup>がある。前者は、想定する映像品質のバリエーションが少ない場合は、有効な方式であるが、今回は不特定多数のマルチソースを用いた

映像配信システムを想定しているため、クライアントから要求があった映像品質を、その時点で生成する後者の方式を採用する。

なお、そのほかに複数の品質の映像を個々に提供する方法として、P2P 技術で一般的に利用される、エンドノード自体が映像品質を変換する方法<sup>15)</sup>がある。しかしながら、本システムでは、受信ホストは画面構成に必要な IP マルチキャストストリームを受信し、自由度の高いレイアウト処理および再生処理を主とする設計としている。そのため、トランスコード機能は IP マルチキャスト網内に置き、エンドノードでの変換は行わない。

(3) 視聴者間の視聴空間の共有とインタラクティブコミュニケーション

提案システムでは、視聴者間のインタラクティブコミュニケーション用に各視聴者が映像を 1 つのストリームとして送信するパブリックビューイング機能を盛り込む。これは、クライアントから視聴者自身の音声や映像を送信することで視聴体験をお互いに共有する機能である。クライアントから送信される映像も、1 つの映像ソースとして扱われ視聴できるようにする。

これに加えて、視聴者間での視聴空間の共有を実現する機能として、スクリーンシェア機能を設計する。この機能は、視聴者間の視聴画面を同期することで、他の視聴者と視聴空間を共有することを可能にする。各視聴者の視聴画面の画面レイアウト情報として、各ウィンドウの位置や大きさを管理サーバで管理し、同期したいクライアント情報を管理サーバから受信することで、視聴者間の視聴映像の同期を実現する。

3.2 提案システムの概要

提案システムは、ビデオ配信サーバ、管理サーバ、トランスコーダ、クライアントにより構成される。システム構成を図 1 に示す。

まず、基本的な映像配信の流れを説明する。コンテンツとなる映像と音声は、カメラとマイクから取り込まれ、それぞれ MPEG-2 (4,096 kbps) と MPEG-1 Audio Layer2 (384 kbps) に符号化された後、コンテナフォーマットを MPEG-2 TS としてビデオ配信サーバから IP マルチキャストでストリーミング配信される。ストリーミングプロトコルには、RTP (Real-time Transport Protocol)<sup>16)</sup> と UDP (User Datagram Protocol) を用いる。ストリーミングされている各コンテンツの情報は、管理サーバで管理されている。クライアントは、管理サーバから配信されているコンテンツの最新情報を取得し、番組表を視聴者に提供する。番組表には、図 2 のようにコンテンツの名前、コンテンツの説明文が表示されており、視聴者はこの情報をもとに任意のコンテンツを選択し、再生を行う。個々の映像コンテンツの表示再生を行うウィンドウにはウィンドウ番号が割り振られており、再生時にウィンドウ番

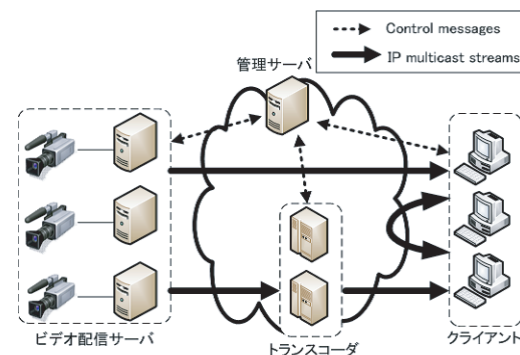


図 1 システム構成図  
Fig. 1 Structure of the system.

contents	comment	transcode	video codec	video rate	audio codec	audio rate	window number	play button
camara1	ピッチャーを追ったカメラ映像です。	<input type="checkbox"/> 変換	h264	4096	mp3	256	1	再生
camara2	バッターを追ったカメラ映像です。	<input type="checkbox"/> 変換	h264	4096	mp3	256	1	再生
camara3	球場全体を撮影しているカメラ映像です。	<input type="checkbox"/> 変換	h264	4096	mp3	256	1	再生

図 2 番組表のイメージ図  
Fig. 2 A sample image of channel list.

号を指定することで、任意のウィンドウにて映像コンテンツの再生を行う。映像品質のカスタマイズを行わない場合は、視聴者が選択したコンテンツに対応するマルチキャストグループにクライアントがジョインすることで、ストリームを受信し、再生を開始する。映像品質のカスタマイズを行う場合の処理は、3.3.3 項で詳細の説明を行う。また、番組表と同様に、クライアントは管理サーバからクライアント一覧を取得し、図 3 に示すスクリーンシェア設定ウィンドウを視聴者に提供する。これは、スクリーンシェア機能の利用時に使用される。

なお、本システムではセッション制御が必要なシグナリングメッセージは TCP を用いるとする。

3.3 管理情報と処理フロー

管理サーバは、システム内のアカウント、コンテンツ、クライアント、トランスコーダ、マルチキャストアドレスの情報をそれぞれデータベース内の、利用者情報テーブル、コンテ

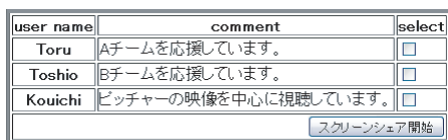


図 3 スクリーンシェア設定ウィンドウのイメージ図

Fig. 3 A sample image of a configuration window of screen share.

表 1 利用者情報テーブル

Table 1 Account information table.

user ID	user name	password	comment
0001	Toru	****	A チームを応援しています .
0002	Toshio	****	B チームを応援しています .
0003	Kouichi	****	ピッチャーの映像を中心に視聴しています .

表 2 コンテンツ情報テーブル

Table 2 Contents information table.

content name	multicast address	comments	video codec	video rate [kbps]	audio codec	audio rate [kbps]
camera1	239.0.0.1	ピッチャーを追った映像	mpeg2v	4,096	mp2	384
camera1-2	239.0.0.2	ピッチャーを追った映像	mpeg4	2,048	mp3	128
camera2	239.0.0.3	バッターを追った映像	mpeg2v	4,096	mp2	384

ンツ情報テーブル, クライアント情報テーブル, トランスコーダ情報テーブル, アドレス情報テーブルにて管理する .

### 3.3.1 管理情報

利用者情報テーブルでは, 表 1 のようにサービス利用開始時の認証用のパスワード, システム内で視聴者を識別するユーザ ID, インタラクティブコミュニケーションで用いるユーザの名称とコメント文を管理する .

コンテンツ情報テーブルでは, 表 2 のようにコンテンツの名称, 配信されているマルチキャストアドレス, コンテンツの説明文, メディアフォーマット情報といった各コンテンツの詳細情報が登録されている . 映像品質のカスタマイズによって生成されたコンテンツについても個別のレコードとして登録される .

クライアント情報テーブルでは, 表 3 のようにクライアントが各ウィンドウにて再生し

表 3 クライアント情報テーブル

Table 3 Table of client information.

user ID	window ID	content name	width	height	x-coordinate	y-coordinate
0001	1	camera1	640	480	5	5
0001	2	camera2	320	240	330	10
0002	1	camera3	640	480	30	30

表 4 トランスコーダ情報テーブル

Table 4 Transcoder information table.

IP address	CPU [%]	memory [kbyte]
192.168.1.1	33	1,500,000
192.168.1.2	60	1,000,000

ているコンテンツと, そのコンテンツを再生しているウィンドウの位置や大きさの情報が登録される . レコードの登録は, ウィンドウ単位で行われるため, クライアントごとに視聴画面に表示しているウィンドウの数だけ登録されることになる . クライアント情報は, スクリーンシェア機能で他の視聴者の視聴画面と自身の視聴画面を同期する際に使用される .

トランスコーダ情報テーブルでは, 表 4 のようなトランスコーダの負荷状況が登録される . トランスコーダは, 映像品質のカスタマイズを提供するために, 映像と音声のストリームを実時間でトランスコードを行う . このトランスコード情報により, 複数のトランスコーダを用いた映像品質のカスタマイズ処理の負荷分散を可能にしている . 処理負荷の尺度として, CPU 使用率, メモリの空き容量を用いており, 新たにトランスコードを行う必要がある際に, どのトランスコーダを使用するかを選択に利用される . トランスコーダが, トランスコーダ情報更新メッセージを管理サーバに定期的に送信することで, トランスコーダ情報テーブルの更新を行う .

クライアントやトランスコーダから動的に送信されるストリームは, マルチキャストアドレスを使用する . そこで, マルチキャストアドレスの管理と割当てを管理サーバが行う . アドレス情報テーブルには, マルチキャストアドレスごとにテーブルが登録されており, マルチキャストアドレスの割当て状態と利用可能期限の情報が登録されている .

### 3.3.2 スクリーンシェア機能

スクリーンシェア機能での処理の流れを図 4 に示す . クライアントは, 定期的に自身の画面レイアウトに変化があるか確認し, 変化があれば, クライアント情報更新メッセージを管

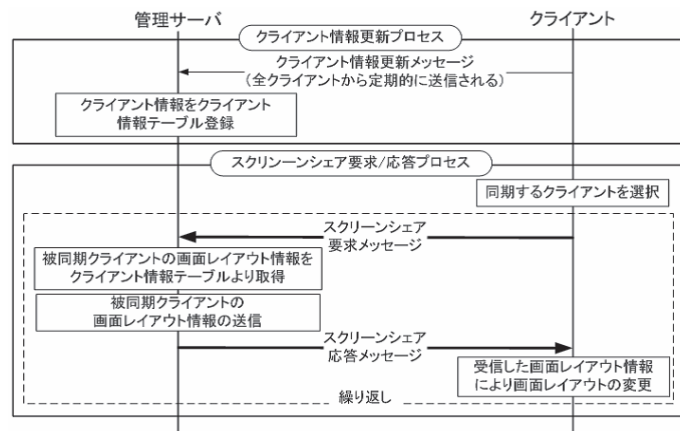


図 4 スクリーンシェア機能の処理フロー  
Fig. 4 Process flow of a screen share function.

理サーバへ送信することでクライアント情報を更新する。スクリーンシェア機能を使用するクライアントは、視聴者が図 3 に示すスクリーンシェア設定ウィンドウで、一覧として提供される視聴者の名前とコメント文を参照し、同期させるクライアントを選択することで、スクリーンシェア要求メッセージを定期的に管理サーバへ送信し、同期情報の要求を行う。管理サーバは、クライアント情報テーブルを参照することで、各クライアントがどのような画面レイアウトでどのコンテンツを再生しているかという同期情報を把握しているので、スクリーンシェア応答メッセージで同期情報を送信する。クライアントは、その同期情報に基づき画面レイアウトを更新する。

スクリーンシェア機能による視聴者間の画面レイアウトの同期間隔は、映像や音声を視聴者間で同期させるようなものとは異なり、厳密な実時間性が求められる場面はほとんどない。そこで、トラフィック量の増加を考慮して、クライアント情報更新メッセージとスクリーンシェア応答メッセージの送信間隔をいずれも 1 秒と設定した。ただし、クライアント情報更新メッセージは画面レイアウトに変化がなければ送信は行わない。また、メッセージは個々のコンテンツを再生しているウィンドウ単位で送信されるため、1 秒間に複数のウィンドウのレイアウトに変化があった場合は、そのウィンドウの数だけ送信される。

スクリーンシェア機能はある同期元ユーザの視聴画面を複数人で共有することを目的としているため、本システムでは双方向での同期は考慮しない。そのため、管理サーバでは、ス

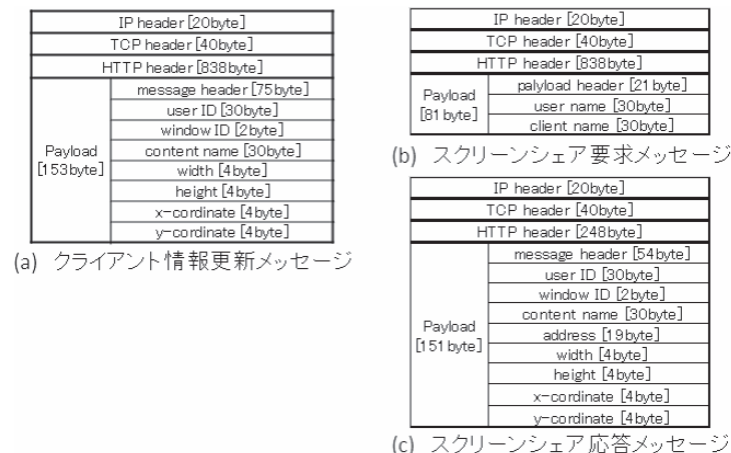


図 5 メッセージフォーマット  
Fig. 5 Message format.

クリンシェア要求を受けた際に、双方向での参照とならないかの確認を行う。

クライアント情報更新メッセージは、HTTP の POST メソッドで画面レイアウト情報を送信するためのメッセージである。図 5 (a) にクライアント情報更新メッセージと各要素のサイズを示す。ただし、画面レイアウト情報は、テキストベースのデータであり、メッセージサイズは可変長となるため、各要素のサイズは最大値をとる。

スクリーンシェア要求メッセージは、HTTP の POST メソッドでクライアントから管理サーバへ同期情報を要求するためのメッセージである。データ部のメッセージ長は、クライアント情報更新メッセージと同様に可変長であるため、ここでは最大メッセージサイズを用いた。図 5 (b) にメッセージフォーマットと各要素のサイズを示す。

スクリーンシェア応答メッセージは、各クライアントからのスクリーンシェア要求メッセージに対する HTTP のレスポンスとして同期情報を送信するためのメッセージである。同期情報のデータ部についても、可変長であるため、ここでは最大メッセージサイズを用いる。図 5 (c) にメッセージフォーマットと各要素のサイズを示す。

### 3.3.3 映像品質のカスタマイズ処理

映像品質のカスタマイズを行う場合には、視聴者自身が嗜好や視聴環境にあわせて自由にビデオ・音声コーデックとそのビットレートの指定を行う。



映像品質のカスタマイズ処理の流れを図 6 に示す。再生したいコンテンツとその映像品質の情報は、クライアントから管理サーバへ再生要求メッセージにより送信される。管理サーバは要求された映像品質のストリームがすでに生成されていないか、コンテンツ情報データベース内を検索する。すでに生成されていれば、そのストリームのマルチキャストアドレスをクライアントへ通知する。該当するストリームが存在しない場合は、トランスコードヘトランスコード要求メッセージを送信し、映像フォーマットの変換要求をする。要求するトランスコードの選定は、CPU 使用率およびメモリ使用率の小さいものから優先的に利用する負荷分散方式を採用している。

トランスコードを、複数配置することでシステム全体の映像品質のカスタマイズの要求に対応する。そこで、4.2.3 項において 1 台のトランスコードの処理性能を定量的に求めることで、前提条件で示した環境におけるトランスコードの性能や設置数の指標について述べる。

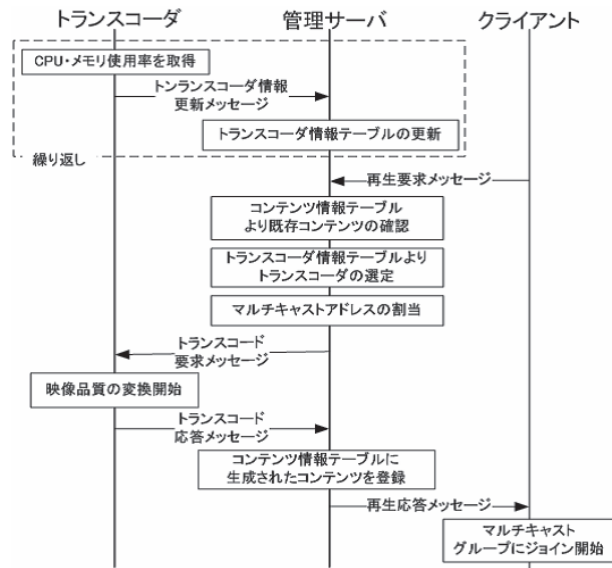


図 6 トランスコードの処理フロー

Fig. 6 Process flow of a transcoding function.

#### 4. プロトタイプシステムの実装と動作検証

##### 4.1 プロトタイプシステムの概要

3 章で提案した機能の一部をプロトタイプシステムとして実装を行った。プロトタイプシステムは、受信ノードで複数ストリームを受信し、合成処理を行うことによる処理負荷を定量的に計測する、提案するシグナリングプロトコルでスクリーンシェア機能が動作するかを実証する、映像品質のカスタマイズ機能がトランスコードにより実現できることを定量的に検証することを目的に実装した。その実装内容と実装環境について述べる。

##### 4.1.1 実装内容

プロトタイプシステムにおける実装状況を表 5 に示す。本論文では、画面レイアウト共有と映像品質のカスタマイズに焦点をあてて評価を行うため、パブリックビューイング機能については未実装としている。

また、プロトタイプシステムの機能モジュール構成を図 7 に示す。クライアントは管理サーバとで、コンテンツ情報の取得やスクリーンシェア機能で用いる制御情報のやりとりを行い、視聴インタフェースの制御を行うクライアント制御メッセージ処理部と、マルチキャストストリームの受信再生を行う映像再生処理部から構成される。管理サーバは、クライアント情報やコンテンツ情報をデータベース (DB) 内で管理し、クライアントとの制御情報のやりとりを行うクライアント管理処理部とトランスコード情報を管理し、トランスコードとの制御情報のやりとりを行うトランスコード管理処理部から構成される。トランスコードは管理サーバとトランスコード情報のやりとりを行うトランスコード制御メッセージ処理部とマルチキャストストリームを受信するストリーム受信部、映像品質の変換を行う映像フォーマット変換部、変換後のストリームをマルチキャストで配信するストリーム送信処理部から構成される。

ビデオ配信サーバからトランスコードやクライアントに配信されるストリームは UDP で

表 5 プロトタイプシステムの実装機能の一覧  
Table 5 Available functions in the prototype system.

システムの構成要素	実装状況
画面レイアウトのカスタマイズ可能な視聴インタフェース	実装済み
パブリックビューイング機能	未実装
スクリーンシェア機能	実装済み
映像品質のカスタマイズ機能	実装済み

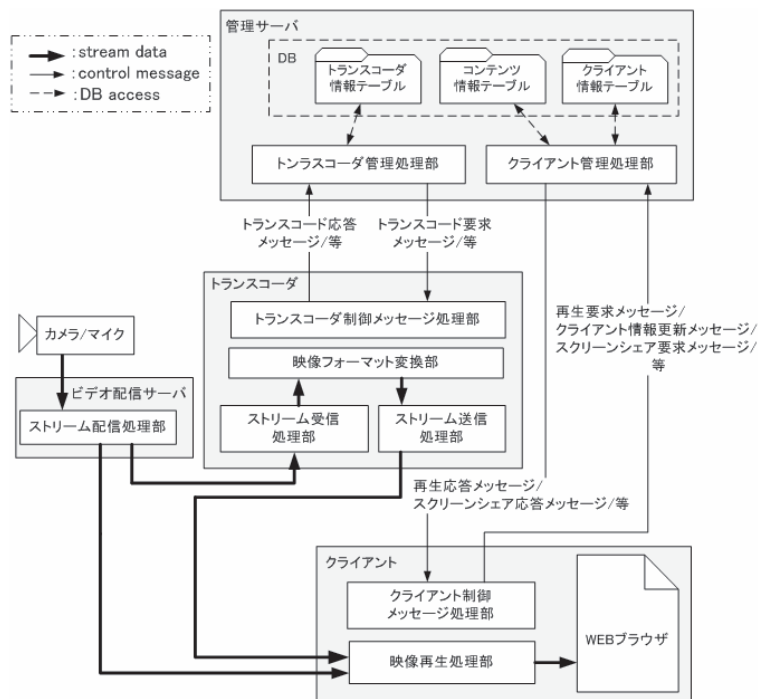


図7 機能モジュール構成図  
Fig.7 Function module diagram.

送信され、管理サーバ、トランスコーダ、クライアント間でやりとりされる制御メッセージは TCP で送受信される。

#### 4.1.2 開発環境

クライアントは、JavaScript により Web アプリケーションとして実装を行い、実行には Windows XP と Firefox3.0.9 を用いた。マルチウィンドウ型の GUI 作成には、JavaScript のライブラリである ext js.2.2 を用いた。また、映像再生処理部には、VLC media player<sup>17)</sup> の Firefox プラグインとその API を用いた。

管理サーバは、Linux 上で PHP4 を用いて実装を行った。クライアントとの HTTP (Hypertext Transfer Protocol) 通信に Apache2.2.3、データベース管理システムに PostgreSQL7.4 を用いた。トランスコーダは、Linux 上で C 言語 (gcc 4.3) により実装を行った。映像フォー

表6 スクリーンシェア機能の動作確認における使用機器

Table 6 PC Specification in experiment for screen share function.

	クライアント	管理サーバ	トランスコーダ	ビデオ配信サーバ
OS	Windows XP SP3	Debian Linux 4.1.1-21	Debian Linux 4.1.1-21	Windows XP SP3
CPU	Core 2 Quad Q8200 2.3 GHz	Pentium Dual E2160 1.80 GHz	Core 2 Duo T7100 1.8 GHz	Core 2 Duo T7100 1.8 GHz
メモリ	4 GB	1 GB	2 GB	2 GB

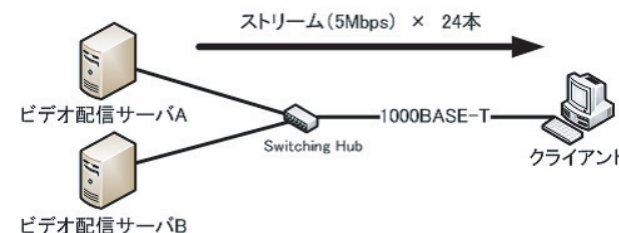


図8 画面合成処理の動作検証における実験構成  
Fig.8 Function verification environment in window layout function.

マット変換部には、FFmpeg<sup>18)</sup> を用いている。映像配信サーバには、既存ストリーミング配信アプリケーションである VLC media player 0.8.6h を用いた。

#### 4.2 プロトタイプシステムの評価

プロトタイプシステムの実装目的としてあげた検証項目について、以下で評価を行った。

##### 4.2.1 スクリーンシェア機能の動作確認

スクリーンシェア機能についてはまず、正常に動作するか確認するため、以下のような動作確認を行った。今回は、あらかじめ管理サーバのクライアント情報テーブルに登録しておいたクライアントとの画面レイアウトの同期を行った。実際に、管理サーバに登録されてる非同期クライアントの画面レイアウト情報と、スクリーンシェア機能を使用したクライアントとの視聴画面を比較し、正しく同期できることを確認した。表6に使用機器を示す。

##### 4.2.2 画面合成処理の動作検証

3.1.2 項の (1) で述べたように、提案システムでは、クライアントにおいて複数映像の画面合成を行う。そこで、クライアントにおいて画面合成処理を行いながら、複数映像の同時再生が可能か検証を行った。実験環境を図8、使用機器を表7に示す。



表 7 画面合成処理の動作検証における使用機器

Table 7 PC Specification in experiment for window layout function.

	ビデオ配信サーバ A	ビデオ配信サーバ B	クライアント
OS	Windows XP SP3	Windows XP SP3	Windows XP SP3
CPU	Xeon 3.4 GHz	Core 2 Duo T7100 1.8 GHz	Core 2 Quad Q8000 2.3 GHz
メモリ	3 GB	2 GB	4 GB
グラフィックカード	—	—	GeForce 9500GT (VRAM 512 MB)

動作確認には、24 本のストリームの画面合成を行った。この 24 という数は、QVGA (320 × 240) サイズの映像をフル HD 解像度 (1,920 × 1,080) のディスプレイで表示できる最大値である。QVGA はワンセグ放送で採用されている解像度であり、そのサイズの映像を一般の映像用ディスプレイとして最高クラスであるフル HD で表示できるだけ表示して、複数映像を視聴したことを想定している。配信には、映像コーデックに MPEG-2、音声コーデックに MP2、コンテナフォーマットに MPEG-2TS を用いた約 5 Mbps のテストデータを使用した。

検証手順は、ビデオ配信サーバからストリームを配信し、クライアントにおいて 24 本のストリームを図 9 に示す A~C の 3 つの画面レイアウトで同時再生できることを確認するものである。また、ストリーム再生中に CPU の使用率を Windows XP のパフォーマンスモニタが可能な Typeperf コマンドを利用して測定した。ビデオ配信サーバは、既存のストリーミング配信アプリケーションである VLC media player 0.8.6h を用い、ビデオ配信サーバ 1 台につき 12 本のストリームを送信した。

同時再生した結果、1 観測者で 30 秒間、目視と耳聴により、どの画面レイアウトにおいても同時再生しているすべての映像と音声にノイズや途切れがないことを確認した。また、各画面レイアウトで 30 秒間測定した結果、平均の CPU 使用率はレイアウト A で 64.0%、B は 69.7%、C は 65.1%であった。

#### 4.2.3 トランスコーダの性能評価

3.3.3 項で述べたように、1 台のトランスコーダでリアルタイムの映像品質の変換を行うときの処理性能を定量的に測定することで、映像品質のカスタマイズ機能の実現可能性を検証するとともにトランスコーダの数についての指標を示す。

図 10 のようにビデオ配信サーバから、マルチキャストで 1~5 本のストリームを配信し、トランスコーダでリアルタイムの映像品質の変換を行ったときの CPU 使用率を vmstat



図 9 画面レイアウト

Fig. 9 Display layouts.

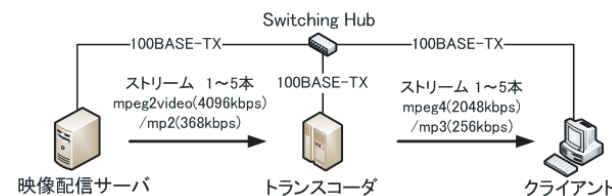


図 10 トランスコーダの性能評価における実験環境

Fig. 10 Experiment environment in performance evaluation of transcoding.

コマンドを用いて 1 秒間隔で 2 分間測定した。映像は MPEG-2 (4,096 kbps) から MP4 (2,048 kbps) に、音声は MP2 (384 kbps) から MP3 (256 kbps) に変換した。コンテナフォーマットには MPEG2-TS を用いた。ストリームの配信と受信再生には VLC media player 0.8.6h を用いた。使用機器を表 8、測定結果から求めた 2 分間の平均 CPU 使用率を表 9 に示す。また、クライアントでは変換後の各ストリームを受信、再生し、1 観測者で 2 分間、音声と映像の途切れやノイズがないことを目視と耳聴にて確認した。

499 画面レイアウト共有可能な複数映像受信システムの設計

表 8 トランスコーダの性能評価における使用機器  
Table 8 PC Specification in experiment for transcoding.

	トランスコーダ	ビデオ配信サーバ	クライアント
OS	Debian Linux 4.1.1-21	Windows XP SP3	Windows XP SP3
CPU	Core 2 Duo Q8200 2.3 GHz	Core Duo 1.06 GHz	Xeon 3.4 GHz
メモリ	4 GB	1 GB	3 GB

表 9 CPU 負荷測定結果  
Table 9 Measurement results of CPU load.

ストリームの本数	1	2	3	4	5
平均 CPU 使用率 (%)	11.16	21.12	31.68	42.33	52.97

4.3 動作検証に関する考察

4.2.2 と 4.2.3 項の動作検証に対する考察を以下で述べる。

4.3.1 画面合成処理性能

3.1.2 項の (1) であげていた、受信ホストにおける複数映像の同時再生と画面レイアウト変更に対する処理負荷については、映像・音声のデコード負荷と比較するとレイアウト変更による負荷増大はほとんど見られず、本環境であれば 20 本程度のストリームの同時再生とレイアウト処理を行えることが確認できた。

4.3.2 トランスコーダの性能評価

表 9 の結果より、ストリームの本数が増えるにつれて CPU 使用率が 10%程度増加していることが分かる。よって、本検証で用いた PC の仕様とメディアフォーマットの変換であれば、1 台のトランスコーダで 8 から 9 本のストリームを同時に変換できると考えられる。仮に、4.2.2 項で用いた 24 本のストリームを映像コンテンツの数とし、各映像コンテンツを 3 種類の映像品質に変換すると考えると、72 本のストリームを生成しなければならない。これに対し、本環境で使用した PC と同程度の仕様であれば、9 台程度のトランスコーダを用意することで対応できると考えられる。

5. システム設計に関する考察

本章では、5.1 節において提案システムの視聴インタフェースの有効性について他のシステムと比較評価し、5.2 節では、管理サーバへのシグナリングの集中が予想されるスクリーンシェア機能において、想定する規模での制御トラフィックを算出し、シグナリング制御の

表 10 機能比較表

Table 10 Functional comparison table.

	RMX 2000	Codian MCU 4500	提案システム
画面レイアウト	固定 (提供パターン数は 24)	固定 (提供パターン数は 50)	マウス操作による 自由な配置が可能
映像の合成方法	MCU 上で 1 ストリームの映像に合成	MCU 上で 1 ストリームの映像に合成	クライアントで映像を画面合成
画面合成後の映像品質の選択	可	可	該当しない
ソースごとの映像品質の選択	不可	不可	可
最大画面合成数	36	40	PC スペックに依存 (24 まで確認)

スケーラビリティについて評価することで、提案システムの設計の有効性を示す。

5.1 機能比較

複数映像を提供する視聴インタフェースを持つシステムとして、既存の MCU と提案システムとの比較を行い、提案システムの有効性を示す。

ここでは、Polycom 社の RMX 2000<sup>\*1</sup>と Tandberg 社の Codian MCU 4500 シリーズ<sup>\*2</sup>と提案システムの機能比較を行った。2 つの MCU は各社の最上位クラスの製品である。機能比較表を表 10 に示す。

2 つの MCU の画面レイアウトのカスタマイズは、あらかじめ用意されている画面レイアウトパターンから選択するという方法であるのに対し、提案システムは個々の映像をマウス操作で配置するという設計であり、任意のレイアウトを作成することができる。これはユーザインタフェースの観点からみても自由度の高い設計となっており、その有効性は高いといえる。

2 つの MCU は合成後の映像品質の選択はできるが、合成前のソースのストリームの映像品質のカスタマイズをすることはできない。たとえば、送信映像 A は高精細な品質で、送信映像 B は多少粗い品質で受信してもよいというような場合にも MCU の場合はどちらも一律に同じ映像品質になってしまう。提案システムではソースストリームを個々にカスタマイズし、それぞれをマルチキャスト配信するという設計により、より柔軟な映像品質のカス

\*1 Polycom 社 RMX 2000 . <http://www.polycom.co.jp/>

\*2 Tandberg 社 Codian MCU 4500 シリーズ . <http://www.tandbergjapan.com/>

タマイズが可能となっている。

以上より、本システムの設計は 2 章や 3 章で述べた「視聴者自身が視聴環境を創る」という本研究の目標に向けて、既存技術の MCU に比べ、画面のレイアウトや映像品質のカスタマイズの柔軟性の点で有効なものとなっているといえる。

## 5.2 シグナリング制御のスケラビリティに関する評価

本評価では、3.1.1 項の前提条件で示した 1,000 人規模（数十人規模の視聴グループが複数存在する場合）の視聴者が存在する環境下で、シグナリング処理を 1 台のサーバで行うときの制御トラフィックを見積もることで、スクリーンシェア機能を 1 台の管理サーバで提供するときのスケラビリティを評価する。

スクリーンシェア機能によるトラフィックは、図 4 に示したように、クライアント情報更新メッセージ、スクリーンシェア要求メッセージ、スクリーンシェア応答メッセージ、の 3 つの制御メッセージから構成される。

まず、トラフィックの算出条件について述べる。視聴者数は 3.1.1 項で示した前提条件より、1,000 人とする。これは、20 人規模の視聴グループが 50 グループ存在すると仮定して、各視聴グループでは、全員がスクリーンシェア機能を利用し、グループ内のある 1 人の視聴者（マスタとよぶ）の画面レイアウトを同期させている場合を考える。グループ内でのマスタの数は少ないほど、シグナリングは多くなるのでここでは 1 人とした。よって全体で見ると、50 人のマスタと 950 人の同期するクライアントが存在することになる。

本評価では、多めにトラフィックを見積もるために、すべてのマスタが 1 秒間隔で 1 つのウィンドウのレイアウトを変更するという非常にレイアウト変更が頻繁に行われる場合を想定する。このとき、管理サーバでは、マスタから受け取るクライアント情報更新メッセージと、他の同期クライアントへのスクリーンシェア要求メッセージ、その応答として管理サーバから送信されるスクリーンシェア応答メッセージ、同期クライアントがレイアウト変更することによるクライアント情報更新メッセージ、の 4 メッセージが 1 秒間に流れることになる。つまり、すべてのクライアント（1,000 人）からの情報更新メッセージを受信し、同期クライアント（950 人）からのスクリーンシェア要求メッセージを受信し、その応答としてスクリーンシェア応答メッセージが送信される。なお、各クライアントが受信している映像コンテンツの数はスクリーンシェアの制御メッセージのトラフィック量とは関係性がないため、本評価では 1 つ以上の映像コンテンツを全クライアントが受信しているとする。また、各クライアントのネットワークは、3.1.1 項の前提条件でも示したように、複数のマルチキャストストリームを受信できる環境を前提とする。

このような前提条件の下で、各メッセージのトラフィック量の算出を行う。まず、各々のメッセージサイズは以下のとおりとなる。

### (1) クライアント情報更新メッセージ ( $msg_{clinfo}$ )

画面レイアウト情報は、テキストベースのデータのためメッセージサイズは可変長となるが、ここでは最大メッセージ長のデータサイズで計算する。図 5 の (a) に示すクライアント情報更新メッセージと各要素のサイズから、メッセージサイズは、ペイロード部 (153 byte) と IP ヘッダ以降のヘッダ部 (898 byte) を合計した 1,051 byte となる。前提条件より各クライアントからは、1 秒間に 1 メッセージ送信される。つまり、1 クライアントから 1 秒間に送信されるトラフィック量の合計は 1,051 byte であり、ビットレートで換算すると 8.408 kbps となる。

このトラフィックの合計の中には、HTTP ヘッダの HOST フィールドなどの稼働環境で可変するデータが 1,051 byte 中に 357 byte 含まれている。そのため、実際には数百 byte 程度の増減が起こりうる。

### (2) スクリーンシェア要求メッセージ ( $msg_{ssreq}$ )

データ部のメッセージ長は、クライアント情報更新メッセージと同様に、可変長であるため、ここでは最大メッセージサイズを用いる。図 5 の (b) に示したメッセージフォーマットと各要素のサイズのサイズから、メッセージサイズは、データ部 (81 byte) と IP ヘッダ以降のヘッダ部 (898 byte) を合計した 979 byte となる。1 秒間隔で送信されるため、ビットレートに換算すると 7.832 Kbps となる。また、クライアント情報更新メッセージと同様に、HTTP ヘッダに稼働環境による可変領域があるため、実際には数百 byte のトラフィックの増減が起こりうる。

### (3) スクリーンシェア応答メッセージ ( $msg_{ssres}$ )

スクリーンシェア応答メッセージは、図 4 に示す処理の流れで、HTTP のレスポンスによって、同期情報を送信する。同期情報のデータ部についても、可変長であるため、ここでは最大メッセージサイズを用いる。図 5 の (c) に示したメッセージフォーマットと各要素のサイズから、メッセージ部は、データ部 (151 byte) と IP ヘッダ以降のヘッダ部 (308 byte) を合計した 459 byte となる。1 秒間隔に送信されるため、ビットレートに換算すると 3.672 kbps となる。

前提条件において、クライアント情報更新メッセージ、スクリーンシェア要求メッセージ、スクリーンシェア応答メッセージによる管理サーバでのトラフィックを算出すると、

## 上りトラフィック

$$\begin{aligned}
 &= (msg_{clinfo} + msg_{ssreq}) \times \text{クライアント数} \\
 &= (8.408 \text{ (kbps)} + 7.832 \text{ (kbps)}) \times 1,000 \\
 &= 16.24 \text{ (Mbps)}
 \end{aligned}$$

## 下りトラフィック

$$\begin{aligned}
 &= msg_{ssres} \times \text{クライアント数} \\
 &= 3.672 \text{ (kbps)} \times 950 \\
 &= 3.488 \text{ (Mbps)}
 \end{aligned}$$

となる。これは、前提条件で示した 1,000 人規模の視聴者で、厳しく想定した利用環境においても、管理サーバにおける必要帯域は、たかだか十数 Mbps 程度であることが分かる。本提案システムでは、数百 kbps から数 Mbps のストリームを伝送できる環境を前提と考えており、シグナリング制御のスケラビリティの観点から考えると、1,000 台程度のクライアント数であれば十分処理できる帯域であると考えられる。

## 6. おわりに

本論文では、IP 通信網を用いた映像配信において、柔軟な画面レイアウトのカスタマイズや視聴空間の共有、視聴者間のインタラクティブなコミュニケーションといった受信環境を高度化する映像受信システムを設計した。画面レイアウトのカスタマイズでは、1 ストリームに 1 ウィンドウを割り当てる視聴インタフェースを設計し、複数映像の個々の映像品質やサイズ、画面レイアウトの変更を可能にした。また、その視聴インタフェースを持つプロトタイプシステムを実装し、定量的な処理負荷を測定することで有効性を示した。画面レイアウトや映像品質の柔軟性の点では、既存の MCU との機能を比較をすることで、提案システムの有効性を示した。スクリーンシェア機能については、想定する 1,000 人程度の視聴者数において、管理サーバでの制御トラフィックを算出することで、シグナリング制御の観点での管理サーバのスケラビリティを示した。

今後の課題として、未実装部分であるパブリックビューイング機能の実装と管理サーバの分散機構を設計する。また、視聴環境のカスタマイズを行うことでの効果を客観的に評価する。

謝辞 本研究を行ううえで、貴重なご助言をいただきました、広島市立大学インターネット工学研究室、広島大学情報メディア教育研究センターの各位に感謝いたします。また、本システムの評価実験で協力していただいた広島市立大学の石國裕一氏、福田将大氏に感謝い

たします。本研究の一部は、総務省戦略的情報通信研究開発通信制度 (SCOPE-地域 ICT, 082308001) および、日本学術振興会科学研究費補助金 (20700066) の支援を受けて実施している。ここに記して謝意を表す。

## 参 考 文 献

- 1) 株式会社 NTT ぷらら：ひかり TV, 株式会社 NTT ぷらら (オンライン). 入手先 <http://www.hikaritv.net/> (参照 2009-05-10)
- 2) 川添雄彦：コンテンツ流通への NTT R&D の取り組み, NTT 技術ジャーナル 2009.4, pp.43-48 (2009).
- 3) オープンインターフェース株式会社：DOING TV, オープンインターフェース株式会社 (オンライン). 入手先 <http://www.doing.tv> (参照 2009-05-10)
- 4) 山田真理子, 鈴木敏明：パーソナライズ映像配信システムの試作, 情報処理学会研究報告, CIVIM, 2008(27), Vol.162, pp.531-536 (2008).
- 5) 株式会社ニワンゴ：ニコニコ動画, 株式会社ニワンゴ (オンライン). 入手先 <http://www.nicovideo.jp/> (参照 2009-05-10)
- 6) Pendarakis, D., Shi, S., Verma, D. and Waldvogel, M.: ALMI: An Application Level Multicast Infrastructure, *Proc. 3rd USENIX Symp. on Internet Technologies and systems*, pp.49-61 (2001).
- 7) 山口弘純, 中村嘉隆, 廣森聡仁, 安本慶一, 東野輝夫, 谷口健一：動画をを用いたコミュニケーションシステム向けのアプリケーション層マルチキャスト, コンピュータソフトウェア, Vol.21, No.2, pp.83-93 (2004).
- 8) 三宅 喬, 美甘幸路, 小林和真：Death Valley の克服に向けた相互接続性確立のための研究開発—IPv6 マルチキャスト技術の応用, 情報処理学会誌, Vol.49, No.10, pp.1159-1164 (2008).
- 9) 大内宏之, 佐藤裕昭, 高橋 賢, 永田広充：IP マルチキャストにおけるコンテンツ切替方式の提案と評価, 電子情報通信学会技術研究報告, IN2003-195, Vol.103, No.650, pp.43-48 (2004).
- 10) 岸田崇志, 河野英太郎, 前田香織, 角田良明：ネットワーク遅延を考慮したグループ内メディア同期機構の提案, 電子情報通信学会技術研究報告, IA2003-39, Vol.103, No.605, pp.19-24 (2004).
- 11) Conklin, G., Geenbaum, G., Lillevold, K. and Lippman, A.: Video Coding for Streaming Media Delivery on the Internet, *IEEE Trans. Circuits and Systems for Video Technology*, Vol.11, No.3, pp.269-281 (2001).
- 12) 山崎達也, 福永 茂, 佐藤範之：トランスコーディングを用いた複数端末への QoS 調整動画配信, 電子情報通信学会論文誌, Vol.85, No.1, pp.50-59 (2002).
- 13) 山崎修一, 孫 タオ, 玉井森彦, 安本慶一, 柴田直樹, 伊藤 実：多種多様な端末に対する効率のよいビデオ配信方式, 情報処理学会研究報告, 2005-DPS-122, Vol.33,

pp.315-320 (2005).

- 14) 阿久津隆志, 木全英明, 清水 淳, 八島由幸, 小林直樹: 低ビットレート様 MPEG-2 リアルタイムソフトウェアトランスコーダ「Trampeg」の開発, 電子情報通信学会論文誌, Vol.84, No.6, pp.1084-1093 (2001).
- 15) 田中貴章, 相田 仁: トランスコードを伴った P2P ビデオストリーミングの検討, 工学技報, Vol.108, No.458, IN2008-200, pp.405-410 (2009).
- 16) Schulzrinne, H., Casner, S., Frederick, R. and Jacobson, V.: RTP: A Transport Protocol for Real-Time Applications, RFC 3550 (2003).
- 17) The VideoLAN Project: VLC media player, The VideoLAN Project (オンライン). 入手先 <http://www.videolan.org> (参照 2009-05-10)
- 18) FFMPEG: FFmpeg, FFmpeg (オンライン). 入手先 <http://ffmpeg.org/> (参照 2009-08-13)

(平成 21 年 5 月 18 日受付)

(平成 21 年 10 月 2 日採録)



安部 光一

2008 年広島市立大学情報科学部・情報メディア工学科卒業。現在、同大学大学院情報科学研究科・情報工学専攻博士前期課程在学中。マルチメディア情報通信に関する研究に従事。



前田 香織 (正会員)

1982 年広島大学総合科学部卒業。広島大学工学部助手,(財)放射線影響研究所技術員, 広島市立大学情報処理センター助教授を経て, 現在, 広島市立大学大学院情報科学研究科教授。博士(情報工学)。コンピュータネットワーク, マルチメディア情報通信に関する研究に従事。電子情報通信学会, 教育システム情報学会各会員。



井上 博之 (正会員)

1987 年大阪大学工学部電子工学科卒業。1989 年同大学大学院工学研究科修士課程修了。同年住友電気工業株式会社入社。2000 年奈良先端科学技術大学院大学情報科学研究科博士後期課程修了。2000 年株式会社インターネット総合研究所入社。2004 年株式会社 IRI ユビテック(現, 株式会社ユビテック)転籍。2007 年広島市立大学院情報科学研究科講師, 現在に至る。博士(工学)。IPv6, 移動体通信, センサーネットワークに関する研究に従事。電子情報通信学会, IEEE 各会員。



近堂 徹 (正会員)

2001 年広島大学工学部第二类(電気系)卒業。2003 年同大学大学院工学研究科博士課程前期修了。2006 年同大学院工学研究科博士課程後期修了。現在, 広島大学情報メディア教育研究センター助教。博士(工学)。IP ネットワーク上でのリアルタイムマルチメディア通信, ネットワークコンピューティング技術等の研究に従事。電子情報通信学会会員。



鈴木 徹

2009 年広島市立大学情報科学部・情報メディア工学科卒業。現在, 同大学大学院情報科学研究科・情報工学専攻博士前期課程在学中。IPv6, マルチメディア情報通信に関する研究に従事。