

プッシュ型情報配信のためのプロトコルと そのコンテンツ配信システムへの適用

石川 憲 洋[†] 木下 真 吾^{††} 高橋 修[†]

インターネット上の新しいサービスとして、プッシュ型情報配信サービスが注目を集めている。しかしながら、これらのサービスの大部分はクライアントが定期的に情報配信サーバにアクセスする「疑似プッシュ型」であり、情報のリアルタイム配信、クライアントの増加に対するスケーラビリティの確保などの課題は解決されていない。一方、プッシュ型情報配信サービスを実現するための基盤技術として IP マルチキャストが注目を集めているが、現在実験段階にあり、実用化には至っていない。本論文では、IP ユニキャスト通信のみをサポートしている現在のインターネット上でプッシュ型情報配信サービスを提供するための新しいアーキテクチャとプロトコルを提案する。我々のアーキテクチャは、情報のリアルタイム配信、クライアントの増加に対するスケーラビリティの確保などのプッシュ型情報配信に対する要求条件を満足することができる。我々のアーキテクチャは、情報配信サーバ、中継サーバ、クライアントから構成される。クライアントの増加に対するスケーラビリティを確保するために、複数台の中継サーバが木構造で接続される構成とした。我々のアーキテクチャを実現するために配信チャンネル管理プロトコル (DCMP) と呼ぶ新しいプロトコルを開発した。本アーキテクチャに基づく実装と、そのコンテンツ配信システム (RealPush Network) への適用についても述べる。

A Protocol for Push Information Delivery and Its Application to Contents Delivery System

NORHIRO ISHIKAWA,[†] SHINGO KINOSHITA^{††} and OSAMU TAKAHASHI[†]

Push information delivery services have been emerging as a new service over the Internet. However, most of those services are pseudo-push information delivery services where clients periodically access information delivery servers. Those pseudo-push information delivery services do not resolve issues such as scalability on the increase of clients and real-time delivery of information. While it is expected that IP multicast is a core technology for push information delivery services over the Internet, IP multicast is now at the experimental stage, and has not been fully deployed over the Internet at this time. In this paper, we propose a new architecture and protocol for push information delivery services over the existing Internet that does not support IP multicast. Our architecture satisfies the requirements on push information delivery, such as scalability on the increase of clients and real-time delivery of information. Our architecture consists of information delivery servers, relay servers and clients. To retain the scalability of our architecture, multiple relay servers form a relationship that has a tree structure. To realize our architecture, we have defined a new protocol called Delivery Channel Management Protocol (DCMP). We have implemented a prototype system based on our architecture. In addition, we have developed a contents delivery system called RealPush network, as an application of our architecture.

1. はじめに

インターネット上の新しいサービスとして、Point-Cast Network に代表されるプッシュ型情報配信サー

ビスが注目を集めている。しかしながら、これらのサービスの大部分はクライアントが定期的に情報配信サーバにアクセスする「疑似プッシュ型 (スマートブル型)」であり、情報のリアルタイム配信、クライアントの数の増加に対するスケーラビリティの確保などの課題は解決されていない。

一方、プッシュ型情報配信サービスを実現するための基盤技術として IP マルチキャスト通信が注目を集めている。IP マルチキャスト通信を使用することに

[†] NTT 移動通信網株式会社マルチメディア研究所
NTT Mobile Communications Network Inc., Multimedia Laboratories

^{††} NTT 情報流通プラットフォーム研究所
NTT Information Sharing Platform Laboratories

より、情報配信サーバは、クライアントの数に関係なく1回の送信ですべてのクライアントに情報を配信することができる。加えて、情報配信サーバ主導の情報配信が可能となり、新しい情報が発生したときに、ただちに配信することができる。しかしながら、IP マルチキャスト通信は、インターネット上の仮想的な実験ネットワークである MBone¹⁾などを使用して実験を実施している段階にあり、商用インターネット上での実用化には至っていない。

本論文では、IP ユニキャスト通信のみをサポートしているネットワーク上で「疑似プッシュ型」ではない真のプッシュ型情報配信サービスを提供するためのアーキテクチャとプロトコルを提案する²⁾。本アーキテクチャにより「疑似プッシュ型」の情報配信サービスが抱える上記の課題を解決することができる。我々は、本アーキテクチャに基づくプロトタイプシステムを様々なプラットフォーム上で実装した。本アーキテクチャに基づく実装と、その RealPush Network と呼ぶコンテンツ配信システムへの適用についても述べる。

本論文の構成は、以下のとおりである。2章では、提案するプッシュ型情報配信のためのアーキテクチャについて述べる。3章では、提案するアーキテクチャを実現するためのプロトコルについて述べる。4章では、提案するプッシュ型情報配信方式の概要について述べる。5章では、本アーキテクチャとIP マルチキャスト通信ネットワークとの相互接続方式について述べる。6章では、本アーキテクチャに基づくプロトタイプシステムの実装について述べる。7章では、IP マルチキャスト通信アプリケーションのプロトタイプシステム上への移植実験について述べる。8章では、本アーキテクチャのコンテンツ配信システム (RealPush Network) への適用について述べる。9章では、プロトタイプシステムの実装実験の経験に基づき、その実用化に向けて解決すべき課題とその解決案について述べる。

2. アーキテクチャ

提案するプッシュ型情報配信のためのアーキテクチャを図1に示す。図1に示すように、提案するアーキテクチャは、情報配信サーバ、中継サーバおよびクライアントから構成される。クライアントの増加に対するスケーラビリティを確保するために、複数台の中継サーバが木構造で接続される構成とした。木構造を構成する中継サーバには上下関係がある。情報配信サーバは最上位の中継サーバと接続する。クライアントは最下位の中継サーバと接続する。

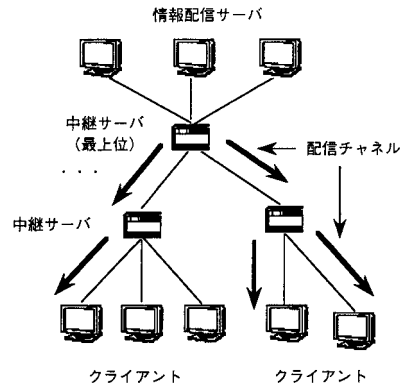


図1 プッシュ型情報配信のアーキテクチャ

Fig.1 Architecture for push information delivery.

情報配信サーバは、配信チャンネルを指定して、最上位の中継サーバに対して情報 (ニュース記事など) の配信を依頼する。中継サーバは、情報配信サーバから受信した情報を、設定した配信チャンネルを利用してクライアントまで配信する。配信チャンネルは、配信される情報の種別 (たとえば、株価情報、スポーツニュース、天気予報など) ごとに定義される。

配信チャンネルは最上位の中継サーバをルートとして、中継サーバ-中継サーバ間、中継サーバ-クライアント間に設定される。

配信チャンネルはクライアントからの要求に基づいて動的に設定される。情報配信の開始を要求するクライアントは、中継サーバに対して配信チャンネルへの加入を要求する。クライアントの加入要求は最上位の中継サーバまで伝播され、同時に配信チャンネルが設定される。情報配信の終了を要求するクライアントは、中継サーバに対して配信チャンネルからの離脱を要求する。クライアントの離脱要求は最上位の中継サーバまで伝播され、同時に設定された配信チャンネルは解放される。

クライアントは、いつでも、配信チャンネルに加入、または加入した配信チャンネルから離脱することができる。配信チャンネルは、配信チャンネル識別子により識別される。配信チャンネル識別子の長さは、IP マルチキャスト通信ネットワークとの相互接続を考慮して、48ビットとした。IP マルチキャスト通信ネットワークと相互接続する場合は、配信チャンネル識別子の上位32ビットにグループアドレス (クラスDのIPアドレス) を、下位16ビットに宛先ポート番号を設定する。IP マルチキャスト通信ネットワークと相互接続しない場合は、配信チャンネル識別子の値は任意である。

配信チャンネルとして、ニュース記事などの信頼性が要求される情報を配信するための高信頼チャンネルと、

ビデオ、オーディオなどのリアルタイム性が要求される情報を配信するためのリアルタイムチャンネルの2種類を定義した。

配信チャンネルごとに、情報配信のルートとなる最上位の中継サーバが定義される。クライアントの増加に対してスケーラブルな情報配信を実現するために、中継サーバは、木構造で多段接続することを可能とした。

本アーキテクチャのスケーラビリティを、同時に1000台以上のクライアントに対してニュースを配信する例で説明する。ルートとなる最上位の中継サーバに対して下位の中継サーバを接続し、下位の中継サーバに対してクライアントを接続する構成とする。この場合、中継サーバが同時に最大50台までのクライアントに対して情報を配信する能力を持つならば、結果として、最大2500台までのクライアントに対して情報を配信することが可能となる。このように、本アーキテクチャにより、クライアントの増加に対してスケーラブルなプッシュ型情報配信が実現可能となる。

3. プロトコル (DCMP)

2章で述べたアーキテクチャを実現するために、DCMP (Delivery Channel Management Protocol) と呼ぶプロトコルを設計した (図2)。

DCMP で定義しているメッセージと主なパラメータを表1に示す。

DCMP は、クライアントからの要求に基づいて配信チャンネルの設定、解放を行う機能、情報配信サーバからクライアントまで設定した配信チャンネルを用いて情報を配信する機能を提供する。

Join Channel は、クライアントが中継サーバに対して、配信チャンネルへの加入を要求するためのメッセージである。Leave Channel は、クライアントが中継サーバに対して、加入した配信チャンネルからの離脱を要求するためのメッセージである。Join Channel メッセージ、Leave Channel メッセージは最上位の中継サーバまで伝播される。配信チャンネルの設定、解放を行うために使用する上記のメッセージは、TCP/IP 上で転送される。

Send メッセージは、情報配信サーバが配信チャンネルを指定して、中継サーバに対して情報の配信を依頼するためのメッセージである。Delivery メッセージは、中継サーバが設定した配信チャンネルを使用して、情報を下位の中継サーバまたはクライアントに対して配信するためのメッセージである。配信チャンネルを用いて情報を配信するために使用する上記のメッセージについては、配信チャンネルの種類により、転送プロトコル

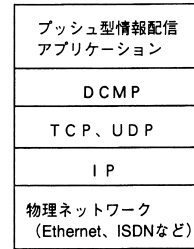


図2 DCMP のプロトコル構成

Fig. 2 Layer model of DCMP.

表1 DCMP のメッセージ

Table 1 DCMP messages.

| メッセージ | 主なパラメータ |
|---------------|----------------------------------|
| Join Channel | 配信チャンネル識別子、認証 |
| Join Ack | 配信チャンネル識別子 |
| Join Nak | 配信チャンネル識別子、理由 |
| Leave Channel | 配信チャンネル識別子 |
| Send | 配信チャンネル識別子、送信元IPアドレス、ペイロードタイプ、情報 |
| Delivery | 配信チャンネル識別子、送信元IPアドレス、ペイロードタイプ、情報 |

が異なる。高信頼チャンネルの場合は、TCP/IP 上で転送される。リアルタイムチャンネルの場合は、UDP/IP 上で転送される。

4. プッシュ型情報配信方式の概要

提案するプッシュ型情報配信方式の概要について述べる (図3)。

4.1 情報配信サーバからの情報配信

情報配信サーバは、新しい情報が発生した場合、クライアントに対して、ただちに配信を行うことができる。情報配信サーバは、クライアントの数に関係なく1回の送信ですべてのクライアントに情報を配信することができる。情報配信サーバは、配信チャンネルを指定した Send メッセージを送信することにより、配信チャンネルのルートとして定義された中継サーバに対して、情報配信を依頼する。

4.2 クライアントへの情報配信の開始

情報配信の開始を要求するクライアントは、加入する配信チャンネルを識別する配信チャンネル識別子を設定した Join Channel メッセージを、そのクライアントが接続する中継サーバに送信する。

Join Channel メッセージは、中間の中継サーバを経由して、配信チャンネルのルートとして定義された最上位の中継サーバまで、順次伝搬される。

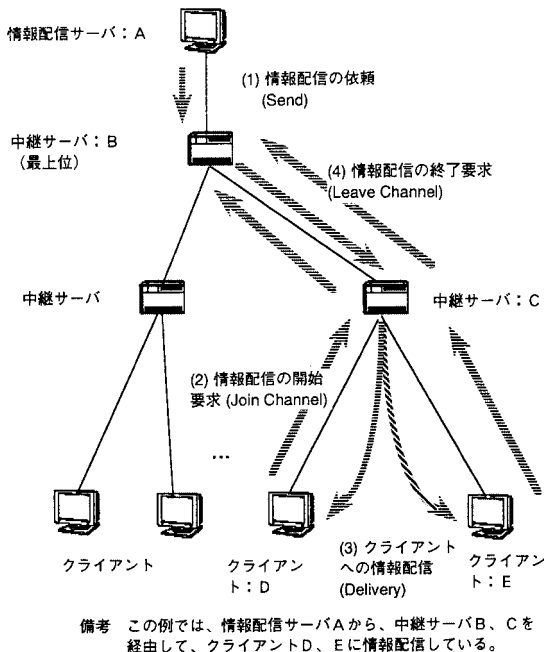


図3 プッシュ型情報配信方式の概要

Fig. 3 Scenario for push information delivery.

ただし、中継サーバが上位の中継サーバに Join Channel メッセージを送信するのは、最初に、クライアントまたは下位の中継サーバから Join Channel メッセージを受信した場合のみで、以降、クライアントまたは下位の中継サーバからその配信チャンネル識別子を設定した Join Channel メッセージを受信した場合は、上位の中継サーバには再度送信しない。

Join Channel メッセージを受信した最上位の中継サーバは、資源不足などの問題がなければ、下位の中継サーバまたはクライアントに対して、Join Ack メッセージで応答する。同様に、Join Ack メッセージを受信した中間の中継サーバは、下位の中継サーバまたはクライアントに対して、Join Ack メッセージで応答する。Join Ack メッセージを送信した時点で、中継サーバは、下位の中継サーバまたはクライアントとの間で、配信チャンネルを設定する。

4.3 中継サーバからの情報配信

配信チャンネルのルートとして定義された最上位の中継サーバは、情報配信サーバからの情報を Send メッセージとして受信する。

最上位の中継サーバは、受信した Send メッセージに設定された配信チャンネル識別子により識別される配信チャンネルを、下位の中継サーバまたはクライアントとの間で設定しているか否か調べる。配信チャンネルを設定している場合、その配信チャンネルを利用して、各

中継サーバまたはクライアントに対して、受信した情報を Delivery メッセージに変換して送信する。

同様に、上位の中継サーバから Delivery メッセージとして情報を受信した中継サーバは、Delivery メッセージに設定された配信チャンネル識別子により識別される配信チャンネルを、下位の中継サーバまたはクライアントとの間で設定しているか否か調べる。配信チャンネルを設定している場合、その配信チャンネルを利用して、各中継サーバまたはクライアントに対して、受信した情報を Delivery メッセージとして送信する。

このように、情報配信サーバから送信された情報は、配信チャンネルのルートとして定義された最上位の中継サーバおよび中間の中継サーバを経由して、すべてのクライアントに、リアルタイム性を損なわずに同時に配信される。

4.4 クライアントへの情報配信の終了

情報配信の終了を要求するクライアントは、離脱する配信チャンネルを識別する配信チャンネル識別子を設定した Leave Channel メッセージを、そのクライアントが接続している中継サーバに送信する。

クライアントから Leave Channel メッセージを受信した中継サーバは、クライアントとの間で設定した配信チャンネルを解放する。

解放した配信チャンネルが、クライアントまたは下位の中継サーバとの間で設定した最後の配信チャンネルである場合、中継サーバは、その配信チャンネルを識別する配信チャンネル識別子を設定した Leave Channel メッセージを上位の中継サーバに送信する。下位の中継サーバから Leave Channel メッセージを受信した中継サーバは、下位の中継サーバとの間で設定した配信チャンネルを解放する。

このように、Leave Channel メッセージは、クライアントから、中間の中継サーバを経由して、最終的にはその配信チャンネルのルートとして定義された最上位の中継サーバまで順次伝搬される。

5. IP マルチキャスト 通信ネットワークとの相互接続

IP マルチキャスト通信は、現在実験段階にあるが、今後、プッシュ型情報配信のための基盤技術として広く利用されることが期待されている。したがって、我々が提案するアーキテクチャが IP マルチキャスト通信ネットワークと相互接続できることは、非常に重要である。IP マルチキャスト通信ネットワークは本質的に信頼性の低いネットワークなので、以下では、IP マルチキャスト通信ネットワークと本アーキテクチャの

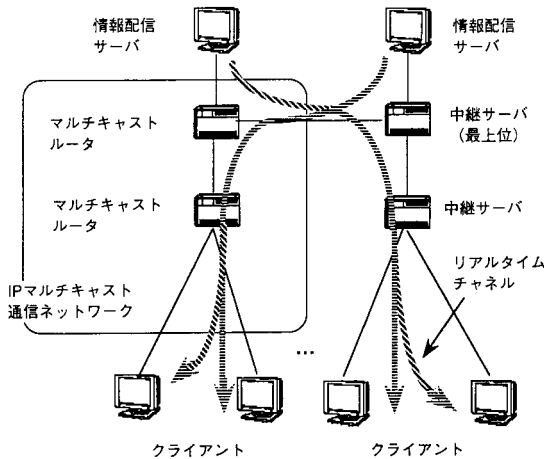


図4 IPマルチキャスト通信ネットワークとの相互接続
Fig.4 Interworking with IP multicast networks.

リアルタイムチャンネルとの相互接続方式について述べる(図4)。本方式では、IPマルチキャスト通信ネットワークとの接続は、配信チャンネルのルートとして定義された最上位の中継サーバに行う。

本アーキテクチャの高信頼チャンネルと相互接続するためには、高信頼マルチキャストが必要となる。高信頼マルチキャストと高信頼チャンネルとの相互接続方式については、8章で述べる。

5.1 IPマルチキャスト通信ネットワーク上の情報配信サーバからの情報配信

IPマルチキャスト通信ネットワーク上の情報配信サーバから、本アーキテクチャのクライアントに情報配信する方式について述べる。

情報配信の開始を要求するクライアントは、Join Channel メッセージを中継サーバに送信する。このJoin Channel メッセージは、配信チャンネルのルートとして定義された最上位の中継サーバまで伝播される。

Join Channel メッセージを受信した最上位の中継サーバは、IPマルチキャスト通信ネットワークに対して、IGMP³⁾のMembership Report メッセージを送信して、IPマルチキャストデータグラムの受信を開始する。このとき、受信したJoin Channel メッセージに設定された配信チャンネル識別子の上位32ビットを、受信を開始するIPマルチキャストデータグラムの宛先グループアドレス(クラスDのIPアドレス)として使用する。

その後、最上位の中継サーバは、受信したIPマルチキャストデータグラムをDelivery メッセージに変換して、下位の中継サーバに送信する。Delivery メッセージは、中間の中継サーバを経由してクライアント

まで配信される。

情報配信の終了を要求するクライアントは、Leave Channel メッセージを中継サーバに送信する。このLeave Channel メッセージは、配信チャンネルのルートとして定義された最上位の中継サーバまで伝播される。

Leave Channel メッセージを受信した最上位の中継サーバは、IPマルチキャスト通信ネットワークに対して、IPマルチキャストデータグラムの受信終了を通知する。

5.2 IPマルチキャスト通信ネットワーク上のクライアントへの情報配信

本アーキテクチャの情報配信サーバから、IPマルチキャスト通信ネットワーク上のクライアントに情報配信する方式について述べる。

配信チャンネルのルートとして定義された最上位の中継サーバは、情報配信サーバから受信したSend メッセージを、IPマルチキャストデータグラムに変換して、IPマルチキャスト通信ネットワークに送信する。このとき、受信したSend メッセージに設定された配信チャンネル識別子の上位32ビットを、送信するIPマルチキャストデータグラムの宛先グループアドレスとして使用する。IPマルチキャストデータグラムは、その宛先グループアドレスを持つIPマルチキャストデータグラムを受信中のクライアントまでルーティングされる。

6. 実装

提案したアーキテクチャとプロトコルに基づくプロトタイプシステムを、WS、PC上で開発した(図5)。

図5に示すように、情報配信サーバ、クライアントの機能はライブラリ(UNIX、Windows 95/NT)として、中継サーバの機能はデーモン(UNIX)およびサービス(Windows NT)として実装した(表2)。加えて、クライアントの機能は、Javaのクラスライブラリとしても実装した。UNIX上では、通信インタフェースとして、UNIXの標準的なソケットインタフェースのみを使用して実装したため、UNIXの種別に依存することなく動作可能である。また、UNIX環境からWindows 95/NT環境への移植も容易に実現できた。中継サーバが、IGMPに準拠したIPマルチキャスト通信の送信ホストおよび受信ホストの機能をサポートしているならば、5章で述べたIPマルチキャスト通信ネットワークとの相互接続が可能である。

情報配信サーバまたはクライアントは、中継サーバと同一システム内に存在してもよい。この場合、情報配信サーバまたはクライアントと中継サーバの間は、

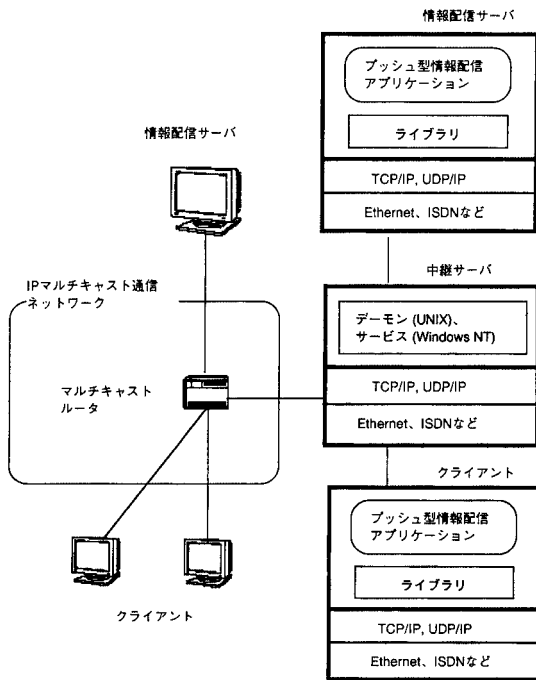


図5 プロトタイプシステムの実装

Fig. 5 Implementation of prototype system.

表2 プロトタイプシステムの実装環境

Table 2 Implementation environment of prototype system.

| 機能 | 実装環境 |
|---------|--|
| 情報配信サーバ | UNIX (ライブラリ)、 Windows 95/NT (ライブラリ) |
| 中継サーバ | UNIX (デーモン)、 Windows NT (サービス) |
| クライアント | UNIX (ライブラリ)、 Windows 95/NT (ライブラリ)、 Java (クラスライブラリ) |

ソケットインタフェースを利用したローカルなプロセス間通信を行う。

ブッシュ型情報配信アプリケーションに提供している主なライブラリ関数(UNIX, Windows 95/NT)を表3に示す。

7. IP マルチキャスト通信アプリケーションの移植実験

提案したブッシュ型情報配信のアーキテクチャの有効性を確認するために、以下に示すIPマルチキャスト通信アプリケーションを、6章で述べた本アーキテクチャのプロトタイプシステム上に移植した。

- vic⁴⁾, vat, RMTTP⁵⁾

vic, vatは、Mbone上のIPマルチキャスト通信実

表3 ブッシュ型情報配信アプリケーションに提供しているライブラリ関数

Table 3 Library functions for push information delivery application.

| 名前 | 機能 |
|-----------------|---------------------------------|
| dcmp_connect | 情報配信サーバが、ルートとして定義された中継サーバと接続する。 |
| dcmp_disconnect | 情報配信サーバが、中継サーバとの接続を切断する。 |
| dcmp_send | 情報配信サーバが、中継サーバに情報配信を依頼する。 |
| dcmp_rcv | クライアントが、中継サーバから情報を受信する。 |
| dcmp_join | クライアントが、中継サーバに情報配信の開始を要求する。 |
| dcmp_leave | クライアントが、中継サーバに情報配信の終了を要求する。 |

験で広く使用されているマルチメディア会議システムであり、vicはビデオの配信を、vatはオーディオの配信を行う。年3回開催されているIETF会合の様子は、vic, vatを使用して、Mbone上で全世界に配信されている。RMTTPは、NTT情報通信研究所と日本IBM東京基礎研究所が共同で開発した高信頼マルチキャスト通信プロトコルである。

表3のライブラリ関数は、ソケットインタフェースで定義しているIPマルチキャスト通信のためのシステムコールとほぼ1対1に対応付けが可能である。したがって、移植作業は、各アプリケーションがIPマルチキャスト通信のためのシステムコールを呼び出している部分を、表3のライブラリ関数で置き換えることでほぼ機械的に行うことができた。

各アプリケーションは、本アーキテクチャのプロトタイプシステム上では、IPマルチキャスト通信の代わりにリアルタイムチャンネルを使用して情報を配信する。

5章で述べた方式に従い、本アーキテクチャ上に移植されたアプリケーション(たとえば、vic, vat)は、IPマルチキャスト通信を利用したもとのアプリケーションとの間で相互通信が可能である。たとえば、vic, vatを用いてMbone上で配信されたIETF会合の様子は、中継サーバを経由して、IPマルチキャスト通信をサポートしていないネットワーク上に接続された本アーキテクチャのクライアント上で受信することが可能である。

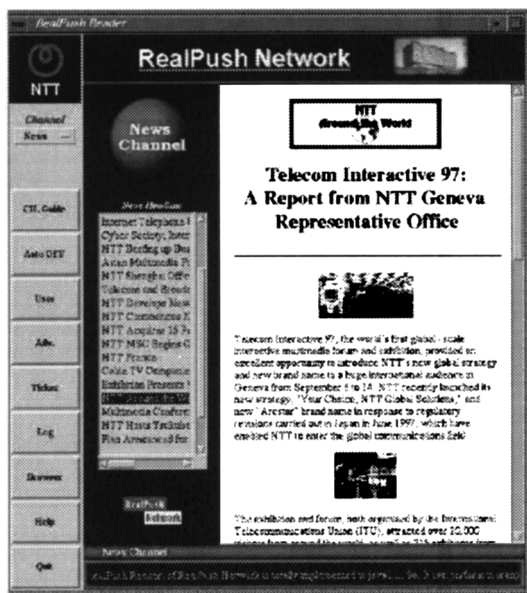


図6 RealPush Reader の画面例
Fig.6 GUI of RealPush Reader.

8. コンテンツ配信システムへの適用

8.1 コンテンツ配信システム (RealPush Network) の概要

我々は、IP マルチキャスト通信を利用したコンテンツ配信システム (RealPush Network) を開発した⁶⁾。RealPush Network は、情報配信サーバ (RealPush Server)、クライアント (RealPush Reader) および IP マルチキャスト通信ネットワークから構成される。

RealPush Server は、配信チャンネルを利用して、ニュース記事、広告情報、ティッカー情報などの様々なコンテンツを配信する。コンテンツは、IP マルチキャスト通信ネットワークを通じて同時にすべてのクライアントに誤りなく配信される。ユーザは、RealPush Reader を利用して、配信チャンネル情報の受信、購読する配信チャンネルの選択、配信されたコンテンツの閲覧などを行う。RealPush Reader の画面例を図6に示す。

RealPush Network のプロトコルは、コンテンツ配信系と配信チャンネルアナウンス系から構成されている (図7)。コンテンツ配信には、我々が開発した IPP (Item Publishing Protocol) と呼ぶプロトコルを使用する。IPP は、機能的に HTTP を IP マルチキャスト通信向けに拡張したプロトコルである。IPP メッセージは、配信するコンテンツとコンテンツの属性などを記述したヘッダから構成される。コンテンツの属性としては、コンテンツ名、コンテンツ概要、配信チャ

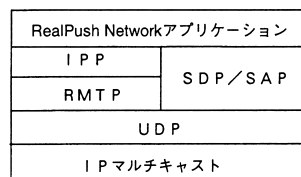


図7 RealPush Network のプロトコル構成
Fig.7 Layer model of RealPush Network.

ネル名、メディアタイプ (コンテンツの MIME タイプ) などがある。IPP メッセージは、RMT P を利用して、同時にすべてのクライアントに誤りなく配信される。配信チャンネルアナウンスには、Mbone で広く使用されている SDP/SAP⁷⁾ を利用している。RealPush Server は、SDP/SAP を利用して、配信チャンネル名、配信チャンネルの概要、使用するマルチキャストアドレス/ポート番号などの配信チャンネル情報を、定期的にクライアントに配信している。

8.2 RealPush Network の移植実験

RealPush Network の適用領域を、IP マルチキャスト通信をサポートしているネットワークから、IP ユニキャスト通信のみをサポートしている現在の大部分のインターネットへと拡大するために、RealPush Network を、6章で述べた本アーキテクチャのプロトタイプシステム上に移植した。

運用コストなどを考慮して、RealPush Server は共通化を図ることとし、RealPush Reader のみを本アーキテクチャのプロトタイプシステム上に移植した。IP マルチキャスト通信ネットワーク上の RealPush Server が配信したコンテンツは、配信チャンネルのルートとして定義された中継サーバで中継され、本アーキテクチャ上に移植された RealPush Reader に配信される (図8)。

RealPush Network の IPP メッセージは、誤りなく配信する必要があるため、本アーキテクチャ上では高信頼チャンネルを利用して配信する。RealPush Network の SDP/SAP メッセージは、本アーキテクチャ上ではリアルタイムチャンネルを利用して配信する。本アーキテクチャ上での RealPush Network のプロトコル構成を図9に示す。

RealPush Reader は Java で記述されているため、RealPush Reader の移植は、RealPush Reader が使用している IP マルチキャスト通信のためのクラスライブラリを、6章で述べた Java クライアントのクラスライブラリで置き換えることにより行った。

IPP メッセージは、中継サーバで RMT P ゲートウェイ方式により中継される。すなわち、中継サーバは、

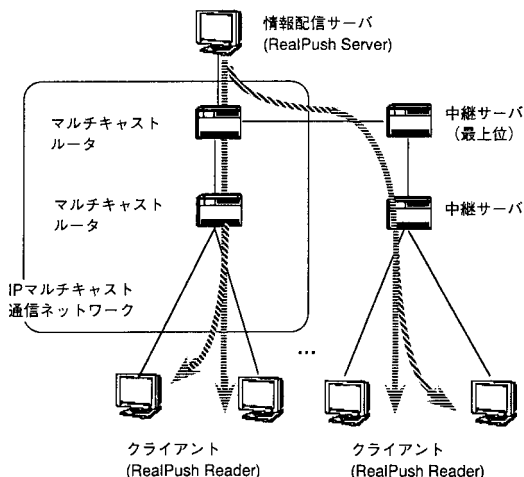


図 8 RealPush Network への適用
Fig. 8 Application to RealPush Network.

| RealPush Networkアプリケーション | |
|--------------------------|--------------------|
| IPP | SDP/SAP |
| DCMP (高信頼チャンネル) | DCMP (リアルタイムチャンネル) |
| TCP/IP、UDP/IP | |

図 9 本アーキテクチャ上に移植した RealPush Network のプロトコル構成
Fig. 9 Layer model of RealPush Network over proposed architecture.

| ゲートウェイ機能 | |
|----------------|-----------------|
| RMTP | DCMP (高信頼チャンネル) |
| UDP/IP マルチキャスト | TCP/IP |

図 10 RMTPゲートウェイのプロトコル構成
Fig. 10 Gateway between RMTP and DCMP (reliable channel).

RealPush Server から RMTP により受信した IPP メッセージを、高信頼チャンネルを利用して本アーキテクチャ上に移植された RealPush Reader に配信する (図 10)。

SDP/SAP メッセージは、中継サーバで 5 章で述べた IP マルチキャスト通信ネットワークとの相互接続方式に従い中継される。すなわち、中継サーバは、RealPush Server から UDP/IP マルチキャストにより受信した SDP/SAP メッセージを、リアルタイムチャンネルを利用して本アーキテクチャ上に移植された RealPush Reader に配信する。

9. システム実現上の課題

6 章, 7 章および 8 章で本アーキテクチャに基づくプロトタイプシステムの実装実験について述べた。本章では、本システムの実装実験の経験に基づき、その実用化に向けて解決すべき課題とその解決案について述べる。

9.1 中継サーバの負荷分散

現在の実装では、クライアントが接続する中継サーバの IP アドレスとポート番号をクライアント内のテーブルに静的に設定している。中継サーバが接続する上位の中継サーバを指定する方法も同様である。この方法では、クライアント数が増加した場合に、負荷均衡化などの観点から問題となる。たとえば、中継サーバの IP アドレスが変更になった場合のアドレス再周知にかかるコストは膨大なものになるし、また、特定の中継サーバにアクセスが集中した場合の回避も困難である。

そこで、本システムの実用化に向けて、中継サーバの指定方法をホストのドメイン名に変更し、BIND 4.9 以降で採用されているラウンドロビン DNS により負荷分散を実現する予定である。ラウンドロビン DNS では、代表ドメイン名に対し複数の実サーバの IP アドレスを対応付け、DNS 問合せに対して順番に異なった IP アドレスを返すことにより負荷分散を実現している。

9.2 情報配信サーバの IP アドレスの通知方法

本アーキテクチャでは、クライアントに情報配信する IP パケットの送信元アドレスには上位の中継サーバの IP アドレスが設定される。しかしながら、アプリケーションによっては、情報配信サーバの IP アドレスが通知されないと正常に動作しないものがある。

そのため、DCMP に情報配信サーバの IP アドレスをクライアントに通知する機能を設け、ライブラリ関数を利用して、クライアントアプリケーションが情報配信サーバの IP アドレスを取得可能とした。

9.3 セキュリティ

現在の実装では、リアルタイムチャンネルのセキュリティに関して何ら保証していない。したがって、不正なユーザが情報配信サーバまたは中継サーバになりすまし、情報をクライアントに配信することを防ぐことができない。この問題は、DCMP の Send メッセージおよび Delivery メッセージに認証オプションを追加することで解決を図る予定である。

9.4 マルチキャスト/ユニキャストの自動選択

現在のクライアント (RealPush Reader) の実装で

は、マルチキャスト (RMTP) による受信か、ユニキャスト (DCMP) による受信かは、ユーザがマニュアルで選択するようになっている。本来、ユーザには受信手段は透過的にしておくほうがよく、マルチキャスト/ユニキャストの自動選択方法を提供することが望ましい。

今後、本システムの実用化に向け、マルチキャスト/ユニキャストの自動選択方法の検討を進める予定である。具体的には、最初にマルチキャストによる受信を選択し、タイムアウトした場合にユニキャストによる受信を実行する方法などについて検討を進める予定である。

10. ま と め

本論文では、IP ユニキャスト通信のみをサポートしているネットワーク上で「疑似プッシュ型」ではない真のプッシュ型情報配信サービスを実現するためのアーキテクチャとプロトコルを提案した。本アーキテクチャに基づくプロトタイプシステムの実装と、そのコンテンツ配信システム (RealPush Network) への適用についても述べた。

今後は、性能評価、実用化に向けた試行サービスの実施、新しいアプリケーションへの適用などについて検討を進める予定である。

参 考 文 献

- 1) Kumar, V.: *MBone: Interactive Multimedia on the Internet*, New Riders Publishing (1996).
- 2) Ishikawa, N., Kinoshita, S. and Takahashi, O.: An Architecture for Push Information Delivery and Its Application to News Delivery System, *INET'99* (1999).
- 3) Fenner, W.: Internet Group Management Protocol, Version 2, RFC 2236 (1997).
- 4) McCanne, S. and Jacobson, V.: vic: A Flexible Framework for Packet Video, *ACM Multimedia'95* (1995).
- 5) 山内長承, 城下輝治, 佐野哲央, 高橋 修: 高信頼同報バルク転送機構, 情報処理学会論文誌, Vol.39, No.6, pp.2009-2019 (1998).
- 6) Kinoshita, S., Shiroshita, T., Nagata, T., Sano, T. and Takahashi, O.: The RealPush Network: A New Push-Type Content Delivery System Using Reliable Multicasting, *IEEE Trans. Consumer Electronics*, Vol.44, No.4 (1998).
- 7) Handley, M. and Jacobson, V.: SDP: Session Description Protocol, RFC 2327 (1998).
(平成 11 年 5 月 7 日受付)
(平成 11 年 12 月 2 日採録)



石川 憲洋 (正会員)

1978 年京都大学工学部情報工学科卒業。1980 年同大学院工学研究科情報工学専攻修士課程修了。同年日本電信電話公社 (現 NTT) 入社。1999 年から, NTT 移動通信網 (株) (NTT ドコモ) マルチメディア研究所に所属。ATM, インターネットプロトコル, マルチメディア通信, モバイルインターネット等の研究開発に従事。



木下 真吾 (正会員)

1968 年生。1991 年大阪大学基礎工学部物性物理工学科卒業。同年 NTT に入社。以来, フォールトトレランス, 分散コンピューティングアーキテクチャ, マルチキャスト通信プロトコルの研究開発に従事。現在, NTT 情報流通プラットフォーム研究所研究主任。IEEE 会員



高橋 修 (正会員)

1975 年北海道大学大学院情報工学専攻修士課程修了。同年, 日本電信電話公社入社。1999 年より, NTT 移動通信網 (株) (NTT ドコモ) マルチメディア研究所主幹研究員。主としてモバイルマルチメディア通信サービスとプロトコルの研究開発に従事。電子情報通信学会会員。