

Pure-P2Pネットワークを用いたユーザ指向型のストリーミングリレー システムの提案

中村 暢大†, 高濱 聡一郎†, バロリ レオナルド††, 馬 建華†††

† 福岡工業大学大学院工学研究科

E-mail: {mgm05008, mgm05007}@ws.ipc.fit.ac.jp

†† 福岡工業大学情報工学部

E-mail: barolli@fit.ac.jp

††† 法政大学情報科学部

jianhua@k.hosei.ac.jp

概要

現在、様々なStreamingの研究されており、実際に利用されている。しかし、コンテンツの種類は多くはないし、リレーは安定していない。その原因の一つとして、配信するためにする行程がとても難しく敷居が高いことと、リレーするためには、Firewallのポートをあける必要があることが関係していると考えられる。本研究では、これらの問題を解決するために、JXTAの特徴である、PureP2Pのネットワークの構築とNAT/Firewallを簡単に通る仕組みとFlashのGUIに対する柔軟なデザインと簡単なビデオや音声の取り込みといった特徴を利用して、新しくリレー形態やコンテンツを流しているPeerの情報や内容などを知らせる方法についても述べる。

A User-oriented Steaming Relay System for Pure P2P Networks

Nobuhiro Nakamura†, Souichirou Takahama†, Leonard Barolli††, Jianhua

Ma†††

† Graduate School of Engineering, Fukuoka Institute of Technology

E-mail: {mgm05008, mgm05007}@ws.ipc.fit.ac.jp

†† Faculty of Information Engineering, Fukuoka Institute of Technology

E-mail: barolli@fit.ac.jp

Faculty of Computer and Information Sciences, Hosei University

jianhua@k.hosei.ac.jp

Abstract

Presently, there is a lot of research about video streaming and also there are many applications. However, there are only few kinds of contents and the relay of data is not steady. The reasons are as follows: the delivery process of video streaming is difficult, the threshold is high, and the Firewall port should be open. In this research, to deal with these problems, we propose a user-oriented streaming system for pure P2P networks. We will implement the proposed system by using JXTA, a architecture to pass easily NAT/Firewall, and Flash for designing and constructing a flexible GUI. We propose a new relay streaming system that can achieve a good delivery of video and voice data.

1. はじめに

近年、インターネットの普及により、個人が持っている情報を送受信したりといった利用が広がっている。さらに、家庭用のパソコンの性能も低価格でありながらも劇的に向上してきており、従来のテキストや静止画像などのデータ容量の小さいコンテンツに加え、音声・動画画像などのより情報量の多いコンテンツも増え、扱うデータ量は増えてきており、全てのファイルをダウンロードしてから視聴するのではなく、ファイルをダウンロードしながら視聴をするストリーミングの配信が行われている。今現在、ストリーミングの配信形態として、Client-Server型のシステムかHybrid-P2P型のシステムに分類される。Client/Server型の特徴としては、サーバーの方の資源に余裕さえあれば、安定して配信が可能である。Hybrid-P2P型[1-7]のアプリケーションのほとんどは、オープンソースでコミュニティによって作成されており、音声や動画などのフォーマットは含められておらず、また、リレーにより、ユーザ数が限定されないといった特徴がある。

2. 現在のストリーミング配信システムの問題点

実際に、動作しているシステムとして、Client/Server型のアプリケーションとしてDarwin Streaming Server, Helix Universal Server, Windows Media Serviceが上げられる。それと、Hybrid-P2P型のアプリケーションでは、PeercastやCoolStreamingが上げられ、実際に利用されている。しかし現在利用されている多くのシステムの多くは以下のような課題があると考えられる。

(1) Client/Server型の場合、Server側で一極集中が起こったり、Hybrid-P2P型では、コンテンツとそれを配信しているPeerの位置情報を管理するために、サーバで一括管理するHybrid-P2P型を採用しており、サーバが必ず必要となり、サーバに障害が発生すると機能停止やServer側のコストがかかる。

(2) P2Pでのストリーミング配信システムを利用しているユーザが少なく、コンテンツの数も少ない。

(3) Hybrid-P2P型の方は、ポートが開いてなかったり、ユーザが消えた時に安定してリレーができなくなり切断が起きやすく、わざわざ、つなぎ直す必要があったりする。

これらの原因として、(1)においては、Client/Server型では、Server側の資源に限界があり、P2P型では、配信されている情報が分散されると、ネットワーク全体で配信されているコンテンツが把握できなくなる。(2)では、動画や音声を配信するために行う行程がとて多く、配信できるものの多くはプラットフォーム依存という点もあり、配信を行うために導入の敷居が高い。(3)においては、ほとんどのコンピュータでファイアウォールを導入しており、ポートを開ける必要があるため、その設定が面倒であるということ。そして、リレー形態としては、Tree型[8-14]を採用して、図1のように、コンテンツを配信するP1のRoot Peerから受信するPeerへとぶら下がるようになる。そのため、コンテンツを探す場合は、上流へ探しにいけばいいのだが、親Peerが消えた時点でどのPeerと接続するかを決めるため毎回探すといったことになるということが原因に上げられる。

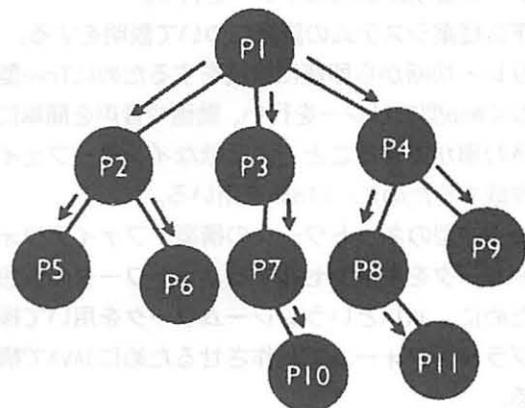


図1. Tree型のリレー形態

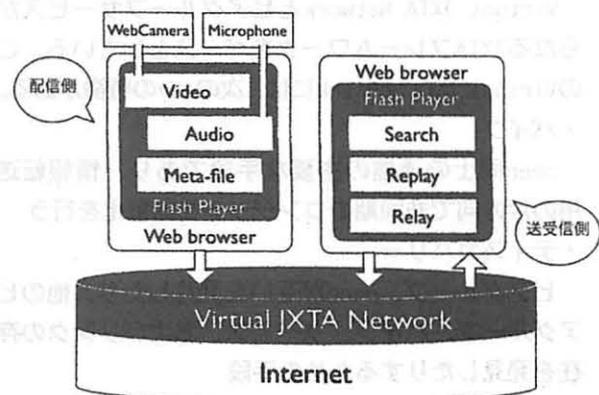


図2. 提案システム

このため、本研究では、サーバーや煩わしい設定を無くし、リレー形態を切断に備えた形態を考え、システム自体を様々なプラットフォームに対応させる事を目的とし、アプローチとして、サーバーに代わる機能、直感的な操作、リレーを切り替える機能、プラットフォーム非依存があげられる。

3. 提案システム

そこで、提案システムでは、図2のようにユーザは、ウェブブラウザ上のFlash Playerから操作をする。

配信側では、配信するコンテンツの内容を記したメタファイルを配ったり、ウェブカメラからビデオ、マイクから音声を取り込んで他の送受信側ヘインターネット上のVirtual JXTA Networkを通して渡す。そして、送受信側では、受け取ったメタファイルからコンテンツを検索して再生したり、他のPeerからリレーの要求があればリレーを行う。

以下に提案システムの詳細について説明をする。

リレー切断から即座に接続をするためにTree型ではなくWeb型でリレーを行い、動画や音声を簡単に取り込む事ができることと、柔軟なインターフェイスを作成するために、Flashを用いる。

Pure-P2P型のネットワークの構築、ファイアウォールやルータをキャンセルするネットワークの仮想化のために、JXTAというフレームワークを用いて様々なプラットフォームで動作させるためにJAVAで構築する。

4. ネットワークアーキテクチャ

Virtual JXTA NetworkとピアグループサービスからなるJXTAフレームワークをベースとしている。このVirtual JXTA Networkには、次の4つの特徴がある。

- ・パイプ

peer同士の通信の主要な手段であり、情報転送用の片方向で非同期のコンジットを抽象化を行う

- ・ディスカバリー

ピアグループ、peerが互いを発見したり、他のピアグループ、パイプ、サービス、およびリンクの存在を発見したりするための手段

- ・リゾルバー

peerが互い同士、peerグループ、パイプまたはサ

ービスを、参照によって間接的に参照することを可能にする

- ・既存のネットワークプロトコルを仮想化

ピアがメッセージを他のどんなピアにでも図3の様に様々なネットワークプロトコルでもXML Messageを運ぶためのEnd Point Protocolとし、図4の様にドライバー化することによって、ファイアウォール、NAT、IPを持たないネットワークでもメッセージを交換する

JXTAは、分散処理、ストレージ、エージェント、コンテンツの配布などP2Pコンピューティングの幅広い範囲をサポートすることができる一般的なフレームワークでありJXTAのオープンソースウェブサイト上では、現在、およそ120のJXTAベースのプロジェクトが行われている。

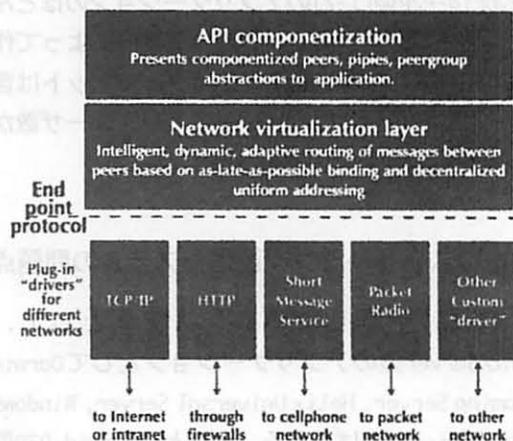


図3. End Point Protocol

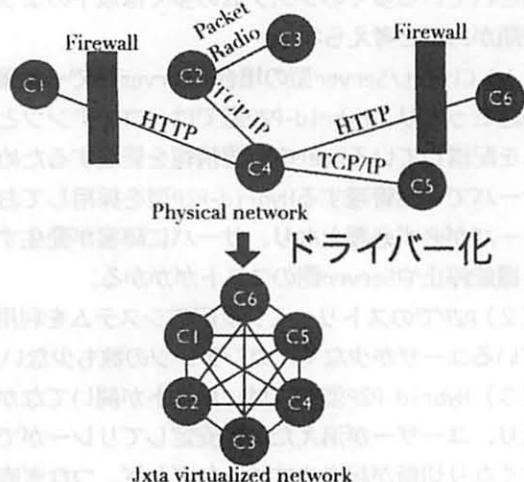


図4. プロトコルのドライバー化

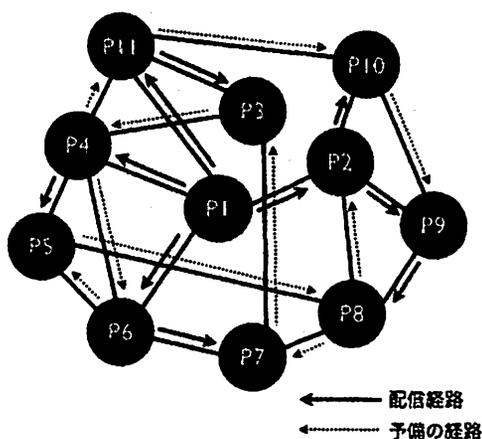


図5. Web型でのリレー

4.1. リレー形態

このシステムでのリレー形態としては、現在多く用いられているTree型のリレー形態ではなく、図5の型のWeb型をとる。Web型の特徴としては、上流も下流も関係なしに、JXTAのサービスによって複数のPeerと接続する。コンテンツの検索は、接続しているPeerに検索依頼を頼むことになり、この接続方法だと、複数接続することができ、リレーを安定させることができると考えるのでWeb型のリレー形態を用いる。リレー形態では、ストリーミングを流す経路以外に予備の経路を用意しておく事で、リレーを切断された時に、即座に予備の経路に切り替える。

4.2. メタファイル

メタファイルとは、コンテンツの在処を教える物である。メタファイルの中身としては、配信PeerのJXTA-ID, コンテンツの要約, コンテンツの詳細を記す。メタファイルの動向としては、クライアントサーバ型やHybrid-P2P型のストリーミングシステムの場合は、メタファイルを検索する場合、サーバが管理している。しかし、今回実装するシステムでは、Pure-P2P型をとるのでサーバは、存在しない。そのため、サーバを必要としない検索方法が必要である。その方法として、2つほど考えられる。

・能動的に検索

検索を行うと、その検索キーワードを含むものが、PeerからPeerへとそれぞれがもっている情報を検索していき、結果がかえってくる。その検索結果から選び出し、リクエストをしてリレーしてもらう。

・受動的に検索

Peerで検索するのではなく、接続した側のPeerは、直接接続しているPeerから、持っているメタファイルを集めて、その中から視聴したいと思うものを選び出して、それから、Peerへリレーしてもらうようにリクエストを出す。

このシステムでは、実際にリレーしているものから選び出す方法を取りたいため、受動的に検索する方法をとる。

4.3. メタファイルの更新

このシステムでは、ある一定条件において、配信されている、下記のような状態になったときにメタファイルの更新を行う。

- 【1】新しくPeerが、ネットワークに参加してきたとき、図6のように、Virtual JXTA Networkに参加してから、サービスによりPeerを探し、Peerが見つかると図のP7のように、P2,P4,P6から全てのメタファイルを取得する。
- 【2】ネットワーク参加時から1分のカウントダウンをし、0になると図7のP2のように、繋がっているP1,P3,P4,P7へメタファイルをマルチキャストする。このカウントダウンは、繰り返して行う。Peerが更新の要求を出すことによっても更新する。Peerが意図的に更新した時も、【1】の条件と同じように、図6のP7のように、P2,P4,P6から新たに全てのメタファイルを取得する。
- 【3】Peerが更新の要求を出すことによっても更新する。Peerが意図的に更新した時も、【1】の条件と同じように、図8のP7のように、P2,P4,P6から新たに全てのメタファイルを取得する。

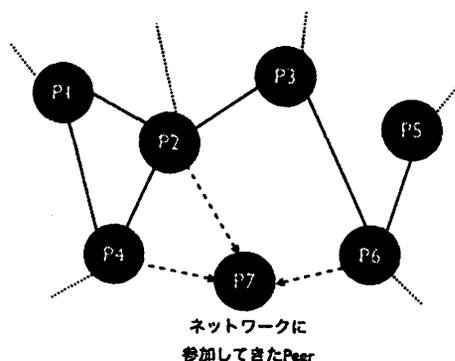


図6. 新しくPeerが参加してきたとき

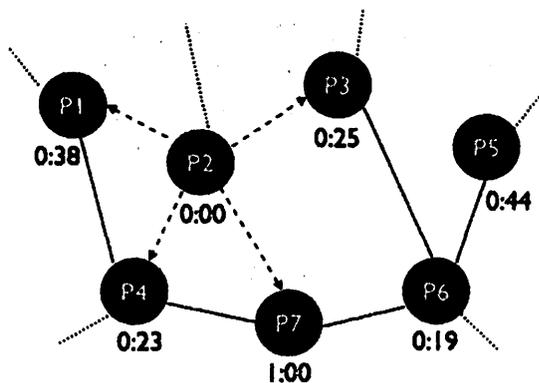


図7. カウントダウンをしているPeer

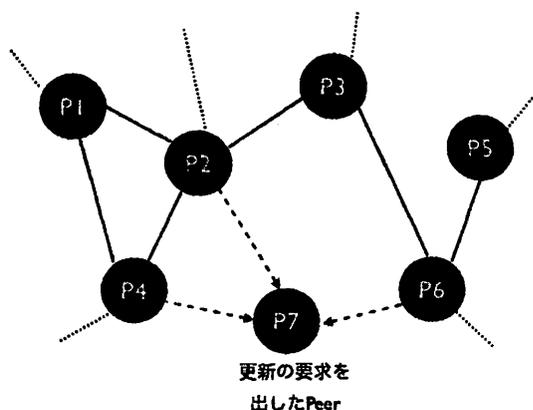


図8. Peerからの更新の要求

5. おわりに

本稿では、Pure-P2Pネットワークを用いたユーザ指向型のストリーミングリレーシステムの提案を行った。今回提案したシステムによって、Pure-P2P型のネットワーク上で、面倒だった設定などをせずに、ストリーミング配信をするシステム。今後の課題としては、提案システムの設計、実装ならびに、パフォーマンスや汎用性についての評価が上げられる。

参考文献

[1] X. Jiang, Y. Dong, D. Xu, B. Bhargava "Gnustream: A P2P Media Streaming System Prototype", Proc of ICMP-2003, 2003
 [2] X. Zhang, J. Liu, B. Li, and T.S.P. Yum, "DONet/CoolStreaming: A Data-driven Overlay Network for Live Media Streaming", Proc. of IEEE INFOCOM'05, Miami, FL, USA, March 2005.

[3] Y.H. Chu, S.G. Rao, and H. Zhang, "A Case for End System Multicast", Proc. of SIGMETRICS'00, 2000.
 [4] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable Application Layer Multicast", Proc. of ACM SIGCOMM'02, 2002.
 [5] D. A. Tran, K. A. Hua, and T. T. Do, "A Peer-to-Peer Architecture for Media Streaming," IEEE Journal on Select. Areas in Comm., Vol. 22, No.1, Jan. 2004.
 [6] K. Sripanidkulchai, A. Ganjam, B. Maggs, H. Zhang, "The Feasibility of Peer-to-Peer Architectures for Large-Scale Live Streaming Application", Proc. of ACM SIGCOMM'04, 2004.
 [7] A. Habib, D. Xu, M. Atallah, B. Bhargava, J. Chuang "Verifying Data Integrity in Peer-to-Peer Media Streaming", Technical Report CERIAS-02-39, Dec 2002.
 [8] Due A. Tran, Kien A. Hua, T. Do "ZIGZAG: An Efficient Peer-to-Peer Scheme for Media Streaming", Proc. of IEEE INFOCOM-2003, 2003.
 [9] M. Hefeeda, A. Habib, B. Botev, D. Xu, B. Bhargava, "PROMISE: Peer-to-Peer Media Streaming Using CollectCast", Proc. of ACM Multimedia 2003, pp. 45--54, Berkeley, CA, November 2003.
 [10] Tai T. Do, Kien A. Hua, M. A. Tantaoui, "P2VoD: Providing Fault Tolerant Video-on-Demand Streaming in Peer-to-Peer Environment", Proc. of IEEE International Conference on Communications, Paris, June 2004.
 [11] S. Khan, "Multiple Description Streaming in Peer-to-Peer and Content Delivery Networks", Proc. of ATNAC-2003, 2003.
 [12] 三村 和, 中内 清秀, 森川 博之, 青山 友紀 "RelayCast: ピアツーピア型ストリーム配信のためのミドルウェア", 電子情報通信学会技術報告, IN2002-42, July 2002.
 [13] R. Rejaie, A. Ortega, "PALS: Peer-to-Peer Adaptive Layered Streaming", Proc. of NOSSDAV-2003, 2003.
 [14] J. Zhang, L. Liu, C. Pu, M. Ammar, "Reliable End System Multicasting with a Heterogeneous Overlay Network", CERCS Technical Report git-cercs-04-19.