

4T-8

JXTA を用いた P2P 型グループウェアの試作 - ビデオ会議システム - Video meeting system of P2P type groupware using JXTA

高柳智行†

山本富士男†

宮崎剛†

Tomoyuki TAKAYANAGI

Fujio YAMAMOTO

Tsuyoshi MIYAZAKI

1. はじめに

現在のグループウェアは、利用可能な範囲が限られているため、社外などからの利用ができない場合が多い。しかしながら実際には、社内だけでなく出先など社外からもグループウェアを利用したい場合も少なくない。また、外部にグループウェアサーバを公開する場合はリスクや費用が伴う。さらに問題点として、サーバに負荷が集中し、サーバがダウンした場合にグループウェア自体の機能が失われ、業務が停止してしまう恐れがある。そのため、1対多などの高負荷なビデオ会議機能が実装されていないのが通例である。

そこで本研究では、これらの問題点の改善を目的として、まずP2Pのフレームワーク技術であるJXTA^[1]を利用して、既存のネットワーク上にオーバーレイネットワークを構築する。次に応用サービスとして、リアルタイム性に厳しく送受信容量の大きな1対多のビデオ会議機能を実装する。これは、単一ネットワークだけでなくあらゆるネットワーク環境下、あらゆるプラットフォーム間で相互利用できる、中央サーバを必要としないPure型P2Pコミュニケーショングループウェアの中核となるものである。

2. 設計方針

本システムで実現するサービスは現在最も多く使われているインスタントメッセージ、ファイルの送受信および、音声通話、ビデオ会議の4つであり、1対1、グループのそれぞれで実現する。本システムの設計方針を次に述べる。

(1) ユーザの利用環境に非依存

Javaで開発し、オーバーレイネットワークを構築することにより、利用するユーザのOSや、ユーザの接続するネットワーク形態にかかわらず、あらゆるネットワーク環境下、あらゆるプラットフォーム間で相互利用することができる。

(2) 特定のPeerに非依存

特定のPeerに依存しないP2Pプロトコルを使用することにより、障害に強いネットワークを構築することができる。ネットワークの負荷を分散することができる。

(3) 利用制限を設けない

P2Pプロトコルにより、リソースを分散することができるのでサーバに対するリソース不足を補うことができる。これによって、同時に利用できるユーザ数を拡大することや、容量の大きなファイルを送受信することが可能となる。

3. 設計

3.1 全体構成

本システムでは、P2Pネットワークに参加・離脱、Peer間リンクの確立、ユーザ管理をする部分を通信層とし、グループウェアのサービスを実現する部分をアプリケーション層とする。システムの全体構成を図.1に示す。

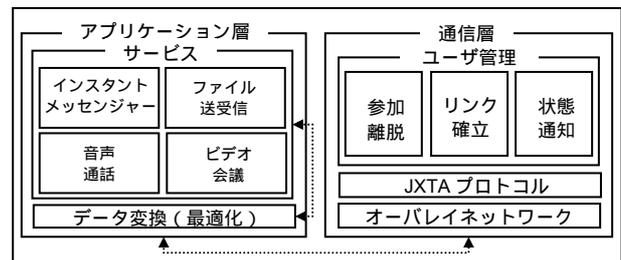


図.1 システム全体構成

3.2 通信層

通信層の主な構成は以下の3つである。

3.2.1 オーバレイネットワーク

アプリケーションレベルのネットワーク空間であるオーバーレイネットワークを構築することにより、ファイアウォールやNATなどのセキュアな環境下でもネットワーク環境を意識することなく通信を可能にする。

3.2.2 JXTAプロトコル

P2Pネットワークの構築にはJXTAを使用する。JXTAはプラットフォームに非依存のプロトコルであり、P2Pネットワークに必要な基本機能(Peerの発見、検索、Peerのグループ化など)を備えており、特定のPeerに依存しないPure型P2Pネットワークを構築することができる。

3.2.3 ユーザ管理

ユーザIDや名前などの情報の登録、Peer間のリンク確立などを管理する。ネットワークに参加する際にはユーザ情報を取得する。ユーザ管理部はユーザ情報を常に最新にするために、ネットワークに参加するPeerを監視し、あらたにネットワークに参加したPeerを発見した場合は現在保持しているユーザ情報を送信する。また、自分自身のユーザ情報が更新された場合も隣接Peerに情報を送信する。JXTAでは、リンク確立(パイプの接続)の際にパイプIDと呼ばれるユニークなIDを利用して、接続するPeerやサービスを指定する。利用する全てのパイプIDはファイルとして保持するか、ユーザ情報に組み込むなどして事前に取得する必要がある。本システムでは、サービスの種類が多いことや、多数のPeerとの接続を想定しているため、従来の方法では効率的ではない。そこで、ユーザIDから事前に決めた法則でパイプIDを生成す

† 神奈川工科大学大学院工学研究科
Department of Information and Computer Sciences,
Kanagawa Institute of Technology

ることにより、パイプ ID を送受信する必要がなくなり、トラフィックを軽減できる。

3.3 アプリケーション層

アプリケーション層の主な構成は、データ変換部分とインスタントメッセージ、ファイル送受信、音声通話、ビデオ会議からなる。

3.3.1 データ変換部分

サービスが扱うデータは文字、ファイル、音声、映像であり、このままでは送受信が非効率であるため、それぞれ適切なデータ形式、サイズに分割し、ヘッダ情報を付加する。

3.3.2 ファイル送受信

P2P 型のシステムでは、送信元 Peer から送信先の Peer までの経路が毎回同じとは限らないため、パケット損失や遅延などの信頼性の低下が見られる。そこでグループの場合は、一定時間経過後に受信側が足りないパケットを再度要求することでこれを解決する。また 1 対 1 の場合は図. 2 に示すように、送信側が分割されたパケットに通し番号を付加し、受信側ではその通し番号とデータサイズを足したものを応答番号として送り返す。そして次のパケットに応答番号を付加して送信する。データや応答番号が消失した場合は、一定時間内に応答番号が受信されないため、送信側が再度同じパケットを送信することにより解決する。

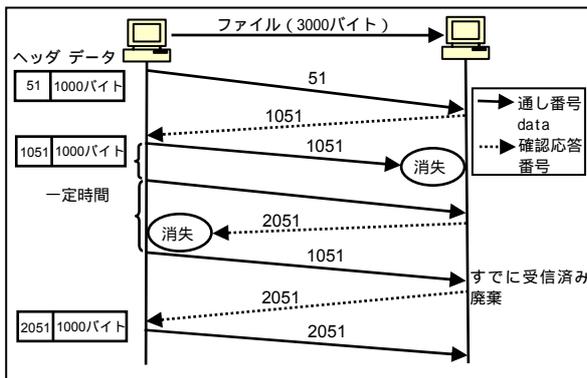


図. 2 ファイル送受信

3.3.3 ビデオ会議

送信側では、音声および動画データは JMF (Java Media Framework^[3]) を用い、マイクと Web カメラにより取得し、適切な形式に変換し、送信する。受信側では、同じく JMF を用いて、音声および動画データとして、スピーカーとディスプレイに出力する。本研究で扱う音声の品質については、一般的な固定電話の音質とされるサンプリング周波数 8.0[kHz]、量子化ビット数 8[bit]、モノラルの PCM で圧縮した音声データを採用する。また動画の品質については、QVGA (320 × 240) サイズの 10[f/s] で、1 フレームごとに JPEG 圧縮をしたデータを採用する。ビデオおよび音声パケットには、ヘッダ情報に通し番号を付加する。受信側で直前に受け取ったパケットの通し番号を確認し、それよりも通し番号が小さい場合は廃棄する。これによりパケットの到着順が、送信した順でない場合も違和感なく出力可能になる。

4. 実験と評価

4.1 実験

実験環境としては、図. 3 のようにそれぞれ異なるネットワーク環境で行う。互いの Peer の探索やリンクの確立に時間がかかる場合があるため、すべての Peer 情報が発見され、必要なリンクが確立した状態からの実験とする。また送出するビデオパケットには、インターネット時刻サーバと同期させた時刻を付加し、受信側で遅延の測定をする。

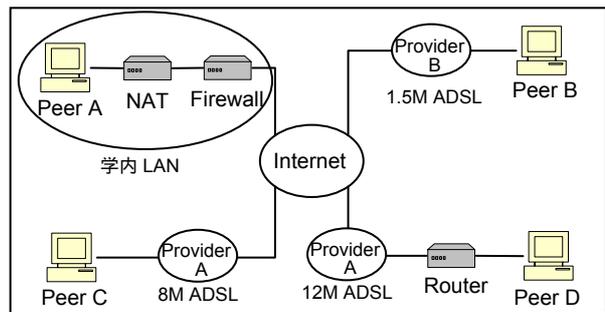


図. 3 実験環境

4.2 評価

P2P 型のシステムとしての動作に関しては、想定どおりに動作した。システムに参加した Peer は所属するサブネットが異なり、またプロバイダの提供するサーバに接続することによって、間接的にインターネットへの接続を行う Peer である。そのため、自身が静的なアドレスは持っておらず、接続に際してアドレスが動的に割り当てられる。よって、それぞれの Peer が通信を行う際には、NAT を通過する必要があったが、その事による動作への支障はなかった。

サービスとしてのビデオ会議機能では、4 台の Peer とも同時に機能し、Peer A を除いては遅延時間も 1.7-4.3[s] と問題なく動作の確認ができた。しかし各 Peer と Peer A の通信遅延時間が 6.7-19.6[s] と大きいことがわかった。

5. まとめ

本研究では、JXTA を利用しオーバーレイネットワークを構築し、ユーザがどのような環境にいても利用できるグループウェアを試作した。さらに、1 対多のビデオ会議機能を実装することにより、高負荷なアプリケーションについても、多少の遅延はあるが問題なく利用できることを確認することができた。

本研究では、それぞれの Peer が発見され、リンクが確立した状態からの評価を行ったが、P2P の性質上、通信相手の探索や、Peer 同士のリンク確立までに時間がかかる場合や、リンクの確率に失敗する場合もある。そこで今後は、効率的な Peer の公開探索や、的確で短時間なリンクの確立についての検討を予定している。

参考文献

- [1] Project JXTA: <http://www.jxta.org/>
- [2] Brendon J. Wilson, "JXTA のすべて—P2P Java プログラミング", 日経 BP 社, 2003
- [3] Rob Gordon, Stephen Talley, "JMF : Java Media Framework", Pearson Education, 1999