

## P2P-Web連携システムにおけるソフトウェア構成法

酒井 孝次<sup>†</sup>, 木村 正二<sup>†</sup>, 須永 宏, 黒川 章

日本電信電話株式会社 NTTネットワークサービスシステム研究所

E-Mail<sup>†</sup> : { sakai.koji, kimura.shoji }@lab.ntt.co.jp

**概要:** 本研究では、ユーザ端末にWebブラウザのみを必要とするP2PネットワークへのP2Pアプリケーションサーバの適用を提案し、そのサービス実現方式、およびソフトウェア構成法を検討する。提案するP2Pアプリケーションサーバは、ユーザを収容管理し、ユーザへのフロントエンドとしてWebアプリケーションサーバ機能を備え、さらにバックエンドとしてそれ自身がP2Pネットワークの1ピアとして参加するものであり、アクセスユーザに対して様々なP2Pサービスを提供するものである。P2Pアプリケーションサーバの適用により、コンテンツやリソースの分散管理が可能となり、管理範囲を特定できるというP2Pシステムのメリットが得られる。

## A Study on the Composition Method of the P2P-Web Coupling System

Koji Sakai, Shoji Kimura, Hiroshi Sunaga and Akira Kurokawa

NTT Network Service Systems Laboratories, NTT Corporation

**Abstract:** This paper proposed a software composition method of the P2P-Application server system. The proposed P2P-Application server providing Web application server functions for Web-client users joins P2P network as one of peers. It is useful to provide various P2P services to thin-clients, which have only Web browsers without a specific application of P2P. And the P2P-Application server has great merits to decentralize managements and controls of various contents and resources.

In this paper, it is suggested clarifying functions for the P2P-Application server, a method of applying software agents for providing services suited each user's preferences and so on.

### 1 はじめに

ブロードバンドネットワーク上のキラーアプリケーションとして注目される P2P システムは、Napster<sup>(4)</sup>, gnutella<sup>(6)</sup>による音楽ファイル交換において脚光を浴びたが、分散コンピューティングやコラボレーションツールなど、その適用領域は非常に幅広く、様々な開発、研究が行われている<sup>(7),(11),(12),(13)</sup>。

一方、インターネット業界に目を向けると、もともとWWW (World Wide Web) により全世界に普及したと言っても過言でなく、Webを取り巻く関連技術の発展はめざましく、今後もWebサービス<sup>(1),(2),(3)</sup>などの台頭により、様々なサービスやビジネスはWebを中心として展開していくことは容易に想像できる。

そこで本研究では、今後もインターネット上にグローバルに発展する可能性が高い“P2P”と“Web”の連携システムを提案し、その提供サービスやソフトウェア実現方式を検討する。

以下、2節にて本研究の背景を述べ、3節において本研究の目的と検討すべき課題を示す。4節では、本稿で提案するP2Pアプリケーションサーバの必要機能、ソフトウェア構成とソフトウェアエージェントの適用方法を明確にする。5節にていくつかのP2Pサービスの実現例を示し、最後に6、7節において、関連技術と本研究の位置付けを示し、本稿のまとめと今後について言及する。

なお本稿では、特に断りがない限り、P2Pネットワークにて流通するリソースやコンテンツを総称して、単に「コンテンツ」と記述する。

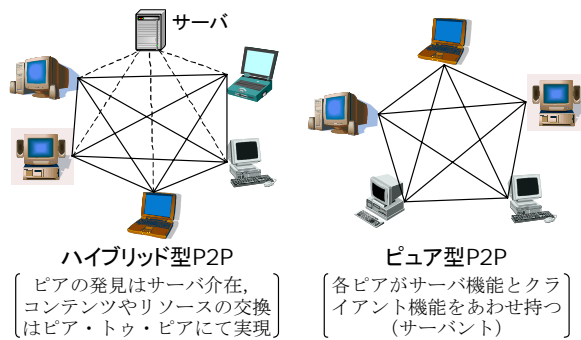


図 1 P2Pネットワークモデル

## 2 P2Pネットワークと本研究の背景

### 2.1 P2Pネットワークの特徴

P2Pのメリットは、「情報の分散化」, 「物理的に広く分散されたリソースを1ヶ所に存在しているように見せるという『仮想的な集中化』」などがあるが, 分散空間から所望のリソースを発見する機能が重要となる. P2Pサービスは, 「ピア発見 (ディスカバリ) → コンテンツ流通 (デリバリ)」という流れが基本になり, ディスカバリを集中サーバで行うモデルを「ハイブリッド型P2P (例, Napster)」, サーバ非介在で各ピア間の連携により検索するモデルを「ピュア型P2P (例, gnutella)」と呼ぶ (図1参照) (13). 両モデルとも, 公開コンテンツはユーザのローカル端末内に格納され, 各端末には専用アプリケーションがインストールされていることが前提となる.

### 2.2 本研究の背景

今後ますますP2Pサービスが発展すると, 携帯電話やPDAなどでもサービス需要が高まり, これらの処理性能 (CPU, メモリ, ストレージ, アプリケーションなど) やネットワーク帯域が制約された端末ユーザに対してもサービスを供給し, さらに上記制約のない端末ピアと混在したP2Pサービスを展開するのは, 既存P2Pモデルでは困難である. また, 現在のP2Pはオンライン端末のみをピアの対象とし, 常時接続が前提となる場合が多

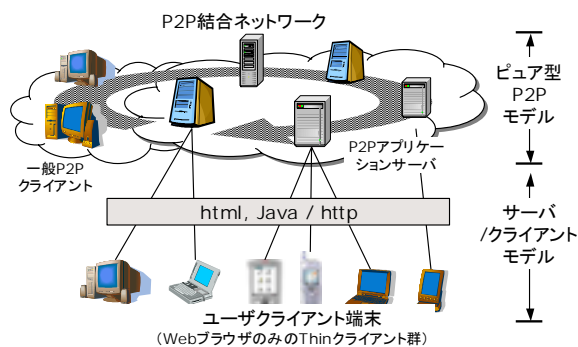


図 2 提案するP2Pネットワークモデル

いが, 依然としてダイアルアップユーザも多く, オフライン時にもユーザが所有するコンテンツを公開させることは不可能である.

一方, 既存のP2Pサービスは, ユーザ端末に専用アプリケーションを必要とし, 末端ユーザはインストールの手間や使用方法を習得する必要がある, といった背景がある.

## 3 本研究の目的

以上から本研究では, ユーザはWebブラウザ (Java実行環境を含む) だけを使用し, P2P専用アプリケーションは必要とせず, Webを通してサービスを展開するP2Pサービスモデルを提案する. その実現のために, 図2に示すような, クライアントユーザを収容管理し, Webアプリケーションサーバ機能を備え, さらにそれ自身がP2P分散ネットワークの1ピアとして参加する『P2Pアプリケーションサーバ (以降, P2P-ASと呼ぶ)』の適用を提案し, そのサービス実現方式, ソフトウェア構成条件を明らかにし, WebユーザへのP2Pサービス提供イメージを明確にすることを目的とする.

### 3.1 提案するサービスモデル

図2の提案モデルにおいて, P2Pネットワーク構成のために, P2P-AS内ではgnutellaやJXTAなど, 既存のP2Pプラットフォーム, またはプロトコルの利用を前提とする. よって, P2P-ASと一般P2Pピア端末で構成されるピュア型P2Pモデルである.

また, 各P2P-ASには信頼されたユーザのアクセスのみを許可するが, オフラインユーザの所有コンテンツも流通対象とし, コンテンツ公開ユーザはオンライン時に, 収容されるP2P-ASにアップロードする必要があるなど, サービス提供上, 一部制約事項もあるので, 適用領域としては, B2B型, またはイントラネットへの適用が考えられる.

本モデルのメリットとして, 情報やコンテンツの分散管理が可能となり, その管理範囲 (管理ドメイン) を特定できることや, Web上で統合化P2Pサービスを提供できることなどがある.

### 3.2 P2P-AS要求条件と検討すべき課題

一般にP2Pはディスカバリ時のトラフィックが高くなること (13, 17) や, Web上のP2P統合サービスの提供, さらにオフラインユーザも対象とするなどの特徴があることから, P2P-ASへの要求条件は,

- req-1. 多数ユーザを収容するが, ネットワークへのトラフィック圧迫を極力低減すること
- req-2. ユーザ端末には, P2P専用アプリケーションを必要とせず, Webブラウザのみでサービ

ス享受可能とすること

**req-3.** Webアプリケーションサーバとして、既存サーバサイドアプリケーションやWebサービスと、P2Pサービスを連携できること

**req-4.** ユーザ毎の環境や嗜好条件に応じてカスタマイズされたサービスを提供すること

**req-5.** オフラインユーザがもつコンテンツをサーバとしてホスティングし、それら进行操作・制御できること

があり、これら要求条件を満たす P2P-AS 実現のために検討すべき課題には以下がある。

**task-1.** 最適P2Pプラットフォームモデル、およびP2P連携方式の明確化 (req-1, 2, 3)

**task-2.** 汎用かつ柔軟な必要機能、およびソフトウェア構成の検討 (req-3, 4)

**task-3.** ユーザへの柔軟なサービス展開を可能とするパーソナル指向ソフトウェア<sup>(14)</sup>の実現 (req-4, 5より)

次節において、これら課題を検討する。

## 4 P2Pアプリケーションサーバの実現

本検討では、P2P-ASのOS、およびクライアント端末のWebブラウザ、双方の汎用性・拡張性を重視し、Javaによる実現を前提とする。

### 4.1 最適P2Pプラットフォーム

P2P-ASは、Webクライアントユーザを多数収容した状態でP2Pネットワークに参加するので、他ピアが所有するコンテンツのディスカバリ時に非常に高いトラフィックが発生し得る<sup>(13),(17)</sup>。したがって、トラフィック特性を考慮したP2Pプラットフォームの選定が重要である。

P2Pトラフィックモデルは表1に示す3つのタイプがある<sup>(17)</sup>が、P2P-ASはアクセスユーザに対してリアルタイムにサービス提供する必要があり、トラフィックを低減できる「定常コネクション型 (モデル3)」の特性をもつP2Pプラットフォームを適用すべきである。

よって、P2P-AS起動時にはP2P空間内の他P2P-AS、および一般P2Pピア端末に対してコネク

表 1 P2Pトラフィックモデル

モデル		特徴
1	定常データ送信型	ピア間の同期をとるために常にデータを送出し続けている (groove, gnutellaなど)
2	データ送信 +コネクション生成 同期型	検索イベントデータ送信時にあわせてピア間のコネクションを生成 (JXTA Shellなど)
	定常コネクション型	ピア起動時にコネクションを生成し、通常時は検索イベントデータ送信のみ (SIONNetなど)

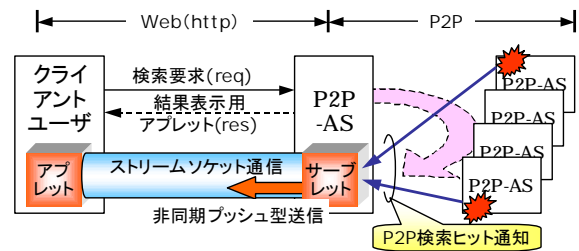


図 3 プッシュ型通知実現方式

ションを生成し、P2P-AS内では多重されたクライアントユーザへの振り分け機能が必要になる。

### 4.2 P2P連携方式

まず、“P2P”と“Web”を連携する上での基本必須機能を明確にする。

通常、P2P空間でのコンテンツ発見後、複数のコンテンツ保有ピアから検索元に検索ヒット (発見) が非同期に通知される。そのため、P2P-ASから検索元Webクライアントユーザへの発見通知はプッシュ型、かつ非同期に通知する機能が必要となる。しかし、クライアントユーザにとって直取P2P-AS以外からの通知を受信できない。そこで本機能実現のため、直取P2P-ASを通知リレーサーバとして利用し、サブレット (P2P-AS側) - アプレット (クライアント端末) 間のストリームソケット通信により実現する (図3)。

次に、ユーザからはバックエンドとなるP2P機能として、P2P-ASに必要な機能を以下で抽出する。

すべてのP2Pアプリケーションサービスフローは、コンテンツの「ディスカバリ→P2Pデリバリ」が基本となることから、P2P-ASを構成する上で、gnutella や、JXTA<sup>(8)</sup>、SIONNet<sup>(9),(10)</sup>など、どのようなP2Pプロトコルを適用する場合でも共通機能となり得、P2P-ASにミドルウェアとして実装すべき機能要素を以下に示す。

**セッション管理:** P2P-AS間を結合し、検索イベントを流通するセッションの管理機能

**イベント送受信制御:** 検索イベントの生成・送信、および受信検索イベントと公開条件の合致時の通知を受信するための機能

**公開情報管理:** P2P空間へのコンテンツの公開情報を整形、公開するための機能

**P2Pデリバリ制御:** P2Pデリバリ用ルートの送受信制御機能 (ただし、P2P-ASではWebサーバという特性上、デリバリに通常のhttpコネクション上のリクエスト/レスポンスでのコンテンツダウンロード機能が適用可能)

以上がP2P連携制御に必要な機能要素であるが、参考のため表2に代表例として、gnutella、

表 2 P2P共通ミドルウェア機能

P2P-AS必要機能	対応する代表的P2Pプラットフォームでの機能		
	gnutella	JXTA	SIONet
セッション管理	•GNUTELLA CONNECT •ping/pong ディスクリプタ	•パイプ	•送受信セッション シェアドリリンク
イベント送受信制御	•query/query hit ディスクリプタ	•Advertisement •Discovery Query /Query Response •メッセージ送受信	•イベント送信 •発火通知 (発火型/ルックイン型)
公開情報管理	•文字列コンペア (実装依存)	•publish	•フィルタ
デリバリ制御	•HTTPコネクション •push ディスクリプタ	•パイプ (TCP/IP,HTTP にバインド)	実装依存

JXTA, SIONetでの上記に相当する機能を示す。

### 4.3 P2P-AS必要機能とソフトウェア構成

P2P-ASは、ユーザにはWebアプリケーションサーバ、バックエンドではP2Pの1ピアとして機能し、さらにP2P空間での公開情報、検索情報を管理するため、以下の機能が必要になる。

- **Webアプリケーションサーバ機能**  
Webサーバ、およびWebアプリケーションサーバ機能
- **P2Pプラットフォーム制御機能**  
コンテンツの検索・発見、およびコンテンツデリバリを実現するためのプロトコル、またはプラットフォーム制御機能（詳細4.2節）
- **ユーザ管理、認証制御**  
収容するユーザの管理、およびアクセス認証制御機能
- **エージェント機能制御**  
Webクライアントユーザ、およびオフラインユーザの代理となってP2P空間とのインタラクションを実現する機能（詳細4.4節）
- **検索情報管理、DB連携制御**  
P2P空間でのコンテンツディスカバリのためのインデックス管理やコンテンツ実体管理のための機能

以上の機能を具備したP2P-ASのソフトウェア構成を図4に示す。図4において、P2Pサービスの追加や変更はWebアプリケーションコンポーネントを追加・変更すればよく、また、適用するP2Pプロトコルを変更する場合には「P2P-PF制御」部にて対応すればよいなど、機能変更や追加を吸収可能なモデルとなっている。

また図4から、本モデルではパーソナライズドサービス提供のためのソフトウェア開発の理解性向上のために、『エージェントがサービスを起動する』というモデルとなっている(14)。すなわち、エージェントは処理・プログラムは内包せず、あくまで個人情報、個人嗜好データのカプセル化とサ

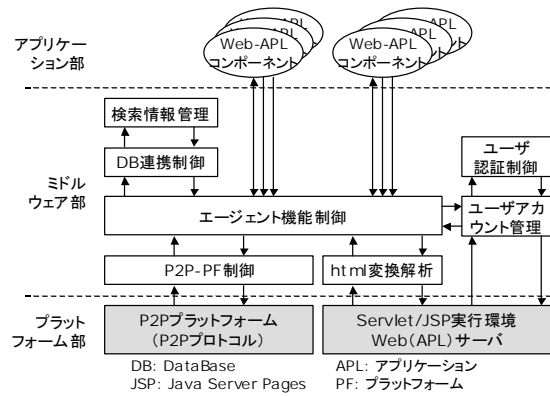


図 4 P2P-ASソフトウェア構成

ービスアプリケーションの起動方法のみを実装したモデルであり、サービスアプリケーションはパーソナライズドエージェント非依存な実行プログラムの実装となっている。

次節にてエージェント適用について述べる。

### 4.4 P2P-ASへのエージェントの適用

ホスティングされたオフラインユーザのコンテンツ操作が必要であること、またパーソナル指向ソフトウェアの実現に適することから、ユーザエージェント(14),(15) (UA: User Agent) を適用する。UAによる仮想ユーザ機能を導入するにあたり、「実ユーザ動作」と「UA動作」の対応付け、および実ユーザがオフライン時のUA動作を明確にする必要がある。

#### 4.4.1 ユーザエージェントの必要機能

UAは、P2P空間へのコンテンツ公開ユーザに対しては公開コンテンツを制御・管理し、一方、検索ユーザにはP2P空間への検索やコンテンツ入手のための制御機能が必要となる。すなわち本稿で提案するP2P-Web連携システムでは、エージェントがピアとなるモデルだと言える。

そこで、もともと既存P2Pはユーザのローカルディスク内に格納されたコンテンツを流通するものであることから、実ユーザに対する“仮想ストレージ (VS: Virtual Storage)”をUAが所有する、というモデルを提案する(図5)。VSの導入により、ユーザのコンテンツをP2P-ASにホスティング(ア

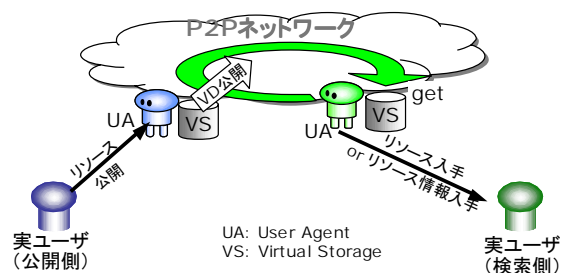


図 5 仮想ストレージ (VS) の適用



アップロード)している場合は、VS内にコンテンツ実体を格納し、一方、ユーザ端末内に実体が存在する場合は、VSにコンテンツインデックス情報を格納することで、既存P2Pと同等の処理モデルにでき、さらにコンテンツの配置、および入手経路(P2Pデリバリルート)をVSにて一元管理でき、プログラム構造上簡略化できるメリットがある。

#### 4.4.2 ユーザ端末プレゼンス管理の必要性

P2P-ASでは、検索/公開ユーザに対応したUAを生成・管理するが、サーバリソースには限りがあることや、プログラムの理解性・記述性向上のため「ユーザ:UA=1:1」とすることから、ユーザがP2Pサービスアクセス時のみ、対応するUAを存在させる必要がある。そこで、ユーザ端末のプレゼンス管理が必要であるが、httpはstatelessプロトコルのため、ユーザ端末のプレゼンスに応じたUAライフサイクルの決定のためには何らかの工夫が必要である。本検討では、Servlet APIを利用した、P2P-ASとユーザ端末間のhttpセッション管理<sup>(16)</sup>を適用し実現する。httpセッションを利用することで、Webアクセスユーザを効率的に管理できることや、onlineユーザ同士のコラボレーションサービスが提供できる、などのメリットがある。

#### 4.4.3 エージェントライフサイクルと最適サービスモデル

ユーザ管理やP2P-ASサーバリソース有効利用の観点から、UAライフサイクルの決定は重要である<sup>(15)</sup>。コンテンツ検索ユーザは、サービス利用時には必ずP2P-ASにアクセスするのでそのライフサイクルは一意に決定でき、4.4.2節に基づいたUAを存在させれば良い。一方、コンテンツ公開側ユーザに対しては図6に示すように

- モデルA: UA不要 (ディスカバリのみ)
- モデルB: UA要 (実ユーザ不要)
- モデルC: UA要 (実ユーザ要)

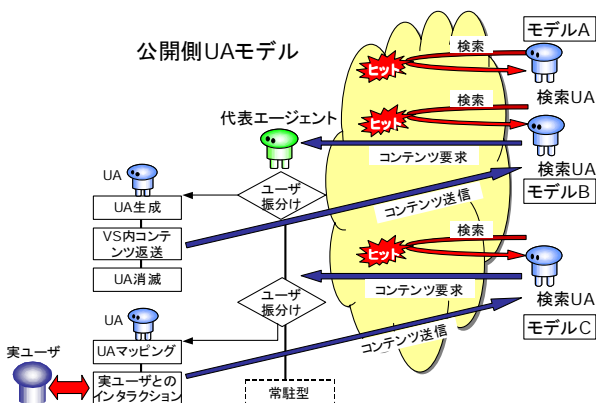


図6 公開ユーザ対応のエージェント動作

表3 UAライフサイクルモデルと最適サービス

UAモデル	ユーザ端末プレゼンス	サービスモデル
モデルA	管理不要	P2Pディスカバリ型 (プロフィール情報、インデックス情報の検索・発見)
モデルB	管理不要	ASバックエンドコンテンツ交換型 (リソース・コンテンツがP2P-AS内にある場合)
モデルC	管理要 ユーザがonlineであることが必要	<ul style="list-style-type: none"> <li>•P2Pサービス公開型(インデックス情報公開時)</li> <li>•ユーザコラボレーション型(IM, チャット, 一部コンテンツ交換など)</li> </ul>

に分類できる。公開側P2P-ASにて、ユーザに対応するUAを生成する場合、4.1節に示したように多重ユーザの振分けが必要なので、常駐型代表エージェントにより適宜実ユーザとの対応付けを行うことで上記サービスが実現可能となる。

以上から、上記のUAモデルとユーザ端末のプレゼンス管理要否、および最適なP2Pサービスモデルの関係を表3に示す。

## 5 P2Pサービス実現例

以上の検討結果を踏まえ、P2P-ASによるP2Pサービス実現方式、本稿では特に“P2Pを用いた検索サービス”および“インスタント・メッセージングサービス”の実現方式を例示する。

### 5.1 P2P検索サービス

まず、あらゆるP2Pサービスの基本となるディスカバリ機能の実現例を図7に示す。本サービスは例えば、ユーザの名刺プロフィール交換やコンテンツインデックス検索時に適用することができる。図7は、P2Pネットワークへの公開インデックス情報だけでなく、DB内に格納された関連属性情報も検索結果通知として返送する場合を示しており、情報公開側エージェントモデルは4.4.3節でのモデルBを適用している。

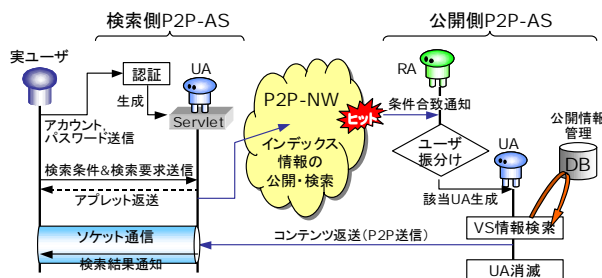


図7 P2P検索サービス実現例

### 5.2 IM (インスタント・メッセージング)

次に、P2Pメッセージコンテンツ交換の代表的アプリケーションであるIMの実現例を図8に示す。本サービス実現時は、検索側・公開側の両ユーザがonlineであることが必須なので、IMメッセージ交換開始前に、5.1節を利用したユーザの検索・発見の後、電子メールなどによりIM開始のためのP2P-ASアクセス勧誘、という動作が必要になる。

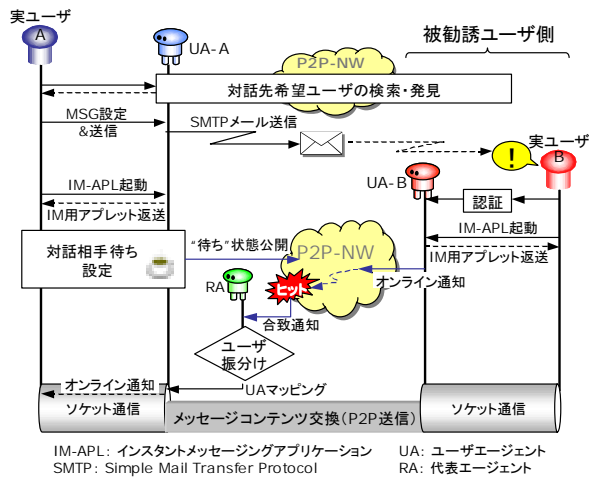


図 8 IMサービス実現例

これは、JavaMail APIを利用したWebアプリケーションコンポーネントの実装で実現可能となる。

図8にて、UA-B参加時にUA-Aと直接通信せず、いったんP2Pネットワーク・代表エージェントを介するのは、UA-A側のP2P-ASにてUA-Bからの受信パケット振分け処理の冗長性を排除し、4.4.3節エージェントモデルCを適用することで、その他のP2Pサービスと同じ処理の枠組みで実現できるというメリットがあるためである。

## 6 関連技術

httpの利用という観点からは、ファイアウォール越えのため、JXTA<sup>(8)</sup>、groove<sup>(5)</sup>、gnutella<sup>(6)</sup>などのP2Pアプリケーションで、デリバリ用のパイプやセッションにhttpを利用しているが、端末上のアプリケーション・GUIなどは独自開発されたものが必要である。

一方、サーバを利用するP2Pモデルとして、Napster<sup>(4)</sup>のようにディスカバリをサーバで行うハイブリッド型P2Pがある。本検討でのP2P-ASは、Napsterでのインデックスサーバ自体がP2P結合されたモデルと考えることができ、ハイスペックなサーバを必要とせず、中央集権的でないP2Pのメリットを生かした高いスケラビリティの確保が可能である。

また、すべてのサービスをWeb上で展開するという観点からはWebサービス<sup>(2),(3)</sup>があり、サービスの発見(UDDI)、サービスの起動(SOAP)というフェーズはP2Pと共通しており、本検討と近い。しかし、本検討では管理の分散というP2Pメリットを生かしており、UDDIサーバのような中央サーバを設置することなく実現できる点で異なる。ただし、P2Pプラットフォーム自体にWebサービス関連技術を適用したモデルも考えられ、今後本研究との融合を検討する余地があると考えられる。

## 7 まとめ

本稿では、すべてのP2PサービスをWebを通して展開し、ユーザ端末にWebブラウザのみを必要とするネットワークモデルへのP2Pアプリケーションサーバ適用を提案し、その必要機能や、サービス提供方式について検討した。

現在、NTTが開発したP2PプラットフォームであるSIONet<sup>(9),(10)</sup>を利用したP2Pアプリケーションサーバの開発を進めており、本検討事項を実装している。今後、実験的に実フィールドでのサービス展開を行い、実装評価、性能評価を行い、報告する予定である。

### 参考文献

- (1) W3C workshop on Web services: Position papers: <http://www.w3.org/2001/03/WSWS-popa/>
- (2) Webservices.org: <http://www.webservices.org>
- (3) IBM developerWorks Web services: <http://www-6.ibm.com/jp/developerworks/webservices/>
- (4) Napster: <http://www.napster.com>
- (5) groove: <http://www.groove.net/>
- (6) gnutella: <http://www.gnutella.com>
- (7) Jnutella: <http://www.jnutella.org>
- (8) Project JXTA: <http://www.jxta.org>
- (9) 星合, 小柳, 他, “意味情報ネットワークアーキテクチャ”, 電子情報通信学会論文誌B, J84-B, No.3, 2001
- (10) 星合, “インターネットの新潮流「ブローカレスモデル」とその実現技術「SIONet: 意味情報ネットワーク」”, 第1回Jnutella P2P Developers Workshop, 2001: <http://www.jnutella.org/jnudev/jws-052201.shtml>
- (11) 大谷, 亀井, 高橋, “P2Pがビジネスを変える”, 翔泳社, 2001
- (12) アール・ホール, “Report1 ピア・コンピューティング・クラウド”, US Insight Silicon Valley Research, Vol.10, July 2001
- (13) 川野, 他, “P2Pネットワークがやってきた”, Software Design 8月号第2特集, 技術評論社, 2001
- (14) 大石, 他, “将来ネットワークソフトウェアにおけるサービス提供技術に関する一検討”, 電子情報通信学会技術研究報告, NS2001-111, Vol.101, No.287, 2001
- (15) 本位田, 飯島, 大須賀, “エージェント技術”, 共立出版, 1999
- (16) 原田, “Javaサーバサイドプログラミング”, 技術評論社, 2001
- (17) 市川, 他, “P2Pアプリケーションのトラヒック評価に関する一考察”, 電子情報通信学会技術研究報告, NS2001-187, Vol.101, No.508, 2001