

# インターネット経路制御パラダイムを適用した ディレクトリシステムアーキテクチャ

江崎 浩 長橋 賢吾

{hiroshi, kenken}@wide.ad.jp

Directory System Architecture based on Internet Routing Control Architecture Paradigm  
Hiroshi ESAKI, Kengo NAGAHASHI

## あらまし

グローバルスケールで自律分散的に動作するディレクトリシステムのアーキテクチャフレームワークの提案を行っている。多様なデジタル通信システムにおけるディレクトリシステムは、(1) グローバルスケールでの動作、(2) アドホック環境での動作、(3) 堅牢性、(4) 自律性、(5) 多様なサービスインスタンスへの対応性を満たす必要がある。特に、アドホック環境への対応は、今後のモバイル/ユビキタス環境の普及とともに、重要な要求条件となる。本論文では、インターネット経路制御アーキテクチャのパラダイムを導入したディレクトリシステムフレームワークの提案を行っている。ディレクトリ構造を階層構造にせず、自律的に定義可能なサービスドメインを定義する(ASに相当)。各サービスドメインは1つ以上のリンク(ピアに相当)で、隣接ドメインと相互接続し、サービスドメイン情報を広告する。これにより、ディレクトリのルートを定義する必要がなく、多様なドメインを自律的に生成削除することができ、さらに、グローバルインターネットに接続されていない環境においても、動作することを可能にする。さらに、動的なクエリパケットの経路制御を実現することによるシステムのロバスト性を向上することができる。

## Abstract

This paper proposes a new directory system framework, which can work as a global distributed autonomous database system. The directory system, that handles various spectrums of communications, must satisfy the following technical requirements; (1) global operation, (2) Ad Hoc operation, (3) robustness, (4) autonomous, and (5) service availability for various service instances. Especially, the adaptability for the ad hoc network is important aspect for upcoming mobile and ubiquitous network environment. In this paper, we propose new directory system architecture based on Internet routing control architecture paradigm. The proposed system defines a service domain, which corresponds to AS in the Internet system, without hierarchical structure. Every service domain connects to the neighbor service domain to advertise the service domain information. By this approach, we must not have any root in the system and can autonomously add and remove the service domain whenever the user wants. As for ad hoc operation, the system can work without the connectivity to the global Internet. Finally, the proposed system provides the robust query packet routing, by means of dynamic routing technique.

**キーワード** ディレクトリサービス、ルーティング、インターネット、経路制御

(注意) 本論文と同様の内容の論文は、電子情報通信学会 和文論文誌B に 投稿されている(2005年2月)。

## 1. はじめに

TCP/IPを用いたインターネットシステムは、基本的には、32ビットあるいは128ビットのIPアドレスを用いてデジタル通信を提供しており、ディレクトリサービスがなくともパケット通信を行うことは可能である。しかしながら、IPアドレスは、2進数の数字の列であり、計算機が記憶するには何の問題もないが、人間が記憶するには都合のよいものとはいえない。したがって、計算機(正確には計算機のインターフェース)に、人間が理解および記憶しやすい文字列からなる名前(FQDN; Fully Qualified Domain Name)をつけ、FQDNの情報からIPアドレスの情報を解決(Resolve/Resolution)するシステムが必要となる。これが、DNS(Domain Name Service)[1]、あるいは、ホスト上に存在する /hostに書かれた情報にあたる。DNSは、グローバルに展開された自律分散型のデータベースシステムであり、IPアドレスとFQDNに関するディレクトリサービスを提供している。DNSシステムは、インターネットにおける基盤的なディレクトリサービスシステムであるが、一般的には、コンピュータシステムは、DNS以外にも、多種多様なディレクトリサービスを必要とする。例えば、ユーザの認証に必要なログインID と パスワードに関するディレクトリサービスは、ユーザのローミングサービスには必須となるディレクトリサービスである。

このような、ディレクトリシステムは、基本的には、DNSやIRR(Internet Routing Registry)[2]、あるいはDHT[3]のようなPeer-to-Peerシステムのディレクトリサービスを除けば、利用するユーザを特定したクローズドシステムとなっている。インターネットシステムは、転送される情報の中味/意味には関係なく、グローバルなインターネット上の任意計算機に、デジタル情報(IPパケット)を転送する いわゆるコモンズ(Commons)[4]の特長を持ったインフラストラクチャであり、これにより、エンド-ツー-エンドアーキテクチャを実現している。IPパケットの転送という観点からみたインターネットシステムと、それを利用して展開されるアプリケーションを支える基盤技術であるディレクトリサービスは、必ずしもエンド-ツー-エンドアーキテクチャに基づいたシステムとはなっておらず、現状では、コモンズの性質を必ずしも持っていない。

DNSはグローバルに展開できたコモンズの特長を持った基盤システムに資する分散ディレクトリサービスであるが、それゆえ、すべてのディレクトリサービスをDNSを用いて実現しようという傾向が見うけられる。一方、個別のディレクトリシステムは、コモンズとしてのグローバル展開を考慮せずにシステムの設計と構築運用が行われる傾向にあり、スケーラビリティへの考慮と多様性への対応が不十分な場合が多い。DNSは、IPアドレスとFQDNとの間のディレクトリサービスという単純で単一の機能を提供することはできるが、それ以外の、ディレクトリサービス機能を実現するには、必ずしも適切なシステムとはいえない。

本論文では、コモンズに資するディレクトリサービスを実現するためのアーキテクチャの提案を行っている。第2章で議論するように、ディレクトリシステムは、以下の6つに関する透過性(Transparency)を実現しなければならない[5]。

1. Location (位置)
2. Duplication (重複性)
3. Fragmentation/Aggregation (分割/集約)
4. Failure (耐故障性)
5. Function (機能)
6. Scale(規模)

本論文で提案するディレクトリシステムでは、インターネットにおける経路制御アーキテクチャのパラダイムを導入することによって、上記の6つの技術的要件を満足する、自律分散性を持ったシステムを実現することを目的としている。提案するシステムは、以下のような

技術的特長を持つ。

- (1) サービスドメインを経路制御におけるAS(Autonomous System)として定義し、自由にディレクトリサービスシステムに参加・離脱可能にする。
- (2) サービスドメインへのクエリメッセージ(問い合わせ)の到達性を提供するために、経路制御におけるBGPのフレームワークを流用する。
- (3) ルートを持たない分散型のディレクトリサービス空間を実現する。今日のコンピュータシステムは、モバイル環境への対応が必須であり、(1)および(3)の技術的特長は、アドホックなネットワーク環境における提案システムの対応を可能とする。

本論文の構成は、以下の通りである。第2章ではディレクトリシステムへの技術的な要求条件を整理する。第3章では既存のディレクトリシステムの問題点を整理する。第4章では提案システムの記述を行う。最後に、第5章でまとめを行う。

## 2. ディレクトリシステムへの要求条件

ディレクトリサービスは、「ある識別子(ID; Identifire)に対して、その識別子の持つプロフィール(Profile)情報自身、あるいは、これにアクセスするための情報を提供するサービス」と定義することができる。例えば、計算機のファイルシステムでは、ディレクトリからそのディレクトリに対応するファイルのディスクリプタ情報を提供する。物理的に分散したファイルシステムにおいては、マウント(mount)を用いて、複数のファイルシステムを結合させることができる。あるいは、DNSシステムにおいては、FQDNの情報から、これに対応するIPアドレスの情報が提供される。

ディレクトリサービスは、IRRやSKYPEのように、1つのサーバを用いて実現することも不可能ではないが、負荷分散と自律的な情報の管理を可能にするために、分散化する必要がある。実際、例えば、IRRシステムをとりあげると、文献[6]に示したように、集中型のデータベースでは、情報の登録やアップデートが不十分になってしまうのが通常である。

データベースの分散化にあたっては、文献[5]に示されているように、以下の6つに関する透過性(Transparency)を満足する必要がある。

### 1. Location (位置)

データの保存場所を自由に選択また変更可能とする。本機能によって、自律的なデータの管理性を実現する。

### 2. Dupilication (重複性)

同じデータを異なる場所に保存することを可能にする。本機能によって、サービスの冗長性を実現させるとともに、負荷分散を実現する。

### 3. Fragmentation/Aggregation (分割/集約)

データの保存場所を複数に分割したり、あるいは、逆に集約可能とする。本機能によって、データ管理の自律性とスケーラビリティが実現される。

### 4. Failure (耐故障性)

ノードおよびノード間の接続性の障害に対して、自律的に対応可能とする。本機能によって、特に、単一点障害(Single Point of Failure)が回避されなければならない。

### 5. Function (機能)

多様な機能への柔軟な対応を可能とする。本機能によって、インターネットシステムにおける エンド-ツー-エンドアーキテクチャによって提供される、サービスの自由度と自律性と同様のサービス種別に関する透過性が実現される。

### 6. Scale (規模)

ノードや情報の増加に対して、十分に対応することを可能とする。本機能によって、ノードの数や情報の量に関して、十分な大規模対応能力という意味での透明性を提供する。

第3章では、上記の6つの透過性という観点から、既存の代表的なディレクトリシステムの分析を行う。

### **3. 既存のディレクトリシステム**

#### **3.1 NFS**

NFS(Network File System)は、UNIXシステムで広く利用されている分散ファイルシステムである。ネットワーク上に分散して存在する複数のディスクを結合させ、仮想的に大きなファイルシステムを構築することができる。マウント(mounting)を用いて、異なるファイルシステムのディレクトリへのポインタを生成することができる。ファイルの格納位置は自由に選択することはできるが、移動の際には、新しいポインタをマニュアルで設定する必要があり、Locationに関する透過性は十分でない。Duplicationの機能はないが、FragmentationとAggregationの機能は備えているといえる。ファイルのディレクトリ構造は、基本的には、階層構造であり、ツリーの中のあるノードに障害が発生すると、そのノード以下のディレクトリがすべて見えなくなってしまうため、Failureに関する透過性はない。自律的にマウントを行うことができるので、十分大きな名記憶空間(64ビットや128ビット)とその識別子が提供されれば、Scaleに関する透明性を持つ。

#### **3.2 DNSシステム [1]**

DNS(Domain Name Service)は、13のルートDNSサーバを頂点とした階層構造の分散データベースである。すなわち、DNSシステムは、マルチルート型の階層的なトポロジーを持つシステムと捕らえることができる。

ドメインの追加削除は、自動で行うことはできず、Delegationの手続きをオペレータが手動(マニュアル)で行う必要がある。その意味で、Locationに関する機能は不十分である。すでに、ルートDNSサーバでも一部導入されているエニキャスト技術を用いて、同一の情報を異なるサーバで提供することは可能であり、Duplication機能は備えているといえる。Delegationされたドメインの分割や集約が可能であるので、FragmentationおよびAggregationに関する機能は備えているといえる。Failureに関しては、DNSシステムでは、以下の2つの観点から、必ずしも十分な透明性を提供しているとはいえない。

- (1) 階層的なディレクトリ構造を持ち、階層の上位から下位へと、順番に再帰的に(Recursive)ディレクトリの検索を行うために、上位のサーバの障害により、それよりも下位のディレクトリサービスを提供するノードへのアクセスができなくなってしまう(接続性は保持されているが、Recursiveな検索のリンクが途切れてしまう)。
- (2) PrimaryとSecondaryという概念を持ち、サーバの2重化を行うことができる。経路制御的に見れば、Static RoutingとDefault Routingを行っているにすぎず、十分な冗長性の確保ができているとはいえない。

Scaleに関する透明性は、上位のサーバ(特にルートDNSとTLDサーバ)への、検索メッセージの集中とその処理能力に依存することになる。実際、Auto-IDのディレクトリサービスであるONSをDNSシステムを用いて実現するという検討が行われているが、上位のサーバへの過負荷

が懸念されている。階層構造における上位サーバへの負荷集中の問題である。

### 3.3 Peer-to-Peerファイルシステム

WinMXやFreenetなど、いわゆる古典的なPeer-to-Peerファイルシステムでは、集中型のディレクトリサーバを持つものと、Peer-to-Peerファイルシステム内で、検索メッセージをフラディング(Flooding)するものの2つに大別することができる。明らかに、前者(集中サーバ型)は、分散データベースとしては定義できない。後者は、分散データベースとして捕らえることができる。

LocationとDuplicationに関しては、自由にファイルを追加削除することが可能であるし、保存場所への制限はまったく存在しないので、十分な透明性を提供しているといえる。Fragmentationに関しても、自由に、保存する内容を分割したり集約化することが可能である。Failureに関しても、Single Point of Failureとはならないシステムとなっている。また、Functionに関する透明度も提供している。

Scaleの透明性に関しては、十分な性能を提供することができるとは考えられない。Flooding手法を用いてサービスノードを検索発見するメカニズムは、基本的には、インターネットにおけるARPに等しい。FloodingはScalability上の問題があることは、よく知られた問題であり、ブリッジにおけるスパンニングツリー(Spanning Tree)の生成を行っても(Peer-to-Peerシステムではこのような手法も導入されていない)、Scalability上十分な性能を提供することはできない。

### 3.4 DHT [3]

DHT[3]は、検索したいオブジェクトから ハッシュ関数を用いて計算される値を用いて、その値に対応するノードへのアクセスを行うというシステムである。ハッシュ値とその値に対応するノードのマッピングを行う必要がある。各ノードは、受信したメッセージ中の宛先のハッシュ値から次ホップのノードを決定するためのテーブルを持つか、あるいはアルゴリズムが決められている。具体的には、例えば、受信したハッシュ値の大小関係で次のホップが決定されたり、あるいは、Staticなテーブルが用意されたりしている。すなわち、基本的には、Static Routingで、クエリメッセージのルーティングを行っているのに等しい。

Location、Duplication Fragmentation / Aggregation、および Functionに関しては、透明性を十分に提供できているといえる。Failureに関しては、基本的にStatic Routingでメッセージのルーティングを行っているために、不十分な透明性となってしまう。Scalabilityに関しては、Static RoutingのScalabilityが問題となってしまう、したがって、十分なScaleに関する透明性を提供しているとは、いいがたい。

### 3.5 Handleシステム [7]

基本的には、DNSシステムと同様のアーキテクチャといえる。Handle Naming Authorityの識別子を定義し、それぞれのAuthorityの中でのサービス実現のアーキテクチャ、データフォーマット、プロトコルなどは、自律的に定義することを可能としたシステムである。Handle Naming Authorityの識別子から、対応するAuthorityへのアクセスのための識別子(= IPアドレスなど)は、Global Handle Registry (GHR)によって提供される。

各Authority内のアーキテクチャは自由に設計することを可能としているので、ここでは議論しない。GHRに関する議論を行う。

Locationの透明性に関しては、各AuthorityはGHRへの登録を行う必要がある。この作業は、

基本的には、手動(マニュアル)で行われる。したがって、十分なLocationに関する透明性が提供されているとは、必ずしもいえない。Duplication、Fragmentation / Aggregationに関する透明性は、十分に提供可能である。Failureに関しては、GHRの障害が全システムの障害につながってしまうので、十分であるとは言いがたい。Scaleに関しても、DNSと同様に、GHRへのアクセスの集中が、課題となる。

### 3.6 まとめ

以上、5つの代表的なディレクトリサービスシステムを、分散データベースに要求される6つの透明性という観点から評価を行った。その結果を、表1に示した。なお、表では、分散データベースが満足すべき6つの透明性に加えて、各システムのアドホックネットワーク環境への対応性の比較評価も行った。アドホックネットワーク環境においては、グローバルインターネットへの接続性がなくても動作することが要求され、かつ、自律的にシステムが動作しなければならない。

表1. 既存ディレクトリシステムの定性的比較  
Table 1. Comparison among legacy directory systems

	NFS	DNS	P2P	HDT	Handle
Location	×	×			×
Duplication	×				
Fragmentation					
Failure	×			×	×
Function	×	×			
Scalability			×	×	
Ad Hoc	×	×		×	×

## 4. 提案システム

### 4.1 提案アーキテクチャ

本章では、グローバルスケールで自律分散的に動作するディレクトリシステムの提案を行う。提案するディレクトリシステムは、以下の5つの技術的な特長を持つ。

- (1) グローバルスケールでの動作(Scalability)
- (2) アドホック環境での動作
- (3) 堅牢性
- (4) 自律性
- (5) 多様なサービスインスタンスへの対応性

特に、アドホック環境への対応は、今後のモバイル/ユビキタス環境の普及とともに、重要な要求条件となる。第3章で評価した既存のディレクトリシステムにおいては、アドホック環境への対応が可能であるのは、Peer-to-Peer型のファイルシステムのみである(表1)。

提案システムは、インターネット経路制御アーキテクチャのパラダイム[8]を導入したディレクトリシステムフレームワークである。ディレクトリ構造を階層構造にせず、自律的なサービスドメインを定義する。サービスドメインは、経路制御システムではASに相当し、

HandleシステムではAuthorityに相当する。各サービスドメインの境界ノード(経路制御におけるBGPルータに相当する)は1つ以上のリンク(経路制御におけるピアに相当する)で、隣接ドメインと相互接続し、サービスドメイン情報を自律的に広告する。サービスドメイン情報とは、サービスドメインの識別子(SDi)、サービスドメインが提供しているサービスの種別(SIDi)、およびサービスドメインディスクリプタ(SDDi)である。SDiおよびSIDiはバイナリの数値である。SDDiは具体的なサービスの内容を示しており、例えば、DNSサービスを本提案システムで実現する場合には、ドメイン名(e.g., wide.ad.jpやu-tokyo.ac.jp)となる。SDDiの表現方法は、本論文では規定しないが、XMLのようなメタ言語での表現が適切であると考えられる。

#### サービスドメイン情報

= [{SDi, SIDi, SDDi}, {SDパス}]

SDパス = {SDi, SD1, SD2, ... SDn}

サービスドメイン情報は、BGPにおける動作と同様に、順次隣接するサービスドメインの境界ノードを経由しながらグローバル空間に伝搬広告される。境界ノードは、BGPにおけるパスベクターの生成と同様の手法を用いて、自サービスドメインから、各サービスドメインへ到達するためのサービスドメインのパスベクター(SDパス)を生成する。生成されたパスベクター情報(SDパス)は、受信した隣接サービスドメイン以外の隣接サービスドメインの境界ノードに広告される。このようにして、すべてのサービスドメインの境界ノードは、すべてのサービスドメインへのアクセスを行うためのサービスドメインのパスベクター(SDパス)のテーブル(BGPにおける経路表に相当する)を生成する。各サービスドメインの中の構成方法は、各サービスドメインで自由に決定することができる。一つのサービスドメインに複数の境界ノードが存在する場合には、BGPにおける iBGP と eBGP と同様のアーキテクチャ/プロトコルを定義することで、柔軟に対応することが可能である。基本的な動作は、BGPと同様であり、BGPが持つ、技術的な特長をすべて持った(Inherit)システムとなる。

すなわち、以下のような特徴を持つ。

- (1) サービスドメインは、その識別子(SDi)にグローバルユニーク性があれば、ディレクトリシステムへの自由な接続(参加)あるいは切断(離脱)が可能となる。すなわち、Locationに関する透明性を実現している。ルートの存在を仮定しないフラットな構造のディレクトリシステムであるので、アドホック環境(あるいはグローバルインターネットに接続されていない環境)での動作も可能となる。アドホック環境においては、サービスドメインを自ノード自身と定義し、すべてのノードが境界ノードとなればよい。
- (2) エニキャストと同様に、同じ情報を持つサービスドメインを複数用意し、これらが、同一のサービスドメイン情報の広告を行うことができる。したがって、Duplicationに関する透明性を実現している。
- (3) サービスドメインは、グローバルユニークなサービスドメイン識別子さえ獲得することができれば、自由に分割あるいは集約を行うことが可能である。すなわち、FragmentationおよびAggregationに関する透明性を実現している。
- (4) ルートを必要としないアーキテクチャであり、各サービスドメインが、複数のサービスドメインと隣接するようなトポロジー構成にすることで、Single Point of Failureとすることを防止することができる。また、BGPと同様に、動的なシグナリング経路(検索メッセージの転送制御)が生成されたために、DNSシステムと比較しても、はるかに高い冗長

性を実現することができる。すなわち、Failureに関して、非常に高い透明性を提供することができる。

- (5) 各サービスドメインは、それぞれ任意のサービスを提供することが可能である。経路制御におけるASの概念では、AS内部でのアーキテクチャは任意とすることができる。それと同様に、サービスドメイン内部でのアーキテクチャは任意とすることができる。すなわち、Functionに関する高い透明性を実現することができる。
- (6) BGPにおける大規模性と同等の大規模性の能力を持つ。BGPのScalabilityは十分に証明されており、同様に、高いScaleに関する透明性を実現する。
- (7) 隣接サービスドメインとの接続は、レイヤ1あるいはレイヤ2である必要はなく、レイヤ3での接続、すなわち、インターネット基盤を利用した隣接ノードとの接続も可能である。したがって、すべてのノードが提案する機能を実装運用する必要はなく、インターネット上に、オーバーレイした形での導入が可能である。

## 4.2 アプリケーション

本節では、提案アーキテクチャを適用したディレクトリサービスの一例として、グローバルなローミングサービスに必要な、ユーザ認証情報のディレクトリサービスを取り上げる。

現在、サービスプロバイダ間でのローミングのためのユーザ認証の共有は、基本的には、各サービスプロバイダ間に1対1のプライベートピアを作り、ユーザ認証に関する情報を共有/交換している。すなわち、ローミングサービスを提供するプロバイダ間に、フルメッシュのプライベートピアを確立した、プライベートな広域インターネットイクスチェンジ (IX; Internet eXchange) と同等の構成となっていると捕らえることができる。例えば、これは、日本テレコム社が提供しているMPLSを用いた広域分散IXサービスである ASSOCI0[9]上でのマルチプロバイダによるBGPを用いたVPNネットワークとほぼ同等である。このような既存のローミングシステムは、グローバルな環境への対応は、ほぼ不可能であり、本論文で示した分散データベースが満足すべき6つの技術的な透過性を実現することはできない。

本論文で提案したアーキテクチャをローミングのためのユーザ認証情報の提供サービスに適用することを考える。各プロバイダは、自身がサービスドメインの識別子(SDi)を持ち、サービスの種別(SIDi)として「ユーザ認証情報」、サービスドメインディスクリプタ(SDDi)として「ドメイン名」(あるいは「サービスプロバイダ名」や「AS番号」)を定義し、これを、隣接するサービスドメインの境界ノードに広告すればよい。ドメイン間の接続は、レイヤ3での接続であり、インターネット基盤を利用した、冗長性を持つパスを利用することが可能である。このようなアーキテクチャにすることで、プロバイダ間でのローミング情報の共有や交換の自由度が大きく向上するとともに、各プロバイダのローミングサービスへの参加や離脱に関する透明性(Location)が実現される。

## 5. まとめ

以上、本論文では、グローバルスケールで自律分散的に動作する新しいディレクトリシステムのアーキテクチャフレームワーク提案を行った。提案したディレクトリシステムは、インターネット経路制御アーキテクチャのパラダイムを導入することにより、(1) グローバルスケールでの動作、(2) アドホック環境での動作、(3) 堅牢性、(4) 自律性、(5) 多様なサービスインスタンスへの対応性を満足することができる。ディレクトリ構造を階層構造にせず、自律的なサービスドメインを定義 (ASに相当) し、各サービスドメインは1つ以上のリンク



(ピアに相当)で、隣接ドメインと相互接続し、サービスドメイン情報を広告する。これにより、ディレクトリのルートを定義する必要がなく、多様なドメインを自律的に生成削除することができ、さらに、グローバルインターネットに接続されていない環境においても、動作することができるとともに、動的なクエリパケットの経路制御を実現することができる。

今後は、本アーキテクチャフレームワークに沿った、具体的なプロトコルの設計と実装を行う予定である。

### **参考文献**

- [1] P.Mockapetris, "DOMAIN NAMES - CONCEPTS AND FACILITIES", IETF RFC1034, November 1987.
- [2] Internet Routing Registry Web, <http://www.ietf.org/>
- [3] The Chord Project, <http://www.pdos.lcs.mit.edu/chord/>
- [4] Lawrence Lessig, “コモンズ”, 翔泳社、ISBN4-7981-0204-0、2003年1月.
- [5] J.B.Rothnie and N.Goodman, "A Survey of Research and Development in Distributed Database Management", VLDB 77, pp.48-62, 1977年.
- [6] 長橋、江崎, "インターネットルーチングレジストリにおける経路整合度の評価に関する研究", 電子情報通信学会 和文論文誌D-1, Vol.J87-D1, No.5, p.553、2004年5月
- [7] S.Sun, L.Lannom, B.Boesch, "Handle System Overview", IETF RFC3650, November 2003.
- [8] Y.Rekhter, T.Li, "A Border Gateway Protocol 4 (BGP-4)", IETF RFC 1771, March 1995.
- [9] MPLS ASSOCIO, <http://www.associo.jp/>