

異機種ルータの相互接続試験用測定ツールの開発

菊池 豊† 石原 丈士†† 菱岡 裕男††† 楠田 友彦†††† 永見 健一†††††

中川 郁夫††††† 江崎 浩††††††

† 高知工科大学 〒 782-0003 高知県香美市土佐山田町宮ノ口 185 高知工科大学連携研究センター 306B

†† 株式会社東芝 研究開発センター 〒 212-8582 神奈川県川崎市幸区小向東芝町 1

†††† 株式会社インテック・ネットコア 〒 136-0075 東京都江東区新砂 1-3-3

†††† インテック・ウェブ・アンド・ゲノム・インフォマティクス株式会社 〒 136-0075 東京都江東区新砂 1-3-3

††† 株式会社ネクステック 〒 060-0807 北海道札幌市北区北 7 条西 5 丁目 8-1 北 7 条ヨシヤビル 8 階

††††† 東京大学 〒 113-8656 東京都 文京区 本郷 7-3-1 東京大学 大学院 情報理工学系研究科

E-mail: †kikuchi.yutaka@kochi-tech.ac.jp

あらまし 広域分散かつ L2 メディア非依存な IX (Internet eXchange) を実現するためのアーキテクチャとして MPLS-IX がある。一般に MPLS-IX は複数の異機種ルータで構成されることより、我々はルータベンダ等と共同で相互接続試験を行ってきた。限られた時間と資源の制約の下で試験を円滑に遂行するため、相互接続試験向けの専用の測定ツールを開発して利用し、効果をあげた。本稿では、測定ツールに対する要求と実現方法および効果について報告する。

キーワード 広域分散 IX、MPLS、次世代 IX 研究会、相互接続試験、ネットワーク測定ツール

Development of Testing Equipment for the Interoperability Test of Heterogeneous Routers

Yutaka KIKUCHI†, Takeshi ISHIHARA††, Hiroo HISHIOKA†††, Tomohiko KUSUDA††††,

Ken'ichi NAGAMI†††††, Ikuo NAKAGAWA†††††, and Hiroshi ESAKI††††††

† Research Institute, Kochi University Of Technology

306B Research Collaboration Center, 185 Miyanokuchi, Tosayamada-cho, Kami-city, Kochi 782-0003 Japan

†† TOSHIBA Corporation, 1 Komukai Toshiba-cho, Saiwai-ku, Kawasaki, 212-8582 Japan

†††† Intec NetCore Inc., 1-3-3 Shinsuna, Koto-ku, Tokyo 136-0075 Japan

†††† INTEC Web and Genome Informatics Corporation, 1-3-3 Shinsuna, Koto-ku, Tokyo 136-0075 Japan

††† Nextech Co., Ltd., 8F Yoshiya bldg., 8-1 N7W5, Kita-ku, Sapporo-city 060-0807 Japan

††††† University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

E-mail: †kikuchi.yutaka@kochi-tech.ac.jp

Abstract MPLS-IX architecture makes IXes widely distributed and Layer 2 media free. There were interoperability problems on routing equipments to deploy the architecture to the Internet, so router manufacturers had held interoperability testing to establish the architecture. We had developed a network testing tool, which had made interoperability tests smoothly. This paper reports requirements, implementation and effects of the tester.

Key words widely distributed IX, MPLS, distix, Interoperability testing, network measurement tool

1. はじめに

MPLS-IX は、MPLS (Multi-Protocol Label Switching) の仮想パスである LSP (Label Switched Path) 上で IP パケットを交換する IX (Internet eXchange) 技術である [1]。これは、従来の IX 技術では困難であった広域に分散可能かつデータリンクメディア非依存な IX を可能とするアーキテクチャである。

MPLS は、ラベル付きパケットを処理する core LSR (Label Switching Router) と、IP パケットとラベル付きパケットとを交換する edge LSR とからなる。MPLS-IX では、MPLS の edge LSR は IX 機能を利用する ISP に、core LSR は IX 機能を提供する IXP に設置する。各 ISP の edge LSR 間で LSP が構成され、IP トラフィックはこの LSP 上を流れる (図 1)。

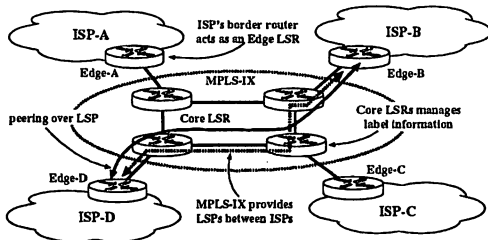


図 1 MPLS-IX の物理アーキテクチャ
Fig.1 MPLS-IX physical architecture

我々は、MPLS-IX アーキテクチャの実用性を高め、参加組織の理解を深め、インターネットでの導入を促進することを目的とし、非営利の任意団体である次世代 IX 研究会^(注1)を 2001 年に組織した [2] [3]。

本来 MPLS は同一 AS 内、すなわち経路制御ポリシーが統一されている組織内での利用が前提の技術である。このため、単一ベンダーのルータのみが用いられ、かつ単一の経路制御プロトコルメイン上で用いられるような想定に基づいてルータの実装が行われている可能性が高い。

一方、MPLS-IX での利用形態はこれとは異なっており、通常の利用では出現しない問題が発生する可能性が強かった。さらに、MPLS-IX の提案当時は MPLS 技術自体が比較的新しい規格であり、ルータ製品での MPLS の実装は必ずしも信頼性の高いものではなかった。よって、MPLS 自体の相互接続性が低い可能性を疑う必要があった。

そこで、研究会内でルータ相互接続ワーキンググループを構成し、2001 年 10 月に開催した第 1 回より 2004 年 11 月に開催した第 10 回まで、MPLS-IX をテーマとした相互接続試験を行った [4] [5] [6] [7]。

相互接続試験は、多数のベンダが参加する中で時間の制約を受けながら試験メニューをこなしていく必要がある。試験では実機で構築した MPLS-IX に対して、測定ツール (以下「ツール」と略す) で機能を確認する。これは既存のツールでは不

分な点が多い。我々は専用のツールを開発して試験で利用することで、試験を円滑に行うことができた。本稿では、MPLS-IX の相互接続試験向けに開発したツールについて述べる。

なお、相互接続試験に参加するのは必ずしもルータメーカーだけでなく、オープンソースの実装を行う大学のチーム、またルータの販社などもある。本稿ではこれらをすべて「ベンダ」と総称することとする。

2. 相互接続試験の手法

本節では次世代 IX 研究会で行った相互接続試験の手法について述べる。

ルータの相互接続試験とは、アーキテクチャの異なるルータを相互に接続し、ルータおよびネットワークが正しく機能するかを調査するものである。次世代 IX 研究会ではルータベンダに呼びかけて、MPLS-IX アーキテクチャの相互接続試験を行った。

研究会や相互接続試験への参加費は無料である。参加するベンダは、試験用の機材、試験に参加する技術者、その技術者の旅費等の負担を行うことで試験に参加できる。個別の試験結果の詳細は WWW で公開^(注2)、誰でも閲覧できるようにしてある。

2.1 試験プロセス

1 回の相互接続試験は、準備開始から完了までで 3ヶ月程度を要する。試験期間は、とある 1 週間の平日 5 日間にとることが多い。おおよそ 1 つの試験項目に対して半日から 1 日かかり、1 週間で 4 ないし 5 程度の試験メニューを消化する。以下に、試験の過程を示す。

- 試験概要
- 日程の決定
- 試験会場の選定
- 試験詳細の決定
- 試験の実施
- 結果の整理と参加者への通知
- 参加者の同意と結果の公開

図 2 は第 1 回相互接続試験の風景である。

MPLS-IX の試験の多くは、持ち寄った実機のルータで MPLS-IX を構成して、期待する機能が実現されているかを確認するものである。

たとえば Edge LSR 試験と呼ばれる試験メニューは、主催者側で用意した Core LSR に各ベンダが Edge LSR を接続することで MPLS-IX を構成し、MPLS-IX としての正しい機能が実現できていることを確認するものである。

図 3 は第 1 回の Edge LSR 試験のトポロジーの例である。左右の側にある 9 つずつのネットワークが MPLS-IX により相互接続される AS であり、試験ではこれらの AS 間での疎通試験を行う。

(注 1) : <http://www.distix.net/>

(注 2) : <http://www.distix.net/router-wg/>

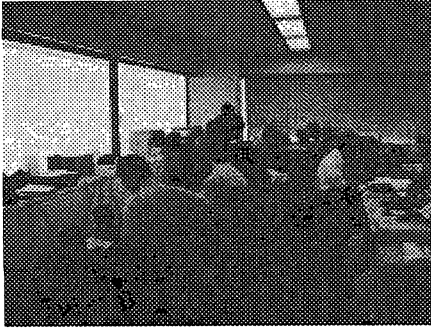


図 2 第 1 回試験風景

Fig. 2 A view of first testing event

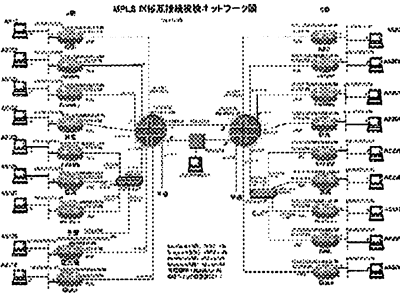


図 3 試験トポロジーの例

Fig. 3 An example of testing topology

3. ツールに対する要求

ここでは試験用に開発したツールに対する要求について述べる。

通常、ネットワークの接続性を確認する際には、ICMP Echo Request/Reply を用いた ping プログラムを手動で利用することが多い。あるいは、ルータテストと呼ばれる製品も出ている。しかし、MPLS-IX の相互接続試験においては以下の点が必要であり、手動による ping やルータテストでは不十分であった。

- 一方向の IP パケットを用いる
- ある時点のスナップショットがとれ、履歴を残せる
- WWW で参加者全員が結果を見ることが出来る
- 拡張性がある
- 低コストである

これらを満足させるためにツールを作成するに至った。以下では、まず必要性和要求について述べ、続いて仕様と実装について述べる。

3.1 片方向 IP パケットによる測定の必要性

一つの大きな理由は技術的な問題である。OS に実装されている通常の ping コマンドは ICMP パケットで往復の通信経路を検査する。ところが、MPLS-IX の相互接続試験で検査したいのは、Edge LSR 間に張られているはずの LSP の状態である。LSP の上を流れるのは IP であり ICMP でない上に、MPLS における LSP は片方向接続の仮想パスであるため、ping の結

果では間接的な情報しか得られない。よって IP パケットを用いた上で、さらに片方向のみのトラフィックを測定する必要があった。

3.2 スナップショットの必要性

技術上のもうひとつの大きな理由は、LSP の本数の多さである。

信頼性の高い結果を得るためには、ネットワークのある瞬間の状況をスナップショットとして見られることが望ましい。ところが、手動による ping コマンド実行による結果を収集していると、ある瞬間のスナップショットを取得することが難しい。

Edge LSR として n 種類のルータが試験に参加している場合を考えてみる。組み合わせ自体が n^2 あり、往復の経路は異なる LSP になる上、さらにパラメータを m 通りに変化させることを想定すると、全部で $2mn^2$ 種類の疎通試験を同時に行う必要があるとわかる。図 3 で長短 2 種類のパケット長での疎通試験を行う場合は、324 種類を同時に検査する必要がある。これを手動で短時間に全部の行うのは困難である。このため、一斉検査を実施できるツールの必要があった。

3.3 相互接続試験遂行上の必要性

もう一つの理由は、運用上の問題である。実験期間と試験項目が決まっている一方で、スタッフには実験をスケジュール通りに進行させたいという要求がある。一部に動作不良箇所があると、不良箇所をかかえる参加者がデータ収集の延期を申し出ることが多く、計測状態が収束しにくい。これはスケジュール通りに試験が進まない大きな要因となる。

ただし、一方的に時間を区切ることで結果とするのは望ましくなく、状況がある程度安定した時点での結果を最終結果としていたと考えていた。これは、設定のケアレスミスが残っていたり、実装のミスを発見してデバッグ中に試験が終了してしまうようなことは避けたいからである。単に終了時刻を定めているだけでは、デバッグ中の過渡的な状態を試験結果として残すことになる可能性がある。

このため、主催者スタッフの手元だけで結果が分かるのではなく、スタッフがどのように結果を収集するのかがきちんと説明でき、結果の収束や測定結果が参加者にはっきり見えるようにする必要があった。

3.4 拡張性に対する要求

基本的に検査したい内容は疎通しているか否かである。しかし、これだけでなくより詳細な情報が必要になる可能性があった。実際、後で述べる TTL やフレーム長の検査が必要になった。いかなる追加が必要になるか事前には想定しきれないこともあり、ツールには出来るだけ柔軟に機能を追加できる必要性があった。

3.5 コストに対する要求

次世代 IX 研究会および相互接続試験の参加は無料である。Edge LSR 試験の場合の Core LSR や、参加者ルータに接続する L2-SW など、必要な機材は主催者側では十分に準備できず、協力組織より借り受けていた。このような環境において、測定に用いる機材も低コストで入手できる必要があった。

4. ツールの実現

MPLS-IX 上のすべての LSP に対して IP パケットの到達性のあるなしを WWW 画面上で一度に表示することができ、かつ時系列で履歴を追いかけられるようにした。このため、定期的に（通常は 3~5 分間隔）スキャンプログラムを起動し、結果を WWW 上に表示するものとした。

相互接続試験では、参加メンバーがルータの設定を行うのとルータの状況を監視するために、1 つのサブネットをメンテナンスセグメントとして準備していた。これ経由で参加メンバー全員が測定結果の WWW を見るができるようにした。

4.1 仕様

スキャンプログラムは全部の LSP の疎通を調べるもので、Sender と Receiver により構成する。

- (1) Sender はトラフィックとしてのパケットを生成する。
- (2) Receiver は受信したパケットの状態を記録する。タイムアウトした場合には疎通がなかったものとする。
- (3) 送出トラフィックは UDP とする。IP TTL およびパケット長はスキャンプログラムの起動ごとに設定可能とする。
- (4) 送出パケット内に時刻情報を埋め込み、Receiver でその時刻情報を抽出して記録する。
- (5) 結果をファイルに書き出す。

パケットごとの疎通試験に対して 1 行 1 エントリで結果をファイルに書き出した。各エントリの内容は次の通りとした。

```
vlanid, srcip, dstip, ipid, ipttl, rcvsiz
vlanid: vlan タグ ID
srcip: 送信元 IP アドレス
dstip: 受信先 IP アドレス
ipid: IP ヘッダの ID フィールド値
ipttl: IP ヘッダの TTL 値
rcvsiz: 受信したフレーム長
```

ここで TTL とフレーム長を計測することは MPLS および MPLS-IX に特有の事項であるので、理由を以下に詳しく述べる。

4.1.1 TTL

何らかの原因で通信経路が循環してしまった場合に、限りなくパケットの伝送が続くことがないように、IP にも MPLS にも TTL (Time to Live) の概念がある。これは経由できるリンクの数を示した値で、送出ルータで正の整数を TTL フィールドに入れ、ルータを経由するごとに 1 減らし、0 になった時点でパケットを廃棄するようになっている。

IP 網の途中で MPLS 網がある場合に、それを連続する網とみなすか、MPLS 網を 1 本の擬似的なリンクとみなすかにより、TTL の算出方法が異なる。これは現在では RFC3443 [8] に定義されている。しかしながら、相互接続試験を開始した当時は RFC に規定が無く、松嶋らの Internet Draft が存在するのみであった^(注3)。通常のルータは前者を実装するものの、後

(注3)：当時は draft-satoru-mpls-1hop-lsp-00.txt として公開されていた。現

者を実装するかどうかはベンダ依存であった。

一方で、経路制御に用いる BGP は、LAN 型や ATM 型などの従来型 IX においては隣接ルータがセッションを張る。このため L2 リンクの障害時には BGP のセッションも切断されるので、通信経路を失ったことを BGP 境界ルータが知ることができる。

MPLS-IX では、離れたルータ同士で BGP セッションを張る BGP multi-hop 手法を用いる。これを用いると、LSP 途中で障害が発生して通信経路が失われた際に、インターネット上の別の通信経路を通じて経路情報が伝わる可能性を残す。この場合には、BGP ルータが MPLS-IX 上の通信経路が消失したことを検出できない。

このため MPLS-IX では、LSP 通過における TTL 処理を「1 だけ減少」とし、かつ BGP のセッションに用いる TTL を 1 とすることで、BGP セッションが LSP 以外の通信経路を通ることを避けるようにしなくてはならない。

この「TTL を 1 だけ減少」を正しく処理しているかを確認するために、ツールで TTL の変化を見られるようにすることにした。

4.1.2 フレーム長

IP の MTU (Maximum Transfer Unit) は Ethernet のフレーム長の仕様で依存して 1500 Octet であることを前提とする場合が多い^(注4)。MPLS を用いる場合には、LSP 上では 4 Octet のラベルが付加されるので、Ethernet のフレーム長も 4 Octet 分長くなる。

アプリケーションによっては IP の DF (Don't Fragment) ビットを on にして、MTU が小さくなる場合のフラグメント処理を禁止するものがある。また、禁止しないまでもフラグメントが起こると、フラグメント処理とアセンブリ処理のオーバーヘッドによるルータ処理性能の低下も予想できる。

ルータのインタフェースが長いフレームを正しく処理することが期待され、これを検査するために、ツールでフレーム長を見られるようにすることとした。

4.2 実装

NetBSD 上に BPF および libnet ライブラリ^(注5)と hping2^(注6)とを用いて検証ツールを実装した。

hping2 は ICMP、UDP、TCP などの任意のパケットを送受信する到達性検査ツールであり、今回の我々の検証ツールに対する要求の大部分を満たしていた。しかし、送信についてはカーネルが持つ IP 層に依存していること、および受信時の VLAN タグ処理をサポートしていないことから、そのままの形で利用することはできなかった。

このため、我々は BPF (Berkeley Packet Filter) デバイスを用いて IP パケットを直接ネットワークに送信できるように変

在は期限切れであり IETF のホームページでは参照できない。

(注4)：本来は Path MTU Discovery が正しく機能する場合には、このような前提は不要なはずであるものの、現実にはこうなっていない場合がほとんどである。

(注5)：<http://www.packetfactory.net/projects/libnet/>

(注6)：<http://www.hping2.org/>

更し、受信時に VLAN タグの存在および値のチェック機能を付加した。BPF 経由でのフレーム送信は libnet ライブラリの関数を利用した。パケットの生成、伝送時間の計算などの処理は hping2 のオリジナルのルーチンをそのまま使用している。

UNIX コマンドの形式としては、従来の hping2 に対して、VLAN ID を指定するコマンドオプションが追加された形となった。変更は C のソースコード換算で 500 行弱である。

Sender はあらかじめ決められた送信元 IP アドレスと受信先 IP アドレスに対して、定期的にパケットを送信する。これは前述のプログラムに与えるコマンドライン引数をシェルスクリプトで生成し、プログラムを cron で定期的に起動することで実現した。

Sender から送信される各パケットのデータ領域には IP アドレスではない送信元と受信先を表す識別子が埋められ、他のパケットとテストパケットとを区別する。また Receiver は、ネットワーク上のパケットを BPF を使って取り込み、パケット内の識別子、IP アドレス、VLAN ID などをチェックして、テストパケットを受信したと判断した場合に、IP アドレス、TTL および受信サイズなどをファイルに出力する。

また、Receiver の結果を整形し随時結果が HTTP で確認できるように、スクリプトを記述し Web サーバを動作させた。

4.3 利用方法

図 4 はツールの外観である。ハードディスクのない、手のひらサイズの PC 互換機を用いている。これは Ethernet インタフェースを 2 つ装備しており、一方を参加者が WWW を閲覧するためのメンテナンスネットワークに接続し、他方を試験に用いることとした。

図 5 は測定時のトポロジーを示している。ここで sender と receiver とは物理的には同一のツールの I/F であり、Tagged VLAN を構成することで限られた物理 I/F で複数 LSP の同時検査を実現している。

図 6 は結果を示す WWW 上の表示例である。A~J は実際の試験活動では分かりやすいようにベンダ名にしてある。表で、○が通過、×が不通を示している。さらにパケットが通過していても、TTL 等のパラメータが不正な場合も表示できるようにしており、実際の WWW 画面では、青い○がパラメータが正常な通過状況を、黄色の○がパラメータが不正な通過状況を、赤い×が不通を示している。

5. 考 察

ツールが無かった第 1 回~第 3 回の相互接続試験に対して、ツールを使うようになった第 4 回目以降では運用の効率が向上した。この試験専用のツールを作成して活用したのは有効であったと考える。本節ではツールの開発がどのような局面で有効であったかを考察する。

5.1 不具合の発見と修正の結果の確認の迅速化

不具合は実装の不備の場合の他に、接続ミスや設定ミスもある。接続性がない場合のトラブルシューティングは容易ではない。一部の設定を変えることで、それまで成立していた別のルータとの接続性を失ってしまうこともある。

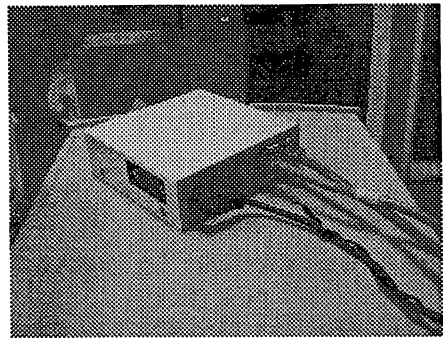


図 4 ツールの外観

Fig. 4 A view of the tool

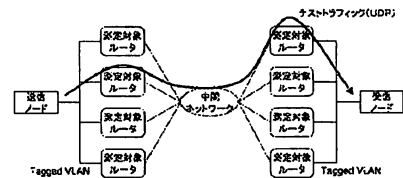


図 5 測定時のネットワークトポロジー

Fig. 5 An example of topology using the tool

	A	B	C	D	E	F	G	H
A	○	○	○	○	○	○	○	○
B	○	○	○	○	○	○	○	○
C	○	○	○	○	x	○	○	○
D	○	○	x	○	○	○	○	○
E	○	○	○	x	○	○	○	○
F	○	○	○	○	○	○	○	○
G	○	○	○	○	○	○	○	○
H	x	○	○	○	○	○	○	○

図 6 測定結果表示

Fig. 6 A result example page of the tool

今回のツールの表示では、接続性が確保されていない箇所が特定ベンダに集中しているのか、あるいは散発的に出ているのかを容易に読み取ることができる。これにより特定ベンダのルータに問題があるのか、特定ベンダ間の相性の悪さといった問題なのかを判定しやすくなった。また、一部の修正が全体での接続性にどのように波及していくかも予想しやすくなった。

5.2 実験の進行の管理

スタッフは全体の進捗を見ながら、必要があれば個別のベンダと相談し、実験全体を進めていく。ツールの WWW による表示は、全体で状況を共有することができるため、スタッフが進捗を把握するにも便利であり、進行管理に役に立った。

たとえば、ツールの表示自体や、その変化や推移を眺めることで、どのベンダに問題が発生しているのか、あるいは大きな作業をしているのかわかる。また、接続不能な組み合わせのうち、どの組み合わせが接続までたどり着けそうで、どれが絶望的なのかもおよそ把握することができた。

ツールの表示の推移を見ながら結果が漸近して来たところを見計らって、次の正時等の切りの良い時刻を示し、その時刻のツールの結果を持って試験の結果とすると宣言することにした。これにより、エンジニアは終了時刻のしばらく前より設定・ファームウェア等に触らないようになり、安定した状態でのスナップショットを試験結果と出来るようになった。

加えて、必死でデバッグするベンダのエンジニアに対して試験時間の終了を告げるのことで、スタッフにとって大きな精神的負担がかかっていたのが軽減されるようになった。

5.3 試験精度の確保

ツールが無かった第1回～第3回の試験では、各ベンダからの報告を元に相互接続性の一覧表を構成した。しかし、各ベンダからの報告をまとめるだけでは「ある瞬間にうまく行ったことがある」という状況の集合が最終的な結果になってしまう。

これは望ましい試験方式ではない。たとえば、ルータAのベンダが、まずルータBと接続できて、その後ルータCとも接続できたとする。この場合ルータAのベンダの報告をまとめると、どちらとも接続できたという結果となる。しかし、ルータBとルータCと同時に接続出来ていたのか、あるいはそのような実装や設定が可能なのかは実際には不明である。

このため、報告から構成した一覧表が必ずしも全体を正しく反映するものにならない。ツールの開発により、同時にすべての通信経路に対して、プローブパケットを投げることができるようになり、正確な試験結果を残せるようになった。

6. まとめ

次世代IX研究会の相互接続試験において、専用のツールを開発して利用することで、試験をスムーズにすることが可能となった。

次世代IX研究会の相互接続試験は、MPLS-IXの実用化を促すのに大きな貢献をした。MPLS-IXはJGN II上に構築した実験網の他、日本テレコムによりmpls ASSOCIOとして商用化されるに至っている^(注7)。

謝 辞

相互接続試験に参加して頂いた各組織に御礼申し上げます。また、事務局を御願っている株式会社三菱総合研究所の諸氏には、常日頃より次世代IX研究会の活動の支援で大変お世話になっている。本活動はTAO I121-401による支援を受けた。

文 献

- [1] I. Nakagawa, H. Esaki, Y. Kikuchi, and K. Nagami, "Design of next generation IX using MPLS technology," IPSJ Journal, vol.43, no.11, pp.3280-3290, Nov. 2002.
- [2] 中川郁夫, 江崎浩, 菊池豊, 永見健一, "MPLSを用いた広域分散IXの実証実験," 情報処理学会誌, vol.43, no.11, pp.1192-1198, Nov. 2002.
- [3] 中川郁夫, 江崎浩, 菊池豊, 永見健一, "MPLSを用いた広域分散IXの実験," 情報処理学会論文誌, vol.43, no.11, pp.3519-3529, Nov. 2002.
- [4] 菊池豊他, "異機種ルータの相互接続試験活動～新しいネットワークアーキテクチャの導入を促進するために～," 信学技法, vol.106, no.15, pp.19-25, Apr. 2006.
- [5] 菊池豊, "次世代IX研究会(distix)における相互接続試験," MPLS Japan 2002 発表資料, Nov. 2002.
- [6] 村井朋生, "実装者から見た相互接続," MPLS Japan 2002 発表資料, Nov. 2002.
- [7] 村上哲也, "相互接続実験での実装の相違による様々な出来事2," MPLS Japan 2003 発表資料, Oct. 2003.
- [8] P. Agarwal, and B. Akyol, "Time to live (TTL) processing in multi-protocol label switching (MPLS) networks," , Jan. 2003, IETF RFC3443, proposed standard.

(注7) : <http://www.associo.jp/>