

# MPLS マルチキャスト転送機構の設計と実装

小柏 伸夫†      宇夫 陽次郎‡      宇多 仁†      篠田 陽一††

† 北陸先端科学技術大学院大学 情報科学研究科  
‡ 株式会社インターネットイニシアティブ 技術研究所  
†† 北陸先端科学技術大学院大学 情報科学センター

現在のインターネットにおける多点間通信 (マルチキャスト) は IP マルチキャスト [1] 技術およびその派生技術が用いられている。IP マルチキャストを実現するためには、パケットを多点間に配送するための技術と同時に、多点間でのパケットの配送経路を規定する配送木の確立などの配送制御技術が必要である。そのため、現時点では主にユニキャスト配送に関する仕様化がなされているマルチプロトコルラベルスイッチング (MPLS) を IP マルチキャストで利用するためには、多くの技術的課題を解決する必要がある。我々は MPLS におけるマルチキャストの実現を目指して、我々が設計開発している MPLS 実装 AYAME 上に MPLS でのマルチキャストパケット配送機構を実装した。本実装は IP マルチキャストを包含した汎用的な多点間通信機構を MPLS 上に実現するための試みである。本論文では、MPLS におけるマルチキャスト配送のフレームワークをまとめたうえで、MPLS パケット配送機構の設計と実装を論じる。

## The Design and Implementation of An MPLS Multicast Forwarding Mechanism

Nobuo Ogashiwa†      Yojiro Uo‡      Satoshi Uda†      Yoichi Shinoda††

† School of Information Science, Japan Advanced Institute of Science and Technology  
‡ Research Laboratory, Internet Initiative Japan Inc.  
†† Center for Information Science, Japan Advanced Institute of Science and Technology

Currently, IP multicast [1] and its derivative are used as multicast communication technologies on the Internet. To realize IP multicast, not only a packet distribution function but also a distribution tree establishment function are required. Therefore, to apply IP multicast technologies to the MPLS which aims to an only unicast packet forwarding function, several problems have to be solved. We implemented a multicast packet distribution function with the MPLS packet forwarding function. This is an experimental implementation for realizing the generic MPLS multicast system which include the IP multicasting in context of the MPLS. In this paper, we offer the framework of the multicast communication in context of the MPLS, and we also discuss the design and implementation of the MPLS multicast system we implemented.

### 1 はじめに

インターネットの爆発的な普及に伴い、ルータにおけるパケット転送処理の高速化や TE (Traffic Engineering) および QoS (Quality of Service) の実現の可能性を持つ技術として MPLS (MultiProtocol Label Switching) 技術に対する期待が高まりつつあ

る。一方で、ネットワークアプリケーションの多様化に伴いマルチキャスト技術も注目されつつあり、これまでに IP マルチキャスト技術として IP を基本とした種々のマルチキャスト技術が提案されてきた。IP マルチキャスト技術はパケット配送だけでなく経路制御の段階においても多くの要素技術を必要とする。そのため、暫定的にユニキャスト

だけを対象として標準化が進められてきた MPLS 上で IP マルチキャストを利用するには、様々な問題を解決しなければならない。我々は、MPLS におけるマルチキャスト配送に関する問題は極めて数多く存在するため段階的に解決していく必要があると考え、MPLS における基本的なマルチキャストパケット転送機能に主眼を置いて MPLS マルチキャスト機構をソフトウェア MPLS 実装である AYAME 上に実装した。本論文では、MPLS マルチキャストのフレームワークをまとめたうえで、我々が実装した MPLS マルチキャスト機構の設計と実装について論じる。

## 2 背景

本章では、本論文の背景として MPLS 技術およびそのマルチキャストへの適用の現状、ソフトウェア MPLS 実装である AYAME について論じる。

### 2.1 MPLS 技術

MPLS とは IETF (Internet Engineering Task Force)<sup>1</sup> で標準化が推進されている技術で、短い固定長の値(ラベル)を用いてパケットを転送することで上位プロトコルの制御に非依存、且つ高速なパケット転送を可能にする技術である。MPLS ではパケットはラベルのスワップ動作によって転送されていく。このパケットが転送されるパスを LSP (Label Switching Path) と呼ぶ。パケット転送機構としての IP は経路制御機能とパケット配送機能という二種の重要な機能を提供している。経路制御機能は経路表を構築する機能でありパケット配送機能は経路表を用いてパケット配送する機能である。IP では、これらの機能は経路表という単一の情報系を共有して動作している。これに対し MPLS は、あらかじめ隣接ルータ間で広告したラベルを用いてパケットの転送時を行なうことでパケット配送機能と経路制御機能を完全に分離している。

この分離した情報系のマッピングに関する隣接ルータ間での折衝はラベル配布プロトコルによって行なわれる。LSR (Label Switching Router) はラベル配布プロトコルを用いて交換したラベル情報 (FIB: Forwarding Information Base) を三種の情報系を用いて保持する。それらは NHLFE (Next Hop Label Forwarding Entry)、ILM (Input Label Mapping)、FTN (FEC to NHLFE) と呼ばれるものであ

<sup>1</sup><http://www.ietf.org/>

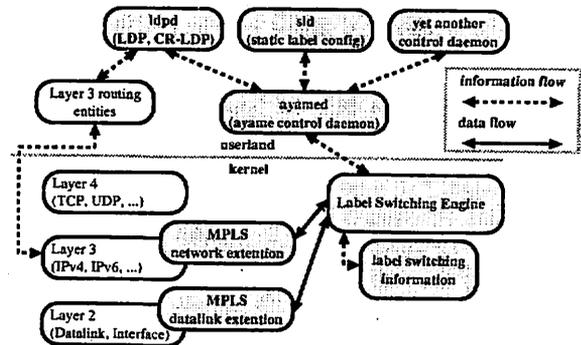


図 1: AYAME のアーキテクチャ

る。NHLFE は、転送するパケットに対する操作(ラベルの PUSH、POP、SWAP)のエントリである。ILM は、ラベルと NHLFE のマッピングであり、あるラベルが付加された入力パケットに対する操作を決定するために用いられる。FTN は、FEC と NHLFE のマッピングであり、あるクラスに分類されたパケットに対する操作を決定するために用いられる。2002 年 9 月の時点では、IP の制御に追従してラベル配布を行なう LDP (Label Distribution Protocol) [5]、特定の制約に基づいてラベル配布を行なう CR-LDP (Constraint-based LDP) [6]、ネットワーク資源の予約と同時にラベル配布を行なう RSVP-TE [7] などがラベル配布プロトコルとして提案されている。

MPLS の基本的な仕様は、2002 年 9 月の時点ではユニキャストを対象としたものしか標準化されておらず、多点間通信、すなわちマルチキャストを対象としたものについてはフレームワークの提示しか行なわれていない。MPLS を利用したマルチキャストには極めて多数の応用が考えられる反面、非常に多くの問題を解決する必要がある。そのため未だに研究的要素が強く、段階的に問題を解決していくことが必要であると考えられる。

### 2.2 AYAME MPLS 実装

AYAME<sup>2</sup> [2] は研究等への利用を主目的の一つとしたソフトウェア MPLS 実装で、NetBSD<sup>3</sup> 上で動作する。AYAME は、NetBSD のカーネル内部に実装されたパケット配送機構、ユーザランドにおける管理デーモンおよびシグナリングデーモンから構成されている。AYAME のアーキテクチャを図 1 に示す。

<sup>2</sup><http://www.ayame.org/>

<sup>3</sup><http://www.netbsd.org/>

LSE (Label Switching Engine) はカーネル内部に実装されたパケット配送機構で、カーネル内部に保持された FIB に基づいて、L3 および L2 に関連する MPLS パケットおよび IP パケット等の入出力を行う。この転送に用いる情報である FIB は、ユーザランドの資源管理デーモンである ayamed と、ラベル制御デーモンによって確立される。各ラベル制御デーモンは、それぞれ個々の制御スキームあるいは同類の制御スキーム毎に個別のデーモンとして動作し、それぞれの制御スキームに従ってラベルマッピングの配布、設定を行う。複数の制御デーモン間で整合性を保持する必要がある情報系は ayamed が管理する。有限の情報空間内の値であるラベル値の整合性や、同様に有限である物理リンクの帯域の分配などは ayamed が管理する。

AYAME では、LDP および CR-LDP を用いてラベルを制御する ldpd、および静的なラベル設定を実現する sld がラベル制御デーモンとして提供されている。ldpd は LDP あるいは CR-LDP を用いてシグナリングを行ないラベルを管理する。sld は管理者が静的にラベルマッピングと分類器のフィルタ設定を記述し LSP を設定するために用いられる。

### 3 MPLS マルチキャストの設計

ここでは、MPLS を用いたマルチキャストのフレームワークをまとめ MPLS マルチキャストのパケット配送機構に関する要求事項を提示し、AYAME を用いた MPLS マルチキャストパケット配送機構の設計について論じる。

#### 3.1 フレームワーク

MPLS では制御と配送は完全に分離しており、ユニキャストだけでなくマルチキャストにおいてもその構造が適用可能である。現在 MPLS を用いたマルチキャストとして IP マルチキャストと MPLS の併用に関する議論が IETF において行なわれているが、MPLS を用いたマルチキャストは IP マルチキャストを併用する方法だけでなく、経路制御には TE、配送するパケットは非 IP というような方法も存在する。以下は、MPLS におけるこのような制御と配送の特徴を IP と IP 以外という観点からまとめたものである。

1. IP 経路制御および IP パケットの配送:  
LDP 等によって IP 経路制御に追従したラベルマッピングを配布し、ラベルを用いて IP パ

ケットを配送する。バックボーンネットワーク等における一般的な MPLS の利用形態。

2. IP 経路制御および IP 以外のパケットの配送:  
LDP (Label Distribution Protocol) 等によって IP 経路制御に追従したラベルマッピングを配布し、ラベルを用いて IP 以外のパケットを配送する。
3. TE 経路制御および IP パケットの配送:  
RSVP-TE や CR-LDP 等のシグナリングプロトコルによって IP 経路制御とは独立したラベルマッピングを配布し、ラベルを用いて IP パケットを配送する。TE を目的とした一般的な MPLS の利用形態。
4. TE 経路制御および IP 以外のパケットの配送:  
RSVP-TE や CR-LDP 等のシグナリングプロトコルによって IP 経路制御とは独立したラベルマッピングを配布し、ラベルを用いて IP 以外のパケットを配送する。

特に上記の 1. については、RFC3353 [8] においてその経路制御とパケット配送に関する要求事項がまとめられている。同 RFC では、上記の 3. および 4. については明示的経路制御の方法として IP マルチキャスト経路制御プロトコルによる配送木を LSP ベースの配送木で上書きする方法、および明示的にユニキャストの経路を用いる方法の二種が考えられると言及するにとどまっておき、IP 以外の経路制御や IP 以外のパケット配送については言及していない。

このように MPLS を用いたマルチキャストについては、単に IP マルチキャストと MPLS の併用だけでなく極めて多数の制御と配送の組み合わせが考えられる。

#### 3.2 要求事項

MPLS マルチキャストの制御は IP マルチキャスト技術を対象とした経路制御や TE を用いたマルチキャスト技術でそれぞれ異なる可能性が極めて高い。そのため、MPLS マルチキャストの制御機構には複数の異なる制御スキームで同一の MPLS マルチキャスト配送機構を利用できるアーキテクチャが要求される。このような中核的な MPLS マルチキャストの基本機能を実現し利用するためには、配送部分は基本的な機能を提供し、その制御インターフェース部分では汎用性の高いインターフェースを提供する必要がある。

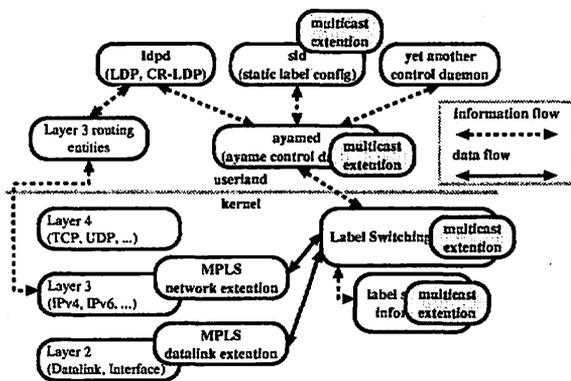


図 2: マルチキャスト機構のアーキテクチャ

このような種々の制御スキームでの応用を目的とする基本的な MPLS マルチキャスト配送機構に求められる基本機能を以下にまとめる。

- 入力されたパケットの複製機能
- 入力されたパケットの複数インターフェースへの同時複数出力機能
- 入力ラベルおよび各種情報によるフィルタを用いたパケット区分化機能

これらの要求事項に加えて、MPLS マルチキャストを行う LSR は単一の FEC に対して入口ノード (ingress)、転送ノード (transit)、出口ノード (egress) のすべての役割を同時に満たすことが要求される。ユニキャストの MPLS の場合、単一のパスについては各ルータで役割が明確に異なるが、マルチキャストの場合、LSP はツリー状であるのである枝については transit、ある枝については egress、そのルータがマルチキャストトラフィックの発生源の場合には ingress としても動作する必要がある。これは、MPLS マルチキャスト LSR は複数インターフェースへの同時複数出力機能として L2.5 および L3 に対する出力機能を持たなければならないということを示している。

### 3.3 AYAME ベースの設計

AYAME をベースとした MPLS マルチキャスト機構の全体像を図 2 に示す。我々は、マルチキャスト配送機構をマルチキャスト拡張 LSE、マルチキャスト拡張 FIB、マルチキャスト拡張制御デモンから構築するという設計を選択した。

マルチキャスト配送に必要な情報は、カーネル空間に存在するマルチキャスト拡張された FIB に保持される。マルチキャスト拡張 FIB は、標準的

な MPLS の FIB と同様 NHLFE、ILM、FTN から構成される。マルチキャスト拡張 NHLFE には、パケットに対する挙動が設定されている。ユニキャストの場合には、POP、SWAP、PUSH の操作、出力インターフェースが設定される。マルチキャスト拡張 NHLFE には、この操作と出力インターフェースのペアがリスト構造で格納される。

パケット複製機能および配送機能は、LSE の拡張として実現される。マルチキャスト拡張された LSE は、マルチキャスト拡張 FIB に基づいて、L3 および L2.5 から入力されたパケットを必要に応じて複製し L3 および L2.5 に適切に出力する。ラベルが付加されたパケットが入力された場合、マルチキャスト拡張 LSE はそのラベル値から NHLFE を取得し、リストの要素の数にパケットを複製し、リストの各要素に従ってパケットに操作を行ない各要素にしたがってパケットを出力する。このリストは出力が L2.5 であっても L3 であっても POP と SWAP というラベル操作として区別なく記述できるという特徴がある。一方、ラベルの付加されていないパケット、すなわち L3 のパケットが入力されてきた場合、L3 の AYAME 拡張によって LSE は FTN を用いてそのパケットに対する操作が設定されている NHLFE を取得する。この NHLFE に従って、ラベルが付加されたパケットが入力された場合と同様の処理を行う。マルチキャスト拡張 LSE とマルチキャスト拡張 FIB の詳細を図 3 に示す。

マルチキャスト拡張 FIB の確立は、ayamed を介してマルチキャスト拡張ラベル制御デモンによって行われる。マルチキャスト用の FIB エントリはユニキャスト用の FIB エントリと同時に存在でき、ユニキャストされるパケットとマルチキャストされるパケットは同一の入出力インターフェースを介して処理されその管理テーブルも同一の管理インターフェースを用いて設定される。また、マルチキャスト拡張ラベル制御デモンは制御スキーム毎に異なる実装が考えられるのでこの ayamed の管理インターフェースは種々のラベル制御デモンで汎用的に利用できる必要がある。

## 4 MPLS マルチキャストの実装

我々は、AYAME の LSE、FIB、ayamed、sld にマルチキャスト拡張を実装することで MPLS マルチキャスト機構を実現した。また、カーネルと ayamed のインターフェース、ayamed と sld のインターフェースにもそれぞれマルチキャスト拡張を行なった。

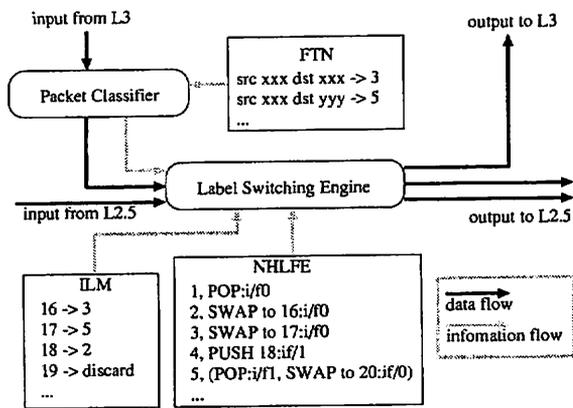


図 3: マルチキャスト配送機構

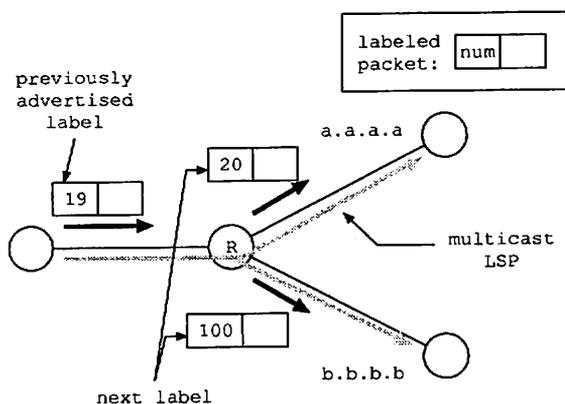


図 4: MPLS マルチキャストの例

我々は、LSEに内部的にラベルスイッチング動作を行わせることでマルチキャスト拡張 NHLFE へ出力インターフェースと操作のペアのリストを格納するという機能を実現した。マルチキャスト拡張 NHLFE でインターフェースと操作のペアに加えて内部的に用いるラベルを保持し、パケットを複製しつつ LSE 内で内部的にラベルスイッチング動作を繰り返すことでマルチキャストラベルスイッチング動作を実現した。この方式は内部的にラベルスイッチング動作を繰り返すためラベルの PUSH、SWAP、POP などの操作を自由に組み合わせることが可能で、非常に柔軟な制御が可能であるという利点を持つ反面、ラベル空間を内部的に消費するという欠点を持つ。我々は、本実装には研究的利用における柔軟な設定が要求されるという点を重視し、このような実装方式を選択した。

このマルチキャスト拡張 NHLFE を構築するための制御デーモンとして我々はマルチキャスト拡張 sld を実装した。sld は静的な LSP を設定するためのデーモンで、マルチキャスト拡張 sld は静

```
! configuration for unicast MPLS
label 0 i8 SWAP y.y.y 10101
label 0 18 filter 1 ANY src 0.0.0.0 0 dst x.x.x.x 32 80
label 0 18 filter 2 ANY src 0.0.0.0 0 dst x.x.x.x 32 8080
! configuration for multicast MPLS
label 0 19 SWAP a.a.a.a 20
label 0 19 SWAP b.b.b.b 100
label 0 19 POP 0.0.0.0
label 0 19 filter 1 ANY src 0.0.0.0 0 dst x.x.x.x 32 0
```

図 5: MPLS マルチキャストの静的設定の例

的な一対多の LSP を設定するためのデーモンである。以下は、マルチキャスト拡張 sld の設定構文である。

1. label {ラベル空間} {ラベル値} {操作} {次ホップ} {次ラベル値}
2. label {ラベル空間} {ラベル値} filter {プロトコル} src {始点アドレス} {始点アドレスマスク長} {始点ポート} dst {終点アドレス} {終点アドレスマスク長} {終点ポート}

1. のコマンドは ILM と NHLFE を同時に設定するためのもので、2. のコマンドはラベル空間とラベル値を NHLFE と対応させるキーとして FTN を設定するためのものである。これらのコマンドは、ユニキャストの場合もマルチキャストの場合も構文自体に差異はない。しかしながら、ユニキャストの場合には、上記の 1. のコマンドはラベル空間とラベル値のペアに対して一意でなければならないが、本論文で提案するマルチキャスト拡張 sld では 1. のコマンドの複数記述を許容できるようにした。これにより、あるラベルのついたパケットを受け取った際のそのパケットに対する LSR の挙動を複数記述でき NHLFE のリストに相当する FIB の構築が可能となった。図 4 に MPLS マルチキャストの例を示す。また図 5 に図 4 の LSR の設定例を示す。

## 5 考察

本章では、MPLS を用いたマルチキャスト配送の実現に必要な、今後解決しなければならない主な問題点に着目して、前章までで論じた我々の実装について論じる。

## 5.1 マルチアクセスリンク

IPでは同一のマルチアクセスリンク上でデータリンクレベルでのマルチキャスト技術が用いられるケースでも、現在の標準的なMPLSでは個々の隣接ノードに対して個別のラベルを付加して送信する必要があり、帯域幅を浪費する可能性がある。我々が実装したMPLSマルチキャストの基本配送機能もこれを踏襲しており、運用の方法によっては帯域幅を浪費する可能性がある。これを本質的に解決するためには、マルチアクセスリンク上での単一ノード対複数ノードのLSPを用いる必要がある。しかしながら、このようなpoint-to-multipointのLSPについては未だ研究段階である。point-to-multipointのLSPはマルチアクセスリンク上で用いられるだけでなく、網全体に対する適用の可能性もあり、また、ユニキャストのpoint-to-pointのLSPと同様の抽象化に基づいた場合にはpoint-to-pointとpoint-to-multipointのLSPを混在させたマルチキャスト技術など、応用の幅が広がることが期待できる。point-to-multipointのLSPは潜在的にIPでのマルチキャスト技術とは根本的に異なるマルチキャスト技術の可能性を保持していると言え、応用の可能性は極めて高く、今後の研究が期待される。我々が実装したMPLSマルチキャスト機構では、マルチアクセスリンク上で単一のマルチキャストフレームを複製して出力することも複製せず出力することも可能であり、柔軟な設定が可能である。

## 5.2 非対称リンク

標準的なMPLSのシグナリングプロトコルではシグナリングの信頼性を高めるため、ピアLSR間での双方向の情報伝達を必要とするケースが多い。衛星リンク等の非対称リンクを用いたマルチキャスト技術をMPLSに適用する場合、なんらかの方法でラベルマッピングを設定する必要があるが、現在の標準的なシグナリングプロトコルは双方向のリンクを前提としているケースが多く問題が多い。我々が実装したラベルマッピングの設定方法は静的な設定でありラベルマッピングの変更が時間的に細粒度で運用する場合には非現実的であるが、非対称リンクなどのシグナリングが困難な場合に関する研究への応用の可能性を示している。

## 5.3 IPマルチキャスト

本論文では、MPLSマルチキャストのフレームワークを示し、さらにその一部としてIPマルチキャストをMPLSに適用するための要求事項のフレームワークを示した。前述のとおり、IPマルチキャストには種々の要素技術が用いられているうえ、伝統的なIPの特徴とも言える制御と配送の非分離を応用している側面もあるため、MPLSにIPマルチキャストを完全に適用するには極めて多くの問題が存在する。MPLSにおいてマルチキャスト型の配送を実現するには、IPマルチキャストで利用されてきた種々の技術をそのままMPLSに適用するだけでなく、MPLSの特徴を利用した新しいマルチキャスト技術を含めて段階的にMPLSマルチキャストを実現していく必要がある。

## 6 まとめ

本論文では、MPLSマルチキャストのフレームワークを提示したうえで、MPLSを用いたマルチキャストの中核的な機能であるパケットの複製および配送機能について我々が実装したパケット配送機構の詳細を論じた。

## 参考文献

- [1] S. Deering., "Host Extensions for IP Multicasting," RFC1112, Internet Engineering Task Force, Aug 1989.
- [2] Yojiro Uo, Satoshi Uda, Nobuo Ogashiwa, Satoshi Ota, Yoichi Shinoda, "AYAME: A design and implementation of the CoS capable MPLS layer for BSD Network stack" INET2000, Internet Society, Jul 2000
- [3] E. Duros et al., "A Link-Layer Tunneling Mechanism for Unidirectional Links," RFC3077, Internet Engineering Task Force, Mar 2001.
- [4] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," RFC3031, Internet Engineering Task Force, Jan 2001.
- [5] L. Andersson, P. Doolan, N. Feldman, A. Fredette, and B. Thomas, "Ldp specification," RFC3036, Internet Engineering Task Force, Jan 2001.
- [6] B. Jamoussi et al., "Constraint-based lsp setup using ldp," RFC3212, Internet Engineering Task Force, Jan 2002.
- [7] D. Awduche et al., "Rsvp-te: Extensions to rsvp for lsp tunnels," RFC3209, Internet Engineering Task Force, Dec 2001.
- [8] D. Ooms et al., "Overview of IP Multicast in a Multi-Protocol Label Switching (MPLS) Environment," RFC3353, Internet Engineering Task Force, Dec 2001.