

OPENFLOW

マスタリング  
TCP/IP

OpenFlow編

あきみち・宮永直樹・岩田 淳 共著



---

---

本書を発行するにあたって、内容に誤りのないようできる限りの注意を払いましたが、本書の内容を適用した結果生じたこと、また、適用できなかった結果について、著者、出版社とも一切の責任を負いませんのでご了承ください。

---

---

本書に掲載されている会社名・製品名は一般に各社の登録商標または商標です。

---

---

本書は、「著作権法」によって、著作権等の権利が保護されている著作物です。本書の複製権・翻訳権・上映権・譲渡権・公衆送信権（送信可能化権を含む）は著作権者が保有しています。本書の全部または一部につき、無断で転載、複写複製、電子的装置への入力等をされると、著作権等の侵害となる場合がありますので、ご注意ください。本書の無断複写は、著作権法上の制限事項を除き、禁じられています。本書の複写複製を希望される場合は、そのつど事前に下記へ連絡して許諾を得てください。

- オーム社開発部「マスタリング TCP/IP OpenFlow 編」係宛、E-mail (kaihatu@ohmsha.co.jp) または書状、FAX (03-3293-2825) にて

## はじめに

OpenFlowは、ネットワーク上で何でも実現できてしまうように思えるほど、さまざまなことができる仕組みです。もちろん、仕様上または機器実装上の制限というものがありますが、工夫をすればかなりのことができます。しかし逆に、設定者が非常に細かく設定を行わないと何もできない仕組み、ともいえます。OpenFlowそのものは、あくまでも細かい設定を可能にする「枠組み」にすぎないからです。事実、何の設定も行われていないOpenFlowネットワークは、一般的なラーニングブリッジ（L2スイッチ）並みのネットワーク機能すら実現できません。

「OpenFlowはプログラマブルである」という表現を耳にしたことはないでしょうか。OpenFlowネットワークにおける各種設定は、パケットを転送するOpenFlowスイッチではなく、複数のOpenFlowスイッチを一括して制御するOpenFlowコントローラで行われます。従来のネットワーク機器は、それぞれが自律分散協調することによってネットワークを構成します。これに対しOpenFlowでは、OpenFlowコントローラがネットワーク内の各機器の細かい挙動を含めて統合管理できるため、OpenFlowコントローラの設定を行うことが、あたかもネットワーク全体を「プログラミング」しているかのような感じになります。これが、「プログラマブル」であるとされるゆえんです。さらに言えば、「OpenFlowそのものをプログラミング言語に例えるとアセンブラだ」というアナロジーもあり得るでしょう。C、C++、Java、Rubyとプログラミング言語の流行が変化していく中で、「かゆいところに手が届くアセンブラ」に似たコンセプトで突っ走ろうというのがOpenFlowというイメージです。

とはいえ筆者らは、OpenFlowをアセンブラに例えるのは、ある意味では間違いであるとも考えています。アセンブラはハードウェアに依存していますが、OpenFlowは単一のベンダーが作成した独自仕様ではなく、さまざまな機器同士が相互接続を行えます。そのような意味では、「OpenFlowそのものは標準ライブラリがないC言語」という比喩も気に入っています。C言語そのものは非常に基本的なプログラミング言語ですが、標準ライブラリやPOSIXなどの標準的なAPIを全く使わずに何かを作るのは大変です。生のOpenFlowも、標準が存在していて相互接続が可能であるという点では、そういった側面があります。将来OpenFlowが実環境で利用されるときには、多くのユーザはOpenFlowを直接使うわけではなく、C言語の標準ライブラリを使うような感じで、統合され整備された環境を経由してOpenFlowを利用するのではないかと筆者らは推測しています。

## ■ 本書が解説する OpenFlow のバージョン

本書は OpenFlow 1.0 を骨子としつつ、OpenFlow 1.3.2 までを解説しています。OpenFlow 1.0 を基本としているのは、OpenFlow 1.0 で OpenFlow の初期設計思想が色濃く反映されているため、OpenFlow を理解するには OpenFlow 1.0 から入るのがわかりやすいと筆者らが考えているためです。

本書の執筆を開始した段階では、OpenFlow スイッチのハードウェアは多くが OpenFlow 1.0 対応として実装されており、OpenFlow 1.1 以降の仕様に対応したハードウェアはあまり存在していませんでした。しかし、本書執筆が終わりに近づいたところに OpenFlow 1.3 ベースの製品が発表され、今後も、OpenFlow 1.3 以降をベースとしたハードウェア実装が増えていくと推測されています。そのため、どのバージョンを中心に解説を行うのかは、執筆中も頭を悩ませる問題のひとつでした。

悩んだ結果、OpenFlow 1.0 をベースに OpenFlow を説明しつつ、本書執筆段階で最新である OpenFlow 1.3.2 までの変化も解説する構成にしました。最初から OpenFlow 1.3 で全部を解説することも検討しましたが、そうしてしまうと、なぜ各機能が追加されていったのかを理解しにくい書籍になってしまうのではという懸念がありました。実際、OpenFlow 1.0 までは、「既存のハードウェアを生かしつつソフトウェアの変更で面白いことをする」という思想がありましたが、OpenFlow 1.3 は、OpenFlow の旧バージョンではできないことを実現するために拡張されていった結果であるという側面も感じられます。

なお、プロトコルに対する変更は、必要に応じて各所で個別に言及するとともに、OpenFlow 1.1、1.2、1.3 の各バージョンについては、それぞれ一章分を割いて解説しています。

## ■ 本書の構成

本書で最も気をつけたのは、特定の実装や製品の紹介にならずに、プロトコルとしての OpenFlow を純粹に伝えるようにすることです。また、プロトコルを利用するうえで陥りやすいと思われる落とし穴に関しても、可能な限り紹介することを目指しました。

第1章「OpenFlow 概要」では、OpenFlow の概要や OpenFlow をとりまく状況などを説明します。第2章「OpenFlow 1.0 の仕組み」は、OpenFlow 1.0 の仕組みやフォーマットを解説します。

第3章から第6章は、OpenFlow の設定例を紹介しつつ、OpenFlow をより深く理解することを目指しました。

第3章「OpenFlow と LLDP」ではネットワークポロジ検出機能を実装するための LLDP を OpenFlow で活用する方法を、第4章「L2 スイッチの機能再現から学ぶ OpenFlow」は OpenFlow を利用してさまざまな L2 ネットワーク機能を実現する方法を、第5章「OpenFlow と仮想化」は OpenFlow によるネットワーク仮想化や VMware などによるサーバ仮想化と OpenFlow の関係などを、

第6章「ユースケースで考えるOpenFlow」はそのほかのさまざまなユースケースを通じてOpenFlowを考察します。これらの設定例をもとに、OpenFlowの実際を伝えられればと考えています。

第7章以降は、OpenFlow 1.0よりも新しいバージョンや今後の動向に着目しています。

第7章、第8章、第9章ではそれぞれ、OpenFlow 1.1、1.2、1.3での変更点を紹介しています。OpenFlow 1.3までの変化をすべてまとめてしまうと、あまりの違いに混乱を誘発するおそれがありそうだと筆者らは考えました。そのため、各バージョンを個別の章にし、それぞれで行われた変更内容から各バージョンアップの設計思想を理解しやすいような構成にしています。

第10章「OpenFlowの留意事項」では、運用などを行う際の留意事項を紹介し、第11章「OpenFlowのこれから」は、標準化のとても重要な要素である相互接続性への取り組みや、OpenFlow仕様に直接含まれない周辺の議論などを紹介しています。

## ■ 謝辞

本書を完成させるにあたりさまざまなご意見をいただいた、大久保修一氏（さくらインターネット株式会社）、菊池之裕（プロケードコミュニケーションズシステムズ株式会社）、安田豊氏（京都産業大学）、@tsubo氏（アイウエオ順）に感謝致します。

さまざまな質問にご回答いただき、また、完成させるにあたりご意見をいただいた、日本電気株式会社の芦原浩司氏、高飛氏、鈴木一哉氏、須堯一志氏、高宮安仁氏（アイウエオ順）、およびテクノバン株式会社の山梨暁氏に感謝致します。

みなさまのご協力なしに、本書完成は不可能でした。専門的な見地からのご助言やご感想をいただけたことを大変感謝しております。みなさま、本当にありがとうございました。

OpenFlowに興味を持つすべての方に本書を楽しんでいただければ幸いです。

2013年7月

著者しるす



---

# 目次

---

<b>第1章</b>	<b>OpenFlow 概要</b>	<b>1</b>
<b>1.1</b>	<b>OpenFlow の成り立ち</b>	<b>2</b>
1.1.1	OpenFlow の歴史	2
1.1.2	ONF (Open Networking Foundation)	3
<b>1.2</b>	<b>既存ハードウェアを活用して面白いことを!</b>	<b>4</b>
<b>1.3</b>	<b>OpenFlow とは、具体的に何か?</b>	<b>5</b>
1.3.1	OpenFlow ネットワークの構成例	5
1.3.2	コントロールプレーンとデータプレーンの分離	6
1.3.3	コントロールプレーン構築手法	7
1.3.4	データプレーン構築手法	8
1.3.5	OpenFlow コントローラと OpenFlow チャンネル	10
1.3.6	フローテーブル	12
<b>1.4</b>	<b>コントローラとスイッチの基本動作</b>	<b>12</b>
1.4.1	OpenFlow スwitch の初期動作	12
1.4.2	プロアクティブ型設定	13
1.4.3	リアクティブ型設定	13
<b>1.5</b>	<b>OpenFlow の動作例</b>	<b>15</b>
1.5.1	動作例 1	15
1.5.2	動作例 2	16
1.5.3	動作例 3	17
1.5.4	その他の動作例	18
<b>1.6</b>	<b>変化し続ける OpenFlow 仕様</b>	<b>19</b>
1.6.1	OpenFlow 1.0 以降、急激に変化する仕様	19
1.6.2	OpenFlow 1.3 標準化後	19
1.6.3	本書での OpenFlow 1.1 以降のバージョン解説について	20

---

## 第2章 OpenFlow 1.0の仕組み 21

---

<b>2.1</b>	<b>OpenFlow 1.0のフローテーブルとフローエントリ</b> .....	<b>22</b>
2.1.1	複数フローテーブル .....	23
<b>2.2</b>	<b>フローエントリ</b> .....	<b>24</b>
2.2.1	ヘッダフィールド .....	24
2.2.2	カウンタ .....	27
2.2.3	アクション .....	28
<b>2.3</b>	<b>アクション</b> .....	<b>29</b>
2.3.1	Forward アクション .....	29
2.3.2	Drop アクション .....	30
2.3.3	Enqueue アクション (Optional) .....	31
2.3.4	Modify-Field アクション (Optional) .....	32
<b>2.4</b>	<b>コントローラとスイッチ間のメッセージ</b> .....	<b>34</b>
2.4.1	セキュアチャネルの確立 .....	34
2.4.2	メッセージフォーマット .....	34
2.4.3	セキュアチャネルの確立および初期設定 .....	36
2.4.4	ハンドシェイク .....	37
2.4.5	Flow-Mod メッセージ .....	41
2.4.6	Packet-In メッセージ .....	49
2.4.7	Packet-Out メッセージ .....	50
2.4.8	Port-Status メッセージ .....	51
2.4.9	Flow-Removed メッセージ .....	52
2.4.10	Error メッセージ .....	54
2.4.11	Barrier メッセージ .....	55
2.4.12	Echo メッセージ .....	56
<b>2.5</b>	<b>OpenFlow Switch Errata Version 1.0.1</b> .....	<b>58</b>
2.5.1	セキュアチャネル切断時の緊急事態モード .....	58
<b>2.6</b>	<b>新しい仕様でいろいろと変わっているので注意</b> .....	<b>59</b>



---

## 第3章      **OpenFlow と LLDP** **61**

---

<b>3.1</b>	<b>LLDP と OpenFlow</b> .....	<b>62</b>
3.1.1	OpenFlow での LLDP 処理概要 .....	62
<b>3.2</b>	<b>LLDP の仕組み</b> .....	<b>64</b>
3.2.1	LLDP で利用される宛先イーサネットアドレス .....	65
3.2.2	LLDP で利用される 3 種類のマルチキャストイーサネット アドレス .....	66
3.2.3	LLDPDU フォーマット .....	67
<b>3.3</b>	<b>OpenFlow における LLDP 活用手法例</b> .....	<b>68</b>
3.3.1	事前の準備 .....	68
3.3.2	Packet-Out メッセージで LLDP フレームを送出 .....	70
3.3.3	Packet-In メッセージで LLDP フレームを OpenFlow コ ントローラへ .....	71
<b>3.4</b>	<b>ダイクストラ法による経路計算</b> .....	<b>72</b>

## 第4章      **L2 スイッチの機能再現から学ぶ OpenFlow** **77**

---

<b>4.1</b>	<b>具体的なネットワーク機器の実現を通して OpenFlow を理解する</b> .....	<b>78</b>
<b>4.2</b>	<b>リピータ HUB</b> .....	<b>79</b>
4.2.1	この例のネットワーク構成 .....	79
4.2.2	プロアクティブ型設定で実現 .....	80
4.2.3	すべてのパケットを OpenFlow コントローラへ Packet- In する方法 .....	82
<b>4.3</b>	<b>ラーニングブリッジ</b> .....	<b>85</b>
4.3.1	この例のネットワーク構成 .....	85
4.3.2	OpenFlow 1.0 でラーニングブリッジに挑戦！ .....	86
4.3.3	ARP を監視してフローエントリを作成してみる .....	93
4.3.4	PC A と PC B が入れ替わるとどうなるか？ .....	99
4.3.5	送信元と宛先イーサネットアドレスのペアで管理する 方法 .....	100
4.3.6	OpenFlow 1.1 以降でのラーニングブリッジ実現方法 ....	100

---

<b>4.4</b>	<b>タグド VLAN</b> .....	<b>101</b>
4.4.1	この例のネットワーク構成 .....	101
4.4.2	タグド VLAN を実現する設定 (OpenFlow スイッチ 1 側) .....	103
4.4.3	タグド VLAN を実現する設定 (OpenFlow スイッチ 2 側) .....	106
4.4.4	この例の注意点 .....	108

---

## 第 5 章      **OpenFlow と仮想化**      **109**

---

<b>5.1</b>	<b>サーバ仮想化とネットワーク仮想化</b> .....	<b>110</b>
5.1.1	サーバ仮想化 .....	110
5.1.2	ライブマイグレーション .....	111
5.1.3	マルチテナント .....	112
5.1.4	ネットワーク仮想化 .....	114
<b>5.2</b>	<b>OpenFlow によるネットワーク仮想化実現手法例</b> .....	<b>115</b>
5.2.1	VLAN ID を利用したもの .....	116
5.2.2	物理ポート/論理ポートを利用したもの .....	116
5.2.3	OpenFlow コントローラの実装のポイント .....	116
5.2.4	その他の方法 .....	117
<b>5.3</b>	<b>FlowVisor</b> .....	<b>118</b>
5.3.1	FlowVisor 概要 .....	118
5.3.2	FlowVisor と OpenFlow コントローラ間の OpenFlow チャンネル .....	118
5.3.3	FlowVisor の多段接続 .....	119
5.3.4	FlowVisor と仮想ネットワーク .....	120
5.3.5	FlowVisor 設定時の注意点 .....	121

---

## 第 6 章      **ユースケースで考える OpenFlow**      **123**

---

<b>6.1</b>	<b>イーサネットアドレスを利用したユーザ管理</b> .....	<b>124</b>
6.1.1	コントローラの実装のポイント .....	125
<b>6.2</b>	<b>ECMP (Equal-Cost Multi-Path)</b> .....	<b>125</b>
6.2.1	この例のネットワーク構成 .....	126
6.2.2	送信元アドレスで分ける場合 .....	126
6.2.3	TCP ポート番号で分ける場合 .....	128

6.2.4	ラウンドロビン方式 .....	128
<b>6.3</b>	<b>簡易ロードバランサ .....</b>	<b>129</b>
<b>6.4</b>	<b>選択的ポートミラーリング .....</b>	<b>130</b>
6.4.1	単純なポートミラーリング .....	130
6.4.2	特定のTCPポートのみをミラーリング .....	131
6.4.3	OpenFlow 1.1 の「グループ」とミラーリング .....	132
6.4.4	複数の OpenFlow スイッチから選択的ミラーリングし つつ監視機器へと転送 .....	132
<b>6.5</b>	<b>セキュリティ製品へのリダイレクト .....</b>	<b>133</b>
<b>6.6</b>	<b>仮想ルータに似た挙動（マルチレイヤスイッチ） .....</b>	<b>134</b>
6.6.1	この例のネットワーク構成 .....	134
6.6.2	同一サブネット内のパケット転送処理 .....	135
6.6.3	仮想ルータを越えるパケット転送処理 .....	136
6.6.4	仮想ルータとして ARP リクエストに返答 .....	137
6.6.5	仮想ルータが ARP でイーサネットアドレス解決 .....	138
6.6.6	TTL の扱い .....	139

---

## **第 7 章      OpenFlow 1.1      141**

---

<b>7.1</b>	<b>OpenFlow 1.1 における変更ポイント .....</b>	<b>142</b>
<b>7.2</b>	<b>マッチフィールドの変更 .....</b>	<b>143</b>
<b>7.3</b>	<b>マルチテーブル仕様の変更（パイプライン処理） .....</b>	<b>144</b>
7.3.1	パイプライン処理 .....	145
7.3.2	メタデータ .....	149
7.3.3	OpenFlow 1.1 でのラーニングブリッジ実装手法を 考える .....	150
<b>7.4</b>	<b>インストラクション .....</b>	<b>151</b>
7.4.1	インストラクションとは .....	151
7.4.2	アクション、アクションセット、アクションリスト、イ ンストラクションの違い .....	152
7.4.3	アクションに対する変更 .....	153
<b>7.5</b>	<b>グループ .....</b>	<b>153</b>
7.5.1	グループテーブル .....	153

---

7.5.2	グループエントリ .....	154
7.5.3	グループタイプ .....	154
7.5.4	グループのグループ .....	159
<b>7.6</b>	<b>仮想ポートの拡張 .....</b>	<b>160</b>
7.6.1	ポート番号の識別子が16ビットから32ビットへ変更 ..	160
<b>7.7</b>	<b>TTLフィールド操作 .....</b>	<b>161</b>
7.7.1	copy TTL inwards / copy TTL outwards .....	161
7.7.2	不正なTTL値のケットを受け取ったときの処理 .....	163
7.7.3	TTLのマッチはできない .....	163
<b>7.8</b>	<b>OpenFlow 1.1 におけるその他の変更 .....</b>	<b>164</b>
7.8.1	MPLSタグとVLANタグのPush/Popに対応 .....	164
7.8.2	OpenFlowハイブリッドスイッチ .....	164
7.8.3	SCTP対応 .....	164
7.8.4	ECN対応 .....	165
7.8.5	OpenFlowスイッチとコントローラ間接続の名称変更 ..	165
7.8.6	緊急事態フローキャッシュ (Emergency Flow Cache) の廃止 .....	166
7.8.7	Vendorメッセージの名称変更 .....	166

## 第8章      **OpenFlow 1.2**      **167**

<b>8.1</b>	<b>OpenFlow 1.2 における変更ポイント .....</b>	<b>168</b>
<b>8.2</b>	<b>OpenFlow eXtensible Match (OXM) .....</b>	<b>169</b>
8.2.1	OXM TLVの基本構造 .....	169
8.2.2	マッチフィールド解析仕様の廃止と Pre-requisite .....	172
8.2.3	OXM マッチフィールド .....	172
8.2.4	OXM におけるワイルドカード .....	174
8.2.5	OXM TLV 例 .....	175
8.2.6	OXM による Set-Field .....	177
8.2.7	TCP、UDP、SCTP、ICMPが同じフィールドをオーバー ロードして使っていたのを廃止 .....	177
<b>8.3</b>	<b>基本的なIPv6対応 .....</b>	<b>177</b>
<b>8.4</b>	<b>複数コントローラ対応 (フェイルオーバーとロードバ ランシング) .....</b>	<b>178</b>
8.4.1	Role .....	178

8.4.2	Role の変更 .....	179
8.4.3	OpenFlow コントローラ間の連携 .....	180
<b>8.5</b>	<b>OpenFlow 1.2 におけるその他の変更.....</b>	<b>181</b>
8.5.1	仮想ポートがロジカルポートと予約ポートに分離 .....	181
8.5.2	Flow-Mod MODIFY/MODIFY_STRICT の仕様変更 .....	181
8.5.3	実験的拡張への対応 .....	182
8.5.4	変更履歴の追加 .....	182
<b>第9章</b>	<b>OpenFlow 1.3</b>	<b>183</b>
<b>9.1</b>	<b>OpenFlow 1.3 における変更ポイント.....</b>	<b>184</b>
<b>9.2</b>	<b>メーターテーブル (QoS 対応) .....</b>	<b>185</b>
9.2.1	メーターエントリ、メーターバンド .....	186
<b>9.3</b>	<b>テーブルミスのデフォルト挙動が Packet-In ではなく Drop に.....</b>	<b>187</b>
9.3.1	テーブルミスフローエントリ .....	187
9.3.2	フローマッチ状態遷移の変更 .....	187
<b>9.4</b>	<b>OpenFlow 1.3 におけるその他の変更.....</b>	<b>188</b>
9.4.1	OpenFlow コントローラと OpenFlow スイッチ間の補助接続 (Auxiliary Connections) .....	188
9.4.2	OpenFlow コントローラと UDP や DTLS での通信が可能に .....	189
9.4.3	IPv6 拡張ヘッダ対応 .....	189
9.4.4	OXM マッチフィールドの追加 .....	190
9.4.5	PBB 対応 .....	190
9.4.6	マルチパートフレームワーク .....	191
9.4.7	ハンドシェイク時の Features 応答メッセージからポート番号情報が削除 .....	191
9.4.8	フローエントリの構成要素が変更 .....	192
<b>9.5</b>	<b>OpenFlow 1.3.1 と 1.3.2.....</b>	<b>192</b>
9.5.1	OpenFlow チャネルにおけるバージョンネゴシエーションの変更 .....	192
9.5.2	OpenFlow コントローラからの OpenFlow チャネル確立 .....	193

## 第10章 OpenFlowの留意事項 195

<b>10.1 Packet-In メッセージの処理負荷</b> .....	<b>196</b>
10.1.1 コントロールプレーンの帯域が狭いことによる障害 .....	197
10.1.2 Packet-In によるメッセージ遅延 .....	198
<b>10.2 マッチとフローに関する留意事項</b> .....	<b>198</b>
10.2.1 TCPのフラグを見られない .....	198
10.2.2 パケットやフレームの任意のフィールドがマッチ可能で あるわけではない .....	199
10.2.3 マッチフィールドの依存関係 .....	200
<b>10.3 実装に依存する事項</b> .....	<b>201</b>
10.3.1 フローエントリ数の上限 .....	201
10.3.2 OpenFlow コントローラが同時に制御可能な OpenFlow スイッチ数の上限 .....	202
10.3.3 buffer_id で示されるバッファが残っているとは限らな い .....	202
10.3.4 OpenFlow チャンネル切断時の再接続タイマー .....	203
<b>10.4 Flow-Mod は下流から</b> .....	<b>203</b>
<b>10.5 Barrier メッセージとエラー</b> .....	<b>204</b>
10.5.1 Flow-Mod 直後の Packet-In の場合 .....	205
10.5.2 別々の OpenFlow スイッチへ Flow-Mod メッセージと Packet-In メッセージを送信する場合 .....	206
<b>10.6 Packet-Out の失敗を検出する方法がない</b> .....	<b>208</b>
<b>10.7 IP フラグメント処理</b> .....	<b>209</b>

## 第11章 OpenFlowのこれから 211

<b>11.1 相互接続性検証</b> .....	<b>212</b>
11.1.1 OF-Test .....	212
11.1.2 プラグフェスト .....	212
11.1.3 2012年に行われた第1回プラグフェスト .....	213
11.1.4 第2回、第3回プラグフェスト .....	213
11.1.5 今後の課題 .....	214
<b>11.2 Northbound API</b> .....	<b>215</b>

---

<b>11.3 OF-CONFIG</b> .....	<b>216</b>
-----------------------------	------------

<b>付録</b>	<b>217</b>
-----------	------------

---

<b>付.1 各バージョンのアクション一覧</b> .....	<b>218</b>
付.1.1 OpenFlow 1.0 のアクション .....	218
付.1.2 OpenFlow 1.1 のアクション .....	219
付.1.3 OpenFlow 1.2 のアクション .....	221
付.1.4 OpenFlow 1.3 のアクション .....	223
<b>付.2 各バージョンのメッセージ一覧</b> .....	<b>225</b>
付.2.1 OpenFlow 1.0 のメッセージ .....	225
付.2.2 OpenFlow 1.1 のメッセージ .....	226
付.2.3 OpenFlow 1.2 のメッセージ .....	227
付.2.4 OpenFlow 1.3 のメッセージ .....	228
<b>付.3 OpenFlow 1.0 から 1.3.2 までの変更点まとめ</b> .....	<b>229</b>
付.3.1 アクションセット .....	229
付.3.2 インストラクション .....	229
付.3.3 アクションリスト .....	229
付.3.4 グループテーブル .....	230
付.3.5 メーターテーブル .....	230
付.3.6 アクション .....	230
<b>付.4 参考文献および URL</b> .....	<b>232</b>
<b>索引</b> .....	<b>237</b>





# 第 1 章

## OpenFlow 概要

本章では、OpenFlow そのものの概要や OpenFlow をとりまく状況などを説明します。ここで紹介する OpenFlow 仕様は、あくまで概要であるため、さらに詳細な情報は第 2 章以降をご覧ください。

# 1.1 OpenFlowの成り立ち

OpenFlowは、もともとは研究用のネットワークを容易に構築するためにスタンフォード大学で考案されたプロトコルでした。ここではOpenFlowの成り立ちを簡単に紹介します。

## 1.1.1 OpenFlowの歴史

ネットワークに関する研究においては、実験用のネットワークを構築するという大事な作業がつきものです。かつては必要な台数のPCを用いて物理的に実験ネットワークを構築する作業が行われていましたが、VMwareなどの仮想化ソフトの登場と普及に伴って、バーチャルマシンを組み合わせた仮想プラットフォームによる実験環境の構築が主流になっていきました。そのような仮想的な実験ネットワーク環境として有名なものに、インターネット全体にわたる広域な分散環境を提供するPlanetLab[1]と呼ばれる仕組みがあります。キャンパス内のネットワークを活用して、インターネット全体ではなくある特定の範囲内のネットワークを仮想的に作る場合もあります。

さらに近年、米国を中心として、新しくインターネットそのものをゼロから作り直すような次世代ネットワーク技術の研究がトレンドとなっています。次世代ネットワーク技術としてはさまざまなものが提案されていますが、そうした新しい仕組みの実験を行うとき、従来のインターネット用通信機器よりもさらに細かい制御が可能な仕組みがあると非常に便利です。

そうした中、さまざまな実験ネットワークを容易に構築できるようにするという目的でスタンフォード大学にて開発されたのがOpenFlowです。実際、米国のNSF (National Science Foundation) が後援する大規模な仮想実験ネットワーク環境であるGENI (Global Environment for Network Innovations) [2] プロジェクトでもOpenFlowが利用されています[3]。

OpenFlowはネットワークの実験や研究にとどまらず広く注目を集め、いくつかのネットワーク機器ベンダーが参加するOpenFlowスイッチコンソーシアムが設立されました。2008年には、このOpenFlowスイッチコンソーシアムによって、OpenFlowの基本的な仕様である「OpenFlow Switching Specification 0.2.0」が策定されました。スタンフォード大学、Deutsche Telekom、Hewlett-Packard、Niciraなどとともに、本書執筆者3人のうちの2人が勤めるNECもOpenFlowスイッチコンソーシアム設立時からメンバーとして参加しています▼。

▼Nicira社は、2012年にVMware社に買収されています。

## 1.1.2 ONF(Open Networking Foundation)

本書執筆現在、OpenFlow仕様を取りまとめているのは、OpenFlowスイッチコンソーシアムではなく Open Networking Foundation (ONF) です。

OpenFlowが盛り上がっていくにつれ、OpenFlowスイッチコンソーシアムに参加を希望する組織が増加し、OpenFlowの標準化に関わりを持ちたいという声も増えました。そのため、OpenFlowスイッチコンソーシアムの活動を引き継ぐという形で、Open Networking Foundationが新たに設立されました。

OpenFlow仕様のうち、OpenFlow 1.1までがOpenFlowスイッチコンソーシアムで策定され、OpenFlow 1.2以降はONFで策定されています（本書執筆現在の最新仕様はOpenFlow 1.3.2）。ただし、OpenFlow 1.0と1.1がONFと全く関係がないわけではありません。OpenFlow 1.0と1.1についてもONFでレビューが行われ、仕様の曖昧な部分などの修正を行ったのち、正式な仕様として認定を行うプロセスを行っています。そのため、OpenFlow 1.0と1.1もONF標準です。

### ■ ONFのディスカッショングループ「Japanese」

ONFでは、標準を議論する前段階の議論を行う場として、ディスカッショングループが用意されています。ONF内にあるディスカッショングループのひとつとして、日本語でやり取りされるメーリングリスト「Japanese」があります（本書執筆時点）。

他のディスカッショングループが技術的な題材であったり、ソフトウェアによって仮想ネットワークの構築や制御を行うSDN (Software Defined Networking)の今後を議論するようなものであることを考えると、日本語でのやり取りを行うためのものが存在するのが非常に特徴的です。それだけ日本におけるOpenFlow熱が他国よりも強いということなのだろうと思います。

## 1.2

# 既存ハードウェアを活用して面白いことを！

OpenFlowの初期設計思想は、「新たなハードウェアを設計せずに、既存のハードウェアに対して新しいソフトウェア更新のみで面白いことをしよう」というものです。そのためにOpenFlowは、多くのネットワーク機器に内蔵されているTCAM (Ternary Content-Addressable Memory) という部品を前提として考案されました。

TCAMは、ビットが「マッチしている」「マッチしていない」「気にしない」という3種類のマッチングを行うための電子部品で、ネットワーク上のスイッチでサブネットマスクやアクセスリスト (Access Control List) をハードウェアで高速に処理するために搭載されています。そのTCAMが、初期のOpenFlowの設計思想において非常に大きな鍵となりました。実際、2008年のACM SIGCOMMに掲載された論文“OpenFlow: Enabling Innovation in College Networks”[3]には以下のように書かれています。

The basic idea is simple: we exploit the fact that most modern Ethernet switches and routers contain flow-tables (typically built from TCAMs) that run at line-rate to implement firewalls, NAT, QoS, and to collect statistics.

日本語訳は、以下のようになります。

基本となるアイデアはシンプルです：最近のイーサネットスイッチやルータには、ファイアウォール、NAT、QoS▼を実現しつつも統計情報を集めることを回線速度に近い性能で行うために、(一般的にはTCAMによって構築された) フローテーブルを持っているという事実を有効に使いましました。

ACM SIGCOMMの論文にあるように、多くのネットワーク機器にはTCAMが搭載されています。OpenFlowは、TCAMが搭載されたネットワーク機器のハードウェア構成を変更することなく、ソフトウェアアップデートだけで動作の変更を実現できるように設計されました。新しくハードウェアを開発せずに高速処理が可能なシステムを実現できるという現実主義的な設計思想です。OpenFlowが大きく注目された背景には、既存の仕組みを活用しつつ仮想的なネットワーク構築が実現できることが計算されたうえで設計されているという要素も大きく影響していると考えられます。

▼Quality of service

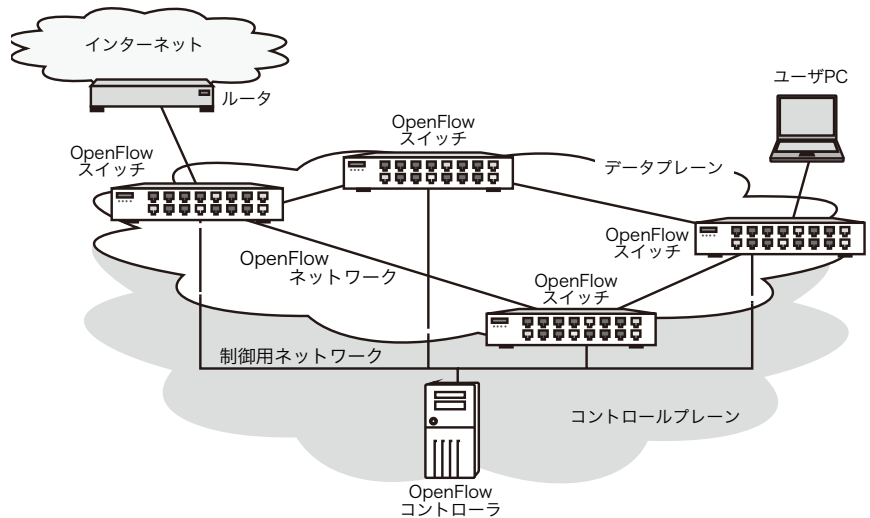
## 1.3 OpenFlowとは、具体的に何か？

OpenFlowは、従来のL2スイッチのようにイーサネットアドレスやVLANタグを使ってスイッチングを行うのではなく、各パケット（もしくはフレーム）が持つイーサネットアドレス、VLANタグ、IPアドレス、TCP/UDPポート番号のような特徴を「フロー」として扱い、それをベースにスイッチングを行い、経路を柔軟に設定できるようにするネットワークを構築するための標準化規格です。

### 1.3.1 OpenFlow ネットワークの構成例

インターネットに接続されたネットワーク内でOpenFlowを利用した構成例を図1.1に示します。

図 1.1  
OpenFlow 構成例



この図では、ユーザPCがOpenFlow ネットワークに接続されていますが、ユーザPCが接続された OpenFlow スwitchの物理ポートは、通常のL2 スwitchと同様の機能をユーザPCに対して提供します。この構成例では、OpenFlow ネットワークはL2 ネットワークなので、ユーザPCはルータとL2接続を行っているように認識します。

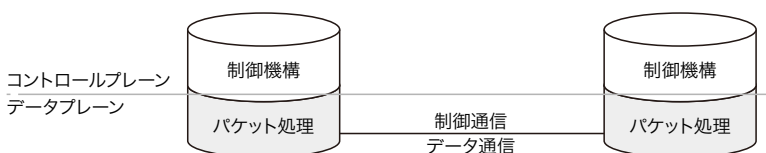
### 1.3.2 コントロールプレーンとデータプレーンの分離

OpenFlowの代表的な特徴として、図1.1のように制御用のネットワークと、パケット転送用のネットワークが分離されている点があげられます。制御用ネットワークは「コントロールプレーン」、パケット転送用ネットワークは「データプレーン」とそれぞれ呼ばれます。

従来のネットワーク機器は、図1.2のようにコントロールプレーンとデータプレーンが同一機器内に同居していました。インターネットは自律分散協調が非常に大きなキーワードであり、各機器が自律して各自で判断するというモデルであるためです。

図1.2

従来のコントロールプレーンとデータプレーン



たとえば、IPプロトコルを利用してパケットの転送を行うルータは、コントロールプレーンとデータプレーンが同居したシステムです。各ルータは、ルーティングプロトコルなどを利用して各自が適切な経路を計算し、その計算結果をもとにパケットの転送を行います。ルータ内に、経路を計算するためのコントロールプレーンと、パケットを転送するためのデータプレーンの両方が存在しています。

OpenFlowで採用されているコントロールプレーンとデータプレーンの分離モデルは、図1.3のようになっています。パケットを転送するデータプレーンを、外部に存在しているコントロールプレーンが制御しています。

図1.3

OpenFlowのコントロールプレーンとデータプレーン

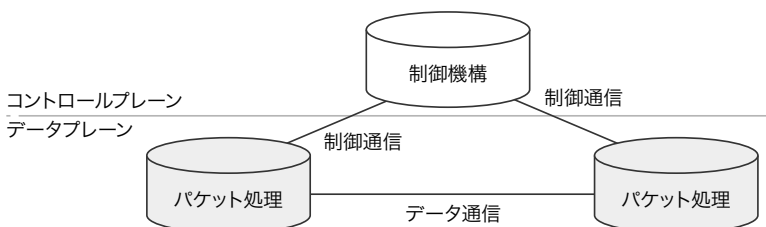


図 1.3 を OpenFlow ネットワークの構成要素で考えると、コントロールプレーン部分が OpenFlow コントローラで、データプレーン部分が OpenFlow スイッチになります。「OpenFlow は中央集権モデルである」という表現がよく使われますが、それはコントロールプレーンとデータプレーンを別機器として分離したうえで、OpenFlow コントローラが複数の OpenFlow スイッチを同時に制御するようにするという、OpenFlow の思想の結果であると筆者らは考えます。

このようなコントロールプレーンとデータプレーンを分けるという思想そのものは特に新しいものではなく、以前からたびたび提唱される方式ではありますが、OpenFlow の仕組みを知るうえで重要な視点です。

### 1.3.3 コントロールプレーン構築手法

OpenFlow ネットワークを実現するうえで、コントロールプレーンをどのように構築するのがポイントのひとつになっています。

データプレーンを実現する OpenFlow スイッチは、コントロールプレーンを実現する OpenFlow コントローラからの命令に従った動きをしますが、その命令をやり取りするために両者が TCP/IP を使った通信ができる環境が必要です。具体的には、コントロールプレーン用のネットワークにおいて、OpenFlow スイッチと OpenFlow コントローラに適切な IP アドレスや経路設定が行われている必要があります。

OpenFlow ネットワークを構築するにあたり、コントロールプレーンとデータプレーンは基本的に別々のネットワークとして用意する必要があります。ただし、どのように「別々のネットワーク」を実現するのかに関しては、いくつかの方式があり、必ずしも別々の物理ネットワークを用意しなければならないというわけではありません。

たとえば、このあと紹介するオーバーレイ方式のように、データプレーンをオーバーレイネットワークとして構築する場合、オーバーレイネットワーク構築に利用する既存網を、そのままコントロールプレーンとして利用することも可能です。このほか、VLAN などを利用して同じ物理ネットワークを論理的に分割したうえで、データプレーンとコントロールプレーンを構築したり、データプレーンとコントロールプレーンを同居させる「In-band コントロールチャネル」という技術を利用することも可能です。

なお、コントロールプレーンとデータプレーン用に、別々の物理ネットワークを整備することももちろん可能です。物理的に分離されていたほうが、障害の切り分けなどが行いやすい傾向があります。

### 1.3.4 データプレーン構築手法

コントロールプレーン用の TCP/IP ネットワークを用意したうえで、OpenFlow ネットワークのデータプレーンをどのように構築するのも大きなポイントです。

データプレーンの構築手法としては、OpenFlow スイッチ同士を直接つなぐ「Hop-by-Hop 方式」、オーバーレイネットワークを通じて OpenFlow スイッチ同士をつなぐ「オーバーレイ方式」、両方を組み合わせた「ハイブリッド方式」の3種類があります。

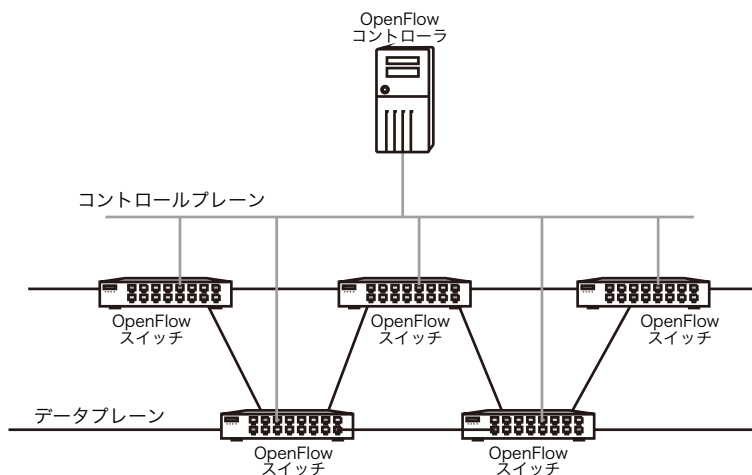
#### ■ Hop-by-Hop 方式

図 1.4 のように、OpenFlow スイッチ同士のデータプレーンが物理的に直接つながっているのが Hop-by-Hop 方式です。

▼ここで述べている「ハイブリッド方式」はデータプレーン構築手法に関するハイブリッドで、Hop-by-Hop とオーバーレイのハイブリッドです。OpenFlow 1.1 から、1つの物理ポート内で OpenFlow と既存 L2 機能が同居した「OpenFlow ハイブリッドスイッチ」(7.8.2項参照)がありますが、その「ハイブリッド」とは無関係なのでご注意ください。

図 1.4

Hop-by-Hop 方式例の物理的ネットワークポロジ



Hop-by-Hop 方式の利点としては、OpenFlow スイッチのデータプレーンが直接接続されていることがあげられます。間に他の機器が入るオーバーレイ方式と比べて、高速な処理が可能であることや、障害の切り分けのための要素が少ないなどの利点があります。

#### ■ オーバーレイ方式

OpenFlow スイッチ同士を直接つなぐことが難しい環境において OpenFlow を導入する手法として、オーバーレイ方式があります。この方式では、データプレーン用に IP トンネルなどの技術を利用してオーバーレイネットワークを組みます。

オーバーレイ方式の例を図 1.5 と 1.6 に示します。図 1.5 は、物理的なネットワークポロジと OpenFlow スイッチ同士をつなぐ IP トンネルを示したもので



す。本書執筆時点では、図 1.5 のように、OpenFlow スイッチは仮想スイッチで実装されることが多く、オーバーレイ方式は仮想化環境を暗黙の前提とされている場合があります。

図 1.5

オーバーレイ方式例の物理的ネットワークポロジ

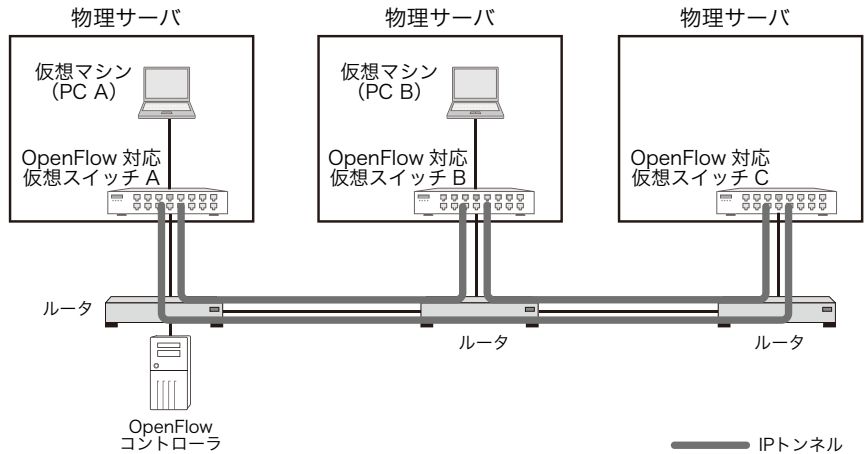
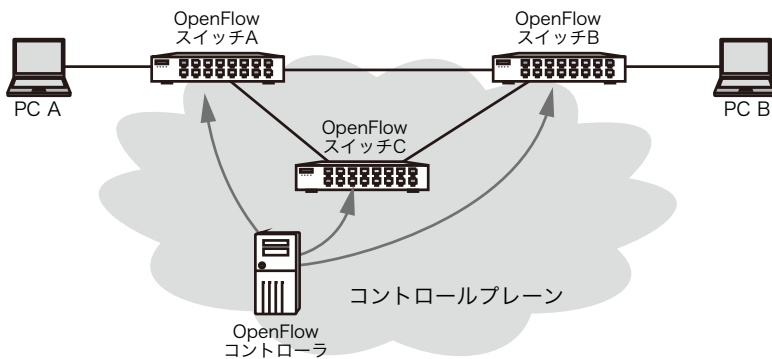


図 1.5 では、各 OpenFlow スイッチと PC は仮想サーバとして同じ物理サーバ内で稼働しています。各 OpenFlow スイッチは、IP トンネルによって互いに接続し、図 1.6 のような論理的構成を構築しています。

図 1.6

オーバーレイ方式例の論理的ネットワークポロジ



各 OpenFlow スイッチはコントロールプレーン用のネットワークで OpenFlow コントローラと接続し、データパケットは OpenFlow スイッチ同士が接続するデータプレーンを利用します。

Hop-by-Hop 方式は、データプレーンのために専用ネットワークを新たに整備することが要求されがちですが、オーバーレイ方式は既存網と OpenFlow 網を同居させやすいという利点があります。一方で、カプセル化を前提としているため、処理性能が Hop-by-Hop よりも低くなりがちであったり、MTU (Maximum

▼ただし、ジャンボパケットやIPトンネルとフラグメンテーションの組み合わせでMTUを小さくしないようにするという手法もあります。

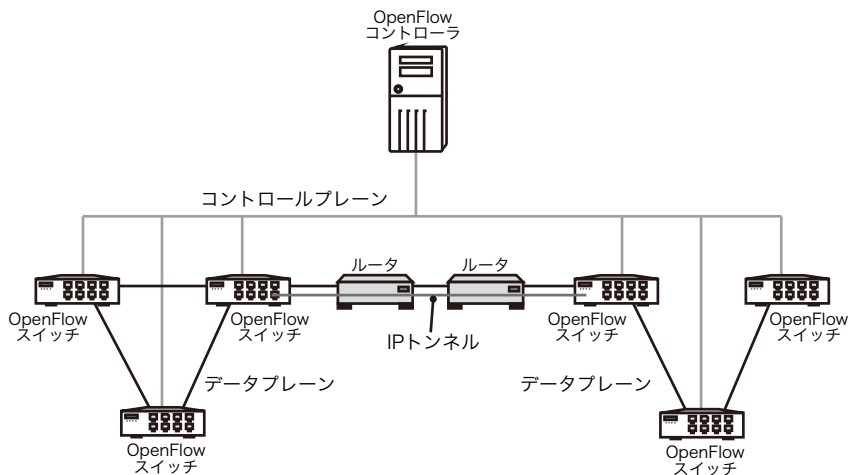
Transfer Unit) が小さくなるという欠点もあります▼。

## ■ ハイブリッド方式

Hop-by-Hop方式とオーバーレイ方式を組み合わせたのが、ハイブリッド方式です。

図 1.7

ハイブリッド方式



ハイブリッド方式では、図 1.7 のように、Hop-by-Hop で接続しあう OpenFlow スイッチと、オーバーレイネットワーク上で接続しあう OpenFlow スイッチの両方が存在しています。

必要に応じて各所で最適な方法を採用しつつ、OpenFlow ネットワークを構築できるのが特徴です。

## ▼ 1.3.5 OpenFlow コントローラと OpenFlow チャンネル

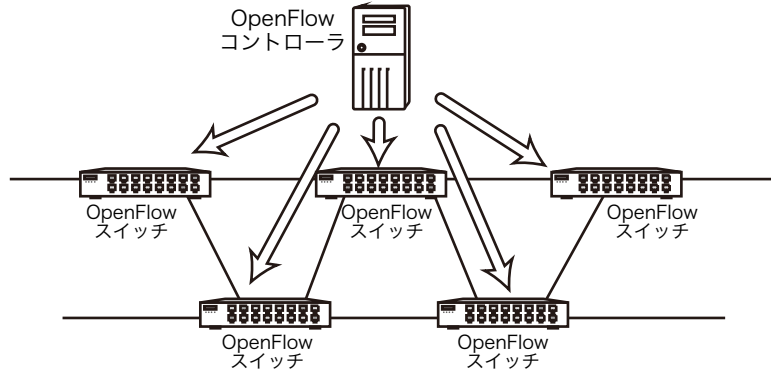
OpenFlow は、OpenFlow コントローラと呼ばれるサーバが OpenFlow ネットワーク上の OpenFlow スイッチのフォワーディングテーブル設定を行う中央集権モデルです (図 1.8)。OpenFlow スイッチは、OpenFlow コントローラから入力される情報を反映するという単純な機能を実現することに徹しています。

OpenFlow コントローラと OpenFlow スイッチは、OpenFlow チャンネル▼による TCP 接続を維持します。これは、OpenFlow コントローラが常に OpenFlow スイッチの状態を監視しながら必要に応じて指令などを出せるようにするためです。

▼OpenFlow 1.0 では、OpenFlow スイッチと OpenFlow コントローラ間のコントロールプレーン用接続が「セキュアチャンネル」という名称になっていますが、この接続が必ず暗号化されているとは限らないため、OpenFlow 1.1 で「OpenFlow チャンネル」と名称が変更されました (詳細は 2.4.1 項および 7.8.5 項参照)。本書では、OpenFlow 1.0 を解説した第 2 章以外では、より正確な用語である「OpenFlow チャンネル」を使うことにしました。

図 1.8

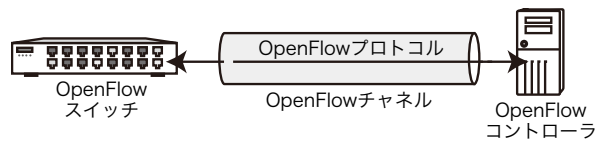
OpenFlow コントローラと OpenFlow スイッチ



OpenFlowを構成する基本要素は、図 1.9 のように、OpenFlow コントローラ、OpenFlow スイッチ、OpenFlow コントローラとスイッチ間の OpenFlow チャンネルによる接続、OpenFlow スイッチでのフローテーブルです。

図 1.9

OpenFlow チャンネル



OpenFlowでは、OpenFlow コントローラがOpenFlow ネットワーク全体を集中管理します。ただ、OpenFlow スイッチのすべての行動に関して逐一命令を出しているわけではありません。あらかじめ「こうなったらこうしなさい」という設定をOpenFlow コントローラがOpenFlow スイッチに対して通知し、OpenFlow スイッチはそれに従います。人間の体のアナロジーでいうと、あらかじめ訓練しておけば、脳の命令を待たずに反射的に行動できるようなものです。「こうなったらこうしなさい」を表現しているのが「フローテーブル」です。

「こうなったらこうしなさい」という OpenFlow からの通知のタイミングは大きく 2 つに分かれます。1 つが、OpenFlow コントローラと OpenFlow スイッチ間の OpenFlow チャンネルが確立した直後など、事前に設定通知が行われる「プロアクティブ型」です。もう 1 つが、OpenFlow スイッチが未知の packet を受け取ったときなどに OpenFlow コントローラにどうすればよいかを問い合わせる「リアクティブ型」です（プロアクティブ型／リアクティブ型設定の詳細は第 2 章を参照してください）。

### 1.3.6 フローテーブル

フローテーブルは、OpenFlow スイッチにおいて、どのようなパケットがどのように処理されるのかといった設定情報や、それらの処理が行われた回数などの記録を記述したものです。OpenFlow スイッチは複数のフローテーブルを持つことが可能です。

各フローの挙動そのものを個別に定義しているのが、フローテーブルに含まれるエン트리（以降、フローエン트리）です。各フローテーブルは、複数のフローエントリを持つことができます。OpenFlow では、OpenFlow コントローラが OpenFlow スイッチに対してフローエントリを設定することで、各種制御を行います。

フローエントリは、各パケットを OpenFlow スイッチがどのように扱うのかを記述したものです。具体的には、OpenFlow スイッチの物理ポート番号などの L1 に関する情報から、イーサネットアドレスなどの L2 情報、IP アドレスなどの L3 情報、TCP/UDP ポート番号などの L4 の情報などに基づいて、そのパケットをどのポートに送出するか、あるいは破棄するかといった処置を記述できます。パケットの送出を行う前にヘッダを変更することもできます。

## 1.4 コントローラとスイッチの基本動作

ここでは OpenFlow における OpenFlow コントローラと OpenFlow スイッチの基本的な動作を説明します。

### 1.4.1 OpenFlow スイッチの初期動作

OpenFlow スイッチが起動すると、OpenFlow コントローラとの「OpenFlow チャネル」と呼ばれる接続を確立します。接続は OpenFlow スイッチ側から OpenFlow コントローラに対して行われます。OpenFlow コントローラは、OpenFlow スイッチからの接続を待ち、接続を行ってもよい相手であれば OpenFlow チャネルによる接続を確立し、維持します。

OpenFlow コントローラとの接続に必要な、OpenFlow コントローラの IP アドレスなどの設定は、あらかじめ OpenFlow スイッチに対して行われている必要があります。また、コントロールプレーンとなるネットワークの設定や、OpenFlow スイッチ自身の IP アドレスなどの設定も必要です ▼。

データプレーンとなる OpenFlow スイッチ間のネットワークは、各 OpenFlow スイッチに設定されたフローエントリによって制御されるため、OpenFlow コントローラが OpenFlow スイッチに対して設定を行う形でネットワーク設定が

▼OpenFlow スイッチの初期設定を行うための OF-CONFIG という仕様もあります。OF-CONFIG に関しては、11.3 節を参照してください。

# ■ 索引 ■

\* 斜体で示すページ番号は傍注に用語があることを示す。

記号・数字	
3GPP .....	165
A	
ACM SIGCOMM (OpenFlow論文) .....	4
actions (Flow-Mod) .....	47(表)
actions_len (Packet-Out) .....	51(表)
ALL	
Forward アクション .....	29(表)
OpenFlow 1.3 .....	181(表)
仮想ポート .....	30
予約された仮想ポート .....	160
all (グループタイプ) .....	154(表), 155
ANY .....	46
ARP	
Gratuitous ARP .....	112
OpenFlow 1.0 .....	27
監視によりフローエントリ作成 .....	93
リクエストに返答 .....	137
ARP opcode (マッチフィールド) .....	143(表)
Authentication (IPv6 拡張ヘッダ) .....	190
Auxiliary Connection .....	188
B	
Barrier 応答メッセージ .....	35(表)
OpenFlow 1.0 .....	225(表)
OpenFlow 1.1 .....	226(表)
OpenFlow 1.2 .....	227(表)
OpenFlow 1.3 .....	228(表)
Barrier メッセージ	
実装依存 .....	204
利用例 .....	55
Barrier 要求メッセージ .....	35(表)
OpenFlow 1.0 .....	225(表)
OpenFlow 1.1 .....	226(表)
OpenFlow 1.2 .....	227(表)
OpenFlow 1.3 .....	228(表)
buffer_id (Flow-Mod) .....	47(表)
パケット消失 .....	202
バッファ削除のタイミング .....	203
buffer_id (Packet-In) .....	50(表)
buffer_id (Packet-Out) .....	51(表)
C	
C-DA .....	190
C-SA .....	190
CGN (Carrier Grade NAT) .....	115
Change-TTL (アクション)	
OpenFlow 1.2 .....	221(表)
OpenFlow 1.3 .....	223(表)
Chassis ID (LLDPDU) .....	67, 68(表)
command (Flow-Mod) .....	47(表)
CONTROLLER	
Forward アクション .....	29(表)
OpenFlow 1.3 .....	181(表)
仮想ポート .....	30
予約された仮想ポート .....	160
copy TTL inwards (アクション) .....	161
copy TTL outwards (アクション) .....	161
copy TTL outwards (アクションセットの実行順) ..	146(表)
CoS .....	33
D	
Data Center TCP .....	165
datapath .....	38(表)
datapath_id .....	38(表)
DDoS 攻撃 (OpenFlow でパケット破棄) .....	134
decrement TTL (アクションセットの実行順) .....	146(表)
Destination options (IPv6 拡張ヘッダ) .....	189
Deutsche Telekom .....	2
DiffServ .....	33, 186
Drop (アクション) .....	30
OpenFlow 1.0 .....	218(表)
OpenFlow 1.1 .....	219(表)
OpenFlow 1.2 .....	221(表)
OpenFlow 1.3 .....	223(表)
DS field .....	33
DSCP .....	44(表), 45, 48(表)
DS バイト .....	33
DTLS .....	189
E	
Echo 応答メッセージ .....	35(表)
OpenFlow 1.0 .....	225(表)
OpenFlow 1.1 .....	226(表)
OpenFlow 1.2 .....	227(表)
OpenFlow 1.3 .....	228(表)
Echo メッセージ .....	56(図)
Echo 要求メッセージ .....	35(表)
OpenFlow 1.0 .....	225(表)
OpenFlow 1.1 .....	226(表)
OpenFlow 1.2 .....	227(表)
OpenFlow 1.3 .....	228(表)
ECMP .....	125
ラウンドロビン方式 .....	128
ロードバランサ (OpenFlow で実現) .....	129
ECN .....	33, 165
Encrypted Security Payload (IPv6 拡張ヘッダ) .....	190
End Of LLDPDU .....	67
End Of LLDPDU (LLDPDU) .....	68(表)
Enqueue (アクション) .....	31, 153, 218(表)
EQUAL (コントローラの Role) .....	178
Error メッセージ .....	35(表), 54(図)
OpenFlow 1.0 .....	225(表)
OpenFlow 1.1 .....	226(表)
OpenFlow 1.2 .....	227(表)
OpenFlow 1.3 .....	228(表)
Ethernet destination address	
ヘッダフィールド .....	24(表)
マッチフィールド .....	143(表)
Ethernet source address	
ヘッダフィールド .....	24(表)
マッチフィールド .....	143(表)
Ethernet type	
ヘッダフィールド .....	24(表)

マッチフィールド .....	143 (表)
EUI-48 .....	65
Experimenter (アクション)	
OpenFlow 1.1 .....	220 (表)
OpenFlow 1.2 .....	222 (表)
OpenFlow 1.3 .....	224 (表)
Experimenter メッセージ .....	166
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)

## F

fail secure mode .....	166
fail standalone mode .....	166
fast failover (グループタイプ) .....	154 (表), 158
Features 応答メッセージ .....	35 (表), 38 (図)
OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)
Features 要求メッセージ .....	35 (表), 38 (図)
OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)
flags (Flow-Mod) .....	47 (表)
FLOOD	
Forward アクション .....	29 (表)
OpenFlow 1.3 .....	181 (表)
仮想ポート .....	30
予約された仮想ポート .....	160
Flow Statistics Request メッセージ .....	48
Flow Stats Request メッセージ .....	45
Flow-Mod メッセージ .....	35 (表), 41, 43 (図)
OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)
設定例 .....	69, 80, 82, 86, 89, 93, 103
直後に依存する Packet-Out を送る .....	205
Flow-Removed メッセージ .....	35 (表), 52, 52 (図)
OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)
FlowVisor .....	118, 178, 213
設定時の注意点 .....	121
多段接続 .....	119
Forward (アクション) .....	29
OpenFlow 1.0 .....	218 (表)
例 .....	30
Fragmentation (IPv6 拡張ヘッダ) .....	189

## G

generation_id .....	180
GENI .....	2
GET_ASYNC 応答メッセージ .....	228 (表)
GET_ASYNC 要求メッセージ .....	228 (表)
GET_CONFIG 応答メッセージ .....	35 (表)
OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)

GET_CONFIG 要求メッセージ .....	35 (表), 40
OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)
Gratuitous ARP .....	112, 117
Group (アクション) .....	153
OpenFlow 1.1 .....	220 (表)
OpenFlow 1.2 .....	222 (表)
OpenFlow 1.3 .....	224 (表)
group (アクションセットの実行順) .....	146 (表)
GROUP_MOD メッセージ	
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)

## H

hard_timeout (Flow-Mod) .....	43, 47 (表)
Hello メッセージ .....	35 (表), 37 (図)
OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)
Hewlett-Packard .....	2
Hop Limit .....	177
Hop-by-hop (IPv6 拡張ヘッダ) .....	189
Hop-by-Hop 方式 .....	8
HUB (OpenFlow で実現) .....	79

## I

ICMP code	
ヘッダフィールド .....	24 (表)
マッチフィールド .....	143 (表)
ICMP type	
ヘッダフィールド .....	24 (表)
マッチフィールド .....	143 (表)
idle_timeout (Flow-Mod) .....	43, 47 (表)
IEEE OUI .....	166
IEEE802.1AB .....	62, 65
IEEE802.1ah .....	190
IEEE802.1Aj .....	66 (表)
IEEE802.1aq .....	125
IEEE802.1D .....	25
IEEE802.1P .....	33
IEEE802.1Q .....	33, 143 (表)
In-band コントロールチャネル .....	7
IN_PORT	
Forward アクション .....	29 (表)
OpenFlow 1.3 .....	181 (表)
仮想ポート .....	30
予約された仮想ポート .....	160
in_port (Packet-In) .....	50 (表)
in_port (Packet-Out) .....	51 (表)
indirect (グループタイプ) .....	154 (表), 157
Ingress port	
ヘッダフィールド .....	24 (表)
マッチフィールド .....	143 (表)
Interop Tokyo 2009 .....	115
Interop Tokyo 2013 .....	19
IP destination address	
ヘッダフィールド .....	24 (表)
マッチフィールド .....	143 (表)
IP protocol	
ヘッダフィールド .....	24 (表)

マッチフィールド .....	143 (表)
IP source address .....	
ヘッダフィールド .....	24 (表)
マッチフィールド .....	143 (表)
IPv6 拡張ヘッダ .....	178, 189
IPv6 対応 .....	177
IPv6 フラグメント .....	178
IP カプセル化 .....	161
IP フラグメント .....	209
IPv6 .....	209
IS-IS .....	72
ISL 方式 (Inter Switch Link) .....	33

**J**

Japanese .....	3
----------------	---

**K**

Keep-alive パケット .....	57
-----------------------	----

**L**

L2 スイッチ (OpenFlow で実現) .....	78
LLDP .....	62
イーサネットタイプ .....	64
活用例 .....	68
仕組み .....	64
マルチキャストアドレス .....	66
LLDPDU .....	64
フォーマット .....	67
LOCAL .....	
Forward アクション .....	29 (表)
OpenFlow 1.3 .....	181 (表)
仮想ポート .....	30
予約された仮想ポート .....	160
LTE .....	165

**M**

MAC-48 .....	65
mac-in-mac .....	190
Management Address (LLDPDU) .....	68 (表)
MAP-E .....	200
MASTER (コントローラの Role) .....	178
Metadata .....	143 (表)
METER_MOD メッセージ .....	228 (表)
Meter インストラクション .....	186
Modify Ethernet destination MAC address .....	32 (表)
Modify Ethernet source MAC address .....	32 (表)
Modify Flow Entry Message .....	41
Modify IPv4 destination address .....	32 (表)
Modify IPv4 source address .....	32 (表)
Modify IPv4 ToS bits .....	32 (表)
Modify transport destination port .....	32 (表)
Modify transport source port .....	32 (表)
Modify-Field (アクション) .....	32, 153, 218 (表)
MPLS label .....	143 (表)
MPLS traffic class .....	143 (表)
MPLS トラフィッククラス .....	143
MPLS ラベル .....	143
MTU .....	9, 209
MULTIPART 応答メッセージ .....	228 (表)
MULTIPART 要求メッセージ .....	228 (表)

**N**

NAT .....	129
-----------	-----

Nearest Bridge .....	66 (表)
Nearest Customer Bridge .....	66 (表)
Nearest non-TPMR Bridge .....	66 (表)
NEC .....	2
NETCONF .....	216
Nicira .....	2
No Next Header (IPv6 拡張ヘッダ) .....	190
NORMAL .....	
Forward アクション .....	29 (表)
OpenFlow 1.3 .....	181 (表)
仮想ポート .....	30
予約された仮想ポート .....	160
Northbound API .....	215
NSF .....	2

**O**

OF-CONFIG 1.0 .....	216
OF-CONFIG 1.1 .....	216
OF-Test .....	212
ofp_action_header 構造体 .....	48 (図)
ofp_action_type .....	48
ofp_config_flags .....	163
ofp_flow_mod_command .....	41
ofp_match 構造体 .....	
OpenFlow 1.0 .....	44 (図), 44 (表)
OpenFlow 1.2 .....	169 (図)
複数の OXM TLV が含まれる例 .....	176
ワイルドカード .....	45
ofp_phy_port 構造体 .....	39
ofp_port 構造体 .....	39
OPPAT_COPY_TTL_IN .....	
OpenFlow 1.1 .....	219 (表)
OpenFlow 1.2 .....	221 (表)
OpenFlow 1.3 .....	223 (表)
OPPAT_COPY_TTL_OUT .....	
OpenFlow 1.1 .....	219 (表)
OpenFlow 1.2 .....	221 (表)
OpenFlow 1.3 .....	223 (表)
OPPAT_DEC_MPLS_TTL .....	
OpenFlow 1.1 .....	219 (表)
OpenFlow 1.2 .....	221 (表)
OpenFlow 1.3 .....	223 (表)
OPPAT_DEC_NW_TTL .....	
OpenFlow 1.1 .....	219 (表)
OpenFlow 1.2 .....	221 (表)
OpenFlow 1.3 .....	223 (表)
OPPAT_ENQUEUE .....	218 (表)
OpenFlow 1.0 .....	48 (表)
OPPAT_EXPERIMENTER .....	
OpenFlow 1.1 .....	220 (表)
OpenFlow 1.2 .....	222 (表)
OpenFlow 1.3 .....	224 (表)
OPPAT_GROUP .....	
OpenFlow 1.1 .....	220 (表)
OpenFlow 1.2 .....	222 (表)
OpenFlow 1.3 .....	224 (表)
OPPAT_OUTPUT .....	30
OpenFlow 1.0 .....	48 (表), 218 (表)
OpenFlow 1.1 .....	219 (表)
OpenFlow 1.2 .....	221 (表)
OpenFlow 1.3 .....	223 (表)
OPPAT_POP_MPLS .....	
OpenFlow 1.1 .....	220 (表)
OpenFlow 1.2 .....	221 (表)

OpenFlow 1.3	223 (表)	OFFPET_BAD_ACTION (Errorメッセージ)	54 (表)
OFFPAT_POP_PBB	223 (表)	OFFPET_BAD_REQUEST (Errorメッセージ)	54 (表)
OFFPAT_POP_VLAN		OFFPET_FLOW_MOD_FAILED (Errorメッセージ)	54 (表)
OpenFlow 1.1	220 (表)	OFFPET_GROUP_MOD_FAILED	160
OpenFlow 1.2	221 (表)	OFFPET_HELLO_FAILED (Errorメッセージ)	54 (表)
OpenFlow 1.3	223 (表)	OFFPET_PORT_MOD_FAILED (Errorメッセージ)	54 (表)
OFFPAT_PUSH_MPLS		OFFPET_QUEUE_OP_FAILED (Errorメッセージ)	54 (表)
OpenFlow 1.1	220 (表)	OFFPET_ROLE_REQUEST_FAILED	180
OpenFlow 1.2	221 (表)	OFFPFC_ADD (Flow-Mod)	42 (表)
OpenFlow 1.3	223 (表)	OFFPFC_DELETE (Flow-Mod)	42 (表)
OFFPAT_PUSH_PBB	223 (表)	OFFPFC_DELETE_STRICT (Flow-Mod)	42 (表)
OFFPAT_PUSH_VLAN		OFFPFC_MODIFY (Flow-Mod)	42 (表)
OpenFlow 1.1	220 (表)	OFFPFC_MODIFY (Flow-Modのコマンド)	181
OpenFlow 1.2	221 (表)	OFFPFC_MODIFY_STRICT (Flow-Mod)	42 (表)
OpenFlow 1.3	223 (表)	OFFPFC_MODIFY_STRICT (Flow-Modのコマンド)	181
OFFPAT_SET_DL_DST		OFFPFF_CHECK_OVERLAP	47 (表)
OpenFlow 1.0	48 (表), 218 (表)	OFFPFF_EMERG	47 (表)
OpenFlow 1.1	219 (表)	OFFPFF_RESET_COUNT	182
OFFPAT_SET_DL_SRC		OFFPFF_SEND_FLOW_REM	47 (表)
OpenFlow 1.0	48 (表), 218 (表)	OFFPGMFC_CHAINING_UNSUPPORTED	160
OpenFlow 1.1	219 (表)	OFFPHFC_EPERM	54
OFFPAT_SET_FIELD		OFFPHFC_INCOMPATIBLE	54
OpenFlow 1.2	221 (表)	OFFPMT_OXM	169
OpenFlow 1.3	223 (表)	OFFPMT_STANDARD	169
OFFPAT_SET_MPLS_LABEL	219 (表)	OFFPP_ALL (仮想ポート)	
OFFPAT_SET_MPLS_TC	219 (表)	OpenFlow 1.0	30 (表)
OFFPAT_SET_MPLS_TTL		OpenFlow 1.1	161 (表)
OpenFlow 1.1	219 (表)	OFFPP_CONTROLLER (仮想ポート)	
OpenFlow 1.2	221 (表)	OpenFlow 1.0	30 (表)
OpenFlow 1.3	223 (表)	OpenFlow 1.1	161 (表)
OFFPAT_SET_NW_DST		OFFPP_FLOOD (仮想ポート)	
OpenFlow 1.0	48 (表), 218 (表)	OpenFlow 1.0	30 (表)
OpenFlow 1.1	219 (表)	OpenFlow 1.1	161 (表)
OFFPAT_SET_NW_ECN	219 (表)	OFFPP_IN_PORT (仮想ポート)	
OFFPAT_SET_NW_SRC		OpenFlow 1.0	30 (表)
OpenFlow 1.0	48 (表), 218 (表)	OpenFlow 1.1	161 (表)
OpenFlow 1.1	219 (表)	OFFPP_LOCAL (仮想ポート)	
OFFPAT_SET_NW_TOS		OpenFlow 1.0	30 (表)
OpenFlow 1.0	48 (表), 218 (表)	OpenFlow 1.1	161 (表)
OpenFlow 1.1	219 (表)	OFFPP_MAX	
OFFPAT_SET_NW_TTL		OpenFlow 1.0	30
OpenFlow 1.1	219 (表)	OpenFlow 1.1	161
OpenFlow 1.2	221 (表)	OFFPP_NONE (仮想ポート)	
OpenFlow 1.3	223 (表)	OpenFlow 1.0	30 (表)
OFFPAT_SET_QUEUE		OpenFlow 1.1	161 (表)
OpenFlow 1.1	220 (表)	OFFPP_NORMAL (仮想ポート)	
OpenFlow 1.2	221 (表)	OpenFlow 1.0	30 (表)
OpenFlow 1.3	224 (表)	OpenFlow 1.1	161 (表)
OFFPAT_SET_TP_DST		OFFPP_TABLE (仮想ポート)	
OpenFlow 1.0	48 (表), 218 (表)	OpenFlow 1.0	30 (表)
OpenFlow 1.1	219 (表)	OpenFlow 1.1	161 (表)
OFFPAT_SET_TP_SRC		OFFPR_INVALID_TTL	163
OpenFlow 1.0	48 (表), 218 (表)	OFFPRRFC_STALE	180
OpenFlow 1.1	219 (表)	OFFPST_FLOW	48
OFFPAT_SET_VLAN_PCP		OFFPT_BARRIER_REPLY	35 (表)
OpenFlow 1.0	48 (表), 218 (表)	OpenFlow 1.0	225 (表)
OpenFlow 1.1	219 (表)	OpenFlow 1.1	226 (表)
OFFPAT_SET_VLAN_VID		OpenFlow 1.2	227 (表)
OpenFlow 1.0	48 (表), 218 (表)	OpenFlow 1.3	228 (表)
OpenFlow 1.1	219 (表)	OFFPT_BARRIER_REQUEST	35 (表)
OFFPAT_STRIP_VLAN	48 (表), 218 (表)	OpenFlow 1.0	225 (表)
OFFPAT_VENDOR	218 (表)	OpenFlow 1.1	226 (表)
OFFPC_INVALID_TTL_TO_CONTROLLER	163	OpenFlow 1.2	227 (表)
OFFPCR_ROLE_MASTER	179	OpenFlow 1.3	228 (表)



OFPT_ECHO_REPLY .....	35 (表)	OFPT_PACKET_IN .....	35 (表)
OpenFlow 1.0 .....	225 (表)	OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)	OpenFlow 1.3 .....	228 (表)
OFPT_ECHO_REQUEST .....	35 (表)	OFPT_PACKET_OUT .....	35 (表), 50
OpenFlow 1.0 .....	225 (表)	OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)	OpenFlow 1.3 .....	228 (表)
OFPT_ERROR .....	35 (表)	OFPT_PORT_MOD .....	35 (表)
OpenFlow 1.0 .....	225 (表)	OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)	OpenFlow 1.3 .....	228 (表)
OFPT_EXPERIMENTER .....		OFPT_PORT_STATUS .....	35 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.0 .....	225 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.1 .....	226 (表)
OpenFlow 1.3 .....	228 (表)	OpenFlow 1.2 .....	227 (表)
OFPT_FEATURES_REPLY .....	35 (表)	OpenFlow 1.3 .....	228 (表)
OpenFlow 1.0 .....	225 (表)	OFPT_QUEUE_GET_CONFIG_REPLY .....	35 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.0 .....	225 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.1 .....	226 (表)
OpenFlow 1.3 .....	228 (表)	OpenFlow 1.2 .....	227 (表)
OFPT_FEATURES_REQUEST .....	35 (表)	OpenFlow 1.3 .....	228 (表)
OpenFlow 1.0 .....	225 (表)	OFPT_QUEUE_GET_CONFIG_REQUEST .....	35 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.0 .....	225 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.1 .....	226 (表)
OpenFlow 1.3 .....	228 (表)	OpenFlow 1.2 .....	227 (表)
OFPT_FLOW_MOD .....	35 (表)	OpenFlow 1.3 .....	228 (表)
OpenFlow 1.0 .....	225 (表)	OFPT_ROLE_REPLY .....	179
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.2 .....	227 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.3 .....	228 (表)
OpenFlow 1.3 .....	228 (表)	OFPT_ROLE_REQUEST .....	179
OFPT_FLOW_REMOVED .....	35 (表)	OpenFlow 1.2 .....	227 (表)
OpenFlow 1.0 .....	225 (表)	OpenFlow 1.3 .....	228 (表)
OpenFlow 1.1 .....	226 (表)	OFPT_SET_ASYNC .....	228 (表)
OpenFlow 1.2 .....	227 (表)	OFPT_SET_CONFIG .....	35 (表)
OpenFlow 1.3 .....	228 (表)	OpenFlow 1.0 .....	225 (表)
OFPT_GET_ASYNC_REPLY .....	228 (表)	OpenFlow 1.1 .....	226 (表)
OFPT_GET_ASYNC_REQUEST .....	228 (表)	OpenFlow 1.2 .....	227 (表)
OFPT_GET_CONFIG_REPLY .....	35 (表)	OpenFlow 1.3 .....	228 (表)
OpenFlow 1.0 .....	225 (表)	OFPT_STATS_REPLY .....	35 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.0 .....	225 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.1 .....	226 (表)
OpenFlow 1.3 .....	228 (表)	OpenFlow 1.2 .....	227 (表)
OFPT_GET_CONFIG_REQUEST .....	35 (表)	OFPT_STATS_REQUEST .....	35 (表)
OpenFlow 1.0 .....	225 (表)	OpenFlow 1.0 .....	225 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)	OFPT_TABLE_MOD .....	
OFPT_GROUP_MOD .....		OpenFlow 1.1 .....	226 (表)
OpenFlow 1.1 .....	226 (表)	OpenFlow 1.2 .....	227 (表)
OpenFlow 1.2 .....	227 (表)	OpenFlow 1.3 .....	228 (表)
OpenFlow 1.3 .....	228 (表)	OFPT_VENDOR .....	35 (表), 225 (表)
OFPT_HELLO .....	35 (表)	OFPTC_TABLE_MISS_CONTINUE .....	147
OpenFlow 1.0 .....	225 (表)	OFPXMC_EXPERIMENTER .....	170 (表)
OpenFlow 1.1 .....	226 (表)	OFPXMC_NXM_0 .....	170
OpenFlow 1.2 .....	227 (表)	OFPXMC_NXM_1 .....	170
OpenFlow 1.3 .....	228 (表)	OFPXMC_OPENFLOW_BASIC .....	170 (表), 172
OFPT_METER_MOD .....	228 (表)	OFPXMT_OFF_MPLS_BOS .....	
OFPT_MULTIPART_REPLY .....	228 (表)	OpenFlow 1.3 .....	190 (表)
OFPT_MULTIPART_REQUEST .....	228 (表)	ONF .....	3
		Open Networking Foundation .....	3

OpenFlow	
L2 ネットワーク構成例	5 (図)
概要	5
起源	2
仕様	2
仕様の変遷	19
動作例	15
歴史	2
OpenFlow 1.0	21
OpenFlow 1.1	141
OpenFlow 1.2	167
OpenFlow 1.3	183
OpenFlow 1.3.1	192
OpenFlow 1.3.2	193
OpenFlow Configuration Point	216
OpenFlow eXtensible Match	169
OpenFlow Lab	213
OpenFlow Switch Errata 1.0.1	58
OpenFlow コントローラ	7
冗長化	178
制御可能なスイッチの数	202
負荷増大	196
負荷分散	178
OpenFlow スイッチ	7
数の制限	202
初期動作	12
設定順	204
OpenFlow スイッチコンソーシアム	2
OpenFlow チャンネル	10, 165, セキュアチャンネルも見よ
逆向きに確立	193
再接続タイマー	203
遅延	198
複数の	178
OpenFlow ハイブリッドスイッチ	164
OpenFlow ヘッド	34
OpenFlow メッセージ	34
OpenFlow 1.0	225
OpenFlow 1.1	226
OpenFlow 1.2	227
OpenFlow 1.3	228
Organizationally Specific TLVs (LLDPDU)	68 (表)
OSPF	72
OUI (製造者識別子)	65
out_port (Flow-Mod)	47 (表)
Output (アクション)	142, Forward も見よ
OpenFlow 1.1	219 (表)
OpenFlow 1.2	221 (表)
OpenFlow 1.3	223 (表)
output (アクションセットの実行順)	146 (表)
OXM	168, 169
OpenFlow 1.3	190 (表)
OpenFlow 1.3 で必須な	200 (表)
依存関係	172, 200
oxm_class	170 (表)
OXM_ETH_TYPE	
OpenFlow 1.2	172 (表)
OpenFlow 1.3	200 (表)
oxm_field	170 (表)
oxm_hasmask	170 (表)
oxm_length	170 (表)
OXM_OF_ARP_OP	173 (表)
OXM_OF_ARP_SHA	173 (表)
OXM_OF_ARP_SPA	173 (表)
OXM_OF_ARP_THA	174 (表)
OXM_OF_ARP_TPA	173 (表)
OXM_OF_ETH_DST	
OpenFlow 1.2	172 (表)
OpenFlow 1.3	200 (表)
OXM_OF_ETH_SRC	
OpenFlow 1.2	172 (表)
OpenFlow 1.3	200 (表)
OXM_OF_ICMPV4_CODE	173 (表)
OXM_OF_ICMPV4_TYPE	173 (表)
OXM_OF_ICMPV6_CODE	174 (表)
OXM_OF_ICMPV6_TYPE	174 (表)
OXM_OF_IN_PHY_PORT	172 (表)
OXM_OF_IN_PORT	
OpenFlow 1.2	172 (表)
OpenFlow 1.3	200 (表)
OXM_OF_IP_DSCP	173 (表)
OXM_OF_IP_ECN	173 (表)
OXM_OF_IP_PROTO	
OpenFlow 1.2	173 (表)
OpenFlow 1.3	200 (表)
OXM_OF_IPV4_DST	
OpenFlow 1.2	173 (表)
OpenFlow 1.3	200 (表)
OXM_OF_IPV4_SRC	
OpenFlow 1.2	173 (表)
OpenFlow 1.3	200 (表)
OXM_OF_IPV6_DST	
OpenFlow 1.2	174 (表)
OpenFlow 1.3	200 (表)
OXM_OF_IPV6_EXTHDR	189, 190 (表)
OXM_OF_IPV6_FLABEL	174 (表)
OXM_OF_IPV6_ND_SLL	174 (表)
OXM_OF_IPV6_ND_TARGET	174 (表)
OXM_OF_IPV6_ND_TLL	174 (表)
OXM_OF_IPV6_SRC	
OpenFlow 1.2	174 (表)
OpenFlow 1.3	200 (表)
OXM_OF_METADATA	172 (表)
OXM_OF_MPLS_LABEL	174 (表)
OXM_OF_MPLS_TC	174 (表)
OXM_OF_PBB_ISID	
OpenFlow 1.3	190 (表)
OXM_OF_SCTP_DST	173 (表)
OXM_OF_SCTP_SRC	173 (表)
OXM_OF_TCP_DST	
OpenFlow 1.2	173 (表)
OpenFlow 1.3	200 (表)
OXM_OF_TCP_SRC	
OpenFlow 1.2	173 (表)
OpenFlow 1.3	200 (表)
OXM_OF_TUNNEL_ID	
OpenFlow 1.3	190 (表)
OXM_OF_UDP_DST	
OpenFlow 1.2	173 (表)
OpenFlow 1.3	200 (表)
OXM_OF_UDP_SRC	
OpenFlow 1.2	173 (表)
OpenFlow 1.3	200 (表)
OXM_OF_VLAN_PCP	173 (表)
OXM_OF_VLAN_VID	172 (表)
OXM マッチフィールド	172-174 (表)

<b>P</b>	
Packet-In メッセージ	13, 35(表), 49
OpenFlow 1.0	225(表)
OpenFlow 1.1	226(表)
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
設定例	72, 83, 95
大量の	196
Packet-Out メッセージ	35(表), 50
OpenFlow 1.0	225(表)
OpenFlow 1.1	226(表)
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
失敗検出	208
設定例	70, 84
フローエントリに依存する	205
PBB I-SID	190
PBB (Provider Backbone Bridging)	190
PCP (Priority Code Point)	143(表)
PlanetLab	2
PlugFest	212
pop (アクションセットの実行順)	146(表)
Port Description (LLDPDU)	68(表)
Port ID (LLDPDU)	67, 68(表)
Port-Status メッセージ	35(表)
OpenFlow 1.0	225(表)
OpenFlow 1.1	226(表)
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
PORT_MOD メッセージ	35(表)
OpenFlow 1.0	225(表)
OpenFlow 1.1	226(表)
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
Pre-requisite	172
Priority	→ 優先度
priority (Flow-Mod)	47(表)
Push VLAN header	164
push (アクションセットの実行順)	146(表)
Push-Tag/Pop-Tag (アクション)	153
OpenFlow 1.1	220(表)
OpenFlow 1.2	221(表)
OpenFlow 1.3	223(表)
<b>Q</b>	
QinQ	164
qos (アクションセットの実行順)	146(表)
QoS (メーターテーブルによる)	185
Queue	31
QUEUE_GET_CONFIG 応答メッセージ	35(表), 41
OpenFlow 1.0	225(表)
OpenFlow 1.1	226(表)
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
QUEUE_GET_CONFIG 要求メッセージ	35, 41
OpenFlow 1.0	225(表)
OpenFlow 1.1	226(表)
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
<b>R</b>	
reason (Packet-In)	50(表)
RED (Random Early Detection)	165
RFC 2460 (IPv6)	33
RFC 2474 (DiffServ)	186
RFC 2474 (DS field)	33
RFC 2863 (MIB)	68(表)
RFC 3168 (ECN)	165
RFC 3260 (DS field)	33
RFC 3418 (MIB)	68(表)
RFC 3882 (セキュリティ)	134
RFC 4960 (SCTP)	164
RFC 5405 (UDPユニキャスト)	189
RFC 5635 (セキュリティ)	134
RFC 5798 (VRRP)	112
RFC 6020 (YANG)	216
RFC 6241 (NETCONF)	216
RFC 6346 (Stateless A+P)	200
RFC 6347 (DTLS)	189
Role	178
Role 応答メッセージ	
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
Role 要求メッセージ	179
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
Router (IPv6 拡張ヘッダ)	189
<b>S</b>	
SBP (Shortest Path Bridging)	125
SCTP	143, 164
SDN	3, 20, 109
select (グループタイプ)	128, 154(表), 156
Set VLAN ID	
CoS	33
OpenFlow 1.0	32(表)
OpenFlow 1.1	164
Set VLAN priority	32(表)
set (アクションセットの実行順)	146(表)
Set-Field	177
Set-Field (アクション)	153
OpenFlow 1.1	219(表)
OpenFlow 1.2	221(表)
OpenFlow 1.3	223(表)
Set-Queue (アクション)	
OpenFlow 1.1	220(表)
OpenFlow 1.2	221(表)
OpenFlow 1.3	224(表)
SET_ASYNC メッセージ	228(表)
SET_CONFIG メッセージ	35(表), 40(図)
OFPC_INVALID_TTL_TO_CONTROLLER	163
OpenFlow 1.0	225(表)
OpenFlow 1.1	226(表)
OpenFlow 1.2	227(表)
OpenFlow 1.3	228(表)
ShowNet (Interop Tokyo 2009)	115
SIGCOMM (OpenFlow 論文)	4
SLAVE (コントローラの Role)	178
Southbound API	215
Spanning Tree Protocol	25
Stateless A+P	200
STATS 応答メッセージ	35(表)
OpenFlow 1.0	225(表)
OpenFlow 1.1	226(表)
OpenFlow 1.2	227(表)
STATS 要求メッセージ	35(表), 41
OpenFlow 1.0	225(表)

OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
STP (Spanning Tree Protocol) .....	25, 125
STRICT (コマンド名) .....	42
Strip VLAN header	
OpenFlow 1.0 .....	32 (表)
廃止 .....	164
System Capabilities (LLDPDU) .....	68 (表)
System Description (LLDPDU) .....	68 (表)
System Name (LLDPDU) .....	68 (表)

## T

TABLE	
Forward アクション .....	29 (表)
OpenFlow 1.3 .....	181 (表)
仮想ポート .....	30
予約された仮想ポート .....	160
TABLE_MOD メッセージ	
OpenFlow 1.1 .....	226 (表)
OpenFlow 1.2 .....	227 (表)
OpenFlow 1.3 .....	228 (表)
TCAM	
TCP/UDP/SCTP destination port .....	143 (表)
TCP/UDP/SCTP source port .....	143 (表)
TCPセッション .....	56
TCP 平文	
OpenFlow チャンネル .....	165
セキュアチャンネル .....	34
TCP ヘッダ (OpenFlow で扱えないフィールド)	
Testing-Interop Working Group .....	212
Timeout (Hard) .....	→ hard_timeout
Timeout (Idle) .....	→ idle_timeout
TLS	
OpenFlow チャンネル .....	165
セキュアチャンネル .....	34
TLV	
LLDPDU .....	67
OXM .....	169
ToS	
ヘッダフィールド .....	24 (表)
マッチフィールド .....	143 (表)
ToS フィールド	
TPMR .....	66 (表)
Traffic Class .....	33, 178
Transport destination port .....	24 (表)
Transport source port .....	24 (表)
TRILL .....	125
TTL (LLDPDU) .....	67, 68 (表)
TTL 値 (不正な) .....	163
TTL フィールド	
0 のパケットの処理 .....	139
OpenFlow による操作 .....	161
に対するマッチ .....	163
U	
UDP .....	189
V	
Vendor (アクション) .....	218 (表)
Vendor メッセージ .....	35 (表), 41, 166, 225 (表)
version (OpenFlow ヘッダ) .....	35 (表)
VID (VLAN ID) .....	24 (表)
VLAN ID	
仮想ネットワークの指定 .....	116

ヘッダフィールド .....	24 (表)
マッチフィールド .....	143 (表)
VLAN Priority	
ヘッダフィールド .....	24 (表)
マッチフィールド .....	143 (表)
VLAN タグ	
OpenFlow 1.0 .....	101
OpenFlow 1.1 .....	164
VMware .....	2, 109
VRF (Virtual Routing and Forwarding) .....	114
VRPP .....	112

## W

watch_port .....	159
Web3 層アーキテクチャ .....	114
WebRTC .....	165

## X

xid (OpenFlow ヘッダ) .....	35 (表)
--------------------------	--------

## Y

YANG .....	216
------------	-----

## ア

アクション .....	152, 230
OpenFlow 1.0 .....	22, 29
OpenFlow 1.1 .....	153
例 .....	22
アクションセット .....	145, 146, 152, 229
アクションリスト .....	152, 229
アクセスリスト (ハードウェア処理) .....	4
宛先ポート番号 .....	24 (表), 143 (表), 200
上書き .....	32 (表)
マスク .....	200
アライメント .....	169
安定	
製品の .....	214

## イ

イーサネットアドレス .....	65
イーサネットファブリック .....	125
イコールコストマルチパス .....	→ ECMP
インストラクション .....	151, 152, 229
インターオペラビリティ .....	19
プラグフェスト .....	212
インターネット	
OpenFlow と接続 .....	5, 161
次世代 .....	2

## オ

オーケストレータ .....	115
オーバーレイ方式 .....	8
同じ IPv4 アドレス (がないか確認する) .....	112

## カ

カウンタ .....	22, 27
仮想化 .....	109
仮想スイッチ .....	9, 110, 116, 216
イーサネットアドレステーブル更新 .....	112
仮想ポート .....	181 (表)
OpenFlow 1.0 .....	30
OpenFlow 1.1 .....	160
廃止 .....	181
仮想マシン .....	110
仮想ルータ .....	110, 134

<b>キ</b>	
キャンパスネットワーク	2
緊急事態フローキャッシュ	47(表)
OpenFlow 1.0.1	58
廃止	166
<b>ク</b>	
クッキー	
OpenFlow 1.0	43
OpenFlow 1.3	192
グループ	
チェーン	159
ポートミラーリングに利用	132
グループテーブル	230
<b>ケ</b>	
経路コスト	126
ケーパビリティ	39(表)
<b>コ</b>	
コスト (経路の)	126
コントロールプレーン	6
構築手法	7
帯域不足	197
<b>サ</b>	
サーバ仮想化	110
最新版	192
<b>シ</b>	
実験的な拡張	182
実験ネットワーク環境	2
実装依存	201
仕様	
OpenFlow	19
OpenFlow 0.2.0	2
最新	3
実装との差	199
周辺の議論	211
状態遷移 (フローマッチ)	
OpenFlow 1.0	25
OpenFlow 1.1	147
OpenFlow 1.3	187
冗長性確保 (コントローラの)	178
自律分散	6
<b>ス</b>	
スタンフォード大学	2, 212
<b>セ</b>	
セキュアチャネル	10, 34
TCPポート番号	34
確立手順	36
障害発生時	58
<b>ソ</b>	
相互運用性	→ インターオペラビリティ
相互接続性	
Testing-Interop Working Group	212
ネットワークポロジ検出	62
送信元ポート番号	24(表), 143(表), 200
上書き	32(表)
マスク	200
<b>タ</b>	
ダイクストラ法	72
タイムアウト	
OpenFlow 1.3	192
フローエントリの削除	42
<b>チ</b>	
中央集権モデル	7
<b>テ</b>	
ディスカッショングループ	3
データセンター	110
データプレーン	6
構築手法	8
テーブルミス	25
Packet-In メッセージ	49
デフォルト設定	26
テーブルミスフローエントリ	187
テストツール (OpenFlowの)	212
<b>ト</b>	
トポロジ	→ ネットワークトポロジ
トポロジ検出	13
トランッキング	128
<b>ニ</b>	
日本	3
認定プログラム	214
<b>ネ</b>	
ネゴシエーション	192
ネットワーク仮想化	114
オーバーレイ型	114
ネットワークポロジ	
Hop-by-Hop 方式	8
オーバーレイ方式	8
ネットワークポロジ検出	62
ネットワークバイトオーダー	169, 171
<b>ハ</b>	
バージョン	
OpenFlow 1.0	21
OpenFlow 1.1	141
OpenFlow 1.2	167
OpenFlow 1.3	183
OpenFlow 1.3.1	192
OpenFlow 1.3.2	193
最新版	192
ネゴシエーション	192
バーチャルアプライアンス	114
バーチャルマシン	2
ハードウェア	4
バイトオーダー	171
ハイパーバイザー	110
パイプライン処理	145
ハイブリッドスイッチ	164
ハイブリッド方式	8, 10
パケットマッチ情報	43
ハンドシェイク	37
～後のメッセージ	41

ヒ	
ビッグエンディアン .....	171
標準ポート .....	181 (表)
平文接続 .....	34

フ	
ファイアウォール .....	114
プール化 .....	114
フェイルオーバー .....	158
OpenFlow コントローラの .....	178
輻輳制御機構 (TCP) .....	165
物理ポート .....	181 (表)
プラグフェスト .....	212
フラグメンテーション .....	209
プロアクティブ型 .....	13
タグド VLAN の例 .....	103, 106
ラーニングブリッジの例 .....	86
リピータ HUB の例 .....	80
フロー .....	5
フローエントリ .....	12, 22
OpenFlow 1.0 .....	22
OpenFlow 1.1 .....	151
OpenFlow 1.3 の構成要素 .....	192
削除 .....	42
設定可能な上限 .....	201
追加 .....	42
複数にマッチ .....	144
フローテーブル .....	12, 22
こうなったらこうしなさい .....	11
サポート数の確認 .....	25
複数の .....	23
フローテーブルエントリ .....	→ フローエントリ
フローマッチフィールド .....	169
フローラベル (IPv6) .....	178

へ	
ヘッダフィールド .....	24
ヘッダフィールド解析 .....	26 (図), マッチフィールド解析も見よ
ベンダー .....	
コンソーシアム設立時に参加 .....	2
力関係 .....	214

ホ	
ポート番号 .....	→ 送信元ポート番号, → 宛先ポート番号
ポート番号 (スイッチの) .....	
OpenFlow 1.0 での識別子 .....	30
OpenFlow 1.0 の最大値 .....	30
OpenFlow 1.1 での識別子 .....	160
OpenFlow 1.1 の最大値 .....	161
ポートミラーリング .....	130
補助的な接続 .....	188

マ	
マスク (OXM マッチフィールド) .....	200
マッチ .....	
OpenFlow 1.0 .....	25
TCAM .....	4
TTL の .....	163
完全に一致させる .....	45
できるもの、できないもの .....	199
何でも .....	46
何でも (OXM) .....	174
フローの .....	22

マッチフィールド .....	143
OpenFlow 1.1 .....	143 (表)
依存関係 .....	172, 200
マッチフィールド解析 .....	148
廃止 .....	172
マルチテーブル .....	144
マルチテナント .....	112, 113
マルチパートフレームワーク .....	191
マルチパス .....	125
マルチレイヤスイッチ (OpenFlow による実現例) .....	134

ミ	
未知のパケット .....	13, → テーブルミス
スイッチのデフォルト挙動 .....	187

メ	
メーターエントリ .....	186
メーターテーブル .....	185, 230
メーターバンド .....	186
メタデータ .....	143
メッセージが遅延 .....	198

ユ	
優先度 .....	25
OpenFlow 1.3 .....	192

ヨ	
予約された仮想ポート .....	160, 181 (表)
予約ポート .....	181, 181 (表)

ラ	
ライブマイグレーション .....	111
OpenFlow での実現例 .....	115
ラウンドロビン方式 .....	128

リ	
リアクティブ型 .....	13
リダイレクト .....	133
リトルエンディアン .....	171
リピータ HUB (OpenFlow で実現) .....	79
リンクアグリゲーション .....	160

ル	
ルータ .....	
OpenFlow による実現例 .....	134
コントロールプレーンとデータプレーン .....	6
ループバックインターフェース .....	160

レ	
歴史 (OpenFlow の) .....	2

ロ	
ロードバランサ (OpenFlow で実現) .....	129
ロードバランシング .....	
OpenFlow コントローラの .....	178
テストケース .....	213
ロジカルポート .....	181, 181 (表)

ワ	
ワイルドカード .....	44 (表), 45
ワイルドカードフィールド .....	45

## 〈著者略歴〉

あきみち

慶應義塾大学政策メディア研究科にて博士を取得。  
ソニー株式会社において、ホームネットワークにおける通信技術開発に従事したのち、  
2007年にソニーを退職し、現在はプログラマー (Geek なページ、<http://www.geekpage.jp/>) として活動を行っている。  
アルファプログラマーアワード 2011 受賞。全日本剣道連盟情報小委員会委員。  
著書に『インターネットのカタチ - 脆さが織り成す粘り強い世界』(オーム社) がある。

宮永直樹 (みやなが なおき)

1994年4月日本電気(株)入社。  
企業向けネットワークの販売・設計・構築に従事。  
2001年セキュリティ運用サービスの立ち上げ。  
2002年から2006年まで Interop Tokyo NOC チームにて ShowNet のセキュリティを担当。  
データセンターネットワークを中心とした要件定義や設計を担当しつつ、現在、OpenFlow 製品の販売・ソリューション企画に従事。

岩田 淳 (いわた あつし)

1988年 東京大学工学部電気工学科卒業。  
1990年 同大学大学院修士課程修了。同年、NEC 入社。  
以来、ATM 網、IP 網、MPLS 網、コンテンツ配信網、広域イーサネット網、メトロ基幹網、次世代インターネット (OpenFlow/SDN) におけるネットワーク制御ならびにシステム研究開発、ならびにサーバシステムにおけるノード仮想化、I/O 仮想化の研究開発に従事。  
University of California, Los Angeles (UCLA)、NEC IP ネットワーク事業部、クラウドシステム研究所を経て、2013年4月に情報・ナレッジ研究所に異動。それぞれの研究所にて OpenFlow/SDN 研究開発全体統括、ならびに同研究所の所長代理業務に従事。  
受賞歴として、電子情報通信学会交換システム研究会研究賞、同学会ネットワークシステム研究会研究賞、フジサンケイ ビジネスアイ第26回先端技術大賞 産経新聞社賞受賞、科学技術と経済の会第一回技術経営・イノベーション賞。  
電子情報通信学会東京支部会計幹事。電子情報通信学会会員。  
著書に『広域イーサネット技術概論』(電子情報通信学会) がある。

- 本書の内容に関する質問は、オーム社開発部「マスタリングTCP/IP OpenFlow 編」係宛、E-mail ([kaihatu@ohmsha.co.jp](mailto:kaihatu@ohmsha.co.jp)) または書状、FAX (03-3293-2825) にてお願いします。お受けできる質問は本書で紹介した内容に限らせていただきます。なお、電話での質問にはお答えできませんので、あらかじめご了承ください。

## マスタリングTCP/IP OpenFlow 編

平成25年7月25日 第1版第1刷 発行

著 者 あきみち・宮永直樹・岩田淳  
企画編集 オーム社 開発局  
発行者 竹生修己  
発行所 株式会社 オーム社  
郵便番号 101-8460  
東京都千代田区神田錦町3-1  
電 話 03 (3233) 0641 (代表)  
URL <http://www.ohmsha.co.jp/>

©あきみち・宮永直樹・岩田淳 2013

装丁デザイン 花本浩一

ISBN 978-4-274-06920-8

