

# SDN最新動向と活用事例

中島佳宏

NTT未来ネット研究所

# 自己紹介

## 中島佳宏

- NTT未来ねっと研究所

- バックグラウンド

- 大学時代

- High-performance computing
- Grid Computing

- 入社後

- 非圧縮映像IP NW伝送装置
- アプリケーション特化のNW制御システム

- 現在

- 高速なソフトウェアスイッチ開発
- Whitebox向けスイッチOSの開発

### **3. SDN最新動向**

## 3.1 標準化動向

ネットワークノードをソフトウェアから制御可能にするための  
データモデル・API・プロトコルの検討

# SDN関連標準化団体と領域

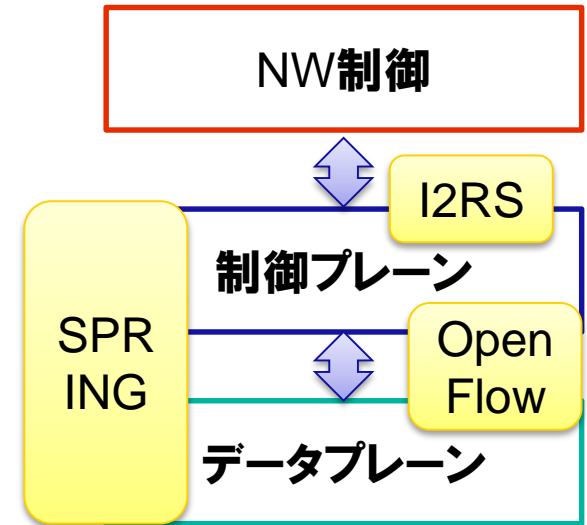
## プロトコル・フレームフォーマット主体の標準化

- IETF
  - I2RS (Interface to the Routing System)
  - SPRING (Source Packet Routing in Networking)
- Open Networking Foundation (ONF)

## アーキテクチャ・情報モデル主体の標準化

- Optical Internetworking Forum (OIF)
- ITU-T SG13

## コントロールプレーンにおける標準化領域マップ



## データプレーンにおける標準化領域

対象	IETF	ONF	OIF	ITU-T
L3				
L2.5	SPRING			アーキテクチャ
L2		Open Flow		
L0 – L1			アーキテクチャ	

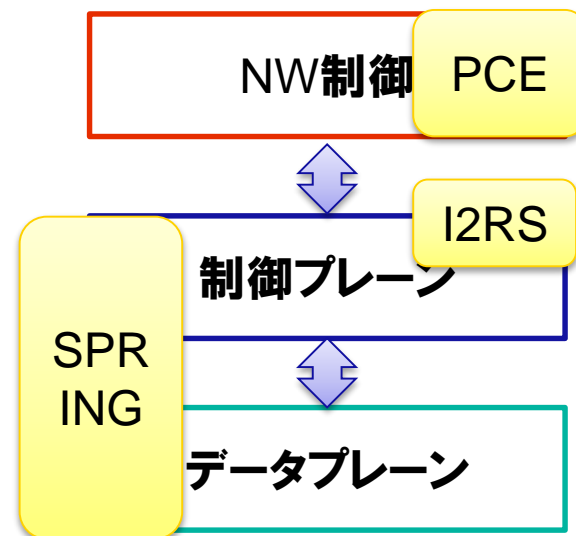
  Protocol系

  アーキテクチャ系

## インターネット技術の標準化推進の組織

## IPやMPLSプロトコルを拡張して SDNを実現する関連プロトコルの標準化

- I2RS (Interface to the Routing System)
  - NW制御システムからルータの内部ルーティング情報の取得や設定のAPIを提供
- SPRING (Source Packet Routing in Networking)
  - ネットワークをセグメントとして扱い、集中型と分散型のハイブリッド型NW
- PCE (Path Computation Element)
  - ネットワークノードからパス計算部分を外部に切り出し集中管理としてNWパス計算を実施



# I2RS (Interface to the Routing System)

## ネットワークエレメントと連携するための標準的なデータモデルの規定

- ルータを対象

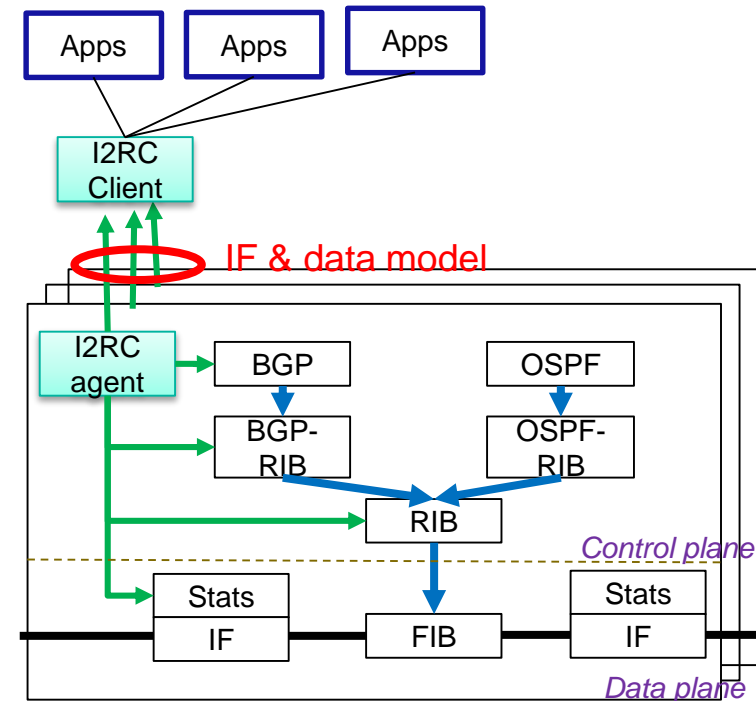
## NEのRIB情報へのIFを提供

- アプリケーションロジックを上位で扱えるようにした抽象化されたルーティング情報
- トポロジデータの収集とエクスポート
- イベント通知
- ステート情報のRIBへプッシュ

## コントロールプレーン (BGP・OSPF)はそのまま

## 参考資料

- [I-D.ietf-i2rs-architecture]
- [I-D.ietf-i2rs-problem-statement]



# SPRING (Source Routing, Segment Routing) 1/2

## NWの集中型と分散型の良い点を利用するハイブリッドのNW制御方式

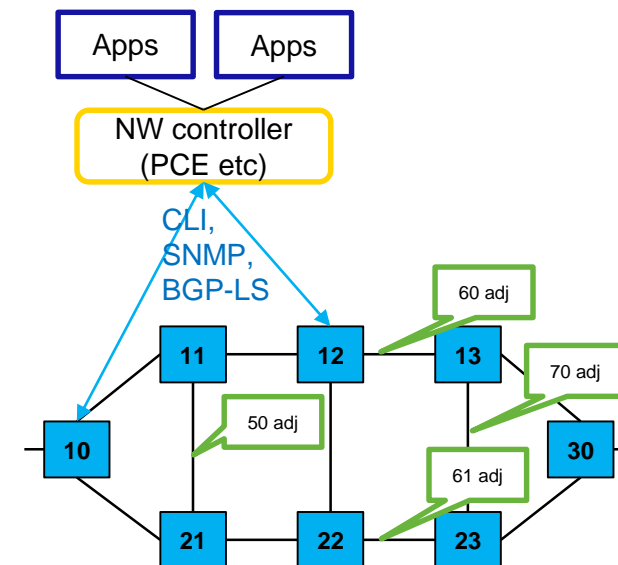
- ネットワークをセグメントで表現 (NodeとAdjacency)
- 集中制御: 可視化・自動化・TE
- 自立分散: 経路計算、障害検知、OAM、迂回
- MPLSもしくはIPv6によるデータプレーン実装

## メリット

- Control Planeのシンプル化
  - IGPでsegment idを配布, 複雑なLDP, RSVPを排除
- 柔軟かつScalableなTE
  - ステートレスかつ, 帯域や遅延を加味したCSPF
- Transport的なパス運用にも適応可能
  - Explicitパス, OAM, BFD, Protectionとの連携

## 参考資料

- [I-D.filsfils-spring-segment-routing],  
[I-D.filsfils-spring-segment-routing-use-cases],  
[I-D.filsfils-spring-segment-routing-mpls].



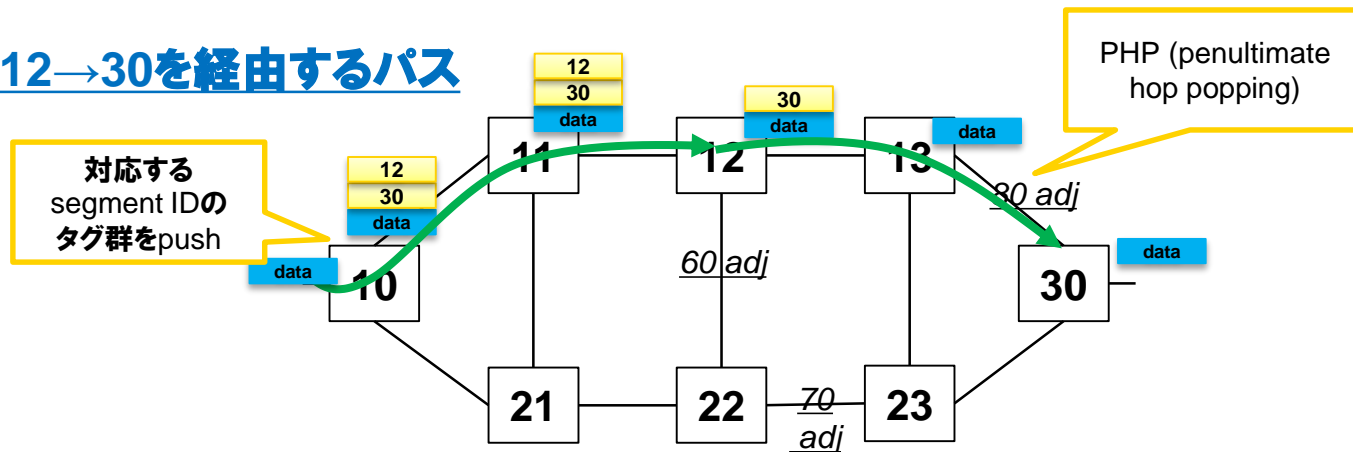


# SPRING (Source Routing, Segment Routing) 2/2

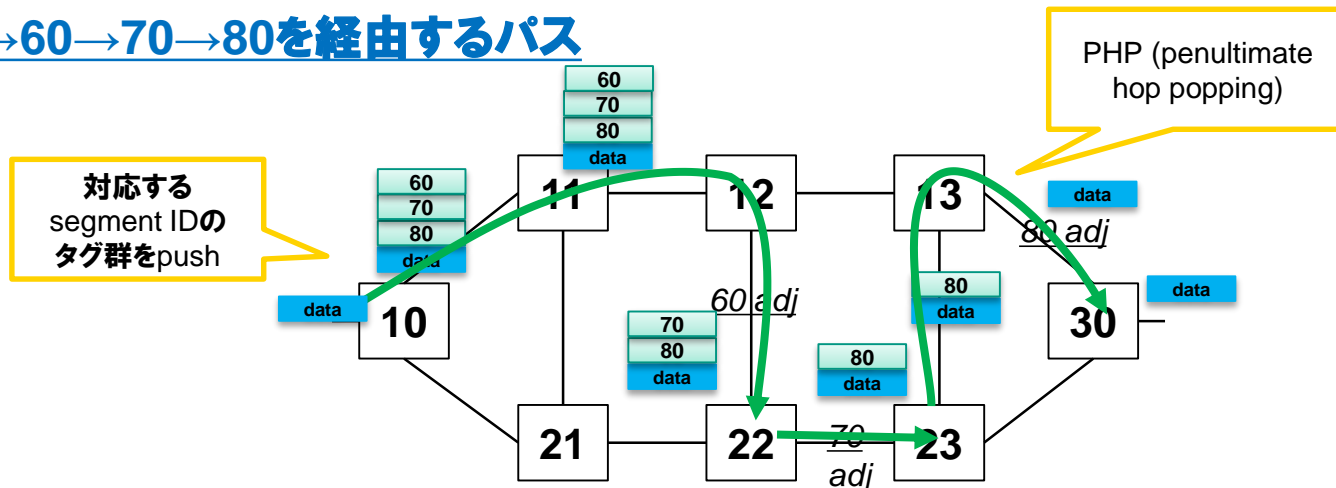
## MPLS実装の場合

- ingressエッジルータで通過するsegment idを表現するMPLS tagをすべてpush
- 各ルータではCSPFに従いパケット転送

### 10→12→30を経由するパス



### 10→60→70→80を経由するパス

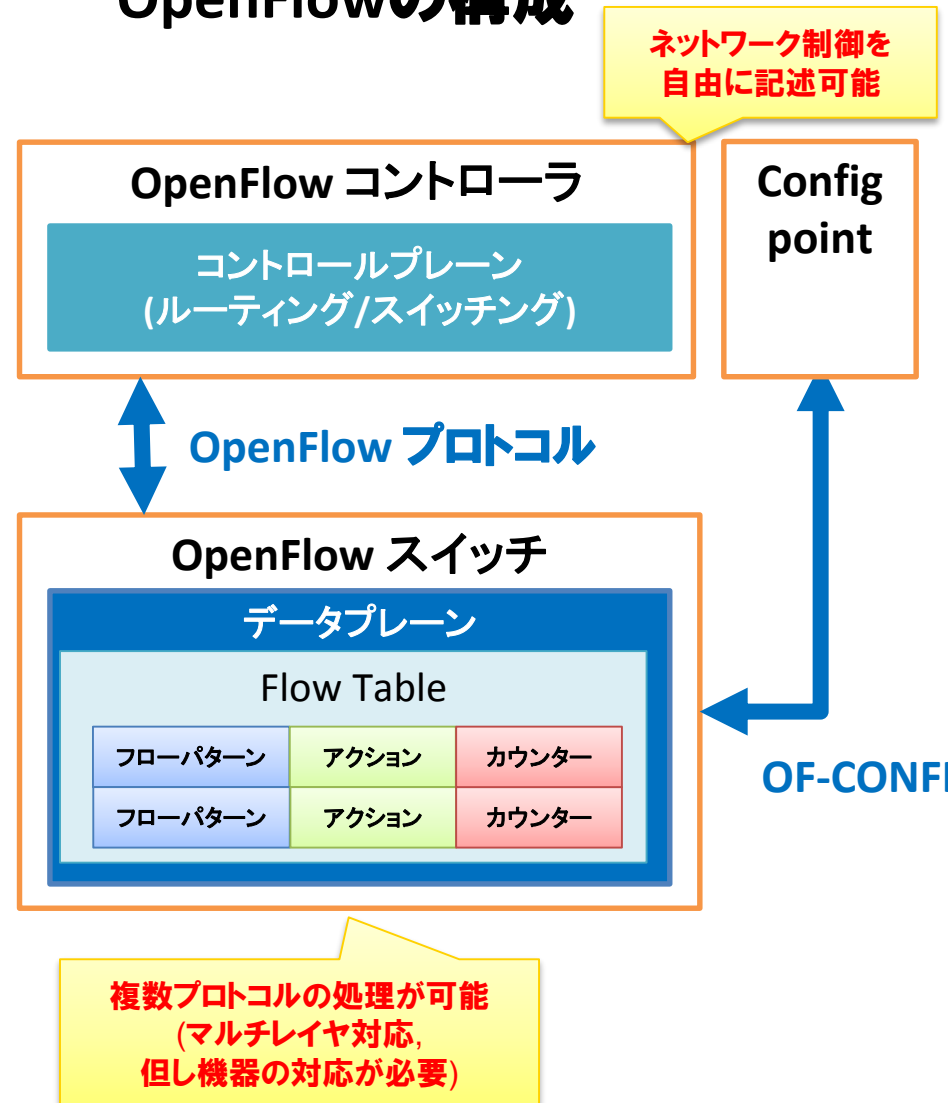


# Open Networking Foundation

## OpenFlowを中心としたオープンな仕様のSDN関連プロトコルの標準化団体

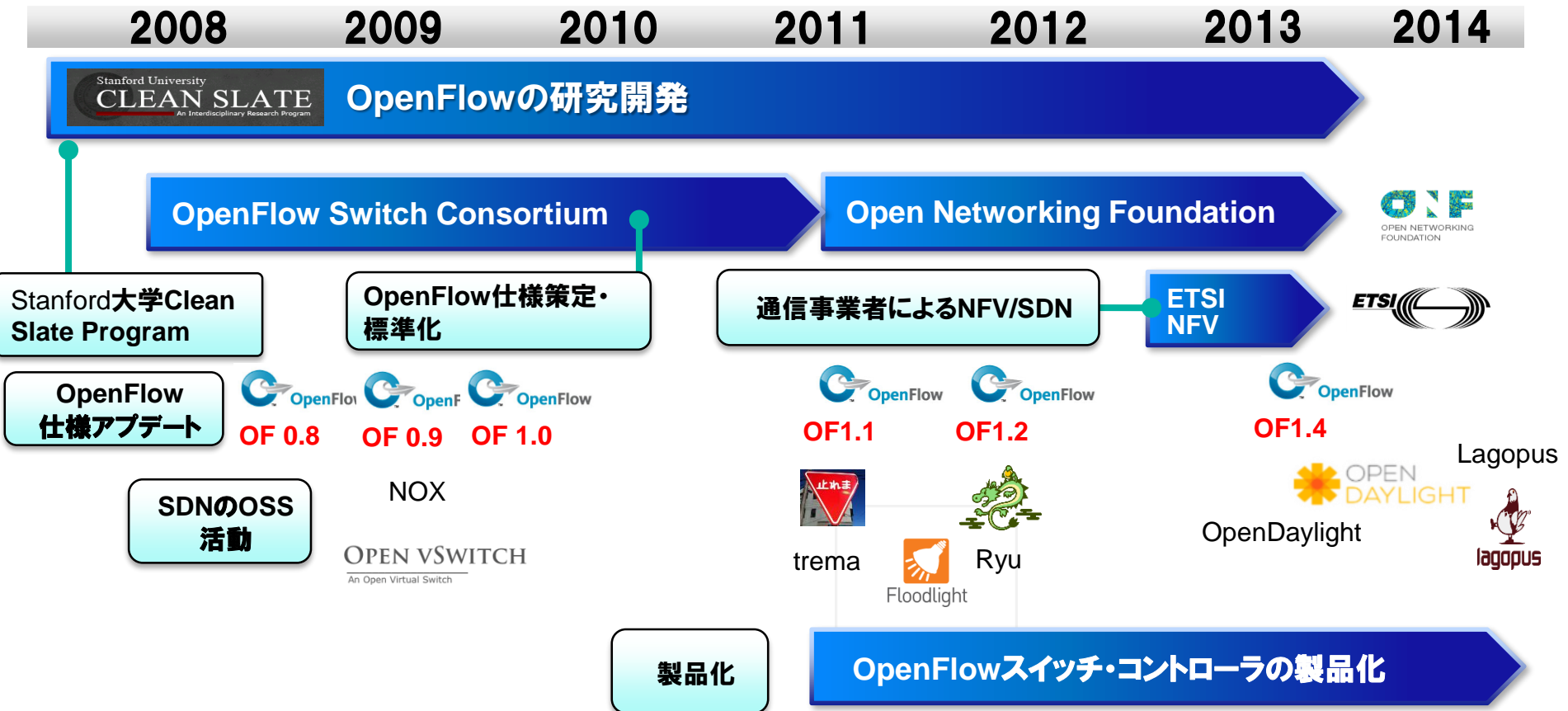
- 2011年3月に設立、150社が加盟 (2014年5月時点)
- OpenFlowの標準スペックをはじめSDNに関わる仕様の策定(最新スペックは1.4, 安定版は1.3)
  - OpenFlow switch specification
    - switchとcontroller間のフロー制御命令
      - » Ethernet, VLAN, IPv4, IPv6, ARP, ICMP, L4 port
      - » MPLS, PBB, QinQ
  - OF-CONFIG
    - NETCONFをベースにしたOpenFlow switchの初期設定プロトコル
    - OpenFlowコントローラやブリッジ、トンネルポイントの設定

## OpenFlowの構成

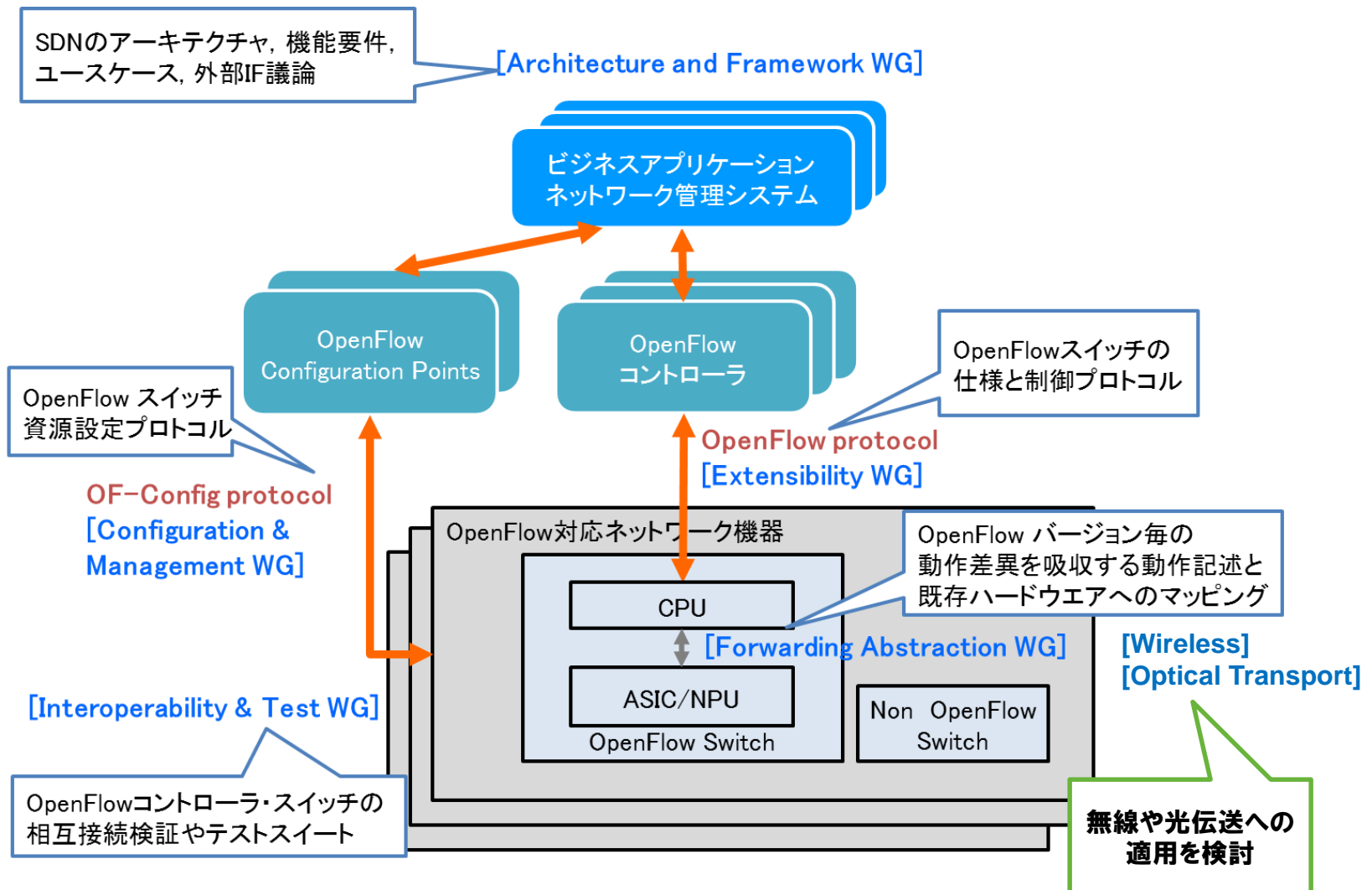


# ONFの標準化の進展

Stanford大のClean Slate ProgramからOpenFlowの開発研究  
 標準化団体ONFが発足しSDN/OpenFlowの標準化をリード



# ONFのワーキンググループとプロトコルの標準化



# OpenFlowのバージョンの進展 (1/2)

## OpenFlow 1.0

- キャンパスネットワークやデータセンタ内のネットワークを意識したプロトコルをサポート
  - Ethernet, VLAN, IPv4, TCP, UDP等

## OpenFlow 1.1

- 広域ネットワーク**で活用されることを想定し機能的な拡張
- データセンタ間, キャリア網**を意識した通信プロトコル対応
- パケットパイプライン処理を実現するための**複数フローテーブル**の導入

## OpenFlow 1.2

- プロトコルの拡張性を意識したパラメータ記述の**拡張マッチング記述方式**の採用
- コントローラのフェイルオーバーを可能にすべく**複数コントローラ**へ対応

## OpenFlow 1.3

- ONFの現在の安定版仕様**
- 制御プレーンのIPv6環境での運用に対応
- トンネリングプロトコル, リンクアグリゲーション**の論理ポートに対応
- フロー毎の帯域制御を実現する**Quality of Service制御**へ対応 (Meter)

# OpenFlowのバージョンの進展 (2/2)

## OpenFlow 1.4

- **光伝送装置制御のための拡張**
- **複数コントローラ対応のためのフロー関連の通知機能・ロールの機能**
- **テーブルの同期機構**
- **TCPの変更 6633 to 6653**

## OpenFlow 1.5 (策定中)

- **Egress Table対応**
- **フローエントリの統計情報と統計トリガ機能**
- **TCPフラグmatch対応**

## 3.2 SDN/OpenFlow **実装最新動向**

# SDN化を進めるために

## OpenFlowでSDNを体験してみたい

### ●NW制御側

1. 自分でコントローラを作る
  - 結構しんどいです
2. フリーなコントローラを見つける
  - OpenDaylight, Ryu, Trema
3. ソリューションを購入する
  - OpenFlow 1.3対応製品を検索してください

### ●データプレーン側

1. 自分でコントローラを作る
  - かなり結構しんどいです
2. フリーなOpenFlowスイッチを見つける
  - Open vSwitch, Lagopus
3. ソリューションを購入する
  - OpenFlow 1.3対応製品を検索してください

## IaaS基盤をSDN化したい

1. 自分でIaaS基盤を作る
  - 超しんどいです
2. OSSなソリューションを見つける
  - A) OpenFlowベース
  - B) OpenContrail
  - C) Midonet
3. ソリューションを購入する
  - SDN対応IaaS基盤の製品を検索してください



## 3.2.1 SDN/OpenFlow **制御系**

# SDN制御系

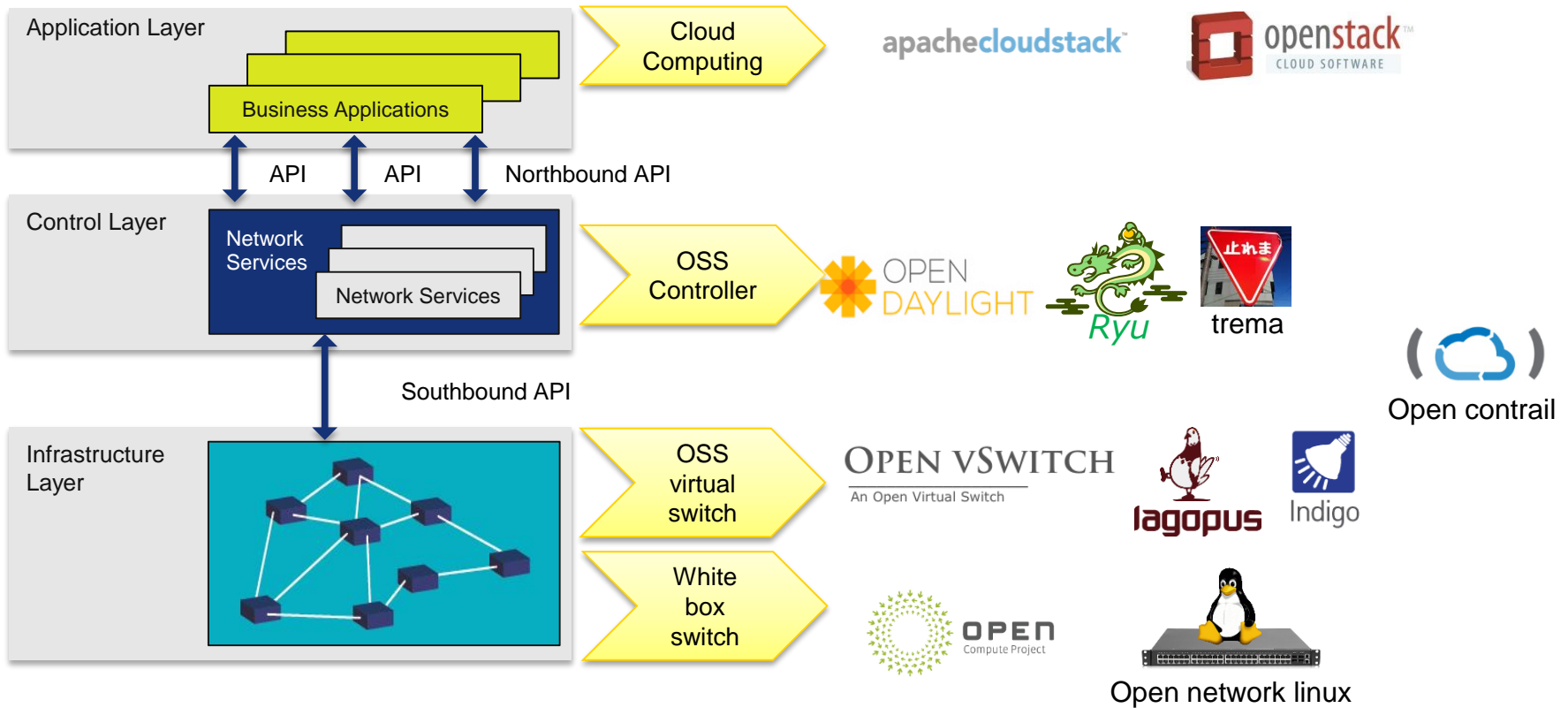
## 自分でNW制御ロジックを書きたい人向け

	OpenDaylight	Ryu	Trema
Language	Java	Python	Ruby, C
License	Eclipse Public v1	Apache v2	GPL
制御可能データプレーン	OpenFlow switch v1.0, 13	OpenFlow switch v1.0 – 1.4 OVSDB対応スイッチ	OpenFlow switch v1.0, 1.3 (trema-edge)
データプレーン設定プロトコル	OVSDB, NETCONF	OVSDB	-
プロトコルライブラリ	○ (BGP, BGP)	○ (いろいろ)	○(いろいろ)
Sample apps	?	○ (L2, L3, LB, firewall)	○ (L2, L3, LB, monitor)

## OSS系でIaaS特化で環境を作りたい

	Ryu	OpenContrail	Midnet
License	Apache v2	Apache v2	GPL
制御可能データプレーン	OpenFlow switch (OVS)	vRouter (contrail)	OpenvSwitch
対応データプレーンプロトコル	GRE, VxLAN	OVSDB	
対応NW	L2 overlay VxLAN, Gre	MPLS over UDP	VxLAN

# SDNを構成するオープンソース



# Open Daylight (1/2)

## SDNを実現するためのOSSのモジュール型コントローラプラットフォーム

- OSGiベースでコントローラの機能を動的にプラグイン可能
  - 機能を実現するためフレームワークSAL (service abstraction layer)の提供
- 上位アプリ向けの双方向のRESTによるNorthbound Interface提供
- 複数のネットワークノードへのSouthbound Interface対応
  - OpenFlow 1.0 & 1.3, NETCONF, PCEP, BGP, BGP-LS

## 大手ネットワークベンダやソリューションベンダが参画 (約40社)

## リリース

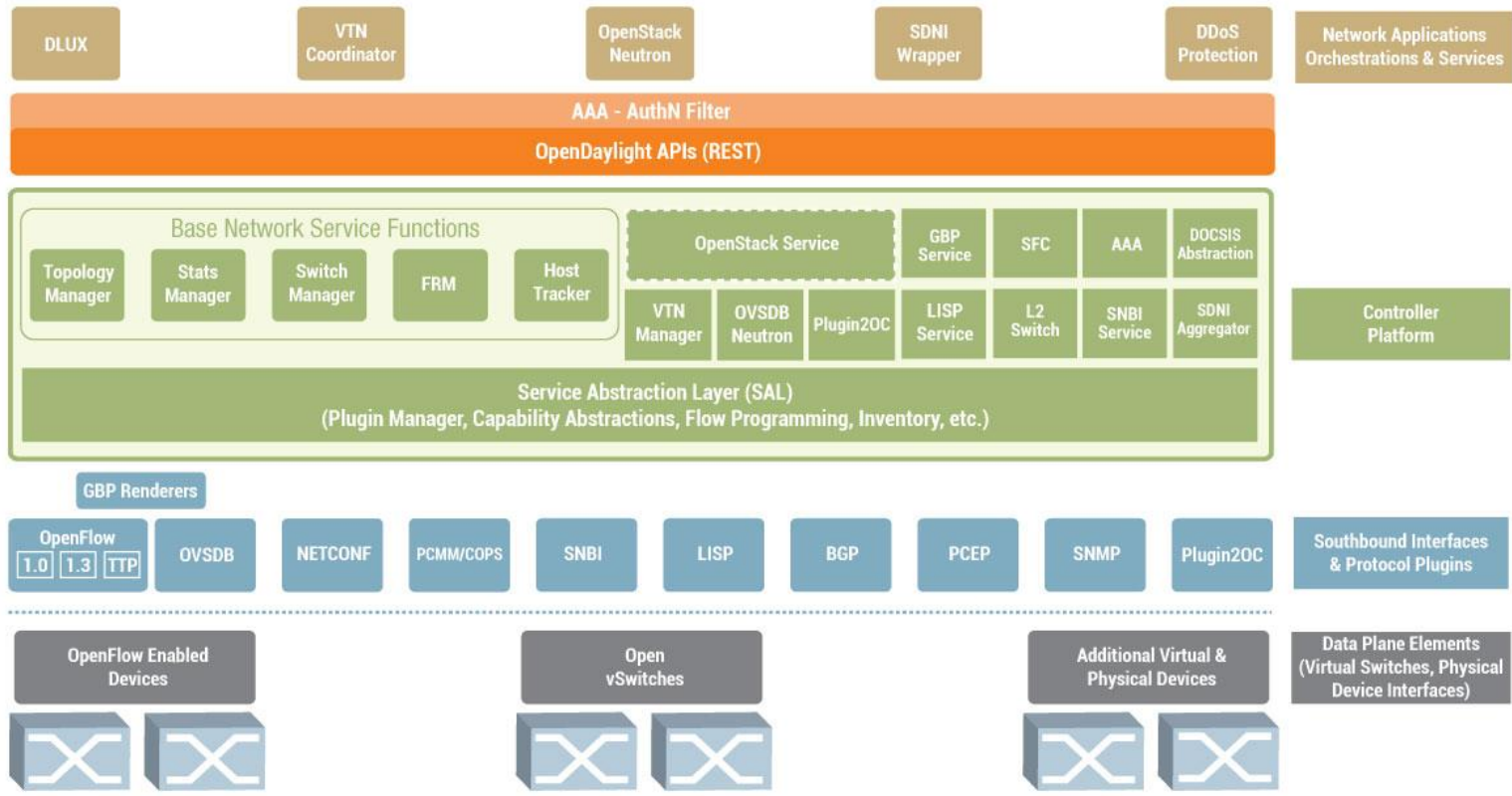
- 2014年2月: Hydrogen
  - Base Edition: OpenFlowを中心とした基本的な機能
  - Virtualization Edition: データセンター向け  
(仮想テナントネットワークやオーバーレイネットワーク、セキュリティ機能)
  - Service Provider Edition: 通信事業者向け  
(BGP-LS, LISP, Metadata service)
- 2014年9月: Heliumをリリース
  - OVSDB, Service Function Chaining, AAAの対応

# Open Daylight (2/2)

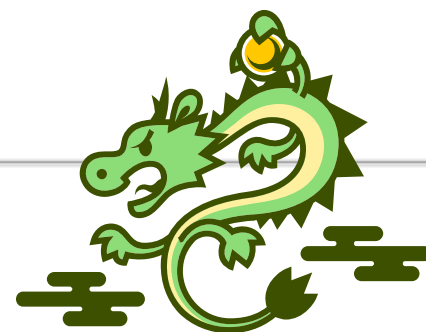


**LEGEND**

- AAA: Authentication, Authorization & Accounting
- AuthN: Authentication
- BGP: Border Gateway Protocol
- COPS: Common Open Policy Service
- DLUX: OpenDaylight User Experience
- DDoS: Distributed Denial Of Service
- DOCSIS: Data Over Cable Service Interface Specification
- FRM: Forwarding Rules Manager
- GBP: Group Based Policy
- LISP: Locator/Identifier Separation Protocol
- OVSDB: Open vSwitch DataBase Protocol
- PCEP: Path Computation Element Communication Protocol
- PCMM: Packet Cable MultiMedia
- Plugin2OC: Plugin To Open-Contrail
- SDNI: SDN Interface (Cross-Controller Federation)
- SFC: Service Function Chaining
- SNBI: Secure Network Bootstrapping Infrastructure
- SNMP: Simple Network Management Protocol
- TTP: Table Type Patterns
- VTN: Virtual Tenant Network



# Ryu: component-based software defined networking framework



http://osrg.github.io/ryu/

## 既存ネットワークの段階的な「SDN化」を実現

- 既存ネットワークプロトコル(BGPなど)のライブラリも提供

## SDNアプリケーションを構築するための部品

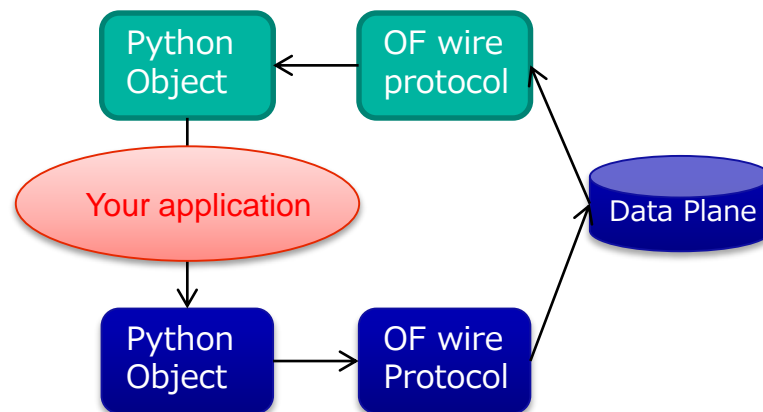
### ● データプレーン制御プロトコル

- OpenFlow 1.0, 1.2, 1.3, 1.4, Nicira Extensions (OpenvSwitch専用)
- OF-config, OVSDB JSON (OpenvSwitch専用)

### ● 共通機能コンポーネント

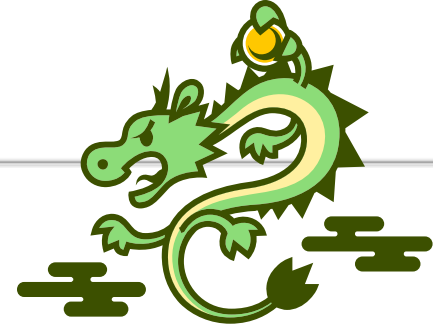
- ルーティング: BGP, MPBGP
- トポロジ検出: LLDP
- クラウドテナントL2分離: VXLAN・GRE・VLAN
- セキュリティ: ファイアウォール
- 高可用化: VRRP・LACP
- パケットライブラリ: BMP
- OpenStack連携: icehouseから新しい機構へ

Ryu converts it

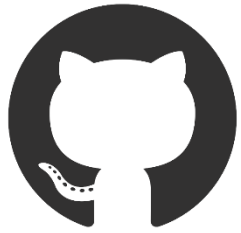


Ryu generates

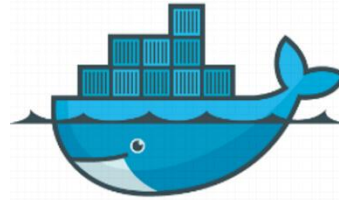
# 自動化されたRyu開発



Unit tests are executed



github



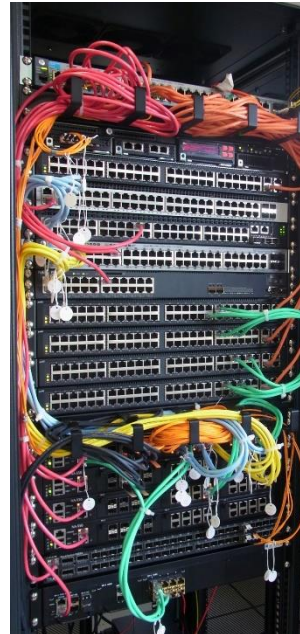
Docker hub image  
is updated

You can update  
your Ryu  
environment  
with one command

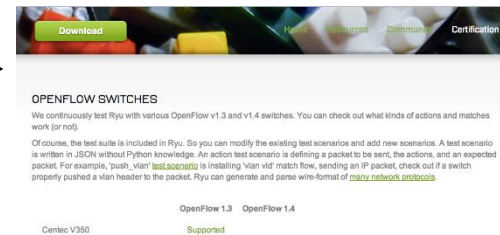
Push the new code



Ryu certification is  
executed on test lab



Ryu certification  
site is updated



# Trema

■ RubyによるコンパクトにまとまったOpenFlow controllerフレームワーク

- コントローラ, OFライブラリ, NWEミュレータ機能を提供
- OpenFlow 1.0, Trema-edgeにてOpenFlow 1.3対応
- 少ないコード数で機能の追加や変更が可能
- 少ない工数で柔軟なNI, SIが可能

■ DCを中心に採用事例あり

■ 入門者にやさしい

- 日本語ドキュメントが揃っている
- <http://trema.github.io/trema/>



trema

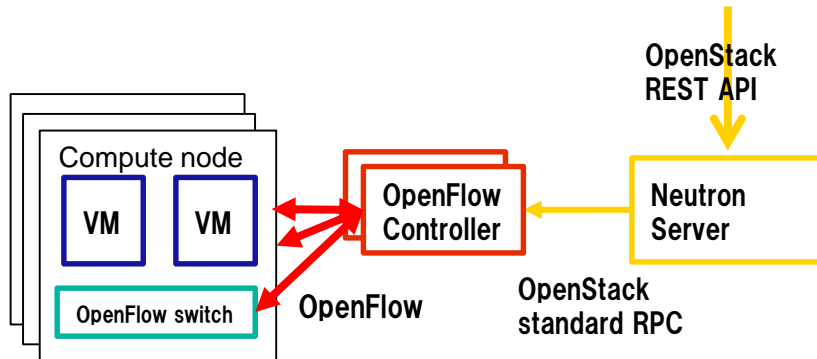


# OpenStack連携にみるコントローラ開発の思想の違い

## 集中管理型

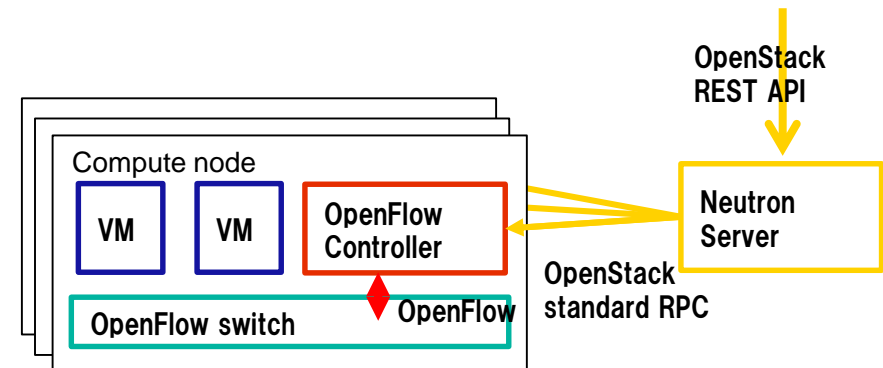
- **実装例**: OpenDaylight, NEC PFとか
- **インテリジェントはOpenFlowコントローラに存在**

- Scalability
- High availability
- Single system image



## 分散管理型

- **実装例**: Ryu (OFAgent driver)
- **インテリジェントはエッジに**
  - Scalability, High-availabilityは他のフレームワークを活用



## 3.2.2 データプレーンの実装

# SDNを構成するデータプレーンの実装

データプレーン 制御プロトコル	ソフトウェア実装	ハードウェア実装		
		ASIC	FPGA	NPU
独自プロトコル	<ul style="list-style-type: none"> <li>• OpenContrail vRouter (XMPP)</li> <li>• Cisco application virtual switch (CISCO ACI)</li> </ul>	<ul style="list-style-type: none"> <li>• Cisco Nexus 9K (Cisco ACI)</li> </ul>		
OpenFlow	<ul style="list-style-type: none"> <li>• Open vSwitch</li> <li>• Lagopus</li> <li>• LINC</li> <li>• Light switch</li> </ul>	<ul style="list-style-type: none"> <li>• NEC Programmable Flow SW</li> <li>• HP Procurve</li> <li>• Pica8</li> <li>• Centec</li> <li>• Accton</li> <li>• EdgeCore</li> <li>• IBM RackSwitch</li> <li>• Juniper MX</li> </ul>	<ul style="list-style-type: none"> <li>• Corsa</li> <li>• Brocade MLX</li> </ul>	<ul style="list-style-type: none"> <li>• NoviFlow NoviKit</li> <li>• Netronome</li> </ul>

## **3.2.2.1 ソフトウェアデータプレーンの実装**

# Intel DPDK

## X86アーキテクチャに最適化されたデータプレーン用ライブラリとNICドライバ

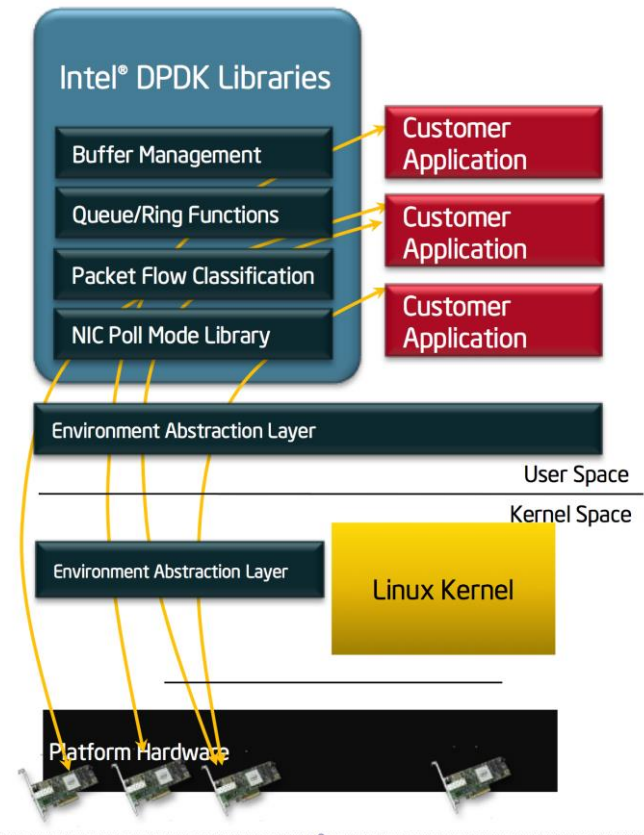
- メモリ構成を考慮したQueue, buffer管理ライブラリ
- packet flow classification
- ポーリングベースのNIC driver

## データプレーン処理に最適化された低遅延 & 高性能なランタイム

## 抽象化されたパケット処理

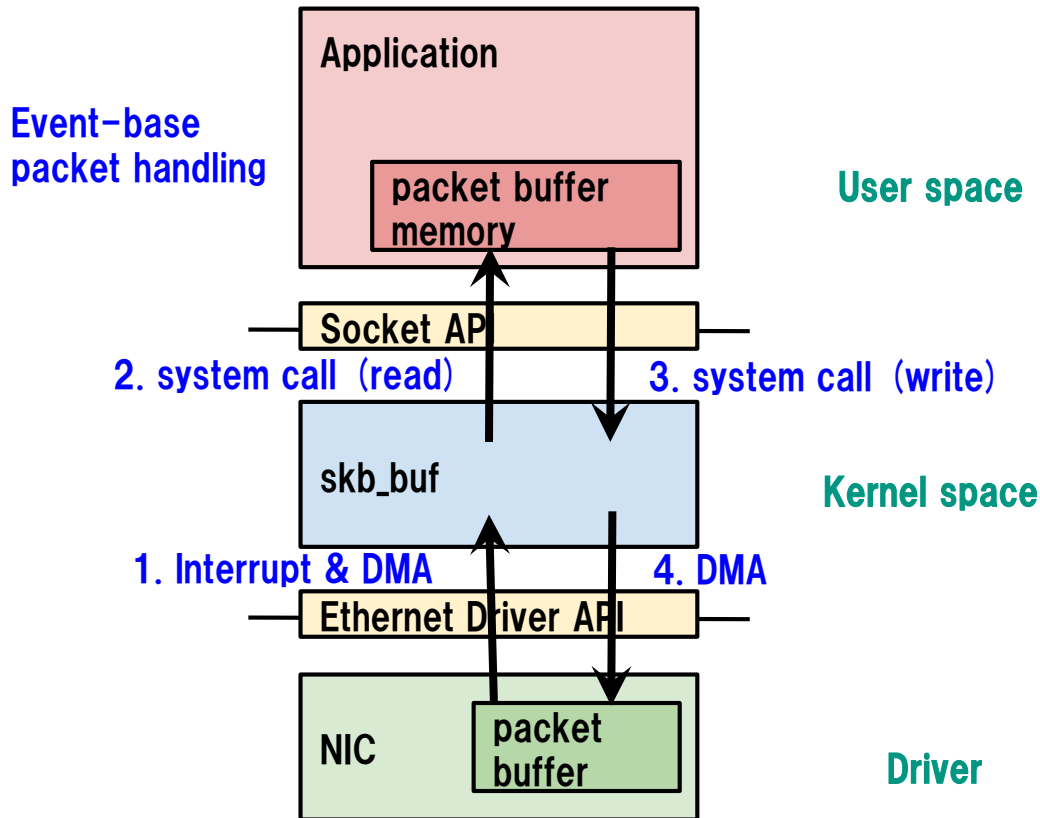
## BSDライセンス :)

- OSSコミュニティにより開発中
- dpdk.org

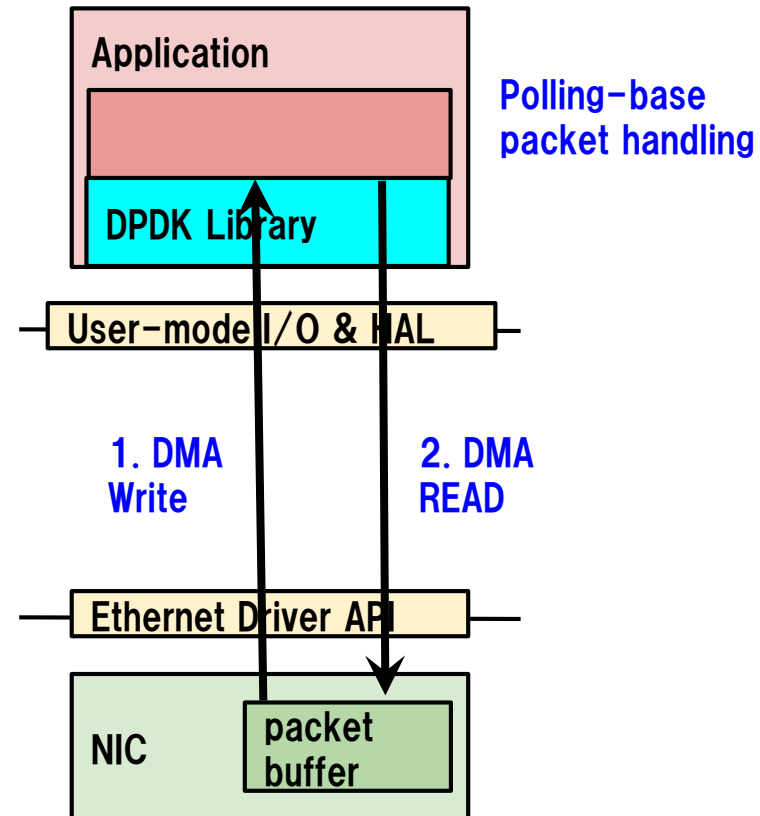


# Intel DPDKによるネットワークI/Oのバイパス

Standard linux application



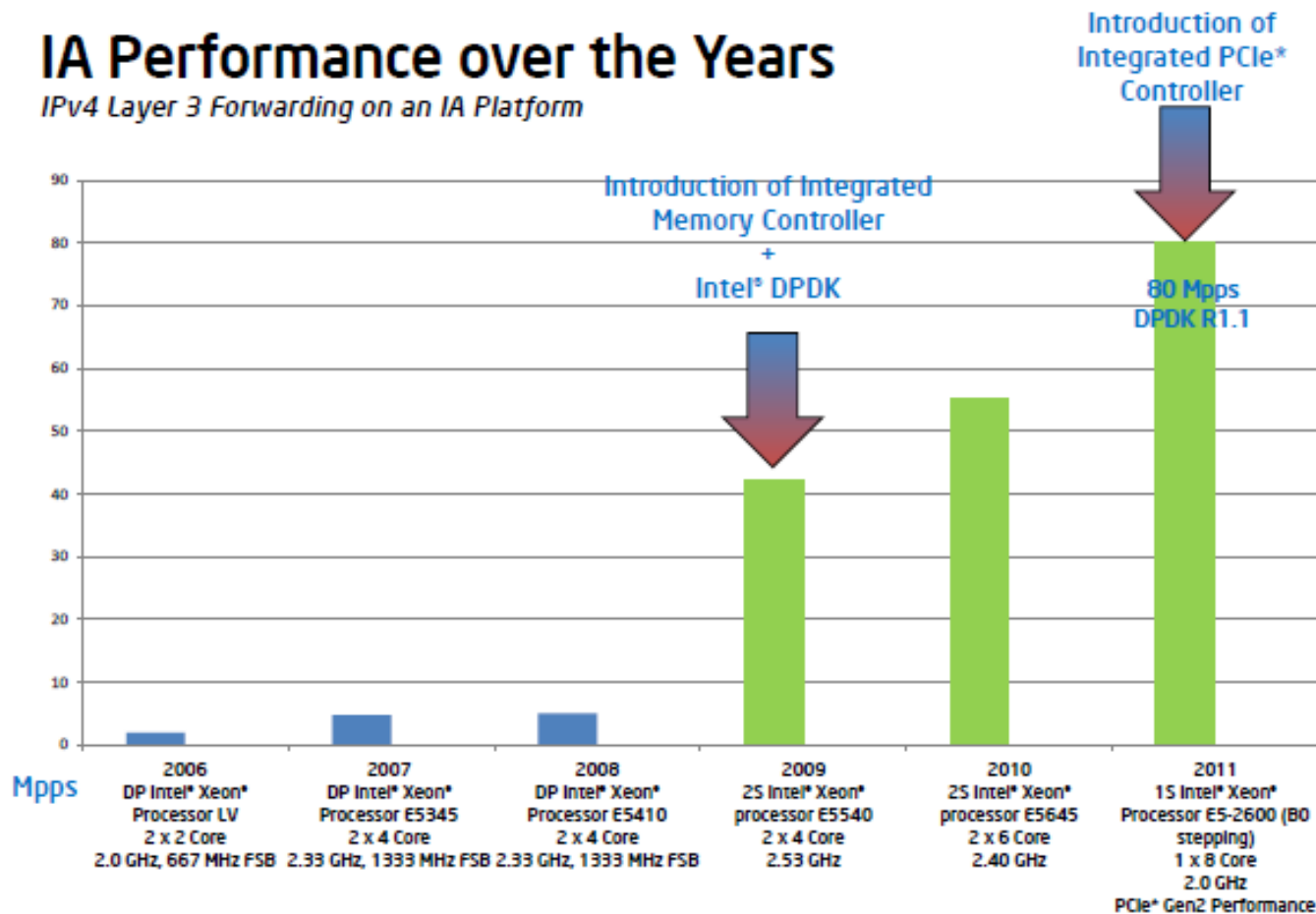
application with intel DPDK



# Intel DPDKの効果

## IA Performance over the Years

IPv4 Layer 3 Forwarding on an IA Platform



Standard "off-the-shelf" IA platform can deliver huge performance.  
Performance jump can be attributed to Core, Memory architecture (iMC) + Intel® DPDK

# 主なSoftware OpenFlow switchの実装

## Open vSwitch

- VMwareが開発を主導
- 様々なLinuxディストリで利用可能
- 様々な仮想化基盤との連携も可能
- Linuxカーネルにデータプレーンのモジュールが取り込まれている
- <http://openvswitch.org/>

## Lagopus

- NTTが開発を主導
- 様々なプロトコルのフレームへのOpenFlow命令の対応
- 高速化および大規模フローへの対応を主眼
- <http://lagopus.github.io/>

## LINC, LINCX

- flowforwardingが開発を主導
- 最新のOpenFlow仕様やOF-CONFIGに対応
- <http://www.flowforwarding.org/>



# Open vSwitch (1/2)

## デファクトスタンダード OpenFlowソフトウェアスイッチ

- Linux kernelにvswitch moduleがマージ済み
- メジャーなLinuxディストリでインストール可能
- 高速なフロー検索機構 megaflow導入による高速化

## NW運用のための統計取得プロトコルサポート

- NetFlow, IPFIX, sFlow, SPAN/RSPAN

## OpenFlowやOVSDBによる集中管理

- Port bonding, LACP, tunneling (VxLAN, GRE, LISP)

## Linuxベースの仮想化やFreeBSDで動作

- Xen, XenServer, KVM, VirtualBox

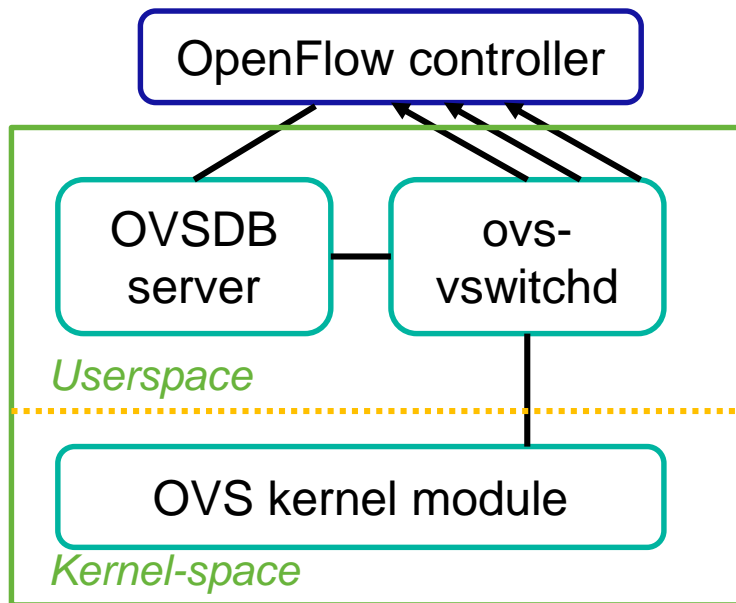
## 様々なデータプレーンへの移植

- Linux Kernel空間のソフトデータプレーン
- DPDKベースのデータプレーン
- 商用スイッチシリコンベースのスイッチ

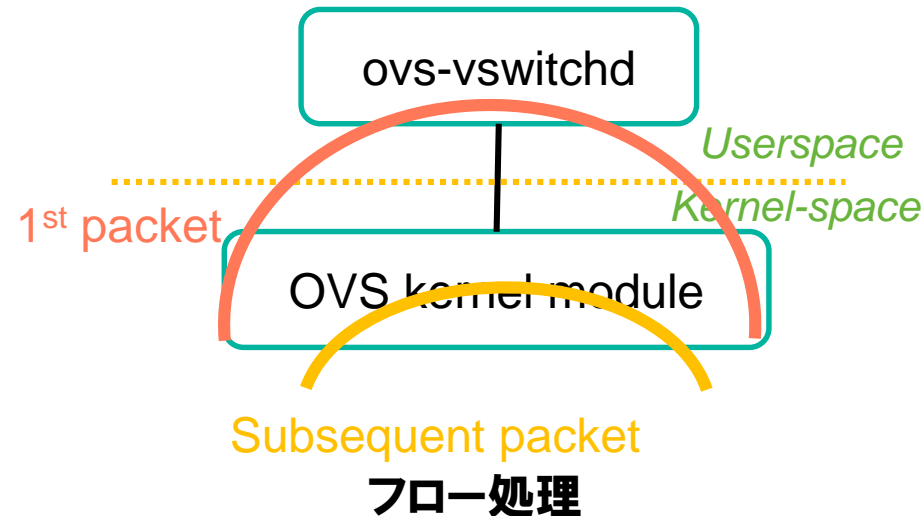
# Open vSwitch (2/2)

## 実装詳細

- <http://openvswitch.org/slides/OpenStack-131107.pdf>



OVSの構成



フロー処理

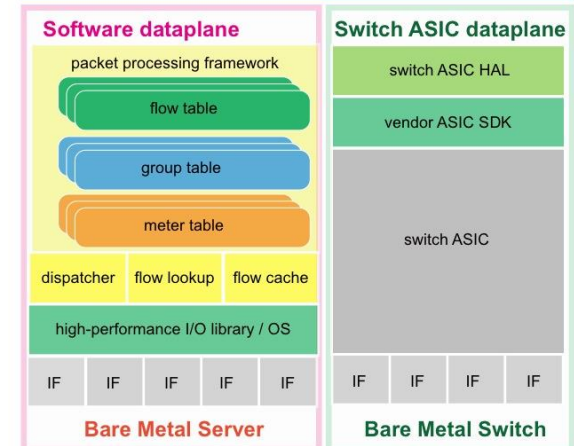
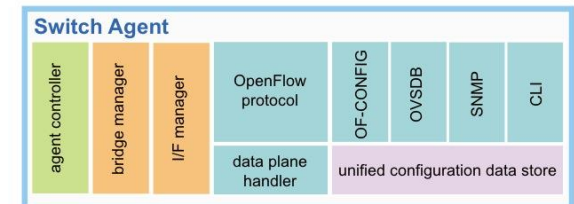
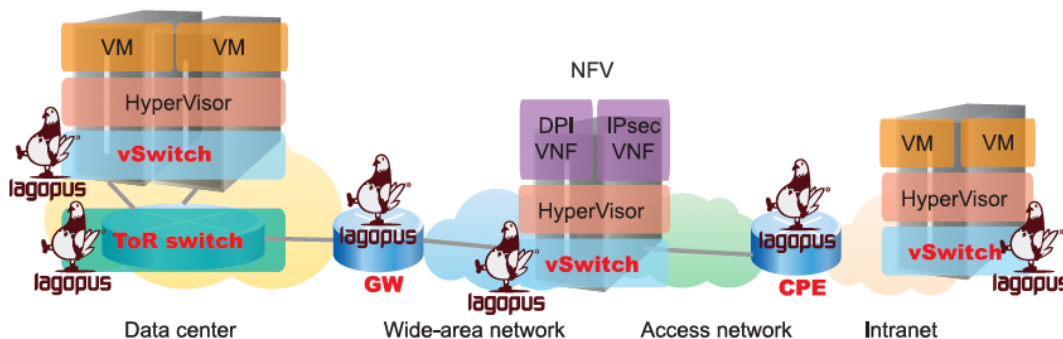
## マルチコアCPUを活用する高性能なソフトウェアOpenFlow switch

- 10Gbps wire-rateのパケット処理
- 100万フローエントリへの対応と高速なフローエントリの投入
- ユーザースペース実装による容易なインストール

## SDNの広域NWへの適用を意識したプロトコルサポート

- data center以外の領域へ
- WAN protocolsのサポート (MPLS and PBB)
- Intel DPDK対応

## ゲートウェイシステムや仮想化基盤向けのvSW



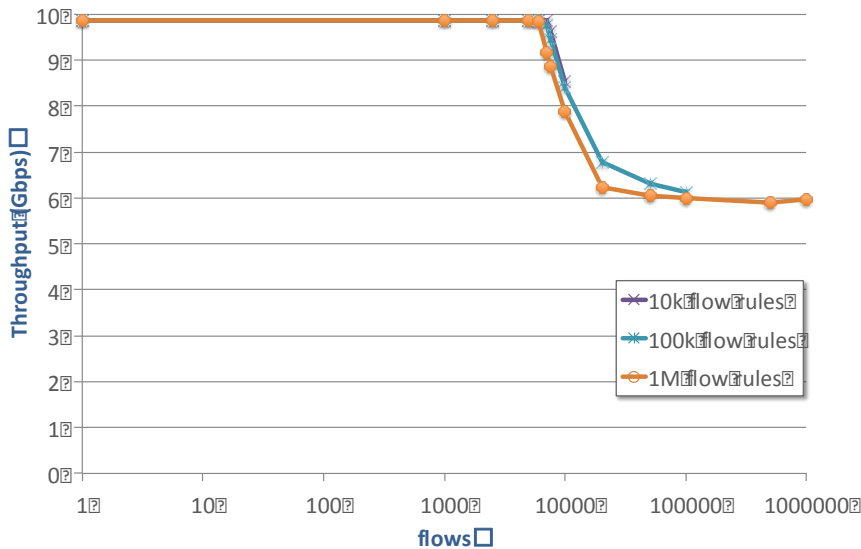
## まとめ

- Throughput: **10Gbps wire-rate**
- Flow rules: **1M flow rules**  
**4000 flowmod add / sec**

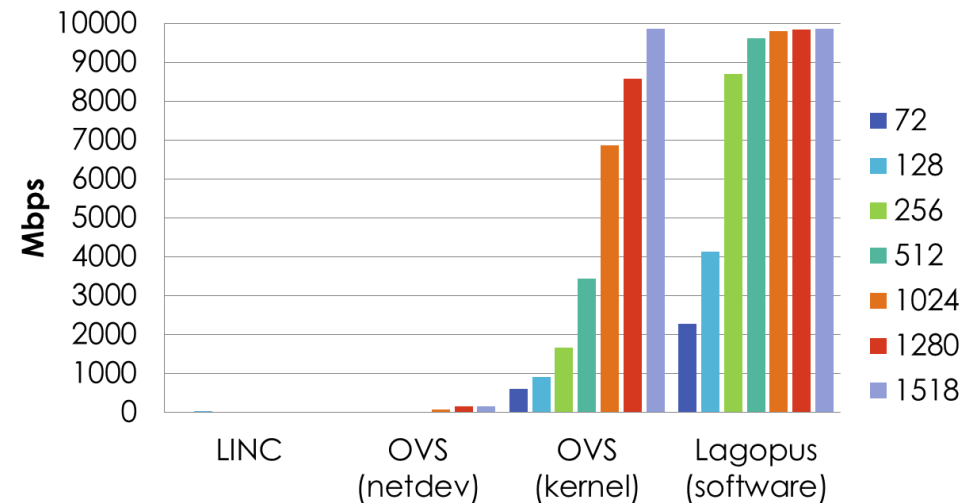
## HWスペック

- Dual Intel xeon E5-2660 (8cores)
- DDR3-1600 64GB
- Intel X520-DA2

Throughput vs flows, 1518 bytes packet



L2 switch performance (Mbps)  
10GbE x 2 (RFC2889 test)



# Ryu OpenFlow 1.3スイッチの機能テスト (2014.10.16)

Type	Action	Set field	Match	Group	Meter	Total
# of test scenario (mandatory, optional)	56 (3, 53)	159 (0, 159)	714 (108, 606)	15 (3, 12)	36 (0, 36)	991 (114, 877)
Lagopus 2014.09.12	56 (3, 56)	150 (0, 159)	714 (108, 606)	15 (3, 12)	30 (0, 30)	976 (114, 862)
OVS (kernel) 2014.08.08	34 (3, 31)	96 (0, 96)	534 (108, 426)	6 (3, 3)	0 (0, 0)	670 (114, 556)
IVS 2014.09.12	17 (3, 14)	46 (0, 46)	323 (108, 215)	1 (0, 1)	0 (0, 0)	387 (111, 276)
ofswitch 2014.09.08	50 (3, 47)	100 (0, 100)	577 (108, 469)	15 (3, 12)	26 (0, 26)	768 (114, 654)
LINC 2014.09.15	24 (3, 21)	68 (0, 68)	428 (108, 320)	3 (3, 0)	4 (0, 4)	527 (114, 413)
Trema 2014.08.24	50 (3, 47)	159 (0, 159)	708 (108, 600)	15 (3, 12)	34 (0, 34)	966 (114, 854)

## **3.2.2.2 ハードウェアデータプレーンの実装**

# ハードウェアのOpenFlowスイッチ

## 大手ベンダからOpenFlow 1.3対応スイッチが販売

- 内部で使用しているチップ・構成により使用可能なOpenFlowの機能はまちまち
- 詳しくは <http://osrg.github.io/ryu/certification.html>
  - 最新のファームウェアに対しての対応結果をCIで公開



Download | Home | Community | Certification

### OPENFLOW SWITCHES

We continuously test Ryu with various OpenFlow v1.3 and v1.4 switches. You can check out what kinds of actions and matches work (or not).

Of course, the test suite is included in Ryu. So you can modify the existing test scenarios and add new scenarios. A test scenario is written in JSON without Python knowledge. An action test scenario is defining a packet to be sent, the actions, and an expected packet. For example, 'push\_vlan' test scenario is installing 'vlan vid' match flow, sending an IP packet, check out if a switch properly pushed a vlan header to the packet. Ryu can generate and parse wire-format of many network protocols.

	OpenFlow 1.3	OpenFlow 1.4
Centec V350	Supported	
CpQd	Supported	
Edge-Core AS4600-54T	Supported	
HP 2820	Supported	
IBM RackSwitch GB264	Supported	
Indigo Virtual Switch	Supported	
Lagopus	Supported	
LINC	Supported	Supported
NEC PF5220	Supported	
NoviFlow NovaGS200	Supported	
Open vSwitch	Supported	Supported
Open vSwitch (netdev)	Supported	Supported
Pica8 P-3290	Supported	Supported
Trema Switch	Supported	

RESOURCES | COMMUNITY | CERTIFICATION | CONTACT US

Home | Resources | Community | Contribution | Copyright © Ryu SDN Framework Community, 2014 | Design by ChuuDai

### Ryu Certification

- IBM RackSwitch-GB264
- Configuration

	OK	ERROR
Action	6	50
(Required)	(3)	(0)
(Optional)	(3)	(50)
set_field	14	156
(Optional)	(14)	(156)
Match	108	606
(Required)	(54)	(54)
(Optional)	(54)	(552)
Group	0	15
(Required)	(0)	(3)
(Optional)	(0)	(12)
Meter	0	36
(Optional)	(0)	(36)
Total	128	863
(Required)	(57)	(57)
(Optional)	(71)	(806)

### Action

	Required	IPv4	IPv6	ARP
OUTPUT	x	OK	OK	OK
PUSH_VLAN	-	ERROR	ERROR	ERROR
PUSH_MPLS	-	ERROR	ERROR	ERROR
PUSH_FIBB	-	ERROR	ERROR	ERROR
PUSH_VLAN (multiple)	-	ERROR	ERROR	ERROR

Whitebox switch



# スイッチのホワイトボックス化の流れ

## スイッチのハードウェア

- 1980年代: CPUの全盛  
自社での製造が主流
- 1990年代: カスタムチップ全盛
- 2000年代: 汎用スイッチシリコン台頭  
OEMベンダでの製造が主流
- 2010年代: 商用スイッチシリコン全盛



汎用スイッチシリコンベース  
ODMベンダ製造のHW



## ソフトウェア

- 1980年代: 専用OS
- 1990年代: 汎用OS (BSD系)の利用
- 2000年代: 汎用OS (Linux系)の台頭
- 2010年代: ほとんどLinux化



スイッチエージェント  
on  
Linux



## 管理

- 1980年代: CLI
- 1990年代: CLI ☺
- 2000年代: CLI ☺☺
- 2010年代: サーバ系設定自動化ツールとの連携



サーバ系設定自動化ツール

# What's white Box Switch

## 商用Switchシリコンを使用したスイッチ

- Broadcom Trident系
- Intel FM6000

## スイッチ制御を汎用CPUで処理

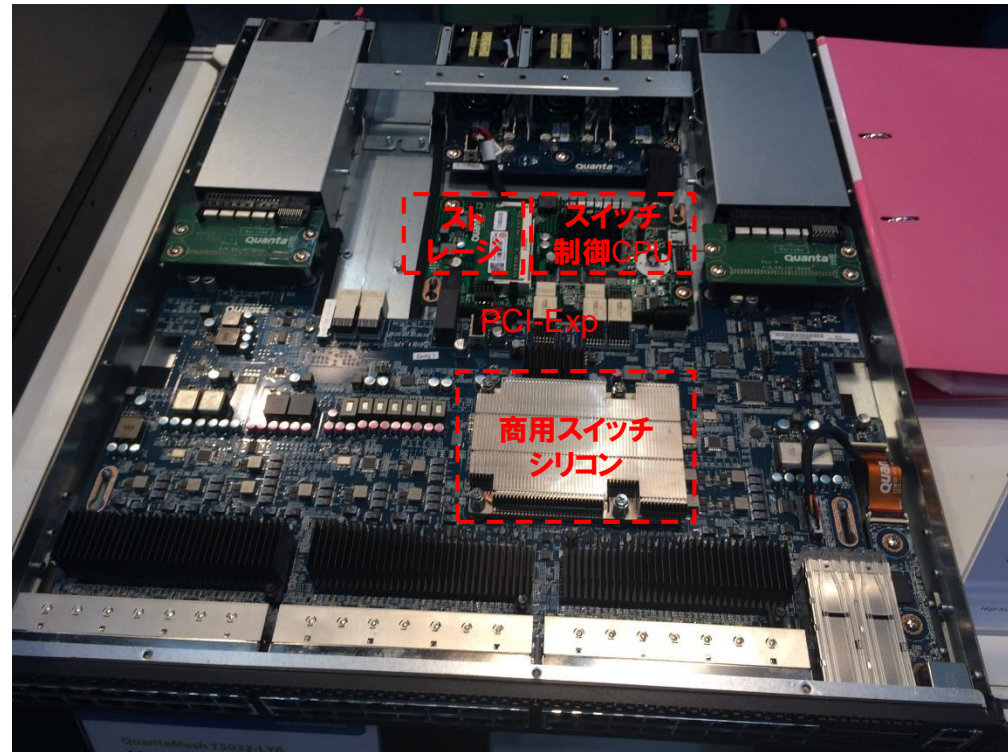
- Intel, ARM, Powerpc

## ODMベンダが製造

- Quanta
- Accton
- Pica8
- Dell
- Advantech
- Penguin computing

## OSは自由に選べる

- Cumulus Linux (Cumulus)
- Switch light OS (Bigswitch)
- PicaOS (Pica-8)
- OpenNetwork Linux (OSS)



# Whitebox switchはどこで買えるの

秋葉原ではまだ買えませんが ☹️ webからは発注可能









OSは別途購入必要

WHITEBOX

HOME OPEN NETWORK SWITCHES - CUMULUS LINUX - QUOTE REQUEST

Home / Switches / Page 1 of 1





### Switches

 <p>SALE</p> <p>Quanta T1048-LB9 \$ 2,499.00 <del>\$ 2,799.00</del></p>	 <p>SALE</p> <p>Edge-Core AS5610-52X \$ 5,995.00 <del>\$ 7,995.00</del></p>	 <p>SALE</p> <p>Edge-Core AS6701-32X with Cumulus OS \$ 8,995.00 <del>\$ 19,995.00</del></p>
 <p>SALE</p> <p>QuantaMesh 3000 Series T3048-LY2 \$ 5,995.00 <del>\$ 7,995.00</del></p>	 <p>No image</p>	 <p>SALE</p> <p>Edge-Core AS4600-54T \$ 2,499.00 <del>\$ 2,799.00</del></p>
 <p>SALE</p> <p>Edge-Core AS5710-54X \$ 2,499.00 <del>\$ 2,799.00</del></p>	 <p>Quanta T5032-LY6 \$ 8,950.00</p>	

bm-switch.com

HOME BARE METAL SWITCHES - CUMULUS SNS - ACCESSORIES - BLOG

### 10G Ethernet switches

 <p>Quanta T3048-LY1 10/40G BM switch with Intel ONS support \$4,800.00</p>	 <p>Quanta T3048-LY2 10/40G BM switch preloaded with ONIE \$5,400.00</p>	 <p>Quanta T3048-LY2R 10/40G BM switch preloaded with ONIE \$4,100.00</p>	 <p>Quanta T3048-LY8 10/40G BM switch preloaded with ONIE \$5,300.00</p>
--	---	--	---

CONTACT INFORMATION  
bm-switch.com  
Phone: + 866-2-05537088  
Email: sales@bm-switch.com

PAYMENT METHODS  
AMERICAN EXPRESS VISA MASTERCARD

USEFUL LINKS  
Search  
About Us  
Our gear  
Bare Metal switches  
Cumulus Networks

© 2014 bm-switch.com, All Rights Reserved.

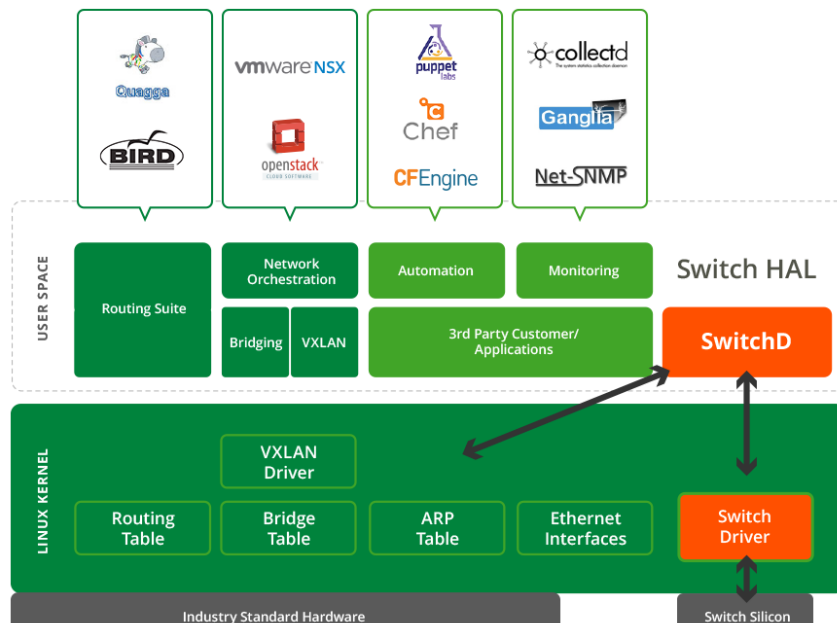
# Cumulus Linux

## LinuxのディストリのようにNetworkを設定

- Bridge系, IP系コマンドを使ったVLAN, IPの設定

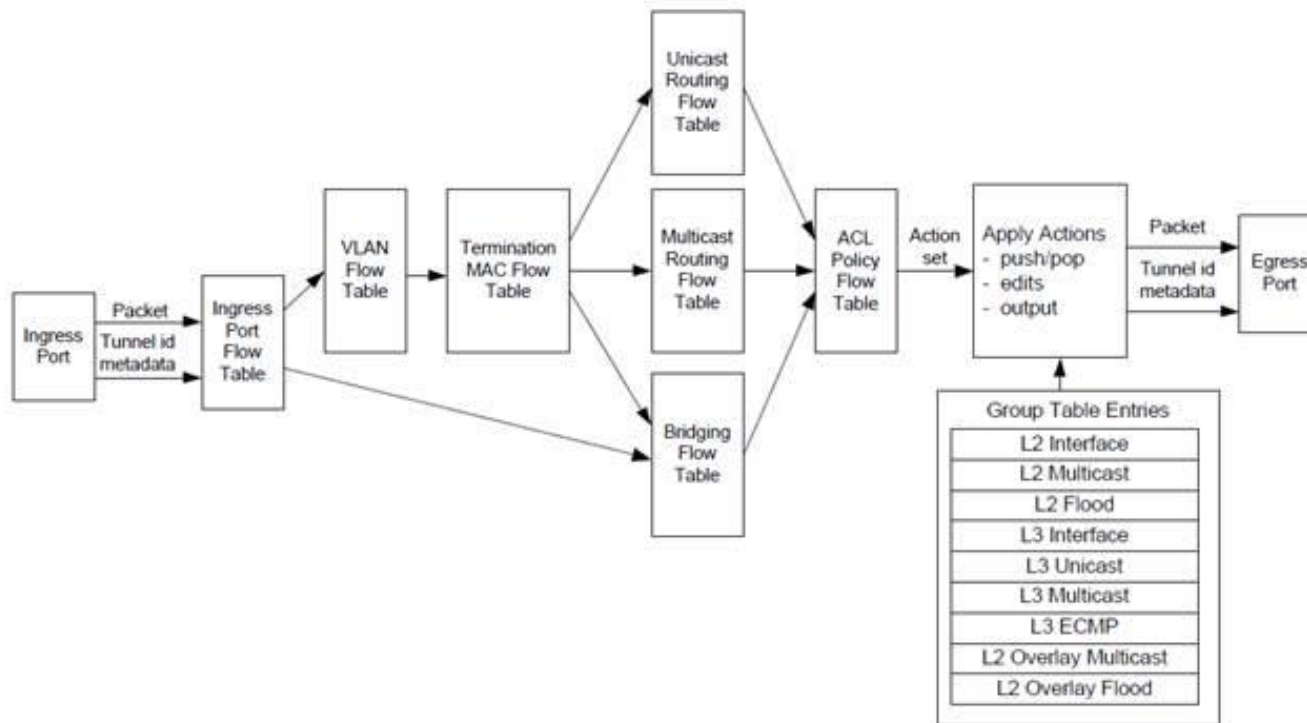
```
# bridgeへのインターフェスの追加
cumulus@switch:~$ sudo brctl addbr my_bridge
cumulus@switch:~$ sudo brctl addif my_bridge swp1 swp2 swp3 swp4
# vlan 10の設定
cumulus@switch:~$ sudo ip link add link swp1 name swp1.10 type vlan id 10
cumulus@switch:~$ sudo ip link add link swp2 name swp2.10 type vlan id 10
```

## 既存自動設定ツールを使った一括設定



## DC向けホワイトボックススイッチ向けのOpenFlow switch agentファームウェア OF-DPAの提供とOpenFlowのFlow Tableデザイン

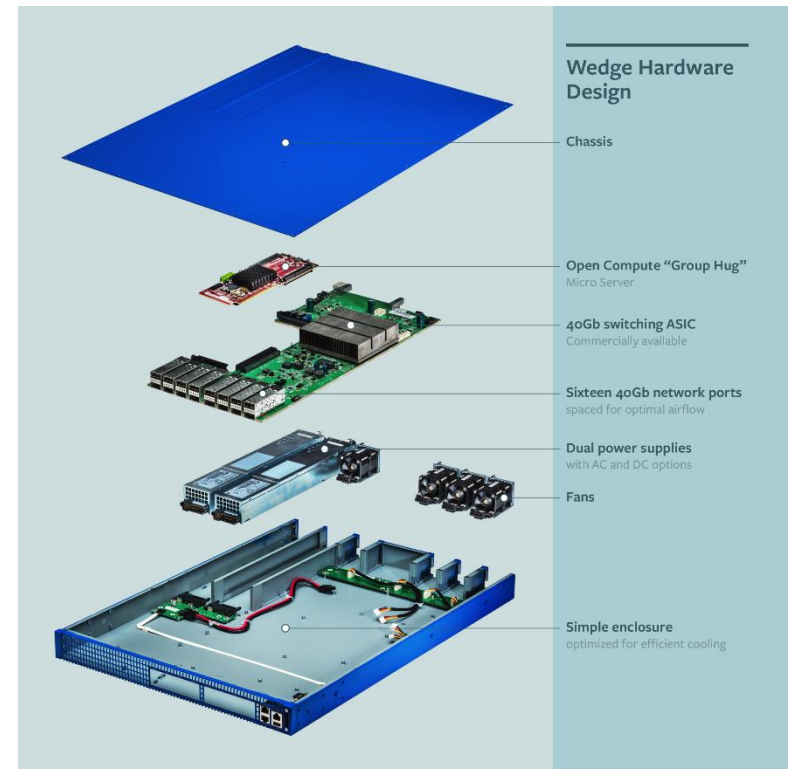
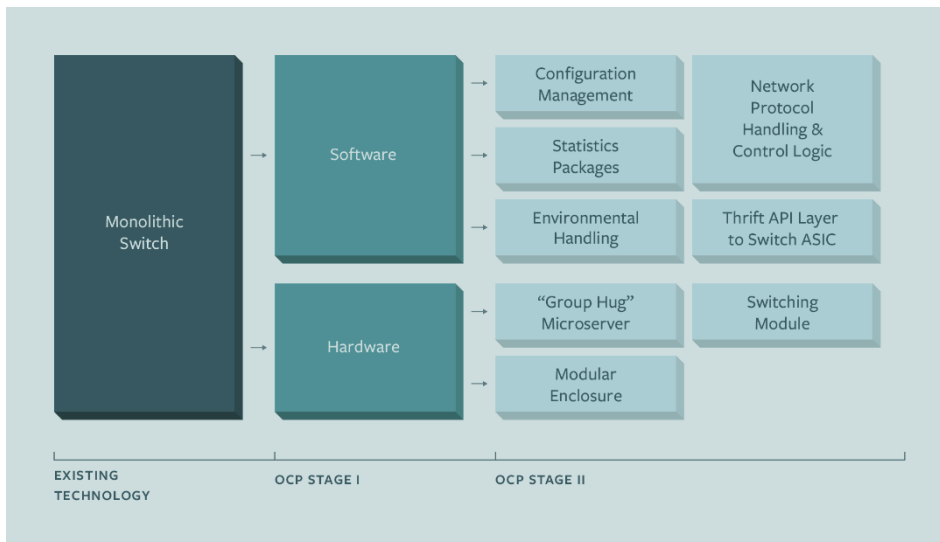
- レイヤリングの考えを取り入れたFlow table
- Broadcom Trident系チップを搭載したホワイトボックススイッチに対応
- <https://github.com/Broadcom-Switch/of-dpa>



# Wedge & FBOSS

## Facebookが主導する商用スイッチシリコンベースのToRのスイッチ

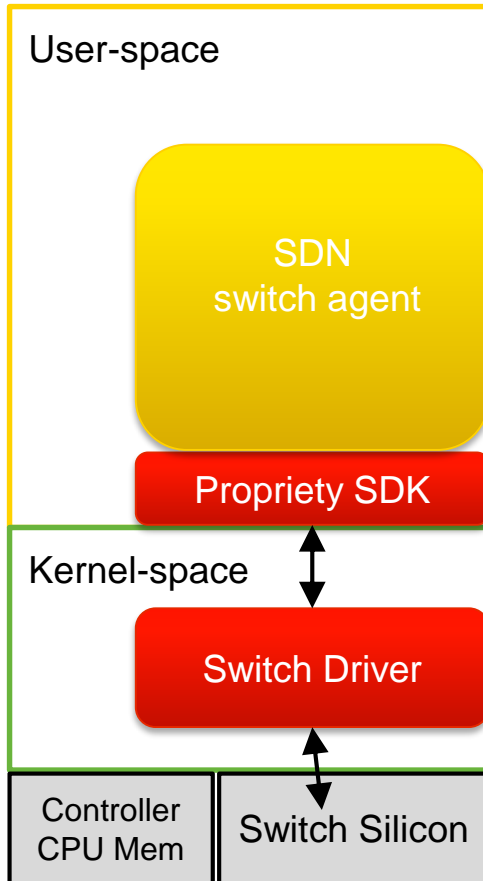
- ToRスイッチをHWとソフトウェアの観点からオープン化を推進
  - OCPのネットワーク部門としても推進
- Wedge: 商用スイッチシリコンを用いたハードウェアのデザイン
- FBOSS: スイッチOS
  - Switch ASICへのThriftベースの抽象化レイヤを提供



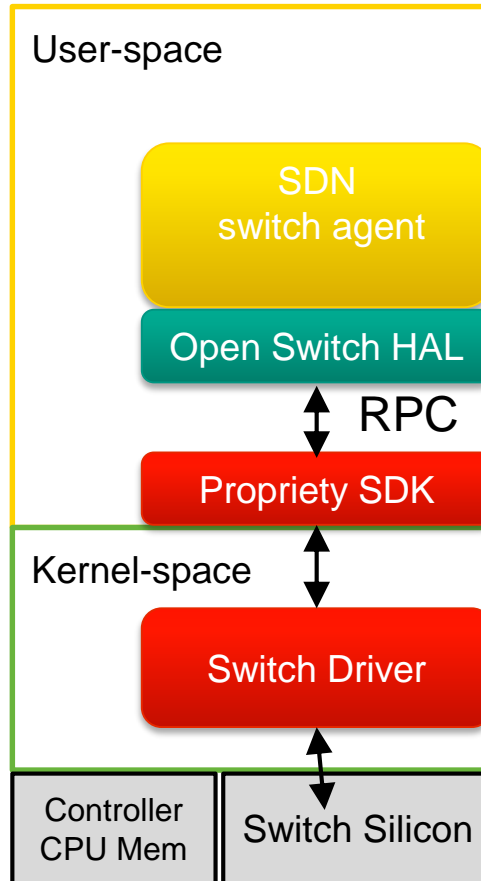
Reference: <https://code.facebook.com/posts/681382905244727/introducing-wedge-and-fboss-the-next-steps-toward-a-disaggregated-network/>

# Whitebox用スイッチOSの構成

現状のスイッチOS  
(ソース公開不可)



上位APIだけオープン  
(上部のエージェントのみOSS化)



真の  
OSSスイッチOS

