

光コアネットワークのSDN化 への取り組み

～広域ネットワークでのSDN実現に向けて～

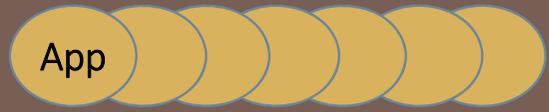
富士通(株) 山田亜紀子

akikoo@jp.fujitsu.com

SDN: 簡単なおさらい



Vertically Integrated
Closed, proprietary
Slow innovation



多種多様なアプリ

----- Open Interface (NBI) ----- プログラマビリティ提供



様々なコントローラ

----- Open Interface (SBI) -----



単純なスイッチ製品



- NEC
- DELL
- HP
- IBM
- Cisco
- Juniper
- Pica8
- Arista
- BROCADE
- FUJITSU
- etc...

出典: "How SDN will shape networking", Nick McKeown, 1st Open Networking Summit (2011/10)

SDNに特に期待するポイント

ネットワークをプログラムで変えられる

ネットワークやさんの視点：

- 手動でのネットワーク変更は怖いけど、SDNコントローラによる変更はテスト済だから安心！
- 集中制御アーキなので、SDNコントローラ機能もいろいろ作って追加しやすい

ネットワークを使うユーザ視点：

- ネットワーク設定をPlug&Playで自動化できるとGOOD
(自動的に設定してくれるアプリがSDNによってできるよね)

どうしてネットワークをプログラムで変えたいの？

●ユーザ視点の例

“いつもはコネクティビティがあればいいけど、大事な会議のときに映像が絶対に途切れないようにしたい”

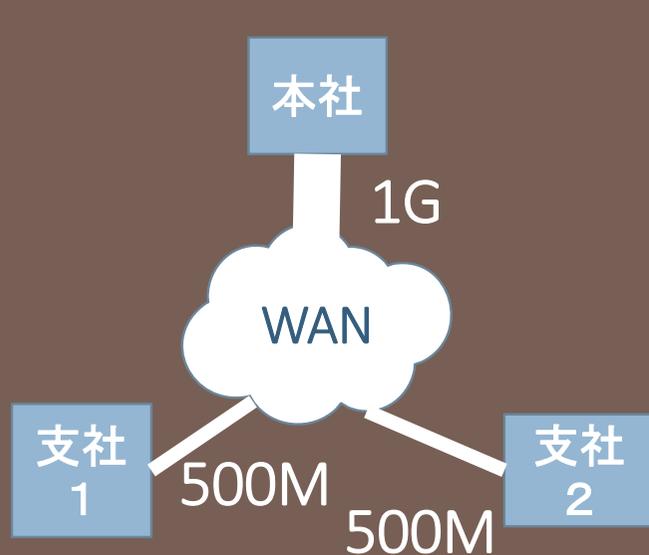
大事な会議のセッティング
とあわせて

●オペレータ視点の例

“ユーザAさんはネットワークを月～金までしか使わない。土日は他の人に使って（買って）もらいたい”

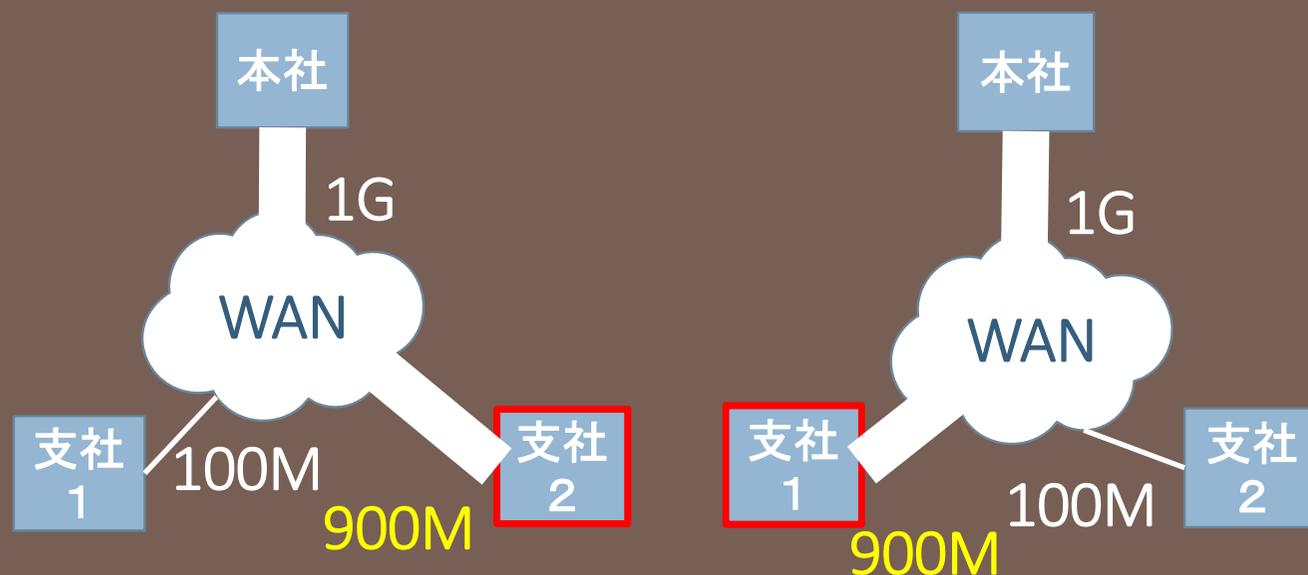
土日に使いたい人に
あわせて

広域SDNで、例えばこんなことができる



契約時に必要帯域を想定

BEFORE : 変更不可



あるときは・・・またあるときは・・・。

AFTER : 自在に変化

広域SDNのメリット

柔軟な広域ネットワーク活用で得られる様々なメリット

ディザスタリカバリのための
データバックアップが超はやい

大事な会議も場所を気にせず開催できる
(移動費よりネットワーク費のほうが安い)

windowsアップデートが多いときは
アップデートファイルを急いで別拠点に送り込んで、
拠点毎に分散してアップデートできる

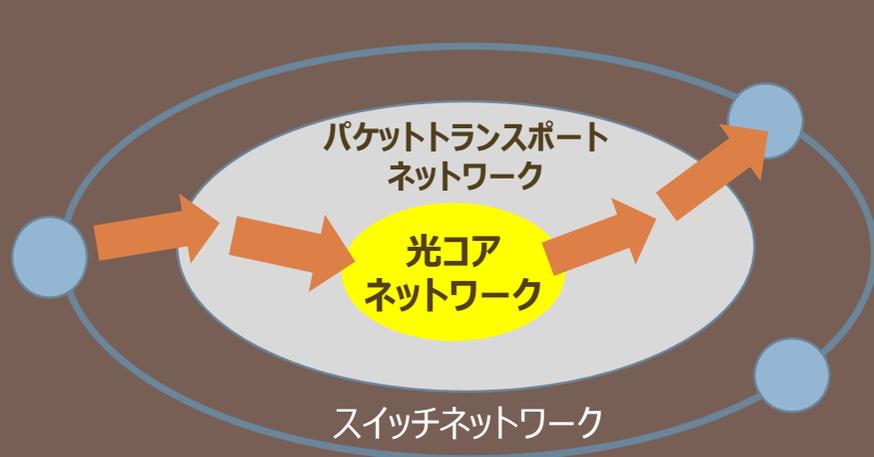
センサーを配置したら
Plug&Playでクラウドにデータ
を集めはじめてくれる

光コアネットワークの SDN

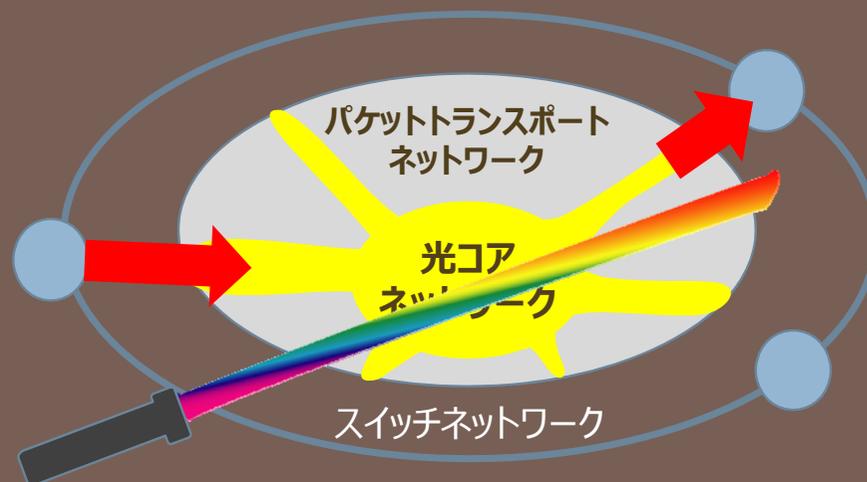
マルチレイヤ連携で、光も使いこなす

光コアネットワークSDN

広帯域が必要な接続には、パケットトランスポートでAggregationせず、広帯域・低遅延保証が可能な光コアネットワークを直接利用することも可能に



BEFORE

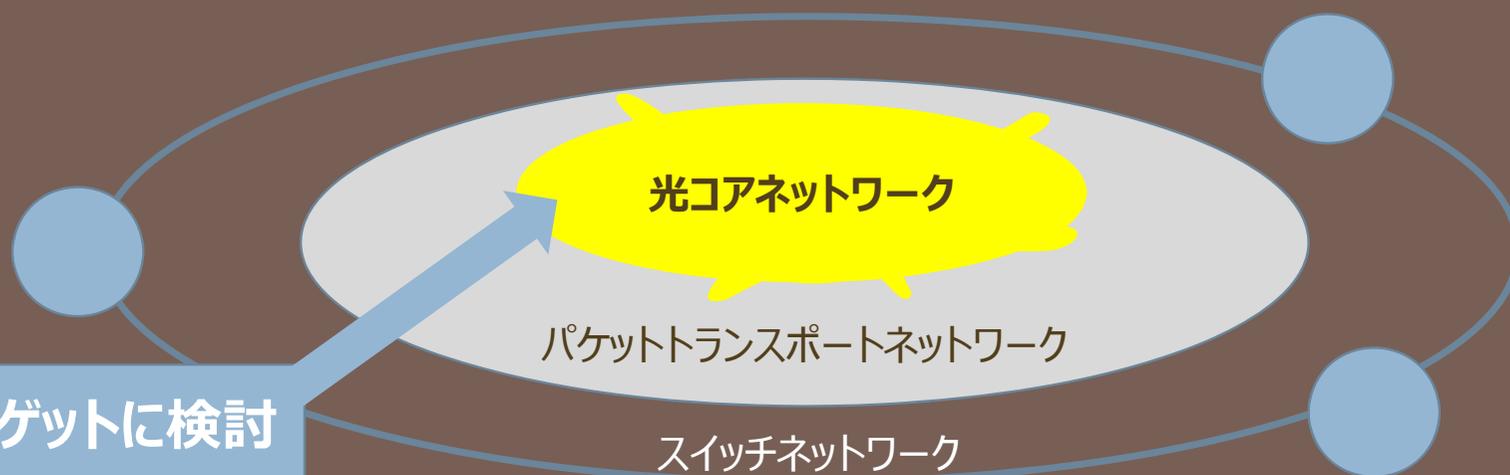


AFTER

光コアネットワークSDNを実現するには

1. SDNによる制御管理のための単純なリソースモデル
 - 複雑な光コアネットワークでのプログラマビリティの実現
2. 上位レイヤサービスとのシームレスな連携（マルチレイヤ）

OTN/WDMをターゲットに検討

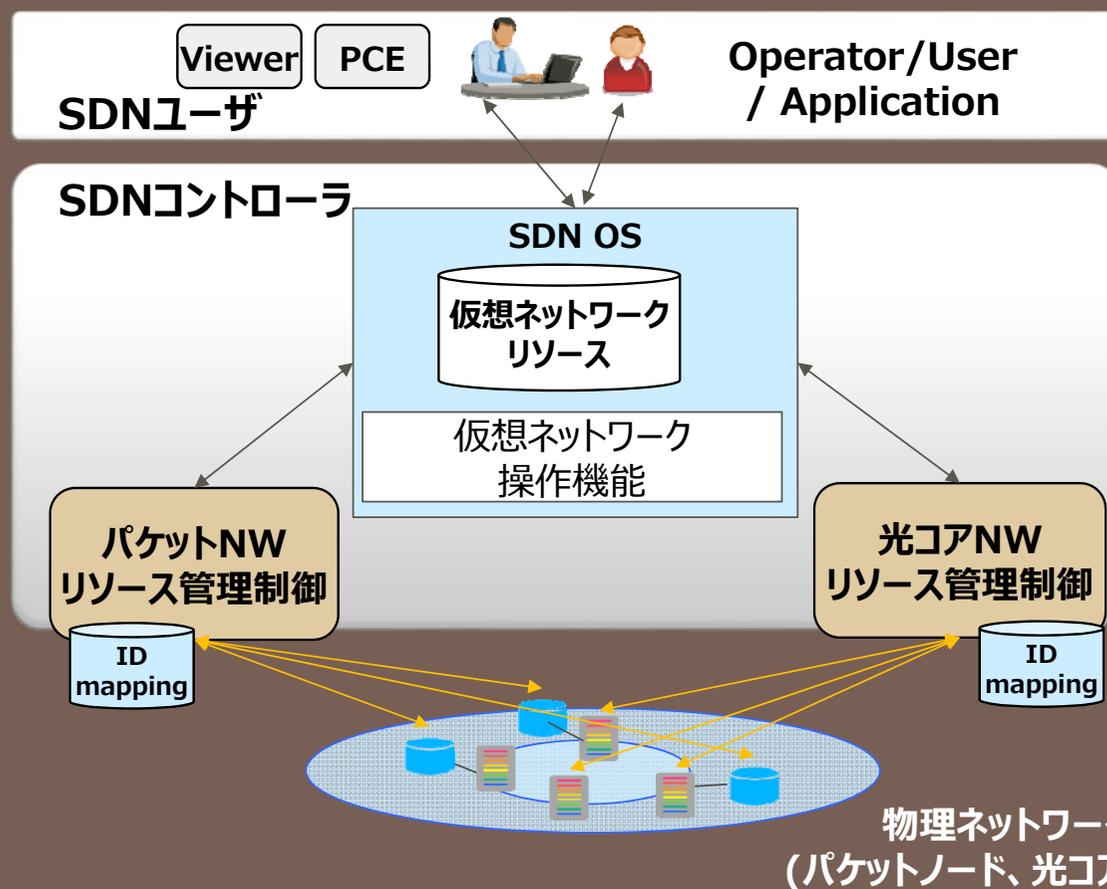


マルチレイヤ連携のために

■マルチレイヤをオーケストレーションするSDNコントローラの実現が必要

- ✓ レイヤに依存しないリソースモデル
- ✓ レイヤ間連携のための汎用的な機能

マルチレイヤSDN制御アーキテクチャ

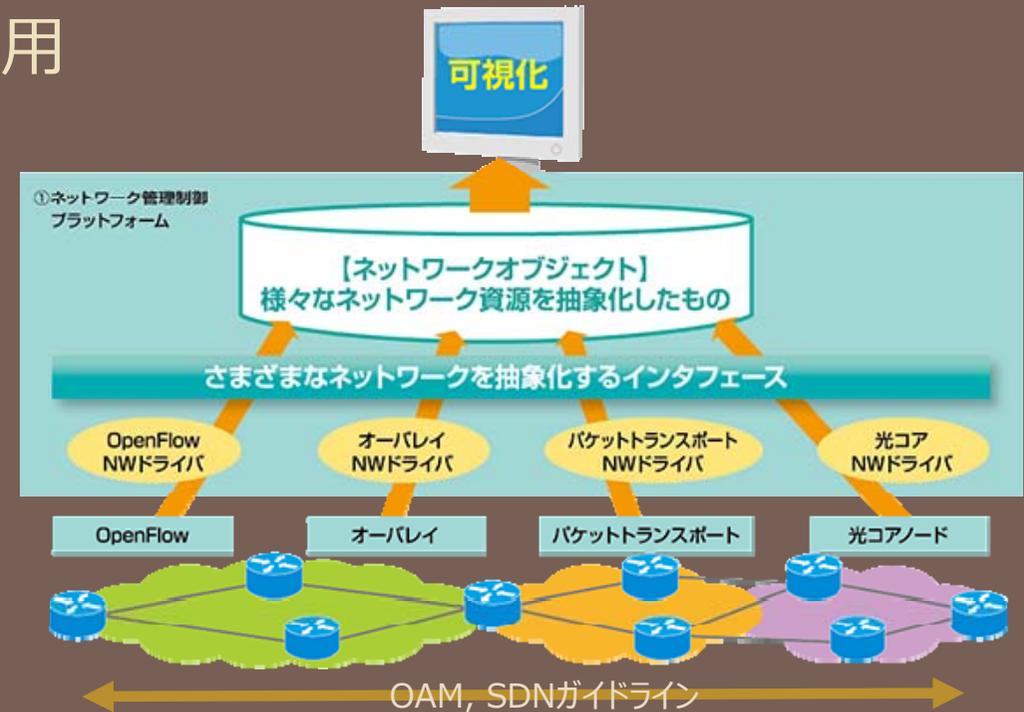
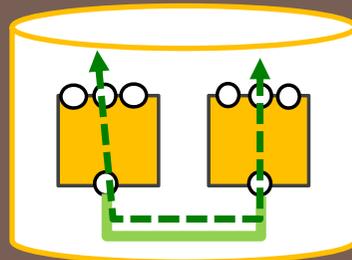
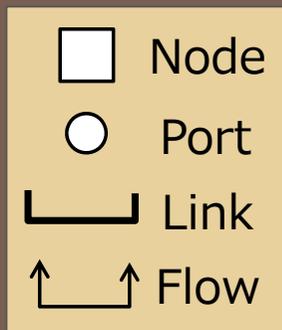


- 仮想ネットワークリソースは、レイヤ非依存で統一的に扱える
- レイヤ間連携のための仮想ネットワーク操作機能を具備
- NWリソースへの要求と連動し、管理制御機能が動作

SDN OS: ネットワーク管理制御プラットフォーム@O3 Project

- ✓ 広域ネットワークのSDN実現のための技術研究開発を推進
- ✓ ネットワーク管理制御プラットフォーム用

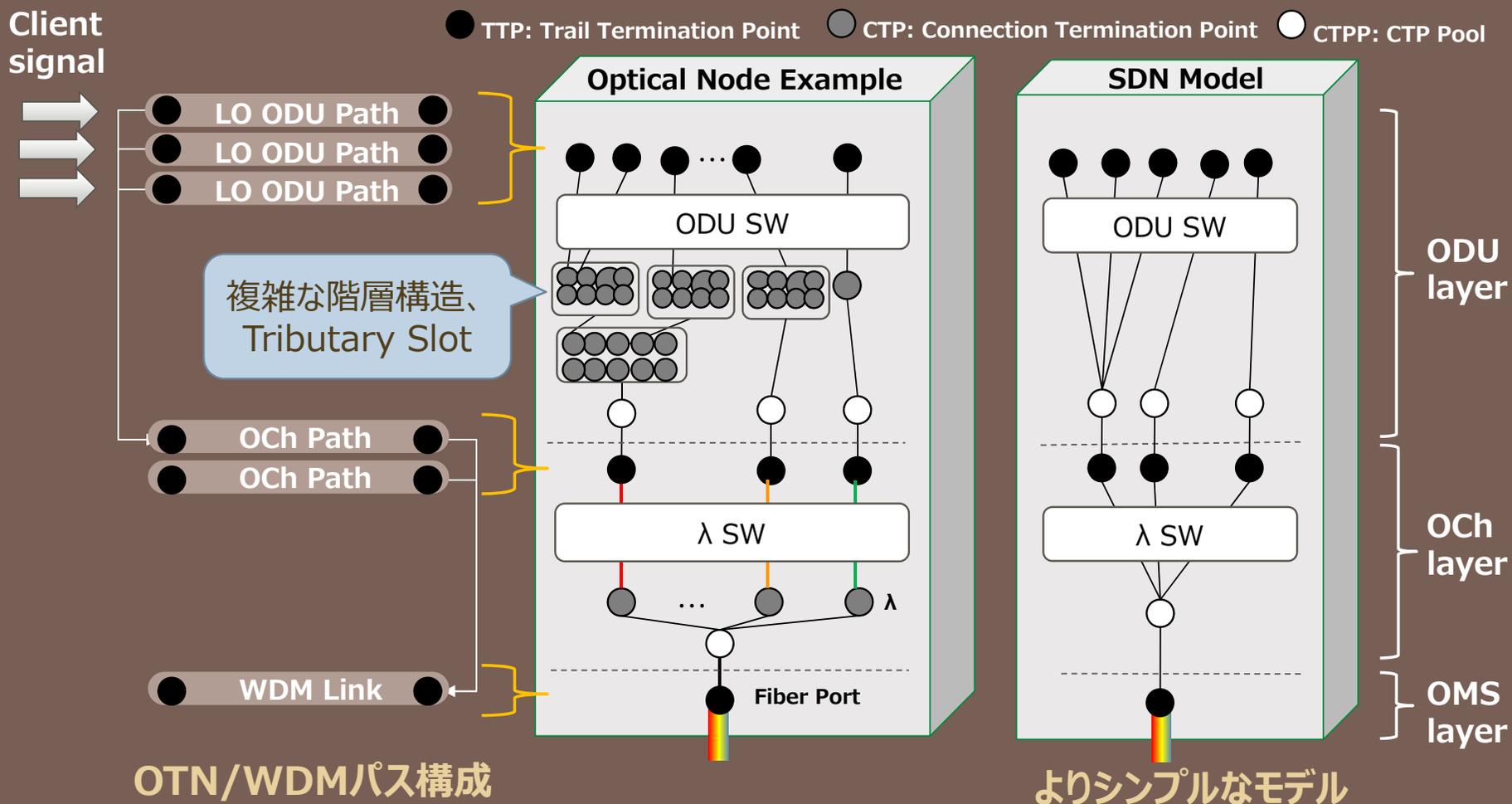
フレームワーク“ODENOS”



光コアリソース：詳細な物理情報をいかに隠蔽するか



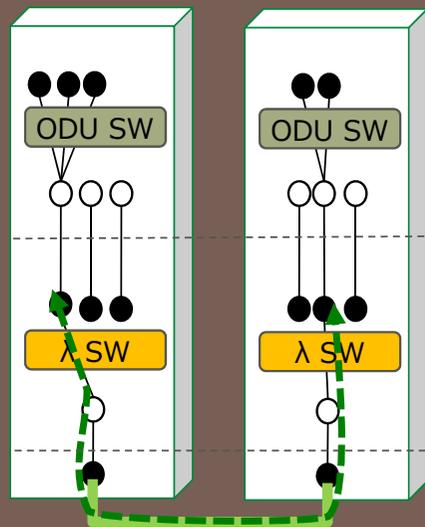
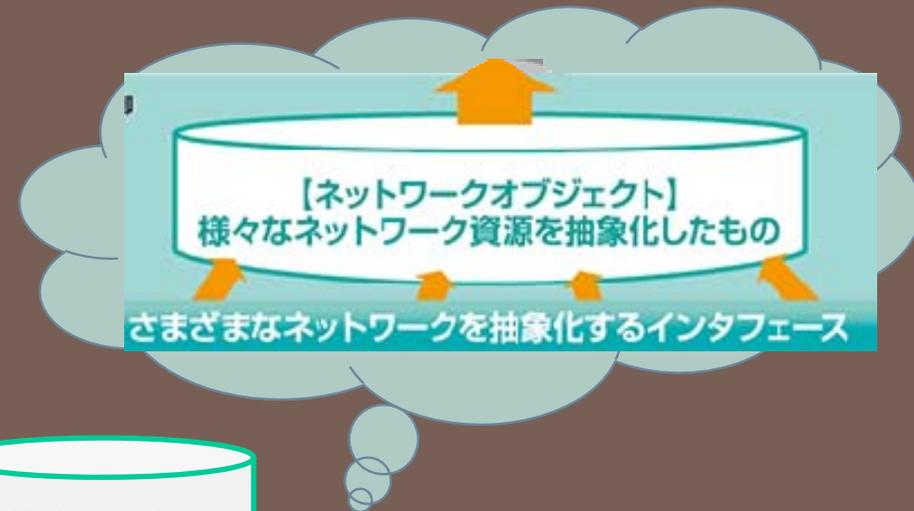
実際の装置 (OTN/WDM)



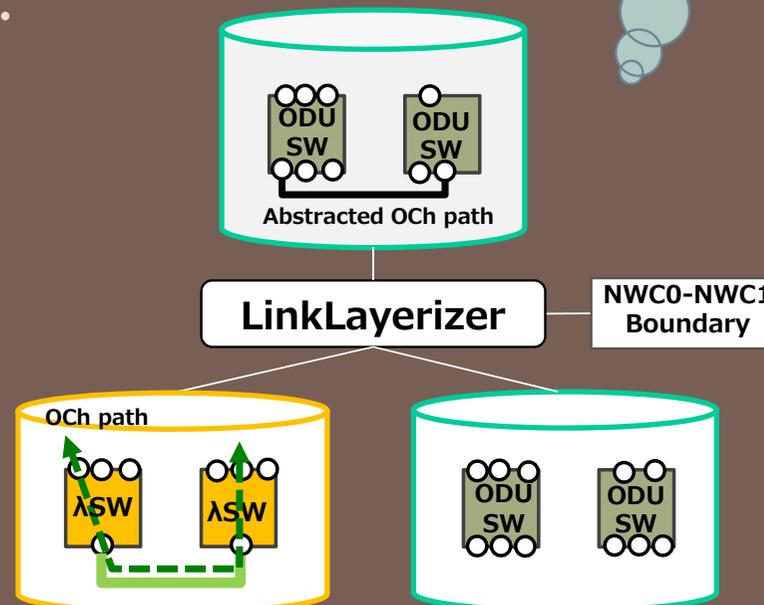
SDN OS: ODENOS

ODENOSでは、ネットワークオブジェクトとして様々なネットワーク資源を抽象化して管理

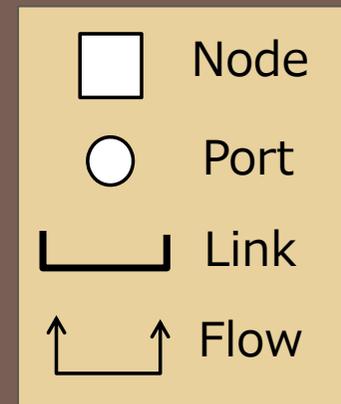
光コアリソース (OTN/WDM) を表現すると...



光コア シンプルモデル表現



光コアリソース (ODENOSモデル)



リソースの表現例 (ごく一部)

```
"links": {  
  "Link1": {  
    "attributes": {  
      "latency": "5",  
      "max_bandwidth": "10000",  
      :  
    },  
    "dst_node": "NODE2",  
    "dst_port": "CTPP3",  
    :  
  },  
  :  
}
```

Link

```
"nodes": {  
  "NODE2": {  
    "attributes": {  
      "vendor": "Fujitsu"  
    },  
    "node_id": "NODE2",  
    "ports": {  
      "CTPP1": {  
        :  
      },  
      "CTPP2": {  
        :  
      }  
    }  
  },  
  :  
}
```

Node

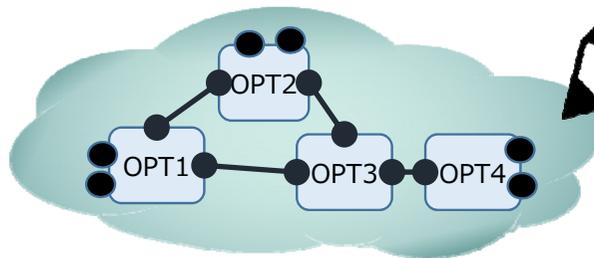
Port

SDNコントローラ動作： マルチレイヤのリソースを書きこみ

ODENOS ネットワークコンポーネント



パケットレイヤ
リソース



光リソース
(OTN/WDM)


パケット担当
マネージャ

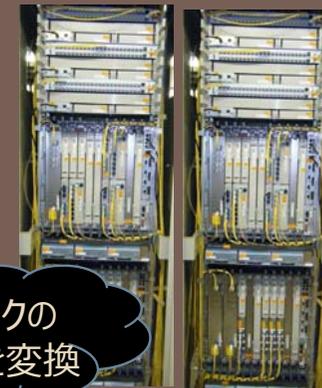

OCNRM
光担当
マネージャ

光ネット
リソース
情報

光ネットワークの
物理情報を変換

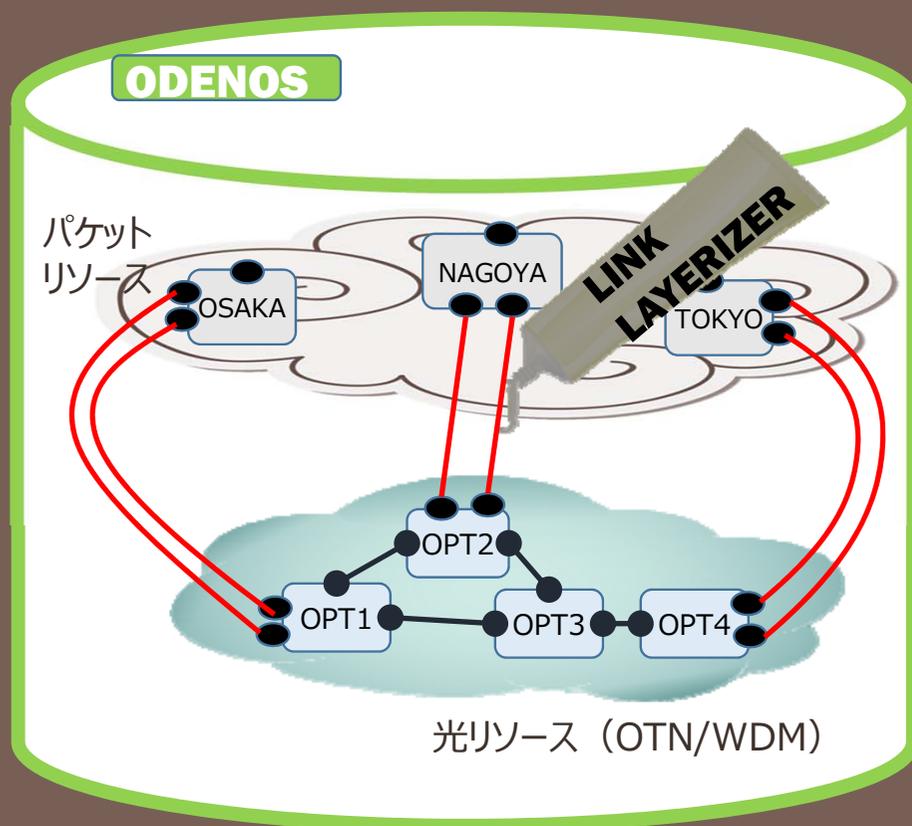
ノードリソースや
最初に設定済みの
パスなどを登録します。

これで、アプリが仮想リソース
として認識！



**OCNRM : Optical Core
Network Resource Manager**

SDNコントローラ動作： マルチレイヤ間の接続情報を持つLinkLayerizer



レイヤ間の接続、情報伝達
(マルチレイヤ対応)は
“LINK LAYERIZER”
というものを
作ってお任せしちゃいます。

SDNコントローラ動作: アプリからの要求

Application

大阪

東京

大阪-東京間に
“超・低遅延”パスを要求

ODENOS

パケット
リソース

OSAKA

NAGOYA

TOKYO

LINK
LAYERIZER

SDN Controller

event

REST

パケット担当
マネージャ

OPT1からOPT4に
超・低遅延パス
を作ればいいのね

OPT1

OPT2

OPT3

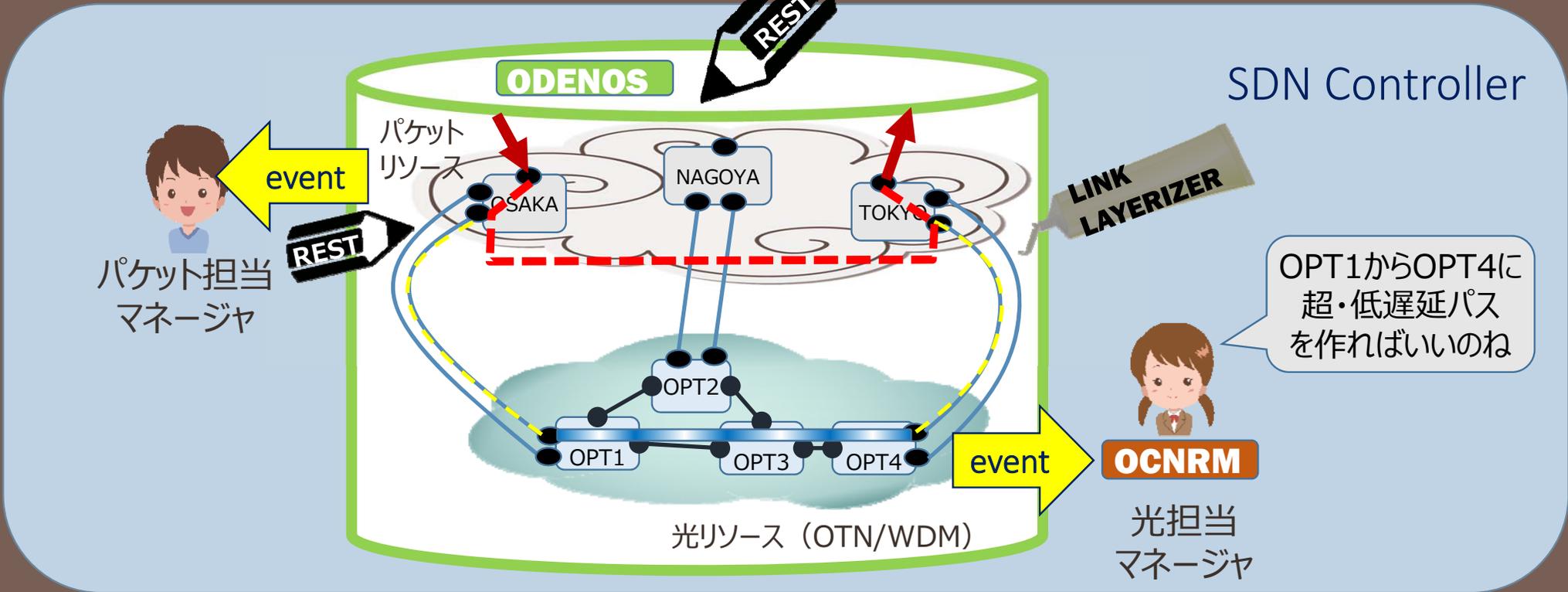
OPT4

event

OCNRM

光担当
マネージャ

光リソース (OTN/WDM)



光コアリソース設定の流れ



① 光パス割り当て (ODENOSモデル)

② 物理設定パラメタに変換
(DPID、Port#、Tributary Slot、...)

③ コマンド発行 (OpenFlow光拡張)

ONF OTWGにて標準仕様策定 (EXT-445,446)



RYU OTN
Extension

簡単なデモをご覧ください

光コア担当マネージャ、アプリ : 富士通

パケット担当マネージャ、アプリ : 日立製作所

ODENOS : NEC



O3Project公式サイト <http://www.o3project.org/ja/>

GitHub <https://github.com/o3project/>

論物変換の話

シンプルなモデルで扱っても、最終的には装置の設定を細々と行う必要がある



最も洗練された方法は??

- ★ SDN OSあるいは装置でプロトコルで定義されたオブジェクト名を扱う?
- ★ 細かいパラメタの割当ては装置がやってくれれば良い?
- ★ SDN OSのモデルでもっと細かいパラメタを抽象化して扱えるようにすべき?

まとめ

必要な時だけ広帯域/低遅延なネットワークを使える光コアSDNを実現

マルチレイヤ連携 / シンプルモデル / OpenFlowの光拡張

OSSを使った動作をご紹介 (物理装置まわりはダミー)

さらに、今年度は柔軟性をアップ、様々な帯域 (1.25Gbps x N) を指定できるFlexible ODUにも対応

是非、O3プロジェクトのデモブースにもお立ち寄り下さい

ご静聴ありがとうございました