

2018/10/19

Open Networking Conference Japan 2018

「ネットワーク図」のモデル化と モデルを起点にした自動化の可能性

TIS株式会社
萩原 学

この発表は何をするもの？

- “NW図” を作る/見る/判断するボトルネックを変えたい
 - オープンに → 人やシステムをまたいで共有できるように
- そういふのがあるのか…
 - RFCで定義されたネットワークトポロジのデータモデル
 - どんない用ができそうなのか (実装例含む)
 - 課題: ネットワークをどう表現するのがベターなのか?
- **何かやってみようかな…**

Focus?

- 話すこと
 - Network Topology Data Model
 - どんなことができる(できそう)か
- 話さないこと
 - 標準化動向
 - 開発ツール
 - 開発プロセス

背景

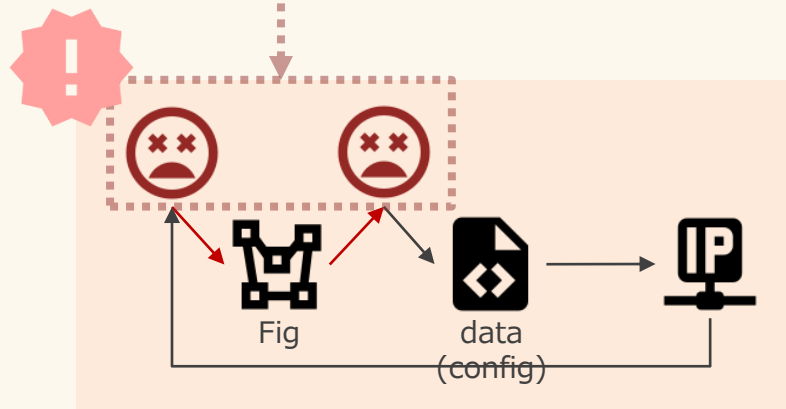
「ネットワーク図」あるある

- 図 ⇔ パラメタ/コンフィグ/設計書のマッピングが難
- 図 ⇔ 相互に連携するノードや機能間の整合性を取るのが難
- レイヤごとに図を分けていて、
複数の図のマッピングを取らないと作業が組めない
- ひとつの問題でどこまで影響がおよぶのかわからない
(図がないと・図を見ても)
- 環境を変えても図が更新されない
- 図は更新されたけど、変更前後の違いがわかりにくい
- 人・案件によって図の書き方・ルール・粒度が違う

「ネットワーク図」を核とする問題

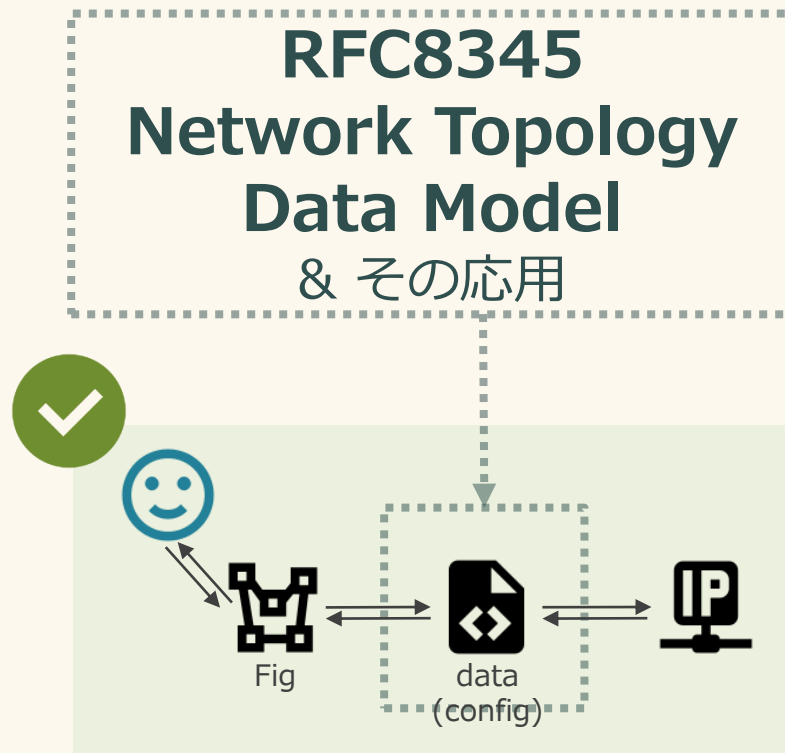
- 運用上の課題
 - 「絵」
 - 内容・書き方がまちまち
 - システム間連携が難
- スケーラビリティの課題
 - 関係性→組合せ→数が膨大
 - 人力作業ではスケールが難
 - “ネットワークは人間には早すぎる”

図から情報を読み取る
図を基に判断する
ことがボトルネックに



解決策

- 図をモデル化(標準化)
 - 図を中心にした解析
 - Validation
 - 自動化
 - Model Driven Networking



方向性

対象ネットワークのモデリングとモデルの評価

モデルを使ってやりたいこと・ユースケース

監視・ログ分析連動

インパクト分析

静的解析
シミュレーション

構成変更管理

設定自動化

運用

いま動いている環境から
情報を集める

設計・構築

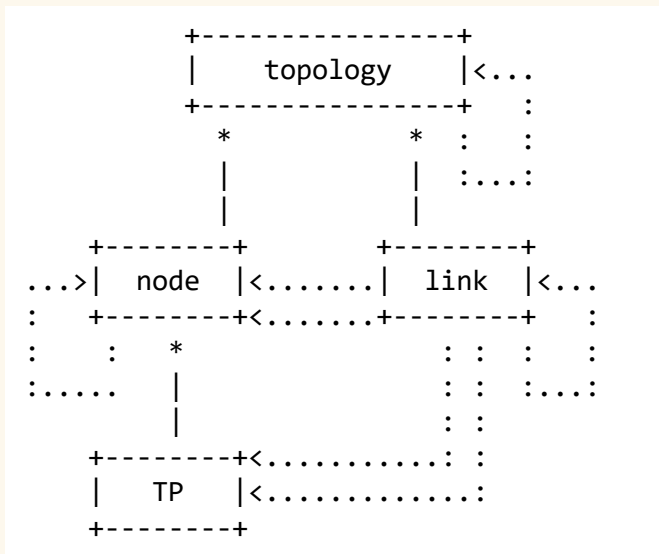
ゼロから自分で考える
最初から作る

モデルデータをどうやって作るか

データモデルと応用例

「ネットワーク図」のモデル

- RFC 8345 - A YANG Data Model for Network Topologies
<https://datatracker.ietf.org/doc/rfc8345/>
- RFC 8346 - A YANG Data Model for Layer 3 Topologies
<https://datatracker.ietf.org/doc/rfc8346/>



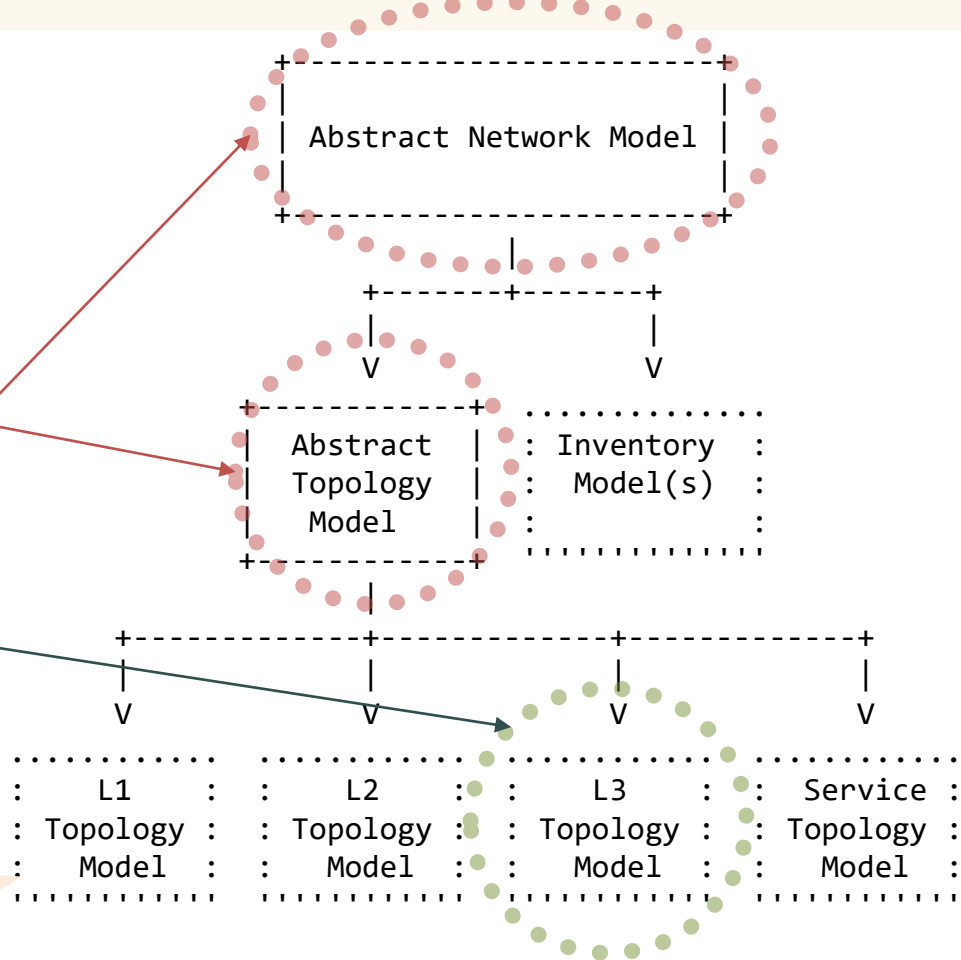
draft-medved-i2rs-topology-im-01 - An Information Model for Network Topologies <https://datatracker.ietf.org/doc/draft-medved-i2rs-topology-im/>

IETF報告会(94th横浜) SDN(網制御・管理)関連報告動画 – JPNIC
<https://www.nic.ad.jp/ja/materials/ietf-report/20151208/20151208-07-tochio.html>

RFC8345

RFC8345では
NW Model
Base Topology
のふたつを定義

RFC8346



データモデル構造と
NW階層の話は**別物**

Figure 1: The Network Data Model Structure

試してみる

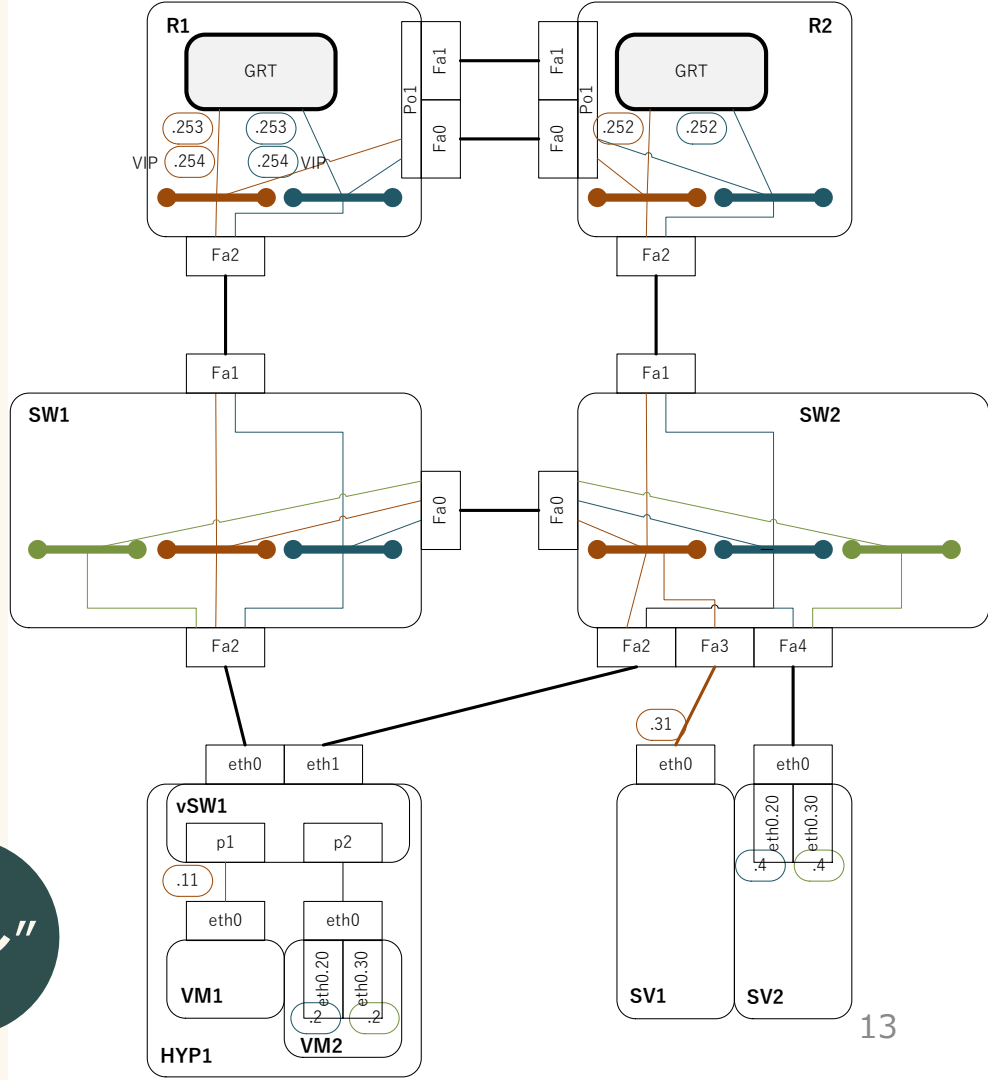
- RFCにモデル規定はあれども実例なし
 - ごくシンプルな例しかない
- 典型的なNWでデータを作ってみる
- 考えどころ
 - “アレはどう描けばいいんだろう?”

典型的なNW(図)

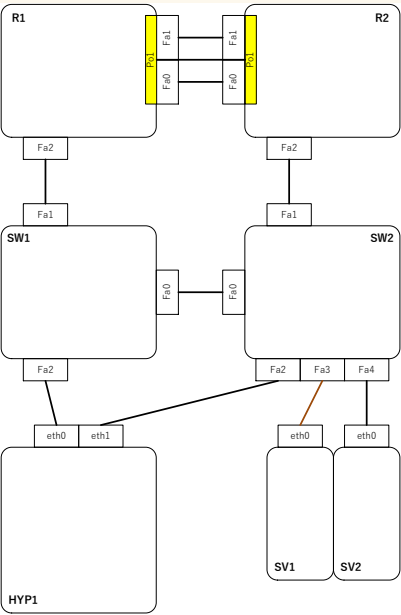
- よくある Enterprise NW
 - L2/L3SW + VLAN
 - 仮想サーバ
- 物理・論理のパターン

1→N	VRF, VLAN, Sub I/F, Hypervisor/VM
N→1	LAG, HSRP

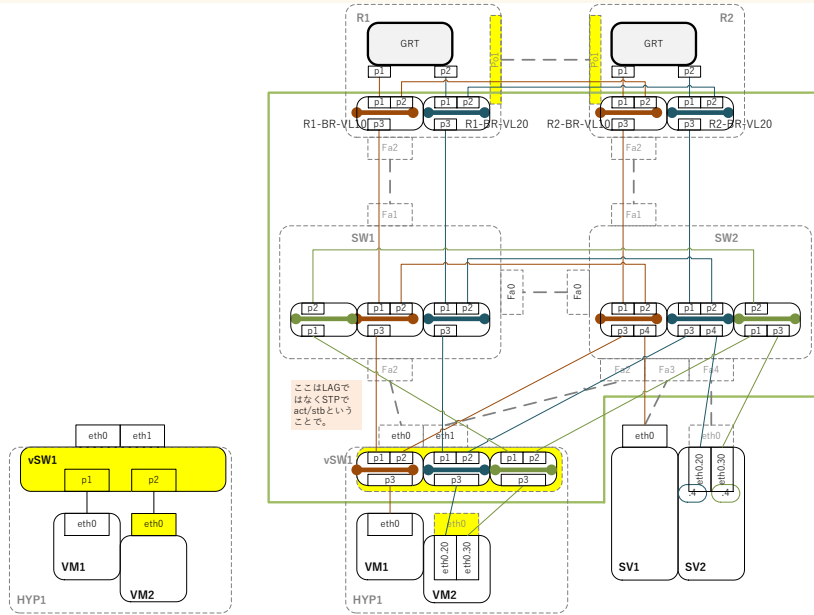
“アレ”



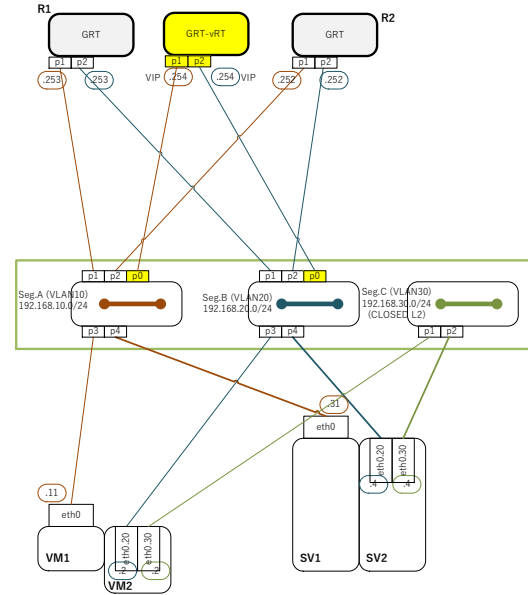
レイヤ別に図を分割



L1



L2



L3



図 → トポロジデータを作成

- YANG → XML, JSON
 - 最初は手書きで…
 - 道具の話は後述

データをもとに描画

<https://netoviz.herokuapp.com/>



Select network to display

L2 Aggregated Model ▾

Select layer(s) to display

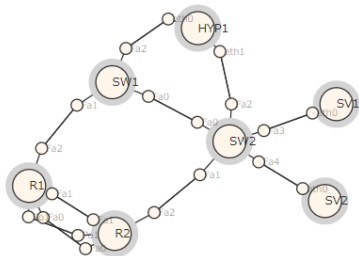
- target-L1
- target-L1.5
- target-L2
- target-L3

Legend



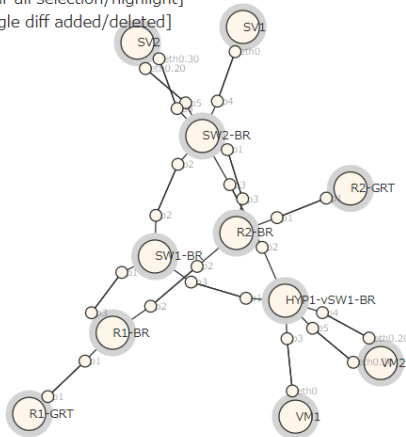
target-L1

[clear all selection/highlight]
[toggle diff added/deleted]



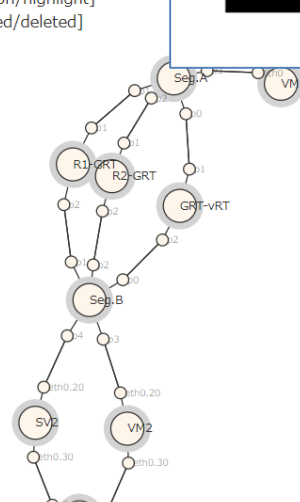
target-L2

[clear all selection/highlight]
[toggle diff added/deleted]

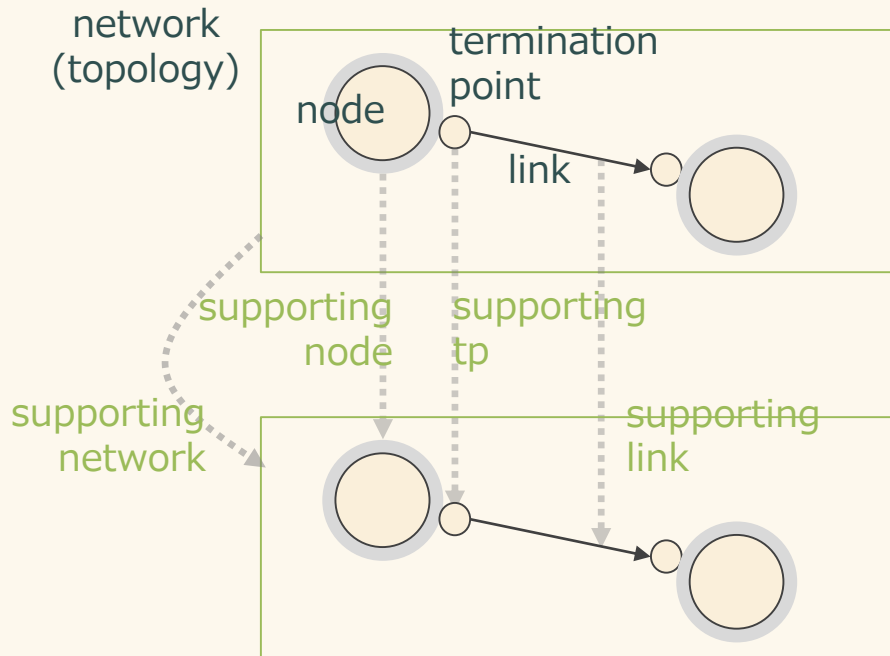
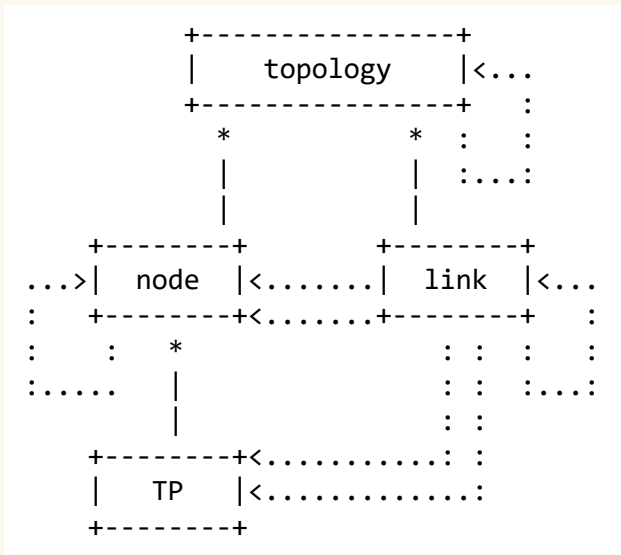


target-L3

[clear all selection/highlight]
[toggle diff added/deleted]

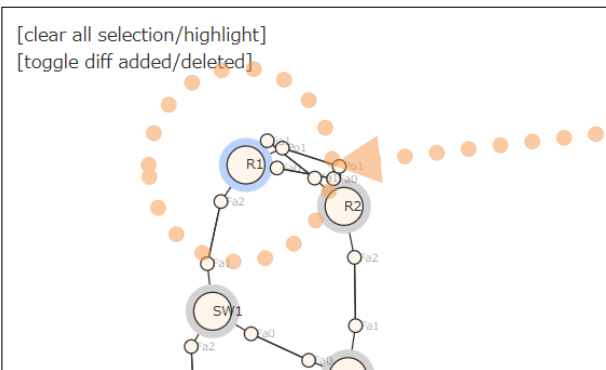


(補足) データモデルと描画オブジェクト

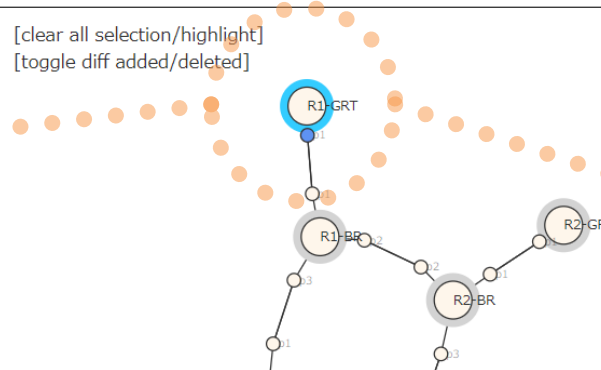


トポロジ間リンクの利用例

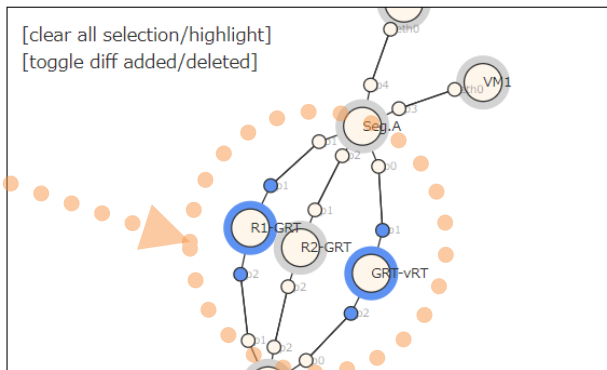
target-L1



target-L2

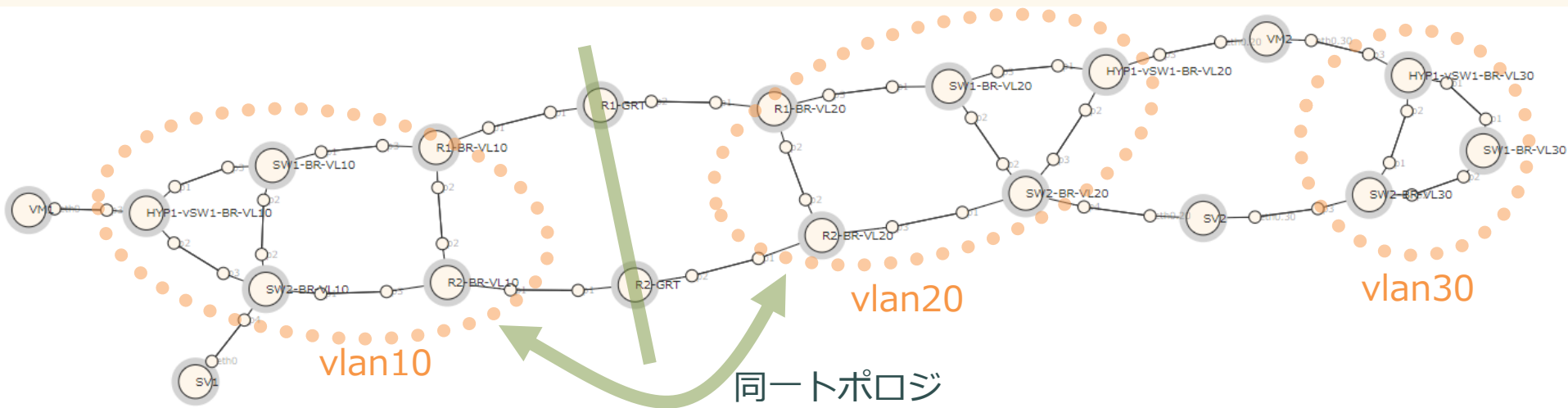


target-L3



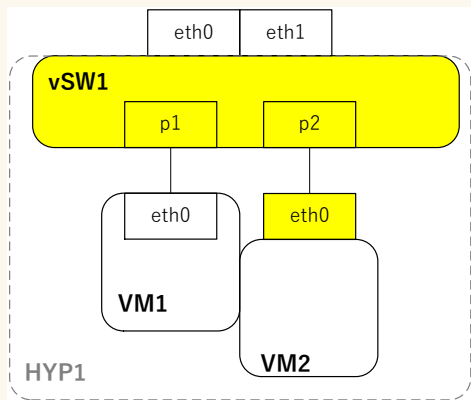
- 上層・下層にあるものを依存関係リンクをたどってハイライト
 - 依存関係リンク → supporting-[network|node|link|tp]

どう描く? (例) VLANトポロジの表現方法

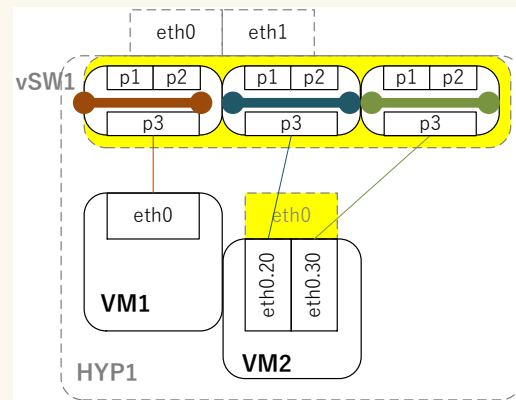


- すべてのセグメント(VLAN)を全部個別に描くのは冗長では…
 - どこまで束ねるべき?
 - ユースケース: STP計算 → STP Instance が同じものは束ねてよいはず
 - Attributeで区別する?

どう描く? (例) 仮想ノードの表現方法



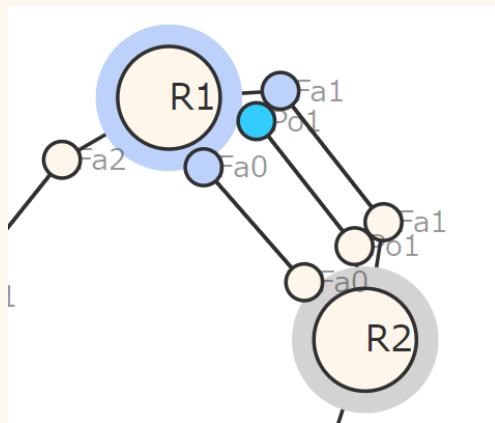
L1.5



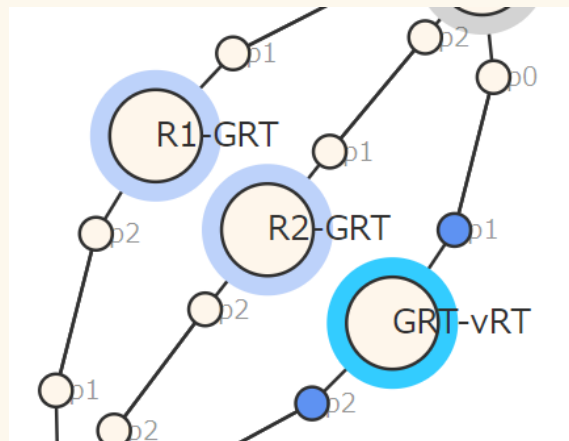
L2

- VLAN → ひとつの L2 bridge としてトポロジ作成
- VM からだと vSwitch は L1 ぽく見える。
が、Layer1/2 でわけると物理実体がないものをマップする場所がない
 - 中間レイヤ: L1.5

どう描く？ (例) 冗長機能の表現方法



L1 LAG



L3 HSRP vRT

- N→1 のパターン
 - 同一レイヤ内での
仮想オブジェクト表現



[RFC8345] 4.4.6. Multihoming and Link Aggregation

Links are terminated by a single termination point, not sets of termination points. Connections involving multihoming or link aggregation schemes need to be represented using multiple point-to-point links and then **defining a link at a higher layer that is supported by those individual links.**

プラクティスの必要性と課題点

- 複数の表現方法…適しているのはどちらか?
 - 図の表現方法はひとつではない
 - **ユースケース評価 → ベストプラクティス**
- **壁**
 - データ作成の手間
 - 試行錯誤が必要 ⇔ データを作るのがきつい
 - 変化を見る(変更確認)のが難しい

「壁」に対する
道具立て

データ作るのがきつい

<https://github.com/corestates55/netoviz/blob/develop/dist/model/target3.json>

```
{
  "ietf-network:networks": {
    "network": [
      {
        "network-types": {},
        "network-id": "target-L1",
        "node": [
          {
            "node-id": "R1",
            "ietf-network-topology:termination-point": [
              {
                "tp-id": "Fa0"
              },
              {
                "tp-id": "Fa1"
              },
              {
                "tp-id": "Fa2"
              },
              {
                "tp-id": "Po1",
                "supporting-termination-point": [
                  {
                    "network-ref": "target-L1",
                    "node-ref": "R1",
                    "tp-ref": "Fa0"
                  },
                  {
                    "network-ref": "target-L1",
                    "node-ref": "R1",
                    "tp-ref": "Fa1"
                  }
                ]
              }
            ]
          }
        ]
      },
      {
        "node-id": "R2"
      }
    ]
  }
}
```

JSON Data
2600+

行

||

DSL
500+

行

```
require 'netomox'

def make_target_layer1
  Netomox::DSL::Network.new 'target-L1' do
    node 'R1' do
      (0..2).each { |n| term_point "Fa#{n}" }
      term_point 'Po1' do
        support %w[target-L1 R1 Fa0]
        support %w[target-L1 R1 Fa1]
      end
    end

    node 'R2' do
      (0..2).each { |n| term_point "Fa#{n}" }
      term_point 'Po1' do
        support %w[target-L1 R2 Fa0]
        support %w[target-L1 R2 Fa1]
      end
    end

    node 'SW1' do
      (0..2).each { |n| term_point "Fa#{n}" }
    end

    node 'SW2' do
      (0..4).each { |n| term_point "Fa#{n}" }
    end

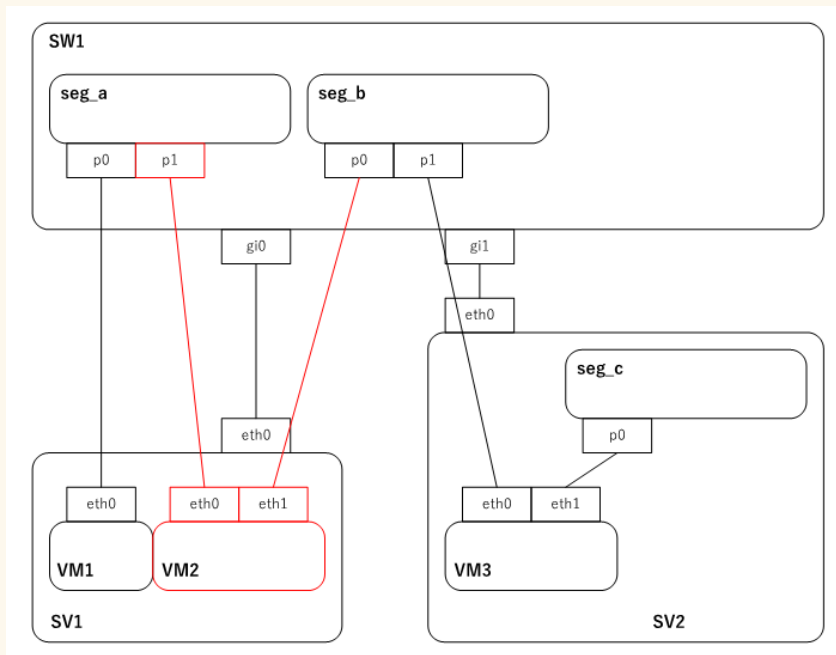
    node 'HYP1' do
      (0..1).each { |n| term_point "eth#{n}" }
    end

    node 'SV1' do
      term_point 'eth0'
    end

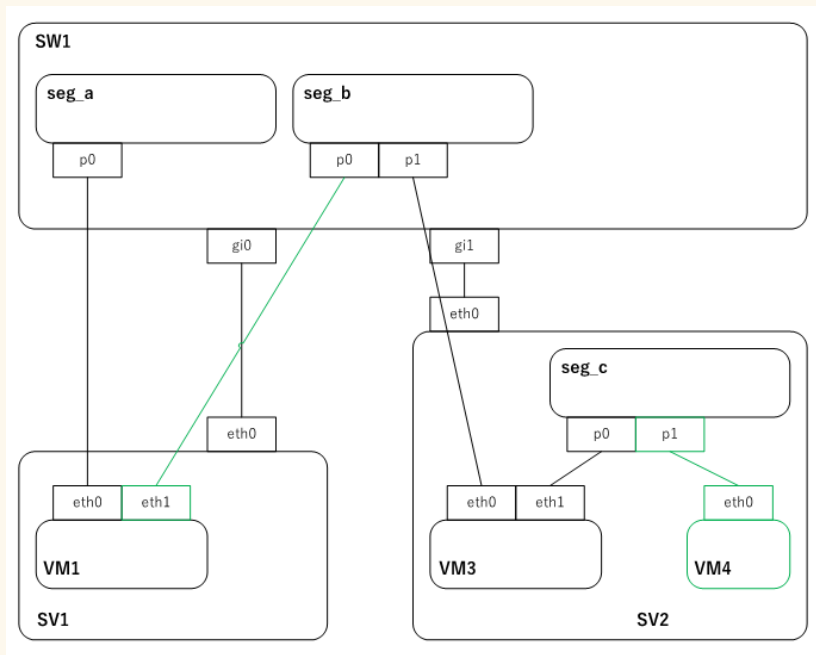
    node 'SV2' do
      term_point 'eth0'
    end
  end
end
```

https://github.com/corestates55/netomox/blob/develop/vendor/model_defs/target3.rb

変化を見るのが難しい



before

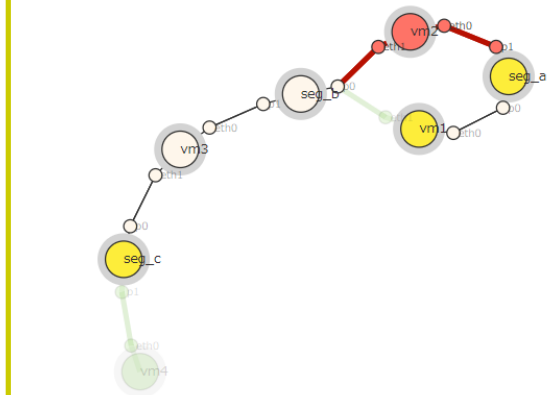


after

オブジェクト単位で差分比較

layer3

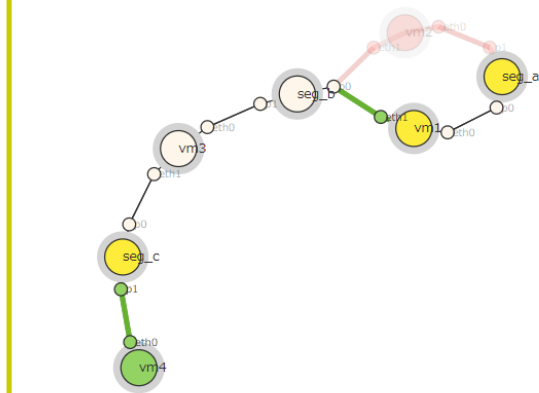
[clear all selection/highlight]
[toggle diff added/deleted]



before

layer3

[clear all selection/highlight]
[toggle diff added/deleted]



after

今後の課題とまとめ

検討ポイント

- より規模が大きな環境での「見せ方」
 - 情報収集と可視化の工夫 … 「絵」とは違う見せ方?
- 複数レイヤのマッピング
 - Tunnelなど overlay/underlay のマッピング
 - クラウド~オンプレなど
異なる管理形態のものが連携するシステム
- 直接「ネットワーク」でないものとの関係性?
 - “アプリ” や “サービス” などとの対応関係?

直近のねらいどころ

対象ネットワークのモデリングとモデルの評価

収集したデータを元に
L3トポロジ可視化
時系列変更差分の可視化

GraphDB応用

Model Diff

運用

**Fixpoint
Kompira 連携**

設計・構築

TISとフィックスポイント、「標準トポロジモデルを応用した
ネットワーク構成の可視化に関する研究」を共同で開始
https://www.tis.co.jp/news/2018/tis_news/20181017_1.html

今後の課題

仲間
探し中

- 実運用を通じた評価やフィールドテスト
 - 誰と・どこで・どうやって?
- プラクティス
 - 標準モデル → いろいろな人がいろいろなユースケースを
 - 新しいユースケース・新しいネットワークモデル…
 - 「おもしろそうだからやってみよう」
 - まずは「おもしろそう」と思ってもらえるものを

まとめ

- 「ネットワーク図」を起点にした自動化
 - RFC8345 ネットワークトポロジのデータモデル
 - いろいろなレイヤ・ドメイン・技術にまたがる関係性の表現
- トポロジモデルの応用を考えよう
 - ユースケースとモデル (表現方法)
 - 道具立て
- **いろんなプラクティスを!**

参照

- TISとフィックスポイント、「標準トポロジモデルを応用したネットワーク構成の可視化に関する研究」を共同で開始
https://www.tis.co.jp/news/2018/tis_news/20181017_1.html
- **netomox**: Network Topology Modeling Toolbox
<https://github.com/corestate55/netomox>
 - データ定義DSL
 - Topology Data (JSON) の CLI diff
- **netoviz**: Network Topology Visualizer
<https://github.com/corestate55/netoviz>
 - <https://netoviz.herokuapp.com/> (demo)
 - Topology Data (JSON) Visualizer