



# 個人デバイス間における ソフトウェア定義のオーバーレイ仮想ネットワーク - PerSoNet -

---

Saumitra Aditya, Kensworth Subratie, Renato J Figueiredo

©2018 IEEE

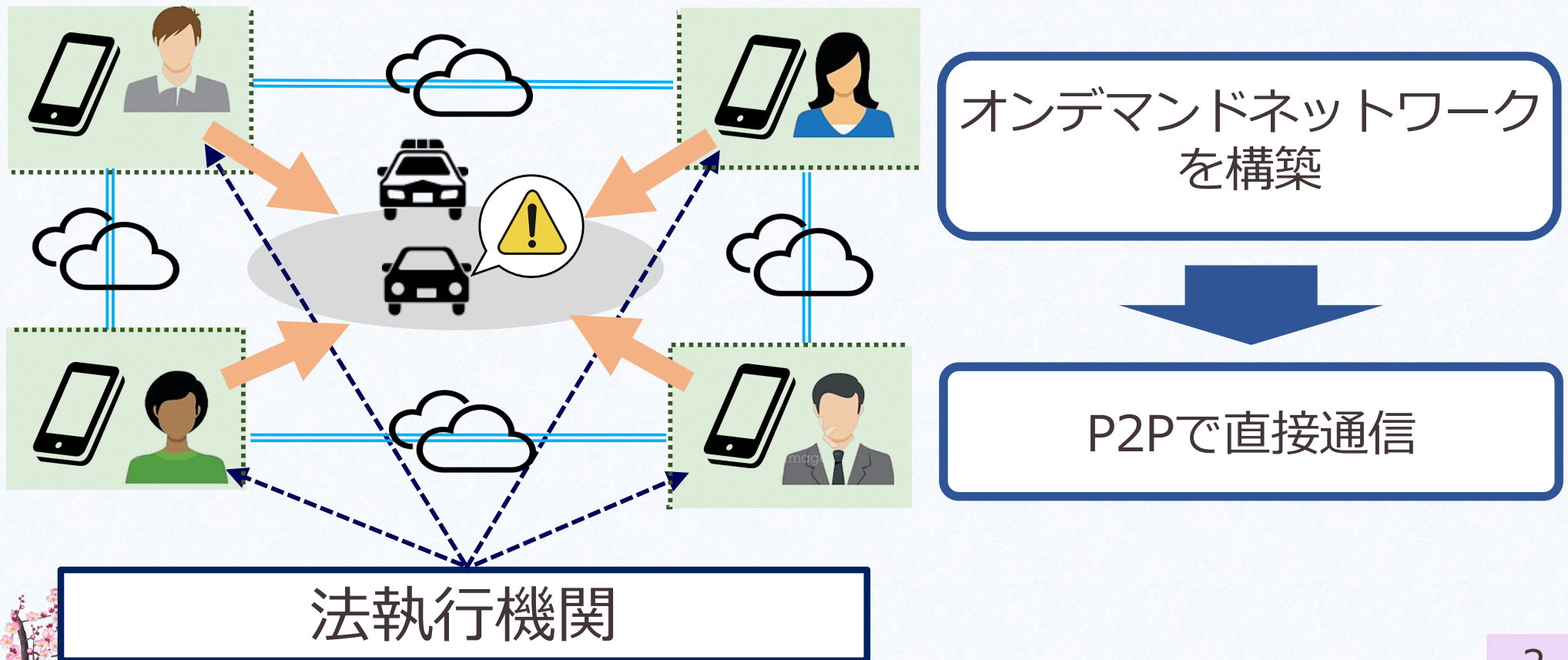
**内藤研究室**  
**K18039 B4 後藤 廉**

**2021.06.02 論文ゼミ**

# OSNを活用したオンデマンド通信



- デバイスや専用リソースは地理的に分散して存在
- エッジで処理されたデータを共有して課題に対処
- OSNで確立された信頼関係を活用
  - ✓ OSN: Online Social Network



# OSN及び分散コンピューティングの課題



フォグ, エッジコンピューティングパラダイムは  
基本的な通信ファブリックが前提



OSNで確立された信頼関係はピアの発見に活用できるが  
エッジバイス間での直接通信は不可能



ユーザ自身がデバイス間のP2Pネットワークを  
管理することは非現実的



これらの課題を解決する  
新たなネットワークを構築する必要がある！





## ユーザとエッジのリソースをP2Pでシームレスに接続する 新しい仮想ネットワーク **PerSoNet** の提案

- 名前, アドレス空間, デバイスの発見, アクセス制御を抽象化
  - 3層構造のオーバーレイネットワークレイヤを構成
  - OSNが提供するメッセージング機能の利用
- ピアデバイス間のプライベートリンクをシームレスに作成, 管理
  - トンネリングを利用したL2 over L3ネットワークによって論理的に接続
  - SDNスイッチを用いてIPパケットをオーバーレイリンク上でルーティング
- 既存のシステムを変更せずに実行可能
  - OS層でのDNSの名前解決を利用し, 同様の3レベルスキームを採用
  - 通信プロセスはユーザの介入を必要としない完全自己構成



# PerSoNet 3層構造アーキテクチャ



## ■ ソーシャルネットワーク層

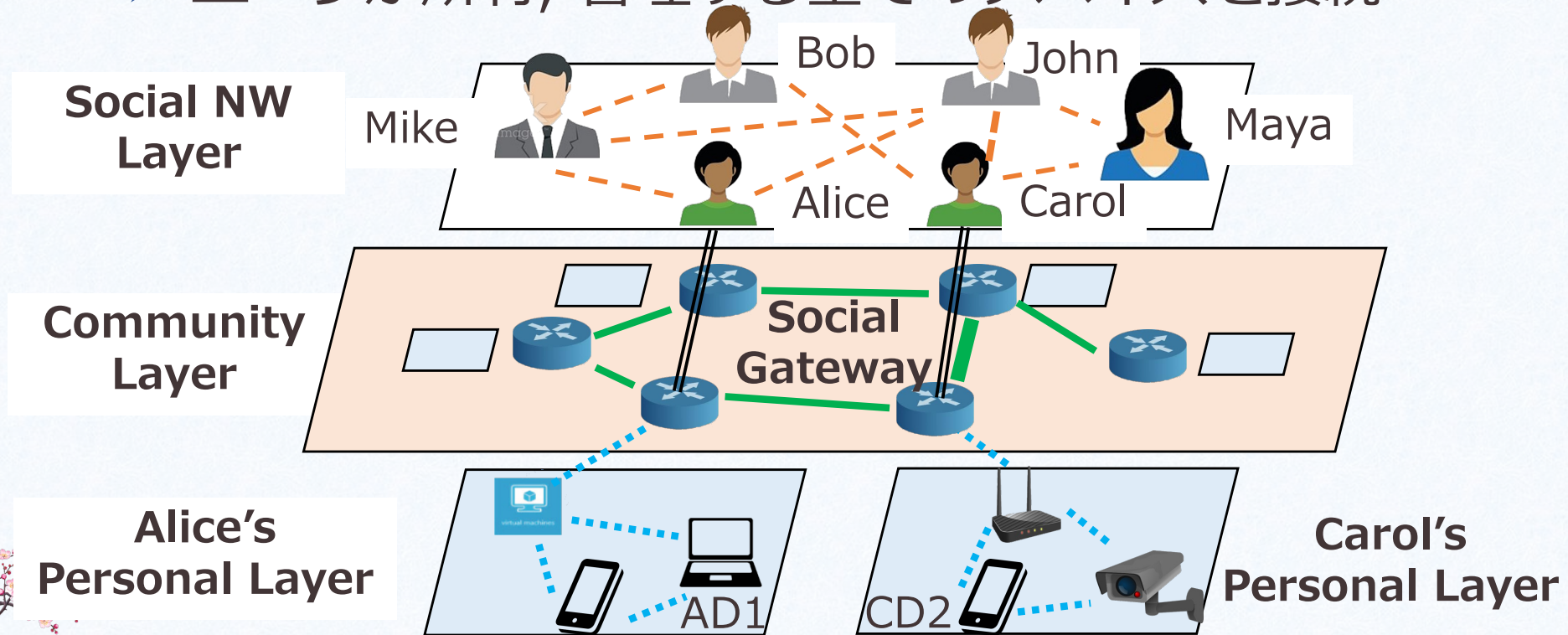
- OSNベースでピア間を管理

## ■ コミュニティ層

- ピア間の関係をマッピングするトポロジを形成

## ■ パーソナル層

- ユーザが所有, 管理する全てのデバイスを接続



# PerSoNet 構成モジュール



## ■ デバイスコンポーネント

- VPNトンネルによるエンド間の直接通信とDNS処理を実行

## ■ ゲートウェイコンポーネント

- 異なるピアデバイス間の接続を組織化
- GM: Gateway Manager
  - ✓ ネットワーク構成指令をOpenFlow SDNルールに変換
  - ✓ マッピングテーブルの維持・管理
- SLCC: Social Layer Communication Client
  - ✓ GM, デバイス・アドミニストレーションコンポーネント間のブリッジとして機能
  - ✓ 相手GMとの通信を制御

## ■ アドミニストレーションコンポーネント

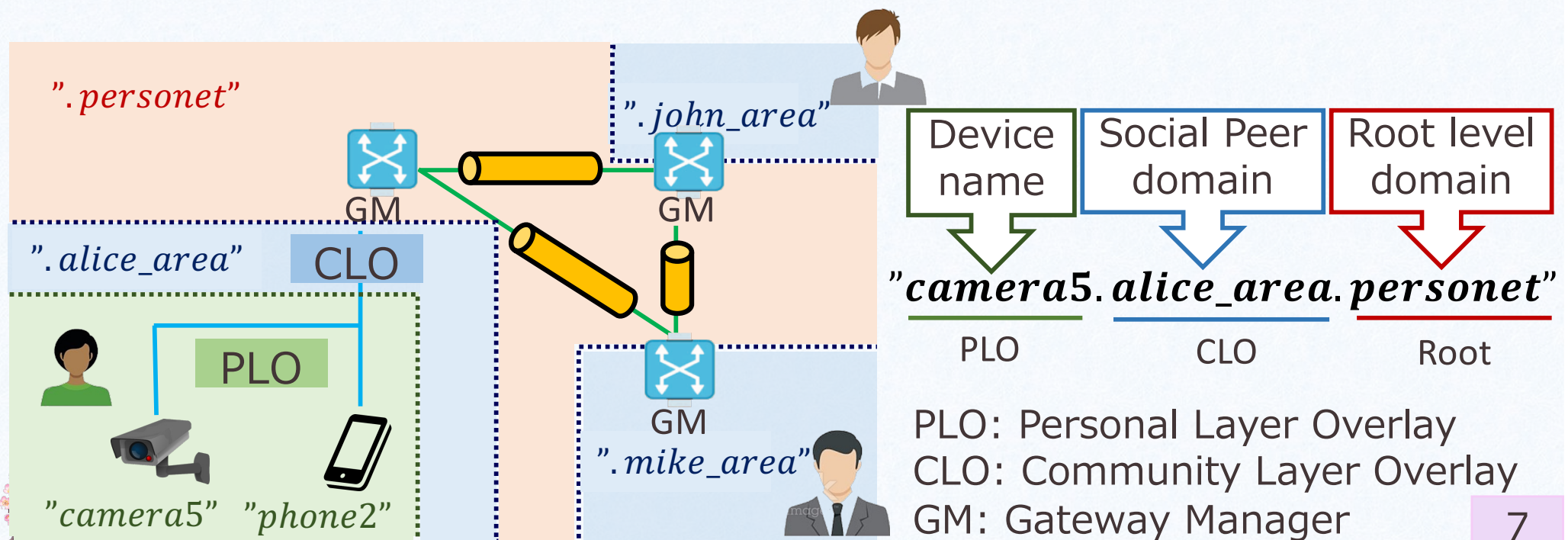
- ユーザにPerSoNetの制御インターフェースを提供



# 名前管理



- DNS階層と同様の3レベルスキームを採用
  - **第3レベル**：PerSoNetに所属するデバイスを識別
    - ✓ 未使用のルートドメイン *".personet"* を使用
  - **第2レベル**：デバイスのソーシャルドメインを識別
    - ✓ OSN上で固有のユーザ名を使用
  - **第1レベル**：個々のデバイスを識別
    - ✓ ユーザ自身が一意なデバイス名を設定

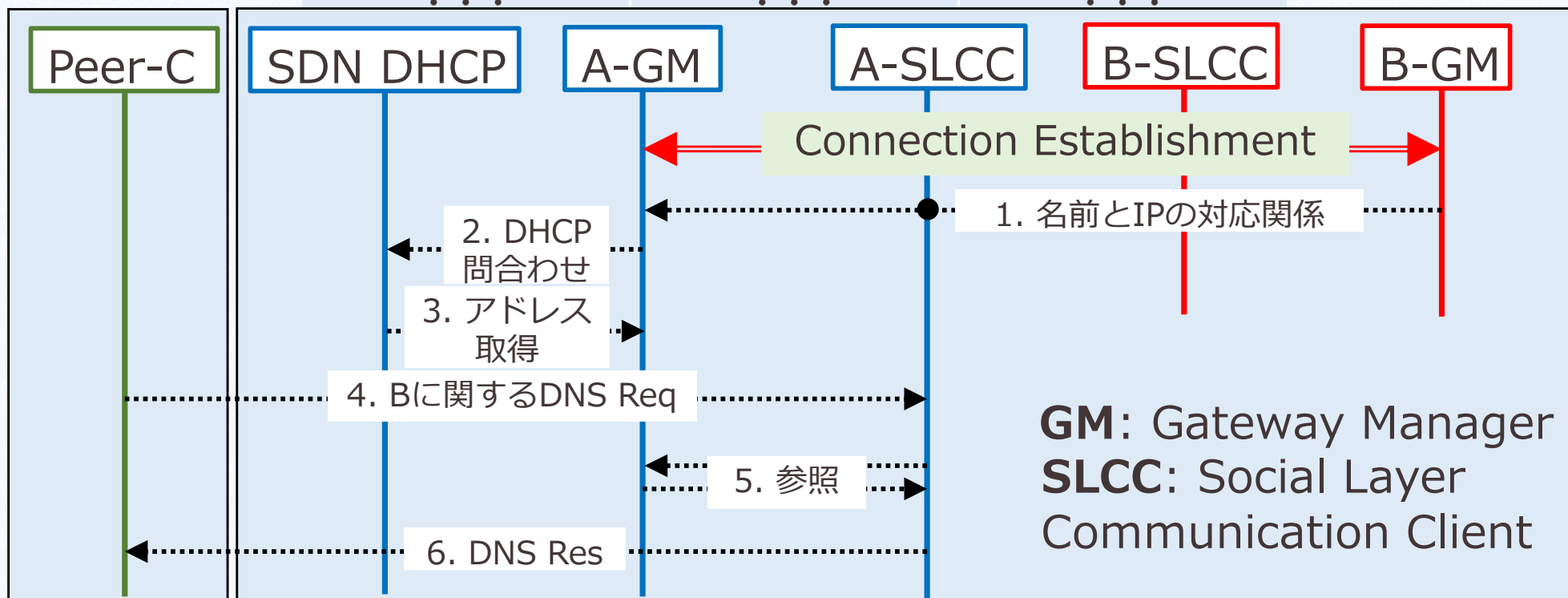


# アドレス管理と名前解決



- ① 相手機器のDNS名をIPアドレスに解決
- ② DNSクエリに対するレスポンスを生成

Device name	Remote address	Mapped address
B: pc1.bob.personet	192.168.1.105/16	192.168.1.201/16
...	...	...



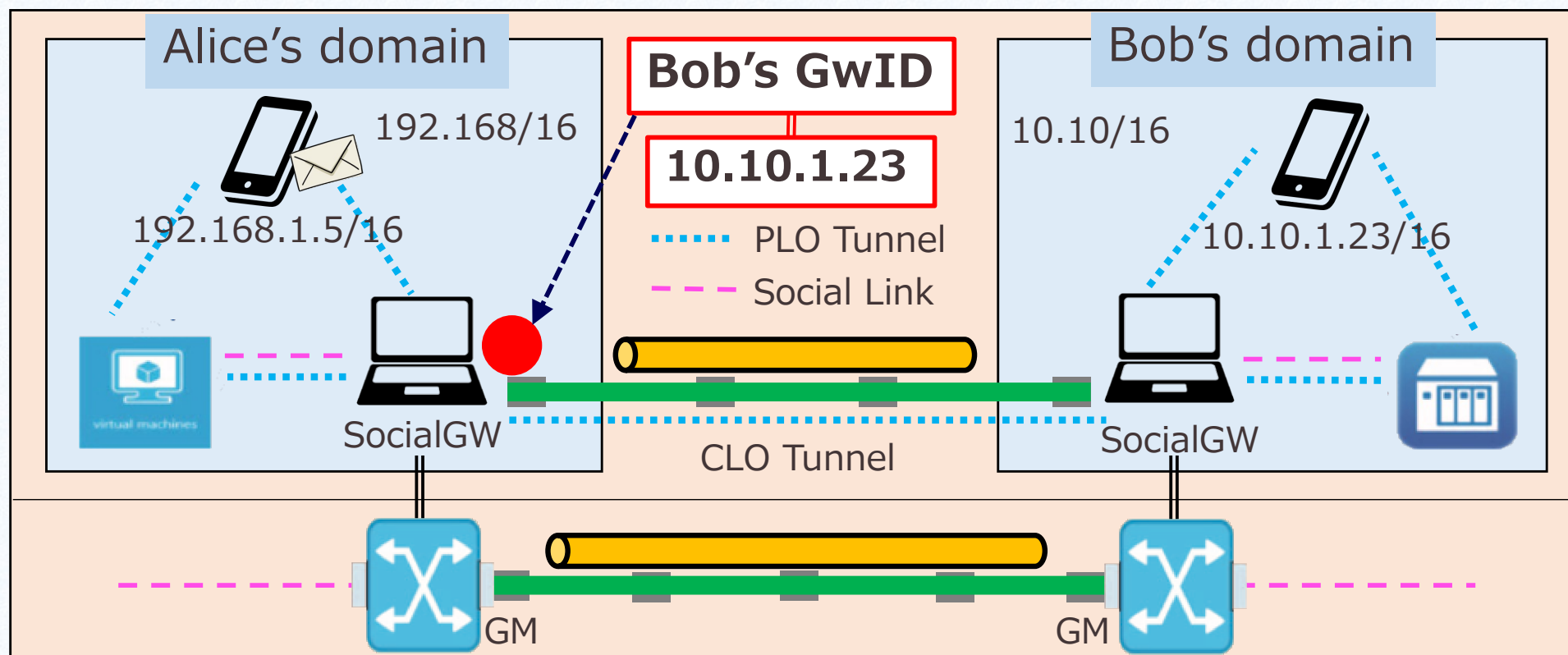
PerSoNetデバイスに対するDNSクエリは、動的にマッピングされたドメイン固有のIPアドレスに解決



# SDNフローによるP2P通信



- ① 送信先と送信元でアドレスのサブネットドメインを統一
- ② ポート検出とCLOでのルーティング
- ③ トンネリングによるエンド間通信の実行



SDNベースのプロセスは完全自己構成であり、  
ユーザが手動で介入することなくシームレスに実行可能





## ■ 発見可能性

“他のソーシャルピアによって自身のデバイスが発見されることを許可できるか”

➡ OSNのメッセージング機能を利用して拡張

## ■ 接続性

“PerSoNet仮想ネットワーク上でパケットを送受信することを許可できるか”

➡ ソーシャルゲートウェイはSDN構成のファイアウォールとして機能

ex:) Jane's rule

Source peer	Destination device name	Policy
Alice	phone1	ACCEPT
Bob	pc3	DROP



# PerSoNet プロトタイプ実装



一時的なアドホックネットワーク構築の際は  
手動による介入を最小限に抑えることが求められる



タスクテンプレートを使用して抽象化

1. **タスクに必要なデバイスとアプリケーション**
2. **仮想リソースへのアプリケーションマッピング**
  - ➡ OSNが提供する認証チャンネルを介して  
オーケストレーションメッセージを交換
3. **物理-仮想リソース間を接続するためのネットワークトポロジ**
  - ➡ コミュニティメンバー間のソーシャルゲートウェイを  
セキュアなCLOトンネルで接続

タスクの開始者はリポジトリから自分の要件に適した  
タスクテンプレートを引き出して設定





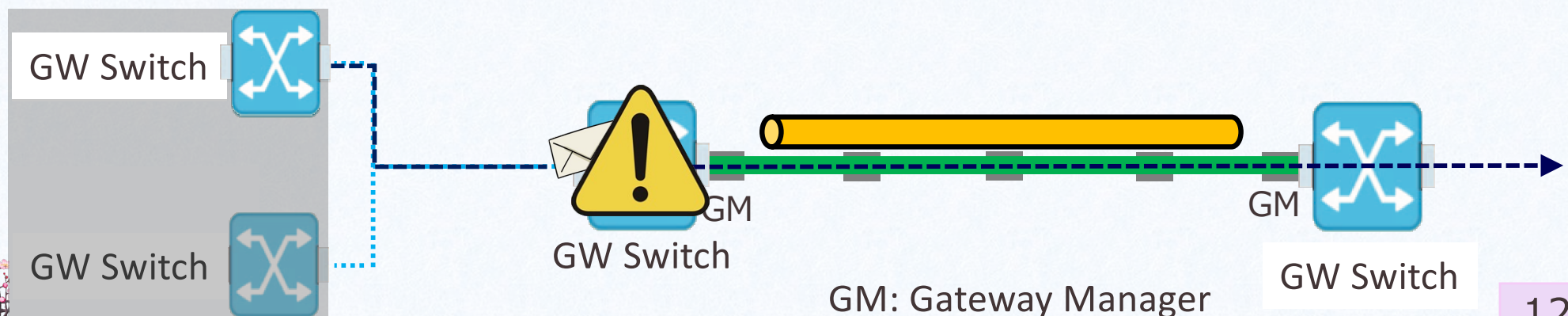
## 懸念される障害に対する対策

### ■ OSNサービス

- ▶ XMPPを実装するメッセージングサービスを利用
  - ✓ Ejabberd, Ignite, Openfire, …etc.
- ▶ フェデレート構造を採用してサービスを冗長に展開

### ■ CLOを接続するソーシャルゲートウェイ

- ▶ 複数のSDNコントローラをアクティブ・スタンバイ構成でデプロイ
- ▶ 障害発生時にスタンバイコントローラがアクティブコントローラの情報自動的に引き継ぐことでフローを維持



# 評価：タスクオーバーレイのセットアップ



PerSoNetの実用性は、“如何に速くソーシャルゲートウェイ間でオーバーレイを作成し、P2P通信を開始できるか” に依存



あるユーザがタスクを開始してから、スマートタスクオーバーレイの作成に成功するまでの経過時間を測定

Resources	Cameras	Compute	Tunnels	Time
20	15	5	19	151.4 s
40	36	4	39	170.8 s
60	50	10	59	177.2 s

ソーシャルピアデバイス増加に伴うP2P通信開始に要する時間は許容可



# 評価：TCPスループット及びレイテンシ



## ■ クライアントとサーバが同じクラウドインフラに存在する場合

	Original	OpenVPN	PerSoNet
Bandwidth	941 Mbps	351 Mbps	248 Mbps
Latency	0.135 ms	0.669 ms	2.66 ms

➤ 帯域幅, レイテンシ共にOpenVPNに少々遅れを取る

## ■ クライアントとサーバが地理的に分散して存在する場合

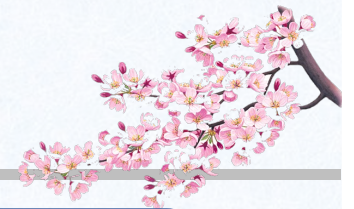
	Original	OpenVPN	PerSoNet
Bandwidth	NA	1.12 Mbps	25.5 Mbps
Latency	NA	304.06 ms	29.27 ms

➤ 帯域幅, レイテンシ共にOpenVPNよりも性能を発揮

リレーベースのオーバーレイよりも  
P2Pトンネルの方が有効に機能



# 評価：ビデオストリーミング性能



市販のWi-Fiを利用してPerSoNetトンネル上でカメラのビデオフィードをリアルタイムでストリーム処理

- ▶ デバイス：LAPTOPx4 Core i5 8GB (内2台 ソーシャルゲートウェイ)
- ▶ ソフトウェア：VLC (ハーフHD 1280×720)

	Physical WiFi	PerSoNet
Bandwidth	37 Mbps	28.8 Mbps
Latency	3.6 ms	17.3 ms
FPS (Video)	22.5	19.37
Frame drop (Video)	0.8 %	1.3 %
FPS (Webcam)	17.6	17.2
Frame drop (Webcam)	0.7 %	1.5 %

PerSoNetは物理ネットワークに近い性能を提供可能



# まとめ



フォグ, エッジコンピューティングにおいてユーザとエッジのリソースをP2Pでシームレスに接続するネットワークが必要



OSNを活用した個人デバイス間におけるソフトウェア定義のオーバーレイ仮想ネットワーク **PerSoNet** を提案



物理ネットワークに近い適切な性能を発揮



フォグ, エッジコンピューティングパラダイムの為のネットワーク層基盤の提供が見込める



以下, 参考スライド





**“地理的に離れたネットワークのサブドメインを統一し、かつその中で重複しないアドレスを持つことができれば、仮想IPアドレスを持ちいらなくてもSDNフロー(トンネリング)を用いて実IPアドレスでP2P通信を実現可能”**



# JSON定義 タスクテンプレート



## ■ タスク例：行方不明のペットを探す

```
{
  "task_name": "locate missing cat",
  # social gateway ID
  "task_initiator": "alice@xmpp10.org",
  # unique identifier
  "task_identifier": "ZJ8S3GE800T6O2W3RB969V4G",
  "devices_needed": ["camera", "linux_computes"],
  # geographical coordinates
  "coordinates": {"lat": 29.651634, "lng": -82.324829},
  # search radius
  "radius": 5,
  # orchestration policy
  "orchestration": "https://github.com/xyz/abc/missing_pet.py"
  # folder containing images of pet
  "target-folder": "/home/sam/pretty-cat"
}
```



# タスクオーバレイの実行手順 その1



- ① タスクを開始する個人(イニシエータ)は自分のアドミンコンポーネントを使ってリポジトリからタスクテンプレートをダウンロードし, 必要な情報を入力
- ② アドミニストレーションコンポーネントがソーシャルIDを入力してタスクの一意的識別子を作成し, タスクテンプレートで指定されたオーケストレーションポリシーを取り込む
- ③ アドミニストレーションコンポーネントがタスクを開始すると、イニシエータの全友人リストを取り込んで、「タスク識別子」「開始者のソーシャルID」「要求されたデバイスの種類」を含むリソース勧誘要求を送信  
(これに続いて、デバイス勧誘の応答を待つためのタイマが開始される)
- ④ ピアのアドミニストレーションコンポーネントで勧誘要求を受信し、もしピアが要求されたタイプのリソースを持っており、かつ自分のリソースをタスクに提供したい場合は、アドミニストレーションコンポーネントを使用して、イニシエータからのトンネル勧誘要求の着信を許可するようにソーシャルゲートウェイに指示  
(これに続いて、「デバイスの名前」「ソーシャルゲートウェイのソーシャルID」「デバイスの機能」「タイプ (位置情報)」を含むデバイス勧誘応答が送信される)
- ⑤ イニシエータではリソース勧誘タイマが切れると、アドミニストレーションコンポーネントが受信した勧誘応答からすべてのリソース情報をまとめ、オーケストレーションポリシーの入力として提供



# タスクオーバレイの実行手順 その2



⑥ オークストレーションポリシーはタスクオーバレイに適したネットワークトポロジを選択するために、デバイスの能力の様々な側面を考慮するようにプログラムすることができる

- プロトタイプの実装では単純な層状の負荷分散されたネットワークトポロジのポリシーを持っており、リーフ層はカメラノードで構成される
- 利用可能なカメラノードと計算ノード（ビデオ処理の負荷を管理する）の数に応じて、カメラノードは画像認識アプリケーションが配置される最後の層の計算ノードに比例して接続される
- マッチしたフレームとメタデータ情報は最終的にルートノードに集約され、そこから配置されたアプリケーションがイニシエータにアラート通知を送ることができる

⑦ インメモリグラフの形でトポロジが作成されると、CLOトンネルを作成するためにトンネル勧誘リクエストがピアのソーシャルゲートウェイに送信される

- 例えば、イニシエータはAliceとBobのソーシャルゲートウェイ間にCLOトンネルを作成するために、両者にトンネル勧誘リクエストを送信する
- このリクエストを受け取ったAliceとBobは、リクエストされたデバイスとポートへのアクセスを互いに許可し、それによってDNSの名前解決が行われる



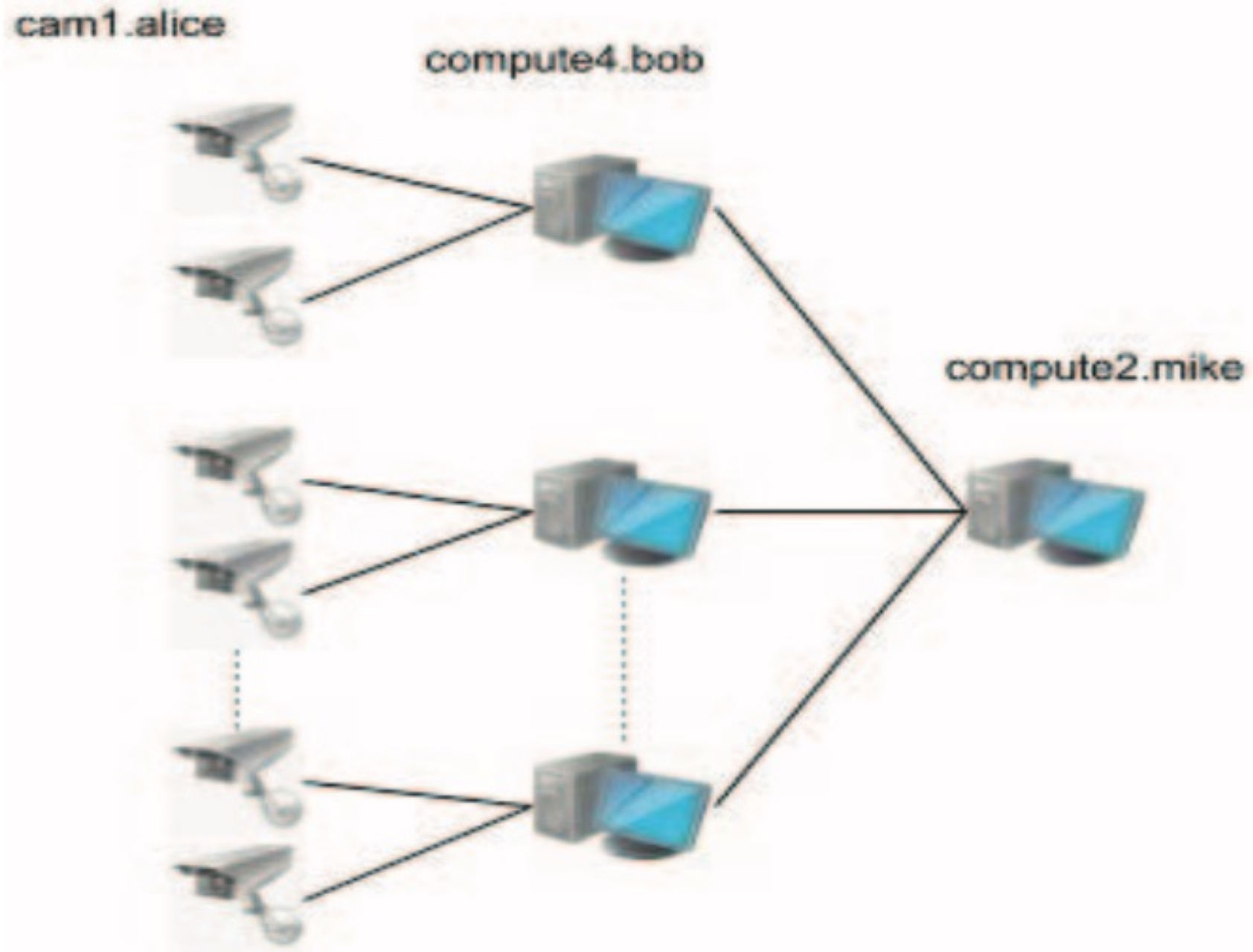
# タスクオーバーレイの実行手順 その3



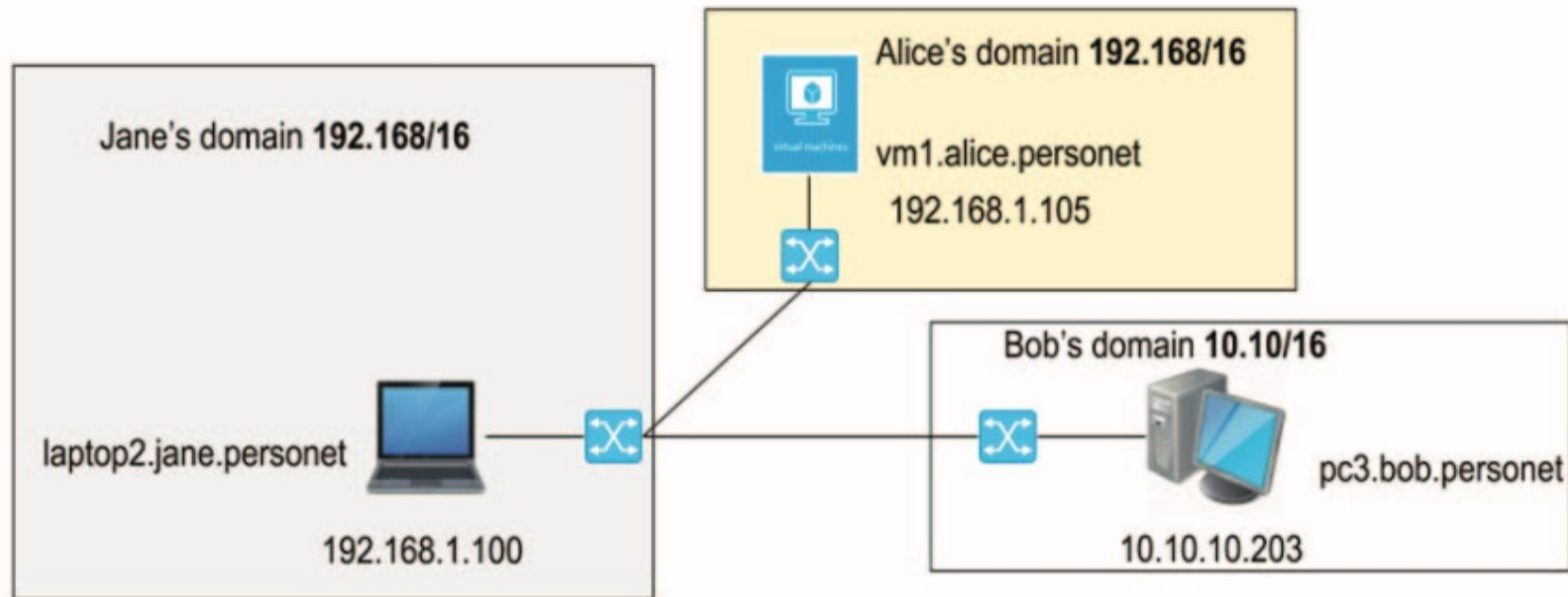
- ⑧ ピアのアドミニストレーションコンポーネントは, イニシエータに対してトンネルの設定とネットワークフロールールの設定が完了すると, ピアのアドミニストレーションコンポーネントはイニシエータにトンネル勧誘応答を返す
- ⑨ タスクオーバーレイが作成されるとオーケストレーションポリシーはリソースノードへのアプリケーションの展開を進める
- ⑩ タスクが終了するとタスクオーバーレイは破棄される



# 負荷分散トポロジ



# アドレス管理



Device name	Remote address	Mapped address
vm1.alice.personet	192.168.1.105	192.168.1.201
pc3.bob.personet	10.10.10.203	192.168.1.198

Jane's addressing table

Device name	Remote address	Mapped address
laptop2.jane.personet	192.168.1.100	192.168.1.103

Alice's addressing table



# 実装に使用する技術及びライブラリ



- PLO間, CLO間通信
  - ✓ NATトラバーサルを備えたP2Pプライベートトンネル(TinCan)を利用して実装
- OSNのメッセージングプラットフォーム
  - ✓ XMPPオープンソースメッセージャーを利用
- デバイスコンポーネント
  - ✓ Python SleekXmppライブラリで実装
- DNS機能
  - ✓ dnscchefを拡張
  - ✓ DNS Req/ResをXMPP上でプロキシすることで実現
- SDNスタック (GM及びSLCC)
  - ✓ ONOSを用いて実装
    - ◆ ONOS(Open Network Operating System) : SDNコントローラフレームワーク
    - ※ 分散コントローラとデータストアのネイティブサポートが存在するため採用
- ONOSとXMPPの連携
  - ✓ XMPPとのインターフェースにはSMACKライブラリを使用
- アドミニストレーションコンポーネントの検証及び評価実験
  - ✓ エミュレーションスクリプトはPythonを用いて実装



# SDN(Software Defined Network)



- ソフトウェア定義ネットワーク
  - ネットワークをソフトウェアによって仮想的に構築
  - 管理者は状況に応じて構成を迅速かつ柔軟に変更可能
- SDNを実現する技術
  - Hop-by-Hop 方式
    - ✓ OpenFlow, …etc.
  - Overlay 方式
    - ✓ VXLAN, …etc.

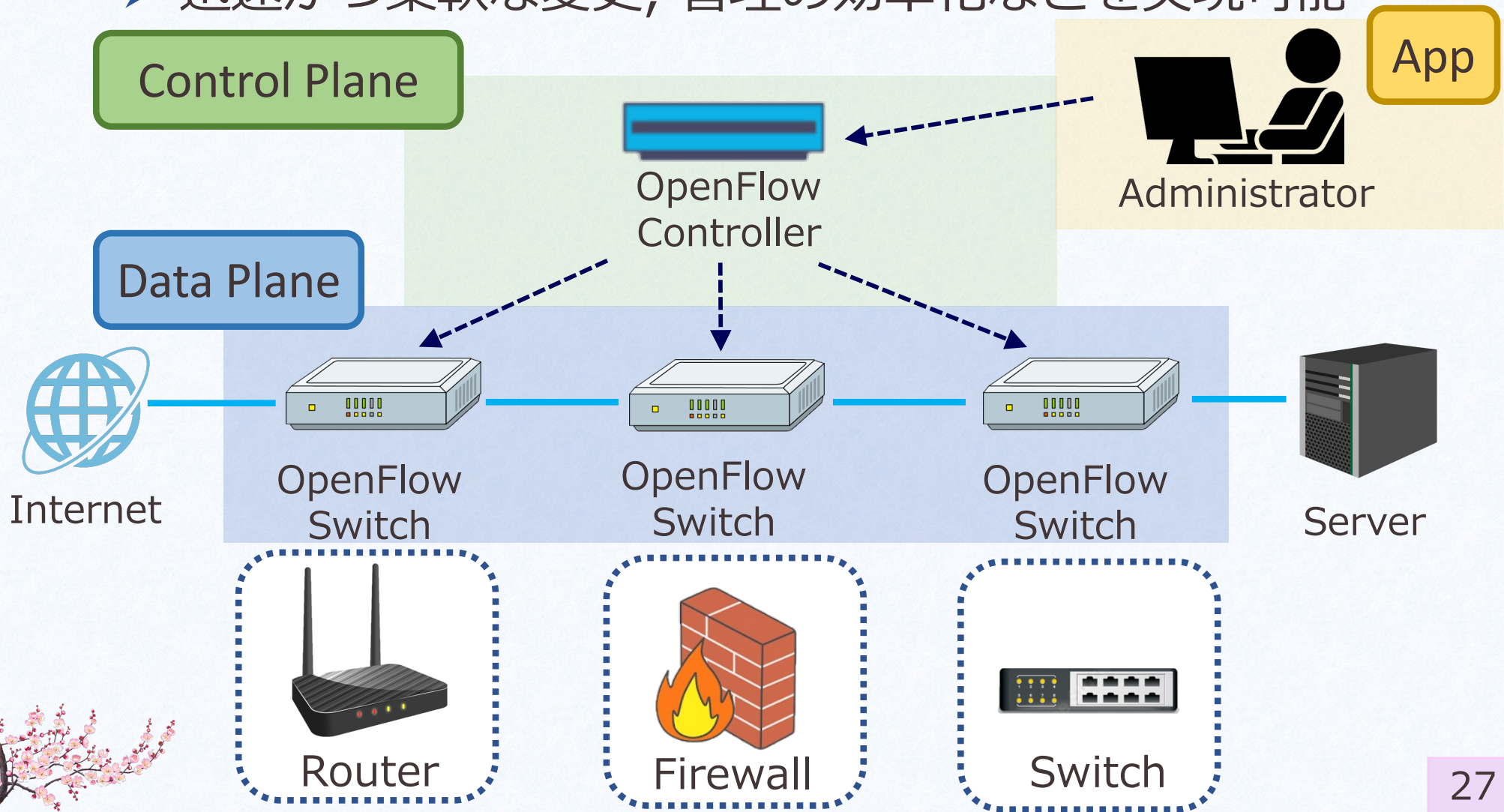


# OpenFlow



## ■ Hop-by-Hop 方式

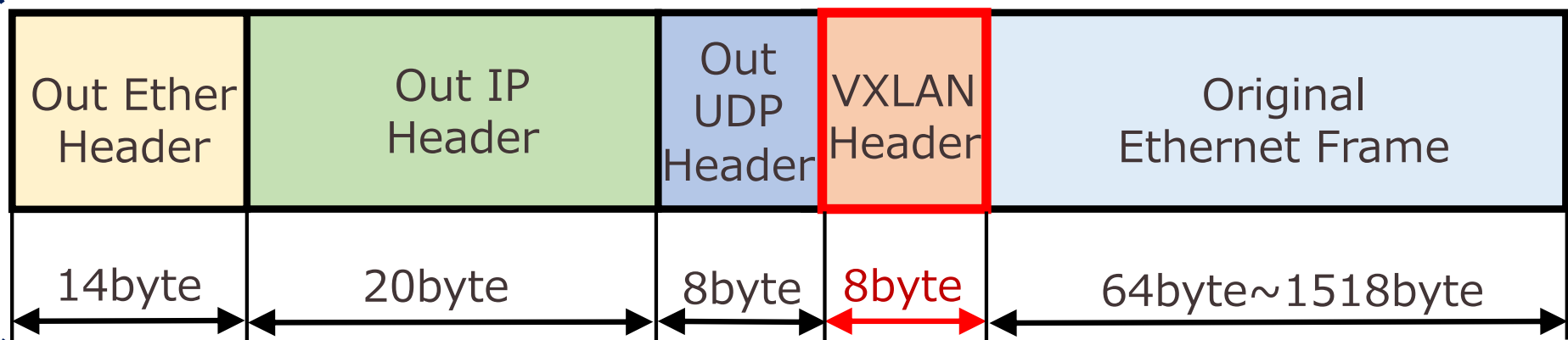
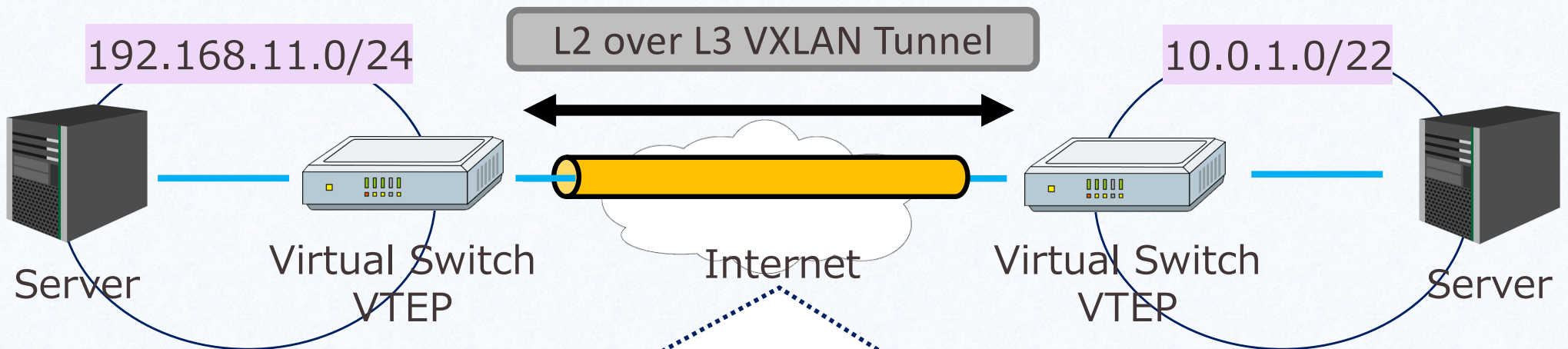
- 「経路制御機能」と「データ転送機能」を分離
- 迅速かつ柔軟な変更, 管理の効率化などを実現可能



# VXLAN(Virtual eXtensible Local Area Network)

## ■ Overlay 方式

- ▶ トンネリングによって仮想ネットワークを構築
- ▶ 既存のネットワーク機器をそのまま使用して実現可能



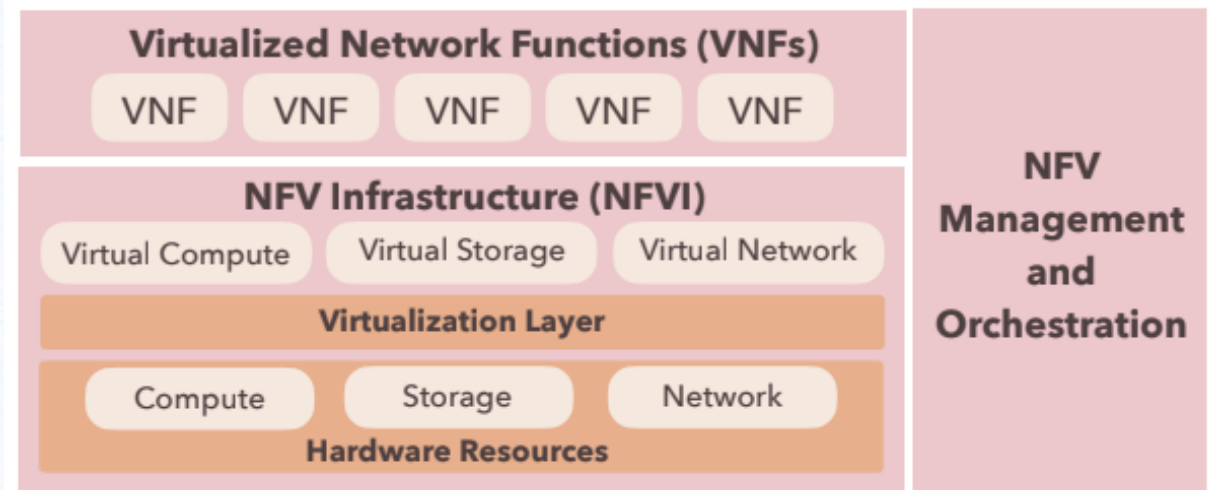
Encapsulation packets with VXLAN

# VNF(Network Function Virtualization)



## ■ ネットワーク機能の仮想化技術

- 専用機器を汎用サーバ上で仮想マシンとして動作
- 物理的な機器を集約可能



# OSN(Online Social Network)



## ■ オンラインソーシャルネットワーク

- インターネットサービスを介して相互に通信する  
分散型コンピュータネットワーク
- ソーシャルネットワーキングサイトは対面でのやり取りを超えて継続的にやり取りするためのスペースを提供
- 社会的・地理的に分散した様々なネットワークのメンバをリンクし, 関係を維持および発展させるのに役立つ
- 膨大なユーザを接続可能
  - ✓ Twitter, Facebook, ...etc.
- 信頼関係をグラフ化し, デバイスをグループ化するのに利用可能

## ■ 問題点

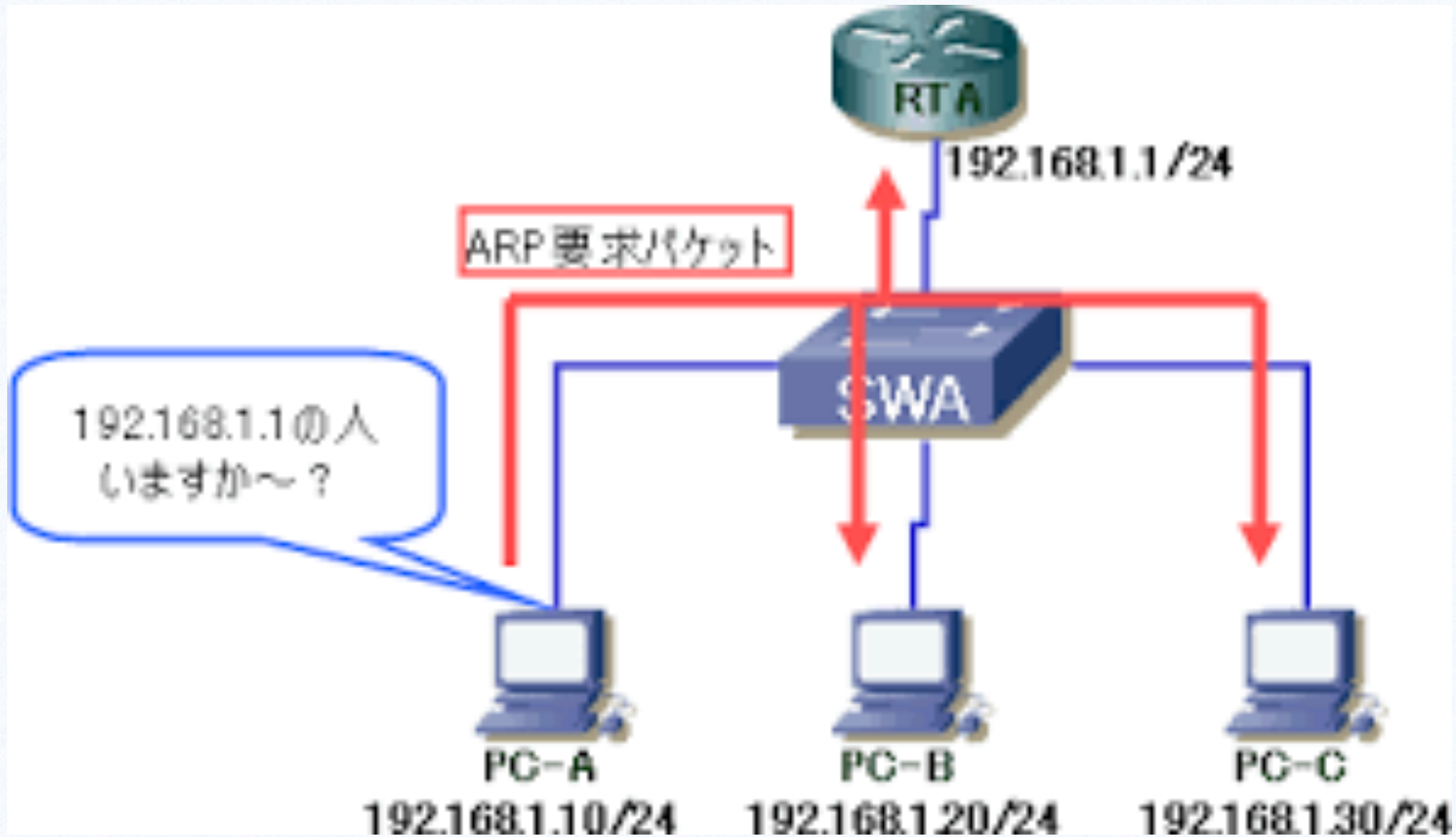
- エッジデバイス間の直接のP2P接続を提供しない
- エッジコンピューティングにおける帯域幅及びレイテンシのメリットを活かせない



# ARP(Address Resolution Protocol)



- L3(IP)アドレスをL2(MAC)アドレスに解決





**[1] PerSoNet: Software-defined Overlay Virtual Networks Spanning Personal Devices Across Social Network Users**

©2018 IEEE - Saumitra Aditya, Kensworth Subratie, Renato J Figueiredo

**[2] TinCan: User-Defined P2P Virtual Network Overlays for Ad-hoc Collaboration**

©2014 IEEE - Juste, K. Jeong, H. Eom, C. Baker, R. J. O. Figueiredo

