

人のための自動化がAIエージェント運用への道を拓く — IOWN APN × VXLAN/EVPNをまたぐマルチレイヤ抽象化の実践

NTT西日本株式会社

芳 政信 鈴木 佑輔

アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
4. 情報取得一元化API
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント



芳 政信
ヨシ マサノブ

- 造園土木職人からネットワークエンジニアへ
- 企業VPNの詳細設計、社内NW運用管理、ネットワークインストラクタなど
- Model Driven Programmability,
Network Programmability, DevNet



鈴木 佑輔
スズキ ユウスケ

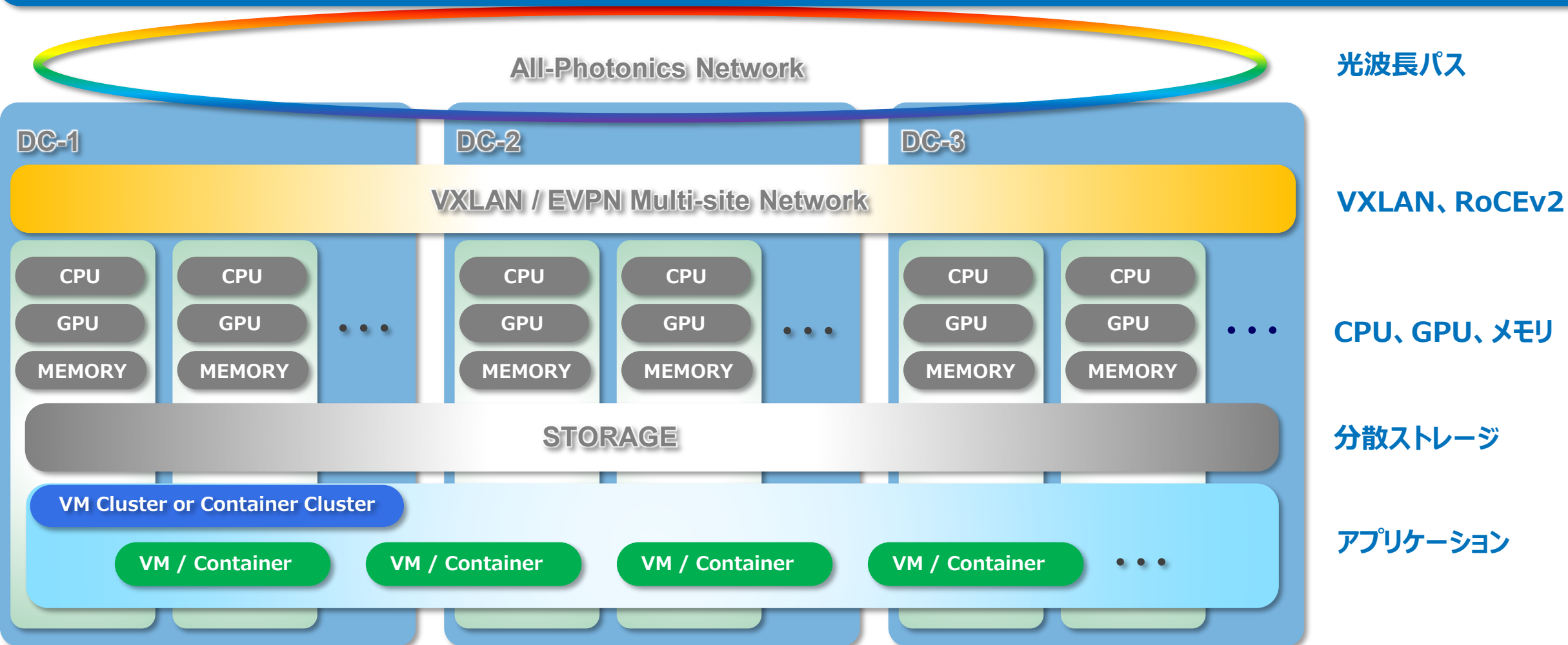
- キャリアネットワーク内で使われているシステムの開発・構築・運用に従事（サーバ寄りのエンジニア）
 - サーバプラットフォーム
 - DHCP/監視などのサーバ、ネットワーク装置の制御サーバ
- Markdown、サーバ仮想化、自動化、自作PC、ガジェット

アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
4. 情報取得一元化API
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント

分散DCの社会実装を目指して

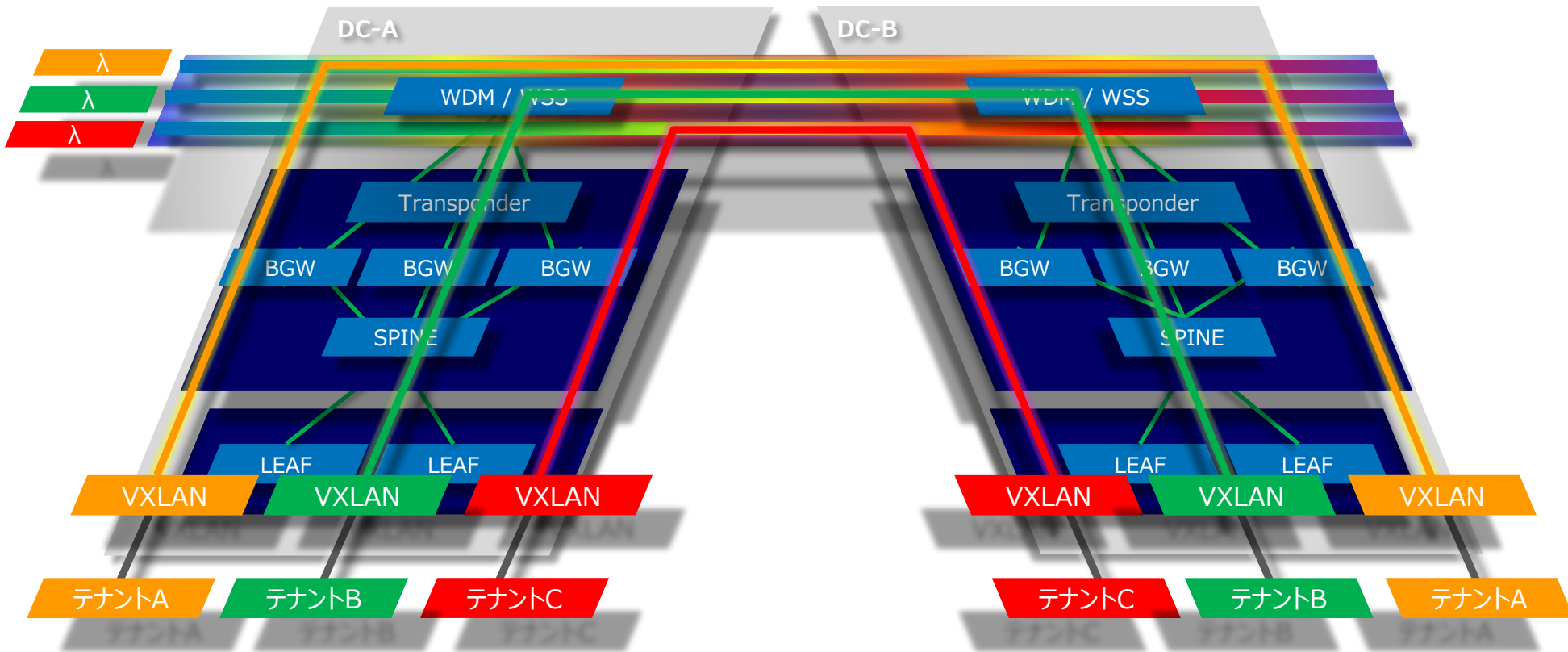
一つのDCでは賄いきれないようなGPUなどのリソース利用のニーズに応えるため、分散DCアーキテクチャが求められている。私たちは、複数拠点のDCをEVPN/VXLANとIOWN APNで結び、DCリソースを自由に活用できる分散DCアーキテクチャを、技術開発・サービス開発を通じて社会実装したいと考えている。



私たちの目指す分散DCアーキテクチャ

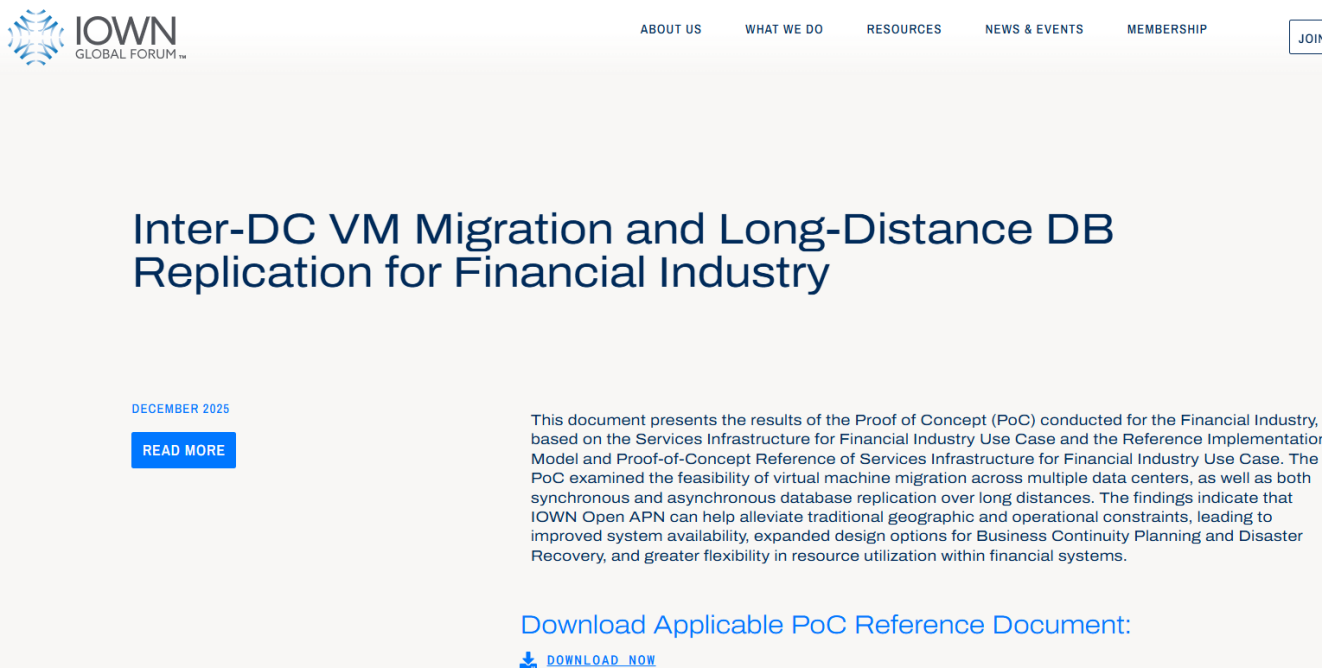
APNの光波長パスとIPオーバーレイの統合制御を目指す。

- ・ テナント、サービス、ワークロードごとの光波長パスも含めた専用のネットワークパス
- ・ オンデマンドによるE2Eのきめ細かなネットワーク制御



テナント毎やサービス毎のVXLANに、光波長を割り当てることでE2Eでのネットワーク制御が可能になる。

IOWN APNを活用した分散DCユースケースに関してニュースリリースやIOWN Global Forum PoCレポートなどを発表し、処理最適化や低遅延接続などの実証を積み重ね、実運用に向けた知見を蓄積してきた。



IOWN GLOBAL FORUM

ABOUT US WHAT WE DO RESOURCES NEWS & EVENTS MEMBERSHIP JOIN

Inter-DC VM Migration and Long-Distance DB Replication for Financial Industry

DECEMBER 2025

[READ MORE](#)

This document presents the results of the Proof of Concept (PoC) conducted for the Financial Industry, based on the Services Infrastructure for Financial Industry Use Case and the Reference Implementation Model and Proof-of-Concept Reference of Services Infrastructure for Financial Industry Use Case. The PoC examined the feasibility of virtual machine migration across multiple data centers, as well as both synchronous and asynchronous database replication over long distances. The findings indicate that IOWN Open APN can help alleviate traditional geographic and operational constraints, leading to improved system availability, expanded design options for Business Continuity Planning and Disaster Recovery, and greater flexibility in resource utilization within financial systems.

Download Applicable PoC Reference Document:

[DOWNLOAD NOW](#)



法人のお客さま 個人のお客さま 企業情報 お問い合わせ一覧 English サイト内検索

シェアする 0 ポスト

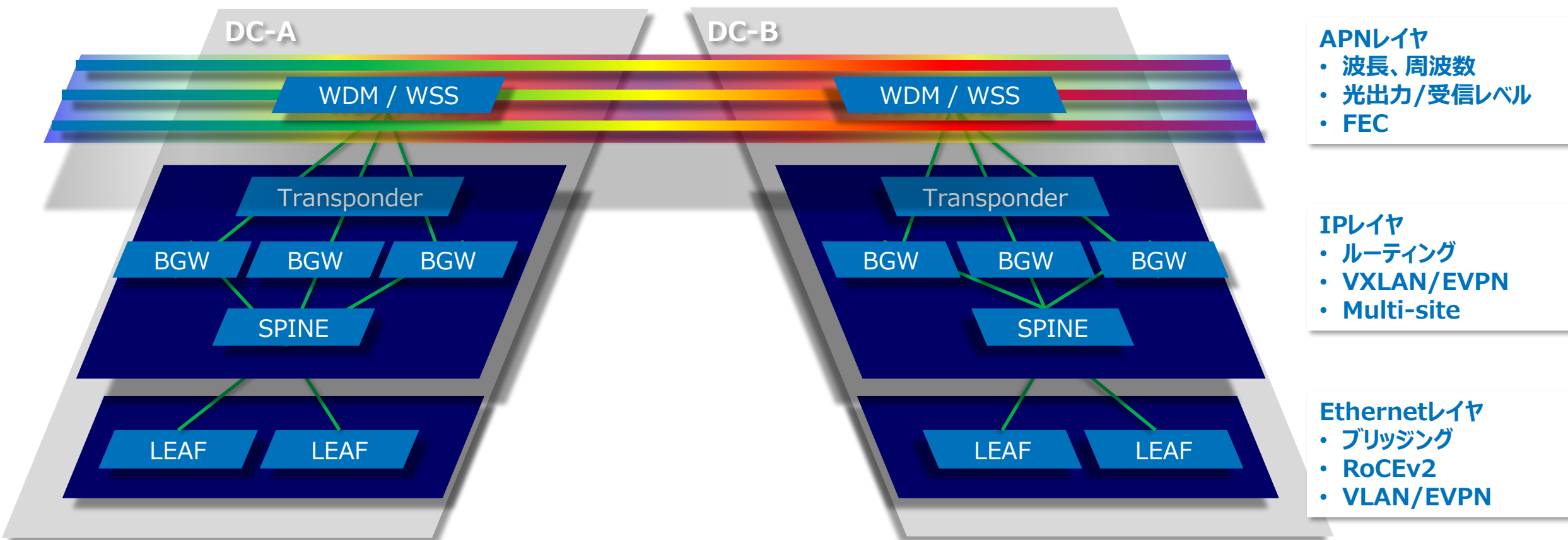
News Release

IOWN APNによる遠隔データセンター間における処理配置最適化の実証実験に成功 ～再生可能エネルギー積極利用によるカーボ ンニュートラルへの貢献～

2025年6月11日
西日本電信電話株式会社
日本電信電話株式会社
株式会社QTnet

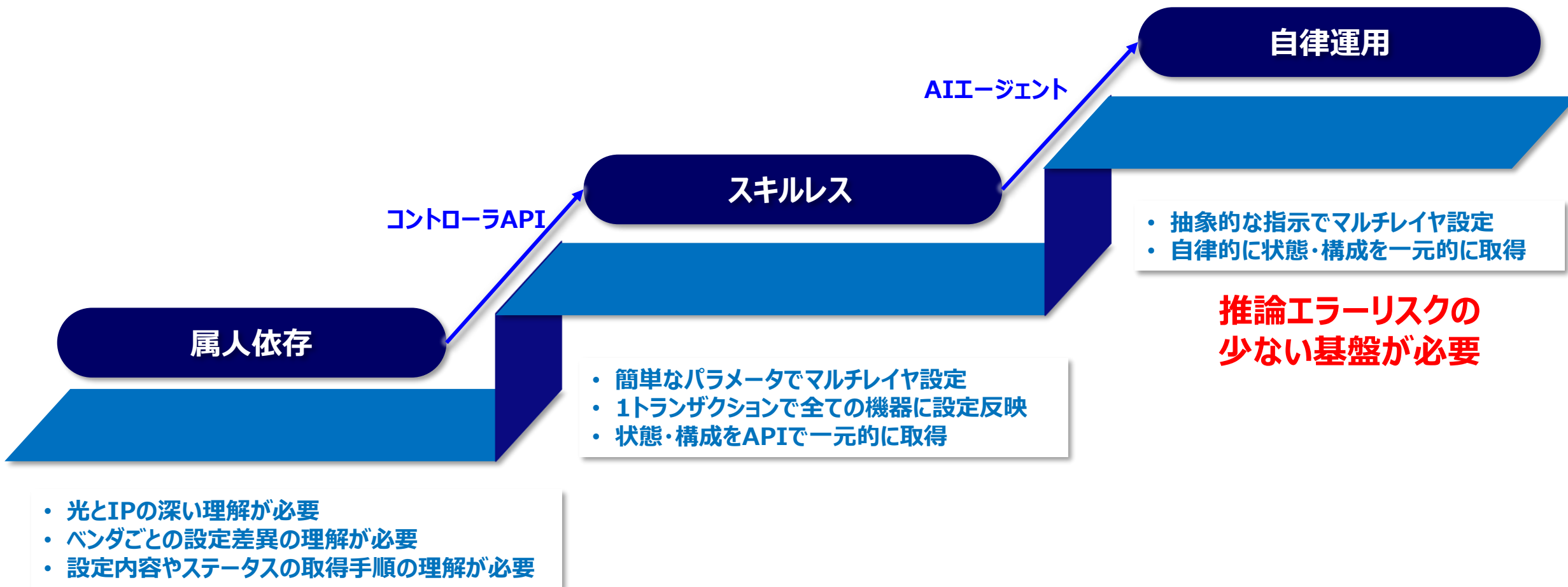
西日本電信電話株式会社（本社：大阪市都島区、代表取締役社長：北村 亮太、以下「NTT西日本」）、日本電信電話株式会社（本社：東京都千代田区、代表取締役社長：島田 明、以下「NTT」）、株式会社QTnet（本社：福岡県福岡市、代表取締役社長：小倉 良夫、以下「QTnet」）は、地理的に離れたデータセンター（以下「DC」）に分散配置された計算処理環境においても、IOWN オールフォトニクス・ネットワーク（All-Photonics Network、以下「APN[®]」）による通信の特徴である大容量・

IOWN APNをDC間バックボーンに用いた3拠点の分散DC基盤を日常的に運用管理しているのは少数のメンバーである。少数での運用に関わらず、求められるスキルの幅は極めて広い。



スキルレスで運用できる基盤づくりから自律運用へ

現在の分散DC運用は少数の高スキル人材に依存しており、属人化によりスケールしない。
スキルレスで運用できる基盤を開発し、その先にAIエージェントによる運用へとつなげたい。



分散DCにおける運用は、マルチレイヤ・マルチベンダ・可視化の分断により複雑化している。
スキルレス基盤はマルチレイヤ制御統合API・トランザクションの一括管理・情報取得一元化APIにより実現する

マルチレイヤ



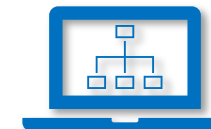
- L1（光）とL2/L3（VXLAN/EVPN）の統合設計が複雑
- レイヤ間のマッピングに高度なスキルが必要
- etc...

マルチベンダ



- ベンダごとの設定となり、一括変更ができない
- 自動ロールバックができない
- etc...

可視化分断



- レイヤ、ベンダごとに管理が分断され、横断的な把握ができない
- ライブステータスを一元的に取得できない
- etc...

スキルレス基盤

マルチレイヤ制御統合API

トランザクション一括管理

情報取得一元化API

スキルレス基盤を実現する①マルチレイヤ制御統合API、②トランザクションの一括管理、③情報取得一元化APIは、AIが判断を誤る余地も排除することができそう

スキルレス基盤要素

自律運用時の推論エラーリスクと、スキルレス基盤を使うことの想定メリット



マルチレイヤ制御統合API

- ・ シンプルな設定パラメータ
- ・ 統合API

パラメータの間違い、APIの間違い

→APIが統合されており、少数パラメータに抽象化されているので、誤る可能性が低い

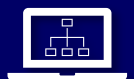


トランザクションの一括管理

- ・ 全機器のトランザクション管理
- ・ ロールバック処理

設定投入機器の間違い、設定順番間違い

→対象機器はAPIのパラメータから自動特定され、AIが個別に指定しない



情報取得一元化API

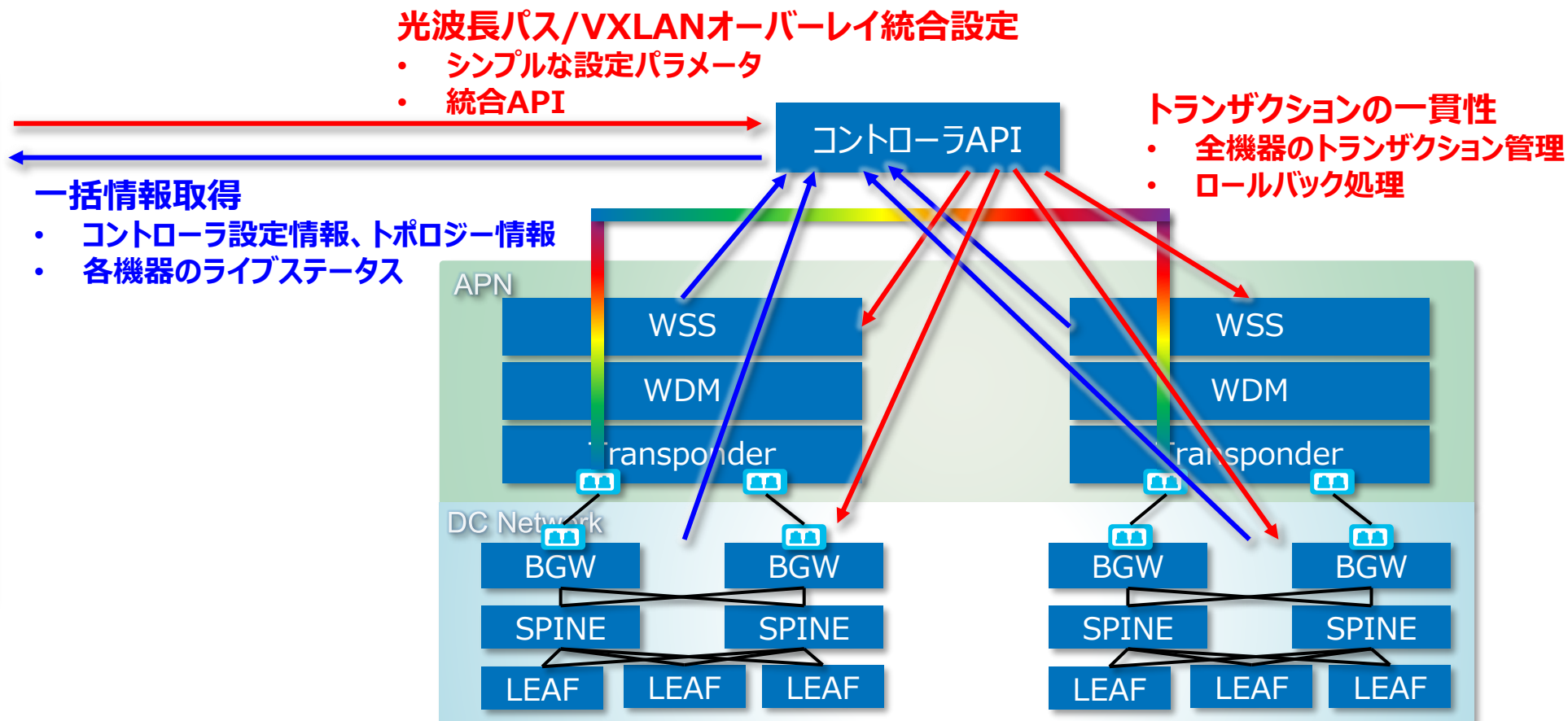
- ・ コントローラ設定情報取得
- ・ 各機器のライブステータス取得

取得情報の間違い、取得機器の間違い

→一元化されたAPIにより、取得方法はレイヤやベンダに依存せず、誤る可能性が低い

仮説：人のためのスキルレス基盤はAIエージェントによる運用にもつながる

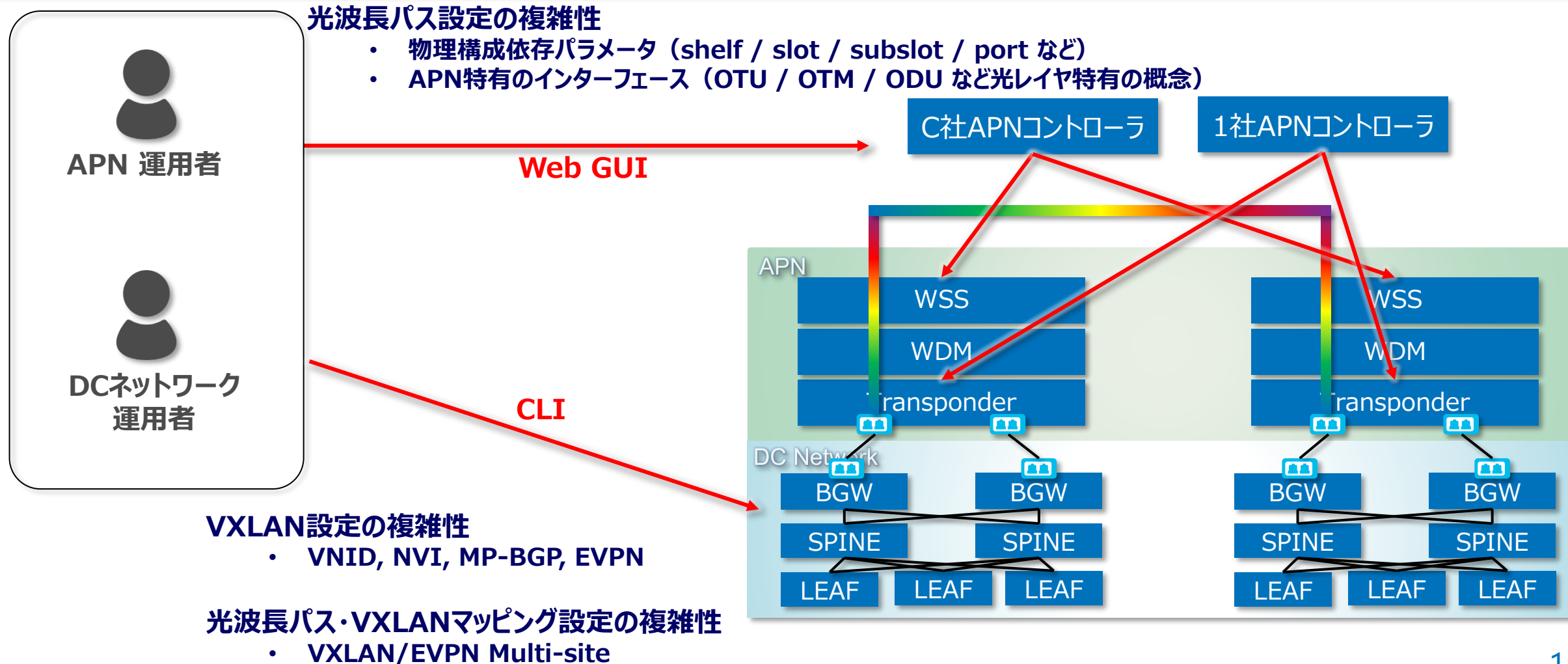
- ① マルチレイヤ制御統合API : ユーザ、AIが扱いやすい光波長パス/VXLANオーバーレイ統合設定を実現
- ② トランザクションの一括管理 : 複数ベンダ・複数機器を設定するトランザクションの一貫性を実現
- ③ 情報取得一元化API : 各種情報の統合取得を実現（対ユーザ、対オペレータ、対AI）



アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
4. 情報取得一元化API
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント

「光波長パスの設定」、「VXLAN設定」、「分散DC特有の光波長パス・VXLANマッピング」といった、単体でも複雑な設定が組み合わさることで、さらに難易度の高い設定となる。
→APNとDCネットワークの個別理解と個別対応が必要



「光波長パスの設定」、「VXLAN設定」、「分散DC特有の光波長パス・VXLANマッピング」といった、単体でも複雑な設定が組み合わさることで、さらに難易度の高い設定となる。

→APNとDCネットワークの個別理解と個別対応が必要

APN 運用者

DCネットワーク
運用者

光波長パス設定の複雑性

- ・ 物理構成依存パラメータ (shelf / slot / subslot / port など)
- ・ APN特有のインターフェース (OTU / OTM / ODU など光レイヤ特有の概念)

ユーザ、AIが扱いやすい 光波長パス/VXLAN オーバーレイ統合設定を実現

VXLAN設定の複雑性

- ・ VNID, NVI, MP-BGP, EVPN

光波長パス・VXLANマッピング設定の複雑性

- ・ VXLAN/EVPN Multi-site



- ① 光波長パスのE2EのBGWスイッチ名
- ② 光波長パスに接続するAPN接続インターフェースと方向



- APNの物理構成情報（機器名、物理インターフェース）を、エンド端末の属性情報としてDBへ登録。
- 光波長パス設定時に属性情報を参照することで、パラメータの抽象化、設定のシンプル化を実現する。

コントローラー YANG DB

devices device BGW-OSAKA

attribute apn-connected interface eth1/53

transponder device TPND-OSAKA

transponder eth-interface 1/0/0/C1

transponder och-interface 1/1/0/E1

wdm och-interface 3/1/0/C1

wss device WSS-OSAKA

wss direction tokyo

client-interface 1/0/0/C1

external-interface 1/0/0/E1

wss direction hakata

client-interface 2/0/0/C1

external-interface 2/0/0/E1

=> **エンド端末名**

=> **エンド端末のトランスポンダーとの接続インターフェース**

=> **接続トランスポンダ名**

=> **トランスポンダーのBGWtとの接続インターフェース**

=> **トランスポンダーのWDMとの接続インターフェース**

=> **WDMのトランスポンダーとの接続インターフェース**

=> **接続WSS名**

=> **光波長パスの方向（使用インターフェースは方向で異なる）**

=> **WSSのWDMや他のWSSとの接続インターフェース**

=> **WSSの他サイトとの接続インターフェース**

=> **光波長パスの方向（使用インターフェースは方向で異なる）**

=> **WSSのWDMや他のWSSとの接続インターフェース**

=> **WSSの他サイトとの接続インターフェース**

(参考) パラメータマッピングによるAPI統合とシンプル化

- ① REST APIから入力されたパラメータ (JSON) を解析
- ② YANG DBを参照して光波長パス設定に必要な物理構成パラメータを取得
- ③ 使用済波長をNetconf APIでAPNから取得し、光波長パス設定に必要な未使用波長を算出
- ④⑤ 光波長パス設定のコンフィグテンプレートにパラメータを埋め込みコンフィグを作成
- ⑥ Netconfで機器設定

YANG DB

REST API

パラメータ (JSON)

- ・ エンドエンドのBGWと、APN接続インターフェースとパスの方向

物理構成情報、光波長パス情報

BGWとAPNの物理構成情報や、光波長パスのエンドポイント情報

①解析

②参照

コンフィグテンプレート

Transponder
Netconf XML

WSS, WDM
Netconf XML

④パラメータマッピング

Python

③使用済波長取得

⑤コンフィグ作成

Transponder用コンフィグ (Netconf XML)
光波長パス設定

WSS, WDM用コンフィグ (Netconf XML)
光波長パス設定

RPC, Netconf API

⑥設定

Netconf API

F社Transponder

F社WSS, WDM

開発ポイント②：シンプルな共通モデルによるAPI化（VXLAN）

VXLANオーバーレイ設定、光波長パス・VXLANマッピング設定を抽象化（①～③）し、共通モデルでAPI化。

- ① VXLANオーバーレイのE2EのVTEP
- ② VTEPで設定するVXLAN情報とインターフェース情報
- ③ VXLANをマッピングする光波長パスの設定ポリシー名



運用管理者



AIエージェント

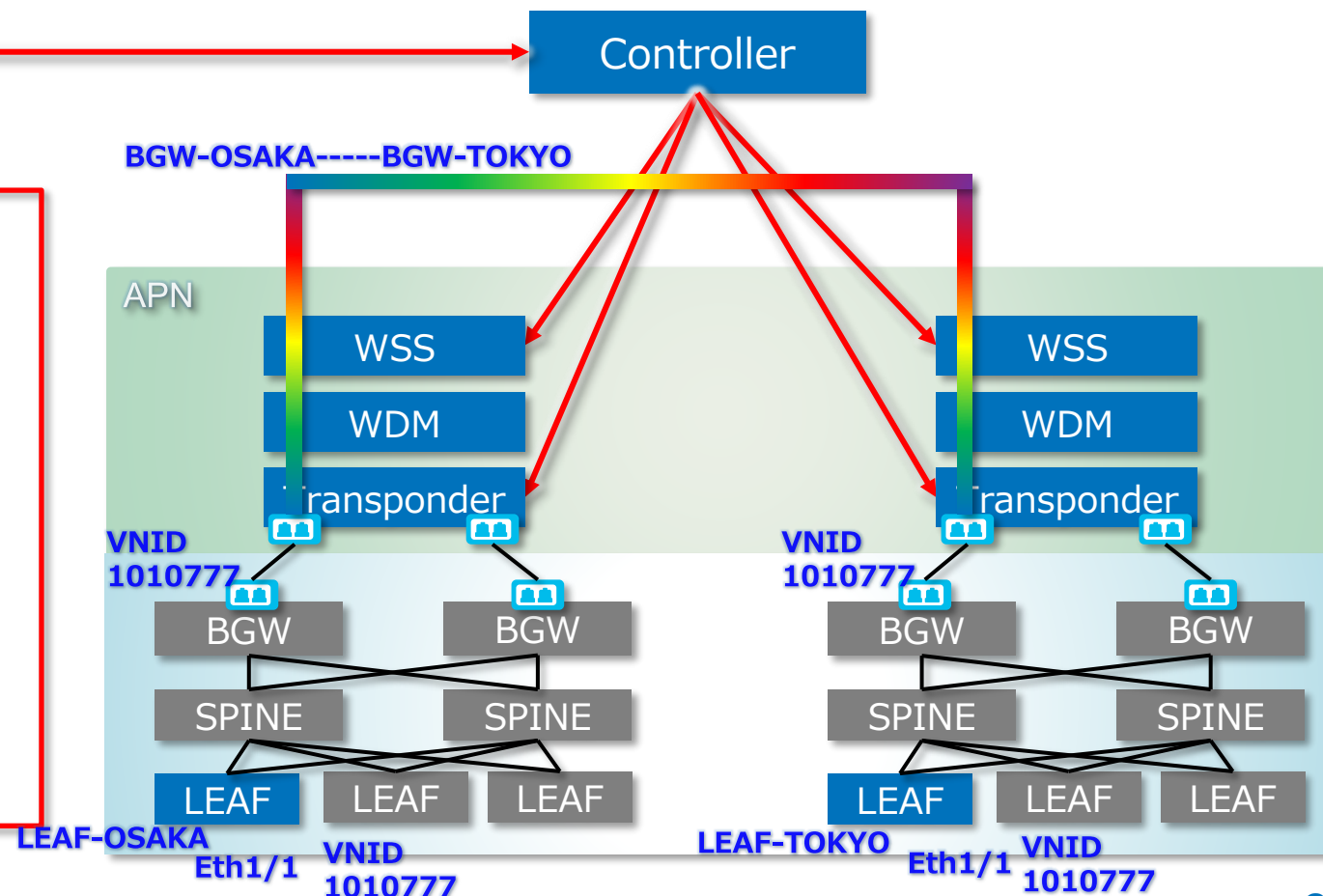
REST API

Controller

```
{
  "device": "LEAF-OSAKA",
  "devices_config": {
    "nail-ddc-overlay:ddc_overlay": {
      "bgw": "disable",
      "l2_vni": {
        "vlans": [{
          "vlan": 777,
          "vni": 1010777
        }]
      },
      "phys_intf": {
        "interfaces": [{
          "interface": "eth1/1",
          "leaves": {
            "trunk": [null]
          }
        }]
      },
      "wavelength_overlay": {
        "wavelength_path_policy": [
          "BGW-OSAKA-----BGW-TOKYO"
        ]
      }
    }
  }
}
```

```
{
  "device": "LEAF-TOKYO",
  "devices_config": {
    "nail-ddc-overlay:ddc_overlay": {
      "bgw": "disable",
      "l2_vni": {
        "vlans": [{
          "vlan": 777,
          "vni": 1010777
        }]
      },
      "phys_intf": {
        "interfaces": [{
          "interface": "eth1/1",
          "leaves": {
            "trunk": [null]
          }
        }]
      },
      "wavelength_overlay": {
        "wavelength_path_policy": [
          "BGW-OSAKA-----BGW-TOKYO"
        ]
      }
    }
  }
}
```

AIエージェントもわかりやすい設定項目

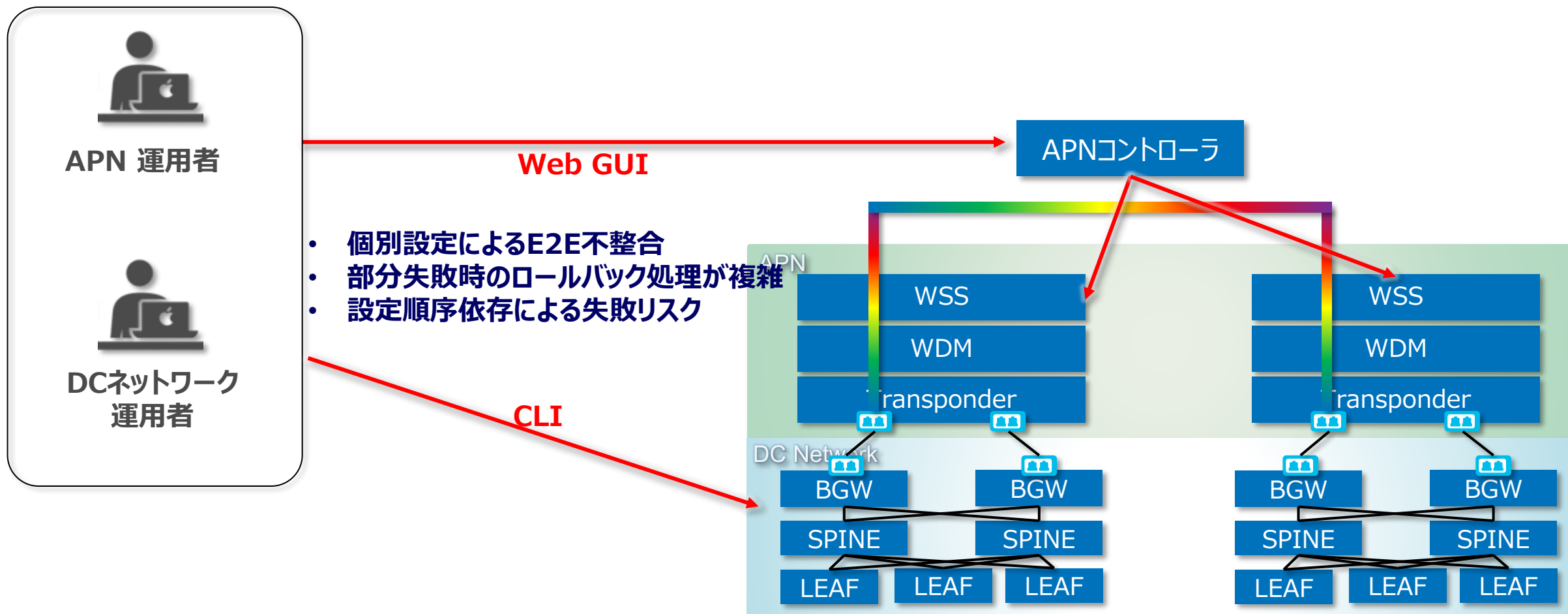


アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
4. 情報取得一元化API
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント

複数ベンダ・複数機器の設定の難しさ

複数ベンダ・複数装置をまたぐ設定では、一括管理ができない場合に設定の整合性が崩れる。
→設定途中での不整合や障害復旧の複雑化。



複数ベンダ・複数装置をまたぐ設定では、一括管理ができない場合に設定の整合性が崩れる。

→設定途中での不整合や障害復旧の複雑化。



APN 運用者

Web GUI

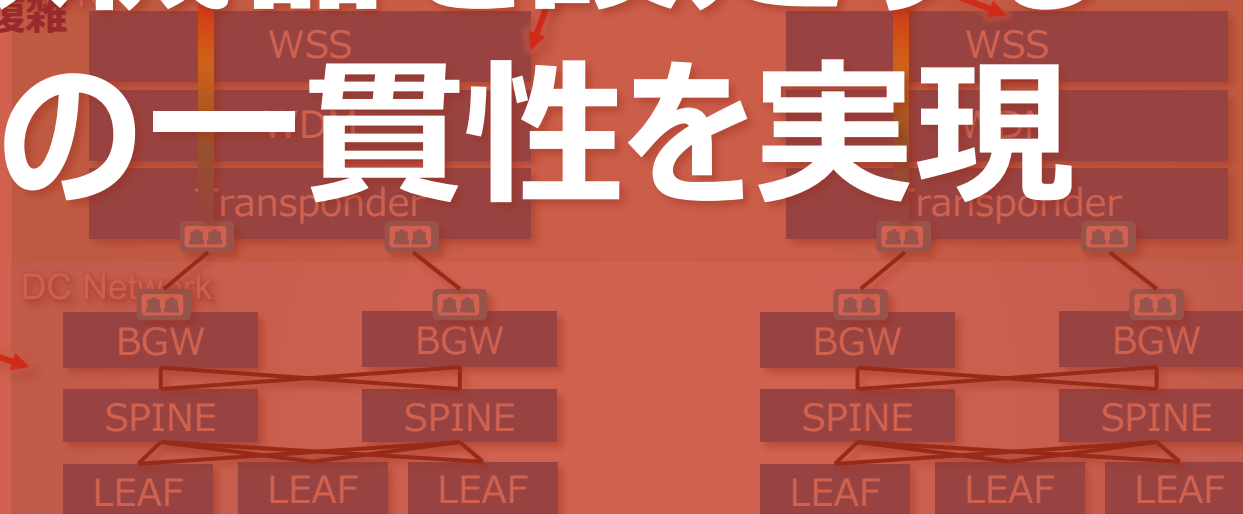
APNコントローラ

複数ベンダ・複数機器を設定する トランザクションの一貫性を実現

- 個別設定によるE2E
- 部分失敗時のロールバック処理が複雑
- 設定順序依存による失敗リスク

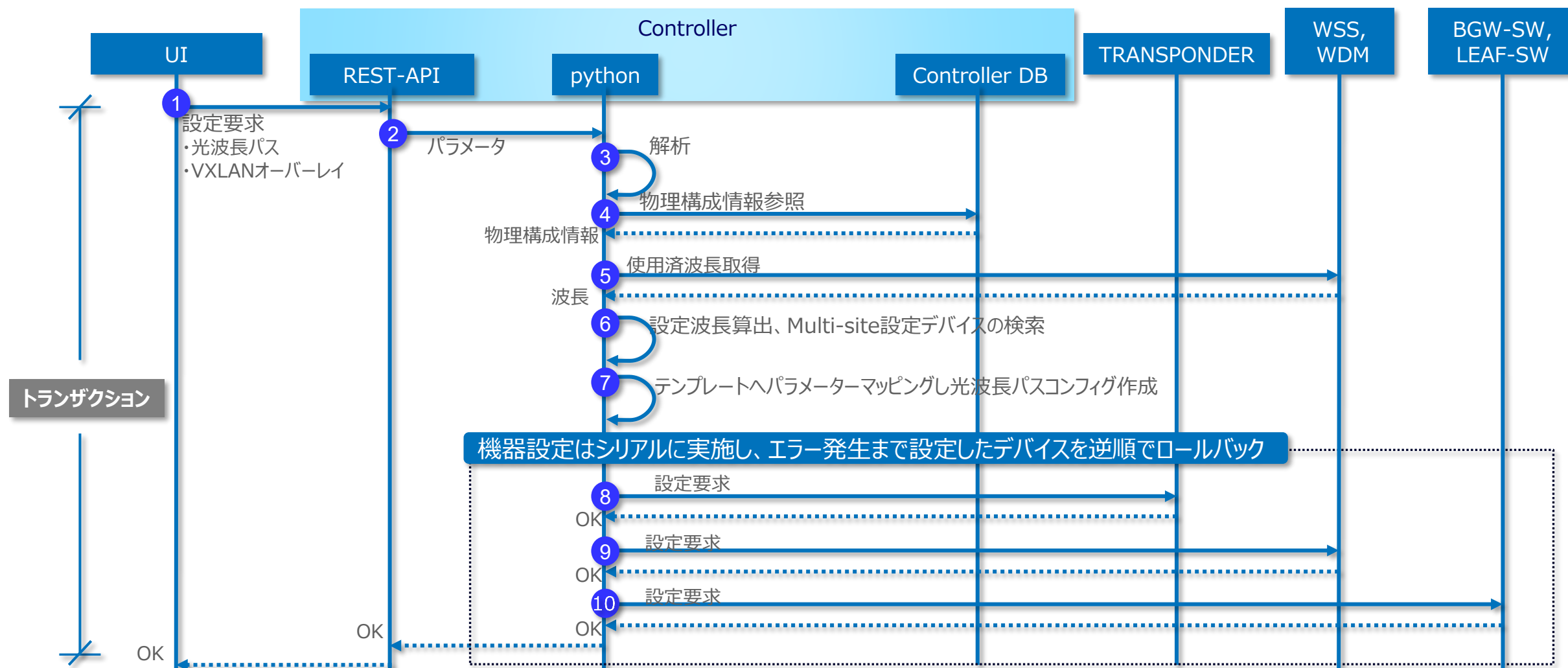
CLI

DCネットワーク
運用者



開発ポイント：トランザクションの一貫性の確保

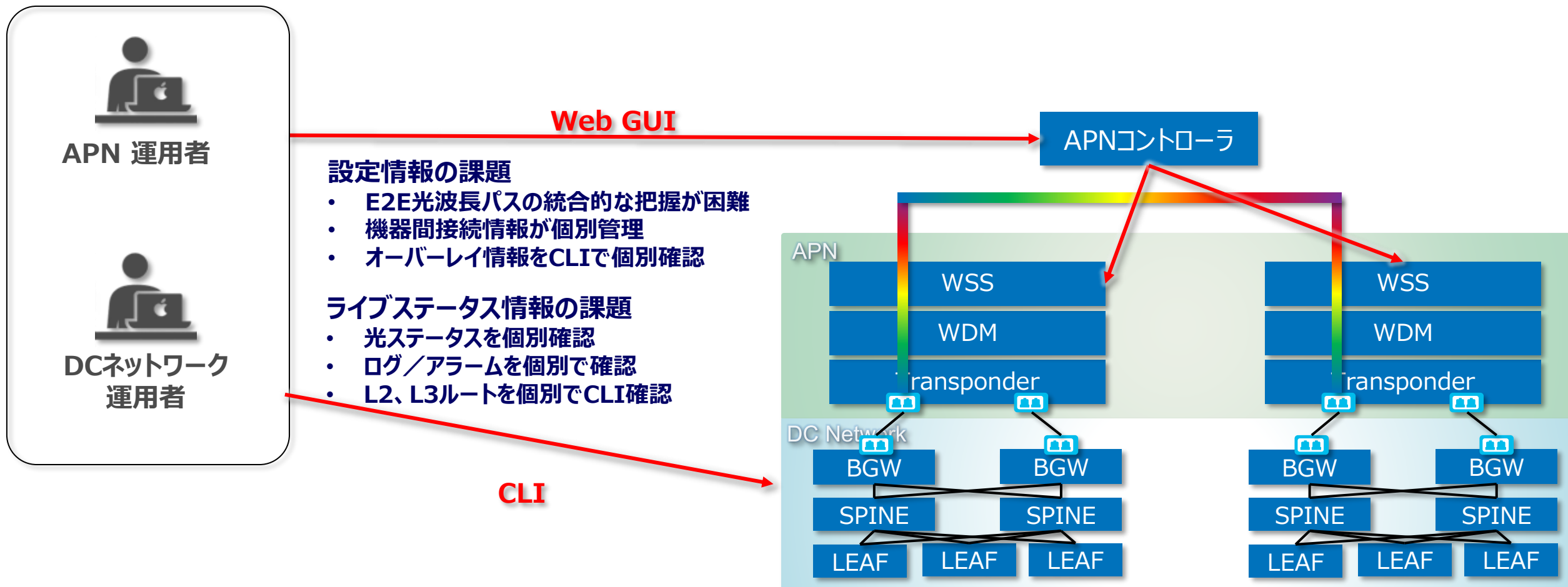
- トランザクションはコントローラで一貫性を確保し全ての機器設定が完了するまで終了させない。
- 途中でエラーが発生した場合は、それ以前の全ての機器設定をロールバックする。



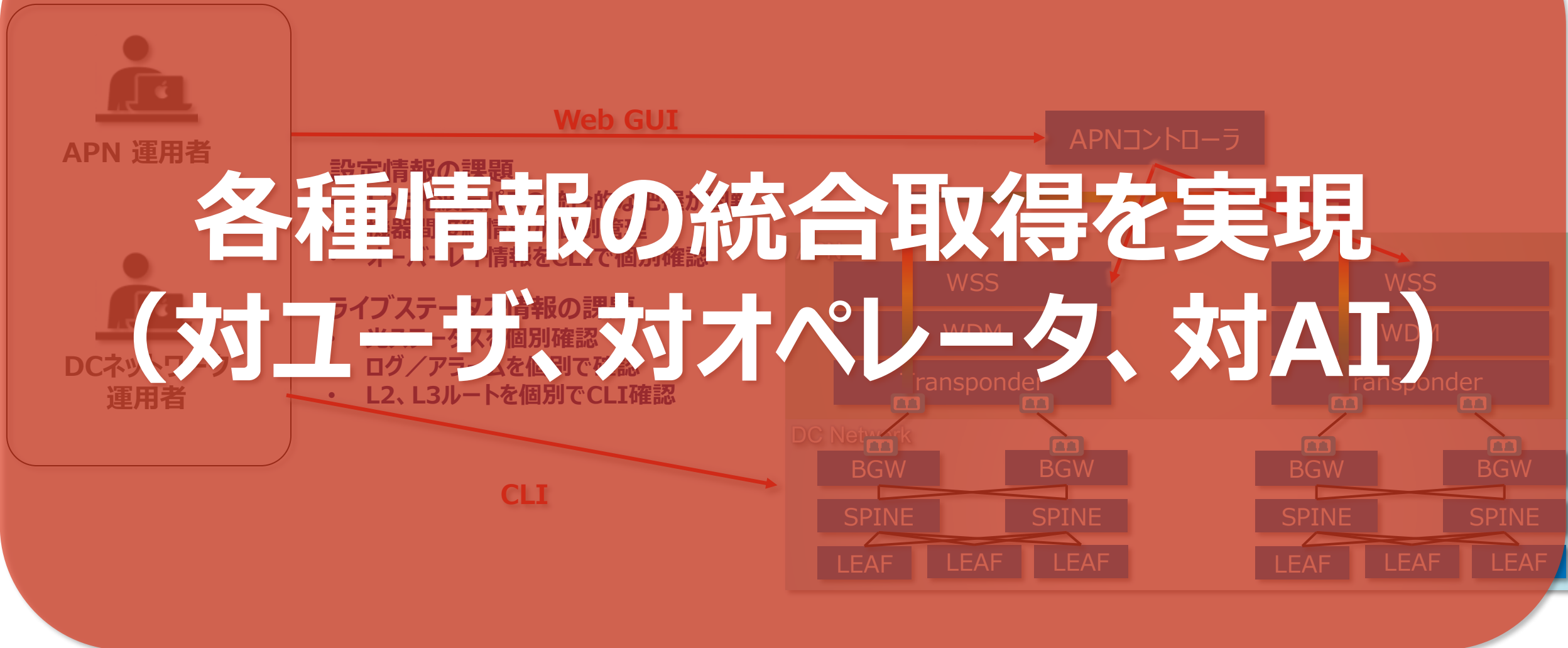
アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
- 4. 情報取得一元化API**
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント

APNの光波長パスとVXLANオーバーレイの統合環境では、情報が分断（サイロ化）され統合可視化が困難。
→E2E視点での状態把握や迅速な障害分析が困難。



APNの光波長パスとVXLANオーバーレイの統合環境では、情報が分断（サイロ化）され統合可視化が困難。
→E2E視点での状態把握や迅速な障害分析が困難。

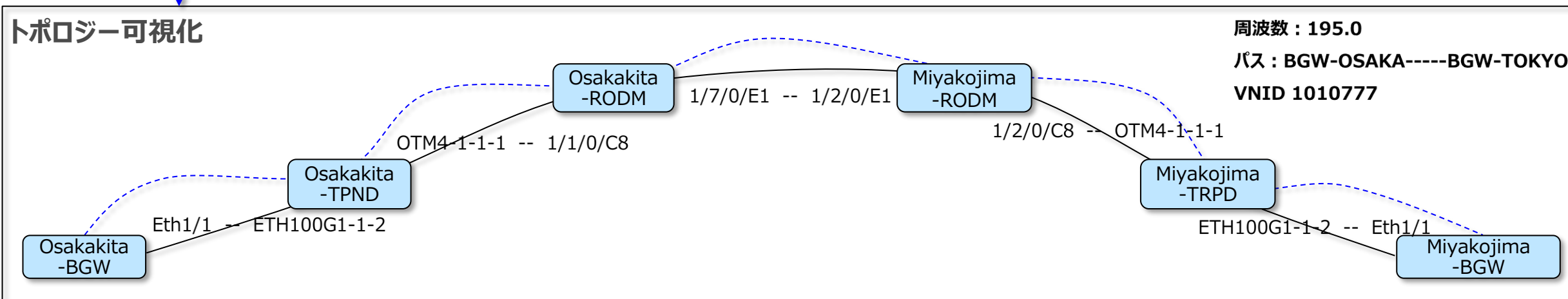


開発ポイント①：コントローラDBから光波長パス設定情報の取得

E2Eの光波長パス設定情報を構造化されたコントローラDBに保存し、APIから参照する。

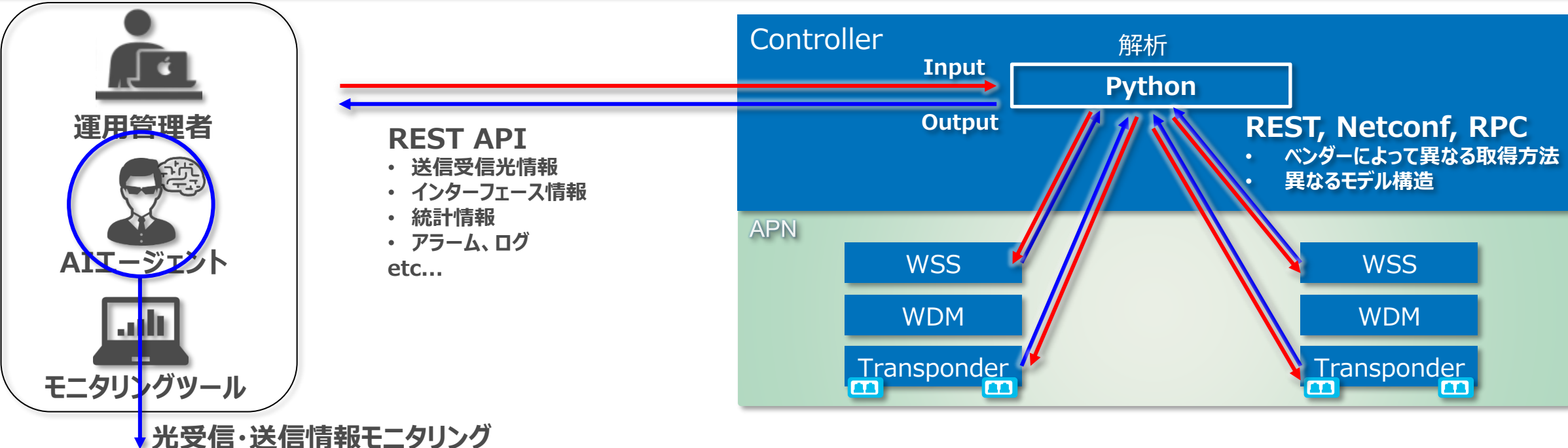


トポロジー可視化



開発ポイント②：APNからのライブステータスの取得

各デバイスのライブステータスを共通のコントローラAPIから取得する。



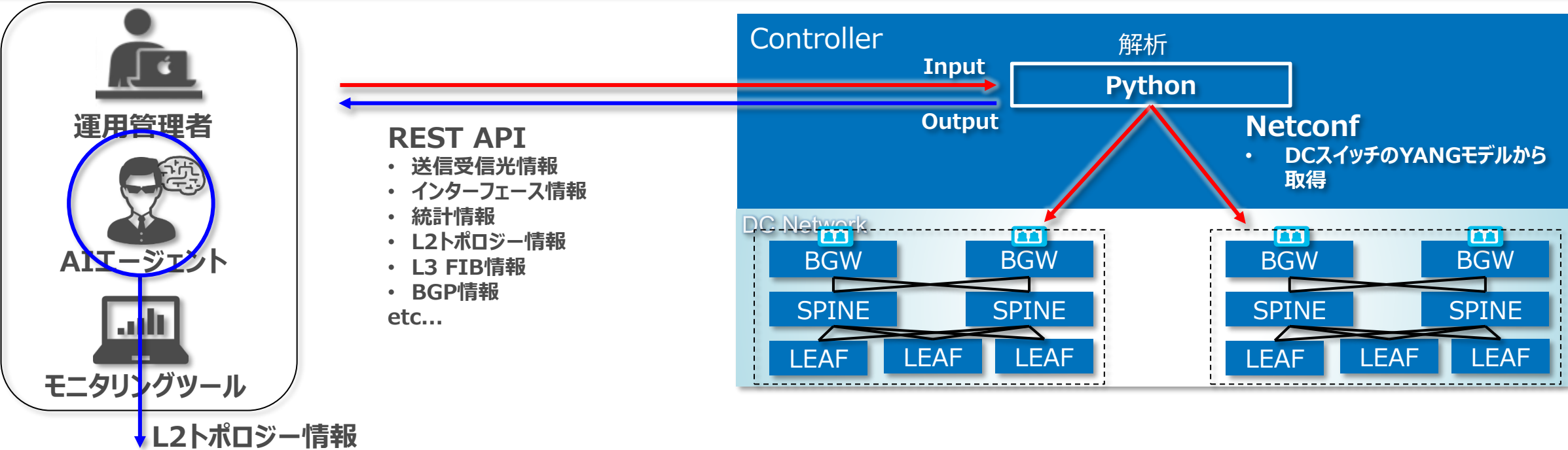
光受信・送信情報モニタリング

↓ 受信 (Receive) 光パワー		
パラメータ	15分値	1日値
OTS 受信パワー	-4.0 dBm	-4.0 dBm
OTS 受信パワー (Min)	-4.0 dBm	-4.0 dBm
OTS 受信パワー (Max)	-3.9 dBm	-3.9 dBm
OTS 受信パワー (Avg)	-3.91 dBm	-3.91 dBm
OMS 受信パワー	-6.8 dBm	-6.8 dBm
OMS 受信パワー (Min)	-6.8 dBm	-6.8 dBm
OMS 受信パワー (Max)	-6.7 dBm	-6.7 dBm
OMS 受信パワー (Avg)	-6.7 dBm	-6.71 dBm
OSC 受信パワー	-7.2 dBm	-7.2 dBm
OSC 受信パワー (Min)	-7.2 dBm	-7.2 dBm
OSC 受信パワー (Max)	-7.2 dBm	-7.1 dBm
OSC 受信パワー (Avg)	-7.19 dBm	-7.19 dBm

↑ 送信 (Transmit) 光パワー		
パラメータ	15分値	1日値
OTS 送信パワー	7.8 dBm	7.8 dBm
OTS 送信パワー (Min)	7.8 dBm	7.6 dBm
OTS 送信パワー (Max)	7.9 dBm	7.9 dBm
OTS 送信パワー (Avg)	7.8 dBm	7.8 dBm
OMS 送信パワー	6.0 dBm	6.0 dBm
OMS 送信パワー (Min)	5.9 dBm	5.9 dBm
OMS 送信パワー (Max)	6.0 dBm	6.1 dBm
OMS 送信パワー (Avg)	6.0 dBm	5.98 dBm
OSC 送信パワー	3.2 dBm	3.2 dBm
OSC 送信パワー (Min)	3.1 dBm	2.7 dBm
OSC 送信パワー (Max)	3.2 dBm	3.3 dBm
OSC 送信パワー (Avg)	3.19 dBm	3.2 dBm

開発ポイント③：DCスイッチからのライブステータスの取得

各デバイスのライブステータスを共通のコントローラAPIから取得する。



Topo ID	VXLAN名	VNI フラグ	VTEP IP	フラッドリストピア
183	Vxlan-1234183	1,234,183 advertise-mac, multisite	10.127.255.103	10.127.255.101
184	Vxlan-1234184	1,234,184 advertise-mac, multisite	10.127.255.103	10.127.255.101
555	Vxlan-155555	155,555 advertise-mac, multisite	10.127.255.103	—
666	Vxlan-155666	155,666 advertise-mac, multisite	10.127.255.103	—
2105	Vxlan-1012105	1,012,105 advertise-mac, multisite	10.127.255.103	10.127.255.101
2161	Vxlan-1042161	1,042,161 advertise-mac, multisite	10.127.255.103	10.127.255.101
2162	Vxlan-1042162	1,042,162 advertise-mac, multisite	10.127.255.103	10.127.255.101
2163	Vxlan-1042163	1,042,163 advertise-mac, multisite	10.127.255.103	10.127.255.101
2164	Vxlan-1112164	1,112,164 advertise-mac, multisite	10.127.255.103	—

アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
4. 情報取得一元化API
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント

VXLANオーバーレイ設定を、従来の属人運用・スキルレス基盤・AIエージェントの3パターンで整理する。
運用者・設定方法・設定ボリューム・必要スキルの違いを明確にした上で、比較検証を行う。

VXLANオーバーレイ設定

	運用者	設定方法	設定ボリューム	設定対象	スキル
パターン①	高度スキル保持者	従来 (CLI / Web GUI)	100行	Switch 4台	VXLAN/EVPN Multi-siteの専門スキル
パターン②	一般スキル運用者	スキルレス基盤 (コントローラAPI)	7パラメータ	コントローラAPI 1台	コントローラAPIの使い方
パターン③	AIエージェント	スキルレス基盤 (コントローラAPI)	7パラメータ	コントローラAPI 1台	不要

スキルレス基盤は属人運用と同等の所要時間を実現し、さらにAIエージェントと組み合わせることで最短時間かつスキル不要の運用を可能にする。

VXLANオーバーレイ設定

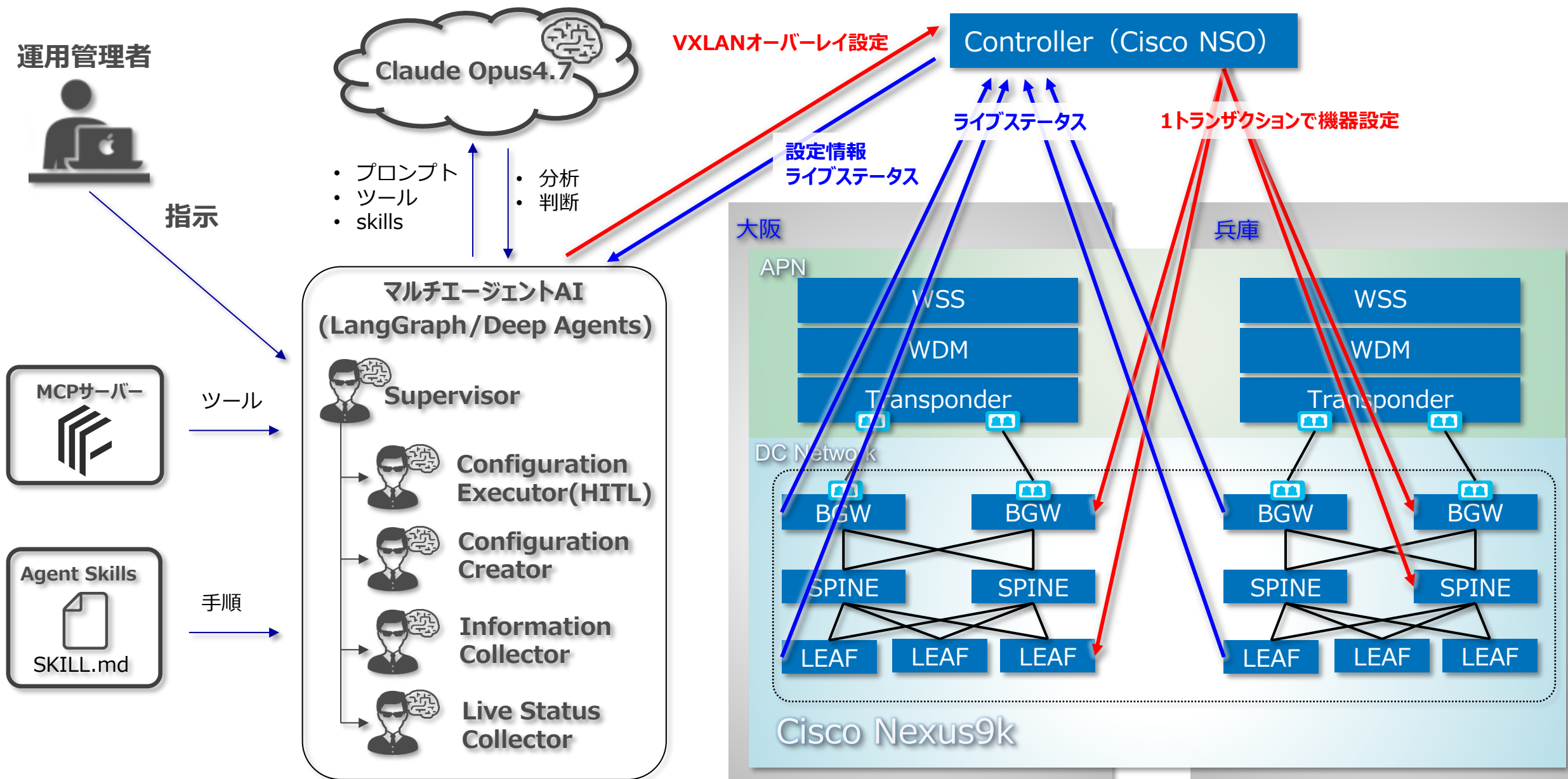
	事前確認	設定	事後確認	トータル	考察
パターン①	5分	6分	8分	19分	- 高スキル者は手順を熟知しており、CLI操作が高速 「どこを見るか」を即座に判断でき、確認時間が短い
パターン②	7分	8分	7分	22分	- 事前確認・事後検証も手順書どおり実施するため時間がかかる - スキルに依存せず誰でも実施可能という別の価値がある
パターン③	-	-	-	8分	- 設定・事前確認・事後検証をSkillsに基づき自律的に実行 - スキルレス基盤があるからこそAIが高速かつ正確に動作できる

パターン①	高度スキル保持者	+	従来 (CLI / Web GUI)
パターン②	一般スキル運用者	+	スキルレス基盤 (NSO)
パターン③	AIエージェント	+	スキルレス基盤 (NSO)

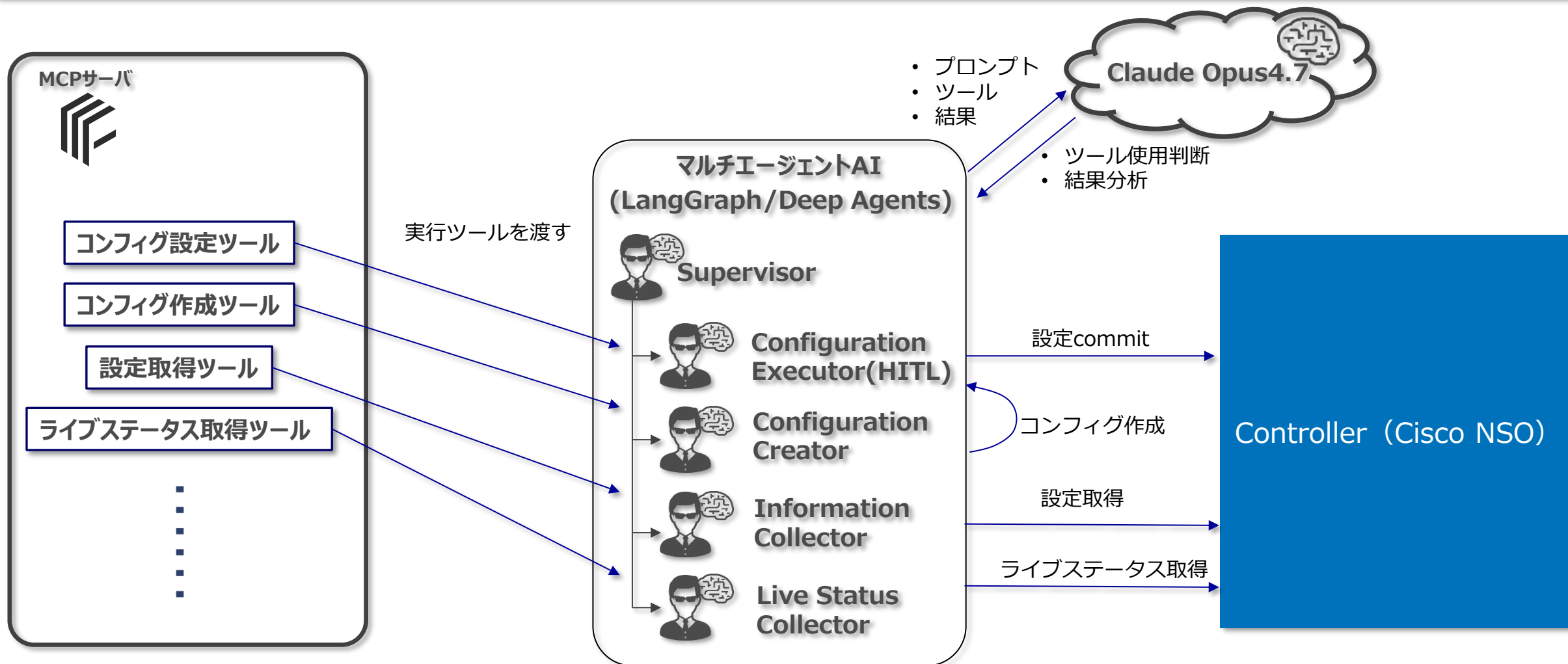
アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
4. 情報取得一元化API
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント

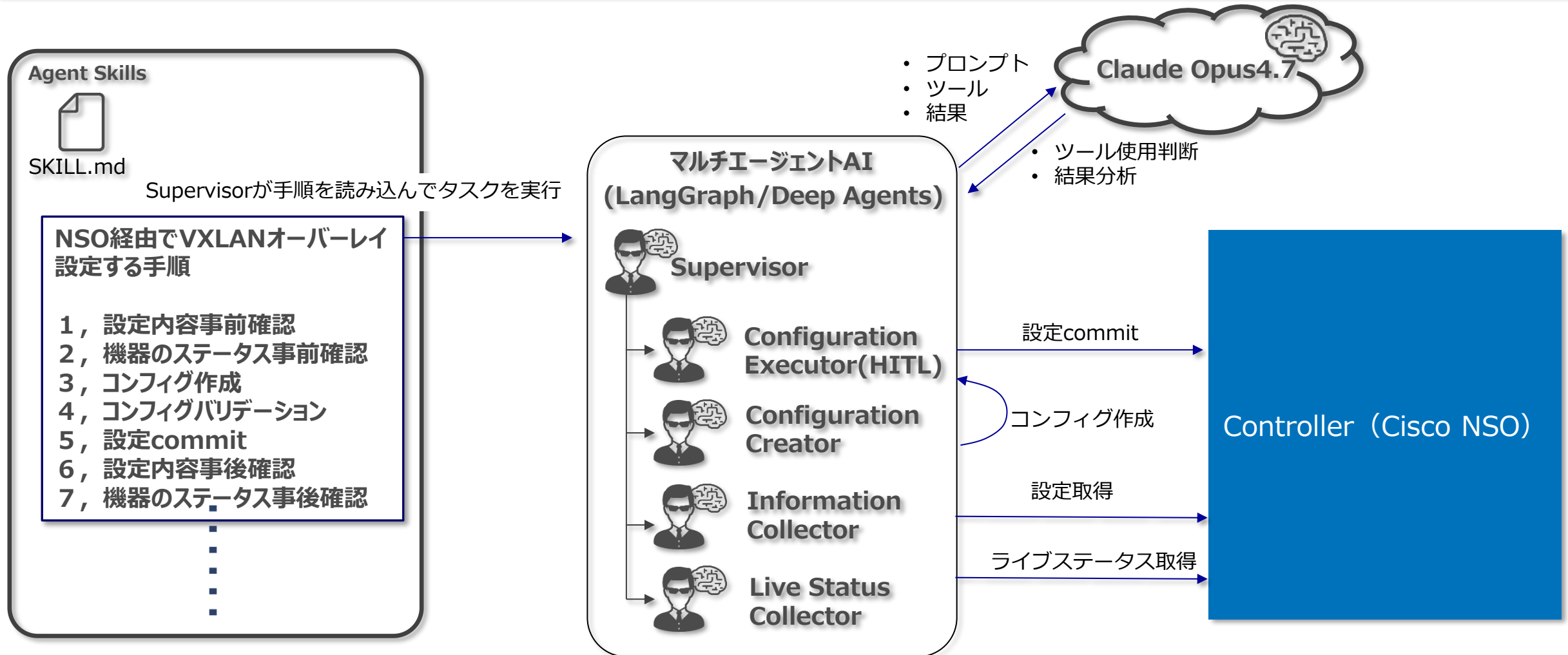
コントローラAPIのデモ（構成）



MCPサーバは、AIエージェントにNSOの各機能（取得・生成・実行）の実行ツールを提供する基盤である。



Skillsは、AIエージェントがネットワーク設定に必要な一連の作業（確認・作成・設定・検証）を再利用可能な手順として定義したものである。



大阪と兵庫でVXLANオーバーレイ

アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
4. 情報取得一元化API
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント

属人的な分散DC運用をスキルレス基盤で解消し、その基盤がAIエージェントの自律運用にもつながることを仮説として設定し、開発・検証を通じて証明した。

- 課題
 - ✓ 分散DC運用は属人化し、少数の高スキル人材に依存
 - ✓ マルチレイヤ／マルチベンダ／可視化分断で運用が複雑
- 仮説
 - ✓ スキルレス基盤はAIエージェントの実行基盤になる
- 解決（スキルレス基盤）
 - ✓ マルチレイヤ制御統合API（少数パラメータで設定）
 - ✓ トランザクション一括管理（自動ロールバック）
 - ✓ 情報取得一元化API（E2Eで状態取得）
- 検証結果
 - ✓ 一般スキル運用者は高スキル保持者と同等品質（スピードは遅い）
 - ✓ AIエージェントは高スキル保持者と同等品質かつ人手より早い
- 結論
 - ✓ 品質とスピードを両立する運用は、AIエージェント化により実現できる
 - ✓ スキルレス基盤はAI自律運用の基盤となる

スキルレス基盤を起点に、AIエージェント活用の深化と、他技術領域への応用の2つの軸で取り組みを進める。

① AIエージェント活用の拡張

- ✓ 設定・運用領域の拡大（光波長パス・VXLANに加え、障害対応・最適化）
- ✓ 自律制御の高度化（Skills拡張、マルチエージェント連携）
- ✓ 人とAIが協調する運用モデルの確立

② 他技術領域への応用

- ✓ 今回の「スキルレス基盤＝AIエージェント実行基盤」の考え方を、他のネットワーク技術にも展開
- ✓ マルチレイヤ・マルチベンダ環境一般への汎用化

③ コミュニティへの還元

- ✓ 知見・課題・解決アプローチの継続的な共有
- ✓ 議論を通じたスキルレス運用・AI活用の促進

アジェンダ

1. 背景と課題
2. マルチレイヤ制御統合API
3. トランザクションの一括管理
4. 情報取得一元化API
5. 比較検証
6. コントローラAPI／AIエージェントデモ
7. まとめと今後の展望
8. 議論ポイント

- 運用の属人化、どう向き合っていますか？
 - ベテラン頼みになっていませんか？
 - スキルトランスファーや自動化、どこまで進められていますか？
- AIEージェント、もう触っていますか？
 - 取り入れていますか？まだ様子見ですか？
 - 触ってみて「これは使える」「これは無理」と感じたのはどんなところですか？
- AIのハルシネーション、怖くないですか？
 - 商用環境で使うとしたら、何があれば安心できますか？
 - 「これがないと無理」という前提条件は何ですか？
- AIEエージェントを動かす土台、何を整備しましたか？
 - API、トランザクション管理、ライブステータス取得、ドキュメント……
 - まず何から手をつけるべきだと思いますか？
- AIと人、どう役割分担しますか？
 - AIに任せたい領域・任せたくない領域は？
 - 運用者の役割はどう変わっていくと思いますか？