



## 概要

---

- モデル駆動型プログラマビリティについて（1ページ）
- プログラム可能なインターフェイスインフラストラクチャについて（2ページ）

## モデル駆動型プログラマビリティについて

Cisco NX-OS デバイスのモデル駆動型プログラマビリティ（model-driven programmability、MDP）により、デバイスの構成と制御を自動化できます。また、デバイスの状態変化をモニターすることもできます。

データモデリングでは、手作業の構成プロセスの代わりに、プログラムを使用した標準ベースの方法でネットワーク デバイスの構成を記述します。データ モデルは、業界で定義された標準言語で記述されます。CLI を使用した設定のほうがわかりやすいですが、データ モデルを使用して設定を自動化すると拡張性が向上します。

Cisco NX-OS デバイスは YANG データ モデリング言語をサポートしています。YANG は、ネットワーク デバイスの構成データと運用データ、リモート プロシージャ呼び出し、および通知を記述するために用いられるデータ モデリング 言語です。

次に、NX-OS でサポートされる標準ベースのプログラマブルインターフェイスを示します。

- NETCONF
- RESTCONF
- gRPC

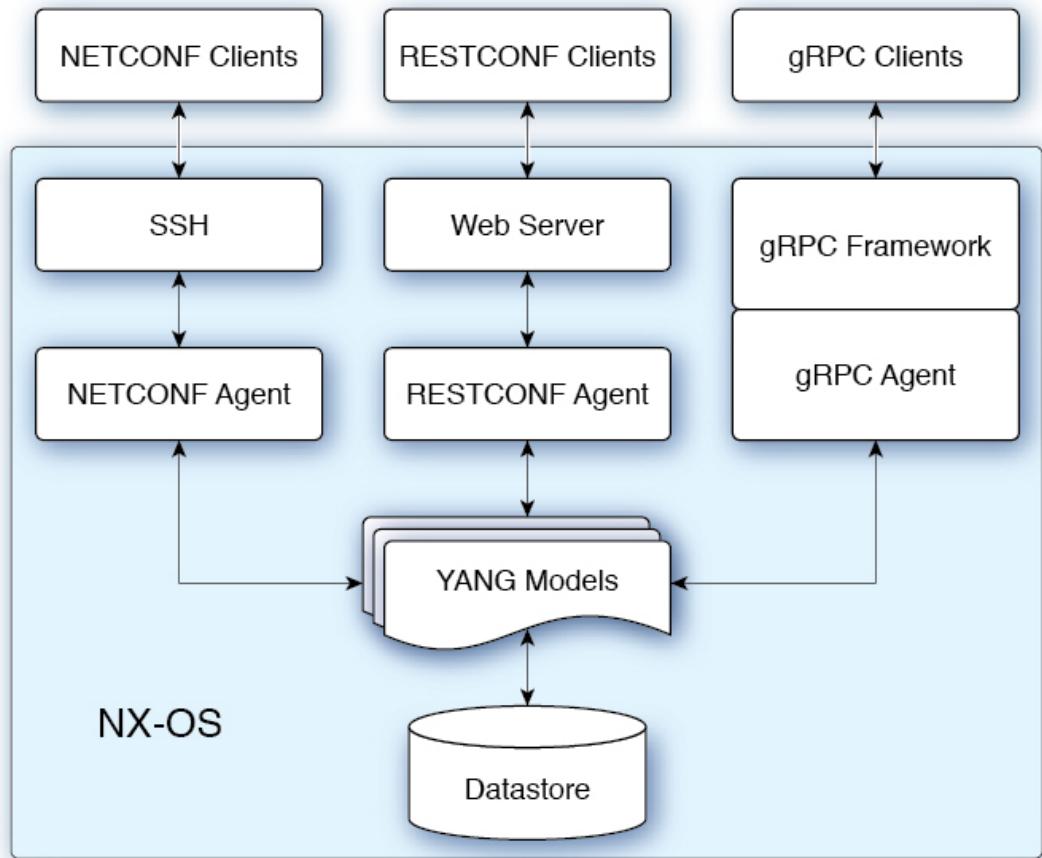
NX-OS でサポートされるオブザーバビリティインターフェイスは次のとおりです。

- NETCONF
- gRPC
- テレメトリ

■ プログラム可能なインターフェイス インフラストラクチャについて

# プログラム可能なインターフェイス インフラストラクチャについて

ここでは、NX-OS プログラマブルインターフェイス インフラストラクチャの概要について説明します。



NETCONF、RESTCONF または gRPC のどの方法で受信した場合でも、要求は、抽象メッセージオブジェクトに変換されます。そのメッセージオブジェクトは、要求内のモデル名前空間に基づいて、基盤となるモデルインフラストラクチャに配布されます。名前空間を使用すると、適切なモデルが選択されて、要求は処理のために渡されます。モデルインフラストラクチャは、基礎となる NX-OS のデバイスデータストアで要求（読み取りまたは書き込み）を実行します。結果は要求元のエージェントに返され、要求元のクライアントに応答が送信されます。

## NX-OS プログラマビリティエージェント

エージェントは、NX-OS デバイスと外部クライアント間のインターフェイスを提供します。ネットワーキングデバイスとの通信の特定のトランスポート、プロトコル、およびエンコーディングを指定します。NX-OS プログラマブルインターフェイスは、NETCONF、RESTCONF、および gRPC の 3 つのエージェントをサポートします。それぞれが YANG モデルを介する構成管理に異なるインターフェイスを提供します。

表 1: NX-OS プログラマブルインターフェイス エージェント

| エージェント   | トランスポート | プロトコル<br>(Protocol)                           | エンコーディング<br>(Encoding) |
|----------|---------|---|------------------------|
| NETCONF  | SSH     | 該当なし  | XML                    |
| RESTCONF | HTTP    | draft-ietf-netconf-restconf-10 <sup>[1]</sup> | XML または JSON           |
| gRPC     | HTTP    | gRPC プロトコル <sup>[2]</sup>                     | Google Protobuf        |

プロトコルの仕様については、次のドキュメントを参照してください。

- RESTCONF エージェント
- gRPC エージェント

## NX-OS オブザーバビリティエージェント

スイッチイベントをモニターするための、読み取り専用ストリーミングインターフェイスが存在することに注意してください。NX-OS は、標準ベースの NETCONF と gRPC、および独自のソフトウェアテレメトリをサポートします。

表 2: NX-OS オブザーバビリティインターフェイスエージェント

| エージェント  | トランスポート       | プロトコル (Protocol) |
|---------|---------------|------------------|
| NETCONF | SSH           | 該当なし             |
| gRPC    | HTTP          | gRPC プロトコル       |
| テレメトリ   | HTTP、UDP、GRPC | NA               |

## モデルインフラストラクチャ

モデルインフラストラクチャは、エージェントから受信した要求を受け取り、要求内の YANG モデルに関連付けられている名前空間を決定し、名前空間に一致するモデルコンポーネントを選択して要求を処理します。選択したモデルコンポーネントが要求処理を完了すると、処理結果が要求元のエージェントに送信され、クライアントに送り返されます。モデルインフラストラクチャは、エージェントプロトコルの指定に従って、認証、ハンドシェイクなどを含むプロトコル開始要求の処理も行います。

## ネイティブ（デバイス）YANG モデル

デバイス構成を記述している YANG モデルは、デバイス YANG モデルと呼ばれます。デバイスモデルは、デバイス名前空間の他のモデルコンポーネントのように、モデルインフラストラクチャに明示されます。

## 共通（OpenConfig）YANG モデル

共通モデルは、パブリックコミュニティによって定義される別の種類のモデルコンポーネントです。共通モデルの YANG パスは、同等のデバイスモデル要素に変換できます。これら同等のデバイス モデル要素は、デバイス YANG コンテキストでデバイス モデルデータの読み取りと書き込みに使用されます。



(注) Cisco NX-OSは、中央集中型のDME データベースを維持します。デバイスとOpenConfig YANG は、DME データベースのラッピング表現です。これにより、すべてのプログラマビリティインターフェイスで一貫性が保証されます。ユーザーが異なるインターフェイスを使用していても、同じスイッチの構成/状態が矛盾なく表示されます。

## その他の YANG 関連資料

YANG の詳細については、<https://handwiki.org/wiki/YANG>を参照してください。

## 翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。