

OpenStack運用実践による
エンタープライズ適用に向けた課題と
今後について
～ベアメタルサービスの現状と
Design Summitでの討議状況～

2015年7月13日
富士通株式会社
椎名宏徳

- Ironicの概要説明
- ベアメタルサービスの現状と
Design Summitでの討議状況

Ironicの概要

■ エンタープライズ適用に向けて必要な機能

■ 仮想化に適さないワークロードへの対応

- I/O負荷が高く、性能要求の厳しいシステム(データベースなど)

→ベアメタルが必要

■ 仮想マシンとベアメタルの共通管理

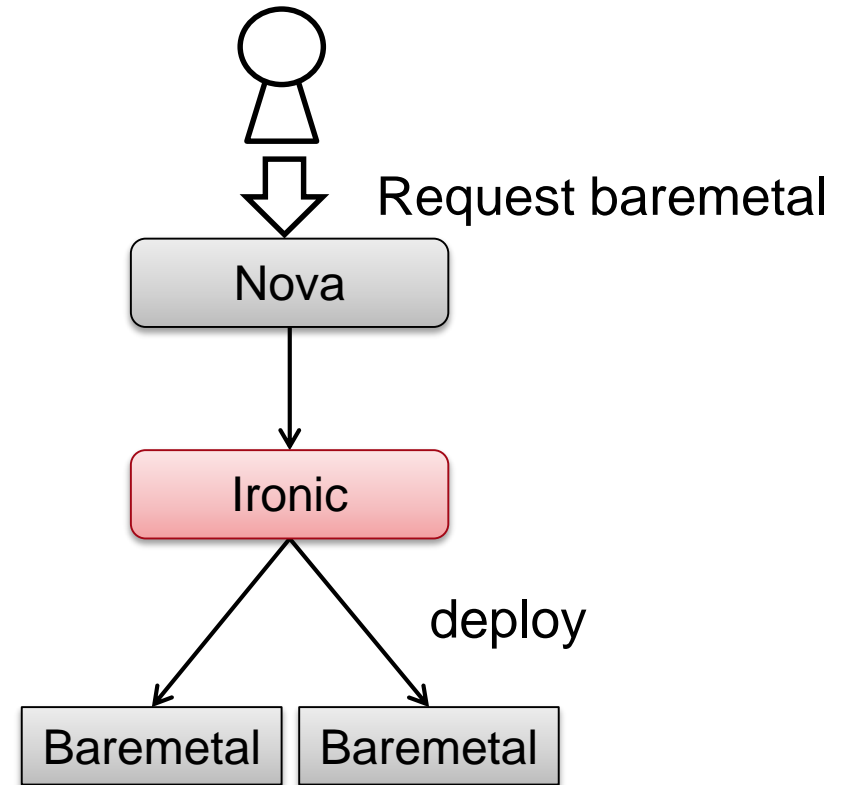
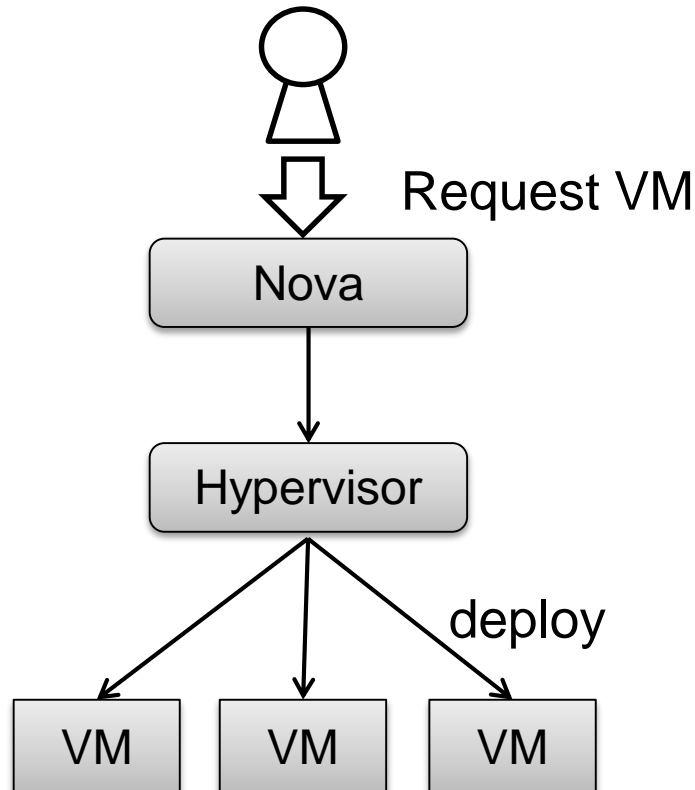
■ OpenStackのベアメタルサービス: Ironic

■ OpenStack Kiloで正式にリリース

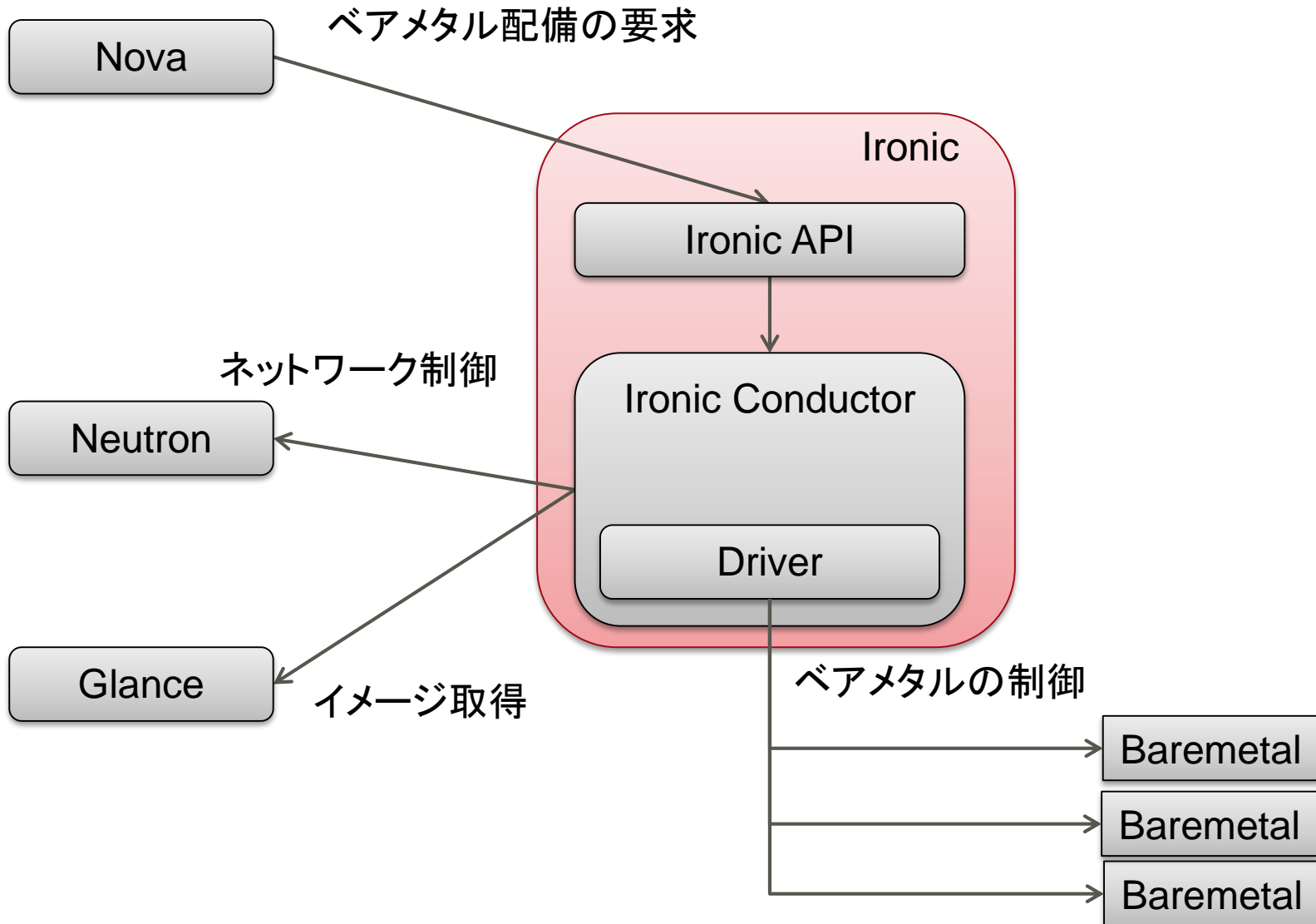
■ Kiloでの修正規模

- 機能追加数: 24件
- バグ修正数: 170件
- ソース追加: 約27,000行

- ベアメタルサービスを提供するコンポーネント
- 利用者は、VM配備と共通の操作でベアメタル配備可能

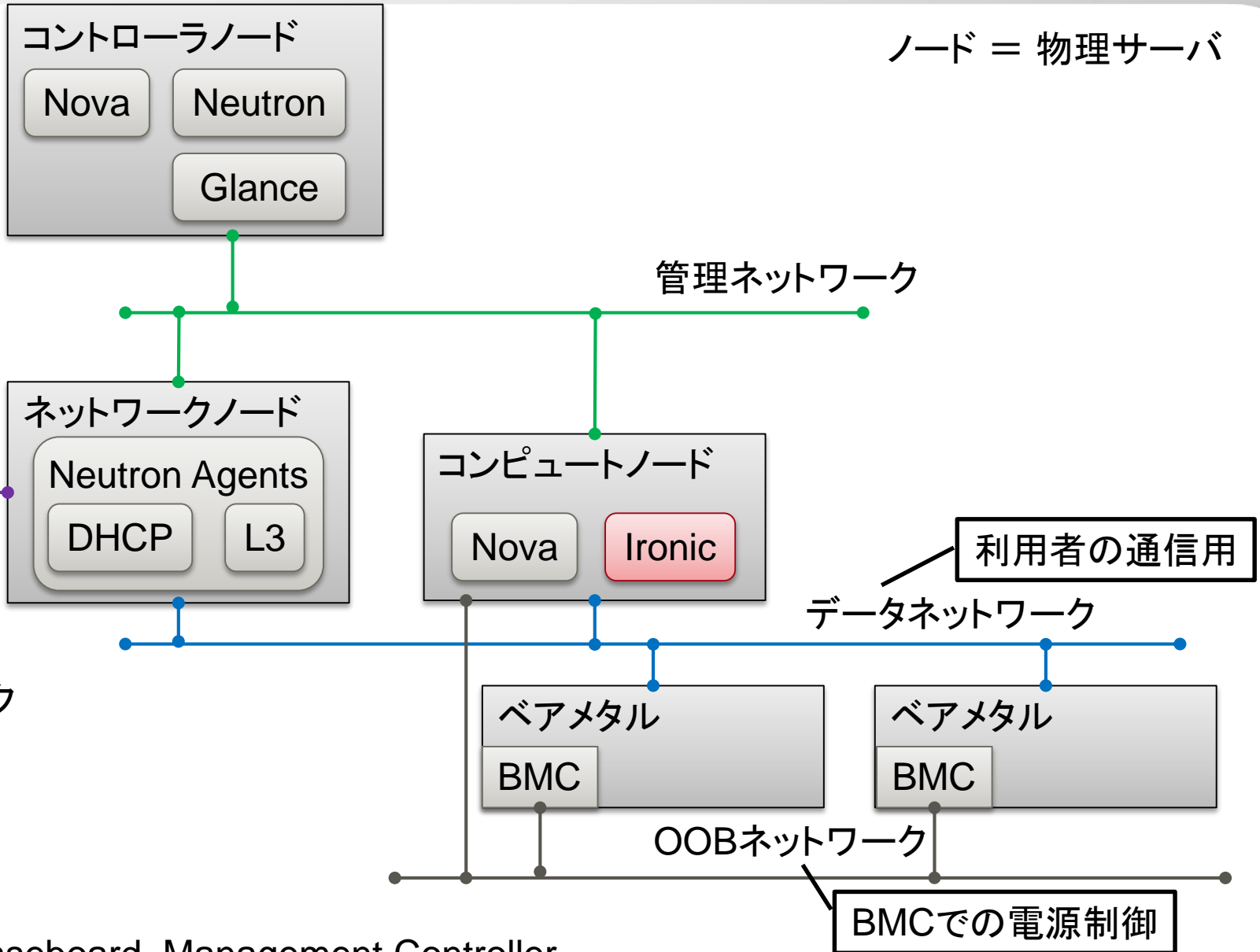


Ironicの構成



- Ironic Conductorに、ベアメタルを制御するインタフェースを提供
- 標準のドライバはIPMIを利用
- ハードベンダも固有のドライバを開発
(性能向上や追加機能のため)
 - iRMC driver (富士通)
 - iLO driver (HP)
 - DRAC driver (Dell)
 - etc...
- ベアメタル配備の方式(ドライバで実装)
 - PXEブート
 - Ironic Python Agent
 - Virtual media機能による配備

システムの全体像

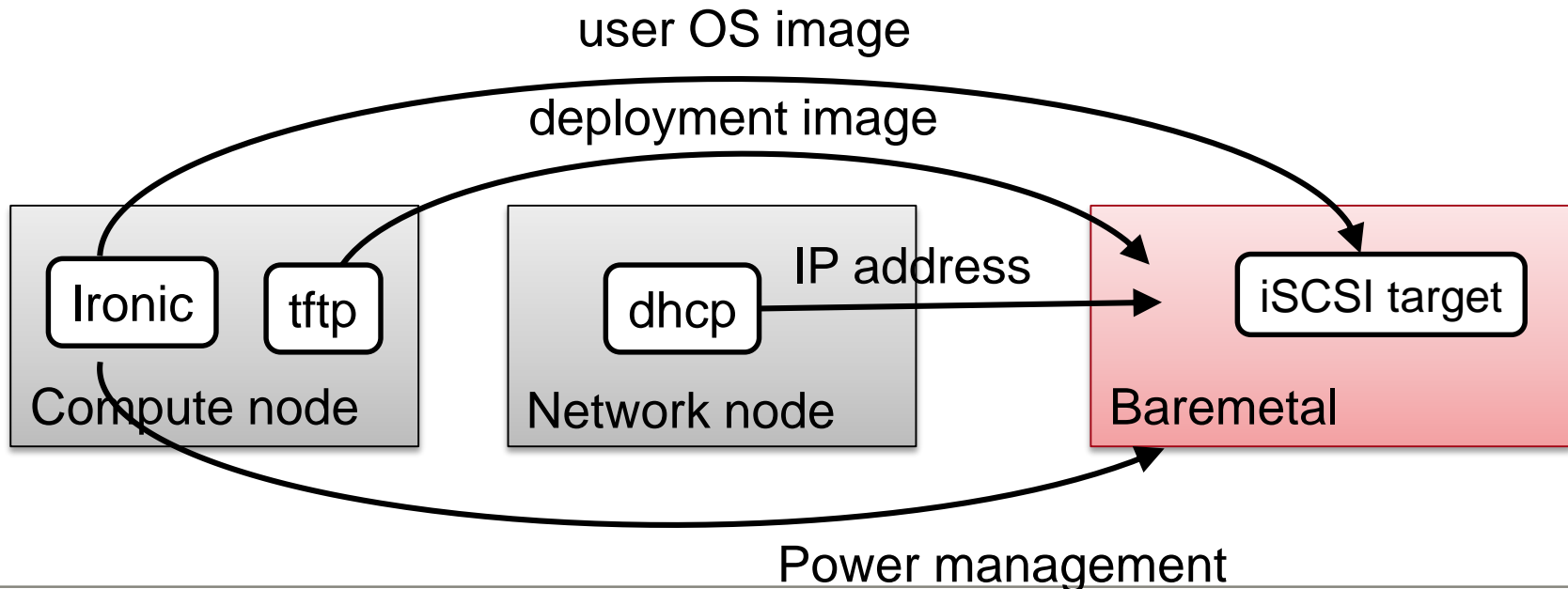


BMC: Baseboard Management Controller

- ベアメタルをIronicに登録
 - 使用するドライバ、BMCの認証情報
 - ベアメタルのスペック情報（CPU数、メモリ量など）
 - MACアドレス
- ベアメタル用のフレーバを作成
(利用者はフレーバを選択して、ベアメタルの配備を要求)
- ベアメタル用のイメージを作成
 - デプロイ用イメージ（配備時のみ利用）
 - bm-deploy-kernel, bm-deploy-ramdisk
 - ユーザ用イメージ
 - user-image, user-image-vmlinuz, user-image-initrd

ベアメタル配備の流れ

1. Ironicがベアメタルの電源を入れる
2. ベアメタルがデプロイ用イメージを取得してブート
3. デプロイ用イメージにより、iSCSIをセットアップ
4. Ironicがユーザ用イメージをベアメタルにコピー
5. Ironicがベアメタルを再起動
6. ベアメタルがユーザ用イメージで起動



ベアメタルサービスの現状と Design Summitでの討議状況

- 普段からIRCミーティングで貢献を積んでいる開発者が次のリリースの大方針、デザインなどを決定する場
 - 10以上のセッションが並列開催
 - 1セッションから5名～50名の参加者
- 座談会方式で、自由に発言
 - 最前列の開発者が中心になって議論
 - 人気セッションでは、PTLやコア開発者を捕まえられない
- セッションでのアジェンダ登録は、事前にIRCミーティングで議論して、EtherPadに登録

Ironic-Neutron integration sessionの様子



ベアメタルの要件とIronicの対応状況

要件	可否	課題
仮想マシンと同一APIでの配備	○	
OSプロビジョニング	○	
SAN Boot構成	×	Cinderとの連携
サーバ故障時の復旧	×	
ボリューム割り当て	×	
マルチテナント対応	×	Neutronとの連携
仮想マシンとのネットワーク接続	×	

今回の発表の範囲

■ Design Summitの目玉テーマの一つ

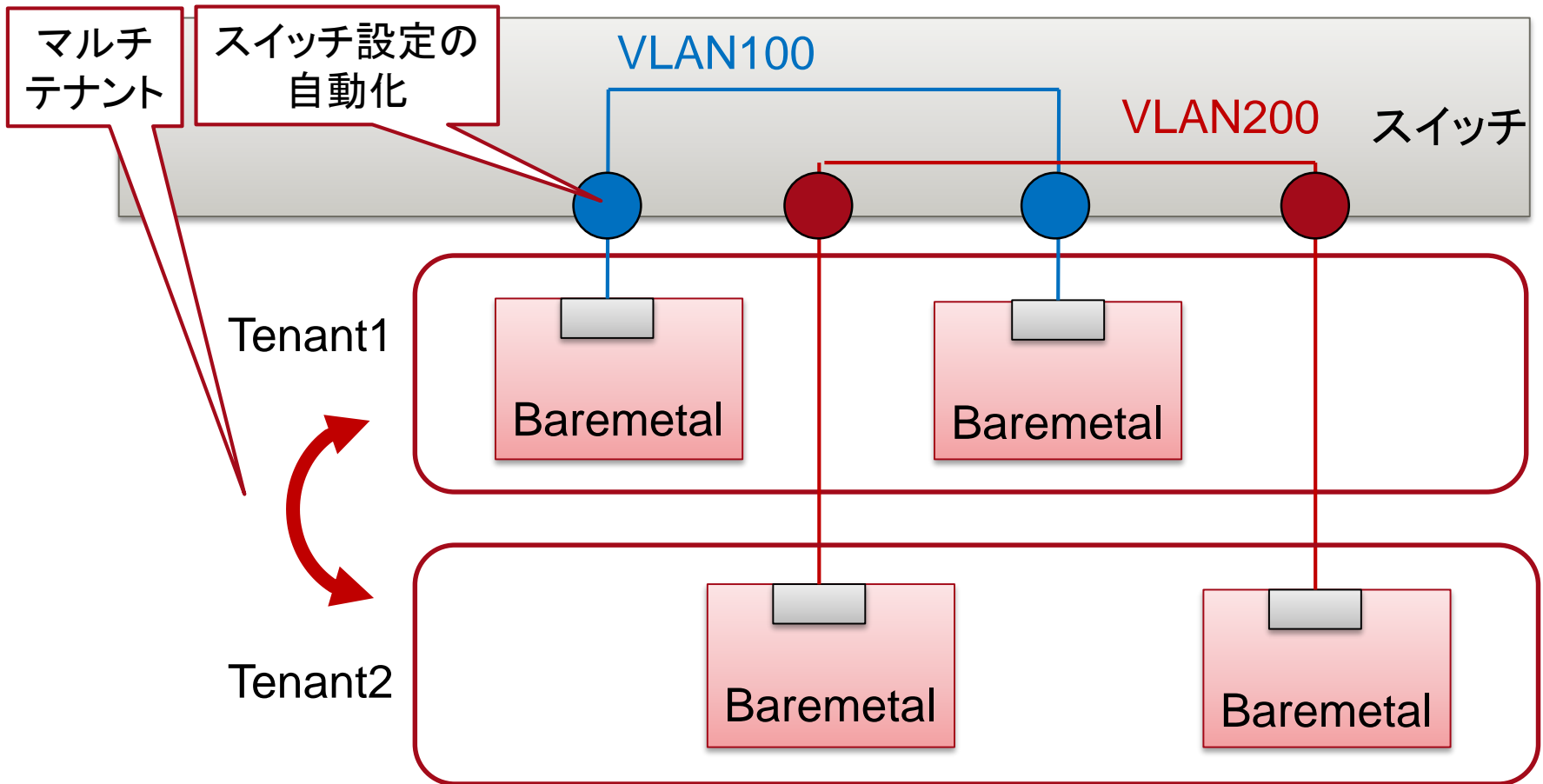
- マルチテナント対応

- 仮想マシンとのネットワーク接続

→新しいクロスプロジェクト“*Ironic-Neutron integration*”の立ち上げ

■ 実装方法の大枠について合意

- マルチテナント対応のためには、ネットワークの分離が必要
- ベアメタルの場合、物理スイッチの設定が必要
- 物理スイッチ設定の自動化を行う



■ 2種類のVLANを利用

■ 配備用VLAN

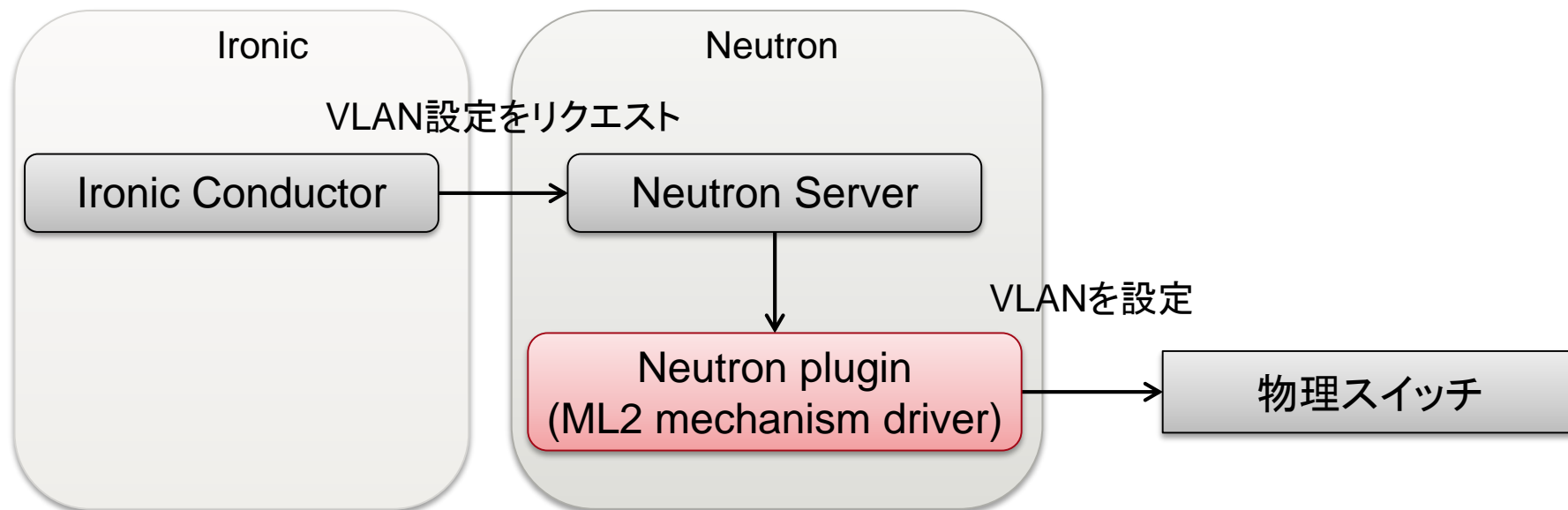
- 管理者が作成するNeutronネットワーク
- Ironicを接続するVLAN
- ベアメタルは配備時のみこのVLANに接続

■ テナント用VLAN

- テナント利用者が作成するNeutronネットワーク
- ベアメタルは配備後にこのVLANに接続

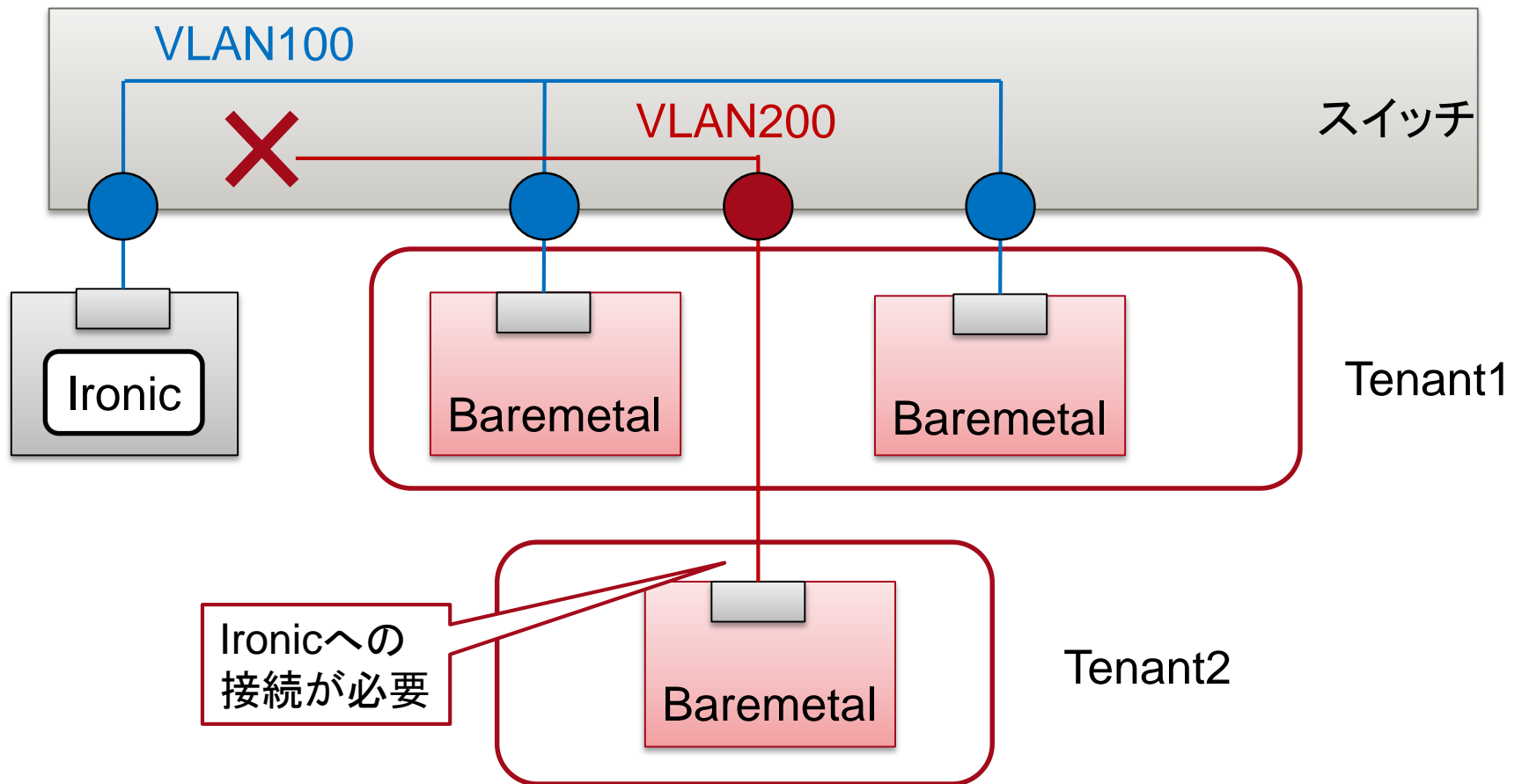
■ 配備の前後で2種類のVLANを切り替える

- Neutronのドライバで物理スイッチを制御
 - スwitchポートのVLAN設定を実施
- NeutronのML2メカニズムドライバを実装する方針

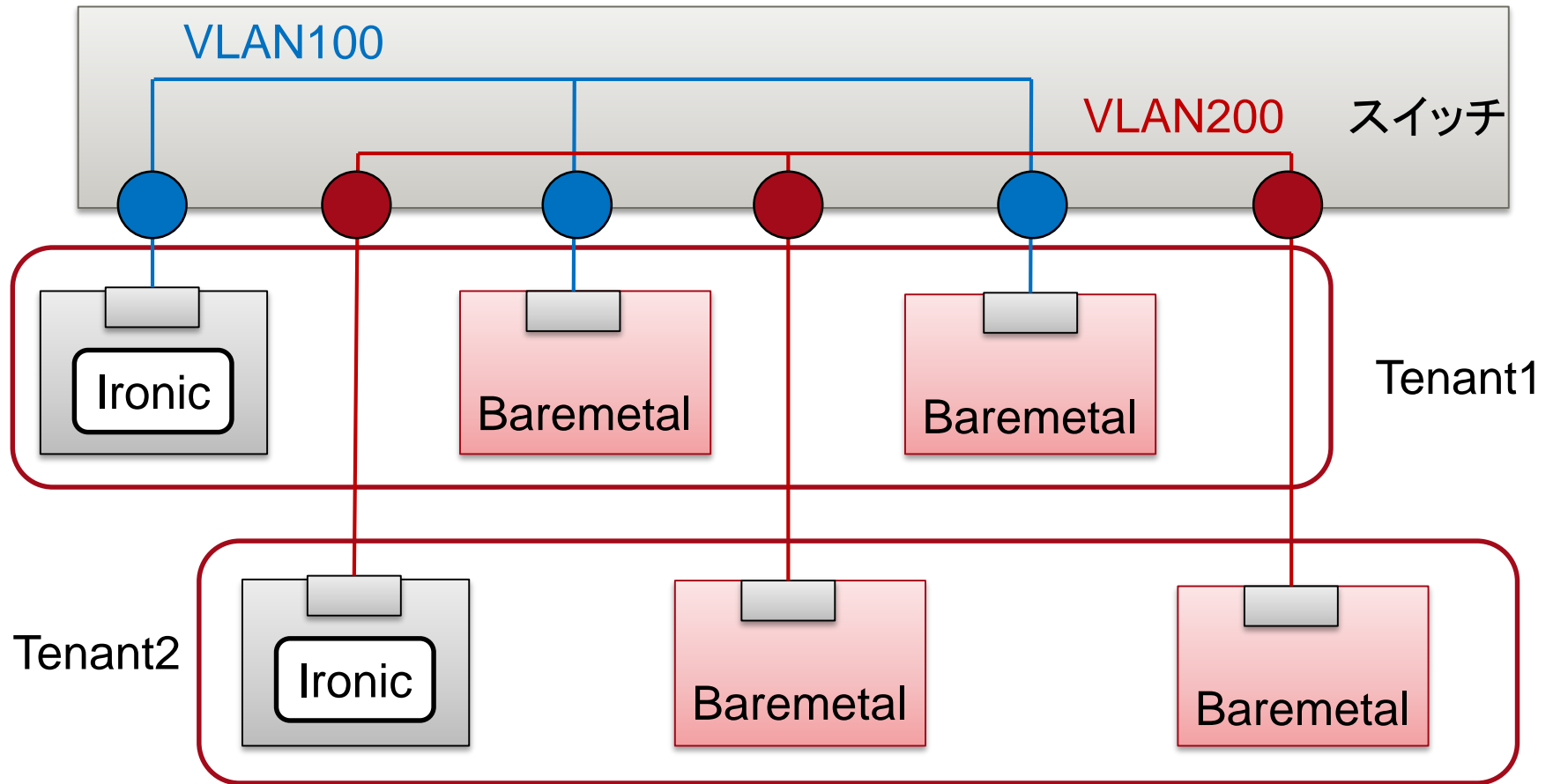


検討の過程(案1)

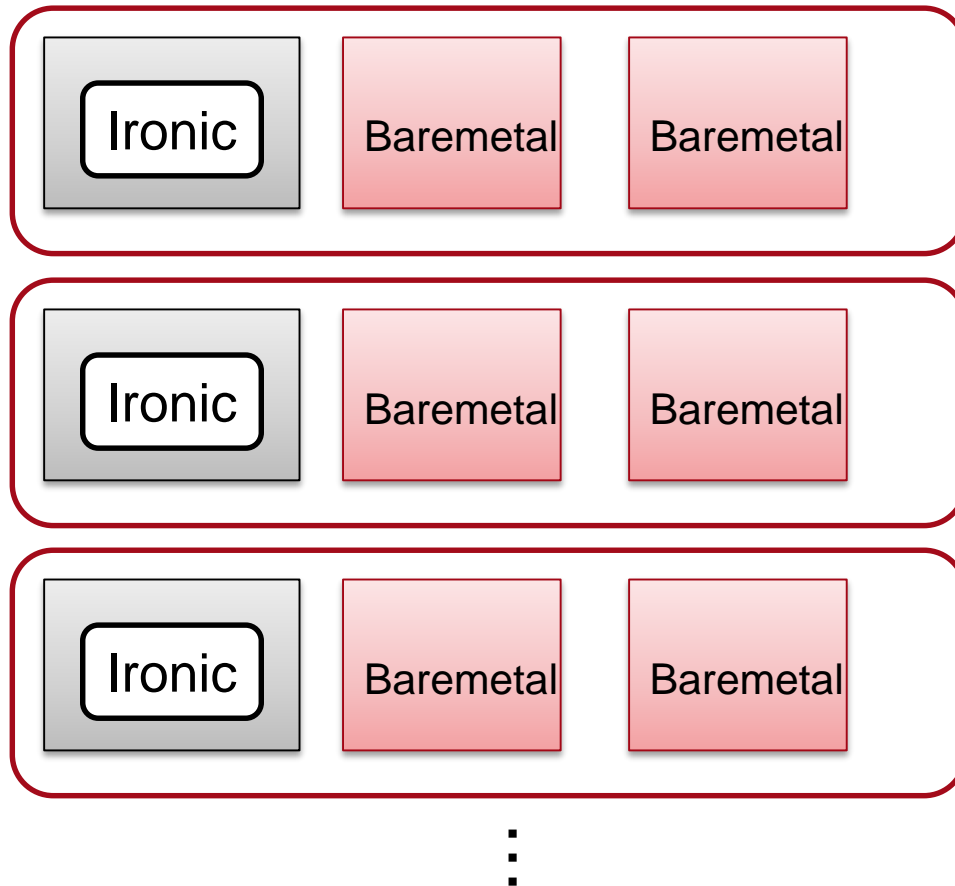
- 単純にVLANでネットワークを分割
- 配備するときに、どのベアメタルもIronicからOSイメージの取得が必要



■ テナントごとにIronicを配置

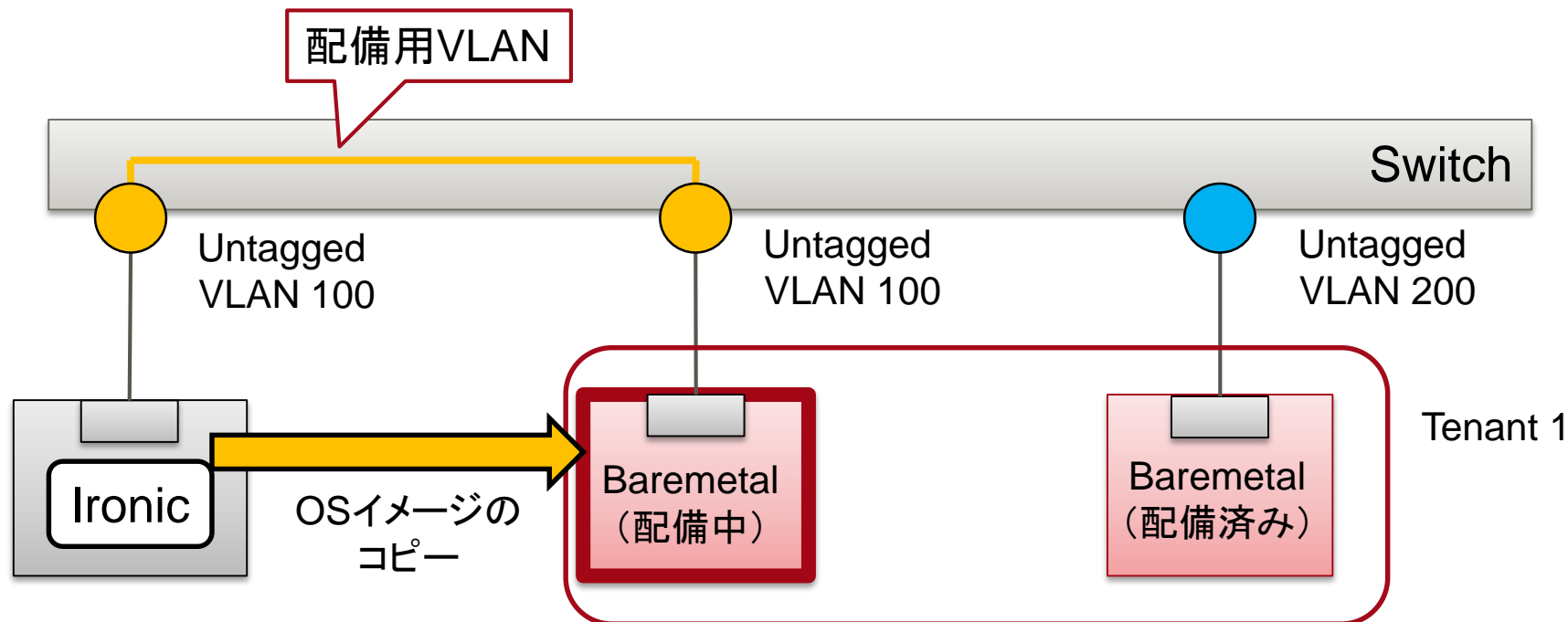


- テナントごとにIronic用のノードを用意するのは非現実的



解決方式の概要 (1/4)

- ベアメタルを配備するときは、配備用VLANを利用



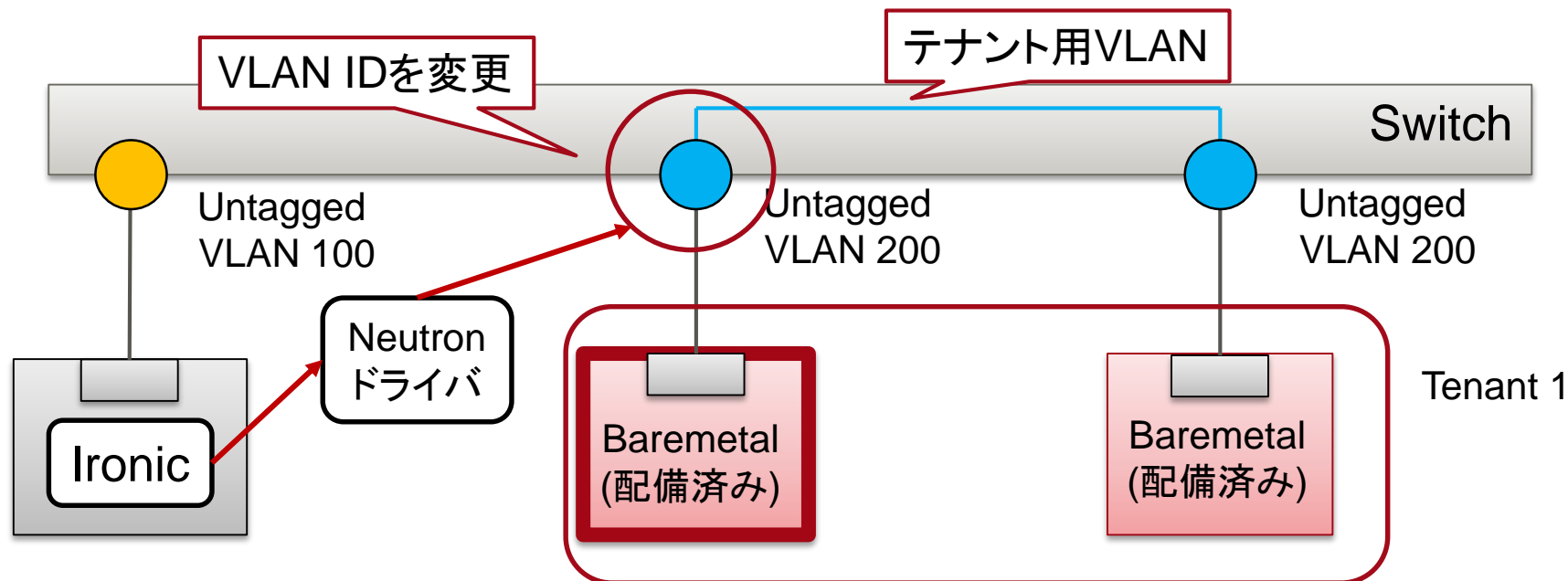
VLAN ID

● 100:Deployment

● 200:Tenant1

解決方式の概要 (2/4)

- 配備完了後に、IronicがベアメタルのVLAN設定を変更
(配備用VLANからテナント用VLANへ)



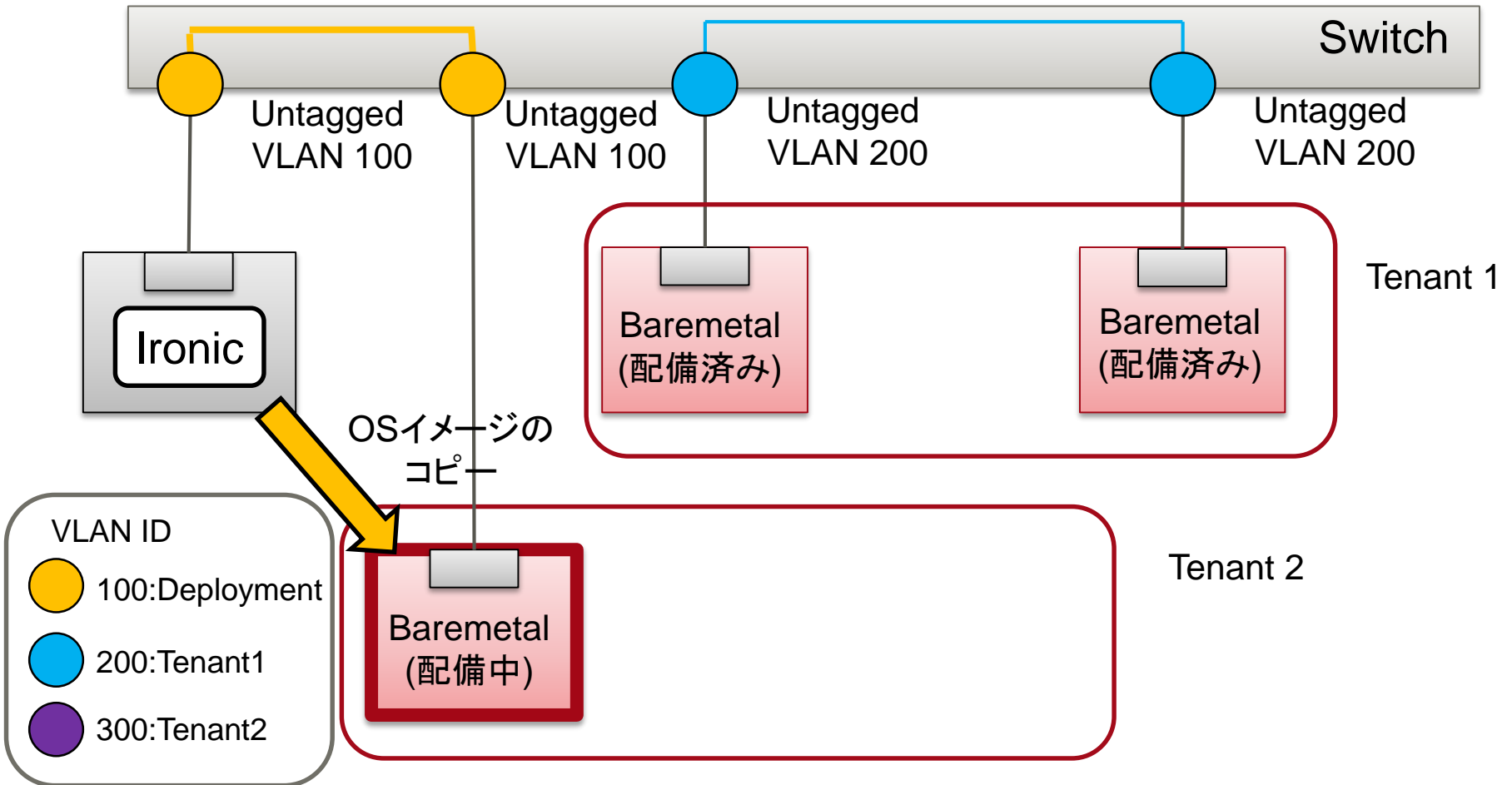
VLAN ID

● 100:Deployment

● 200:Tenant1

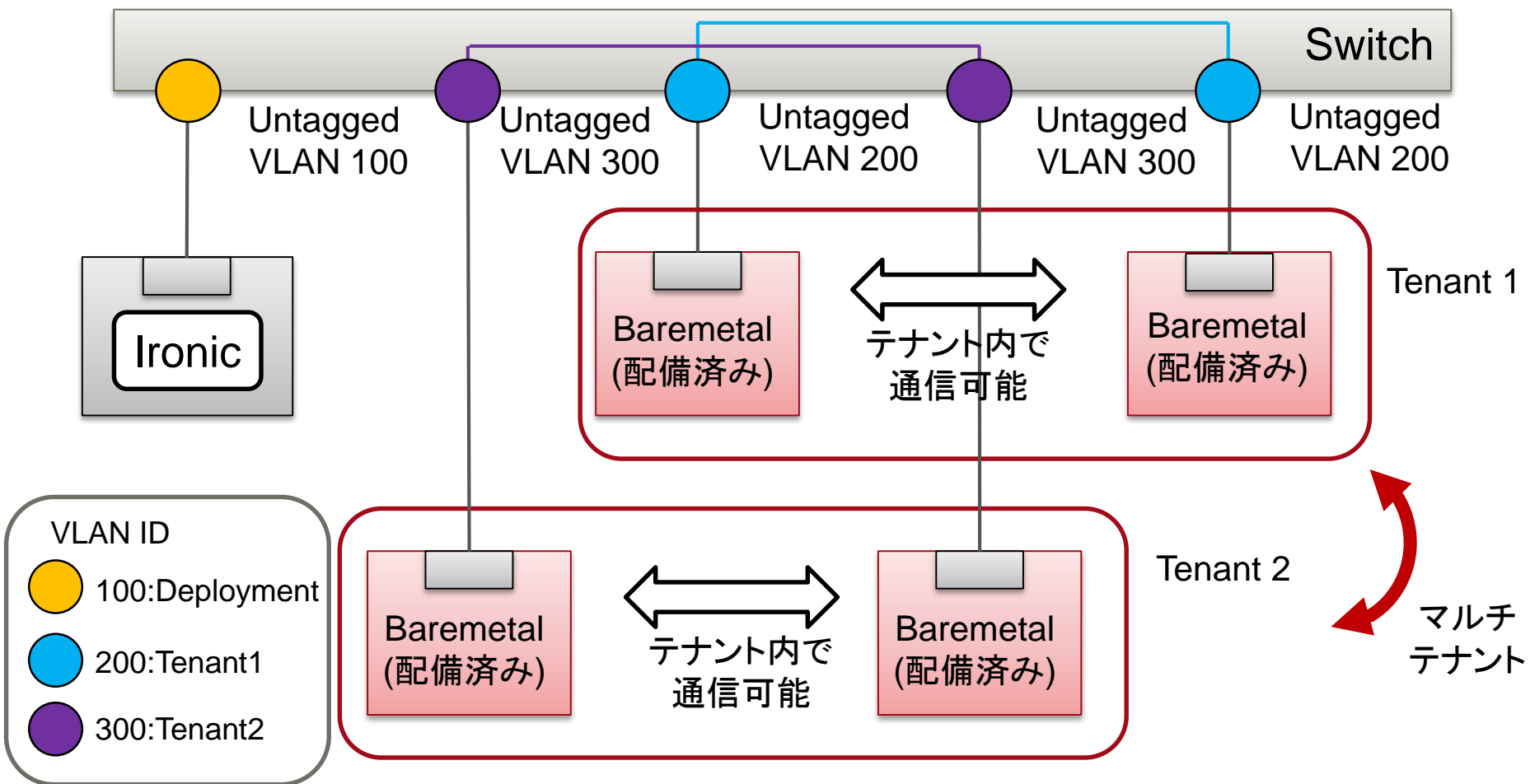
解決方式の概要 (3/4)

- 別のテナントも配備用VLANを使ってベアメタルを配備可能



解決方式の概要 (4/4)

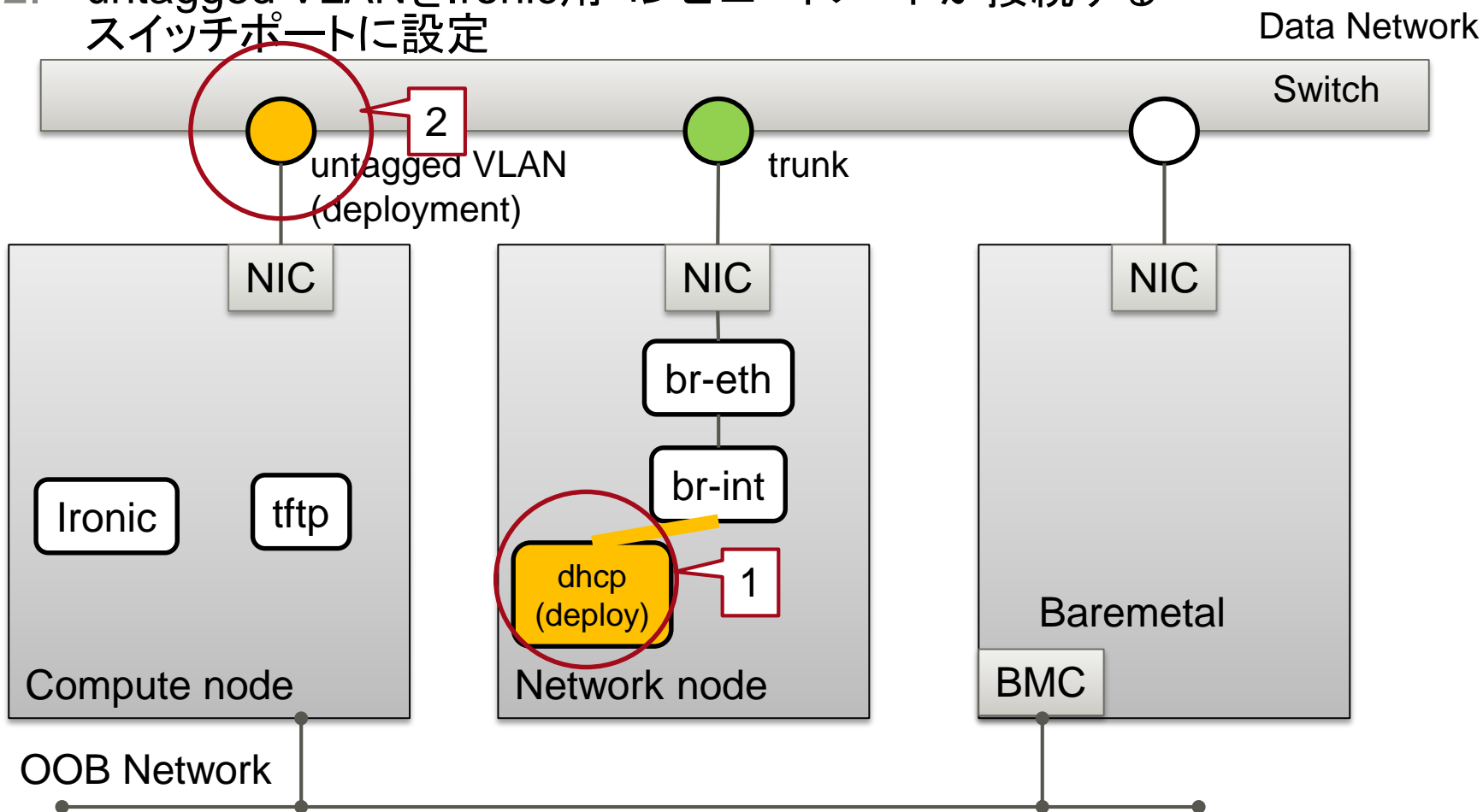
- VLANを切り替えることで、Ironicが全テナントのベアメタルを配備可能



配備の詳細（準備）

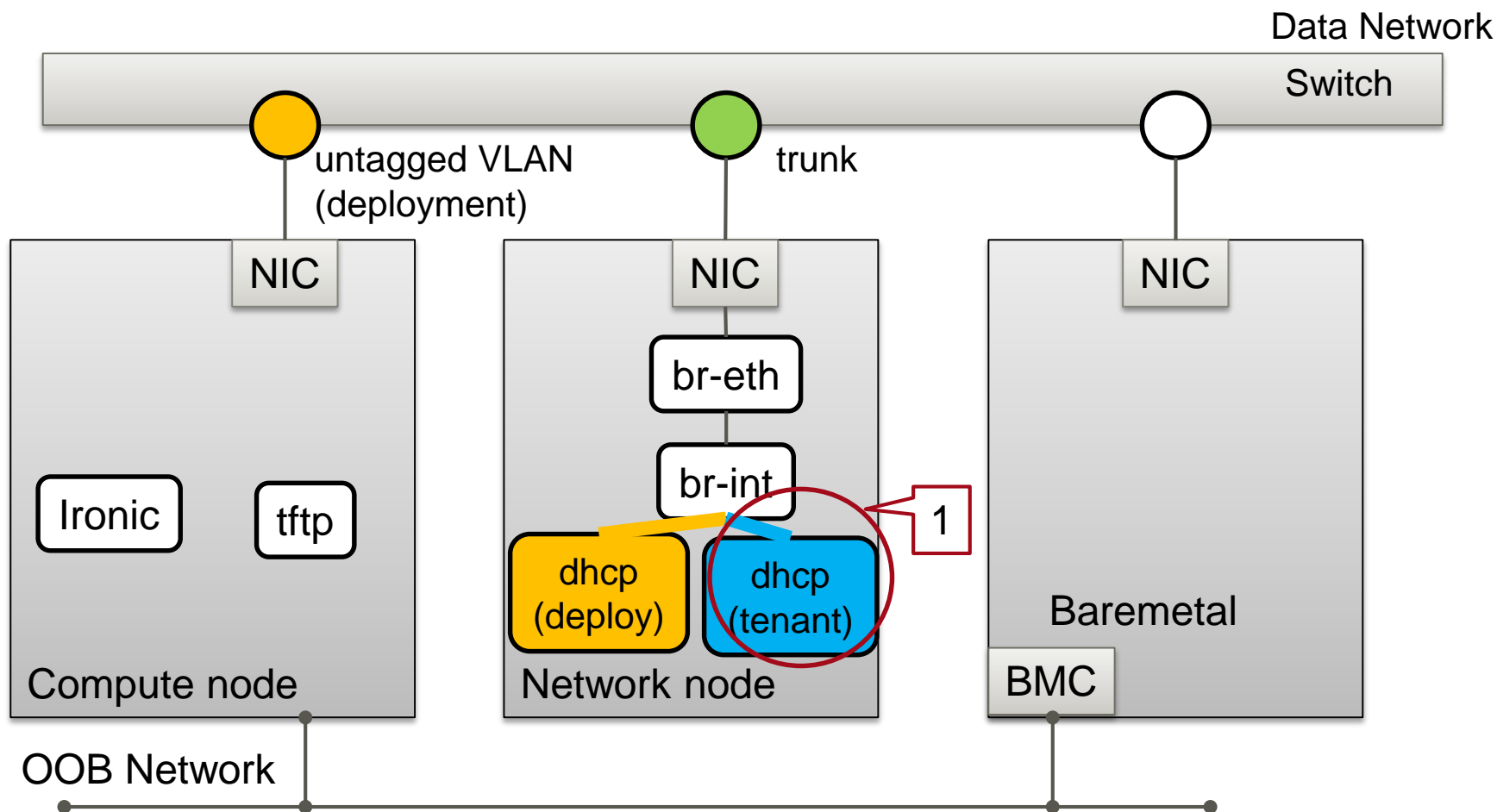
■ 管理者のオペレーション

1. Neutronネットワークを作成（配備用VLAN）
→ NeutronがDHCPサーバを作成
2. untagged VLANをIronic用コンピュータノードが接続する
スイッチポートに設定



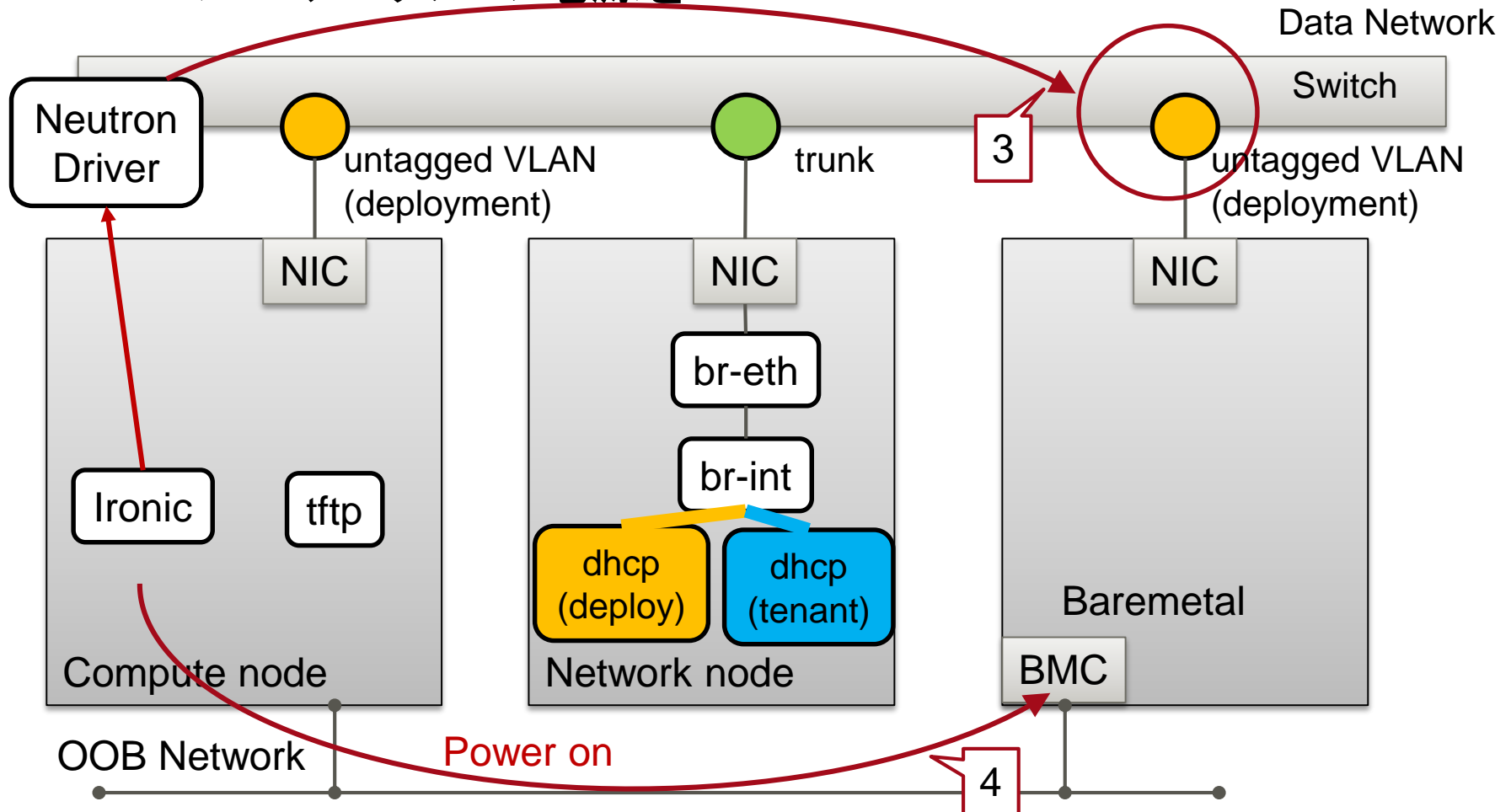
配備の詳細 (1/6)

1. テナント利用者がネットワークを作成(テナント用VLAN)
→NeutronがDHCPサーバを作成
2. テナント利用者がベアメタルの配備をリクエスト



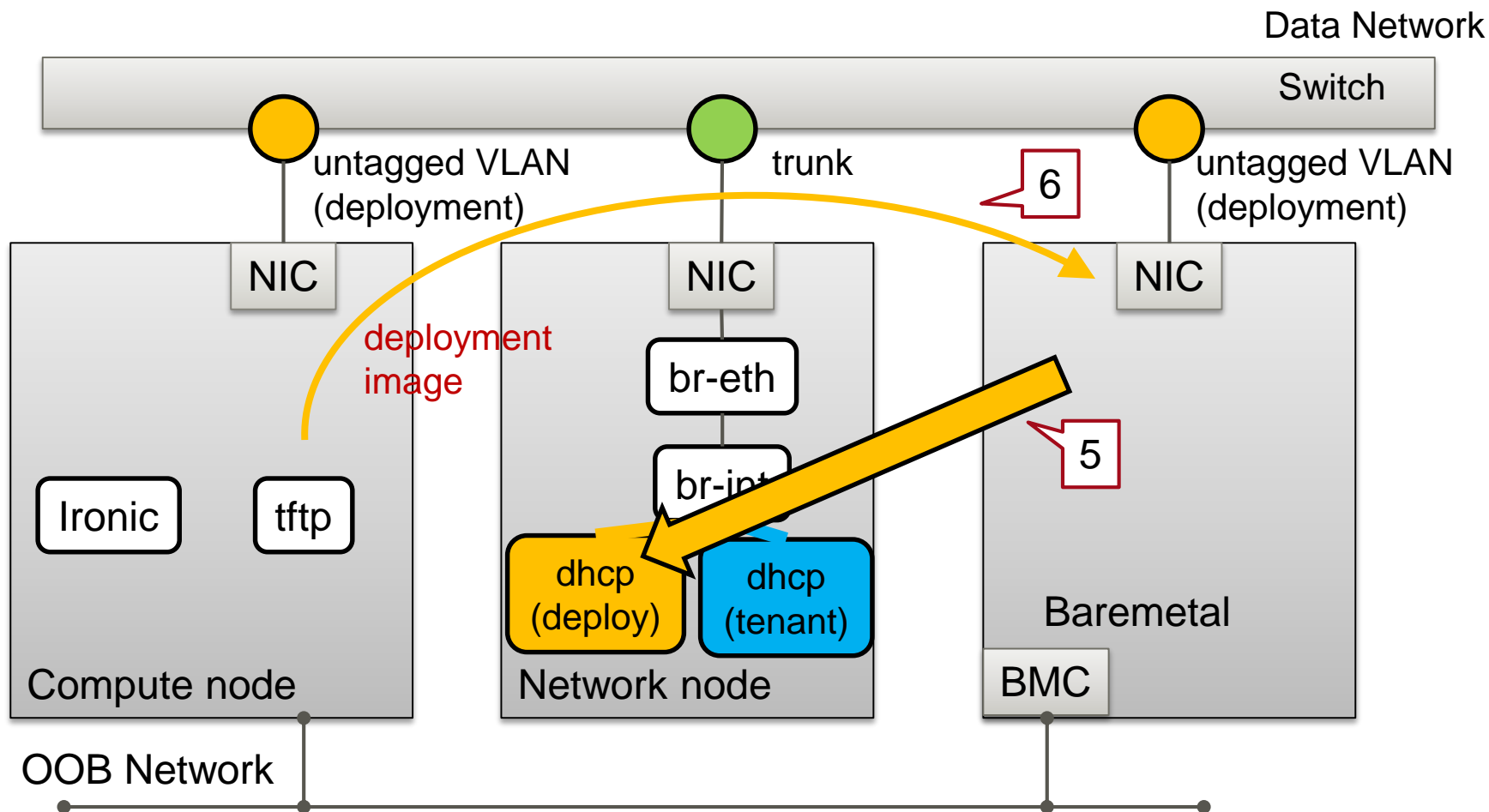
配備の詳細 (2/6)

- 3. Ironicが配備用VLANのVLAN IDをベアメタルが接続しているスイッチポートに設定
- 4. Ironicがベアメタルの電源をON



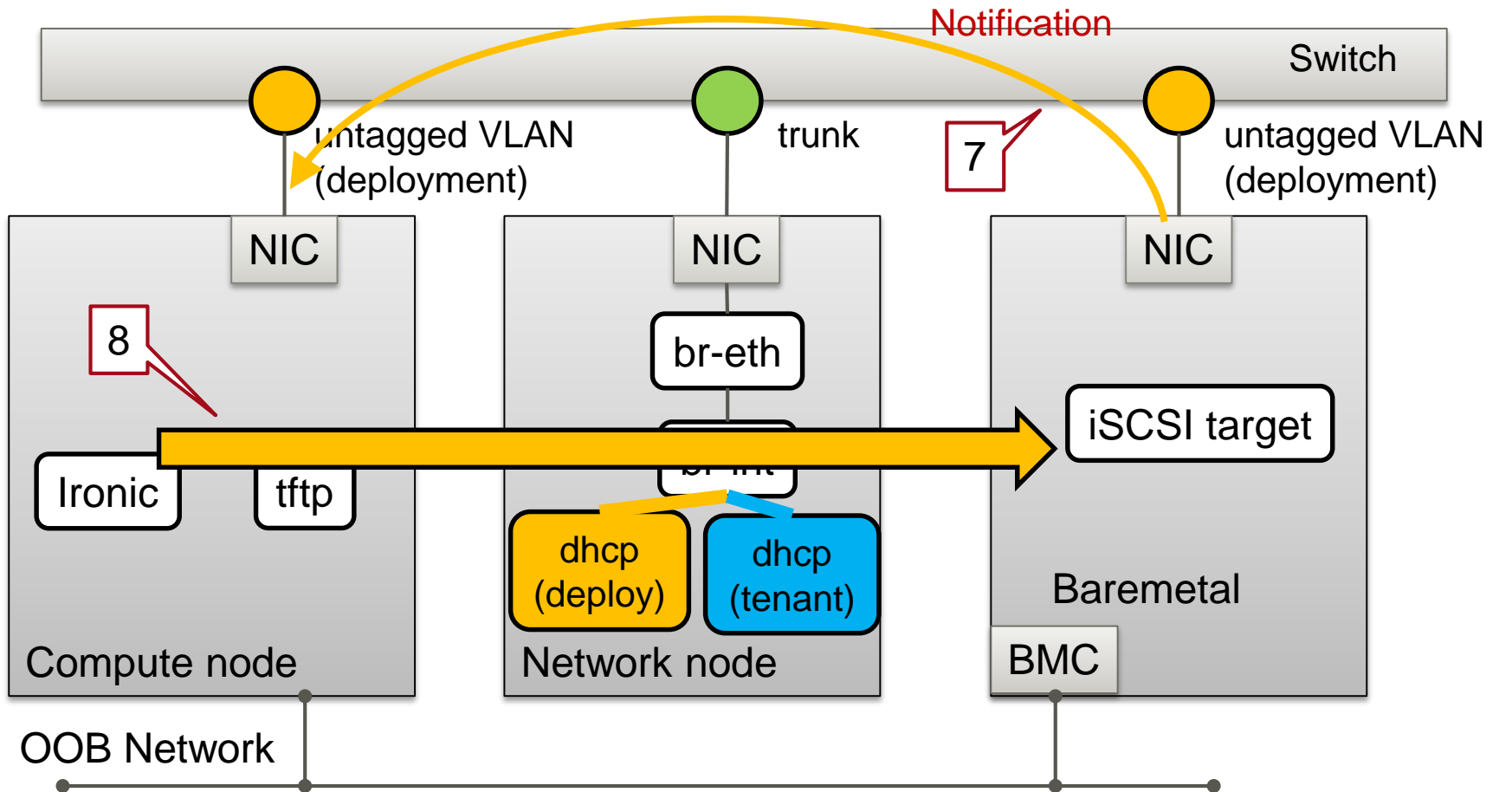
配備の詳細 (3/6)

- 5. ベアメタルが配備用VLANのDHCPサーバからIPアドレスを取得
- 6. ベアメタルがデプロイ用イメージをtftpサーバから取得



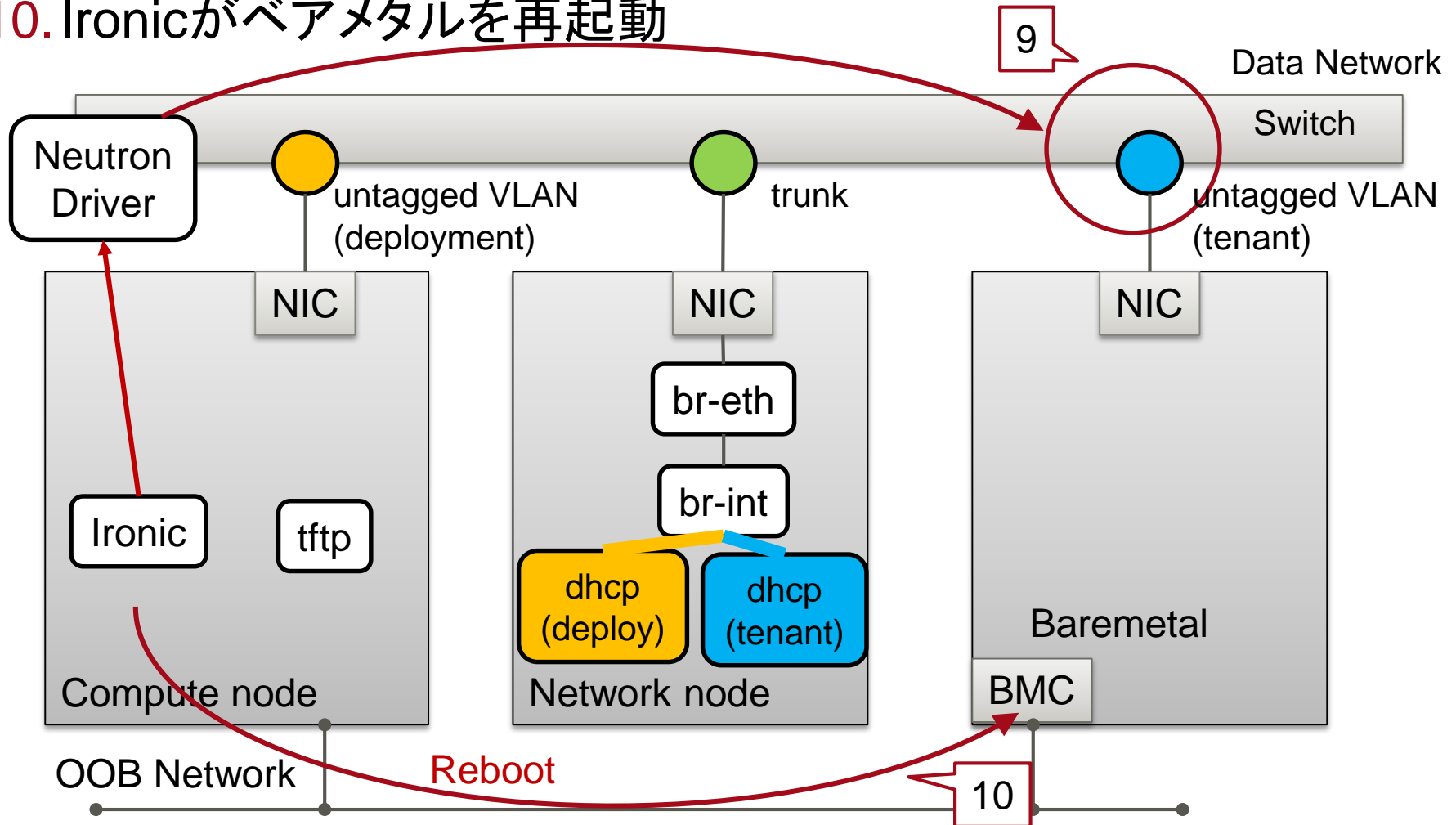
配備の詳細 (4/6)

- 7. ベアメタルが、デプロイ用イメージでブート後、iSCSIターゲットを用意し、Ironicに通知
- 8. 通知を受けたIronicがユーザ用OSイメージをベアメタルにコピー

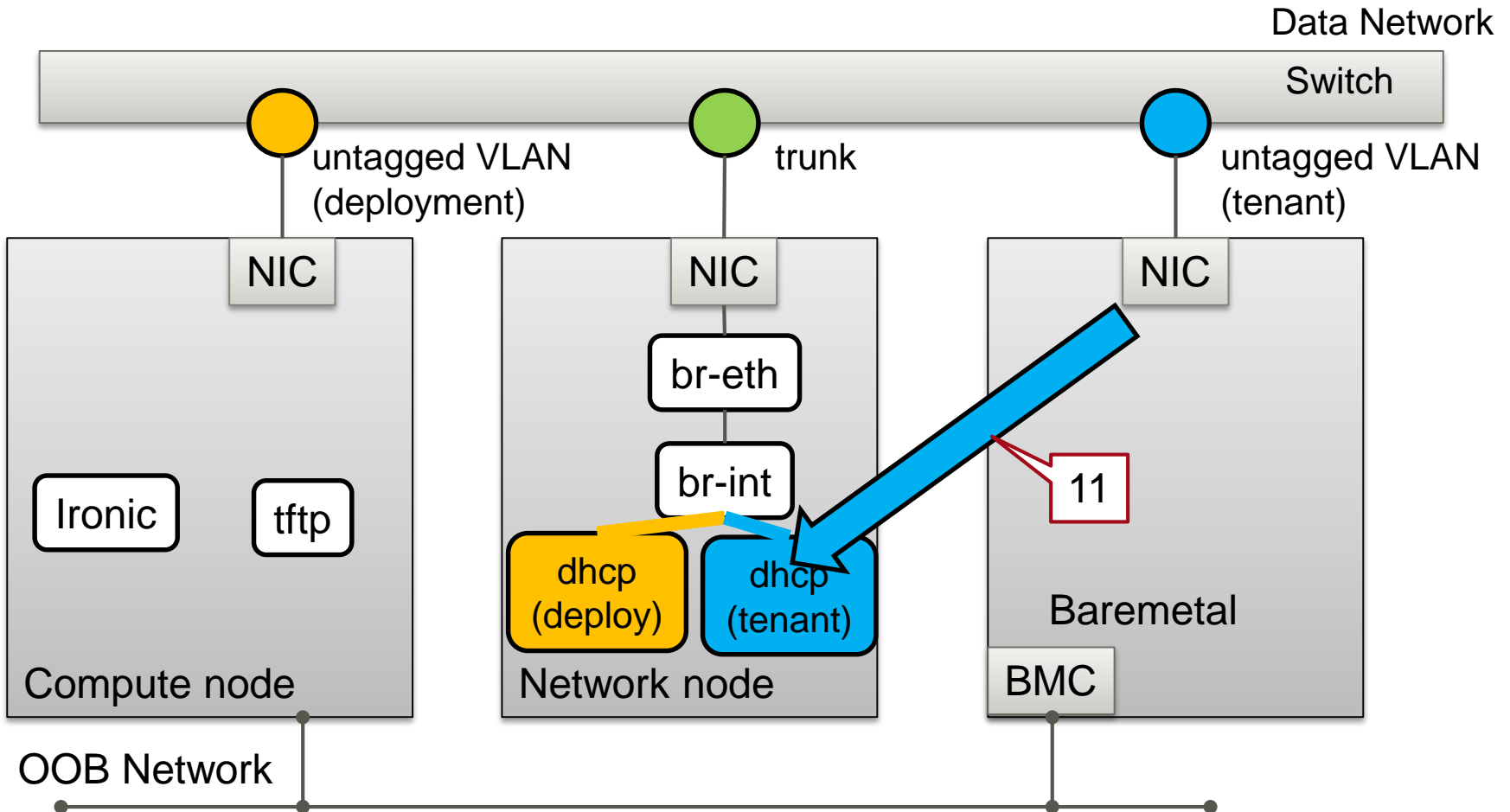


9. Ironicがテナント用VLANのIDを、ベアメタルが接続するスイッチポートに設定

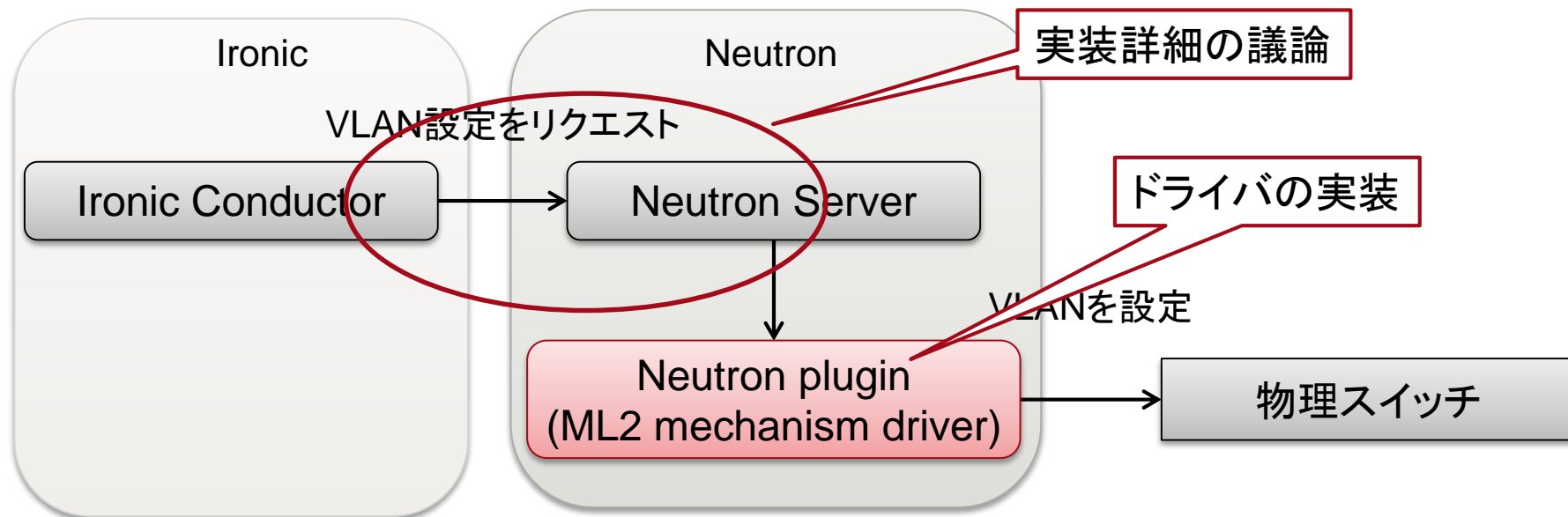
10. Ironicがベアメタルを再起動



- 11. 再起動後、ベアメタルが、テナント用VLANのDHCPサーバからIPアドレスを取得
- 12. テナント利用者がベアメタルを利用可能




- 専用IRCで実装の詳細を討議し、Libertyでの実現に貢献
 - Ironic-Neutron IRCミーティングで議論
毎週月曜16:00(UTC) / 25:00(JST)
- ドライバ実装のよい前例となることを目指して、ベンダ固有ドライバを開発中
 - NeutronのML2メカニズムドライバを開発中
 - neutronにRFE(Request For Enhancement)を発行
<https://bugs.launchpad.net/neutron/+bug/1464190>



- Ironicの機能強化に向けて
Design Summitでの課題の提案、議論に貢献

要件	可否	課題
仮想マシンと同一APIでの配備	○	
OSプロビジョニング	○	
SAN Boot構成	×	Cinderとの連携
サーバ故障時の復旧	×	
ボリューム割り当て	×	
マルチテナント対応	×	Neutronとの連携
仮想マシンとのネットワーク接続	×	



FUJITSU

shaping tomorrow with you