



Red Hat OpenStack Platform 13

ネットワーク機能仮想化 (NFV) の製品ガイド

ネットワーク機能仮想化 (NFV) の概要

Red Hat OpenStack Platform 13 ネットワーク機能仮想化 (NFV) の製品ガイド

ネットワーク機能仮想化 (NFV) の概要

OpenStack Team
rhos-docs@redhat.com

法律上の通知

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

本ガイドでは、ネットワーク機能仮想化 (NFV)、その利点、サポートされる設定、アーキテクチャー、コンポーネント、インストール、統合に関する情報を紹介します。

目次

前書き	3
第1章 RED HAT ネットワーク機能仮想化 (NFV) の理解	4
1.1. NFV の利点	4
1.2. NFV デプロイメントでサポートされている構成	4
第2章 ソフトウェア	6
2.1. ETSI NFV アーキテクチャー	6
2.2. NFV ETSI のアーキテクチャーおよびコンポーネント	6
2.3. RED HAT NFV のコンポーネント	7
2.4. インストールの概要	8
第3章 ハードウェア	9
第4章 NFV データプレーンの接続性	10
4.1. 高速データパスのオプション	10
第5章 NFV のパフォーマンスの考慮事項	12
5.1. CPU と NUMA ノード	12
5.2. NUMA ノードの例	12
5.3. CPU ピニング	13
5.4. HUGE PAGE	13
第6章 その他の参考資料	15

前書き

Red Hat OpenStack Platform は、Red Hat Enterprise Linux 上にプライベートまたはパブリックの Infrastructure-as-a-Service (IaaS) クラウドを構築するための基盤を提供します。これにより、スケーラビリティが極めて高く、耐障害性に優れたプラットフォームをクラウド対応のワークロード開発にご利用いただくことができます。

ネットワーク機能仮想化 (NFV: Network Functions Virtualization) では、仮想化を使用してネットワークノード機能をビルディングブロックに変換し、それらのビルディングブロックを相互に接続して通信サービスを作成することができます。NFV とは、専用のハードウェアアプライアンスをソフトウェアや自動化に置き換えることによって、ネットワークの定義、作成、管理を行う新しい方法です。

本ガイドでは、Red Hat OpenStack Platform を使用した NFV デプロイメントを加速するための Red Hat の取り組みについて簡単に説明します。

第1章 RED HAT ネットワーク機能仮想化 (NFV) の理解

ネットワーク機能仮想化 (NFV: Network Functions Virtualization) とは、通信事業者 (CSP) が従来のプロプライエタリーハードウェアの範囲を超えて、運用コストを抑えつつ、効率性および俊敏性を向上するのに役立つソフトウェアベースのソリューションのことです。

NFV は汎用のクラウドベースのインフラストラクチャーでネットワーク機能を仮想化して、コストの削減とイノベーションの拡大を図りつつ、レガシーのインフラストラクチャーよりも高い俊敏性、柔軟性、容易性、効率性、スケーラビリティを提供します。

NFV 環境では、スイッチ、ルーター、ストレージなどの標準のハードウェアデバイス上で実行する標準の仮想化技術を使用して仮想化インフラストラクチャーを提供することで、IT およびネットワークの収束が可能になり、ネットワーク機能を仮想化します。管理およびオーケストレーションロジックにより、これらのサービスをデプロイおよび維持します。NFV には、手動操作の必要性を軽減するシステム管理、自動化、ライフサイクル管理も含まれます。これにより、DevOps などの最新のツールが確実に使用されるので、作業時間が短縮され、規模が拡大されます。

1.1. NFV の利点

NFV 実装の主要な利点は以下のとおりです。

- ビジネスニーズの変化に対応するために特化されたハードウェアを新たにインストールする必要はないので、新たなネットワークサービスを迅速にデプロイすることが可能となり、市場投入までの時間が短縮されます。NFV により、通信事業者はサービスを試し、開発して、増加する顧客のニーズを満たすことができるため、新規サービスの展開に伴うリスクが軽減されます。
- NFV により、需要の変化に対応してサービスを迅速にスケーリングできるため、俊敏性と柔軟性が提供されます。また、サービスの開発者は実稼動環境で使用するのと同じ環境を使用してリソースやプロトタイプを自己管理できるので、イノベーションがサポートされます。
- セキュリティやパフォーマンスを犠牲にせず、週/日単位ではなく、時間/分単位で顧客のニーズに対応します。
- カスタマイズされた高額な設備ではなく、商用オフザシェルフ (COTS) のハードウェアを使用するため、資本支出が削減されます。
- 運用の合理化と自動化で日常のタスクを最適化することにより運用コストを削減し、従業員の生産性を向上させます。

1.2. NFV デプロイメントでサポートされている構成

Red Hat OpenStack Platform では、OVS-DPDK および SR-IOV 設定の自動化が追加され、NFV デプロイメントがサポートされています。さらに、ハイパーコンバージドインフラストラクチャー (HCI: Hyper-converged Infrastructure) ソリューションをお求めのお客様は、コンピュータのサブシステムと Red Hat Ceph Storage ノードを共存させることができます。NFV ユースケースにおいて、このハイパーコンバージドモデルを使用すると、初期コストの軽減、初期デプロイメントのフットプリントの縮小、稼働率の最大化、管理の効率化などが実現されます。また、一部制約はありますが、OVS ハードウェアオフロードなどのテクノロジープレビューサービスのデプロイメントもサポートしています。テクノロジープレビューとして提供されている機能のサポート範囲に関する詳しい情報は、[「テクノロジープレビュー機能のサポート範囲」](#)を参照してください。

コンポーザブルロールを使用して、カスタムのデプロイメントロールを作成できるようになりました。コンポーザブルロールを使用すると、各ロールにサービスを追加/削除することができます。

コンポーザブルロールに関する詳しい情報は「[Composable Roles and Services](#)」を参照してください。

第2章 ソフトウェア

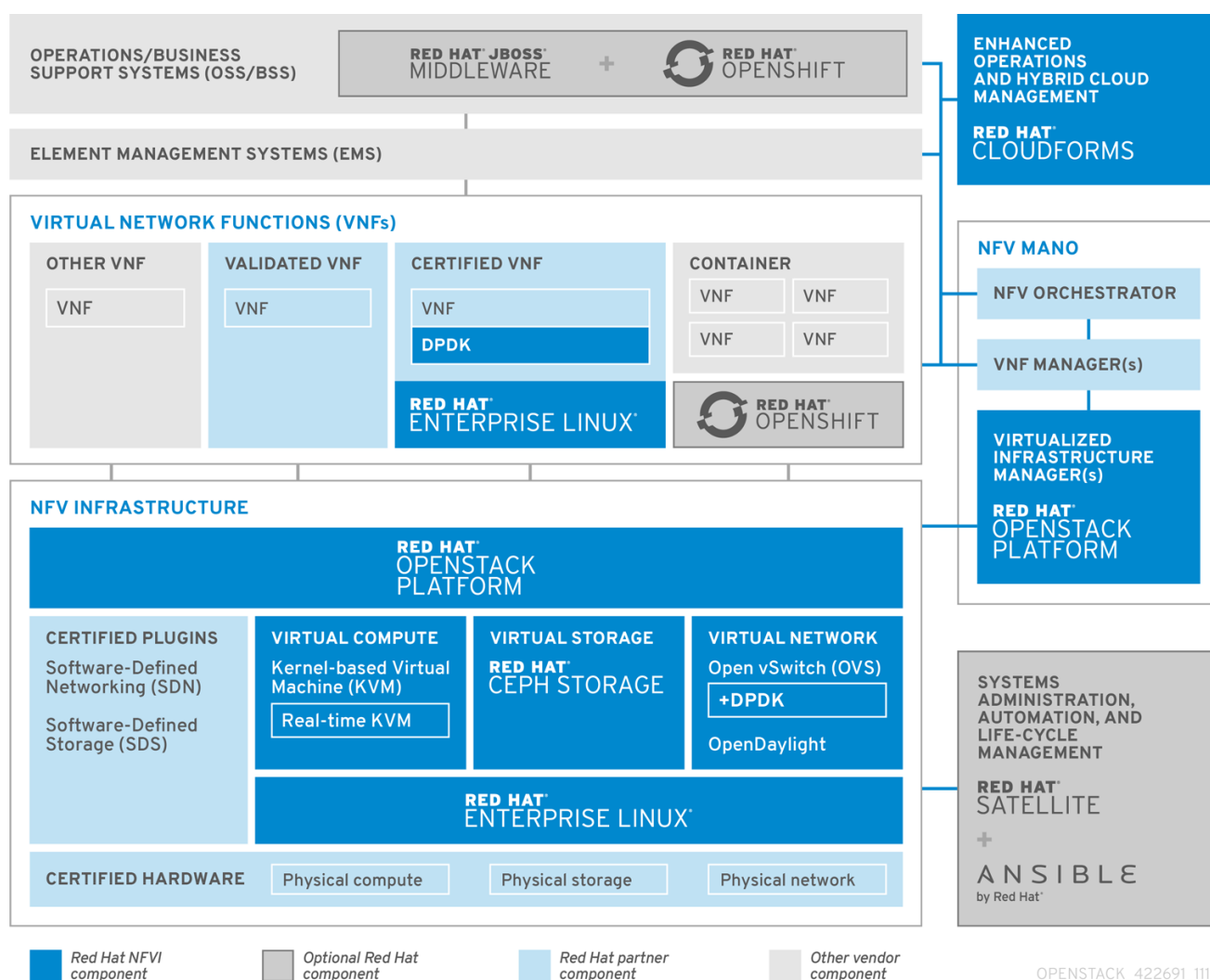
2.1. ETSI NFV アーキテクチャー

欧州電気通信標準化機構 (ETSI) は、ヨーロッパの情報通信技術 (ICT: Information and Communication Technology) の標準を開発する独立した標準化組織です。

ネットワーク機能の仮想化 (NFV) は、プロプライエタリーのハードウェアデバイスの使用に伴う問題への対処に重点を置いています。NFV を使用すると、ユースケースの要件や経済的な利点によっては、ネットワーク固有の設備をインストールする必要性が軽減されます。ETSI Industry Specification Group for Network Functions Virtualization (ETSI ISG NFV) は、VF を確実にサポートするために必要な要件、リファレンスアーキテクチャー、インフラストラクチャー仕様を設定します。

Red Hat では、通信事業者 (CSP) の IT とネットワークのコンバージェンス実現を支援するオープンソースベースのクラウド最適化ソリューションを提供しています。Red Hat OpenStack には、SR-IOV や DPDK-OVS などの NFV 機能を追加しています。

2.2. NFV ETSI のアーキテクチャーおよびコンポーネント



通常、NFV プラットフォームには以下のコンポーネントが含まれます。

- **Virtualized Network Functions (VNF):** ルーター、ファイアウォール、ロードバランサー、ブロードバンドのゲートウェイ、モバイルパケットのプロセッサ、サービス提供ノード、シグナリング、位置情報サービスなどのネットワーク機能のソフトウェア実装。

- **NFV Infrastructure (NFVi):** インフラストラクチャーを構成する物理リソース (コンピュータ、ストレージ、ネットワーク) および仮想化層。このネットワークには、仮想マシン間、ホスト全体でパケットを転送するためのデータパスが含まれます。これにより、基盤のハードウェアの情報を考慮せずに VNF をインストールできます。NFVi は、NFV スタックの基盤を形成します。NFVi は、マルチテナントをサポートし、Virtual Infrastructure Manager (VIM) で管理されます。Enhanced Platform Awareness (EPA) により、低レベルの CPU および NIC アクセラレーション機能が VNF に公開され、Red Hat Openstack Platform は仮想マシンのパケット転送のパフォーマンス (スループット、レイテンシー、ジッター) を向上することができます。
- **NFV Management and Orchestration (MANO):** VNF のライフサイクル全体で必要とされる全サービス管理タスクにフォーカスする管理およびオーケストレーション層。MANO の主要な目的は、オペレーターが顧客に提供するネットワーク機能のサービス定義、自動化、エラーの相関、監視、ライフサイクルを物理インフラストラクチャーから切り離せるようにすることです。このような切り離しを行うには、Virtual Network Function Manager (VNFM) が提供する管理層が追加で必要になります。VNFM は、直接対話するか、VNF ベンダーが提供する Element Management System (EMS) を使用して、仮想マシンのライフサイクルや VNF を管理します。MANO が定義するコンポーネントで他に重要なのは、NFVO として知られるオーケストレーターです。NFVO は、最上部のオペレーション/ビジネスサポートシステム (OSS/BSS) や、最下部の VNFM など、さまざまなデータベースやシステムにインターフェースを提供します。NFVO は、顧客向けの新規サービスを構築する場合には、VNFM に対して VNF のインスタンス化をトリガーするかを尋ねます (これにより、複数の仮想マシンが作成される場合があります)。
- **オペレーション/ビジネスサポートシステム (OSS/BSS: Operations/Business Support System):** オペレーションサポートや請求など必要不可欠なビジネス機能アプリケーションを提供します。OSS/BSS は、NFV に適応する必要がある、レガシーシステムと新規の MANO コンポーネントを統合しています。BSS システムは、サービスサブスクリプションをベースにポリシーを設定して、レポートと請求を管理します。
- **システム管理、自動化、ライフサイクル管理:** NFVi プラットフォームのインフラストラクチャーコンポーネントやライフサイクルのシステム管理、自動化を管理します。

2.3. RED HAT NFV のコンポーネント

Red Hat の NFV 向けのソリューションには、ETSI モデルに含まれる NFV フレームワークの異なるコンポーネントとしての役割を果たすことのできる各種製品が含まれます。NFV ソリューションには、Red Hat ポートフォリオの以下の製品が統合されます。

- Red Hat OpenStack Platform: IT および NFV ワークロードをサポートします。Enhanced Platform Awareness (EPA) 機能は、SR-IOV や OVS-DPDK をサポートする CPU ピニング、ヒュージページ、Non-Uniform Memory Access (NUMA) アフィニティー、ネットワークアダプター (NIC) などを使用して、決定的なパフォーマンスの向上を図ることができます。
- Red Hat Enterprise Linux および Red Hat Enterprise Linux Atomic Host: VNF として仮想マシンやコンテナを作成します。
- Red Hat Ceph Storage: サービスプロバイダーのワークロードに関するニーズすべてに対応する弾力性があり、統一された高性能なストレージ層を提供します。
- Red Hat JBoss Middleware および Red Hat 提供の OpenShift Enterprise: オプションで OSS/BSS コンポーネントの最新化に使用することができます。
- Red Hat CloudForms: VNF マネージャーを提供し、統合された画面に、VIM や NFVI などの複数のソースのデータを表示することができます。

- Red Hat Satellite および Red Hat 提供の Ansible: オプションとして、システムの管理、自動化、ライフサイクル管理を強化します。

2.4. インストールの概要

Red Hat OpenStack Platform director は完全な OpenStack 環境をインストールおよび管理します。director は、アップストリームの OpenStack TripleO プロジェクトをベースとしています。TripleO とは、「OpenStack-On-OpenStack」の頭文字の「O」が3つあることを意味しています。このプロジェクトは、OpenStack コンポーネントを活用して、完全に機能する OpenStack 環境をインストールします。これには、アンダークラウドと呼ばれる、最小限の OpenStack ノードが含まれます。アンダークラウドは、オーバークラウド (実稼働環境用の OpenStack ノードとして使用される一連のベアメタルシステム) のプロビジョニングと制御を行います。director は、リーンかつ堅牢な Red Hat OpenStack Platform の完全な環境を簡単にインストールできる方法を提供します。

アンダークラウドおよびオーバークラウドのインストールに関する詳しい情報は『[Red Hat OpenStack Platform director のインストールと使用方法](#)』を参照してください。

NFV 機能をインストールするには、以下の設定手順を追加で行う必要があります。

- SR-IOV および PCI パススルーのパラメーターを **network-environment.yaml** ファイルに追加して、CPU をチューニングするために **post-install.yaml** ファイルを更新し、**compute.yaml** ファイルを変更してから **overcloud_deploy.sh** スクリプトを実行し、オーバークラウドをデプロイします。
- NIC から直接データをポーリングして、高速なパケット処理を行えるように DPDK ライブラリーおよびドライバーをインストールします。**network-environment.yaml** ファイルに DPDK パラメーターを追加し、CPU チューニング向けに **post-install.yaml** ファイルを更新します。また、**compute.yaml** ファイルを更新して DPDK ポートでブリッジの設定を行い、**controller.yaml** ファイルを更新して VLAN 構成のブリッジとインターフェースを設定し、**overcloud_deploy.sh** スクリプトを実行してオーバークラウドをデプロイします。

NFV のプランニングガイドラインと設定については、『[Network Function Virtualization Planning and Configuration Guide](#)』を参照してください。

第3章 ハードウェア

OpenStack ノードのハードウェア選択に関する考慮事項は、[『director のインストールと使用方法』](#)を参照してください。

NFV 向けのテスト済み NIC の一覧は、[「Network Adapter Support」](#)を参照してください (カスタマーポータルログインが必要です)。

第4章 NFV データプレーンの接続性

NFV が導入され、従来のデバイスを VNF として実装を始めているネットワークベンダーが増えています。これらのネットワークベンダーの大半が仮想マシンに着目していますが、選択した設計に合わせたコンテナベースのアプローチに注目しているベンダーもあります。OpenStack ベースのソリューションは、主に以下の2つの理由からリッチかつ柔軟性でなければなりません。

- **アプリケーションの即応性:** ネットワークベンダーは現在、デバイスを VNF に変換している段階であるため、市場にある VNF ごとに異なる成熟度レベルとなっており、即応性に関する共通の障害として、API での RESTful インターフェースの有効化、データモデルのステートレスへの進化、自動化管理オペレーションの提供などが挙げられます。OpenStack は、すべてに共通のプラットフォームを提供します。
- **幅広いユースケース:** NFV には、異なるユースケースに対応する多様なアプリケーションが含まれます。たとえば、Virtual Customer Premise Equipment (vCPE) は、ルーティング、ファイアウォール、VPN、NAT など多くのネットワーク機能を顧客のサイトで提供することを目的としています。また、Virtual Evolved Packet Core (vEPC) は、LTE (Long-Term Evolution) ネットワークのコアコンポーネントにコスト効果の高いプラットフォームを提供するクラウドアーキテクチャーで、ゲートウェイやモバイルエンドポイントをダイナミックにプロビジョニングして、スマートフォンやその他のデバイスからの増加するデータトラフィック量に対応できるようにします。
これらのユースケースは本質的に、異なるネットワークアプリケーション、プロトコルを使用して実装され、インフラストラクチャーとは異なる接続性、分離、パフォーマンスの特性が必要となります。また、コントロールプレーンのインターフェースとプロトコル、実際の転送プレーンを分離することが一般的です。OpenStack には、さまざまなデータパスの接続性オプションを十分に提供できるように柔軟性が重要です。

基本的に、仮想マシンにデータプレーンの接続性を提供する一般的なアプローチは2種類あります。

- **ハードウェアへのダイレクトアクセス:** PCI パススルー (OpenStack では SR-IOV PF と表記) や VF (Virtual Function) や PF (Physical Function) の両パススルー向けの Single Root I/O Virtualization (SR-IOV) などの技術を使用して、Linux カーネルを迂回して、物理ネットワークインターフェースカード (NIC) にセキュアな Direct Memory Access (DMA) を提供します。
- **仮想スイッチ (vswitch) の使用:** ハイパーバイザーのソフトウェアサービスとしての実装。仮想マシンは、仮想インターフェース (vNIC) を使用して vSwitch に接続され、仮想マシン間や仮想マシンと物理ネットワーク間のトラフィックを転送することができます。

4.1. 高速データパスのオプション

高速データパスには、以下のようなオプションがあります。

- **Single Root I/O Virtualization (SR-IOV):** 単一の PCI ハードウェアデバイスを複数の仮想 PCI デバイスのように見せる標準のことで、PF (Physical Function) を導入して機能します。PF とは、物理ハードウェアポートと VF (Virtual Function: 仮想マシンに割り当て可能な軽量機能) の機能を果たすフル装備の PCIe 機能です。仮想マシンは、VF をハードウェアと直接通信する通常の NIC とみなします。NIC は複数の VF をサポートします。
- **Open vSwitch (OVS):** 仮想化サーバー環境内で仮想スイッチとして使用されるように設計されたオープンソースのソフトウェアスイッチです。OVS は、通常の L2-L3 スwitch のケーパビリティだけでなく、ユーザー定義のオーバーレイネットワーク (例: VXLAN) を作成する OpenFlow などの SDN プロトコルもサポートします。OVS は、物理 NIC を使用して仮想マシン間およびホスト全体のパケットを切り替えるのに、Linux のカーネルネットワークを使用します。OVS は、iptables/ebtables で Linux ブリッジのオーバーヘッドを回避する接続トラッキング

グ (Conntrack) と内蔵のファイアウォール機能をサポートします。Red Hat OpenStack Platform 環境の Open vSwitch は、カスタマイズなしに OVS と OpenStack Networking (neutron) を統合できます。

- **Data Plane Development Kit (DPDK):** 高速なパケット処理に向けてライブラリーセットや Poll Mode Driver (PMD) で構成されています。DPDK はユーザー空間で大半を実行するように設計されており、アプリケーションが NIC から直接独自のパケット処理を実行できるようになります。DPDK は、レイテンシーを減らし、処理するパケット数を増やすことができます。DPDK Poll Mode Drivers (PMDs) はビジーループで実行され、ホストの NIC ポートやゲストの vNIC ポートにパケットが到着していないかを絶えずスキャンします。
- **DPDK accelerated Open vSwitch (OVS-DPDK):** Linux カーネルバイパスと物理 NIC への Direct Memory Access (DMA) を用いた高性能のユーザー空間ソリューションに向け DPDK がバンドルされた Open vSwitch。標準の OVS カーネルデータパスを DPDK ベースのデータパスに置き換えて、内部で DPDK をパケット転送に使用するユーザー空間の vSwitch をホストに構築するという概念です。このアーキテクチャーの利点は、OVS の基本機能や公開されるインターフェース (OpenFlow、OVSDb、コマンドライン) はほぼ変わらないので、大半がユーザーに透過的となっている点です。

第5章 NFV のパフォーマンスの考慮事項

NFV ソリューションが有用であるためには、VF が物理実装のパフォーマンス以上である必要があります。Red Hat の仮想化技術は、OpenStack およびクラウドのデプロイメントで一般的に使用されている高パフォーマンスの Kernel-based Virtual Machine (KVM) ハイパーバイザーをベースにしています。

5.1. CPU と NUMA ノード

以前は、x86 システムの全メモリーは、システム内のどの CPU からでも同等にアクセスできていたために、システム内で操作を行う CPU、Uniform Memory Access (UMA) を参照する CPU がどれでも、メモリーのアクセス時間は同じでした。

Non-Uniform Memory Access (NUMA) では、システムメモリーは、特定の CPU またはソケットに割り当てられるノードと呼ばれるゾーンに分割されます。CPU のローカルにあるメモリーには、そのシステムのリモートの CPU に接続されているメモリーにアクセスするよりも高速です。通常、NUMA システム上のソケットにはそれぞれローカルのメモリーノードがあり、別の CPU のローカルにあるノードのメモリーや、全 CPU で共有されるバス上のメモリーよりもコンテンツに早くアクセスできます。

同様に、物理 NIC はコンピュータノードのハードウェア上の PCI スロットに配置されます。これらのスロットは、特定の NUMA ノードに関連付けられた特定の CPU ソケットに接続されます。パフォーマンスを最適化するには、CPU の設定で指定されているのと同じ NUMA ノードにデータパス NIC を接続します。

NUMA を使用しない場合のパフォーマンスへの影響は大きく、一般的に、パフォーマンス 10 % 以上が影響を受けます。各 CPU ソケットには、仮想化を目的とする個別の CPU として扱われる複数の CPU コアを配置することができます。

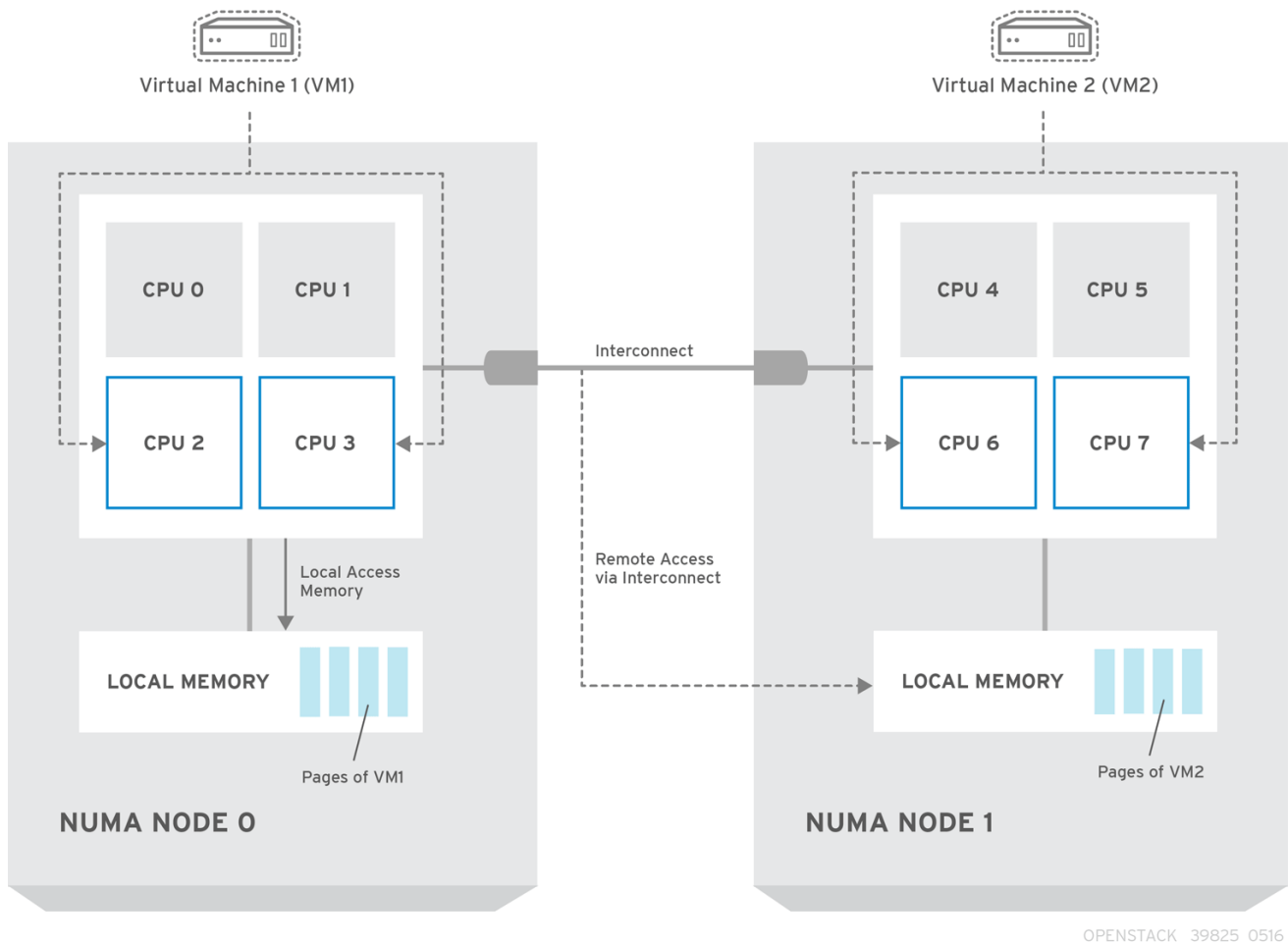
OpenStack Compute は、インスタンスの起動時にスマートなスケジューリングや配置の意思決定を行います。管理者はこのような機能を活用すると、NFV や高性能コンピューティング (HPC) などの特化されたワークロードを対象にするカスタマイズされたパフォーマンスフレーバーを構築することができます。

ヒント

NUMA についての予備知識は、[「What is NUMA and how does it work on Linux?」](#) の記事に記載されています。

5.2. NUMA ノードの例

以下の図では、2 つのノードからなる NUMA システムの例と、CPU コアとメモリーページを利用可能にする方法の例が提供されています。



注記

Interconnect 経由で利用可能なリモートのメモリには、NUMA ノード 0 からの VM 1 に NUMA ノード 1 の CPU コアがある場合 **のみ** アクセスされます。このような場合には、NUMA ノード 1 のメモリは、VM 1 の 3 番目の CPU コアのローカルとして機能しますが (例: 上記の図では、VM1 は CPU4 が割り当てられています)、同じ VM の別の CPU コアに対してはリモートメモリとして機能します。

5.3. CPU ピニング

CPU ピニングは、指定のホスト内にある特定の物理 CPU 上で特定の仮想マシンの仮想 CPU を実行する機能のことです。vCPU ピニングでは、ベアメタルシステムへのピンングタスクと同様の利点が得られます。仮想マシンは、ホストのオペレーティングシステムのユーザー空間タスクとして実行されるので、ピンングすることでキャッシュの効率性が向上されます。

詳しい情報は、[「Configure CPU Pinning with NUMA」](#) を参照してください。

5.4. HUGE PAGE

物理メモリは、ページと呼ばれる連続した一連のリージョンに分割されます。効率化を図るため、システムは、各メモリーバイトにアクセスするのではなく、ページ全体にアクセスしてメモリーを取得します。このような変換を実行するには、システムは、最新のページまたは頻繁に使用されるページの物理から仮想アドレスのマッピングが含まれるトランスレーションルックアサイドバッファ (TLB: Translation Lookaside Buffers) をチェックします。検索したマッピングが TLB にはない場合には、プロ

セッサーは全ページテーブルで同じ処理を反復して、アドレスマッピングを判断する必要があるため、パフォーマンスペナルティーの原因となります。そのため、対象のプロセスがなるべく TLB ミスを回避できるように TLB を最適化することが推奨されます。

x86 システムの一般的なページサイズは 4KB で、他にこれよりもページサイズが大きいものもあります。ページサイズが大きいと、全体的なページ数が少なくなるので、TLB に仮想から物理アドレスの変換を保存可能なシステムメモリー量が増えることになります。その結果、TLB ミスの可能性が低くなり、パフォーマンスが向上します。ページサイズが大きいと、プロセスはページに割り当てる必要があるため、メモリーを無駄にする可能性が高くなりますが、すべてのメモリーが必要となることはあまりありません。そのため、ページサイズを選択する際には、より大きいページを使用してアクセス時間を早くするか、より小さいページを使用して最大限にメモリーが使用されるようにするかで、トレードオフが生じます。

第6章 その他の参考資料

以下の表には、その他の参考となる Red Hat のドキュメントの一覧を記載しています。

Red Hat OpenStack Platform のドキュメントスイートは [Red Hat OpenStack Platform の製品ドキュメントスイート](#) から参照してください。

表6.1 参考資料一覧

コンポーネント	参考情報
Red Hat Enterprise Linux	Red Hat OpenStack Platform 10 は Red Hat Enterprise Linux 7.4 でサポートされています。Red Hat Enterprise Linux のインストールに関する情報は、 Red Hat Enterprise Linux のドキュメント から対応するインストールガイドを参照してください。
Red Hat OpenStack Platform	<p>OpenStack のコンポーネントとそれらの依存関係をインストールするには、Red Hat OpenStack Platform director を使用します。director は、アンダークラウド を使用して、基本的な OpenStack 環境のインストールと設定を行って、最終的には オーバークラウド で OpenStack ノードを管理します。アンダークラウドのインストールには、デプロイしたオーバークラウドに必要な環境に加えて、アンダークラウドのインストールには、追加のホストマシンが必要となる点に注意してください。詳しい手順は、『Red Hat OpenStack Platform director のインストールと使用方法』 を参照してください。</p> <p>Red Hat OpenStack Platform director を使用して、ネットワークの分離、ストレージ設定、SSL 通信、一般的な設定方法など Red Hat OpenStack Platform のエンタープライズ環境の高度な機能設定に関する情報は 『Advanced Overcloud Customization』 を参照してください。</p>
NFV のドキュメント	SR-IOV および OVS-DPDK を実装する Red Hat OpenStack Platform のデプロイメントの計画および設定に関する詳細は、 『Network Function Virtualization Planning and Configuration Guide』 を参照してください。