



日本アイ・ビー・エム株式会社
グローバル・テクノロジー・サービス事業本部
ITS デリバリー
技術理事
ディステイニングイッシュド・エンジニア

山下 克司



日本アイ・ビー・エム株式会社
GTS 事業、ソリューション
第一サービスライン SOL
シニア IT スペシャリスト
IBM クラウドマイスター
認定クラウド上席システムズ・エンジニア

田中 良典

実践 OpenStack を最大限活用する設計のツボ

クラウド環境を構築するためのオープンソース・ソフトウェアのスタンダードとして定着している OpenStack。その機能は年々進化を遂げ、コンポーネントの数も充実している。しかし、この OpenStack を使って最適化されたクラウド環境を設計するためには、一定の知識が必要になる。本資料では、OpenStack を活用してクラウド環境を構築する際のポイントを3つの「ツボ」としてまとめ、最適化されたクラウド環境実現のためのノウハウについて解説する。

ツボ その1 要件に応じて必要なコンポーネントと既存の技術を柔軟に組み合わせる

OpenStack はコンピュー、ネットワーク、ストレージの3つのリソースを抽象化して管理するものであり、それぞれがコンポーネントに分かれている。コア・サービスとしては、nova、cinder、neutron、keystone、glance、swift の6つのコンポーネントがあり、それ以外にもオプションのサービスを含めると50種類以上のコンポーネントが用意されている。これらのコンポーネントの連携は疎結合となっているので、柔軟に組み合わせ活用することができる。そこでどのコンポーネントを選んでどのように組み合わせればいいのかという判断をしなければならぬが、その際大切なポイントは、OpenStack のコンポーネントをできるだけシンプルに使うことと、実現したい機能を必ずしも OpenStack だけで構成する必要はないということである。これが1つ目のツボで、使い慣れていない OpenStack のコンポーネントを無理に使うのではなく、コア・サービスを中心に各機能を活用しつつ、要件に応じて既存の技術を組み合わせることで、効率的にクラウド環境を構築することが可能になるのだ。以下、3つの具体例を取り上げながらそのポイントについて解説する。

● OpenStack によるマルチリージョン／マルチハイパーバイザー管理

1つ目はマルチリージョン／マルチハイパーバイザー環境における OpenStack の活用例 (図1) である。この例では、VMware の環境と UNIX の環境を個別に運用していたが、ここに OpenStack を導入することでハイパーバイザーを抽象化し、統合的に管理可能な環境を構築した。OpenStack のコンポーネントとしては、nova、glance、cinder、keystone といったコア・サービスのものを活用し、マルチリージョン／マルチハイパーバイザー環境の統合を実現している。

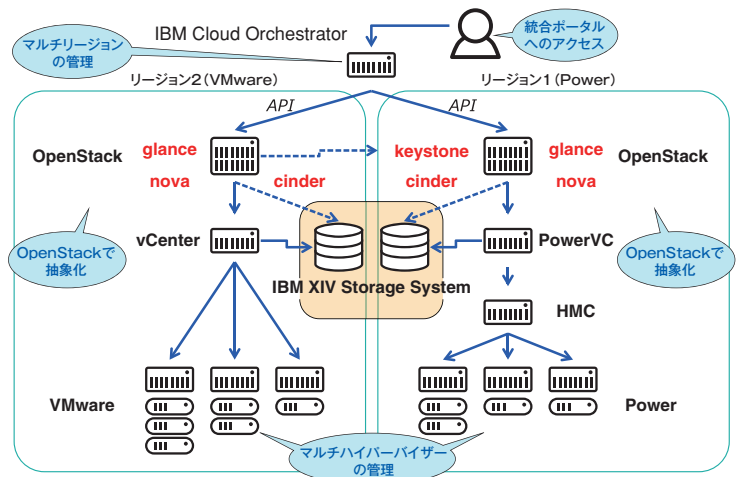
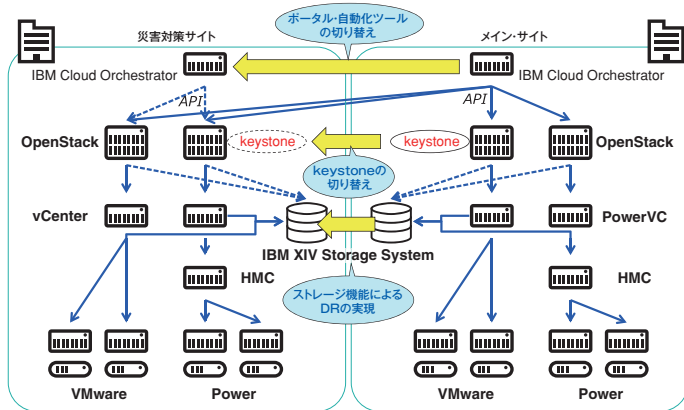


図1. OpenStack によるマルチリージョン／マルチハイパーバイザー管理



● OpenStackによるマルチリージョン/
マルチハイパーバイザー管理 (災害対策サイト構成)

この例は先に紹介したマルチリージョン/マルチハイパーバイザー環境と似ているが、災害対策サイトの構成を目的としたものである。ここでもOpenStackのコンポーネントは、コア・サービスのものを活用しているが、災害時の回復 (Disaster Recovery: 以下、DR) の仕組みをOpenStackのコンポーネントで組み込むことは非常に難易度が高い。そこでDRについてはリモート・コピーなどのストレージの機能を活用することで効率的に災害対策サイトを構成することが可能になる。このようにOpenStackのコンポーネントを使わなければならないという固定概念を捨てて、既存の技術を柔軟に組み合わせることが非常に有効だといえる。

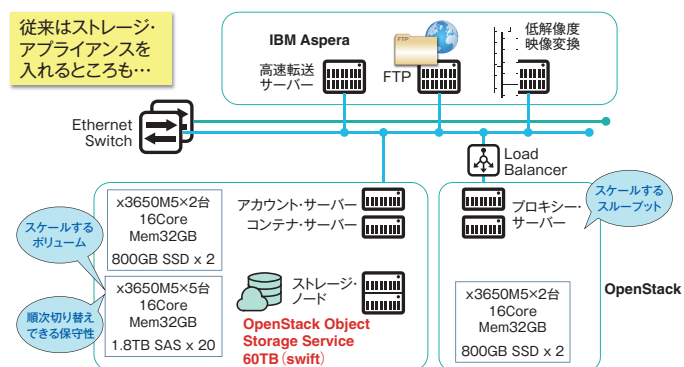


keystone: Identity

図2. OpenStackによるマルチリージョン/マルチハイパーバイザー管理 (災害対策サイト構成)

● OpenStack Object Storage (swift) の活用~ 4K映像
クラウド・サービスにおけるスケールする分散ストレージ・ノード

3つ目は映像の転送・変換システムの例で、そこで映像を保管するストレージの構成方法について解説する (図3)。従来は大容量のストレージを導入することで解決していたが、映像データは将来的にどの程度のスケールになるのか予想することが難しい。そこでswiftによる分散型のオブジェクト・ストレージを活用すれば、柔軟にボリュームを拡充できる環境を構築することが可能だ。ソフトウェアで分散環境を管理しているので、一部のマシンが老朽化した場合でも容易に切り替えることができ、保守性の向上が実現する。この例のようにOpenStackからswiftだけを切り出して活用するという方法も非常に有効である。



swift: Object Storage

図3. OpenStack Object Storage (swift) の活用

ツボ その2
自動化適用範囲を見定め、オープンソースの
自動化ツールを活用することで運用を劇的に改善

2つ目のツボは、自動化の適用範囲をいかに見定めるかということだ。システムの企画、構築から運用に至るライフサイクル全体を見通して、トータルで自動化を図ることが重要である。その自動化を実現するのがOpenStackだ。OpenStackといえばインターネット・ビジネスを展開する企業向けという印象があったが、近年は一般企業内のシステムにOpenStackを適用し、構築・運用の効率化を図るといった事例も増えている。

通常OpenStackを導入するとエンドユーザー向けのセルフサービス・ポータルをフロントに設置し、そこから自動化ツールを呼び出すことになる。さらに、ITILを使ってプロセス管理をしている場合は、変更管理、リリース管理、構成管理などのプロセスを経て、承認されれば構築の時点でOpenStackに命令が下される。

次にOpenStackを使ってオーケストレーション処理とテストを自動化する例を紹介する。OpenStackはコンピュータ (仮想マシン)、ストレージ、ネットワークのプロビジョニングを主たる目的としているが、要件によってはミドルウェアの自動化や複数サーバー展開、サーバー間の接続の自動化、Webアプリケーション・システムのロード・バランサーや監視サーバーの設定の自動化なども求められる。そこで活用するのがオーケストレーション・サービスを提供するコンポーネントのheatおよびChef、Ansibleといったオープンソースの構成管理ツールである。これらを組み合わせることで、OSの設定、ミドルウェアの設定、外部との連携などを実現する。

ここまでは構築の話だが、これをリリースするためにはテストが必要になる。このテストの過程を手動で行っていたら自動化の効果が半減してしまう。そこでオープンソースのServerspecなどを活用してサーバー構成テストを自動化することが有効になる。例えばChefで構成した設定をServerspecで自動的にテスト・検証し、その結果をユーザーに通知するという流れで構築、設定、テストまでを自動化することができる。

さらに自動化ツールを活用することでDevOpsを実現することも可能になる (図4)。アジャイルを導入するとアプリケーションをスピーディーにリリースできるようになるが、インフラがそれに追い付ければ全体として効率的なサイクルを回すことができない。そこでOpenStackを使ってAPIを導入し、Vagrant、Chef、Serverspecといったツールを活用してスピードアップを図ることになる。そして究極的にはインフラもコードで管理するInfrastructure as Codeを実現し、アプリケーションと同様にコードで管理しながらCI (継続的インテグレーション) ツールで運用していくことになり、最終的にはContinuous Delivery (継続的デリバリー) が視野に入ってくる。このように自動化の適用範囲を見定めた上で、オープンソース系の構成管理・自動化ツールをフルに活用することで運用を劇的に改善することが可能になる。実際に、基幹業務をOpenStackベースでハイブリッド・クラウド化することで構築と運用を効率化し、構築期間を2カ月から10日間に短縮したという事例もある。

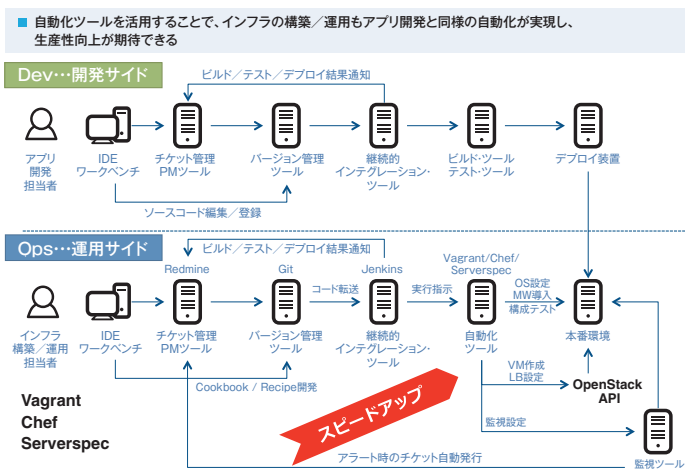


図4. DevOpsを実現するオープンソース自動化ツールの利用

ツボ その3 データセンターを物理的なネットワークから解放して柔軟性を重視する

続いてのツボはネットワークに関するポイントである。コンピュータやストレージのエリアが自動化されても、ネットワーク部分がボトルネックとなっていては、OpenStackの利点を最大限に引き出すことは難しい。そこで物理的なネットワークに頼らず、クラウド内のネットワークをソフトウェアで制御することで柔軟性を高めることが重要になる。クラウドの外部拠点からはインターネットやWANなどで接続してOpenStackのコントロール・パネルにアクセスし、クラウド内ではネットワークのコンポーネントであるneutronを活用してネットワークを構成するのだ。neutronにVLANの構築、アクセス・コントロールの実行、ロード・バランサーの構成などを命令することでOpenStack内のネットワークを構成することができる。従来はスイッチにコンソールをつなげるなどの作業が必要だったところが、すべてコマンドで指示するだけで済むようになるのだ。

OpenStackの基本的なモデルはインターネット上のWebサイトに例えることができる。つまり、OpenStack内では外部とは関係ないIPアドレスで管理されているが、ユーザーはその内部構造を把握せずともOpenStackを活用することが可能だ。OpenStackはインターネットのクラウド環境で使われてきたテクノロジーなので、Webサーバーや分散ストレージに適した構造になっている。

こうした特徴を有するOpenStackをエンタープライズで活用する場合、エンタープライズが使っているネットワーク・アドレスの領域とOpenStackの中で使っているネットワーク・アドレスの領域を同一のネットワーク・アドレスの管理体内に収めることを求められるケースが多い。その課題はOpenStack環境内にNFV(Network Function Virtualization)を持ち込むことで解決できる。NFVによりソフトウェア化されたルーターやゲートウェイを構築することで、OpenStack内とそれ以外のエンタープライズ領域の間の橋渡しの役割を持たせることが可能になる。

OpenStackではネットワークをソフトウェアで実現するという基本的なポリシーがあるため、以下を実現することができる。

- ・ SDN (Software-Defined Network) による集中管理されたソフトウェア制御のデータセンター・ネットワーク
- ・ サーバーやストレージの仮想化のスピードに対応
- ・ NFVによるネットワーク機能の仮想化がネットワークの柔軟性を向上させる

IBM SoftLayerのデータセンターでは、より安全かつ容易にネットワークを構成できるように、お客様環境と専用線で直接接続できるサービスを提供している。

OpenStackとVMware、どちらを選択するのか

OpenStackの基本的なアーキテクチャーは、疎結合されたコンポーネントがそれぞれAPIから呼び出されるという形になっている。従って柔軟性に富んでいるのではあるが、まだ不慣れなユーザーも多いと思われる。一方でVMwareはデータセンターの仮想化を目的として開発されたもので、クラウドを実現するものではない。とはいえVMwareは長い実績があることから、使い慣れているユーザーは数多く、機能も豊富に用意されている。しかし、それぞれの機能は密結合となっているため柔軟性に劣るといえるだろう。

ハイパーバイザーとして両者を比較する場合、OpenStackのコンポーネントであるKVMに着目する必要がある。KVMのコンソールは不慣れな場合が多く、さまざまなスキルが求められる。しかし、OpenStack全体を組み上げ、horizon portalを構築すれば、KVMのコンソールを使う必要はなくなる。そして、このレベルになるとさまざまなコンポーネントやオープンソースのツールを使うことができるようになるので、機能面でも充実してくる。要件に応じてOpenStackが求められる場合とVMwareが求められる場合があるが、データセンターの仮想化のレベルから脱却し、柔軟にスケールするクラウドのインフラストラクチャーに変革し、スピードと品質を向上させるためには、OpenStackの活用が鍵となるであろう。

IBM製品を活用したOpenStack環境の管理

IBMではOpenStack環境を管理するための製品を数々そろえている。クラウドの管理ツールとしては、インフラ視点でクラウドを管理するIBM Cloud Orchestratorとアプリケーションのデプロイやリリースを管理するIBM UrbanCode Deployを用意している。IBM Cloud Orchestratorはコミュニティ版のOpenStackのほかに、各社が提供しているIBM商用ディストリビューション、IBMのOpenStackのマネージド・サービスであるBlue Boxなどを管理することが可能だ。

Blue BoxはOpenStackの環境をマネージド・サービスとして提供するものだ。OpenStackはオープンソースであることから、パッケージ製品と比較した場合、取り扱いが難しい面がある。そこでOpenStackの難しい設定などをIBMが引き受けることで、お客様が自動化されたサービスをすぐに活用できることを実現した。IBMは今後もOpenStackのコミュニティが提供する機能などを常に取り入れ、お客様のニーズに即したサービスや製品を提供していく。

IBMクラウドに関する詳細情報は下記のWebサイトをご覧ください。
<http://www.ibm.com/cloud-computing/jp/ja/>



日本アイ・ビー・エム株式会社

〒103-8510 東京都中央区日本橋箱崎町 19-21

©Copyright IBM Japan, Ltd. 2016
All Rights Reserved

Printed in Japan
March 2016

本資料の情報は2016年3月現在のものです。仕様は予告なく変更される場合があります。本資料中に記載の肩書や数値、固有名詞等は初掲載当時のものであり、閲覧される時点では、変更されている可能性があることをご了承ください。
また、記載の事例は特定のお客様に関するものであり、すべての場合において同等の効果が得られることを意味するものではありません。効果はお客様の環境その他の要因によって異なります。製品、サービスなどの詳細については、弊社の営業担当員にご相談ください。

IBM、IBMロゴ、ibm.comは、世界の多くの国で登録されたInternational Business Machines Corp.の商標です。他の製品名およびサービス名等は、それぞれIBMまたは各社の商標である場合があります。現時点でのIBM商標リストについては www.ibm.com/legal/copytrade.shtml をご覧ください。Microsoft、Windows、Windows XPは、Microsoft Corporationの米国およびその他の国における商標です。
