

OpenStackで始める クラウド環境構築入門

日本仮想化技術株式会社
VirtualTech.jp

VirtualTech Japan

VirtualTech Japan

日本仮想化技術株式会社 概要

- 社名: 日本仮想化技術株式会社
 - 英語名: VirtualTech Japan Inc.
 - 略称: 日本仮想化技術/VTJ
- 設立: 2006年12月
- 資本金: 14,250,000円
- 本社: 東京都渋谷区渋谷1-1-10
- 取締役: 宮原 徹(代表取締役社長兼CEO)
- 伊藤 宏通(取締役CTO)
- スタッフ: 8名(うち、5.5名が仮想化技術専門エンジニアです)
- URL: <http://VirtualTech.jp/>
- 仮想化技術に関する研究および開発
 - 仮想化技術に関する各種調査
 - 仮想化技術に関連したソフトウェアの開発
 - 仮想化技術を導入したシステムの構築

ベンダーニュートラルな
独立系仮想化技術
専業会社



仮想化環境構築をトータルサポート

戦略立案



設計



導入・移行



運用保守

- 戦略立案
 - コスト削減、社内標準化、将来プランのコンサルティング
- 設計
 - 要求仕様の策定
 - サーバ、ストレージからネットワークまでアプリケーションまで考慮した設計最適化
 - キャパシティプランニング(ベンチマーク)
- 導入
 - 仮想化ソリューションパッケージの提供
 - 仮想化統合 (P2V 既存環境移行)
- 運用保守
 - エンジニア教育
 - 技術サポートの提供
 - OSSソースコードレベルサポート

ベンダーニュートラルなワンストップ・サポートをご提供

3



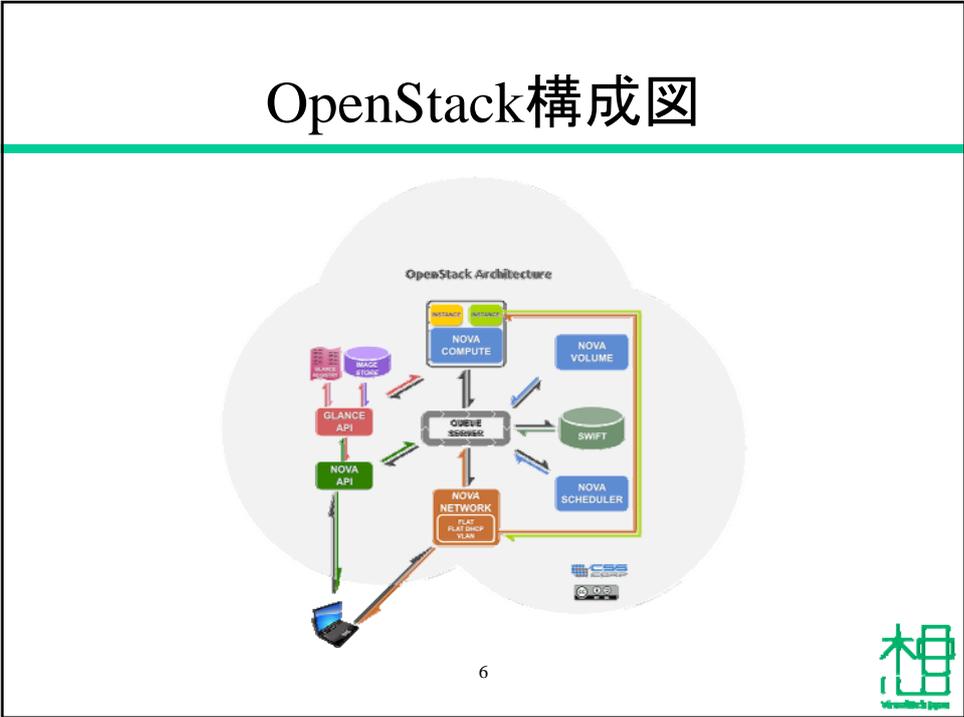
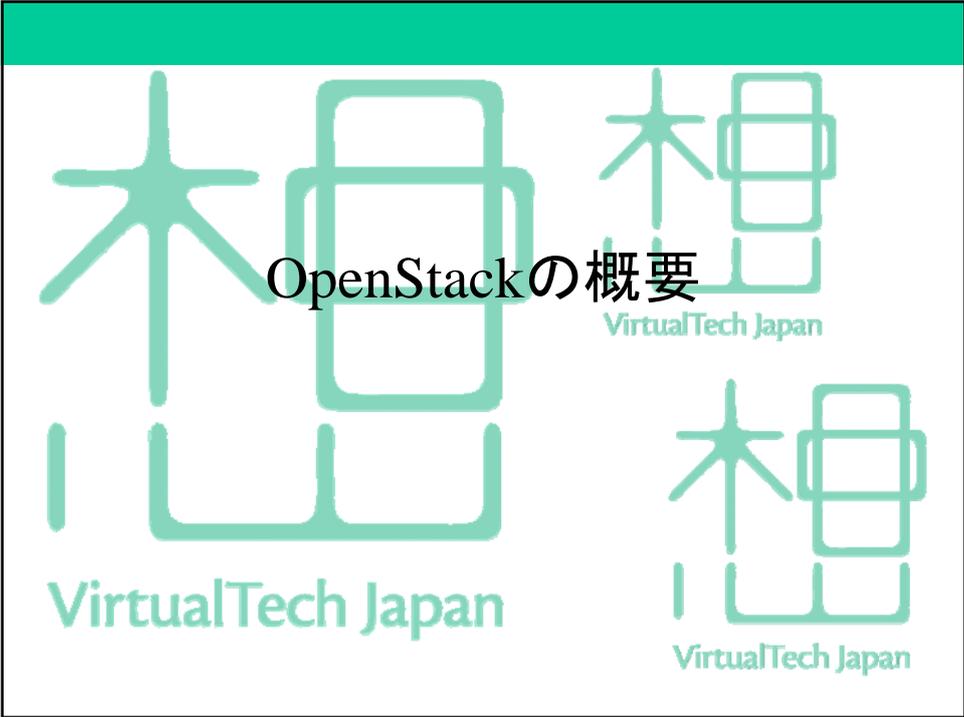
本日のアジェンダ

- OpenStackの概要
- OpenStack導入手順
 - Ubuntu Server 11.10のインストールと設定
 - 各種コンポーネントのインストールと設定
 - イメージの作成
 - インスタンスの起動

まずはOpenStackを体験してみたい人向け

4





OpenStackの構成要素

- Nova
 - コンピュータ(計算機)インフラ
 - 仮想マシンやネットワークを管理
- Swift
 - ストレージインフラ
 - 今回は使わない
- Glance
 - イメージ管理
 - 仮想マシンのイメージを登録

7



Novaの主なコンポーネント

- API Server (nova-api)
 - 外部にAPIを提供
- Message Queue (rabbit-mq server)
 - 各コンポーネントへの通信をキューイング
- Compute Workers (nova-compute)
 - インスタンスの管理
- Network Controller (nova-network)
 - ネットワークの管理
- Volume Worker (nova-volume)
 - ボリュームの管理
- Scheduler (nova-scheduler)
 - API呼び出しのコンポーネントへの割り当て

8



今回の方針

- Ubuntu Server 11.10をベースに構築
 - 12.04は現在検証中
- すべてを1台でまかなう
 - 必要に応じて複数台での構成も可能
- 仮想マシンのイメージはGlanceに登録
- Swiftは使用しない
- ゲストOSもUbuntu Server 11.10で構築

9



Ubuntu Server 11.10の
インストールと設定

VirtualTech Japan

VirtualTech Japan

Ubuntu Server 11.10のインストールと設定

1. ベースOSとしてのインストール
 - KVMを有効にしておく
 - デスクトップ環境のインストール
 - 仮想マシンマネージャーのインストール
2. IPアドレスの設定
 - 固定IPアドレスに設定

11



ベースOSとしてのインストール

1. インストール時に以下のコンポーネントを追加でインストールしておく
 - OpenSSH server
 - Virtual Machine host
2. デスクトップ環境のインストール
 - # apt-get update
 - # aptitude -y install ubuntu-desktop
3. システムの再起動
4. 仮想マシンマネージャーのインストール
 - # apt-get -y install virt-manager

12



固定IPアドレスの設定

1. /etc/network/interfacesの編集

```
auto eth0
iface eth0 inet static
    address 192.168.0.40
    network 192.168.0.0
    netmask 255.255.255.0
    broadcast 192.168.0.255
    gateway 192.168.0.1
    dns-nameservers 192.168.0.1
```

1. networkingサービスの再起動

– # /etc/init.d/networking restart



各種コンポーネントの
インストールと設定

VirtualTech Japan

VirtualTech Japan

各種コンポーネントのインストールと設定

1. Novaコンポーネントのインストール
2. MySQLのインストール
3. Novaデータベースの初期化
4. nova.confの設定と適用
5. ユーザーとプロジェクトの作成
6. novarcの修正
7. ネットワークの作成
8. Glanceのインストールと設定
9. nova-computeのインストール

15



Novaコンポーネントのインストール

- Message Queue (rabbit-mq server)
 - API Server (nova-api)
 - Network Controller (nova-network)
 - Scheduler (nova-scheduler)
1. Novaコンポーネントのインストール
 - # apt-get -y install rabbitmq-server python-greenlet python-mysqldb nova-common nova-doc python-nova nova-api nova-network nova-objectstore nova-scheduler
 2. eucaツールのインストール
 - # apt-get -y install euca2ools unzip

16



MySQLのインストール

1. MySQLのインストール
 - # apt-get -y install mysql-server
 - パスワードを3回聞かれるが何も入力せず Enterキー
2. /etc/mysql/my.cnfの修正

bind-address = 0.0.0.0
3. MySQLの再起動
 - # restart mysql

17



Novaデータベースの初期化

1. MySQLへの接続
 - # mysql
2. データベースの作成
 - mysql> create database nova;
3. 権限の付与
 - mysql> grant all privileges on nova.* to 'nova'@'%' identified by 'password';
4. MySQL接続の終了
 - mysql> exit

18



/etc/nova/nova.confの設定(1)

```
--verbose
--dhcpbridge_flagfile=/etc/nova/nova.conf
--dhcpbridge=/usr/bin/nova-dhcpbridge
--logdir=/var/log/nova
--state_path=/var/lib/nova
--lock_path=/var/lock/nova
--s3_host=127.0.0.1
--rabbit_host=127.0.0.1
--cc_host=127.0.0.1
--ec2_url=http://192.168.0.40:8773/services/Cloud
--fixed_range=10.0.0.0/8
--FAKE_subdomain=ec2
--sql_connection=mysql://nova:password@127.0.0.1/nova
--network_manager=nova.network.manager.VlanManager
```

19



/etc/nova/nova.confの設定(2)

```
—image_service=nova.image.glance.GlanceImageService
—glance_api_servers=127.0.0.1:9292

—allow_admin_api=true

—vlan_start=1100
—use_s3=true
—libvirt_type=kvm
—use_syslog=false
—node_availability_zone=nova
—use_cow_images=true
—auth_driver=nova.auth.dbdriver.DbDriver
—scheduler_driver=nova.scheduler.zone.ZoneScheduler
—use_ipv6=false
—use_deprecated_auth
```

20



Novaの設定の適用

1. データベースの同期
 - # nova-manage db sync
 - 警告が出るが気にしないでも良い
2. 各サービスの再起動
 - # restart nova-api
 - # restart nova-network
 - # restart nova-objectstore
 - # restart nova-scheduler

21



ユーザーとプロジェクトの作成

1. ユーザーの作成
 - # nova-manage user create --name=user1
2. プロジェクトの作成
 - # nova-manage project create --project=project1 --user=user1
3. 各種情報をまとめたZIPファイルの作成
 - # nova-manage project zipfile --project=project1 --user=user1
4. ZIPファイルの展開
 - # unzip nova.zip

22



Novaの動作状況を確認

1. novarcの読み込み

– # source novarc

2. ゾーン情報の確認

– # nova zone-info

```
+-----+-----+
| Property | Value |
+-----+-----+
| hypervisor | xenserver;kvm |
| name      | nova    |
| os       | linux;windows |
+-----+-----+
```

23



novarcの内容(抜粋)

```
export EC2_ACCESS_KEY="7b2c0f28-fc0c-420b-
be0a-c47a81404e5e:project1"
export
EC2_URL="http://192.168.0.40:8773/services/Cloud"
export S3_URL="http://192.168.0.40:3333"
export NOVA_API_KEY="7b2c0f28-fc0c-420b-be0a-
c47a81404e5e"
export NOVA_URL="http://192.168.0.40:8774/v1.1/"
```

24



ネットワークの作成

1. nova-manageコマンドの実行
 - # nova-manage network create
--label test1
--fixed_range_v4=10.1.1.0/24
--vlan=1011
--bridge_interface=eth0
 - # nova-manage network create
--label test2
--fixed_range_v4=10.1.2.0/24
--vlan=1012
--bridge_interface=eth0
2. ネットワーク作成の確認
 - # nova-manage network list

25



Glanceのインストールと設定

1. Glanceのインストール
 - # apt-get -y install glance
2. MySQLへの接続
 - # mysql
3. データベースの作成
 - mysql> create database glance;
4. データベース権限の付与
 - mysql> grant all privileges
on glance.* to 'glance'@'%' identified by
'password';
5. MySQL接続の終了
 - mysql> exit

26



Glanceの設定

1. /etc/glance/glance-registry.confの修正

```
sql_connection = mysql://glance:password@127.0.0.1/glance
```

1. サービスの再起動

- # restart glance-api
- # restart glance-registry

27



nova-computeのインストール

1. nova-computeのインストール

- # apt-get -y install
nova-compute

2. nova-computeサービスの再起動

- # restart nova-compute

28



Novaコンポーネントの状況を確認

1. Novaのサービスリストを表示

– # nova-manage service list

```
Binary      Host      Zone Status State Updated_At
nova-network openstack nova enabled :-) 2011-11-14 17:22:24
nova-scheduler openstack nova enabled :-) 2011-11-14 17:22:22
nova-compute openstack nova enabled :-) 2011-11-14 17:22:18
```

2. StatusがXXXの場合は異常

– しばらく待ってみる

– サービスの再起動などで正常状態にする



仮想マシンイメージの作成

VirtualTech Japan

VirtualTech Japan

ゲストOSのインストール

- 仮想マシンマネージャーを使ってゲストOSとしてUbuntu 11.10をインストール
- 仮想ディスク作成の注意点
 - server.imgとして作成することを想定
 - 仮想マシン名をserverとする
 - LVMにしないこと
 - 1パーティションとすること
 - swap領域は作らないこと
 - /パーティションをext4で初期化すること

31



ゲストOSの設定

1. OSインストール後、仮想マシンを再起動
2. アップデートの適用
 - # apt-get update
 - # apt-get upgrade
 - # apt-get -y install openssh-server [cloud-init](#)
3. /etc/fstabの修正
 - [LABEL=uec-rootfs](#) / ext4 defaults 0 0
4. udevルールの削除
 - # rm /etc/udev/rules.d/70-persistent-net.rules

32



VMイメージのext4イメージ化

- 仮想マシンマネージャーを使って作成されたVMイメージからext4領域部分だけを切り出す
- パーティションの開始セクターを確認し、開始セクターまで飛ばしてext4イメージ化
 - 通常はWindows系に合わせて2048セクターから第1パーティション開始

33



イメージのループバックマウント

1. ゲストOSのシャットダウンとVMの停止
2. イメージのループバックマウント
 - # losetup -f
/var/lib/libvirt/images/server.img
3. ループバックマウントのデバイス名を確認
 - # losetup -a

```
/dev/loop0: [fc00]:12845192  
(/var/lib/libvirt/images/server.img)
```

34



イメージの開始セクターの確認

1. 開始セクターの確認

– # fdisk -l /dev/loop0

デバイス	ブート	始点	終点	ブロック	Id	システム
/dev/loop0p1	*	2048	16775167	8386560	83	Linux

2. ループバックマウントの解除

– # losetup -d /dev/loop0

3. オフセットを設定して再度ループバックマウント

– オフセット値は開始セクター数 × 512バイトの数

– # losetup -f -o 1048576
/var/lib/libvirt/images/server.img

– # losetup -a

/dev/loop0: [fc00]:12845192 (/var/lib/libvirt/images/server.img), offset 1048576

35



イメージの書き出し

1. RAWイメージの書き出し

– # dd if=/dev/loop0 of=serverfinal.img

2. ループバックマウントの解除

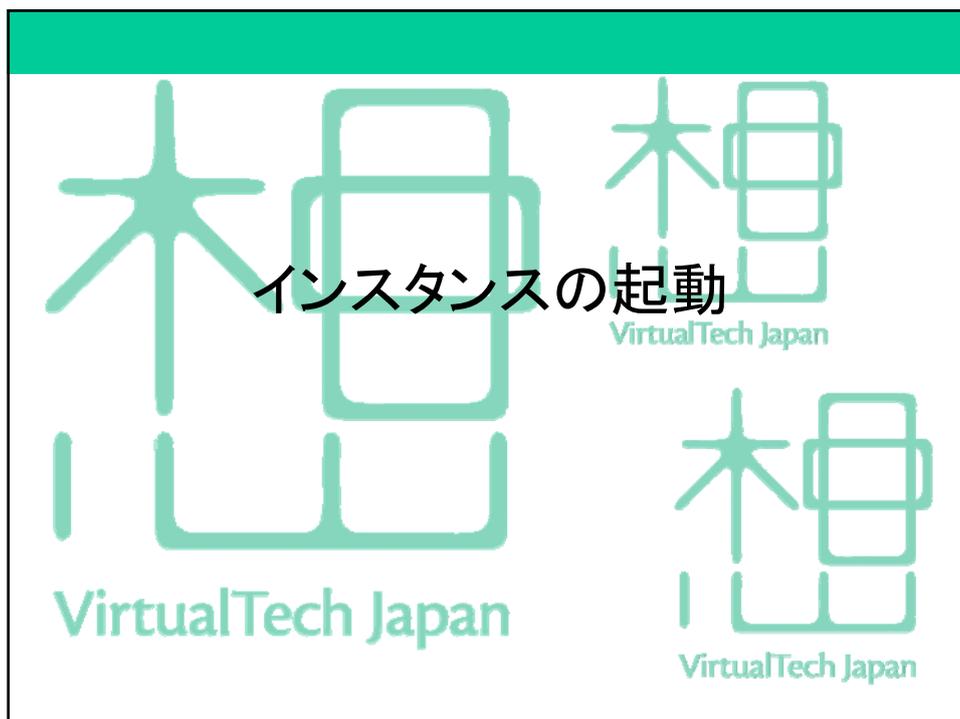
– # losetup -d /dev/loop0

3. ラベルの設定

– # tune2fs -L uec-rootfs serverfinal.img

36





カーネルの登録

1. カーネルイメージバンドルの作成
 - # euca-bundle-image -i /boot/vmlinuz-3.0.0-12-server -p kernel --kernel true
2. カーネルイメージバンドルのアップロード
 - # euca-upload-bundle -b base1 -m /tmp/kernel.manifest.xml
3. カーネルイメージの登録
 - # euca-register base1/kernel.manifest.xml

初期化RAMディスクの登録

1. 初期化RAMディスクイメージバンドルの作成
 - # euca-bundle-image -i /boot/initrd.img-3.0.0-12-server -p ramdisk --ramdisk true
2. 初期化RAMディスクイメージバンドルイメージバンドルのアップロード
 - # euca-upload-bundle -b base1 -m /tmp/ramdisk.manifest.xml
3. 初期化RAMディスクイメージバンドルイメージの登録
 - # euca-register base1/ramdisk.manifest.xml

39



ディスクイメージの登録

1. ディスクイメージバンドルの作成
 - # euca-bundle-image -i serverfinal.img -p machine --kernel aki-1 --ramdisk ari-2
2. ディスクイメージバンドルのアップロード
 - # euca-upload-bundle -b base1 -m /tmp/machine.manifest.xml
3. ディスクイメージバンドルの登録
 - # euca-register base1/machine.manifest.xml
4. 登録にはかなり時間がかかる

40



もう少し楽なイメージ登録方法

- カーネルの登録
 - # nova-manage image kernel_register
 - path=/boot/vmlinuz-3.0.0-12-server --owner=admin
 - name=kernel_1 --public=T --arch=x86_64
- initrdの登録
 - # nova-manage image ramdisk_register
 - path=/boot/initrd.img-3.0.0-12-server --owner=admin
 - name=ramdisk_1 --public=T --arch=x86_64
- イメージの登録
 - # nova-manage image image_register --path=serverfinal.img --owner=admin --name=image_1 --public=T --arch=x86_64
 - kernel=aki-1 --ramdisk=ari-2
- 3つ同時に登録
 - # nova-manage image all_register --image=serverfinal.img
 - kernel=/boot/vmlinuz-3.0.0-12-server
 - ram=/boot/initrd.img-3.0.0-12-server --owner=admin
 - name=image_2 --public=T --arch=x86_64



インスタンスの起動

1. 登録イメージの一覧
 - # euca-describe-images
2. インスタンスの起動
 - # euca-run-instances [AMIのID](#)
3. インスタンスの一覧
 - # euca-describe-instances
4. インスタンスへの接続
 - # ssh [username@IPアドレス](#)
5. インスタンスの終了
 - # euca-terminate-instances [インスタンスID](#)



トラブルシューティング

- 何かうまく動作しない
 - イメージサービスは処理完了に時間がかかる
 - topコマンドでプロセス状態を確認
 - 複数サービスの連動なので、とりあえず再起動してみましょう
- ログはどれを見ればいい？
 - /var/log/nova/ディレクトリのnova-compute.log、nova-api.logあたりを見る
- ドキュメントは？
 - <http://docs.openstack.org/> を参照

43



いくつかの宿題

- 複数台によるOpenStack環境の構成
 - nova-computeを複数台とか
- 外部からのネットワーク接続
 - Floating IPを利用してグローバル接続
 - VPNを利用してプライベート接続
- Swiftの利用

44



スタッフ大募集中！

- 日本仮想化技術株式会社は常時スタッフを募集中
- 『クラウド方面で何かチャレンジしたい！』という人
- これまでのスキルを活かしたキャリアを探している人
- 将来起業してみたいという人

自由闊達な社風の我が社と一緒に働きませんか？

まずは気軽に recruit@VirtualTech.jp まで

45



仮想化について相談したい

無料コンサルティング実施中

- 既存環境を仮想化移行する概要設計
- 他社提案に対する「セカンドオピニオン」
- もちろんOpenStack構築・カスタマイズも

お気軽にお問い合わせください

46



お問い合わせ先

「仮想化環境を構築したいが、どこに相談すればいいの？」

まずは我々にご相談ください



VirtualTech Japan

日本仮想化技術株式会社

<http://VirtualTech.jp/>
sales@VirtualTech.jp
050-7571-0584

