

# OS 授業向けマルチユーザ VM 環境の構築

平良太貴<sup>†1</sup> 河野真治<sup>†2</sup>

ブレードサーバ上の VM を自由に起動・停止ができる VM 管理ツール、ie-virsh を開発した。本学科ではローカルな環境で動作した VM イメージを、豊富なリソースを持つブレードサーバ上で動作させることが可能である。ie-virsh はマルチユーザに対応し、複数台の VM を管理できる。更に ie-virsh を使用すれば、学生が作成した VM のセキュリティやネットワークの設定が保証され、ブレードサーバへのアップロードや起動・停止を自由に行うことができる。それにより学生が VM を管理できる幅が広がり、学習効率が向上する。

## Suggest multi user VM environment for OS class

TAIKI TAIRA<sup>†1</sup> and SHINJI KONO<sup>†2</sup>

Cerium Task Manager is a parallel programming framework. To achieve good performance in GPGPU using Open CL, various tuning is needed. In particular, it is necessary to implement the dependency of task in Cerium by the function of Open CL. But, to match specialization for OpenCL spoiles of flexibility of framework. Balance of flexibility and the performance is considered. We evaluate example Sort, Word count, and FFT.

### 1. 研究の目的

情報工学科では、Operating System という授業がある。その授業では OS についてのみではなく IT 技術に関する様々な課題を生徒に出し、提出させる。授業 Operating System での課題の中に仮想化に関する課題があり、その課題ではローカルまたはブレードサーバ上に置いた VM でのプログラムの実装と計測を行う。

情報工学科ではブレードサーバを保持しており、当研究室ではそのブレードサーバのうち 4 台を使用できる。授業 Operating System で出される課題に対応するため生徒がブレードサーバへ VM をアップロードし、VM を動作させ、自由に操作できる必要がある。授業 Operating System に対応するため、本研究では ie-virsh の実装を行った。ie-virsh は本研究室の保持するブレードサーバ上の VM を起動・停止することができ、更にマルチユーザで使用することができる。授業で使用する複数名の生徒一人一人が、複数の VM をブレードサーバで作成し、起動・停止することが可能となった。

### 2. ie-virsh

ie-virsh は、virsh をラップして作られた VM 管理用のプログラムである。生徒は ssh で学科アカウントを使用して VM に接続後、ie-virsh を使用して VM を操作することができる。表 1 が ie-virsh の機能である。

表 1 ie-virsh のコマンド

define	XML の template を元に domain を作成
undefine	define で作成した domain を削除
list	define で作成した domain を一覧表示
start	指定した domain 名の VM を起動
destroy	指定した domain 名の VM を停止
dumpxml	domain の XML を参照

ie-virsh には virsh にあるような、ネットワークの構成などの管理者側がすべき操作はなく、管理者でない生徒は操作できないようになっている。

生徒が ie-virsh を使用して VM を起動する手順はこうである。まず生徒のノートパソコンで、VMWare や Virtual Box を使って Linux をインストールし、イメージを作成する。作成したイメージをブレードサーバにアップロードする。イメージを qcow2 に変換して指定のディレクトリに配置し、以下のコマンドを実

<sup>†1</sup> 琉球大学理工学研究科情報工学専攻  
Interdisciplinary Information Engineering, Graduate  
School of Engineering and Science, University of the  
Ryukyus.

<sup>†2</sup> 琉球大学工学部情報工学科  
Information Engineering, University of the Ryukyus.

行する。

```
% ie-virsh define [domain name]
```

そうすると、template XML を元に domain が生成される。ie-virsh は XML の template を持ち、その template は生徒が VM のイメージをブレードサーバにアップロードして define した際に使用される。そこには virsh 上で VM を使用するために必要な設定が記述されている。生成された domain は以下のように起動することができる。

```
% ie-virsh start [domain name]
```

自身が持っている VM の状態を、下記で見ることができる。

```
% ie-virsh list
```

### 2.1 XML template

生徒が作成する VM は、XML のテンプレートを元に作成される。テンプレートは VM にしておくべき設定が記述されており、ie-virsh はこのテンプレートを元に生徒の XML ファイルを生成する。XML template にされている設定は以下ようになる。

- ネットワークの設定
- I/O 設定
- VM イメージのフォーマット
- CPU 数
- メモリ容量

ie-virsh は XML template を元に以下の設定を追記したファイルを作成する。

- VM 名
- UUID
- VM イメージの配置

以上が生徒が利用するための XML テンプレートの設定である。

## 3. libvirt

仮想マシンの制御を抽象化したライブラリである。VM の情報を習得、操作することが可能な API 群となっている。C 言語の API を持っており、それ以外の言語にもバインディングされている。

図 1 は libvirt のアーキテクチャの概要である。アプリケーションから libvirt API を呼び出すと、API に従って内部の VMM API もしくは資源管理 API を呼び出し、制御する。

libvirt は VM の管理だけでなく、仮想ネットワーク、仮想ストレージも管理することができる。もともとは Xen に対して API を提供していたが、2014 年現在、多くのハイパーバイザに対応している。以下が対応しているハイパーバイザの例である。

- Xen
- Qemu
- Kernel-based Virtual Machine (KVM)
- Linux Containers (LXC)

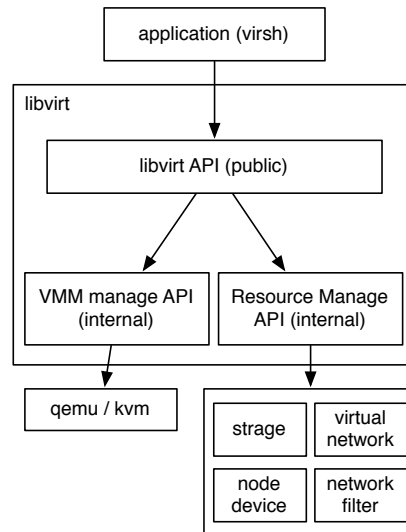


図 1 libvirt architecture

- User Mode Linux (UML)
- VirtualBox
- OpenVZ
- VMWare ESX / GSX
- VirtualBox
- Bhyve

libvirt は XML にパラメータを保存することができる。以下が XML に保存できるパラメータである。

- VM 名 (domain 名)
- 割り当てる CPU・メモリ
- ディスクの形式
- 起動オプション
- ネットワーク設定
- コンソール設定

libvirt でこれらを管理することにより、煩雑なオプションの管理をしなくてすむ。

### 3.1 virsh

libvirt には virsh というコマンドラインインターフェイスがあり、libvirt の API でできる制御の殆どを virsh で制御できる。VM の起動や停止、情報の表示、ゲストが接続しているネットワークの管理をすることができる。また、virsh を使用することでゲストを別のホストへ移行することも可能である。

## 4. Kernel-based Virtual Machine (KVM)

Linux 自体を VM の実行基盤として機能させるソフトウェアである。完全仮想化により、OS の仮想化環境を提供する。

図 2 は、KVM のアーキテクチャである。KVM は Linux 用のカーネルモジュールとして実装されており、OS が持つメモリ管理プロセスやスケジューリング機

表 2 vagrant の主なコマンド

up	Vagrant Box を起動
destroy	仮想マシンの削除
halt	起動している Vagrant Box の停止
ssh	起動している Vagrant Box へ ssh 接続
status	ステータスの確認
box add	新しい Vagrant Box の追加

能を利用している。そのため他の仮想マシンソフトウェアに比べ、KVM 自体のコードは簡潔なものになっている。

Intel VT や AMD-V などの仮想化支援機能を持つプロセッサや BIOS を持っているパソコン上で動かすことができる。

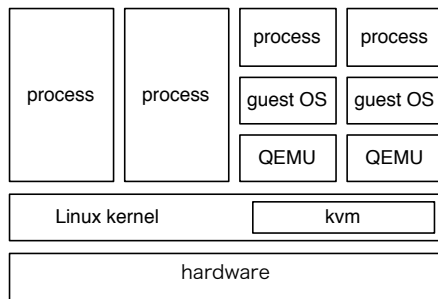


図 2 KVM architecture

## 5. Vagrant

Vagrant は異なる環境に移行可能な開発環境を簡単に構築・管理・配布することが出来る開発環境作成ツールである。手軽にテスト環境を導入することができ、変更が加わっても開発環境・本番環境に自動的に適用される。また、環境を気軽に捨てることも可能である。ホスト環境として、VirtualBox や VMWare などで動かすことができる。表 2 は Vagrant で使用することの主な機能である。

### 5.1 Vagrant Box

Vagrant で仮想マシンを利用する際に、仮想マシンのベースとなるイメージファイルである。Vagrant で Vagrant Box に開発環境を構築し、配布することができる。また配布されている Vagrant Box を取得し、Vagrant で起動し使用することが可能である。

## 6. Oracle Cluster Filesystem(OCFS2)

OCFS2 は汎用の、共有ディスククラスタファイルシステムである。一つのブロックデバイスを複数のパソコンから同時に読み書きすることができる。

各計算ノードがそれぞれファイルシステムへの処理を行い、ストレージに対して個別に読み書きをする。

一貫性を持った読み書きを実現する機構として Distributed Lock Manager (DLM) が使われる。DLM で他の計算ノードと矛盾しないように調整しながらそれぞれの計算ノードが並行してストレージへの読み書きを行うことで、全体として一貫性のあるファイルシステムを実現している。

標準のファイルシステムインターフェイスを通じてすべてのノードが並行してストレージに読み書きできるため、クラスタにまたがって動作するアプリケーションの管理が容易になる。

### 6.1 NAS との比較

SAN を ext3 ・ ext4 のようなファイルシステムを使用する場合、別々のパソコンから同時に書き込むと整合性が取れなくなる。今回複数のブレードサーバから読み書きをされるため、並行で I/O が起きてしまうと整合性が取れなくなり、書き込むデータが破損してしまう恐れがある。OCFS2 と比べて書き込みを制御しなければならなくなるため、使用することができない。

## 7. 評価

授業 Operating System で実際に使用した、VM 環境の構成を説明する。授業 Operating system は、受講者 60 名の授業である。

### 7.1 サーバの構成

ホストサーバの環境は以下である。

- OS:debian
- CPU: Intel(R) Xeon(R) CPU X5650 @ 2.67GHz
- Memory: 24GB
- Filesystem: OCFS2

また OCFS2 との接続は、図??となっている。複数台のブレードサーバから OCFS2 でフォーマットされたファイルシステムへ接続し、書き込みを行う。ブレードに内蔵されている SSD では大量の VM イメージを保存するには足りないため、外部の記憶装置を利用する必要があるためである。また別のブレードサーバ上に KVM をたてた場合に移行が簡単になる。

情報工学科では、グローバル IP アドレスを取得することができる。生徒は取得した IP アドレスを使用して VM へ ssh アクセスする。そのため、virsh が作る仮想ネットワークではなく、情報工学科の DHCP サーバによって生徒の IP アドレスが受け取られるように設定する必要がある。それには仮想ブリッジを配置してその問題に対処した。

図 4 は ie-virsh を使用して生徒が VM を配置するホストの構成である。

また、情報工学科で使用されている LDAP サーバの情報を使用して、学科のアカウントで ssh ログイン可能な状態に設定した。そうすることで、生徒一人一人のアカウントを登録する必要がなく、またアカウント

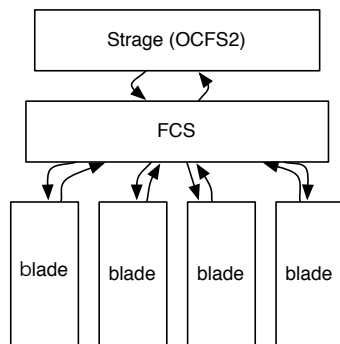


図 3 All Server structure

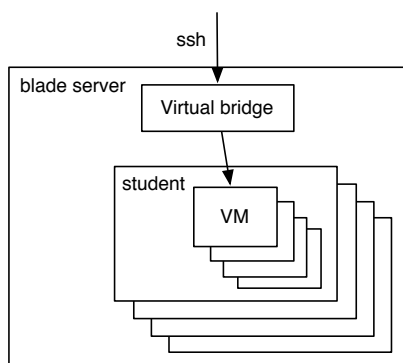


図 4 Server structure

ト名は学籍番号になっているため管理が容易となる。

## 7.2 マルチユーザでの利用

ie-virsh はマルチユーザで利用できるように実装している。今回授業 Operating System で使用した際は、受講者 60 人が使用し、授業 Operating system の課題を行った。生徒一人一人が VM をブレードサーバにアップロードし、ssh 接続し操作することができた。

## 7.3 Vagrant Box の利点・欠点

今回生徒のノートパソコン上で VM を作成・設定する際に、Vagrant Box を利用した。Vagrant は、既にインストールされている Vagrant Box イメージをノートパソコンで起動することができる。生徒はその VM イメージを使用して設定し、その後ブレードサーバにアップロードする。アップロードした後に ie-virsh で起動し、サーバとして動作させる。

そうすることによって、始めから OS をインストールをすることなく手軽に開発環境を構築することができる。

しかし今回、セキュリティ対策をしていないため Vagrant Box のセキュリティに関する設定が甘かった。そのため生徒は対策しないままブレードサーバへアップロードしてしまうということが起きた。パスワードやファイアウォールの設定の自動化が必要となっ

ると考えられる。よい学習環境を整えるために、VM の設定を考えることが必要である。

## 7.4 vSphere Client との比較

授業 Operating System での VM を使用しての授業は、今回の例に挙げたものだけではなく毎年行われている。生徒に VMWare ESX を利用させるために、vSphere Client を使用した。vSphere Client を利用するためには、Windows PC が必要となる。情報工学科では主に Mac OS X を使用しているため、ssh で接続すればよい ie-virsh と比較すると vSphere Client を使用するためには手間がかかる。

また、vSphere Client は VM を操作するために必要な機能以外の機能が豊富なため、UI が煩雑となっており授業で必要な操作を学ぶのに手間がかかる。ie-virsh であれば、コマンドはシンプルで数も少なく覚えやすい。

## 7.5 Vagrant と virsh の比較

ie-virsh とは別に、KVM をハイパーバイザとして Vagrant を使って VM を操作可能にした。ie-virsh と同様にマルチユーザとして利用できるようにしようとしたが、Vagrant を KVM をハイパーバイザとして動かすと、複数の VM をネットワークに接続して ssh で入る、ということができなかった。

Vagrant Box も上記にあるようにグローバル IP アドレスを設定してインターネットに接続するにはセキュリティの面を考慮する必要があるため、対策する必要がある。

## 7.6 Web Service としての実装との比較

## 8. ま と め

### 参 考 文 献

- 1) 眞大千, 河野真治: Cerium Task Manager におけるマルチコア上での並列実行機構の実装, 第 53 回プログラミング・シンポジウム (2012).
- 2) Aaftab Munshi, Khronos OpenCL Working Group: *The OpenCL Specification Version 1.0* (2007).
- 3) Khronos OpenCL Working Group: *OpenCL 1.2 Reference Pages* (2012).
- 4) 北山洋幸: OpenCL 応用 メニーコア CPU & GPGPU 時代の並列処理, カットシステム (2012).
- 5) 金城裕, 河野真治, 多賀野海人, 小林佑亮 (琉球大学): ゲームフレームワーク Cerium TaskManager の改良, 情報処理学会システムソフトウェアとオペレーティング・システム研究会 (OS) (2011).