



**Hewlett Packard**  
Enterprise

# Integrity VM で やさしく学ぶ サーバー仮想化

サーバー仮想化は、たとえば言えば「タクシーの快適さをバス料金で提供する」という技術革新を実現したテクノロジーです。そのサーバー仮想化技術が、いま企業の基幹システムにも導入され始めています。過去 8 年にわたり「ミッションクリティカル環境のためのサーバー仮想化技術」を追求してきた HPE は、その最新のサーバー仮想化の新機能として「Integrity VM」を昨年 12 月にリリースしました。本連載では、この Integrity VM を実際に使いながら、サーバー仮想化のメリットをわかりやすく紹介します。

《連載期間：2006 年 4 月～2007 年 1 月》

## —目次—

### 第 1 回 Integrity VM って何だ？

Integrity VM の特徴を解説し、どのような局面でそのアドバンテージを生かせるかについて説明します。

### 第 2 回 Integrity VM をインストールする

Integrity VM の VM ホストと VM ゲストそれぞれが消費するメモリ領域やディスク領域の計算方法を紹介し、Software Distributor (SD-UX) の `swinstall` コマンドによる Integrity VM のインストール方法を解説します。

### 第 3 回 ストレージとネットワークの仮想化

Integrity VM では、個々の VM ゲストについて、仮想ストレージ・デバイスと呼ばれる仮想的な SCSI ディスクを設け、それに VM ホストの物理ディスクや論理ボリューム、ファイルを割り当て可能です。今回は、Integrity VM におけるストレージとネットワークの仮想化について説明します。

### 第 4 回 VM ゲストの作成

Ignite-UX サーバーからのインストール方法と、仮想 DVD ドライブからのインストール方法の両方を説明します。また、VM ゲストを起動する 2 種類のコマンド、`hvvmstart` コマンドと `hvvmconsole` コマンドの使い方を解説します。

### 第 5 回 VM ゲストへのリソース配分

Integrity VM における「仮想 CPU」へのリソース配分方法を説明します。仮想 CPU とは、「それぞれの VM が利用可能な最大 CPU 数」を表します。

### 第 6 回 Integrity VM の構成管理

VM ゲストや仮想スイッチの構成管理の方法を紹介します。

### 第 7 回 Virtualization Manager と Capacity Advisor による統合管理

仮想化で重要となる「Virtualization Manager」と「Capacity Advisor」による統合管理について説明します。

### 第 8 回 Ignite-UX によるネットワーク・インストール・その 1

Ignite-UX を用いて、HP-UX の OS イメージをネットワーク経由で VM ゲストにインストールする基本的な手順を説明します。

### 第 9 回 Ignite-UX によるネットワーク・インストール・その 2

各 VM に共通する設定で OS やソフトウェアをあらかじめインストールした「ゴールド・イメージ」を使用することで、標準的なソフトウェアをインストール済みの OS をすばやく準備できます。

### 最終回 Ignite-UX によるネットワーク・インストール・その 3

Ignite-UX サーバーによるネットワーク・インストールで用いる「ゴールド・イメージ」の作成手順を紹介します。

## 第 1 回

# Integrity VM って何だ？

2006 年 4 月 テクニカルライター 小林聡史

サーバー仮想化は、たとえて言えば「タクシーの快適さをバス料金で提供する」という技術革新を実現したテクノロジーです。そのサーバー仮想化技術が、いま企業の基幹システムにも導入され始めています。過去 8 年にわたり「ミッションクリティカル環境のためのサーバー仮想化技術」を追求してきた HPE は、その最新のサーバー仮想化の新機能として「Integrity VM」を昨年 12 月にリリースしました。本連載では、この Integrity VM を実際に使いながら、サーバー仮想化のメリットをわかりやすく紹介します。今回は、Integrity VM の特徴を解説し、どのような局面でそのアドバンテージを生かせるかについて説明します。

### 「タクシーの快適さをバス料金で提供する」サーバー仮想化

いま「VPS(Virtual Private Server)」と呼ばれる新しいサービスが、サーバーホスティング分野で急速に普及しつつあります。VPS とは、「仮想専用サーバー」つまりあたかも 1 台の専用サーバーのように使える仮想的なサーバーを意味します。

従来のサーバーホスティングでは、アプリケーションごとにマシンを 1 台ずつ占有させる「専用サーバー」か、同じサーバーに複数のアプリケーションを収容する「共用サーバー」が一般的でした。VPS は、これら両者のいいところを兼ね備えたサービスです。つまり、共用サーバー並みの低コストでありながら、専用サーバーと同じく 1 つの OS をまるごと占有できる使いやすさを実現します。たとえて言えば「タクシーの快適さをバス料金で提供できる」のが VPS です。

そんなにわかには信じがたい技術革新を可能にしたのが、「サーバー仮想化」のテクノロジーです。例えばバイエムウェア社の VMware、マイクロソフト社の Virtual Server、そしてオープンソースソフトウェアの Xen といったサーバー仮想化ツールを利用することで、1 台のサーバー上でいくつもの仮想サーバーを動作させることが可能になりました。つまりサーバー仮想化は、「1 台のサーバーから何台ものサーバーを生み出せる術」なのです。

### ミッションクリティカル環境+サーバー仮想化=VSE

とはいえ、こうしたサーバー仮想化製品の大半はいずれも登場して日が浅く、安定性が求められる環境に導入できるだけのハードウェアと一体となった設計思想や豊富な実績がありません。そのため、企業の IT システム構築では、サーバー仮想化の導入事例はまだまだ少数にとどまっています。そうしたなか、1997 年に最初の仮想化製品をリリースして以来、過去 8 年にわたり「ミッションクリティカル環境のためのサーバー仮想化」を追求し、着実に実績を重ねているのが HPE です。

HPE では、VSE(Virtual Server Environment)と呼ばれる、ミッションクリティカル環境向けのサーバー仮想化技術ポートフォリオを提供しています。

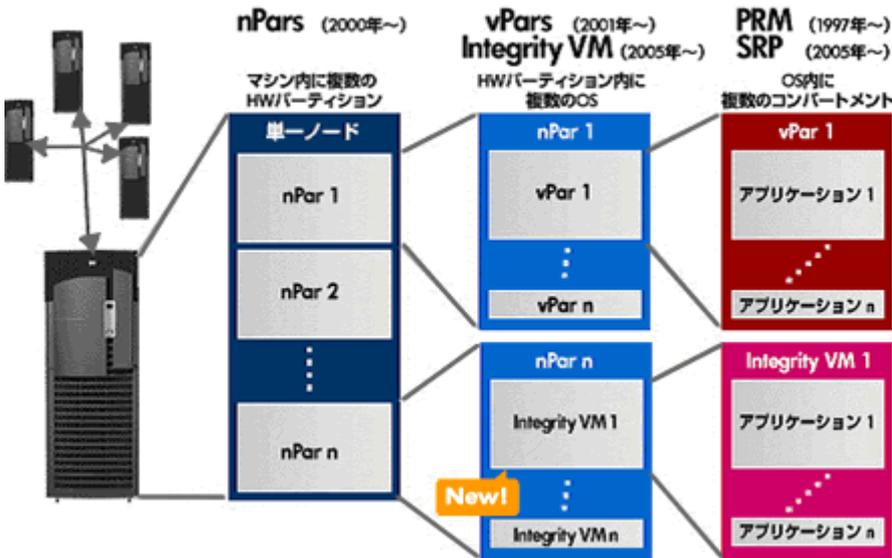


図 1 : HP VSE を構成するサーバー仮想化技術

VSE では、大まかに以下の 3 種類のサーバー仮想化技術を提供しています。

- ハードパーティション(nPars)
- ソフトパーティション(vPars、Integrity VM)
- リソース・パーティション(PRM、SRP)

まず、ハードパーティションである「nPartitions(nPars)」は、サーバーのハードウェア(CPU、メモリ、I/O など)を物理的にパーティション分割し、1 台のサーバー内部に複数のサーバーを作り出す技術です。それぞれのパーティションは完全に独立しており、OS レベルやハードウェアレベルの障害が発生しても、互いに影響を及ぼすことはありません。

つづいて、ソフトパーティションである Virtual Partitions(vPars) および Integrity VM は、サーバーをソフトウェア・レベルによってパーティション分割する技術です。1 台のサーバー内部で複数の OS を同時に動作させることができ、OS 内部でのトラブルを封じ込めることができます。また個々のパーティションに割り当てる CPU パワーや I/O リソースをリアルタイムに調節できるのが特徴です。

リソース・パーティションである PRM(Process Resource Manager) および SRP(Secure Resource Partition) は、1 つの OS のリソースを分割し、個々のアプリケーションに配分する技術です。PRM では CPU やメモリ、I/O など一定のリソースを各アプリケーションに%単位で正確に配分でき、他のアプリケーションの過負荷による影響を防ぐことができます。さらに SRP では、それぞれのアプリケーションをセキュリティ的に完全に隔離できます。

この VSE が提供するサーバー仮想化技術の豊富さ、そしてミッションクリティカル環境での実績の多さで、HPE は他のベンダーの追随を許していません。サーバー仮想化の優れた能力を活用したい、しかし同時に、企業の根幹を担う IT システムを新技術導入のリスクにさらしたくない——と悩むエンジニアにとって、VSE はうってつけのソリューションと言えるでしょう。

そして VSE のなかでも、もっとも新しい製品として昨年 12 月にリリースされたのが、「Integrity VM」です。後半では、この Integrity VM の特徴を掘り下げていきます。

## Integrity VM とは何か

Integrity VM は、その名が示すとおり、Integrity サーバーをまるごとソフトウェアで再現する仮想マシンです。つまり、Integrity サーバー上で Integrity VM を動作させることで、仮想的な Integrity サーバーを 2 台、3 台、4 台……といくらでも増やすことができます。よってサーバー・リソースに余裕がある限りは、もはや新しいサーバーを購入する必要はありません。

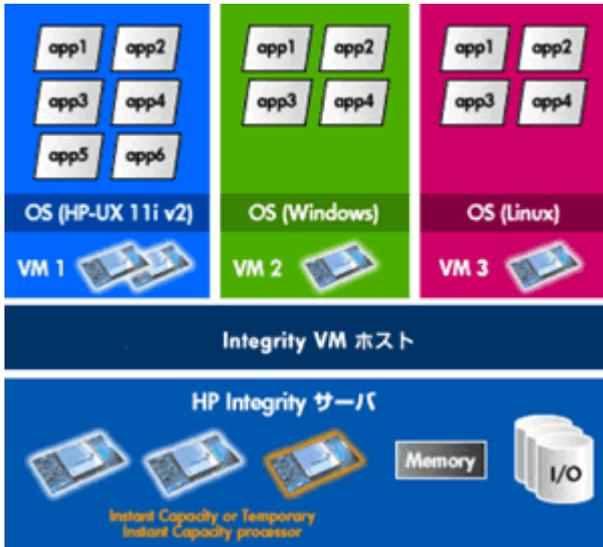


図 2 : Integrity VM の動作イメージ

Integrity VM では、Integrity VM 自体を動作させるための OS、すなわち「ホスト OS」として HP-UX 11i v2 に対応しています。一方、Integrity VM 内部で動作する OS、すなわち「ゲスト OS」としては、HP-UX 11i v2 に加えて、2006 年後半から段階的に Windows および Linux に対応する予定です。つまり、HP-UX・Windows・Linux という 3 つの OS を用途に応じて自在に選択し、仮想マシンに搭載できるのが、Integrity VM の大きな特徴です。

## vPars との違い

以前から HP-UX を利用していた方であれば、従来から提供されていたソフトパーティション技術、vPars と Integrity VM の類似性にお気づきでしょう。この両者は、どちらも Integrity サーバーの CPU やディスク、I/Oなどをソフトウェア的に分割するという点でよく似ています。

vPars と Integrity VM の異なる点は、前者が「ハードウェアコンポーネント単位の分割」をファームウェアレベルで実現するのに対し、後者は「ハードウェアの共有」をホスト環境により可能にする点です。例えば CPU リソースをパーティション分割するとき、vPars では各パーティションに最低でも 1 個の CPU を占有させます。一方 Integrity VM では、1/20 個分の CPU リソースさえ割り当てれば VM を構成できます。言い換えれば、CPU 1 個を最大 20 の VM で共有できる計算です。同様に、ディスク領域やネットワーク帯域についても、Integrity VM ではホスト環境により共有し、仮想化しますので、vPars より柔軟性の高いアサインメントが可能です。

こうした「粒度の細かなサーバー仮想化」が可能のため、Integrity VM はハイエンドやミッドレンジサーバーだけでなく、エントリーモデルの Integrity サーバーでの仮想化にも適しています。一方 vPars は、粒度が大きいため、ミッドレンジ以上のモデルの仮想化を実現し、ほとんどオーバーヘッドがないというメリットを備えています。

<表 : vPars と Integrity VM の比較>

	vPars	Integrity VM
サポート対象サーバー	ハイエンド/ミッドレンジ(セルベース)の Integrity サーバーおよび HP 9000 サーバー	Integrity サーバー全機種
サポートするゲスト OS	<ul style="list-style-type: none"> <li>HP-UX 11i v2 2005 年 5 月版以降</li> <li>HP-UX 11i v1(PA のみ)</li> </ul>	<ul style="list-style-type: none"> <li>HP-UX 11i v2 2005 年 5 月版以降</li> <li>Linux(2006 年末以降)</li> <li>Windows(2006 年後半)</li> </ul>
1CPU コアあたりの分割数	1 パーティション	20 仮想マシン
CPU の動的割り当て単位	CPU 単位	CPU 利用率単位
I/O 共有	不可	可能

## Integrity VM はこう使う

では、この Integrity VM はどのように活用すればよいのでしょうか。まず考えられるのは、開発環境やテスト環境のコンソリデーション(集約)です。

常時稼働するプロダクション環境とは異なり、開発環境やテスト環境はつねに利用されているわけではありません。特に、プロジェクトのカットオーバー以降はほとんど利用されなくなるケースが多いでしょう。とはいえ、運用中のトラブル対応や将来的な機能拡張を考えると、開発・テスト環境として用いてきたサーバーをすぐに他の用途へと回すことはできません。その結果、あまり使われることなく放置されるムダなサーバー・リソースとなってしまうがちです。

Integrity VM を導入することで、こうした開発・テスト環境のコンソリデーションが実現します。Integrity VM では、1つの VM の内容すべてが1つのファイル(もしくは論理ボリューム、ディスクなど)に記録されています。この VM を起動しない限り、サーバー・リソースを消費することはありません。そこで、開発やテスト作業のピーク時には豊富なサーバー・リソースを配分し、カットオーバー後には VM を閉じておくといった使い方が可能になります。

また、Integrity VM ではファイルコピー感覚で VM の複製や移設が可能のため、ミドルウェアやアプリケーションがインストール済みの VM をコピーし、検証用環境を瞬時に用意することもできます。

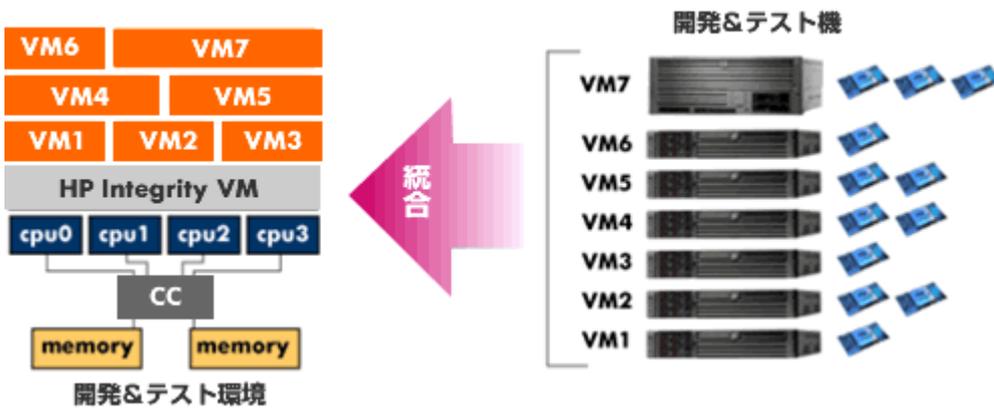


図 3 : Integrity VM による開発・テスト環境のコンソリデーション

ちなみに、Integrity VM ではマイグレーションシナリオとして、他の VM 環境間の移行を最新バージョンで提供し、「VM から物理サーバーへの移行」および「物理サーバーから VM への移行」を今後サポートする予定です。これにより、VM 環境間での連携はもちろんのこと、例えば VM 上に構築されたステージング(本番検証)環境をプロダクション用の物理サーバーに移行したり、または物理サーバー上の既存の開発環境を VM に移行してコンソリデーションしたり——といった使い方も可能になります。このように Integrity VM は、システム管理者にとってはきわめて重宝なツールとなります。

以上、今回は HP VSE が提供するサーバー仮想化技術と、その最新製品である Integrity VM の概要を紹介しました。次回以降は、Integrity VM を実際にインストールし、その使い方や動作メカニズムを解説していく予定です。

## 第 2 回

# Integrity VM をインストールする

2006 年 5 月 テクニカルライター 小林聡史

HPE では Integrity VM のトライアル版を無償で公開しており、Integrity サーバーと HP-UX 11i v2 2005 年 5 月版以降さえあれば簡単に試してみることが可能です。そこで今回は、Integrity VM のトライアル版を用いて実際にインストールする手順を説明します。Integrity VM の VM ホストと VM ゲストそれぞれが消費するメモリ領域やディスク領域の計算方法を紹介し、Software Distributor (SD-UX) の `swinstall` コマンドによる Integrity VM のインストール方法を解説します。また、インストール後に実施する仮想スイッチの作成コマンドの使い方を説明します。

## Integrity VM のシステム要件

今回は、Integrity VM のトライアル版を用いて実際にインストールする手順を説明します。

まずは、Integrity VM の動作に必要なとされるシステム要件を確認します。

<表 : Integrity VM のシステム条件>

項目	条件
ハードウェア	Integrity サーバー全機種
OS	HP-UX 11i v2 2005 年 5 月版 (0505) 以降
メモリ容量	<ul style="list-style-type: none"> <li>VM ホスト：750MB+オーバーヘッド (残りメモリ容量の 7.5%)</li> <li>VM ゲスト：HP-UX のメモリ容量+オーバーヘッド (7%)</li> </ul>
ディスク容量	<ul style="list-style-type: none"> <li>VM ホスト：50MB</li> <li>VM ゲスト：HP-UX のディスク容量</li> </ul>
ネットワーク	VM ゲストへのネットワーク接続には最低 1 枚の NIC が必要

Integrity VM は、Integrity サーバー全機種で動作可能です。一方、Integrity VM を動作させるための OS (VM ホストと呼びます) としては、HP-UX 11i v2 2005 年 5 月版 (0505) 以降が必要となります。ちなみに、いま使用している HP-UX のリリース ID を調べるには、以下のコマンドを入力します。

```
# swlist | grep HPUX11i
```

これにより、OS のリリース ID が次のように表示されます。

```
HPUX11i-OE      B.11.23.0603  HP-UX Foundation Operating Environment Component
```

この例では、「B.11.23.0603」という部分がリリース ID を示しており、この OS が HP-UX 11i v2 (すなわち B.11.23) の 2006 年 3 月版 (0603) であることが分かります。

## メモリ容量とディスク容量の確認

続いて、メモリ容量について確認します。Integrity VM 本体 (VM ホスト) は、およそ 750MB のメモリ領域を消費します。また、全メモリ容量から 750MB を差し引いた容量の 7.5% をオーバーヘッドとして消費します。例えば 16GB メモリを搭載した Integrity サーバーの場合、オーバーヘッドは  $(16\text{GB} - 750\text{MB}) \times 7.5\% = 1.15\text{GB}$  となり、VM ホストが消費するメモリは  $1.15\text{GB} + 750\text{MB} = 1.9\text{GB}$  になります。

一方、VM ホスト上で動作するゲスト OS (VM ゲストと呼びます) としては、HP-UX がまるごと一式動作しますので、HP-UX の最低メモリ容量 (1GB) とアプリケーションのメモリ容量、さらにオーバーヘッドとして 7% が消費されます。例えば VM ゲスト用に 3GB を確保するには、 $3\text{GB} + (3\text{GB} \times 7\%) = 3.21\text{GB}$  が必要です。これらの値をトータルすると、VM ゲストをひとつ運用する場合、 $1.9\text{GB} + 3.21\text{GB} = 5.11\text{GB}$  のメモリを確保しなくてはなりません。

次にディスク容量を計算します。VM ホストが 50MB のディスク領域を消費するほか、VM ゲストの HP-UX が必要とするディスク領域、さらに VM ゲストのメモリサイズに合わせたスワップ領域が必要となります。

## Integrity VM トライアル版のダウンロード

システム要件が確認できたところで、続いては Integrity VM のトライアル版のソフトウェア・バンドルを以下のサイトからダウンロードします。

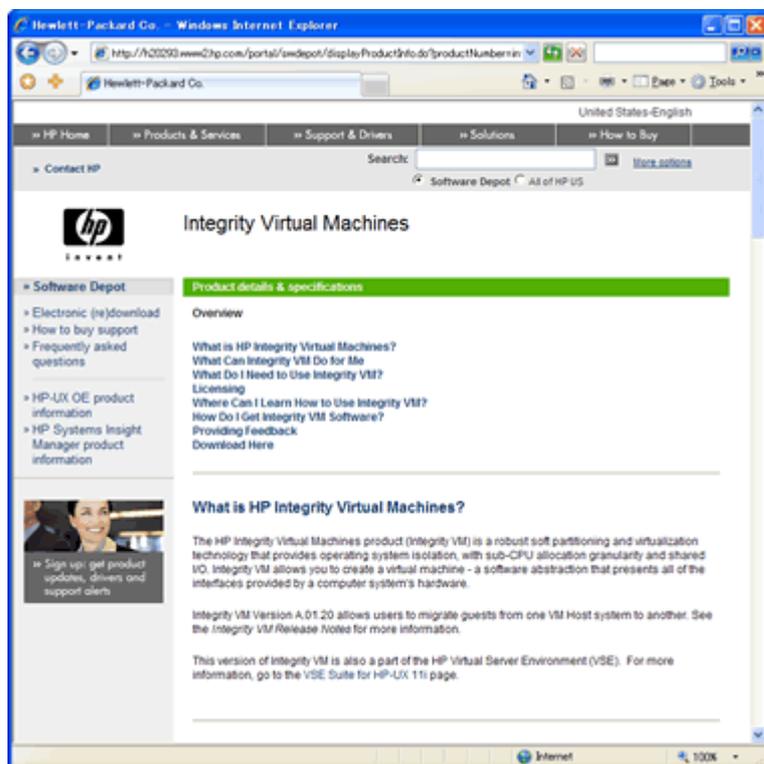


図 1 : Integrity VM トライアル版のダウンロードページ

このページの一番下にある「Receive for Free」をクリックし、つづいて表示されるフォームに氏名や住所などを入力したのち「Next」をクリックします。これにより、Integrity VM トライアル版ソフトウェア・バンドルの depot ファイルをダウンロードするリンクが表示されます。

後半では、ダウンロードした depot ファイルをインストールする手順を紹介します。

## Integrity VM のインストール

HP-UX の通常のソフトウェア・インストールと同様に、Integrity VM もまた Software Distributor (SD-UX) の swinstall コマンドによりインストールします。以下では TUI によるインストール例を紹介しますが、GUI でも手順はほとんど同じです。ちなみに、今回インストールするのは図 2 における VM ホスト部分となります。VM ゲスト部分のインストール手順については、次回説明する予定です。

## Integrity VM でやさしく学ぶサーバー仮想化

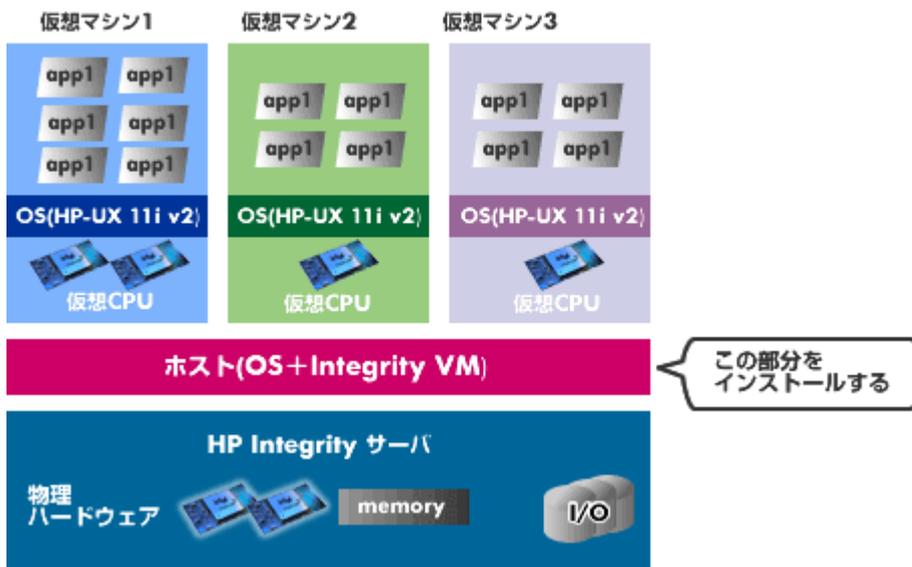


図 2 : Integrity VM の VM ホストと VM ゲスト

ではまず、swinstall を起動します。

```
# swinstall
```

swinstall を起動すると、depot ファイルの場所を指定するダイアログが表示されます。このダイアログの「Source Depot Path...」部分にて、先ほどダウンロードした depot ファイルのパスを入力し、「OK」を選択します。

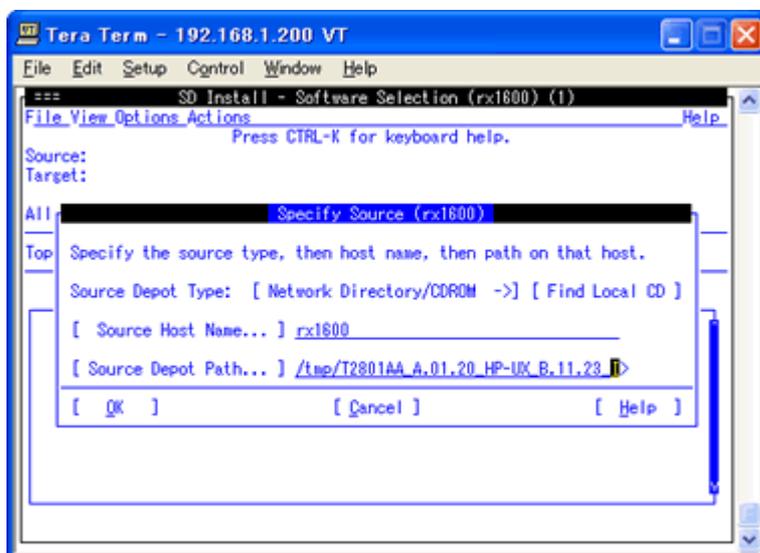


図 3 : depot ファイルの指定

これにより depot ファイルが読み込まれ、ソフトウェア・バンドルに含まれる内容が一覧表示されます。

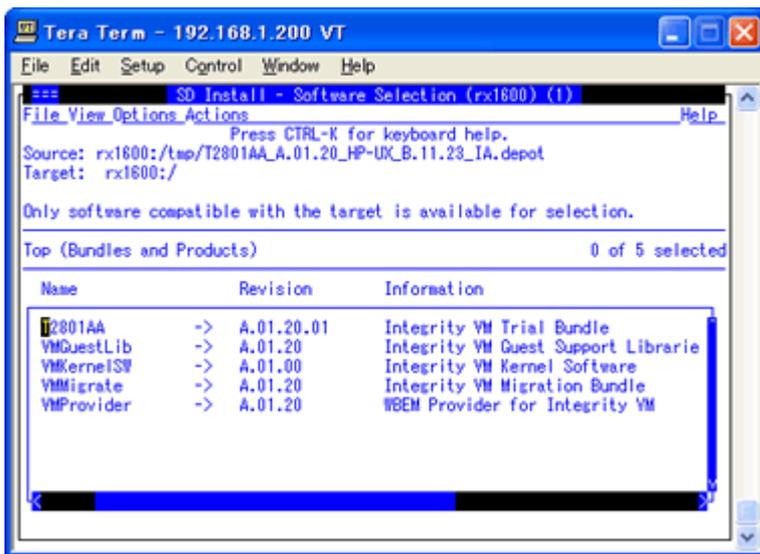


図 4 : Integrity VM のソフトウェア・バンドル内容)

このバンドルには、Integrity VM 本体のほか、WBEM (Web-Based Enterprise Management) プロトコルを通じて Integrity VM を管理するためのソフトウェアなどが含まれています。今回はこれらすべてのソフトウェア・プロダクトをインストールしますので、個々のプロダクトを選択し、メニューより「Actions」→「Mark For Install」を選択してマークを付けていきます。すべてのプロダクトにマークを付けたならば、メニューより「Actions」→「Install...」を選択してインストールを開始します。

swinstall によるインストールの解析が終了したら、「OK」を選択してインストールを開始します。インストールは 1~2 分で終了しますので、インストールが完了したら「Done」を選択します。これにより、システムは自動的にリブートします。

## インストール結果の確認

Integrity VM は以下のディレクトリ構成でインストールされます。

<表 : Integrity VM のディレクトリ構成>

パラメーター	内容
/opt/hpvm	Integrity VM のソフトウェア本体
/opt/hpvm/bin	Integrity VM のコマンド類
/var/opt/hpvm	Integrity VM のデータファイルやゲストの情報など

また Integrity VM のインストールにともない、カーネルパラメーターが以下の内容で変更されます。

<表：Integrity VM のインストールによるカーネルパラメーター変更>

パラメーター	内容	初期値	変更後
dbc_max_pct	バッファ・キャッシュに利用するメモリの割合の最大値	50%	1%
dbc_min_pct	バッファ・キャッシュに利用するメモリの割合の最小値	5%	1%
maxdsiz_64bit	データセグメントの最大サイズ	4,294,967,296 バイト	34,359,738,368 バイト
swapmem_on	物理メモリによる疑似スワップの利用	1 (オン)	0 (オフ)

では、Integrity VM が正しくインストールされているか確認するために、以下のコマンドを実行します。

```
# hpvminfo
```

このとき、以下のように表示されればインストールは成功しています。

```
hpvminfo: Running on an HPVM host.
```

また以下のコマンドを入力することで、VM ゲストの動作状況を確認できます。

```
# hpvmstatus
```

もちろん、この段階では VM ゲストをひとつも作成していないので、以下のようなメッセージが表示されます。

```
hpvmstatus: No guest information is available.
```

```
hpvmstatus: Unable to continue.
```

## 仮想スイッチの作成

Integrity VM のインストールの仕上げとして、「仮想スイッチ」を作成します。仮想スイッチについて詳しくは本連載で追って説明しますが、簡単に言えば VM ホスト内部に用意される仮想的な LAN スイッチであり、VM ゲストを物理 NIC や他の VM ゲストに接続するために用います。VM ゲストの作成時にはこの仮想スイッチ名を指定する必要があるため、先に仮想スイッチを作成しておくのが便利です。

ちなみに、Integrity VM をプロダクション環境で用いる場合には、このほかにも「VM ホストのデバイスのアクセス制限」や「VM ゲストのコンソール・アクセスの制限」といったセキュリティ設定を実施しておくことが推奨されます。これらについては Integrity VM ドキュメント第 3 章の「Post installation Procedures」を参照してください。

例えば、clan0 という名称の仮想スイッチを作成するには、hvvmnet コマンドを以下のように入力します。

```
# hvvmnet -c -S clan0 -n0
```

ここで、オプション-c は仮想スイッチの作成を表し、-S は名称の指定、そして-n0 は仮想スイッチに関連づける VM ホストの NIC の PPA (Physical Point of Attachment) 番号を表します。ちなみに lanscan コマンドを実行すると、そのマシンが備える NIC の PPA 番号を表示できます。

仮想スイッチを作成したならば、hvvmnet コマンドにオプション-b を付けて実行し、仮想スイッチを起動します。

```
# hvvmnet -S clan0 -b
```

最後に、hvvmnet コマンドにオプション-v を実行すると、作成された仮想スイッチの情報が表示されます。

```
# hvvmnet -v
Version A.01.20.00
Name      Number State   Mode      PPA      MAC Address      IP Address
=====
localnet  1      Up      Shared    N/A      N/A
clan0     3      Up      Shared    lan0     0x00306e4c1984  192.186.1.200
```

この例では、作成した仮想スイッチ clan0 が起動しており、NIC である lan0 および IP アドレス 192.168.1.200 と関連づけられていることがわかります。

以上、今回は Integrity VM のインストール手順をひとつお介绍しました。次回は、Integrity VM の VM ゲストの作成方法を説明します。

## 第 3 回

# ストレージとネットワークの仮想化

2006 年 6 月 テクニカルライター 小林聡史

Integrity VM では、個々の VM ゲストについて、仮想ストレージ・デバイスと呼ばれる仮想的な SCSI ディスクを設け、それに VM ホストの物理ディスクや論理ボリューム、ファイルを割り当て可能です。例えばファイルに対応づけることで、仮想的な Integrity サーバーが 1 台まるごと 1 つのファイルに収まり、ファイルコピーだけで「サーバーの移設」が可能となります。さらに Integrity VM では、各 VM ゲストや物理 NIC を結ぶ「仮想スイッチ」も構成できます。今回は、Integrity VM におけるストレージとネットワークの仮想化について説明します。

## hvvmcreate コマンドのオプションを知る

前回は、Integrity VM の「VM ホスト」、すなわち Integrity VM 本体のインストール手順を紹介しました。今回は、VM ホスト上で動作する仮想マシンである「VM ゲスト」の作成に必要なコマンド・オプションの指定方法を説明します。

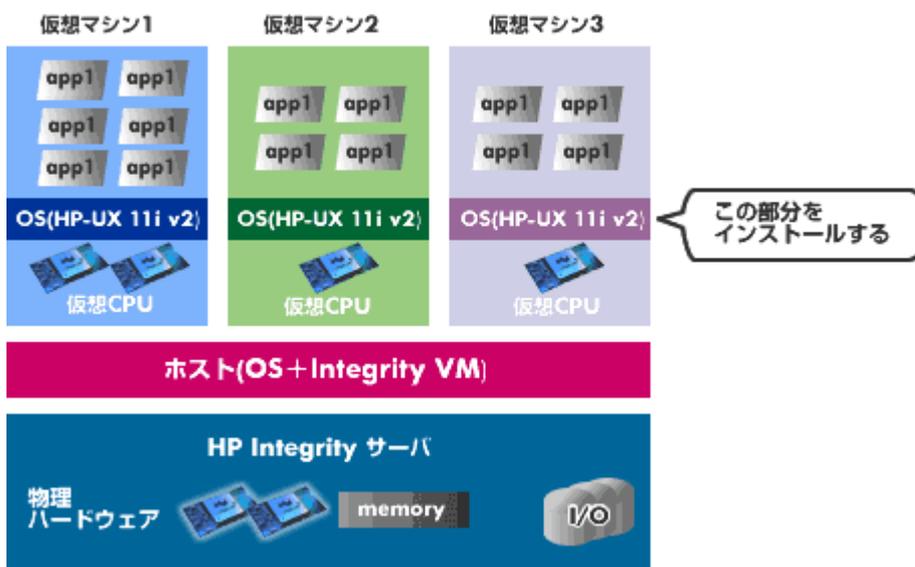


図 1 : Integrity VM の VM ホストと VM ゲスト

VM ゲストの作成には、hvvmcreate コマンドを用います。このコマンドを実行する際に、VM ゲストをどのような構成で作成するかを、以下のオプションで細かく指定します。

表 : hvvmcreate コマンドのオプション

VM ゲストの構成内容	コマンド・オプション
ゲスト名	-P <ゲスト名>
OS の種類	-O <OS タイプ>

仮想 CPU 数	-c <仮想 CPU 数>
CPU エンタイトルメント	-e <パーセント値>または -E <CPU サイクル数>
メモリサイズ	-r <メモリサイズ>
仮想ネットワーク・デバイス 仮想ストレージ・デバイス	-a <仮想デバイス指定>
スタートアップ時の動作	-B <スタートアップ時の動作指定>
管理者アカウント名	-u <ユーザーグループ> -g <グループ>

これらのオプションのうち、「仮想 CPU 数」や「CPU エンタイトルメント」、「メモリサイズ」の意味と設定方法については次回説明します。今回は、オプション-a で指定する「仮想ストレージ・デバイス」と「仮想ネットワーク・デバイス」について解説します。

## 仮想ストレージ・デバイスとは

Integrity VM では、個々の VM ゲストについて、仮想ストレージ・デバイスと呼ばれる仮想的な SCSI ディスクを割り当てることができます。この仮想ディスクは、実際には以下のいずれかの物理ストレージ・デバイスに対応付けられています。

- 物理ディスクや SAN（論理ユニット）
- LVM もしくは VxVM の論理ボリューム
- ファイル

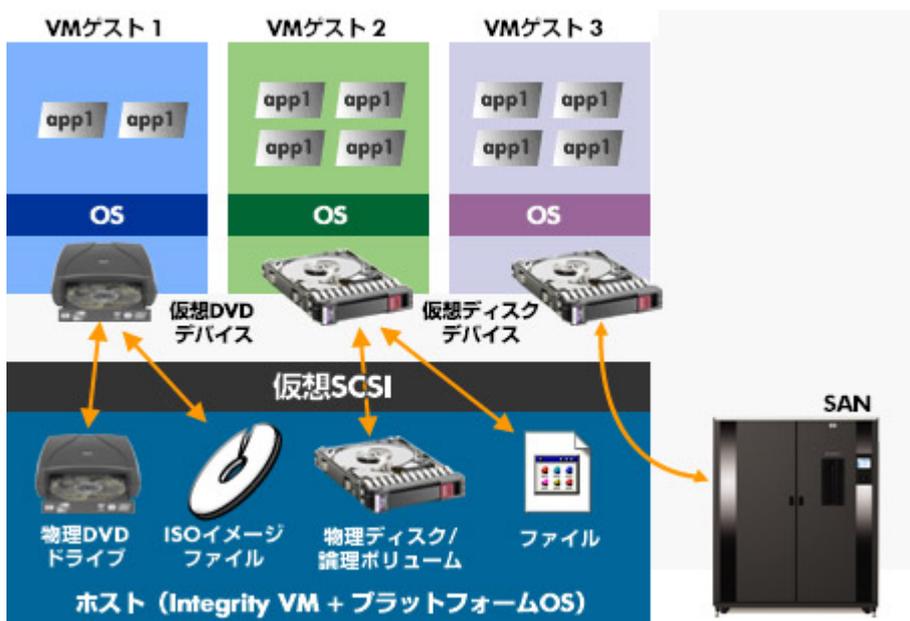


図 2 : Integrity VM の仮想ストレージ・デバイス

これらのうち、もっともパフォーマンスが高いのは、物理ディスクや SAN の論理ユニットをまるごと仮想ストレージ・デバイスに割り当てる方法です。ただしこの場合、割り当ての粒度が数 10~数 100GB と大きくなってしまいうえ、VM ゲストのコピーや移動が容易ではないというデメリットがあります。

一方、LVM もしくは VxVM の論理ボリュームや、ファイルシステム上のひとつのファイルを仮想ストレージ・デバイスに割り当てることも可能です。この場合、パフォーマンスのオーバーヘッドが上昇するものの、割り当ての粒度を細かくでき、VM ゲストのポータビリティも向上します。例えばファイルに割り当てた場合は、仮想的な Integrity サーバーが 1 台まるごと 1 つの巨大ファイルに収まるかたちとなります。つまり、このファイルをコピーするだけで「サーバーの移設」が可能です。

また Integrity VM では、ディスクの他に DVD ドライブの仮想化にも対応しています。「仮想 DVD」に対して物理 DVD を対応付けられるほか、ISO ディスク・イメージを記録したファイルに対応付けることもできます。したがって、例えばアプリケーション・インストール用 DVD を ISO イメージ・ファイルとして保管しておけば、同ファイルを仮想 DVD としてマウントし、アプリケーションのインストール作業を行うことができます。何枚もの DVD メディアを手作業で交換するような手間もかかりません。

ちなみに、2006 年末には Integrity VM において「Direct I/O」と呼ばれるモードがサポートされる予定です。Direct I/O では、物理ストレージ・デバイスを特定の VM ゲストに直結することで、よりオーバーヘッドの少ない I/O パフォーマンスが期待されます。

## -a オプションによるストレージ設定

さて、VM ゲストの仮想ストレージ・デバイスを設定するには、hpvmcreate コマンドのオプション-a に続いて以下の書式で記述します。

```
<デバイス・タイプ>:<アダプタ・タイプ>:[ハードウェア・アドレス:]<ストレージ・タイプ>:デバイス
```

ここで、それぞれの項目には以下の内容を記述します。

### 【仮想ストレージ・デバイスの指定】

- **デバイス・タイプ**—デバイスのタイプとして、ディスクの場合は「disk」、DVD ドライブの場合は「dvd」と記述します。
- **アダプタ・タイプ**—ホストバス・アダプタのタイプとして「scsi」と記述します。
- **ハードウェア・アドレス (任意指定)**—デバイスのハードウェア・アドレスを以下の書式で指定できます。ただし通常は自動的に割り当てられるため、ハードウェア・アドレスを記述する必要はありません。

```
<PCI バス番号>,<PCI スロット番号>,<SCSI ターゲット番号>
```

### 【物理ストレージ・デバイスの指定】

- **ストレージ・タイプおよびデバイス**—仮想ストレージ・デバイスに割り当てる物理ストレージ・デバイスを以下のように指定します。

表：物理ストレージ・デバイスの指定

ストレージ・タイプ	ストレージの種類	デバイス指定例
disk	ディスクもしくは DVD (raw デバイス)	/dev/rdisk/c4t3d2
lv	LVM もしくは VxVM の論理ボリューム (raw デバイス)	/dev/vg01/rlv012
file	ローカル・ディスク上のファイル (HFS ファイルは利用不可)	/guestfiles/diskfile
null	DVD メディアイメージの ISO ファイル (保管先ディレクトリを指定)	/docs

例えば、LVM のあるボリューム全体を VM ゲストに割り当てるには、以下のように指定します。

```
-a disk:scsi::lv:/dev/vg01/rlv022
```

この場合、仮想ストレージ・デバイスとしては「SCSI ディスク」を指定し、それに割り当てる物理ストレージ・デバイスとしては LVM の論理ボリューム「/dev/vg01/rlv022」を指定しています。つまり VM ゲストからは、この論理ボリュームの領域全体が 1 台の仮想的な SCSI ディスクとして認識されるという仕組みです。

また、以下の例では、仮想 DVD ドライブを設定しています。

```
-a dvd:scsi:0,0,1:null:/docs
```

ここでは、物理ストレージ・デバイスのタイプとして「null」を指定することで、「/docs」ディレクトリに格納された DVD メディアの ISO イメージ・ファイルを仮想 DVD ドライブとしてマウントしています。

後半では、仮想ネットワーク・デバイスの指定方法を説明します。

## 仮想ネットワーク・デバイスとは

Integrity VM では、ストレージと同様に、ネットワークの仮想化もサポートしています。以下は、そのメカニズムを示した図です。

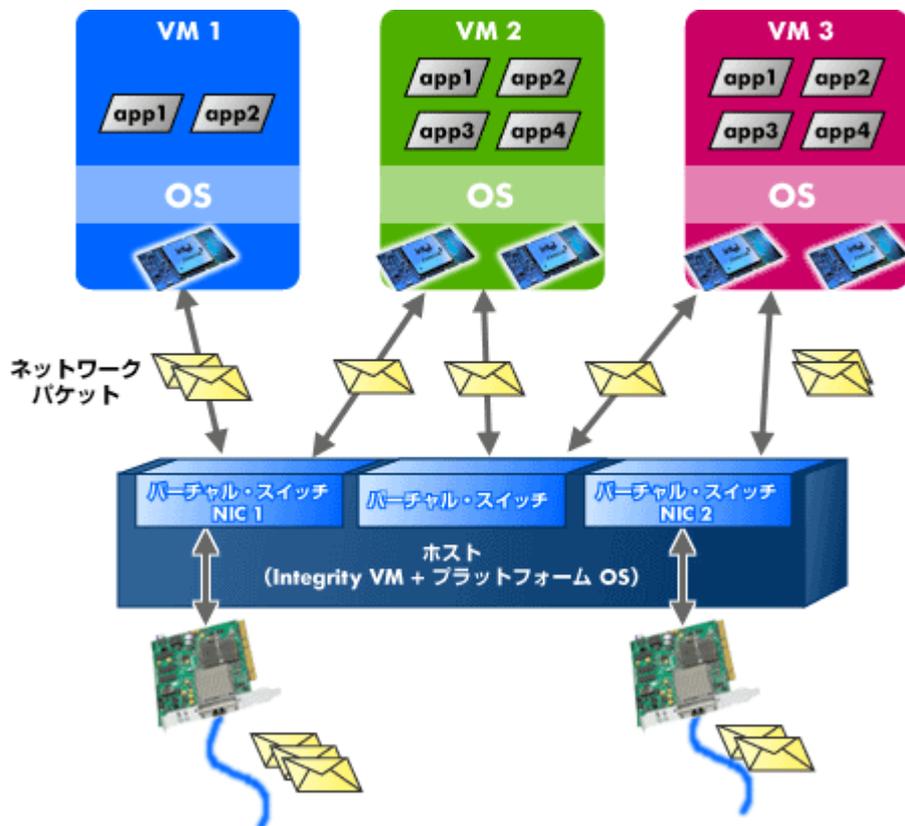


図 3 : Integrity VM の仮想ネットワーク・デバイス

図 3 において、VM ホスト上にバーチャル・スイッチ（仮想スイッチ）が形成されている部分に注目してください。Integrity VM では、仮想スイッチを作成することで、各 VM ゲストや物理 NIC を結ぶ仮想的なネットワークセグメントを構成できます。よって VM ゲスト間のネットワーク通信は、物理 NIC を介さずにホスト OS の中だけで実現可能です。

例えばアプリケーション開発においてテスト環境を構築するようなケースでは、仮想のクライアントやサーバーとなる VM ゲストを作成し、それらを仮想スイッチで結ぶだけで環境設定は完了します。すべてがコマンド操作だけで完結するため、物理的なサーバーを並べ、それらをネットワークハブに接続し……といった手間のかかる作業は不要となります。また、物理ネットワークとは切り離れた別セグメントを構成することで、セキュリティ面の隔離性も得られます。

ただし、現状の Integrity VM における仮想スイッチは一定のオーバーヘッドをとるため、ネットワーク I/O 負荷の大きいアプリケーションでの利用には注意が必要です。オーバーヘッドを下げるため、物理 NIC を特定の VM のみに割り当てることも可能です。

### -a オプションによるネットワーク設定

VM ゲストの仮想ネットワーク・デバイスを設定するには、`hvvmcreate` コマンドのオプション-a に続いて以下の書式で記述します。

```
network:lan:[<ハードウェア・アドレス>:]vswitch:<仮想スイッチ名>
```

ここで、それぞれの項目には以下の内容を記述します。

・**ハードウェア・アドレス (任意指定)** — 仮想ネットワーク・デバイスのハードウェア・アドレスを以下の書式で指定できます。ただし通常は自動的に割り当てられるため、ハードウェア・アドレスを記述する必要はありません。

```
<PCIバス番号>,<PCIスロット番号>,<MACアドレス>
```

・**仮想スイッチ名** — 仮想ネットワーク・デバイスを接続する仮想スイッチの名称を指定します。

なお hpvmcreate コマンドにて仮想ネットワーク・デバイスを指定する際には、あらかじめ hpvmnet コマンドにより仮想スイッチを作成しておく必要があります。同コマンドの利用方法は、前回説明したとおりです。

例えば、仮想スイッチ clan0 に接続する仮想ネットワーク・デバイスを指定する場合は、オプション-a を以下のように記述します。

```
-a network:lan:vswitch:clan0
```

以上、今回は VM ゲストを作成する hpvmcreate コマンドのオプション-a を中心に説明しました。次回は、同コマンドを実際に実行して VM ゲストを作成し、起動する手順を紹介する予定です。

## 第 4 回

# VM ゲストの作成

2006 年 7 月 テクニカルライター 小林聡史

前回は、VM ゲストを作成する `hpvmcreate` コマンドに指定するオプションのうち、仮想ストレージ・デバイスと仮想ネットワーク・デバイスについて説明しました。今回は、同コマンドにより VM ゲストを実際に作成する手順を紹介します。作成したばかりの VM ゲストには OS がインストールされていないため、VM ゲストの初回起動時にはインストール・メディアを指定する必要があります。ここでは、Ignite-UX サーバーからのインストール方法と、仮想 DVD ドライブからのインストール方法の両方を説明します。また、VM ゲストを起動する 2 種類のコマンド、`hpvmstart` コマンドと `hpvmconsole` コマンドの使い方を解説します。

## VM ゲストの作成

前回は、VM ゲストを作成する `hpvmcreate` コマンドに指定するオプションのうち、仮想ストレージ・デバイスと仮想ネットワーク・デバイスについて説明しました。今回は、同コマンドにより VM ゲストを実際に作成する手順を紹介します。

表 : `hpvmcreate` コマンドのオプション

VM ゲストの構成内容	コマンド・オプション
ゲスト名	-P <ゲスト名>
OS の種類	-O <OS タイプ>
仮想 CPU 数	-c <仮想 CPU 数>
CPU エンタイトルメント	-e <パーセント値>または -E <CPU サイクル数>
メモリサイズ	-r <メモリサイズ>
仮想ネットワーク・デバイス 仮想ストレージ・デバイス	-a <仮想デバイス指定>
スタートアップ時の動作	-B <スタートアップ時の動作指定>
管理者アカウント名	-u <ユーザーグループ> -g <グループ>

例えば、サンプルの VM ゲスト `compass1` を作成するには、`hpvmcreate` コマンドを以下のように実行します。

```
# hpvmcreate -Pcompass1 -c1 -r1G \
-a network:lan::vswitch:clan0\
-a disk:scsi::lv:/dev/vg01/rlv022\
-a dvd:scsi:0,0,1:null:/dev
```

この例で指定している各オプションの意味を説明します。

- 「-Pcompass1」——VM ゲスト名を compass1 とする。
- 「-c1」——仮想 CPU を 1 個割り当てる。
- 「-r1G」——メモリを 1GB 割り当てる。
- 「-a network:lan::vswitch:clan0」——仮想スイッチ clan0 に接続する仮想ネットワーク・デバイスを設定する。
- 「-a disk:scsi::lv:/dev/vg01/rlv022」——論理ボリューム rlv022 を仮想ストレージ・デバイスとして割り当てる。
- 「-a dvd:scsi:0,0,1:null:/dev」——「/dev」ディレクトリ内の ISO イメージ・ファイルを仮想 DVD ドライブとしてマウントする。

つづいて、VM ゲストの作成例をもうひとつ見てみます。

```
# hpvmcreate -Pcompass2 -c2 -r3G -e50 \
-a disk:scsi:1,0,2:disk:/dev/vg01/rlv023
```

この VM ゲスト compass2 の例では、仮想 CPU を 2 個、メモリを 3GB、そしてエンタイトルメント（詳しくは次回以降で説明）として 50%を割り当てています。また仮想ストレージ・デバイスとしては、論理ボリューム/dev/vg01/rlv023 を設定します。仮想ネットワーク・デバイスは割り当てていないため、ローカル・ネットワーク経由でのみアクセス可能な VM ゲストとなります。

次に、以下の作成例を見てください。

```
# hpvmcreate -Pcompass3 -c1 -r2G \
-a disk:scsi::disk:/dev/vg01/rlv024 \
-a network:lan::vswitch:clan1
HPVM guest compass3 configuration problems:
Warning 1: Insufficient free memory for guest.
These problems may prevent HPVM guest compass3 from booting.
hpvmcreate: The creation process is continuing.
```

この例では、VM ホストのメモリが不足しており、VM ゲスト compass3 に指定したメモリが割り当てられなかったことを示すワーニングが発生しています。この場合、VM ゲストは作成されるものの、このままの状態では VM ゲストを起動することができません。よって、VM ゲストの構成を変更する hpvmmodify コマンドを用いてメモリサイズを少なくするか、他の VM ゲストを停止するなどの対処が必要です。

以上の手順で VM ゲストを作成したならば、hpvmstatus コマンドを実行して各 VM ゲストの状態を確認できます。

```
# hpvmstatus
[Virtual Machines]
Virtual Machine Name VM # OS Type State # vCPUs # Devs # Nets Memory
-----
compass1 15 HPUX Off 1 2 2 1 GB
compass2 16 HPUX Off 2 1 0 3 GB
compass3 17 HPUX Off 1 1 1 2 GB
```

図 1 : hpvmstatus コマンドの実行例

## VM ゲストを起動する

VM ゲストを起動するには、次に挙げる 2 つのコマンドのいずれかを使用します。

- hpvmstart コマンド
- hpvmconsole コマンド

作成したばかりの VM ゲストには OS がインストールされていません。よって VM ゲストの初回起動時には、インストール・メディアを指定する必要があります。このとき、Ignite-UX を用いてネットワーク上のサーバーから OS イメージの供給を受ける場合は、通常の Integrity サーバーのインストール時と同じ操作を行います。つまり、VM ゲストのホスト名、MAC アドレス (hpvmstatus コマンドで表示されるもの)、そして IP アドレスを Ignite-UX サーバーに設定します。また VM ゲストのコンソールからは、LAN ブートデバイスを設定します。

一方、DVD メディアからインストールを行う場合は、VM ゲストに仮想 DVD ドライブを以下の手順で割り当てておく必要があります。

1. VM ホスト上で ioscan コマンドを実行し、VM ホストの物理 DVD ドライブのキャラクタ・デバイス・パスを確認します。

```
# ioscan -funC disk
```

2. VM ゲストに設定する仮想 DVD ドライブとして、このキャラクタ・デバイス・パスを指定します。

```
# hpvmmodify -P compass1 -a dvd:scsi::disk:/dev/rdisk/c0t0d0
```

## hpvmstart コマンドによる起動

これにより、DVD メディアから VM ゲストのインストールを行う準備が整いました。hpvmstart コマンドを以下のように実行し、VM ゲストを起動します。

```
# hpvmstart -Pcompass1
(C) Copyright 2000 - 2005 Hewlett-Packard Development Company, L.P.
Initializing System Event Log
Initializing Forward Progress Log
Opening minor device and creating guest machine container
Creation of VM, minor device 1
Allocating guest memory: 64MB
    allocating low RAM (0-4000000, 64MB)
```

```

/opt/hpvm/sbin/hpvmapp (/var/opt/hpvm/uuids/ce17ee10-3131-11da-9845-00306e39f70b
/vmm_config.current): Allocated 67108864 bytes at 00
allocating firmware RAM (ffaa0000-ffab5000, 84KB)
/opt/hpvm/sbin/hpvmapp (/var/opt/hpvm/uuids/ce17ee10-3131-11da-9845-00306e39f70b
/vmm_config.current): Allocated 86016 bytes at 0x600
Loading boot image
Image initial IP=102000 GP=5F4000
Initialize guest memory mapping tables
Starting event polling thread
Starting thread initialization
Daemonizing...
hpvmstart: Successful start initiation of guest 'compass1'

```

VM ゲストが起動したら、hpvmstatus コマンドを実行し、動作状態を確認します。

```

# hpvmstatus
[Virtual Machines]
Virtual Machine Name VM # OS Type State # vCPUs # Devs # Nets Memory
-----
compass1 15 HPUX On 1 2 2 1 GB
compass2 16 HPUX Off 2 1 0 3 GB
compass3 17 HPUX Off 1 1 1 2 GB

```

図 2 : hpvmstatus コマンドによる動作状態の確認

## hpvmconsole コマンドによる起動

hpvmconsole コマンドを利用して、VM ゲストのコンソールから起動することも可能です。この場合、以下のようにコマンドを入力します。

```
# hpvmconsole -c 'pc -on' -P compass1
```

これにより、ちょうど Integrity サーバーをブートした時と同様に、EFI Boot Manager が立ち上がります。

```
EFI Boot Manager ver 1.10 [14.62]
```

```
Please select a boot option
```

```

Acpi(PNP0A03,0)/Pci(3|1)/Ata(Primary,Slave)
HP-UX Primary Boot: 0/16/1/3/1.0.0.0
EFI Shell [Built-in]
Boot option maintenance menu
Use ^ and v to change options. Press Enter to select an option.

```

ここで「Boot option maintenance menu」を選択し、「Add a Boot Option」を選択します。

```
EFI Boot Maintenance Manager ver 1.10 [14.62]
```

Add a Boot Option. Select a Volume

```
Removable Media Boot [Acpi(PNP0604,0)] *1
Load File [Acpi(PNP0A03,0)/Pci(1|0)/Mac(763AE48F393F)] *2
Load File [EFI Shell [Built-in]]
Legacy Boot
Exit
```

ここで仮想 DVD からインストールする場合は「Removable Media Boot」\*1 を選び、Ignite-UX サーバーからインストールする場合は MAC アドレスが記載されている「Load File」\*2 を選びます。この一連の操作により、インストールが開始します。インストール手順が完了すると、OS が自動的にリブートします。

以上、今回は VM ゲストの作成方法と OS インストールの手順を紹介しました。

---

## 第 5 回

# VM ゲストへのリソース配分

2006 年 8 月 テクニカルライター 小林聡史

今回は、Integrity VM における「仮想 CPU」へのリソース配分方法を説明します。仮想 CPU とは、「それぞれの VM が利用可能な最大 CPU 数」を表します。この CPU 数の合計が物理 CPU の数を上回る場合、すべての仮想 CPU に 100% の処理能力を配分することはできません。そこで「エンタイトルメント」と呼ばれる設定を行い、高負荷時の CPU リソース配分をあらかじめ決めておきます。こうした Integrity VM における CPU リソースのきめ細かな配分は、HPE のリソース・パーティション技術 PRM(Process Resource Manager)で利用されている FSS(Fare Share Scheduler)によって実装されており、CPU 利用率を%単位で厳密に履行できるのが特徴です。

### 「仮想 CPU」とは

前回は、hpvmcreate コマンドを用いて VM ゲストを作成し、hpvmstart コマンドおよび hpvmconsole コマンドにより VM ゲストに OS をインストールする手順を紹介しました。今回は、VM ゲストへの CPU リソースの配分方法や、VM ゲスト作成時のトラブルシューティング方法を説明します。

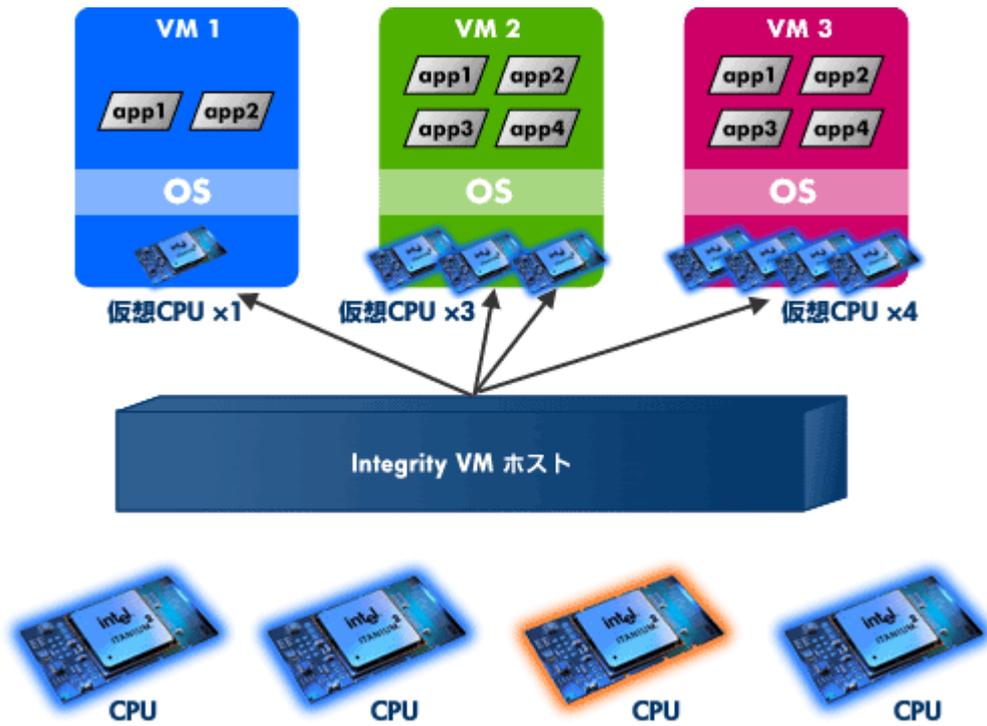


図 1 : Integrity VM における CPU リソースの仮想化

図 1 は、Integrity VM における CPU リソースの仮想化を表した図です。この例では、4 個の CPU を搭載した Integrity サーバー上で、3 つの VM を運用しています。ここで、それぞれの VM に配分されている「仮想 CPU」に注目してください。仮想 CPU とは、「それぞれの VM が利用可能な最大 CPU 数」を表します。よって図 1 の例では、左端の VM1 は 1 個分、VM2 は 3 個分、そして VM3 は 4 個分までの CPU 処理能力を受け取ることができます。VM 内部で並列処理可能なスレッド数も、仮想 CPU 数に比例します。

ちなみに 1 つの VM には、サーバーの物理 CPU 数を超える仮想 CPU 数を割り当てることはできません。そのため、もし 2-way サーバーで Integrity VM を利用する場合は、各 VM に配分できる仮想 CPU 数は最大 2 個までとなります。また、多数の CPU を搭載するサーバーであっても、仮想 CPU 数の上限は、現在のところ 4 個までです。

### 仮想 CPU の「エンタイルメント」

さて図 1 の例では、4 個の物理 CPU に対し、合計 8 個の仮想 CPU を各 VM に割り当てているので、当然のことながらすべての VM の仮想 CPU に 100%の処理能力を配分することは不可能です。そこで Integrity VM では、「エンタイルメント (entitlement)」と呼ばれる設定を行うことで、高負荷時の CPU リソース配分をあらかじめ決めておくことができます(図 2-上)。また、VM3 がアイドル状態で負荷がない場合は、その分リソースを他の VM が使うことができます(図 2-下)。

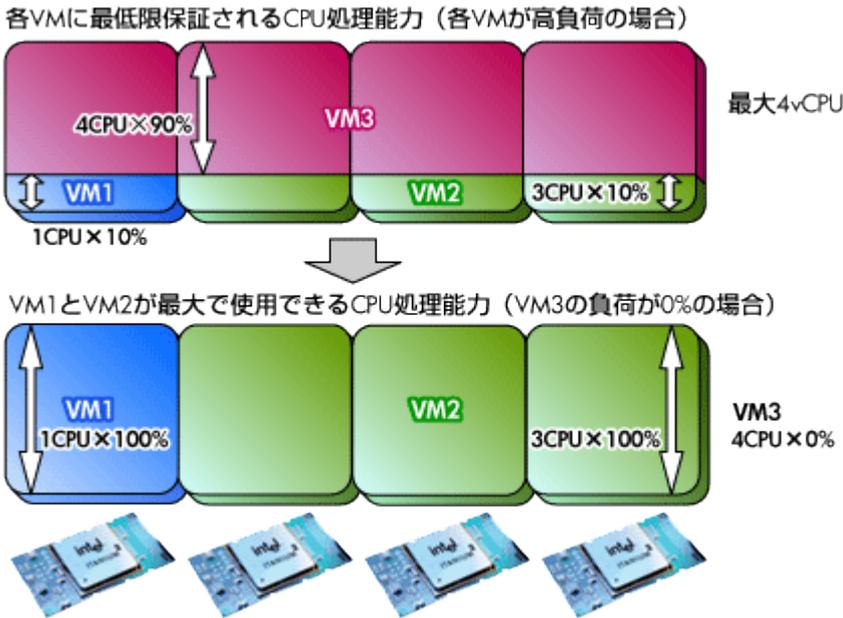


図 2 : Integrity VM のエンタイトルメントの設定と CPU 使用率

図 2 の例では、エンタイトルメントが以下のように設定されています。

表 : エンタイトルメントによって配分される CPU 処理能力

VM ゲスト	エンタイトルメント	最小	最大
VM1	仮想 CPU 1 個 x 10%	10%	100%
VM2	仮想 CPU 3 個 x 10%	30%	300%
VM3	仮想 CPU 4 個 x 90%	360%	400%

エンタイトルメントは、「各 VM に最低限保証される CPU 処理能力」を表します。つまり、すべての VM が高負荷状態にある場合でも、エンタイトルメントで設定した CPU 使用率はかならず確保されます。例えば上記例では、たとえ VM2 や VM3 の内部で高負荷のアプリケーションが動作していても、VM1 はつねに 10%の CPU リソースを利用できます。また VM2 や VM3 の負荷が低い場合は、その剰余分を利用して、最大 100%(仮想 CPU 1 個分)の処理能力が VM1 に動的に配分される仕組みです。ちなみに、エンタイトルメントの設定の最小値は 5%であるため、CPU 1 個あたり最大 20 の VM を作成することができます。こうした Integrity VM における CPU リソースのきめ細かな配分は、HPE が 1997 年から提供しているリソース・パーティション技術である PRM(Process Resource Manager)の FSS(Fair Share Scheduler)によって実現されています。FSS は、HP-UX カーネルに組み込まれた専用のスケジューラーであり、個々のプロセスの CPU 利用率を%単位で厳密に履行します。こうした実績豊富なテクノロジーによって実装されている点も、Integrity VM の優位性のひとつと言えます。

後半では、VM ゲスト作成時のトラブルシューティング方法を説明します。

## VM ゲスト作成時のトラブルシューティング

VM ゲストの起動時、VM ホストはリソース配分を動的に計算し、VM ゲストの実行に十分なリソースが存在するか確認します。もし問題があれば、以下の警告メッセージのいずれか 1 つ以上が表示されます。

```
HPVM guest badguest configuration problems:
```

```
Warning 1: Guest's vcpus exceeds server's physical cpus.
Warning 2: Guest's vcpus exceeds the supported maximum, 4 vcpus.
Warning 3: Insufficient free memory for guest.
Warning 4: Insufficient swap resource for guest.
Warning 5: Insufficient cpu resource for guest.
Warning 6 on item /dev/rdisk/c2t1d0: Device file '/dev/rdisk/c2t1d0' in use by another guest.
Warning 7 on item /dev/vg00/rhostswap: Device file '/dev/vg00/rhostswap' in use by server.
Warning 8 on item /dev/rdisk/foo: '/dev/rdisk/foo' backing device does not exist.
Warning 9 on item hostnet: Guest MAC address for switch hostnet is in use.
Warning 10 on item offnet: Vswitch offnet is not active.
Warning 11 on item badnet: Vswitch badnet does not exist.
```

```
These problems may prevent HPVM guest badguest from booting.
```

```
hpvmstart: Unable to continue.
```

これらの警告メッセージが表示された場合、VM ゲストの作成は可能ですが、起動は行えません。また、以下のようなエラーメッセージが表示された場合は、VM ゲストの作成と起動の両方が行えません。

```
hpvmcreate: ERROR (badguest): Duplicate backing device '/dev/rdisk/c2t1d0'.
hpvmcreate: ERROR (badguest): Illegal blk device '/dev/dsk/c2t1d0s5' as backing device
```

これらのメッセージが表示された場合は、hpvmmodify コマンドを実行して VM ゲストの構成を必要に応じて変更するか、ほかの VM ゲストを停止するなどしてリソース不足を解消します。

ちなみに、VM ホストに対して実行した Integrity VM 関連コマンドの履歴は、以下のファイルに記録されているので、参考になるでしょう。

```
/var/opt/hpvm/common/command.log
```

## その他のトラブルシューティング

VM ゲストがハングアップした場合(ping コマンドに反応しないときや、hpvmconsole コマンドがフリーズしたときなど)は、hpvmconsole コマンドのセッションで「**^B**(Ctrl+B)」を入力します。これにより、セッションが VM コンソールに戻るため、CM コマンド、TC コマンドをそれぞれ実行し、VM ゲストの OS に対して INIT コマンドを送ります。VM ゲスト自体のリブートは行いません。

ちなみに、VM ホストで hpvmcollect コマンドを実行することで、トラブルシューティングに役立つ Integrity VM の情報を収集できます。同コマンドを実行すると、収集された情報(tar ファイル)の記録場所が以下のように末尾に表示されます。

```
# hpvmcollect -Pcompass1
```

```
HPVM host crash/log collection tool version 0.8
Gathering info for post-mortem analysis of guest 'compass1' on host
```

```
Collecting I/O configuration info ..... OK
Collecting filesystem info ..... OK
Collecting system info ..... OK
Collecting lan info ..... OK
```

<中略>

```
The collection is "/var/opt/hpvm/common/hpvmcollect_archive/compass1_Oct.04.05_165043EDT.tar.gz"
```

この tar ファイルには、例えば仮想ストレージ・デバイスとして指定したディレクトリに十分な空きスペースがない場合、以下のようなエラー・メッセージが記録されます。

```
msgcnt 10 vxfs: mesg 001: vx_nospace - /dev/vg00/lvol5 file system full(1 block extent)
```

以上、今回は VM ゲストへのリソース配分の方法と、トラブルシューティングの方法について説明しました。

---

## 第 6 回

# Integrity VM の構成管理

2006 年 9 月 テクニカルライター 小林聡史

VM ゲストを作成したものの、サーバーのリソース不足などにより、そのままの構成では起動できないケースがあります。こうした場合は、VM ゲストの構成を変更し、リソース不足を解消する必要があります。また Integrity VM では、VM ゲストの仮想 NIC を接続する「仮想スイッチ」を作成できます。仮想スイッチとは、VM ホスト内部に用意される仮想的な LAN スイッチであり、VM ゲスト間の通信や、VM ゲストと物理ネットワークとの通信を媒介します。今回は、VM ゲストや仮想スイッチの構成管理の方法を紹介します。

## VM ゲストの構成変更

前回までの内容で、VM ゲストの作成や起動の方法についてひととおり説明しました。今回は、作成後の VM ゲストの管理方法を紹介します。

VM ゲストを作成したものの、サーバーのリソース不足などにより、そのままの構成では起動できないケースがあります。例えば、以下の例を見てください。

```
# hpvmstart -Pcompass1
HPVM guest compass1 configuration problems:
Warning 1: Insufficient free memory for guest.
Warning 2: Insufficient cpu resource for guest.
These problems may prevent HPVM guest compass1 from booting.
hpvmstart: Unable to continue.
```

ここでは、「compass1」という VM ゲストの起動に失敗し、その理由として「メモリ不足」および「CPU リソース不足」が示されています。そこで、compass1 に割り当てる仮想 CPU とメモリの量を変更するには、hpvmmodify コマンドを以下のように入力します。

```
# hpvmmodify -Pcompass1 -c1 -r1GB
```

ここでは、仮想 CPU 数を 1 個、およびメモリサイズを 1GB にそれぞれ設定しなおしています。VM ゲストの構成を変更したならば、再度 hpvmstart コマンドを実行してみます。以下のように表示されれば、今度はリソース不足が解消し、問題なく起動したことがわかります。

```
# hpvmstart -Pcompass1
(C) Copyright 2000 - 2005 Hewlett-Packard Development Company, L.P.
Initializing System Event Log
Initializing Forward Progress Log
Opening minor device and creating guest machine container
Creation of VM, minor device 1 Allocating guest memory: 1024MB
  allocating low RAM (0-40000000, 1024MB)
/opt/hpvm/lbin/hpvmapp (/var/opt/hpvm/uuids/17e4af4c-34fc-11da-94e3-00306e39f700
  allocating firmware RAM (ffaa0000-ffab5000, 84KB)
/opt/hpvm/lbin/hpvmapp (/var/opt/hpvm/uuids/17e4af4c-34fc-11da-94e3-00306e39f700
Loading boot image
Image initial IP=102000 GP=5F4000
Initialize guest memory mapping tables
Starting event polling thread
Starting thread initialization
Daemonizing...
hpvmstart: Successful start initiation of guest 'compass1'
```

また、hpvmstatus コマンドを実行し、VM ゲストの稼働状況を確認してみます。

```
# hpvmstatus
[Virtual Machines]
Virtual Machine Name VM # OS Type State # vCPUs # Devs # Nets Memory
-----
compass1 15 HPUX Off 1 0 1 1 GB
compass2 16 HPUX Off 2 1 0 5 GB
compass3 17 HPUX On 1 1 0 1 GB
#
```

図 1 : hpvmstatus コマンドによる稼働状態の確認

なお、VM ゲストの構成は、`/var/opt/hpvm/guests/<VM ゲスト名>`ディレクトリ以下に構成ファイルとして保存されます。このディレクトリには、以下の 3 種類の構成ファイルが作成されます。

- `vmm_config.current`——現在の VM ゲスト構成を含みます
- `vmm_config.prev`——変更前の VM ゲスト構成を含みます
- `vmm_config.next`——VM ゲストの起動後に変更された VM ゲスト構成を含みます。  
この構成を反映させるには、VM ゲストをリブートします

また、`hpvmstatus` コマンドにオプション `-V` を付加することで、VM ゲストの詳細な構成情報を表示させることが可能です。

表 : `hpvmstatus -V` コマンドのおもな表示内容

項目名	内容	表示例
Virtual Machine Name	仮想マシン名	compass1
Virtual Machine UUID	仮想マシンのユニーク ID	17e4af4c-34fc-11da-94e3-00306e39f70b
Virtual Machine ID	仮想マシン ID	15
VM's Model Name	仮想マシンのモデル名	server Integrity Virtual Machine
VM's Serial Number	仮想マシンのシリアル番号	VM00540000
VM's Version Number	仮想マシンのバージョン番号	0.16.0
VM's Version Label	仮想マシンのバージョンラベル	HPVM V0.16.0 clearcase opt Thu Sep 29 2005 05h12m13s T
Operating System	OS	HPUX
State	稼働状態	On
Boot type	起動方法	Manual
Console type	コンソール種別	vt100-plus
EFI location	EFI 格納場所	/opt/hpvm/guest-images/common/efi
Pattern File location	パターンファイル格納場所	/opt/hpvm/guest- images/common/patterns.vmmpat

## &lt;仮想 CPU 情報&gt;

Number Virtual CPUs	仮想 CPU 数	1
Minimum Virtual CPUs	仮想 CPU の最小数	1
Maximum Virtual CPUs	仮想 CPU の最大数	32
Percent Entitlement	エンタイルメント値	5.0%
Maximum Entitlement	エンタイルメントの最大値	100.0%

## &lt;メモリ情報&gt;

Total memory	メモリ合計	1 GB
Minimum memory limit	メモリの最小サイズ	32 MB
Maximum memory limit	メモリの最大サイズ	128 GB
Reserved memory	予約済みメモリ	64 MB
Minimum reserved limit	予約済みメモリの最小サイズ	32 MB
Maximum reserved limit	予約済みメモリの最大サイズ	128 GB
VHPT Size	VHPT サイズ	1 MB

## &lt;ネットワーク・インターフェイス情報&gt;

Interface	仮想 NIC 名	vswitch
Guest Adaptor type	アダプタ種別	lan
Backing	仮想スイッチ名	clan1
Bus	バス番号	0
Device	デバイス番号	0
Function	機能番号	0
Mac Address	MAC アドレス	12-40-62-b4-99-61

### <その他のインターフェイス情報>

Guest Device type	デバイス種別	serial
Guest Adaptor type	アダプタ種別	com1
Interface	インターフェイス	tty
Physical Device	物理デバイス	console

## 仮想スイッチの構成

本連載の第 2 回目でも説明したとおり、Integrity VM では、VM ゲストの仮想 NIC を接続する「仮想スイッチ」を作成できます。仮想スイッチとは、VM ホスト内部に用意される仮想的な LAN スイッチであり、VM ゲスト間の通信や、VM ゲストと物理ネットワークとの通信を媒介します。

現在 VM ホストに定義されている仮想スイッチの一覧を表示するには、hvvmnet コマンドを次のように入力します。

```
# hvvmnet
Name      Number State  Mode      PPA      MAC Address  IP Address
-----
localnet  1 Up     Shared   N/A      N/A
lan0      2 Up     Shared   N/A      N/A
compnet   4 Down   Shared   lan0     16.116.14.205
clan1     5 Up     Shared   lan1     0x00306e3977ab
clan2     6 Down   Shared   lan2
```

図 2 : hvvmnet コマンドの実行例

一方、個々の VM ゲストの構成ファイル (vmm\_config.current) には、VM ゲストがどの仮想スイッチに接続されているかを示す、以下のようなエントリが含まれています。

```
# Virtual Network Devices
#
lan(0,0).0x00306E39F70B = switch(clan1)
```

このエントリは、次のような意味を表します。

表：仮想ネットワーク・デバイスのエントリ内容

文字列	内容
lan(0,0)	仮想 NIC のハードウェアパス (バス番号 0、デバイス番号 0)
0x00306E39F70B	仮想 NIC の MAC アドレス
switch(clan1)	仮想 NIC の接続先の仮想スイッチ (clan1)

一方、この VM ゲストの内部で lanscan コマンドを実行すると、以下のような結果が表示されます。

```
compass1# lanscan
Hardware Station      Crd Hdw  Net-Interface  NM  MAC      HP-DLPI DLPI
Path   Address          In# State NamePPA        ID  Type     Support Mjr#
0/0/3/0 0x00306E39F70B 0   UP   lan0 snap0      1  ETHER   Yes    119
0/1/2/0 0x00306E3977AB 1   UP   lan1 snap1      2  ETHER   Yes    119
0/4/1/0 0x00306E4CE96E 2   UP   lan2 snap2      3  ETHER   Yes    119
```

図 3：lanscan コマンドの実行結果

ここで、VM ゲストの仮想 NIC である lan0 のハードウェアパスと MAC アドレスが、上述のエントリ内容と一致している点に注目してください。この仮想 NIC が仮想スイッチ clan1 に接続されていることが分かります。

ちなみに、仮想スイッチが接続する物理 NIC を変更したい場合は、hvvmnet コマンドを用いてその仮想スイッチを削除します。つづいて、同じ名称で仮想スイッチを再作成し、その際に変更先の物理 NIC を指定します。例えば、以下のような具合です。

```
# hvvmnet -c -S clan0 -n0
```

ここで、オプション-c は仮想スイッチの作成を表し、-S は名称の指定、そして-n0 は仮想スイッチに関連づける物理 NIC の PPA (Physical Point of Attachment) 番号を表します。

以上、今回は VM ゲストおよび仮想スイッチの構成の管理方法を説明しました。

## 第 7 回

# Virtualization Manager と Capacity Advisor による 統合管理

2006 年 10 月 テクニカルライター 小林聡史

HPE では、Integrity VM の VM ゲストを管理するためのツール「VM Manager」、および仮想化環境のためのシミュレーション・ツール「Capacity Advisor」を提供しています。VM Manager は、各 VM の CPU、メモリ、ディスク、ネットワークなどのリソース使用率やエンタイトルメントを、GUI 画面上で分かりやすく表示するツールです。一方、Capacity Advisor は、「サーバー仮想化によって CPU リソースの利用率はどの程度まで高められるか?」「サーバー仮想化によって、既存のサーバー上に追加のアプリケーションを収容できるか?」といった問いに答えるためのツールです。いずれも Virtualization Manager に統合され、シームレスに管理が可能です。

### 仮想化で重要となる「統合管理」

前回までの内容で、Integrity VM の使い方について一通り説明しました。今回は、Integrity VM の統合管理のための GUI ツール「Virtualization Manager」およびプランニング・ツール「Capacity Advisor」について紹介します。

Integrity VM のような仮想化技術は大変便利なものですが、その優れた柔軟性は、諸刃の刃でもあります。例えば、個々の VM に対してどの物理リソースを割り当てているかといった「仮想環境と物理環境のマッピング」を手作業でドキュメント化したり、CPU、メモリ、I/O などの配分状況を個別のツールで管理したりする手間が発生します。

しかし、手作業によって作成されたドキュメントはすぐに古くなってしまいます。IT インフラ管理の効率化をもたらすはずの仮想化技術が、逆に現場のオペレーションの足を引っ張りかねないのです。

そこで HPE では、この問題を解決する統合管理ツール「Integrity Essentials Virtualization Manager」（以下、Virtualization Manager）を 2006 年 4 月にリリースしました。Virtualization Manager は、ネットワーク上のすべての Integrity サーバーや HP 9000 サーバーから自動的に構成情報を収集し、仮想環境と物理環境の最新のマッピング状況をグラフィカルに表示、管理できるツールです。また同時に、各サーバーや仮想環境の負荷状況もグラフ表示します

Integrity VM でやさしく学ぶサーバー仮想化

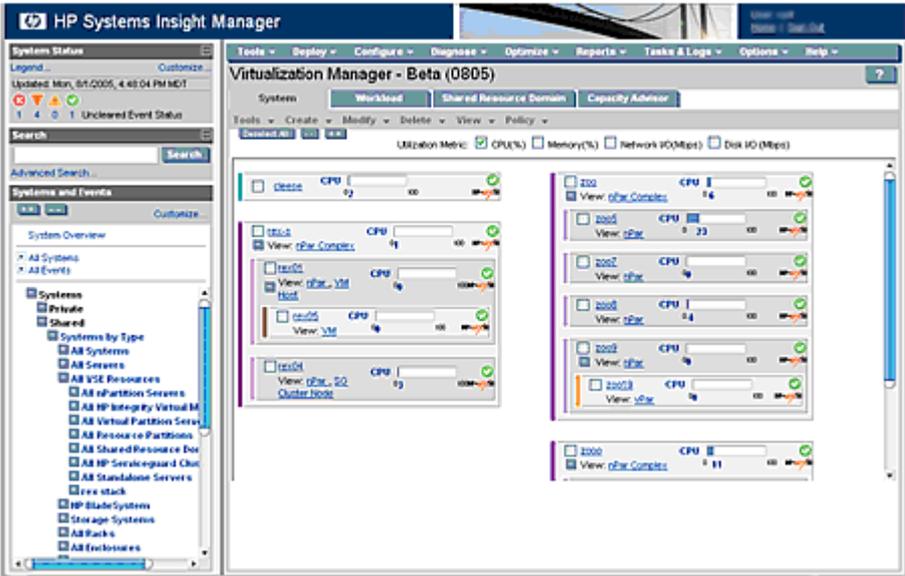


図 1 : Virtualization Manager

図 2 は、Virtualization Manager によって仮想環境と物理環境がどのように GUI 表示されるかを示す画面です。

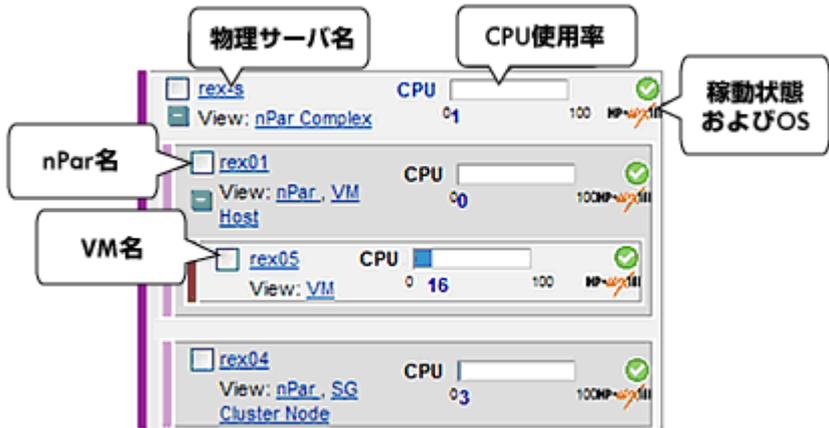


図 2 : Virtualization Manager による表示例

ここで、もっとも外側を囲む領域全体が、1 台の Integrity サーバーを表します。この例では、「rex-s」という名称のサーバーであることがわかります。また画面上部には現在の CPU 利用率を示すバググラフがあり、さらに右上には、稼働状態を示す緑のチェックマーク、そして稼働 OS を示すロゴマークが表示されています。

さて、HPE の仮想化技術ポートフォリオ VSE を構成する nPartitions (nPars) 、Virtual Partitions (vPars) 、Integrity VM といった仮想化技術は、それぞれを入れ子にして組み合わせることができます。この VSE のポートフォリオを反映して、Virtualization Manager の GUI 表示も入れ子構成になっています。例えば図 2 の例では、物理サーバー rex-s の内部には nPars 「rex01」 および 「rex04」 が存在します。また、nPars である rex01 の内部には Integrity VM 「rex05」 が動作しています。

この Virtualization Manager の特徴は、他の管理ツールとの連携機能です。例えば、物理サーバー名 rex-s をクリックすれば、同サーバーのハードウェア管理画面が開きます。また CPU 利用率のバーグラフは、後述する「[Capacity Advisor](#)」へのリンクになっています。同様に、nPars や vPars のパーティション管理画面、[Serviceguard](#) クラスターの管理画面、[gWLM](#) によるワークロード管理画面にもクリックひとつでジャンプできる仕組みになっています。

## VM Manager による Integrity VM 管理

[Virtualization Manager](#) は、Integrity VM の VM ゲストを管理するためのツール「VM Manager」を統合しています。例えば、図 2 の画面において Integrity VM のホスト名をクリックすると、以下のような VM Manager 画面が表示されます。

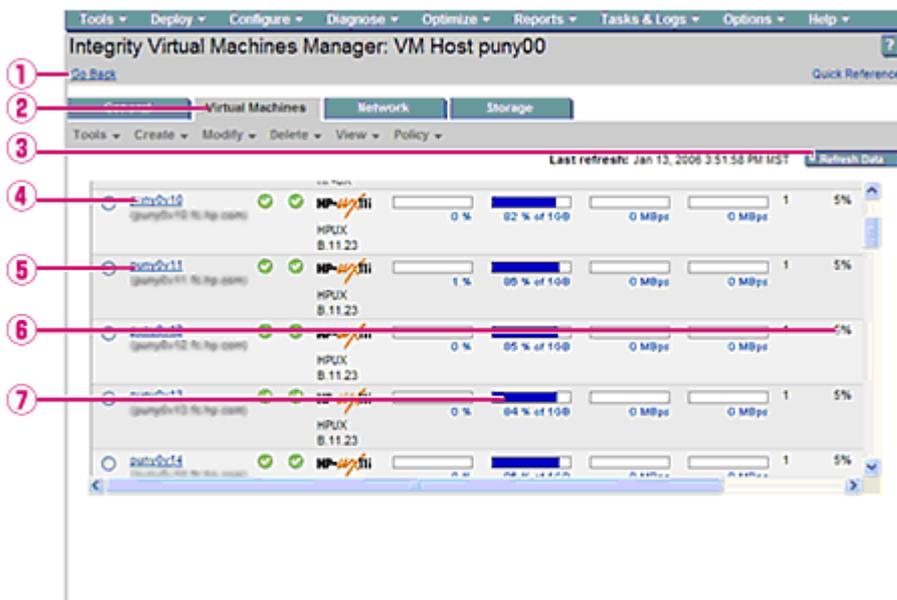


図 3 : VM Manager

1. Virtualization Manager 画面に戻るリンク
2. ホストサーバー上で動作するすべての VM を一覧表示するタブ
3. 画面内容を最新のものに更新するリンク
4. VM の詳細情報を表示するリンク
5. VM を選択し、さまざまなアクションを実行するためのラジオボタン
6. VM に割り当てられた CPU リソースを表すエンタイトルメント値
7. VM のリソース利用率

図 3 のように、VM Manager では各 VM の CPU やメモリ、ディスク、ネットワークなどのリソース使用率や、エンタイトルメントを確認できます。また VM Manager では、仮想ネットワーク・デバイスや仮想ストレージ・デバイスなどのマッピング状態も GUI 表示可能です。

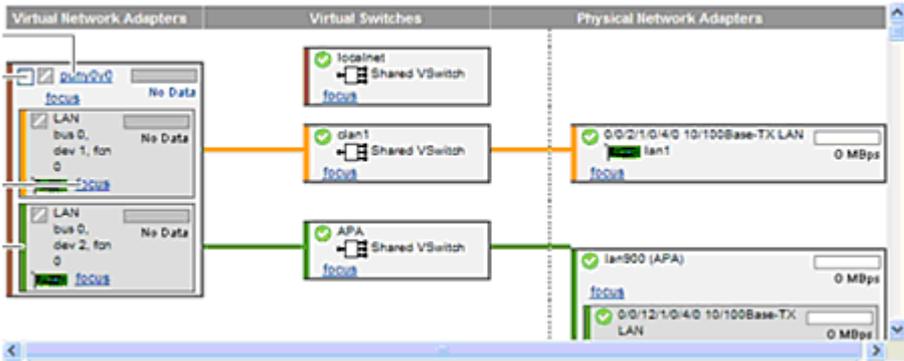


図 4 : 仮想ネットワーク・デバイスの GUI 表示例

上記例では、画面左側の仮想 NIC と画面右側の物理 NIC が、画面中央の仮想スイッチによってどのように“結線”されているかが図示されています。このほか VM Manager では、物理ディスクと仮想ディスク間のマッピング表示や、VM の作成・起動・リセット・停止・削除といった構成作業を実施できます。

### Capacity Advisor による仮想化シミュレーション

Virtualization Manager には、もうひとつの仮想化環境向けツール「Integrity Essentials Capacity Advisor」(以下、Capacity Advisor) が統合されています。このツールを利用することで、以下のようなポイントについてシミュレーションが可能です。

- サーバー仮想化によって CPU リソースの利用率はどの程度まで高められるか？
- サーバー仮想化によって、既存のサーバー上に追加のアプリケーションを収容できるか？
- パーティション間の CPU 配分を変更すると、どのような影響があるか？
- 新しいアプリケーションはどの VM に配置すべきか？
- TiCAP (テンポラリー・インスタント・キャパシティ) の導入でどのようなメリットが得られるか？
- クラスターのフェイルオーバーが発生したとき負荷状況はどのように変化するか？
- gWLM の導入でどのようなメリットが得られるか？

以下の画面は、Capacity Advisor で CPU 利用率の履歴を表示した例です。

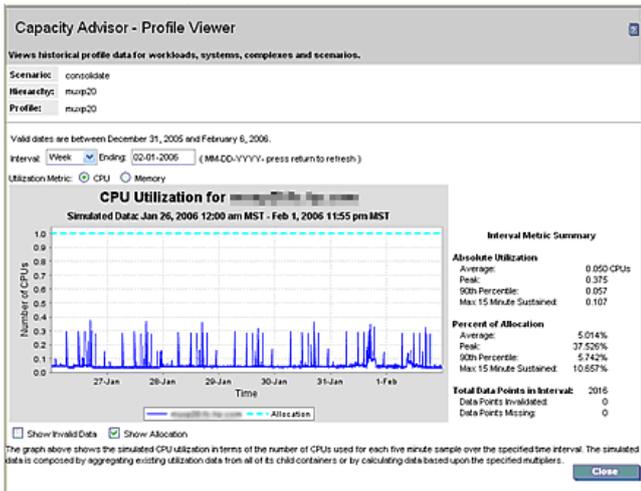


図 5 : Capacity Advisor

具体的には、以下のような作業に活用できます。

- 既存のワークロード情報のインポート——これまでのアプリケーション運用で計測されたワークロード情報のインポート
- ワークロード情報の編集——今後のワークロードのトレンド予測を反映させる
- 新しいワークロード情報の作成——新たに発生すると予想されるワークロード情報を手作業で入力する
- ワークロード移動時のシミュレーション
- VM 移動時のシミュレーション
- 新しいシステム構築時のシミュレーション（スタンドアロン構成や VM ホスト構成の作成）
- 既存システム変更時のシミュレーション（CPU の動作スピードや CPU 数、メモリ量の変更）

では、Capacity Advisor による実際の仮想化シミュレーションの例を紹介しましょう。

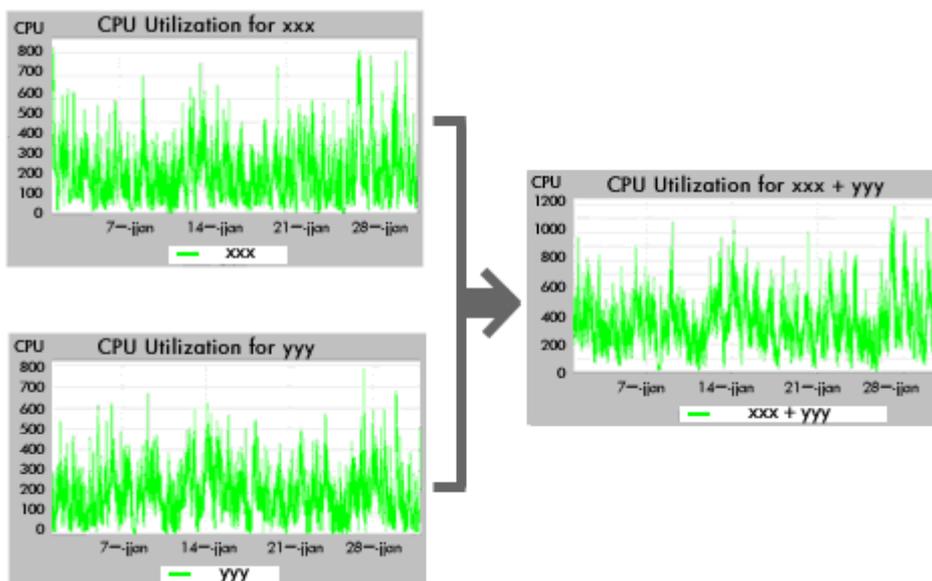


図 6：ワークロード移動時の仮想化シミュレーションの例

この例では、画面左側に示す 2 つのワークロードをひとつのマシン上にコンソリデーションした場合のシミュレーション結果が画面右側のグラフに表示されています。もともとのワークロードは、それぞれが 8 CPU 構成の独立したマシンで運用されていたもので、ピーク時負荷の数値は 700%~800%（CPU が 7~8 個分）に達しています。しかし、この 2 つのワークロードをひとつに集約するシミュレーションを実施すると、画面右側のグラフが示すようにピーク時負荷は 1200% 以下に収まることわかります。つまり、CPU 8 個+8 個=16 個ではなく、CPU 12 個分のサーバー・リソースで十分なのです。

このように Capacity Advisor によるシミュレーションによって、ワークロードやリソースのさまざまな組み合わせを事前に試すことが可能になります。

以上、今回は Integrity VM をはじめとする HPE の仮想化技術を統合管理するツール群について紹介しました。

## 第 8 回

# Ignite-UX によるネットワーク・インストール・その 1

2006 年 11 月 テクニカルライター 小林聡史

HP-UX のネットワーク・インストール・ツール Ignite-UX を利用することで、Integrity VM の VM ゲストに対する OS インストール作業を簡素化できます。今回は、この Ignite-UX を用いて、HP-UX の OS イメージをネットワーク経由で VM ゲストにインストールする基本的な手順を説明します。

## Ignite-UX を使って OS インストール

まずは Integrity VM から Ignite-UX を利用する前提条件として、VM ゲストがネットワーク上の Ignite-UX サーバーに接続可能であるか確認します。つまり、VM ゲストに仮想スイッチが設定済みであり、さらにその仮想スイッチに関連づけられた物理 NIC が Ignite-UX サーバーと同じサブネットに接続されている状態です。

例えば、Ignite-UX サーバーが VM ホストの lan0 と同じサブネットに属しているとします。hvvmnet コマンドを利用することで、この lan0 に割り当てられている仮想スイッチの有無と同スイッチの情報を確認できます。

```
# hvvmnet
Name      Number State   Mode   PPA   MAC Address   IP Address
-----
vswA      8 Up      Shared lan0  0x00110a319869 15.99.87.15
vswB      11 Up     Shared lan1  0x00110a3177c6 16.72.120.11
```

仮想スイッチのネットワーク接続状態の確認

この出力例は、仮想スイッチ vswA が lan0 に割り当てられていることを示しています。よって vswA に接続されている VM ゲストであれば、Ignite-UX サーバーに接続可能です。一方、VM ゲストが vswA に接続されていることを確認するには、hvvmstatus コマンドを以下のように実行します。

```
# hvvmstatus -P myvm02
...
[Network Interface Details]
Interface Adaptor   Name/Num   Bus Dev Ftn Mac Address
-----
vswitch   lan       vswA      0  1  0  86-ab-c8-5d-75-09
```

VM ゲストの仮想スイッチ接続状態の確認

この出力例では、VM ゲスト myvm02 が vswA に接続されていることを示しています。この例のように、VM ゲストと仮想スイッチの接続が確認できない場合は、仮想スイッチとの接続を新たに定義する必要があります。例えば、hvvmmodify コマンドを実行します。

```
# hvvmmodify -P myvm02 -a network:lan::vswitch:vswA
```

## Ignite-UX サーバーのインストール

つづいて、Ignite-UX サーバーのインストール手順を説明します。HPE では、Integrity VM の OS インストール作業に用いる Ignite-UX サーバーについて、HP-UX 11i v2 を搭載した Integrity サーバーの利用を推奨しています。HP 9000 サーバーや旧バージョンの Ignite-UX サーバーでは、VM ゲストとの通信が行えない場合があるので注意してください。なお、以下に説明するインストール手順は、Ignite-UX バンドル B5725AA に基づいています。Ignite-UX のインストールに関する最新情報は、HP-UX ドキュメント『Ignite-UX 管理ガイド』を参照してください。

Ignite-UX サーバーのインストールや初期化の作業は、Ignite-UX に備わるユーザーインタフェース上で実行できます。ただし、このユーザーインタフェースを Integrity サーバー上で使用するには、次に説明する 2 種類のネットワーク設定のいずれかをあらかじめ実施しておく必要があります。

### 固定 IP アドレスの場合

Ignite-UX クライアント（つまり今回の場合は VM ゲスト）に IP アドレスを固定で割り当てる場合は、Ignite-UX サーバーとなる Integrity サーバーの設定ファイル/etc/bootptab を編集します。同ファイルでは、デフォルト・エントリを参照するかたちでクライアント・エントリ（VM ゲスト向けのエントリ）を追加します。以下に、2 つのクライアント・エントリを追加した例を示します。

```
IADEF:¥
ht=ethernet:¥
hn:¥
bf=/opt/ignite/boot/nbp.efi:¥
bs=48:¥
bp=15.99.87.15:¥
sm=255.255.248.0:¥
gw=15.99.87.254:¥
ds=15.99.52.11:¥
vm=rfc1048:

myvm01:tc=IADEF:ip=15.99.87.205:ha=86abc85d751a:
myvm02:tc=IADEF:ip=15.99.87.38:ha=86abc85d7509:
```

/etc/bootptab の設定例

この例は、1 つのデフォルト・エントリ（IADEF）と 2 つのクライアント・エントリ（myvm01 および myvm02）が定義されています。デフォルト・エントリでは、次の 4 つの値を変更する必要があります。

- bp — クライアントに返答する Ignite-UX サーバーの IP アドレス
- sm — クライアントが使用しているサブネット・マスク
- gw — ネットワーク・ゲートウェイのアドレス
- ds — DNS サーバーのアドレス

一方、クライアント・エントリでは、クライアントのホスト名から始まり、残りの 3 つの属性を次のように定義します。

- tc — このクライアントが使用するデフォルト・エントリ（上記例では IADEF）

- ip —— クライアントの IP アドレス (固定)
- ha —— クライアントの MAC アドレス

なお、VM ゲストの MAC アドレスは、`hvvmstatus` コマンドまたは VM コンソールで確認できます。

## 動的 IP アドレスの場合

VM ゲストを DHCP クライアントとして動作させ、動的な IP アドレスの割り当てを実施したい場合は、設定ファイル `/etc/dhcptab` を以下のように設定して、DHCP サーバーを構成します。

```
dhcp_device_group:¥
re:¥
ncid:¥
class-id="PXEClient:Arch:00002:.*":¥
lease-time=300:¥
subnet-mask=255.255.248.0:¥
addr-pool-start-address=15.99.87.6:¥
addr-pool-last-address=15.99.87.253:¥
bf=/opt/ignite/boot/nbp.efi
```

`/etc/dhcptab` の設定例

ここで定義されている各属性は、以下のような意味を持ちます。。

- `dhcp_device_group` —— DHCP デバイス・プール・グループを有効にします。これにより、DHCP クライアントが送信するブート・メッセージ中に特定の `class-id` が含まれる場合、DHCP サービスを提供します。
- `class-id` —— DHCP クライアントの `class-id` を指定します。Integrity サーバーまたは VM ゲストを DHCP クライアントとして用いる場合、同クライアントが送信するブート・メッセージには「PXEClient:Arch:00002」で始まる `class id` が入ります。よって `class id` を「PXEClient:Arch:00002:.\*」と指定することで、Integrity サーバーまたは VM ゲストからのブート・メッセージにのみ応答する設定となります。
- `re` —— `class-id` をデフォルトのリテラル照合ではなく正規表現で照合するように指示します (HP-UX 11i v2 リリース 11.23 で新しく追加されたオプションです)。
- `ncid` —— DHCP クライアントからのメッセージへの応答する際に、`class-id` を返送しないように指示する必須のオプションです (HP-UX 11i v2 リリース 11.23 で新しく追加)。
- `lease-time` —— DHCP クライアント提供する IP アドレスのリース時間です (秒単位)。
- `addr-pool-start-address` —— DHCP クライアントに提供する IP アドレス範囲の先頭アドレスです。
- `addr-pool-last-address` —— DHCP クライアントに提供する IP アドレス範囲の末尾のアドレスです。
- `subnet-mask` —— DHCP クライアントが使用するサブネット・マスクです。
- `bf` —— ブート時に使用する EFI ネットワーク・ブート・プログラムです。

以上のエントリを Ignite-UX サーバーの `/etc/dhcptab` ファイルに追加した後、`bootpd` を再起動します。`bootpd` は通常 `inetd` が起動するため、`/etc/inetd.conf` ファイルに `bootpd` の設定が含まれているはずですが。その場合は、`bootpd` のプロセスを終了することで、`inetd` によって `bootpd` が再起動されます。

以上で Ignite-UX サーバーの設定は完了です。次回は、この Ignite-UX サーバーに対して VM ゲストから接続し、ネットワーク・インストールを実施する手順を説明します。

## 第 9 回

# Ignite-UX によるネットワーク・インストール・その 2

2006 年 12 月 テクニカルライター 小林聡史

複数の VM を運用する場合、それぞれの VM で OS やソフトウェアの設定を繰り返すのは面倒です。そこで、各 VM に共通する設定で OS やソフトウェアをあらかじめインストールした「ゴールド・イメージ」を作成することで、こうした繰り返し作業を効率化できます。ゴールド・イメージとは、OS やソフトウェアのインストール、各種設定を事前に済ませた OS イメージを指します。こうしたゴールド・イメージを使用することで、標準的なソフトウェアをインストール済みの OS をすばやく準備できます。

### ネットワーク・インストールの実行

今回は、Integrity VM のネットワーク・インストールのための事前設定を、Ignite-UX サーバーに行う手順を説明しました。つづいては、Integrity VM へのネットワーク・インストール作業を進めます。

まずは、Integrity VM の仮想コンソールにアクセスします。hvvmstart コマンドで VM を起動したのち、hvvmconsole コマンドで仮想コンソールを開きます(図 1)。



図 1 : VM 初回起動後の仮想コンソール

つづいて、次の手順でネットワーク・ブートを行います。まずは「Boot option maintenance menu」を選択して Enter キーを押します。つづいて表示される「Boot Maintenance Manager」メニュー(図 2)で「Boot from a File」を選択し、Enter キーを押します。

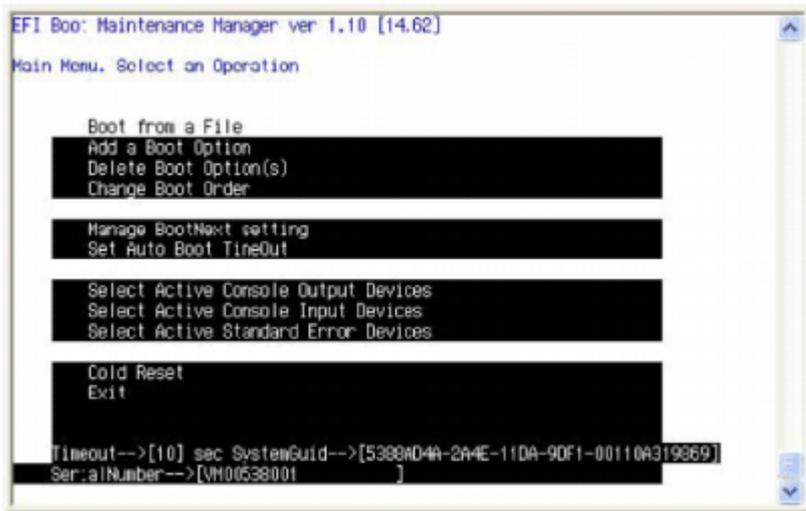


図 2 : Boot Maintenance Manager メニュー

次に、「Boot from a File」メニュー(図 3)で、Ignite-UX で使用する MAC アドレスが表示された Load File エントリを選択してください。このとき複数の Load File エントリが表示された場合は、前回説明した Ignite-UX サーバーの/etc/bootptab ファイルに登録した VM の MAC アドレスが表示されているエントリを選択してください(IP アドレス固定の場合)。

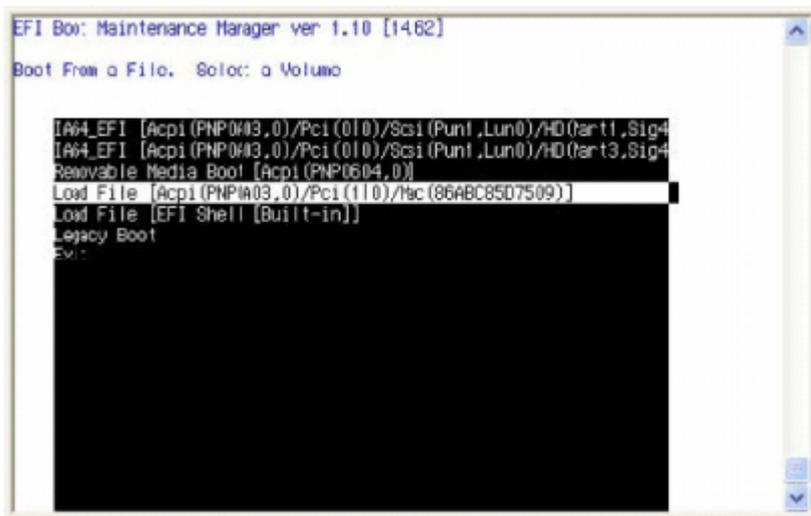


図 3 : Boot From a File メニュー

適切な Load File エントリを選択すると VM が Ignite-UX サーバーへ要求を送信し、すべての設定が正しければ Ignite-UX サーバーが応答します。このとき図 4 のメッセージが表示され、VM と Ignite-UX サーバーの通信が開始されます。

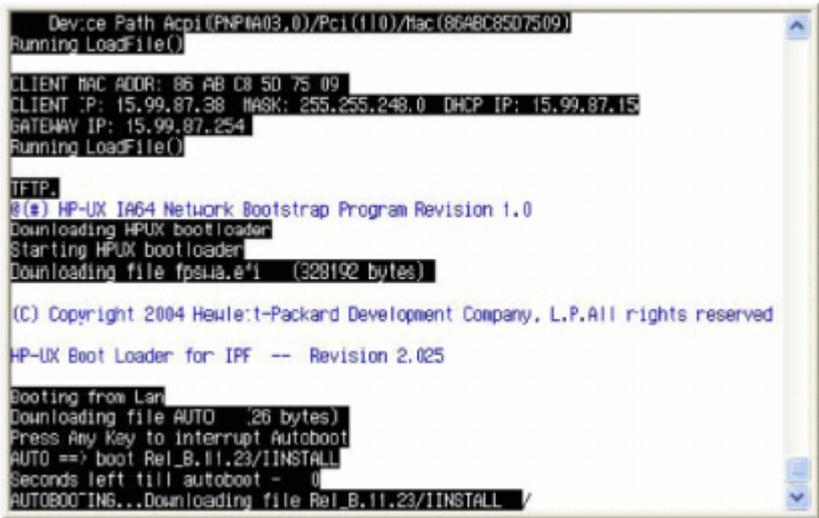


図 4 : Ignite-UX サーバーとの通信時に表示されるメッセージ

しばらくすると「HP-UX installation」メニュー(図 5)が表示されます(なお、このメニューが表示されるまでの間に、ターミナル・タイプの指定を求めるプロンプトが表示されることがあります)。

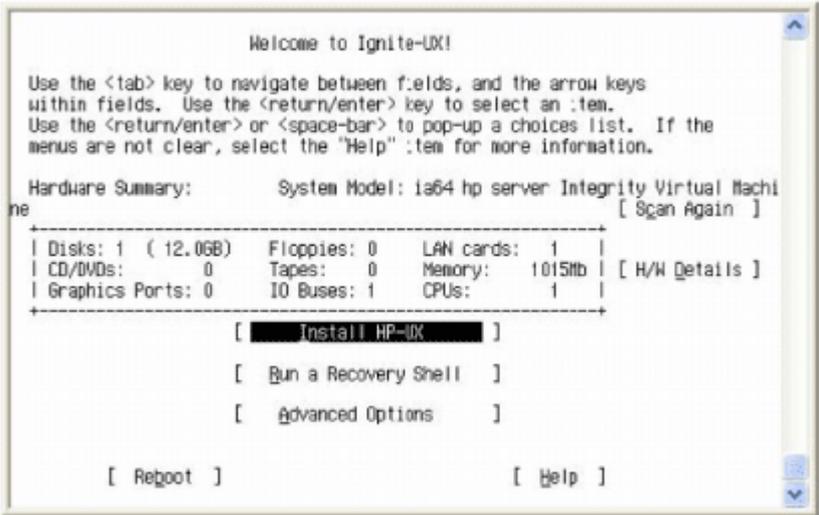


図 5 : HP-UX installation メニュー

あとは、通常の HP-UX インストールの手順を進めます。

### より高度なネットワーク・インストール

複数の VM を運用する場合、それぞれの VM で OS やソフトウェアの設定を繰り返すのは面倒です。そこで、各 VM に共通する設定で OS やソフトウェアをあらかじめインストールした「ゴールド・イメージ」を作成することで、こうした繰り返し作業を効率化できます。

ゴールド・イメージとは、OS やソフトウェアのインストール、各種設定を事前に済ませた OS イメージを指します。Ignite-UX では、OS 設定を tar または cpio 形式でアーカイブしてゴールド・イメージを作成し、それを Ignite-UX サーバー経由で各マシ

ンにネットワーク・インストール可能です。こうしたゴールド・イメージを使用することで、標準的なソフトウェアをインストール済みの OS をすばやく準備できます。

ゴールド・イメージの作成手順は、『Ignite-UX 管理ガイド』の「ゴールド・イメージ」の項で詳しく説明されています。ここでは、標準的な Integrity VM のゴールド・イメージの作成方法を紹介します。ゴールド・イメージの基本的な作成手順は次のとおりです。

1. ゴールド・イメージの作成に使用するシステムを用意
2. ゴールド・イメージに含める Integrity VM ソフトウェアをインストール(省略可)
3. オペレーティングシステムのアーカイブ(ゴールド・イメージ)を作成
4. ゴールド・イメージを認識するように Ignite-UX サーバーを設定

これらの各手順について、以下に説明します。

## ゴールド・イメージを作成するシステムの用意

ゴールド・システム(ゴールド・イメージを作成するシステム)を準備するには、まず VM に OS およびソフトウェアをインストールします。なお VM にソフトウェアをインストールする場合は、必要なソフトウェア・バンドルを一か所の SD-UX ソフトウェア・デポに収めておけばよいでしょう。これにより、すべてのソフトウェアを一度に選択してインストールでき、何度も再起動する必要がありません。

## Integrity VM ソフトウェアのインストール

Integrity VM 用のゴールド・システムを作成する場合は、Integrity VM に固有のソフトウェア・バンドルとして以下の 3 つを必要に応じてインストール可能です。

VMGuestLib	Integrity VM に関連するシステムの基本情報を提供するライブラリ・セットです。
HPVM-Guest	HP-UX を VM 向けにチューニングするためのツール群です。VM の動作状態を確認する 2 つのユーティリティ・コマンド(hpvminfo、hpvmcollect)を含みます。
VMProvider	Integrity VM WBEM プロバイダ。Virtualization Manager(vman)、Integrity VM GUI(vmmgr)等の Virtual Server Environment(VSE)ツールに VM の動作情報を提供します。VMProvider の利用には、VMGuestLib または HPVM-Guest が必要です。

これらのうち VMGuestLib と VMProvider は、HP-UX 11i v2(2005 年 12 月版)から製品の一部として提供されています。したがって、ゴールド・システム設定にこれ以降の新しい HP-UX を使用する場合は、これら 2 つのバンドルをあらためてインストールする必要はありません。これらのバンドルは Integrity VM のインストール・メディアにも収録されており、メディアからコピー(swcopy)または直接インストール(swinstall)できます。

一方、HPVM-Guest バンドルは、Integrity VM をインストールしたホスト上に SD-UX 配布テープ形式ファイル(/opt/hpvm/guest-images/hpux/hpvm\_guest\_depot.sd)として配置されています。このファイルに含まれている HPVM-Guest をデポにコピーし、VM にインストールします。デポにコピーするには、次のように swcopy コマンドを使用します。

```
# swcopy -s /opt/hpvm/guest-images/hpux/hpvm_guest_depot.sd \  
\* @ <host>:./<depot path>/
```

なお、HPVM-Guest バンドルはインストール後に再起動する必要があるため、他のソフトウェアと共に 1 つのデポに収納して一括インストールすることをおすすめします。

以上、今回は Ignite-UX サーバーによるネットワーク・インストールの流れを説明し、ゴールド・イメージの作成方法を解説しました。次回もひきつづきゴールド・イメージの策定手順を説明する予定です。

## 最終回

# Ignite-UX によるネットワーク・インストール・その 3

2007 年 1 月 テクニカルライター 小林聡史

連載の最終回となる今回は、Ignite-UX サーバーによるネットワーク・インストールで用いる「ゴールド・イメージ」の作成手順を紹介します。ゴールド・イメージとは、OS やソフトウェアのインストール、各種設定を事前に済ませた OS イメージを指します。こうしたゴールド・イメージを使用することで、標準的なソフトウェアをインストール済みの OS をすばやく準備することができます。

## オペレーティング・システムのアーカイブを作成

今回は、前回に引き続き、Ignite-UX サーバーによるネットワーク・インストールで用いるゴールド・イメージの作成手順を紹介します。

ゴールド・システムの設定後は、Ignite-UX ソフトウェアをインストールするか、`/opt/ignite/data/scripts/make_sys_image` ファイルをゴールド・システムの `/tmp` にコピーしてください。このファイルをコピーした場合は、`/tmp/make_sys_image` ファイルのパーミッションを「実行可能」に変更します。

つづいて Ignite-UX サーバーで、ゴールド・イメージのアーカイブを保管するファイルシステムに、十分な空き領域があることを確認します。デフォルトの保管先は `/var/tmp` です。通常、`/var` の領域は他のファイルシステムに比べて拡張しにくいいため、保管先を他のファイルシステムに変更できます。

次に、以下の手順でゴールド・イメージを作成します。

1. ゴールド・システムで、次のコマンドを実行します。

```
# /tmp/make_sys_image -s <Ignite-UX server IP address> \ -d /<path to target directory> -n <archive file name>
```

具体的な実行例を次に示します。

```
# /tmp/make_sys_image -s 15.99.87.15 \ -d /var/opt/ignite/data/Rel_B.11.23 -n B.11.23_VM_archive.gz
```

このコマンドは、IP アドレス 15.99.87.15 のホストで /var/opt/ignite/data/Rel\_B.11.23 ディレクトリにゴールド・イメージを作成します。ゴールド・システムは Ignite-UX サーバーの /.rhosts ファイルに登録しておく必要があります。登録していない場合、次のエラーメッセージが表示されます。

```
ERROR: Cannot remsh server system_name (check server .rhosts file).
```

通常、make\_sys\_image は数分で終了します。実行中にはゴールド・システムで、他の処理を実行しないでください。処理が完了すると、Ignite-UX サーバーに B.11.23\_VM\_archive.gz (gzip 形式の) ファイルが作成されます(上記の例では、/var/opt/ignite/data/Rel\_B.11.23/B.11.23\_VM\_archive.gz)。

2. ゴールド・イメージを格納しているディレクトリを NFS マウントできるように設定する必要があります。上記の例では、Ignite-UX サーバーの /etc/exports ファイルに、/var/opt/ignite/data/Rel\_B.11.23 を次のように登録します。

```
/var/opt/ignite/data/Rel_B.11.23 -anon=2
```

登録後に exportfs -a を実行して、ディレクトリをエクスポートします。

## ゴールド・イメージを認識する Ignite-UX サーバーの設定

ゴールド・イメージの作成後は、Ignite-UX サーバーで使用できるように設定する必要があります。手順を次に示します。

1. 次のコマンドをアーカイブがあるディレクトリで実行して、設定ファイルで使用する impacts 文を取得してください。

```
# /opt/ignite/lbin/archive_impact -t -g B.11.23_VM_archive.gz
```

たとえば、ゴールド・イメージが /var/opt/ignite/data/Rel\_B.11.23 ディレクトリにある場合は、次のコマンドを実行します。

```
# /opt/ignite/lbin/archive_impact -t -g \ ./B.11.23_VM_archive.gz > impacts.txt
```

このコマンドの実行には、1~2 分を要します。

2. サンプルの設定ファイルをシステムのアーカイブがあるディレクトリにコピーして、設定ファイルを作成してください。実行例を示します。

```
# cp /opt/ignite/data/examples/B.11.23.golden_image.cfg \ /var/opt/ignite/data/Rel_B.11.23/vm.cfg
```

作成した設定ファイルは、以下の手順に従って変更する必要があります。

3. 設定ファイルの sw\_source セクションにある nfs\_source の定義を、Ignite-UX サーバーと格納先のパスに合わせて変更してください。たとえば、前述した Ignite-UX サーバーにゴールド・イメージが存在する場合、

```
nfs_source = "10.2.72.150: /var/opt/ignite/archives/Rel_B.11.23"
```

を次のように変更します。

```
nfs_source = "15.99.87.15: /var/opt/ignite/data/Rel_B.11.23"
```

4. 同じ設定ファイルの `init sw_sel` セクションを、使用するゴールド・イメージに合わせて変更してください。サンプルの設定ファイルでは 2 つの異なるイメージが指定されています。1 つは Itanium ベースのシステム用 (ia64) で、もう 1 つは PA-RISC システム用 (hppa) です。PA-RISC システム版 HP-UX のイメージを作成しなかった場合は、PA-RISC システムのイメージ ("is\_hppa") の `init sw_sel` セクションを削除してください。続いて、次の手順を実行してください。

a) `description` のパラメーターを、使用するゴールド・イメージの説明に変更します。たとえば、

```
description = "B.11.23 IA golden image archive"
```

を次のように変更してください。

```
description = "HP-UX 11.23 image for VMs"
```

b) `archive_path` を、実際のアーカイブ・ファイル名に変更してください。たとえば

```
archive_path = "B.11.23_archive_IA.gz"
```

を前述した例と同じアーカイブ・ファイル名にするには、次のように変更します。

```
archive_path = "B.11.23_VM_archive.gz"
```

c) `impacts` の各行を、手順 1 で取得した内容に置き換えてください。たとえば、次の行を

```
impacts = "/" 7659Kb
impacts = "/dev" 11Kb
}impacts = "/etc" 76229Kb
impacts = "/home" 1Kb
impacts = "/opt" 2223298Kb
impacts = "/sbin" 109002Kb
impacts = "/stand" 51127Kb
impacts = "/usr" 2379518Kb
impacts = "/var" 848454Kb
```

以下のように置き換えます。

```
impacts = "/" 833Kb
```

```

impacts = "/dev" 13Kb
impacts = "/etc" 82077Kb
impacts = "/home" 133696Kb
impacts = "/opt" 2536512Kb
impacts = "/sbin" 111333Kb
impacts = "/scratch" 135843Kb
impacts = "/stand" 52187Kb
impacts = "/usr" 2396952Kb
impacts = "/var" 1603044Kb

```

「=」の後には、手順 1 で作成した impacts.txt ファイルから取得した内容です。 以上の変更を加えると、init sw\_sel セクションは次のようになります。

```

(is_ia64) {
  init sw_sel "B.11.23 IA golden image archive" {
    description = "HP-UX 11.23 image for VMs"
    sw_source = "golden image archive"
    sw_category = "HPUXEnvironments"
    archive_type = gzip tar

    # For NFS, the path to the archive is relative to the mount point
    # specified in the "nfs_source" keyword within the sw_source stanza
    # above:
    archive_path = "B.11.23_VM_archive.gz"

    # Here we describe how much space is used in each of the mentioned
    # subdirectories. This information is obtained by running the
    # "archive_impact" command against an archive as follows:
    #   /opt/ignite/lbin/archive_impact -tg B.11.23_archive_IA.gz
    # The output from the above command should replace the "impacts"
    # statements below.
    #
    impacts = "/" 833Kb
    impacts = "/dev" 13Kb
    impacts = "/etc" 82077Kb
    impacts = "/home" 133696Kb
    impacts = "/opt" 2536512Kb
    impacts = "/sbin" 111333Kb
    impacts = "/scratch" 135843Kb
    impacts = "/stand" 52187Kb
    impacts = "/usr" 2396952Kb
    impacts = "/var" 1603044Kb
  } = TRUE}

```

5. Ignite-UX の新しい設定をインストールするために、/var/opt/ignite/INDEX ファイルを編集してください。新しい cfg セクションをこのファイルに追加します。追加する cfg セクションは次のように定義します。

```

cfg "HP-UX B.11.23 VM Config" {
  description "This selection supplies the VM configuration."
  "/opt/ignite/data/Rel_B.11.23/config"
  "/var/opt/ignite/data/Rel_B.11.23/vm.cfg"
  "/var/opt/ignite/config.local"
}

```

6. 次のコマンドを実行して、追加した設定情報の構文をチェックしてください。

```
# /opt/ignite/bin/instl_adm -T
```

このコマンドで問題が見つからなければ、設定は完了です。

## 新しい設定をインストール

上記の手順をすべて完了すると、「Ignite-UX による VM のロード」で前述した手順に従って Ignite-UX を使用し、新しい VM 設定をインストールできます。HP-UX installation メニューから User Interface and Media Options 画面まで実行してください。ここで Guided Installation オプションを選択して、Select an overall system configuration 画面まで実行してください。カーソルを Configurations: フィールドに位置付け Enter キーを押すと、選択可能な VM 設定が表示されます。

この時点で、ゴールド・イメージ (B.11.23\_VM\_archive.gz など) が見つからない旨のエラーメッセージが表示されることがあります。通常、このエラーの原因は、ゴールド・イメージを含むファイルシステムが、NFS クライアントにエクスポートされていないためです。Ignite-UX サーバーで適切なファイルシステムがエクスポートされていることを確認し、インストールをやり直してください。

以上、今回は Ignite-UX サーバーによるネットワーク・インストールで用いるゴールド・イメージの作成手順を紹介しました。

## HP-UX

[www.hpe.com/jp/hpux](http://www.hpe.com/jp/hpux)

---

© Copyright 2018 Hewlett Packard Enterprise Development LP.

本書の内容は、将来予告なく変更されることがあります。日本ヒューレット・パカード製品およびサービスに対する保証については、当該製品およびサービスの保証規定書に記載されています。本書のいかなる内容も、新たな保証を追加するものではありません。日本ヒューレット・パカードは、本書中の技術的あるいは校正上の誤り、脱字に対して、責任を負いかねますのでご了承ください。