

ホスト OS と協調して動作する仮想計算機の提案

五明 将幸[†] 新城
佐藤 聡[†] 中井

靖[†] 白石 光隆^{††}
央^{††} 板野 耕三[†]

1. はじめに

近年、安価で高速なパーソナルコンピュータ (Personal Computer, PC) が一般に普及したことにより、仮想化技術の適用範囲が PC に及ぶようになってきている。仮想計算機を構築・管理するソフトウェアのことを仮想計算機モニタ (Virtual Machine Monitor, VMM) と呼ぶ。VMM はその動作形態により、Type-I と Type-II の二種類に分類される。Type-I VMM は実機上で直接動作するのに対し、Type-II VMM はオペレーティング・システム (Operating System, OS) のプロセスとして動作する。

PC 上で動作する代表的な Type-II VMM として、VMware Workstation¹⁾、KVM²⁾ などが挙げられる。既存の Type-II VMM の中には、VMware Workstation の VMware tools や User Mode Linux の hostfs など、ゲスト OS に何らかの手を加えることで、ホスト OS とゲスト OS でファイルを利用可能にするものも存在する。しかし、基本的にはホストと複数のゲスト OS が独立して動作するように設計されている。そのため、このような VMM 上で、ゲスト OS 上からホスト OS 上の資源 (ファイルに加え、システムコール、ライブラリルーチン、アプリケーションなど) にアクセスするには、ネットワーク越しに利用する必要がある。しかし、一旦ネットワークを介することで、ユーザの識別が曖昧になり、ホスト OS のアクセス制御がうまく働かなくなる。

本稿では、上記の問題を解決するために、既存の VMM の設計とは全く異なったアプローチである協調型仮想計算機を提案する。協調型仮想計算機とは、ゲスト OS がホスト OS と協調して動作することを可能とする仮想計算機である。これに対して、従来の仮想計算機を隔離型仮想計算機と呼ぶ。図 1 に隔離型仮想計算機と協調型仮想計算機の違いを示す。隔離型仮想計算機では同一の物理計算機上で動作しているにも関わらず、ゲスト OS 上からホスト OS 上の資源を直接利用することができない。それに対し、協調型仮想計算機ではホスト OS 上の資源をゲスト OS 上から直接利用することが可能になる。

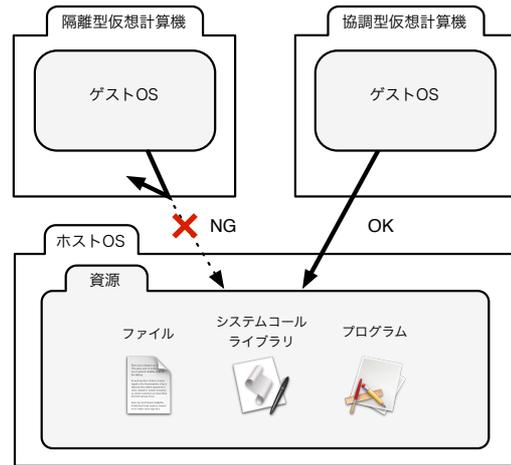


図 1 隔離型仮想計算機と協調型仮想計算機の相違点

2. 協調型仮想計算機

協調型仮想計算機では、ゲスト OS 上からホスト OS 上の資源にアクセスする際、実際に資源にアクセスする主体をホスト OS 上のプロセスが担うことにより、ホスト OS におけるユーザ管理の枠組みから外れないようにする。

ハイパバイザコールは VMM をゲスト OS から呼び出すためのインターフェースである。協調型仮想計算機では、従来の VMM が提供するハイパバイザコールを拡張した、ホストコールというものを新たに提供する。ホストコールは、ゲスト OS 上からホスト OS 上の資源を利用するためのインターフェースである。ゲスト OS 上のプロセスは、ホストコールを利用してホスト OS 上の資源にアクセスすることが可能になる。

図 2 に協調型仮想計算機が動作する様子を示す。ゲスト OS 上からホスト OS 上の資源にアクセスする際、ゲスト OS 上のプロセスはホストコールを用い VMM に要求を伝える。VMM はその要求をホスト OS 上の VMM の管理下にある、ホスト OS の資源にアクセスするためのプロセスに中継する。このプロセスを、シャドウ・プロセスと呼ぶ。最終的に、シャドウ・プロセスがホスト OS 上の資源にアクセスする。

シャドウ・プロセスは、ホストコールが呼ばれた際に開かれたファイル等の資源を保持するために、ホストコールを行う主体の数だけ生成される。

[†] 筑波大学システム情報工学研究科コンピュータサイエンス専攻

^{††} 筑波大学図書館情報メディア研究科

^{†††} 筑波大学第三学群情報学類

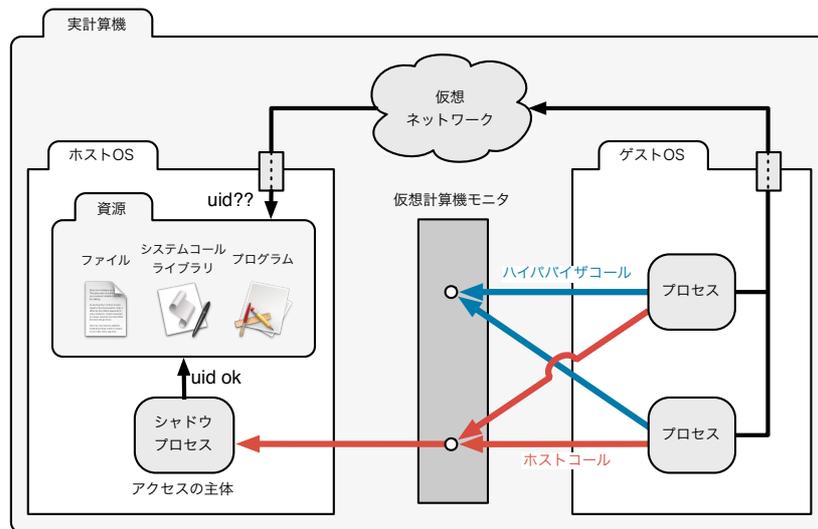


図 2 協調型仮想計算機の動作図

3. 二重 OS シェル

協調型仮想計算機で動作するシステムプログラムとして、ユーザがゲスト OS 上とホスト OS 上の両資源に対してアクセスできるユーザ・インタフェースを備えたシェルを開発している。これを二重 OS シェルと呼ぶ。二重 OS シェルは、ゲスト OS 上で動作し、協調型仮想計算機が提供するホストコールを用いたホスト OS 上の資源へのアクセスを提供する。

二重 OS シェルは、ホスト OS 上とゲスト OS 上の両資源にアクセスできるという特徴を生かし、ホスト OS 上の資源とゲスト OS 上の資源を組み合わせる仕組みを備える。次の例では、ゲスト上の `test.c` というテキストファイルを `a2ps` コマンドによりポストスクリプトに変換し、その出力結果を Unix のパイプの機能を用いてホスト OS 上の `lpr` コマンドに与え、ホスト OS 上のプリンタに出力している。この例において、`#!`記号がホスト OS 上でコマンドを実行することを意味している。

```
$ a2ps -o - test.c | !lpr
```

二重 OS シェルを用いることで、ホスト OS 上にしかないデバイスやホスト OS 上でしか動かないアプリケーションをゲスト OS 上から直接利用できる。また、ホスト OS にインストールされているアプリケーションを重複してゲスト OS にインストールする必要がなくなる。

4. 実 装

現在、協調型仮想計算機を KVM²⁾ をベースに実装している。KVM は Intel VT/AMD-v などの仮想化支援機能を備えた x86 アーキテクチャ上の Linux で

動作する Type-II の VMM である。KVM で提供されているハイパバイザコールを拡張する形で、ホストコールを実現する。

5. おわりに

本稿では、従来の隔離型の仮想計算機とは異なった目的である、協調型仮想計算機を提案した。協調型仮想計算機を利用することで、ゲスト OS 上からホスト OS 上の資源をに利用可能にする。また、協調型仮想計算機上で動作するシステムプログラムとして、二重 OS シェルを提案した。

今後は、実装を完了させ、機能と性能を評価する。

参 考 文 献

- 1) Sugerman, J., Venkitachalam, G. and Lim, B.-H.: Virtualizing I/O Devices on VMware Workstation's Hosted Virtual Machine Monitor, *Proceedings of the General Track: 2002 USENIX Annual Technical Conference*, Berkeley, CA, USA, USENIX Association, pp. 1-14 (2001).
- 2) Avi Kivity, Yaniv Kamay, Dor Laor, Uri Lublin and Anthony Liguori: kvm: the Linux Virtual Machine Monitor, *Proceedings of the Linux Symposium*, Ottawa, Ontario, pp. 225-230 (2007).