

# 実験で理解する！ ハイパーバイザの動作原理

坂井 弘亮

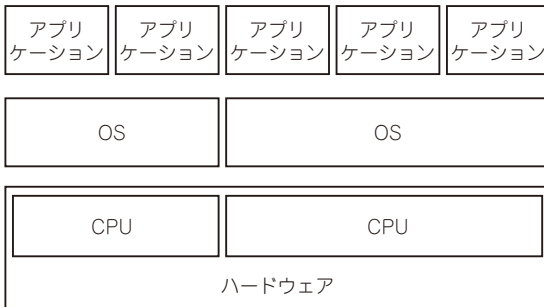


図1 マルチコアによる複数OSの動作  
シンプルな構成だが、OSや資源ごとに細かい調整が必要

## はじめに

現在、仮想化技術はコンピュータ技術の中でも特に重要なものとなっており、大きな位置を占めています。実際のところ、今となってはほとんどのプログラムは何らかの仮想化基盤の上で動いていて、そしてその仮想化も2重、3重にされていることが当たり前になっており、そうしたことに気づかずに利用していることも多々あることでしょう。

そして仮想化を行うための重要な技術として、ハイパーバイザがあります。ハイパーバイザについては「OSを動かすためのVM (Virtual Machine) 環境を提供する」という漠然とした理解があると思いますが、それを実現するための具体的な手法については実装を見る以外にはなかなか情報が得られず、理解を深めることが難しい部分があるように感じます。

そこで本稿ではハイパーバイザへの理解を深めることを目的に、まずハイパーバイザの概要と基本的かつ代表的な動作原理について説明し、さらにCPUのユーザ命令をネイティブに実行しつつも保護された命令は仮想化してエミュレーション実行するという実験を試みます。

## 1 ハイパーバイザ概要

### ● ハイパーバイザとは何か？

ハイパーバイザ (hypervisor) はOS (スーパーバイザ) に対して仮想コンピュータ (VM: Virtual Machine) の環境を提供することで仮想化を実現するための、OSよりも下に置かれる層のことです。

つまり機能としてはOSを動作させるためのシステム・エミュレータと同等ですが、エミュレータがコンピュータの動作をエミュレーションすることに対して、ハイパーバイザは多くのCPU命令をネイティブに実行するため、本来の実行速度に近い速度で動作させられます。このためエミュレータならばあらゆるアーキテクチャが動作可能となりますが、ハイパーバイザではそのOSをネイティブ動作させられる環境であるという前提が必要となります。

- エミュレータ…低速だが、あらゆるアーキテクチャが動作対象となる
- ハイパーバイザ…ネイティブ実行できるアーキテクチャに限られるが、高速に動作させられる

ハイパーバイザが必要とされるのは、多くは複数のOSを動作させたい場合です。そのようなとき、もっともシンプルな構成として、マルチコアのシステムならば図1のようにして各OSにCPUを割り当てて動作させるという方法が考えられます。

しかし、この構成では、それぞれのOSはシステム全体の資源の利用が衝突しないように、互いに調整しておく必要があります。この資源はこちらのOSで使うので、別のOS側では使わないようにしておく、といった細かい調整がOSごと、資源ごとに必要になる訳です (これはOSカーネルがない状態で、複数のプログラムを動かしてしまったらどうなるかということと似ている。プログラムごとに資源の利用が衝突しないよう、互いに気をを使う必要が出てしまう)。

そこで図2のようにハイパーバイザを最下層に置き、システム資源をハイパーバイザが管理することで各OSはそのような調整を必要とせず、動作できるよ