

仮想化ソフトウェア Xen の シミュレータとしての活用例

鈴木 量三朗

仮想化技術を使って ARM Cortex-A9 コア上に複数の OS を走らせることができる仮想化ソフトウェア Xen (ゼン) のシミュレータとしての応用事例について解説する。また仮想環境上だけでなく、実機上でも動作を確認している。

(編集部)

1 シミュレーション環境としての Xen

本章では、オープン・ソースの仮想化技術として QEMU と並んで有名な Xen について解説します。

QEMU は、ほかの章で解説されているとおり、ハードウェアや CPU を仮想化しエミュレーションします。そのため、i386 の Linux や Windows 上で、ARM のプラットフォーム (CPU を含めたハードウェア) を丸ごとエミュレーションすることができます (図1)。

一方、Xen が仮想化するのは、OS を動作させる環境であり、主に MMU (Memory Management Unit) の機能です。ハイパーバイザと呼ばれる仮想化の技術を用い、ゲストの OS を複数実行させることが可能です (図2)。また、QEMU と組み合わせて使うことにより、QEMU 単体で動作させるよりも高速に動作する仮想環境を提供することができます。

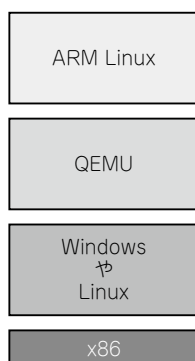


図1 QEMUの基本構成

x86上でARMプラットフォームを仮想化する。

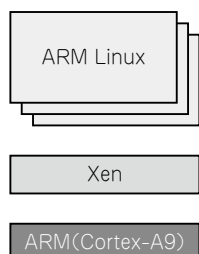


図2 Xenの基本構成

ARMプラットフォームで複数のOSを仮想化することができます。

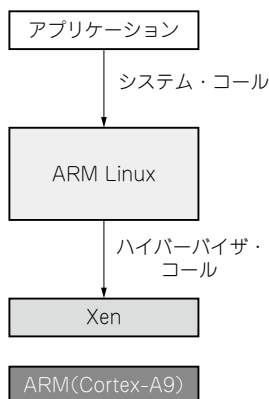


図3 準仮想化(PV)

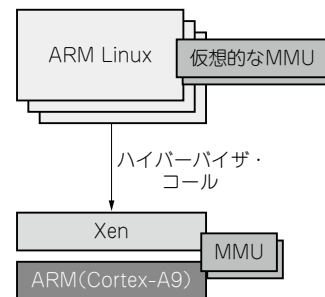


図4 MMUの個数

各ゲストOS(Linux)ごとに仮想的なMMUをもつが、実際にはコアの数しかない

2 Xenの特徴

Xenがエミュレーションするのは、MMUを含めたOSのためのハードウェア環境です。Xenではその方式として、準仮想化方式(Para Virtualization)と完全仮想化方式の2種類の方法を用意しています。

● 準仮想化(PV)

準仮想化(PV)におけるXenは、OSにとってのさらに上のOSを管理する超OS的な存在(ハイパーバイザ)です。アプリケーションから見ると、OSの上にOSがあるように見えるでしょう(図3)。OSにとって必要な機能は、ハイパーバイザ・コールによってXenに要求されます。例えば、ページの属性の管理やメモリの要求といった、通常はOSが直接(CPUが持つ)MMUの機能を制御する処理を、ハイパーバイザへ依頼します。ハイパーバイザは、これらの機能を