Intel® VT vs AMD AMD-V
ハードウェア仮想化　徹底比較

株式会社びぎねっと
伊藤　宏通

Begi:net

ハードウェア仮想化とは

- ハードウェアによる仮想化支援機能では、タイトルとして長すぎたので短縮
- ハードウェア(主にCPU)に仮想化を支援する機能を追加
- 追加された機能を使うことで、仮想化を実現するソフトウェアの実装がシンプルになる

実装一覧

- IA-32
  - Intel VT-x
  - AMD Virtualization(AMD-V)
- IA-64
  - Intel VT-i
- UltraSPARC
  - UltraSPARCArchitecture2005(UltraSPARC T1)
- POWER
  - Logical Partitioning (LPAR)
- ARM
  - TrustZone

メリット

- 完全仮想化の実現が簡単になる
  - x86では特に
- オーバーヘッドが少なくなる

Intel VT とは

- Intel® Virtualization Technology の略称
  - 開発コード"Vanderpool Technology”
  - IA-32用のVT-x
  - IA-64用のVT-i
  - I/O仮想化のVT-d
    - Virtualization Technology for Directed I/O

Intel VT-x

- 動作モードが追加
  - VMXモード
- 制御構造が追加
  - virtual-machine control structure(VMCS)
- 10個の命令が追加 (VMX Instruction Set)
  - VMXモードを管理するする命令が5つ
  - VMCSを管理する命令が5つ
- 制御用レジスタが拡張
  - CR4レジスタ
  - MSR(Model Specific Register: モデル別レジスタ)

VMXモードとは

- リングと関係を持たない新たなモード
  - リング0より一段高い特権レベルのような働きをする
- VMXモードには2つの種類がある
  - VMX rootモード（VMMが動作）
  - VMX non-rootモード（仮想マシンが動作）
  - ２つのモード間を移行することをVMX transitionsと呼ぶ

VMX transitions

- VMX transitionsにも2つの種類がある
  - VM entries
    - VMX rootモードからVMX non-rootモードに移行すること
  - VM exits
    - VMX non-rootモードからVMX rootモードに移行すること

VMX rootモード

- 従来のプロセッサとほぼ同じ動作をするモード
- 相違点
  - VMX命令が利用できること
  - VMX関連の制御レジスタの操作が一部制限されている

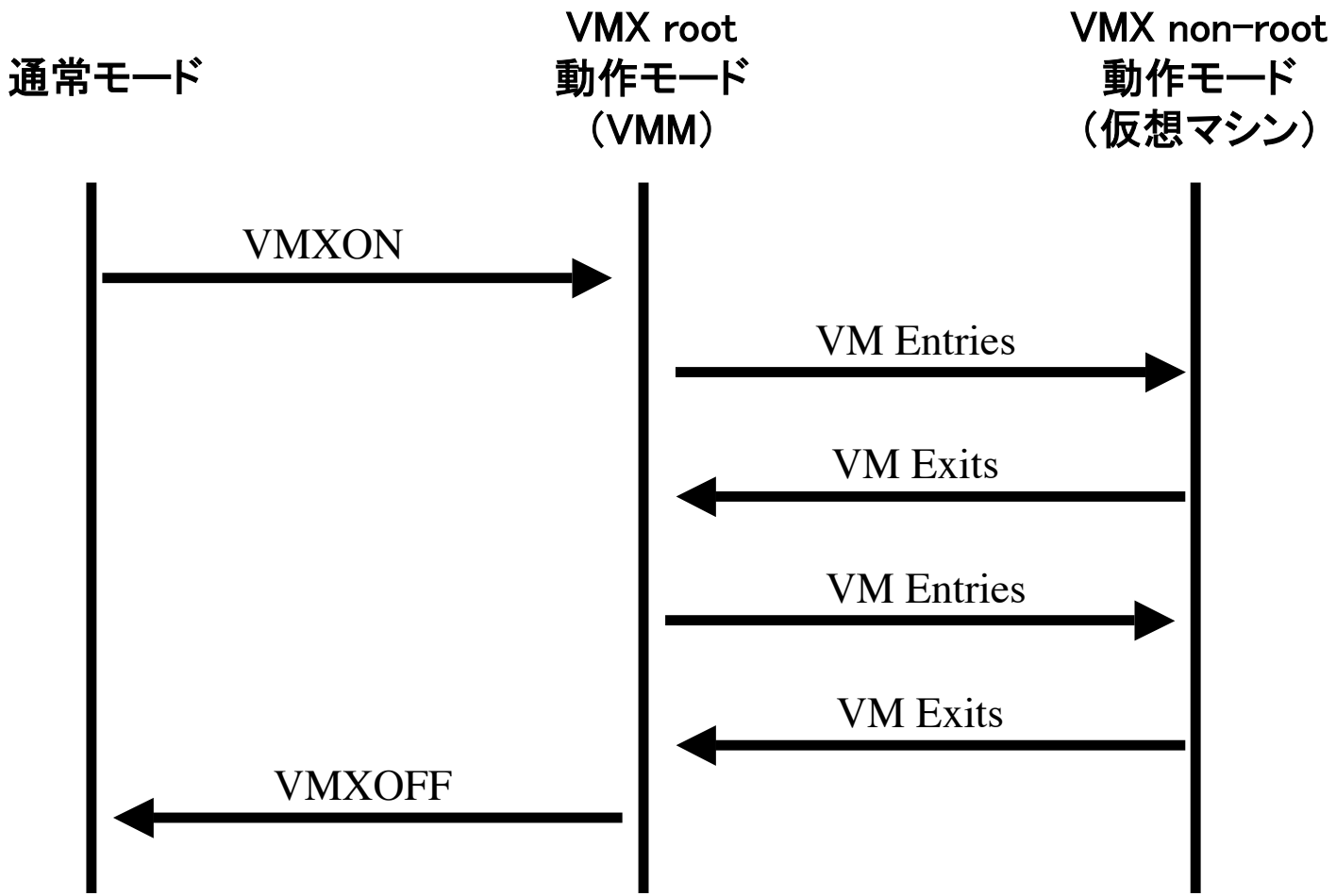
VMX non-rootモード

- 仮想化をシンプルにするために動作に制限や変更が加えられている
  - 仮想化に影響のある命令やイベントはVM exitsを引き起こすようになっている
    - VM exits発生
    - VMX rootモードのVMMに制御が移行
    - VMMがVMCSの情報を元にVM exitsの原因を調べて、適切な処理を行う
    - VMMがVMX non-rootモードに戻す

VMX non-rootモード（続き）

- このモード上で動作する仮想マシンは、リングを利用しなくても制御できる
  - Ring Aliasing(リングの付け替え)を行う必要がなくなる
  - Xen用にOS修正する必要がなくなる
- VMMからリング関連の処理を省くことができる
  - 実装が大幅に簡素化される

モードの遷移



制御構造VMCSとは

- VMX non-rootモードとVMX tansitionsを制御するためのデータ構造
  - VMMはこれを利用して仮想マシンを制御する
  - 現時点ではVMCS regionと呼ばれる4KBのメモリ領域を利用（将来拡張される模様）
  - 6つの論理グループに分けられている
    - Guest-state エリア
    - Host-stateエリア
    - VM-execution 制御フィールド
    - VM-exit 制御フィールド
    - VM-entry 制御フィールド
    - VM-exit 情報フィールド

制御構造VMCSとは（続き）

- Guest-state エリア
  - VM exits時にVMX non-rootモードで動作していたプロセッサの状態を保存
  - VM entry時に、このエリアに保持されている状態がプロセッサに読み込まれる
- Host-stateエリア
  - VM exits時に、このエリアに保持されている状態がプロセッサに読み込まれる

制御構造VMCSとは（続き）

- VM-execution 制御フィールド
  - VMX non-rootモードにおけるプロセッサの動作を制御するフィールド
- VM-exit 制御フィールド
  - VM exitsを制御するフィールド
- VM-entry 制御フィールド
  - VM entriesを制御するフィールド
- VM-exit 情報フィールド
  - VM exitsが起きた原因や種類の情報が格納されるフィールド

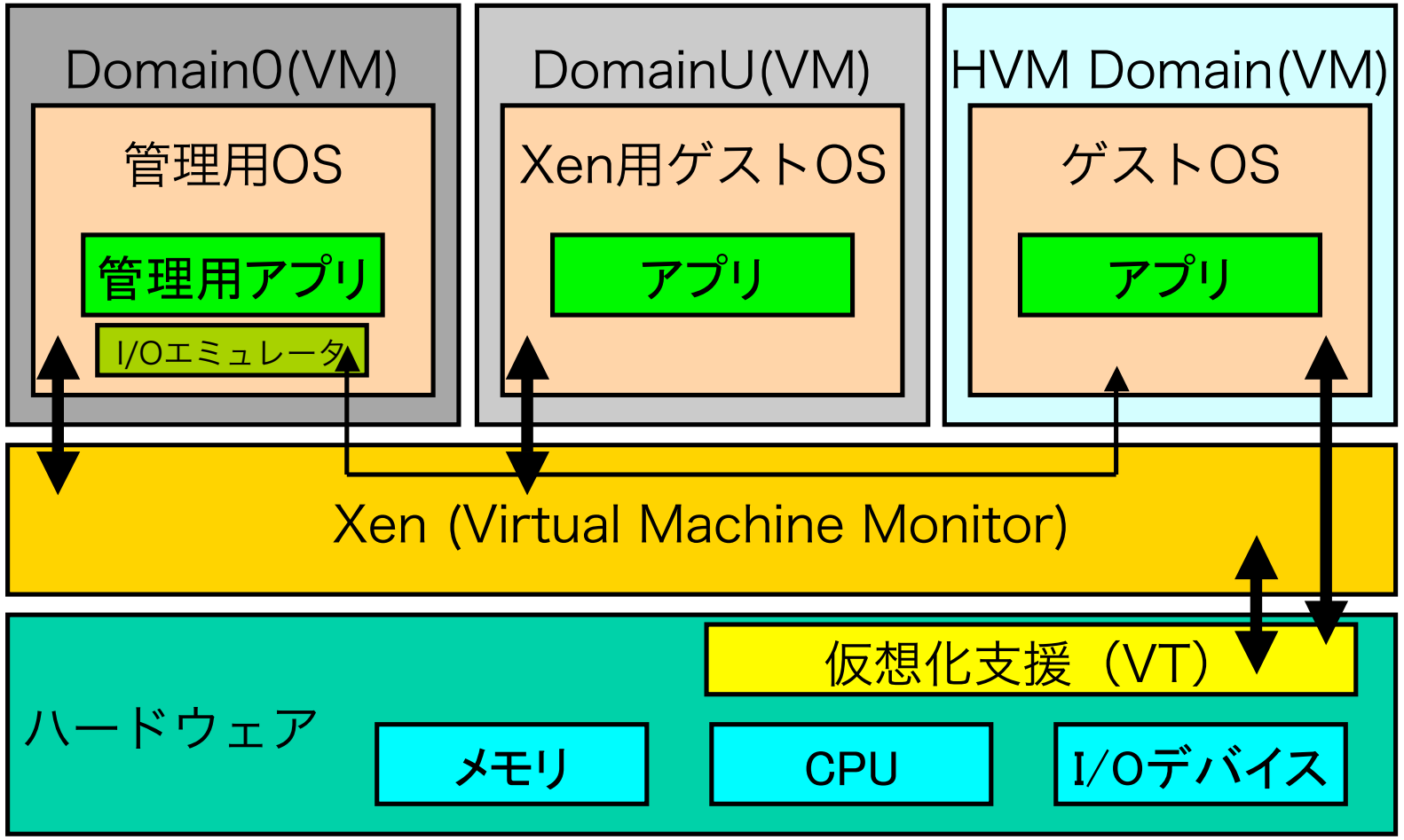
追加される10個の命令

- VMCSを管理する命令
  - VMPTRLD
    - VMCS用に確保したメモリをプロセッサに読み込ませる命令
  - VMPTRST
    - 現在のVMCSの内容をメモリに保存させる命令
  - VMCLEAR
    - VMCS用に確保したメモリの初期化を行う命令
  - VMREAD
    - VMCSのデータを読み込む命令
  - VMWRITE
    - VMCSにデータを書き込む命令

追加される10個の命令（続き）

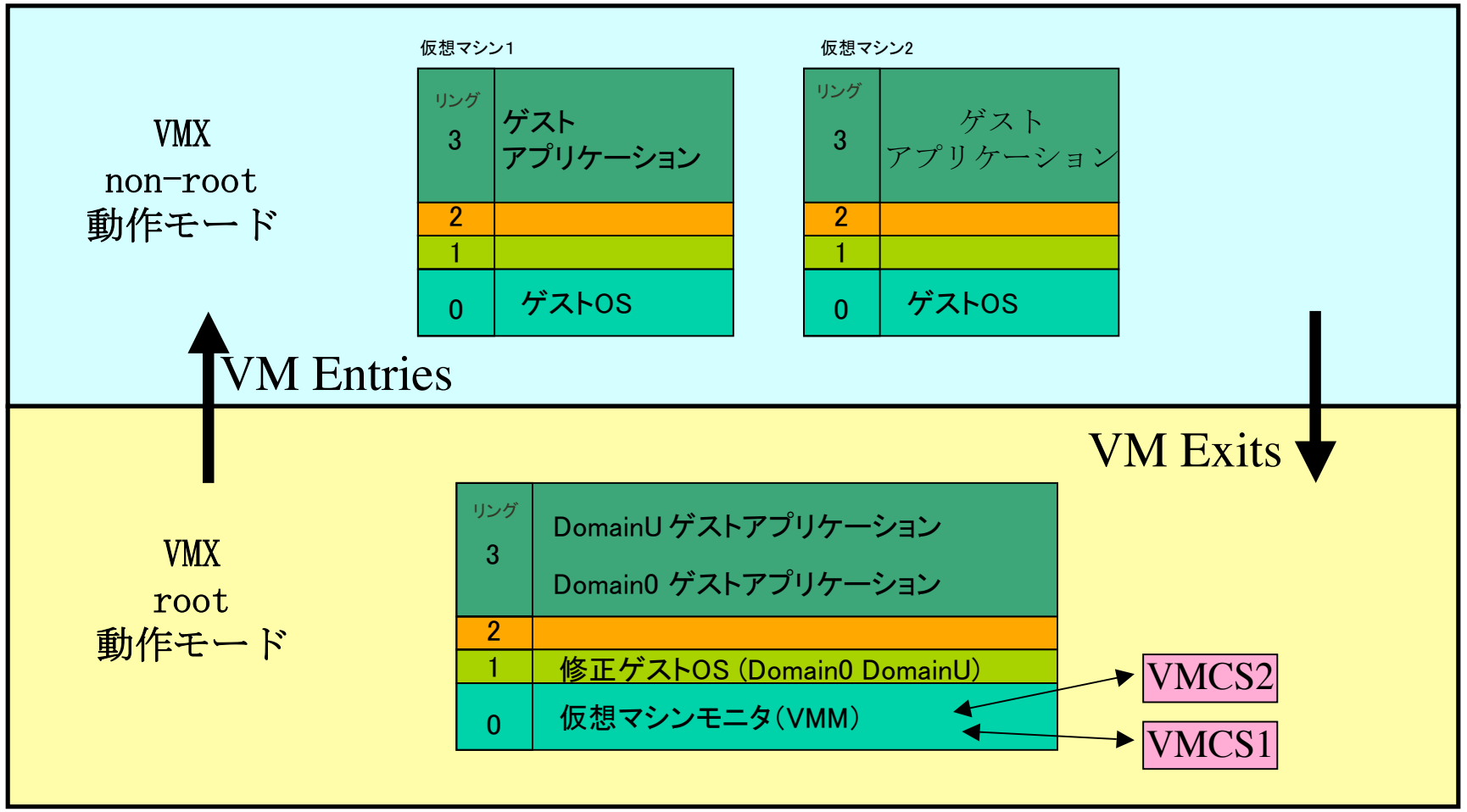
- VMXモードを管理する命令
  - VMCALL
    - VMX non-rootモードからVMMを呼び出す時に使う
      - （使用するとVM exitsが発生してVMMに制御が移行する）
  - VMLAUNCH
    - 仮想マシン（論理プロセッサ）を起動する
  - VMRESUME
    - 仮想マシン（論理プロセッサ）を再開する
  - VMXOFF
    - VMXモードから抜ける
  - VMXON
    - VMXモードに入る

Xen+VTの構造



Begi:net

Xen+VTにおける動作モード



Begi:net

AMD-V とは

- AMD Virtualizationの略称
- AMD SVM(Security and Virtual Machine architecture)がドキュメントでは使われている
- 開発コード ” Pacifica”
- I/O仮想化のAMD IOMMU(AMD I/O Virtualization Technology)

AMD-Vとは(続き)

- 下記の2つの機能から構成されている。
  - 仮想化サポート(Virtualization Support)
  - セキュリティサポート(Security Support)
- 動作モードが追加
  - Guestモード
- 制御構造
  - Virtual Machine Control Block(VMCB)
- 9個の命令が追加 （SVM instruction set)
  - 仮想化サポート用が8つ
  - セキュリティサポート用が1つ
- 制御用レジスタが拡張

AMD-Vとは(続き)

- LocalAPICを使う仮想化された割り込みの実装をサポート
- 外部アクセス（DMA）からVMMや仮想マシンのメモリを保護
- タグ付けされたTLB（tagged TLB）を持つ
- Nested Paging機能を持つ
  - Intelも同様の機能EPT（Extended Page Tables）を追加予定

Guestモード

- VT-xのVMX non-root operation モードに相当するもの
- SVMでは、「Guest Mode」からその他のモードに遷移することをVMEXITと呼んでいる。
- このモード内では、仮想化に影響のある命令の振る舞いが変更されている。
- このモードには、予めVMCBを作成してVMRUN命令で入ることができる。

Guestモード(続き)

- このモード内で、VMMによりVMCB上に設定されたトリガーとなるイベント、例えば指定された命令が実行された場合や例外などが発生した場合、VMEXITが発生して制御をVMMに移行する。
- VMEXITが発生した場合、VMCB上にその原因が記録されているので、それを元にVMMは、適切な処理を行うことができる。

VMCB

- VT-xのVMCS（Virtual-Machine Control data Structure）に相当するものである。
- VMCBには、4KBでアライメントされたメモリ領域を適切なサイズ（現在の実装ではページサイズである4KB）で割り当てる
- VMCBは、２つの論理エリアに分かれていて１つは、Control Area 2つ目は、State Save Areaと呼ばれている。

Control Area

- その名の通り、様々な仮想マシンを制御するために必要な情報で構成されている。
- どのようなトリガーとなるイベントでVMEXITが発生するのかを設定したり、VMEXITが発生した原因などが記録される。

State Save Area

- State Save Areaは、仮想マシンの状態を保存しておくために必要な情報で構成されている。

SVM instruction set

- VMRUN
  - Guest Modeを開始して仮想マシンを動作させる
- VMSAVE
  - 仮想マシンのプロセッサの状態をVMCBに保存する
- VMLOAD
  - 仮想マシンのプロセッサの状態をVMCBから読み込む
- VMMCALL
  - 仮想マシンからVMMを呼び出す
- STGI
  - グローバル割り込みフラグをセットする

SVM instruction set

- CLGI
  - グローバル割り込みフラグをクリアする
- INVLPGA
  - ASIDで指定したTLBのエントリを無効にする
- MOV (CRn)
  - CR8を含むコントロールレジスタを読み書きできるようにする
- SKINIT
  - セキュリティを考慮した方法で、Security Kernelを起動する

VT-x と AMD-V の違い

- 「The architectures are from 10,000 ft. view they are very similar Each has their own advantages/disadvantages 」

  米IBM Leendertvan Doorn Xen開発者

  直訳

  10,000フィート上空から見れば両者はとても似ている。両者ともそれぞれ得意不得意な部分を持っている。

類似点

- 動作モードが追加されている
  - VMXモードと Guestモード
- 制御構造が追加されている
  - VMCSと VMCB
- 命令が追加されている
- 制御用レジスタが拡張されている

相違点

- 動作モードの遷移の仕方が違う
- 制御構造の中身が違う
- 命令が違う
- 拡張されたレジスタが違う
- メモリ管理が違う

Xenでの実装

- 両者の違いを吸収するレイヤがある
  - Hardware Virtual Machine Abstraction Layer
  - HVM
- このレイヤの名前から完全仮想化された仮想マシン（ドメイン）はHVMドメインと呼ばれる

制御構造の違い

- VT-xのVMCSには、ホスト（VMM）側の状態を保存しておく領域がある

- SVMのVMCBにはなく、SVMの制御レジスタに状態を保存するメモリ領域（4KBにアラインメントされた適切なサイズ）のアドレスを登録しておくことで、VMRUN命令が実行された時点のホスト側の状態を保存する

命令の違い

- 相当する命令が存在するが異なる

SVM	VT-x
VMRUN	VMLAUNCH、 VMRESUME
VMSAVE、VMLOAD	VMWRITE、VMREAD、 VMPTRLD、VMPTRST、 VMCLEAR
VMMCALL	VMCALL

命令の違い(続き)

- 相当する命令が存在しない
  - STGIとCLGI
    - 仮想マシンを起動もしくは再開させる処理を行っている間の割り込み処理を適切に行うための命令(VT-xには不要)
  - INVLPGA
    - SVMの特徴であるタグ付けされたTLBに関連する命令
  - MOV(CRn)
    - 制御レジスタの読み込み書き込みを高速化する命令
  - SKINIT
    - TPMなどの高度のセキュリティを要求されるプログラムを安全に読み込んで起動することをサポートするための命令

メモリ管理の違い

- AMD-Vは、タグ付けされたTLB（tagged TLB）を持つ
  - ページテーブル
    - 仮想アドレスのページ番号と実アドレスのページ番号を結びつけたページテーブルエントリ(PTE)から構成された対応表
  - TLBとは
    - TLB（Translation Lookaside Buffer）と呼ばれるPTE専用のキャッシュメモリです。
  - タグとは
    - AMD-V を搭載したプロセッサは、ASID(Address Space IDentifier)と呼ばれる、アドレス空間を識別する識別子を持っている。このASIDを使ってTLBにタグ付けすることができる。

メモリ管理の違い（続き）

- タグ付けされたTLB（tagged TLB）のメリット
  - 仮想マシンのVMCBに、それぞれ1つのASIDに結び付ける
  - 仮想マシン上で動作するOSがTLBに対するフラッシュ（初期化）を要求してきた際に、ASIDを参照することで、その仮想マシンに割り当てられたTLBだけをフラッシュにすることができる。
  - TLBのフラッシュ回数を減らせるため、TLBミスの発生が抑制される
  - 性能が向上する

課題

- I/Oが遅い
  - VT-d や AMD IOMMUは来年
  - パラバーチャルドライバは実装作業中
- マイグレーション関連の実装がまだ
  - 開発ML上で進行中
  - メモリ管理手法の変更が提案されている
- まだ安定していない
  - もう少し時間が必要

パラバーチャルドライバとは

- ゲストOS用に作成された仮想マシン環境専用のドライバ
- インストールすることで、仮想化のオーバーヘッドを軽減することができる
- VMwareのVMtoolsに含まれるドライバもパラバーチャルドライバ
- ドライバなのでOS本体を修正する必要はない(Windows用のドライバも作成中？)

I/O 仮想化支援とは

- 仮想マシンから直接I/Oにアクセス可能な仕組みを提供する
  - 他の仮想マシンやVMMに影響を与えないための仕組みが必要
  - 策定中のPCI-Expressの新しい規格に準拠したデバイスを利用すると、複数の仮想マシンで1つのデバイスを共有することも可能
    - PCI-Express IOV

デモ

- AMD SVM上で動作する
  - Windows
  - Linux