Technology Consulting Company
Research, Development &
Global Standard

BitVisor 2016年の主な変更点

榮樂 英樹
株式会社イーゲル

2016-11-30 BitVisor Summit 5

BitVisor 2016年の主な変更点

igel

- Intel GbE MSI + virtio-net MSI-X対応
- 性能改善
  - Thread, Nested Paging
- バグ修正
  - mm_lockレースコンディション、UEFIスタックアラインメント、ATAコマンドREAD LOG DMA EXT、Virtio-net割り込みステータス
- その他
  - iMac SDカードリーダー対応、iMac EFI variablesアクセスエラー修正、Intel最新CPU対応

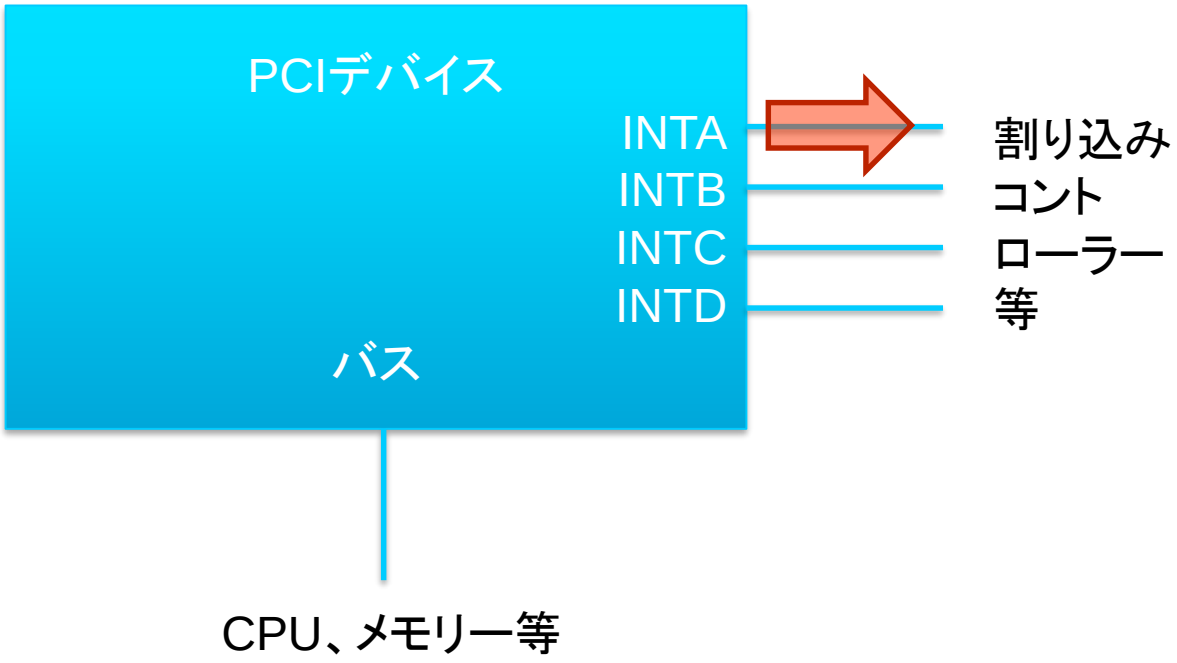
INTEL GBE MSI + VIRTIO-NET MSI-X対応

背景: PCIeデバイスの割り込み

- INTx (PCI互換)
- MSI
- MSI-X

背景: PCIeデバイスの割り込み INTx (PCI互換)

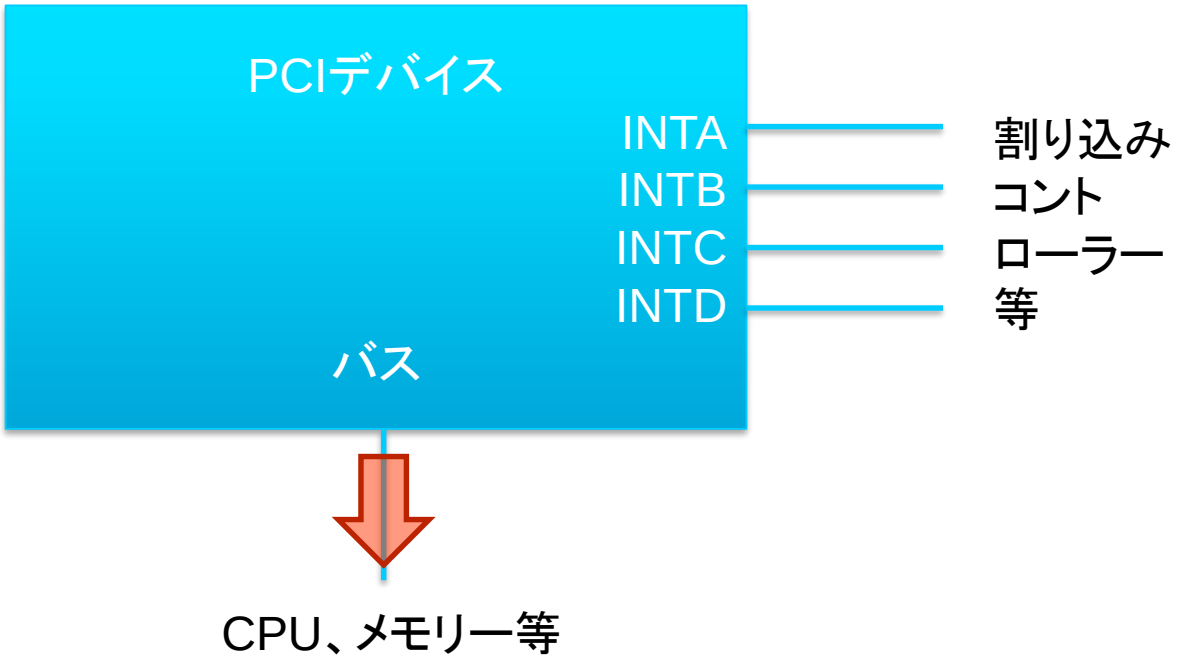
- PCIでは4本の割り込み信号線を使用
- PCIeではINTxに相当するメッセージを使用



背景: PCIeデバイスの割り込み MSIおよびMSI-X

igel

- メモリー書き込みによって割り込みを生成



背景: PCIeデバイスの割り込み MSIおよびMSI-X

MSI (Message Signaled Interrupt)

- アドレスおよび内容はconfiguration registerで指定
- 各デバイスは最大32種類のメッセージを使用可能

MSI-X

- MSIの拡張で、アドレスおよび内容をベースアドレスレジスターで示されたメモリー領域で指定
- 各デバイスは最大2048種類のメッセージを使用可能で、内容は個別に設定可能

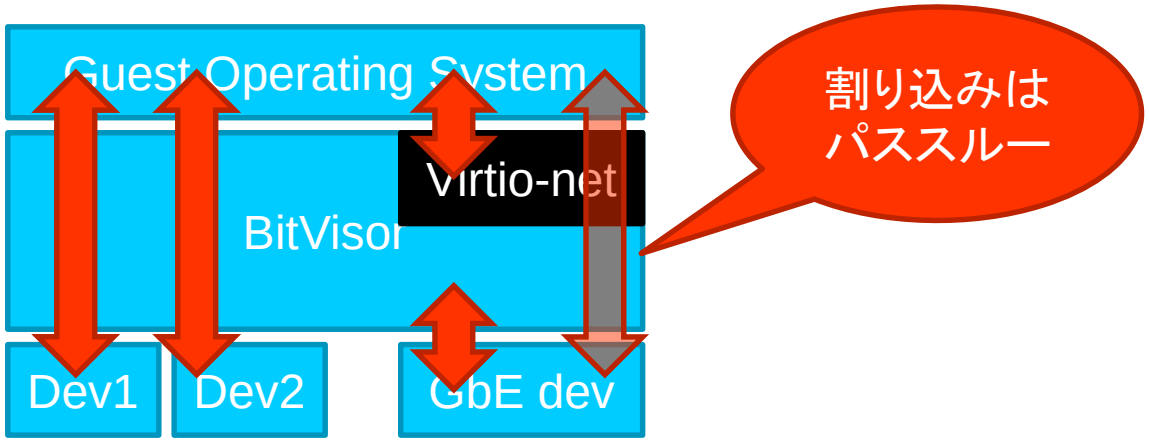
背景: PCIeデバイスの割り込みの組み合わせ

igel

- PCIeの仕様上の組み合わせ
  - INTx + MSI
  - INTx + MSI-X
  - INTx + MSI + MSI-X

- 例
  - Virtio-net: INTx + MSI-X
  - Intel 82572EI: INTx + MSI
  - Intel 82574L: INTx + MSI + MSI-X
  - Intel I219-LM: INTx (?) + MSI

背景: BitVisorのvirtio-net実装 (BroadcomおよびIntel GbE用)



- ネットワークデバイスのIDやベースアドレスレジスターを偽装、ゲストOSにはvirtio-netデバイスとして見せる
- 割り込みはINTxのみ対応として見せて、実際のネットワークデバイスのINTxをそのまま流用する
  - PCIeでINTxのみ対応のデバイスはありえないが、Windows, LinuxおよびmacOSは特に問題なく動作する

Intel I219-LM (Skylake世代)

igel

■ INTxによる割り込みが発生しない問題

	BitVisor virtio-net	Linux pci=nomsi
F社製PC	発生しない	発生する
D社製PC	発生しない	**発生しない**

➡ **INTxは使えない**

MSIを使用すれば問題ない (I219-LMはMSI-Xには未対応)

➡ **MSI対応が必要**

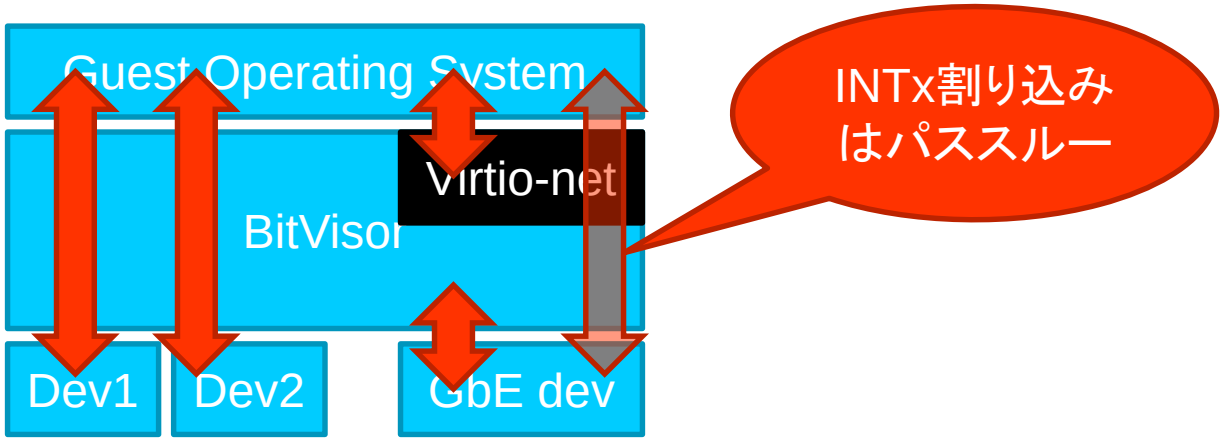
BitVisorのvirtio-net実装とMSI

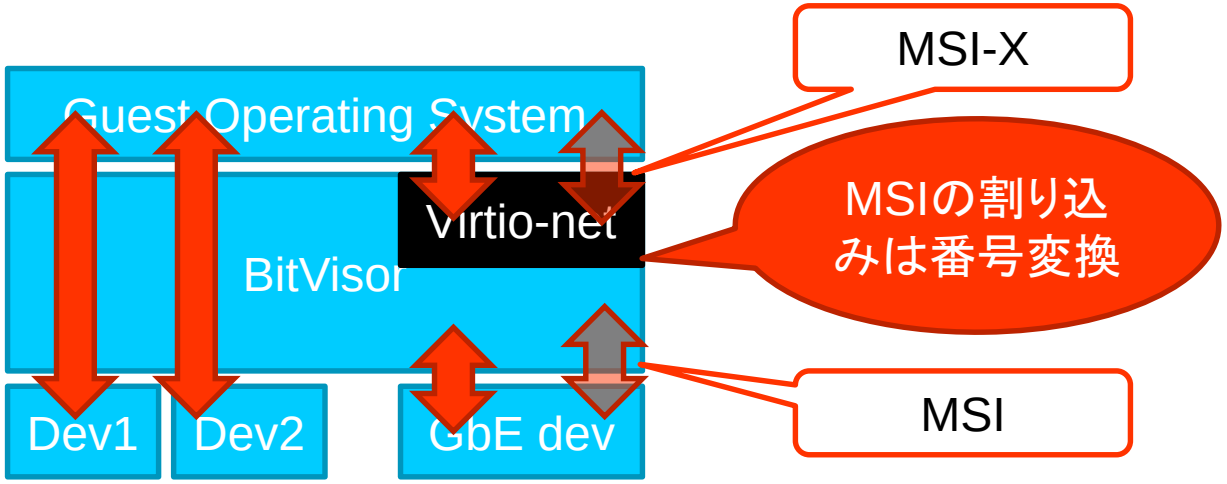
igel

- ネットワークデバイスをMSIで動作させたときに、ゲストOSに見せるvirtio-netデバイスの割り込みをどうするか?
  - INTx: MSIから割り込み生成ができない
  - MSI: Virtio-netの仕様にMSIのことが書かれていないためどのように実装すべきかわからない
  - MSI-X: 送信・受信それぞれに割り込みを個別に割り当てることができるため、MSIで生成された割り込みからの変換が必要

  ➡ MSI-Xが適切
- Virtio-net INTx ➡ ネットワークデバイスINTx (今まで通り)
- Virtio-net MSI-X ➡ ネットワークデバイスMSI、割り込み番号変換

BitVisorのvirtio-net実装の割り込み (MSI-X対応)





実装: MSI-Xテーブル用メモリー空間

igel

- 割り込みをどのように発生させるか、アドレスと書き込み内容を示すテーブルが、ベースアドレスレジスターの指すメモリー空間内に必要
- メモリー割り当ての単純化のため、Intel GbEデバイスのレジスターが割り当てられているメモリー空間をそのままMSI-Xテーブル用としてゲストOSに見せる
- ゲストOSのアクセスはMMIOフックによりBitVisorで管理する

実装: MSI用割り込み番号 (VMMのみが使用)

igel

Intel GbEデバイスのMSI用に割り込み番号を割り当てる

- 0x00-0x0FはAPIC・MSIの仕様により使用不可能
- 0x20-0xFFはゲストOSが外部割り込みに割り当てる可能性があり、VMMで識別できないと使用不可能
- 0x10-0x1Fを使用
  - この範囲はCPUの例外やBIOS呼び出し用のソフトウェア割り込みに使われており、割り込みハンドラーで外部割り込みかどうかの識別は困難なため通常使われない
  - VMMは#VMEXITの情報から例外・ソフトウェア割り込みと外部割り込みを区別することができる

実装: 割り込み番号変換

igel

- 外部割り込みパススルー処理で割り込み番号を取得したときに、0x10-0x1Fの範囲であれば登録されているコールバック関数を呼び出し、割り込み番号の変換を行う
- Virtio-netドライバーのコールバック関数内でMSI-Xテーブルを参照し割り込み番号を取得する
  - 本当はLocal APICを使用してIPIにより割り込み生成をするべきだと考えたが、うまくいかなかったため番号を変換して直接割り込みを生成している

現在のBitVisorのvirtio-net実装

igel

Virtio-netデバイスで利用可能な割り込み

- Broadcom GbE (bnx): INTxのみ
- Intel GbE MSI使用不可能: INTxのみ
- Intel GbE MSI使用可能: INTx + **MSI-X**

制約事項

- I219-LM環境ではpci=nomsiなどによりゲストOSがMSI-Xを使用しないようにすると割り込みが発生しない
- MSI-X使用時はconfig.vmm.no_intr_intercept=0でなければならない

性能改善

性能改善: Thread

igel

- 1.4までのspinlock版ではfairnessの問題があった
  - 特定のCPUがspinlockを高速に繰り返すと他のCPUが長時間待たされる ➡ scheduleを高速に繰り返すと他のCPUが止まる
- 以前のロックフリー版ではごくまれに原因不明のpanicが発生していた
  - "thread: td_runnable_put failure %u %u"
- 1.4までのspinlockをticketlockに変えて問題を修正できたが、性能は大きく低下した
- 性能改善のためのコンパイル時configとして **CONFIG_THREAD_1CPU**を導入した

CONFIG_THREAD_1CPU

igel

- CONFIG_THREAD_1CPU=0 [遅い] (デフォルト)
  各CPUでできるだけスレッドを実行する

CPU0 ➡	VM	★1	★2	★3	★VM	★2	★VM					
CPU1 ➡	VM							★1	★3	★VM		

- CONFIG_THREAD_1CPU=1 [速い]
  VM以外のスレッドをひとつのCPUで最後まで回す

CPU0 ➡	VM	★1	★2	★3	★VM			★VM				
CPU1 ➡	VM		★VM				★1	★2	★3	★VM		

★ schedule()呼び出し　　VM, 1, 2, 3: スレッド

性能改善: Nested Paging

igel

2MiBページによる性能改善

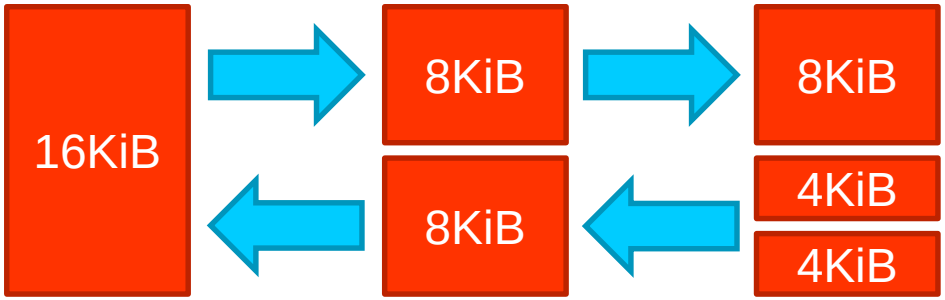
- 効果
  - EPT violation / NPF #VMEXITの回数削減
  - テーブルあふれ回数削減
- 2MiBページ単位用の以下の処理を追加
  - 仮想マシンの物理アドレスを実際の物理アドレスに変換
  - MMIOフック対象領域が含まれないことの確認
  - MTRRキャッシュ設定の複数領域にまたがっていないことの確認
- 上の処理のいずれかに失敗したら従来通り4KiBページを用いる

バグ修正

バグ修正: mm_lockレースコンディション

igel

- ページ単位のメモリー確保・解放処理にレースコンディションが存在した (初期のバージョンから)
  - 内部では2のべき乗単位のページ数のサイズ別リストで管理



  - 確保の際にPAGE_TYPE_FREE → PAGE_TYPE_ALLOCATEDの状態遷移をロック外で実行、解放の際に、隣接する同サイズの領域が解放済みであれば連結するという処理があり、その処理と競合していた
  - 隣接する同サイズの領域の解放が同時に行われない限り問題が表面化していなかった

バグ修正: UEFIスタックアラインメント

igel

- iMac 27-inch Retina 5K 2015において、UEFIによる文字表示 (ConOut->OutputString) 呼び出しの際、スタックポインターを16バイト境界に合わせておかないと、でハングアップする問題が発生

- 調査の結果、現在のUEFIの仕様に16バイトアラインメントの指定があるようなので修正

バグ修正: ATAコマンドREAD LOG DMA EXT

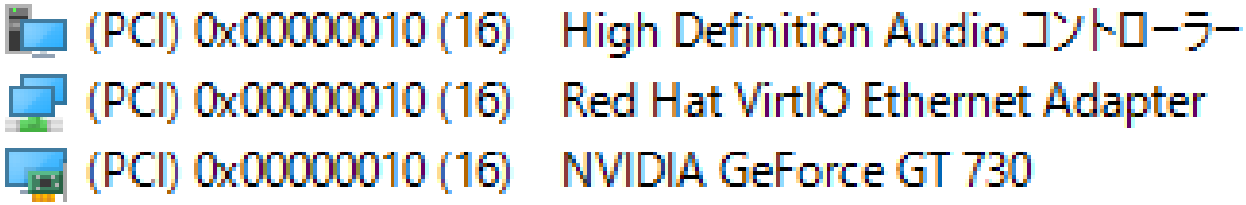
igel

- READ LOG EXTコマンドのパススルーのみ実装済みで、READ LOG DMA EXTコマンドのパススルーが入っていなかったため追加した

バグ修正: Virtio-net割り込みステータス

- 割り込みステータスが常に同じ値を返していた部分を修正した

- ネットワークデバイスとその他のPCIeデバイスが割り込み線を共有している場合の問題への対応



  - ネットワークデバイスが自身の割り込みだというステータスを返すことで、残りのデバイスドライバーの割り込み処理が行われない問題がWindowsで発生した
  - レベルトリガーの場合割り込みが発生し続けてハングアップしたような状態となる

その他

iMac SDカードリーダー対応

- ネットワークデバイスとSDカードリーダーがマルチファンクションデバイスとして実装されている
  - 03:00.**0** Ethernet controller [0200]: Broadcom Corporation NetXtreme BCM57766 Gigabit Ethernet PCIe [14e4:1686] (rev 01)
  - 03:00.**1** SD Host controller [0805]: Broadcom Corporation BCM57765/57785 SDXC/MMC Card Reader [14e4:16bc] (rev 01) (prog-if 01)
- Virtio-netをmultifunction=1で使用するとWindowsでBSoDが発生
- Virtio-netがPCIe capabilitiesを完全に隠していたのが原因、対応済み

iMac EFI variablesアクセスエラー修正

iMac 27-inch Retina 5K 2015においてBitVisor上でEFI variablesへのアクセスがエラーになる問題が発生

- SMIを使用する一部BIOS環境で発生していた問題と似ている
- メモリーマップでEfiMemoryMappedIOとされている領域を事前にマップしておくことで解決
- Nested Paging環境のみの対応

Intel最新CPU対応

igel

- x2APIC対応
  - L社製PCにおいてx2APICが積極的に使われるケースを確認
  - x2APICはMSRでのアクセスとなるため、対応していない旧バージョンではマルチコア開始を追跡できない
  - MSRはVMMで扱うにはMMIOより都合がいい
  - 対応はしたが動作未確認

- XSAVES/XRSTORS命令対応
  - F社製PCにおいてXSAVES/XRSTORS命令が使用されるケースを確認
  - パススルーで問題ないのでenableにするようにして対応

まとめ

igel

BitVisor 2016年の主な変更点

- Intel GbE MSI + virtio-net MSI-X対応
  - I219-LM対応、MSI - MSI-X割り込み番号変換
- 性能改善
  - Thread, Nested Paging
- バグ修正
  - mm_lockレースコンディション、UEFIスタックアラインメント、ATAコマンドREAD LOG DMA EXT、Virtio-net割り込みステータス
- その他
  - iMac SDカードリーダー対応、iMac EFI variablesアクセスエラー修正、Intel最新CPU対応

今後の予定

igel

- BitVisor 2.0？
  - Unsafe nested virtualization (AMD-V) (VT-x？)
  - AHCI piggyback改良
  - PCI改良
  - MMCONFIG対応
  - Virtio-net対応
  - Broadcom NetXtreme GbE driver追加
  - USB 3.0 XHCI対応
  - PCID対応 for Ubuntu 16.10