LinuxにおけるCPU/Memory/IOのHot Plugサポート

菅沼 公夫 (NEC Solutions (America), Inc.)
河内 隆仁 (NEC Solutions (America), Inc.)
青野 寛 (NEC コンピュータソフトウェア事業部)

概要

- 背景
- Hot Plugの活用方法
- CPU Hot Plug
- Memory Hot Plug
- ACPI
- IO Hot Plug
- まとめ

背景

□Hot Plugとは？

○システムを停止せずにデバイスを追加・削除する機能

□Hot Plugの目的

○システム運用中に故障部品の切り離し/交換が可能になり、可用性の向上に貢献

□なぜ NECが?

○RAS機能の充実はエンタープライズ用システムで重要

○Atlas Projectに参加し大規模システム向けLinux機能を開発

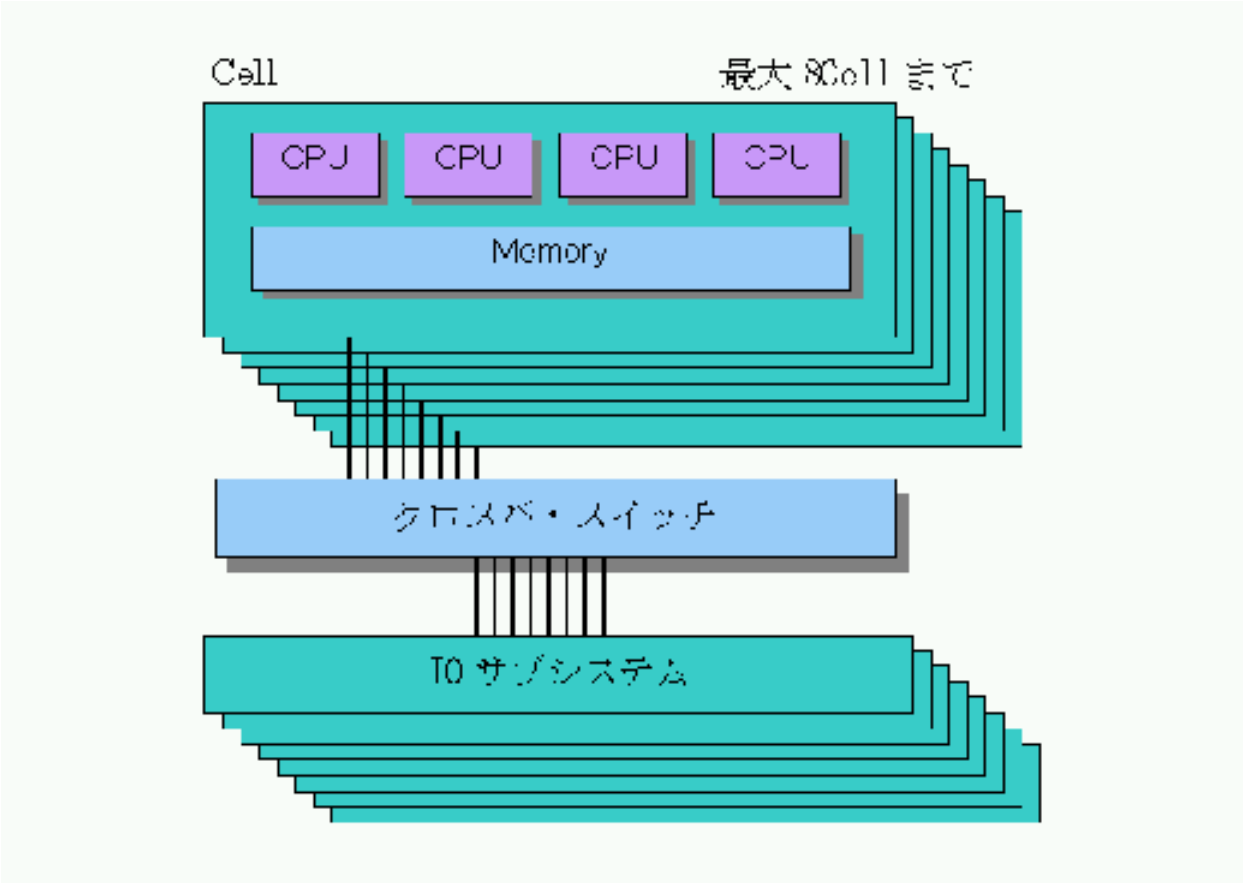
Atlas Project

- Static Domain Partitioning
  - SMPのシステム上で複数のOSを個別に動作可能に
- Machine Check Handling
  - ハードウェア障害のログ/リカバリ機能
- ccNUMA
  - ccNUMAシステムでの性能向上
- Hot Plug Processor and Memory Support
  - CPU/Memoryの Hot Plugサポート
- Hot Plug IO Node Support
  - PCI/IOノード Hot Plug
- ACPI 2.0 Support
  - ハードウェア構成情報を抽象化、種々の機能強化のための仕組

プラットフォーム

NEC TX7/i9000シリーズ

プラットフォーム(構成図)



Hot Plugの活用方法

Hot Plugの活用方法 (1)

□Hot Swap

○故障部品の切り離しによるシステムダウンの回避

○システムを停止せずに、部品の交換&組み込み

MCA(Machine Check Architecture)と連携することで高可用性を実現

□例

○CPU cache故障で 1bitエラー発生

○MCAによりログを採取

○故障した CPUをHot Remove

○CPUを交換

○交換した CPUをHot Add

Hot Plugの活用方法 (2)

□Hot Add

○システムを停止せずにリソースを追加

システム構成のフレキシブルな拡張を実現

□例

○必要最小限の規模で運用開始

○処理量の増加でMemory/Diskが枯渇

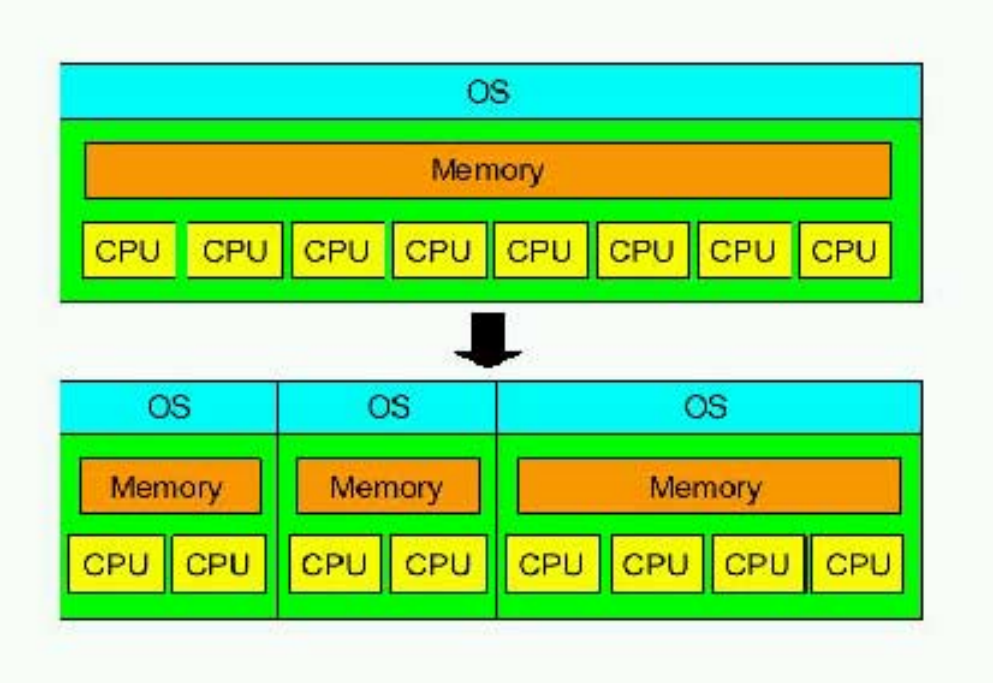
○Memory、SCSIカード、DISKを追加

○Memory Hot AddでMemoryを組み込み

○PCI Hot AddでSCSIカードを組み込み

Hot Plugの活用方法 (3)

□ダイナミック・パーティショニング



Hot Plugはシステムの仮想化技術の基盤

Hot Plugの活用方法 (4)

- □省電力
  - ○未使用CPUの電源断で省電力
- □swsusp(Software Suspend)での利用
  - ○Hot PlugでCPUを1つに減らし、SMP特有の問題を回避

多くの可能性を秘めている！

CPU Hot Plug

機能

- CPU Hot Add
  - CPUを運用中に初期化、システムに組み込む
- CPU Hot Remove
  - 運用中にCPUを停止

開発内容

- □CPU数不変の前提の撤廃
  - ○smp_num_cpusの撤廃、置換
  - ○各種マクロの提供
- □Hot Add
  - ○CPU初期化処理をシステム初期化処理から切り出し、運用中に実行可能にする
  - ○CPU起動処理の再構築
  - ○register、タイマー、IRQなどの初期化
  - ○idle/ksoftirqd/migration_threadなどのCPU毎に必要なカーネルスレッドの起動

開発内容 (cont.)

□Hot Remove

- ○CPU毎に所持する構造体・変数などの解放
- ○カーネルスレッドの停止
- ○IRQの設定変更
- ○停止CPUのrunqueue上のプロセスの移動
- ○停止CPUのcache flush、CPU HALT処理
- ○同期処理

□インターフェース

- ○ACPIとのインターフェース
- ○ユーザインターフェース

開発状況

- Rusty Russellを中心に数名のエンジニアと共同で作業中
- Hot Add/Hot Remove 機能を完全にサポート
- i386/Itanium/ppc/s390 をサポート
- 2.5カーネルに順次取り込まれている
- 2.4カーネル(Itanium版)は Atlas Projectの成果物として公開 (http://sourceforge.net/projects/atlas-64/)

Memory Hot Plug

機能

- Memory Hot Add
  - Memoryを運用中に追加
- Memory Hot Remove
  - Memoryを運用中に削除

開発状況

- 試作段階
- Memory Hot Addのみ作成
- discontigmemをベースに
- Itaniumのみサポート
- Cell単位でのHot Add
- SourceForge上で公開 (http://sourceforge.net/projects/lhms/)

課題

- discontigmemをベースにしてて良いのか？
- 仲間作り
- よりgenericに
- Hot Removeを実現させるには…

ACPI

ACPIとは

- APM(Advanced Power Management)の後継
- Advanced Configuration and Power Interface
- 電源管理/構成管理
- 中間言語の利用によるCPU非依存性
- 500ページの仕様書:(

ACPIとは (cont.)

□LinuxとACPIの関係

- ○2.4カーネルから利用可能
- ○2.5で新Driver Modelと統合、サスペンド等が現実的に
- ○http://sourceforge.net/projects/acpi

□Hot PlugとACPIの関係

- ○構成情報の取得
- ○IOに関する各種リソースの管理
- ○Hot Plug用標準メソッドの利用
- ○Hot Plugイベントの扱い

IO Hot Plug

IO Hot Plugとは

- □IOデバイスのHot Plug (USB,PC Card, …)
- □PCI Hot Plug
- □PCIホストバスブリッジのHot Plug

PCI Hot Plug

- □PCI Hot Plugコントローラが必要
  - ○Compaq
  - ○IBM
  - ○ACPI
- □PCI Hot Plugの機能
  - ○PCIカードの挿抜の検出
  - ○Hot Add時のPCIデバイスへのリソースの割当
  - ○ドライバのロード・アンロード
- □ユーザーインターフェース
  - ○pcihpfsファイルシステム

ACPIを利用したPCI Hot Plug

- ACPIからの構成情報を利用したPCIスロットの検出
- 挿抜イベントのハンドリング
- PCIデバイスに割り当てるのリソースの管理

- 同様にCPU/MemoryのHot Plugも実装可能

ACPI PCI Hot Plugの開発状況

- リリース済、2.4.20以降で利用可能
- Itanium/i386をサポート
- 2.5カーネルに移植中
- http://sourceforge.net/projects/pcihpd

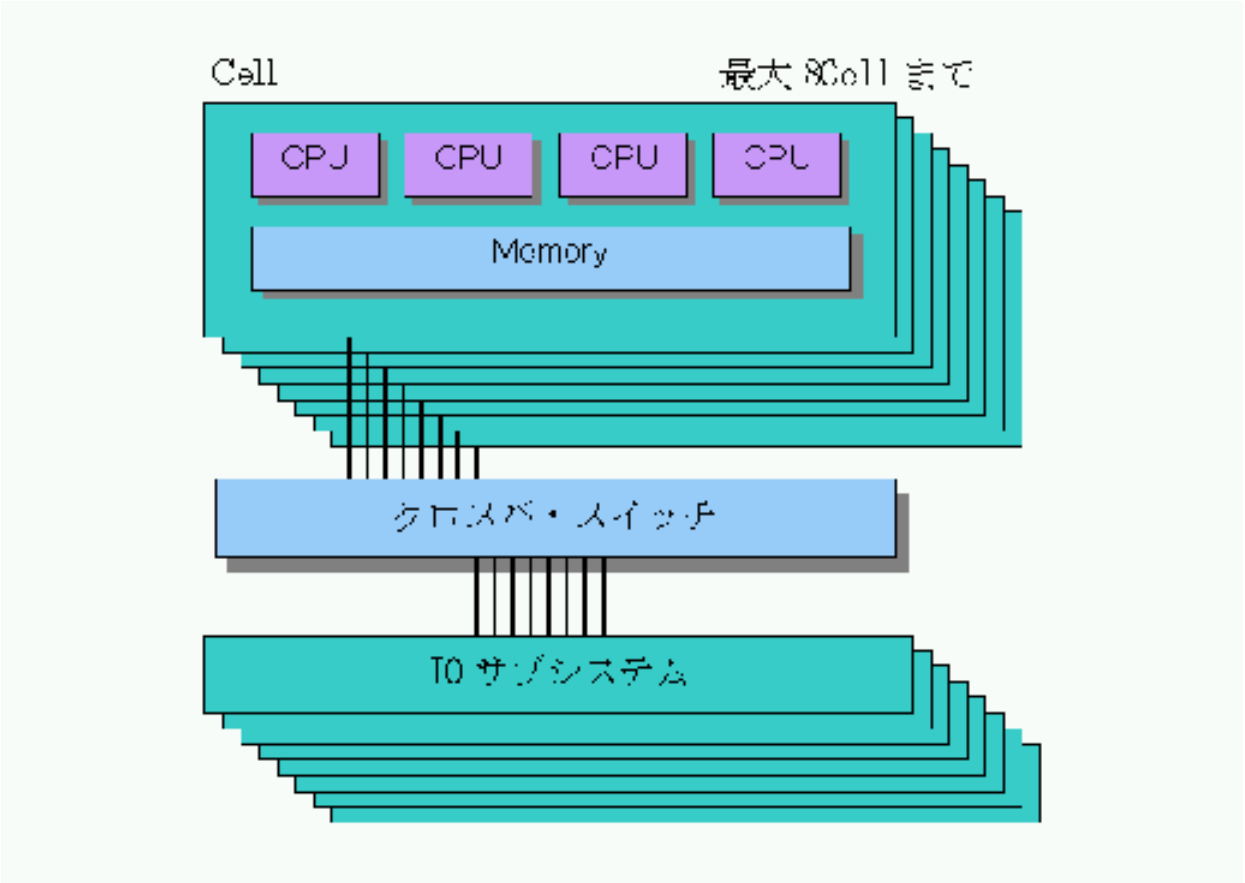
PCI Hot Plugの問題点

- □各デバイスドライバがHot Plug対応である必要あり
  - ○新しいPCI driver APIを利用する必要性
- □カードがHot Plugできても...
  - ○カードの位置は?
  - ○eth0, eth1...
  - ○umount?

PCIホストバスブリッジのHot Plug

- □ホストバスブリッジとは?
- □PCIバス全体のHot Plug
  - ○PCIホストバスブリッジ単位でのHot Plug
- □PCIカードのHot Plugに加えて...
  - ○チップセットデバイスのHot Plug
  - ○割り込みコントローラのHot Plug

プラットフォーム(構成図)



PCIホストバスブリッジのHot Plug(cont.)

```
Bus  1, device   0, function  0:
  PIC: PCI device 1033:00af (NEC Corporation) (rev 0).
    Master Capable.  No bursts.  Min Gnt=8.Max Lat=8.
Bus  1, device   2, function  0:
  SCSI storage controller: QLogic Corp. QLA12160 (rev 6).
    IRQ 52.
    Master Capable.  Latency=64.  Min Gnt=64.
    I/O at 0x1300 [0x13ff].
    Non-prefetchable 32 bit memory at 0xf7fff000 [0xf7ffffff].
Bus  2, device   4, function  0:
  SCSI storage controller: QLogic Corp. QLA2200 (rev 5).
    IRQ 53.
    Master Capable.  Latency=64.  Min Gnt=64.
    I/O at 0x1400 [0x14ff].
    Non-prefetchable 32 bit memory at 0xf7bff000 [0xf7bfffff].
```

□現在の状況～実装作業中

まとめ

- □CPU Hot Plug
  - ○2.5カーネルに採用され 2.6カーネルで実用化
- □Memory Hot Plug
  - ○Hot Addのみ試作済
- □IO Hot Plug
  - ○PCI Hot Plugは 2.4カーネルに採用済
  - ○ホストバスブリッジHot Plugは 2.5での採用を目指し開発中
- □ACPIインタフェース
  - ○各Hot Plug用のドライバを開発中

課題

- 2.5カーネル向けのパッチの提出
- 他のRAS機能との連携
- 他アーキテクチャへの展開

関連URL

- Atlas Project
  http://sourceforge.net/projects/atlas-64
- Linux CPU Hotplug Support
  http://sourceforge.net/projects/lhcs
- Linux Memory Hotplug Support
  http://sourceforge.net/projects/lhms
- PCI Hot Plug for Linux
  http://sourceforge.net/projects/pcihpd
- ACPI
  http://sourceforge.net/projects/acpi

Thank You!