ARM Hibernation調査（前半部分）

@sabakawa

発表の主旨と謝意

- 今回の発表内容はAndroidである必然性はありません。Linuxカーネルのメモリ管理に関する内容となります。
- 現在でも調査・開発中です。（モノはできていません）この発表で同様の話題で会話できたら良いな、という思いがあり強引にAndroidに結び付けました。
- 本日、発表する機会をいただき、ありがとうございます。

最終形は組込み機器の待機電力ゼロ化

- ノートパソコンは残電力が少なくなるとハイバネーション機能が起動して使用状態をハードディスクに残して、電源が復旧した後に使用状態も復旧します。

- （主として）スマートフォンはハイバネーションが存在しません。存在しない背景として様々な理由はあると思いますが、個人的には必須な機能の一つです。

オマケとしてARM Androidの高速起動

- 最近はこちらの方が目立っているらしいです。
  以下にソリューション化している会社さんを記載します。
  - ユビキタスさんのQuickBoot
  - リネオさんのWarp!!

技術的なお話：仮想メモリの抽象モデル

- 物理メモリはLinuxカーネル内で抽象化されて管理されています。（埋め込みスライド参照）

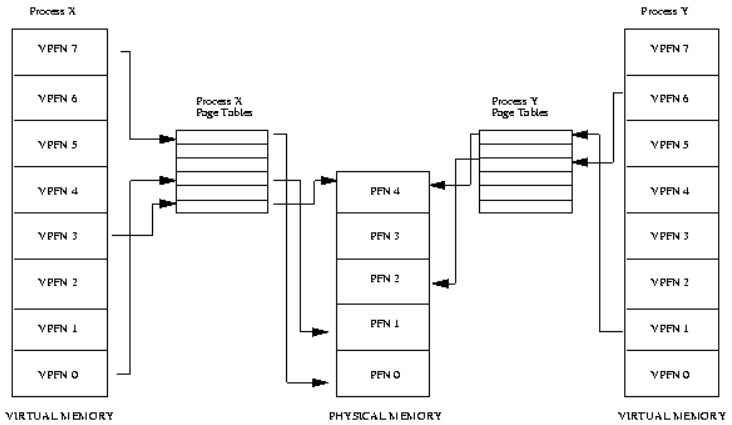


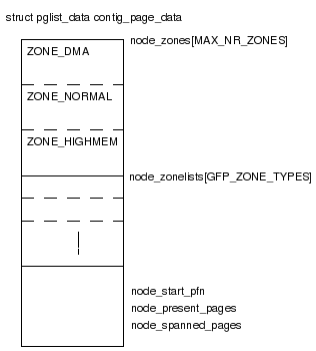
図1 仮想アドレスから物理アドレスへのマッピングに関する抽象モデル

出所：http://archive.linux.or.jp/JF/JFdocs/The-Linux-Kernel-4.html

- 非常に簡単に述べるとハイバネーションとは、この物理メモリと内部レジスタをストレージに残して再仮想化を行うことを指します。

技術的なお話：ZONE構造体

- ZONEという概念　linux v2.4から導入された概念
  メモリの断続的な実装が発生したため、ゾーン分けして管理するようにした。



ZONE_DMA
先頭から16MBの物理ページ。デバイスのDMAなどで使用される。DMAは低位の物理アドレスしか使えないデバイスがあるので物理メモリ先頭に配置されている。

ZONE_NORMAL
LowMemory領域の物理ページ。
ZONE_DMA以降の880MBがZONE_NORMALとなる。この領域はカーネルのアドレス空間から物理ページへストレートにマッピングされている。

ZONE_HIGHMEM
HighMemory領域の物理ページ。
ZONE_NORMAL以降のページ(先頭から896MB以降)。
HighMemoryはユーザープロセスのデータやPageCacheに使われる。

図2　ZONE説明（x86アーキテクチャの場合）

出所：http://wiki.bit-hive.com/linuxkernelmemo/pg/%CA%AA%CD%FD%A5%DA%A1%BC%A5%B8%B4%C9%CD%FD

- 歴史的な背景はパソコンの世界で（Windows2000が普及したあたりから？）、大容量のメモリを使い出して、追従するようにLinuxカーネルも大容量対応となった。（と、思う）

技術的なお話：保存メモリの最適化

- 使用中のページだけ保存します。
  - 保存先ストレージの最小化
  - 保存時間／再開時間の最小化（この辺りが高速起動に寄与）

具体的にはZONEの各free_areaに管理されているPFNに印をつけて保存の対象に入れないようにします。以下のような感じ

```
for (order = MAX_ORDER - 1; order >= 0; --order) {
    for(type = 0; type < MIGRATE_TYPES; type++) {
                list_for_each(curr, &zone->free_area[order].free_list[type]) {
                start_pfn = page_to_pfn(list_entry(curr, struct page, lru));
                for (i = 0; i < (1UL << order); i++) {
                    SetPageNosaveFree(pfn_to_page(start_pfn + i));
                }
    }
}
```

技術的なお話：多様な保存先ストレージ

- 保存対象が判明したらストレージに保存しますが、ここから組込み的なバリエーションが発生します。
  - PC（x86）なら保存先はIDEインタフェース先ということなります。開発者はIDEを意識したら良いということになります。
  - 組込み機器（例えばAndroid搭載の組込み機器）の場合、NAND/NOR FLASH、PCMCIA先のストレージ、IDEインタフェース先のハードディスクなど様々なドライバの開発が必要です。

Thank you for listening !

本日の発表は保存までの内容となります。
次回は再開処理が出来たらご報告したいと思います。

ご清聴ありがとうございました。