

Mintオペレーティングシステムにおける 高速で柔軟な起動方式に関する研究

中原 大貴

岡山大学大学院 自然科学研究科

電子情報システム工学専攻

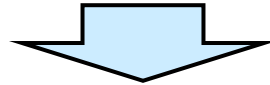
平成24年2月16日

はじめに

ハードウェアは高性能化が進み、メニーコアプロセッサが登場

1つのチップに数百個のコアを搭載したCPU

OSの観点からは、メニーコアを効率的に使用方法が課題



複数のOSを同時に走行させてメニーコアを効率的に利用

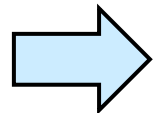
<1台の計算機上で複数のOSを走行させる方式>

(1) 仮想計算機方式

問題点: 仮想化によるオーバヘッドが発生

(2) Mint

仮想化によらず複数のLinuxを独立に同時走行

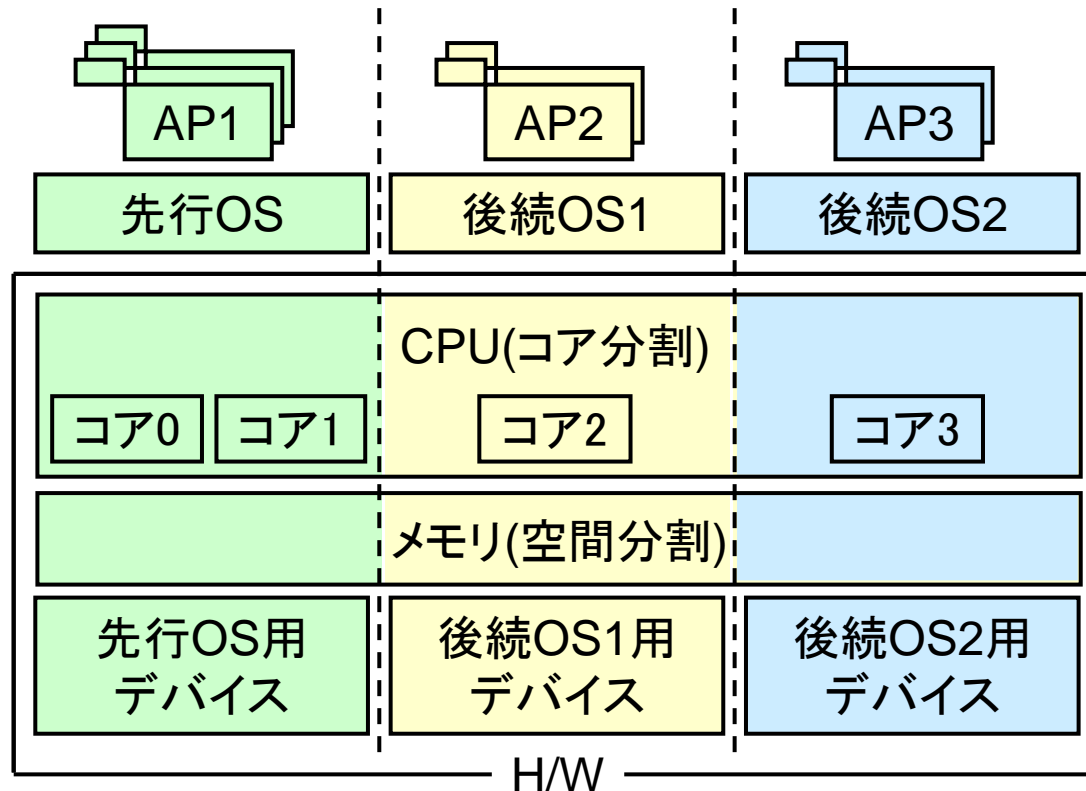


仮想化によるオーバヘッドの問題を解決可能

メニーコアの効率的な利用を考慮したMintの起動方式を実現

Mintの特徴

- (1) 1台の計算機上で複数のLinuxが走行
- (2) 各OSは, 1つ以上のコアを占有
- (3) 全てのOSが相互に処理負荷の影響を与えずに走行可能
- (4) 全てのOSが入出力性能を十分に利用可能



メニーコアの利用に関する要求

役割の異なる多数のOSを同時に起動させて密に連携

➡ メニーコアを効率的に利用

(要求1) 柔軟な起動方式の実現

動作OSと同数のカーネルイメージが必要となり、管理が煩雑

➡ 単一のカーネルイメージから複数のOSを起動可能となる柔軟な起動方式が必要

Kexecを応用した後続OS起動方式の実現

(要求2) 起動処理の高速化

多数のOSの起動処理を完了するための時間は長大

➡ 高速に多数のOSを起動させる必要

後続OS起動処理の並列化

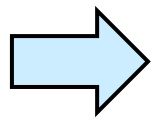
Kexecを応用する利点

<Kexecの特徴>

- (1) Linuxの高速な再起動方式
 - (2) BIOS, ブートローダ, セットアップルーチンを経由せずに再起動
 - (3) 指定した実メモリ上に再起動用のカーネルイメージを配置可能
- これらの特徴はMintにおいて後続OSの起動を実現するために有用

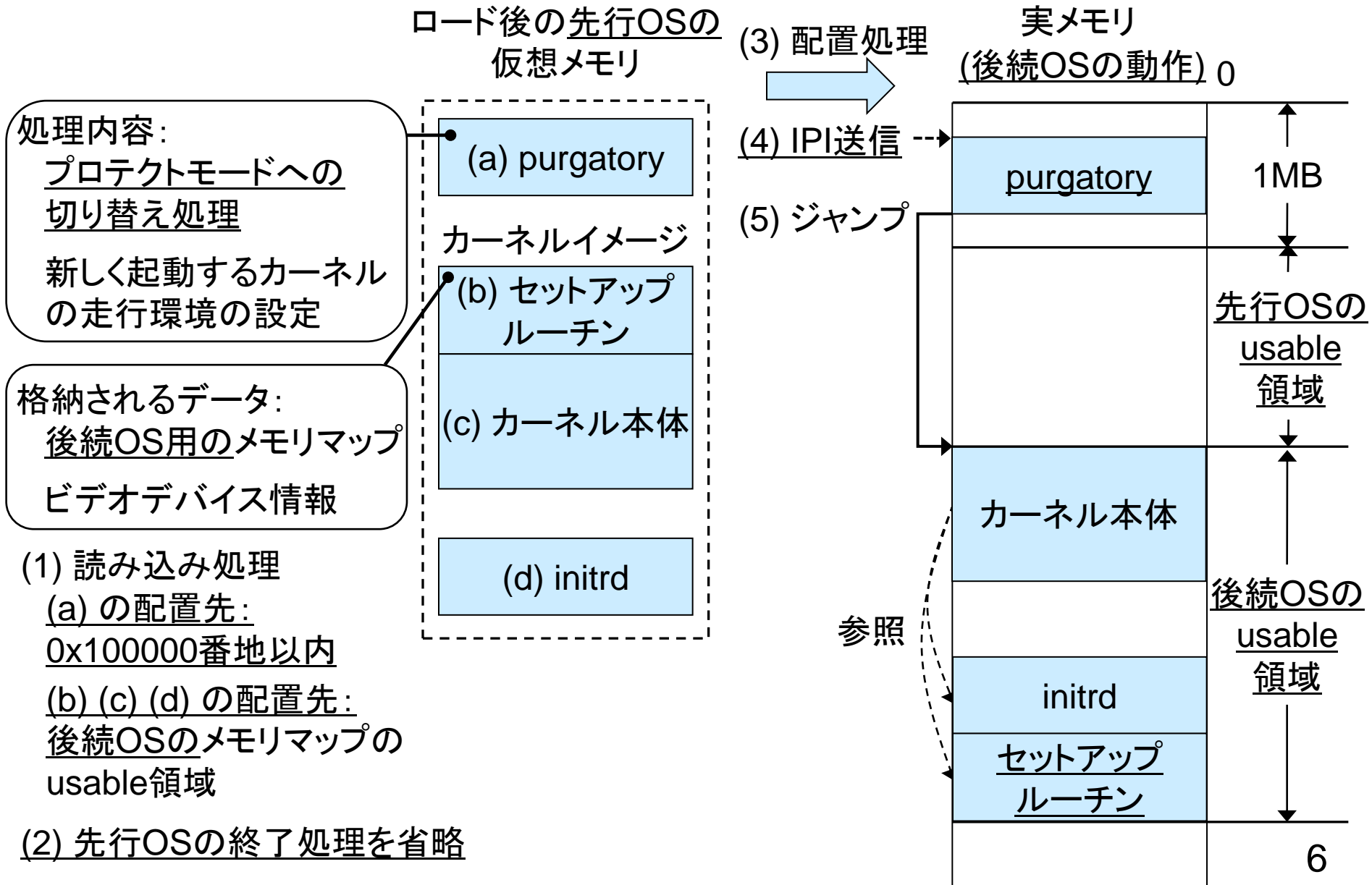
<Kexecを応用する利点>

指定した実メモリ上にカーネルイメージを配置する機能を利用



- (1) 単一のカーネルイメージを別々の実メモリ領域に配置可能
- (2) 単一のカーネルイメージから複数のOSを起動可能

Kexecを応用した後続OS起動方式



カーネルイメージの単一化

<カーネルイメージの単一化における課題>

Mintの各OSの差分は、占有するハードウェア資源のみ

➡ 各OS起動時に占有するハードウェア資源を指定する必要

<指定が必要なハードウェア資源>

CPU, **実メモリ**, デバイス

<実メモリの指定方法>

通常、カーネルの配置先アドレスは、カーネルのビルド時に確定

➡ 複数のカーネルイメージを作成する必要

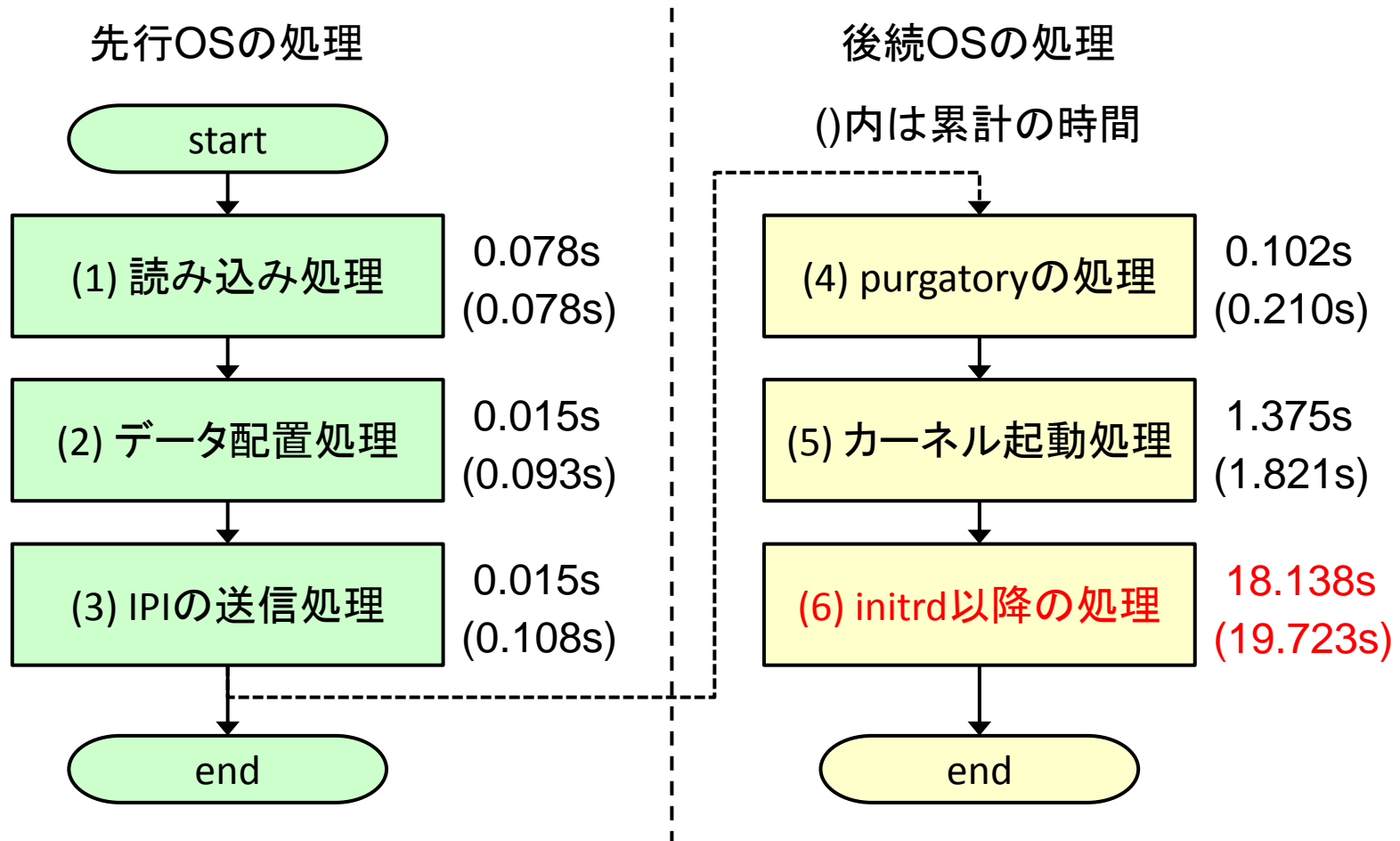
再配置可能な圧縮カーネルイメージを利用

圧縮カーネルイメージが配置されたアドレスから展開

Kexecによって圧縮カーネルイメージを任意の実メモリ領域に配置

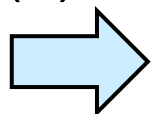
➡ 各OSに対して**起動時に**異なる実メモリ領域を割り当て可能

後続OS起動処理の並列化



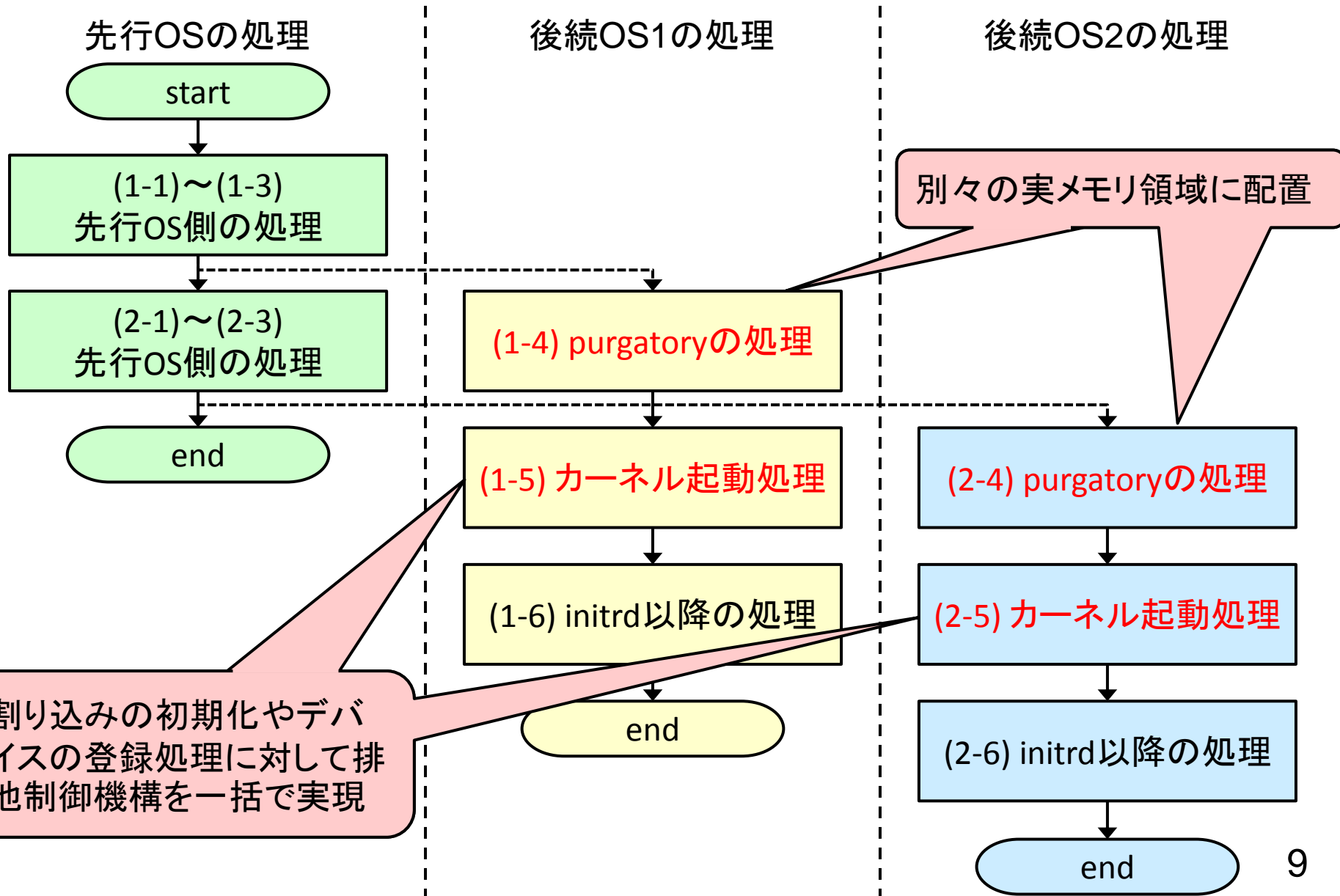
(6)自体はカーネル起動処理が完了しているため、並列に実行可能

(6)以外の部分に並列に実行不可能な箇所が存在



並列に実行不可能な箇所に対して排他制御を実現

並列起動の実現



32/64bitカーネルの混載

役割の異なるカーネルの連携例

64bit CPUの普及に伴い, OSの64bit化も活発化

<問題点>

64bitアーキテクチャにおけるCPUの走行モードの仕様

➡ 32bit OS用に記述された一部のソフトウェアを使用不可能
例:カーネル, デバイスドライバ

<対処>

CPUの走行モードは, コアごとに切り替え可能

走行モードの異なる32bit OSと64bit OSを同時に走行

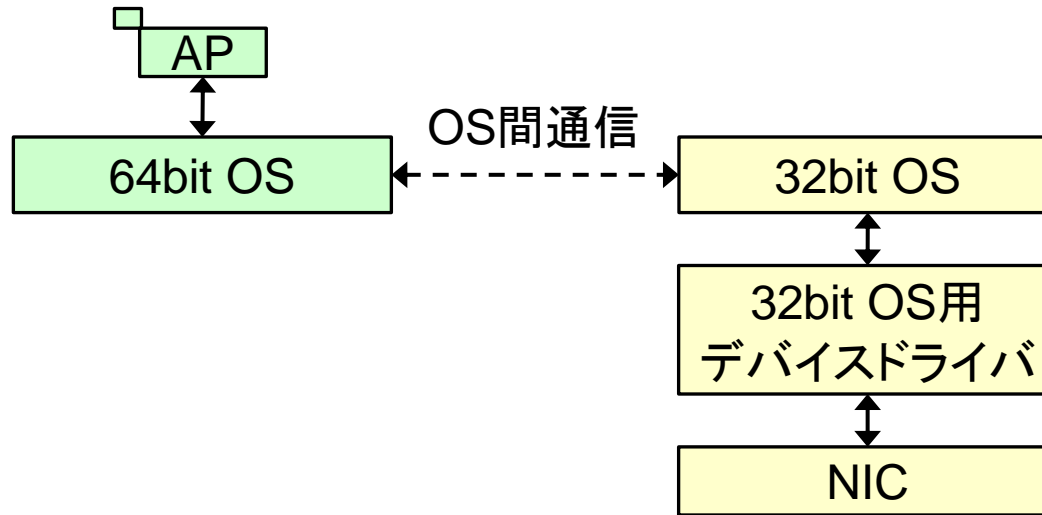
➡ 大量のメモリと32bit OS用のソフトウェアの両方を利用可能

Mintを64bitアーキテクチャに対応

➡ 32/64bitカーネルの混載を実現

混載による利用例

<利用例1>



64bit OSでは利用不可能なNICを32bit OSを介して利用

<利用例2>

32bitカーネルは64bitカーネルよりも必要な実メモリ量が少ない

4GB以上の実メモリを必要とするOSを64bitで動作させ、
実メモリをあまり必要としないOSを32bitで動作

➡ **メモリの節約が可能**

評価

<評価対象>

- (1) 起動処理に要する時間を測定
- (2) Intel Core i7(2.8GHz)を搭載した計算機を利用

<評価結果1>

(1) 未変更のLinuxの起動時間	21.4s
(2) 未変更のLinuxのKexecによる再起動時間	1.97s
(3) Kexecの応用によるMintの後続OS起動時間	0.309s

※ 測定の誤差を減らすため、kernel_initスレッド実行までを測定

<評価結果2>

- (1) デーモン起動処理まで含めた後続OS起動時間は、19.7s
- (2) 2つの後続OSを逐次起動するのに要する時間は、39.4s
- (3) 2つの後続OSを並列起動するのに要する時間は、21.6s

おわりに

メニーコアの効率的な利用を考慮したMintの起動方式を実現

(1) 柔軟な起動方式の実現

単一のカーネルイメージから複数のOSを起動可能となる柔軟な起動方式が必要

 Kexecを応用した後続OS起動方式の実現

(2) 起動処理の高速化

高速に多数のOSを起動させる必要

 後続OS起動処理の並列化

(3) 役割の異なるカーネルの連携例

32/64bitカーネルの混載を実現

(4) 評価

起動処理に要する時間を測定