

Mintオペレーティングシステムにおける OS環境の高速な複製手法

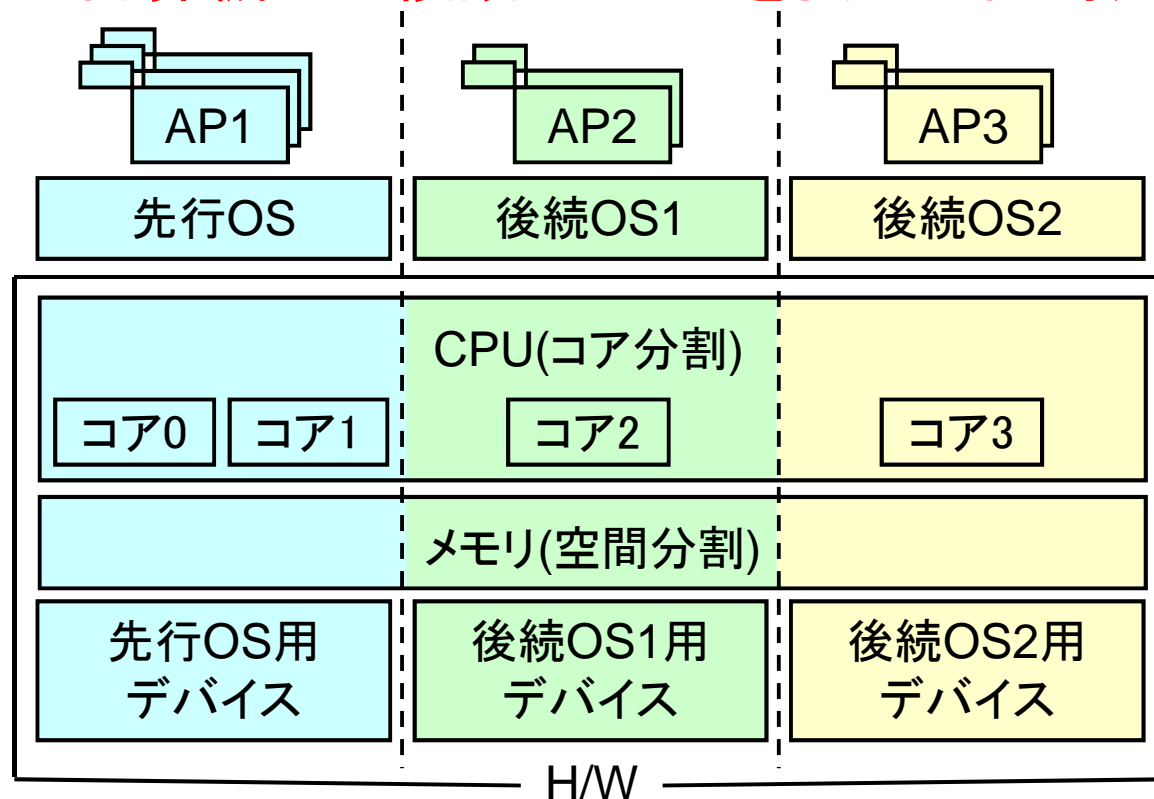
西田 潤

岡山大学 工学部 情報工学科

平成24年 2月17日

研究背景

<Mint> 1台の計算機上で複数のLinuxを独立に同時走行させる方式



<後続OS起動における問題点>

- (1) 起動させるOSの数だけHDDから読み込みが発生
- (2) 複数回にわたるシステムコールの呼び出し

➡ 後続OSの起動に時間がかかる

目的

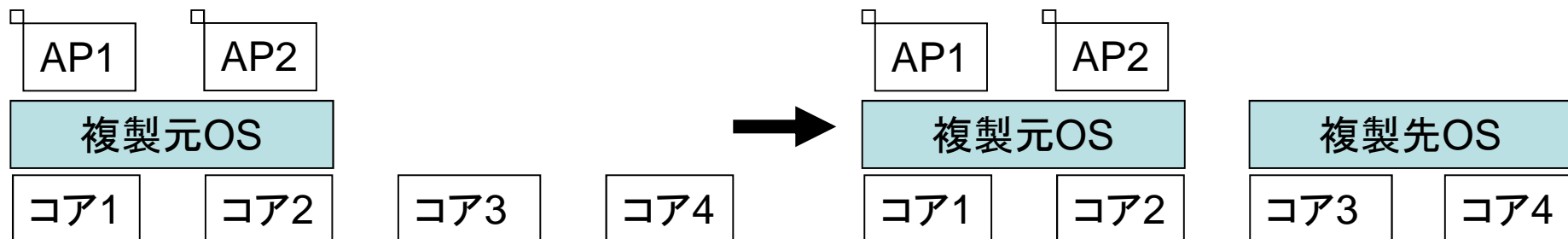
メモリ上に展開されている先行OSを複製する手法を提案

- (1) HDDからの読み込み処理を省略
- (2) システムコール呼び出し回数を削減

➡ 複製先OSを高速に起動

<複製手法の手順>

- (1) 走行中の複製元カーネルを複製
- (2) 複製元OSが占有しているコアとは別のコアで複製先OSを起動
- (3) 複製元OSの走行環境を複製先OSで再現



課題の一覧

複製したカーネルの初期化ルーチンの変更

(1) ページテーブルの初期化

両カーネルの仮想アドレスを同一にする必要

(2) コアの走行環境の再現

走行中の複製元OSはプロテクトモードで走行

➡ 複製先のコアの走行環境を再現することが必要

(3) 各OSのデバイスの初期化处理

(4) 初期化関数とデータの保護

OSを複製する手法の設計

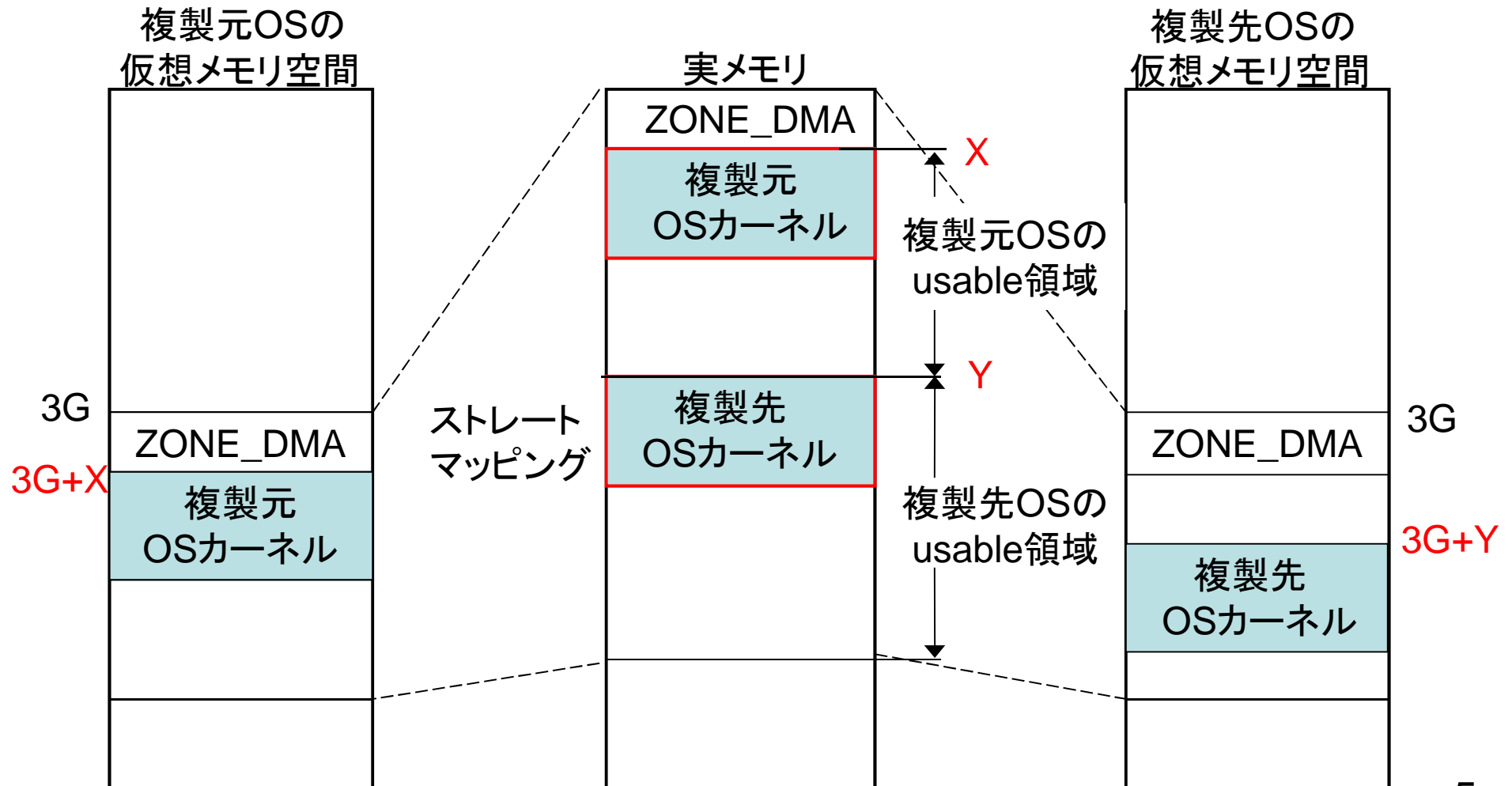
(5) カーネルを含んだ実メモリ領域を指定の領域へ複製

それぞれのOSが使用する実メモリ空間は互いに使用禁止

(課題1)の詳細

(課題1) ページテーブルの初期化

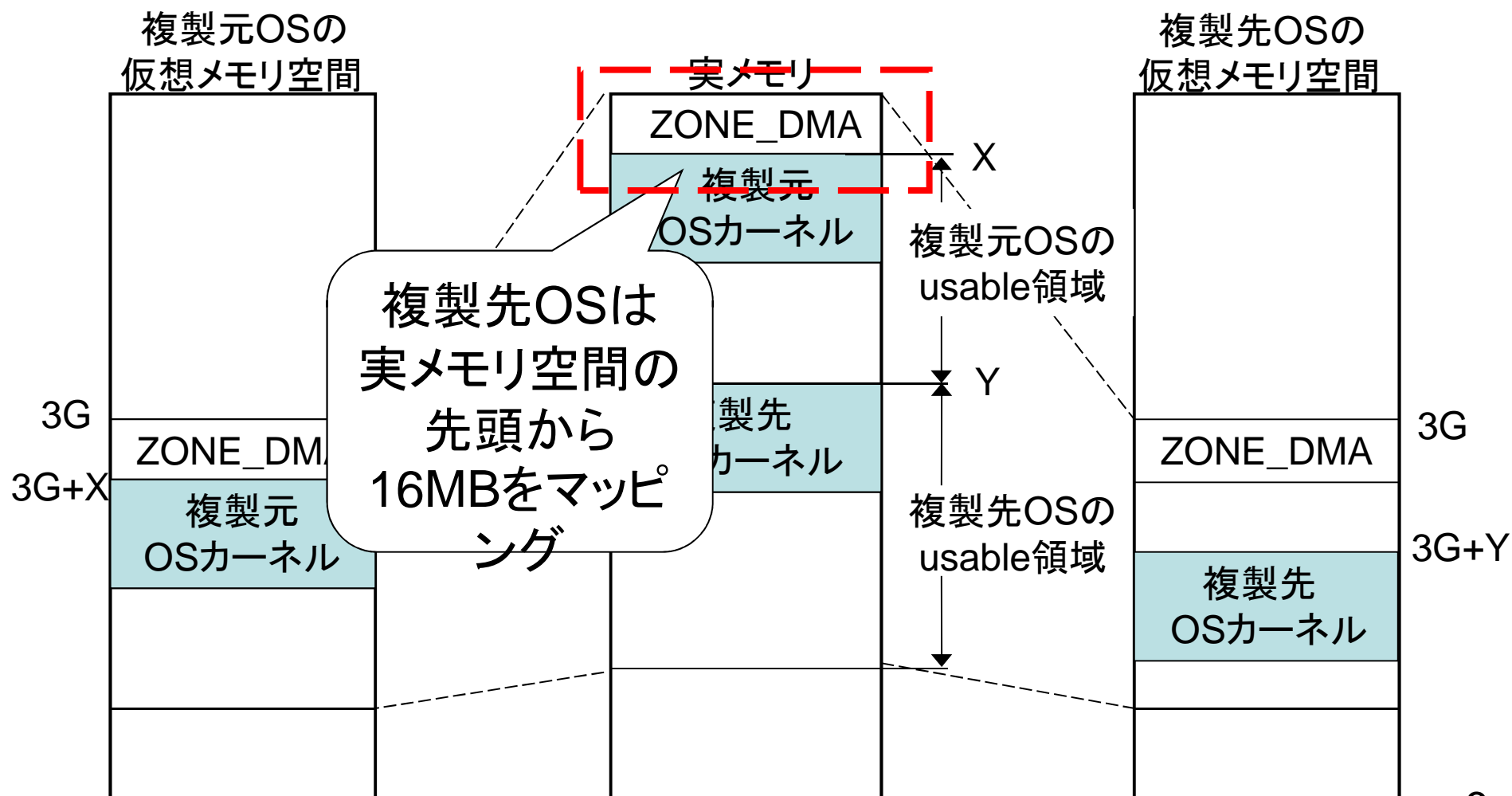
複製元OSと複製先OSでカーネルの仮想アドレスを同一にする必要



(課題1)の詳細

(課題1) ページテーブルの初期化

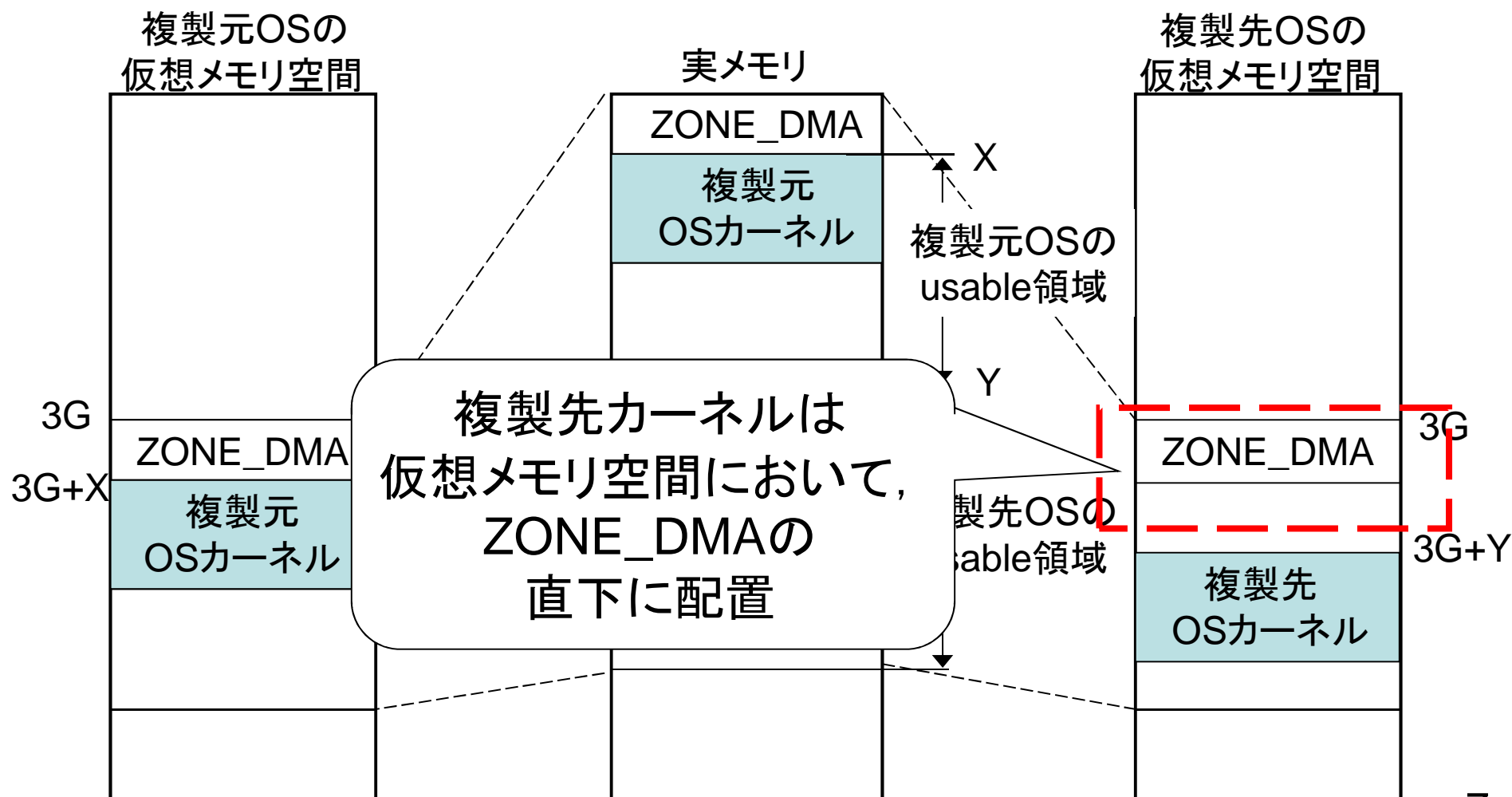
複製元OSと複製先OSでカーネルの仮想アドレスを同一にする必要



(課題1)の詳細

(課題1) ページテーブルの初期化

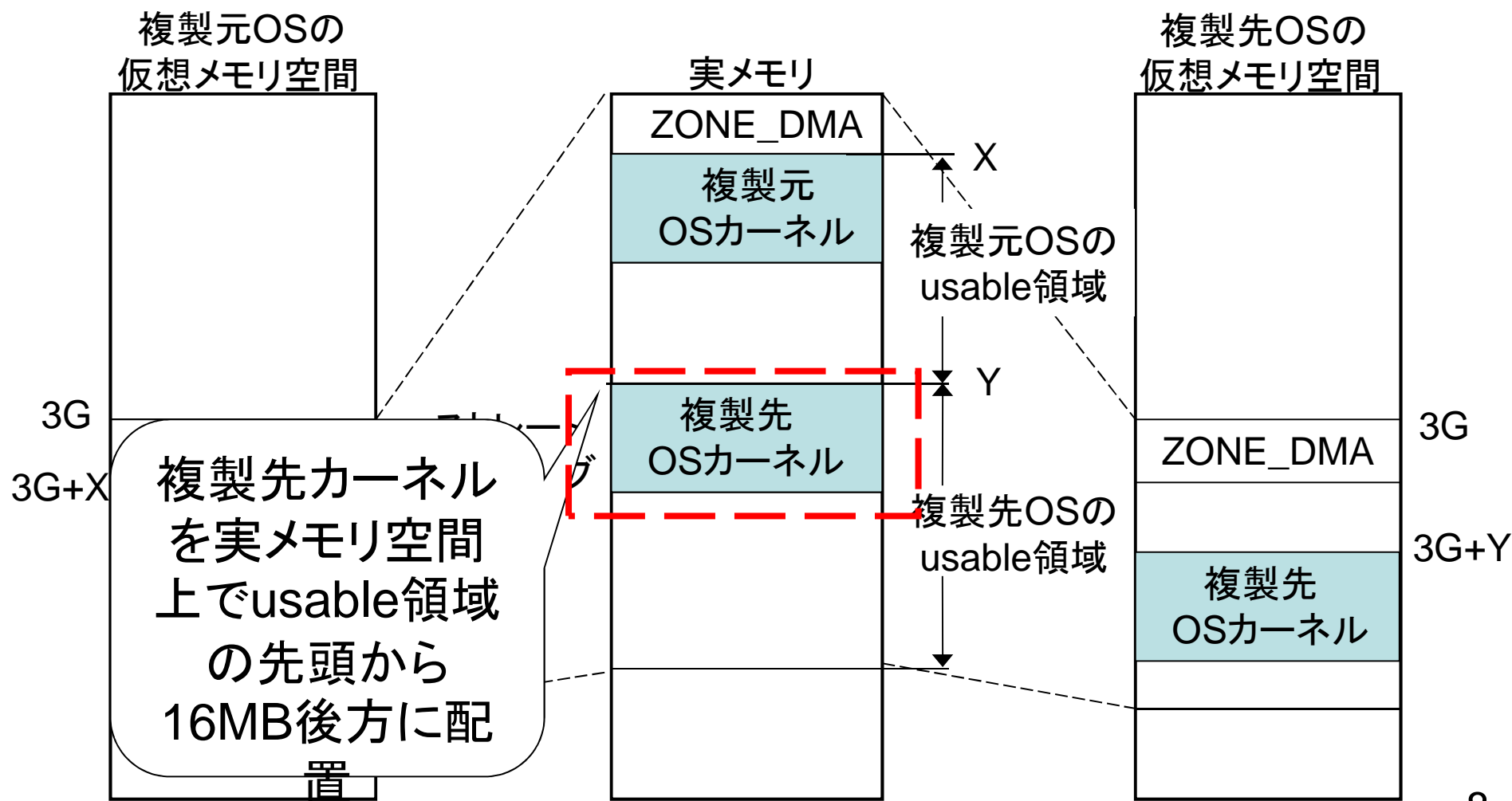
複製元OSと複製先OSでカーネルの仮想アドレスを同一にする必要



(課題1)の詳細

(課題1) ページテーブルの初期化

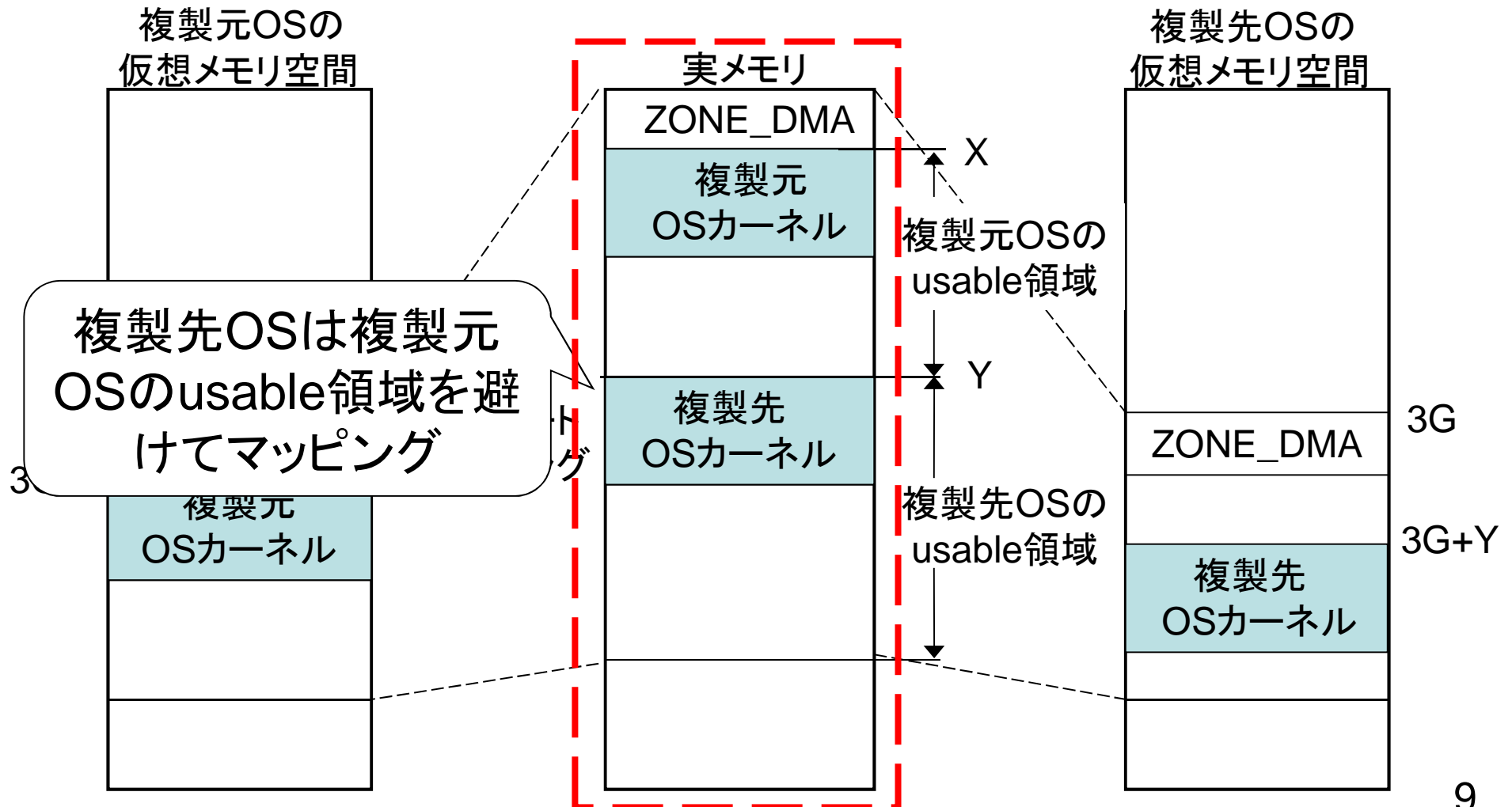
複製元OSと複製先OSでカーネルの仮想アドレスを同一にする必要



(課題1)の詳細

(課題1) ページテーブルの初期化

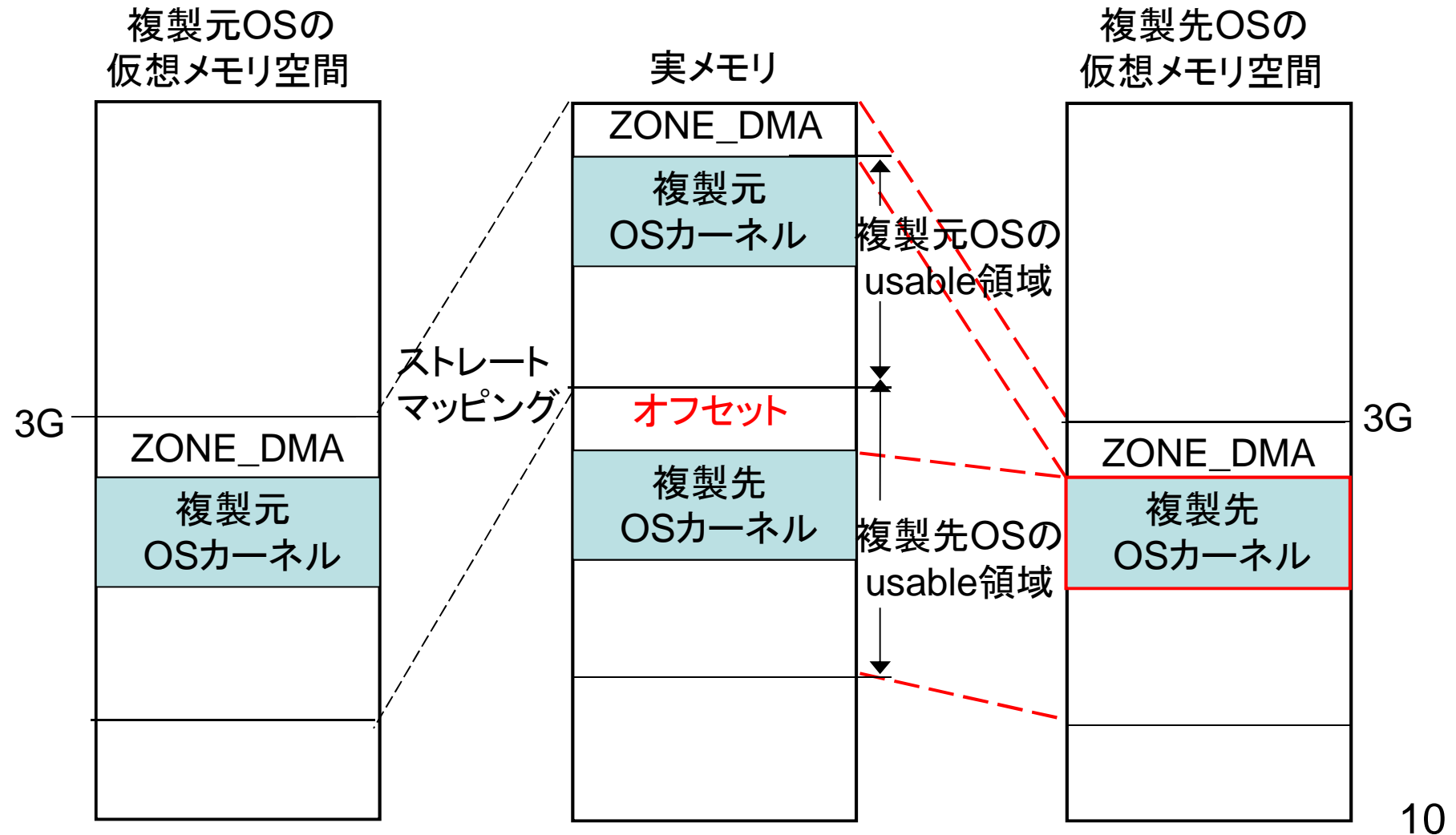
複製元OSと複製先OSでカーネルの仮想アドレスを同一にする必要



(課題1)の対処

(対処1)

それぞれのOSの仮想メモリ空間の同じアドレスにマッピング



実装

複製先カーネルのページテーブルの初期化処理を実験的に実装

- (1) ページテーブルの初期化処理を変更
- (2) 仮想アドレスと実アドレスを相互変換する処理を変更

コード変更量

	コード行数	hunk数
全体	46	6
arch/x86/kernel/head_32.S	40	2
arch/x86/config/i386_defconfig	2	2
arch/x86/include/asm/page_32.h	2	1
arch/x86/include/asm/page.h	2	1

本発表のまとめ

<実績>

- (1) 初期化ルーチンの変更手法の設計
- (2) OSを複製する処理を設計
- (3) ページテーブルの初期化を実装

<残された課題>

- (1) 残りの初期化ルーチンを実装
- (2) OSを複製する処理を実装
- (3) ページテーブルの初期化処理をpurgatoryに移行