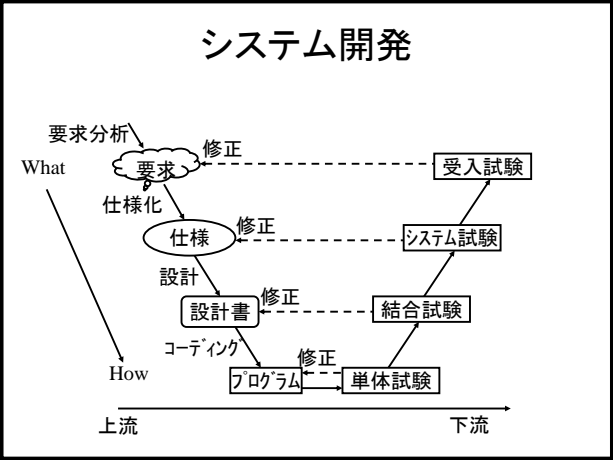
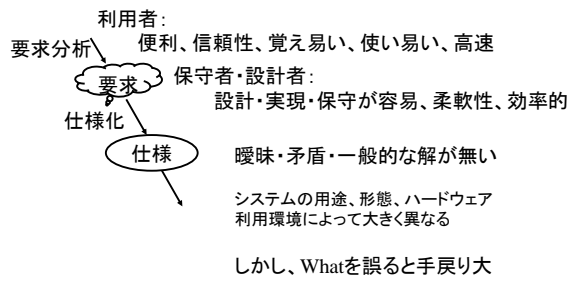


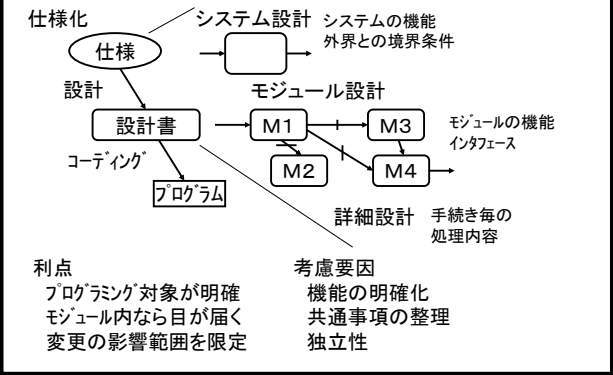
第12回 設計原理とOSの構成法



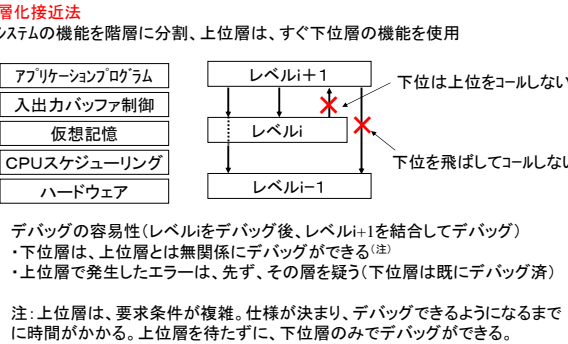
上流工程の困難性



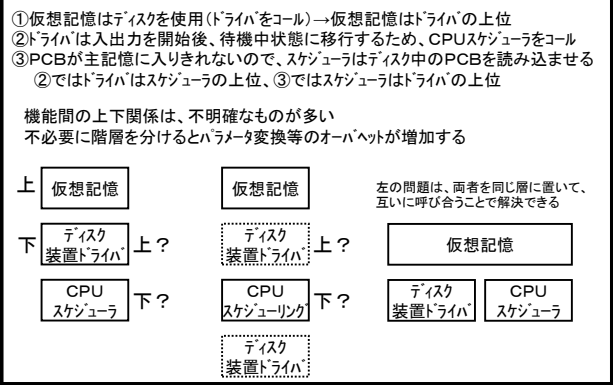
モジュール化



層化接近法 (layered approach)



参考: レベルの定義



CP/M層構造

MS/DOSの原型

レベル3: AP、コマンド解釈実行プログラム
レベル2: 論理(ファイル)入出力(BDOS)
レベル1: 物理入出力、装置ドライバ(BIOS)
レベル0: ハードウェア

当時のPC用のOSで、  
シングルユーザ、シングルプロセス  
のOSとして開発  
(多重プログラミングは行わない  
という前提)

Basic Input / Output System  
システムの基本ハードウェア(CPU、HDD、KB等)を制御するマイクロコ

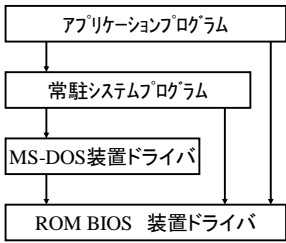
参考 システム立ち上げ手順

コンピュータの電源を入れてからの基本的な流れ。



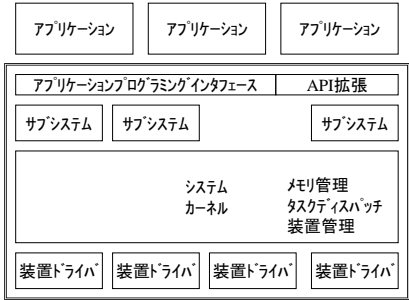
MBR: Master Boot Record (ハードディスクの先頭セクタ. bootのローダ、パーティション情報)

参考: MS-DOSの層構造



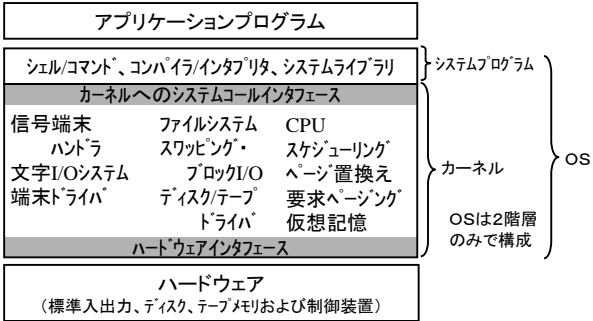
(1) 階層化の配慮は無い。  
最小の資源でPCに必要な  
機能を提供するのが目的  
  
例  
アプリケーションから直接ディスク  
へ書き込める  
  
(2) ハードウェア(8088)の制約  
特権モード、非特権モードの区  
別が無い。  
ハードウェアによる保護機能が無い

参考: OS/2の層構造



MS-DOSの後継システム、マルチタスクとデュアルモード操作を追加  
階層構成が明確(利用者が直接下位レベル機能をアクセスできない)

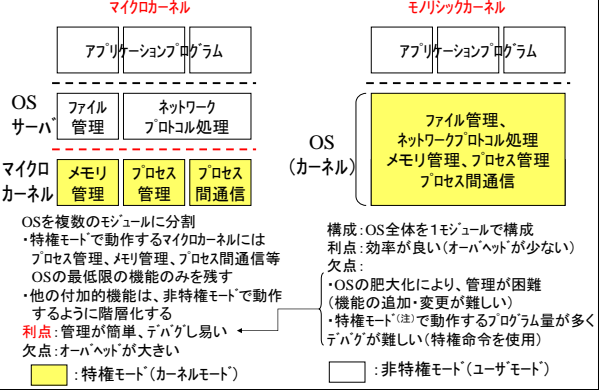
初期のUNIXの層構造



モリシクカーネル<sup>(注)</sup>: 多くの機能がカーネルの1階層に押し込められている。  
(機能追加をしていくなかでカーネルが肥大化)

注: モリシク=一枚岩

重要: モリシクカーネルとマイクロカーネル



OSを複数のモジュールに分割  
・特権モードで動作するマイクロカーネルには  
プロセス管理、メモリ管理、プロセス間通信等  
OSの最低限の機能のみを残す  
・他の付加的機能は、非特権モードで動作  
するように階層化する  
利点: 管理が簡単、デバッグしやすい  
欠点: オバヘッドが大きい

構成: OS全体を1モジュールで構成  
利点: 効率がよい(オバヘッドが少ない)  
欠点:  
・OSの肥大化により、管理が困難  
(機能の追加・変更が難しい)  
・特権モード<sup>(注)</sup>で動作するプログラム量が多く  
デバッグが難しい(特権命令を使用)

注: 特権モード(カーネルモード)

注: 非特権モード(ユーザモード)

