Systèmes d'exploitation et Programmation Concurrente

Yves Denneulin Yves.Denneulin@grenoble-inp.fr

Jacques Mossière

Sebastien Viardot

L’équipe « SEPC »

- Le cours
  - Yves Denneulin
- Les TD/TP
  - Louis Boulanger, Florence Maraninchi, Grégory Mounié, Alain Tchana, Frédéric Wagner

L’organisation du semestre

- 1h30 Cours /semaine
  - vidéo disponible avant le cours
  - créneau utilisable pour poser des questions
- 3h de TD/TP SEPC tous les 15 jours
  - TPs en libre service

Et le semestre 2 ?

- Semestre 2 : Projet de conception de système d'exploitation - approfondissement
  - Optionnel ISI
- Projet filé
  - En système ou en archi

Références

- Tanenbaum, Andrew « Modern operating systems » Prentice Hall
- Silberschatz, Galvin, Gagne « Operating system concepts », Wiley
- Krakowiak, Sacha « Principes des systèmes d ’exploitation des ordinateurs », Dunod

Quelques liens utiles

- http://www.wiley.com/college/silberschatz
- http://www.cs.unm.edu/~crowley
- http://www.pearsonhighered.com

Les transparents

- Ceux faits spécialement pour ce cours
- Ceux empruntés à Tanenbaum ou Silberschatz
- Ne constituent qu'un plan détaillé
- Sont disponibles avant le cours

Les résumés

- Une version pour certains chapitres est disponible
- Il ne s'agit que du résumé des points traités en cours (pas de double emploi avec un livre)
- Description en pseudo-C d'un système jouet (PedagOS)
- Tout commentaire constructif est le bienvenu

Les questions

- Porte sur des points du cours pendant la séance
- Utilisés pour vérifier que le message est passé
  - Ou pas…
- La réponse est donnée juste après
  - vérifier que vous avez bien compris l'explication

Plan de la séance

- Fonctions d'un système d'exploitation
- Différentes classes de systèmes
- Objectifs du cours
- Principaux chapitres

Essai de définition

- Le matériel n'est (presque) jamais utilisé seul
  - Ensemble de logiciels plus ou moins proches de l'application (entrées-sorties, fichiers, compilateurs, etc.)
- Le système d'exploitation regroupe les logiciels les plus proches du matériel
  - Ceux qui sont toujours présents en mémoire pendant l'exécution des applications

« Un système d'exploitation, c'est une collection de choses qui ne tiennent pas dans un langage. Ça ne devrait pas exister. »

Dan Ingalls, *Design Principles Behind Smalltalk*, *Byte Magazine*, August 1981.

Une brève histoire des systèmes d'exploitation

http://j.mp/2c0aFoc

Fonctions d’un système

- Les systèmes que vous avez utilisés
  - Unix, Windows, Android, …
- Machine virtuelle ou étendue
  - Langage de commande « shell »
  - Appels systèmes
- Partage de ressources
  - Fonctions d’un serveur

Un peu d'histoire

- Évolution parallèle des systèmes et des architectures matérielles
  - Plus le matériel a de fonctionnalités et plus l'OS est complexe
- D'abord partage de ressources, puis prise en compte de la simplicité d'utilisation
- Recherche pionnière 1965-75
  - multics, puis unix
- Travaux actuels
  - Systèmes répartis, intergiciels, virtualisation, auto-administration, exploitation des multi-coeurs, stockage, mémoire persistente

Rôles de l'OS

- Abstraire/cacher le matériel (55-)
  - Faciliter (factoriser) la programmation
  - Utiliser des abstractions de plus haut niveau
- Partager et faire coopérer les ressources matérielles
  - Optimiser l'utilisation du matériel pour un ou un ensemble de critères (temps de réponse, équité,…)
  - Seconde époque 65-
  - Début du multi-tâche (multi-programmation)
    - Et des problèmes de synchronisation associés!
  - Exemple : couvrir les temps de communication avec du calcul

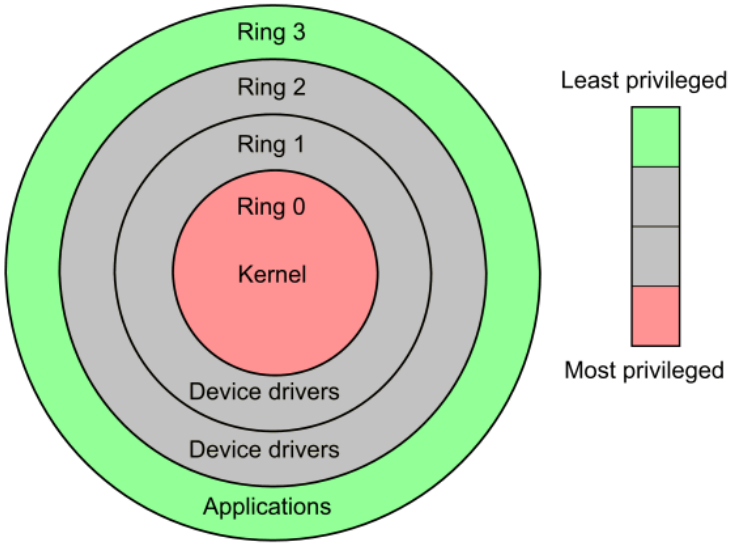
Rôles de l'OS (2)

- Assurer le cloisonnement (75-)
  - Entre les applications
  - Entre applications et système
  - Définition de rôle et de droits + accès et possession de ressources
    - L'OS fait respecter les droits
    - Utilise des fonctions spécifiques des processeurs

- Plus le matériel a de fonctionnalités et plus l'OS est complexe

Structure de l'OS

- Un programme particulier
  - S'exécute dans un mode processeur particulier, **protégé**
  - Séparé des applications
    - Mécanisme de séparation fourni par le processeur
  - Le **noyau** du système
- Lien avec les applications
  - Les bibliothèques système
  - Différents programmes
- Performance : élément clé



©Coding Horror

Passage entre les modes

Mode application

Interruption matérielle
ou logicielle

Retour d'Interruption

Mode protégé (noyau)

Passage entre les modes est coûteux (temps, vidange des caches)

Différentes classes de systèmes

- Ordinateur individuel
  - *Smartphones*, tablettes
- Systèmes à transactions
- Commande de procédés industriels
  - Systèmes embarqués

Ordinateur individuel

- Qualité de l'interface utilisateur
  - Temps de réponse
- Simplicité d'utilisation
- Conservation d'informations
  - Attention aux sauvegardes
- Dominante : machine virtuelle
  - Mais plusieurs applications coexistent donc partage de ressources aussi

Smartphone

- Temps de réponse
- Isolation entre les applications
- Consommation d'énergie

- Dominante : partage de ressources
  - Processeur
  - Mémoire vive

Systèmes à transactions

- Systèmes bancaires, de réservation de places
- Grand nombre de clients qui demandent des opérations simples
- Grand volume de données à longue durée de vie
- Dominante : cohérence et conservation des données à long terme, rendement, réactivité

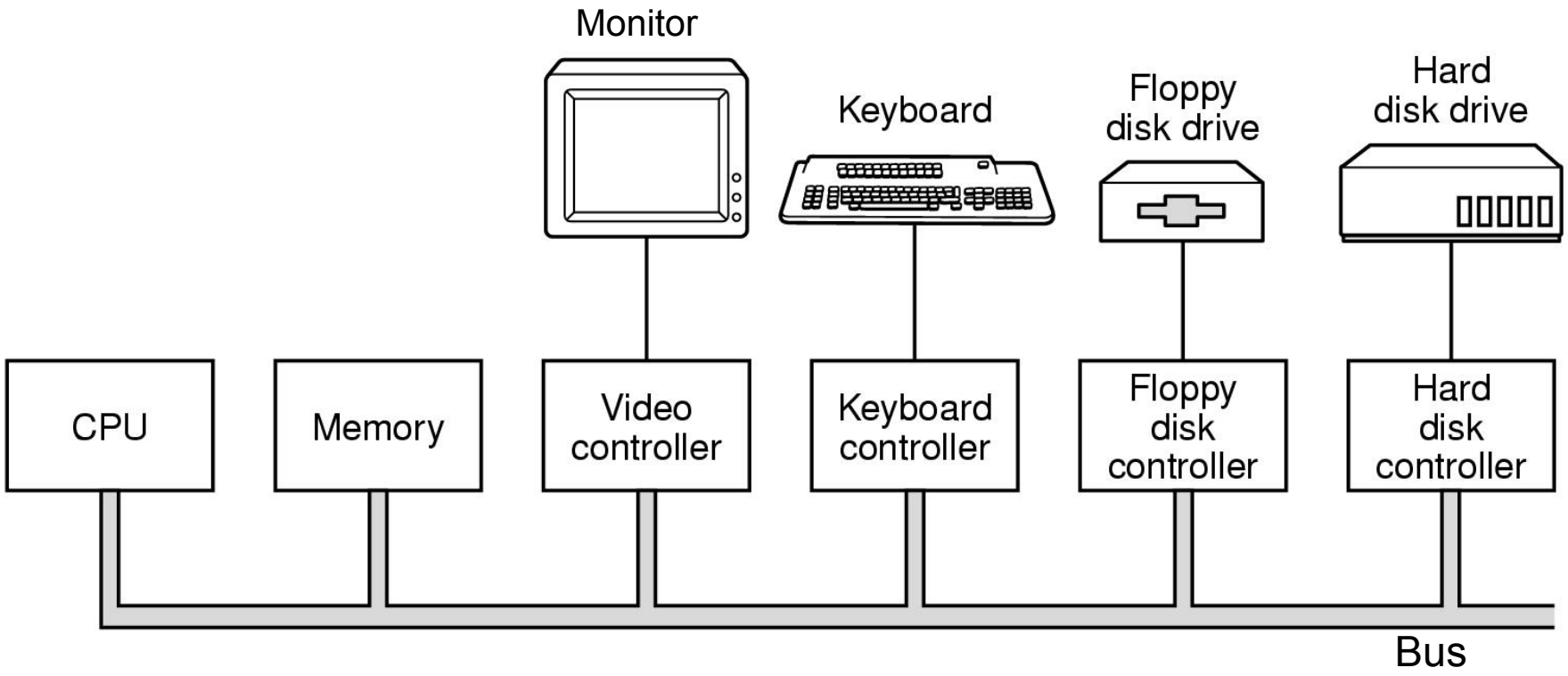
Commande de procédés industriels

- Observation et contrôle d'un procédé physique
  - Fusée, raffinerie, machine à laver
- Boucle observation, calcul, action
- Contraintes de temps physique
- Dominante : sûreté de fonctionnement et dimensionnement du système

Rappels d'architecture

- CPU
  - état, registres
  - Modes de fonctionnement
- Mémoire
  - Mémoire principale
  - disques

Computer Hardware Review (1)



- Components of a simple personal computer
- OS = chef d'orchestre

The Storage Latency Hierarchy

Technology	Latency	Size (e.g.)
L1 CPU Cache	4 cycles (~1 nsec)	32K
L2 CPU Cache	10 cycles (3 nsec)	256K
LLC CPU Cache	40 cycles (13 nsec)	1 MB
DRAM	240 cycles (80 nsec)	16 GB
NVRAM	1200 cycles (400 nsec)	128 GB
RDMA Read	6K cycles (2 usec)	16 GB
FLASH Read	150K cycles (50 usec)	128 GB
FLASH Write	1500K cycles (500 usec)	128 GB
HDD Write min	1500K cycles (500 usec)*	4 TB
HDD Read min	15000K cycles (5 msec)	4 TB
HDD Read max	75000K cycles (25 msec)	4 TB
Tape File Access	150000000K cycles (50 sec)	6 TB

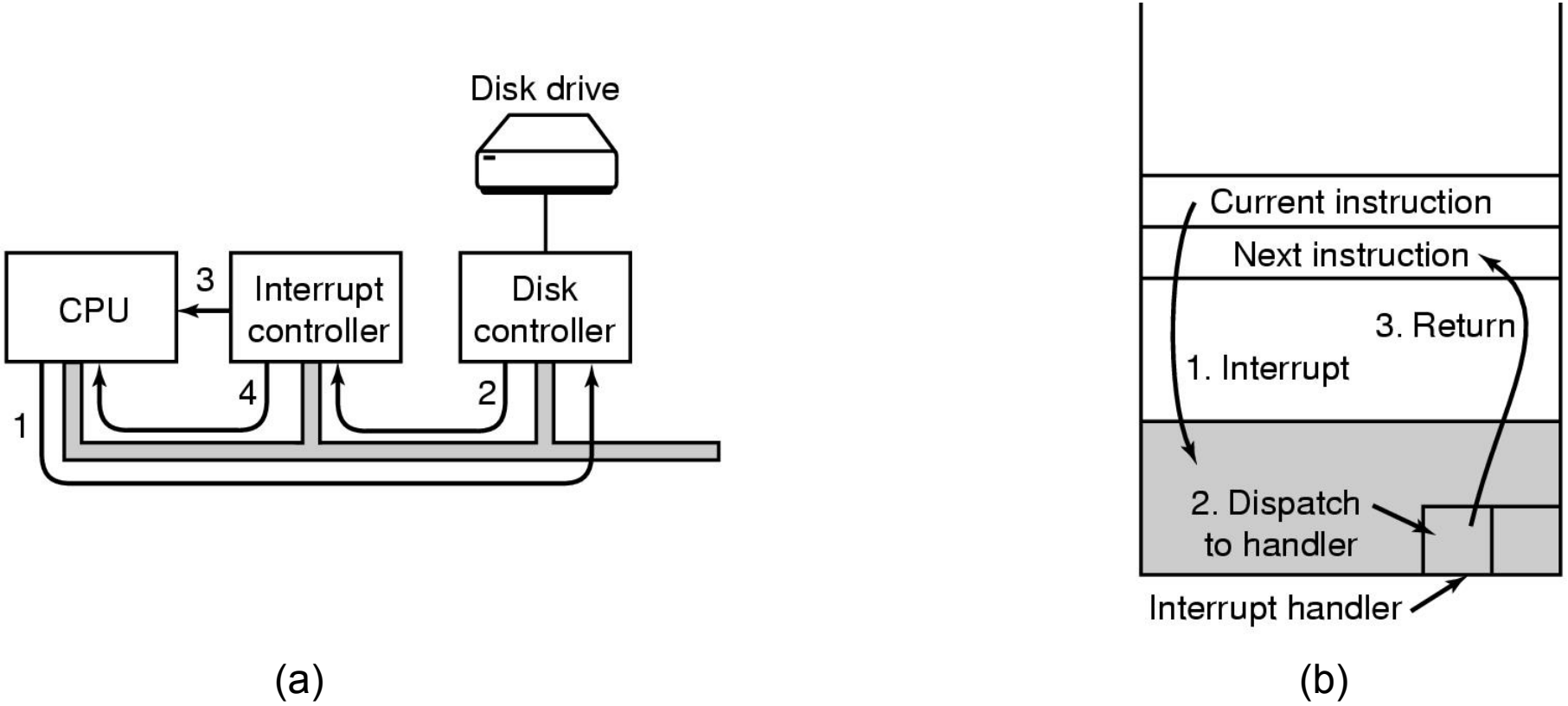
2017-8

* Write to track cache

Communication entre unité centrale et périphérique

- Notion d'interruption

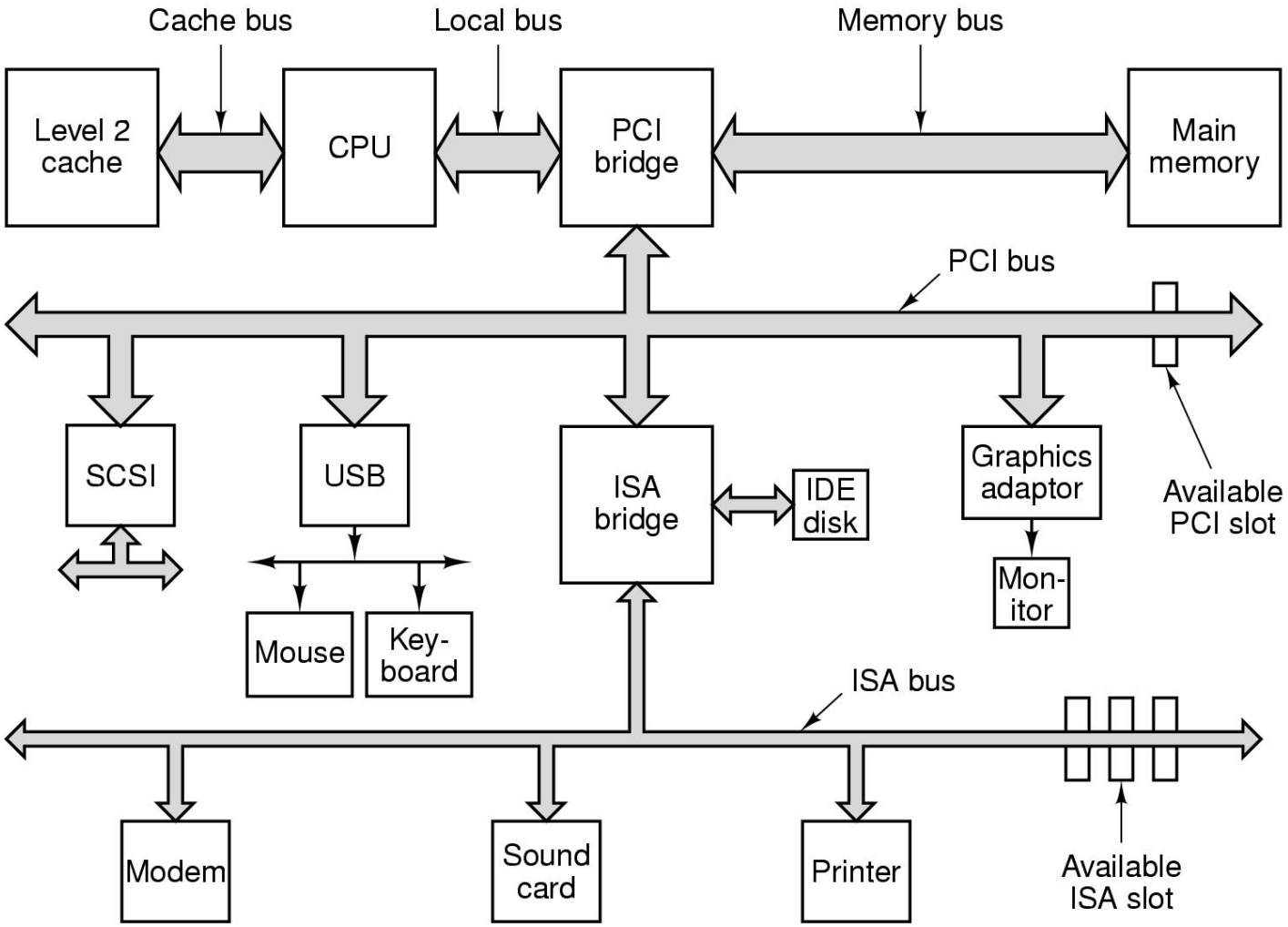
Computer Hardware Review (4)



(a) Steps in starting an I/O device and getting interrupt

(b) How the CPU is interrupted

Computer Hardware Review (5)

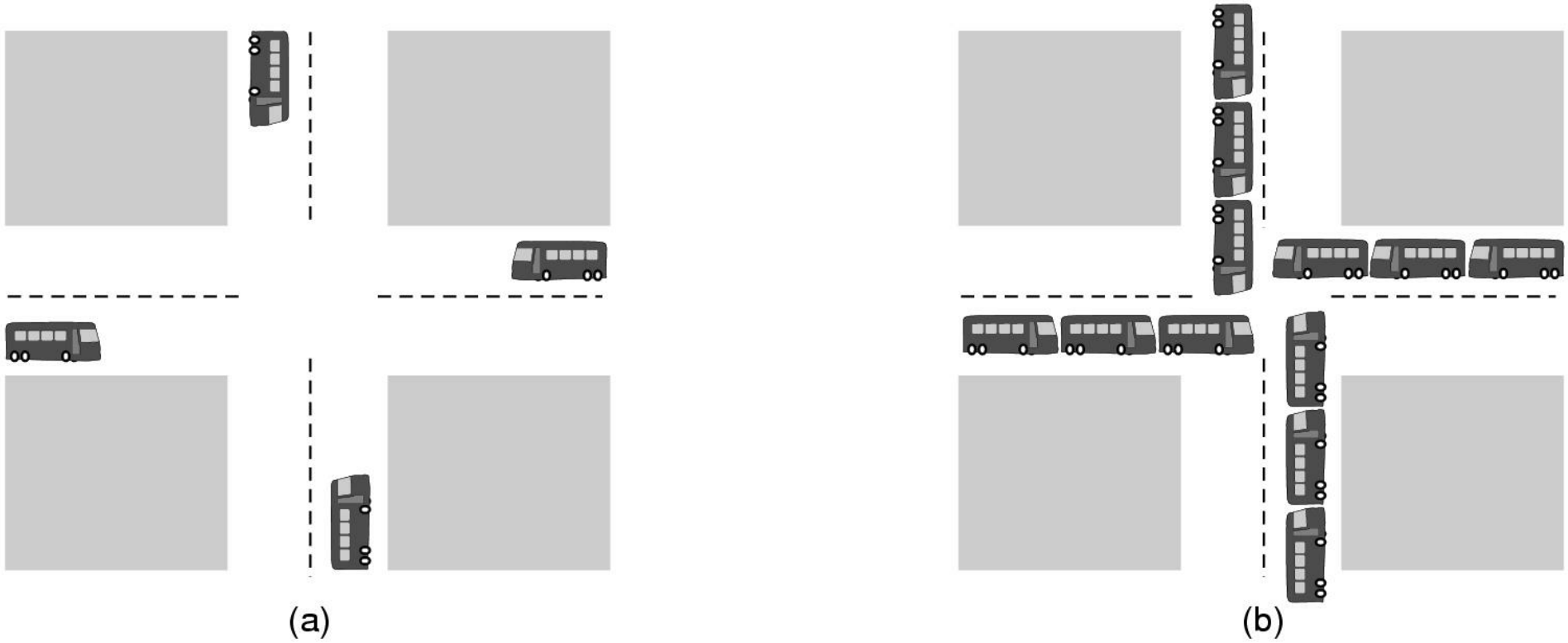


Structure of a large Pentium system

Concepts principaux

- Activités parallèles
  - Synchronisation
  - Inter blocages
- Partage du temps

Operating System Concepts (2)



(a) A potential deadlock. (b) an actual deadlock.

Conservation des informations

- Fichiers, catalogues
  - Abstraction de plus haut niveau
- Persistance
  - redondance
- Sécurité - protection

Gestion de la mémoire

- Partage entre les différentes activités
- Mémoire virtuelle
  - indépendance de la taille de la mémoire physique
  - cloisonnement entre les applications et le système
  - meilleure utilisation de la mémoire physique

Objectifs du cours

- Pré Requis : structure d'un ordinateur, logiciel de base, algorithmique
- Comprendre le fonctionnement d'un système d'exploitation
- Savoir invoquer directement les services du système (C)
- Comprendre et résoudre un problème de synchronisation
- Effectuer de la programmation « système », écrire un pilote de périphérique par exemple
- Effectuer des adaptations à des logiciels existants
- (Écrire des systèmes complets)

Pourquoi à l'Ensimag ?

- Cours qui concerne toutes les filières
  - Programmation concurrente : toutes les architectures le sont!
    - monte-carlo efficace
  - Efficacité : surcoût minimal, utilisation optimale des ressources
  - Partage de ressources : équité et protection
  - Les applications modernes sont des applications réparties donc avec des synchronisations
- Proche du matériel mais dimension algorithmique aussi
- Enseignements voisins: architecture, réseaux, évaluation de performances, applications réparties

Principaux chapitres

- Gestion des activités parallèles
  - outils de synchronisation et problèmes associés
- Entrées-sorties, interruptions
- Gestion de mémoire principale
- Conservation des informations et fichiers

En résumé

- Le système d'exploitation est le logiciel qui se situe entre le matériel et les applications
- Il sert à abstraire le matériel, répartir les ressources et isoler les applications
- Il doit gérer différents éléments matériels et logiciels en simultané, c'est un programme parallèle
- Une bonne utilisation des ressources par les programmes dépend de lui
- Il est constitué :
  - d'un programme noyau qui s'exécute dans un mode particulier du processeur lui donnant accès à toutes les ressources,
  - d'une interface permettant aux applications de faire appel à ses services.

One more thing…