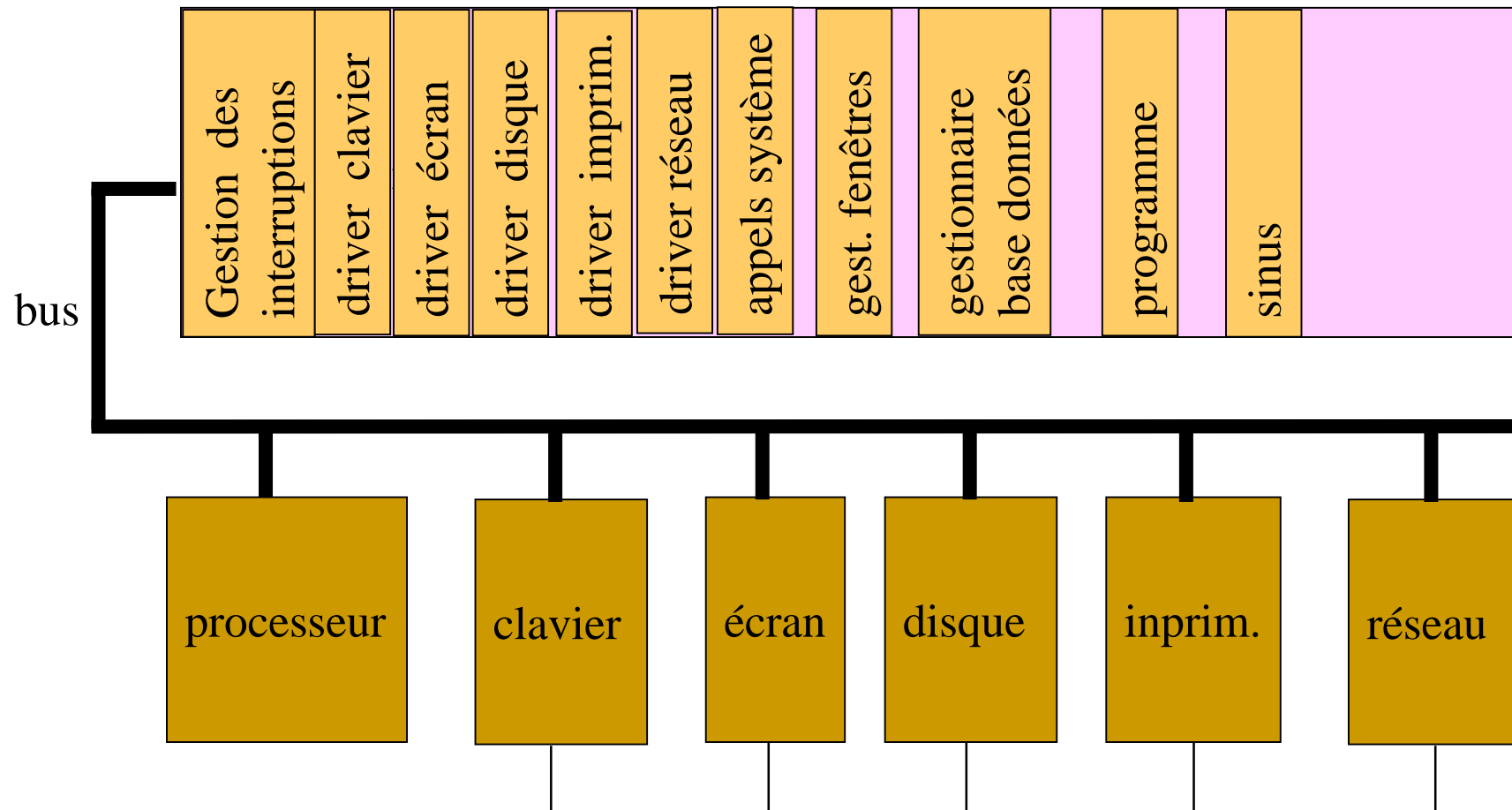
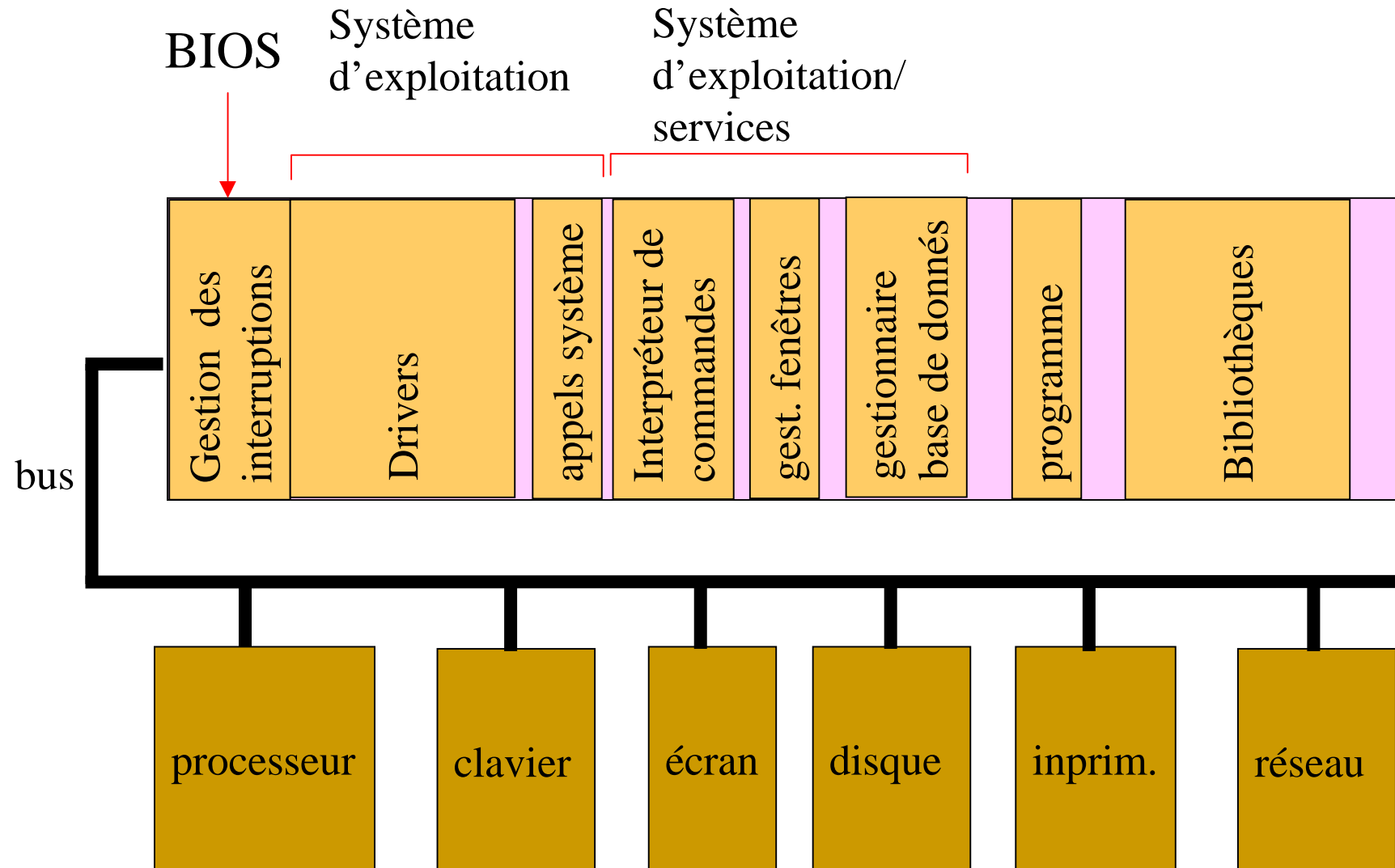


Introduction aux enseignements de Systèmes informatiques

introduction aux systèmes informatiques





■ Question :

Quelle est l'entité qui installe un programme dans la mémoire?

• Réponse :

Le système d'exploitation à la demande d'un autre programme

■ Question :

Quels sont les programmes qui peuvent demander au système d'exploitation l'installation d'un programme dans la mémoire puis son exécution

• Réponse :

L'interpréteur de commandes le plus souvent mais aussi n'importe quel autre programme

- Question :

Quelle est l'entité qui installe le système d'exploitation dans la mémoire ?

- Réponse :

Le BIOS qui réside en permanence en mémoire non volatile (ROM)

■ Question :

Pourquoi existe-t-il plusieurs systèmes d'exploitation?
(VMS, GCOS, MVS, AIX, ULTRIX, HPUIX, SOLARIS , QNX,
WINDOWS, LINUX.....)

- Réponse : Satisfaire des besoins différents tels
 - ordinateur personnel
 - centre de calcul scientifique
 - Gestion base de Données (Banque.....)
 - nœud réseau
 - gestion de trafic (aérien, routier, ferroviaire, maritime)
 - systèmes embarqués (téléphones mobiles, cartes à puces)

■ Question :

En quoi les systèmes d'exploitation différent-ils ?

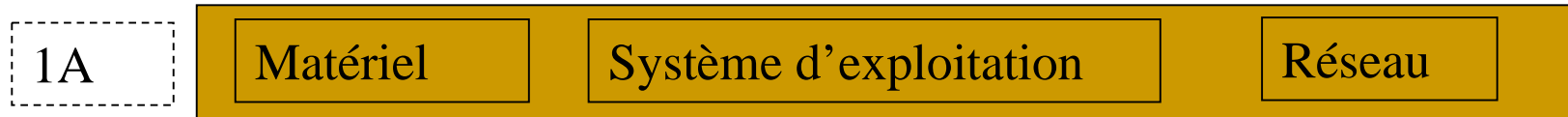
• Réponse :

- la liste des appels systèmes qu'ils proposent
- le format des fichiers exécutables

• Conséquence :

- un programme compilé (préparé) pour un système d'exploitation donné ne peut être exécuté sur un autre.

Systèmes informatiques : ISI



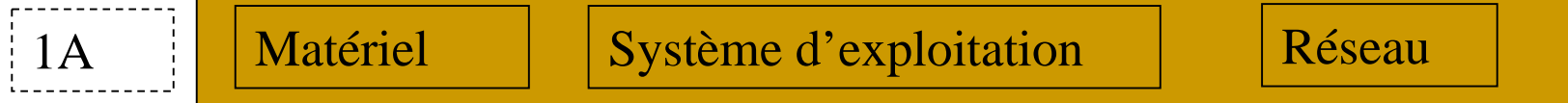
- Systèmes informatiques (1A) :
 - Aspects liés au Matériel
 - Architecture d'un processeur
 - codages des entiers
 - codages des rationnels
 - minimisation des fonctions booléennes

Systèmes informatiques : ISI



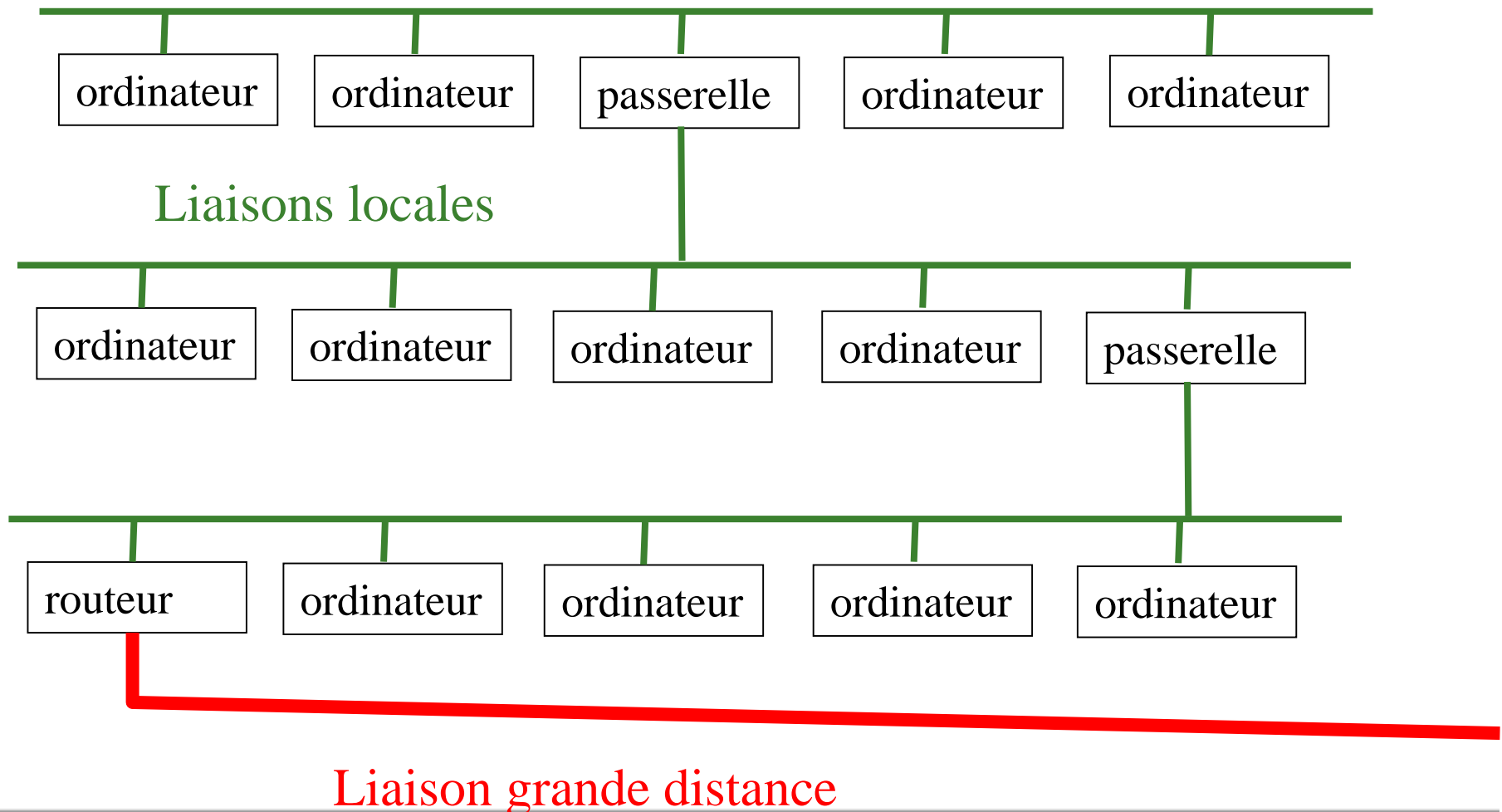
- **Systèmes informatiques (1A) suite**
 - **Système d'exploitation :**
 - interpréteur de commandes
 - chargement des programmes et bibliothèques
 - appels systèmes
 - gestion des ressources matérielles
 - processeur
 - mémoire
 - disques

Systèmes informatiques



introduction aux systèmes informatiques

architecture réseau



introduction aux systèmes informatique

pile de protocoles

Ordinateur 1

Ordinateur 2

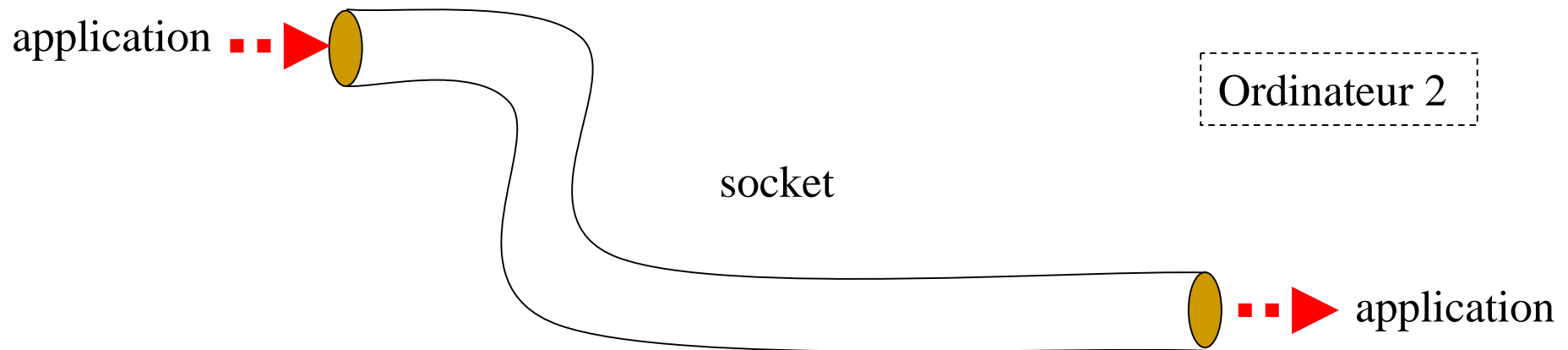
Niveau application
Niveau transport
Niveau réseau
Niveau liaison
Niveau physique

Niveau application
Niveau transport
Niveau réseau
Niveau liaison
Niveau physique

introduction aux systèmes informatiques

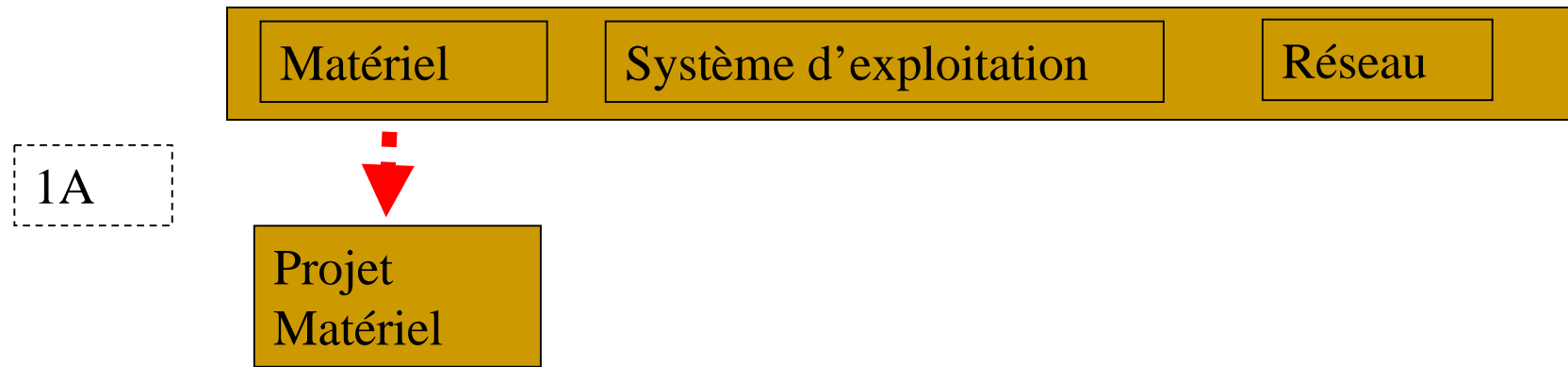
sockets

Ordinateur 1



Flux d'octets

Systèmes informatiques



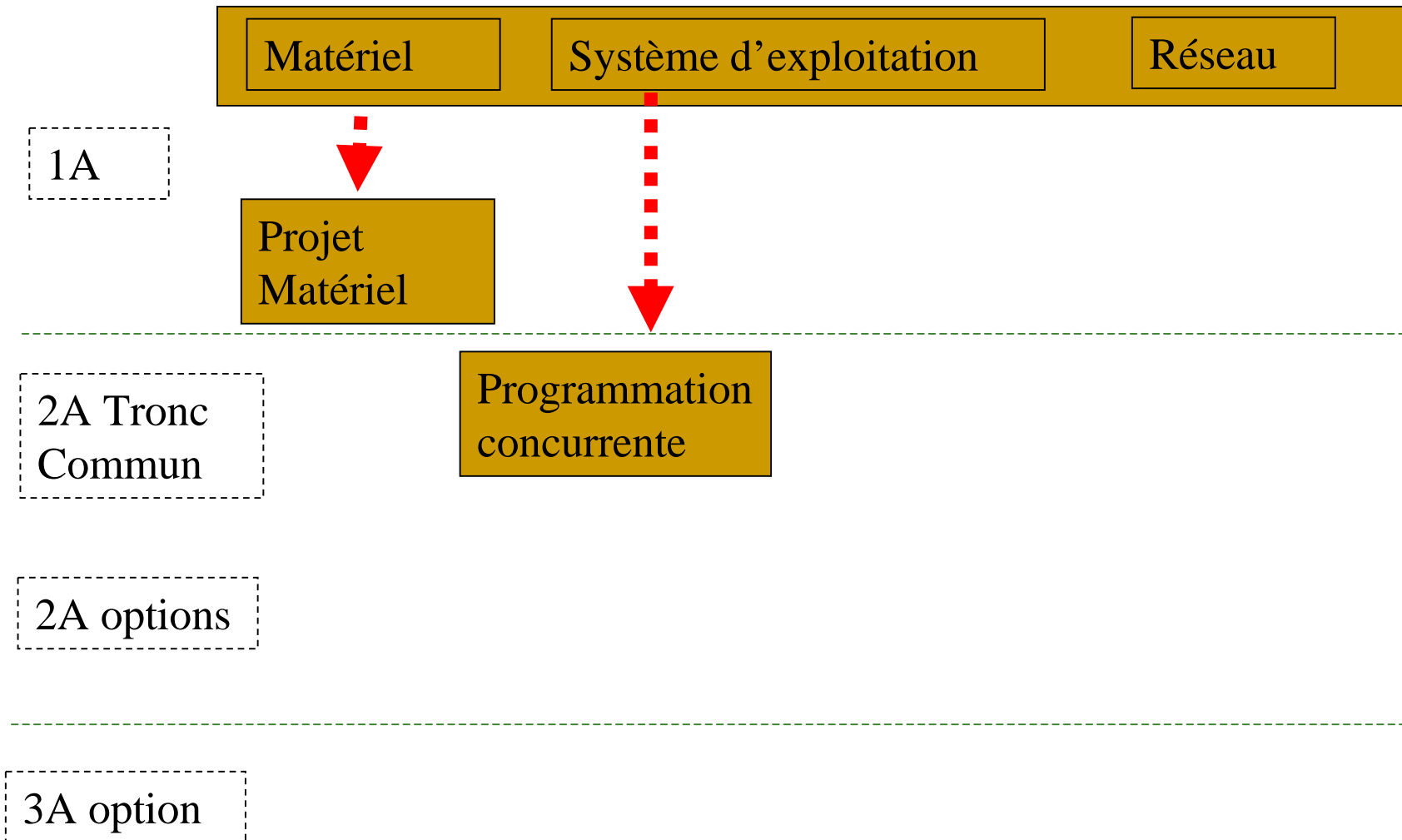
2A Tronc
Commun

2A options

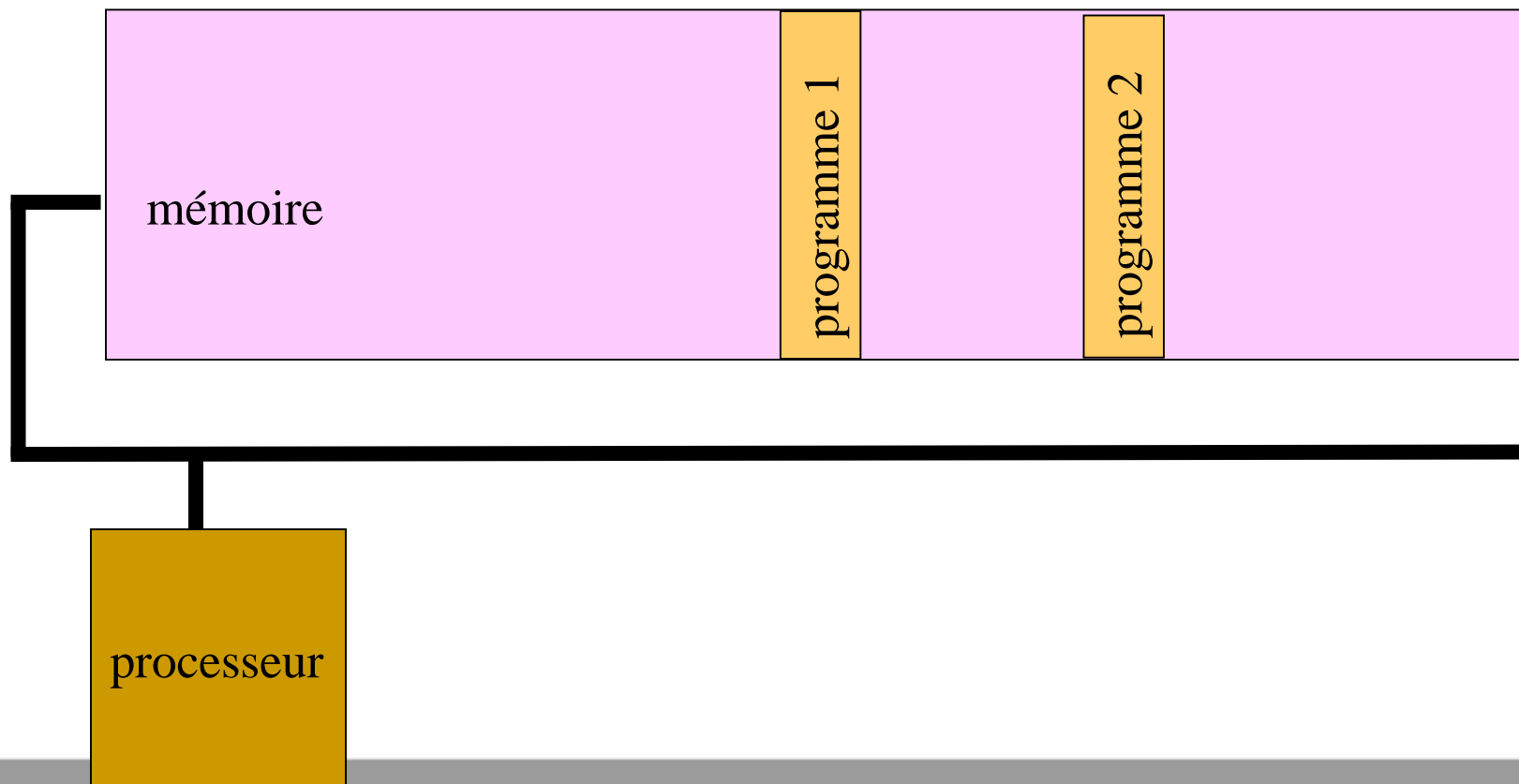
3A option

- **Projet matériel (1A) :**
 - composants matériels (circuits)
 - programmation assembleur

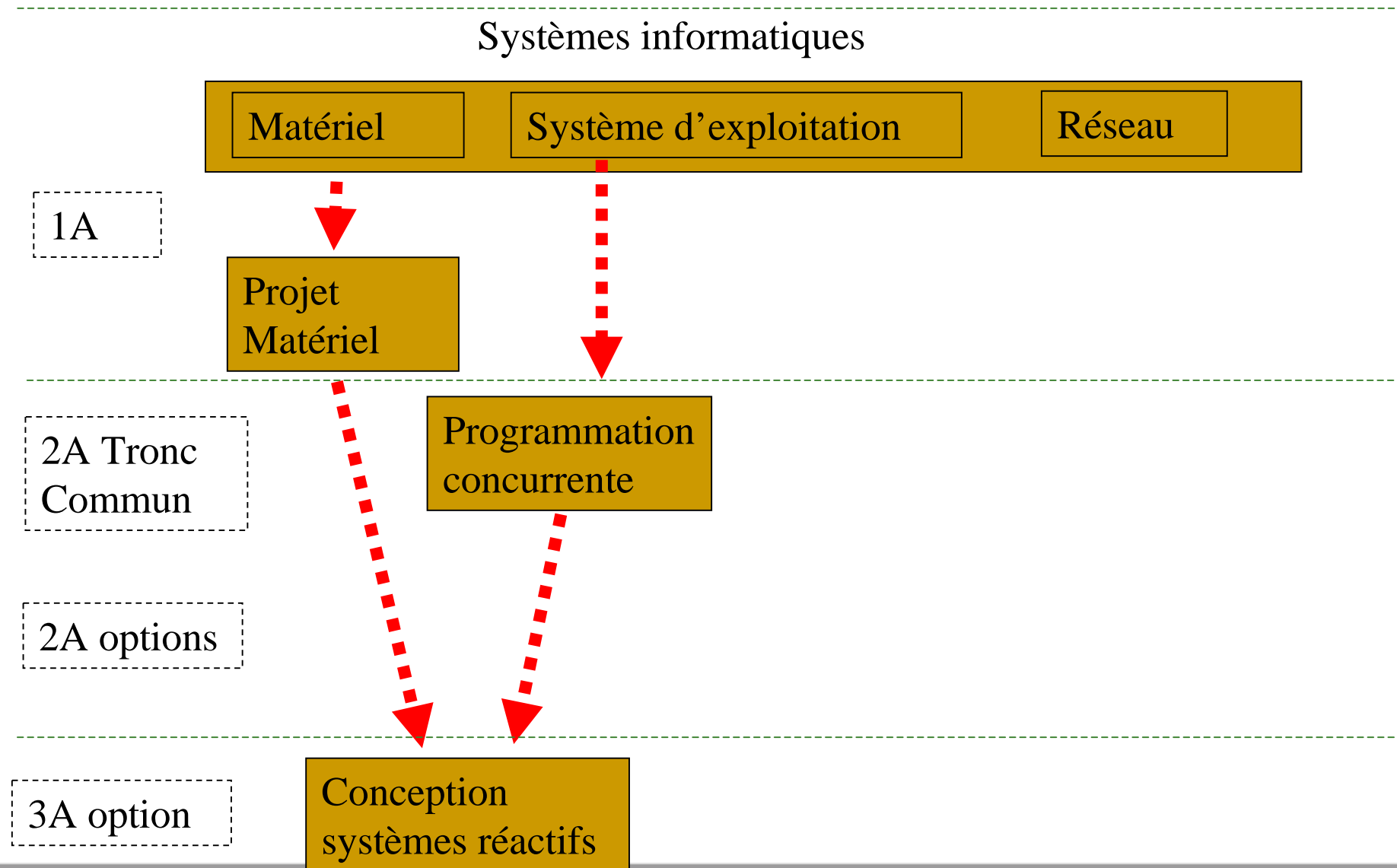
Systemes informatiques



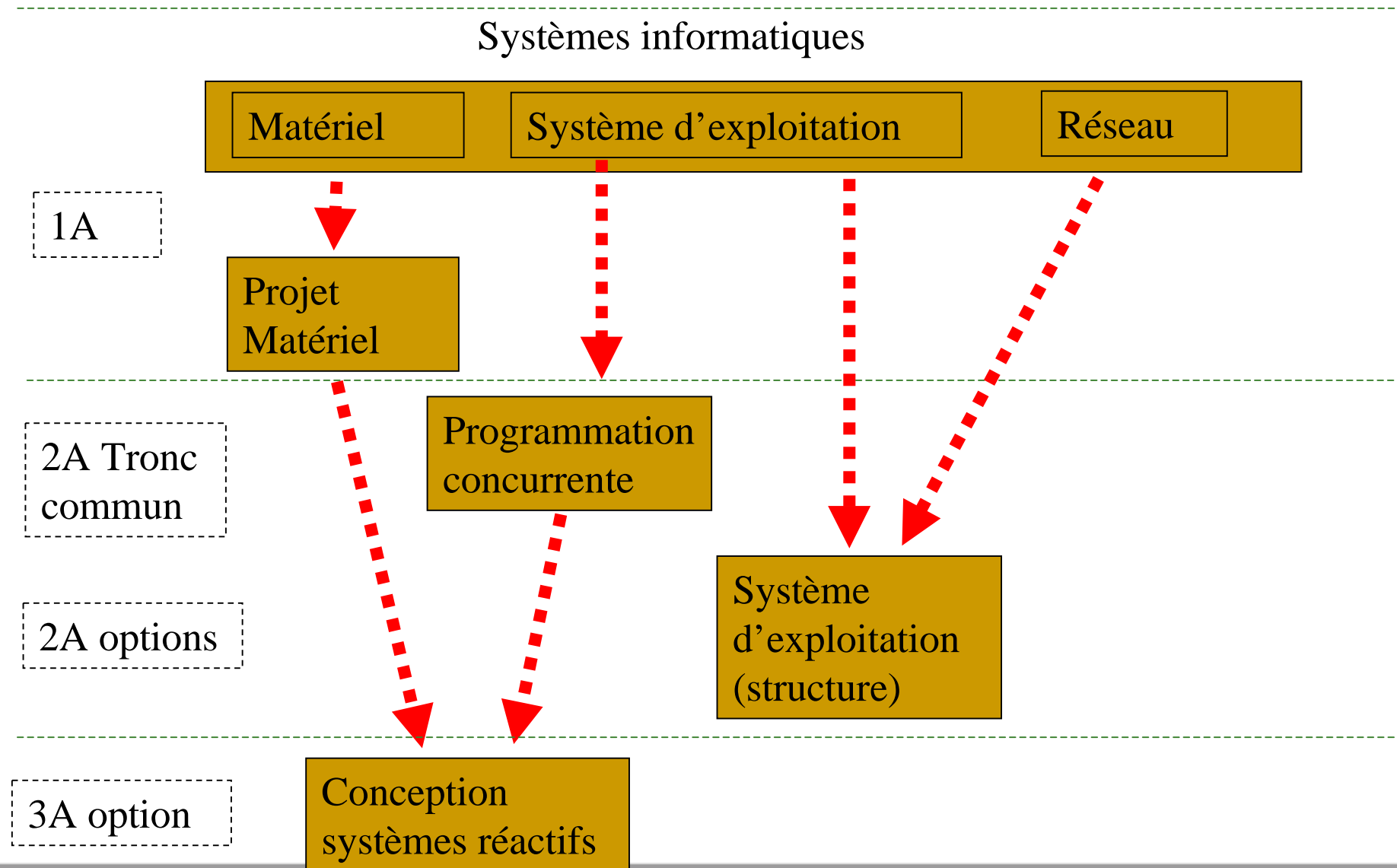
■ Programmation concurrente (2A)



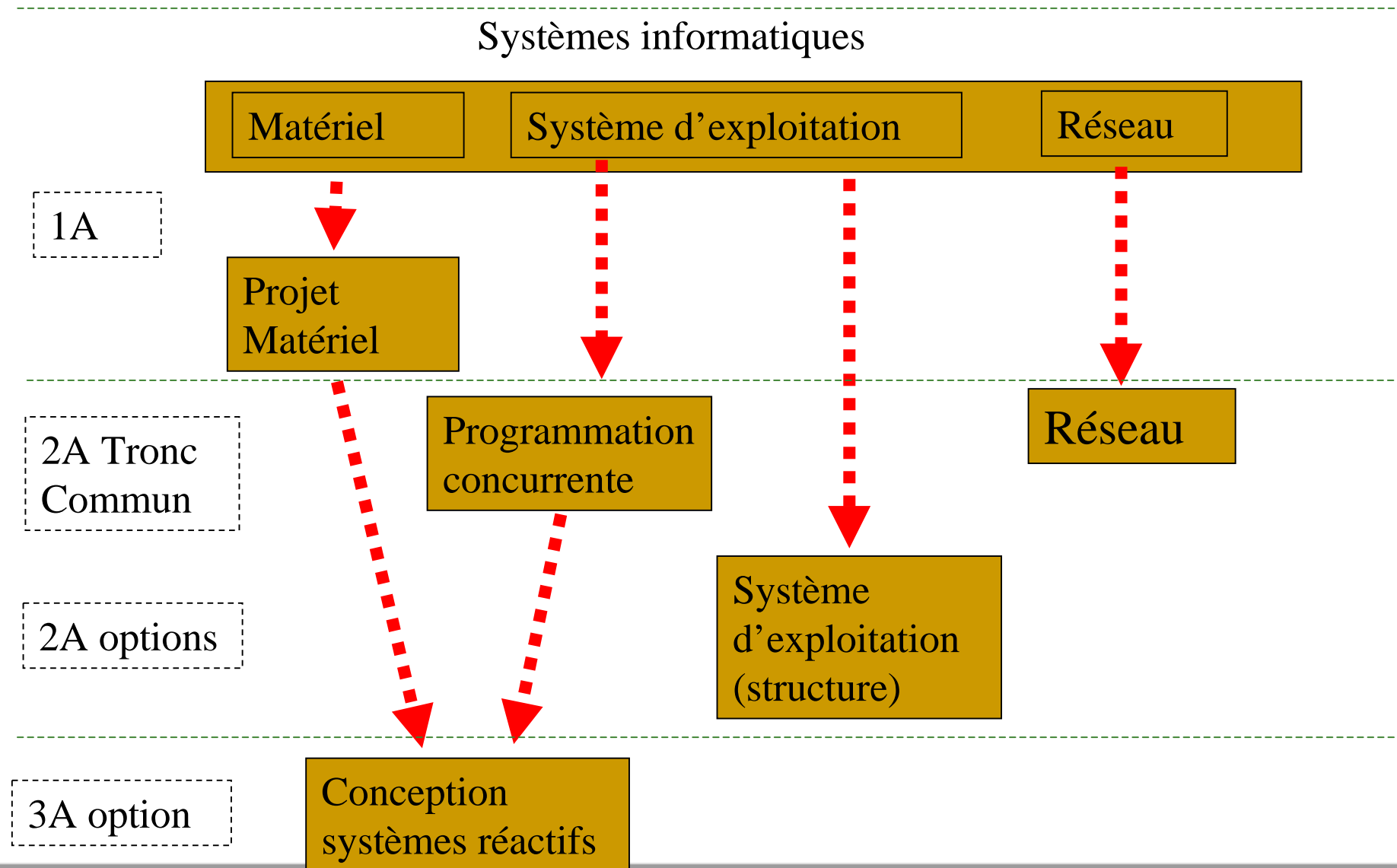
- Programmation concurrente (2A)
 - Qu'obtient-t-on lorsque deux programmes essayent de modifier la même information en même temps ? **des résultats incohérents sauf si on impose une exclusion mutuelle**
 - Que se passe-t-il lorsque plusieurs programmes doivent partager un ensemble fini de ressources ? **des interblocages**



- Conception Vérification Application
interactives ou Réactives (option 3A)
 - Conception et vérification de systèmes
embarqués ou réactifs
 - systèmes temps réel (pilotage d'un procédé)



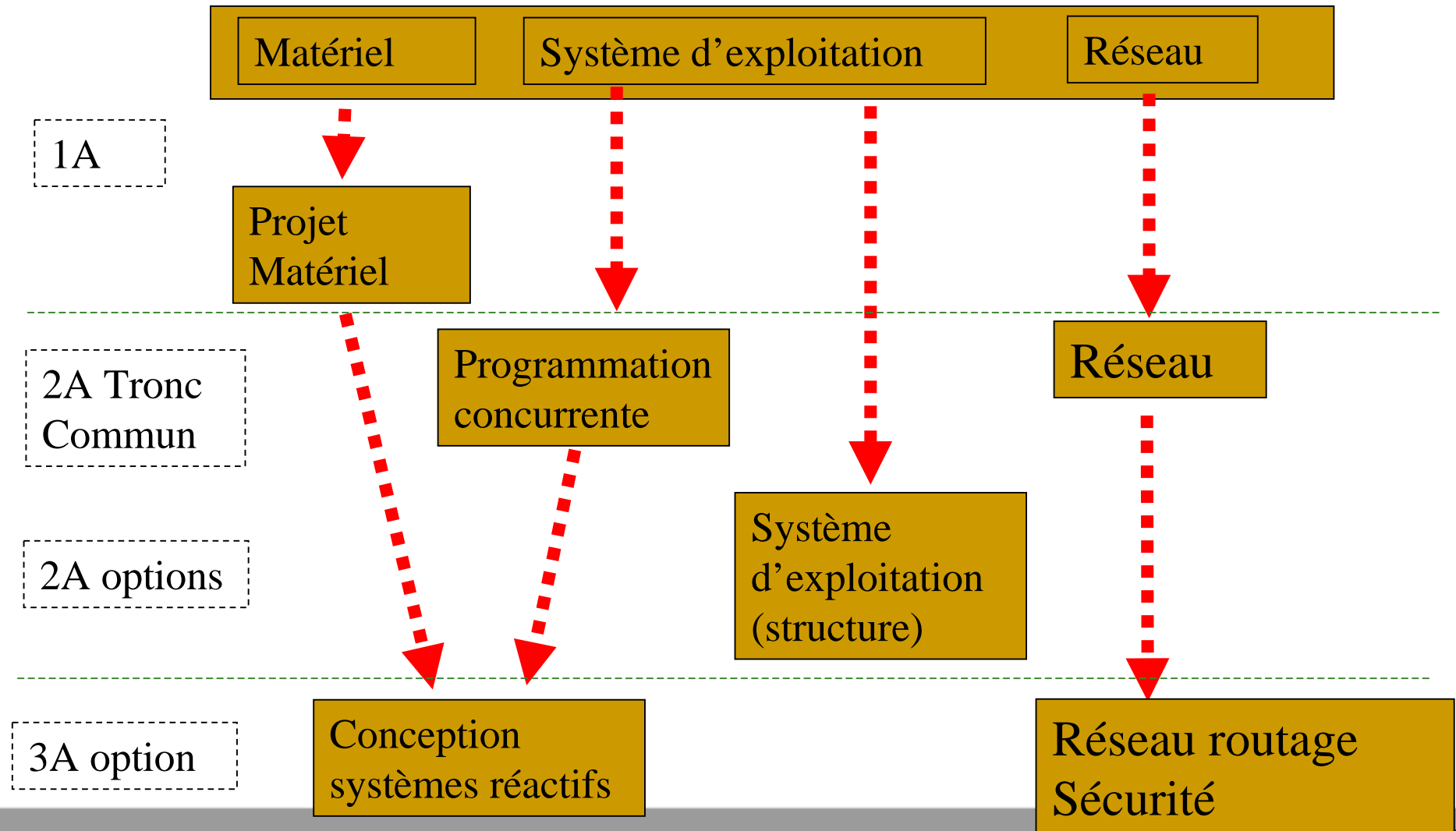
- Système d'exploitation et Administration de réseau local (option 2A)
 - anatomie d'un système d'exploitation (linux)
 - mise en place d'un réseau local et des services associés



■ Réseaux (2A)

- protocoles IP/TCP et compagnie
- construction d'application réparties (CORBA)

Systèmes informatiques



- Réseaux et multimédia (option 2A)
 - protocoles de routage
 - protocoles multimédia
 - cryptographie, authentification, sécurité

- Système informatique : dispositif qui permet d'exécuter des applications sur des collections de données pour créer d'autres collections de données. Le Système d'exploitation est le logiciel qui permet de démarrer les applications et de gérer les collections de données.
- notion de fichier : c'est une collection de données ou une collection d'instructions regroupées parce qu'elles vont logiquement ensemble.
- Les données d'un fichier sont éparpillées sur un disque (dur, disquette, clé USB) à des emplacements connus du seul système de fichiers (composant de l'OS) pour masquer ces détails pénibles à gérer.

prise en main des machines

- Il faut éviter de mettre tous les fichiers au même endroit: fouillis non listable, problème d'homonymie.
- Il faut faire du rangement en mettant les fichiers dans des containers à fichiers : les répertoires ou directory.
- Comme un répertoire peu contenir un fichier ou un répertoire on crée ainsi une espèce d'arborescence
- les répertoires utilisateurs sont regroupés dans /home/. Ensuite un utilisateur doit continuer à classer par matières/programmes, textes etc.

prise en main des machines

- interface graphique versus interface console (terminal, shell, invite de commandes).
- Une IG est plaisante. Attractive pour un utilisateur novice ou occasionnel (le grand public).
- grave inconvénient pour les informaticiens : on ne peut automatiser les traitements (ce qui est la raison d'être de l'informatique). Exemples :
 - renommer 500 fichiers
 - installer un logiciel sur 25 machines
 - un élève de l'ENSIIE compile des milliers de programmes et à chaque fois il faut mettre des options et donner des noms aux fichier produits.
- Ceci NE PEUT SE FAIRE **SIMPLEMENT** AVEC une IG.

prise en main des machines

commandes de base

- **ls** pour voir le contenu d'un répertoire, **ls | less** pour ne pas être submergé si trop de fichiers
- **man** pour voir les indications sur une commande. voir aussi **ls --help**
- **mkdir** pour créer un répertoire, **rmdir** pour supprimer
- **cd** pour changer le répertoire de travail , **cd ..** pour remonter d'un niveau, **cd -** pour le précédent
- **pwd** pour savoir où on se trouve dans l'arborescence
- **mv** pour déplacer un fichier
- **cp** pour copier un fichier, **scp** pour un fichier distant
- complétion des noms par la touche "**tab**"
- rappel des commandes précédentes par "**arrowup**"

BIBLIOGRAPHIE

- ARCHITECTURE DES MACHINES ET DES SYSTEMES INFORMATIQUES 3EME EDITION
 - CAZES (DUNOD)
 - ISBN: 978-2-10-052087-9 (08/2008)
- ARCHITECTURE DE L'ORDINATEUR SYNTHEX
 - LAZARD (PEARSON EDUCATION)
 - ISBN: 978-2-7440-7378-6 (01/2009)
- ARCHITECTURE DE L'ORDINATEUR + CORRIGES
 - TANENBAUM (PEARSON EDUCATION FRANCE)
 - ISBN: 978-2-7440-7183-6 (03/2006)
- Architecture des ordinateurs, une approche quantitative, 3e éd.
 - David a. PATTERSON, John I. HENNESSY (VUIBERT 8/2003)

BIBLIOGRAPHIE

- PRINCIPES DES SYSTEMES D'EXPLOITATION 6EME EDITION
 - SILBERSHATZ (VUIBERT INFORMATIQUE)
 - ISBN: 978-2-7117-4869-3
- SYSTEMES D'EXPLOITATION DES ORDINATEURS
 - BLOCH (VUIBERT INFORMATIQUE)
 - ISBN: 978-2-7117-5322-2 (02/2003)
- CONCEPTION DE SYSTEMES D'EXPLOITATION: LE CAS LINUX
 - CEGIELSKI (EYROLLES)
 - ISBN: 978-2-212-11479-9 (09/2004)
- LES SYSTEMES D'EXPLOITATION
 - BOUZEFRANE (DUNOD)
 - ISBN :978-2-10-007189-0 (10/2003)

BIBLIOGRAPHIE

- ARCHITECTURE DE L'ORDINATEUR 4EME EDITION
 - TANENBAUM (DUNOD)
 - ISBN: 978-2-10-005158-8 (01/2001)
- SYSTEMES D'EXPLOITATION 3EME EDITION
 - TANENBAUM (PEARSON EDUCATION FRANCE)
 - ISBN: 978-2-7440-7299-4 (09/2008)
- RESEAUX 4EME EDITION
 - TANENBAUM (PEARSON EDUCATION FRANCE)
 - ISBN: 978-2-7440-7001-3 (05/2003)

prise en main des machines

commandes de base

- `ls` pour voir le contenu d'un répertoire , avec `ls | less` pour ne pas être submergé si trop de fichiers
- `man` pour voir les indications sur une commande. voir aussi `ls --help`
- `mkdir` pour créer un répertoire, `rmdir` pour supprimer
- `cd` pour changer le répertoire de travail , `..` pour remonter d'un niveau (et aussi `cd -`)
- `pwd` pour savoir où on se trouve
- `mv` pour déplacer un fichier
- `cp` pour copier un fichier
- complétion des noms par la touche "tab"
- rappel des commandes précédentes par `arrowup`
- `tar`, `ssh`, `scp` pour travailler sur une autre machine que la sienne

- Faire un répertoire contenant plusieurs centaines de fichiers de types variés : .doc .txt .tmp .jpg .html (utiliser le script générateur qui marche sous bash) et mélangés puis montrer comment on peut le ranger rapidement avec quelques commandes simples (ça peut se faire sous cygwin, sauf que l'arborescence est bizarre et que les protections ne marchent que pour le propriétaire, le système FAT ne gérant pas les accès par plusieurs utilisateurs différents) . Basculer entre fenêtre et console avec alt-tab pour visualiser.
- ls pour voir le contenu, avec ls | less pour ne pas être submergé si trop de fichiers
- man pour voir les indications puis taper /recurs pour rechercher la rubrique concernant la récursivité. voir aussi ls --help
- mkdir texte, mkdir texte/document pour créer des répertoires
- cd pour changer le répertoire de travail , expliquer le .. pour remonter (et aussi cd -)
- pwd pour savoir où on se trouve
- mv *.txt texte pour déplacer
- cp pour copier
- montrer qu'on peut faire les commandes à distance sans faire des cd tout le temps
- montrer l'expansion de nom par tab et le rappel des commandes passées par arrowup
- montrer qu'on peut faire les commandes à distance sans faire des cd tout le temps
- rmdir,
- tar, ssh, scp pour travailler sur une autre machine que la sienne
- faut-il expliquer le mécanisme des fichiers personnels centralisés sur un serveur et accessibles de partout.