

Logiciel de Base

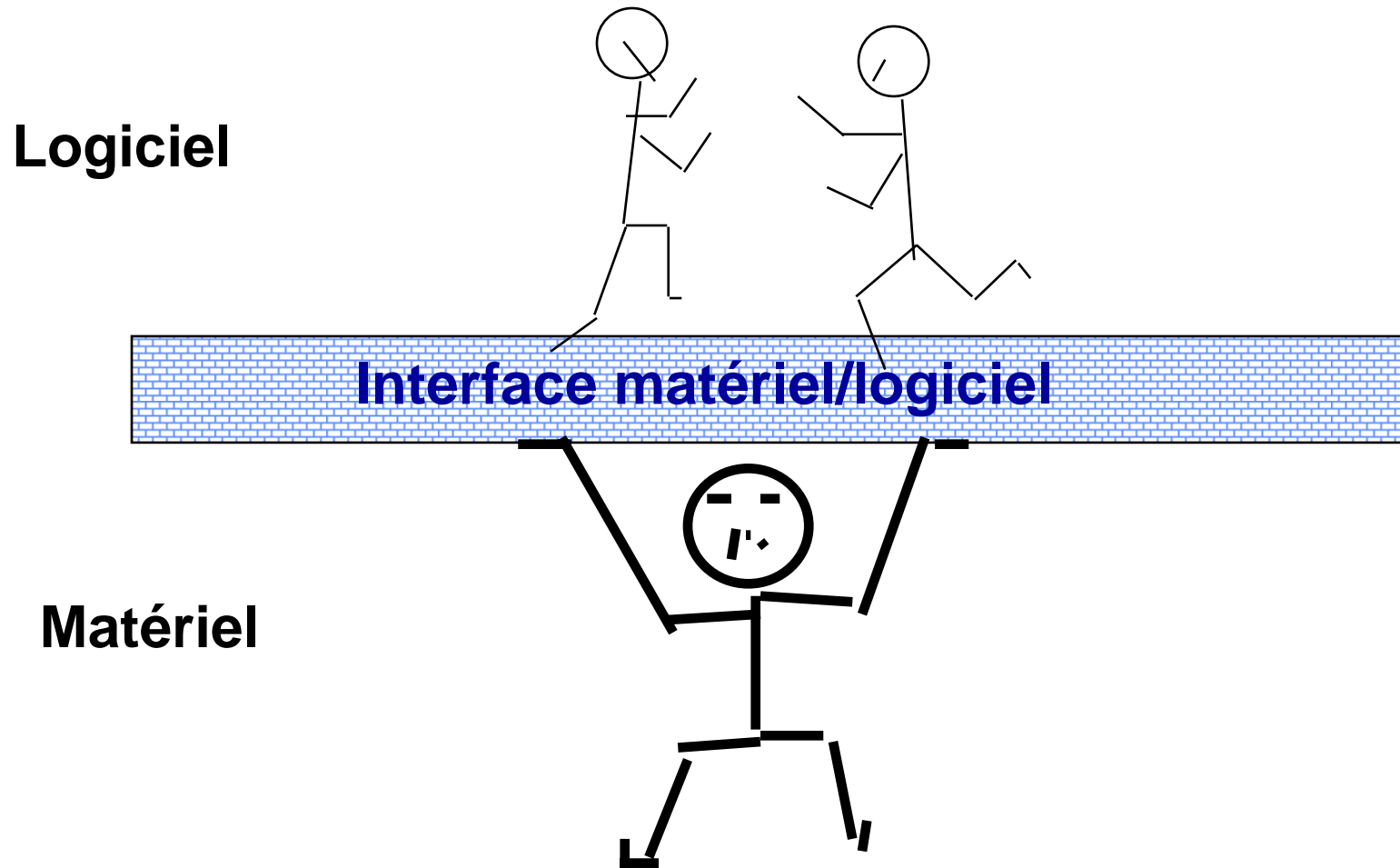
A1- 06/07

Léon Mugwaneza

ESIL/Dépt. Informatique (bureau A118)
mugwaneza@univmed.fr

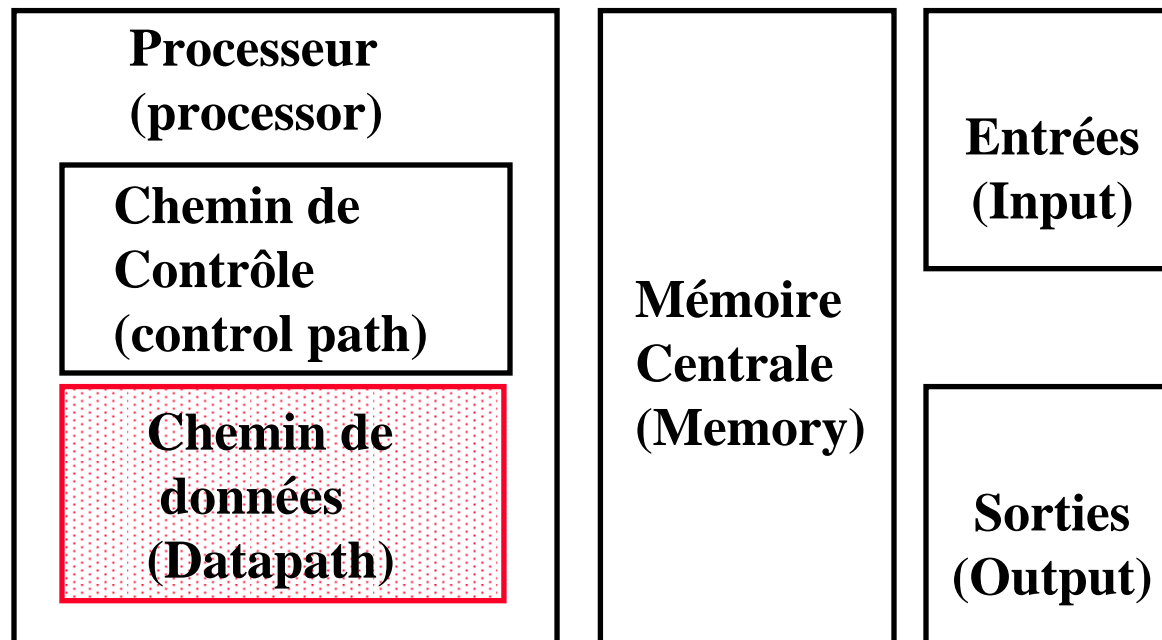
Introduction

Ordinateur = matériel + logiciel



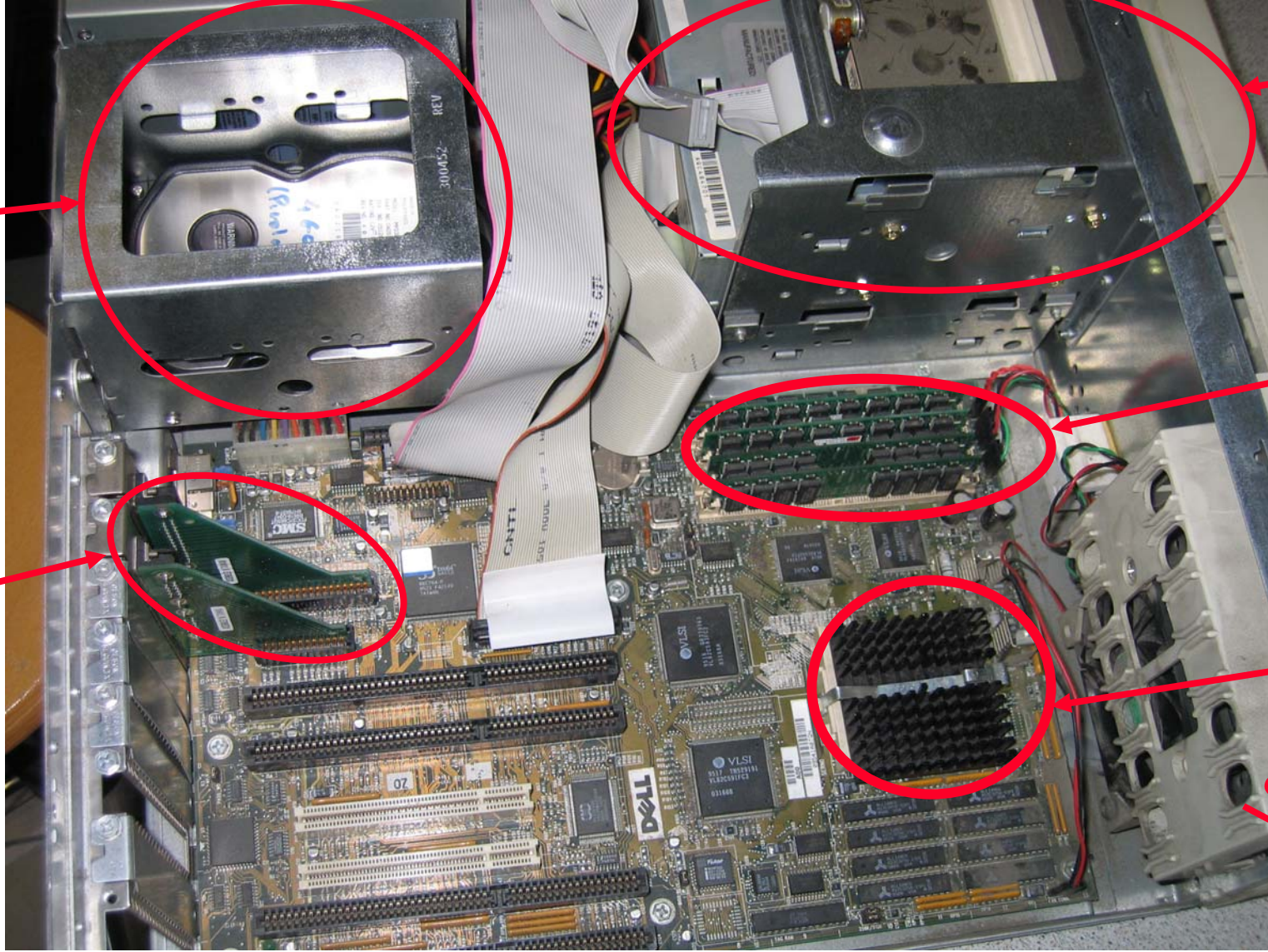
Le matériel

- Les 5 composants d'un ordinateur



Le matériel

Vers écran, clavier, ... disque dur



lecteurs CD et disquette

La mémoire centrale

Le processeur

Un processeur Pentium

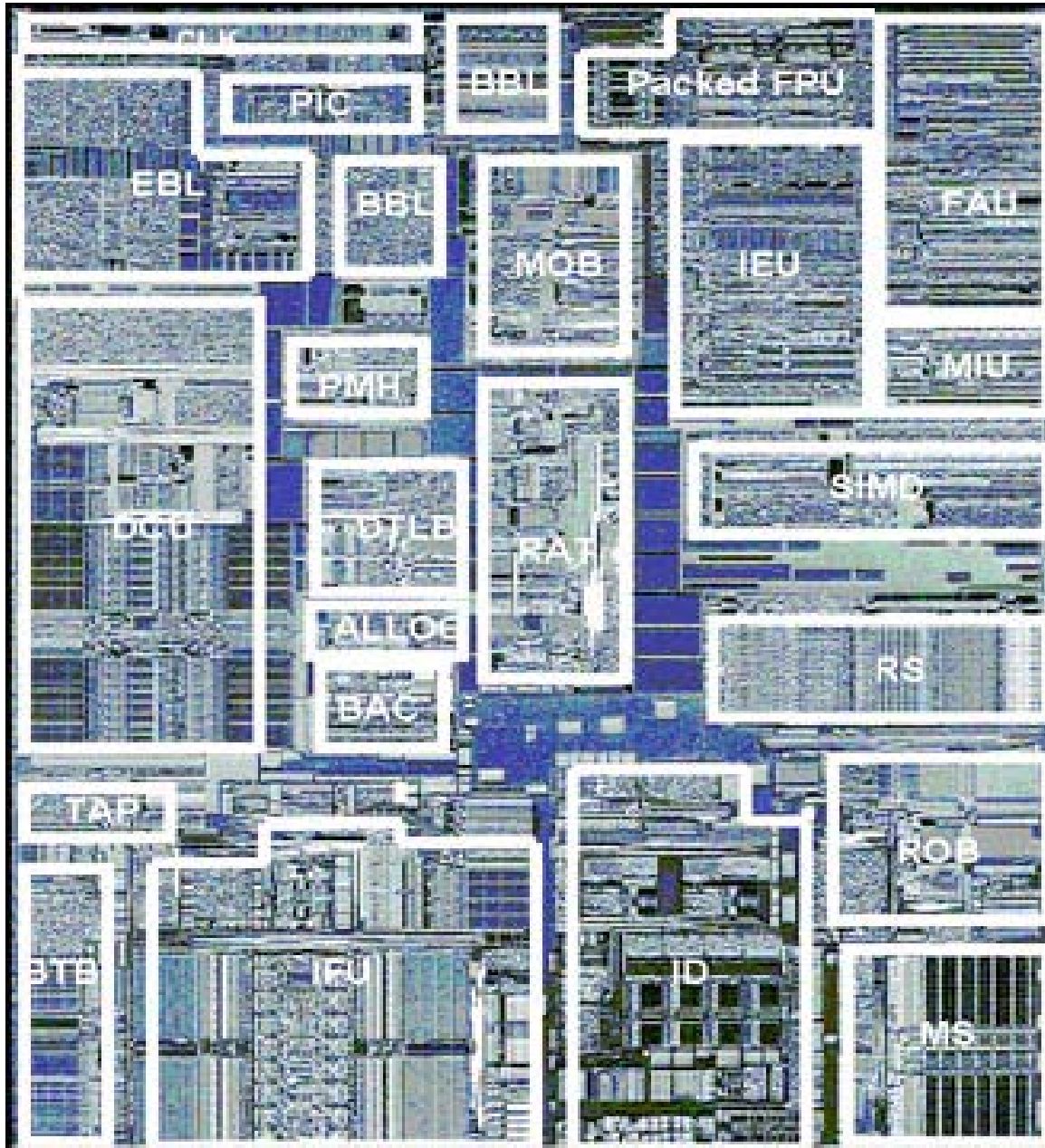


De dessus



De dessous

un processeur pentium III (de dedans)



L'ESIL, le campus, et Marseille vus du Mt Puget



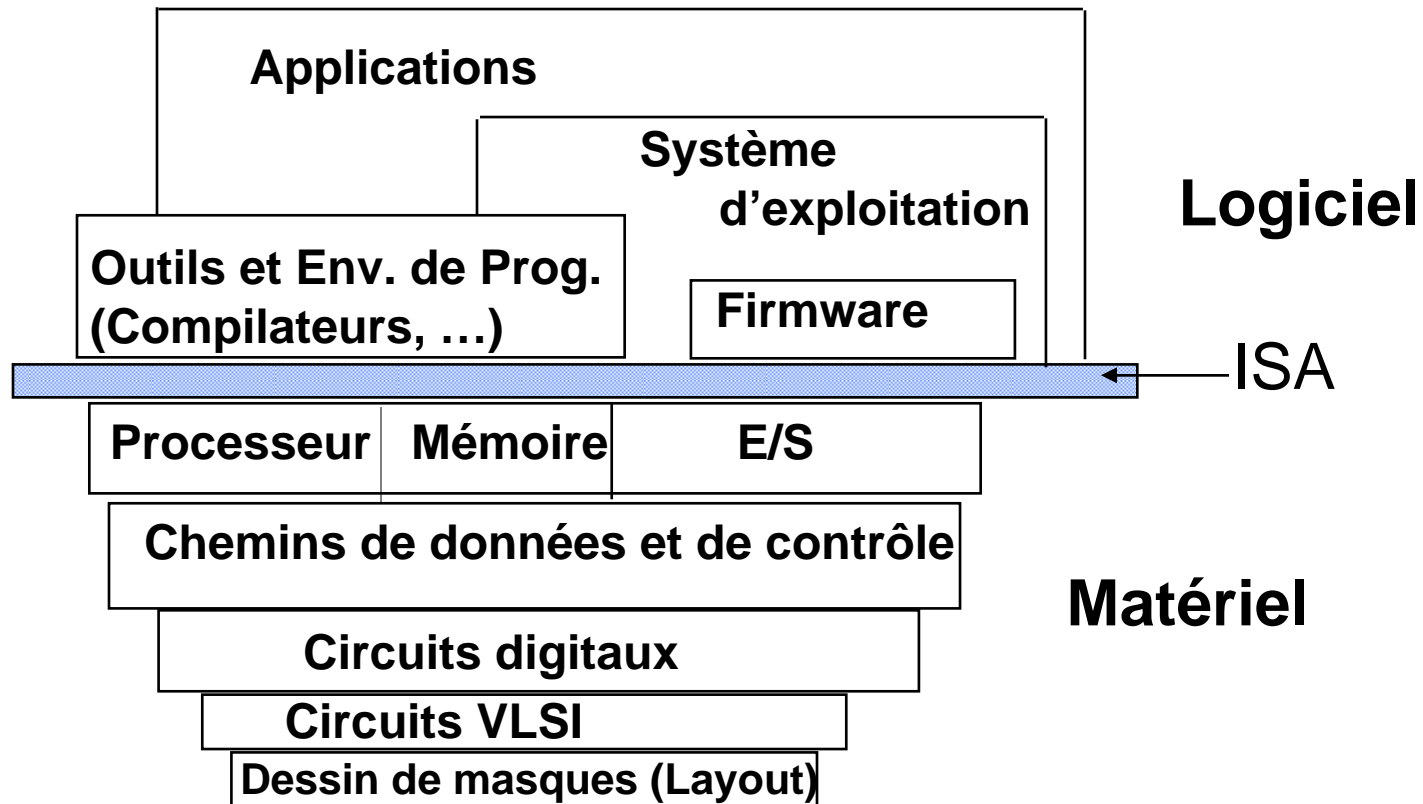
Les 5 composants d'un ordinateur

- Le processeur lit les instructions et les données de la mémoire, écrit les données en mémoire
- Les unités d'entrée écrivent les données dans la mémoire
- Les unités de sortie lisent les données de la mémoire
- Le chemin de données comprend des opérateurs pour implanter le répertoire d'instructions (partie opérative)
- L'unité de contrôle génère les commandes pour la réalisation des opérations sur le chemin de données, la mémoire, les entrées et les sorties

Q : Interface offert par le matériel ?

**⇒ Jeu d'instructions (ISA: Instruction Set Architecture)
(on parle aussi de Langage Machine)**

“Structure” des ordinateurs



- **Plusieurs niveaux d'abstraction**
- **Le logiciel de base est l'ensemble des programmes entre les applications et le matériel**

Un ordinateur : plusieurs « couches »

Addition de 2 et 3

C=A+B;

ldc num1
ldc num2
add

Application

Langage haut niveau

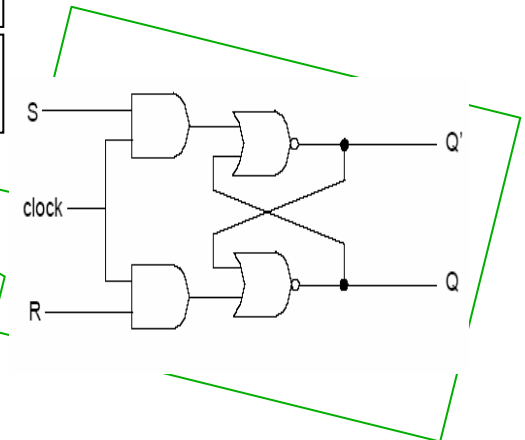
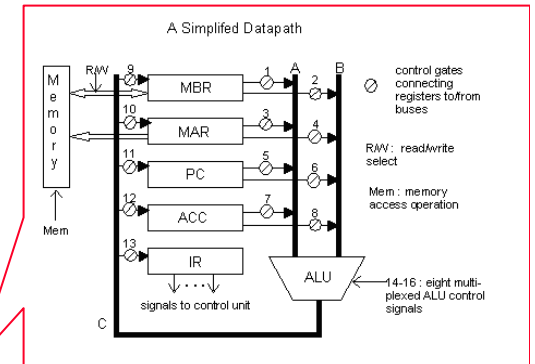
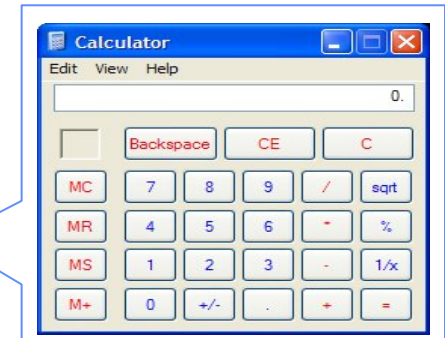
Langage d'assemblage

Machine conventionnelle

Microarchitecture

Circuits digitaux

10011101111000111011111110010000	9DE3BF90
10010010000100000010000000000000	92102000
10010000000100000010000000000001	90102001
10010010000000100100000000001000	92024008
10010000000001000100000000000001	90022001
10000000101000100010000001100100	80A22064
00100100101111111111111111111110	24BF7FFE
10010010000000100100000000001000	92024008
00010001000000000000000000000000	11000000

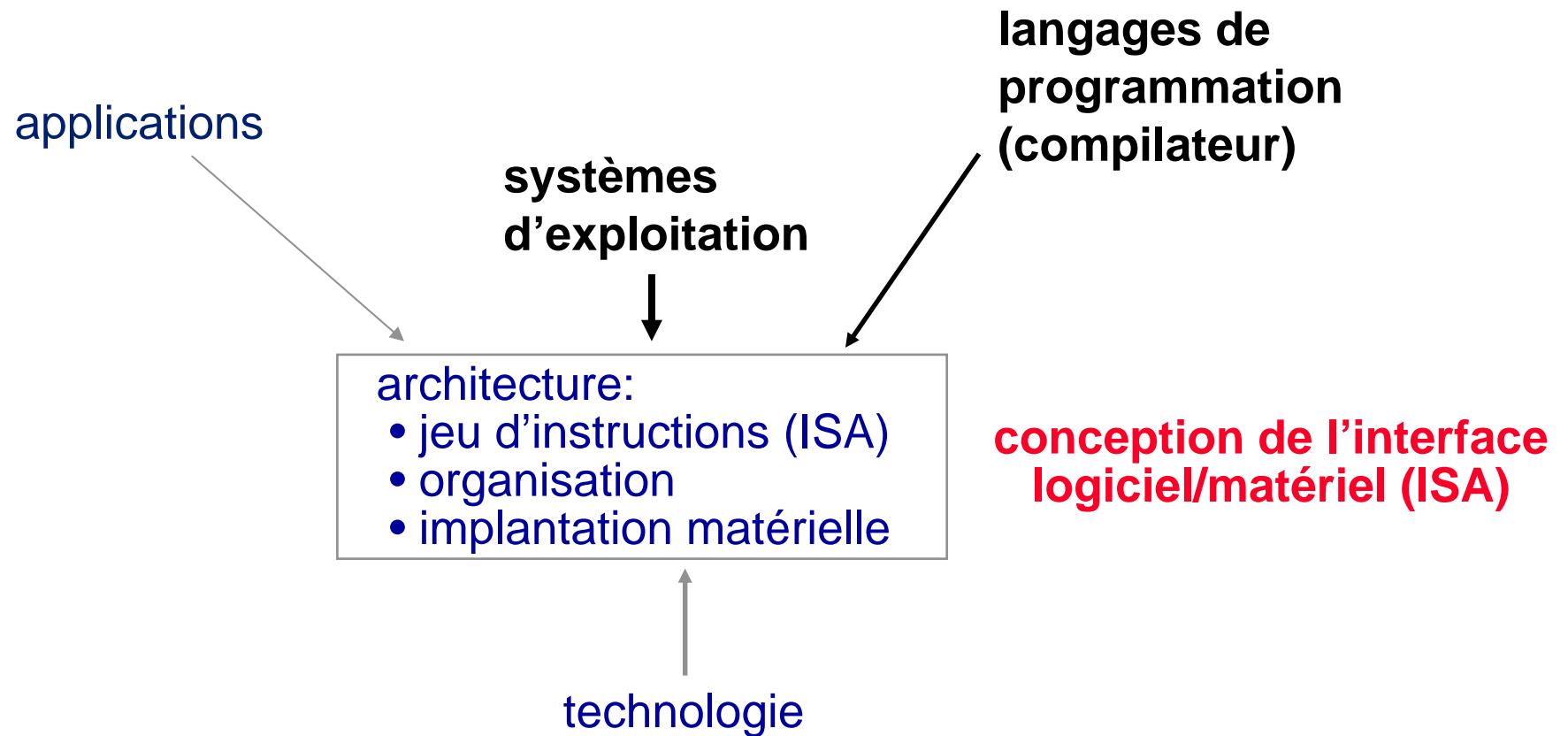


Le logiciel de base

- **Outils pour la préparation des programmes à l'exécution**
 - compilateurs (gcc, cc, gpc, javac, ...)
 - assembleurs (as, gas, nasm, spim, ...)
 - éditeurs de liens (ld, ...)
 - interpréteurs (lisp, perl, jvm, sh, ...)
 - **Le système d'exploitation (operating system)**
 - gestion mémoire, gestion des entrées et sorties,
 - gestion des fichiers
 - partage du processeur
 - **Le SE assure 2 fonctions :**
 - Virtualisation du matériel (cache les détails du matériel en définissant un interface de plus haut niveau, un ordinateur plus simple à utiliser)
 - la gestion, le partage et la protection des “ressources”
- 👉 **Note** : pour certains, le SE comprend tout ce que nous appelons ici «logiciel de base» (des fois même certaines applications)

Objectif du cours

Comprendre l'interface entre le Logiciel et le Matériel



Contenu du cours

- **Préparation d'un programme pour exécution (interface compilateur)**
 - Langage machine (ISA)
 - Langage d'assemblage
 - Traduction des langages de « haut niveau »
 - L'assemblage
 - L'édition des liens et le chargement
- **Mécanismes à la base des systèmes d'exploitation**
 - Exceptions (appels système, erreurs, interruptions)
 - Notion de processus (multiprogrammation)
 - gestion de la mémoire (mémoire virtuelle)
 - gestion des entrées/sorties
 - protection

👉 avant de commencer : représentation des nombres

Articulation avec les autres cours?

- **Pré-requis pour le cours de logiciel de base :**
 - **A1 - Introduction à la programmation**
 - » connaissance d'un langage de haut niveau, style impératif (C, Pascal, ...)
 - **A1 - Introduction à l'informatique (unix)**
 - » utilisation d'un système d'exploitation
 - niveau utilisateur «curieux» («conscient»)
- **Le cours de logiciel de base est un pré-requis pour :**
 - **A1 - Architecture des ordinateurs**
 - » jeux d'instructions (illustration sur le MIPS)
 - **A2 - Compilation**
 - » traduction des langages de haut niveau
 - **A2 - Systèmes d'exploitation**
 - » mécanismes à la base des systèmes d'exploitation
 - **A2 - Architectures avancées**
 - » jeux d'instructions (illustration sur le MIPS)

Bibliographie

- **Hennessy and Patterson, « *Computer Architecture & Design : The Hardware/Software Interface*», 2nd Ed., (Morgan Kaufman Publishers, 1997) [MIPS - simulateur spim-, Dec Vax 11] (Annexe A fournie) (1 ex. à la BU)**
 - ☛ **3^{ème} édition sortie en 2005 (le cours est illustré à en utilisant comme exemple le jeu d'instructions de l'architecture MIPS-I décrit dans l'annexe A de la 2^{ème} édition)**
- **Bryant and O'Hallaron, «Computer Systems : A programmer 's perspective», (prentice hall 2003) [INTEL IA32]**
- **Andrew Tanenbaum, «*Structured Computer Organization*», 4th Ed., (Prentice Hall, 1999) [ultraSparc II, Sun JVM, Pentium II] (1 ex. à la BU)**
- **A. J. van de Goor , «*Computer Architecture & Design*», (Addison Wesley, 1995) [Motorola 68020 et Dec Vax 11]**
- **Kane and Heinrich, « *MIPS RISC Architecture*», (Prentice Hall, 1994)**
 - plus généralement les livres de référence des processeurs
- **Paul Amblard et al., « *Architectures logicielles et matérielles*», (Dunod, 2000) [Sparc, Motorola 68000]**
- **Andrew Tanenbaum, « *Architecture de l 'ordinateur*», 3^{ème} ed. (Interedition,1991) [motorola 680x0] (des exemplaires à la BU)**

Organisation pratique du cours

- **Cours de «logiciel de base»**
 - 30 heures de cours
 - 30 heures de TD (2 groupes)
 - les fiches de TD sont données avant la séance
 - ⇒ à préparer avant la séance
 - 12 heures de TP (4 groupes)
 - 6 séances de 2h à partir du 28/11/2006
 - les fiches de TP seront données avant la séance
 - **Evaluation :**
 - 1 examen écrit à la fin du cours (fin Janvier 2007)
 - ECTS : 5
 - des points de participation (rajoutés à la note d'examen)
 - passage volontaire au tableau en TD
 - réponses aux questions «rémunérées» (en cours, TD, TP)