

TACT factory mobile agency



Fournisseur **souple** et **réactif**
d'applications mobiles innovantes
avec une **méthodologie** projet
rigoureuse.

Présentation

Mickael Gaillard (Architecte logiciel)

mickael.gaillard@tactfactory.com

Yoan Pintas (Lead Developer)

yoan.pintas@tactfactory.com

Erwan LeHuitouze (Lead Developer)

erwan.lehuitouze@tactfactory.com



Planning session 1

Le système d'exploitation :

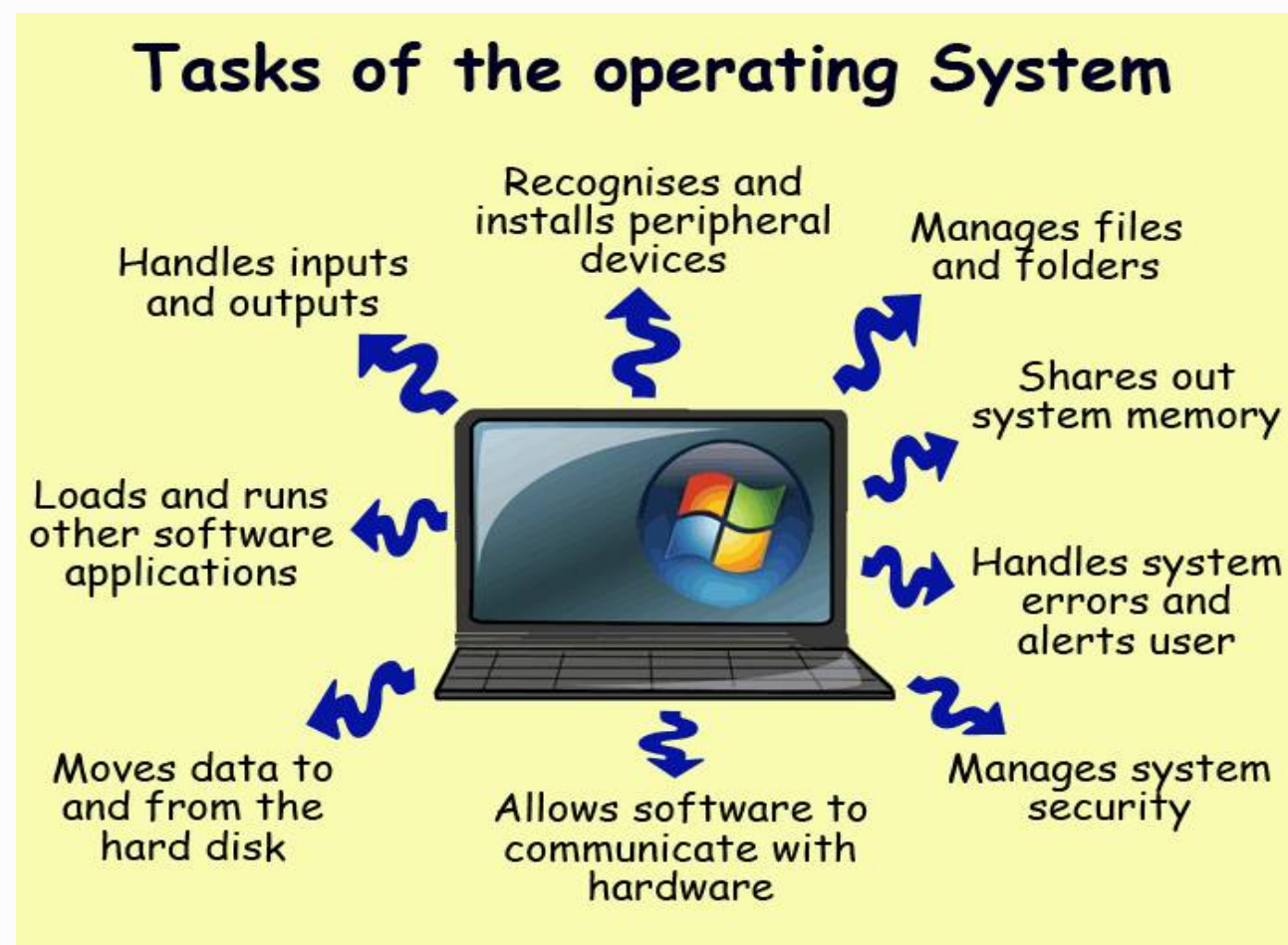
- Définition
- Les types d'OS
- L'architecture
- Le noyau
- Historique et évolution
- Les OS mobiles

Architecture ARM

- CISC vs RISC
- Historique
- Version ARM (Architectures/Familles)
- Business Model
- Jeux d'instructions
- ARM et mobilité

Operating System (système d'exploitation)

Définition : Ensemble de programmes qui assurent et gèrent la liaison entre les ressources matériel, l'utilisateur et les applications.



Les types

Première génération : Traitement par lots => Tâches exécutées en lot les unes après les autres

Deuxième génération : la multiprogrammation => Tâches exécutées en parallèles

Troisième génération : le temps partagé => Répondre rapidement à plusieurs demandes envoyées simultanément

Quatrième génération : le temps réel => Répondre aux demandes dans les délais déterminés

Cinquième génération : les systèmes distribués => Utilisation de plusieurs ordinateurs

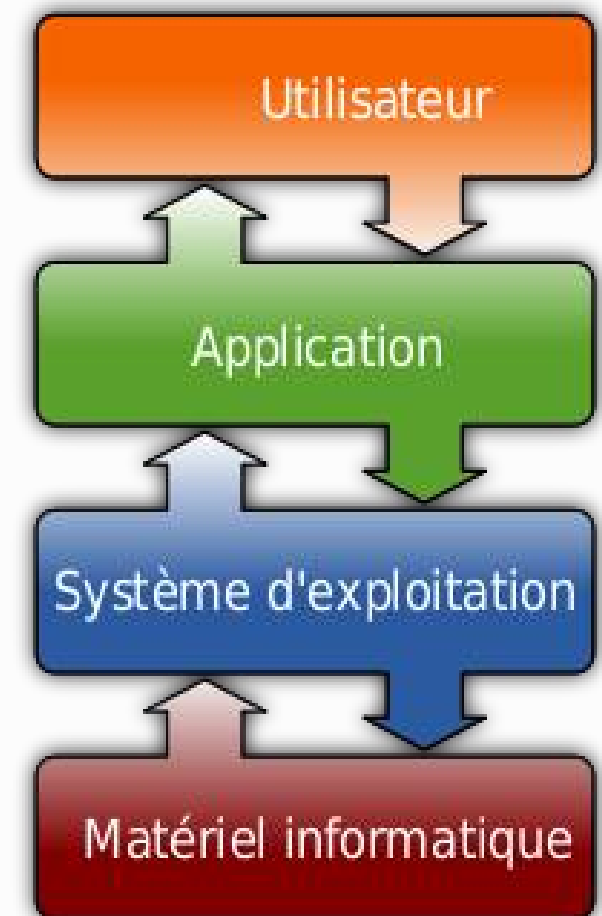
Architecture d'un OS ⁽¹⁾

Les acteurs :

le matériel : Motherboard - CPU - RAM - GPU - HDD - Devices

Les applications : Internet - Bureautique - Jeux vidéo

L'utilisateur : IHM (Interface Homme Machine)



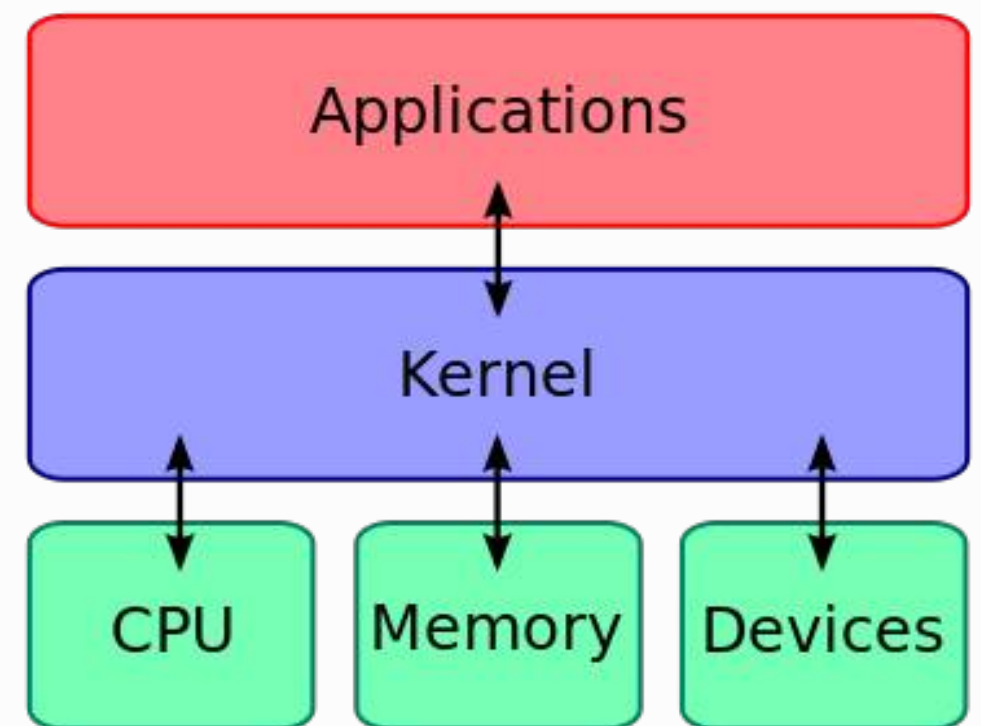
Architecture d'un OS ⁽²⁾

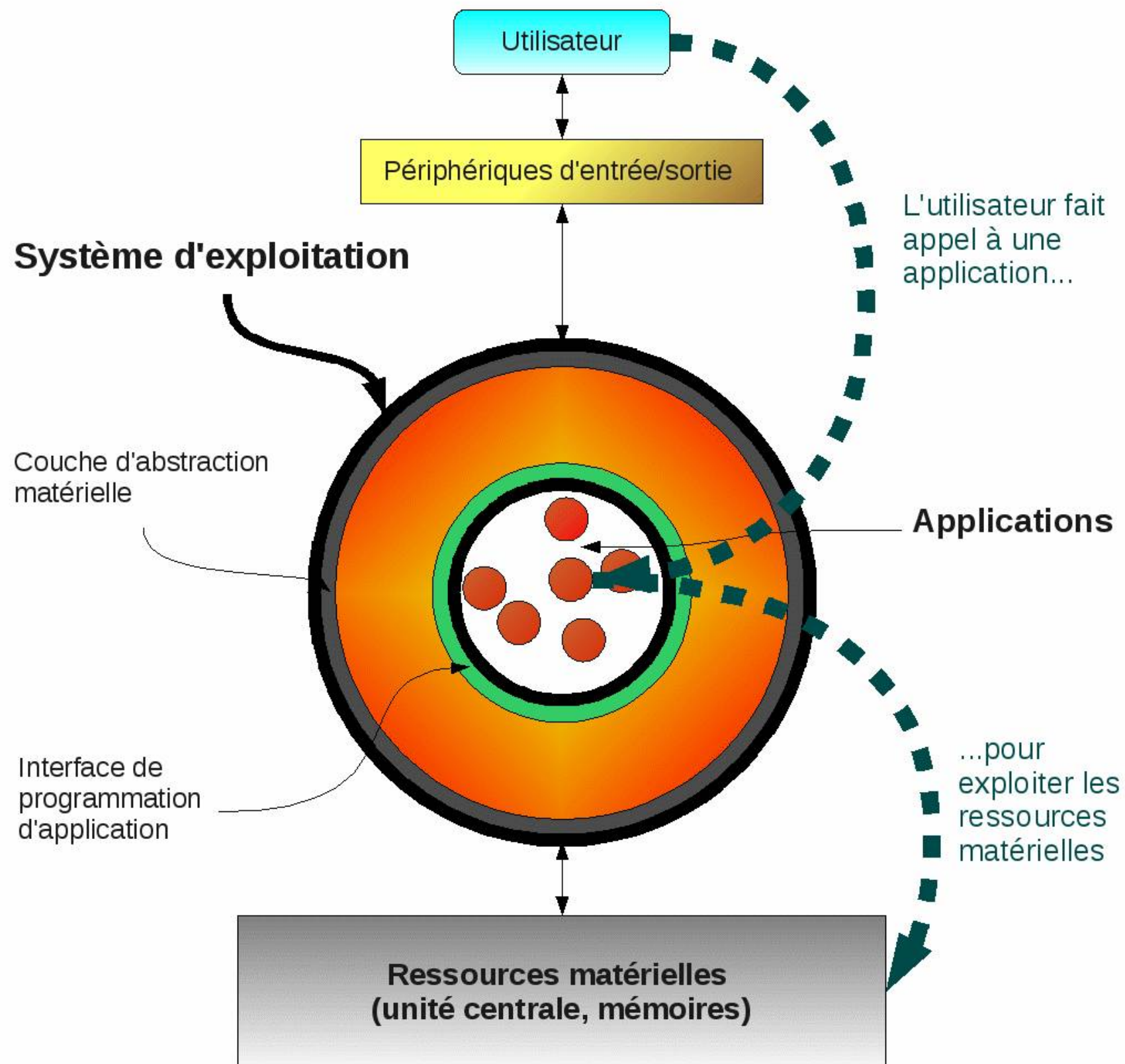
Les couches :

Applications (User Space) : interface de programmation avec les logiciels applicatifs y compris l'API fournie par le système d'exploitation

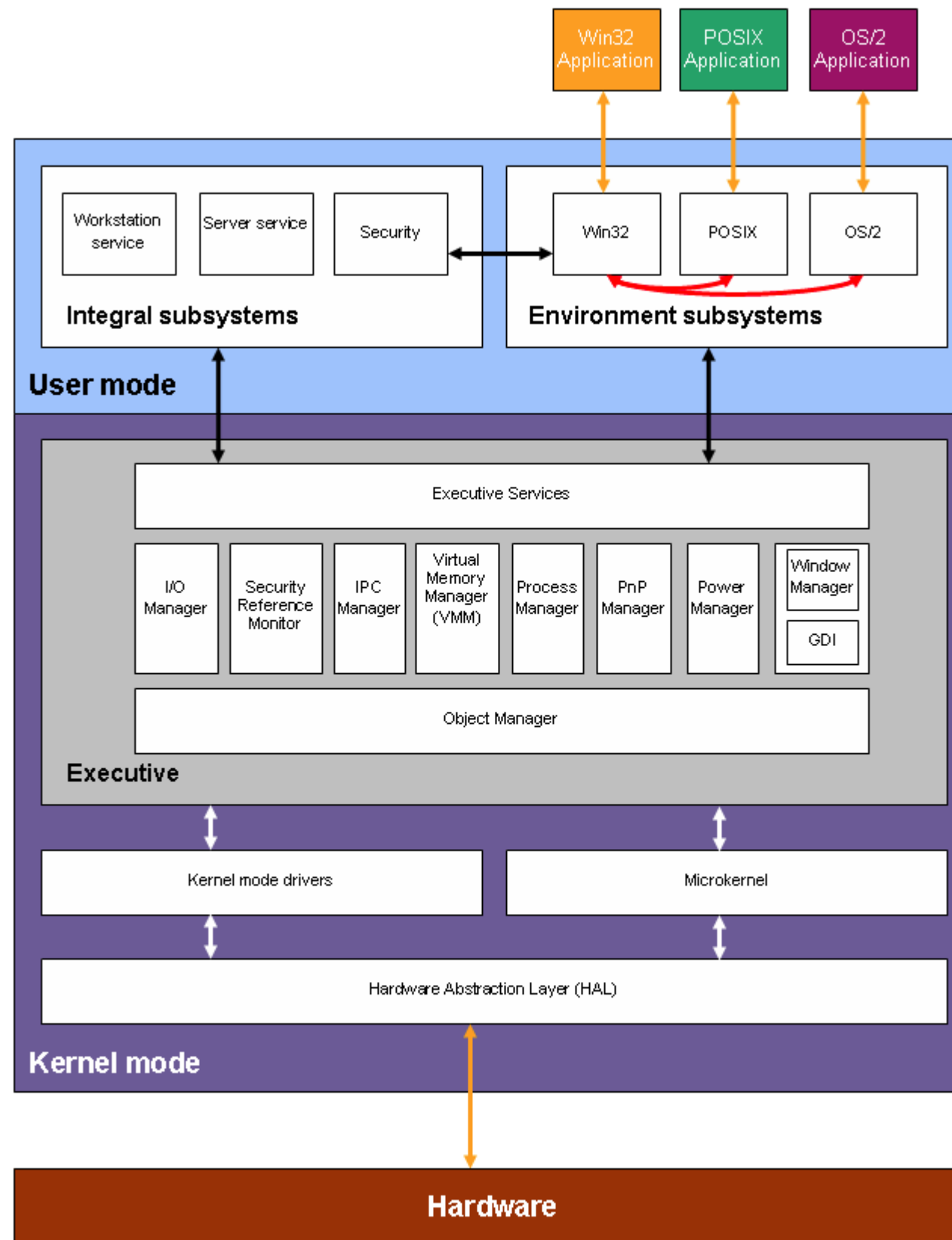
Noyau (Kernel) : Emplacement isolé contenant le cœur du système d'exploitation

Matériel (Hardware) : Couche d'abstraction matérielle





Les relations entre utilisateur, applications, système d'exploitation et matériel

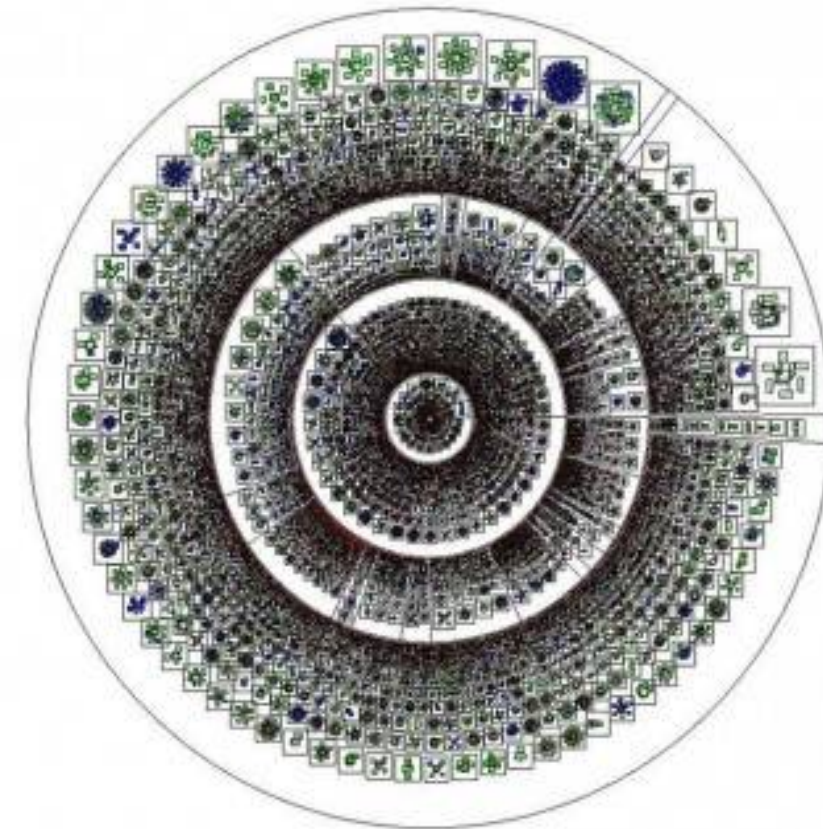


Questions ?

Le noyau ⁽¹⁾

Les fonctions de base du kernel:

- Gestion des ressources matériels (Processeurs, Mémoires, Périphériques)
- Gestion des divers logiciels (tâches) d'une machine (lancement des programmes, ordonnancement...)
- Communication entre matériels et logiciels

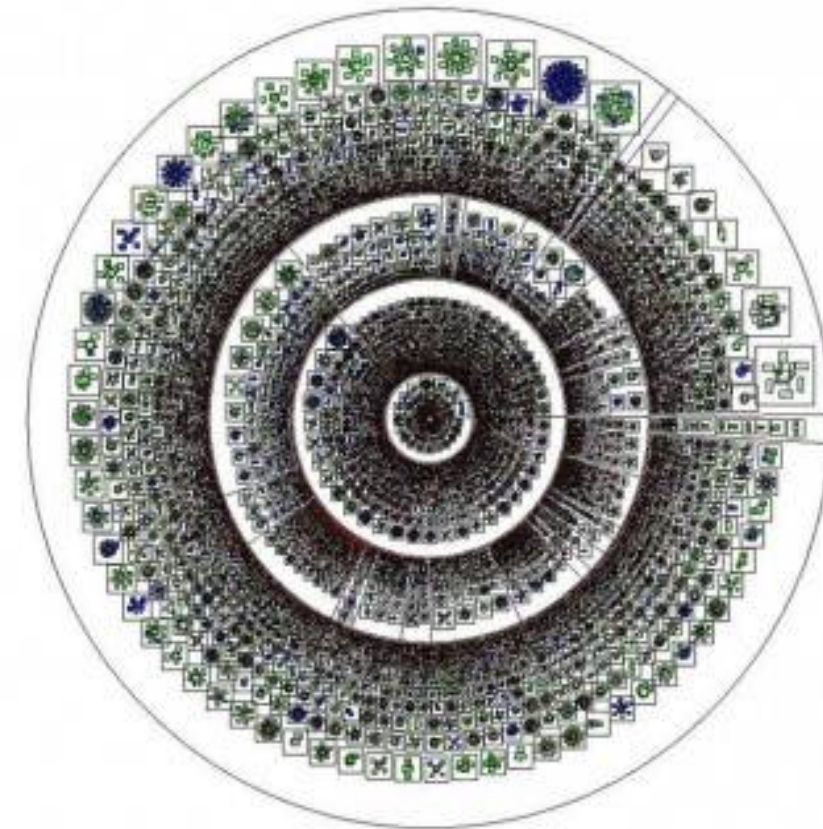


Le noyau garanti des performances élevées et la stabilité du système !!!

Le noyau (2)

Les fonctions avancées du kernel:

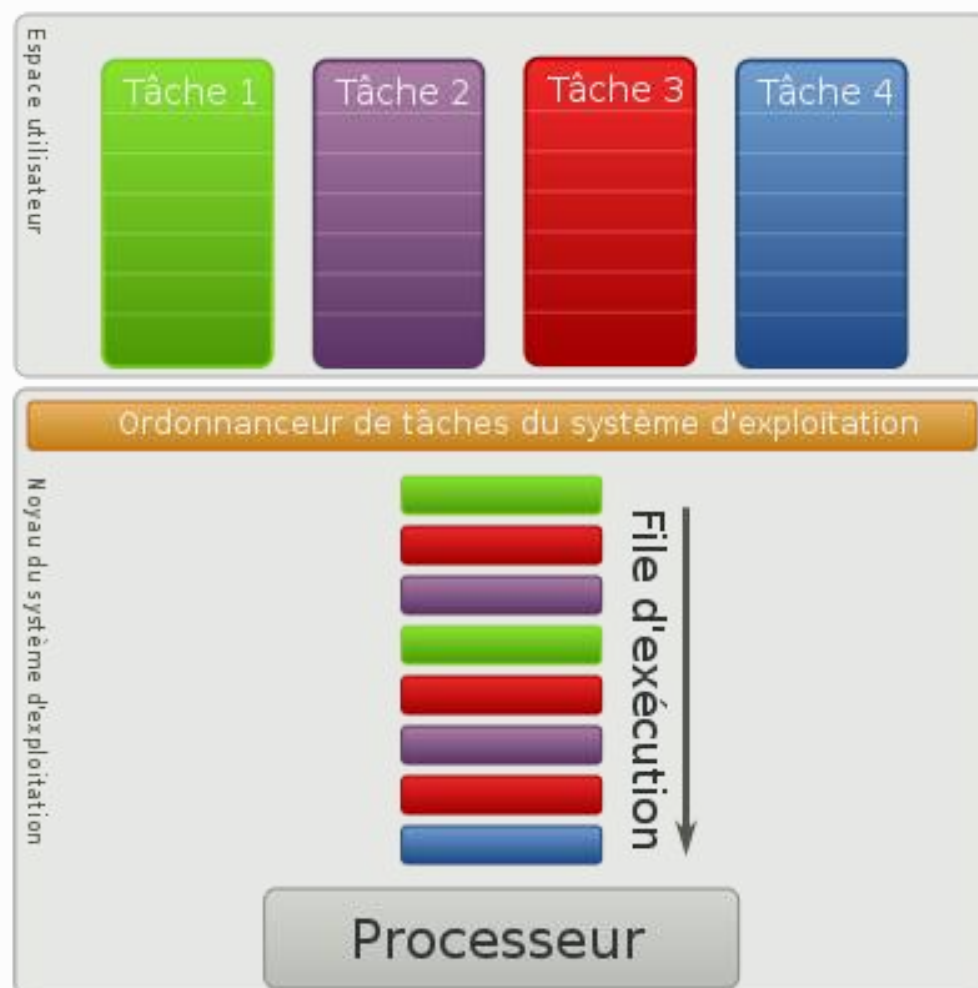
- Gestion des systèmes de fichiers
- Plusieurs ordonnanceurs spécialisés (batch, temps réel, entrées/sorties, etc.)
- Notions de processus étendues telles que les processus légers
- Supports réseaux (TCP/IP, PPP, pare-feu, etc.)
- Services réseau (NFS, Samba, etc.).



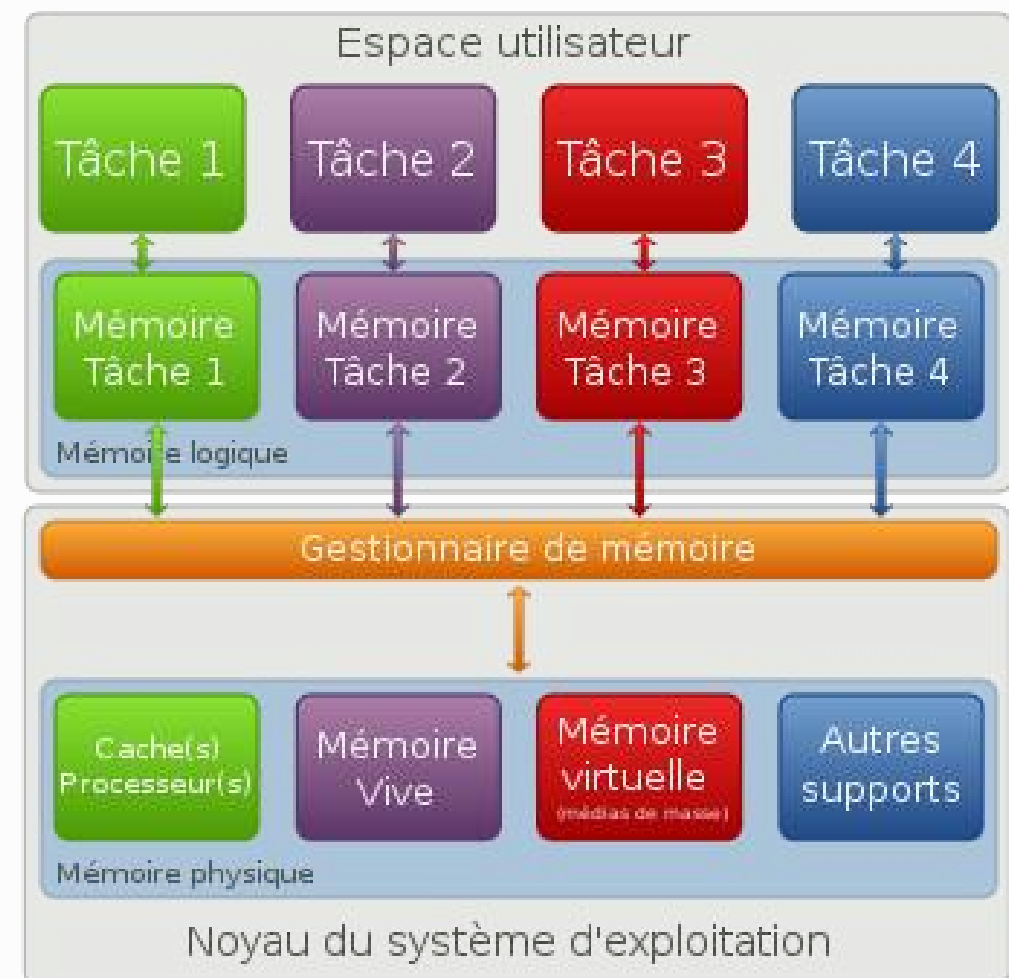
Le noyau garanti des performances élevées et la stabilité du système !!!

Le noyau (3)

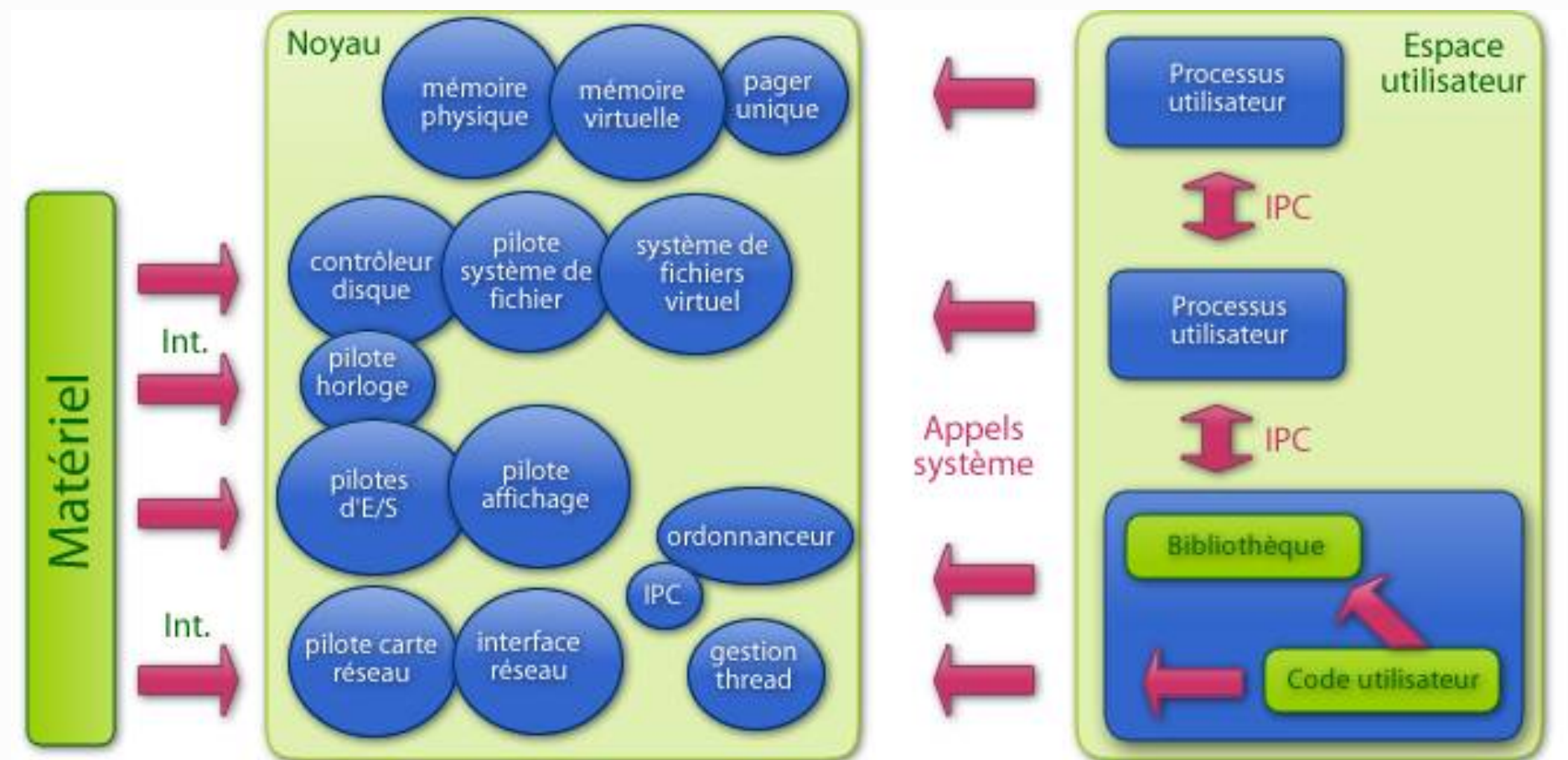
L'ordonnanceur



La gestion mémoire



Le noyau monolithique

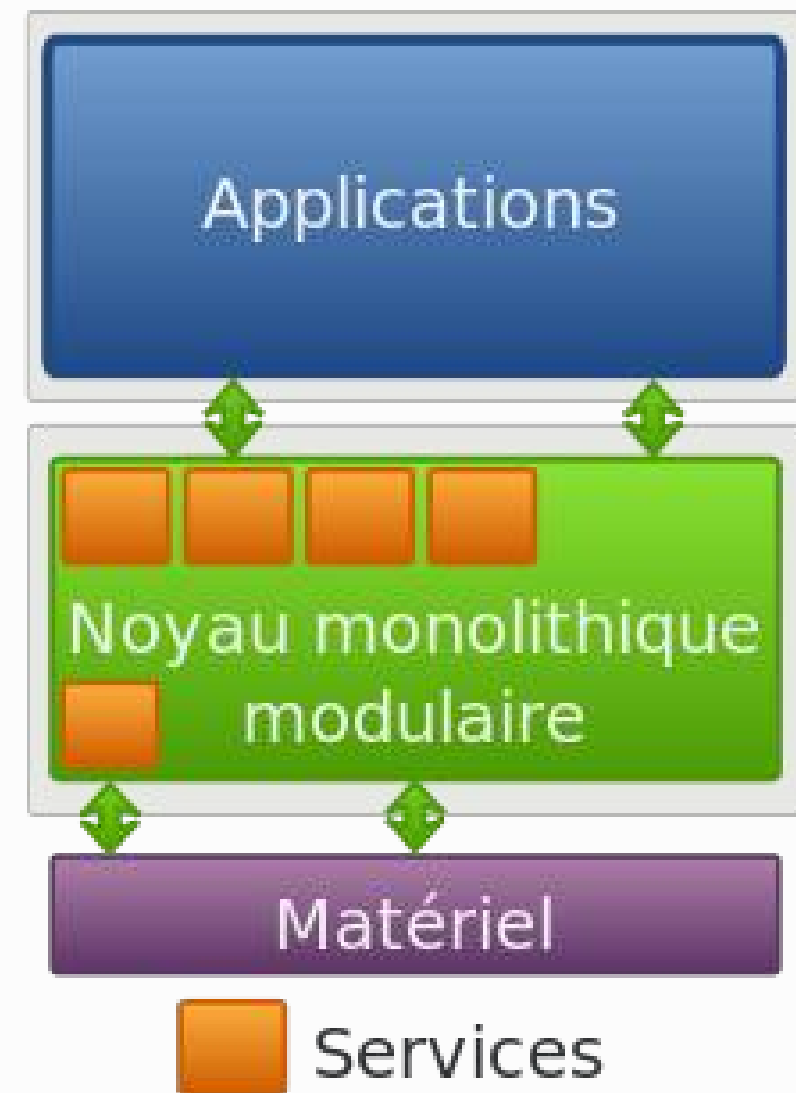


MS-DOS, BSD, HP-UX, AIX, Windows 98, LINUX

Le noyau monolithique modulaire

Les fonctions de bases sont regroupés dans un bloc unique

Les autres fonctions (ex : pilotes) sont découpées en modules séparables



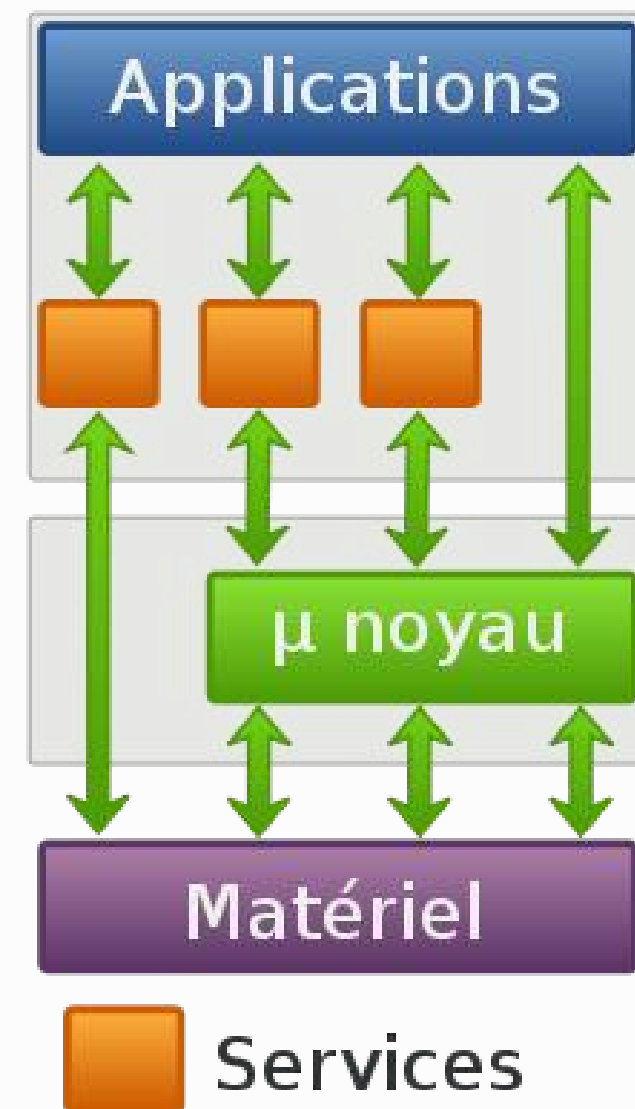
Le micro noyau

Le micro noyau contient les fonctions fondamentales.

Les autres fonctions sont découpées en services et se trouvent dans l'espace utilisateur.

L'ensemble forme un micro noyau « enrichi »

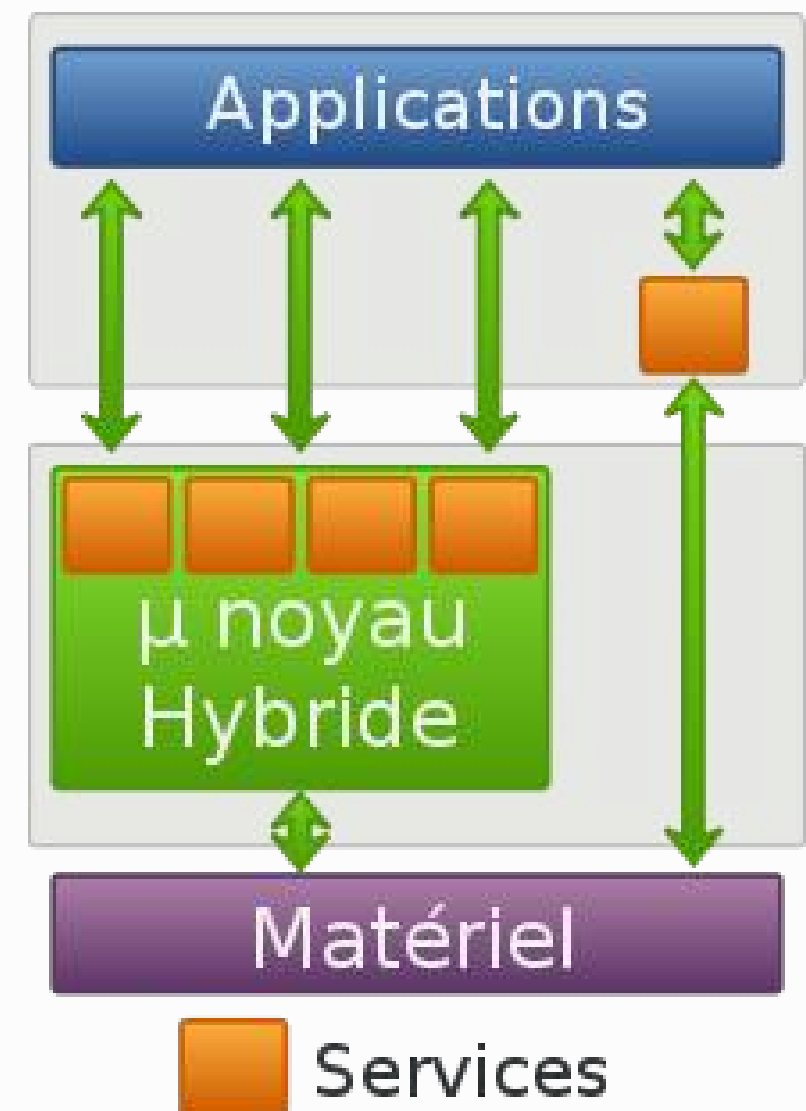
Ex : Mach, Minix, QNX, L4



Le noyau hybride

Le noyau hybride regroupe les concepts du noyau monolithique et du micro noyau.

Les fonctions qui font énormément d'appels système sont réintégrées dans l'espace noyau.

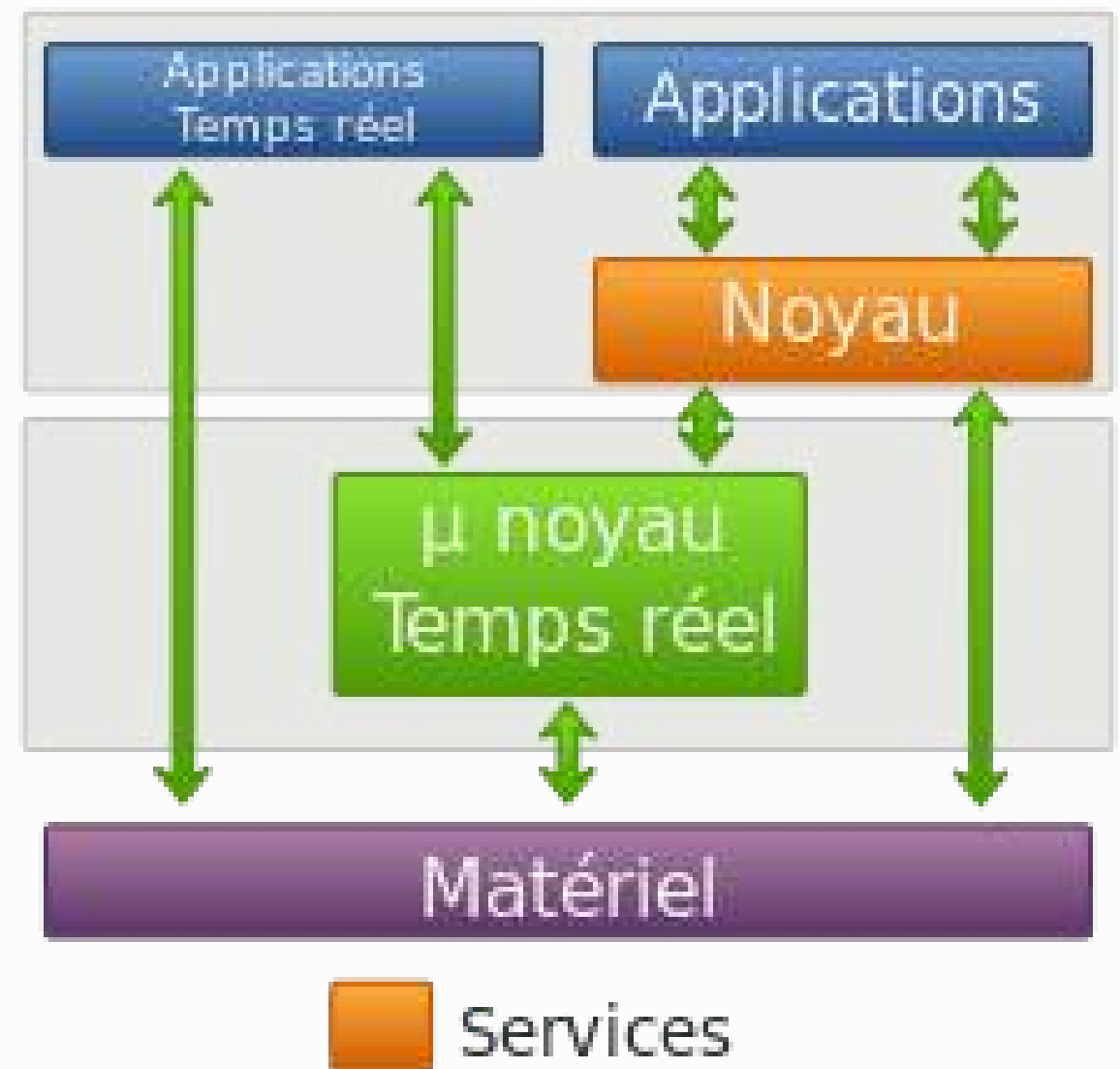


MAC OSX (XNU), Microsoft Windows NT, 2000, XP, Vista, and 7

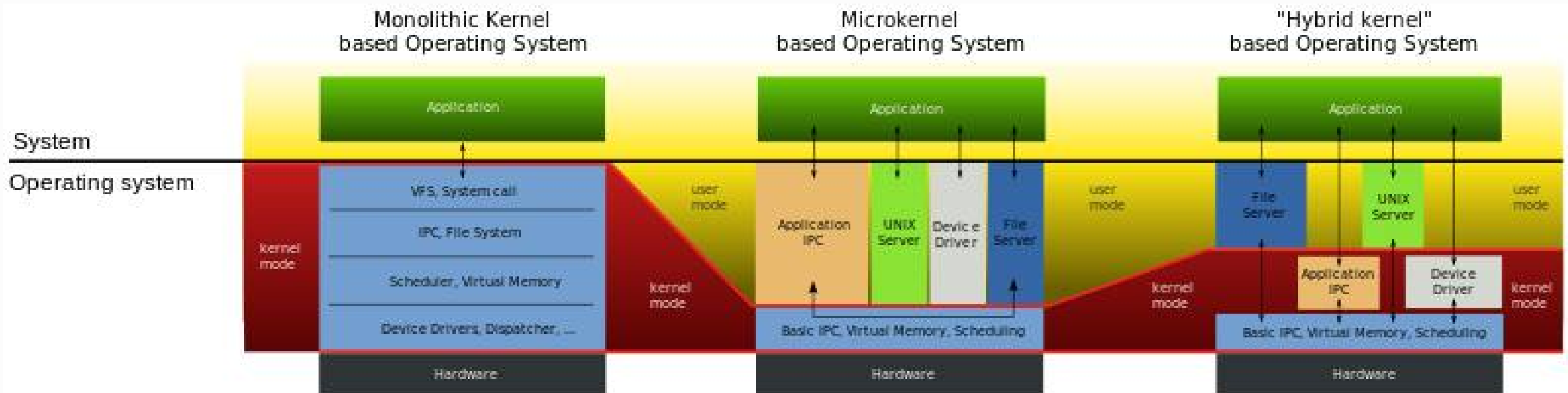
Le noyau temps réel

Micro noyau qui a pour fonction de garantir les temps d'exécutions des tâches.

Très utilisé dans le monde de l'électronique embarquée.



Comparatif noyaux



Questions ?

Historique Mobile

Smartphone : Téléphone intelligent

Historique Smartphone :

Simon (IBM- 1992) > BlackBerry (RIM-1999) > Windows CE (1996) > Windows Mobile (HTC - 2003) > iPhone (Apple - 2007) > Android (HTC - ~~2007~~ -2008) > Firefox OS (zte - 2014)

Historique Tablette :

Linus Write-Top (1987) > iPad (2010) > Samsung Galaxy Tab (2010) > iPad 2 (2011) > PlayBook (2011) > iPad 3 (2012) > Nexus 7/10 (2012)

Historique Mobile

Watch : Montre connectée

Historique Watch :

Tizen (Samsung - 2011) > Android Wear (LG - 2014) > Apple Watch (Apple - 2014)

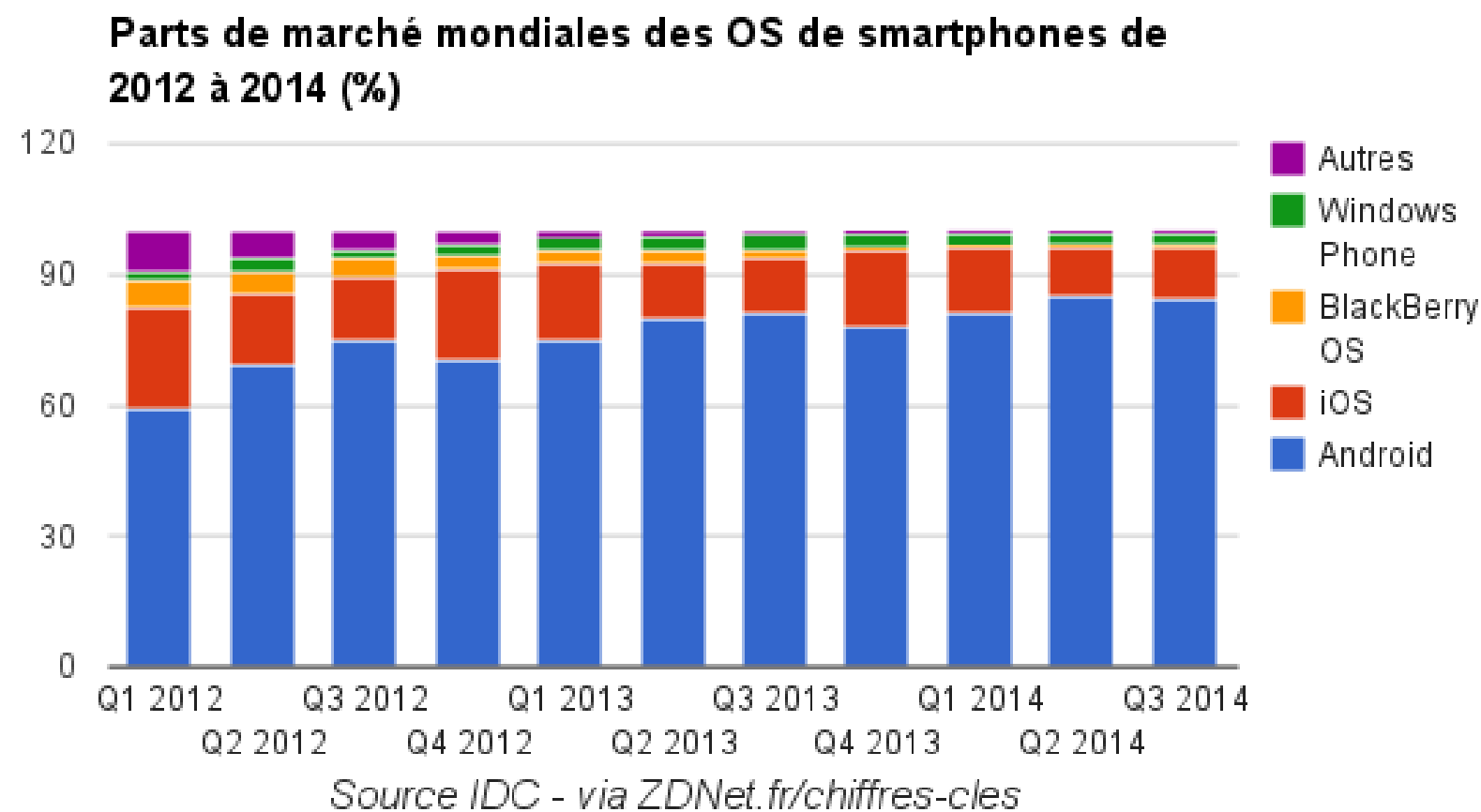


Les OS Mobiles

OS	Éditeur	Part de marché
Android	Google	84,4%
iOS	Apple	11,7%
Windows Phone	Microsoft	2,9%
RIM	Blackberry	0,5%
Symbian	Symbian LTD	Moins de 0,5 %
Bada	Samsung	Moins de 0,5 %
Tizen	Samsung	N.C.
Ubuntu Touch	Canonical	N.C.

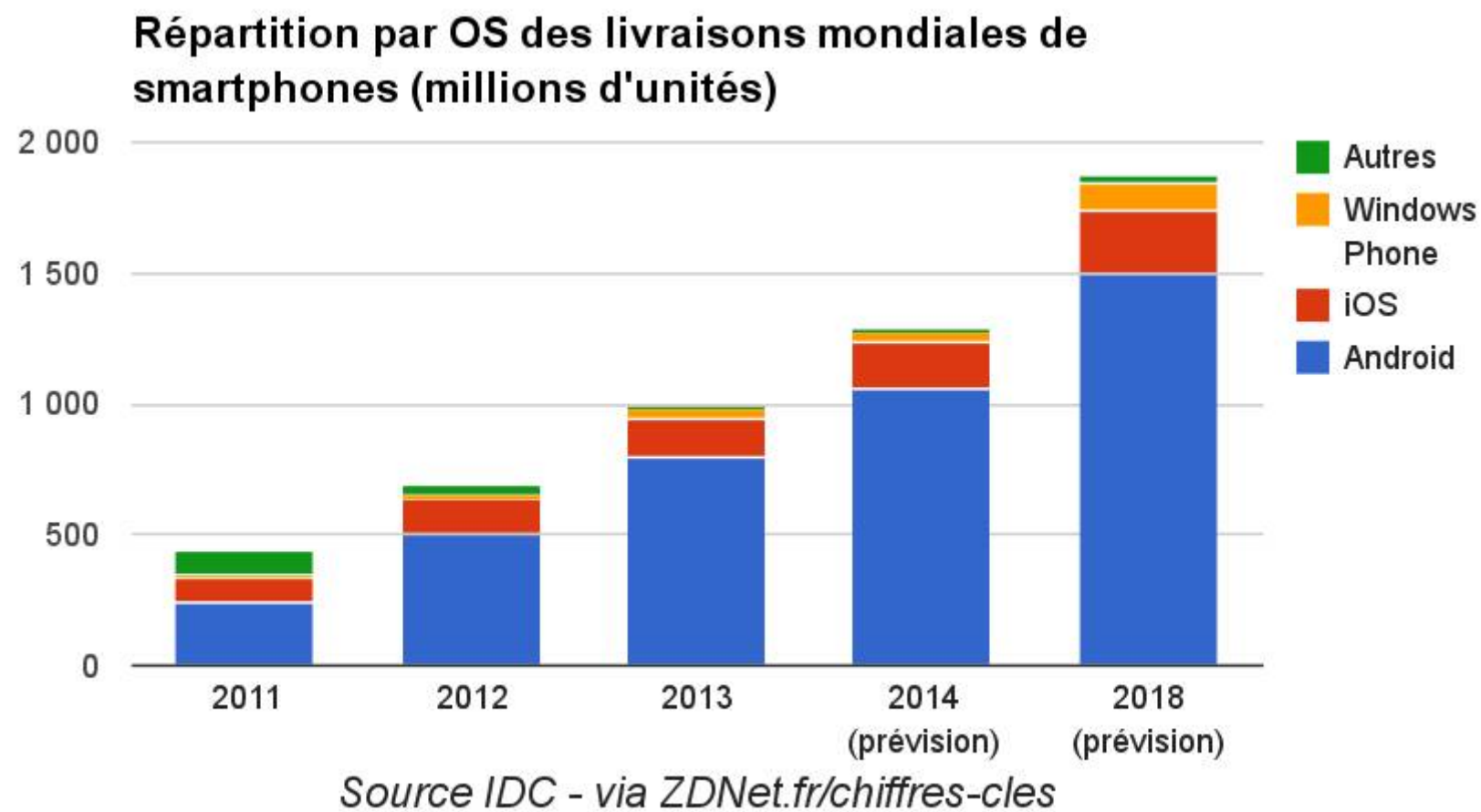
Répartition OS

Forte domination d'Android au détriment d'iOS :



Répartition OS

Prévision 2014 à 2018 :



Les éditeurs

Les principaux éditeurs :

- Google (Android)
- Apple (iOS)

Worldwide Smartphone Sales to End Users by Operating System in 2Q13 (Thousands of Units)

Operating System	2Q13 Units	2Q13 Market Share (%)	2Q12 Units	2Q12 Market Share (%)
Android	177,898.2	79.0	98,664.0	64.2
iOS	31,899.7	14.2	28,935.0	18.8
Microsoft	7,407.6	3.3	4,039.1	2.6
BlackBerry	6,180.0	2.7	7,991.2	5.2
Bada	838.2	0.4	4,208.8	2.7
Symbian	630.8	0.3	9,071.5	5.9
Others	471.7	0.2	863.3	0.6
Total	225,326.2	100.0	153,772.9	100.0

Source: Gartner (August 2013)

Autres éditeurs :

Microsoft - Samsung - Nokia

Les constructeurs

Les principaux constructeurs :

- Samsung
- Apple
- Huawei

Worldwide Smartphone Sales to End Users by Vendor in 4Q13 (Thousands of Units)

Company	4Q13 Units	4Q13 Market Share (%)	4Q12 Units	4Q12 Market Share (%)
Samsung	83,317.2	29.5	64,496.3	31.1
Apple	50,224.4	17.8	43,457.4	20.9
Huawei	16,057.1	5.7	8,666.4	4.2
Lenovo	12,892.2	4.6	7,904.2	3.8
LG Electronics	12,822.9	4.5	8,038.8	3.9
Others	106,937.9	37.9	75,099.3	36.2
Total	282,251.7	100.0	207,662.4	100.0

Source: Gartner (February 2014)

Autres constructeurs :

Lenovo - HTC - Sony - Motorola - Nokia - BlackBerry - ZTE - LG - Wiko - Archos

Questions ?

Les Processeurs en 1980

Intel :

- 1982 : 80286
- 1980 : 8051

Motorola :

- 1979 : 68000 (16/32 bits)

Université de Berkeley (David Patterson) :

- 1980 à 1984 : RISC I et II

Université de Stanford :

- 1981 à 1984 : MIPS

Architecture CISC

CISC : Complex Instruction Set Computer

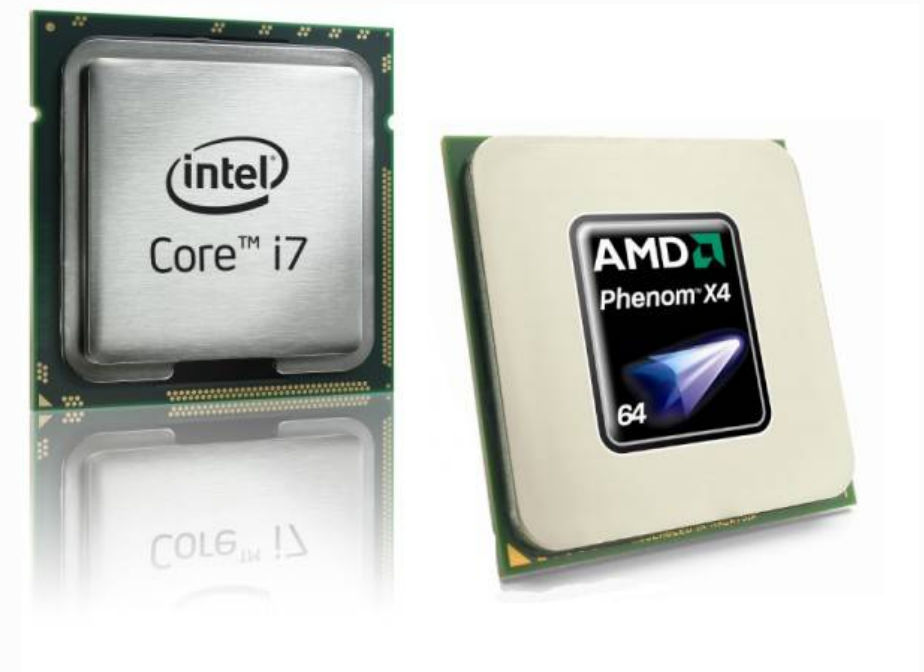
- Microcode ou microprogramme d'instruction
- Nombre d'instructions élevé avec multiples modes d'adressage

Avantages :

- Il faut moins de code de haut niveau pour une fonction. Besoin de moins de jeux d'instructions.
- Instructions très complexes et très rapides

Inconvénients :

- Processeur volumineux. Surface de silicium importante. Forte consommation électrique
- Instructions complexes et peu utilisées par les compilateurs



Architecture RISC

RISC : Reduced Instruction Set Computer

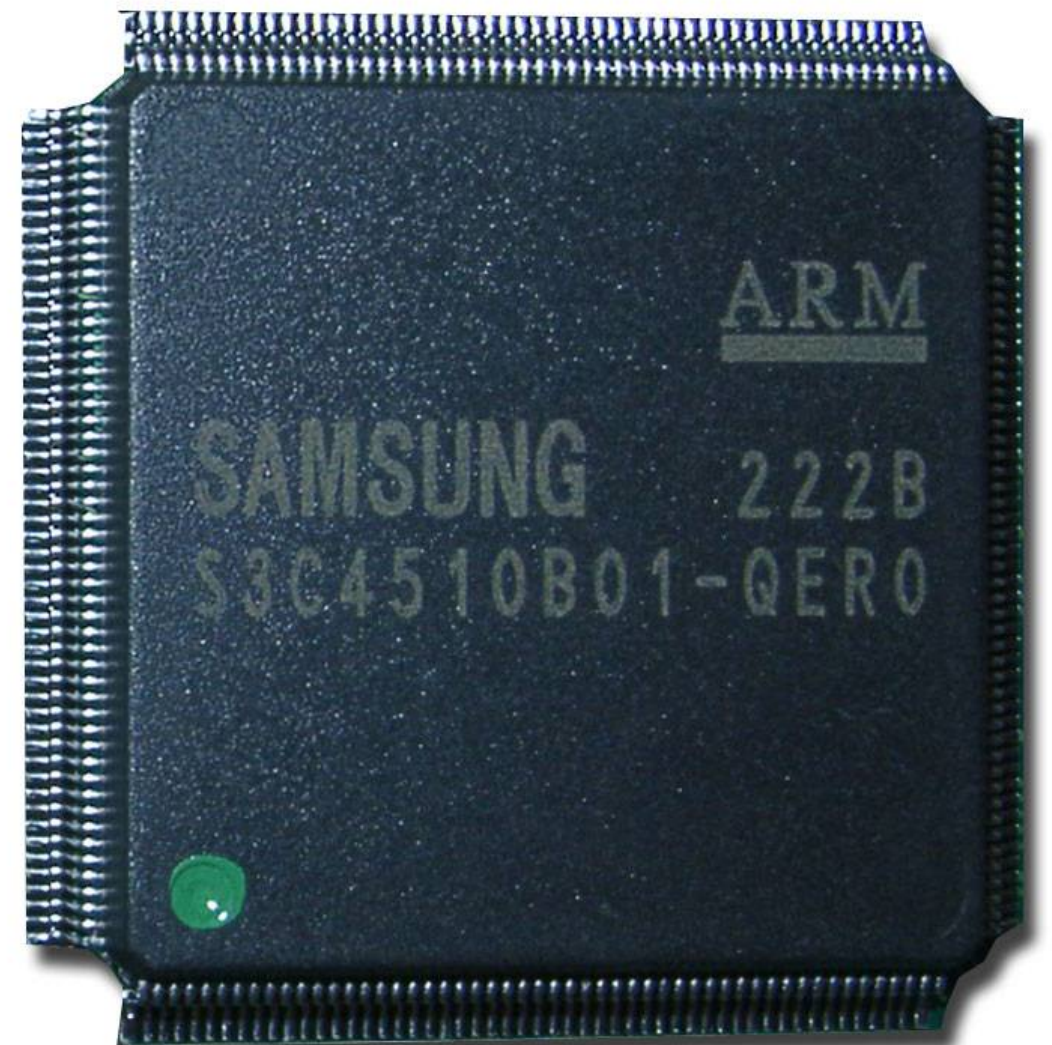
- Instructions de taille fixe
- Limitation des modes d'adressage

Avantages :

- Architecture moins complexe
- Instructions orthogonales
- Interruptions plus rapides
- Compilateur plus simple et efficace

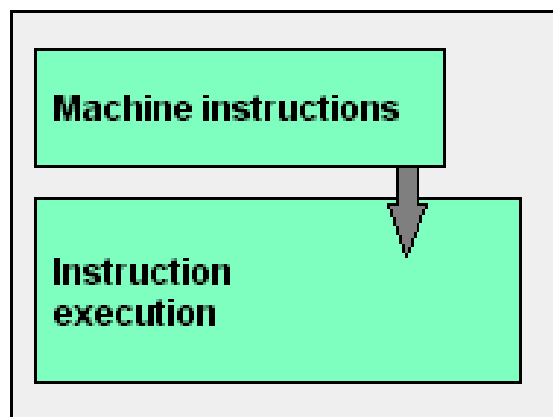
Inconvénients :

- Taille du programme plus importante
- Traitements avec accès multiples à la mémoire pénalisés

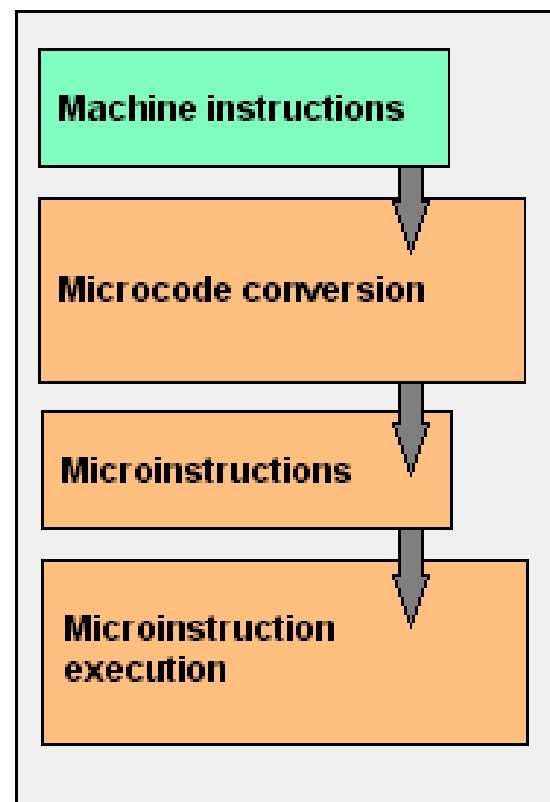


CISC vs RISC

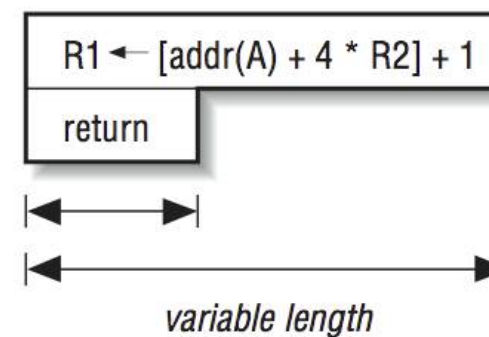
RISC



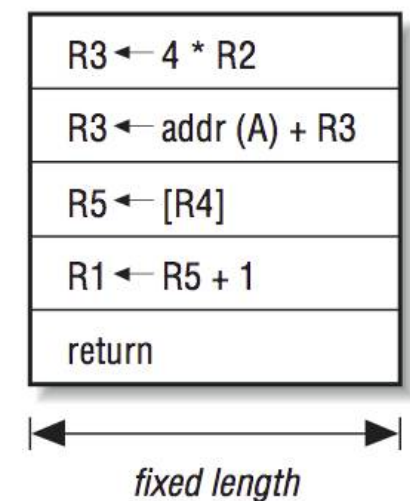
CISC



CISC



RISC



Historique ARM

Développée par l'entreprise anglaise Acorn Computer.

Équipe de développement dirigée par Roger Wilson et Steve Furber.

Inspiré des travaux des universités Berkeley et Stanford.

1985 : 1^{er} prototype ARM (collaboration avec le fondeur VLSI)

1986 : mise en production de l'ARM 2 (30 000 transistors - 4 MIPS@8Mhz)

1990 : création de la société ARM : Advanced Risc Machine limited

1992 : Apple travail en collaboration avec Acorn. ARM 610 (17MIPS@20MHz) pour PDA Newton.



Architecture	Famille(s)
ARMv1	ARM1
ARMv2	ARM2, ARM3
ARMv3	ARM6, ARM7
ARMv4	StrongARM, ARM7TDMI, ARM8, ARM9TDMI
ARMv5	ARM7EJ, ARM9E, ARM10E, XScale, FA626TE, Feroceon, PJ1/Mohawk
ARMv6	ARM11
ARMv6-M	ARM Cortex-M0, ARM Cortex-M0+, ARM Cortex-M1 (voir ARM Cortex-M)
ARMv7-A	ARM Cortex-A (ARM Cortex-A8, ARM Cortex-A9 MPCore, ARM Cortex-A5 MPCore, ARM Cortex-A7 MPCore, ARM Cortex-A12 MPCore, ARM Cortex-A15 MPCore, Scorpion, Krait, PJ4/Sheeva, Swift
ARMv7-M	ARM Cortex-M (ARM Cortex-M3)
ARMv7-R	ARM Cortex-R (ARM Cortex-R4, ARM Cortex-R5, ARM Cortex-R7)
ARMv8	ARM Cortex-A50 (ARM Cortex-A53, ARM Cortex-A57), X-Gene, Denver, Cyclone

ARM Business Model

Société « fabless » :

ARM Ltd. ne fabrique et ne vend pas ses processeurs.

Vends de l'IP (Intellectual Property) :

Elle vend les licences de ses processeurs :

- Implementation Licence :

 - Hard Core pour les fondeurs

 - Soft Core pour les spécifications

- Architecture Licence :

 - Utilisation du jeu d'instruction

Fabricants : http://fr.wikipedia.org/wiki/Architecture_ARM#Fabricants_de_processeurs_ARM

Jeu d'instruction

Architecture :

- Architecture inspirée des principes de conception RISC.
- 16 registres généraux de 32 bits.
- Instructions codées sur 32 bits jusqu'à ARMv7. 64 bits à partir d'ARMv8

Optimisation :

- Thumb : jeu d'instruction 16 bits pour un gain de mémoire
- VFP (vector floating point) : jeu d'instruction pour les calculs des formats virgule flottante
- Jazelle DBX (Direct ByteCode execution) : exécution du bytecode java directement dans l'architecture ARM

ARM et mobilité

De nombreux systèmes d'exploitation sont compatibles avec ARM :

- Symbian S60 avec les Nokia N97 ou Samsung Player HD ;
- iOS avec l'iPhone et l'iPad ;
- Linux, avec la plupart des distributions ou avec Android ;
- BlackBerry OS avec les BlackBerry
- Windows CE, Windows Phone 7 et Windows RT2, une version de Windows 8.
- le système PlayStation Vita
- Risc OS

ARM et mobilité

Pourquoi ARM ?

- Architecture simple
 - Faible consommation électrique.
- Gain en autonomie sur batterie !!!**
- Faible coût de production
 - De nombreux fabricants (concurrence)
 - Chauffe peu
 - Architecture en fonction des besoins



Questions ?

Creative Commons 2013 TACTfactory.

Change log

- Yoan PINTAS 2014 : Initial document