CONSERVATOIRE
NATIONAL
DES ARTS
ET METIERS

Architecture des Serveurs : Évolution des technologies

Novembre 2001

René J. Chevance

Contenu

- Revue des besoins
- Transactionnel et décisionnel - Web
- Introduction aux options d'architecture
- Évolution des technologies - Matériel
  - Semi-conducteurs et microprocesseurs
  - Notions de hiérarchie de mémoire
  - Notions de parallélisme
  - Compatibilité binaire - Java
  - Entrées-sorties
    - PCI, SCSI et Fibre Channel
    - Nouvelles architectures d 'entrées-sorties (InfiniBand)
  - Stockage des données
    - SAN et NAS
    - RAID
  - Communications
- Évolution des technologies - Logiciel
  - Mémoire virtuelle
  - Architecture 64 bits
  - Systèmes d'exploitation
  - Client/serveur
  - Middleware
- Quelques considérations économiques
  - Évolution de la structure de l'industrie
- Perspectives concernant la technologie

Revue des besoins

- **Disponibilité de packages d'application et d'outils**
- **Intégrité des données**
- **Disponibilité du serveur**
- **Sécurité**
- **Performance**
- **Scalabilité (capacité de traitement, de stockage, de communication)**
- **Prix (coût de possession et d'opération - TCO Total Cost of Ownership)**
- **Support du Client/Serveur**
- **Maturité de l'architecture**
- **Pérennité des investissements**

Transactionnel et décisionnel

Caractérisation du transactionnel (OLTP - On Line Transaction Processing) et du décisionnel (DSS - Decison Support Systems)

Systèmes Transactionnels

- **Partage:**
  - **en Lecture et Écriture**
  - **par l'ensemble des utilisateurs**
  - **Propriétés ACID**
- **Flux de requêtes irrégulier**
- **Travail répétitif**
  - **Répertoire de fonctions pré-défini typiquement O(100) fonctions**
- **Fonctions simples**
  - **Fonctions peu complexes (typiquement de 10^5 à 10^7 instructions et 10 E/S)**
- **Possibilité de traitement de type batch (avec respect des propriétés ACID)**
- **Grand nombre de terminaux (1000-10000)**
- **Clients intelligents (stations, PC, autres systèmes, terminaux)**

Systèmes d'aide à la décision

- **Partage base de données**
  - **en lecture (essentiellement)**
  - **base spécifique (^{1} base de production)**
  - **bases spécialisées (Datamarts)**
- **Flux de requêtes irrégulier**
- **Travail non répétitif**
  - **Pas de répertoire de fonctions pré-défini**
- **Fonctions complexes**
  - **Requêtes souvent complexes mettant en jeu de grands volumes de données**
- **Batch pour requêtes longues**
- **Petit nombre de stations de travail**
- **Clients intelligents (stations, PC)**

Transactionnel et décisionnel (2)

Systèmes Transactionnels

- **Haute disponibilité requise**
- **Recouvrement effectué par le système**
  - **Fondé sur les propriétés ACID**
- **Taille des bases de données**
  - **Proportionnelle à l'activité de la Société**
- **Peu de données "touchées" par une transaction**
- **Équilibrage de charge automatique**
  - **Recherche de la performance au moyen du parallélisme inter-requête**
- **Performance : haut débit et temps de réponse garanti**
- **Scalabilité : exigence typique**

Systèmes d'aide à la décision

- **La haute disponibilité n'est pas une exigence typique**
- **Le temps de génération/régénération de la base est un paramètre important**
- **Taille des bases de données**
  - **Proportionnelle à l'histoire de la Société**
- **Beaucoup de données "touchées" par une requête**
- **Pas d'équilibrage de charge automatique**
  - **Recherche de la performance au moyen du parallélisme intra-requête**
- **Performance : recherche de temps de réponse courts**
- **Scalabilité : exigence typique**

Caractéristiques du Web

- **Les utilisations des sites Web se rangent dans deux grandes catégories non-exclusives :**
  - **Serveurs de documents avec des procédures de recherche et de navigation particulières (moteurs de recherche, liens)**
  - **Serveur transactionnel pour l 'enregistrement de commandes (e-commerce)**
  - ***Note : Une opération commerciale est précédée d'une ou plusieurs phases de recherche d'information***
- **Différentes études ont permis de mettre en évidence les caractéristiques des sites :**
  - **Martin F Arlitt, Carrey L Williamson " Web Server Workload Characterization : The Search for Invariants " Department of Computer Science University of Saskatchewan March 1996**
  - **James E Pitkow « Summary of WWW Characterizations » Xerox Palo Alto Research Center 1998**
  - **Daniel A Menascé et al. « In Search of Invariants for E-Business Workloads » Proc. Second ACM Conference on Electronic Commerce, Minneapolis MN, October 17-20, 2000**

Caractéristiques du Web(2)

Invariants de Arlitt/Williamson 1996

Dérivés de l'observation de 6 sites Web

Invariant	Nom	Description
1	Taux de succès	Le taux de succès pour les accès sur le serveur est ≈ 88%
2	Types de fichiers	90 à 100% des requêtes pour des fichiers HTML et des images
3	Taille moyenne des transferts	Taille moyenne des transferts <= 21 Ko
4	Requêtes distinctes	Moins de 3% des requêtes sont relatives à des fichiers distincts
5	Références uniques	Environ un tiers des fichiers et des octets accédés ne le sont qu'une seule fois
6	Distribution des tailles de fichiers	La distribution des tailles des fichiers accédés est une loi de Pareto avec 0,40<k<0,63
7	Concentration des références	10% des fichiers accédés représentent 90% des requêtes au serveur et 90% des octets transférés
8	Intervalles entre références	Les intervalles entre références (à des documents distincts) suivent une distribution exponentielle et sont indépendants
9	Requêtes provenant de sites distants	≥70% des accès au serveur proviennent de sites distants et ≥60% des octets transférés le sont vers des sites distants
10	Distribution géographique	Les serveurs Web sont accédés par des milliers de domaines mais 10% des domaines comptent pour ≥75% des utilisations

Caractéristiques du Web(3)

Synthèse de Pitkow 1998

Synthèse des mesures publiées

Numéro	Caractéristique	Description
1	Popularité des fichiers accédés	Distribution de Zipf avec a = 1
2	Taux de ré-occurrence des références	Environ 50% des fichiers sont requis plus d'une fois par le même client. La probabilité de re-référence dans les t minutes étant proportionnelle à log(t)
3	Taille des fichiers	Loi de Pareto avec une taille moyenne de 4 à 6 Ko (médiane 2 Ko) pour HTML, taille moyenne des images de 14 Ko
4	Trafic	Les images de petite taille représentent la majorité du trafic et la taille des documents est inversement proportionnelle à la fréquence des requêtes
5	Trafic HTTP auto-similaire	Trafic avec pics, auto-similaire entre les domaines allant de la seconde à la minute
6	Caractère périodique du trafic HTTP	Le caractère périodique du trafic peut être représenté par des séries portant sur des domaines allant de l'heure à la semaine
7	Popularité des sites	25% des serveurs représentent 85% du trafic
8	Durée de vie des documents	Environ 50 jours, les fichiers HTML étant modifiés et supprimés plus fréquemment que les autres types de fichiers
9	Taux d'occurrence de liens brisés durant le surf	De 5 à 8% de l'ensemble des fichiers requis
10	Taux d'occurrence des redirections	De 13 à 19% de l'ensemble des fichiers requis
11	Nombre de visites à une page	Distribution gaussienne inverse avec une moyenne de 3, une déviation standard de 9 et un mode de 1
12	Temps de lecture par page	Distribution avec une moyenne de 30, une médiane de 7 et une déviation standard de 100 secondes

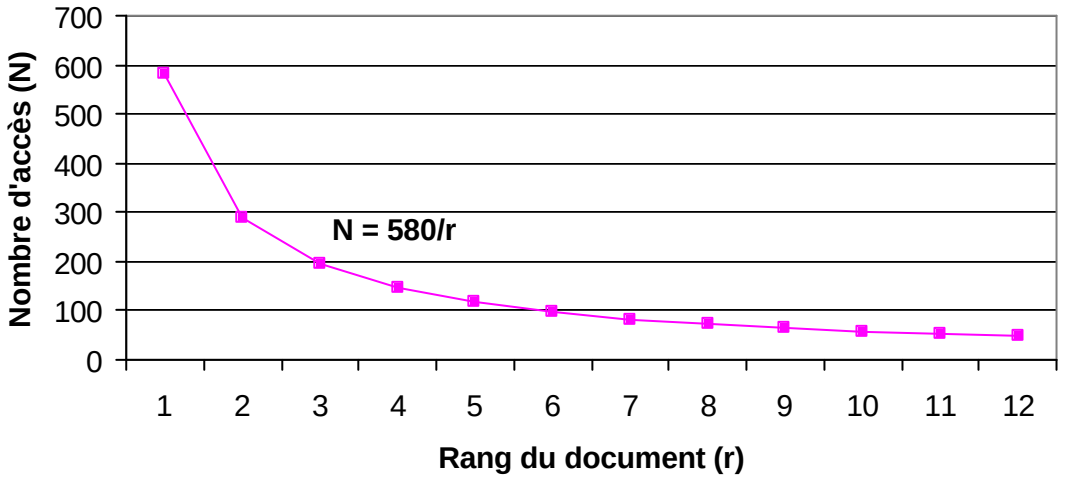
Caractéristiques du Web(4)

- **Invariants e-commerce de Menascé/Almeida 2000**
  - **Étude de deux sites de e-commerce :**
    - **Vente d'ouvrages (uniquement par e-commerce)**
    - **Vente aux enchères de noms de domaines Internet**
- **Résultats :**
  - **La plupart des sessions durent moins de 1 000 secondes**
  - **Plus de 70% des fonctions exécutées sont relatives à la sélection des produits**
  - **Le nombre d'accès à un document suit une distribution de Zipf en relation avec la popularité du document (représentée par son rang r). Nombre d'accès au document de rang r :**
    - **N = k / r (où k est une constante positive)**
  - **Il existe une très forte corrélation dans l'arrivée des requêtes :**
    - **longues séquences de variations à la hausse et à la baisse**
    - **caractère auto-similaire**
  - **Au moins 16% des requêtes sont engendrées par des agents (robots)**
  - **88% des sessions ont moins de 10 requêtes**
  - **La longueur des sessions, mesurée en nombre de requêtes, est une distribution à queue importante « heavy tailed » (si la valeur moyenne est faible, des valeurs très grandes sont possibles bien que peu probables), tout particulièrement pour les sites recevant des requêtes engendrées par des agents**

Rappel : Loi de Zipf

- **Loi de Zipf**
  - **Le linguiste George Kingsley Zipf, professeur à l'Université de Harvard (1902-1950), s'est intéressé à l'utilisation des mots de la langue anglaise. Il a dérivé une loi qui relie leur popularité et leur fréquence d'utilisation. Dans le cas d 'un site Web, si l 'on classe les documents en fonction de leur popularité désignée par *r* (i.e. leur rang en fonction du nombre de fois qu'ils sont requis), le nombre de références, désigné par *N*, au document est exprimé par la loi :**
    - **N = k/r où k >0**

Exemple de loi de Zipf
(Source Mesnacé/Almeida)

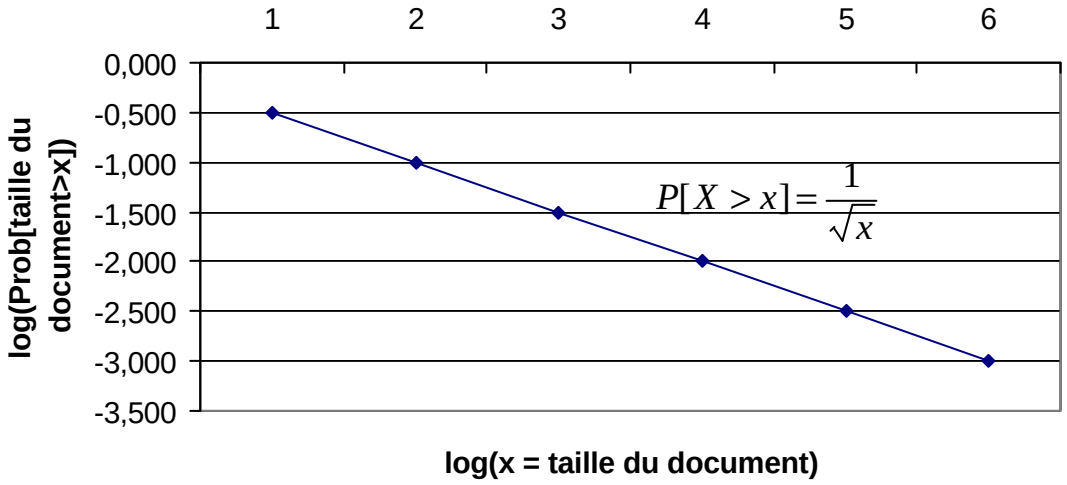


Rappel : Loi de Pareto

- **Loi de Pareto**
  - **L'économiste et sociologue Vilfredo Pareto, professeur à l'Université de Lausanne 1848-1923), s'est intéressé aux revenus des personnes. Il a dérivé une loi qui relie le fait qu'une personne ait des revenus supérieurs à un certain montant à la valeur de ce montant. Dans le cas d'un site Web, cette loi exprime la probabilité qu'un document ait une taille supérieure à une certaine valeur x sous la forme suivante :**

$$\Pr[X \geq x] \sim \left(\frac{m}{x}\right)^k \quad m > 0, k > 0, x \geq m$$

Exemple de loi de Pareto
(Source Mesnacé/Almeida)

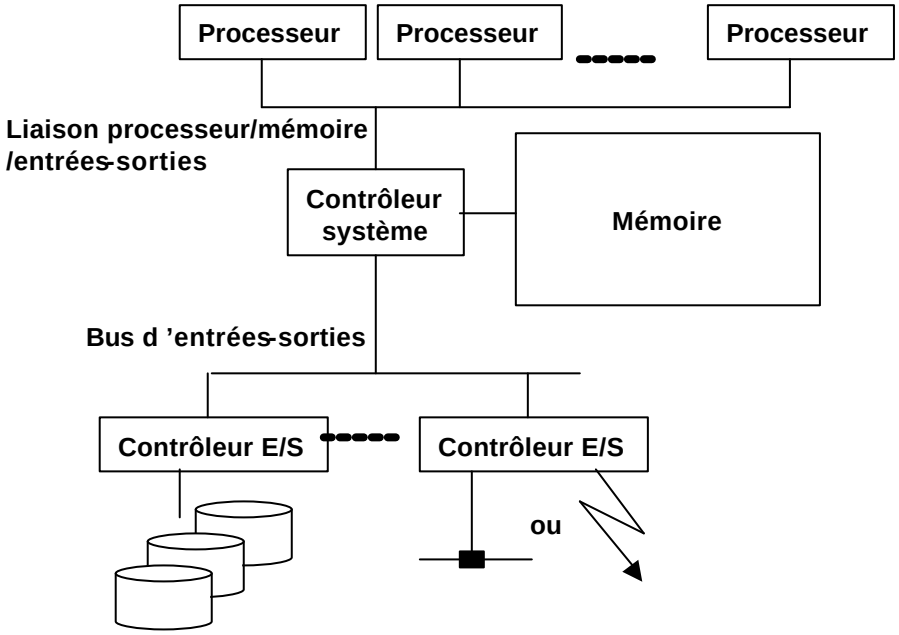


Introduction aux options d'architecture

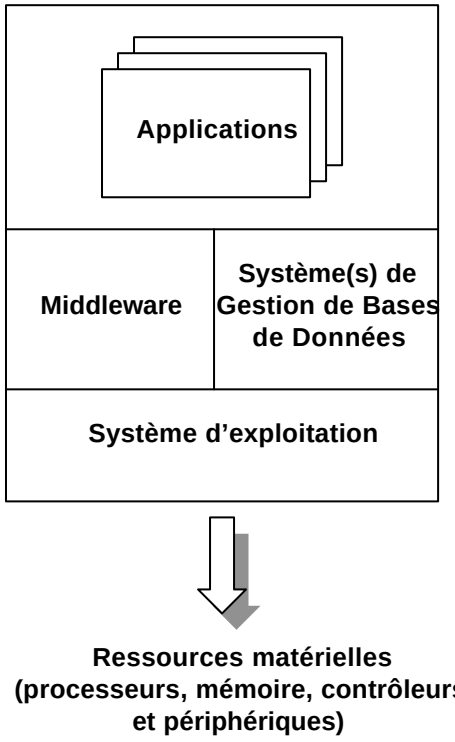
- **Recherche de la scalabilité au moyen des architectures multiprocesseurs**
- **Deux grandes familles d'architecture :**
  - **Couplage serré ou multiprocesseur symétrique (SMP Symmetric Multiprocessor) : les processeurs se partagent une mémoire cohérente ainsi que l'ensemble des ressources du système. Le système fonctionne sous le contrôle d'un seul système d'exploitation**
  - **Couplage lâche (Loosely Coupled). Le système est composé d 'un ensemble de systèmes indépendants appelés nœuds. Chacun des nœuds possède les ressources nécessaires à son exécution et fonctionne sous le contrôle de sa propre copie du système d 'exploitation. Les clusters et les machines massivement parallèles (MPP Massively Parallel Processing) sont des exemples de ce type de d'architecture**

Introduction aux options d'architecture(2)

Architecture à couplage serré (SMP)



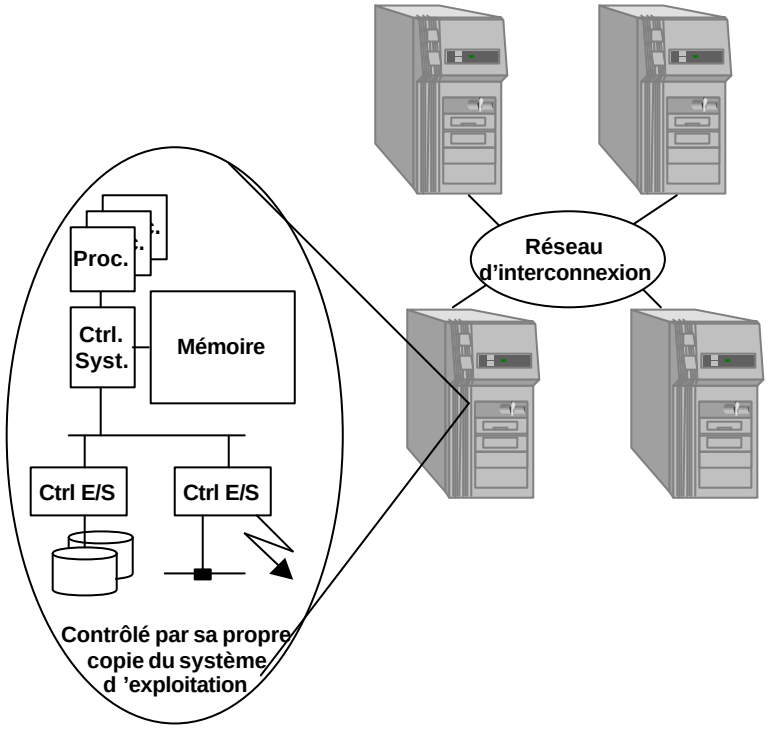
a) Vision matérielle de l'architecture multiprocesseur symétrique



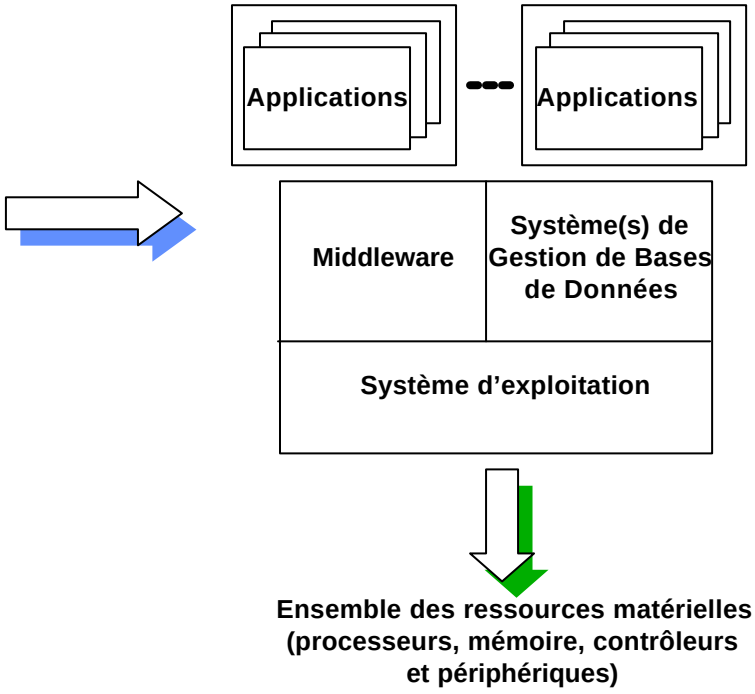
b) Vision logicielle de l'architecture multiprocesseur symétrique

Introduction aux options d'architecture(3)

Architecture à couplage lâche (Cluster/MPP)



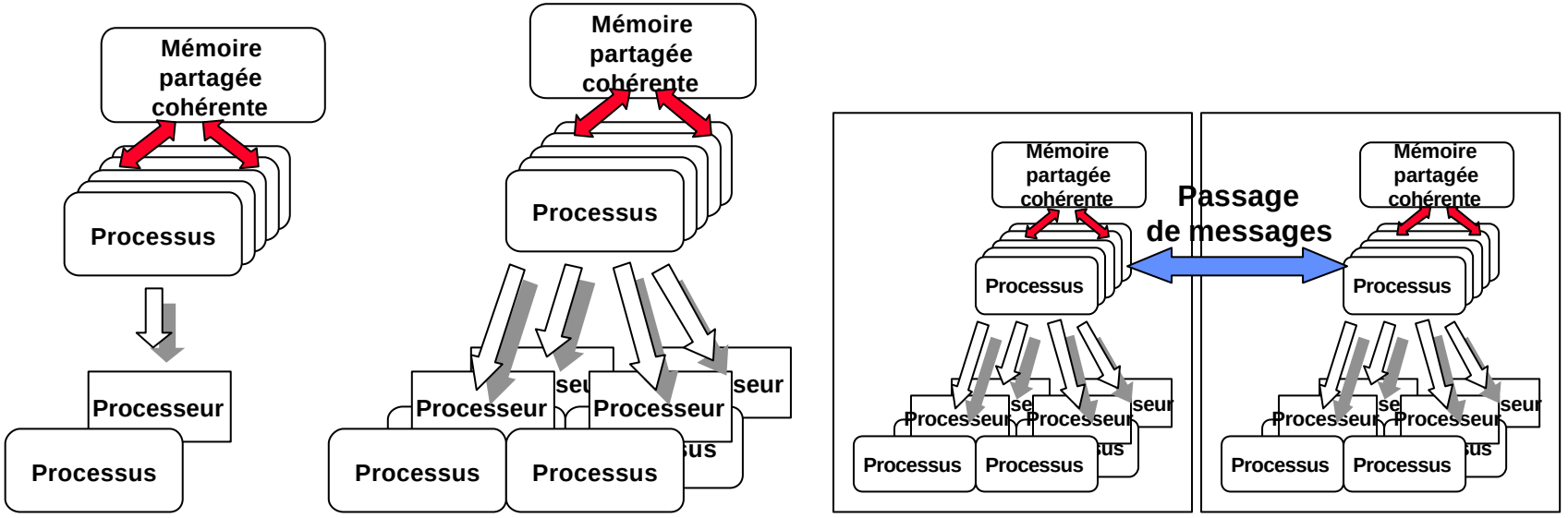
a) Vision technique de l'architecture à couplage faible



b) Vision système idéale de l'architecture à couplage faible

Introduction aux options d'architecture(4)

Modèles d 'exécution et de programmation



a) - Monoprocesseur

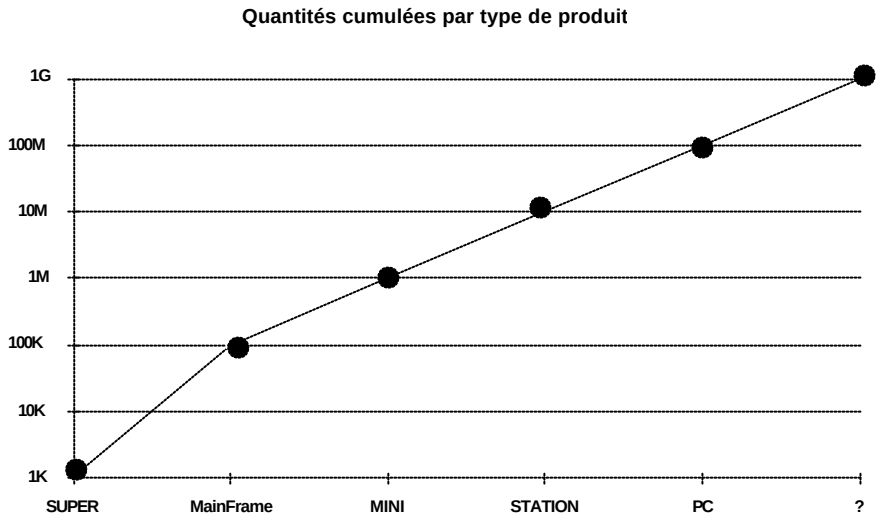
b) - Multiprocesseur à couplage serré

c) - Multiprocesseur à couplage lâche

Evolution des technologies

- **Éléments conducteurs de l'évolution**
- **Matériel:**
  - **Semi-conducteurs et microprocesseurs**
  - **Hiérarchie de mémoire**
  - **Parallélisme**
  - **Compatibilité binaire**
  - **Entrées-sorties**
    - **Sous-systèmes de stockage**
    - **Sous-systèmes de communication**
- **Logiciel**
  - **Mémoire virtuelle**
  - **Architecture 64 bits**
  - **Systèmes d'exploitation**
  - **Client/serveur**
  - **Internet/Intranet/Extranet**
- **Structure de l'industrie**

Éléments conducteurs de l'évolution



Ce graphique montre, de façon schématique, les quantités cumulées des différents types de matériels (les quantités indiquées ne sont que des approximations, on ne cherche ici qu'à identifier des tendances). A l'évidence, les éléments conducteurs (drivers) sont les objets de type station de travail et surtout PC. On a placé sur ce graphique un nouvel élément conducteur (noté ?) qui pourrait être constitué par des appareils d'accès à l'information (Information Appliances tels que Web-phone, télévision interactive, ...).

Les avancées technologiques (circuits intégrés, réseaux locaux, interfaces utilisateur conviviaux, logiciels de productivité personnelle,...) ont permis cette évolution (réduction des coûts et pénétration du marché). La taille des marchés concernés a dicté les investissements industriels. Pour illustrer ce point, on peut noter que l'industrie des disques magnétiques a été transformée par le monde des PC et que les grands systèmes de traitement de l'information qui utilisent les mêmes disques que les PC.

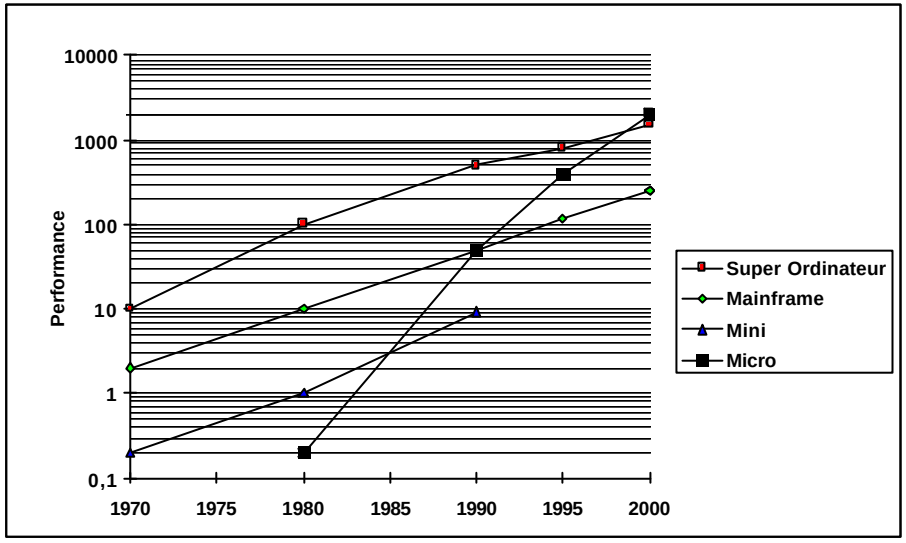
Cette influence concerne tout autant l'industrie du logiciel que celle du matériel: certains des "leaders" de l'industrie aujourd'hui étaient pratiquement inexistants il y 15 ou 20 ans.

Évolution des technologies
Matériel

Semi-conducteurs et microprocesseurs

- **Évolution de la performance des processeurs**
  - **La performance des microprocesseurs leur permet pratiquement d'adresser la quasi-totalité des applications**

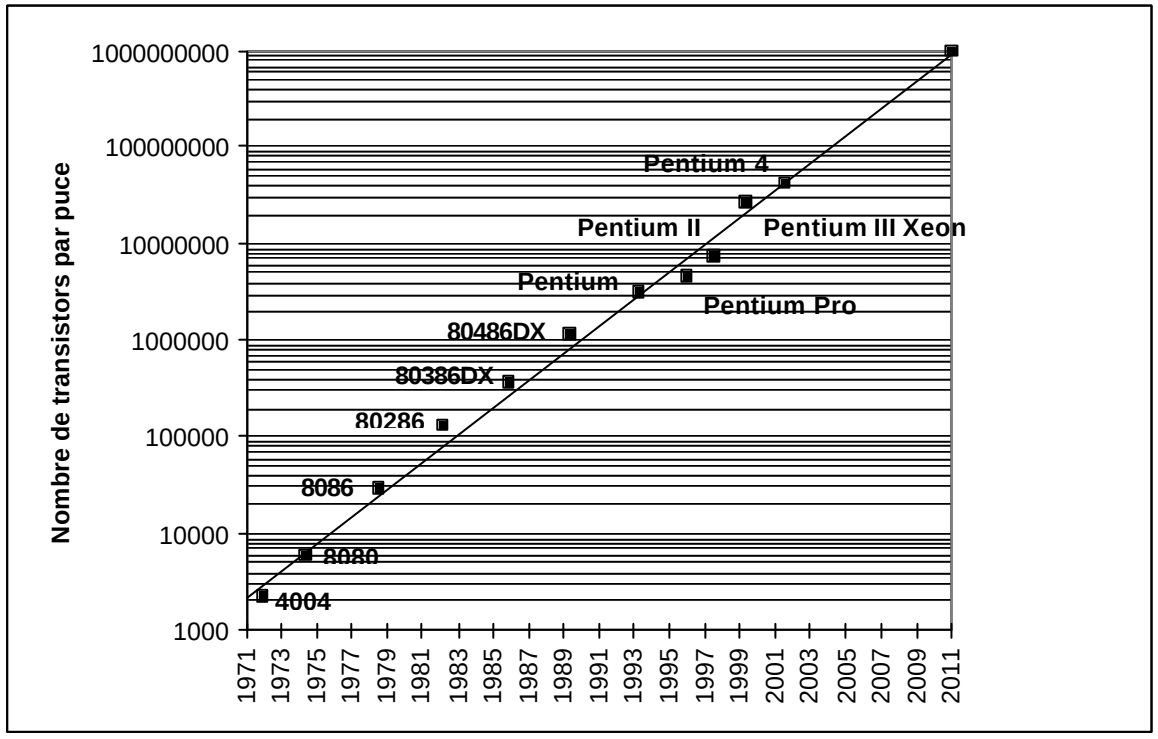
Évolution de la performance relative des processeurs (dérivée de [HEN91])



- **Première loi de Moore : la densité des circuits intégrés double tous les 18 mois**
- **Dérivation : la performance des microprocesseurs double tous les 18 mois**
- **Observation : la performance des microprocesseurs double tous les**
  - **24 mois en fait**
  - **19 mois d'après les projections des fournisseurs (les « Road Maps »)**

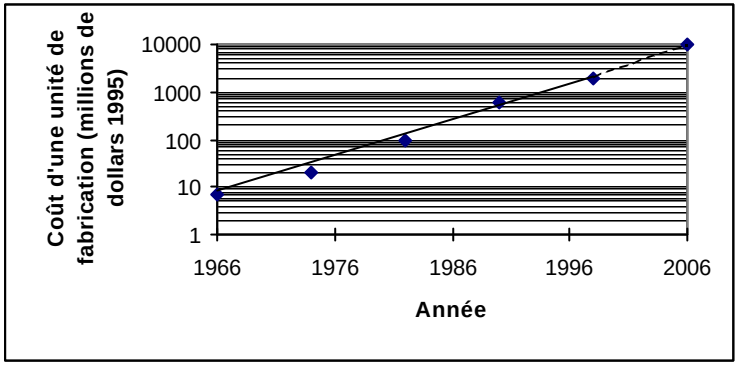
Illustration de la loi de Moore

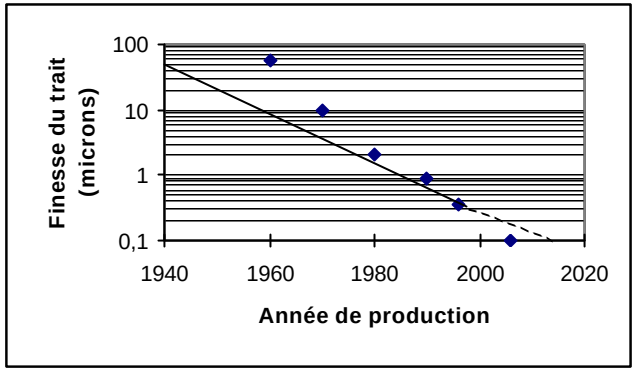
Évolution du nombre de transistors par puce (Source Intel)



Note: des évolutions similaires peuvent être observées chez d'autres fournisseurs

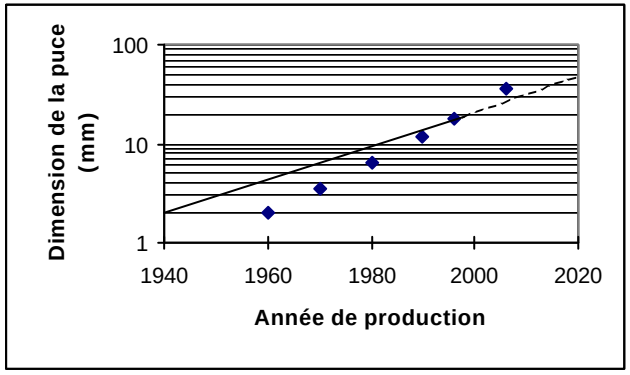
Semi-conducteurs (2)

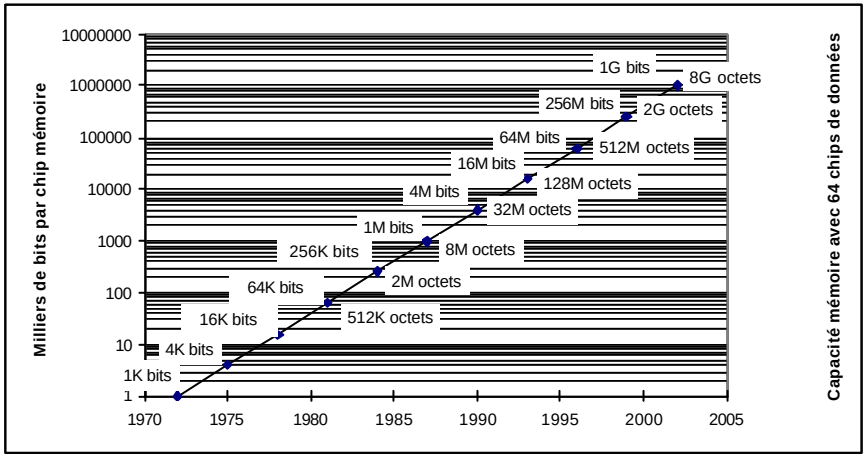




Chiffre Intel octobre 2001 : Nouvelle usine (Chandler AZ) 0.13µ/tranches de 200 mm coût $2B, 4200 personnes et 18 mois de construction.

Note : diamètre d'un cheveu humain = 70µ

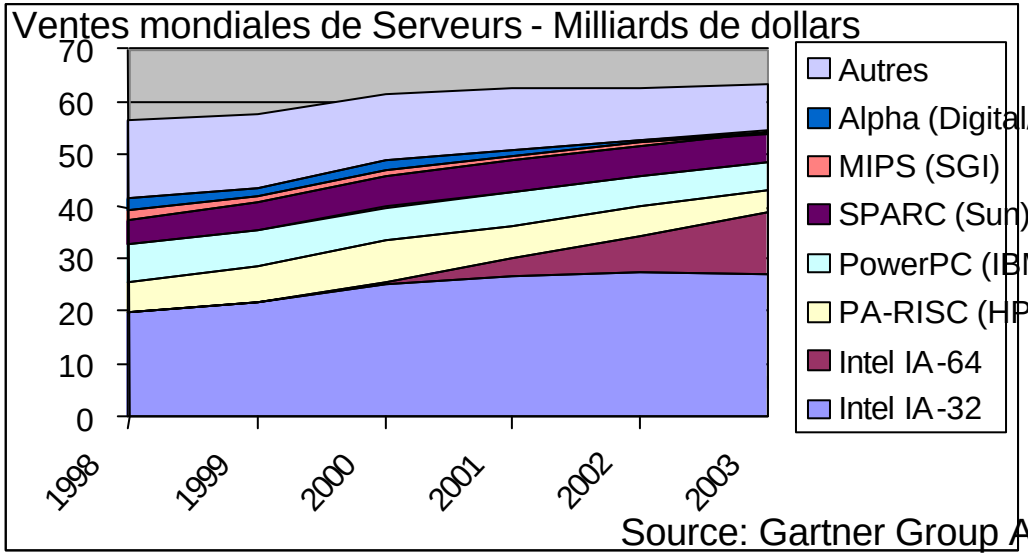




Si n est la finesse, Nombre de transistors par puce $O(n^2)$, Fréquence de la puce $O(n)$ -> Potentiel d'amélioration $O(n^3)$

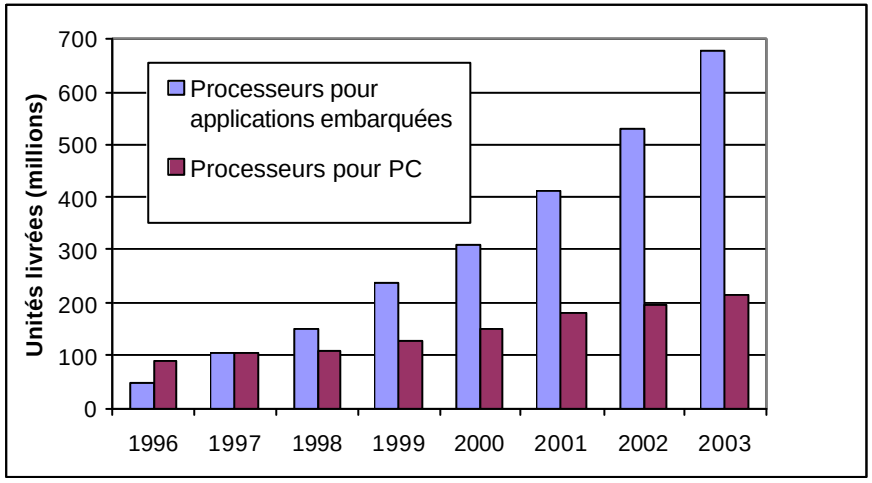
Facteurs économiques

- **Parts de marché des serveurs pour les différentes architectures**



Source: Gartner Group Avril 99

- **Prévision des ventes de microprocesseurs (32 et 64 bits) pour les applications de traitement de l'information et pour les applications embarquées [HEN99]**



Facteurs économiques (2)

- **Évolution des coûts de conception [HEN99]**

Microprocesseur	Année d'introduction	Nombre de transistors (millions)	Nombre de participants au développement	Durée du développement (mois)	Estimation du coût de la main d'œuvre (Millions de dollars)	Coût de la validation (pourcentage de l'effort total)
R2000	1985	0,1	20	15	2,5	15 %
R4000	1991	1,4	55	24	11	20 %
R10000	1996	6,8	>100	36	30	> 35 %

Coût de développement d'une nouvelle implémentation

- **Nouvelle micro-architecture : évolution majeure O(100 millions de $)**
- **Évolution technologique (fréquence, taille des caches) : O(10 millions de $)**
- **Nouvelle architecture O(milliard de $)**

- **Coût de fabrication d'un microprocesseur O(10$) ou O(100$)**
- **Conséquences pratiques**
  - **Seuls des grands volumes de production (plusieurs millions d'unités) peuvent permettre d'amortir les frais de R&D**
  - **Concentration de l'industrie informatique autour de quelques architectures (2 à 3)**

Performance des microprocesseurs

- **Équation de base de la performance d'un processeur**

  temps_tâche = nombre_instructions_tâche x cycles_instruction x temps_cycle

  - **Facteurs contribuant aux termes de cette équation**
    - **Nombre d'instructions par tâche:**
      - **Algorithme, compilateur optimisant, répartition des fonctions entre programme d'application et système d'exploitation,**
      - **adéquation de l'architecture au problème à traiter**
    - **Nombre de cycles par instruction:**
      - **Compilateur optimisant, définition de l'architecture, implémentation de l'architecture (pipeline, superscalaire, super-pipeline,....)**
    - **Temps de cycle:**
      - **Définition de l'architecture, technologie, implémentation de l'architecture**
- **Amélioration des performances : le parallélisme. Deux voies possibles, mais complémentaires, au niveau des microprocesseurs :**
  - **Au niveau des instructions : ILP (Instruction Level Parallelism), on agit sur le terme nombre de cycles par instructions**
  - **Au niveau des processus et processus légers : TLP (Thread Level Parallelism), on cherche à exécuter simultanément plusieurs flots d 'instructions**

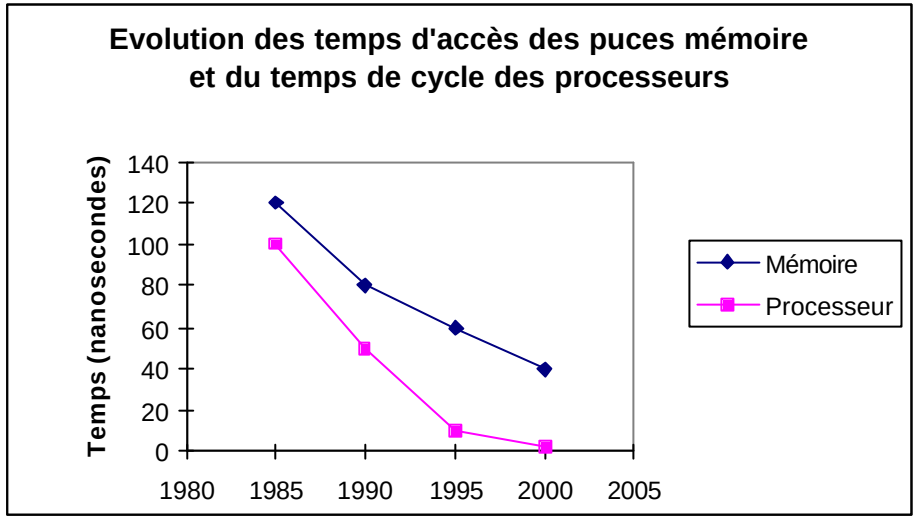
Microprocesseurs et parallélisme

- **Augmentation de la performance au niveau d'un flot d'instructions : ILP pour Instruction Level Parallelism**
  - **Moyens**
    - **Définition d'une nouvelle architecture permettant de mettre en évidence le parallélisme. Exemples : IA-64 (Itanium) d'Intel/HP, VLIW (Very Large Instruction Word) : Crusoe de Transmeta**
    - **Techniques d'amélioration de la performance au niveau de l'implémentation de l'architecture (micro-architecture) et s'appliquant aussi bien à une architecture existante (exemple IA-32) qu'à une nouvelle architecture :**
      - **Renommage des registres**
      - **Exécution dans le désordre**
      - **Exécution spéculative**
      - **Prédiction de branchement**
      - **....**
- **Augmentation de la performance globale (sur plusieurs flots d'instructions) : TLP pour Thread Level Parallelism**
  - **Moyens**
    - **Multi-threading simultané (SMT - Simultaneous Multithreading)**
      - **Plusieurs contextes de threads partagent les mêmes ressources de traitement. Micro-commutation de contexte en cas d'attente sur un thread**
      - **Exemple : PowerPC G3 (NorthStar), Successeur Itanium (fondé sur le projet Alpha 21464)**
    - **Multiprocesseur sur une puce (MPC - MultiProcessor Chip)**
      - **Plusieurs processeurs indépendants sur la même puce**
      - **Exemple : Power 4 IBM**

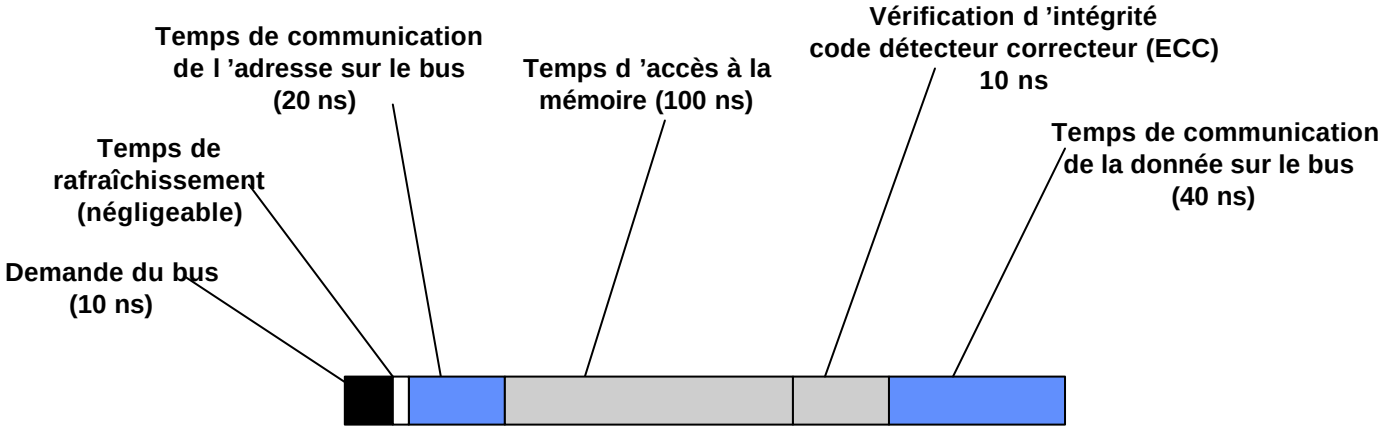
Notions de hiérarchie de mémoire

Hiérarchie de mémoire

- **Écart grandissant entre les temps d'accès aux boîtiers mémoire et le temps de cycle des microprocesseurs**

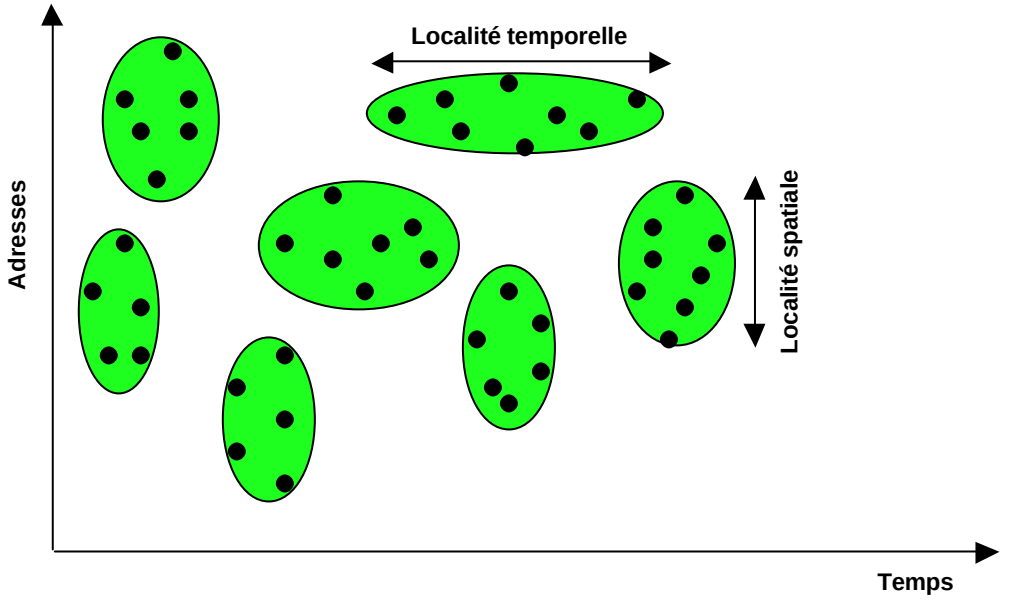


- **Le temps d'accès aux boîtiers n'est qu'une composante du temps d'accès mémoire (exemple indicatif, lecture d'un granule de 64 octets)**



Hiérarchie de mémoire(2)

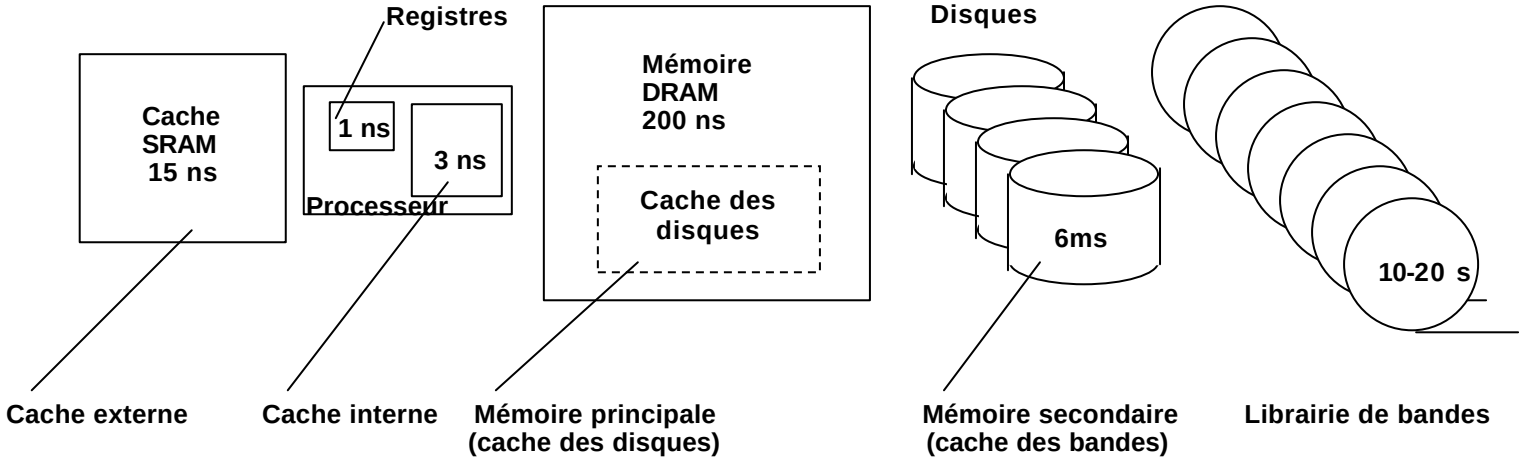
- **Propriété de localité spatio-temporelle**
  - **Spatiale : si une donnée est référencée, il est fort probable que les données voisines le seront dans un avenir proche**
  - **Temporelle : si une donnée est référencée, il est fort probable qu'elle le soit de nouveau dans un avenir proche**



- **Principe du cache : maintenir les informations récemment référencées dans des niveaux de mémoire rapide**
- **Échanges entre niveaux fondés sur le concept de granule (exemple : 32 ou 64 octets)**

Hiérarchie de mémoire(3)

Niveaux de mémorisation dans un système



Caractéristiques comparées des niveaux de mémoire

Technologie	Temps d'accès typique	Échelle humaine	Capacité (indicative)	Indication de prix ($/Mo)
Registre de processeur	1 à 2 ns	1 s	64 x 64 bits	(fait partie du microprocesseur)
Cache intégré	2 ns	2 s	64/512 ko	(fait partie du microprocesseur)
Cache externe	15 ns	15 s	4/8 Mo	~ 10
Mémoire principale	~180 ns	~ 3 min	1 Go	~0.125
Disque	~7 ms	~ 81 jours	73 Go par unité	~0,005
Bande	~10 à 20 s	~ 315 ans et plus	100 Go par unité	< 0,001

Caractéristiques indicatives à fin 2001

Hiérarchie de mémoire(4)

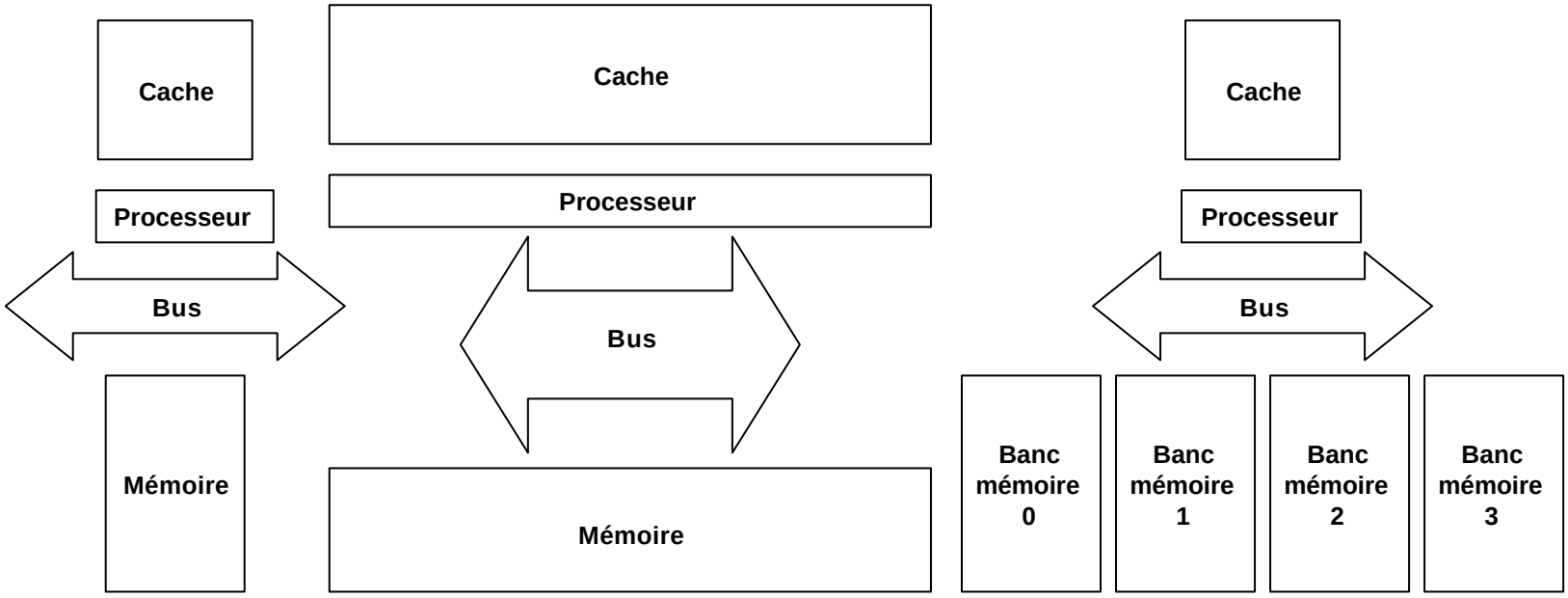
- **Exemple de temps d'accès apparent (caches de niveau 1 et 2, mémoire) sur l'exemple précédent en supposant des taux de présence (hit ratio) dans les caches de 95% et 98%.**

  temps_apparent = 0.95 x 3 + 0.03 x 15 + 0.02 x 180 = 6,9 ns

- **Paramètres de conception des caches :**
  - **Taille du granule**
  - **Caches séparés ou caches unifiés**
  - **Placement des granules dans le cache : associatif (Fully Associative), association directe (Direct Mapped),associatif par sous-ensemble (Set Associative)**
  - **Adressage virtuel ou réel**
  - **Politique de remplacement des granules**
  - **Stratégie d'écriture (Write Through, Write Back)**
  - **Accès en écriture**
- **Problèmes de cohérence des caches (voir SMP)**

Hiérarchie de mémoire(5)

- **Problème de débit des mémoires : le débit demandé par les processeurs croît beaucoup plus vite que le débit des boîtiers**
- **Organisations mémoires pour améliorer le débit :**



(a) - Mémoire de largeur un mot (Référence)

(b) - Mémoire de grande largeur

(c) - Mémoire avec entrelacement

- **Problèmes :**
  - **Mémoire de grande largeur : modularité, coût**
  - **Entrelacement : modularité**

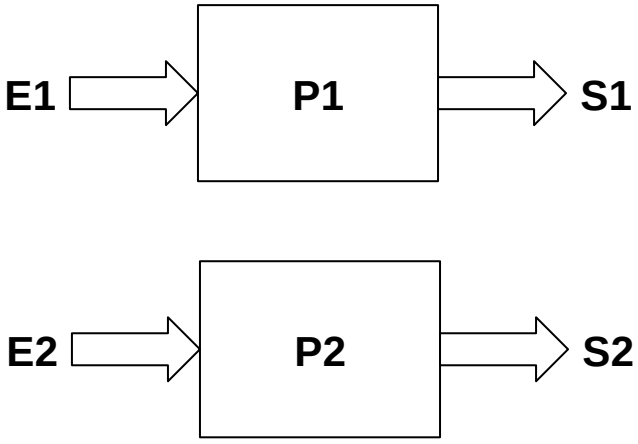
Hiérarchie de mémoire(6)

Synthèse

Type de cache/Propriétés	Niveau 1 et Niveau 2	Externe	Cache disque	Hiérarchie de mémoire de stockage
Emplacement (logique)	(intégré au microprocesseur)	Entre microprocesseur et mémoire	En mémoire	Sur disque
Technologie	SRAM intégrée au microprocesseur	SRAM	DRAM	Disque
Nature de la mémoire cachée	Cache externe ou mémoire	Mémoire DRAM	Disque	Cartouche dans un robot
Caractéristiques				
Capacité	O(10 ko/100 Mo)	O(1 Mo)	O(100 Mo)	O(1 Go)
Taille du granule	O(10/100 o)	O(100 o)	O(10 ko)	O(100 ko)
Temps d'accès	3 ns	15 ns	~180 ns	~6 ms
Débit	O(Go/s)	O(Go/s)	~ 1 Go/s	O(100 Mo/s)
Gestionnaire (logique)	Matériel	Matériel	Logiciel (système de fichiers ou SGBD)	Logiciel (système de gestion de hiérarchie de mémoire secondaire)

Notions de parallélisme

Parallélisme - Définitions (1)



Conditions de Philip Bernstein (1966)

- Les deux programmes P1 et P2 peuvent être exécutés en parallèle (P1||P2) ssi:
  - $\{E1 \cap S2 = \varnothing , E2 \cap S1 = \varnothing , S1 \cap S2 = \varnothing \}$
- Plus généralement, un ensemble de programmes {P1,P2,…,Pn} peut être exécuté en parallèle ssi les conditions de Bernstein sont satisfaites
  - $Pi||Pj \ \forall\{i,j\}$ avec $i \neq j$
- Notion de grain : gros grain et grain fin (coarse grain, fine grain)

Parallélisme - Définitions (2)

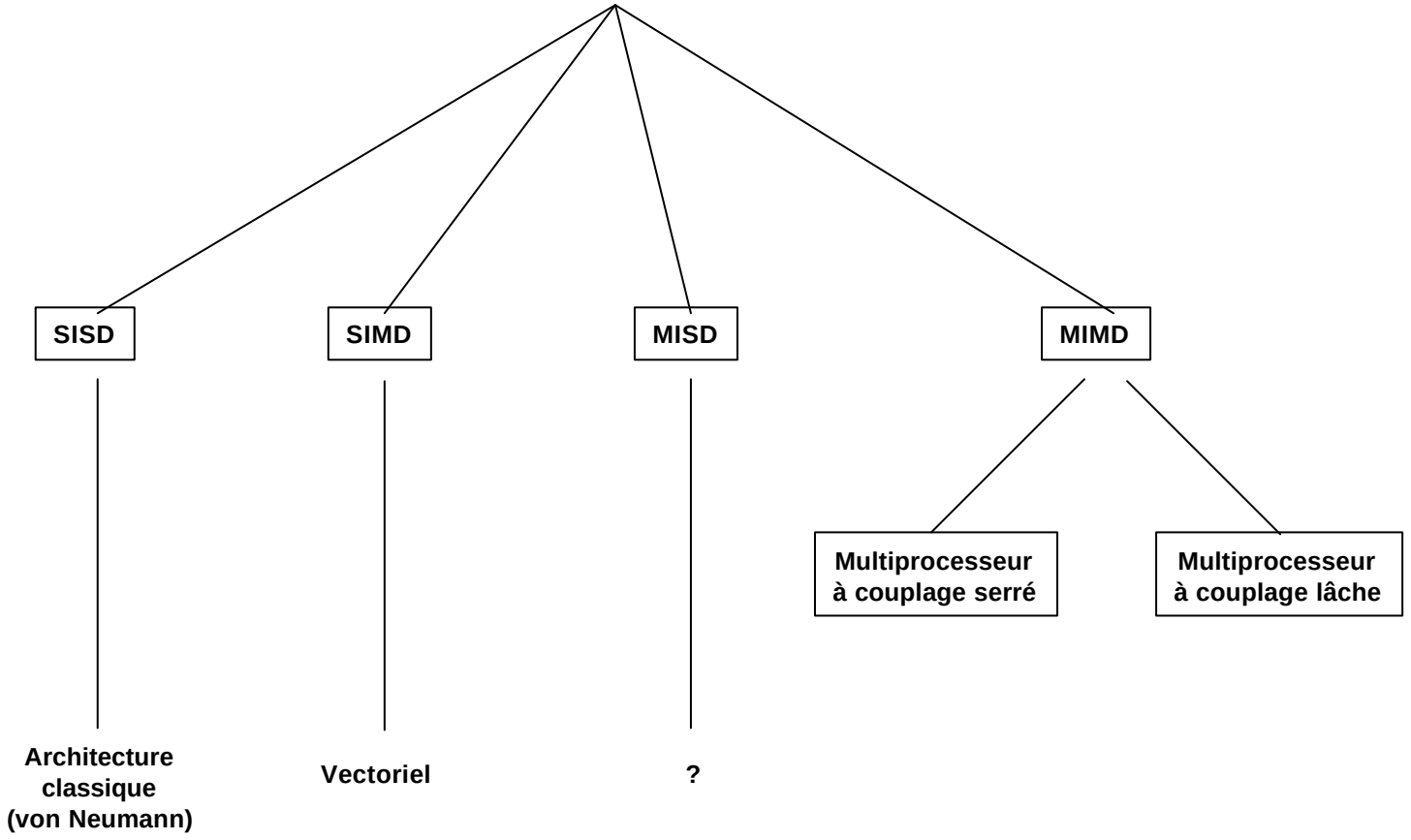
- **Sources du parallélisme**
  - **Parallélisme de données :** même opération effectuée par des processeurs différents sur des ensembles disjoints de données
  - **Parallélisme de contrôle :** des opérations différentes sont réalisées simultanément. Le programme est composé de parties indépendantes ou bien certaines structures de contrôle (telles que des boucles) sont susceptibles d'être exécutées en parallèle.
  - **Parallélisme de flux :** les opérations sur un même flux de données peuvent être enchaînées avec un certain niveau de recouvrement, c'est-à-dire que l'opération suivante peut être amorcée avant que la précédente soit terminée. C'est le mode travail à la chaîne ou mode pipeline.

Parallélisme - Définitions (3)

- **Classification de Michael Flynn [FLY72]**
  - **SISD** (*Single Instruction Single Data Stream*) ou flot unique d'instructions et flot unique de données. (modèle de John von Neumann)
  - **SIMD** (*Single Instruction Multiple Data Stream)*, ou flot unique d'instructions et flots multiples de données, dans laquelle le même flot d'instructions est appliqué à des ensembles disjoints de données
  - **MISD** (*Multiple Instruction Single Data Stream*), dans laquelle plusieurs flots d'instructions sont appliqués à une même flot de données
  - **MIMD** (*Multiple Instruction Multiple Data Stream)* dans laquelle des flots d'instructions indépendants sont appliqués à des ensembles de données indépendants. Dans le cas des SGBD parallèles, c'est le même programme qui est exécuté simultanément (mais sans synchronisme) sur des ensembles disjoints de données, et on parle alors de SPMD (*Single Program Multiple Data Stream)*

Parallélisme - Définitions (4)

- **Relation entre la classification de Flynn et les différentes options d'architecture**



Limites du parallélisme (1)

Loi d'Amdahl

La loi d'Amdahl exprime l'amélioration de la performance en fonction de :

- a : fraction du programme devant être exécutée de façon séquentielle (i.e. 1-a est la partie susceptible d'être améliorée)

- P : facteur d'amélioration

$$Accélération\_Maximale \leq \frac{1}{a + \frac{(1-a)}{P}} \leq \frac{1}{a}$$

Exemple :

- fraction susceptible d'être améliorée = 0.40,

- facteur d'amélioration = 10

alors

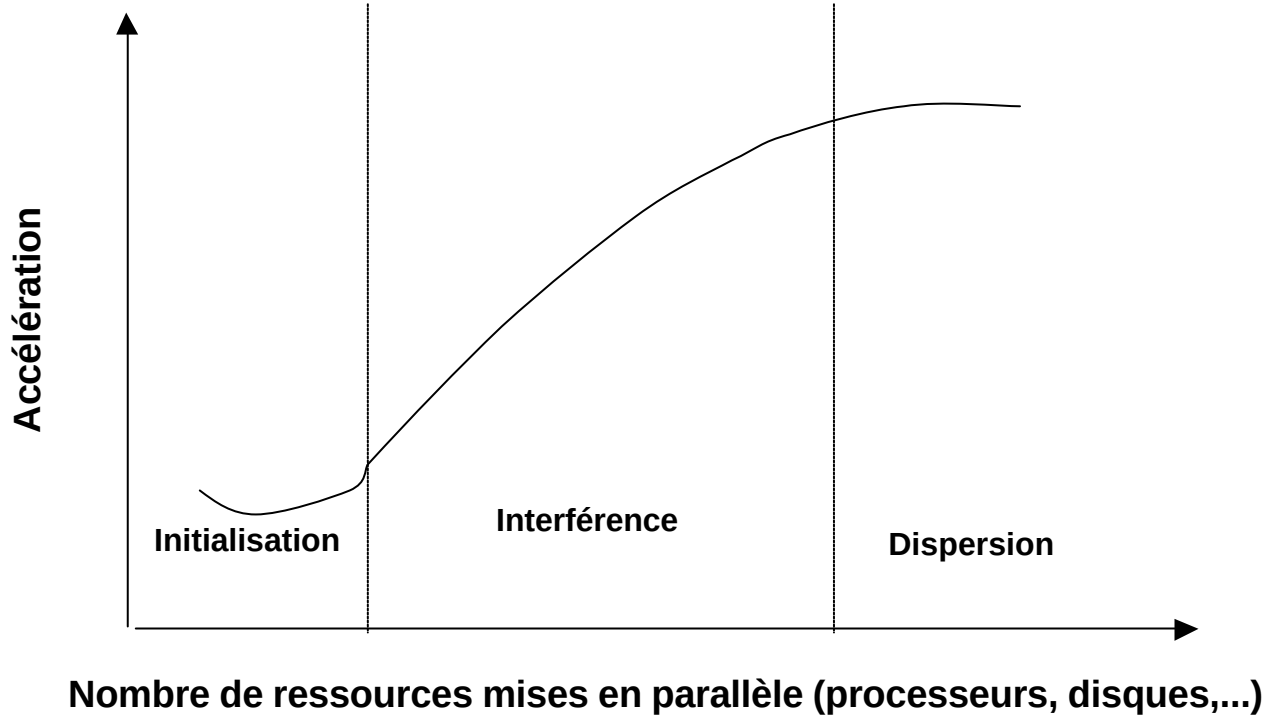
- accélération résultante = 1.56.

Si le facteur d'amélioration était infini, l'amélioration résultante ne serait que de 1.67!

Dans le cas du parallélisme, 1-a correspond à la partie « parallélisable » du programme.

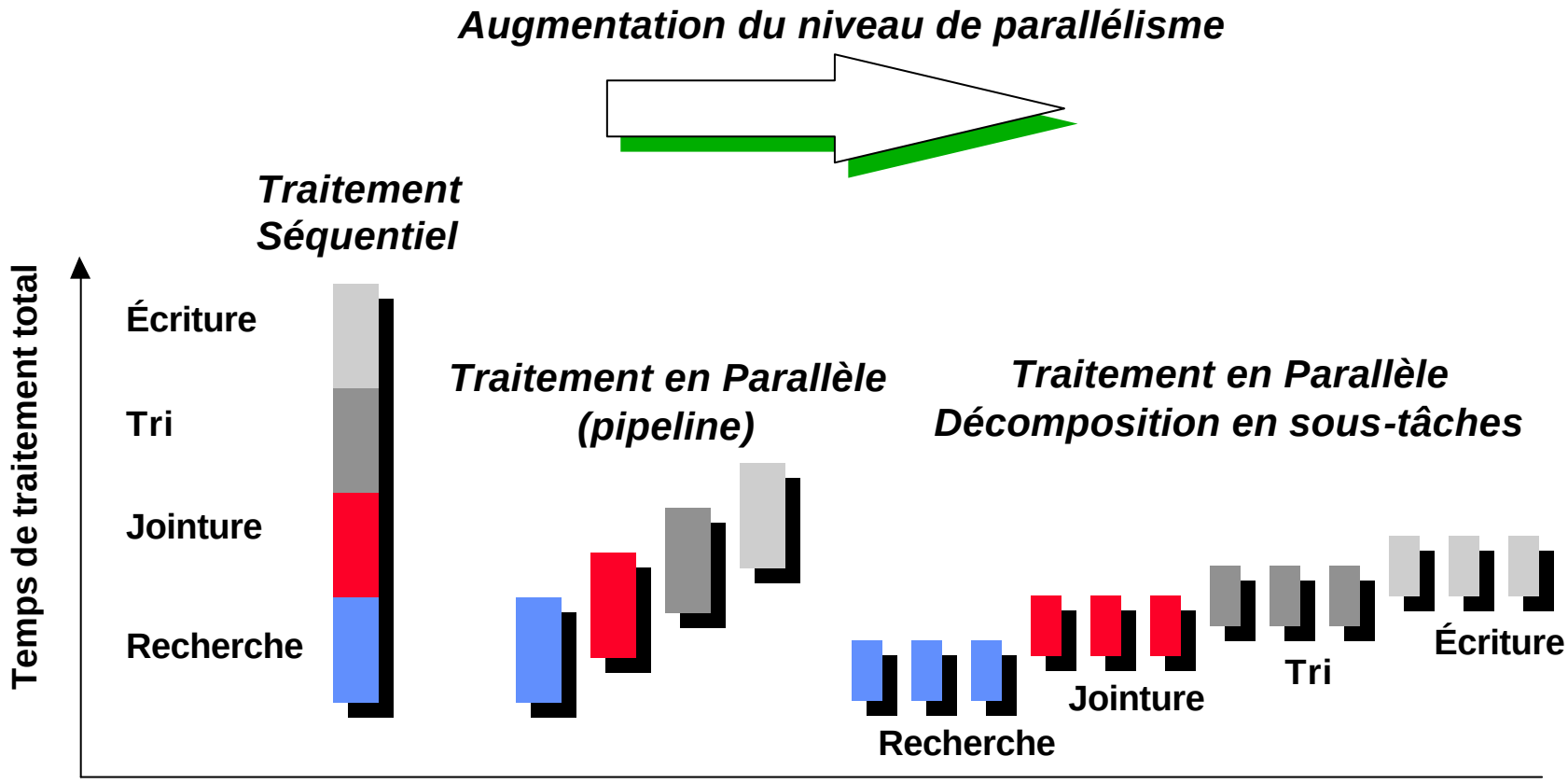
Limites du parallélisme (2)

- **Comportement typique des systèmes parallèles**



Exemple de parallélisme dans les SGBD

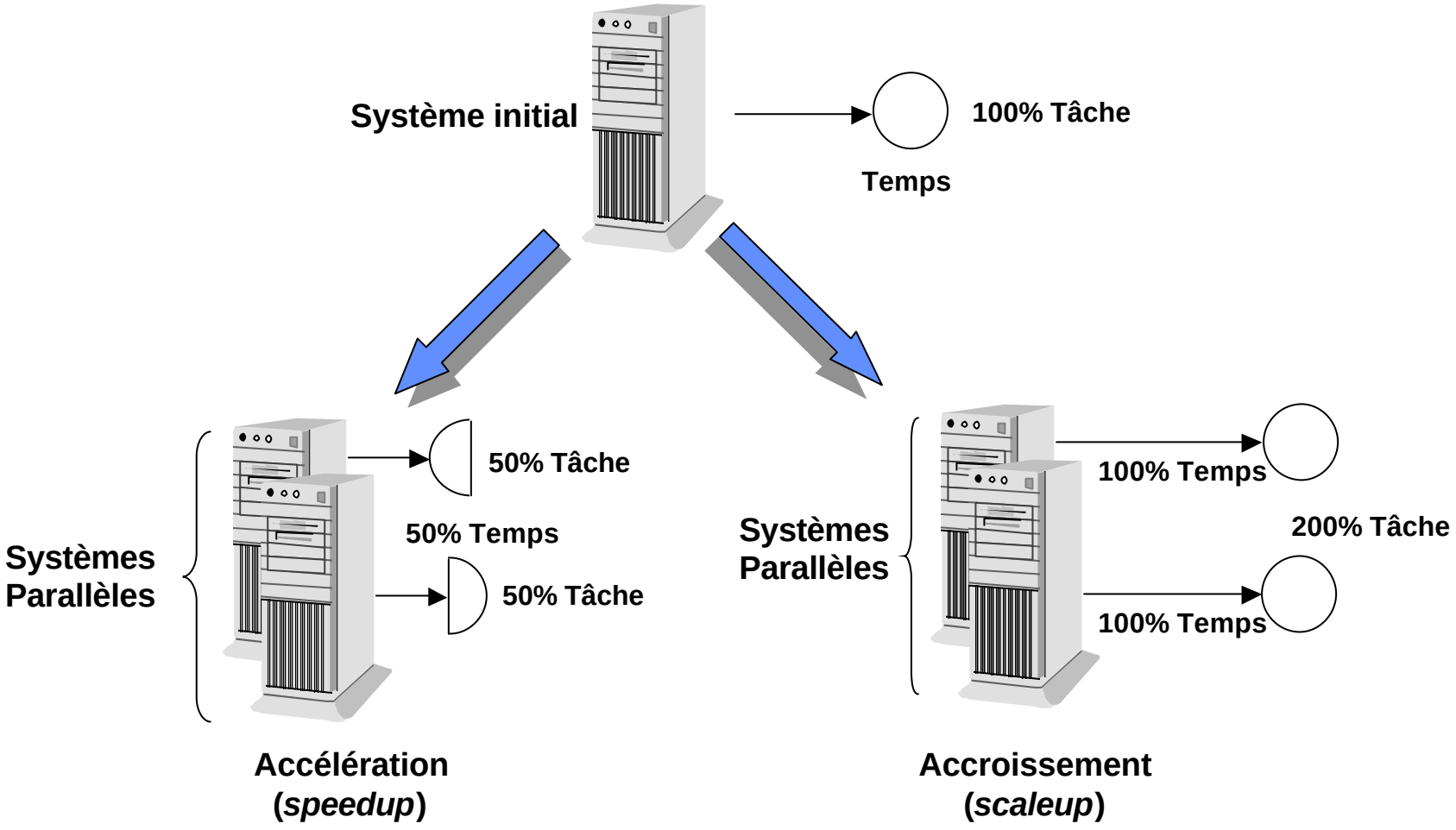
Parallélisme intra-requête



SGBD et parallélisme

- **Recherche de la performance par l'exploitation en parallèle des ressources du système**
- **Un système parallèle "idéal" possède les deux propriétés suivantes (DeWitt/Gray [DEW92]):**
  - **Linear Speedup (Accélération linéaire)**
    - **N fois plus de ressources permettent de traiter un même problème en N fois moins de temps (cas typique du DSS)**
  - **Linear Scaleup (Accroissement linéaire)**
    - **N fois plus de ressources permettent de traiter, dans le même temps, un problème N fois plus important (cas typique de l'OLTP, mise à jour d'une base N fois plus importante par N fois plus d'utilisateurs)**

Speedup et Scaleup



Note: Pour la plupart des applications transactionnelles, la synchronisation fait que Speedup et Scaleup linéaires ne peuvent être obtenus.

Possibilités de parallélisation des SGBD

- **Deux possibilités de parallélisation existent [MOH94]:**
  - **Parallélisme de traitement.** La requête est décomposée en requêtes élémentaires, qui sont exécutées en parallèle
  - **Parallélisme de données.** L'exécution de la requête s'opère en parallèle sur des sous-ensembles des données
- **En pratique**:
  - Le parallélisme de traitement est limité par le nombre d'opérateurs mis en œuvre dans les requêtes et les dépendances entre les opérateurs
  - Le parallélisme de données offre des possibilités bien supérieures en divisant une relation en plusieurs sous-tables (partitionnement)
  - On peut associer les deux formes de parallélisme en exécutant en parallèle des ordres s'adressant à des sous-tables

Situations de saturation

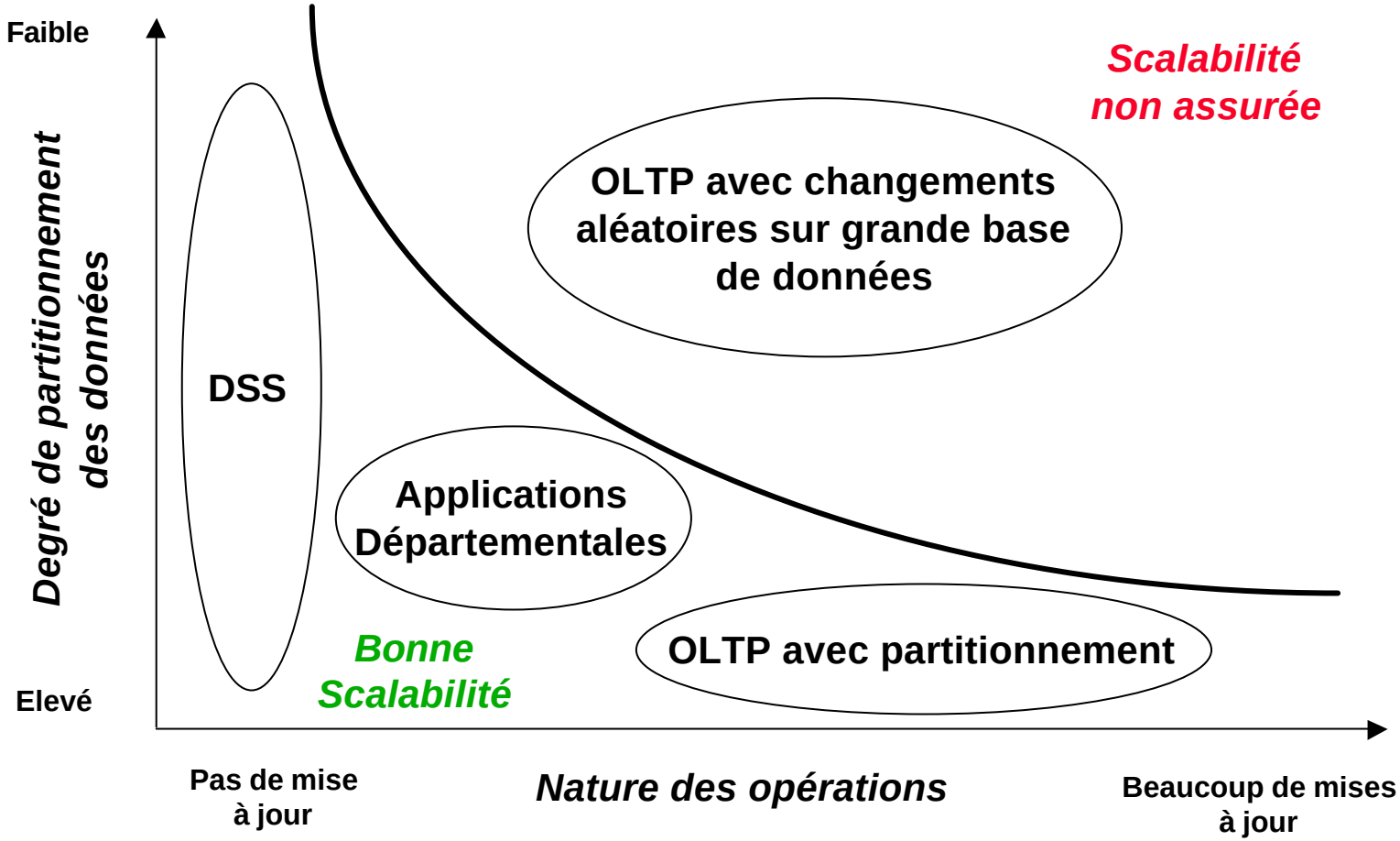
- **Deux cas typiques de saturation de l'utilisation des ressources d'un système informatique par un SGBD**
  - **Saturation des ressources de traitement.** Cette situation est appelée *CPU bound*, parce que la performance du système est limitée par les processeurs
  - **Saturation des entrées-sorties.** Cette situation est appelée *I/O bound*, parce que la performance du système est limitée par les entrées-sorties
- *Compte tenu de l'évolution du potentiel des technologies, on cherche toujours à se placer dans la situation « CPU bound »*

Quand utiliser les SGBD parallèles?

- **Applications DSS (Aide à la décision)**
  - **OK si peu de modifications de la base (update, insert, delete)**
- **Applications OLTP**
  - **OK si données partitionnées et placées de façon adéquate**
  - **OK pour applications mettant à jour des bases de données séparées (e.g. partitionnement par "département")**
  - **Difficultés possibles dans le cas de changements "random" sur de grandes bases de données**

Quand utiliser les SGBD parallèles? (2)

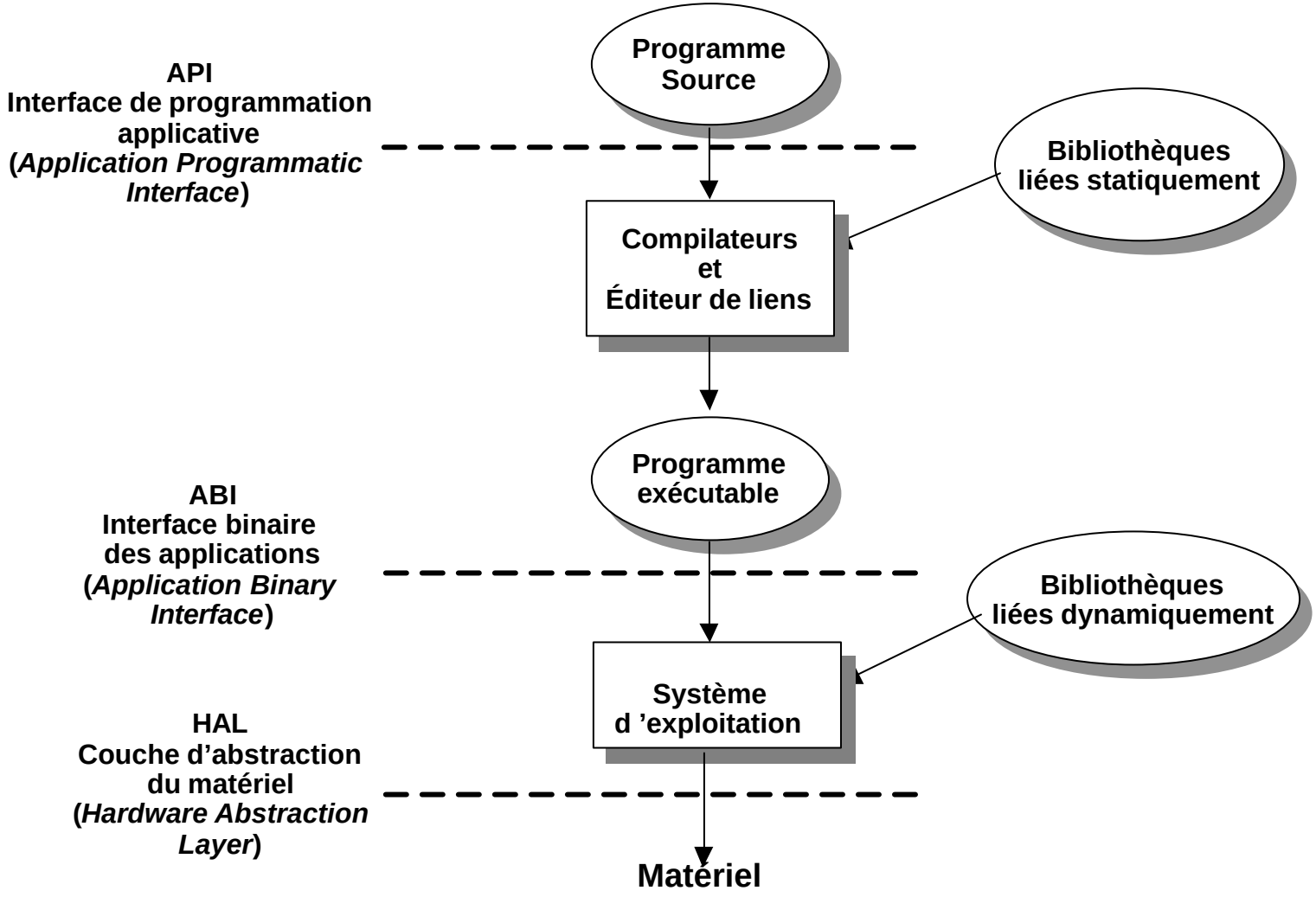
Scalabilité des applications



Source: Oracle

Compatibilité binaire - Java

Niveaux de compatibilité

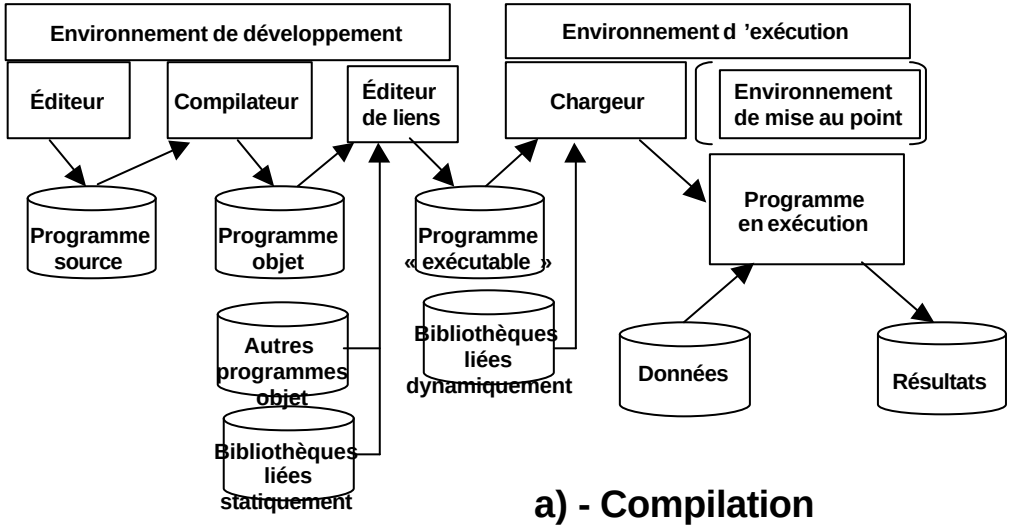


Compatibilité binaire(2)

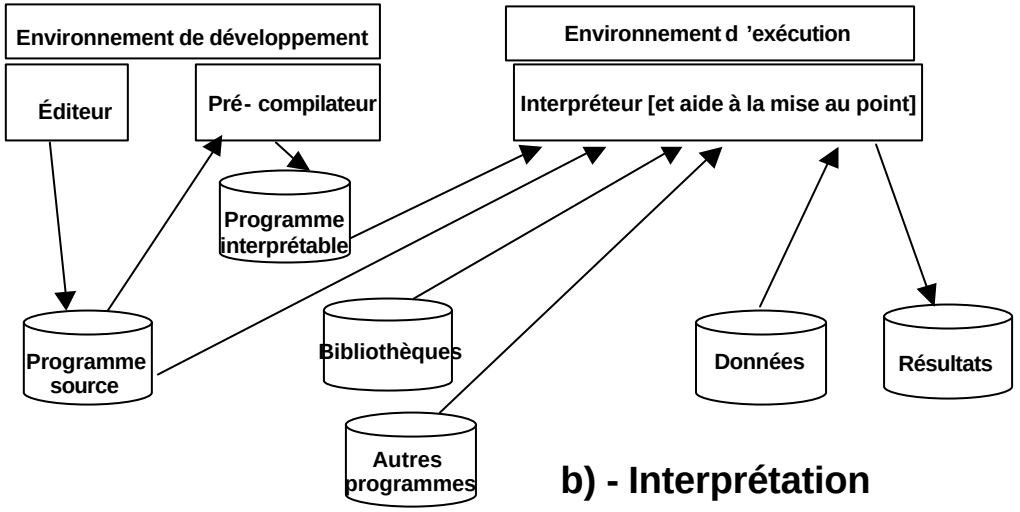
- **Résulte de**
  - **Architecture du processeur (répertoire d'instructions)**
  - **Conventions d'adressage et de communication imposées par le système d 'exploitation**
  - **Interfaces avec le système d 'exploitation et les bibliothèques**
  - **Représentation des données**
- **Standard de distribution des logiciels**
- **Frein à l'introduction de nouvelles architectures et de nouveaux systèmes d'exploitation**
- **Une tentative de s'affranchir de cette contrainte : le langage JAVA (Sun)**
  - **Dérivé de C++**
  - **Génération d'un code interprétable indépendant des architectures**
  - **Applications pouvant être distribuées par les réseaux**
    - **Applets (application niveau stations, PC ou NC)**
    - **Servlets (application niveau serveur)**
- **Reciblage des architectures : le niveau de puissance des microprocesseurs leur permet de supporter des architectures existantes pour lesquelles les développements de nouvelles générations de processeurs ne se justifient plus. Différentes techniques sont utilisées : reciblage des compilateurs, émulation, traduction de code : statique ou dynamique (DOCT, Code Morphing),...**

Compatibilité binaire(3)

Compilation et interprétation

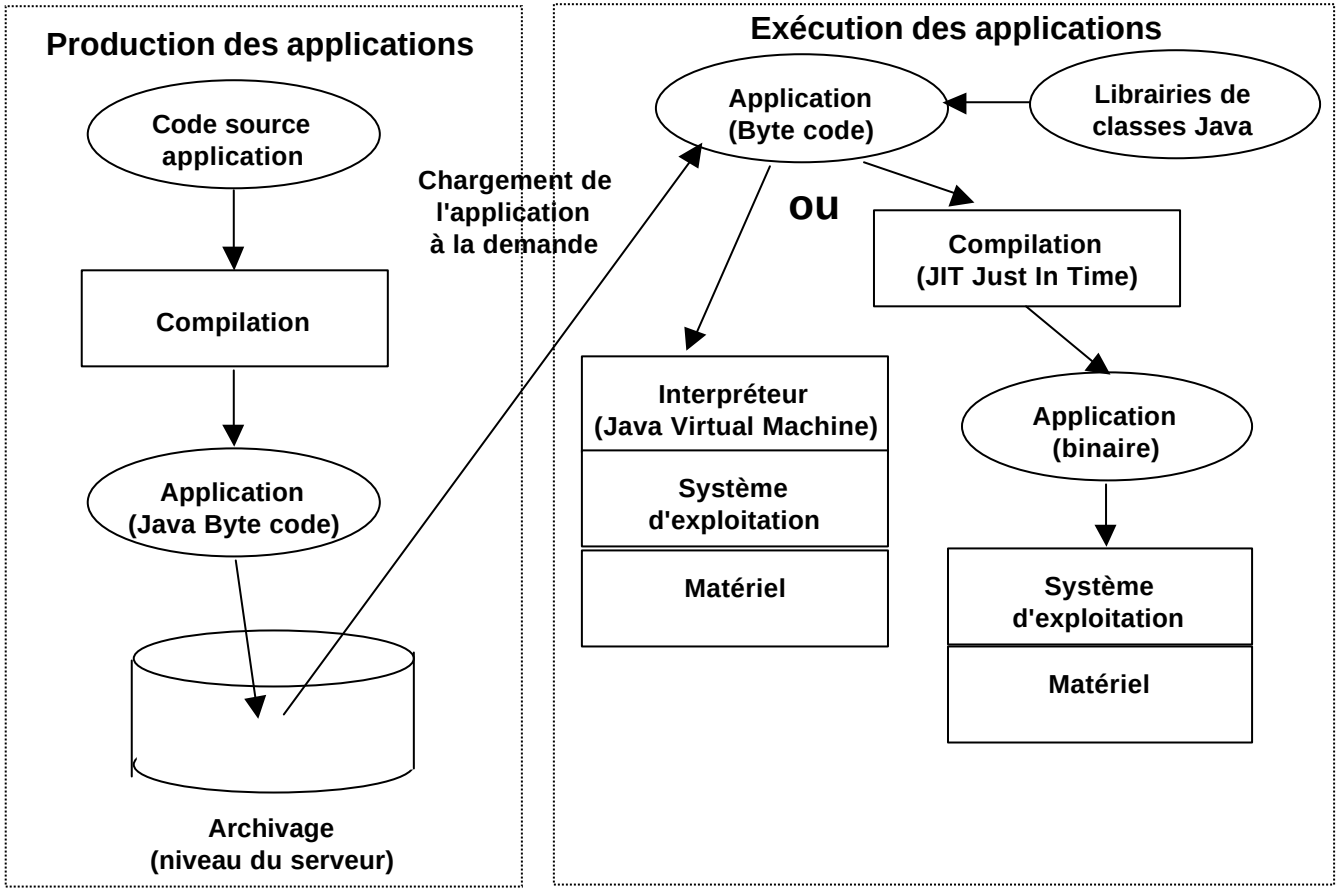


a) - Compilation



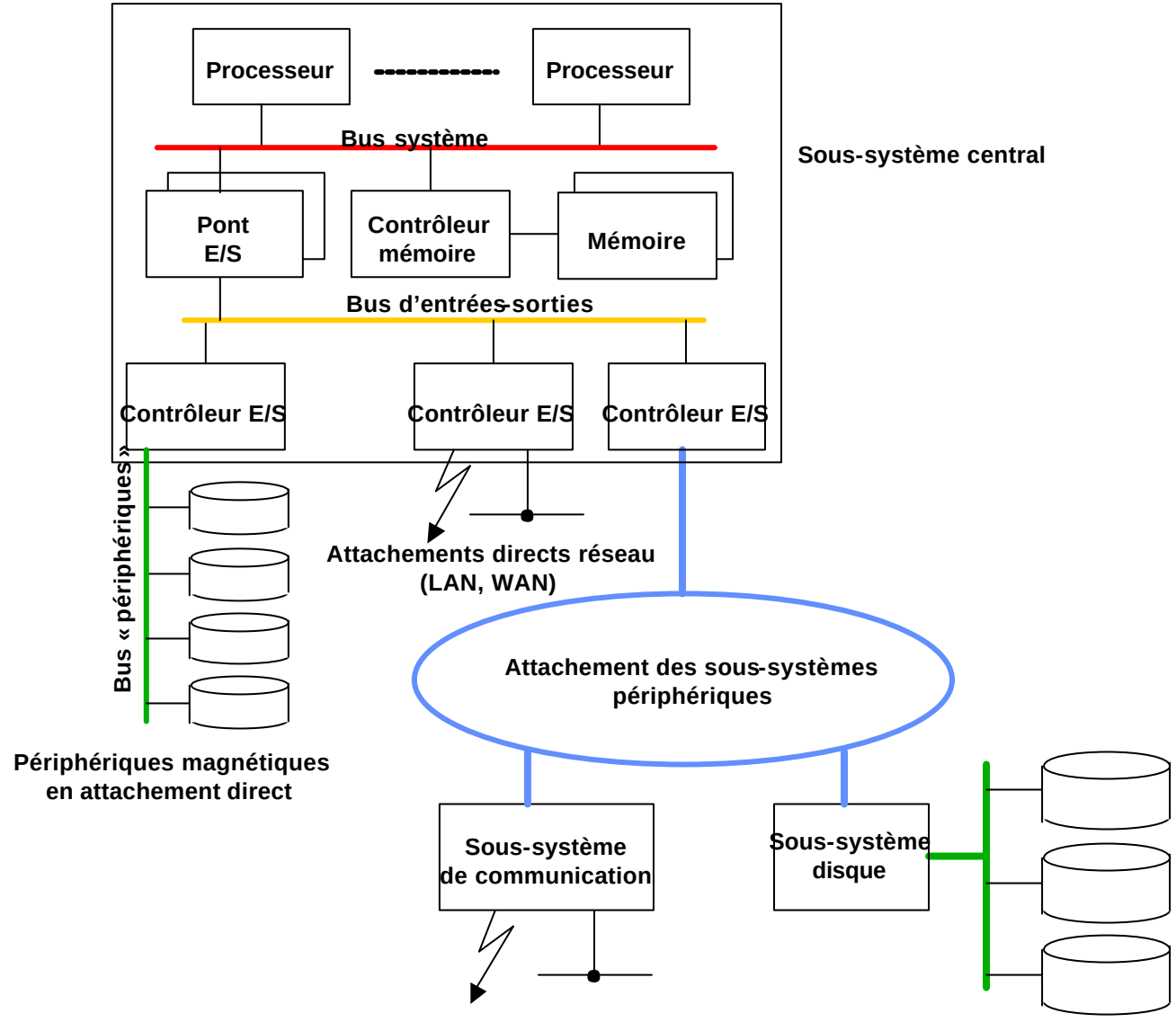
b) - Interprétation

Java : Write Once, Run Everywhere. Ne l'écrivez qu'une fois, il s'exécute partout



Entrées-sorties

Architecture générique des entrées-sorties



Entrées-Sorties(2)

- **Évolution des entrées-sorties**
  - **Alignement sur des standards de l'industrie au détriment des interfaces « propriétaires »**
    - **PCI**
    - **SCSI**
    - **Fibre Channel**
  - **Initiatives de l 'industrie pour le développement de nouveau standards (et de la technologie correspondante) :**
    - **I_2O (Intelligent I/O)**
    - **VIA (Virtual Interface Architecture)**
    - **InfiniBand**
  - **Emergence du concept de réseau de stockage SAN (Storage Area Network)**
- **Évolution des périphériques**

PCI

- **Standard de l'industrie du PC, son usage se généralise**
- **Caractéristiques résumées :**
  - **Deux « largeurs » de bus : 32 et 64 bits**
  - **Deux cadences : 33 et 66 Mhz**
  - **Débits : 133, 266, 532 Mo/s**
  - **Support simultané de différents types de contrôleurs sur le même bus**
  - **Possibilité de reconnaissance automatique (plug and play)**
  - **Possibilité de déconnexion/connexion de contrôleurs sans interrompre le fonctionnement (hot plug)**
- **Évolution avec PCI-X vers un nouveau standard (mais compatible PCI) supportant 133 Mhz (1064 Mo/s) et s'intégrant dans InfiniBand**

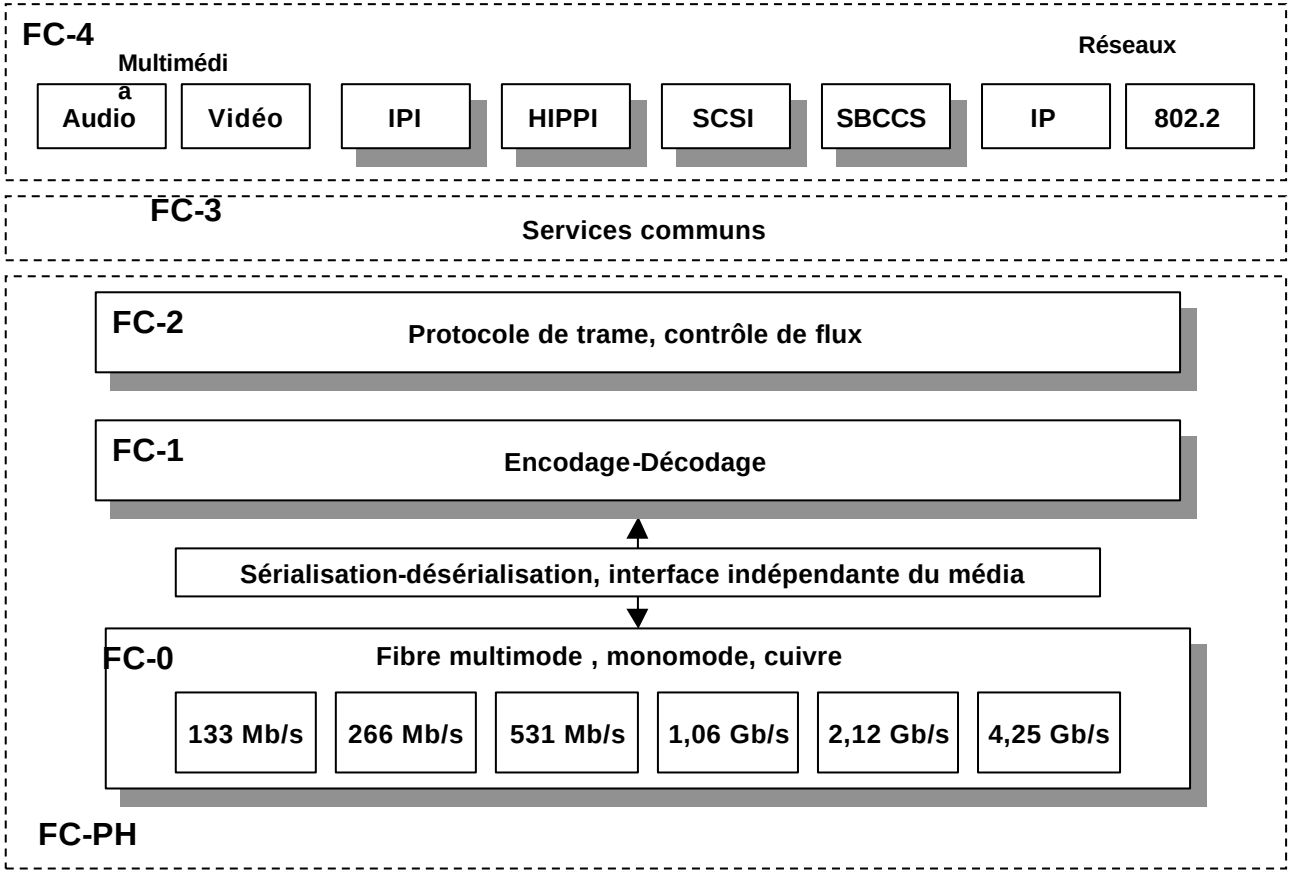
SCSI et Fibre Channel

- **SCSI : Interface parallèle vers les périphériques et les sous-systèmes. Principale limitation : longueur de la liaison**
- **Fibre Channel : Interface sérielle. Deux types de liaison :**
  - **Fibre Channel Arbitrated Loop (FC-AL) pour la connexion des périphériques**
  - **Fibre Channel pour la connexion des sous-systèmes ainsi que la connexion inter-systèmes**
- **Fibre Channel permet de supporter différents protocoles**

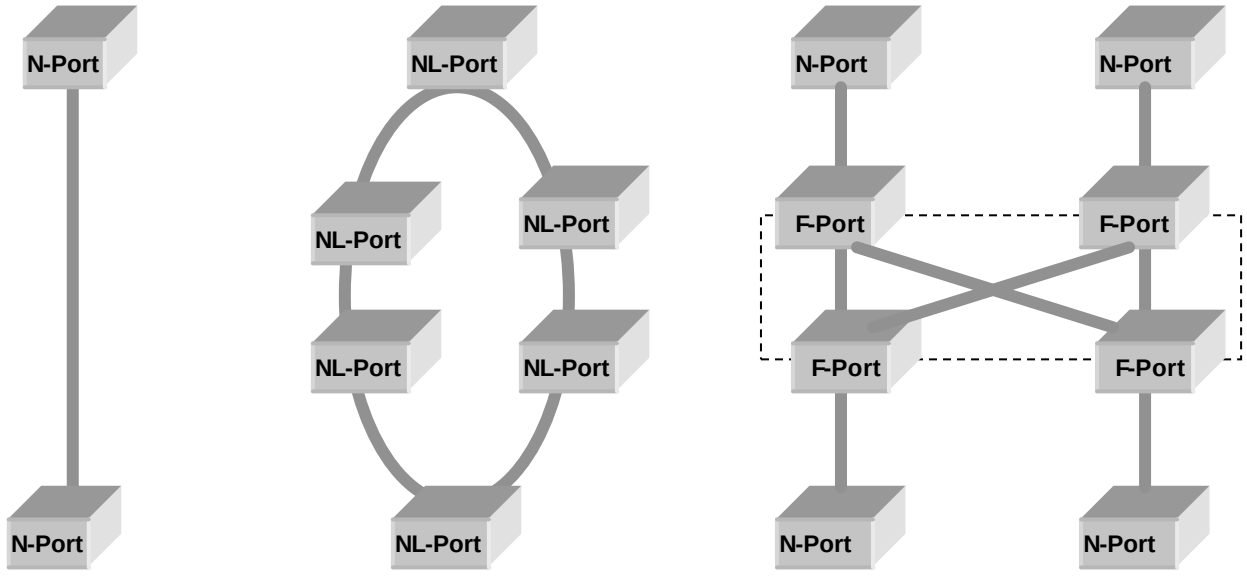
Caractéristiques comparées des canaux d'entrées-sorties, des réseaux et de Fibre Channel

Canaux d'entrées-sorties	Réseaux	Fibre Channel
• Implémentés essentiellement au niveau du matériel	• Implémentés essentiellement au niveau du logiciel	• Implémentés essentiellement au niveau du matériel
• Haute vitesse O(10 Mo/s)	• Vitesse modérée O(Mo/s)	• Haute vitesse O(100 Mo/s)
• Faible connectivité	• Forte connectivité	• Forte connectivité
• Faible latence O(µs)	• Latence modérée O(100 µs)	• Faible latence O(10 µs)
• Courte distance O(m)	• Grande distance	• Grande distance (liaison optique)
• Relation maître/asservi	• Relation de type égal à égal	• Relation maître-asservi et de type égal à égal
• Forte intégrité des données	• Fragilité	• Forte intégrité de données
• Détection d'erreur au niveau le plus fin (matériel)	• Détection d'erreur au niveau élevé (par logiciel)	• Pas de station de gestion

Architecture en couches de Fibre Channel



Topologies de Fibre Channel



Point à point (100 Mo/s)

Boucle FC-AL (100 Mo/s)

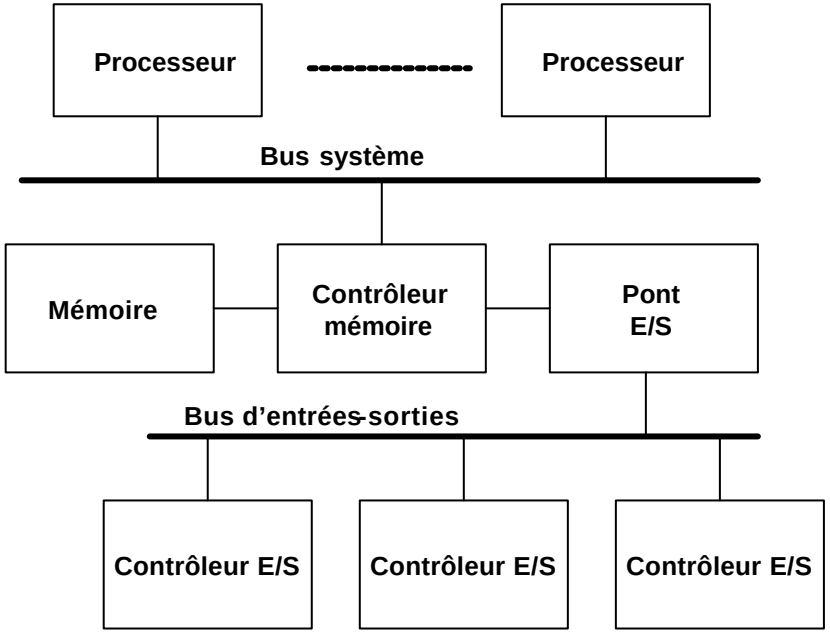
Fabric (n x 100 Mo/s)

Comparaison SCSI/Fibre Channel

Caractéristique	SCSI	Fibre Channel (boucle)
Nombre d'unités supportées	15	126
Débit	40 ou 80 Mo/s	100 Mo/s
Longueur de la liaison	Jusqu'à 25 m	- 30 m (cuivre) - 175 à 500 m (fibre optique de 62,5 ou 50 µm) - plusieurs kilomètres (fibre optique de 9 µm)
Interface physique	Câble 68 fils	Câble 4 fils ou 2 fibres optiques

Nouvelle architecture d'E/S

- **Plusieurs initiatives de l'industrie devraient améliorer les entrées-sorties des systèmes (serveurs en particulier) : InfiniBand, I_2O (Intelligent I/O) et VIA (Virtual Interface Architecture). InfiniBand et I_2O reprennent les concepts des entrées-sorties des mainframes.**
- **Problématique des entrées-sorties :**



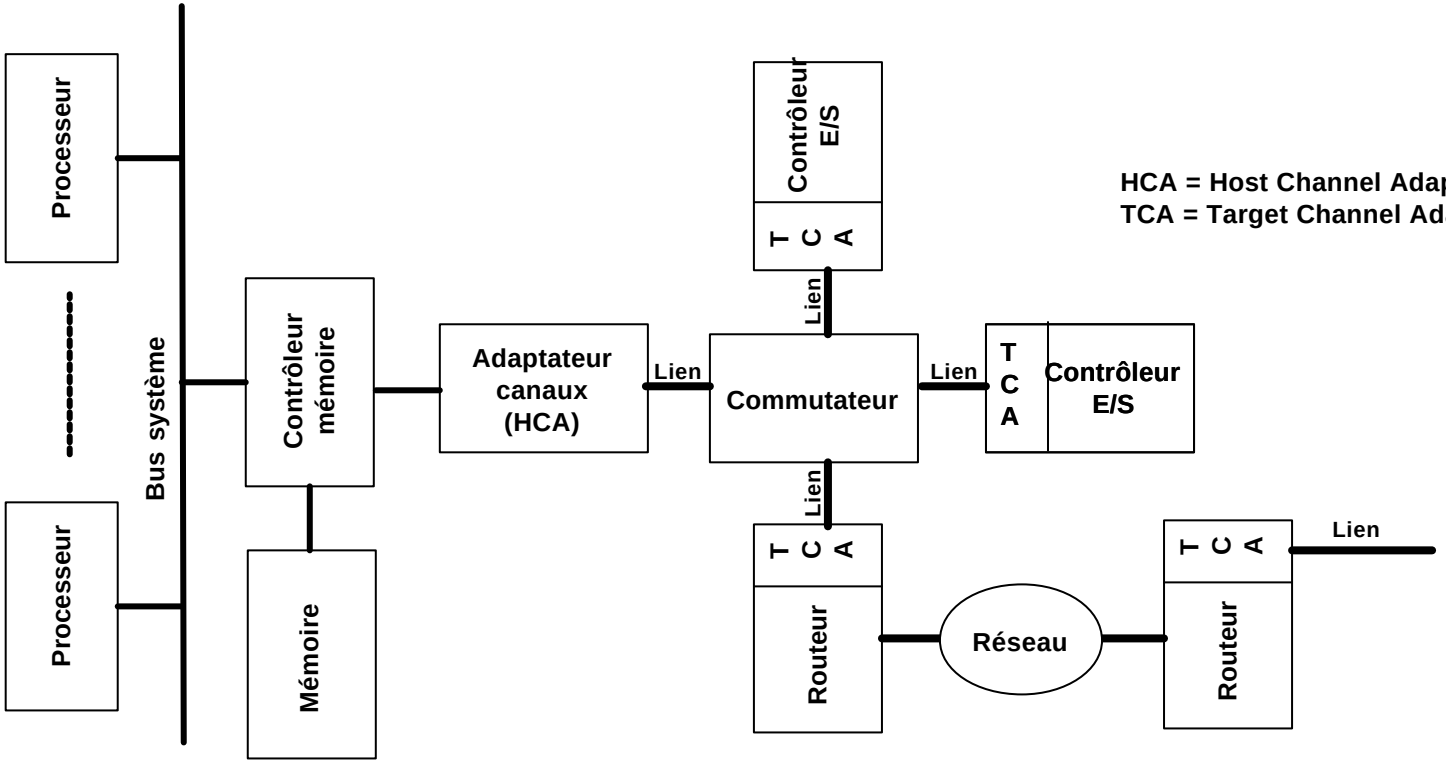
Architecture "classique" d'entrées-sorties

- **Le pont constitue un goulet d 'étranglement (1/N de la bande passante pour N contrôleurs)**
- **Contrainte sévère de distance entre le pont et le contrôleur mémoire**
- **Fréquence de fonctionnement relativement faible pour le bus afin de maintenir le nombre de contrôleur et la longueur à des valeurs acceptables (le débit double tous les 3 ans)**
- **Le partage du bus d 'E/S ne facilite pas la localisation des fautes**
- **Le mode de dialogue entre les contrôleurs et le système à un impact sur la performance**
- **Une défaillance d 'un composant peut entraîner la défaillance de l'ensemble**
- **Avec le mode « memory mapped », un contrôleur peut corrompre la mémoire du système**
- **Pont d'entrées-sorties complexe et difficile à mettre au point**

InfiniBand

- **Objectifs de l ’architecture :**
  - **Scalabilité :**
    - **architecture à base de commutateurs**
    - **capacité de raccordement (milliers de points)**
  - **Diminution de la latence et des interruptions**
  - **Évolution de la bande passante compatible avec l ’évolution de la performance des microprocesseurs**
  - **Haute disponibilité**
  - **Passation de messages**
  - **Moyen de communication unifié :**
    - **entre processus**
    - **avec le stockage**
    - **avec les réseaux**
  - **Possibilités du protocole de communication**
    - **Contrôle de flux (statique et dynamique, Qualité de Service)**
    - **Partitionnement**
    - **Multicast**
    - **Compatibilité Intranet (adressage IPv6,...)**
  - **Diminution des coûts du fait de la standardisation**
  - **Produits en 2001**

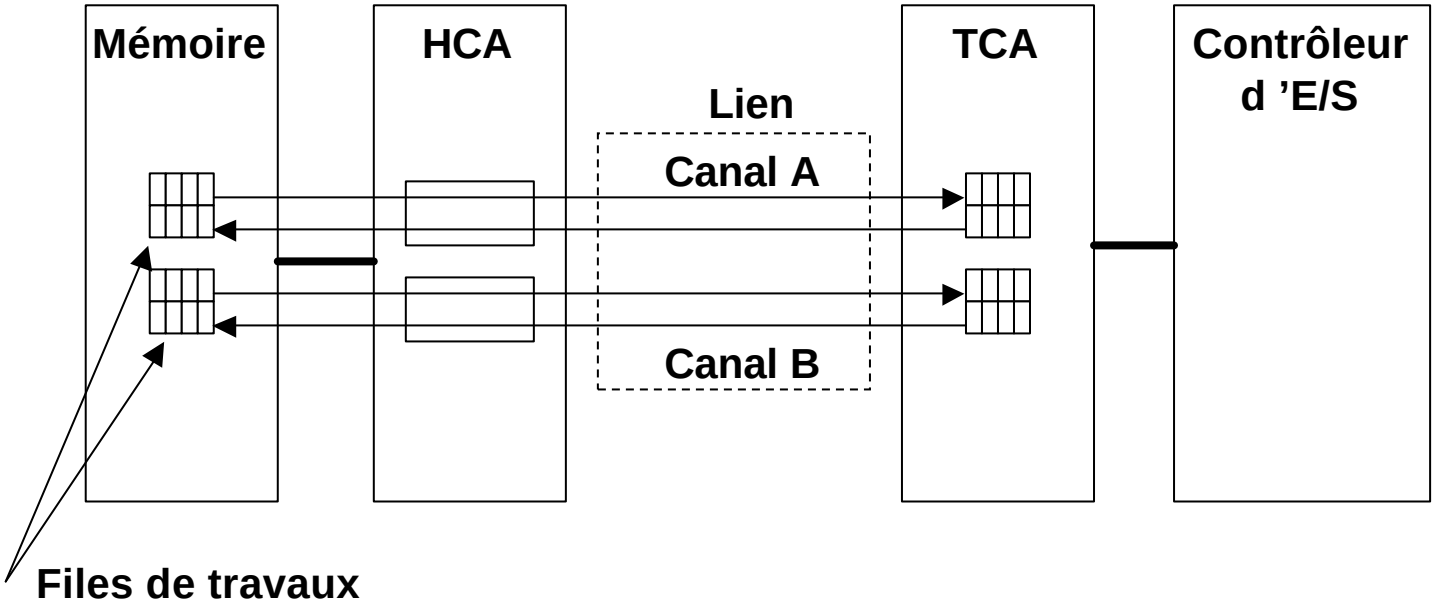
Architecture InfiniBand



Technologie des liens
- **Liens full-duplex**
- **3 « largeurs » : 1, 4 ou 12**
- **Débits :**
  - **Unidirectionnel : 256 Mo/s, 1 ou 3 Go/s**
  - **Bi-directionnel : 512 Mo/s, 2 ou 6 Go/s**
- **Capacité à évoluer vers des débits supérieurs**

Architecture InfiniBand(2)

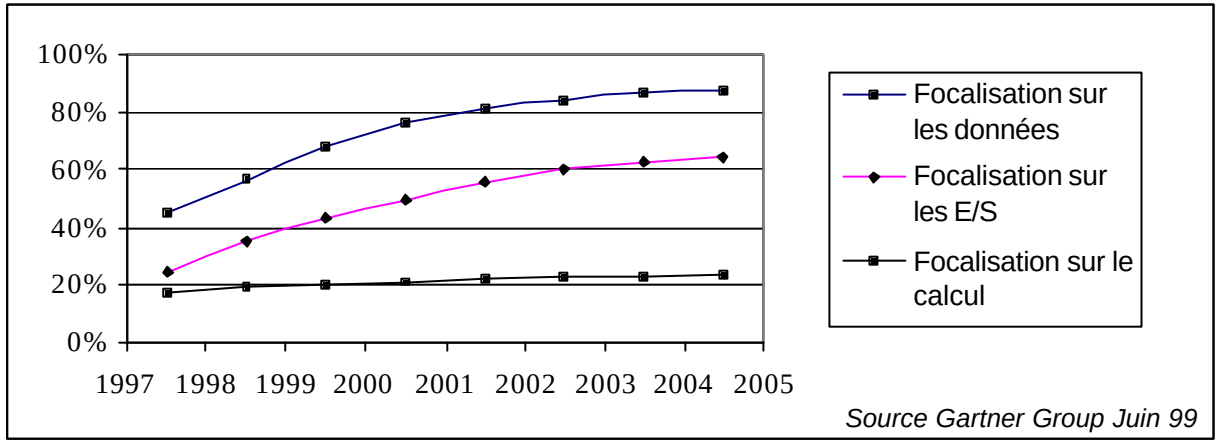
- **Notion de canal d ’entrées-sorties**
  - **Connexion logique entre deux espaces d ’adresses**
  - **Capacité DMA (Dynamic Memory Access) à chaque extrémité**
  - **Concept de queues de travaux**
  - **Concept de programme canal**



Stockage des données

- **Évolution de la part du stockage dans les coûts**
- **Évolution des disques**
- **Technologie RAID**
- **Concepts de DAS, SAN, NAS et iSCSI**
- **Exemple de TCO pour le stockage**

Évolution de la part du stockage dans les dépenses en matière de serveurs

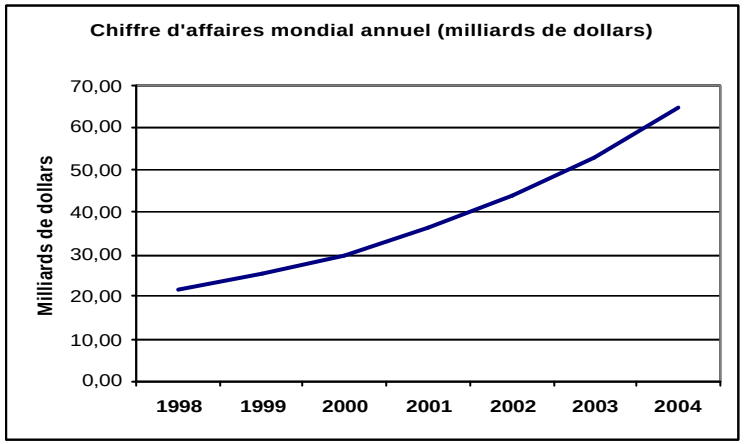


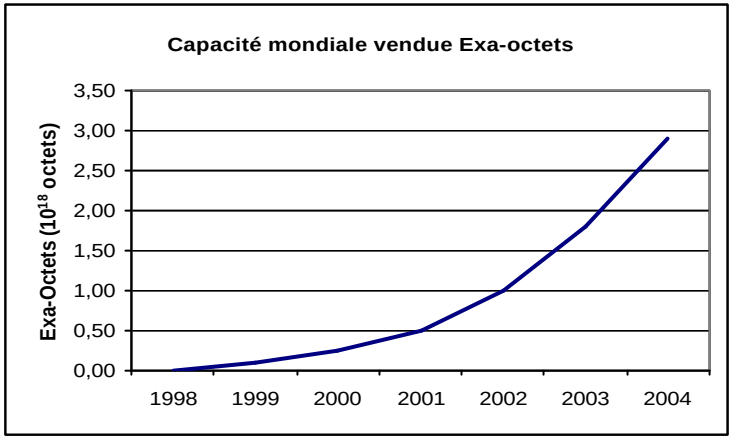
Source Gartner Group Juin 99

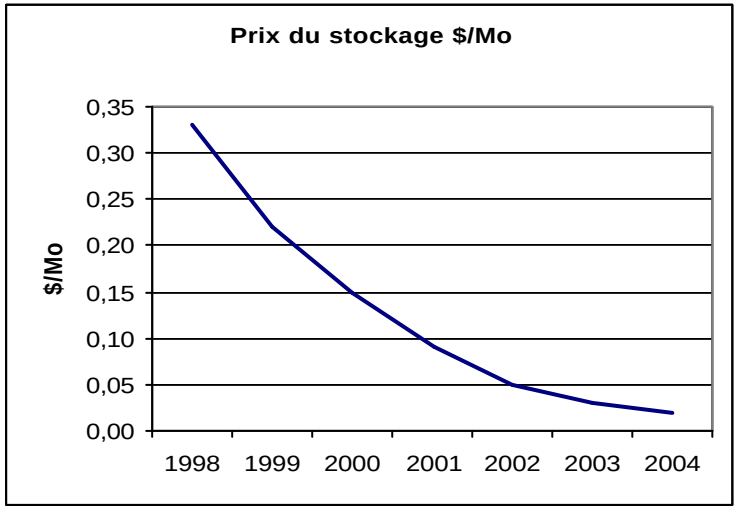
- **Focalisation sur les données : conservation de la quasi-totalité des données «en ligne»**
- **Focalisation sur les entrées-sorties : le temps d'accès aux données est une dimension critique (OLTP)**
- **Focalisation sur le calcul : les caractéristiques du stockage des données ne sont pas critiques**

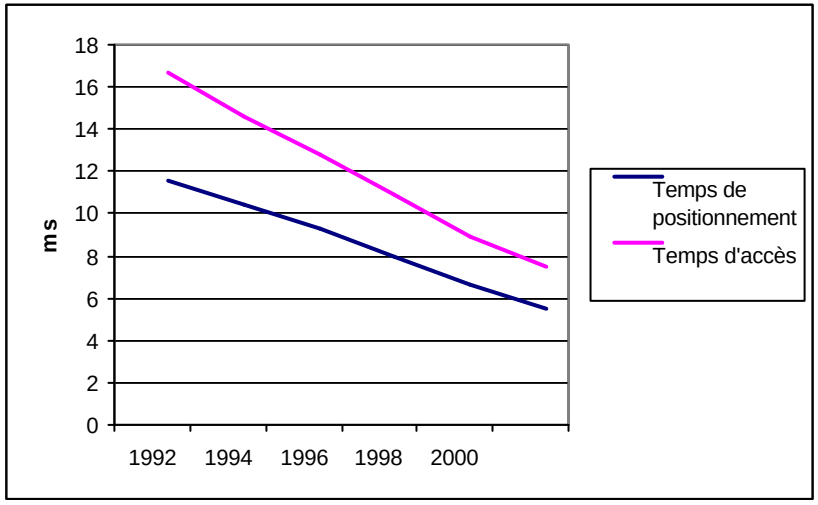
Évolution des disques magnétiques

- **Marché entraîné par le PC**
- **Forte progression des capacités**
- **Progression modérée des temps d'accès (délai rotationnel, positionnement de la tête de lecture, temps de transfert)**



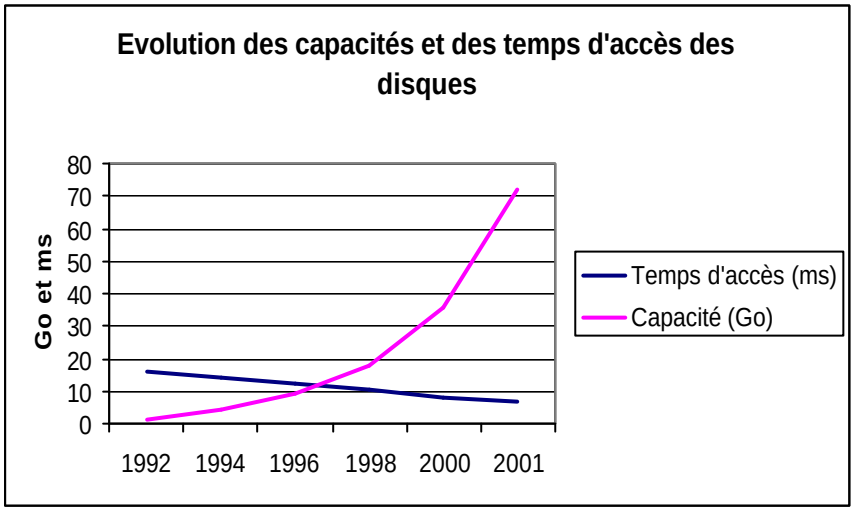


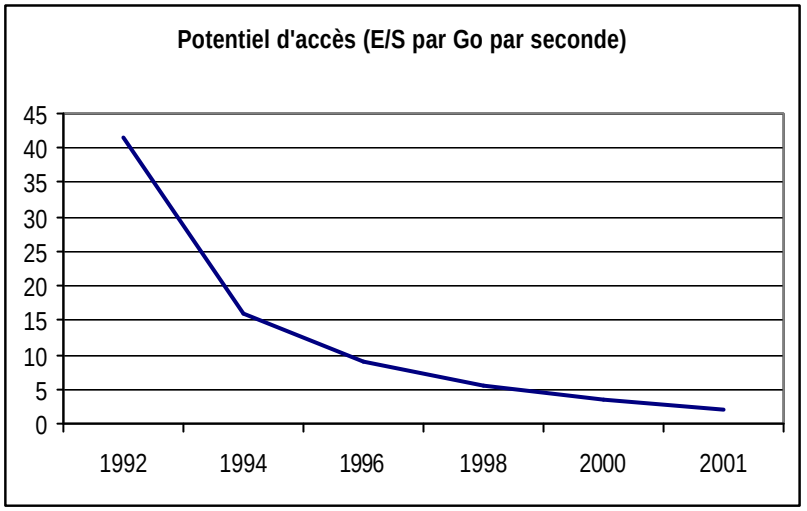




Évolution des disques magnétiques(2)

- **Évolution de la capacité et des temps d 'accès**

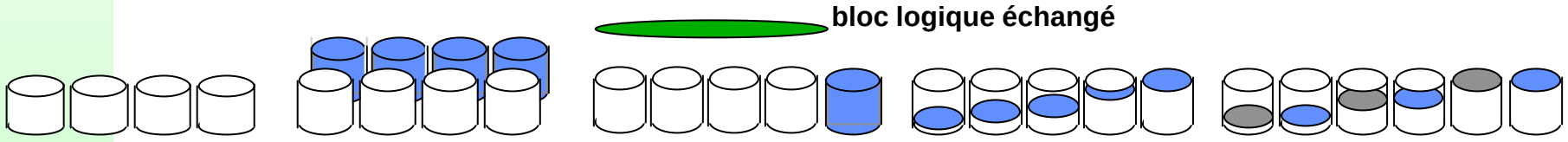




- **La forte augmentation des capacités et la (relativement) faible progression des temps d'accès entraîne que l'accès aux disques devient, de plus en plus, un facteur critique**
- **Diminution du nombre de disques à capacité constante (➔ problème de performance) :**
  - **Répartir les données sur plusieurs disques en parallèle (RAID)**
  - **Placer les données en mémoire (technique de cache)**
  - **Écriture en mémoire stable et acquittement rapide (cache sécurisé)**
- **Vers des disques intelligents? (utilisation des capacités de mémorisation et de traitement contenues dans les unités de disques)**

Technologie RAID - Tableaux de disques

- **RAID : Redundant Array of Inexpensive Disks**
- **La technologie RAID a été formalisée par des chercheurs de l'Université de Berkeley [PAT88]**
- **Principe: groupement de petits disques pour constituer un ensemble de grande capacité, à grande performance et à haute disponibilité :**
  - **Répartition des données sur plusieurs disques et transferts en parallèle**
  - **Redondance économique (utilisation de disques de parité)**
- **On présente ici les niveaux de RAID les plus fréquemment utilisés (parmi les 7 niveaux identifiés de RAID 0 à RAID 6). Le choix entre les différents niveaux de RAID dépend de l'utilisation (voir page suivante)**



RAID 0
Non Redondant
Data Stripping

RAID 1
Redondant
Miroir

RAID 3
Redondant
Disque de parité

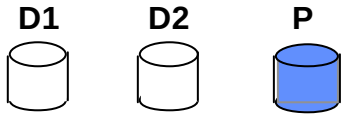
RAID 5
Redondant
Parité distribuée

RAID 6
Redondant
Parité distribuée
P+Q
(e.g. code Reed Solomon)

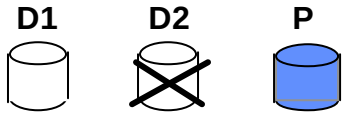
Redondance fondée sur la technique du ou exclusif (XOR)

	0	1
0	0	1
1	1	0

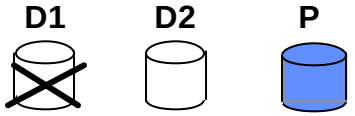
Rappel : définition XOR



P = 1101 XOR 1000 = 0101



D2 = D1 XOR P = 1101 XOR 0101 = 1000



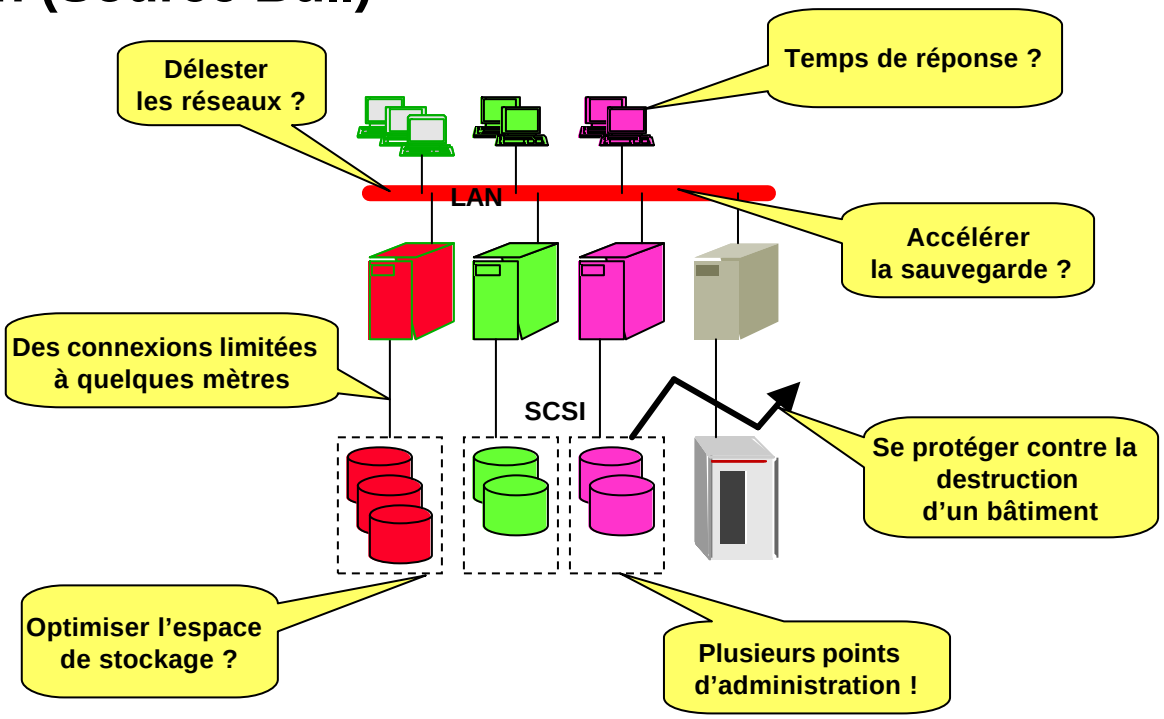
D1 = D2 XOR P = 1000 XOR 0101 = 1101

Cas d'utilisation :

- RAID 0: performance sans redondance
- RAID 1: performance et redondance coûteuse (2 x disques)
- RAID 3: redondance économique (1 disque de parité pour n disques de données) et performance pour les grands transferts de données
- RAID 5: redondance économique (1 disque de parité pour n disques de données) et performance pour les petits transferts de données
- RAID 6: mêmes caractéristiques que RAID 5 mais capacité à résister à la défaillance de deux disques.

Problématique du stockage

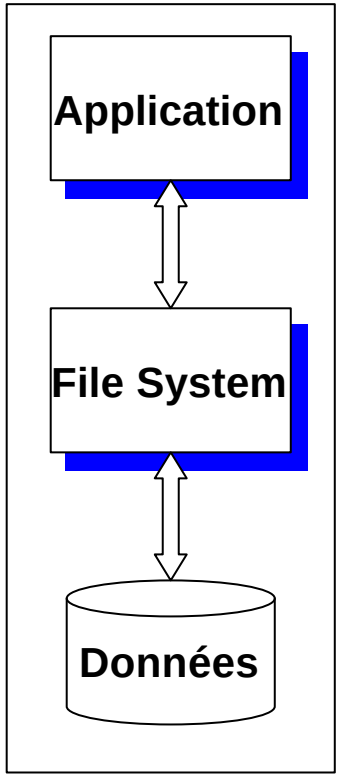
Illustration (Source Bull)



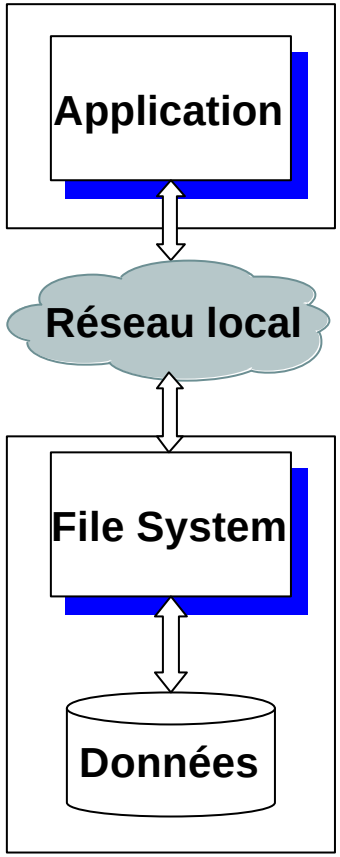
- Allègement de la charge sur le réseau local
- Optimisation des temps de réponse
- Diminution/masquage des temps de sauvegarde
- Distance entre serveur(s) et sous-système de stockage
- Protection contre les catastrophes
- Partage et optimisation des ressources de stockage
- Simplification de l 'administration

Architectures de stockage

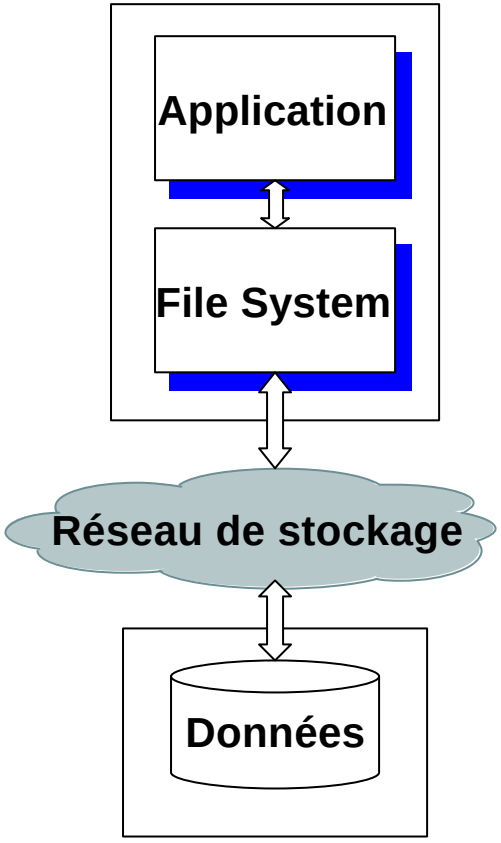
- **Plusieurs architectures en conflit :**



DAS : Direct Attached Storage

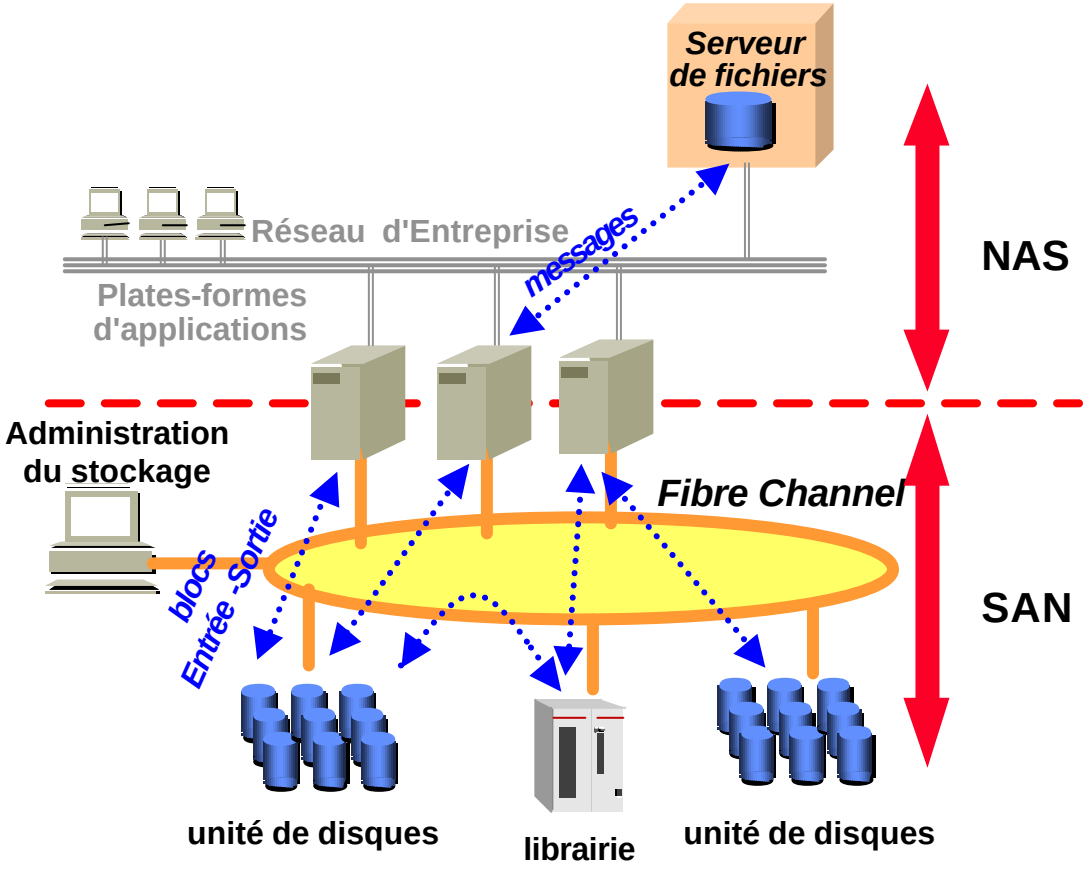


NAS : Network Attached Storage



SAN : Storage Area Network

Concepts de SAN et de NAS (Bull)



Différences entre SAN et NAS (Bull)

	NAS	SAN
Réseau	Réseau d'entreprise existant (Ethernet, FDDI, etc.)	Réseau spécialisé (Fibre Channel)
Fonction des unités de stockage	Serveur de fichiers Gestion multiprotocole	Serveur de ressources de stockage Aide à la protection, au partage et au mouvement de données
Protocole d'échange	Type message (NFS au-dessus de TCP/IP)	Type entrée-sortie (SCSI)
Bande passante	Liée au réseau existant Ethernet : 10/100/1 000 Mbits/s FDDI : 100 Mbits/s ATM : 155/622 Mbits/s	Avec Fibre Channel : $n \times 1$ Gbits/s
Administration du stockage	Le stockage est administré à travers le serveur de fichiers	Administration directe incluant l'infrastructure SAN

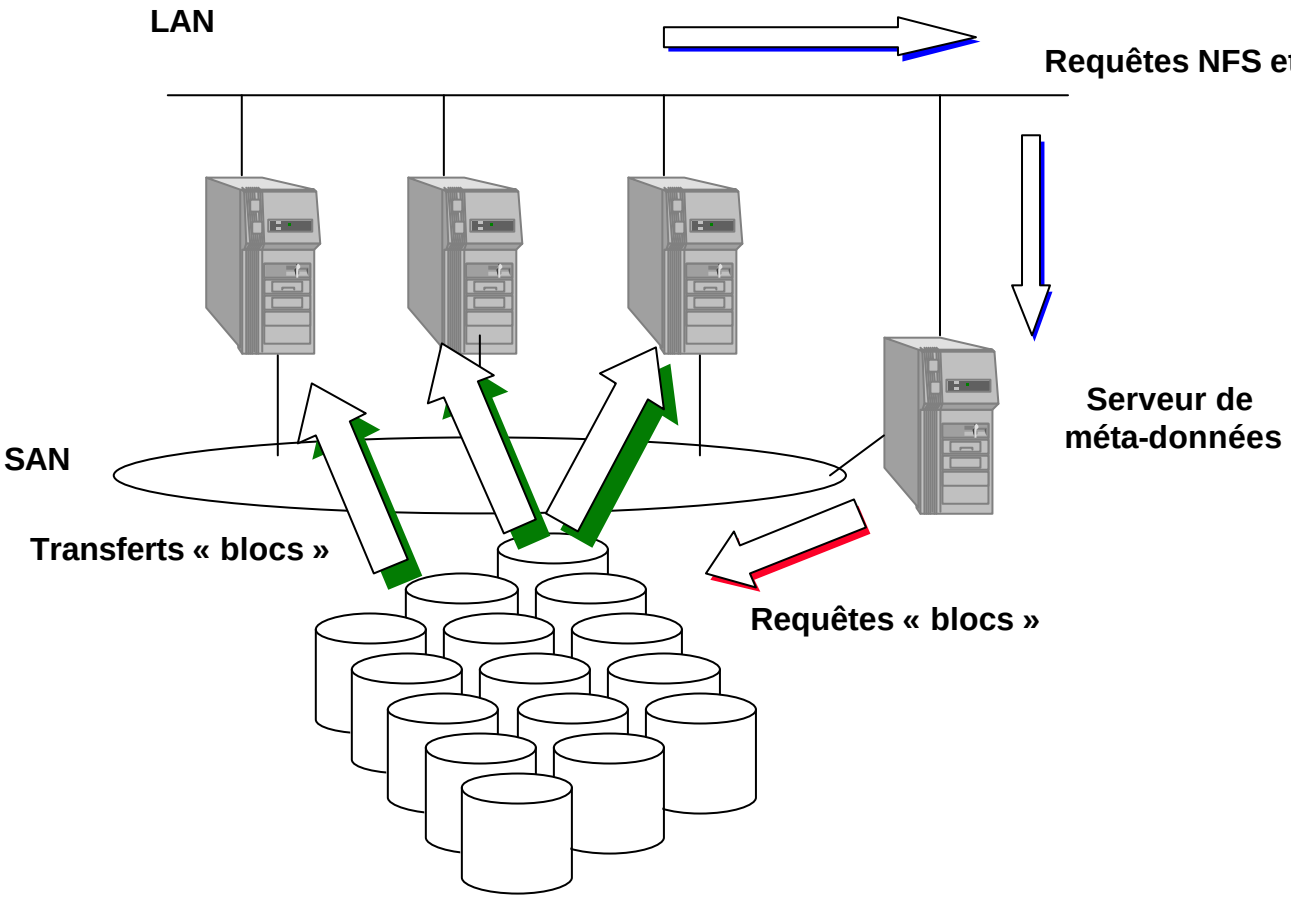
Caractérisation SAN :
- **Connexion entre tous les points**
- **Optimisation des mouvements de données**
- **Partage de ressources de stockage**
- **Partage du réseau d'interconnexion**
- **Communication égal à égal ou maître/esclave**
- **Communication par blocs de données**

Avantages du SAN :
- **Administration centralisée**
- **Partage de périphériques**
- **Souplesse d'utilisation**

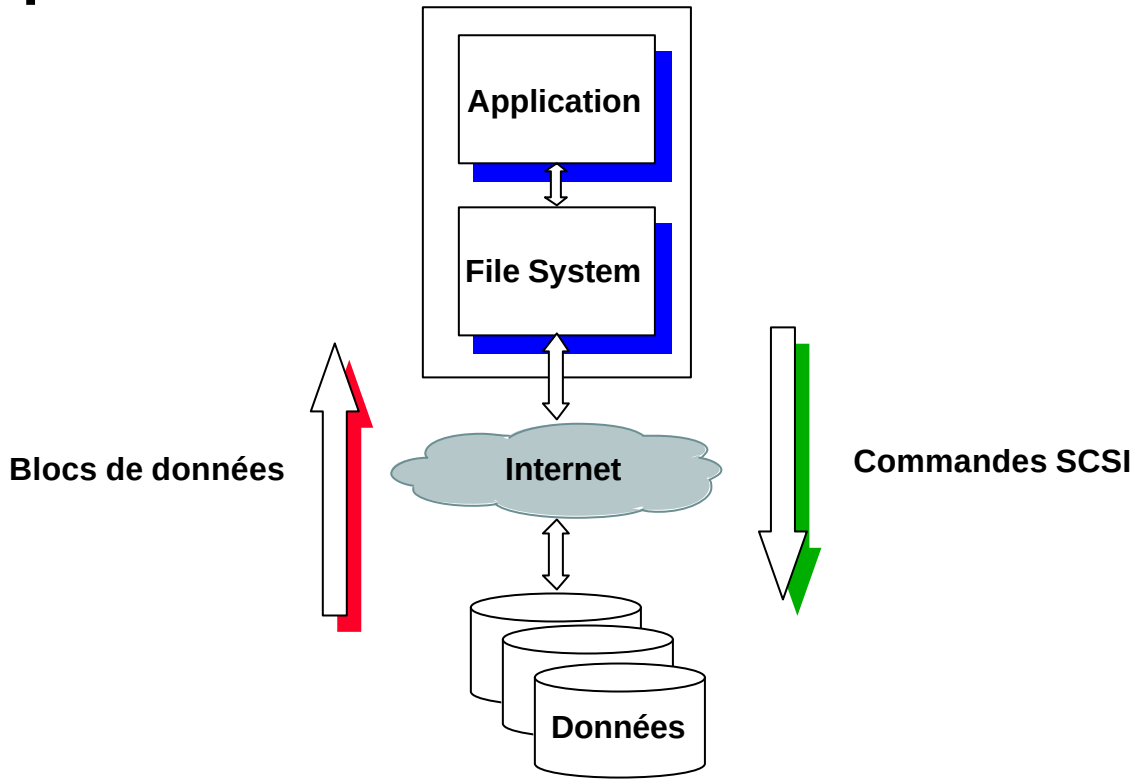
Difficulté :

- **Interopérabilité des offres de constructeurs différents**

- **SANergy Offre IBM/Tivoli : fonctionnalité NAS sur un SAN au moyen d'un serveur de méta-données**



- **Standard proposé par IBM et CISCO pour accéder à des périphériques SCSI via Internet (encapsulation de commandes SCSI dans IP). Premiers produits disponibles.**



Comparatif technologique DAS, NAS, SAN et iSCSI

	DAS	NAS	SAN	iSCSI
Type de liaison	SCSI/FC-AL/,,,	Ethernet	FC	Internet
Partage des données	Difficile	Natif	Difficle/Futur	Difficile
Sécurité	par le serveur	par le réseau	par les serveurs et le réseau	par les serveurs et le réseau
Installation	spécifique au serveur	Facile	Difficile actuellement	Difficile
Gestion	Traditionnelle	type SNMP	Difficile	Difficile
Sauvegarde	Traditionnelle	Approches variées	Server-less LAN-less	Difficile
Disaster Tolerance	Gérée par le serveur	Futur	Solutions d'égal à égal	Futur (solutions d'égal à égal)
Type d'E/S	niveau bloc	niveau fichier	niveau bloc	niveau bloc
Performance	élevée	Limitée par le réseau	élevée	Limitée par le réseau

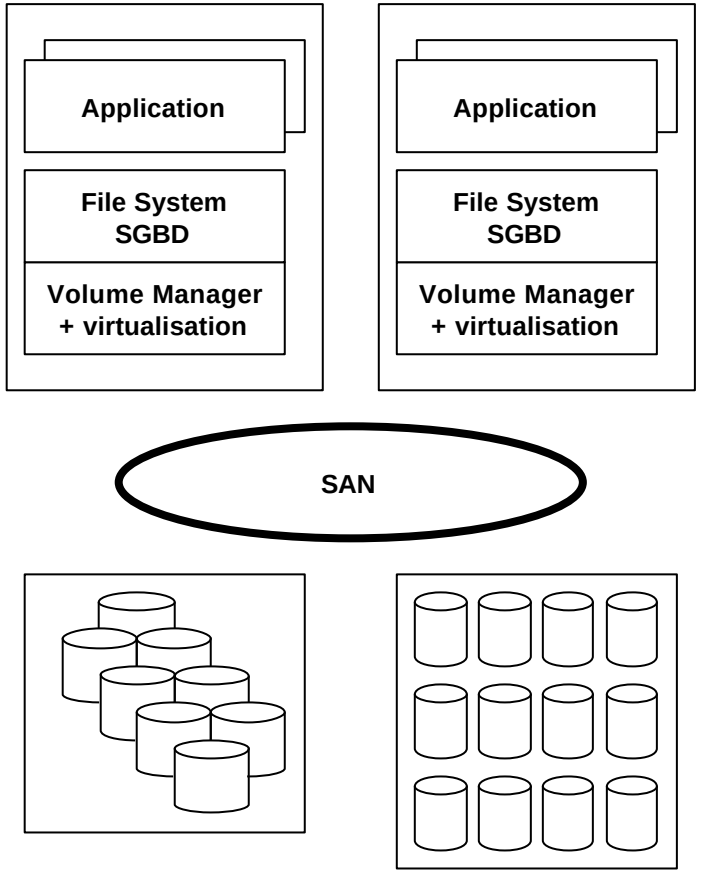
Comparatif fonctionnalité DAS-NAS, SAN et iSCSI

	DAS	NAS	SAN	iSCSI
Connexion à distance	Non	Oui	Oui	Oui
Réduction des coûts (mutualisation)	Non	Oui	Oui	Oui
Scalabilité	Limitée	Oui	Oui	Assez limitée
Séparation des investissements (stockage et traitement)	Non	Oui	Oui	Oui
Centralisation du support et de la gestion	Non	Oui	Oui	Oui
Image unique des données pour des utilisateurs ayant des systèmes différents	Non	Oui	Futur	Futur
LAN Free Backup	Non	Dépend du serveur NAS	Oui	Dépend du serveur iSCSI
Server Free Backup	Non	Dépend du serveur NAS	Oui	Dépend du serveur iSCSI
Disponibilité des données	Limitée	Dépend du serveur NAS	Oui	Dépend du serveur iSCSI

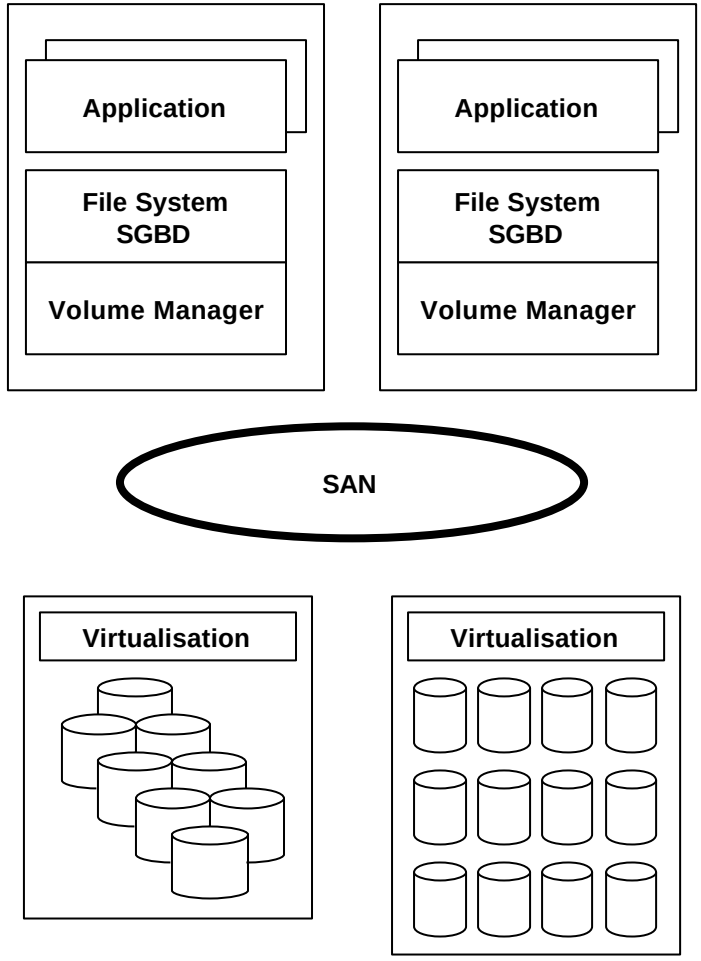
Virtualisation du stockage

- **Problèmes posés par une architecture SAN supportée par des baies de disques en RAID**
  - **Le disque est l'unité d'allocation d'espace de stockage aux serveurs : granularité trop importante**
  - **Difficulté d'accueillir, au sein d'une même configuration, des unités de stockage de différentes technologies**
- **Solution : Virtualisation du stockage**
  - **Introduction d'un niveau d'abstraction entre les unités de stockage proprement dites et les systèmes les utilisant**
- **Plusieurs modèles d'architecture en compétition en fonction de l'endroit où la virtualisation est supportée :**
  - **Systèmes clients du SAN**
  - **Sous-système de stockage**
  - **Équipement de communication du SAN**
  - **spécifique**

Modèles d'architecture

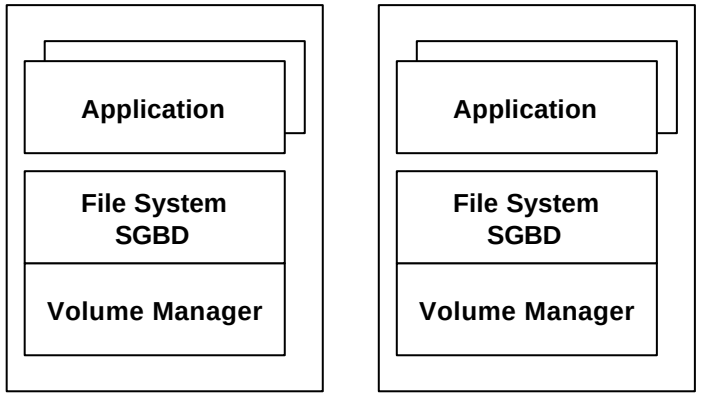


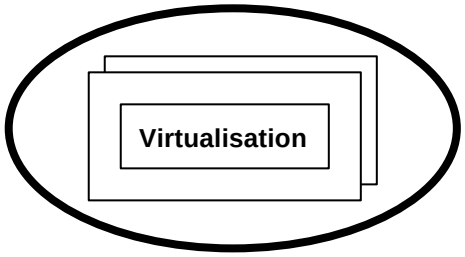
Systèmes clients du SAN

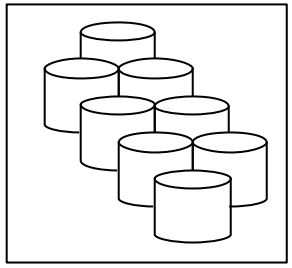


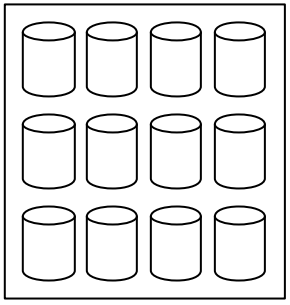
Sous-systèmes de stockage

Modèles d'architecture

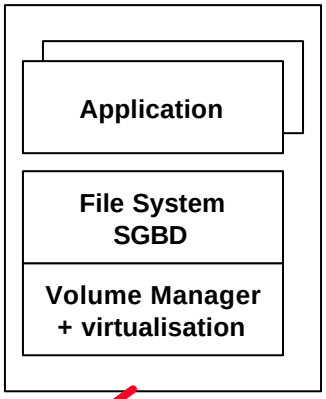


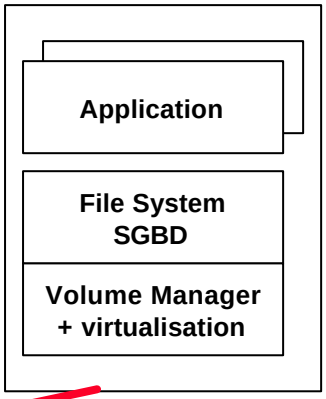


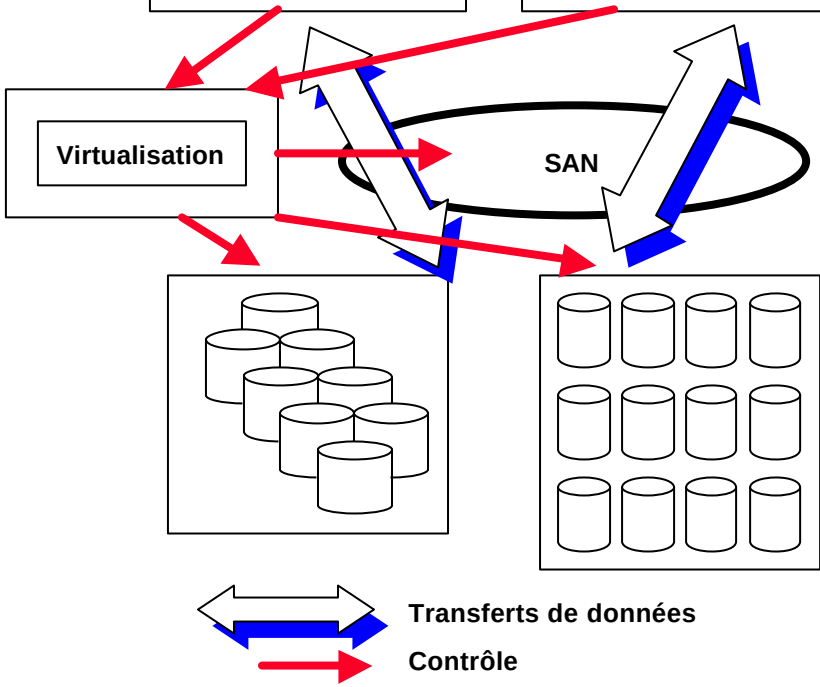




Équipements SAN







Équipements spécifique

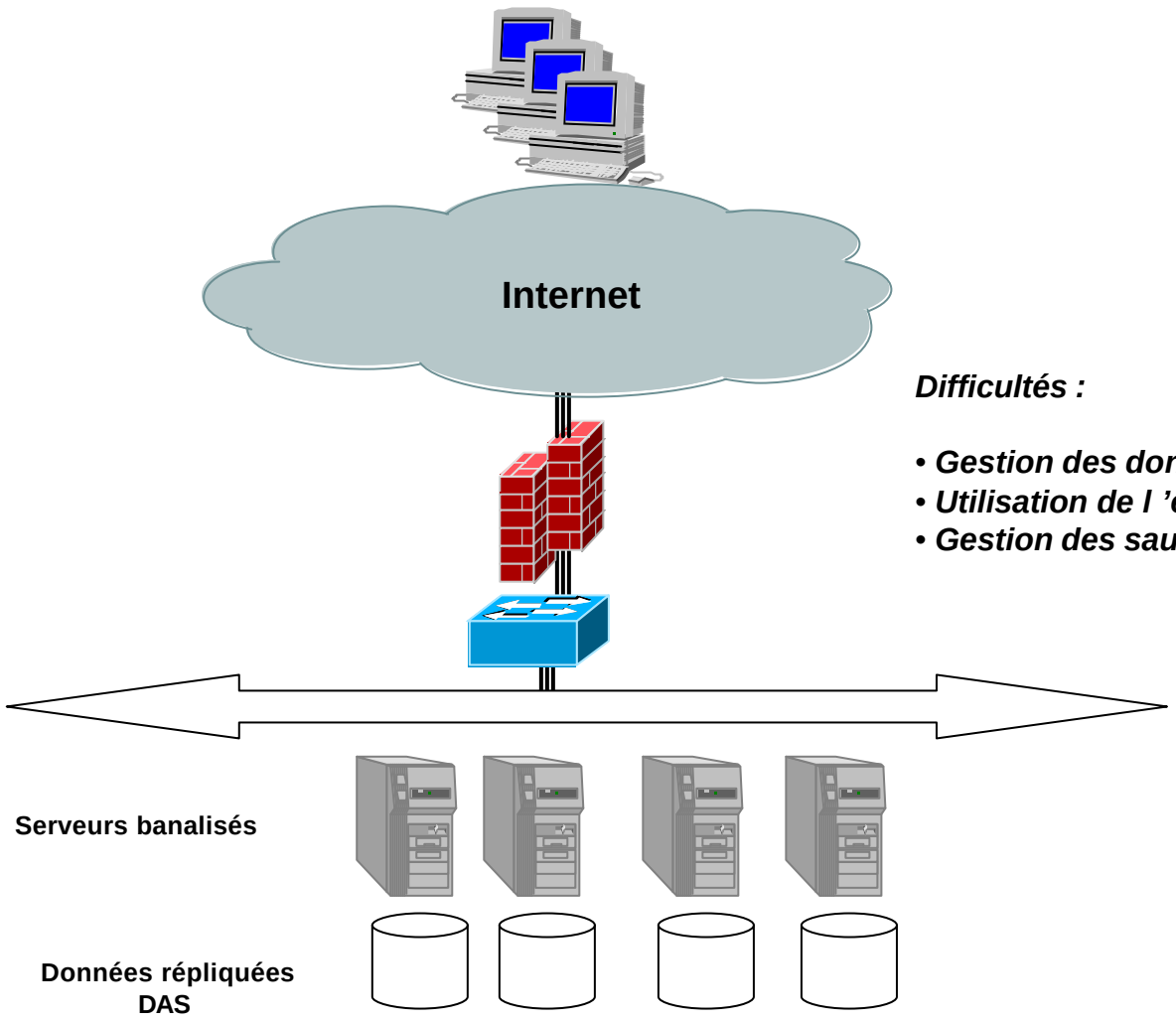
Comparaison des solutions

	Systèmes clients du SAN	Sous-sytèmes de stockage	Equipements du SAN	Equipement spécifique
Avantages	. Virtualisation fondée sur des technologies de gestionnaire de volume logique éprouvées . Intégration étroite avec les File Systems et les SGBD	. Permet de supporter l'hétérogénéité du stockage (technologie et fournisseurs)	. Possibilité de connecter des clients de natures diverses	. Contrôle centralisé . Haute performance du fait de la séparation du contrôle et des mouvements de données . Support de clients hétérogènes
Inconvénients	. La visibilité globale du stockage impose de disposer de techniques de "clusterisation" du stockage .Complexité d'administration	. Plusieurs points d'administration . Solution spécifique de chaque fournisseur . La virtualisation portant sur plusieurs sous-systèmes implique l'usage de techniques de "clustérisation" . Coût initial des différents sous-systèmes	. La visibilité globale du stockage impose de disposer de techniques de "clusterisation" du stockage . Nécessité de techniques de clusterisation pour la continuité de service . Nécessite des équipements capables de supporter les couches de virtualisation . Interopérabilité entre des équipements de différents	. Nécessite des pilotes spécifiques au niveau des clients . Qualification de la solution en environnement hétérogène . Complexité de la connectique

Inspiré par un White Paper de VERITAS « Storage Virtualization »

Exemple comparatif DAS - NAS pour un site Web

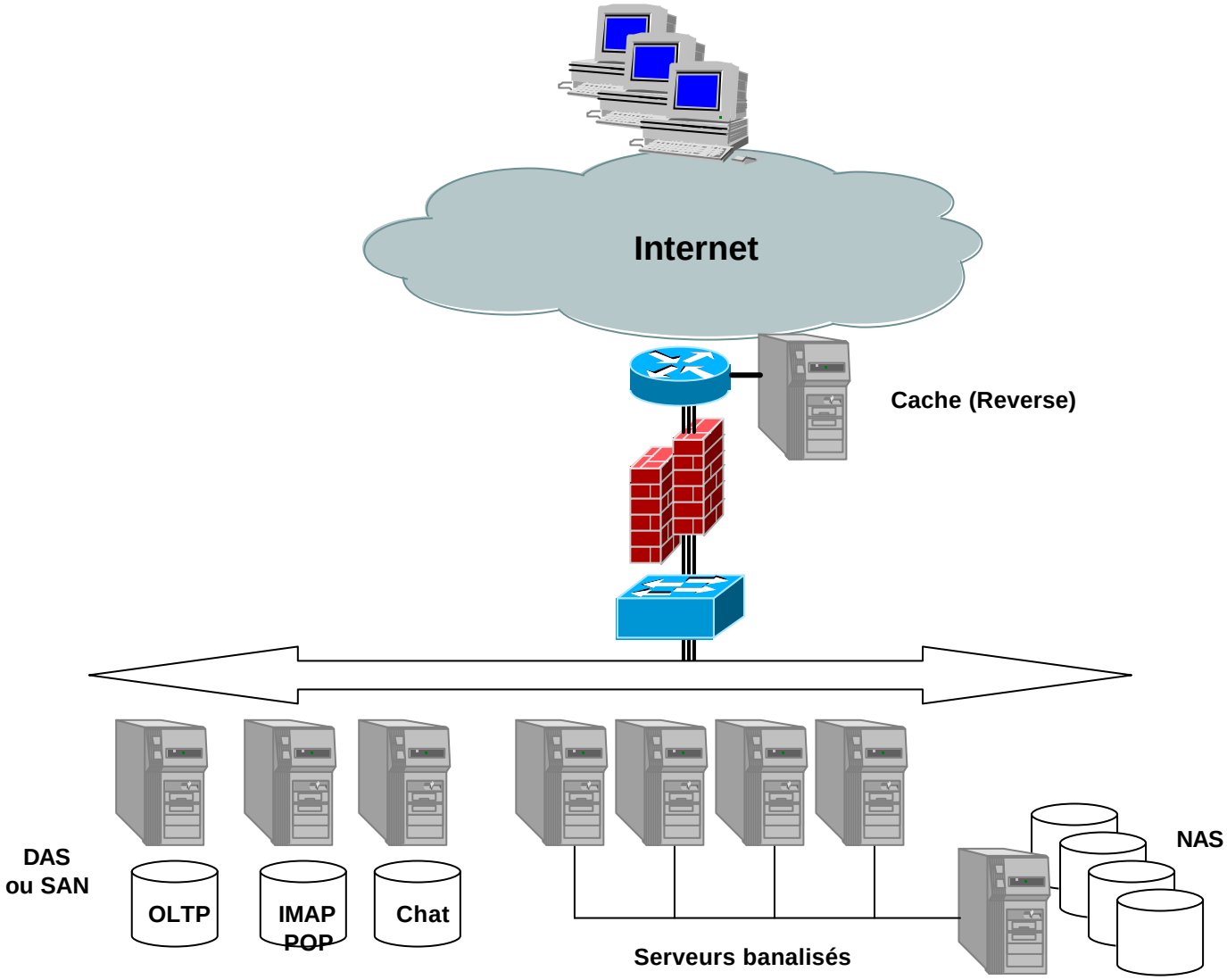
Modèle « plat »



Difficultés :

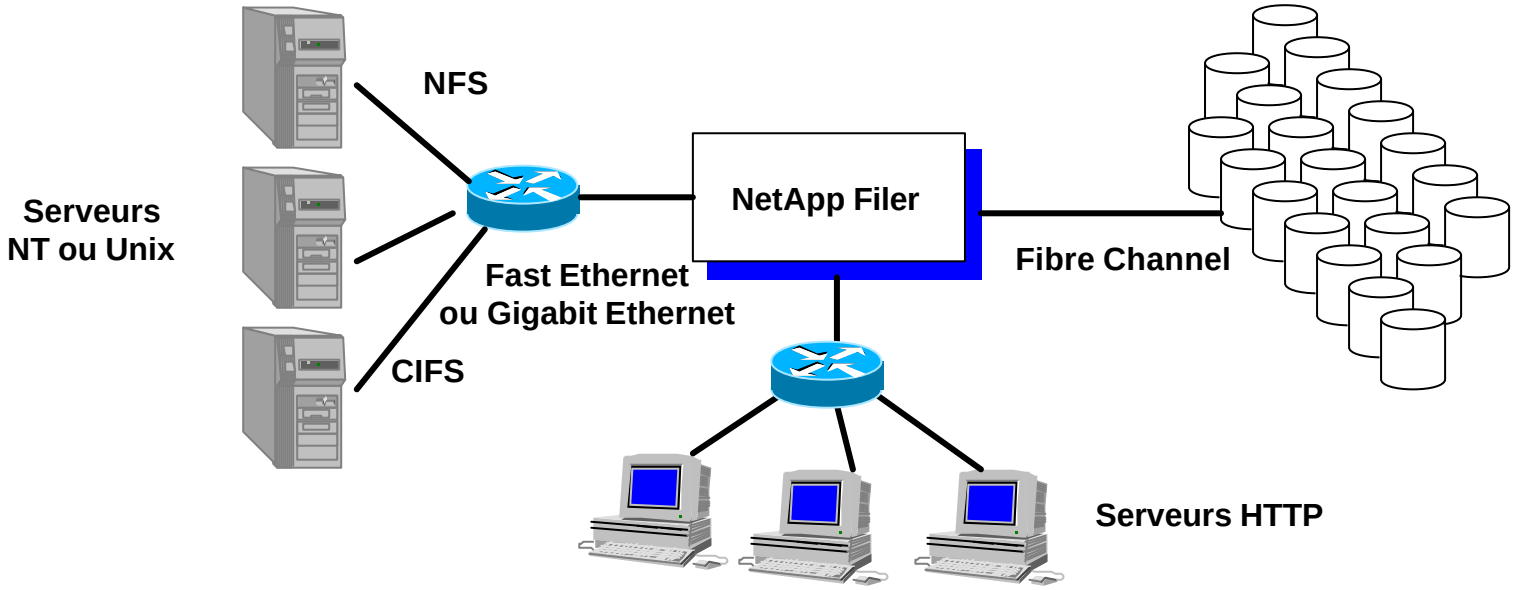
- ***Gestion des données répliquées***
- ***Utilisation de l 'espace disque***
- ***Gestion des sauvegardes***

Modèle hiérarchisé



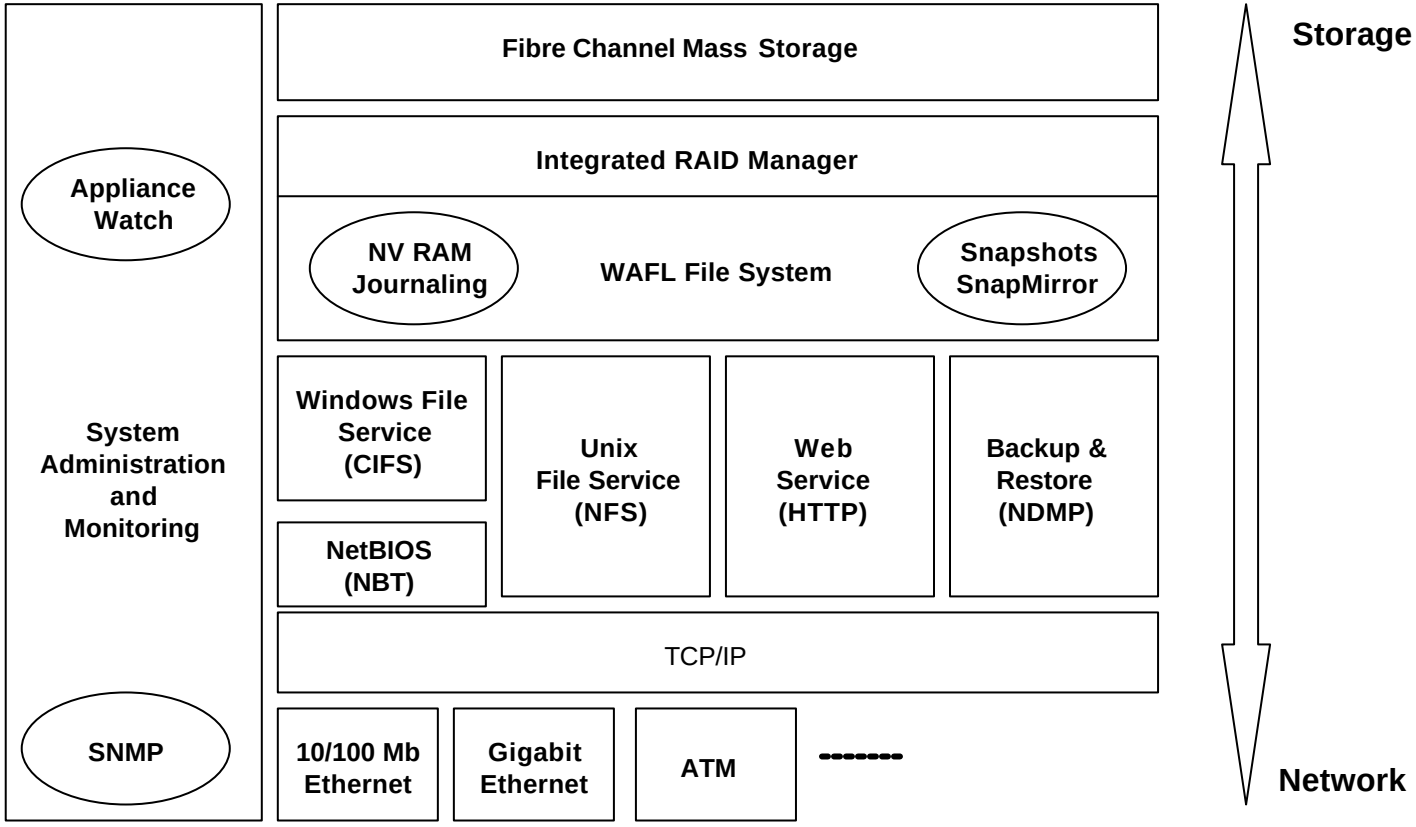
Quelques exemples de produits de stockage

NetApp Filer



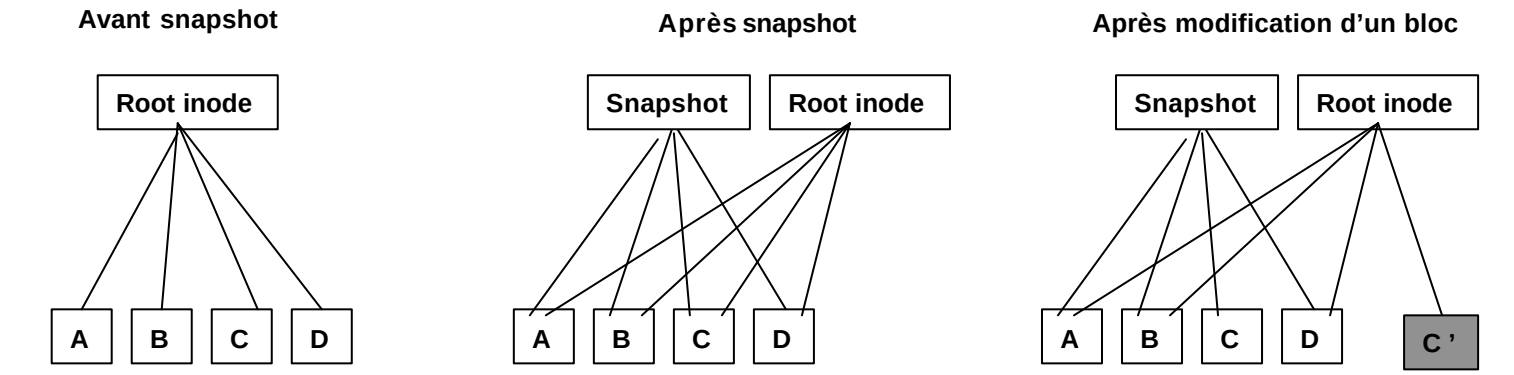
- Architecture matérielle classique fondée sur un processeur Alpha

Architecture du logiciel



WAFL = Write Anywhere File Layout (optimisé pour l'écriture)
CIFS = Common Internet File System

Création de Snapshots

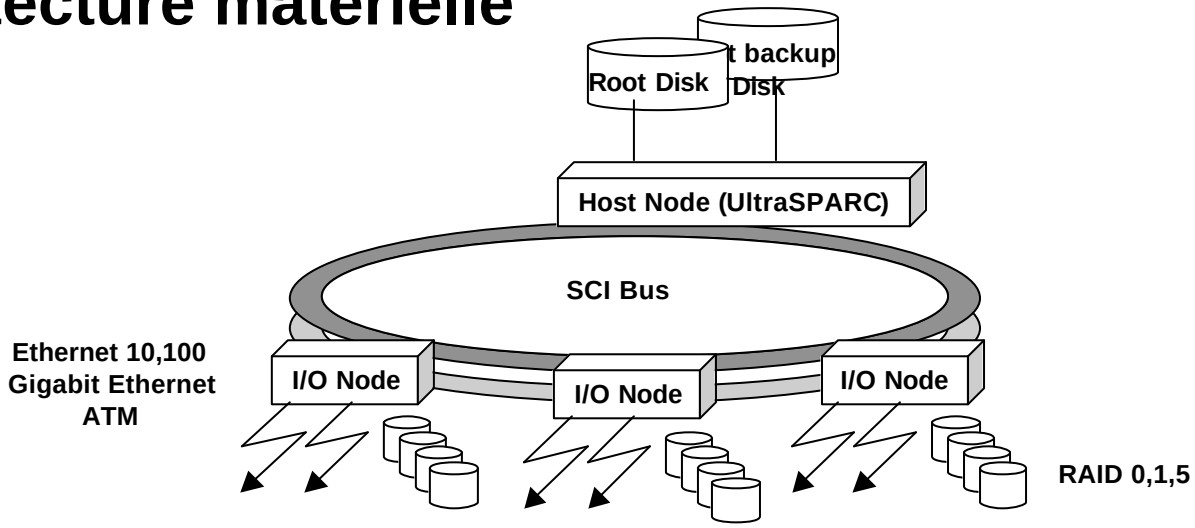


Blocs de données sur disque

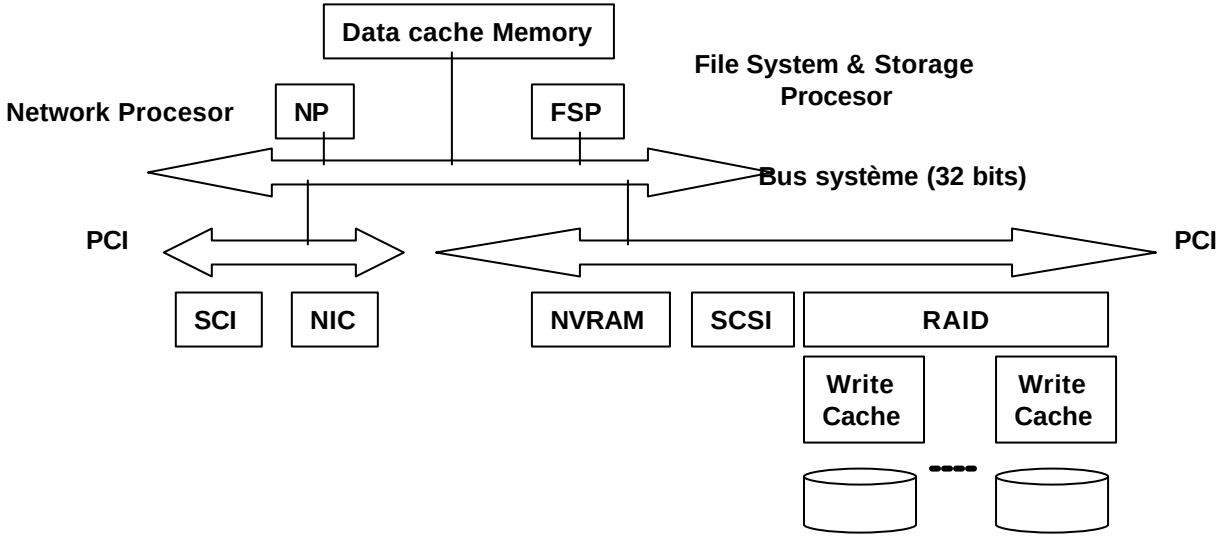
Bloc modifié

Auspex

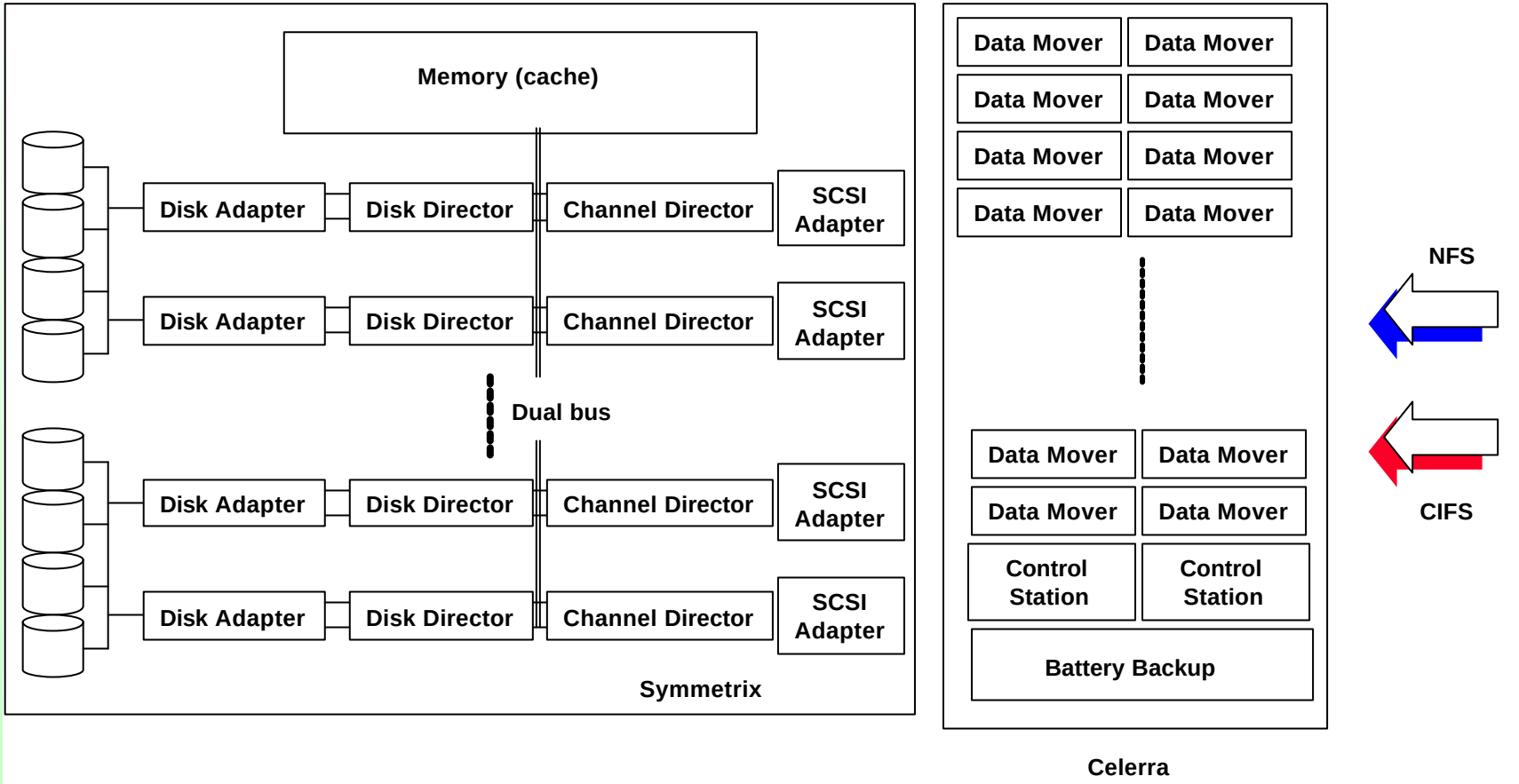
Architecture matérielle



Architecture I/O Node



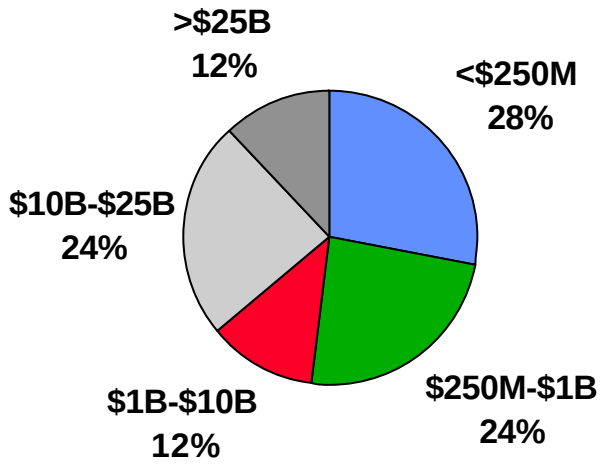
Celerra File Server/Symmetrix



Exemple de TCO du stockage

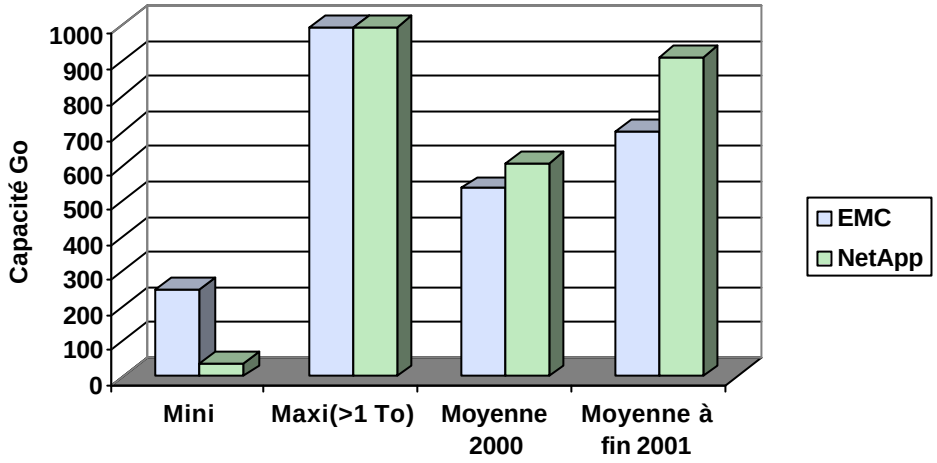
Contexte de l'étude

- **Extrait d'une étude publiée par INPUT en 2001 (www.input.com) menée auprès de 25 sociétés utilisant les produits :**
  - **Oracle + Network Appliance Filer (~50%)**
  - **Oracle + EMC Symmetrix (~50%)**
- **Revenus des sociétés :**

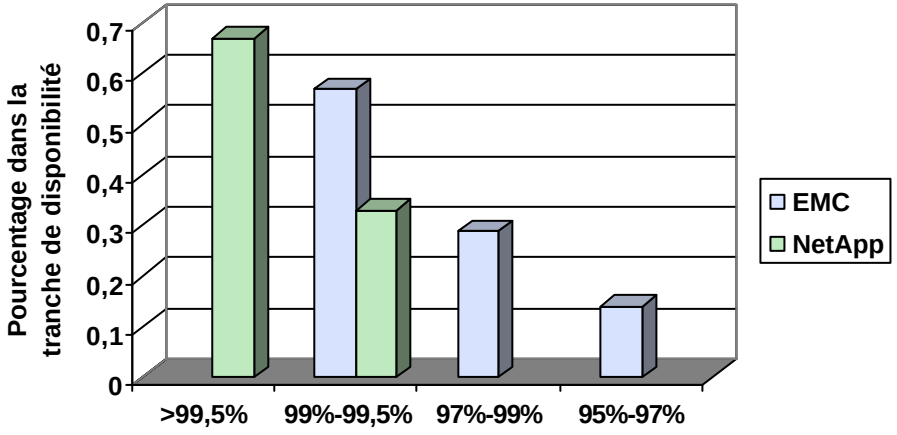


Résultats

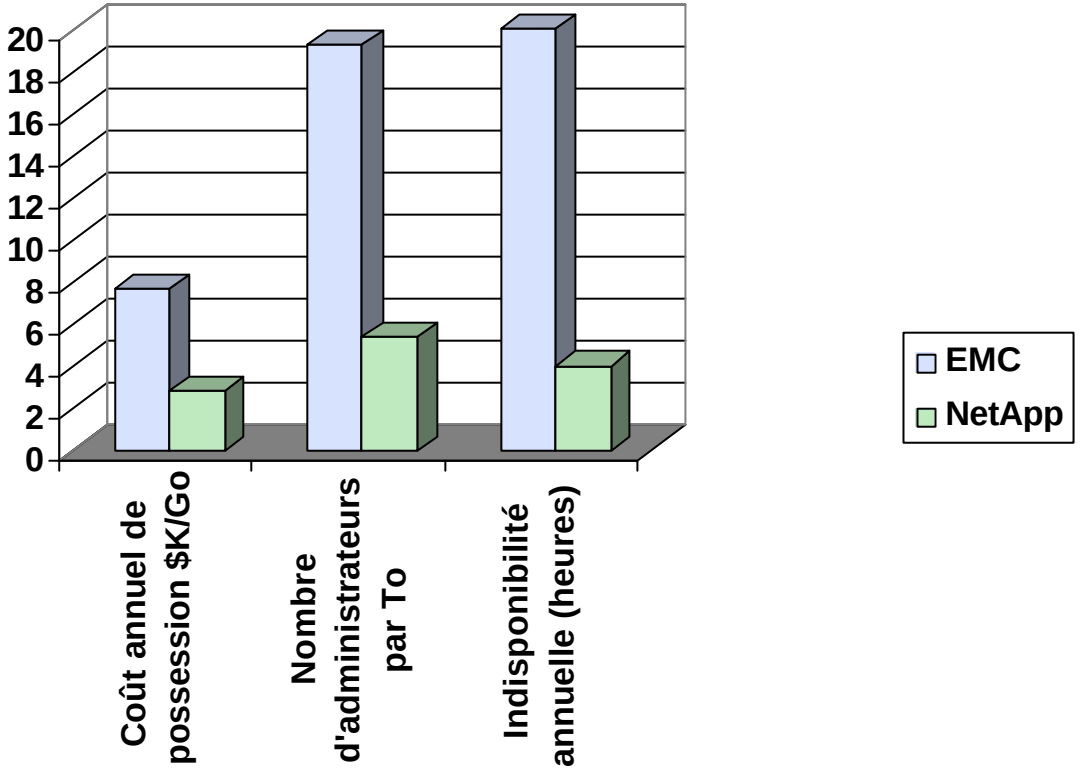
- **Taille des bases de données (en Go) - Analyse 2000 et à fin 2001**



- **Disponibilité des données**

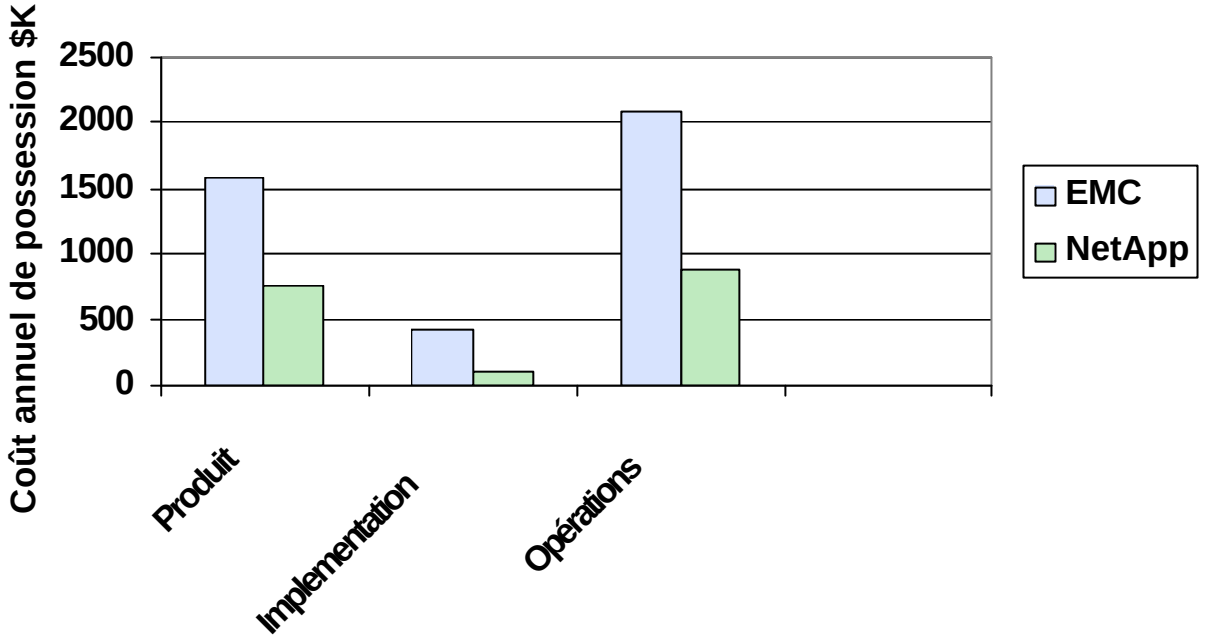


TCO, Administrateurs et indisponibilité



Résultats(3)

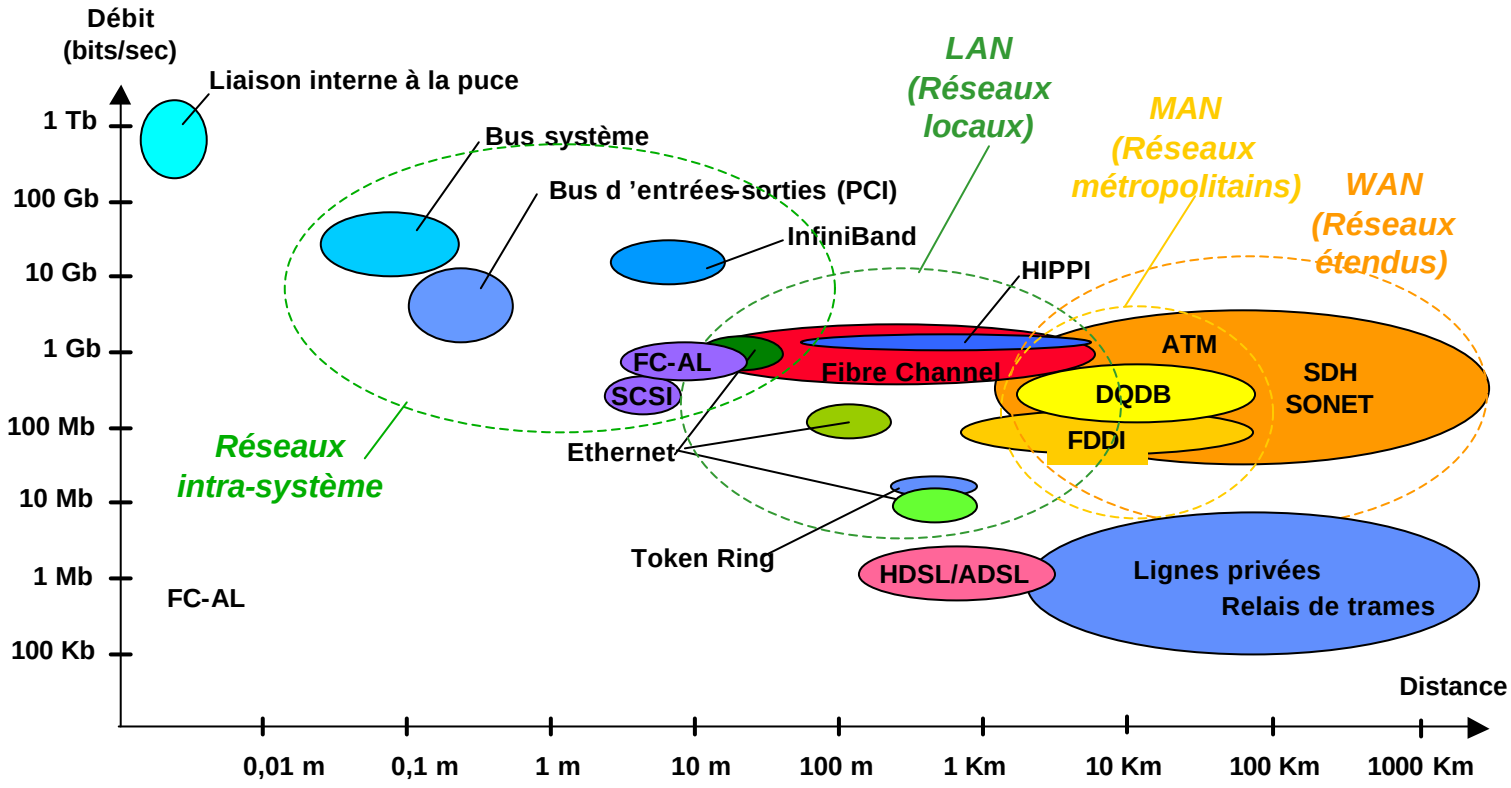
- **Éléments du TCO**



- **Leçons :**
  - **La facilité de mise en œuvre et d'exploitation est un critère clé**
  - **Se méfier des entrants!**

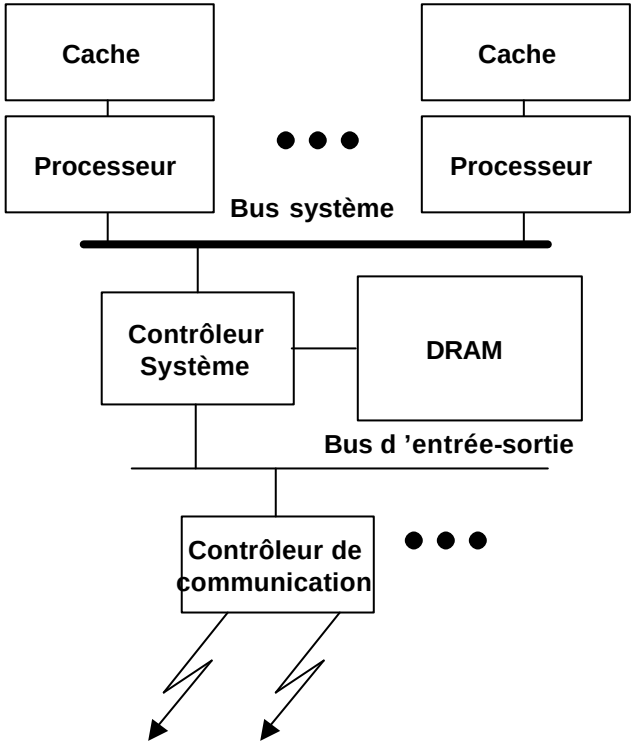
Communications

Technologies de communication

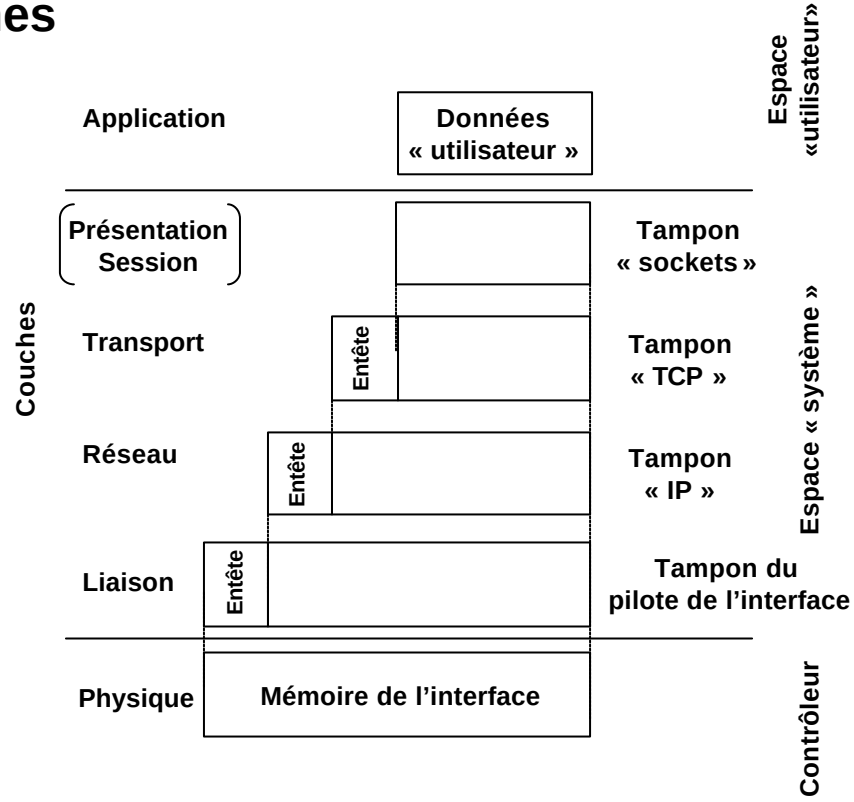


Serveurs : Support des communications

- **Problèmes posés par le support des communications au niveau des serveurs :**
  - **Nombre élevé d ’interruptions**
  - **Copie de zones mémoire**
  - **Interaction avec les caches**

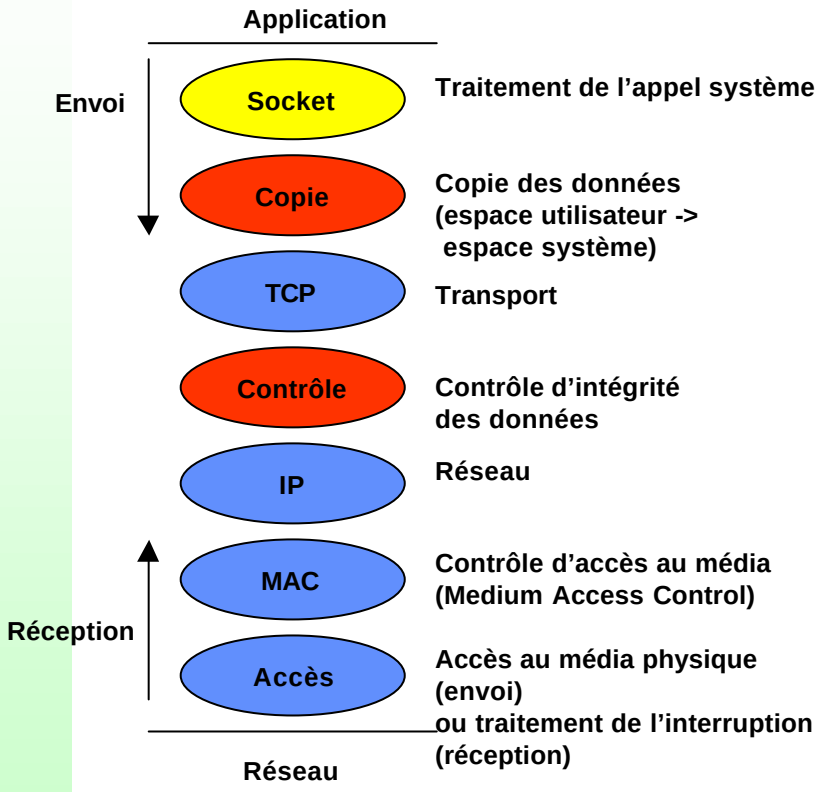


Architecture «classique»



Structuration en couches et élaboration des données

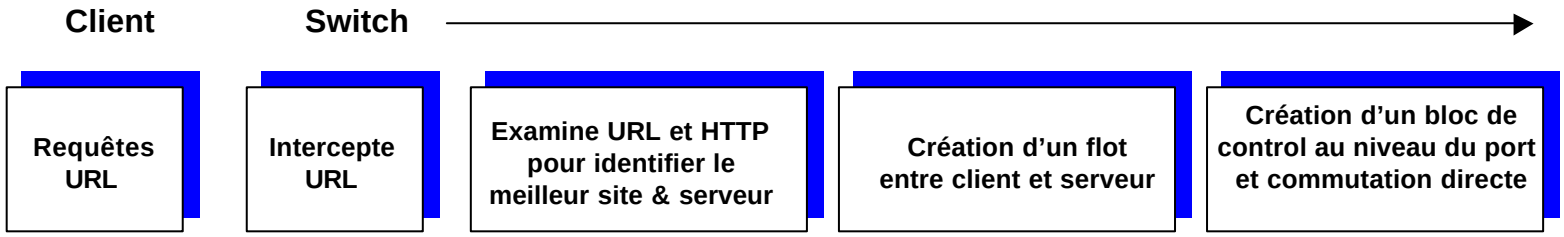
Support des communications(2)



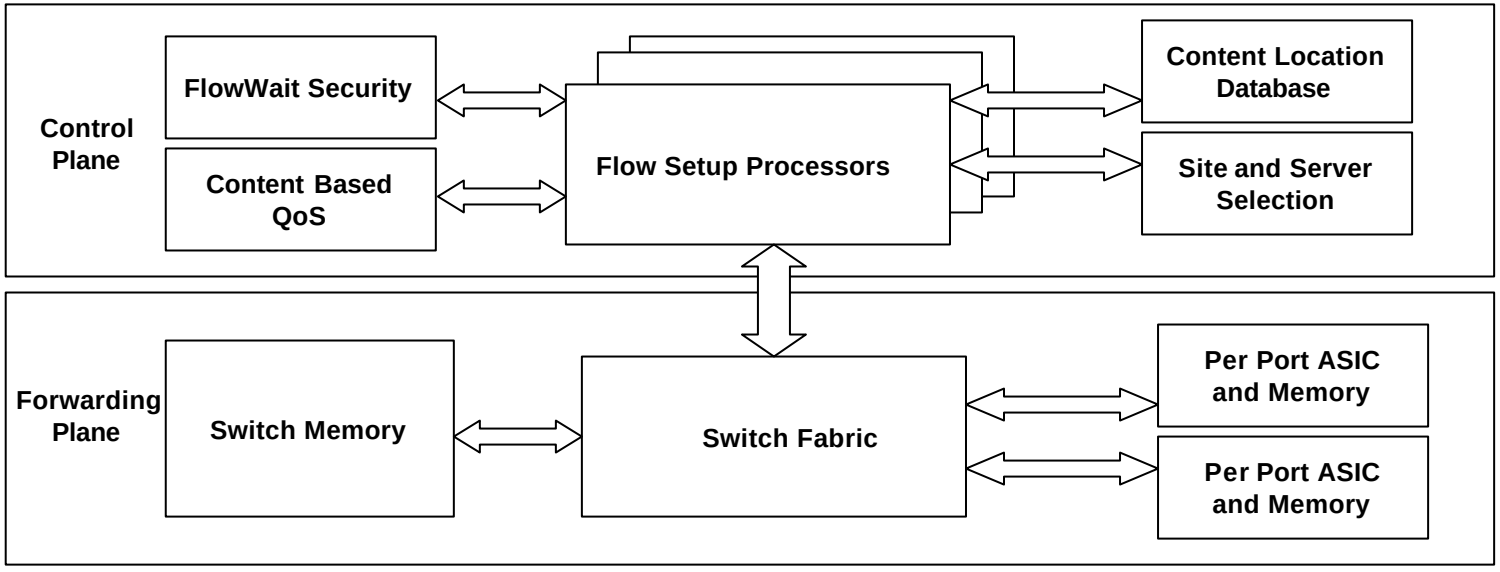
- **Optimisation du support des communications au niveau des serveurs :**
  - **Réduction du nombre d'interruptions**
    - **DMA (Dynamic Memory Access) optimisé : deux listes de descripteurs (entrées, sorties) et information d'état**
  - **Diminution des mouvements de données**
    - **DMA avec liste de descripteurs chaînés**
    - **Circuit spécialisé pour le calcul de la «checksum»**
  - **Consistance de l 'information**
    - **les points précédents tendent à diminuer les manipulations d 'information par les processeurs et donc les interactions des avec les caches**

Nouvelle génération de commutateurs

- **Web Switching (dit niveau 5 TCP/IP ou niveau 7 ISO) ou Wire Speed Switching**
  - **Exemple : ArrowPoint (racheté par CISCO)**
    - **Logique de fonctionnement**

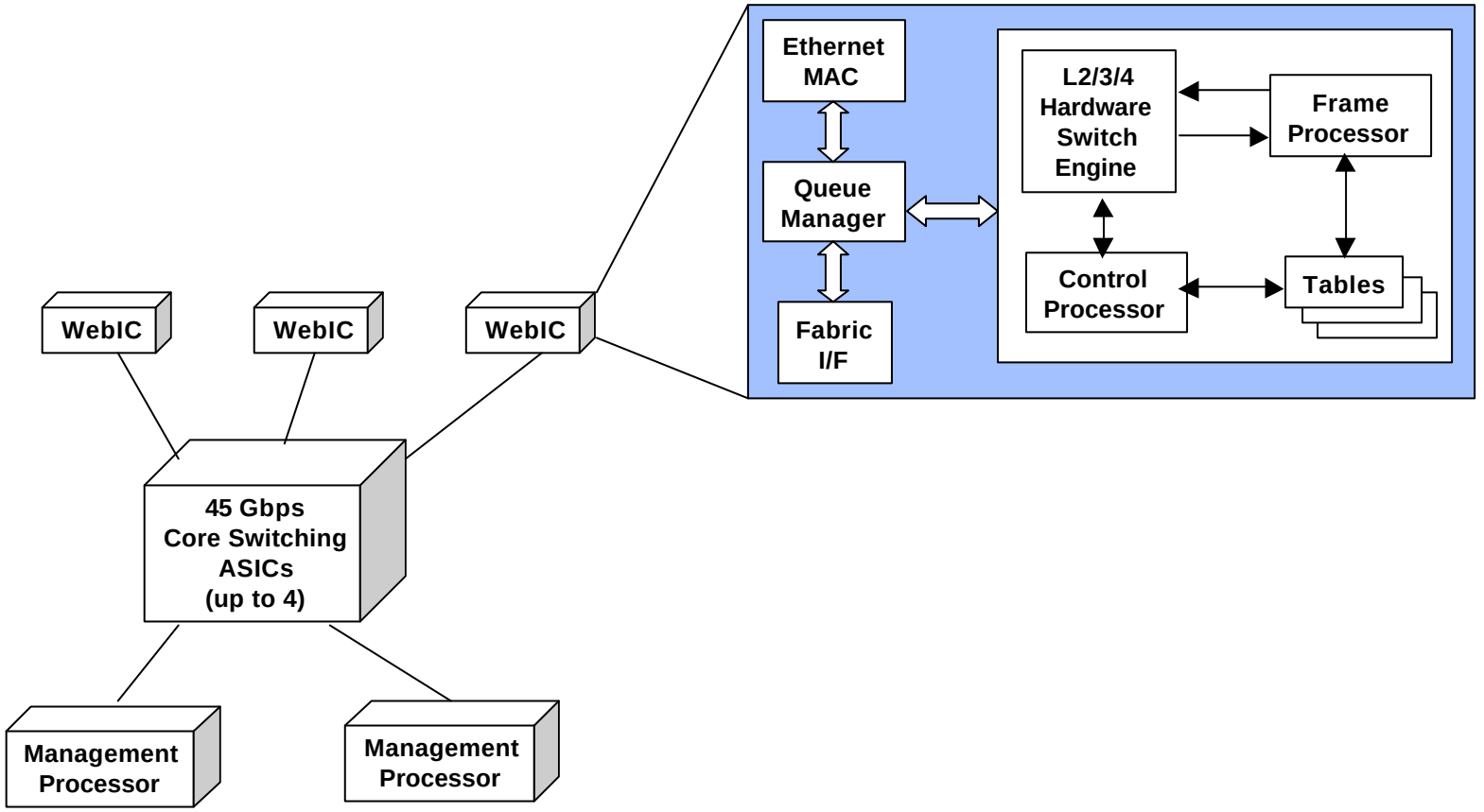


    - **Architecture logique du Web Switch**



Nouvelle génération de commutateurs(2)

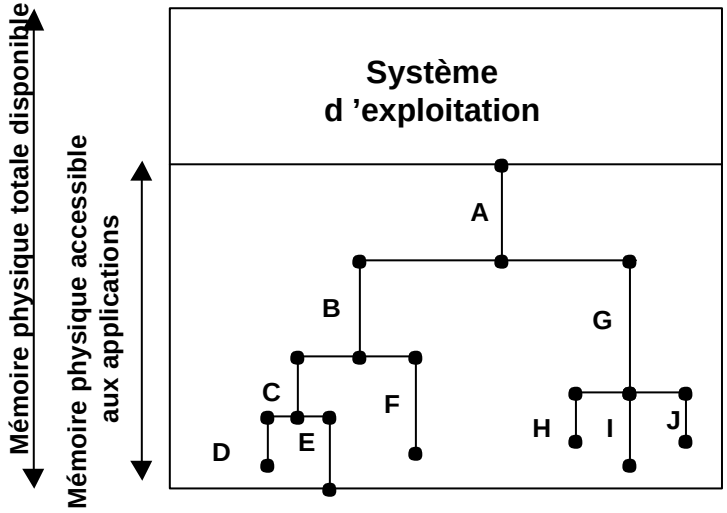
- **Autre exemple : Alteon (racheté par Nortel)**
  - **Architecture matérielle**



Évolution des technologies Logiciel

Mémoire virtuelle

- **Permet d 'affranchir la programmation de la gestion de ressources mémoire limitées**
  - **Illustration dans le cas des programmes, la situation est identique avec les données (gestion de tampons)**

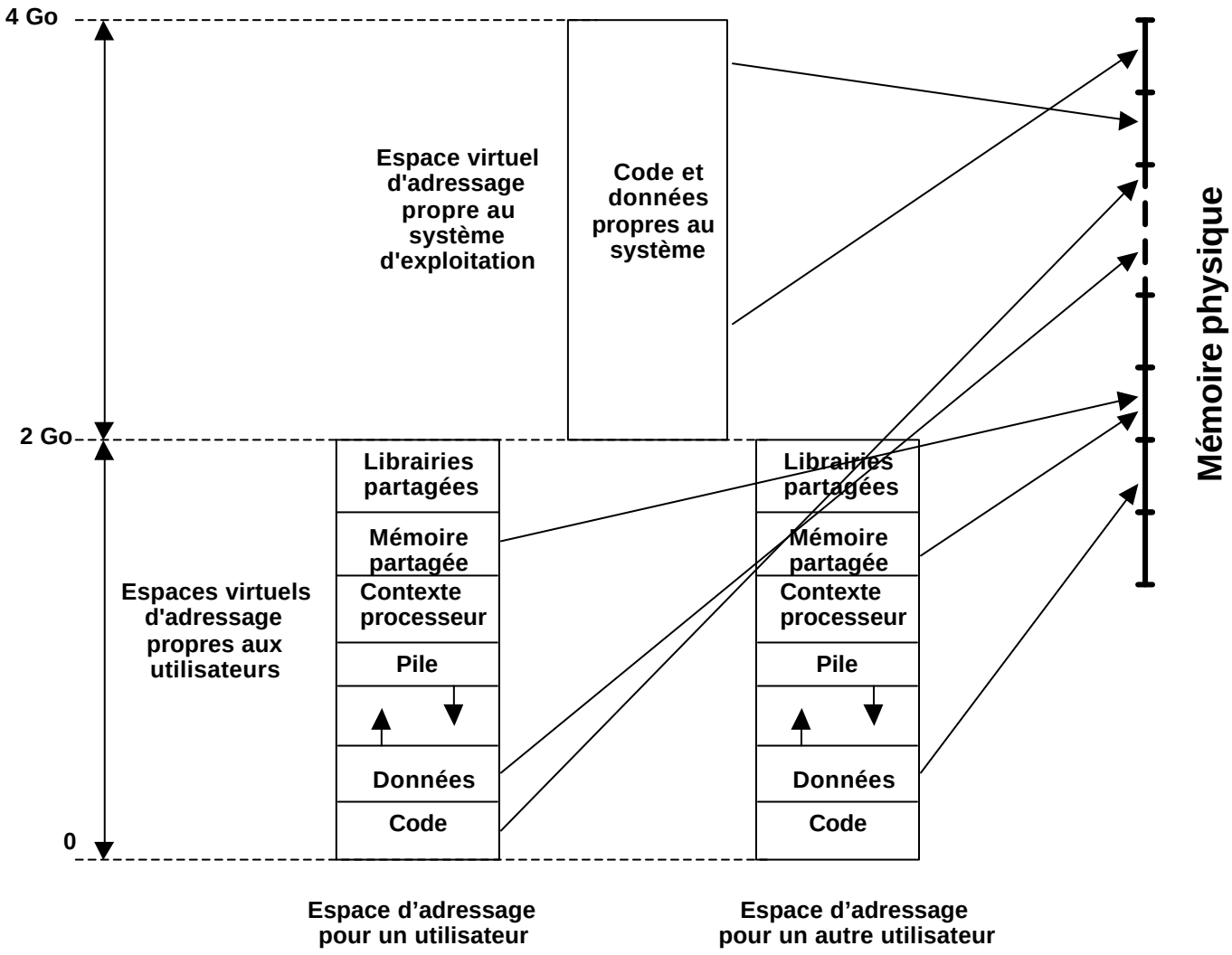


Structure de recouvrement «Overlay»

- **Concept de mémoire virtuelle**
  - **Les espaces virtuels sont réalisés sur la mémoire physique sur la base de pages (par exemple 8 Ko)**
  - **Les pages virtuelles peuvent ne pas avoir de correspondance en mémoire physique, il y a alors défaut de page et c'est le système d'exploitation qui se charge d 'amener la page en mémoire physique (depuis le disque)**

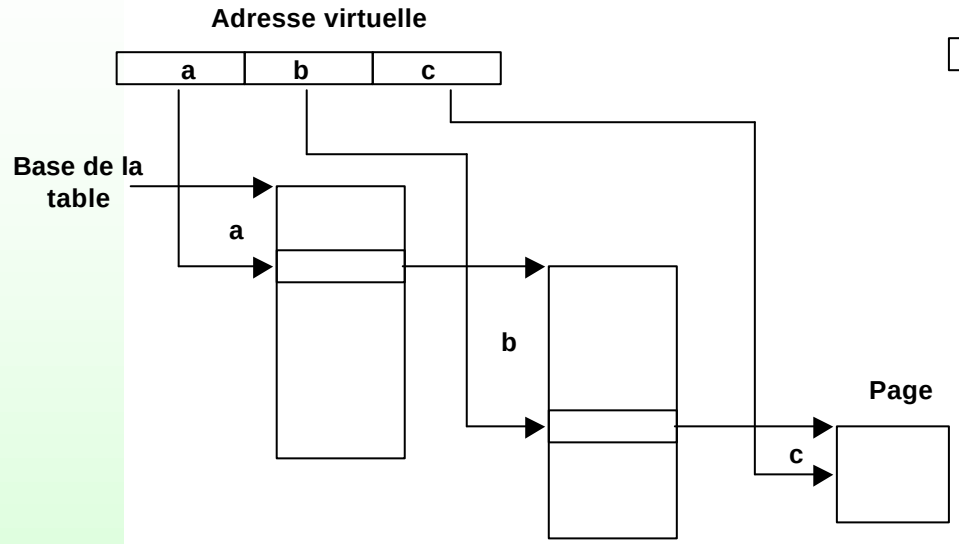
Mémoire virtuelle(2)

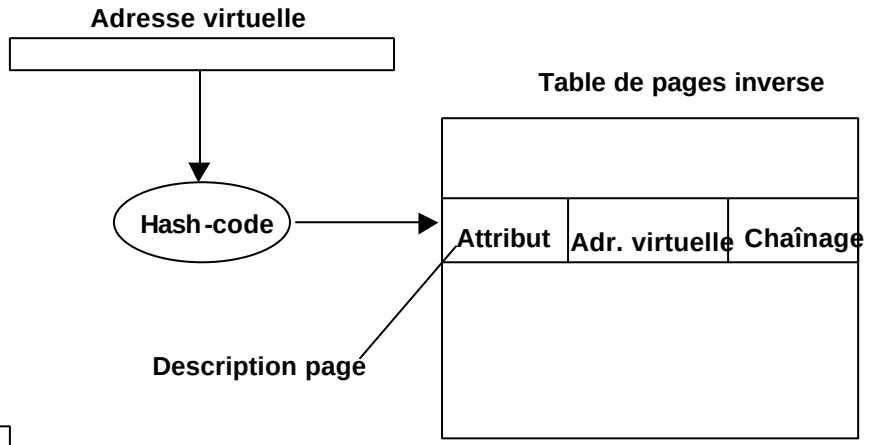
- **Illustration du concept de mémoire virtuelle (Unix, NT)**



Mémoire virtuelle(3)

- **Traduction des adresses virtuelles en adresses réelles**





1) - Schéma classique par tables de pages **2) - Schéma par tables de pages inverse**

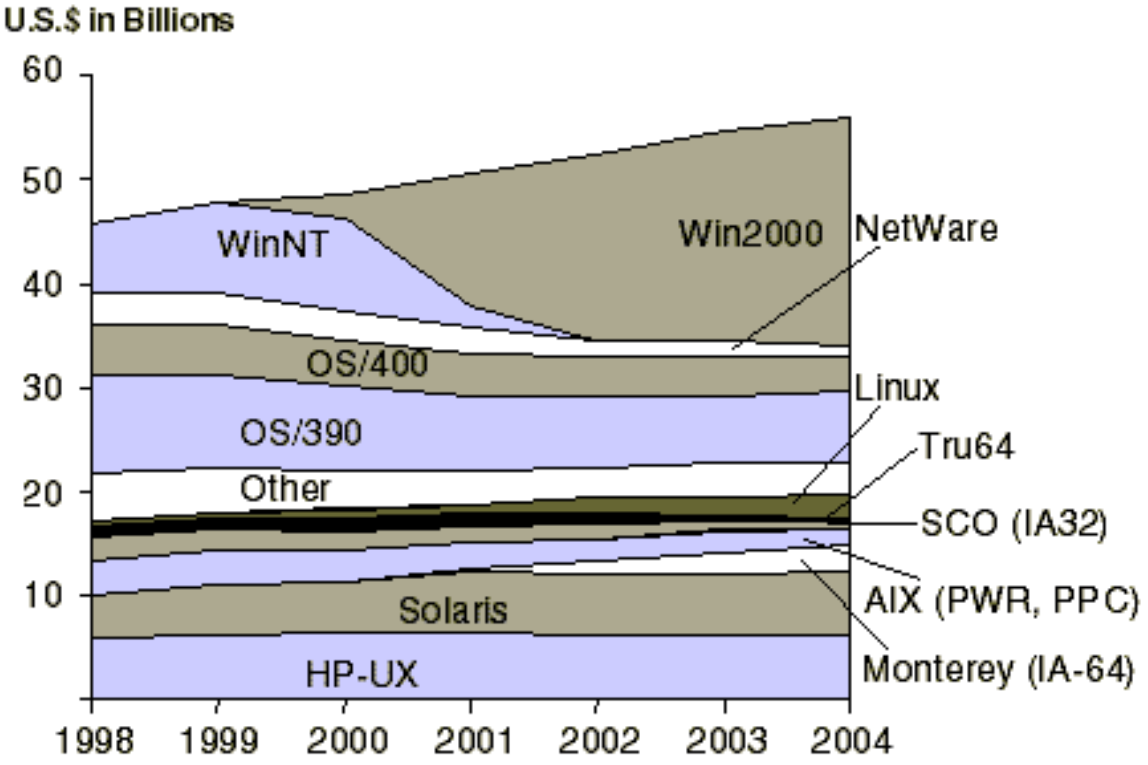
- **La traduction est assurée par les microprocesseurs au moyen de mécanismes de cache (TLB Translation Lookaside Buffer)**
- **La gestion du cache de traduction peut être assurée par le matériel (efficacité) ou, cas de plus en plus fréquent, par le logiciel (souplesse)**

Architecture 64 bits

- Supportée par les microprocesseurs RISC et IA-64 (Itanium)
- Support par les systèmes d ’exploitation :
  - **UNIX : disponible**
  - **Windows 2000 : support progressif**
- Avantages de l'architecture 64 bits
  - **Support de grands objets (fichiers) directement en mémoire ➡ Performances**
    - **Traduction des adresses de fichiers par le matériel**
    - **Mouvements de données via le mécanisme de pagination à la demande (demand paging)**
    - **Suppression du "multiplexage" des adresses par logiciel**
  - **Support de systèmes de fichiers >2 GB**
  - **Gestion des grandes mémoires physiques**
  - **Exigence des SGBD et des logiciels de CAO**

Systèmes d'exploitation

Parts de marché des systèmes d'exploitation pour les serveurs (Gartner Group Octobre 1999)



Commentaires :
- **Le phénomène de consolidation des serveurs (up-sizing) permet aux systèmes propriétaires de maintenir leurs positions**
- **En raison des coûts de développement et de maintenance, concentration probable autour de quelques versions d'Unix, phénomène accentué par l'apparition d'IA-64**
- **Percée importante de Windows 2000 en bas et milieu de gamme**
- **Percée de Linux en particulier pour des serveurs dédiés**

Note : Ces chiffres ne couvrent que la vente des serveurs dans une configuration minimale (sans les sous-systèmes périphériques, disques notamment) et avec le système d'exploitation mais sans les autres logiciels.

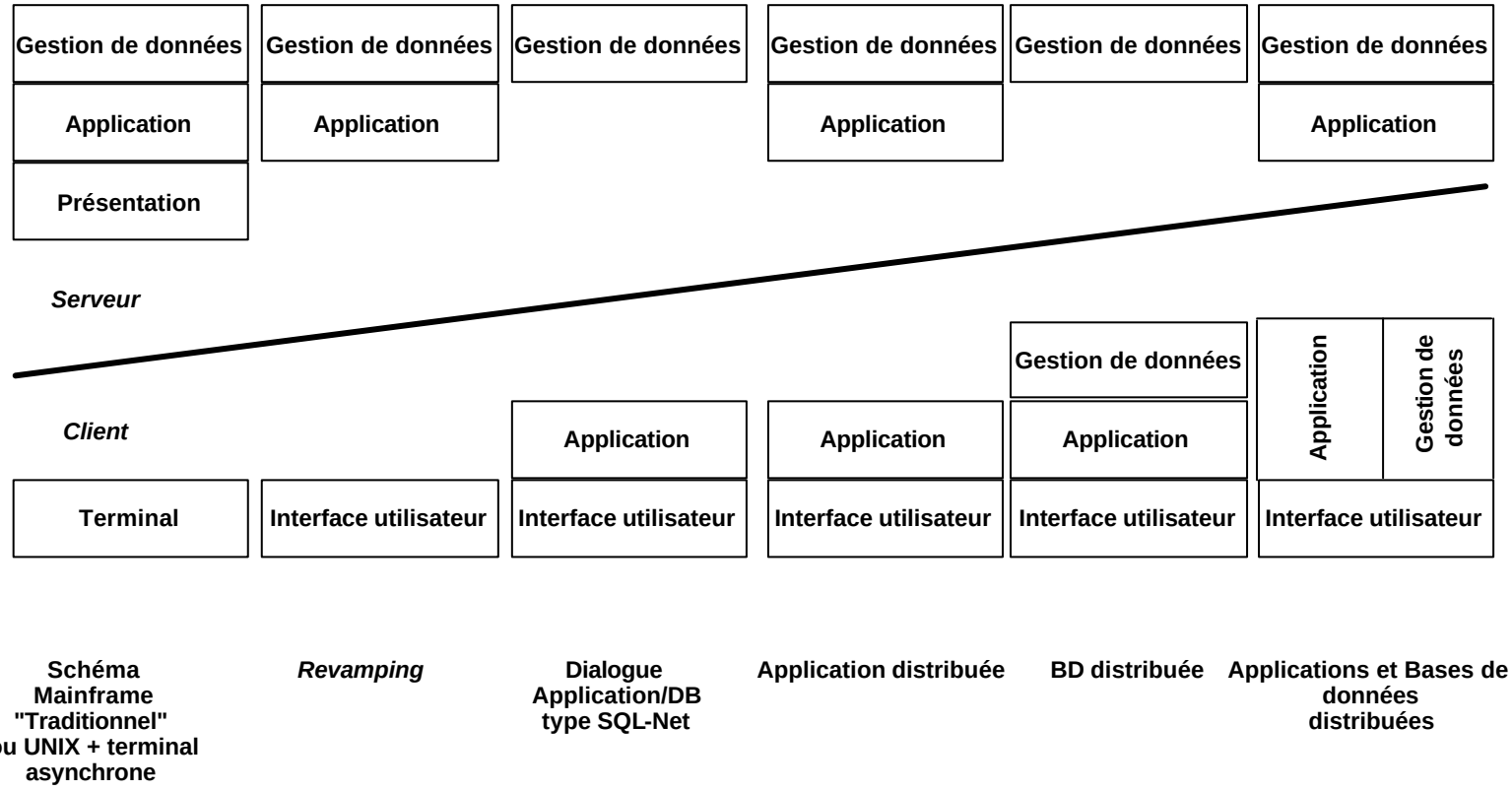
Fonctionnalité des systèmes d'exploitation

- **Aspects techniques**
  - **Scalabilité**
  - **Fiabilité, disponibilité et aptitude à être maintenu en état de bon fonctionnement**
    - **Masquage des défaillances du matériel**
    - **Possibilité de reconfiguration**
    - **Support de la mise à jour «en ligne» du matériel et du logiciel**
    - **Point de reprise et redémarrage**
    - **Support du partitionnement et des configurations de type cluster**
  - **Système de fichiers**
    - **Système de fichiers «journalisé»**
    - **Support des grands fichiers**
    - **Sauvegarde/Restauration**
  - **Support d'Internet**
    - **TCP/IP et IPv6 (128 bits d'adresse)**
    - **Extension, outils et services, Navigateurs**
    - **Messagerie, Commerce électronique, ...**

Fonctionnalité des systèmes d'exploitation(2)

- **Aspects techniques (suite)**
  - **Gestion du système (System Management)**
    - **Gestion de la configuration matérielle**
    - **Gestion de la configuration logicielle**
    - **Gestion des utilisateurs**
    - **Gestion des ressources**
    - **Possibilité d 'administration à distance**
    - **Analyse des performances**
    - **Optimisation des travaux du type traitement par lot**
  - **Services distribués**
    - **Service de désignation (annuaires - Directory Services)**
    - **Sécurité**
    - **Système de fichiers distribué**
    - **Service d'appel de procédure à distance**
    - **Service transactionnel**
  - **Support des clients de type PC et services associés**

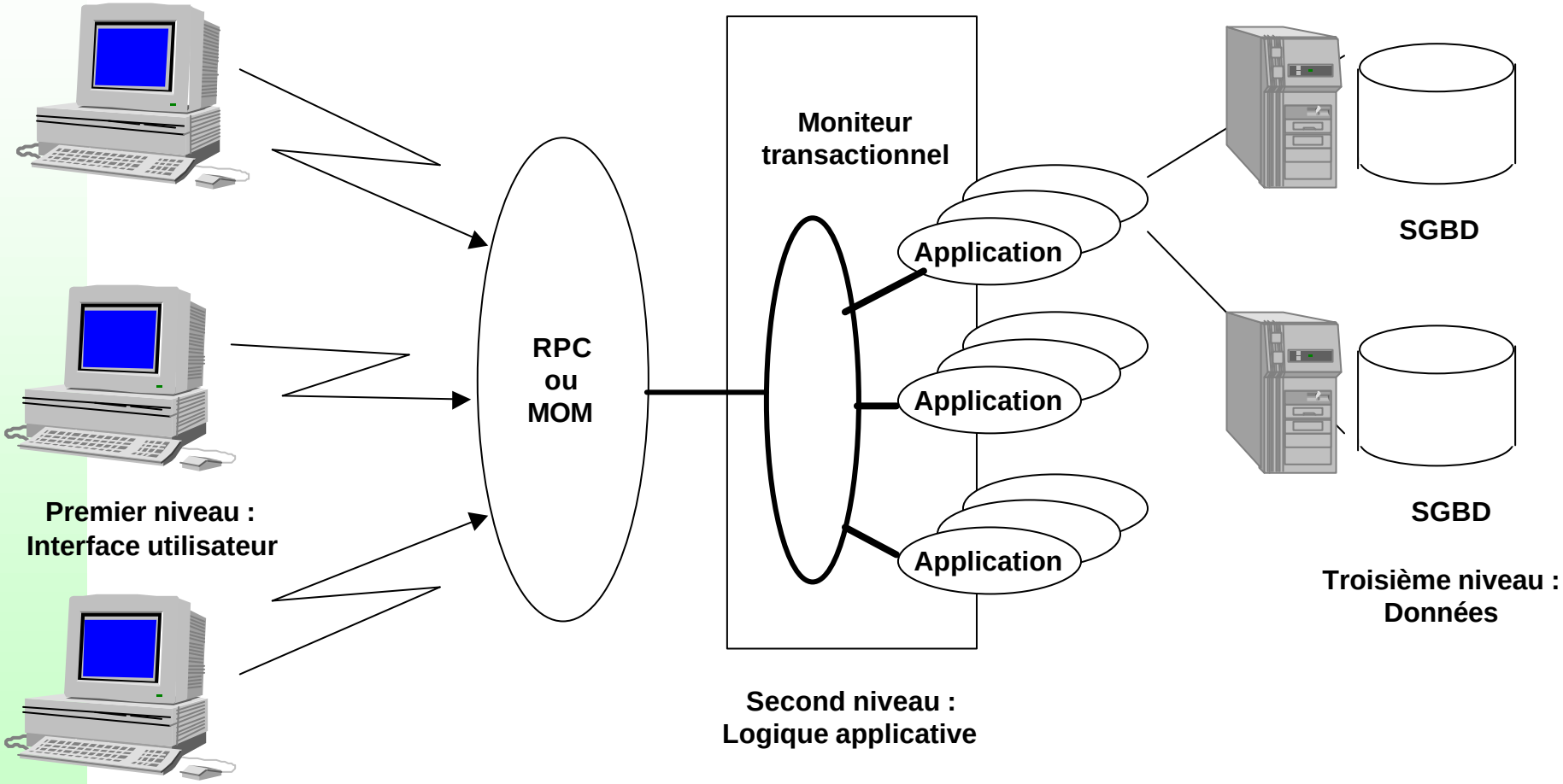
Options d'architecture Client/serveur



Note: Dans un environnement Java, les applications fonctionnant sur les clients sont des applets, les applications fonctionnant sur les serveurs sont des Servlets

Client/Serveur à 3 niveaux

Modèle d ’implémentation Client/Serveur à 3 niveaux



Moniteur transactionnel : Multiplexage des utilisateurs sur des processus serveurs

Composants du middleware

	Internet	Dialogue	Accès aux données	Transactionnel	Objets	
Services applicatifs	HTTP S-HTP SSL	HTML, Windows, Applets Java	SAG/CLI, RDA, DRDA, ODBC, JDBC	X/Open (Tuxedo, Encina, CICS 6000,...)	ORB (CORBA, COM+, …)	● ● ●
Services d'environnement distribué	Administration système (SNMP, ..)	Annuaires (LDAP)	Sécurité (Kerberos)	Temps distribué	PVM MPI	● ● ●
Services de système d 'exploitation réseau	RPC	Files de messages MOM	IPC Communication Inter-Processus distants	Fichiers distribués (NFS, DFS)		● ● ●
Services de communication	TCP/IP					
Services système d 'exploitation	Processus et threads	IPC Communication Inter-Processus locaux	Fichiers locaux			● ● ●

Note : Différents composants du middleware seront analysés dans des présentations spécifiques.

Quelques considérations économiques

Effets d'échelle

- **Matériel**
  - **Baisse des coûts continuelle du fait des volumes :**
    - **Exemple : pour les microprocesseurs, en deçà de plusieurs millions d'unités, les coûts de conception dominent**
- **Logiciel**
  - **Coût de fabrication voisin de 0 :**
    - **Distribution via Internet**
    - **Documentation "en ligne"**
  - **Coût de conception et de développement d'un logiciel important : 10 millions de $**
    - **Bill's Laws (d'après Jim Gray et Gordon Bell)**
      - **Bill Joy (Sun) " Ne pas développer de logiciel pour un marché de moins de 100 000 d'utilisateurs"**
        - **$10M x 10/100 000 -> Prix de vente = 1000 $ (1)**
      - **Bill Gates (Microsoft) " Ne pas développer de logiciel pour un marché de moins de 1 000 000 d'utilisateurs"**
        - **$10Mx10 /1 000 000 -> Prix de vente = 100 $ (1)**

(1) en faisant l'hypothèse d'une marge brute de 10 (un ratio R&D/CA de 10%). Les 90% restants se répartissent entre le coût des activités de vente, d 'administration, de support, les frais de structure, les bénéfices et les taxes. Le ratio de R&D/Chiffre d 'affaire est supérieur à 10% pour les sociétés ne produisant que du logiciel et bien inférieur à ce chiffre pour les sociétés qui se consacrent au PC. Voir explication complémentaire sur la page suivante

Taille de quelques logiciels

- **Nombre de lignes de code de différents systèmes d ’exploitation [MOO99]**

Système d’exploitation	Estimation du nombre de lignes de code (millions de lignes)
Windows 3.11	3
Windows 95	14
Windows 98	18
Windows NT 4.0	16,5
Windows 2000	35
OS/2	2
Netware 5.0	10
UNIX (moyenne)	12
Linux	5 à 6 (en croissance)
OS/400 (v4.r3)	40
MVS (OS 390 et extensions)	9-18

- **Commentaires sur ces chiffres :**
  - **Pour certains systèmes, ces chiffres doivent probablement intégrer des éléments tels que l ’interface homme/machine ou le SGBD**
  - **Coût de l’homme/an (US) : 150,000 dollars**
  - **Productivité : 2000 lignes de code par an**
  - **10 millions de dollars ® 133,000 lignes (nouvelles) de code**

Évolution de la structure de l'industrie

- **Années 70-80 - Dernier stade de la verticalisation**
  - **Intégration verticale (constructeurs généralistes)**
  - **Produits spécifiques des constructeurs ("propriétaires")**
  - **Clients captifs**
  - **Marges élevées**
- **Années 90 - Vers l'horizontalisation**
  - **Spécialisation par étape de valeur ajoutée (processeur/périphériques/système d'exploitation/SGBD/...)**
  - **Le constructeur informatique est devenu un intégrateur**
  - **Client moins captif mais confronté au problème d'intégration de produits variés**

Structure de l'industrie(2)

- **Les fournisseurs tendent à se spécialiser par étape de valeur ajoutée**
- **La banalisation des produits se traduit par une baisse des coûts et la recherche de la rentabilité par les volumes**
- **Les fournisseurs «leaders» dans un domaine tendent à empiéter sur les domaines voisins à la recherche de valeur ajoutée**

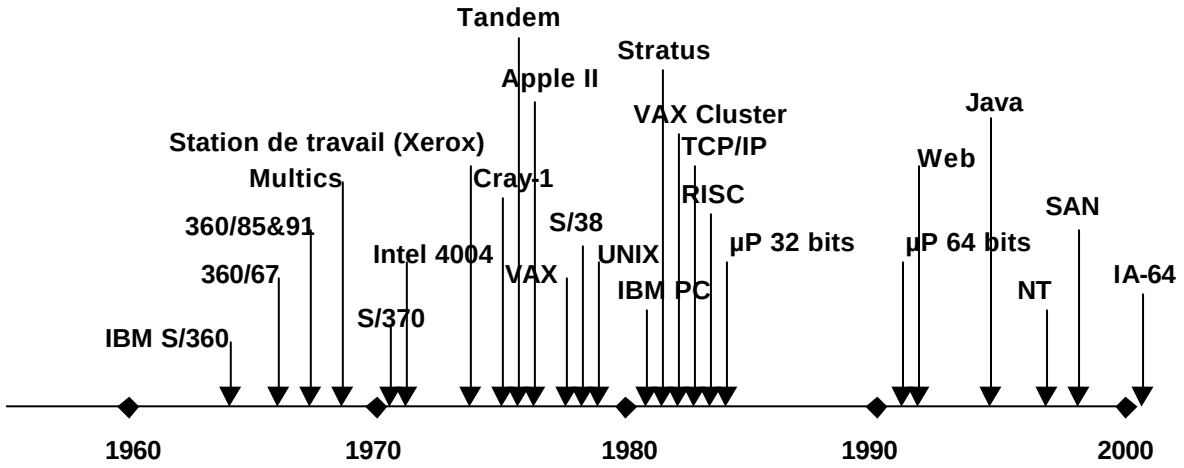
Fonction :	*Qui ?* (Exemples)
Processeur / périphériques	Intel / Seagate
Système	Compaq
Logi. de base	Microsoft
"Middleware"	Oracle
Applications	SAP
Intégration	EDS
Exploitation	CSC / Client

Structure de l'industrie(3)

- **Standardisation**
  - **Conséquence de l'horizontalisation**
  - **Standardisation de fait (par le produit et le marché plutôt que par un comité)**
  - **Standardisations concurrentes (UNIX - NT, PC - Mac,...)**
- **"Commoditization"**
  - **Baisse des prix sous l'effet conjugué de la technologie et de la production de masse**
  - **Banalisation d'un certain nombre de produits à haute technicité (processeurs, systèmes d'exploitation,....)**
  - **Transformation de l'industrie : fusions, disparitions,.....**

Perspectives

- **Un peu d ’histoire - Étapes marquantes en matière d ’architecture de système**



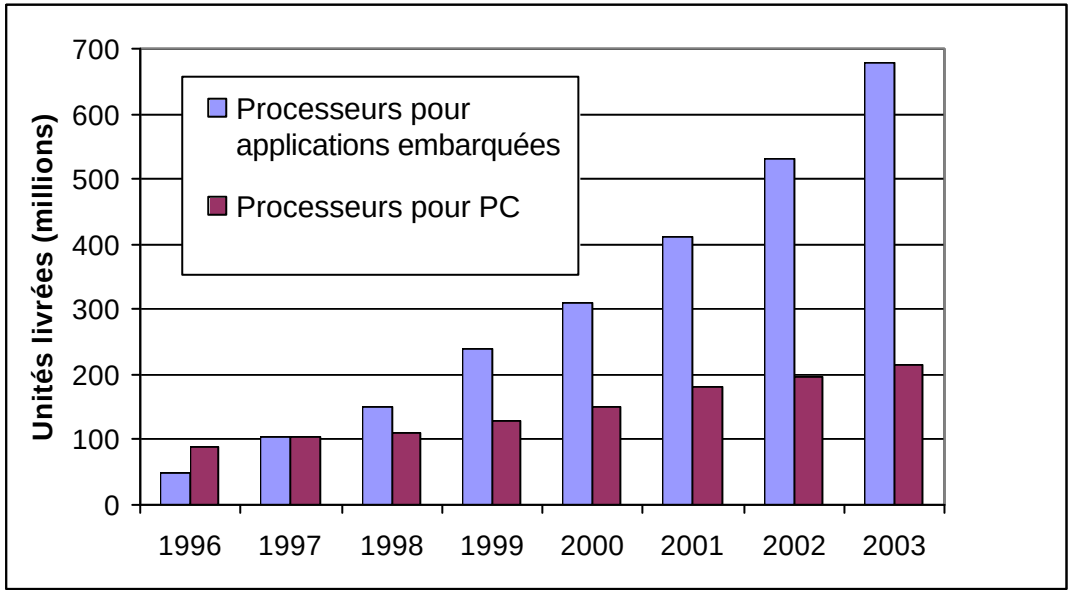
- **Matériel**
  - **Microprocesseurs. Deux axes de progression :**
    - **Nouvelle architecture (IA-64)**
    - **Amélioration des implémentations : techniques d’amélioration de la performance, puce multiprocesseur, intégration de la dimension système sur la puce, …..**
  - **Freins à la progression**
    - **Poids de la compatibilité binaire**
    - **Coût de développement**
  - **Degrés de liberté plus importants dans le domaine des microprocesseurs pour applications embarqués (à condition que les volumes soient au rendez-vous) et les systèmes spécialisés (sous-systèmes de stockage, commutateurs)**

Perspectives(2)

Potentiel de la technologie des circuits intégrés

	Année						
Caractéristique	**1997**	**1999**	**2001**	**2003**	**2006**	**2009**	**2012**
Processus (technologie)	0,25µ	0,18µ	0,15µ	0,13µ	0,10µ	0,07µ	0,05µ
Nombre de transistors (millions)	10	15	40	76	200	520	1 400
Fréquence (MHz)	350	700	1 100	1 500	2 000	2 500	3 000
Surface puce (mm^2)	200	250	320	400	500	620	750

Facteurs économiques [HEN99]



Perspectives(3)

Facteurs économiques (suite) [HEN99]

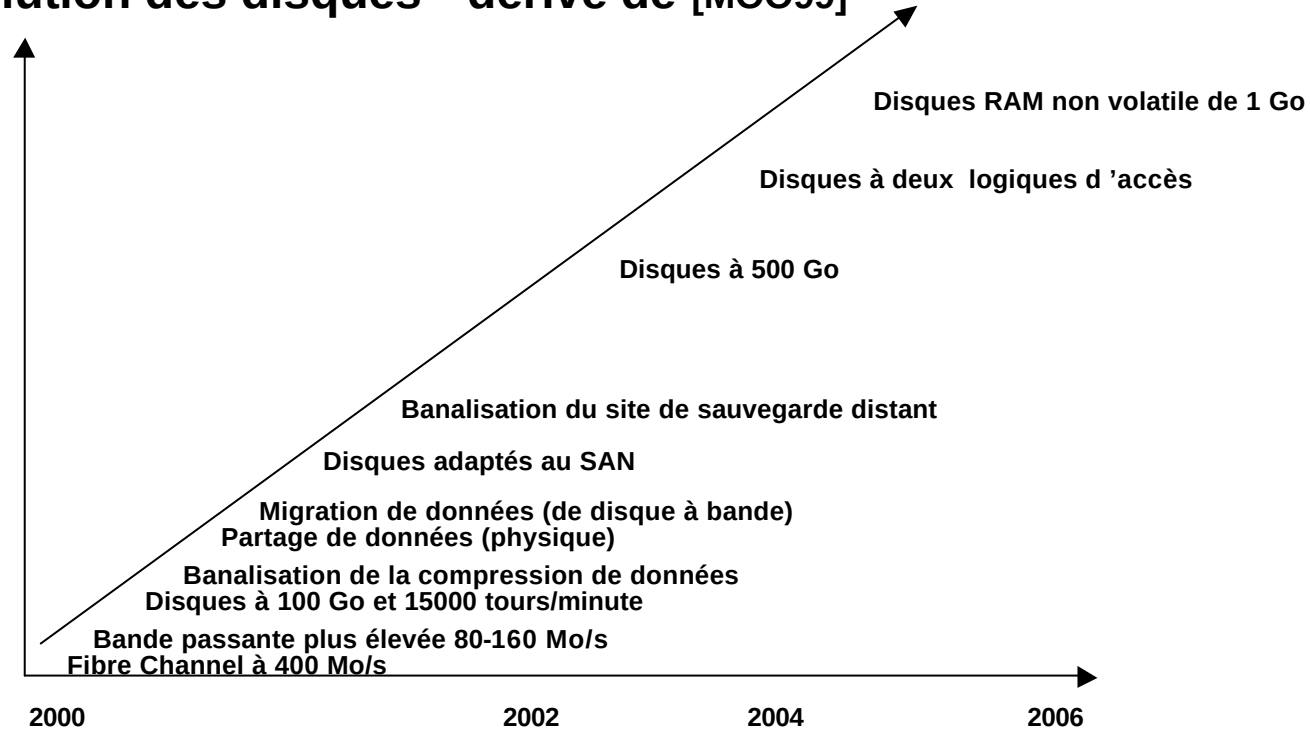
Microprocesseur	Année d'introduction	Nombre de transistors (millions)	Nombre de participants au développement	Durée du développement (mois)	Estimation du coût de la main d'œuvre (Millions de dollars)	Coût de la validation (pourcentage de l'effort total)
R2000	1985	0,1	20	15	2,5	15 %
R4000	1991	1,4	55	24	11	20 %
R10000	1996	6,8	>100	36	30	> 35 %

Mémoire intelligente

- **Principe : intégrer un microprocesseur au sein des puces mémoires et déporter certaines opérations de traitement au niveau des puces mémoire**
- **Avantages :**
  - **Faible latence et bande passante élevée**
  - **Efficacité énergétique et faible consommation**
- **Difficultés :**
  - **Différences dans les processus de production des puces mémoire et des microprocesseurs**
  - **Limitation de la capacité mémoire «par processeur» à celle de la puce**
  - **Caractère inflexible du ratio puissance de traitement/capacité mémoire**
  - **Adhésion de l'industrie (logiciel en particulier car nouvelle architecture)**
  - **Coût du test**

Perspectives(4)

- **Entrées-sorties**
  - **Poursuite de la standardisation**
    - **PCI, SCSI, Fibre Channel**
    - **InfiniBand**
  - **Généralisation des sous-systèmes :**
    - **Stockage**
    - **Communication**
- **Évolution des disques - dérivé de [MOO99]**



Perspectives(5)

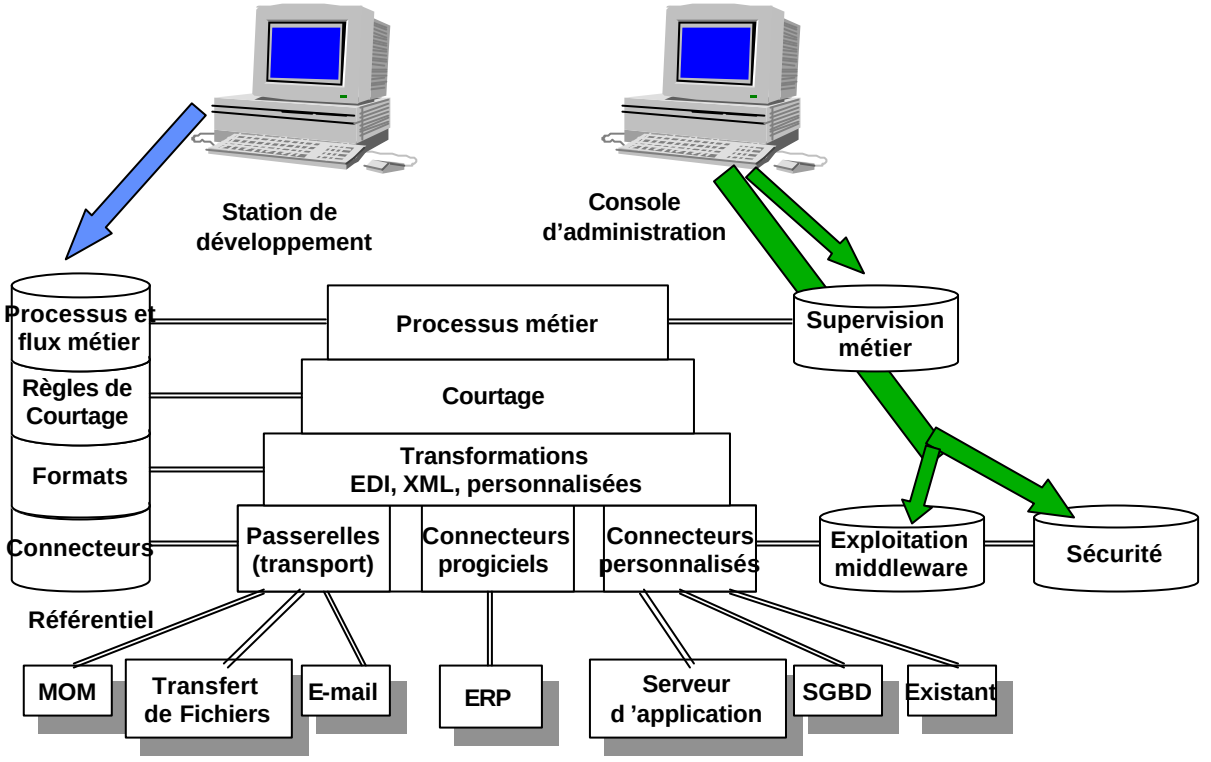
- **Disques intelligents**
  - **Exploitation des capacités de traitement et de mémorisation contenues dans les unités de disques magnétiques**
  - **Déport de certaines fonctions du système de gestion de fichiers et/ou des SGBD dans les unités de disques (exemple : filtrage à la volée)**
  - **Difficulté : adhésion de l ’industrie du logiciel**
- **Considération générale : le processus d’évolution se caractérise par l’alternance de phases de stabilité et de phases de rupture :**
  - **Phase de stabilité : changement homothétiques (ou incrémentaux) ne remettant pas en cause les équilibres établis**
    - **exemple : évolution des architectures CISC au début des années 80**
  - **Il existe parallèlement des évolutions technologiques qui remettent en cause les équilibres et conduisent à une rupture (exemples : capacité des puces mémoire et compilateurs optimisants qui ont conduit aux architectures RISC toujours dans les années 80)**
  - **Une phase de rupture se caractérise par un bouillonnement des idées parmi lesquelles le marché «fait le tri»**

Perspectives(6)

- **Logiciel**
  - **Poursuite de la structuration induite par le modèle Client/Serveur**
    - **SGBD relationnels**
    - **Moniteurs transactionnels**
    - **RPC, MOMs, CORBA/COM+**
    - **EJB**
    - **Généralisation de l'interface homme/machine «Web»**
  - **Portabilité du code avec Java**
  - **Échange de données avec XML**
  - **Évolution vers la réalisation d'applications par intégration de composants standard COTS (Components Off The Shelf)**
    - **Difficultés de l 'approche COTS :**
      - **Conformité des composants à leurs spécifications**
      - **Existence de propriétés émergentes et/ou immergentes**
      - **Procédures de reprise sur défaillance**
      - **Synchronisation des différentes composantes**
      - **Dépendance vis-à-vis des fournisseurs**
      - **Difficulté de substitution d'un composant par un autre composant**

Perspectives(7)

- **Intégration des applications d ’entreprise et urbanisation des systèmes d’information**
  - **Principe : ne pas remettre en cause l ’existant mais permettre la coopération entre les différentes parties constitutives du système d’information**
- **Exemple de composants techniques d ’une offre d ’intégration [OCT99]**



Tendances - Les lois du logiciel

- **Lois de Nathan Mryvold (Centre de recherches Microsoft)**
  - **1ère Loi :** le logiciel a les propriétés d'un gaz : au cours du temps, il s'étend de façon à occuper toute la capacité du système qui le supporte.
    - de 93 à 97, taille de NT a doublé tous les 866 jours (33,9% par an)
    - de 95 à 97, taille du code d'un navigateur a doublé tous les 216 jours (223% par an)
  - **2ème Loi** : le logiciel croît jusqu'à ce qu'il soit limité par la loi de Moore (doublement de la performance tous les 18 mois).
    - Croissance suivant la première loi qui se trouve limitée par les possibilités du matériel, les possibilités de la nouvelle génération étant très rapidement consommées
  - **3ème Loi** : la croissance du logiciel favorise le fait que la loi de Moore se vérifie
    - Comme le logiciel « butte » sur les possibilités du matériel, les utilisateurs investissent dans de nouveaux matériels (proposés au même prix que la précédente génération)
  - **4ème Loi** : le logiciel est seulement limité par l'ambition humaine et par les possibilités de financement de l'activité de développement

Références et adresses utiles

Cette présentation est fondée sur l'ouvrage suivant :

[CHE00] René J. Chevance « Serveurs multiprocesseurs, clusters et architectures parallèles »
Eyrolles 2000

Autres références :

[HEN91] John L. Hennessy, P. Joupi " Computer Technology and Architecture: An Evolving Interaction "
Computer Vol.24,No.9, September 1991

[HEN99] John L. Hennessy, "The Future of Systems Research",
Computer, Vol. 32, No 8, August 1999, pp. 27-33.

[MOO99] Fred Moore, "Storage Panorama 2000"
StorageTech, http://www.storagetech.com

[OCT99] OCTO Technology, " Le livre blanc de l'EAI - Intégration des applications d'entreprise ".
http ://www.octo.fr

Sites de quelques fournisseurs :
http://www.bull.com, http://www.cisco.com,
http://www.compaq.com,
http://www.emc.com, http://www.hp.com,
http://www.ibm.com,http://www.intel.com,
http://www.microsoft.com,
http://www.seagate.com,http://www.sun.com

Quelques autres adresses utiles

Jim Gray http://research.microsoft.com/~Gray
Microprocessor Report http://www.MDRonline.com
ACM http://www.acm.org
IEEE http://www.ieee.org
http://www.techniques-ingenieur.com
Diffusion de nouvelles http://www.theregister.uk