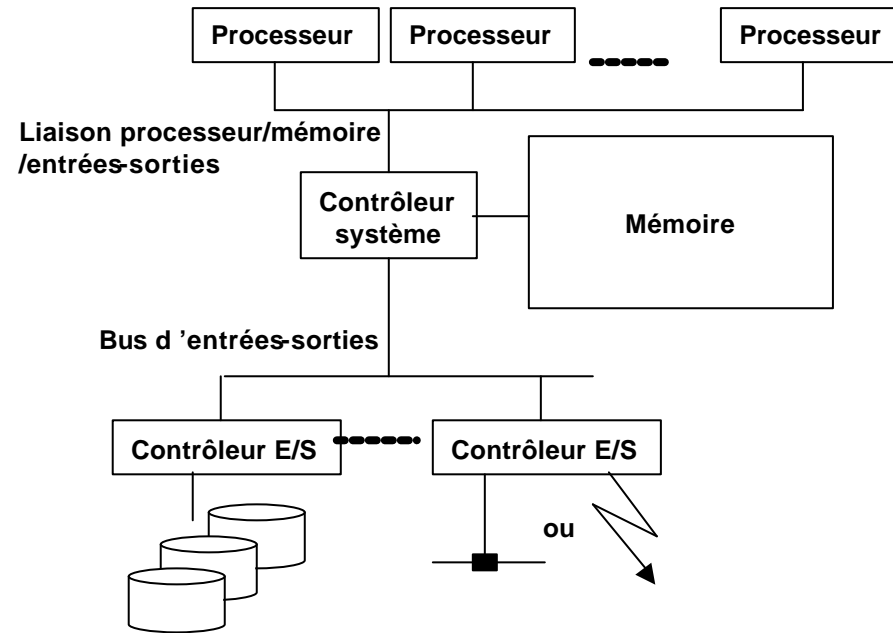


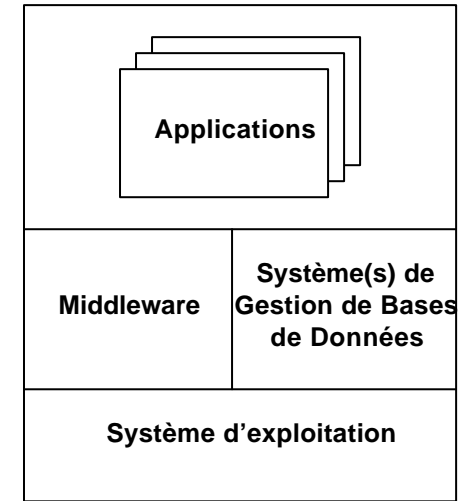
Rappel des options d'architecture

- **Deux grandes familles :**
 - **Couplage serré ou multiprocesseur symétrique (SMP pour Symmetric Multiprocessor))**
 - **Couplage lâche**
 - **Clusters**
 - **Massively Parallel Processing (MPP)**
- **Comparaison**
- **Une autre vision des architectures**
 - **Share Everything**
 - **Shared Disks**
 - **Share Nothing**

Couplage serré - SMP



a) Vision matérielle de l'architecture multiprocesseur symétrique



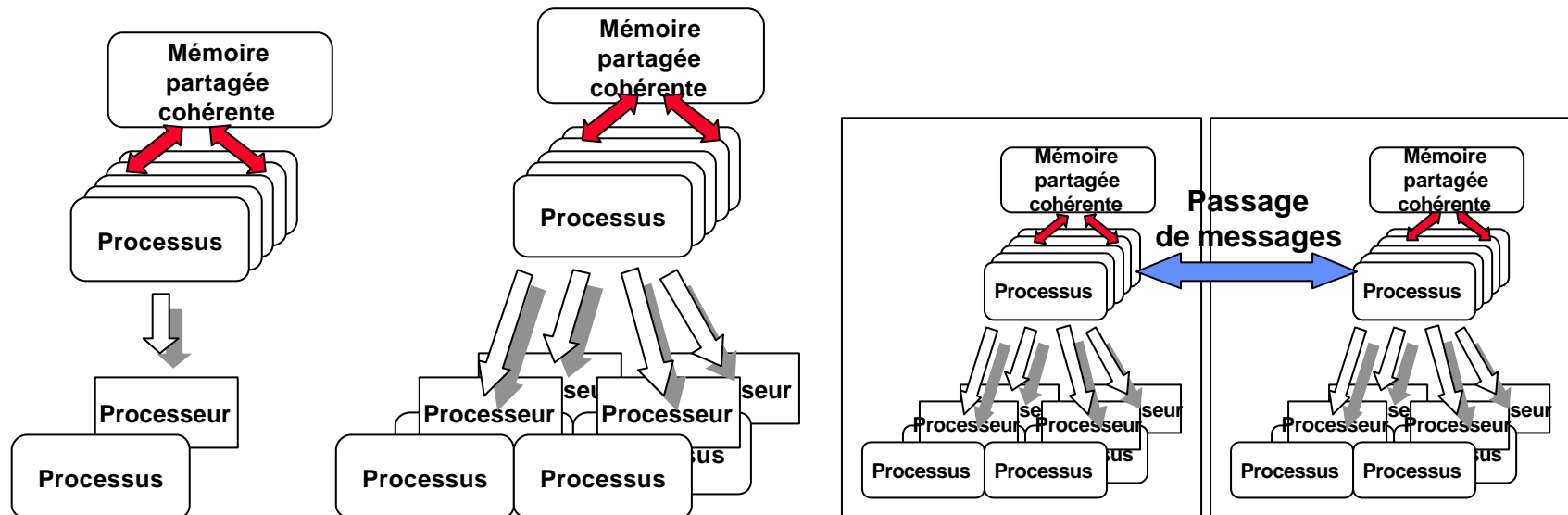
Ressources matérielles
 (processeurs, mémoire, contrôleurs
 et périphériques)

b) Vision logicielle de l'architecture multiprocesseur symétrique

- ***Dans un SMP, tous les processeurs peuvent accéder à toutes les ressources du système (mémoire, dispositifs d'entrées-sorties). Un SMP fonctionne sous le contrôle d'un seul système d'exploitation, qui gère donc l'ensemble des ressources du système.***

- ***Le modèle naturel de programmation est le partage de mémoire entre les processus.***

■ Modèles d'exécution et de programmation



a) - Monoprocesseur

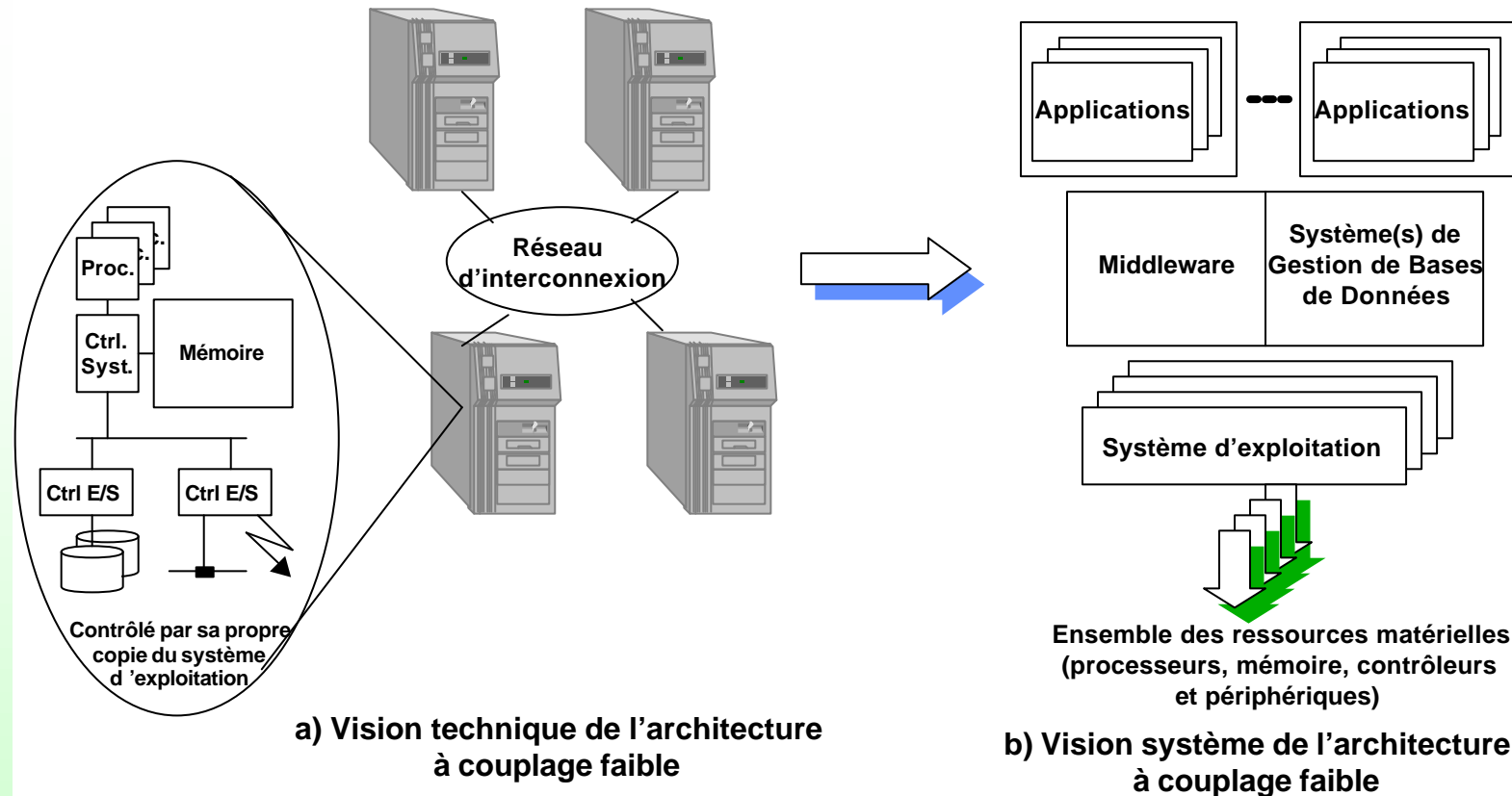
b) - Multiprocesseur à couplage serré

c) - Multiprocesseur à couplage lâche

Couplage serré - Avantages et inconvénients

- + "Scalabilité" du système à un coût modéré;
- + Augmentation de performance aisée: ajout d'une carte ou d'un module processeur;
- + Efficacité multiprocesseur: l'ajout d'un processeur augmente la puissance (dans des limites définies);
- + Simplicité et efficacité du modèle de programmation
- + Transparence des applications: les applications pour "mono-processeur" s'exécutent sans changement mais seules les applications "multi-threaded" tirent parti de l'architecture;
- + Disponibilité: suite à la panne d'un processeur, le système peut re-démarrer avec les processeurs restants;
- + Possibilité de partitionner le système
- Existence d'un point de défaillance unique constitué par le système d'exploitation (ainsi que par des éléments matériels)
- Les opérations de maintenance nécessitent, généralement, l'arrêt du système
- Limitation du nombre de processeurs du fait des conflits d'accès au niveau matériel (bus) et logiciel (système d'exploitation, SGBD,);
- Complexité du matériel et du logiciel pour les SMP à grand nombre de processeurs (CC-NUMA);
- Adaptation et réglage coûteux du système d'exploitation;
- Adaptation nécessaire des applications pour tirer profit de la performance disponible.

Couplage lâche

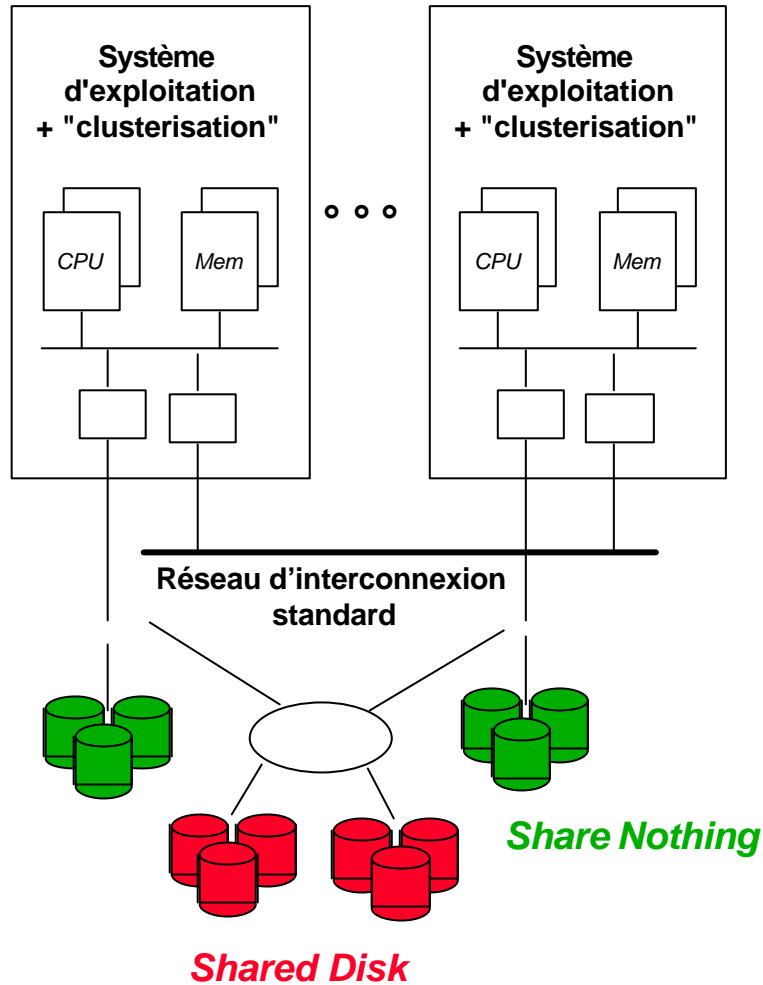


- Le système est constitué par l'interconnexion, au moyen d'une technologie de réseau local rapide, d'un certain nombre de systèmes indépendants (appelés nœuds) , chacun de ces systèmes possède ses propres ressources (processeurs, mémoire, entrées-sorties) et fonctionne sous le contrôle de sa propre copie du système d'exploitation.

- C'est un couplage lâche car les nœuds ne partagent pas de mémoire (cas général).
- Différences par rapport à un système distribué : homogénéité des différents nœuds (fournisseur et système d'exploitation), proximité géographique et image système unique pour certaines ressources.
- Le mode naturel de communication des applications est le passage de messages.

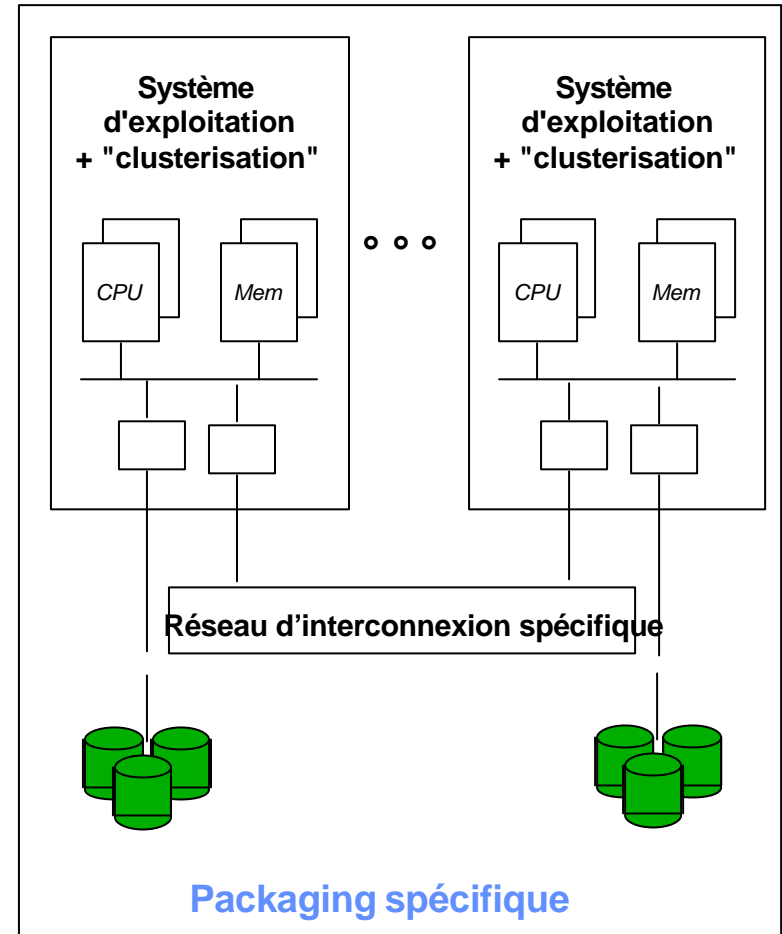
Cluster et MPP

Cluster



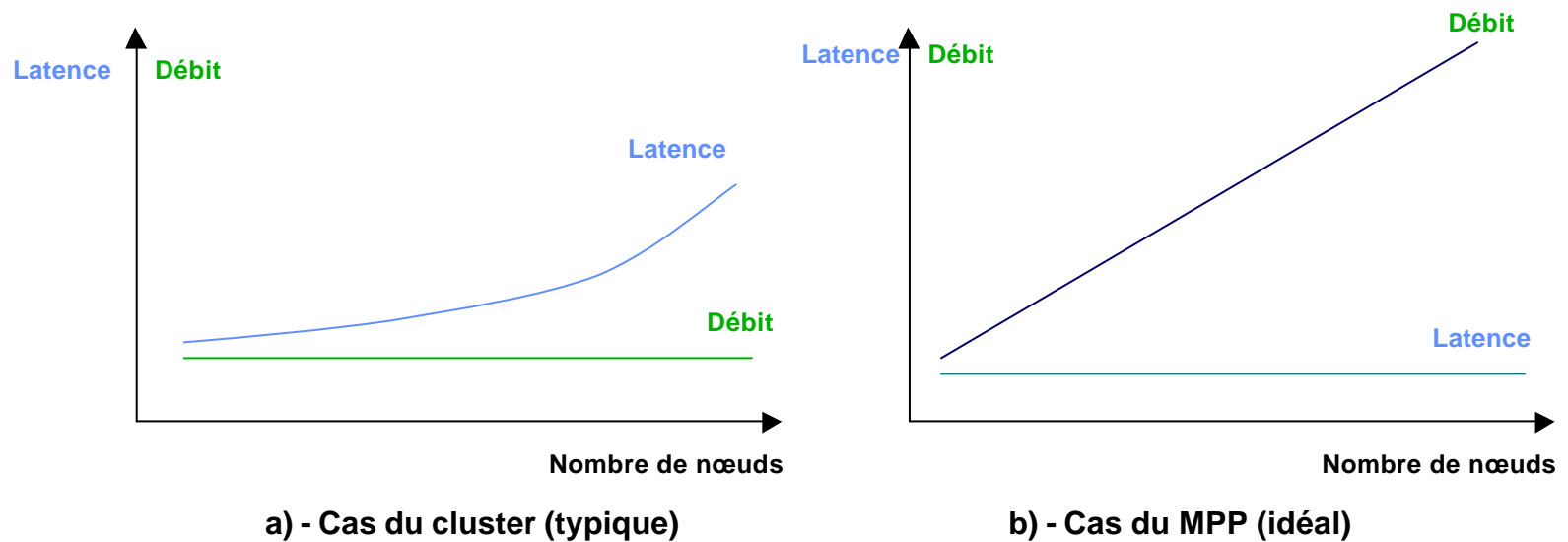
**Deux options d'architecture
 au niveau des disques**

MPP



**Typiquement une seule option
 d'architecture au niveau des disques**

■ Caractéristiques comparées



- Ensemble de systèmes (noeuds) interconnectés
- Réseau d'interconnexion de technologie standard e.g. Ethernet, FDDI, Fibre Channel
- Chacun de ces noeuds dispose de processeur(s), de mémoire, de ressources locales d'entrées-sorties et dispose de sa propre copie du système d'exploitation
- Système d'exploitation dérivé d'un système existant (e.g. VMS, Unix, NT) avec :
 - Vision de système unique (Single System Image ou SSI) pour l'administrateur
 - Accès transparent à certaines ressources dites "clusterisées" quelque soit leur localisation (e.g. système de fichiers, adresse IP unique pour le cluster, files des travaux d'impression, lignes de communication,....)
(Single System Image - SSI- pour les utilisateurs)
- Concept introduit par Tandem (fin des années 70, objectif : FT) et popularisé par DEC (1983, objectif : croissance)

- + **Haute disponibilité intrinsèque (indépendance des noeuds);
Implémentation aisée au niveau du matériel;**
- + **Compatibilité avec les systèmes mono ou multiprocesseur au niveau des applications;**
- + **Augmentation de performance pour les SGBDs (OLTP s'il y a peu d'interactions entre noeuds, DSS);**
- + **Intégration aisée de nouvelles technologies (processeurs et nouvelles versions des systèmes d'exploitation);**
- + **Partage transparent des ressources "clusterisées ».**
- **Efficacité multiprocesseur limitée (par rapport au SMP)**
- **Implique des modifications du système d'exploitation pour le fonctionnement en cluster (SSI difficile);**
- **Nécessite de modifier les applications pour pouvoir tirer profit de l'augmentation de puissance procurée par le cluster (i.e. capacité de l'application à exploiter simultanément plusieurs noeuds). En pratique, seuls les SGBDs ont été adaptés pour tirer profit de telles architectures (e.g. Oracle Parallel Server Option);**
- **Standard de programmation des application parallèles émergents;**
- **Limitation du nombre de systèmes interconnectés à quelques unités (de l'ordre de la dizaine au maximum);**
- **Difficultés d'administration du cluster.**

- **Des fonctionnalités similaires mais une grande diversité d'implémentations**
- **Des solutions spécifiques (DEC, HP, IBM, NCR, Sun,...) dépendantes de chaque fournisseur**
- **Des solutions éprouvées**
- **La fonctionnalité de File System qui faisait cruellement défaut sur les différents clusters Unix (alors qu'elle était disponible que Vax Cluster depuis 1983) commence à arriver (TruCluster de Compaq, AIX, Solaris)**
- **Absence d'API et de SDK standard**
- **Pratiquement, seuls des fournisseurs de SGBD et d'ERP ont adapté leurs produits aux clusters les plus vendus sur le marché**

Massively Parallel Processing - MPP

- Ensemble de noeuds interconnectés au moyen d'un réseau spécialisé (minimisation de la latence, maximisation du débit et "scalabilité")
- Packaging adapté au support d'un grand nombre de noeuds et aux extensions du nombre de noeuds
- Grand nombre de noeuds potentiel ($O(100)$ à $O(1000)$, pratiquement 10 - 100)
- Chaque noeud exécute sa propre copie du système d'exploitation (cas général)
- Objectif : recherche de la performance au moyen du parallélisme des applications
- Peu d'applications parallélisées en pratique (SGBD essentiellement et calcul numérique intensif)
- Convergence de l'industrie vers le couplage de noeuds SMP
- Pratiquement, élimination de la totalité des start-ups

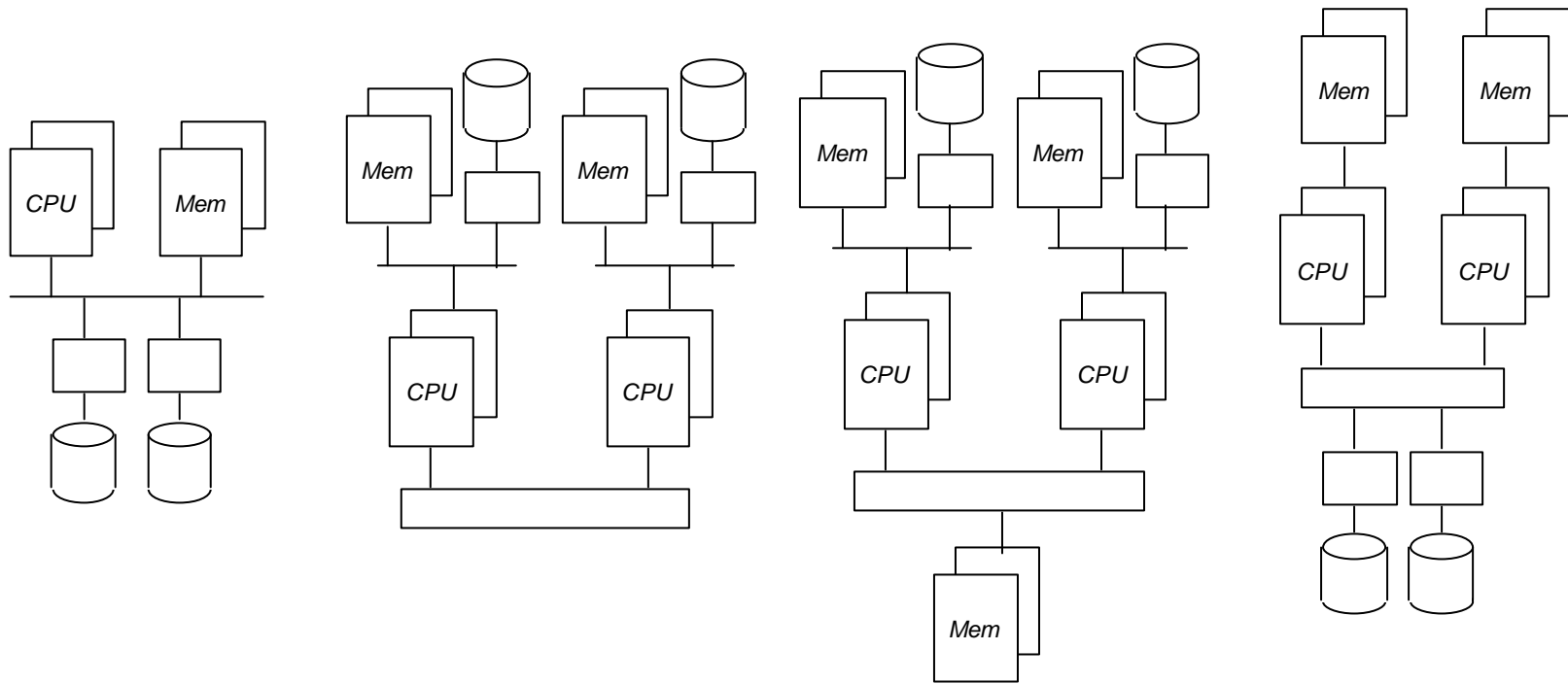
- + **Avantage coût/performance vis à vis des super-ordinateurs pour les applications scientifiques**
- + **Scalabilité en performance limitée seulement par le degré de parallélisation de l'application (i.e. le MPP n'a pas les limitations en nombre de processeurs du SMP ou du cluster)**
- + **Potentiel de haute performance (à la fois Speedup et Scaleup) en environnement transactionnel (OLTP) et décisionnel (avec SGBDs adaptés)**
- + **Haute disponibilité (potentielle)**
- **Efficacité limitée**
- **Technologie émergente**
- **Réseau d'interconnexion spécifique**
- **Implique des modifications du système (e.g. SSI)**
- **Standard de programmation des applications parallèles émergent**
- **Difficulté de développement des applications « parallèles »**
- **Difficultés d'administration du système**
- **Nombre limité d'applications adaptées à ce type d'architecture**

SMP, Cluster et MPP (1ère comparaison)

- **Un système est un SMP si il a:**
 - plusieurs processeurs;
 - une seule copie du système d'exploitation;
 - une mémoire cohérente partagée.
- **Un système est un Cluster si il a:**
 - plusieurs noeuds interconnectés (typiquement quelques unités à une dizaine);
 - une copie du système d'exploitation par noeud et un accès transparent à certaines ressources;
 - un système d'interconnexion "lâche" et standard (Ethernet, FDDI, FCS) puisque l'objectif est d'autoriser la coexistence de générations technologiques différentes au sein d'un même cluster.
- **Un système est un MPP si il a:**
 - plusieurs noeuds interconnectés (typiquement plusieurs dizaines à quelques centaines) au sein d'un packaging matériel spécifique (permettant l'extension du nombre de noeuds);
 - une copie du système d'exploitation par noeud;
 - un système d'interconnexion "serré" puisque l'un des objectifs est de fournir un support efficace aux communications au sein des applications parallèles. Généralement, ce mécanisme d'interconnexion est spécifique et il présente des limitations dans la co-existence de noeuds de générations technologiques différentes.

Une autre vision des architectures

■ Une autre vision de l'architecture en relation avec le support des SGBD



*Share Everything
(SMP)*

*Share Nothing
(typique MPP)*

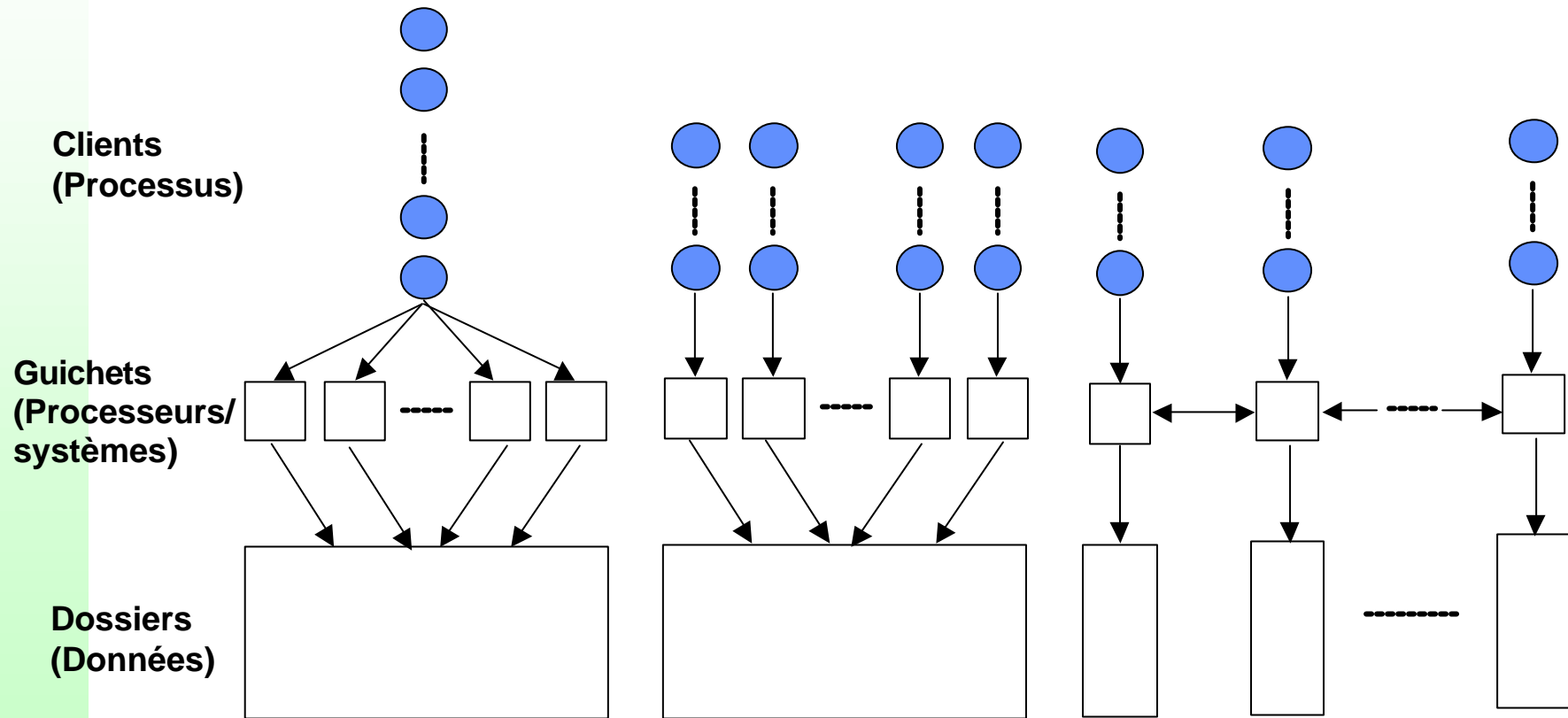
*Share Intermediate Memory
(e.g. IBM Sysplex,
Reflective Memory)*

*Shared Disk
(certains clusters)*

Relation SGBD - Architectures Share*

- Analogie entre les architectures Share Everything, Share Nothing et Share Disk avec l'organisation de guichets de service

Note : D'après une idée de J Papadopoulo (Bull)



Modèle Share Everything

Modèle Share Disk

Modèle Share Nothing

Comparaison Share* pour support SGBD

	Share Everything	Shared Disk	Share Intermediate Memory	Share Nothing
A V A N T A G E S	<ul style="list-style-type: none"> + Simple pour parallélisme inter-requêtes + Assez simple pour le parallélisme intra-requête + Bonne utilisation des ressources, équilibrage de charge « naturel » + Communication inter-processeur efficace (i.e. utilisation de la mémoire partagée cohérente) + Solution en cours de banalisation en bas de gamme (4 à 8 processeurs) 	<ul style="list-style-type: none"> + Bonne fiabilité et bonne disponibilité + Bonne scalabilité (100 processeurs et plus) + Faible coût du fait de la réutilisation de composants standard + Bon équilibrage de charge (les données à fort taux de partage en lecture peuvent être répliquées) 	<ul style="list-style-type: none"> + Bonne scalabilité (mais qui dépend de l'implémentation de la mémoire partagée) + Faible coût du fait de la réutilisation de composants standard + Bonne performance du fait des tampons partagés et de la communication par mémoire partagée 	<ul style="list-style-type: none"> + Bonne fiabilité et disponibilité + Très bonne scalabilité (plusieurs centaines de processeurs) + Faible coût du fait de la réutilisation de composants standard
I N C O N V E N I E N T S	<ul style="list-style-type: none"> - Scalabilité limitée (de 16 à 64 processeurs) mais peut être étendue avec CC-NUMA - Disponibilité du système difficile à assurer - Solution coûteuse pour un grand nombre de processeurs 	<ul style="list-style-type: none"> - Interaction entre les nœuds pour la synchronisation des mises à jour des données - Saturation du réseau d'interconnexion par les transferts entre nœuds et disques - Coût du maintien de la cohérence de copies multiples (si réplication) en particulier en cas de mises à jour fréquentes - le mécanisme d'interconnexion des disques limite le nombre de noeuds 	<ul style="list-style-type: none"> - Le coût de l'accès à la mémoire partagée est clé pour la performance du système - Mémoire partagée spécifique - La fiabilité et la disponibilité du système réclament une conception spécifique de la mémoire partagée - Modèle de programmation? - Absence de volonté de la part des fournisseurs de SGBD standard de supporter des particularités architecturales 	<ul style="list-style-type: none"> - Equilibrage des charges difficile - Difficile à administrer et à optimiser du fait du partitionnement des données - Forte dépendance de la performance vis-à-vis des caractéristiques d réseau d'interconnexion - Coût de la parallélisation, même pour des requêtes simples - Coût du maintien de la cohérence de copies multiples (si réplication), en particulier si les mises à jour sont fréquentes

Offres SGBD et options d'architecture

Type d'architecture	Share Everything (SMP)	Shared Disk (clusters et MPP)	Share Nothing (clusters et MPP)
SGBD	<ul style="list-style-type: none"> - IBM DB2 - Informix - Oracle - SQL Server - Sybase - Tandem ServerWare SQL - Teradata 	<ul style="list-style-type: none"> - Oracle Parallel Server 	<ul style="list-style-type: none"> - IBM DB2 (Parallel Edition) - Informix Extended Parallel Option - SQL Server (en projet) - Sybase (en projet) - Tandem ServerWare SQL - Teradata

Commentaires:

- **Share Disk (Oracle).** Tout noeud doit pouvoir accéder à toutes les données. Les échanges de données transitent par les disques.
- **Share Nothing.** Informix et DB2 sont basées sur une architecture de type "Function Shipping" i.e. envoi de la demande de fonction à exécuter au noeud ayant l'accès aux données.
- **Share Disk** suppose que l'efficacité du réseau d'interconnexion entre les noeuds et les disques est suffisante pour s'affranchir de la répartition des données.
- **Share Nothing** repose sur l'hypothèse que les données sont réparties de façon telle que les débits d'E/S et les capacités de traitement sont utilisées de façon optimale (rôle clé de l'administrateur système dans la répartition).

Offres SGBD et options d'architecture(2)

- **Les notions Share Everything, Shared Disks et Share Nothing dépendent :**
 - de l'architecture du matériel;
 - du système d'exploitation
- **Par définition, un système d'exploitation pour SMP supporte Share Everything**
- **En cas de couplage lâche, les options peuvent différer**
 - Exemple : un cluster de systèmes ayant une connexion avec les disques de type SAN est de type Shared Disks au niveau du matériel mais le système d'exploitation peut imposer une architecture de type Share Nothing
- **Il est toujours possible de simuler une option d'architecture au dessus d'une architecture matérielle et d'un système d'exploitation ayant retenu une option différente :**
 - Shared Disk au dessus de Share Nothing : concept d'entrées-sorties distantes
 - Share Nothing au dessus de Shared Disks : partitionnement des ressources

Rapprochement des besoins, des options et comparaison des options

Synthèse des caractéristiques des SMP, Clusters et MPP

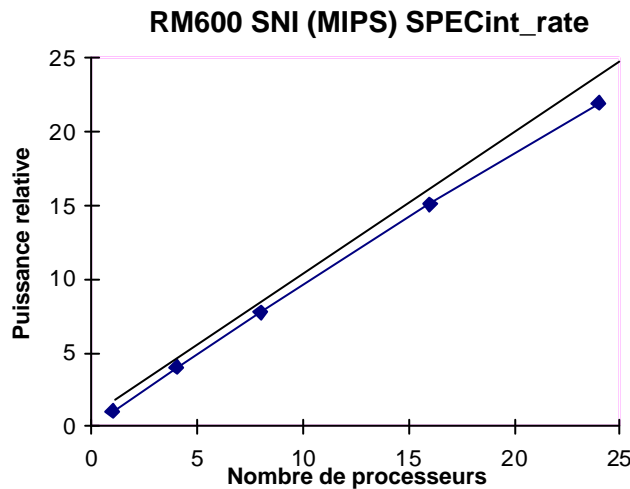
Caractéristiques	SMP	Cluster	MPP
Accélération (speed up) ou Accroissement (scale up)	Accroissement Accélération	Accroissement Accélération	Accélération Accroissement
Équilibrage de charge	Implicite	Nécessite l'intervention d'un logiciel	Nécessite l'intervention d'un logiciel
Haute disponibilité	Typiquement non	Objectif principal	Possible (n'est généralement pas un objectif)
Configuration importante (100 processeurs et au-delà)	Limitée sur technologie de commodité, des technologies spécifiques sont requises pour des configurations importantes	Limitée par les caractéristiques du réseau d'interconnexion (souvent de technologie standard)	Objectif principal (réseau d'interconnexion spécifique)
Image système unique	Complète (par définition)	Limitée	Limitée
Partage	Tout (y compris la mémoire et le système d'exploitation)	Limité (typiquement les disques et les connexions réseau)	Limité (typiquement les connexions réseau)
Programmation	Processus unique ou processus multiples et threads permettant d'exploiter le parallélisme	Programmation spécifique nécessaire dans la mesure où l'objectif est d'exploiter le parallélisme	Programmation spécifique nécessaire afin d'exploiter le parallélisme (élément plus crucial que pour les clusters)
Flexibilité pour l'intégration de technologies de générations différentes	Très limitée	Oui	Limitée
Facilité de maintenance	Limitée (implique souvent l'arrêt de l'exploitation)	Aisée (n'implique pas l'arrêt du système)	Aisée (n'implique pas l'arrêt du système)

Critères de choix et architectures

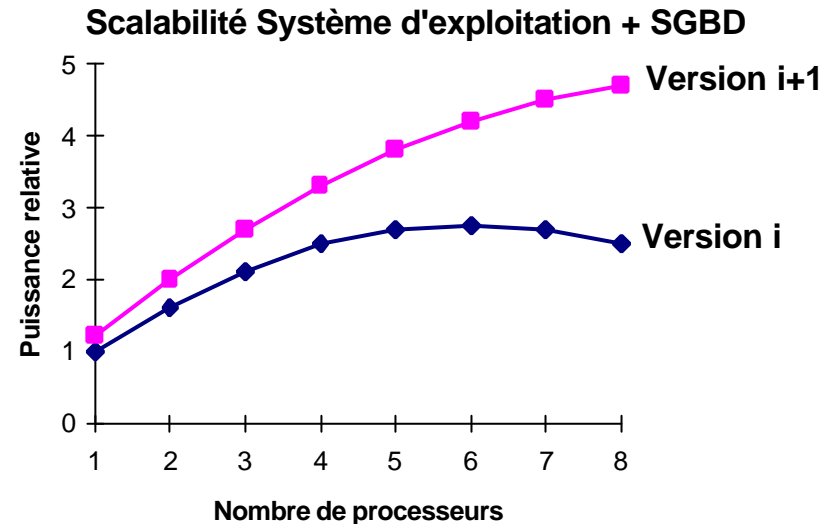
Besoin	Multiprocesseur symétrique (SMP)	Cluster	Machines massivement parallèles (MPP)
Disponibilité des applications et des outils de développement	***	*	*
Intégrité des données	*** (Dépend des SGBD et des moniteurs transactionnels)	*** (Dépend des SGBD et des moniteurs transactionnels)	*** (Dépend des SGBD et des moniteurs transactionnels)
Disponibilité	** (Existence de points de défaillance uniques)	*** (Objectif essentiel de l'architecture)	* (N'est généralement pas un objectif)
Performance	*** (Dans les limites de la configurabilité)	** (Dépend des caractéristiques de l'application)	** (Dépend des caractéristiques de l'application)
Scalabilité	*** (Dans les limites de la configurabilité)	* (Dépend des caractéristiques de l'application)	* (Dépend des caractéristiques de l'application)
Prix	*** (Configurations petites et moyennes)	**	* (Coût du réseau d'interconnexion et caractère novateur)
Support du client-serveur	**	*** (Architecture multiserveur)	** (Architecture multiserveur)
Maturité de l'architecture	*** (Plus de trente ans d'expérience)	** (Plus de quinze ans d'expérience)	* (Technologie émergente)
Pérennité des investissements	** (Limitée à une, voire deux générations)	*** (Possibilité de mixer différentes générations)	* (Technologie émergente)

SMP - Amélioration de la performance

■ Quelques exemples d'accroissement de performance en SMP



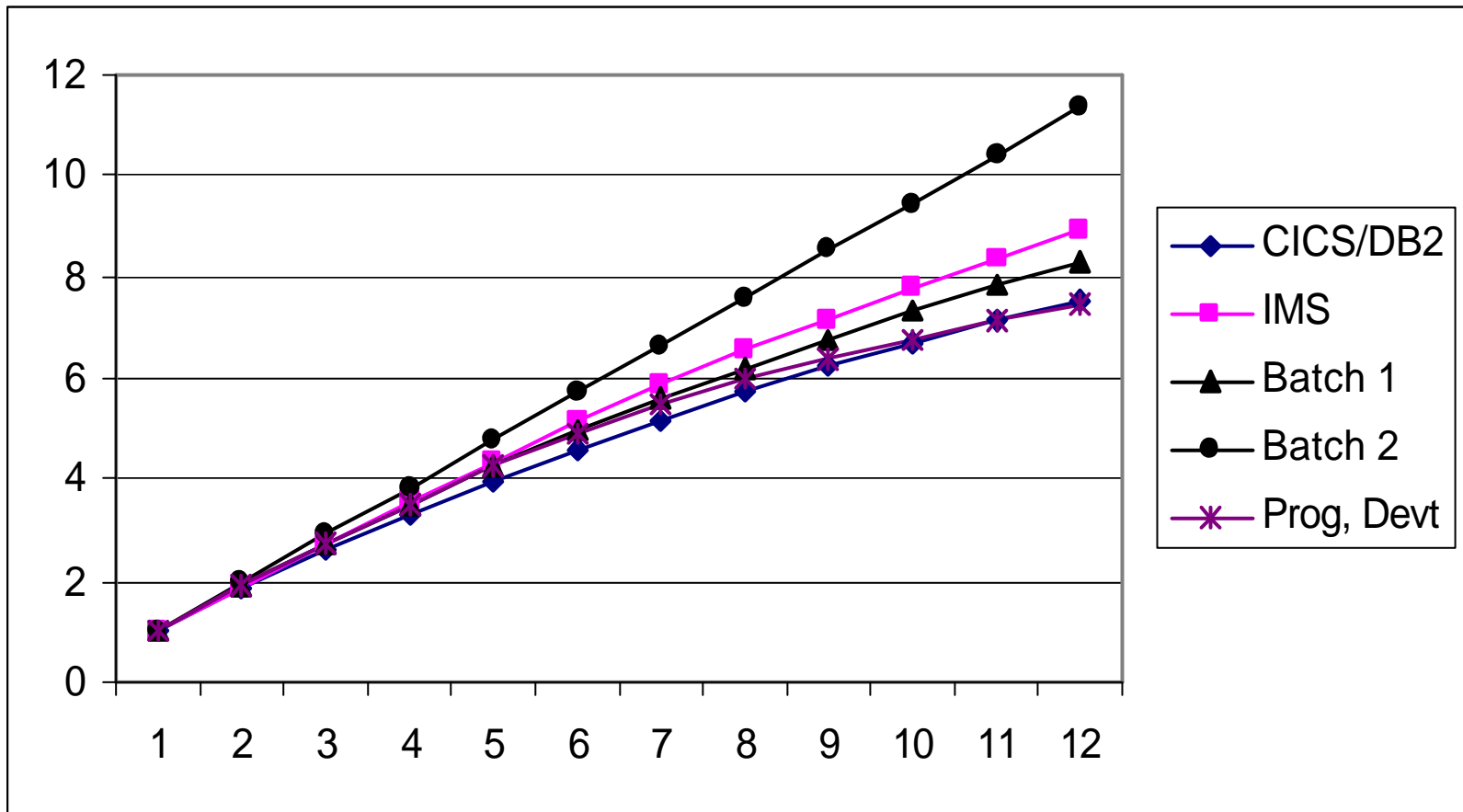
*Sans réelle signification
 compte tenu du caractère
 non représentatif de l'étalon*



*Représentatif compte tenu de
 l'étalon utilisé.
 Montre l'incidence des modifications
 du système et du logiciel de gestion
 de base de données (SGBD)*

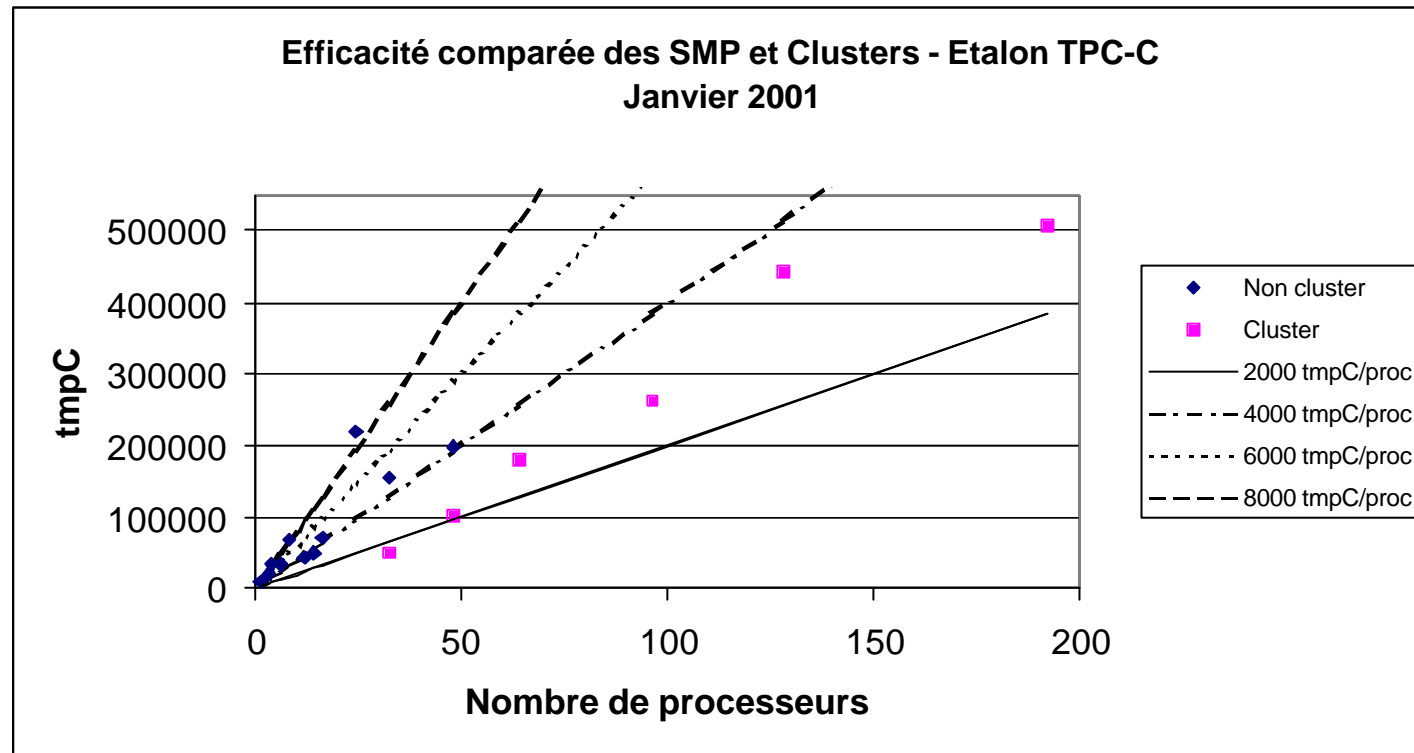
Exemple de scalabilité SMP

■ Chiffres tirés de Large System Performance Report d'IBM et portant sur la série S/390



Comparaison des performances

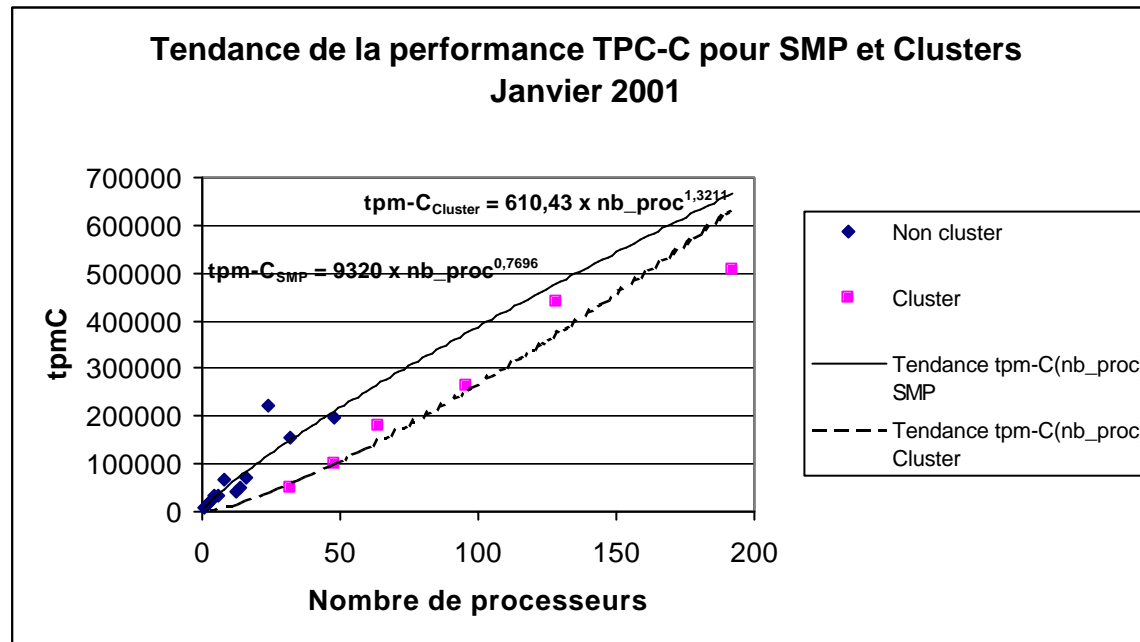
■ Efficacité des SMP et des clusters pour le transactionnel



- Les SMP sont plus efficaces que les clusters : communication par mémoire partagée, un seul SGBD alors que dans les clusters les instances du SGBD doivent dialoguer et se synchroniser via le réseau d'interconnexion. Les clusters ont un ratio de performance par processeur compris entre 2000 et 4000 tmpC par processeur alors que les SMP ont des ratios voisins ou supérieurs 8000 tmpC par processeur.
- Un SMP à 24 processeurs présente une efficacité surprenante. Les éléments pouvant expliquer cette différence peuvent se situer au niveau de la taille des caches (16 Mo de L2) et aussi par le fait qu'il s'agit d'une version 64 bits du SGBD doté d'un réglage probablement très bien adapté du SGBD sur la plateforme système.

Comparaison des performances (2)

■ Tendence de la performance SMP et clusters



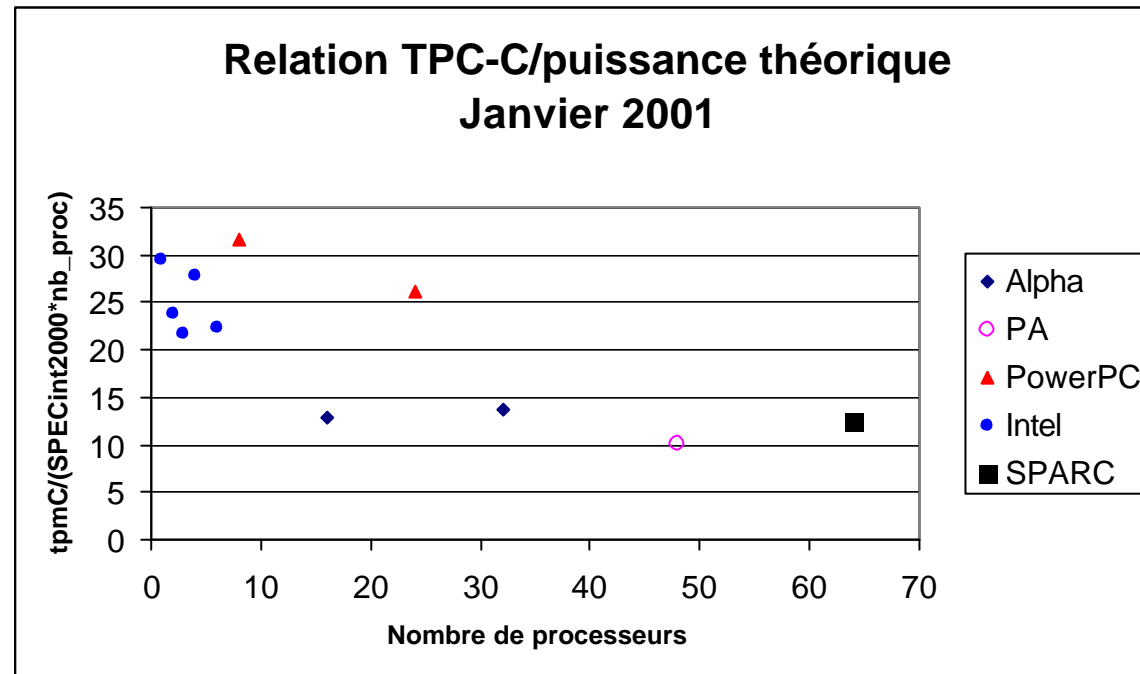
Deux facteurs contribuent à la tendance :

- Scalabilité des architectures multiprocesseurs
- Augmentation de la taille de la base de données proportionnelle à la performance du système

Manque d'information sur ce dernier point pour différencier l'influence de ces facteurs

Comparaison des performances (3)

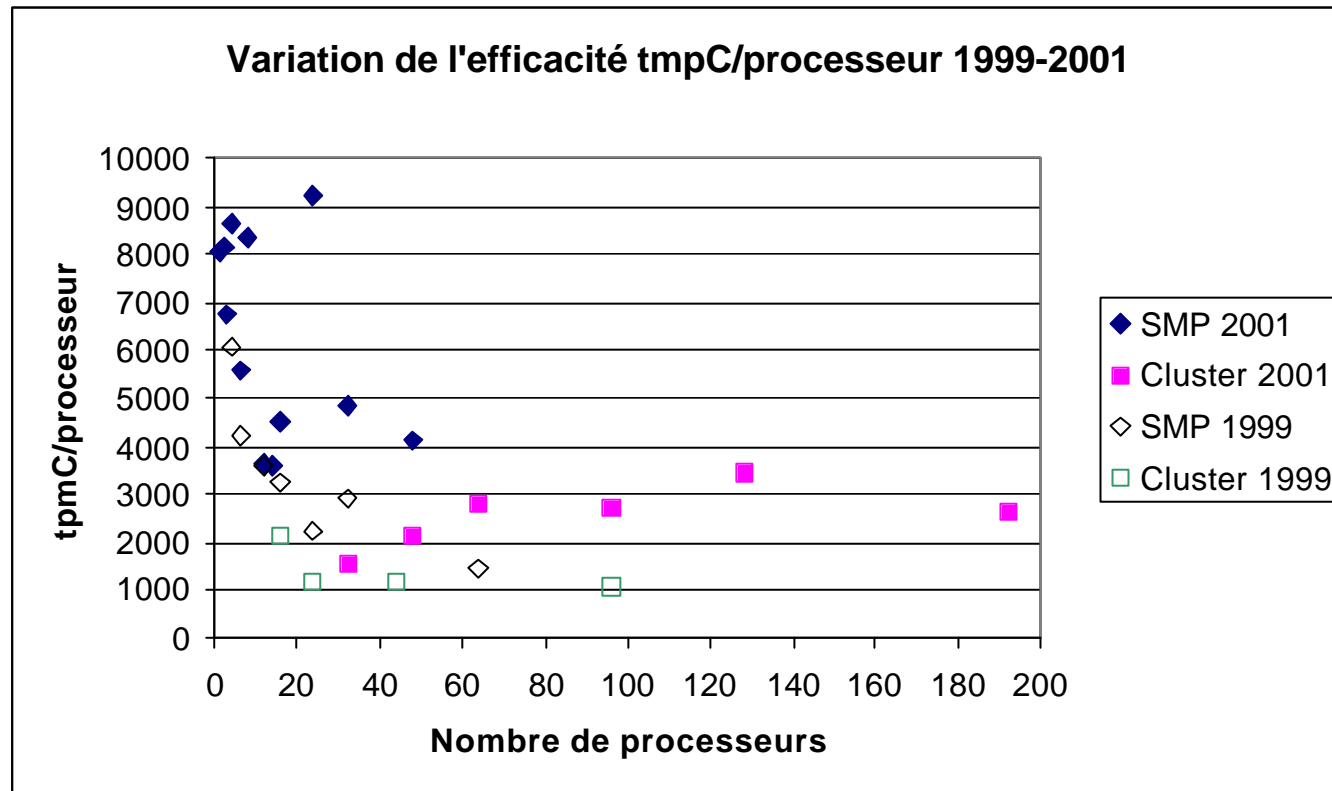
- Relation entre la puissance en transactionnel (TPC-C) et la performance processeur théorique disponible (SPECint2000)



- Interprétation délicate car différents SGBD sont mis en œuvre et de plus sous différentes versions
- L'intensité de la compétition IA-32 dans la zone 4-8 processeurs favorise les progrès
- Les résultats PowerPC à 12 processeurs et à 24 processeurs montrent une très grande efficacité.
- Les résultats pour les systèmes Sequent CC-NUMA à 32 et 64 processeurs n'ont pas été intégrés car ils n'ont pas été mis à jour depuis 1999 (et ils sont donc très en retrait).

Comparaison des performances (4)

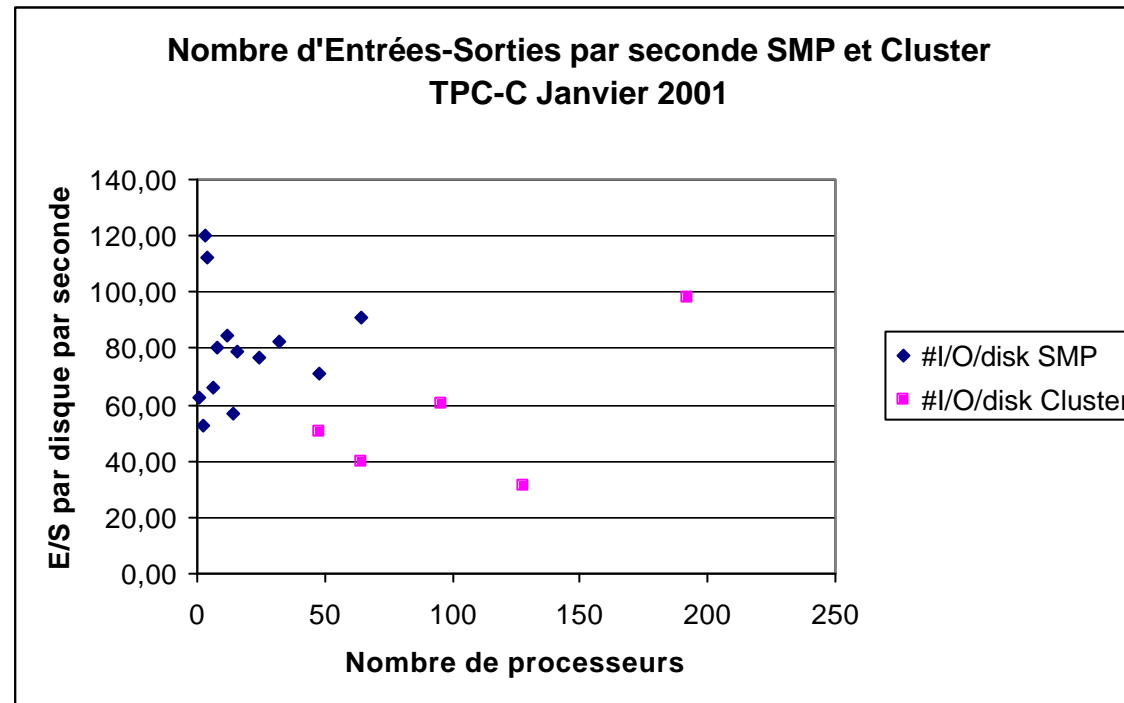
■ Variation de l'efficacité SMP et Cluster entre 1999 et 2001



- Deux facteurs majeurs expliquent cette tendance :
 - Amélioration de la performance des processeurs (x 2 tous les 22 mois en pratique)
 - Amélioration de la performance des SGBD

Comparaison des performances (5)

■ Demandes d'entrées-sorties en environnement TPC-C



- Pour ce calcul, on fait l'hypothèse que chaque tpmC engendre une demande de 30 E/S et l'on s'est servi du nombre de disques utilisés dans la configuration mesurée :
 - Valeur moyenne aux environs de 80 E/S par seconde et par disque
 - Deux systèmes montrent même des valeurs de l'ordre de 120 E/S par seconde et par disque ce qui est voisin de la saturation
- Les configurations moyennes de clusters demandent moins d'entrées-sorties que les SMP
- Compte tenu de la relative faible progression des performances des E/S, l'amélioration des E/S est un point clé

Positionnement des options d'architecture

