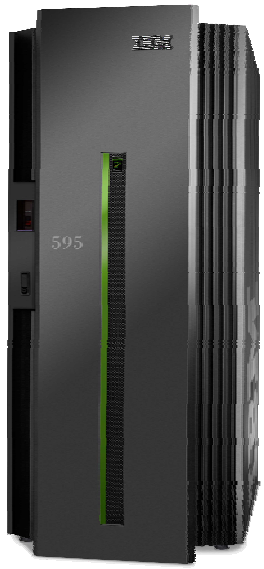
IBM System p, Virtualisation



Mai 2008

Emmanuel Tetreau
tetreau@fr.ibm

alain.lechevalier@fr.ibm.com

Agenda

Intro : Virtualiser pour Consolider

Virtualisation :
- **Ressources CPU, Mémoire, I/O**

Retour d'expérience :
- **Utilisation du VIO**
- **Étude du comportement de l'hyperviseur**

Nouvelles fonctionnalités (Power6, AIX6)

Demo Live Partition Mobility

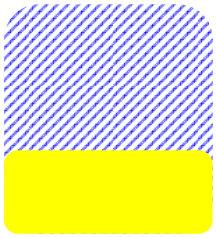
Virtualiser pour consolider

Deux constatations :

1/ $140Md de systèmes non utilisés dans le monde:

- **Imprécision de dimensionnement (marge de sécurité)**
- **Prévision pour les pics de charges**
- **Contraintes techniques (nombre de processeurs mini par ex)**

Taux d'utilisation moyen d'un serveur



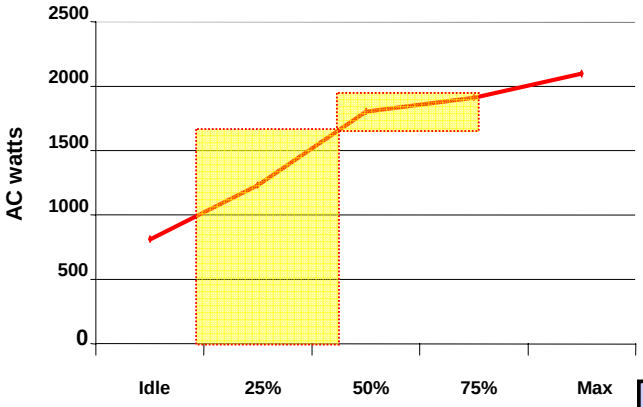
Unix 30 - 40 %

Windows 15 – 20 %

Virtualiser pour consolider

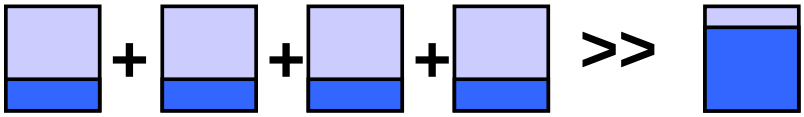
2) L'augmentation de consommation d'énergie est maximum dès les plus faibles taux de charges

Dans cet exemple :
Les premiers 50% coûtent 1700w,
les 50% complémentaires ne coûtent « que » 400w



Les premiers 50% de charge coûtent plus cher en énergie que les 50 suivants

ie : 4 machines chargées à 20% sont moins efficaces en terme de consommation électrique qu'une seule chargée à 80%



Virtualiser pour consolider

→ **Consolider pour mieux utiliser les ressources**

Pour cela, le système doit être :

Puissant, **pour supporter de nombreuse charges de travail**

→ **Processeur Power6**

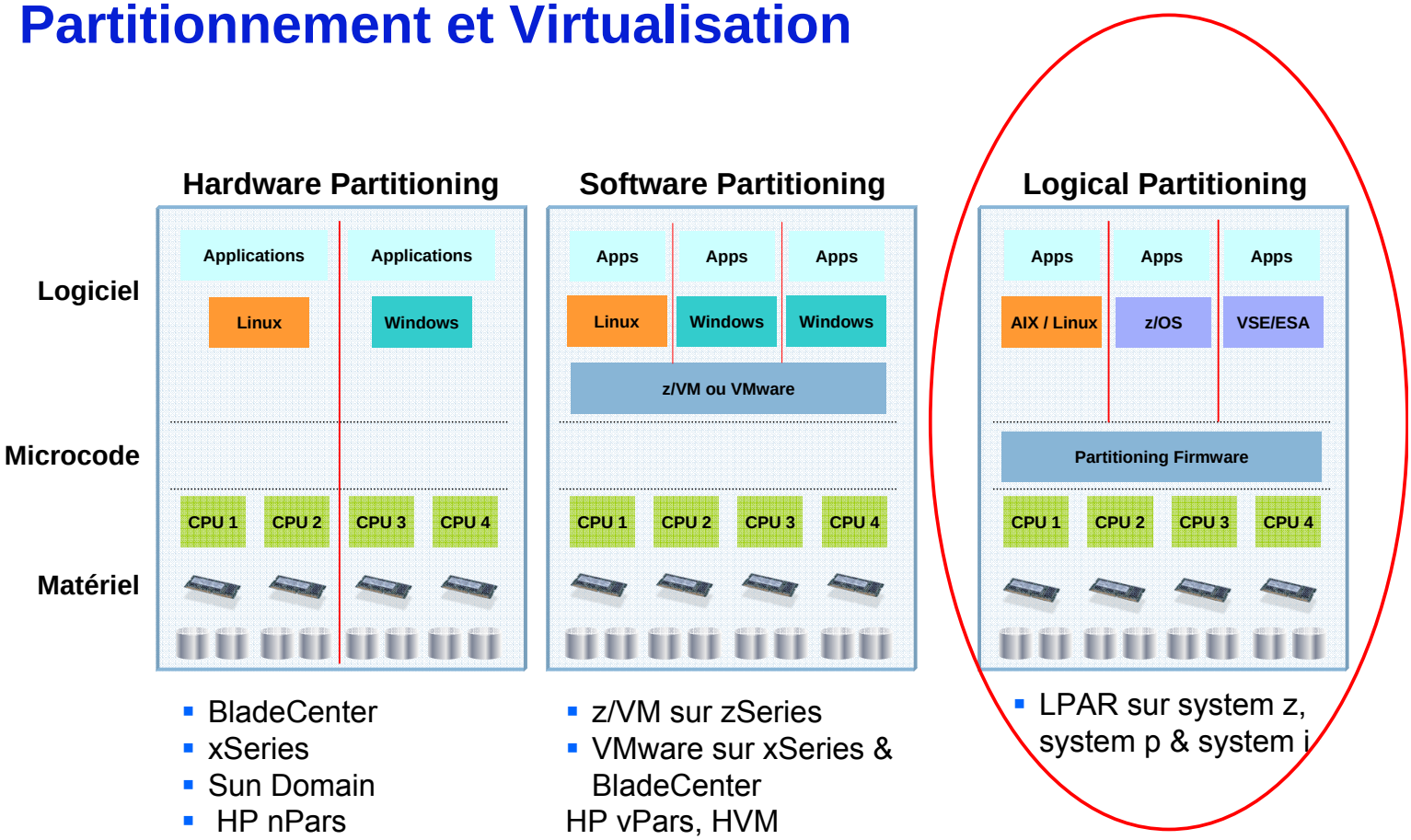
Robuste, **pour assurer un bon niveau de service**

→ **Gamme System p**

Flexible, **pour être capable de dimensionner au plus juste (10ème de proc) et affecter uniquement ce qui est nécessaire à un instant donné.**

→ **Hyperviseur**

Partitionnement et Virtualisation



- BladeCenter
- xSeries
- Sun Domain
- HP nPars

- z/VM sur zSeries
- VMware sur xSeries & BladeCenter

HP vPars, HVM

- LPAR sur system z, system p & system i

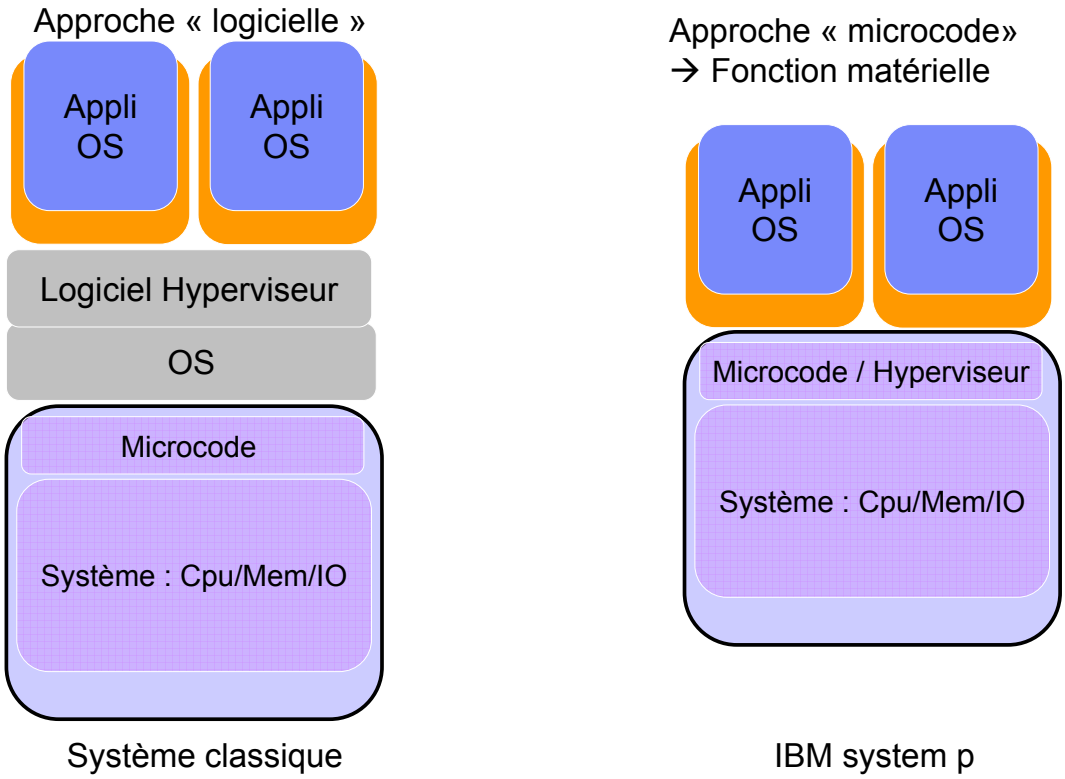
Différentes technologies de partitionnement

Trois technologies de partitionnement :

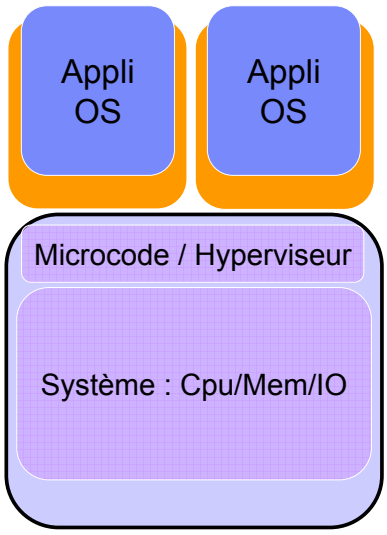
- ***Matériel (hardware)*** **– Les ressources sont allouées aux partitions en mode « un pour un » en respectant les contraintes matérielles. Pas de partages des ressources matérielles.**
- ***Logiciel (software)*** **– Les ressources sont gérées par une couche logicielle. Elles sont regroupées dans un pool de ressources partagées puis présentées aux utilisateurs sous forme de systèmes virtuels (présentation multiple d'une même ressource physique)**
- ***Logique (logical)*** **– Les ressources sont gérées par le microcode du système (firmware) et allouées aux différentes partitions pour créer des environnements virtuels. Les ressources matérielles sont partagées entre les partitions.**

Hyperviseur : 2 implémentations

Virtualisation : Différentes implémentations possibles



p5 Virtualisation : Expérience IBM



IBM system z9 (Mainframe) : 1989 PR/SM

IBM system p5 (Unix/Linux): 2001 (hyperviseur)
2004 (virtualisation)

IBM power6 (Unix/Linux): 2007 (mobilité)

Virtualisation : 40 ans de développement IBM

Power6: Partition Mobility **2007**

Power5: System p Micro-partition, Virtual I/O Server **2004**

Power4 : iSeries sub-processeur LPAR, pSeries LPAR **2001**

LPAR sur AS/400 **1999**

Début du développement LPAR pour Power **1997**

1990 ES/9000

1987 PR/SM, EMIF, MHPG

1985 Start Interpretive Execution (SIE)

1973 S/370 Model 158 et 168 – VM Assists

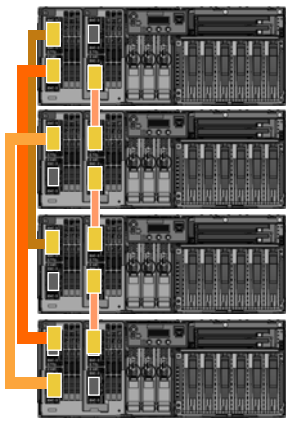
1972 VM/370

1967 CP-67, CMS Timesharing

Virtualisation : Architecture cohérente

Tous les niveaux participent …

→ Approche Paravirtualisation



Systèmes d'Exploitation

- Les OS peuvent *redonner* les ressources processeur inutilisées

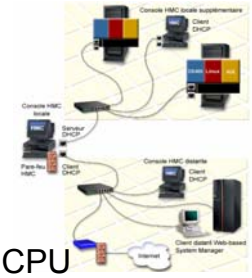
Hyperviseur (Firmware)

- Assure l'interface entre le matériel et sa représentation virtuelle

Matériel (Power5 – Power6)

- Le processeur génère les intervalles de temps pour l'hyperviseur

Virtualisation : Avantages

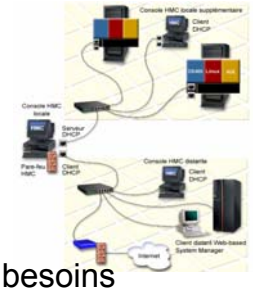


Fiabilité / Sécurité / Efficacité

- Code de l'Hyperviseur maîtrisé – Expérience z9 (mainframe)
  - Relativement peu de ligne de code : fiabilité et efficacité (surconsommation CPU limitée)
- Hyperviseur intégré au système – Pas d'accès utilisateur, pas de code inutile (OS …)
  - Sécurité garantie (pas de virus, rootkit etc.).
- Étanchéité totale des partitions – Tables mémoire et E/S gérées par l'Hyperviseur
  - Fiabilité de l'ensemble du système.

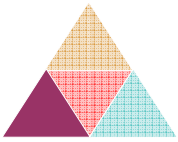
- Hyperviseur toujours actif dans le système
  - Pas de validation spécifique des logiciels
  - Données de performance (bench) fournies avec l'hyperviseur actif
- Disponible depuis Août 2004 – plusieurs centaines de milliers de systèmes installés
  - Retour d'expérience important
  - Utilisé dans des environnements de production « lourds »

Virtualisation : Avantages



Simplicité

- Une seule méthode de partitionnement – Suffisamment souple pour répondre aux besoins
- Pas de contrainte de configuration ni d'interdépendance
  - pas de lien entre : cpu - mémoire - slot E/S
- Création rapide d'environnements
- Automatisation des actions (par ex interface avec le système de cluster HACMP)
- Interface graphique / Ligne de commande (scripts)
- Virtualisation des ressources CPU et E-S
- Entièrement dynamique (modification CPU - Mem – E/S, à chaud)



Hyperviseur intégré : Capacités des systèmes

De base

Système « simple »

Système partitionné
Processeurs entiers
Mémoire fractionnée
I/O physiques

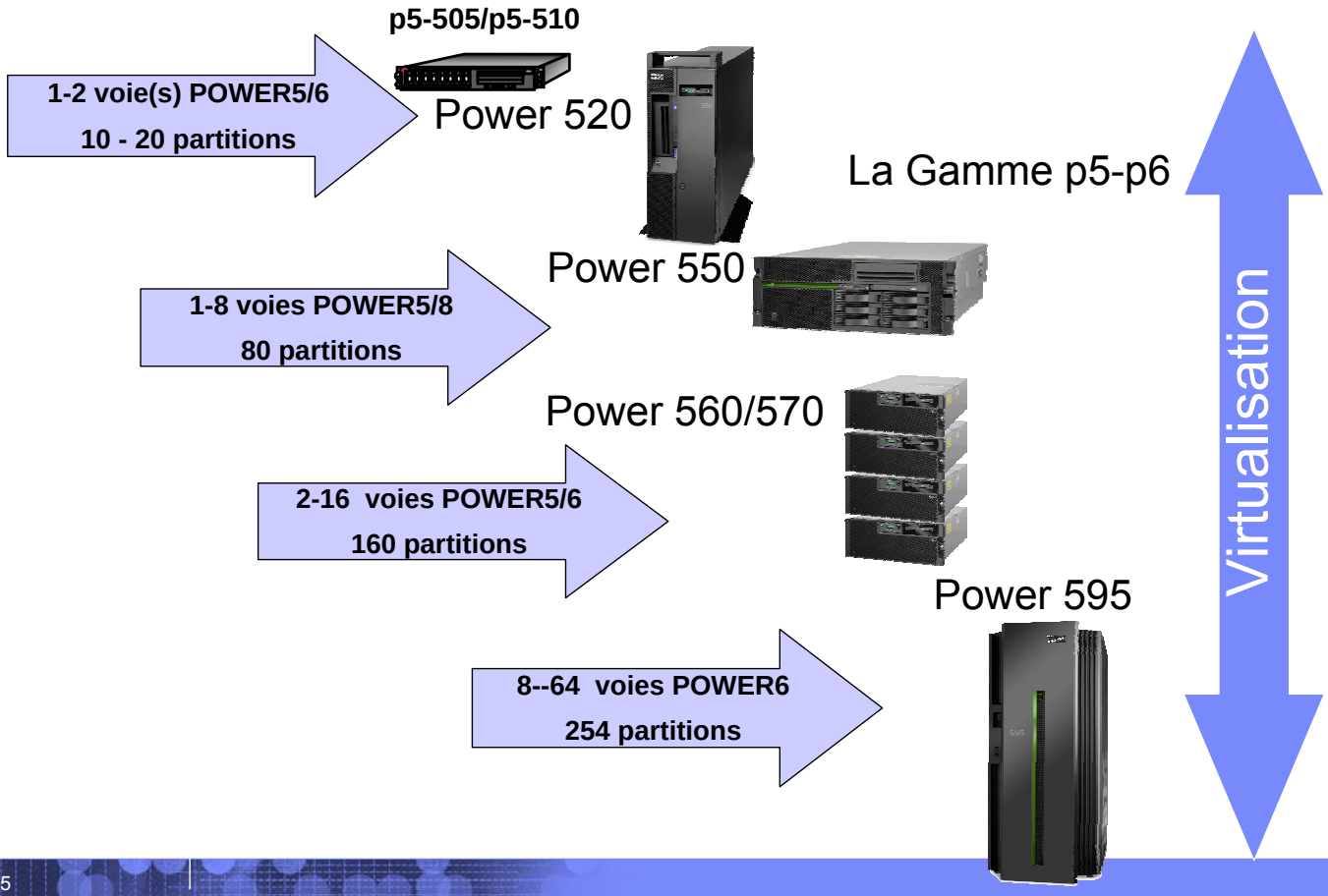
Console HMC

Virtualisation activée

Système virtualisé
Processeurs fractionnés
Mémoire fractionnée
I/O physiques et virtuelles

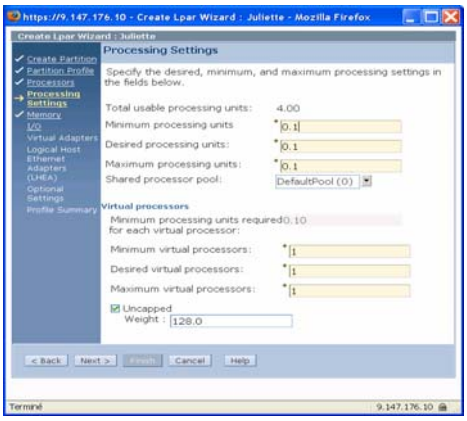
Console HMC

Disponible sur tout serveur Power5 et Power6

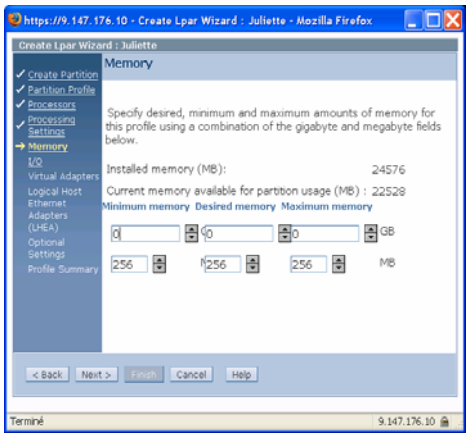


Virtualisation : Micropartitionnement

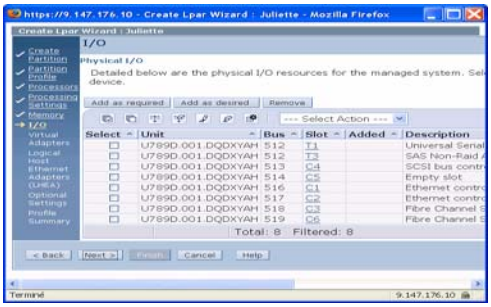
Création d'une machine virtuelle : juste 3 étapes



Affectation ressources CPU



Affectation mémoire



Affectation E/S

Affectation des ressources processeur

Virtualisation : Ressource Processeur

Les processeurs du systèmes sont dans un « pool » partagé

- **L'unité de puissance processeurs est le CE (Capacity Entitlement)**
- **1 processeur physique = 1 CE**
- **Les micro partitions reçoivent chacune un nombre de CE représentant des fractions de processeurs.**

Par exemple un pool de 6 processeurs offre 6 CE à partager entre les micro partitions
Un CE est divisible en centièmes

Pour une partition :

- **Minimum : 0,1**
- **Maximum : nombre de coeurs dans la machine (jusqu'à 64)**
- **Incrément : 0,01**

Micro-partitions : CE Capacity Entitlement

- **Partage des CE**
  - Chaque partition reçoit un CE égale au minimum à 1/10 de processeur physique.
  - Incréments par 1/100 jusqu'à la taille maximum du pool
- **mais le système d'exploitation ne connaît que la notion de processeur ….**
  - Une partition va être constituée de 1 ou plusieurs processeurs virtuels qui *portent* les ressources configurées.

- **L'affectation des ressources processeur est indépendante de l'affectation des ressources mémoire ou de slot I/O.**

Micro-partitions : Exemple

- 3 processeurs dans le pool
- Capacité d'exécution (CE) du pool = 3.00
- Chaque partition peut recevoir une capacité d'exécution entre 0.10 et 3.00
- La somme des CE des partitions doit être inférieure ou égale à 3.00 (CE du pool)
- Une partition est constituée de processeurs virtuels qui se partagent la capacité d'exécution.
  - **Partition 1** - *Database* : **CE=1.80**, VP = 3 (0,60 par processeur)
  - **Partition 2** - *Applications* : **CE=0.80**, VP = 2 (0,40 par processeur)
  - **Partition 3** - *Env de test* : **CE=0.20**, VP = 1 (0,20 par processeur)

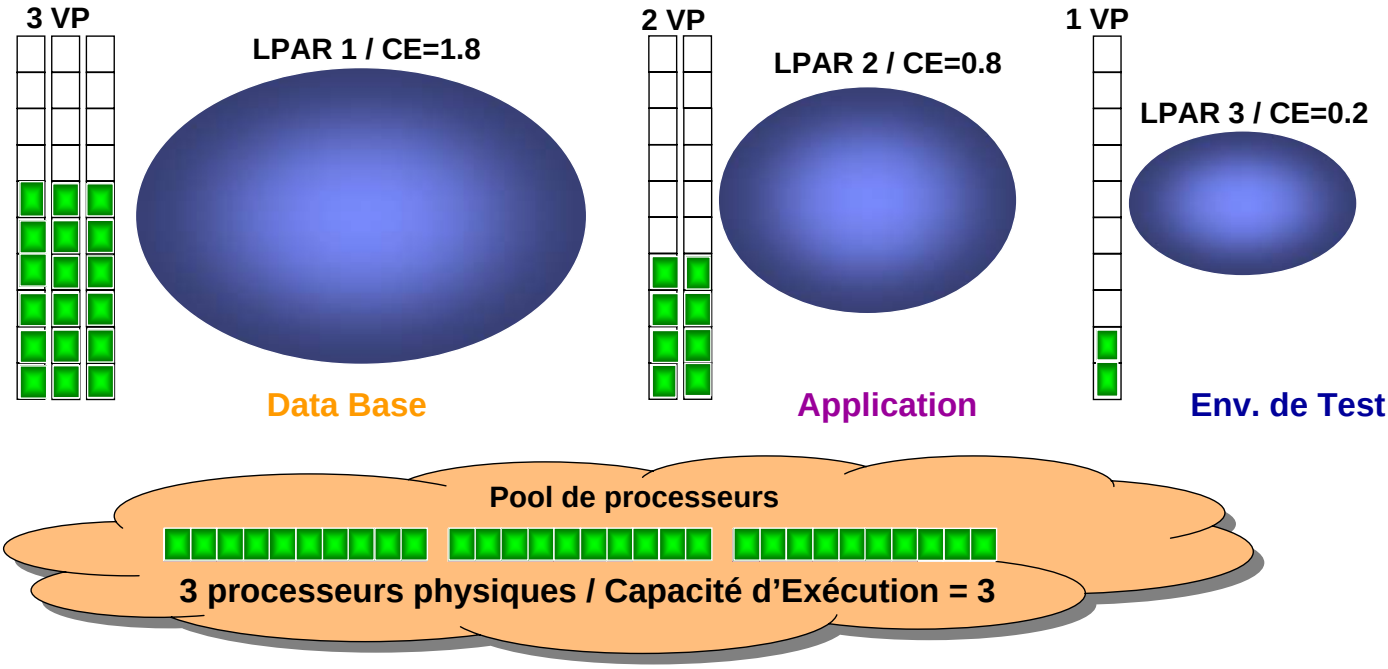
- Total CE=2.80, Total VP = 6 - reste 0.20 CE disponible

Micro-partitioning : CE Capacité d'Exécution

3 processeurs dans le pool - **C**apacité d'**E**xécution du pool = 3.00 (3x10x0,1)

Partition 1 : **Data Base** CE=1.80, Virtual Proc = 3 (0,60 par processeur)
Partition 2 : **Application** CE=0.80, Virtual Proc = 2 (0,40 par processeur)
Partition 3 : **Env. de Test** CE=0.20, Virtual Proc = 1 (0,20 par processeur)

Total CE= 2.80, Total Virtual Proc = 6 (reste 0.20 CE disponible)

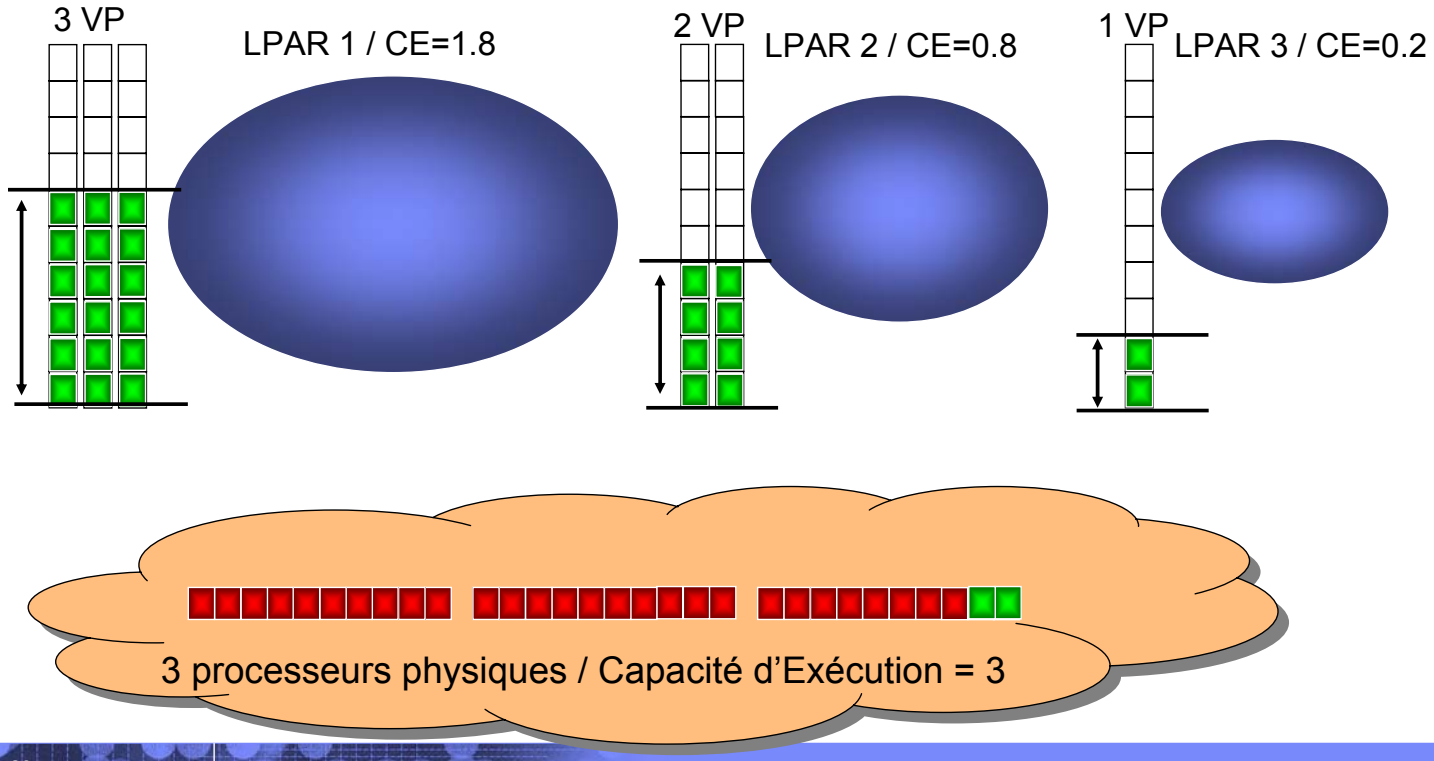


Optimisation de l'utilisation des ressources

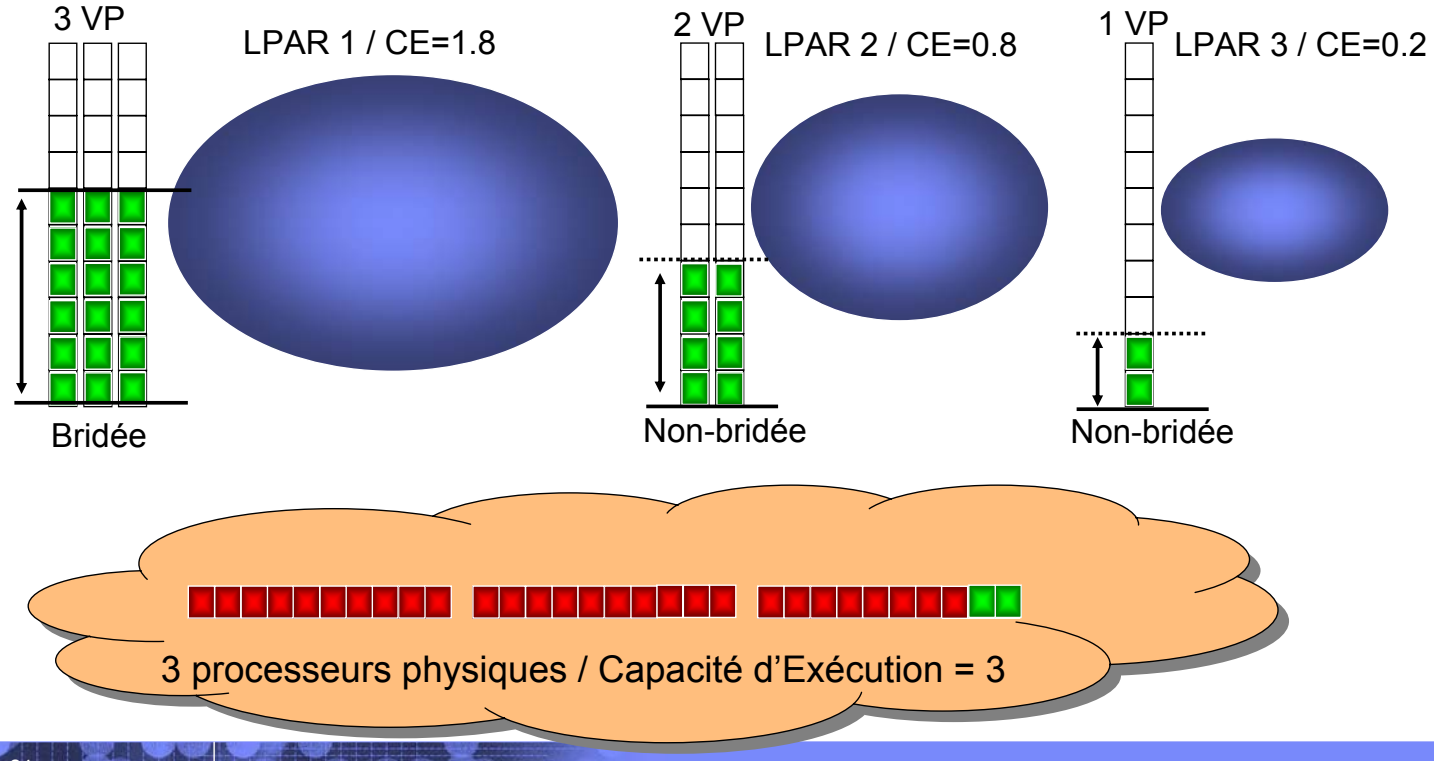
Une partition peut être *bridée* ou *non-bridée*

- **Bridée / Non-bridée (Capped / Uncapped)**
  - **Bridée: Les partitions sont strictement limitées à leur valeur de CE maximum définie.**
  - **Non-bridée: une partition peut utiliser des ressources** disponibles **dans le pool, à concurrence du *remplissage* des processeurs virtuels.**
- **Priorité (Capacity weight)**
  - **Prioritisation de l'affectation des ressources supplémentaires entre partitions.**
  - **Valeur 0-255**

Micro-partitions: Bridées (*Capped*)



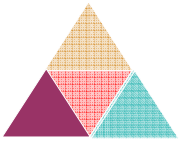
Micro-partitions: Non Bridées (Unc*apped*)



Participation du Système d'Exploitation

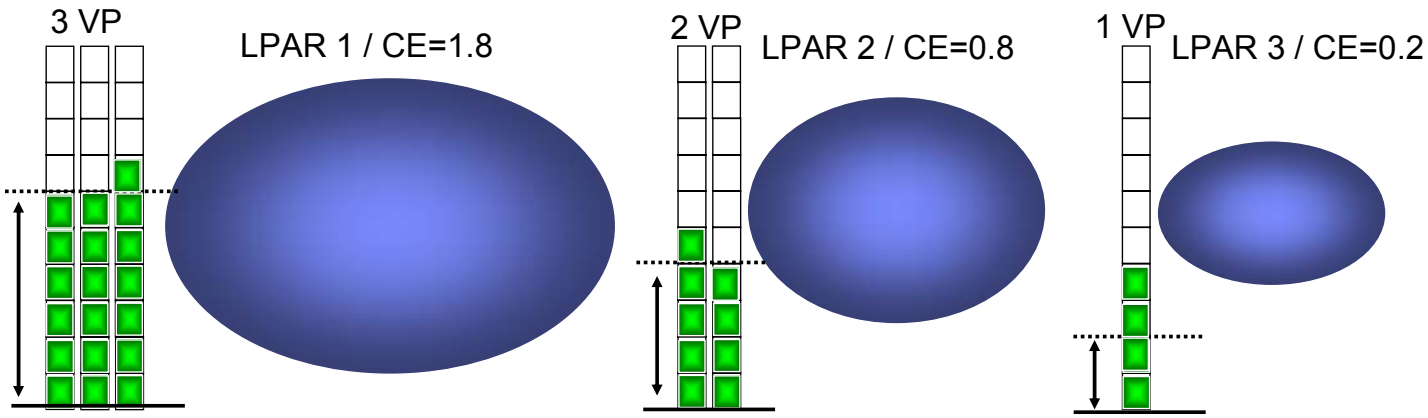
La virtualisation des processeurs permet de mieux utiliser les ressources

- **Si une partition n'a pas besoin de ressource à un instant donné, le système d'exploitation rend (cède) son temps CPU**
  - **Evite de perdre de la ressource processeur**
    - **Comme par exemple une partition utilisant son CE à attendre une fin d'E/S**
  - **Permet une meilleure utilisation du pool**
- **Le temps peut être affecté à un autre processeur virtuel de la même partition si besoin**
- **En retour, le processeur virtuel est potentiellement réactivable dans le même intervalle de temps si nécessaire**

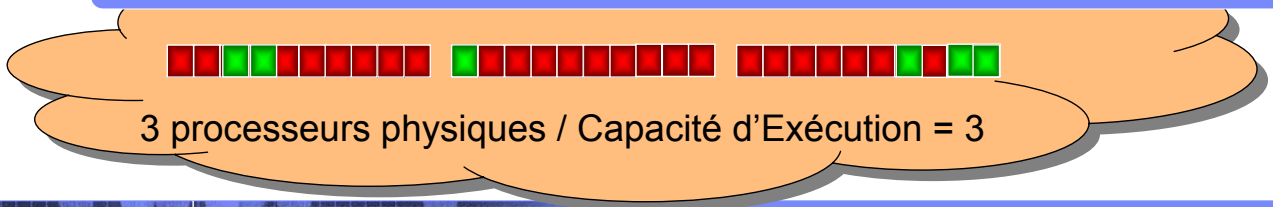


Cet ajustement se fait 100 fois par seconde !!

Micro-partitions: Ajustement des puissances



La virtualisation permet de faire varier en temps réel et d'une façon transparente la « puissance d'un processeur »



Répartition des ressources mémoire

Virtualisation : Ressources mémoire

- La mémoire physique du système est répartie entre les partitions.
- L'hyperviseur assure l'étanchéité totale entre les partitions
- Chaque partition va recevoir une fraction de la mémoire physique
- La gestion de la mémoire virtuelle (pagination) est supportée dans les partitions

Pour une partition :

- Minimum : 128MO (256MO pour AIX)
- Maximum : Taille de la mémoire du système (jusqu'à 2TO)
- Incrément : 16Mo

Répartition des ressources Entrées / Sorties

Virtualisation : Entrées / Sorties

- **Chaque partition reçoit des contrôleurs d'entrées / sorties réels et/ou virtuels**
- **Contrôleurs réels : On affecte à la partitions des slots PCI présents dans le système. Il n'y a pas de contrainte de nombre ou de localisation.**
- **Contrôleurs virtuels : En utilisant une partition VIOS (Virtual I/O server), on peut mutualiser les contrôleurs physiques entre plusieurs partitions.**

Pour une partition (contrôleurs réels) :

- **Minimum : 0 contrôleur réel**
- **Maximum : Tous les contrôleurs du système**
- **Incrément : 1 slot PCI**

Virtualisation : Entrées / Sorties

Virtual I/O Server (VIOS)

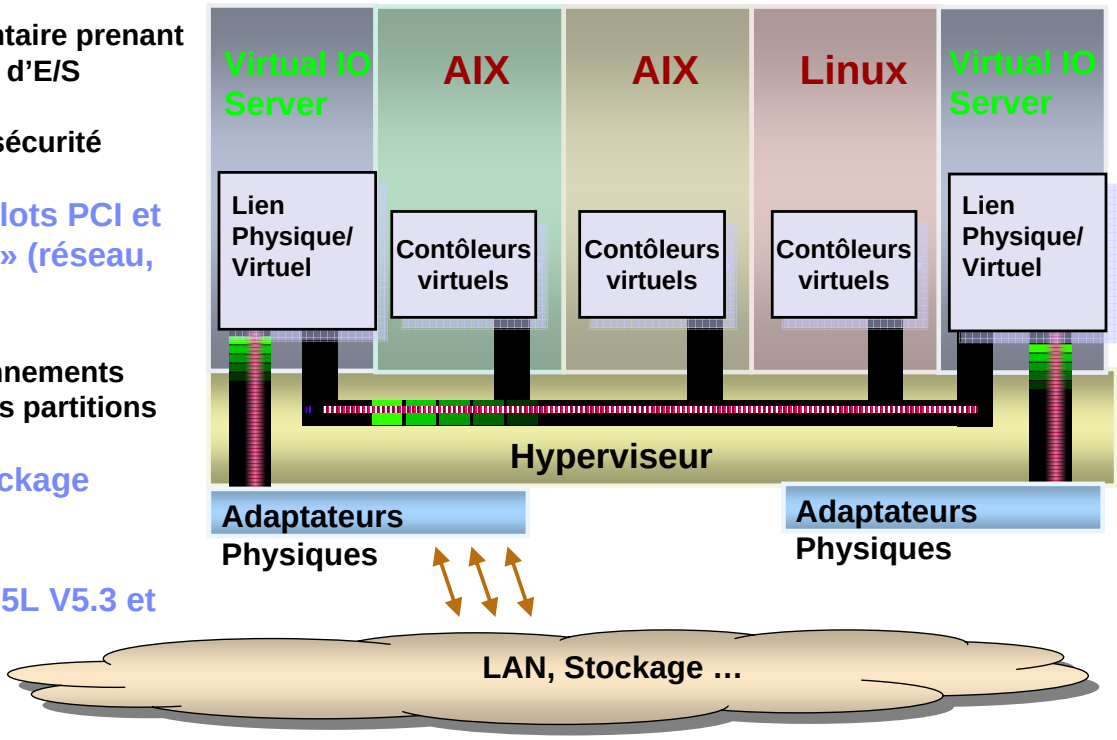
- VIOS : partition supplémentaire prenant en charge la mutualisation d'E/S physiques
- Peut être doublée pour la sécurité

Objectif : économiser des slots PCI et des ports sur les « switchs » (réseau, SAN).

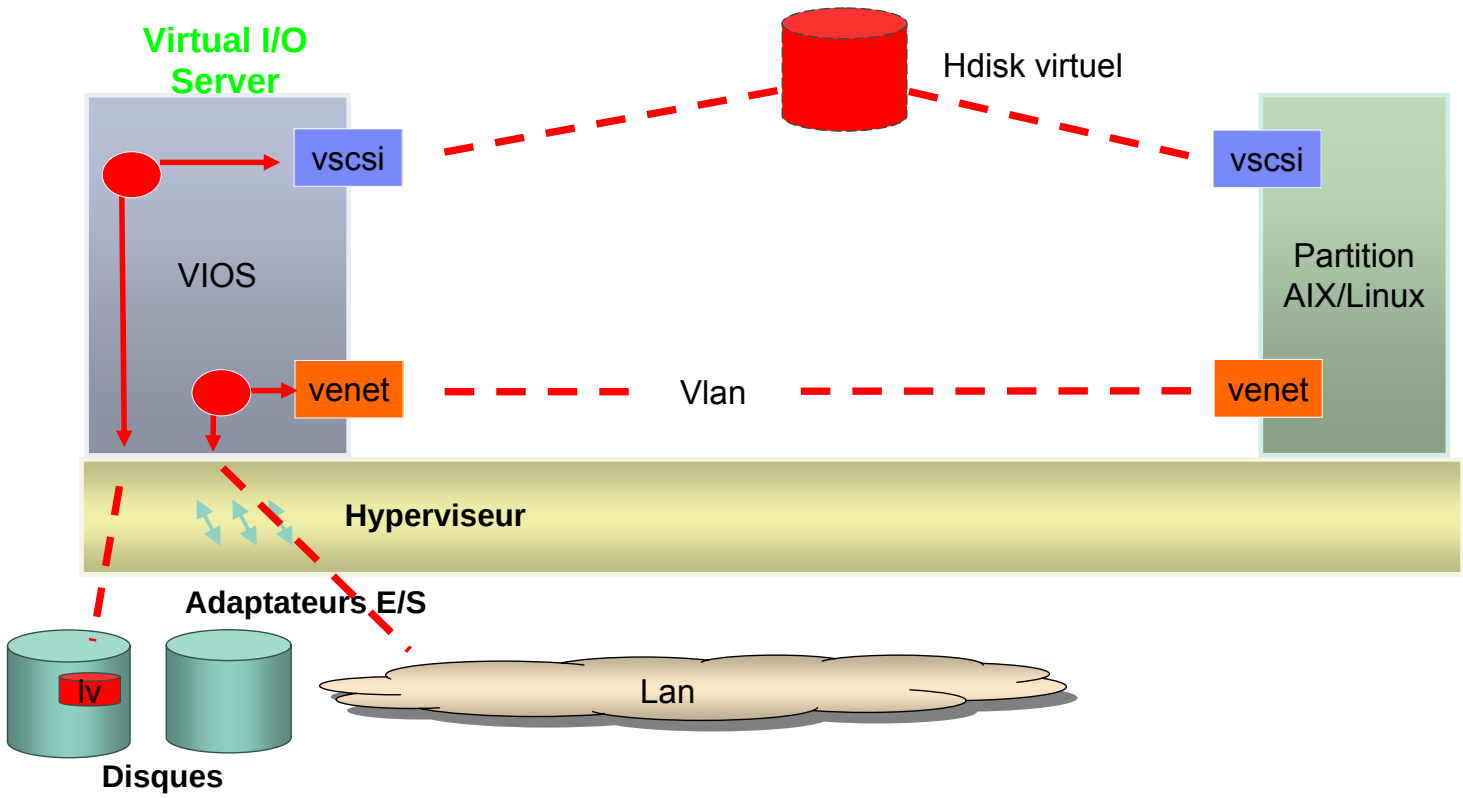
- Important dans les environnements comportant de nombreuses partitions

Virtualisation : Réseau, Stockage

Supporte les partitions AIX 5L V5.3 et Linux

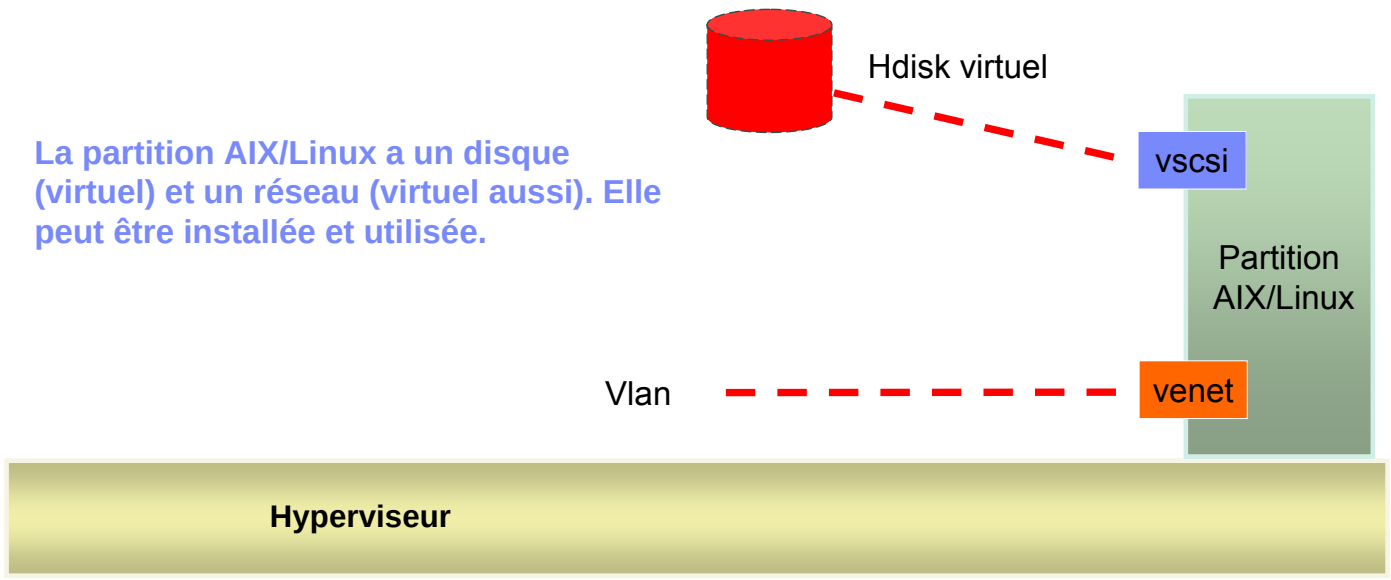


Virtual I/O Server : Principe



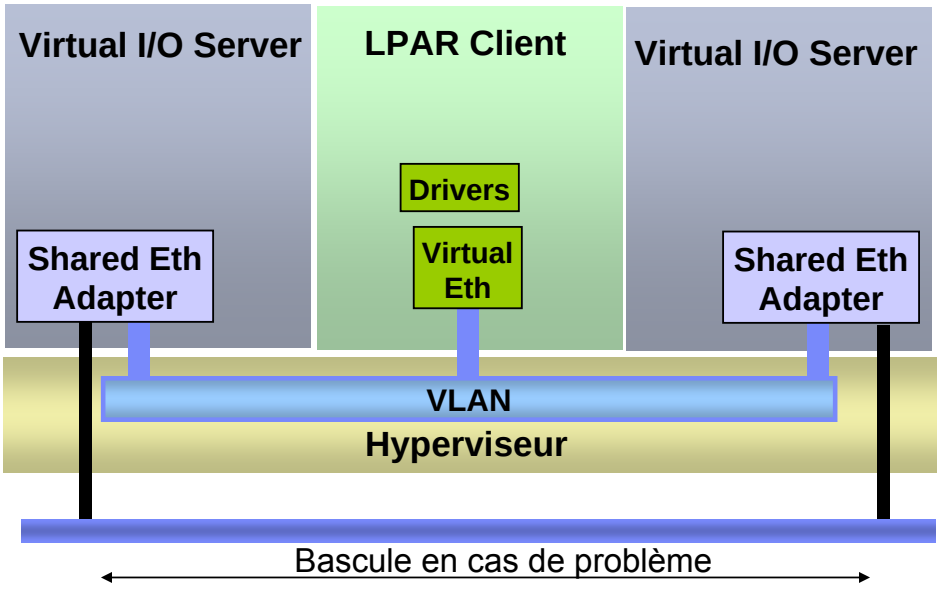
Virtual I/O Server : Principe

La partition AIX/Linux a un disque (virtuel) et un réseau (virtuel aussi). Elle peut être installée et utilisée.



VIOS: Sécurisation accès Ethernet

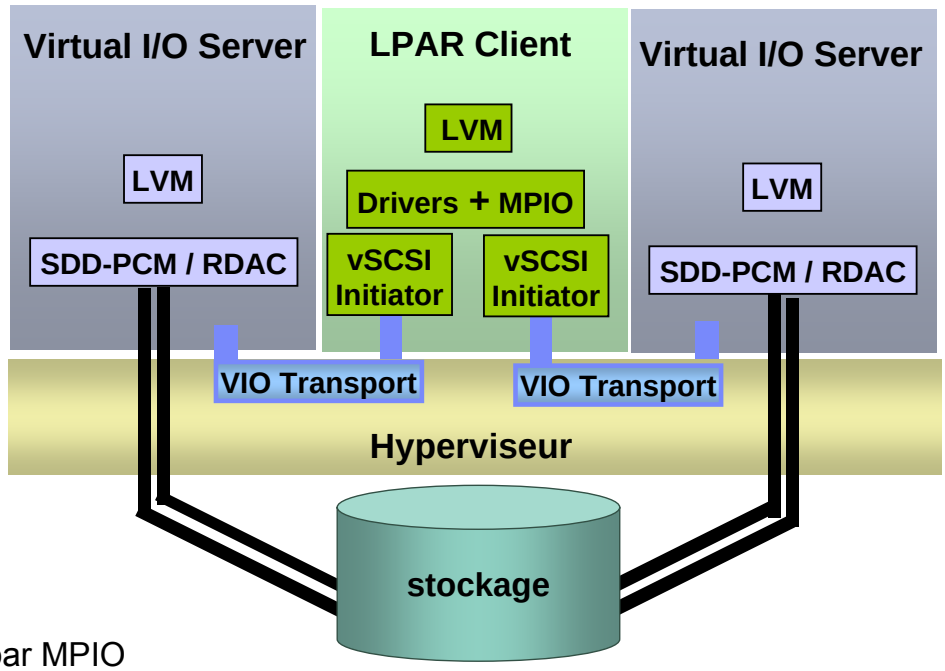
Sécurisation par le VIOS (1.2)



Protection contre un arrêt du VIOS, perte d'un lien Ethernet
Sécurisation de l'adaptateur et/ou du disque effectuée par les partitions VIO

VIOS: Sécurisation accès stockage

Sécurisation du VIOS et des adaptateurs



Client: Protection par MPIO

Dynamic LPAR et Micro-partitions

Le micro partitionnement est entièrement dynamique

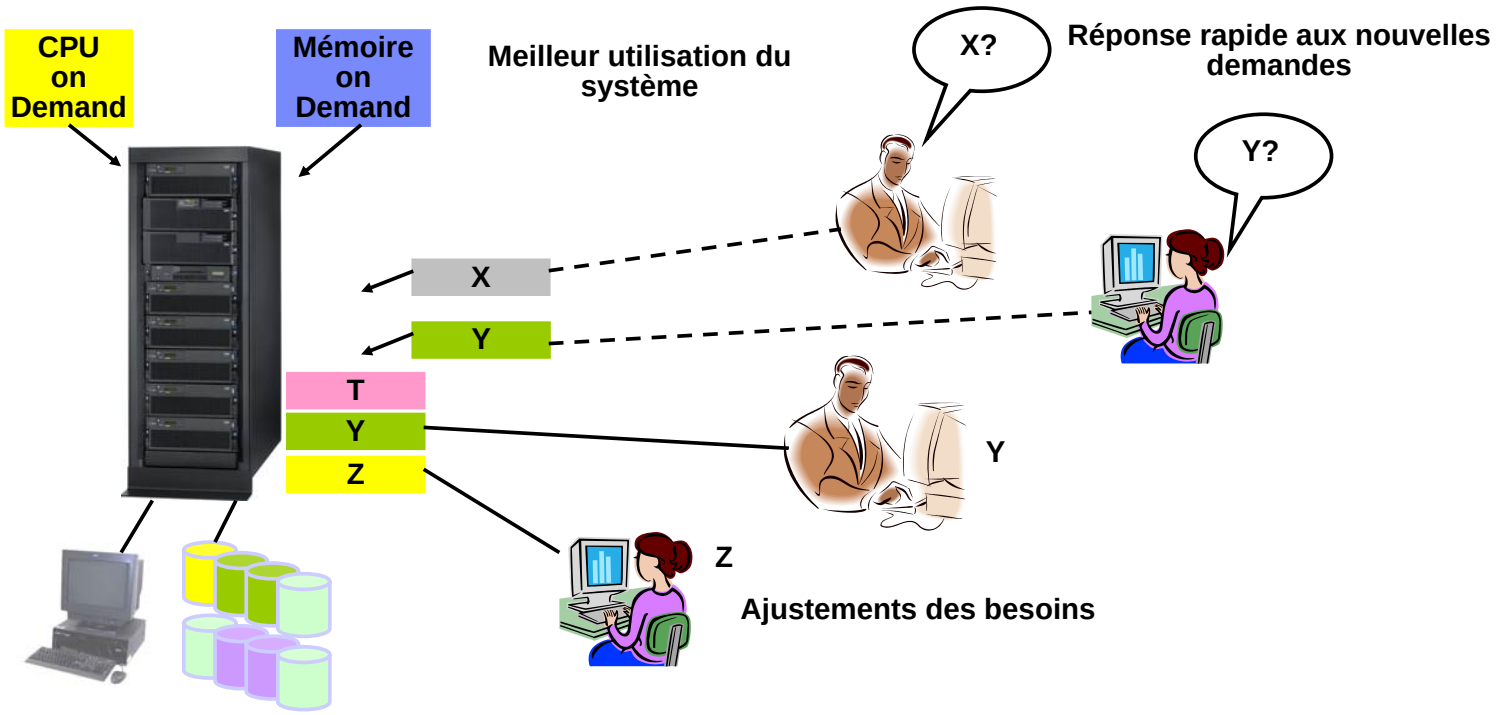
- **Les processeurs réels peuvent être ajoutés ou retirés du *pool* partagé.**
  - **Utilisation du COD**
- **L'affectation des ressources processeurs (CE) peut-être modifiée**
- **Les processeurs virtuels peuvent être ajoutés ou retirés d'une micro-partition,**
- **La mémoire peut être ajoutée ou retirée**
- **Les slots (réels ou virtuels) peuvent être ajoutés ou retirés**

Grande souplesse et simplicité d'administration

Virtualisation des besoins

Approche “virtuelle” : Répondre aux besoins des utilisateurs

Principe : Dématérialisation des besoins.



Virtualisation : Retour d'expérience

Virtualisation : Utilisation du VIO

L'adoption du VIO c'est faite en douceur…

- **La technologie date de 2004.**
- **D'abord introduite dans les environnements de tests / dev**
- **Maintenant utilisée dans les environnements de production lourds**
- **Nos grands clients sont moteur dans cette adoption; ils ont une bonne expérience du fonctionnement des VIOs dans leur environnement.**
- **Réduction des besoins de contrôleurs et de connexions (ports réseau, SAN …)**
- **Nécessaire pour bénéficier des nouvelles fonctionnalités**

Virtualisation : Comportement de l'Hyperviseur

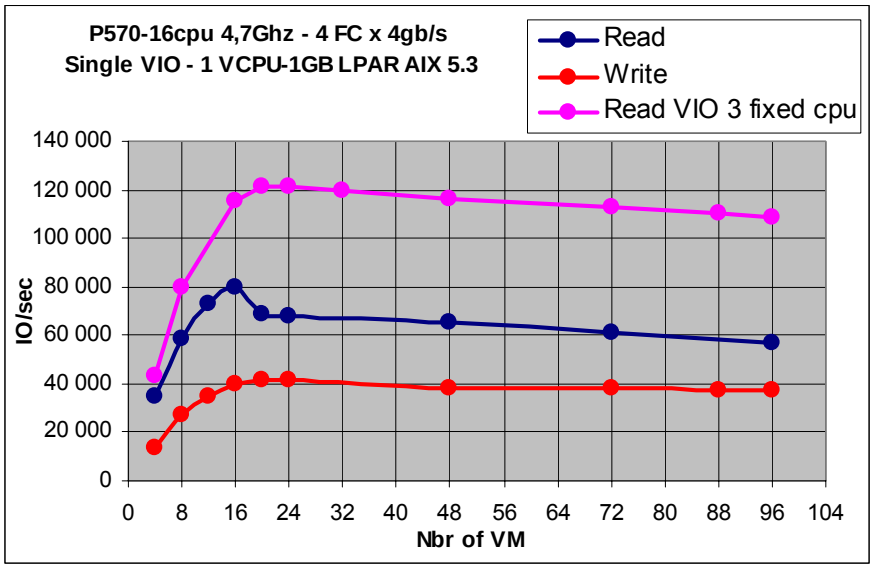
Résultats d'un benchmark interne IBM fait à Montpellier en Décembre 2007

Système : p570 16 cœurs Power6 @ 4.7 Ghz

- **96 partitions en AIX 5.3**
- **1 Virtual I/O Serveur (v1.4)**
- **4 cartes FC 4Gbs**
- **8 Ports Gigabits Ethernet**

Virtualisation : Quelques chiffres

Scalabilités E/S : Disques



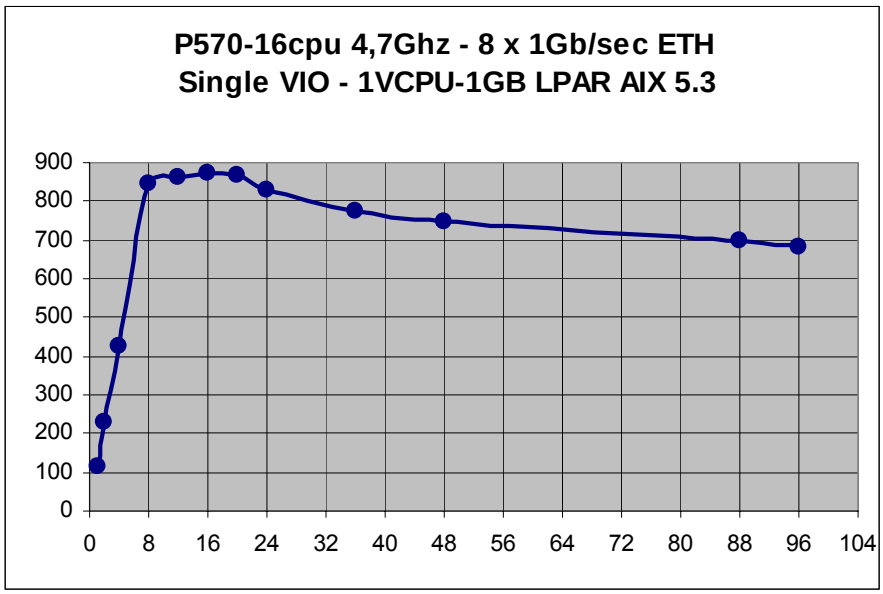
Jusqu'à 96 partitions

Jusqu'à 120 000 I/O par seconde

CPU dédiés plus performants pour les très hauts débits (moins de latence)

Virtualisation : Quelques chiffres

Scalabilités E/S : Réseau



Jusqu'à 96 partitions

1 Ethernet virtuel par partition

Jusqu'à 900 Mo par seconde

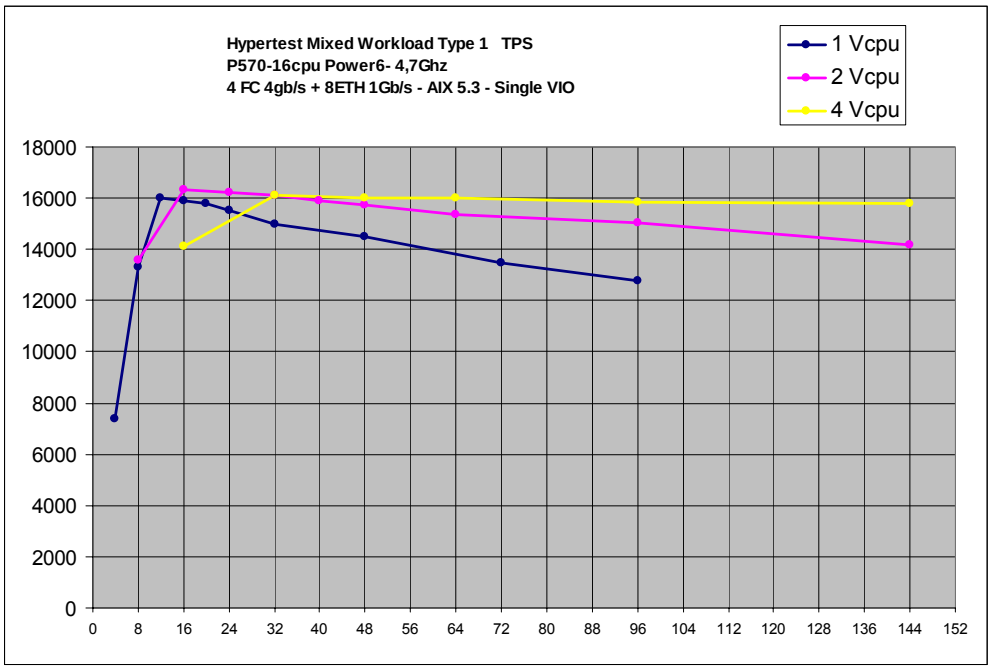
Virtualisation : Quelques chiffres

Scalabilité workload mixte :

- **Calcul CPU (110 x calcul entier)**
- **Accès Mémoire (100 x 8ko)**
- **Accès réseau (10 x 1ko)**
- **Accès I/O (1 x 8ko)**

Virtualisation : Quelques chiffres

Scalabilité workload mixte : Efficacité de l'Hyperviseur



1 à 4 Virtual CPU par partition

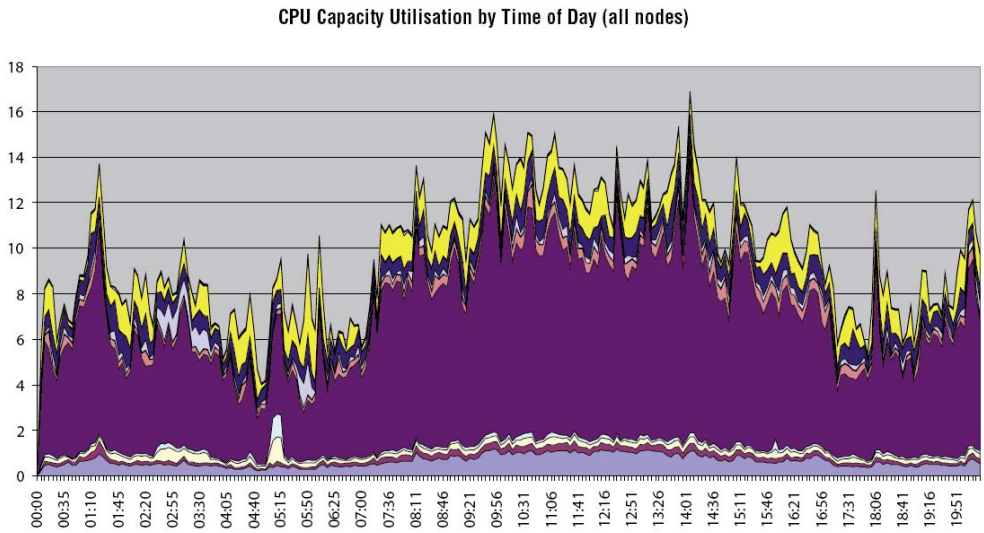
- **36 partitions à 4 vcpu (144)**
- **72 partitions à 2 vcpu (144)**
- **96 partitions à 1 vcpu (96)**

Virtualisation : Référence SAP / DB2

Équipementier automobile : 50000 employés

Architecture entièrement virtualisée sur 2xp5-595 (préparation à LPM)

4 VIOS (2 prod, 2 non-prod) par machine



41 Bases de données

4 TO pour la base principale

DS8300

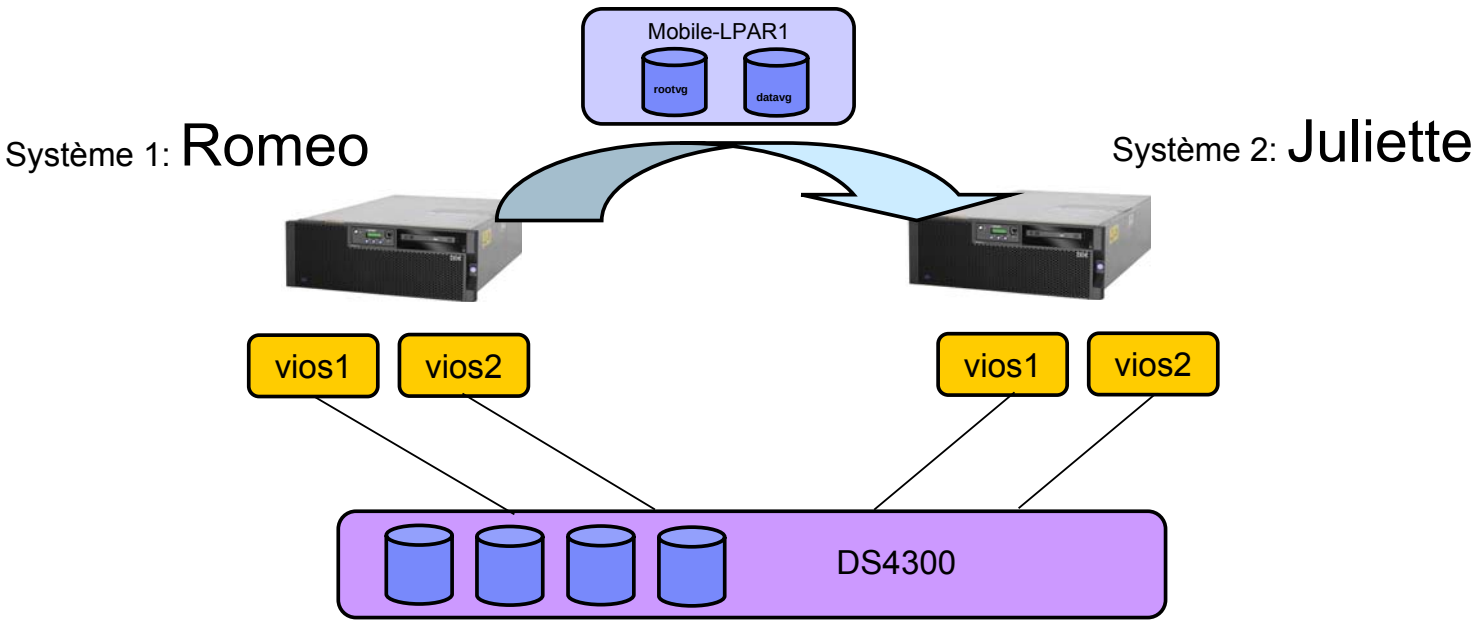
Disques: 20 000 I/O par secondes

Réseau: 50 MO / seconde

Consommation des VIOS <= 2 Cpu

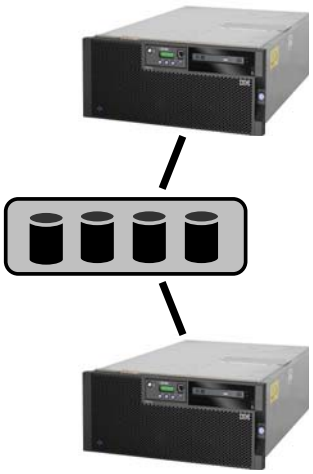
Démonstration Partition Mobility

Demo PM : Systèmes p6



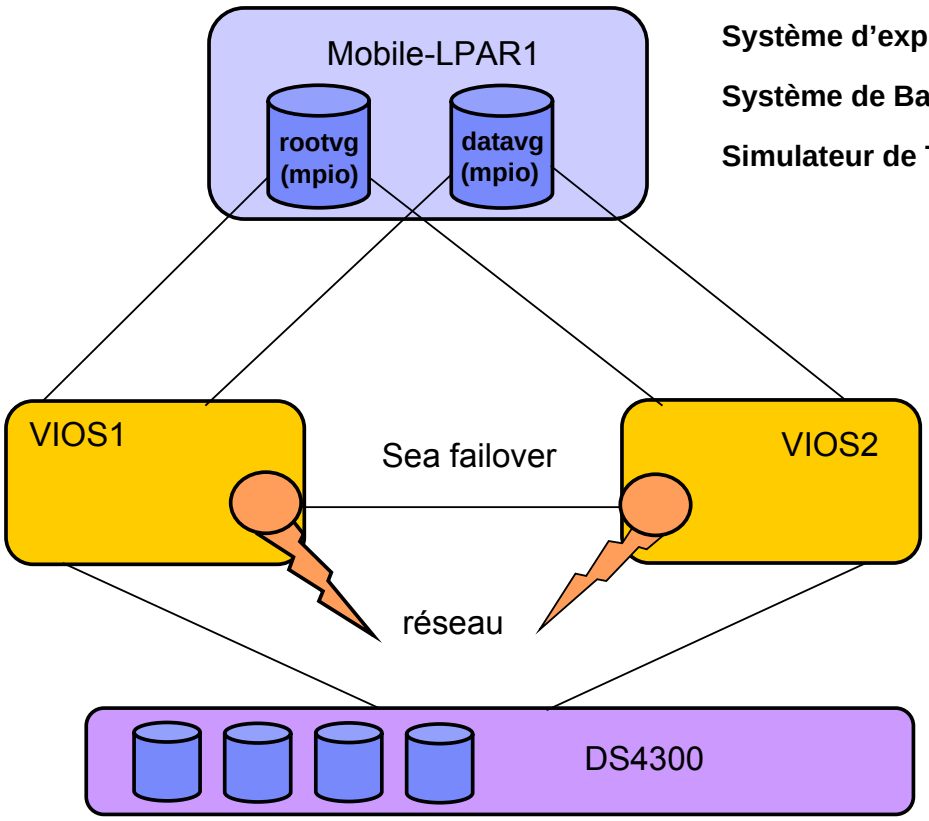
Demo PM : Systèmes p6

Système 1: Romeo



Système 2: Juliette

Demo PM : Configuration de la partition



Système d'exploitation : AIX 5.3 TL6

Système de Base de données : DB2 V9.1

Simulateur de Transaction : Websphere 6.1

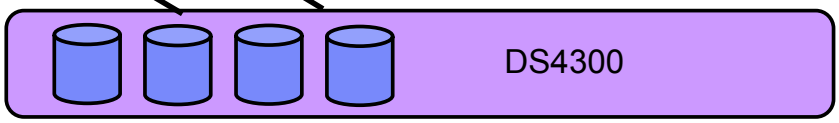
Demo PM : Les étapes

Système 1: Romeo

Système 2: Juliette

vios1 vios2

vios1 vios2



Demo PM : Les étapes (1/5)

Vérifications :

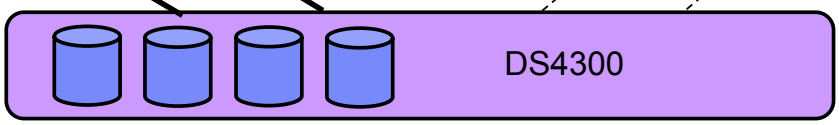
Ressources disponibles sur la cible
Accès au stockage

Système 1: Romeo

Système 2: Juliette

vios1 vios2

vios1 vios2



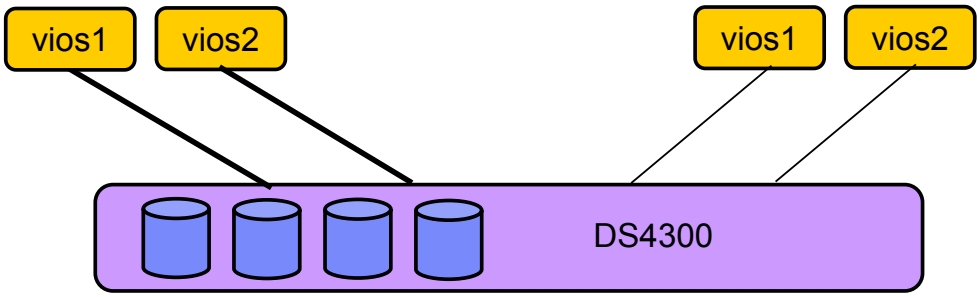
Demo PM : Les étapes (2/5)

Création :

Partition cible
Liens stockage

Système 1: Romeo

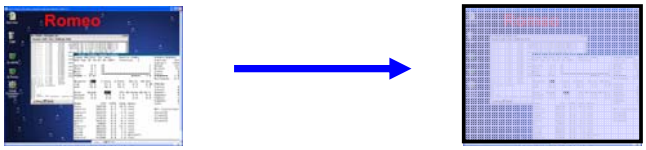
Système 2: Juliette



Demo PM : Les étapes (3/5)

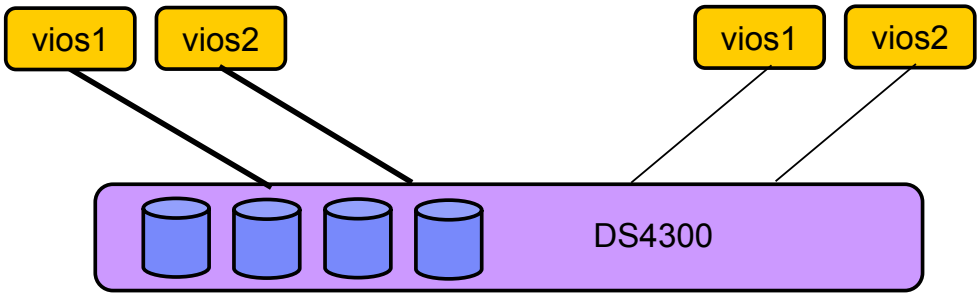
Copie :

Mémoire source → cible



Système 1: Romeo

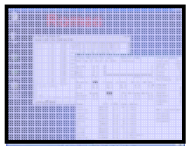
Système 2: Juliette



Demo PM : Les étapes (4/5)

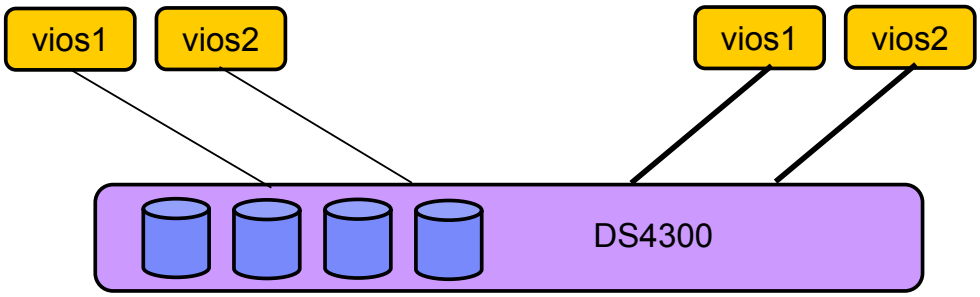
Activation :

Gel de 2 secondes environ



Système 1: Romeo

Système 2: Juliette



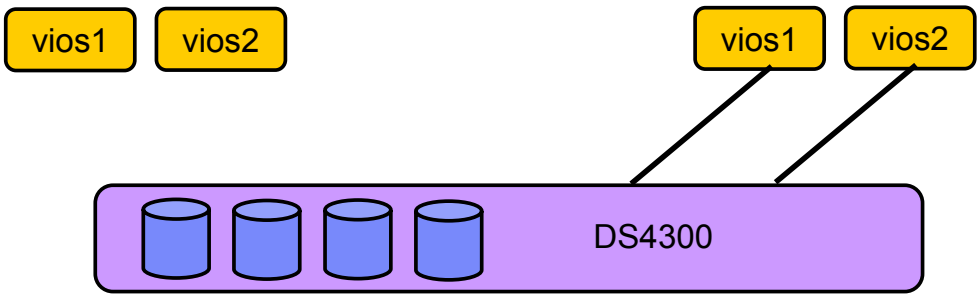
Demo PM : Les étapes (5/5)

Nettoyage :

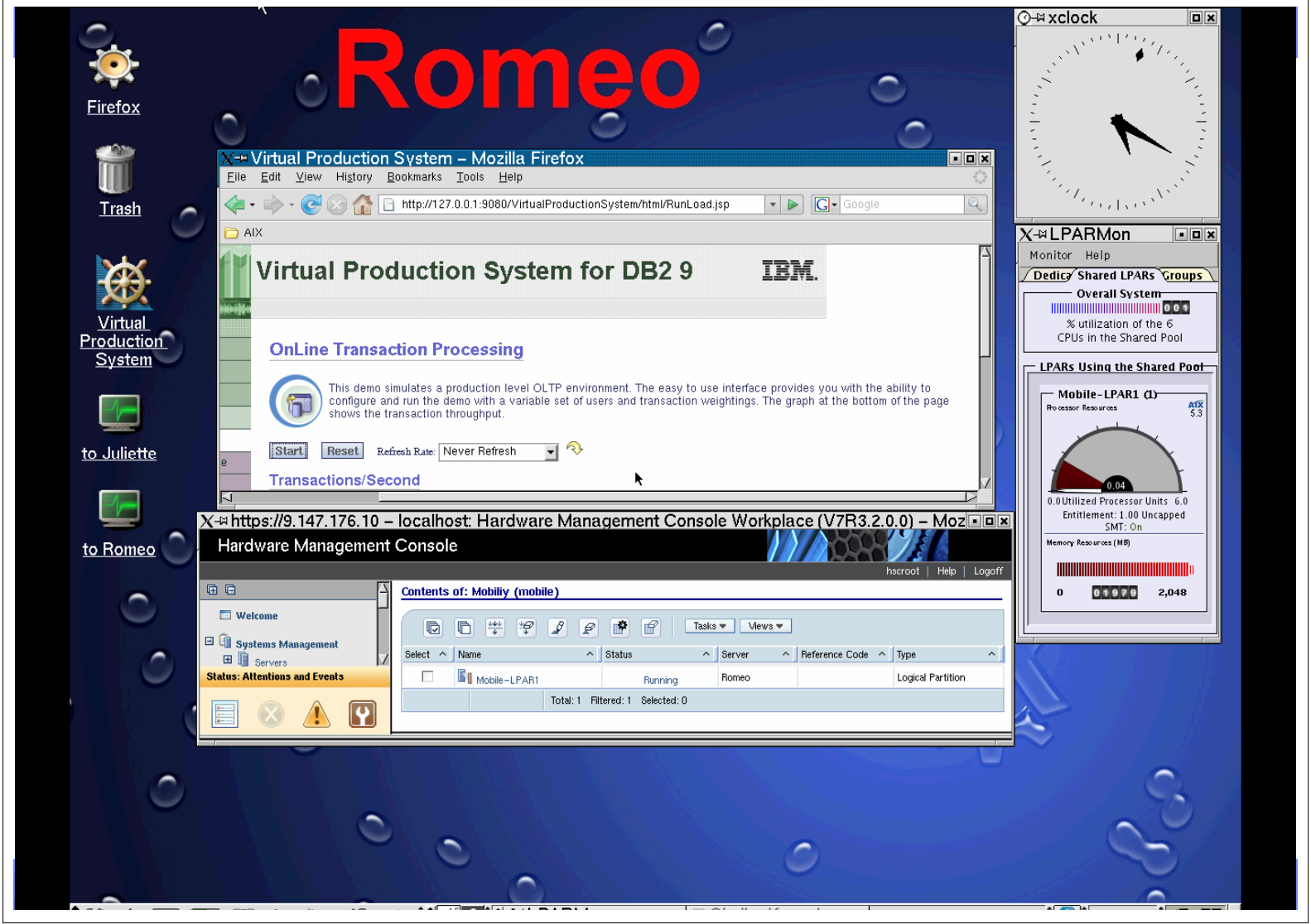
Retrait des définitions sur la source

Système 1: Romeo

Système 2: Juliette



Demo...


Romeo
Firefox
Trash
Virtual Production System
to Juliette
to Romeo
Virtual Production System – Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://127.0.0.1:9080/VirtualProductionSystem/html/RunLoad.jsp
AIX
Virtual Production System for DB2 9
IBM
OnLine Transaction Processing
This demo simulates a production level OLTP environment. The easy to use interface provides you with the ability to configure and run the demo with a variable set of users and transaction weightings. The graph at the bottom of the page shows the transaction throughput.
Start
Reset
Refresh Rate:
Never Refresh
Transactions/Second
https://9.147.176.10 – localhost: Hardware Management Console Workplace (V7R3.2.0.0) – Moz
Hardware Management Console
hscroot | Help | Logoff
Welcome
Systems Management
Servers
Status: Attentions and Events
Contents of: Mobiliy (mobile)
Tasks
Views
Select
Name
Status
Server
Reference Code
Type
Mobile-LPAR1
Running
Romeo
Logical Partition
Total: 1 Filtered: 1 Selected: 0
xclock
LPARMon
Monitor Help
Shared LPARs
Overall System
% utilization of the 6 CPUs in the Shared Pool
LPARs Using the Shared Pool
Mobile-LPAR1 (1)
Processor Resources
0.04
0.0 Utilized Processor Units 6.0
Entitlement: 1.00 Uncapped
SMT: On
Memory Resources (MB)
0
01979
2,048


M@rci.
e-business