

Titre	Convertir une machine physique (Linux/Windows) en machine virtuelle (= Physical To Virtual = P2V)
Propriétaire	jose.tavares@hesge.ch
Classification	
Date dernière modification	14 Octobre 2009
Chemin\NomFichier	http://www.tdeiq.ch/vmware/P2V.pdf

Synthèse

Il est conseillé, dans la mesure du possible, d'éviter la conversion (P2V) et donc de réinstaller le système que l'on souhaite virtualiser dans une machine virtuelle (VM), car un système installé physiquement n'est pas optimisé pour s'exécuter dans une VM.

De plus il comporte des pilotes dont il n'aura plus besoin, ainsi que des informations hardware qui peuvent être fausses/manquantes (GUID du disque dur, chemins, partitions, ...), un certain nombre de ces problèmes a été rencontré, notamment au §6

Le P2V peut toutefois être très utile pour virtualiser un système complexe, qui n'est pas simple à installer ou qui manque de documentation.

Cependant, le P2V reste propre à chaque distribution et chaque configuration hardware (surtout dans le monde Linux), nous avons toutefois essayé de donner une marche à suivre plus ou moins générale.

Ce document traite :

- P2V d'un système Linux Ubuntu Desktop 8.04 LTS 32 bits, voir §3
- P2V d'un système Debian 4.0 32 bits, voir §6
- Redimensionnement des partitions Linux (nécessite parfois de faire V2V), voir §4-5
- P2V d'un système Windows (XP/Vista), voir §7

Table des matières

1	BUTS ET CONFIGURATION HARDWARE.....	3
2	RÉSUMÉ DES OPÉRATIONS GÉNÉRALES	3
3	P2V DU SERVEUR WEB (UBUNTU DESKTOP 8.04 LTS 32BITS).....	5
3.1	Scénario et résultats.....	5
3.2	Manipulations.....	5
3.2.1	Résumé	5
3.2.2	Détails disques source et destination.....	7
3.2.3	Sauvegarder les données de la partition data à l'aide d'un tar	8
3.2.4	Création de la machine virtuelle et copie du disque.....	9
3.2.5	Création d'un 2ème disque virtuel (2ème partition).....	9
3.2.6	Démarrer le système virtualisé et appliquer le tar	10
3.2.7	Reconfiguration sur le système virtualisé.....	10
3.2.8	Création d'un backup avec VMware Converter Standalone Client	10
4	REDIMENSIONNEMENT D'UNE PARTITION LINUX	11
5	V2V DU WEBSERVER	12
5.1	Scénario et résultats	12
5.2	Manipulations.....	12
5.2.1	Disques et copie	12
5.2.2	Attacher le disque virtuel existant.....	13
5.2.3	Reconfigurer le point de montage	13
5.2.4	Reconfiguration des interfaces réseau virtuelles.....	13
5.2.5	Contrôle des points de montage.....	13
6	P2V DU SERVEUR DNS (DEBIAN 4.0 32BITS)	14
6.1	Scénario et résultats	14
6.2	Manipulations.....	14
6.2.1	Détails disques source et destination.....	14
6.2.2	Création de la machine virtuelle et copie du disque.....	15
6.2.3	Modifier la configuration disque sur le système virtualisé	15
6.2.4	Reconfiguration des interfaces réseau virtuelles.....	16
7	P2V WINDOWS AVEC CONVERTER	17
	ANNEXE – DIFFÉRENCES ENTRE ESXI3.5 ET ESXI4	23

1 Buts et configuration hardware

Convertir une machine physique Linux en machine virtuelle (VM), c'est ce qu'on appelle faire du P2V Linux.

Le but étant de remplacer les serveurs physiques de notre DMZ en serveurs virtuels

Il est conseillé, dans la mesure du possible, d'éviter la conversion et donc de réinstaller depuis le début le système que l'on souhaite virtualiser.

Cependant, cela n'est pas toujours possible, car il existe certains serveurs compliqués à réinstaller (on manque de documentation des différentes manipulations qui ont été effectuées), ou par manque de temps. Dans ce cas, le P2V peut être une solution intéressante, bien que le système transféré par P2V ne soit pas optimisé pour s'exécuter dans une machine virtuelle !

J'ai aussi profité d'essayer la version 4 d'ESXi, et de noter quelques différences par rapport à la version ESXi 3.5

La machine utilisée en tant que serveur ESXi est un serveur DELL PowerEdge 1850

- 2x CPU Intel Xeon 2.8Ghz
- 2 GB RAM
- Disque Dur SCSI 136GB
- 2 cartes réseau Intel Pro 1000

2 Résumé des opérations générales

Attention

Il ne faut pas lancer le P2V pendant que le système physique exécute votre système Linux, il faut toujours lancer le P2V depuis un LiveCD ! Si on ne fait pas ceci, il risque d'y avoir des changements sur le système pendant le P2V, ce qui va corrompre la copie !

Le P2V Linux est spécifique pour chaque distribution et pour chaque configuration hardware physique.

En effet, une machine physique peut avoir des disques IDE et souvent sur les machines virtuelles on a du S-ATA ou SCSI ce qui nécessite une reconfiguration, ou avoir des dépendances sur tel matériel, avoir stocké les ID disques ou réseau, etc. Tout ceci varie en fonction du hardware disponible et des distributions Linux, ce qui rend le P2V Linux spécifique à chaque cas.

Cependant, en règle générale, les opérations à effectuer pour le P2V Linux sont décrites comme suit, avec l'axe du temps allant de haut en bas :

Machine Physique Linux

Noter la config. Hardware (CPU, RAM, HDD)
Taille des partitions (commande `sfdisk`, `fdisk`)

Sauvegarde des données non-système en gardant
les permissions des fichiers (tar)

Booter sur un LiveCD Linux

Machine Virtuelle

Créer la machine virtuelle (RAM suffisante, taille disque
système légèrement supérieure à la machine physique)

Booter sur un LiveCD Linux

-----> transfert système avec netcat + dd ----->

Eteindre la machine virtuelle
Créer une nouvelle partition pour accueillir les données
non-système préalablement sauveées avec un tar

Appliquer (sur LiveCD ou sur le système démarré) le tar à la
nouvelle partition (attention à respecter les points de
montage, comme sur la machine physique)

Reconfigurer les interfaces réseau, reconfiguration diverses
dépendant de la distribution et du hardware

Au besoin, redimensionner les partitions avec Gparted
(LiveCD) puis effectuer un V2V comme ci-dessus avec
netcat + dd

3 P2V du serveur web (Ubuntu Desktop 8.04 LTS 32bits)

3.1 Scénario et résultats

La conversion P2V c'est bien déroulée et a été basée sur :
http://conshell.net/wiki/index.php/Linux_P2V

La partition système à transférer faisait **40GB**, ce qui a pris environ **5heures** de transfert sur un réseau à **100Mb/s**

3.2 Manipulations

3.2.1 Résumé

Voici un résumé des opérations effectuées pour le P2V du serveur web ubuntu.

Ce résumé correspond aux manipulations effectuées au §3, 4 et 5, en spécifiant à chaque fois les chapitres concernés :

Machine Physique Ubuntu Desktop 8.04 LTS 32bits

Noter la config. Hardware (P4 1.7Ghz, 512 RAM, HDD 40+160GB)
Taille des partitions (commande `sfdisk`, `fdisk`) §3.2.2

Sauvegarde des données non-système en gardant
les permissions des fichiers (tar) §3.2.3

Booter sur un LiveCD Linux §3.2.4

Machine Virtuelle

Créer la machine virtuelle (RAM suffisante, disque système
légèrement supérieur à celui de la machine physique) §3.2.4

Booter sur un LiveCD Linux §3.2.4

-----> transfert système avec netcat + dd §3.2.4 ----->

Eteindre la machine virtuelle

Créer une nouvelle partition pour accueillir les données
non-système préalablement sauvées avec un tar §3.2.5

Appliquer (sur LiveCD ou sur le système démarré) le tar à la
nouvelle partition (attention à respecter les points de
montage, comme sur la machine physique) §3.2.6

Reconfigurer les interfaces réseau §3.2.7

Redimensionner les partitions avec Gparted (LiveCD) §4

Effectuer un V2V avec netcat + dd §5

Reconfiguration des points de montage et du réseau,
contrôle §5.2.3 à §5.2.5

3.2.2 Détails disques source et destination

Commençons par consulter les propriétés des disques et partitions

```
root@webserver:~# fdisk -l
```

```
Disk /dev/sda: 40.0 GB, 40037760000 bytes
255 heads, 63 sectors/track, 4867 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0xcaddcadd
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	4680	37592068+	83	Linux
/dev/sda2		4681	4867	1502077+	5	Extended
/dev/sda5		4681	4867	1502046	82	Linux swap / Solaris

```
Disk /dev/sdb: 160.0 GB, 160041885696 bytes
255 heads, 63 sectors/track, 19457 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x38950e49
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	19457	156288321	83	Linux

On constate que notre système possède 2 disques durs (**sda** et **sdb**)
La taille affichée (**40GB** pour **sda** et **160GB** pour **sdb**) est la taille brute des disques (capacité maximum).

Ce qui nous intéresse réellement, c'est la taille totale des partitions

Nous devons aussi savoir où se trouve notre système ! Se trouve-t-il dans **sda1** ? **sdb1** ?

Regardons les points de montage avec la commande suivante :

```
eig@webserver:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1       36G   3.1G   31G   9% /
varrun          252M  228K  252M   1% /var/run
varlock         252M     0  252M   0% /var/lock
udev            252M   56K  252M   1% /dev
devshm          252M     0  252M   0% /dev/shm
lrm             252M   38M  215M  15% /lib/modules/XXXX
ile
/dev/sdb1       148G  541M  140G   1% /data
```

On voit que **sdb1** est monté dans **/data**, c'est en réalité une partition de données

sda1 possède quant à lui le système Linux (racine **/**), c'est ce que l'on doit copier avec l'outil **dd**

Quelle taille totale fait notre disque système ?

Il faut pour cela, additionner les blocks de la commande `fdisk -l` (bien que la commande `df -h` nous indique que sda1 fait environ 36GB, il nous manque la partition étendue et de swap (indiquées dans la commande `fdisk -l`), qu'il est préférable de copier en même temps afin d'éviter des reconfigurations inutiles et qui peuvent être compliquées) :

37592068 + 1502077 (une seule fois, en comptant seulement la partition Extended qui contient toutes les partitions logiques comme sda5) = 39094145 blocks

Puis on divise cette valeur 2 fois par 1024 (car 1block = 1Kb) :
(39094145/1024) / 1024 = 37.3 GB

On constate que le système Linux utilise environ 37.3GB, entre les partitions système et swap, ce qui doit être copié via l'outil dd.

Concernant la partition data d'une taille d'environ 160GB, on voit qu'il y a très peu de données dessus (environ 540MB).

Il serait donc très peu judicieux de copier 160GB via dd juste pour récupérer les 540MB de données !

La solution va être de créer un fichier tar qui garde les permissions de chaque fichier de cette partition, que l'on pourra installer par la suite sur une autre partition de taille quelconque ! Ceci nous permet de nous passer de la longue copie avec dd (160GB c'est très long...), ainsi que de créer une partition d'aussi grande taille alors que cette dernière est très peu exploitée !

3.2.3 Sauvegarder les données de la partition data à l'aide d'un tar

Afin de créer le fichier tar de la partition data en gardant les permissions des fichiers :
`root@webserver:~# tar -cpf data.tar /data`

Remarque : On sauvegarde ici tout ce qui est monté dans /data (car/dev/sdb1 renvoie bien vers /data, comme vu dans la commande `df -h` du § précédent)

Ce tar sera sauvegardé sur la partition sda (et donc sera automatiquement copié avec dd), je profite aussi de le récupérer sur mon poste de travail (via le SFTP de Bitwise Tunnelier par exemple)

On crée ensuite un disque virtuel de taille plus petite (20GB au lieu des 160GB) pour la machine virtuelle, où l'on y appliquera le tar !

3.2.4 Création de la machine virtuelle et copie du disque

Maintenant que l'on a confirmé l'espace disque utilisé par le système, on va devoir créer une machine virtuelle sous ESX ou ESXi d'une taille un peu plus grande. Cette taille plus grande que d'origine est nécessaire afin de pouvoir copier toute la partition à l'aide de l'outil dd

Démarrer la nouvelle VM à l'aide d'un liveCD Linux (j'ai utilisé ubuntu 8.10 Desktop 32bits, mais on peut utiliser beaucoup d'autres LiveCD Linux !)

Attendre le lancement live d'ubuntu, puis exécuter la commande :
`sudo nc -l -p 9001 | dd of=/dev/sda`
(AltGr et 1 pour faire le symbole « | »)

Puis sur le serveur web physique :
`root@webserver:~# dd if=/dev/sda | nc 10.1.6.251 9001`
(AltGr et 1 pour faire le symbole « | »)

L'adresse IP ci-dessus est bien entendu l'adresse IP attribuée via DHCP du système Live

Le système de **40GB** c'est copié durant environ **5heures** via un réseau **100Mb/s**

3.2.5 Création d'un 2ème disque virtuel (2ème partition)

Pour créer la 2ème partition :
Dans ESXi, ajouter un nouveau disque de 20GB à la machine virtuelle
Puis dans le système ubuntu live, exécuter
`fdisk /dev/sdb`

Source :
<http://www.linux-kheops.com/doc/linux-f/lf226.htm>

Remarque : Il est aussi possible de démarrer le système, et effectuer ceci dans le mode de maintenance !

Utiliser la commande p pour constater qu'il n'y a aucune partition
p

Créer une nouvelle partition avec la commande
n

La choisir en tant que partition primaire numéro 1 (/dev/sdb1)
Laisser les blocs par défaut

Lorsqu'elle est créée, afficher les informations avec la commande
p

Quitter avec la commande w pour sauvegarder les changements :
w

Formater la partition :
`mke2fs -j /dev/sdb1`

Source :
<http://www.linux-france.org/article/sys/ext3fs/>

3.2.6 Démarrer le système virtualisé et appliquer le tar

Redémarrer la machine virtuelle (quitter le CD live puis démarrer le nouveau système ubuntu qui vient d'être virtualisé)

Le système risque d'émettre quelques avertissements car il possède de nouveaux disques et une nouvelle configuration

Lorsqu'ubuntu aura démarré, nous pourrons alors appliquer le tar avec les données sauvegardées de la 2ème partition
Pour cela, se mettre dans le répertoire parent au répertoire /data (qui est déjà créé)
`cd /`

Effectuer un ls du répertoire /data pour confirmer qu'il est vide :
`ls /data`

« Dézipper » le tar à cet endroit
`tar xfv /home/eig/data.tar`

Effectuer un ls du répertoire /data pour confirmer que les données y sont bien stockées
`ls /data`

3.2.7 Reconfiguration sur le système virtualisé

Redémarrer la VM (on quitte l'environnement LiveCD et on démarre pour la première fois notre système virtualisé)

Ubuntu stocke la relation entre adresse MAC et l'interface (eth0, eth1, etc)
Etant donné que notre système a changé de hardware et donc d'adresse MAC, le système ubuntu ne trouve plus l'adresse MAC associée à eth0.
La nouvelle adresse MAC du système est alors associée à eth1, ce qui pose problème
Il faut donc reconfigurer le système en exécutant la commande suivante afin de supprimer le fichier contenant les correspondances MAC-ethX (ce fichier sera automatiquement recréé au prochain redémarrage)

```
sudo rm /etc/udev/rules.d/70-persistent-net.rules
```

Redémarrer la machine virtuelle pour appliquer les changements
`sudo reboot`

3.2.8 Création d'un backup avec VMware Converter Standalone Client

Le backup a créé une appliance de 20.9GB en 1h25mn sur un réseau à 100Mb/s

Les disques durs virtuels faisant 40GB + 20GB, on arrive à une appliance d'une taille d'un tiers de la taille totale, cependant cela reste encore volumineux, le système et les données ainsi que le swap faisant seulement une taille totale d'environ 5GB

Le principal inconvénient de la copie disque via dd est l'obligation d'avoir une partition cible légèrement supérieure à la partition source, on va donc essayer de redimensionner la partition système

4 Redimensionnement d'une partition Linux

Le redimensionnement d'une partition Linux n'est pas aussi simple qu'avec une partition Windows

Les utilisateurs Linux ont souvent des problèmes en effectuant ce genre de manipulations, cependant comme nous avons créé précédemment une sauvegarde de la machine virtuelle, on peut essayer !

J'ai pour cela utilisé Gparted (gratuit), qui est un outil dédié aux opérations sur les disques et partitions, comme Partition Magic sous Windows

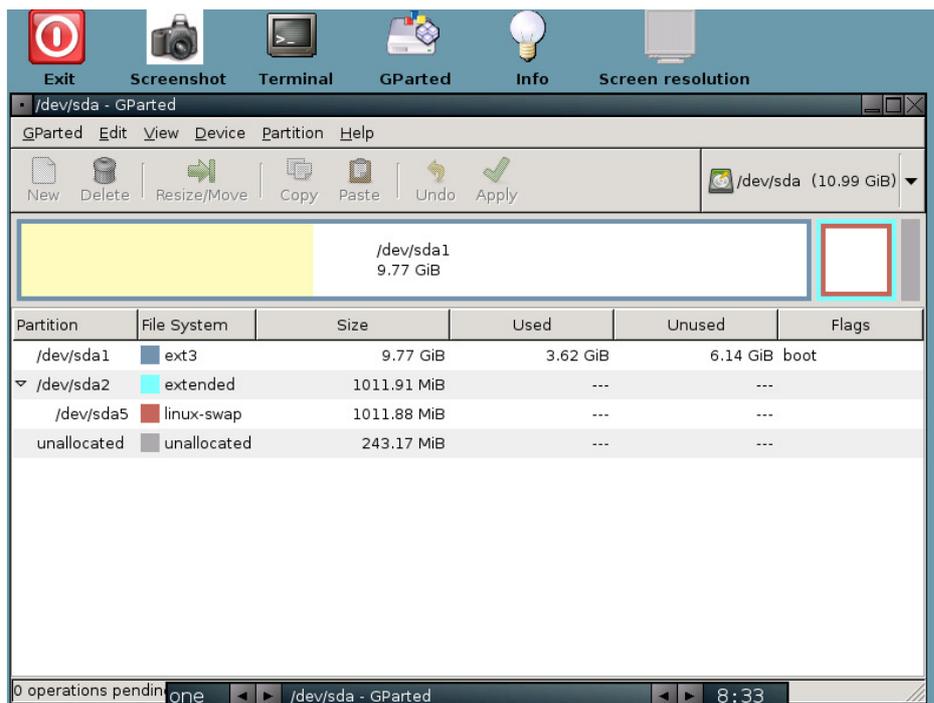
On boot la machine virtuelle à l'aide du CD Gparted, puis on suit les instructions

Je me suis basé sur le document suivant :

<http://www.simplehelp.net/2008/11/04/how-to-resize-linux-partitions-using-gparted/>

Le but étant de redimensionner la partition système et de la déplacer si besoin au début du disque, idem pour la partition swap

Voici le résultat des partitions du webserver, après avoir effectué le §4 et §5 :



5 V2V du webserviceur

5.1 Scénario et résultats

Maintenant que nous avons redimensionné le disque système de notre webserviceur, nous avons encore le problème suivant :

- Le disque vmdk conserve sa taille de 40GB, et je n'ai trouvé aucun moyen de le faire devenir plus petit (non-supporté par VMware).
Il existe l'outil vmkfstool permettant de diminuer la taille d'un fichier vmdk, mais cela agit comme un ciseau, en coupant le fichier indépendamment des données qui y sont stockées, ce qui n'est vraiment pas une bonne idée !

La solution consiste à effectuer du V2V (virtual to virtual) de notre webserviceur virtualisé, vers une nouvelle machine virtuelle qui cette fois possèdera un disque vmdk d'un peu plus de 10GB !

Cette opération s'est avérée concluante, le transfert des 10GB a duré un peu moins d'une heure entre 2 machines virtuelles présentes sur le même serveur ESXi 4.0 !

Concernant la 2^{ème} partition data, j'ai tenté d'attacher le fichier vmdk existant (de la VM source) à la nouvelle VM. Bien entendu la VM source doit être arrêtée. Cette opération c'est avérée moins concluante : le disque est bien vu par le système et les données sont bien dans /data, mais le point de montage (affiché par `df -h`) n'existe pas pour cette partition... Il y a cependant une solution au §5.2.2 et §5.2.3

5.2 Manipulations

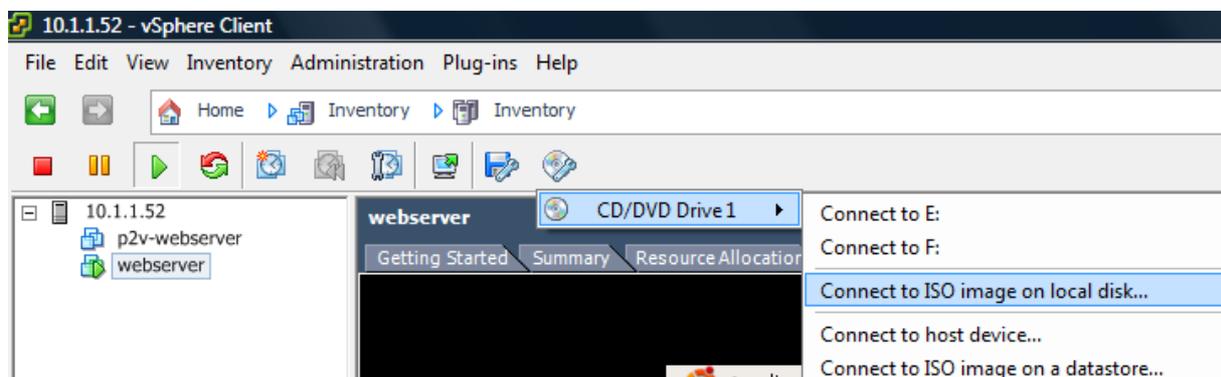
5.2.1 Disques et copie

Les manipulations restent identiques aux points §3.2.2 et §3.2.4

La principale différence réside dans l'image de boot utilisée : Jusqu'à présent on a utilisé un CD boot afin de booter nos machines virtuelles sur un système ubuntu live. Cependant, il n'est pas possible de booter 2 machines virtuelles en même temps avec le même CD !

Sous ESXi4 (et le nouveau vSphere Client), il y a une nouvelle option très pratique qui permet d'attacher une image ISO aux différentes VM !

J'ai donc utilisé l'image ISO du CD Live, que j'ai attaché aux 2 différentes VM en même temps !



5.2.2 Attacher le disque virtuel existant

Il est relativement simple d'attacher un disque vmdk existant à une machine virtuelle

Dans vSphere Client :

Clic droit sur la machine virtuelle – *Edit Settings...* - *Add...*

Sélectionner *Hard Disk – Next*

Sélectionner *Use an existing virtual disk*

On peut ensuite démarrer la nouvelle VM

Comme indiqué dans le §5.1, le disque virtuel ajouté est bien vu par la VM et les données sont bien présentes dans `/data`, mais le point de montage (affiché par `df -h`) n'existe pas pour cette partition...

5.2.3 Reconfigurer le point de montage

Exécuter la commande :

```
mount /dev/sdb1 /data
```

Effectuer un `ls /data` pour voir ce qui se trouve dans ce point de montage

5.2.4 Reconfiguration des interfaces réseau virtuelles

Afin de reconfigurer le réseau (attribuer une nouvelle adresse MAC à l'interface réseau virtuelle et ainsi réinitialiser ces interfaces), exécuter la commande `rm /etc/udev/rules.d/70-persistent-net.rules`

Voir §3.2.7 pour plus d'explications

Redémarrer la machine virtuelle

5.2.5 Contrôle des points de montage

```
eig@webserver:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1       9.7G  3.6G  5.7G  39% /
varrun          252M  240K  252M   1% /var/run
varlock         252M    0  252M   0% /var/lock
udev            252M   52K  252M   1% /dev
devshm          252M   12K  252M   1% /dev/shm
lrm             252M   38M  215M  15% /lib/modules/XXXX
ile
/dev/sdb1       20G   597M   19G   4% /data
gvfs-fuse-daemon 9.7G  3.6G  5.7G  39% /home/eig/.gvfs
```

Le point de montage de notre seconde partition `/dev/sdb1` est cette fois-ci bien listé, tout semble être OK

6 P2V du serveur DNS (Debian 4.0 32bits)

6.1 Scénario et résultats

La conversion P2V a été plus compliquée que pour la distribution ubuntu, car il y a eu plus de choses à reconfigurer. Le P2V a cependant tout de même été réussi !

La partition système à transférer faisait **40GB**, ce qui a pris environ **2heures** de transfert sur un réseau à **100Mb/s**

Les opérations ont été effectuées sur la base des tests concluants du §3, avec quelques complications :

- Redimensionnement des partitions, dont l'espace disque utilisé est peu exploité (déjà vu précédemment)
- Reconfiguration des disques virtuels : le serveur physique possédait des disques hda (IDE), alors que notre serveur virtuel est en sda (SCSI), ce qui posait problème lors du premier démarrage

6.2 Manipulations

6.2.1 Détails disques source et destination

Sur la machine physique :

```
dns_public:~/home/eig# fdisk -l
```

```
Disk /dev/hda: 40.0 GB, 40037760000 bytes
255 heads, 63 sectors/track, 4867 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/hda1	*	1	34	273073+	83	Linux
/dev/hda2		35	4867	38821072+	5	Extended
/dev/hda5		35	642	4883728+	83	Linux
/dev/hda6		643	1007	2931831	83	Linux
/dev/hda7		1008	1195	1510078+	82	Linux swap / Solaris
/dev/hda8		1196	1244	393561	83	Linux
/dev/hda9		1245	4867	29101716	83	Linux

On constate que tous nos disques sont en **hdaX**, ce qui correspond à de **l'IDE** et risque de poser problème...

```
dns_public:~/home/eig# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/hda1       259M   94M  152M  39% /
tmpfs           253M    0  253M   0% /lib/init/rw
udev            10M    68K   10M   1% /dev
tmpfs           253M    0  253M   0% /dev/shm
/dev/hda9        28G  173M   26G   1% /home
/dev/hda8       373M   11M  343M   3% /tmp
/dev/hda5       4.6G  390M  4.0G   9% /usr
/dev/hda6       2.8G  736M  1.9G  28% /var
```

On constate que **hda9** est de grande capacité, mais que cette partition n'utilise presque aucun espace disque, il serait donc bien de la redimensionner (voir §3.2.3 et §4)

6.2.2 Création de la machine virtuelle et copie du disque

Idem qu'au §3.2.2

6.2.3 Modifier la configuration disque sur le système virtualisé

Après copie, voici la configuration disque sur la machine virtuelle
La commande suivante a été exécutée sur un système Ubuntu Live (Boot CD)

```
root@ubuntu# fdisk -l
```

```
Disk /dev/sda: 44.0 GB, 44023414784 bytes
255 heads, 63 sectors/track, 5352 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x456c4d56
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	34	273073+	83	Linux
/dev/sda2		35	4867	38821072+	5	Extended
/dev/sda5		35	642	4883728+	83	Linux
/dev/sda6		643	1007	2931831	83	Linux
/dev/sda7		1008	1195	1510078+	82	Linux swap / Solaris
/dev/sda8		1196	1244	393561	83	Linux
/dev/sda9		1245	4867	29101716	83	Linux

En quittant le mode Live puis en démarrant la machine virtuelle normalement, on obtient le message d'erreur :

```
Waiting for root file system
```

Ce message est logique, puisque le système cherche le disque hda qu'il ne trouve pas...

Puis après quelques minutes :

```
/dev/hda1 does not exist...
```

Il faut donc essayer de reconfigurer les liens vers les différentes partitions

J'ai rebooté sur le LiveCD, puis j'ai effectué les modifications suivantes :

Il faut tout d'abord commencer par monter la partition sda1 qui contient notre système Linux Debian. Pour cela il faut tout d'abord créer un dossier pour le montage, puis monter la partition dedans :

```
mkdir /mnt/rescue
mount /dev/sda1 /mnt/rescue
```

Le système Debian se trouve ensuite monté dans /mnt/rescue et peut être modifié

```
nano /mnt/rescue/etc/fstab
```

Puis modifier tous les hdaX par sdaX, sauf pour le CDRROM. Ce dernier devra être configuré plus tard

```
nano /mnt/rescue/etc/mtab
```

Puis comme précédemment, modifier tous les les hdaX par sdaX

```
nano /mnt/rescue/boot/grub/device.map
```

La ligne de ce fichier doit être :
(hd0) /dev/sda

```
nano /mnt/rescue/boot/grub/menu.lst
```

Ce fichier est assez long, il faut chercher tous les hdaX et les remplacer par des sdaX

Eventuellement appliquer les changements au boot menu GRUB, pour cela voir le lien ci-dessous

Source :

<http://www.synergeek.fr/2009/05/conversion-physique-vers-virtuelle-p2v-vmware-d%E2%80%99une-linux-debian-equipee-de-disques-ide/>

6.2.4 Reconfiguration des interfaces réseau virtuelles

Afin de reconfigurer le réseau (attribuer une nouvelle adresse MAC à l'interface réseau virtuelle et ainsi réinitialiser ces interfaces), exécuter la commande
rm /etc/udev/rules.d/70-persistent-net.rules

Voir §3.2.7 pour plus d'explications

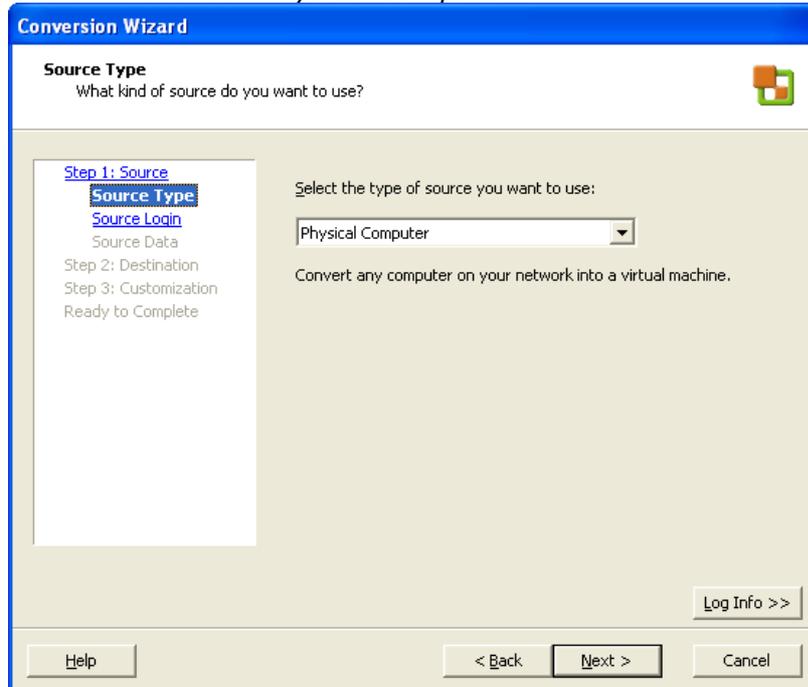
Redémarrer la machine virtuelle

7 P2V Windows avec Converter

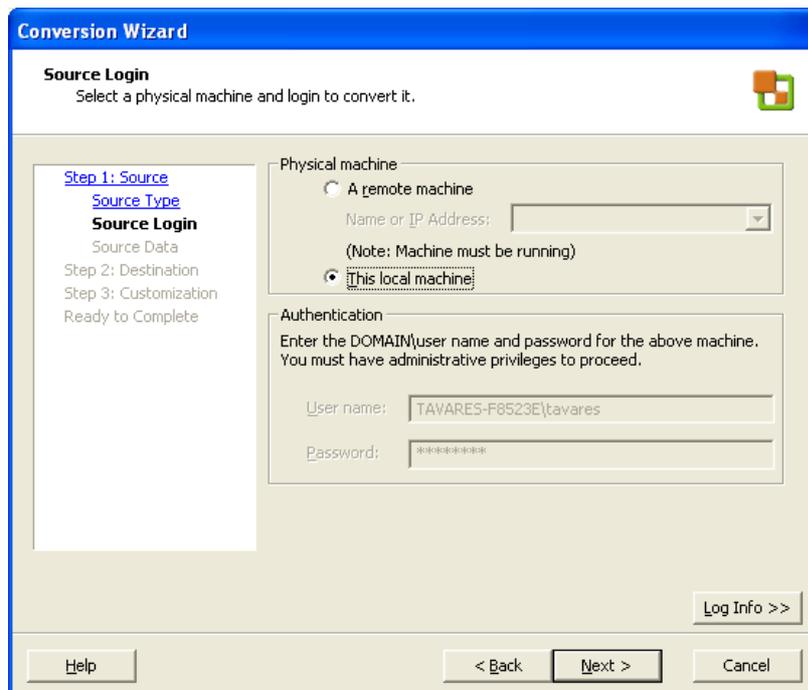
Cette opération est très simple, il suffit d'installer Converter sur le poste Windows que l'on souhaite virtualiser, puis d'effectuer les actions suivantes :

Start – Programs – VMware – VMware Converter

Choisir la source : *Physical Computer*

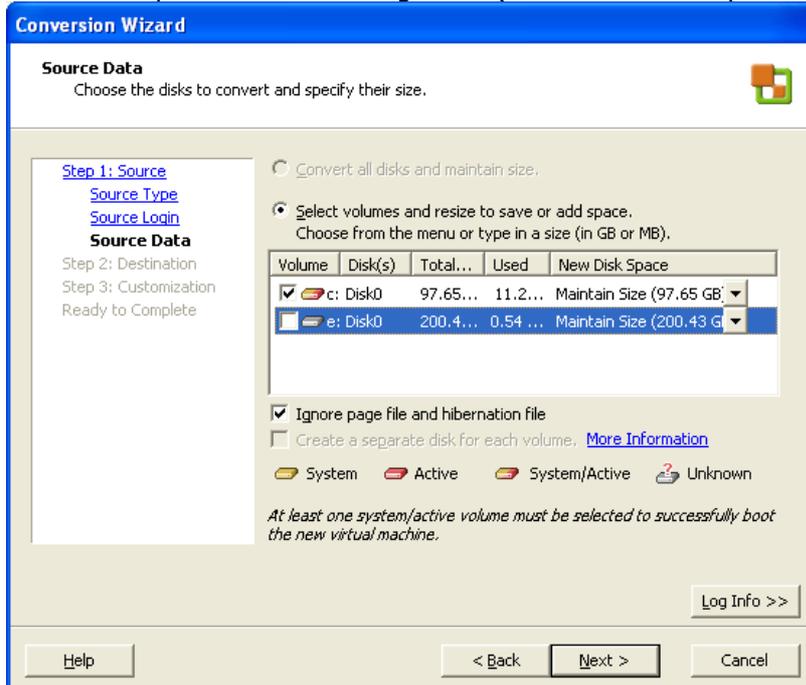


Next



Next

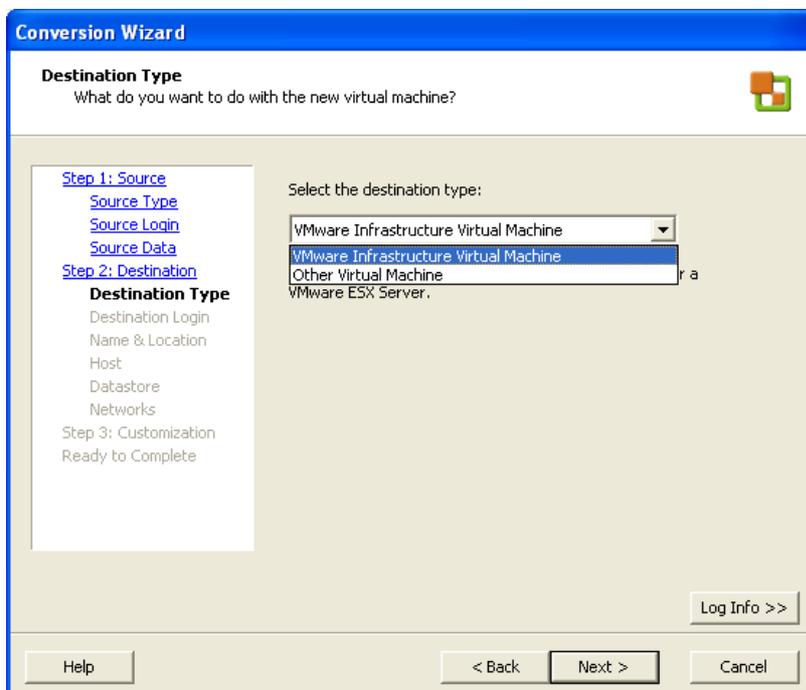
Choisir les partitions à sauvegarder (ici seulement la partition système C:)



Next

Choisir la destination :

- *VMware Infrastructure Virtual Machine* pour créer directement une VM sous ESX/ESXi
- *Other Virtual Machine* pour créer une appliance compatible avec l'outil souhaité (Workstation, Player, VMware Server, VMware Fusion, ...)



Next

Donner l'adresse IP du serveur ESX/ESXi

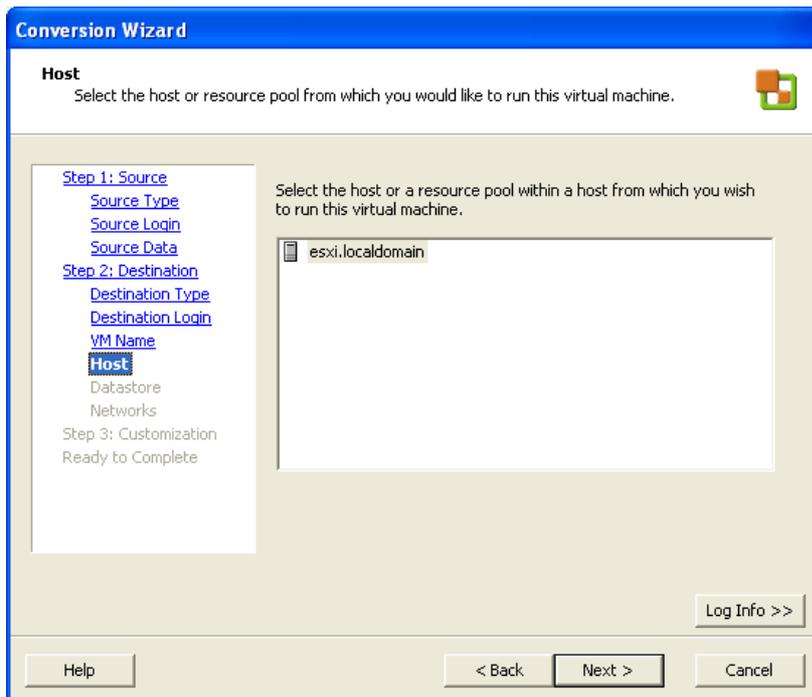
The screenshot shows the 'Conversion Wizard' dialog box with the 'Destination Login' step selected. The title bar reads 'Conversion Wizard'. The main heading is 'Destination Login' with the subtitle 'Specify server and login credentials'. A navigation pane on the left lists steps: Step 1: Source (with sub-items Source Type, Source Login, Source Data), Step 2: Destination (with sub-items Destination Type, Destination Login), and Step 3: Customization (with sub-item Ready to Complete). The 'Destination Login' step is highlighted. The main area contains the instruction: 'Specify the name or IP address of the VirtualCenter Server, or the host, where you want your converted virtual machine to be stored. Enter administrator credentials to the server you specify.' Below this are three input fields: 'Server:' with a dropdown menu showing '10.1.1.53', 'User name:' with a text box containing 'root', and 'Password:' with a text box containing '*****'. A 'Log Info >>' button is located at the bottom right of the main area. The bottom of the dialog features 'Help', '< Back', 'Next >', and 'Cancel' buttons.

Next

Donner un nom à la VM

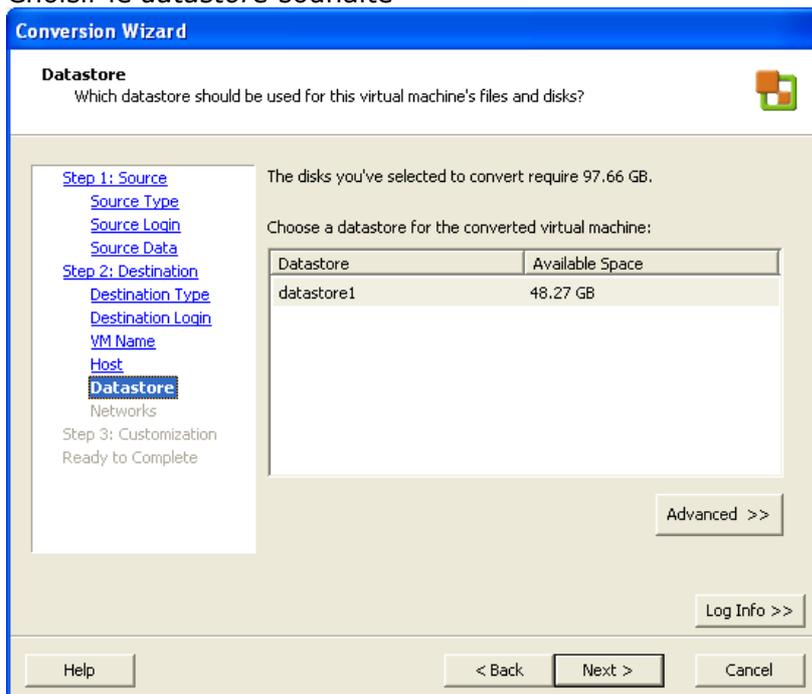
The screenshot shows the 'Conversion Wizard' dialog box with the 'Virtual Machine Name' step selected. The title bar reads 'Conversion Wizard'. The main heading is 'Virtual Machine Name' with the subtitle 'What would you like to name the virtual machine?'. A navigation pane on the left lists steps: Step 1: Source (with sub-items Source Type, Source Login, Source Data), Step 2: Destination (with sub-items Destination Type, Destination Login), and Step 3: Customization (with sub-item Ready to Complete). The 'Virtual Machine Name' step is highlighted. The main area contains the instruction: 'Virtual machine name: (maximum 80 characters)'. Below this is a text box containing 'Nom_de_la_VM'. A 'Log Info >>' button is located at the bottom right of the main area. The bottom of the dialog features 'Help', '< Back', 'Next >', and 'Cancel' buttons.

Next



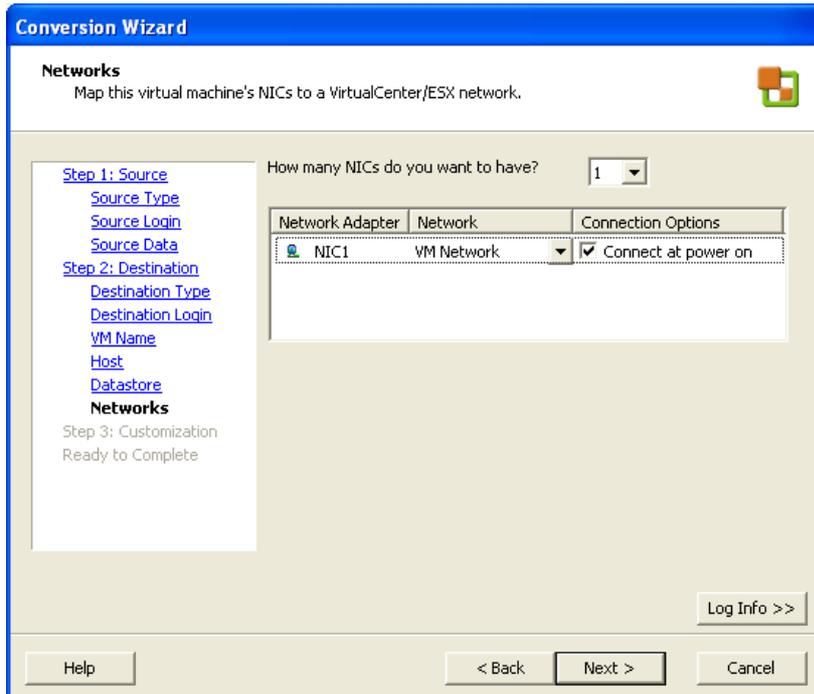
Next

Choisir le *datastore* souhaité



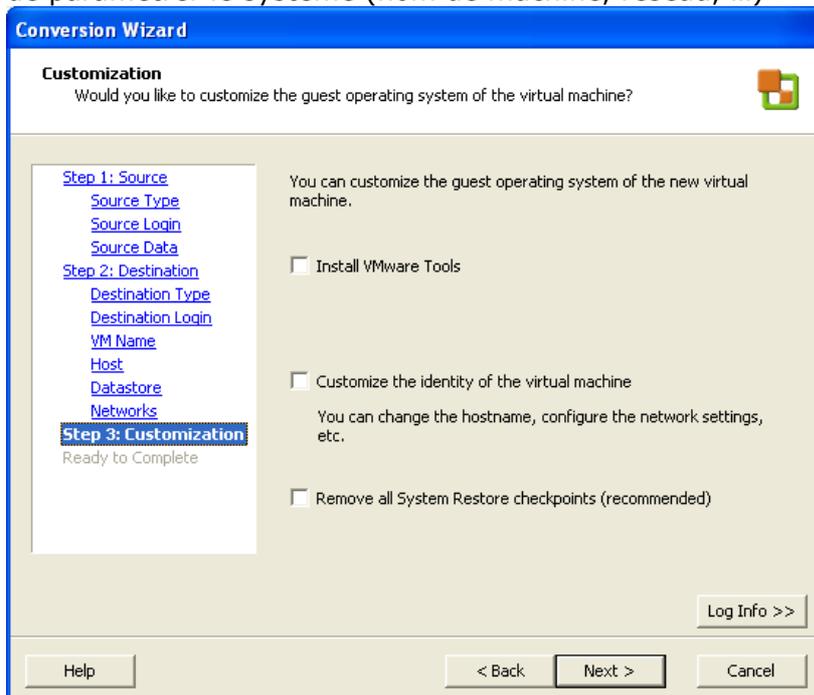
Next

Attacher l'interface réseau virtuelle au vSwitch souhaité



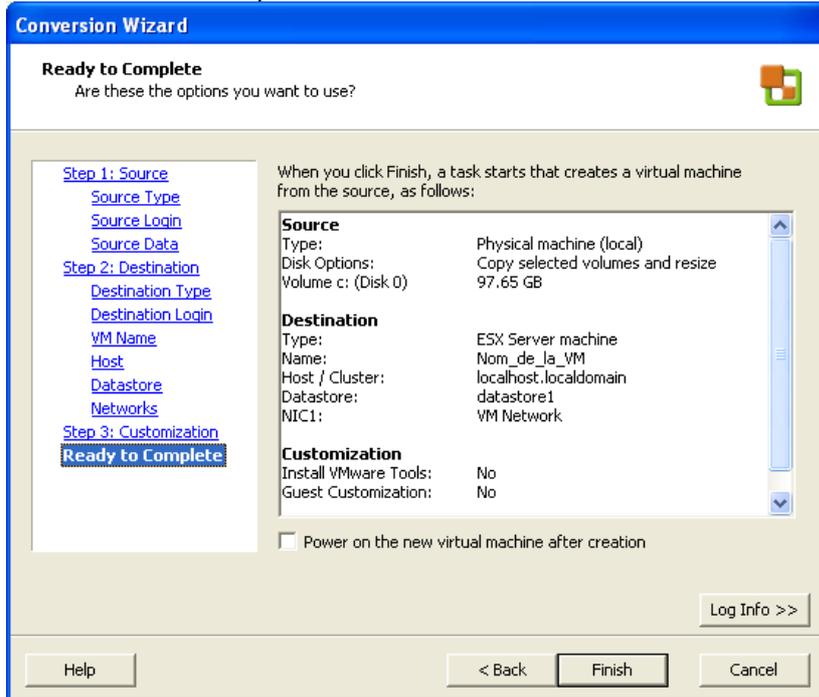
Next

Pour un système Windows, il est possible d'installer en même temps les *VMware Tools* ou de paramétrer le système (nom de machine, réseau, ...)



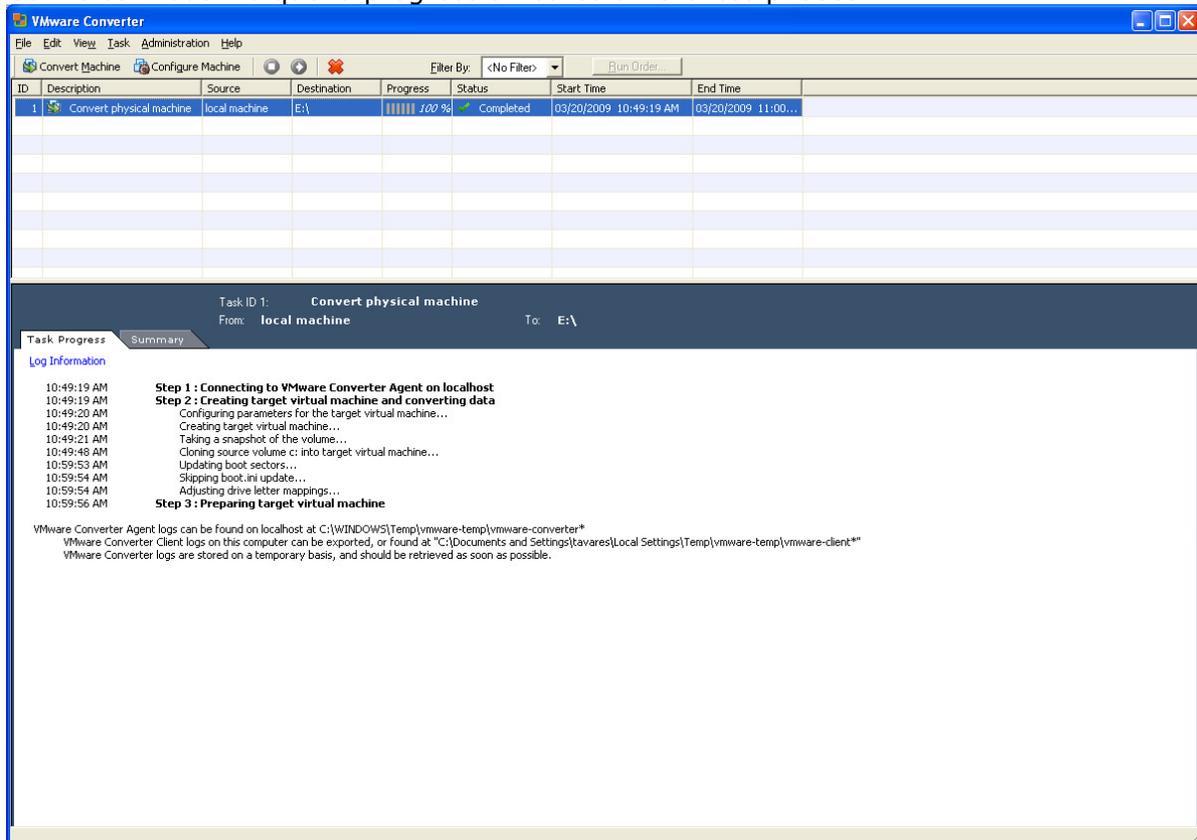
Next

Vérifier le résumé présenté



Lancer la conversion en appuyant sur *Finish*

Convertir nous indique la progression et les différentes phases



ANNEXE – Différences entre ESXi3.5 et ESXi4

Possibilité d'attacher une image ISO à plusieurs VM. Cette image ISO doit se trouver sur le poste exécutant vShere Client (très pratique car cela évite de graver des CD, et surtout permet à plusieurs VM d'avoir accès à la même image ISO en simultanée !

Le hardware virtuel (hardware des machines virtuelles) est différent entre ESXi3.4 et ESXi4. On ne peut pas utiliser une machine virtuelle ESXi4 sur ESXi3.5, mais une machine virtuelle ESXi3.5 peut être ajoutée dans ESXi4