

Communiquer, contrôler à distance un système informatique autonome

Autres activités du TD : voir https://www.ensta-bretagne.fr/lebars/tutorials/TD_wifi_rtk_multiboot_rpi.pdf.

Rendre sur Moodle dans un fichier de la forme DATE_NOM1_NOM2_..._NOMN.zip tous les documents utiles permettant de montrer le travail effectué : notes techniques, codes, fichiers de config, logs, données brutes et/ou traitées, infos précises sur les versions logicielles et matérielles utilisées, schémas en CAO ou manuels, captures d'écran, photos et vidéos à différentes étapes, etc.

Tout ce qu'il y a dans ce document ne sera a priori pas faisable dans le temps imparti mais il serait souhaitable d'arriver jusqu'à la partie **Restauration du boot loader GRUB**. Les parties restantes ont des infos qui pourront être potentiellement utiles dans certaines situations.

Activité 2 : Préparation de PCs avec multiboot Windows et Ubuntu via images disques

Nb max personnes : 4 quadrinômes

Matériel nécessaire par équipe : 1 NUC et son alim et son ou ses adaptateurs pour écran, 1 câble eth, 1 à 2 SSD USB/clé USB, 1 PC de salle info, 1 PC portable d'étudiant

Durée estimée : 1 demi-journée ou plus



Le but de cette activité va être de rapidement préparer plusieurs PC avec la même configuration logicielle. L'idée est que lors de la mise en place de meutes de robots ou si un PC de robot doit être remplacé, il faut savoir rapidement restaurer une installation d'OS+applications et être capable de la configurer de la manière la plus flexible possible.

Les PC à installer seront des Intel NUC (ou similaires). Des SSD USB ou clés USB seront utilisés pour les installer.

Note sur les clés USB : bien que certains sites web indiquent qu'on peut désormais débrancher sans avoir cliqué sur Ejecter dans Windows, en pratique c'est plus compliqué vu que ça dépend de la version de Windows 10 et aussi des clés USB (ou adaptateurs de cartes SD, etc.), certaines étant reconnues comme des disques durs. Pour éviter de se retrouver avec des données corrompues, toujours s'assurer que la clé ou le disque ont été éjectés proprement avant de les débrancher !

Les installations seront d'abord faites en mode de démarrage UEFI+GPT (le plus moderne), puis il faudra essayer de passer au mode BIOS+MBR sans avoir à réinstaller complètement les OS.

2.1 Remarques préliminaires sur les installations de Windows et Ubuntu

Sur un PC où il y a un risque qu'on doive passer d'un mode de démarrage UEFI+GPT à BIOS+MBR (ou inversement), il est préférable d'organiser le disque système avec :

- 1 partition souvent appelée **boot** au début (seulement utilisée par Windows si BIOS+MBR, utilisée par Windows et Ubuntu si UEFI+GPT et souvent appelée **EFI** dans ce cas), de 100 Mo à 260 Mo en général.
- 1 partition spéciale de 16 Mo parfois appelée **MSR** utilisée par Windows en mode UEFI+GPT. Sinon, étendre la partition de boot à 116 Mo pour pouvoir facilement récupérer 16 Mo en cas de passage d'un mode BIOS+MBR à UEFI+GPT, tout en limitant le nombre de partitions (limite de 4 partitions principale en mode BIOS+MBR).
- La partition qui contiendra l'essentiel des fichiers et données de Windows, souvent appelée **Windows**.
- Si besoin on peut prévoir une partition de **recovery** juste après la partition Windows (un peu compliqué à mettre en place et probablement inutile si on est capable d'installer manuellement un système d'exploitation via une image disque). Certaines mises à jour de Windows peuvent silencieusement en créer une en réduisant automatiquement la taille de la partition Windows.

Note : si on souhaite n'utiliser que le mode BIOS+MBR et qu'on veut optimiser son espace disque, on peut forcer Windows à mettre ses fichiers de boot sur sa partition Windows et ne pas prévoir de partition de recovery.

Si après avoir installé Windows on souhaite manuellement installer Ubuntu, le plus simple est en général de réduire la taille de la partition Windows et créer une grande partition dans l'espace libéré pour l'essentiel des fichiers et données d'Ubuntu, suivi éventuellement d'une petite partition de swap (probablement inutile si le PC a beaucoup de RAM (> 16 GB)). Il est en général possible de supprimer la partition de recovery de Windows sans le perturber dans son fonctionnement normal, il faut cependant être bien conscient que l'on perd alors potentiellement des outils permettant la restauration de la configuration logicielle de l'ordinateur dans son état d'usine s'il était vendu avec Windows et d'autres logiciels préinstallés (ceci n'est pas gênant si on se sent capable de réinstaller Windows manuellement et qu'on n'a pas besoin des éventuels logiciels préinstallés, ou qu'on sait comment les récupérer par ailleurs, noter aussi en général que les infos de licence de Windows 8-10 sont enregistrées dans l'UEFI) et s'assurer que c'est bien une partition qui n'est pas utilisée pour d'autres choses plus importantes (chaque modèle d'ordinateur peut avoir ses particularités).

Note : attention, si on est en mode BIOS+MBR et que des partitions extended sont utilisées, il est possible que l'installation d'Ubuntu rende illisibles ces partitions pour Windows. L'outil **testdisk** (https://www.cgsecurity.org/wiki/TestDisk_Download) peut parfois aider à réparer ces partitions en les convertissant en primary, dans la limite de 4 partitions primary par disque (cela reste dans tous les cas une réparation où il y a un gros risque de vraiment perdre ses données).

Note : combinaisons possibles en général pour les modes UEFI, BIOS, MBR, GPT :

- UEFI+GPT (parfois juste désigné par UEFI ou GPT seul par abus de langage)
- UEFI+MBR (mode de compatibilité)
- BIOS+MBR (parfois juste désigné par BIOS ou MBR seul par abus de langage)
- BIOS+GPT n'est pas possible

2.2 Remarques préliminaires sur Windows PE

Windows PE est un Windows minimal pouvant démarrer à partir d'une clé USB ou d'un CD, souvent utilisé pour installer, modifier ou réparer Windows (en fait, lorsqu'on démarre sur une clé USB ou un DVD d'installation de Windows (e.g. préparé avec <https://www.microsoft.com/software-download/windows10>), c'est Windows PE qui démarre et lance immédiatement Windows Setup (en appuyant sur **MAJ+F10**, on peut toutefois faire apparaître une invite de commandes si on souhaite faire autre chose, ce qui peut être pratique pour investiguer en cas d'erreur au cours de l'installation (notamment la commande **notepad** est disponible et avec le menu **File\Open** et en sélectionnant **All Files** dans la boîte de dialogue on peut naviguer dans les dossiers et éventuellement faire des copies, etc.), tenter une installation manuelle de l'image **install.wim/.esd** avec les commandes **diskpart**, **dism**, **bcdboot**, etc.)). Il faut en général s'assurer que la version de Windows PE utilisée est la même ou au moins plus récente que celle que l'on souhaite installer, modifier ou réparer. A noter que Windows Recovery est aussi basé sur Windows PE et lance des outils de réparation au lieu de Windows Setup. Des versions plus complètes et personnalisables existent/existaient, e.g. Win8.1PE SE (voir <http://www.irongeek.com/i.php?page=security/winbuilder-win7pe-se-tutorial> et <http://win81se.cwcodes.net/>) ou **Windows To Go** (voir <https://technet.microsoft.com/fr-fr/library/hh831833.aspx>), qu'on peut/pouvait tester en téléchargeant une version d'évaluation de **Windows 10 Enterprise Edition** (ou en modifiant une installation de **Windows 10 Pro** avec la commande **slmgr /ipk NPPR9-FWDCX-D2C8J-H872K-2YT43** pour lui donner une licence temporaire Enterprise).

2.3 Préparation d'un SSD USB avec Windows PE

Connecter un des SSD USB fourni à un PC sous Windows avec les droits administrateurs, lancer une invite de commandes et taper :

```
diskpart
```

```
Puis dans diskpart :
```

```
list disk
```

En fonction de la liste qui s'affiche, déterminer le numéro du disque correspondant au SSD USB et le sélectionner avec la commande :

```
select dis 9
```

9 étant un exemple de numéro de disque. Effacer toutes les partitions dessus (**attention : toujours vérifier attentivement si c'est le bon disque !**) avec la commande :

```
clean
```

Créer une nouvelle partition NTFS, qui sera accessible sous la lettre K (choisir une autre lettre si déjà utilisée):

```
create part pri
```

```
active
```

```
format fs=ntfs quick
```

```
assign letter K:
```

Taper:
exit
pour quitter diskpart.

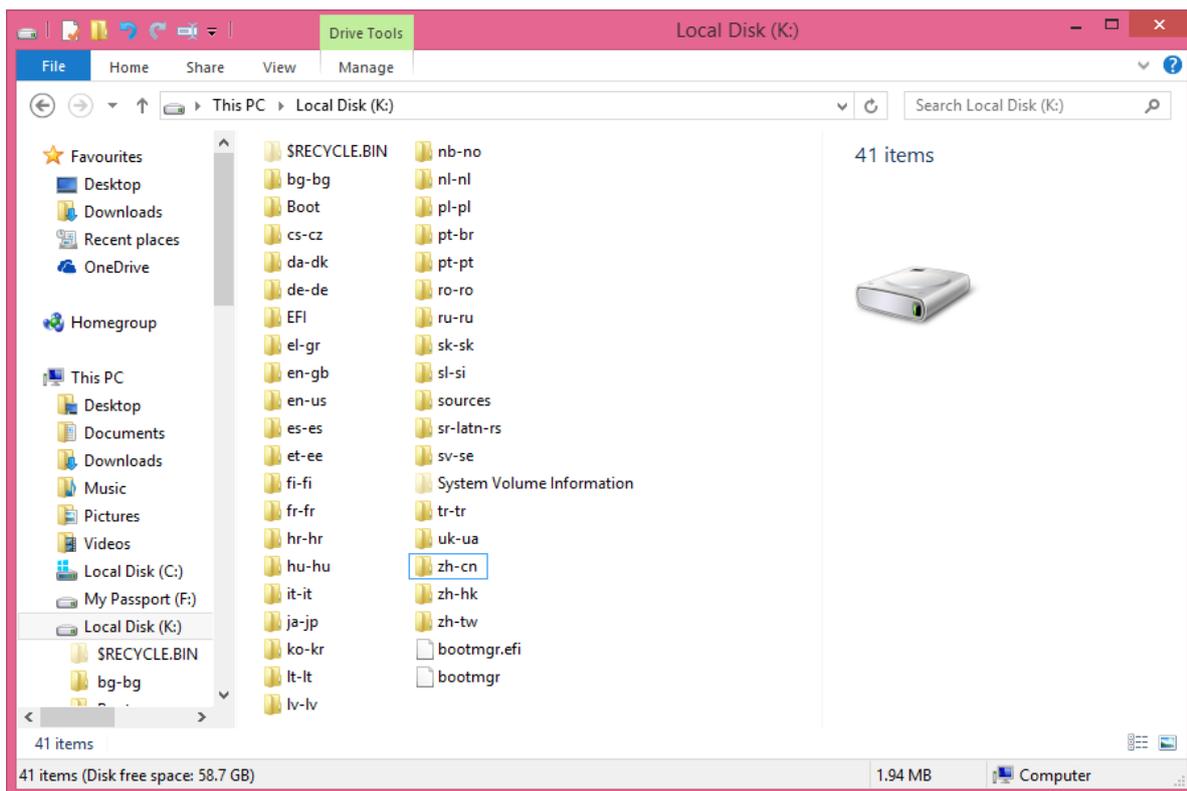
Pour rendre bootable (en mode MBR) le SSD, taper la commande :
bootsect /nt60 K: /mbr /force

Si des options de boot supplémentaires doivent être modifiées, on peut utiliser l'outil **bootice** (à télécharger si nécessaire).

A ce stade, il est possible d'éjecter le SSD USB pour l'utiliser sur un PC de salle info pour copier les fichiers suivants.

Extraire avec **7-Zip** le dossier https://www.ensta-bretagne.fr/lebars/Share/WinPE_ramdisk.zip dans **K:**

Vérifier que la disposition des fichiers est bien similaire (à quelques dossiers ou fichiers près) à ceci, couper-coller les fichiers si nécessaire (i.e. les fichiers ne doivent pas être dans un sous-dossier **WinPE_ramdisk** qui pourrait avoir été créé lors de l'extraction, il faut qu'ils soient à la racine) :



Copier ensuite sur le SSD depuis un PC de salle info (disques réseau) le fichier **win.wim** disponible sur **enseignement\rob\...\Robotique pratique\imvm10x64** (regarder dans **public\share** si **enseignement\rob** n'est pas disponible). Ce fichier de plusieurs GB contient une copie compressée d'une installation de Windows déjà faite sur une machine virtuelle VMware, avec quelques réglages et applications utiles pour la robotique (voir partie **Capturer l'image Windows** pour plus d'infos, à noter que sur une clé USB classique d'installation de Windows préparée avec <https://www.microsoft.com/software->

[download/windows10](#) , on trouve dans le dossier **sources** un fichier **install.wim** (ou **install.esd**, qui peut être converti en suivant e.g. <https://www.intel.com/content/www/us/en/support/articles/000023992/memory-and-storage/intel-optane-memory.html>) qui est a priori le fichier de base installé par Windows Setup (différent de **boot.wim**, qui est juste le Windows PE lançant Windows Setup) probablement en utilisant en interne la commande **dism** comme on va le faire dans la suite). Il restera peut-être à installer certains drivers propres au PC, en général téléchargeables depuis le site du fabricant.

Ceci peut prendre plusieurs min.

Pendant ce temps, il devrait être possible de faire la partie suivante : **Configuration du BIOS/UEFI.**

Fermer ensuite toutes les fenêtres de l'explorateur affichant **K:** et éjecter le SSD USB.

2.4 Configuration du BIOS/UEFI

Connecter tout le nécessaire (il devrait être possible de garder branché sur le même écran le PC de la salle info et le PC fourni et de passer de l'un à l'autre en utilisant les touches de l'écran si l'écran à 1 port branché sur le PC de la salle info et un 2ème port branché sur le PC fourni, seuls clavier et souris devront être passés de l'un à l'autre) et régler les paramètres courants du BIOS/UEFI comme indiqué dans le cours. Les paramètres importants seront a priori :

- Load optimized default settings
- Réglage de l'heure et date
- Il y a 1 option UEFI et 1 option Legacy à cocher, les 2 pouvant être cochées en même temps, si les 2 sont cochées, les périphériques bootables sont listés 2 fois, avec UEFI au début si c'est détecté comme mode UEFI (a priori mettre le disque interne en premier)
- Network boot disabled or set to last (if needed)
- After power failure power on
- Failsafe watchdog disabled
- Suppress alert messages at boot enabled

Note : lors d'une installation standard de Windows (e.g. via une clé USB préparée avec <https://www.microsoft.com/fr-fr/software-download/windows10>) il est possible que Windows indique qu'il n'arrive pas à accéder correctement au disque dur interne si celui-ci n'est pas listé en premier dans le BIOS/UEFI.

2.5 Démarrage sous Windows PE

Brancher le SSD préparé pour l'installation de Windows.
Démarrer le PC et appuyer sur **F2** ou **F10** pour pouvoir choisir de démarrer temporairement sur le SSD USB (dans catégorie **Hard Disk**) en mode **UEFI**.

Choisir **Windows PE x64 UEFI** dans le menu de BOOTMGR.

Dans l'invite de commandes (attention : le clavier est supposé QWERTY par défaut donc le 'a' est en lieu et place du 'q', 'w' à la place de 'z', ':' à la place de 'm', etc.), taper **wpeutil SetKeyboardLayout 0409:0000040c start**

pour ouvrir une nouvelle invite de commandes. Celle-ci reconnaitra bien le clavier normalement maintenant. **Attention : ne pas fermer la première invite de commande sinon l'ordinateur redémarrera !**

2.6 Partitionnement du disque dur à restaurer en mode UEFI+GPT

Dans l'invite de commandes, taper :

diskpart

Puis dans diskpart :

list disk

En fonction de la liste qui s'affiche, déterminer le numéro du disque correspondant au SSD USB et le sélectionner avec la commande :

select dis 9

9 étant un exemple de numéro de disque. Sélectionner ensuite sa première partition et changer sa lettre d'accès à **P** (choisir une autre lettre si déjà utilisée) avec les commandes :

select par 1

remove

assign letter P:

Refaire :

list disk

et en fonction de la liste qui s'affiche, déterminer le numéro du disque correspondant au disque interne du PC (qui fait 80-256 GB) et le sélectionner avec la commande :

select dis 8

8 étant un exemple de numéro de disque. Effacer toutes les partitions dessus (**attention : toujours vérifier attentivement si c'est le bon disque !**) avec la commande :

clean

et convertir le disque au format GPT :

convert gpt

Créer une nouvelle partition de **boot** FAT32, qui sera accessible sous la lettre **S** (choisir une autre lettre si déjà utilisée), ainsi que la partition **MSR** (qui sert probablement pour des données internes au fonctionnement du boot loader de Windows) :

create part efi size=100

format fs=fat32 quick

assign letter S:

create part msr size=16

Créer une nouvelle partition NTFS, qui sera accessible sous la lettre **C** (choisir une autre lettre si déjà utilisée) :

create part pri

format fs=ntfs quick

assign letter C:

Vérifier que tout a été créé comme voulu avec :

list part
list volume

Taper:

exit

pour quitter diskpart.

2.7 Restauration de la partition principale

Facultatif : accélérer la restauration de l'image en définissant le mode de gestion de l'alimentation **Performances élevées** :

powercfg /s 8c5e7fda-e8bf-4a96-9a85-a6e23a8c635c

Aller à la racine du SSD USB (ici sa lettre est **P:**, tester avec d'autres si c'est différent, les fichiers peuvent être listés avec la commande **dir**) :

cd /d P:

et taper :

md C:\Recycler\Scratch

dism /apply-image /imagefile:win.wim /index:1 /ApplyDir:C:\ScratchDir:C:\Recycler\Scratch

pour lancer l'extraction de l'installation de Windows fournie sur la partition créée.

Ceci prendra environ 15 min.

Pendant ce temps, il devrait être possible de faire la partie suivante : **Préparation d'un SSD USB avec Ubuntu à partir de Windows. Ne pas oublier de faire la partie Restauration du boot loader BOOTMGR quand la commande dism se terminera**, sinon Windows ne démarrera pas !

Note : Windows PE étant une version minimale de Windows destinée à lancer une installation ou restauration de Windows, il n'y a pas d'explorateur de fichiers. Une solution de contournement peut être de lancer :

notepad

et aller dans le menu **File\Open** pour pouvoir naviguer dans les dossiers (penser à sélectionner **All Files** dans la boîte de dialogue). On peut alors exécuter des programmes en faisant un clic droit dessus et **Run as Administrator**.

Note : Windows PE est déjà entièrement chargé en RAM donc il est théoriquement possible de retirer le SSD USB (si on ne l'utilise pas spécifiquement pour des copies de fichiers (comme c'est le cas ici) ou autre...).

Note : Un fichier **.wim** ou **.esd** peut contenir parfois plusieurs éditions de Windows, correspondant au paramètre **/index** à choisir dans la commande **dism /apply-image**. Une commande comme **dism /get-imageinfo /imagefile:win.wim** permet de lister les éditions et index correspondants si besoin. C'est notamment utile si on veut choisir d'installer manuellement une image **install.esd** d'un CD officiel de Windows, qui contient habituellement plusieurs éditions telles que Home, Pro, etc.

2.8 Préparation d'un SSD USB avec Ubuntu à partir de Windows

Connecter un des SSD USB fourni à un PC sous Windows avec les droits administrateurs et voir <http://rufus.ie> pour préparer le SSD avec **Ubuntu 22.04 64 bit** (disponible depuis un PC de salle info (disques réseau) sur **enseignement\rob\...\Robotique pratique** (regarder dans **public\share** si **enseignement\rob** n'est pas disponible) ou e.g. <http://ubuntu.mirrors.ovh.net/ubuntu-releases/>). Choisir NTFS comme File system pour supporter les fichiers de plus de 4 GB.

A ce stade, il est possible d'éjecter le SSD USB pour l'utiliser sur un PC de salle info pour copier les fichiers suivants.

Copier sur le SSD le dossier **boot-repair** dessus et extraire à la racine du SSD l'exécutable **fsarchiver** version **x86_64** et **static** (static version is supposed to be an almost self-contained executable with few external dependencies) disponible sur <https://github.com/fdupoux/fsarchiver/releases/> . Copier ensuite depuis un PC de salle info (disques réseau) le fichier **rootfs.fsa** disponible sur **enseignement\rob\...\Robotique pratique\imvmubuntu64** (regarder dans **public\share** si **enseignement\rob** n'est pas disponible). Ce fichier de plusieurs GB contient une copie compressée d'une installation d'Ubuntu déjà faite sur une machine virtuelle VMware, avec quelques réglages et applications utiles pour la robotique (voir partie **Capturer l'image Ubuntu** pour plus d'infos). Il restera peut-être à installer certains drivers propres au PC.

Note: le fait de choisir NTFS comme système de fichier pour le SSD USB pourrait poser des problèmes de compatibilité avec certains BIOS ou UEFI. En cas de problème, il faudra plutôt choisir FAT32 (a priori proposé par défaut par Rufus) et soit utiliser une autre clé USB en NTFS ou ext3/4 pour pouvoir contenir le fichier rootfs.fsa (qui est trop gros pour être supporté par FAT32), soit essayer de repartitionner le SSD USB après l'utilisation de Rufus pour y rajouter une partition pouvant contenir rootfs.fsa (mais en pratique il semble que les outils de partitionnement refusent de le repartitionner potentiellement car Rufus n'a pas préparé de MBR ou GPT comme sur un disque interne et qu'il n'y a qu'un VBR, à noter que si besoin la commande Windows **convert K: /FS:NTFS /NoSecurity /X** peut convertir une partition K: (son **volume label**, e.g. affiché dans ses propriétés, sera aussi demandé) de FAT32 à NTFS a priori sans perte de données)...

2.9 Restauration du boot loader BOOTMGR

Une fois la partition restaurée avec les fichiers de Windows et les applications, pour rendre bootable le disque interne, taper la commande :

```
C:\Windows\System32\bcdboot C:\Windows /s S: /f UEFI
```

Note : certains paramètres du bootloader doivent parfois être mis à jour (dans le store **EFI\Microsoft\BOOT\BCD** quand on est en mode UEFI+GPT ou dans **BOOT\BCD** quand on est en BIOS+MBR, il faut préciser explicitement le store sinon par défaut ce serait celui de Windows PE qui serait modifié, au lieu de celui du Windows qu'on cherche à restaurer).

Taper :

```
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /enum
```

pour afficher les paramètres courants. Si nécessaire, taper :

```
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} osdevice partition=C:
```

```
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} device partition=C:
```

et vérifier les changements avec:

```
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /enum
```

pour mettre à jour les informations sur la partition contenant des infos pour le boot loader et la partition contenant l'OS (typiquement celle contenant le dossier Windows).

Autres options potentiellement utiles pour un robot :

```
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} nx AlwaysOff  
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /displayorder {ntldr} /addlast  
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /timeout 3  
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} bootmenupolicy legacy  
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} recoveryenabled NO  
bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} bootstatuspolicy  
ignoreallfailures
```

Pour redémarrer, fermer toutes les fenêtres. Si tout a bien fonctionné, le SSD USB peut alors être réutilisé pour d'autres choses, notamment la partie **Préparation d'un SSD USB avec Ubuntu à partir de Windows** si un autre SSD USB n'était pas disponible.

Après redémarrage, il faudra mettre à jour le nom réseau de l'ordinateur (voir étiquette dessus) en allant dans le **Control Panel\System and Security\System\Remote settings\Computer Name\Change** et redémarrer.

Identifiant : **Administrator**

Password : **Workstation**

Note : pour faire les choses proprement, il faudrait dans l'idéal laisser Windows accéder à Internet et installer tous les drivers qu'il peut, et ensuite installer les drivers manquants via le site du fabricant, vérifier aussi les options d'alimentation (e.g. désactiver mise en veille automatique, désactiver **Fast startup**), etc.

Pour info, l'installation restaurée a été faite approximativement avec les logiciels listés dans <https://www.ensta-bretagne.fr/lebars/Share/Unattended.bat> (voir aussi certaines infos ici : <https://www.ensta-bretagne.fr/lebars/Share/Windows.txt>).

Facultatif : voir comment monter sous **S:** la partition de **boot** et regarder un peu ce qu'il y a dedans...

On peut ensuite préparer l'installation d'Ubuntu en réduisant la taille de la partition Windows d'au moins 25 Go via **Disc Management** ou la commande **diskpart** (chercher comment utiliser l'un ou l'autre pour y parvenir)...

Facultatif : créer une partition de **recovery** (voir [https://msdn.microsoft.com/fr-fr/library/windows/hardware/dn898510\(v=vs.85\).aspx](https://msdn.microsoft.com/fr-fr/library/windows/hardware/dn898510(v=vs.85).aspx)) juste pour montrer que ça ne pose pas de problèmes majeurs qu'elle soit là ou non (redémarrer après création et après suppression), sauf pour les PCs utilisant WimBoot (<https://docs.microsoft.com/en-us/windows/win32/w8cookbook/windows-image-file-boot--wimboot-> , pour certains PCs avec SSD ou disque dur système de faible capacité, e.g. 64 GB).

Note : dans ce scénario de restauration d'image Windows sur différentes machines, il est possible de gérer les clés de produit Windows avec la commande **slmgr**. En particulier, on doit en général remettre à 0 la période limite d'activation avec la commande :
slmgr /rearm

(cette opération ne peut être faite que 3 fois sur une même installation). Ne pas lancer l'activation de Windows s'il le demande car ce sont des installations temporaires.

2.10 Configuration du réseau (nécessite une connexion au réseau sur câble ou via le réseau Wi-Fi iot (ou DARTAP))

Note : si on n'a pas de périphériques réseaux adéquats disponibles pour fournir Internet sur câble à partir d'un PC portable qui a Internet via Wi-Fi et en plus une carte réseau Ethernet libre, on peut utiliser un PC portable Windows comme routeur avec la fonction partage de connexion internet (ICS pour Internet Connection Sharing) de Windows (ou éventuellement la fonction de pont réseau/network bridge) et un câble réseau croisé (il se peut aussi que les cartes réseaux gèrent les câbles droits dans cette configuration), chercher sur Internet si nécessaire et voir <https://www.ensta-bretagne.fr/lebars/Share/Windows.txt> pour certains détails.

Facultatif : Pour chaque connexion réseau, ajouter **ensta-bretagne.fr,ensieta.fr,ensieta.ecole** dans **TCP/IPv4 Properties\Advanced TCP/IP Settings\DNS\Append these DNS suffixes**.

OLD : Si problèmes d'accès à Internet liés au proxy de l'école, dans **Control Panel\Network and Internet\Internet Options\Connections\LAN settings** choisir **Automatically detect settings** ou **Use a proxy server** et mettre les réglages suivants dans **Advanced** :

http 192.168.1.17 8080

https 192.168.1.17 8080

ftp 192.168.1.17 8080

socks 192.168.1.10 822

ou

http 192.168.1.10 3128

https 192.168.1.10 3128

ftp 192.168.1.10 3128

socks 192.168.1.10 822

et rajouter **;*ensta-bretagne.fr;*ensieta.fr;*ensieta.ecole** dans **Do not use proxy**.

(il y a potentiellement plusieurs serveurs proxy à l'école selon les zones, etc.).

Malgré ces réglages, certaines applications particulières peuvent ne pas réussir à se connecter correctement au réseau, dans ce cas il faut les traiter au cas par cas. Par exemple, **Firefox** ne prend pas toujours en charge les réglages de proxy du système et nécessite sa propre configuration, lancer **Firefox** et dans **Options\Advanced\Network\Settings**, choisir **Auto-detect proxy settings for this network** ou mettre des réglages manuels semblables à ce qui a été fait précédemment.

De plus, il peut être nécessaire de taper :

```
set https_proxy=192.168.1.10:3128
```

```
set http_proxy=192.168.1.10:3128
```

```
set ftp_proxy=192.168.1.10:3128
```

```
set socks_proxy=192.168.1.10:822
```

```
set no_proxy=localhost,127.0.0.0/8,ensieta.ecole,ensieta.fr,ensta-bretagne.fr
```

dans une invite de commandes dans lequel on travaille pour que les commandes suivantes tapées dans cette invite de commandes puissent utiliser les réglages de proxy.

Vérifier que :

wget <http://www.google.fr>

télécharge bien un fichier **index.html**, sinon vérifier les différents réglages réseau.

2.11 Facultatif : accès aux lecteurs réseaux de l'école

Pour se connecter au dossier **public/share** de l'école et accéder à vos dossiers réseaux habituels (nécessite d'être connecté à un réseau où ils sont accessibles), s'inspirer de <https://www.ensta-bretagne.fr/lebars/Share/Network%20drives%20ENSIETA.bat> : changer **lebarsfa** par votre **identifiant** et **utilisateurs1** par utilisateurs2,3,4,5,6,21,22,23 ou 24 (à tester ou chercher dans les propriétés de **U:** sur un PC de salle info sous Windows), de plus **webperso**, **recherche** et **labos** ne sont peut-être pas accessibles par des étudiants. Les identifiants à utiliser seront vos identifiants habituels sur les PC de l'école précédé de **ENSIETA**, e.g. :

User name : **ENSIETA\lebarsfa**

Password : **XXXXXX**

C'est normal si le mot de passe ne s'affiche pas du tout à mesure qu'on le tape.

Note de sécurité informatique : toujours être vigilants quand on vous demande de rentrer vos identifiants dans un contexte inhabituel !

2.12 Accès à distance du PC installé à partir d'un PC de salle info

On peut se connecter à distance au bureau du PC installé avec le protocole RDP (attention le port par défaut est 3389 mais ici il a été mis sur le port 443 (ou 80) car le 3389 est parfois bloqué par des pare-feu). Exécuter **mstsc /admin** et taper **PC-NAME:443** dans **Computer** (remplacer **PC-NAME** par le nom réseau de l'ordinateur défini plus haut (voir étiquette sur le PC)).

Utiliser **PuTTY** pour accéder en **SSH** au PC. Ceci permet d'exécuter des commandes dessus à distance (attention : si la connexion est interrompue, les commandes en cours d'exécution éventuelles seront aussi stoppées...).

Le PC installé est dans le groupe de travail **ROBOTICS**, son nom réseau a été défini plus haut (voir étiquette sur le PC) et a un dossier réseau **Share**, ainsi qu'un partage caché **C\$** de la partition **C:**. Chercher comment y accéder avec l'outil **Connecter un lecteur réseau** de Windows pour pouvoir copier des fichiers. Tester aussi avec **FileZilla** pour se connecter au PC en **SFTP** et **FTP**.

Note de sécurité informatique : sur un PC partagé, il peut être nécessaire d'effacer les identifiants enregistrés. Certains (lecteurs réseaux, RDP) sont regroupés sous Windows et gérés en tapant la commande **control userpasswords2, dans **Advanced\Manage Passwords**.**

2.13 Démarrage d'Ubuntu en RAM

Brancher le SSD USB préparé pour l'installation d'Ubuntu.

Démarrer le PC et appuyer sur **F2** ou **F10** pour pouvoir choisir de démarrer temporairement sur le SSD USB (dans catégorie **Hard Disk**) en mode **UEFI**.

Le programme d'installation d'Ubuntu démarre, choisir **Try Ubuntu** pour qu'il soit juste chargé en RAM.

Note : par défaut, le SSD USB doit rester quand même branché car il n'est pas entièrement chargé en RAM. Si l'ordinateur a suffisamment de RAM, il est possible de tout charger le nécessaire en RAM :

- Highlight **Try Ubuntu** using the arrow keys and then press **E** to edit the boot script.
- Search for the line starting with linux and add the option **toram noprompt** instead of **splash**.
- Press **F10** to boot the modified script.
- When Ubuntu Desktop appears, you can eject/unmount the USB disk, so that it can be used on another computer.

Un fois le bureau apparu, changer la **langue d'entrée du clavier** s'il n'est pas bien reconnu (dans la barre de notifications, aller dans **Settings\Keyboard** pour rajouter un **Input source** Français, changer l'ordre de priorité ou supprimer les autres si besoin).

2.14 Partitionnement du disque dur à restaurer

Dans un terminal :

sudo gparted

pour pouvoir partitionner le disque dur interne (qui fait 80-256 GB).

Les **disques durs** SATA sont en général nommés **/dev/sda** pour le 1^{er}, **/dev/sdb** pour le 2nd (parfois aussi **/dev/hda**, **/dev/hdb**). Sur le disque **/dev/sda**, la **première partition primaire** sera nommée **/dev/sda1**, la 2^{ème} **/dev/sda2** (voir e.g. <https://doc.ubuntu-fr.org/partitions> pour plus de détails). Un disque NVMe est en général nommé **/dev/nvme0n1** et sa première partition sera **/dev/nvme0n1p1**.

Faire **unmount** si nécessaire sur les partitions du disque **/dev/sda** (**attention : toujours vérifier attentivement si c'est le bon disque !**) et créer une partition **ext3** ou **ext4** après la partition Windows. Ne pas oublier de cliquer sur **Apply All Operations**, puis quitter **GParted**.

Note : il se peut que des changements manuels soient nécessaires dans le fichier **/etc/fstab** de l'installation d'Ubuntu qu'on restaure (UUID des partitions, suppressions de lignes faisant référence à des partitions qui n'existent plus, e.g. si la partition EFI a changé, remplacer l'UUID correspondant à **/boot/efi** dans **/etc/fstab** avec celui donné par la commande **blkid | grep -i efi**)...

2.15 Restauration de la partition principale

S'assurer que **/dev/sda4** n'est pas en cours d'utilisation et correspond bien à la partition Ubuntu voulue :

sudo umount /dev/sda4

et se placer dans le dossier du SSD :

cd /isodevice

ou

cd /media/cdrom

et s'assurer que la commande **fsarchiver** est exécutable :

chmod +x fsarchiver

Restaurer la partition principale sauvegardée avec (**attention : bien vérifier que c'est la bonne partition !**) :

```
sudo ./fsarchiver -j2 restfs rootfs.fsa id=0,dest=/dev/sda4
```

Ceci prendra plusieurs min.

Pendant ce temps, il devrait être possible de faire la partie suivante optionnelle :

Configuration du réseau. **Penser à revenir ensuite ici pour les étapes suivantes !**

Note : les avantages de la commande **fsarchiver** par rapport à **dd** (commande bas niveau classique permettant de lire/écrire bit par bit un disque dur) sont :

- Les données sont compressées, et les zones vides de la partition sont ignorées, contrairement à **dd**.
- La partition cible peut être de n'importe quelle taille du moment qu'elle est capable de contenir les données extraites, alors qu'avec **dd**, il est impossible de faire la restauration sur une partition plus petite que l'originale, même si elle n'était pas remplie entièrement, car on ne peut pas prévoir où sont stockées les données sur la partition, si on ne copie pas la partition entière on aura probablement des fichiers corrompus...

Les inconvénients sont que l'on doit recréer soi-même les partitions, reconfigurer le boot loader et s'assurer que c'est cohérent avec ce qu'il y a dans l'image disque à restaurer.

Une fois la partition restaurée, on va pouvoir y accéder avec :

```
sudo mkdir -p /mnt/sda4
sudo mount /dev/sda4 /mnt/sda4
```

Ses données seront alors visibles dans **/mnt/sda4**.

Lister les UUID des partitions (sortes d'identifiants) avec :

```
sudo ls -l /dev/disk/by-uuid
```

et corriger si nécessaire les correspondances dans le fichier suivant :

```
sudo gedit /mnt/sda4/etc/fstab
```

Mettre à jour le nom réseau de l'ordinateur (voir étiquette dessus) :

```
sudo gedit /mnt/sda4/etc/hostname
```

Changer aussi si nécessaire le nom à la ligne 127.0.1.1 avec le nouveau nom réseau dans :

```
sudo gedit /mnt/sda4/etc/hosts
```

2.16 Facultatif : configuration du réseau

Se connecter au réseau Wi-Fi **iot** (ou **DARTAP**).

OLD : Si problèmes d'accès à Internet liés au proxy de l'école, dans **Settings\Network\Network proxy** choisir **Manual** et mettre les réglages suivants et cliquer sur **Apply system wide** :

```
http 192.168.1.17 8080
```

```
https 192.168.1.17 8080
```

```
ftp 192.168.1.17 8080
```

```
socks 192.168.1.10 822
```

ou

```
http 192.168.1.10 3128
https 192.168.1.10 3128
ftp 192.168.1.10 3128
socks 192.168.1.10 822
```

Malgré ces réglages, certaines applications particulières peuvent ne pas réussir à se connecter correctement au réseau, dans ce cas il faut les traiter au cas par cas. Par exemple, la commande **apt-get** a son propre fichier de configuration

```
sudo gedit /etc/apt/apt.conf
```

(il y a aussi parfois `/etc/apt/apt.conf.d/proxy`) et mettre ce qui est dans <http://www.ensta-bretagne.fr/lebars/Share/apt.conf> . Pour y accéder, il faudra lancer **Firefox** et dans

Options\Advanced\Network\Settings, choisir **Auto-detect proxy settings for this network** (**Firefox** aussi ne prend pas toujours en charge les réglages de proxy du système et nécessite sa propre configuration).

De plus, il peut être nécessaire de taper :

```
export https_proxy=192.168.1.10:3128
export http_proxy=192.168.1.10:3128
export ftp_proxy=192.168.1.10:3128
export socks_proxy=192.168.1.10:822
```

```
export no_proxy=localhost,127.0.0.0/8,ensieta.ecole,ensieta.fr,ensta-bretagne.fr
```

dans le terminal dans lequel on travaille pour que les commandes suivantes tapées dans ce terminal puissent utiliser les réglages de proxy. Si les commandes utilisent **sudo**, il faudra peut-être utiliser **sudo -E** pour qu'elles prennent en compte les variables définies.

Vérifier que :

```
sudo apt-get update
```

se passe correctement, sinon vérifier les différents réglages réseau.

De plus, dans le menu **Activities**, il faudra peut-être lancer **Software & Updates** et activer les dépôts **main**, **universe**, **restricted**, **multiverse** pour avoir accès aux paquets habituellement disponibles sur une version non-Live d'Ubuntu.

Pour se connecter au dossier **public/share** de l'école et accéder à vos dossiers réseaux habituels (nécessite d'être connecté à un réseau où ils sont accessibles), s'inspirer de https://www.ensta-bretagne.fr/lebars/Share/remote_ensieta.sh : changer **lebarsfa** par votre **identifiant** et **utilisateurs1** par utilisateurs2,3,4,5,6,21,22,23 ou 24 (à tester ou chercher dans les propriétés de **U**: sur un PC de salle info sous Windows), de plus **webperso**, **recherche** et **labos** ne sont peut-être pas accessibles par des étudiants. Les identifiants à utiliser seront vos identifiants habituels sur les PC de l'école. C'est normal si le mot de passe ne s'affiche pas du tout à mesure qu'on le tape.

Note de sécurité informatique : toujours être vigilants quand on vous demande de rentrer vos identifiants dans un contexte inhabituel !

Pour info, ces étapes de configuration du réseau sont en principe optionnelles et souvent complexes quand on est sur un réseau d'école ou d'entreprise, pour aller plus vite il est préférable de tout préparer le nécessaire sur une clé USB.

2.17 Restauration du boot loader GRUB

Installer et lancer l'outil **Boot Repair** (au choix : soit avec la méthode en commentaires qui télécharge via réseau, soit via les paquets déjà téléchargés, besoin de proxy bien réglé ou alors se déconnecter de tout réseau sinon bloque parfois, rajouter des **-E** dans **sudo** des commandes s'il demande de taper des commandes manuellement...):

```
## sudo -E might be needed to inherit environment variables depending on how proxy settings were set...
```

```
#sudo -E add-apt-repository ppa:yannubuntu/boot-repair
```

```
#sudo -E apt-get update
```

```
#sudo -E apt-get install -y boot-repair
```

```
cd /media/cdrom/boot-repair
```

```
sudo dpkg -i libx86emu3_3.1-2_amd64.deb libhd21_21.72-1_amd64.deb
```

```
glade2script_3.2.4_ppa23_all.deb glade2script-python3_3.2.4_ppa23_all.deb
```

```
pastebinit_1.5.1-1ubuntu1_all.deb hwinfo_21.72-1_amd64.deb
```

```
sudo dpkg -i boot-info_4ppa2056_all.deb boot-sav_4ppa2056_all.deb boot-
```

```
repair_4ppa2056_all.deb boot-sav-extra_4ppa2056_all.deb os-
```

```
uninstaller_4ppa2056_all.deb efibootmgr_17-1ubuntu2_amd64.deb grub-
```

```
customizer_5.2.3-0ubuntu1_ppa1j_amd64.deb
```

```
sudo -E boot-repair
```

Aller dans **Advanced**, vérifier que les options paraissent cohérentes (partition de boot sur **/dev/sda1**, lancement de l'OS sur **/dev/sda4**, éventuellement désactiver les log files et autres options nécessitant Internet) et cliquer sur **Apply**. Ceci peut prendre plusieurs minutes...

Facultatif : Après redémarrage, il se peut que les cartes réseaux aient été réinstallées. Pour chaque connexion réseau, changer **IPv4 Settings\Additional search domains** avec **ensta-bretagne.fr,ensieta.fr,ensieta.ecole** si nécessaire.

De plus, pour éviter les conflits avec les réglages d'heure de Windows, on peut régler Ubuntu pour qu'il suppose que l'heure du BIOS/UEFI est locale au lieu de UTC :

```
timedatectl set-local-rtc 1 --adjust-system-clock
```

Il faudra peut-être encore corriger l'heure 1 ou 2 fois après redémarrage sous l'un et l'autre.

Pour info, l'installation restaurée a été faite approximativement avec les réglages listés ici : <https://www.ensta-bretagne.fr/lebars/Share/Ubuntu.txt> (qui fait notamment appel à <https://www.ensta-bretagne.fr/lebars/Share/unattended.sh>).

Identifiant : **user**

Password : **Workstation**

2.18 Facultatif : accès à distance du PC installé à partir du PC de l'école

Sur le PC de salle info, installer **NoMachine Enterprise Client** (voir <https://www.nomachine.com/download-enterprise>) dans **C:\Temp** et trouver comment l'utiliser pour se connecter au bureau du PC installé...

Lancer **Xming** et utiliser **PuTTY** pour accéder en **SSH** au PC (cocher l'option **Enable X11 forwarding** dans les options de **PuTTY**). Ceci permet d'exécuter des commandes dessus à distance (attention : si la connexion est interrompue, les commandes en cours d'exécution éventuelles seront aussi stoppées...). Tester e.g. la commande :

nautilus &

Le PC installé est dans le groupe de travail **ROBOTICS**, son nom réseau a été défini plus haut (voir étiquette sur le PC) et a un dossier réseau **Share**. Chercher comment y accéder avec l'outil **Connecter un lecteur réseau** de Windows pour pouvoir copier des fichiers. Tester aussi avec **FileZilla** pour se connecter au PC en **SFTP**.

2.19 Ajout de la possibilité de redémarrer sous Ubuntu à partir de Windows lorsqu'on est en mode UEFI+GPT

Facultatif : sous Windows utiliser **diskpart** et **assign letter=s** pour voir comment la partition de boot a été modifiée, ouvrir avec Wordpad grub.cfg.

Note : en mode UEFI+GPT, on ne peut parfois pas rajouter de GRUB 2 Entry Ubuntu dans la liste proposée par BOOTMGR pour des raisons de sécurité (la logique de Microsoft est probablement qu'un produit non signé par Microsoft est potentiellement non sécurisé et doit donc être bloqué à ce niveau critique du démarrage de l'ordinateur)...

Note : Voir aussi <http://www.rodsbooks.com/refind/index.html> (et <https://askubuntu.com/a/484456> (section « For UEFI devices »)) pour des outils qui pourraient potentiellement permettre d'installer Ubuntu à partir de commandes tapées sous Windows et sans avoir besoin d'accéder aux options de l'UEFI et sans clé USB...

Facultatif : observer la sortie de **bcdedit /enum all** et essayer de reconnaître certains éléments...

Note : pour changer le timeout de BOOTMGR : **bcdedit /timeout 3**

Note : pour éditer facilement BOOTMGR ou vérifier sa configuration : **easybcd, bootice**

Note: taper:
%systemroot%\sysnative\Bootsect /nt60 c: /uefi
pour réinstaller les parties chargées en premier par BOOTMGR.

Taper :
bcdedit /set {fwbootmgr} default {bootmgr}
pour forcer le démarrage par défaut avec le boot loader de Windows (BOOTMGR).

Taper :
bcdedit /enum firmware
et en fonction de la sortie remplacer **{db09802d-3e9d-11e8-bbcb-000c290a0070}** dans **bcdedit /set {fwbootmgr} default {db09802d-3e9d-11e8-bbcb-000c290a0070}**
pour forcer le démarrage par défaut avec le boot loader d'Ubuntu (GRUB 2).

Note : si on a supprimé la possibilité de démarrer sous Ubuntu (voir la sortie de **bcdedit /enum firmware**), en supposant qu'on a bien le dossier **\EFI\ubuntu** sur la partition EFI (sinon, utiliser **boot-repair**) et la partition d'Ubuntu toujours bons on peut restaurer l'entrée à partir de Windows (remplacer **{aeb1e0aa-3df8-11e8-bb04-000c290a0070}** par la valeur retournée par la 1^{ère} commande) :

```
bcdedit /copy {bootmgr} /d "Ubuntu"  
bcdedit /set {aeb1e0aa-3df8-11e8-bb04-000c290a0070} path \EFI\ubuntu\shimx64.efi  
bcdedit /set {fwbootmgr} displayorder {aeb1e0aa-3df8-11e8-bb04-000c290a0070}  
/addfirst
```

2.20 Ajout de la possibilité de redémarrer sous Windows à partir de Ubuntu lorsqu'on est en mode UEFI+GPT

Note : pour changer le timeout de GRUB 2, modifier GRUB_HIDDEN_TIMEOUT et GRUB_TIMEOUT:

```
sudo nano /etc/default/grub  
sudo update-grub
```

Note : Pour éditer facilement GRUB 2 ou vérifier sa configuration :

```
sudo -E add-apt-repository ppa:danielrichter2007/grub-customizer  
sudo -E apt-get update  
sudo -E apt-get install -y grub-customizer  
sudo -E grub-customizer
```

Taper :

```
sudo efibootmgr  
et en fonction de la sortie remplacer 5,4,0,1,2,3 dans  
sudo efibootmgr -o 5,4,0,1,2,3  
pour changer l'ordre de démarrage des boot loaders.
```

Note : s'il y a des entrées dupliquées, on peut supprimer les entrées recréées automatiquement avec e.g. **sudo efibootmgr -b 9 -B**

Note : si on a supprimé la possibilité de démarrer sous Ubuntu (voir la sortie de **sudo efibootmgr**), en supposant qu'on a bien le dossier **\EFI\ubuntu** sur la partition EFI (sinon, utiliser **boot-repair**) et la partition d'Ubuntu toujours bons on peut restaurer l'entrée à partir d'Ubuntu Live :

```
sudo efibootmgr -c -g -l \\EFI\ubuntu\shimx64.efi -d /dev/sda -L Ubuntu
```

Facultatif : se connecter à distance via **ssh** et faire le nécessaire pour passer d'un OS à l'autre en ligne de commande sans toucher physiquement au PC contrôlé (s'inspirer éventuellement des scripts sur <https://www.ensta-bretagne.fr/lebars/UEFI.zip>).

2.21 Passer de UEFI+GPT à BIOS+MBR sans tout réinstaller

Sous Ubuntu Live (en redémarrant sur le SSD USB déjà préparé), supprimer la partition de boot et MSR de 16 Mo (ne pas supprimer les partitions principales Windows et Ubuntu contenant l'essentiel des systèmes d'exploitation). Convertir le disque en mode MBR avec:

```
gdisk /dev/sda  
and type in the prompt  
rgwy
```

Redémarrer et dans le BIOS/UEFI mettre mode BIOS, legacy CSM, désactiver UEFI boot, etc.

Sous Windows PE (s'assurer avant de formater que c'est bien les bonnes partitions qu'on formate et qu'elles ont les bons numéros associés aux bonnes lettres!) :

```
diskpart
list disk
select disk 0
create part pri
list part
list volume
format fs=fat32 quick
active
assign letter S:
select part 2
inactive
assign letter C:
select part 3
inactive
list part
list volume
exit
```

et

```
bootsect /nt60 S: /mbr
```

```
C:\Windows\System32\bcdboot C:\Windows /s S: /f BIOS
```

pour recréer la partition de boot de Windows avec BOOTMGR.

Sous Ubuntu Live, utiliser **boot-repair** pour réinstaller GRUB 2 (s'assurer qu'il l'installe sur **/dev/sda** (MBR) et aussi sur **/dev/sda3** (VBR de la partition Ubuntu) éventuellement en le relançant 2 fois, pour déjà préparer la partie suivante, éventuellement désactiver les log files et autres options nécessitant Internet) et vérifier que les 2 OS redémarrent bien.

Éventuellement utiliser **grub-customizer** (sous Ubuntu) et/ou **bcdedit /enum** et **bcdedit /delete** (sous Windows) pour supprimer les entrées de boot inutiles.

2.22 Ajout de la possibilité de redémarrer sous Ubuntu à partir de Windows et inversement lorsqu'on est en mode BIOS+MBR

Sous Windows, installer **easybcd** et ajouter une **GRUB 2 Entry** Ubuntu, et désactiver l'option **Metro bootloader** pour rendre plus rapide le menu.

Note : si ne veut utiliser que des commandes de base (remplacer **{9bcde5bf-256f-11e7-b6b8-001018ea0fce}** par la valeur retournée par la 1^{ère} commande) :

```
bcdedit /create /d "Ubuntu" /application bootsector
```

```
bcdedit /set {9bcde5bf-256f-11e7-b6b8-001018ea0fce} device partition=c:
```

```
bcdedit /set {9bcde5bf-256f-11e7-b6b8-001018ea0fce} path \mbr_grub.bak
```

```
bcdedit /displayorder {9bcde5bf-256f-11e7-b6b8-001018ea0fce} /addlast
```

et sous Ubuntu sauvegarder le MBR :

```
sudo dd=if=/dev/sda count=1 bs=512 of=mbr_grub.bak
```

et mettre le fichier **mbr_grub.bak** à la racine de la partition Windows.

Note : GRUB 2 étant souvent juste installé sur le MBR, il faut aussi dans ce cas s'assurer qu'il soit aussi installé sur le VBR de la partition Ubuntu (normalement c'est ce qu'on a fait), si ce n'est pas le cas faire sous Ubuntu Live :

sudo grub-install --root-directory=/media/ubuntu/XXX /dev/sda3 --force
sudo update-grub

(dans **nautilus** double-cliquer sur le volume correspondant à la partition Ubuntu pour qu'il se monte et remplacer **XXX** par son identifiant qui apparaît dans **/media/ubuntu**). Si on souhaite le réinstaller sur le MBR, remplacer **/dev/sda3** par **/dev/sda**. Si nécessaire, refaire **sudo update-grub** sur le Ubuntu installé.

Note : en mode UEFI+GPT, BOOTMGR ne nous autorise pas à lister des entrées non-Microsoft, en mode BIOS+MBR il est plus permissif...

Taper:

%systemroot%\sysnative\Bootsect /nt60 S: /mbr

pour réinstaller les parties chargées en premier par BOOTMGR et forcer le démarrage par défaut avec le boot loader de Windows.

Il suffit alors juste de changer la partition active (avec **diskpart** sous Windows, **gparted** or **fdisk** sous Ubuntu) pour choisir de démarrer par défaut avec l'un ou l'autre des bootloaders (bien s'assurer que les autres partitions sont inactive, il ne faut pas qu'il y ait plusieurs partitions active en même temps) et du coup l'un ou l'autre des OS selon l'ordre selon lequel chacun est listé dans chaque boot loader (utiliser les commandes **bcdedit /default** sous Windows, **sudo grub-set-default** sous Ubuntu si nécessaire, voir plus bas).

Attention, si une mise à jour de GRUB 2 se produit (e.g. après un **sudo apt-get upgrade**), il est possible que le code sur le MBR soit remplacé par celui de GRUB 2 et dans ce cas le choix de démarrage par défaut ne peut plus se faire en changeant la partition active. Dans ce cas, il faut retaper **%systemroot%\sysnative\Bootsect /nt60 S: /mbr** sous Windows, ou taper **sudo apt-get install syslinux; sudo dd if=/usr/lib/syslinux/mbr/mbr.bin of=/dev/sda** sous Ubuntu pour restaurer un MBR qui convient (bien vérifier et changer si besoin **S:** et **/dev/sda** avec les bons noms). Mais idéalement pour éviter que cette situation se produise, faire **sudo debconf-show grub-pc**, **sudo dpkg-reconfigure grub-pc** and change the place where it should install, see <https://ubuntuforums.org/showthread.php?t=2189643>.

Note: sous Ubuntu on peut faire :

sudo grub-reboot 4 && sudo reboot

pour redémarrer temporairement sous Windows et

sudo grub-set-default 4

pour forcer le démarrage par défaut sous Windows (si Windows est la 4ème **menuentry** dans **/boot/grub/grub.cfg**).

Note: sous Windows (mais ne semble pas avoir toujours d'effet...) on peut faire (remplacer **{975a8204-9658-11dd-993e-9aea7965e9da}** par la bonne valeur retournée par **bcdedit /enum firmware**) :

bcdedit /bootsequence {975a8204-9658-11dd-993e-9aea7965e9da} & shutdown /r /t 0

pour redémarrer temporairement sous Ubuntu et

bcdedit /default {975a8204-9658-11dd-993e-9aea7965e9da}

pour forcer le démarrage par défaut sous Ubuntu.

Facultatif : se connecter à distance via **ssh** et faire le nécessaire pour passer d'un OS à l'autre en ligne de commande sans toucher physiquement au PC contrôlé (s'inspirer éventuellement des scripts sur <https://www.ensta-bretagne.fr/lebars/MBR.zip>).

2.23 Facultatif : ajout de la possibilité de démarrer Ubuntu sous Windows à partir d'une machine virtuelle (sans installer un 2^{ème} Ubuntu, pour configuration BIOS+MBR)

Attention : peu fiable, risque de corruption des données !

Faire les réglages nécessaires pour que le PC démarre par défaut sur Ubuntu (car le MBR courant va être recopié par la machine virtuelle).

Installer **VMware Workstation** sous Windows et le lancer via clic droit sur **Run as administrator** de préférence. Créer une nouvelle machine virtuelle avec **BIOS** (attention, il se peut que ce soit un **UEFI** par défaut, supprimer la ligne **firmware = "efi"** dans le fichier **.vmx** de la machine virtuelle créée si nécessaire) et supprimer le disque dur par défaut puis en rajouter un en SATA (si le vrai disque est en SATA) et choisir l'option **Use a physical disk** avec le bon **PhysicalDrive** et choisir l'option **Use individual partitions** et choisir la ou les partitions propres à Linux. VMware va alors créer un disque dur virtuel en recopiant le contenu du MBR du vrai disque (mais pas forcément le VBR des partitions). Le reste des données est directement accédé sur la partition du vrai disque. S'il y a des erreurs au lancement d'Ubuntu, essayer de réparer GRUB 2.

Note : il peut être utile de rajouter **bios.bootdelay = 3000** dans le fichier **.vmx** de la machine virtuelle pour rajouter un délai supplémentaire pour pouvoir accéder plus facilement aux options du BIOS du PC virtuel.

Note : si la machine virtuelle n'accède pas bien à Internet, choisir une connexion réseau de type **Bridged**.

Note : si on souhaite installer une machine virtuelle dans la machine virtuelle, il faut rajouter **vhv.enable="TRUE"** dans le fichier **.vmx** de la machine virtuelle hôte pour de meilleures performances.

Il se peut que les cartes réseaux aient été réinstallées. Pour chaque connexion réseau, changer **IPv4 Settings\Additional search domains** avec **ensta-bretagne.fr,ensieta.fr,ensieta.ecole** si nécessaire.

Installer éventuellement **VMware Tools** si ce n'est pas déjà installé pour pouvoir facilement utiliser le copier-coller entre la machine virtuelle Ubuntu et Windows :

```
tar xvfz VMwareTools.x.x.x-xxxx.tar.gz
cd vmware-tools-distrib
sudo ./vmware-install.pl
```

Attention : ne pas monter la partition Ubuntu dans Windows (e.g. avec Ext2Fsd, Ext2IFS, il se peut même qu'on doive les désinstaller...) quand la machine virtuelle Ubuntu est lancée, risque de corruption de données ! De même, ne pas essayer de démarrer Windows à partir de la machine virtuelle Ubuntu ! Pour partager des fichiers, monter plutôt les différentes partitions via des protocoles réseaux.

2.24 Facultatif : ajout de la possibilité de démarrer Windows sous Ubuntu à partir d'une machine virtuelle (sans installer un 2^{ème} Windows, pour configuration BIOS+MBR)

Attention : peu fiable, risque de corruption des données !

Tout se passe de manière symétrique à la partie précédente (lancer **VMware Player** ou **VMware Workstation** avec **sudo vmplayer** ou **sudo vmware**).

Note : en cas de problème d'installation/lancement de VMware Player 12.5.9, installer <https://github.com/mkubeczek/vmware-host-modules/archive/player-12.5.9.zip> (ou e.g. p15.0.4.zip selon la version de VMware Player...).

2.25 Passer de BIOS+MBR à UEFI+GPT sans tout réinstaller

Attention : il ne faudra plus utiliser les machines virtuelles créées précédemment, risque de corruption des données.

Sous Ubuntu Live (en redémarrant sur le SSD USB déjà préparé), si nécessaire réduire de 100 Mo la dernière partition pour s'assurer qu'elle ne va pas jusqu'au bout du disque (mieux vaut modifier les partitions d'Ubuntu avec Ubuntu Live et les partitions de Windows avec Windows PE). Convertir le disque en mode GPT avec :

gdisk /dev/sda

and type in the prompt

wy

Note: **mbr2gpt /convert** peut faire la même chose (see <https://docs.microsoft.com/en-us/windows/deployment/mbr-to-gpt>)...

Redémarrer et dans le BIOS/UEFI mettre mode UEFI+GPT, etc.

Sous Windows PE (s'assurer avant de formater que c'est bien les bonnes partitions qu'on formate et qu'elles ont les bons numéros associés aux bonnes lettres!) :

diskpart

list disk

select disk 0

create part efi size=100

list part

list volume

format fs=fat32 quick

assign letter S:

create part msr size=16

sel part 3

assign letter C:

list part

list volume

exit

et

C:\Windows\System32\bcdboot C:\Windows /s S: /f UEFI

pour recréer la partition de boot et MSR de Windows avec BOOTMGR.

Sous Ubuntu Live, utiliser **boot-repair** pour réinstaller GRUB 2 (partition de boot sur **/dev/sda1**, lancement de l'OS sur **/dev/sda4**, éventuellement désactiver les log files et autres options nécessitant Internet) et vérifier que les 2 OS redémarrent bien. Éventuellement utiliser **grub-customizer** et/ou **bcdedit /enum firmware** et **bcdedit /delete** pour supprimer les entrées de boot inutiles.

2.26 Facultatif : utilisation de l'outil testdisk pour récupérer les données de partitions effacées par erreur

Pour simuler une situation où on aurait effacé par erreur une partition de données importante, tester le formatage (uniquement formatage rapide, pas celui qui remplace chaque bit par 0 ou d'autres versions plus sécurisées ou qui changerait le type de système de fichiers !) des partitions de données (1 par 1), vérifier que les OS ne démarrent plus et utiliser **testdisk** pour réparer les partitions...

Note : si d'une manière ou d'une autre des données ont été réécrites par-dessus certaines données, celles-ci ne seront pas récupérables. Si ce n'est que la zone du système de fichiers qui définit l'arborescence de noms de dossiers et noms de fichiers (y compris extensions, dates de modification, permissions d'accès), normalement les données des fichiers eux-mêmes ont une chance d'être encore présentes mais ce sera assez difficile de distinguer rapidement les différents fichiers (éventuellement en regardant leur taille...).

2.27 Facultatif : ajout de la possibilité de démarrer Ubuntu sous Windows à partir d'une machine virtuelle (sans installer un 2^{ème} Ubuntu, pour configuration UEFI+GPT)

Attention : peu fiable, risque de corruption des données !

Faire les réglages nécessaires pour que le PC démarre par défaut sur Ubuntu (car le MBR courant va être recopié par la machine virtuelle, même si en fait il ne sera probablement pas utilisé).

Installer **VMware Workstation** sous Windows et le lancer via clic droit sur **Run as administrator** de préférence. Créer une nouvelle machine virtuelle avec **UEFI** (ajouter la ligne **firmware = "efi"** dans le fichier **.vmx** de la machine virtuelle créée si nécessaire) et supprimer le disque dur par défaut puis en rajouter un en SATA (si le vrai disque est en SATA) et choisir l'option **Use a physical disk** avec le bon **PhysicalDrive** et choisir l'option **Use individual partitions** et choisir la ou les partitions propres à Linux. VMware va alors créer un disque dur virtuel en recopiant le contenu du MBR du vrai disque (mais pas forcément le VBR des partitions). Le reste des données est directement accédé sur la partition du vrai disque. Ajouter ensuite un 2^{ème} disque dur SATA (si le 1^{er} disque est en SATA) de 150 MB, qui lui sera entièrement virtuel et contiendra une partition FAT32 pour le bootloader. Cette partition est nécessaire pour l'UEFI de la machine virtuelle et on ne peut pas réutiliser celle du vrai disque car il y aurait des conflits. Pour préparer cette partition, il va falloir démarrer la machine virtuelle avec un live CD D'Ubuntu (s'assurer que la version d'Ubuntu sur le live CD est la même que sur le disque dur) et utiliser **GParted** pour créer une **GPT part table** sur le disque dur virtuel de 150 MB (**/dev/sdb** a priori) et une partition FAT32 (**/dev/sdb1** a priori). Monter ensuite cette partition ainsi que celle contenant Ubuntu

(/dev/sda4 a priori). Taper (si besoin, activer les dépôts **main, universe, restricted, multiverse** dans **Software & Updates**) :

```
sudo apt install grub-efi
```

```
sudo grub-install --root-directory=/media/ubuntu/XXX /dev/sdb --efi-  
directory=/media/ubuntu/YYY
```

avec /media/ubuntu/XXX le point de montage de la partition contenant Ubuntu et /media/ubuntu/YYY celui de la partition FAT32 de 150 MB. Ensuite, utiliser **GPARTED** pour rajouter les **esp** et **boot** flags à cette partition (cela la cachera pour les utilisateurs normaux).

Note : un bug (de VMware ?) fait parfois que GRUB ne trouve pas les disques durs, aller dans le BIOS/UEFI et choisir la bonne entrée de boot manuellement.

Note : Ubuntu 18.04 plante parfois au cours du démarrage de la machine virtuelle dans ce cas essayer de le démarrer en recovery mode, peut-être que ça dépend du PC...

Note : il peut être utile de rajouter **bios.bootdelay = 3000** dans le fichier **.vmx** de la machine virtuelle pour rajouter un délai supplémentaire pour pouvoir accéder plus facilement aux options du BIOS du PC virtuel.

Note : si la machine virtuelle n'accède pas bien à Internet, choisir une connexion réseau de type **Bridged**.

Note : si on souhaite installer une machine virtuelle dans la machine virtuelle, il faut rajouter **vhv.enable="TRUE"** dans le fichier **.vmx** de la machine virtuelle hôte pour de meilleures performances.

Il se peut que les cartes réseaux aient été réinstallées. Pour chaque connexion réseau, changer **IPv4 Settings\Additional search domains** avec **ensta-bretagne.fr,ensieta.fr,ensieta.ecole** si nécessaire.

Installer éventuellement **VMware Tools** si ce n'est pas déjà installé pour pouvoir facilement utiliser le copier-coller entre la machine virtuelle Ubuntu et Windows :

```
tar xvfz VMwareTools.x.x.x-xxxx.tar.gz
```

```
cd vmware-tools-distrib
```

```
sudo ./vmware-install.pl
```

Attention : ne pas monter la partition Ubuntu dans Windows (e.g. avec Ext2Fsd, Ext2IFS, il se peut même qu'on doive les désinstaller...) quand la machine virtuelle Ubuntu est lancée, risque de corruption de données ! De même, ne pas essayer de démarrer Windows à partir de la machine virtuelle Ubuntu ! Pour partager des fichiers, monter plutôt les différentes partitions via des protocoles réseaux.

2.28 Facultatif : ajout de la possibilité de démarrer Windows sous Ubuntu à partir d'une machine virtuelle (sans installer un 2^{ème} Windows, pour configuration UEFI+GPT)

Attention : peu fiable, risque de corruption des données !

Presque tout se passe de manière symétrique à la partie précédente (lancer **VMware Player** ou **VMware Workstation** avec **sudo vmplayer** ou **sudo vmware**). A la place d'un live CD d'Ubuntu, utiliser Windows PE pour préparer le disque dur virtuel de 150 MB :

list disk

et en fonction de la liste qui s'affiche, déterminer le numéro de ce disque et le sélectionner avec la commande :

select disk 8

8 étant un exemple de numéro de disque. Effacer toutes les partitions dessus (attention : toujours vérifier attentivement si c'est le bon disque, normalement il est déjà vide) avec la commande :

clean

et convertir le disque au format GPT :

convert gpt

Créer une nouvelle partition de boot **FAT32**, qui sera accessible sous la lettre **S** (choisir une autre lettre si déjà utilisée), ainsi que la partition **MSR** :

create part efi size=100

format fs=fat32 quick

assign letter S:

create part msr size=16

S'assurer que **C:** correspond bien à la partition contenant Windows et installer le boot loader de Windows avec :

C:\Windows\System32\bcdboot C:\Windows /s S: /f UEFI

et mettre à jour les autres options de **bcdedit** si certaines ont été changées pour le vrai PC, e.g.

bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} nx AlwaysOff

bcdedit /store S:\EFI\Microsoft\BOOT\BCD /displayorder {ntldr} /addlast

bcdedit /store S:\EFI\Microsoft\BOOT\BCD /timeout 3

bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} bootmenupolicy legacy

bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} recoveryenabled NO

bcdedit /store S:\EFI\Microsoft\BOOT\BCD /set {default} bootstatuspolicy

ignoreallfailures

Note : en cas de problème d'installation/lancement de VMware Player 12.5.9, installer <https://github.com/mkubeczek/vmware-host-modules/archive/player-12.5.9.zip> (ou e.g. p15.0.4.zip selon la version de VMware Player...).

2.29 Info : Capturer l'image Windows

Sous Windows PE, attribuer la lettre **C** à la partition Windows et éventuellement la lettre **S** à la partition de boot à l'aide de la commande **diskpart** si ce n'est pas déjà le cas (changer les numéros de disques et partition si besoin) :

diskpart

list disk

select disk 0

list part

list volume

select part 1

assign letter S:

select part 3

assign letter C:

list part
list volume
exit

Facultatif : accélérer la capture de l'image en définissant le mode de gestion de l'alimentation

Performances élevées :

powercfg /s 8c5e7fda-e8bf-4a96-9a85-a6e23a8c635c

Capter éventuellement la partition de boot (en pratique ce n'est pas vraiment utile vu qu'il faudra très probablement la régénérer lors d'une restauration) avec :

dism /Capture-Image /ImageFile:efi.wim /CaptureDir:S:\ /Name:"efi"

Capter la partition de Windows :

md C:\Recycler\Scratch

dism /Capture-Image /ImageFile:win.wim /CaptureDir:C:\ /Name:"win"
/ScratchDir:C:\Recycler\Scratch

Le fichier **win.wim** contiendra alors une image compressée de la partition Windows (C:).

2.30 Info : Capturer l'image Ubuntu

Sous Ubuntu Live, s'assurer d'abord que la partition Ubuntu à capturer n'est pas montée (changer les numéros de disques et partition si besoin) :

sudo umount /dev/sda3

(ou utiliser **gparted** pour mieux voir quelle partition).

On peut éventuellement sauvegarder le MBR et VBR :

sudo dd if=/dev/sda of=mbr.bak bs=8192 count=1

sudo dd if=/dev/sda3 of=vbr.bak bs=8192 count=1

Pour sauvegarder une image compressée de la partition Ubuntu dans le fichier **rootfs.fsa**, utiliser **fsarchiver** :

sudo fsarchiver -j2 savefs rootfs.fsa /dev/sda3