

U6N2A1EP1 - Dual boot

UF6 - Instal·lació de Programari
MP01 - Muntatge i Manteniment d'Equips



INSTITUT
CENDRASSOS

Alumne 1: Pau Morales Alcantara

DNI 1: 41569347K

Alumne 2: Pau Olmo Ardila

DNI 2: 41654318F

1er SMXC

Professor: Genís de Tuero

Arxiu: SMX-MP01-UF6-Dual Boot-Morales_Pau-Olmo_Pau

Data: 28/04/2024

Índex

1. Fer els pen drives instal·lables Linux i windows:.....	pàg 3
1.1. Creació de taula de particions al pen drive mitjançant fdisk.....	pàg 3
1.2. Creació i configuració de les particions pels instal·ladors.....	pàg 3
1.3. Copiar fitxers dels instal·ladors.....	pàg 6
1.4. Instal·lació i configuració del GRUB.....	pàg 7
Webgrafia.....	pàg 27

1. Fer els pen drives instal·lables Linux i windows:

Jo faré que el pen drive (de 16 GiB) també funcioni en dual boot, instal·lant i configurant el GRUB (GRand Unified Bootloader) per a que ens permeti escollir entre l'instal·lador de l'Ubuntu i el de Windows, tant per BIOS (NTLDR per carregar el BOOTMGR) com per UEFI (chainloader per carregar el bootmgfw.efi).

1.1. Creació de taula de particions al pen drive mitjançant fdisk

Creem una taula de particions GPT mitjançant fdisk:

```
Pau Morales
└─$ sudo fdisk /dev/sde
[sudo] password for spz:

Welcome to fdisk (util-linux 2.39.3).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS (MBR) disklabel with disk identifier 0x54e46ca5.

Command (m for help): g
Created a new GPT disklabel (GUID: 70DCB381-B75E-4780-9445-9F1D03CA1209).

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

1.2. Creació i configuració de les particions pels instal·ladors

Ara, creem el següent esquema de particions:

- Partició 1 - **grub2 core.img** (1 MiB): Aquesta partició serà la partició per arrencar el grub en sistemes que arrenquin per BIOS, ja sigui pel firmware que tenen o perquè utilitzen el CSM (Compatibility Support Module). Aquesta partició es sol anomenar "BIOS_GRUB" o "BIOS Boot partition", i conté la imatge del bootloader del grub (grub.img), ja que els sistemes BIOS no suporten arrencar aplicacions EFI (des de ESPs), i necessiten un bootloader emmagatzemat a l'MBR, però com que utilitzem una taula de particions GPT, això no és possible. L'identificador GPT d'aquest tipus de particions és "**21686148-6449-6E6F-744E-656564454649**", que té un nom curt (*label*) de "**bios_grub**". Al ser una partició tan petita, no requereix de tenir sistema de fitxers, ja que la partició sencera serà el bootloader del grub.
- Partició 2 - **ESP - EFI System Partition** (100 MiB): Aquesta partició contindrà l'aplicació EFI (bootloader) per arrencar el GRUB en sistemes UEFI. Té un identificador GPT de "**C12A7328-F81F-11D2-BA4B-00A0C93EC93B**", o "**esp**".

Seguint les regles de les particions ESP, l'haurem de formatar amb un sistema de fitxers FAT32.

- Partició 3 - **BOOT** (100 MiB): Aquesta partició emmagatzemarà el directori `"/grub"`, junt als seus fitxers de configuració (**grub.cfg**) i mòduls (fitxers `.mod`). La formatarem com a ext4, ja que encara que suporti altres sistemes de fitxers gràcies a mòduls implementats a les versions més noves, el grub funciona de forma nativa amb ext4. Aquesta partició tindrà un GUID GPT de **"0FC63DAF-8483-4772-8E79-3D69D8477DE4"**, també anomenat "Linux filesystem".
- Partició 4 - **Instal·lador Ubuntu 22.04 LTS** (5 GiB - 5120 MiB): Aquesta partició formatada en ext4 contindrà tots els fitxers de la imatge ISO de l'Ubuntu 22.04 LTS. Utilitzarà el mateix identificador GPT de tipus de partició que la partició del grub (partició 3).
- Partició 5 - **Instal·lador Windows 10 x64 i x86 versió 22H2** (8 GiB - 8192 MiB): Aquesta partició formatada amb NTFS contindrà tots els fitxers de la imatge ISO del Windows 10. Tindrà un ID de tipus de partició GPT de **"EBD0A0A2-B9E5-443387C0-68B6B72699C7"**, que s'anomena **"Microsoft Basic Data Partition"** o **"msfdata"**.

```
Pau Morales
└─$ sudo fdisk /dev/sde

Welcome to fdisk (util-linux 2.39.3).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): n
Partition number (1-128, default 1):
First sector (2048-30719966, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-30719966, default 30717951): +1M

Created a new partition 1 of type 'Linux filesystem' and of size 1 MiB.

Command (m for help): t
Selected partition 1
Partition type or alias (type L to list all): 4
Changed type of partition 'Linux filesystem' to 'BIOS boot'.

Command (m for help): n
Partition number (2-128, default 2):
First sector (4096-30719966, default 4096):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (4096-30719966, default 30717951): +100M

Created a new partition 2 of type 'Linux filesystem' and of size 100 MiB.

Command (m for help): t
Partition number (1,2, default 2): 2
Partition type or alias (type L to list all): 1

Changed type of partition 'Linux filesystem' to 'EFI System'.

Command (m for help): n
Partition number (3-128, default 3):
First sector (208896-30719966, default 208896):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (208896-30719966, default 30717951): +100M

Created a new partition 3 of type 'Linux filesystem' and of size 100 MiB.

Command (m for help): n
Partition number (4-128, default 4):
First sector (413696-30719966, default 413696):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (413696-30719966, default 30717951): +5G

Created a new partition 4 of type 'Linux filesystem' and of size 5 GiB.

Command (m for help): n
Partition number (5-128, default 5):
First sector (10899456-30719966, default 10899456):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (10899456-30719966, default 30717951): +8G

Created a new partition 5 of type 'Linux filesystem' and of size 8 GiB.
```

```

First sector (10899456-30719966, default 10899456):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (10899456-30719966, default 30719966): +8G
Created a new partition 5 of type 'Linux filesystem' and of size 8 GiB.

Command (m for help): t

Partition number (1-5, default 5):

Partition type or alias (type L to list all): 11

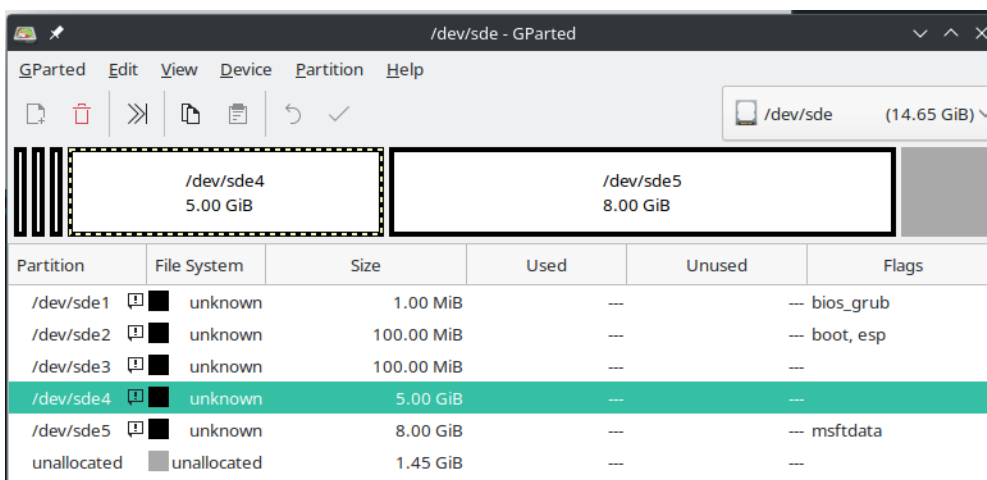
Changed type of partition 'Linux filesystem' to 'Microsoft basic data'.

Command (m for help): w

The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

```

Ara, amb el GParted veurem les següents particions al pendrive:



Com que tot està correcte, podem començar a formatar les particions:

```

sudo mkfs.fat -F 32 /dev/sde2 -n "ESP"
mkfs.fat 4.2 (2021-01-31)
sudo mkfs.ext4 -L "B00T" /dev/sde3
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 102400 1k blocks and 25584 inodes
Filesystem UUID: f784e95e-256f-4aa1-88de-8c833e823dcc
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

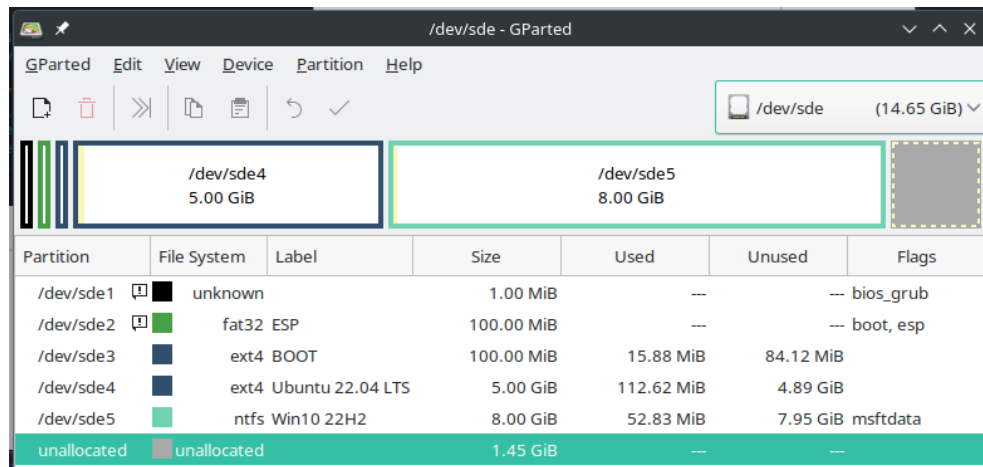
sudo mkfs.ext4 -L "Ubuntu 22.04 LTS" /dev/sde4
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 1310720 4k blocks and 327680 inodes
Filesystem UUID: efc1315a-69b4-4bfa-95c4-0d2bf0650e14
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

sudo mkfs.ntfs -L "Win10 22H2" /dev/sde5
Cluster size has been automatically set to 4096 bytes.
Initializing device with zeroes: 100% - Done.
Creating NTFS volume structures.
mkntfs completed successfully. Have a nice day.

```

I el GParted ens mostrarà el següent:



1.3. Copiar fitxers dels instal·ladors

Ara, muntarem cada partició i imatge ISO a un directori diferent sota /mnt, per tenir-ho tot a mà:

```
Pau Morales
└─$ sudo mkdir /mnt/ISO-UBUNTU /mnt/ISO-WINDOWS
└─$ sudo mkdir /mnt/ESP /mnt/BOOT
└─$ sudo mkdir /mnt/UBUNTU /mnt/WINDOWS
└─$ sudo mount -o ro,loop /run/media/spz/4_TB/Pau/VM/ISOs/Linux/Ubuntu/LTS/22.04/ubuntu-22.04.3-desktop-amd64.iso /mnt/ISO-UBUNTU
└─$ sudo mount -o ro,loop /run/media/spz/4_TB/Pau/VM/ISOs/Microsoft/Windows_10/Home/es-ES/Windows.iso /mnt/ISO-WINDOWS
└─$ sudo mount /dev/sde2 /mnt/ESP
└─$ sudo mount /dev/sde3 /mnt/BOOT
└─$ sudo mount /dev/sde4 /mnt/UBUNTU
└─$ sudo mount /dev/sde5 /mnt/WINDOWS
└─$ lsblk | grep /mnt
loop0  7:0  0  4,7G  1 loop /mnt/ISO-UBUNTU
loop1  7:1  0  7,8G  1 loop /mnt/ISO-WINDOWS
└-sde2  8:66  1  100M  0 part /mnt/ESP
└-sde3  8:67  1  100M  0 part /mnt/BOOT
└-sde4  8:68  1    5G  0 part /mnt/UBUNTU
└-sde5  8:69  1    8G  0 part /mnt/WINDOWS
```

Copiem tots els fitxers de la ISO d'Ubuntu a la partició corresponent, de forma recursiva (-r *recursive*) mantenint els permisos (-a *archive*), i mostrant informació del progrès (-v *verbose*):

```
└─$ sudo cp -rav /mnt/ISO-UBUNTU /mnt/UBUNTU
'/mnt/ISO-UBUNTU' -> '/mnt/UBUNTU/ISO-UBUNTU'
'/mnt/ISO-UBUNTU/boot.catalog' -> '/mnt/UBUNTU/ISO-UBUNTU/boot.catalog'
'/mnt/ISO-UBUNTU/md5sum.txt' -> '/mnt/UBUNTU/ISO-UBUNTU/md5sum.txt'
'/mnt/ISO-UBUNTU/ubuntu' -> '/mnt/UBUNTU/ISO-UBUNTU/ubuntu'
'/mnt/ISO-UBUNTU/.disk' -> '/mnt/UBUNTU/ISO-UBUNTU/.disk'
'/mnt/ISO-UBUNTU/.disk/base_installable' -> '/mnt/UBUNTU/ISO-UBUNTU/.disk/base_installable'
'/mnt/ISO-UBUNTU/.disk/casper-uuid-generic' -> '/mnt/UBUNTU/ISO-UBUNTU/.disk/casper-uuid-generic'
'/mnt/ISO-UBUNTU/.disk/cd_type' -> '/mnt/UBUNTU/ISO-UBUNTU/.disk/cd_type'
'/mnt/ISO-UBUNTU/.disk/info' -> '/mnt/UBUNTU/ISO-UBUNTU/.disk/info'
'/mnt/ISO-UBUNTU/.disk/release_notes_url' -> '/mnt/UBUNTU/ISO-UBUNTU/.disk/release_notes_url'
'/mnt/ISO-UBUNTU/boot' -> '/mnt/UBUNTU/ISO-UBUNTU/boot'
'/mnt/ISO-UBUNTU/boot/memtest86+.bin' -> '/mnt/UBUNTU/ISO-UBUNTU/boot/memtest86+.bin'
'/mnt/ISO-UBUNTU/boot/grub' -> '/mnt/UBUNTU/ISO-UBUNTU/boot/grub'
'/mnt/ISO-UBUNTU/boot/grub/grub.cfg' -> '/mnt/UBUNTU/ISO-UBUNTU/boot/grub/grub.cfg'
'/mnt/ISO-UBUNTU/boot/grub/loopback.cfg' -> '/mnt/UBUNTU/ISO-UBUNTU/boot/grub/loopback.cfg'
'/mnt/ISO-UBUNTU/boot/grub/fonts' -> '/mnt/UBUNTU/ISO-UBUNTU/boot/grub/fonts'
'/mnt/ISO-UBUNTU/boot/grub/fonts/unicode.pf2' -> '/mnt/UBUNTU/ISO-UBUNTU/boot/grub/fonts/unicode.pf2'
'/mnt/ISO-UBUNTU/boot/grub/i386-pc' -> '/mnt/UBUNTU/ISO-UBUNTU/boot/grub/i386-pc'
'/mnt/ISO-UBUNTU/boot/grub/i386-pc/915resolution.mod' -> '/mnt/UBUNTU/ISO-UBUNTU/boot/grub/i386-pc/915resolution'
```

No estic del tot segur per què s'ha creat un directori "ISO-UBUNTU" a la partició. Per tant, mouem tots els fitxers del directori /mnt/UBUNTU/ISO-UBUNTU al directori /mnt/UBUNTU:

```

sudo mv /mnt/UBUNTU/ISO-UBUNTU/* /mnt/UBUNTU
sudo mv /mnt/UBUNTU/ISO-UBUNTU/.*/ /mnt/UBUNTU
sudo rm -rf /mnt/UBUNTU/ISO-UBUNTU

```

Important no oblidar-nos dels fitxers i directoris ocults que pengin de l'arrel (.disk).

Ara, fem el mateix amb la ISO del Windows 10:

```

sudo cp -rav /mnt/ISO-WINDOWS/* /mnt/WINDOWS
'/mnt/ISO-WINDOWS/autorun.inf' -> '/mnt/WINDOWS/autorun.inf'
'/mnt/ISO-WINDOWS/boot' -> '/mnt/WINDOWS/boot'
'/mnt/ISO-WINDOWS/boot/es-es' -> '/mnt/WINDOWS/boot/es-es'
'/mnt/ISO-WINDOWS/boot/es-es/bootsect.exe.mui' -> '/mnt/WINDOWS/boot/es-es/bootsect.exe.mui'
'/mnt/ISO-WINDOWS/boot/fonts' -> '/mnt/WINDOWS/boot/fonts'
'/mnt/ISO-WINDOWS/boot/fonts/chs_boot.ttf' -> '/mnt/WINDOWS/boot/fonts/chs_boot.ttf'
'/mnt/ISO-WINDOWS/boot/fonts/cht_boot.ttf' -> '/mnt/WINDOWS/boot/fonts/cht_boot.ttf'
'/mnt/ISO-WINDOWS/boot/fonts/jpn_boot.ttf' -> '/mnt/WINDOWS/boot/fonts/jpn_boot.ttf'
'/mnt/ISO-WINDOWS/boot/fonts/kor_boot.ttf' -> '/mnt/WINDOWS/boot/fonts/kor_boot.ttf'
'/mnt/ISO-WINDOWS/boot/fonts/malgunn_boot.ttf' -> '/mnt/WINDOWS/boot/fonts/malgunn_boot.ttf'
'/mnt/ISO-WINDOWS/boot/fonts/malgun_boot.ttf' -> '/mnt/WINDOWS/boot/fonts/malgun_boot.ttf'
'/mnt/ISO-WINDOWS/boot/fonts/malgun_console.ttf' -> '/mnt/WINDOWS/boot/fonts/malgun_console.ttf'
'/mnt/ISO-WINDOWS/boot/fonts/meiryon_boot.ttf' -> '/mnt/WINDOWS/boot/fonts/meiryon_boot.ttf'

```

A aquesta ISO no hi ha cap fitxer que comenci amb un ".", ja que a Windows el tema de l'ocultació de fitxers funciona amb atributs NTFS del propi fitxer, i no té res a veure amb el nom. Per tant, no cal que englobem aquests fitxers.

Un cop acabi (a mí m'ha trigat 45 minuts, degut a que el pendrive és USB 2.0), ja podem començar a instal·lar el grub.

1.4. Instal·lació i configuració del GRUB

Com he mencionat a l'inici del document, farem instal·lacions del grub tant per sistemes BIOS com UEFI. Començarem per UEFI.

Utilitzaré la utilitat **grub-install**, que vé inclosa amb el paquet "**grub**" (obtenible mitjançant pacman, per distribucions basades en Arch). Per distribucions basades en Debian (APT), el nom del paquet és "**grub2-common**".

Ara, farem una instal·lació nova del grub (per EFI) a les particions del pen drive que havíem creat prèviament:

```

sudo grub-install --target=x86_64-efi --removable --no-floppy --boot-directory=/mnt/BOOT --efi-directory=/mnt/ESP --bootloader-id="MULTIBOOT-PMORALES"
Installing for x86_64-efi platform.
Installation finished. No error reported.

```

Aquesta configuració ha instal·lat el GRUB per sistemes EFI-64, preparat per arrencar des d'un pendrive, amb els fitxers de configuració a la partició BOOT, les aplicacions EFI per arrencar-lo a la partició ESP, i amb un nom de "MULTIBOOT-PMORALES".

Ara, observem què ens ha instal·lat la utilitat:

```
Pau Morales
├─ /mnt/ESP
│ └─ EFI
│     └─ BOOT
│         └─ BOOTX64.EFI
├─ 3 directories, 1 file
├─ /mnt/ESP/EFI/BOOT/BOOTX64.EFI
│   └─ /mnt/ESP/EFI/BOOT/BOOTX64.EFI: PE32+ executable (EFI application) x86-64 (stripped to external PDB), for MS Windows, 4 sections
├─ /mnt/BOOT
│ └─ grub
│     └─ locale
│         ├── ast.mo
│         ├── ca.mo
│         ├── da.mo
│         ├── de_CH.mo
│         ├── de@hebrew.mo
│         ├── de.mo
│         ├── en@arabic.mo
│         ├── en@cyrillic.mo
│         ├── en@greek.mo
│         ├── en@hebrew.mo
│         ├── en@iglatin.mo
│         ├── en@quot.mo
│         └─ en.mo
```

A la partició ESP ens ha creat un directori "EFI" (que és l'estàndard per les ESP), i a dins un directori "BOOT", que conté el bootloader (aplicació EFI) del grub. Aquest executable està signat amb les un certificat que correspon a les claus del Secure Boot que utilitza el grub, però no conté les claus com a tal.

A la partició BOOT s'ha creat un directori anomenat "**grub**", que conté directoris com ara "**fonts**", que conté les fonts que podem fer que el grub utilitzi, "**locale**", que conté fitxers .mo per diferents traduccions del grub. A més, conté un directori "**themes**", on podem instal·lar temes, i per últim, pot contindre els directoris "**x86_64-efi**", "**x86_32-efi**" i "**i386-pc**", que són directoris que contenen els mòduls del grub específics per a un firmware de pc específic. En el nostr cas, de moment només tenim el directori "**x86_64-efi**", ja que hem instal·lat el grub per sistemes UEFI de 64 bits (x64). Al directori hi ha un fitxer anomenat "**grubenv**", on podem configurar les variables d'entorn del grub (s'ha d'editar amb **grub-editenv**, ja que aquest fitxer no conté text clar):

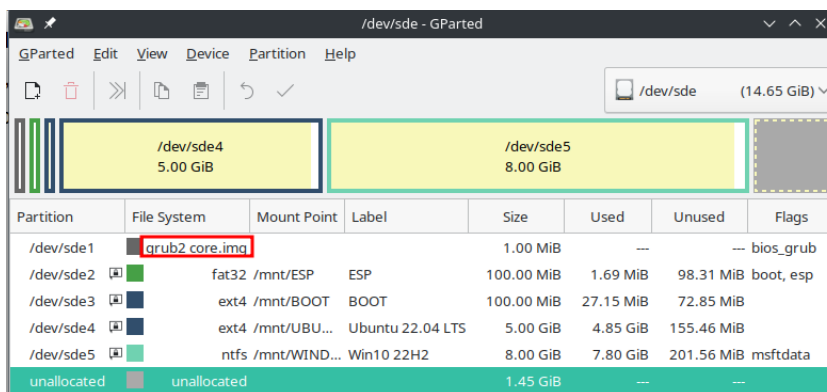
```
Pau Morales
├─ /mnt/BOOT/grub
│ └─ cat grubenv
│     └─ GRUB Environment Block
│         └─ WARNING: Do not edit this file by tools other than grub-editenv!!!
│             └─ [Large block of text representing grub environment variables]
```

I per últim, el directori **grub** pot contindre fitxers **.cfg**, que són els fitxers de configuració del grub, on podem especificar ordres que volem que segueixi el grub, com ara mostrar entrades d'arrencada (*boot entries*), agafar un tema, canviar el fons, la font... El nom de fitxer **.cfg** més comú (i l'estàndard) és "**grub.cfg**".

Ara, toca instal·lar el grub per a sistemes BIOS (i386-pc):

```
Pau Morales
└─$ sudo grub-install --target=i386-pc --removable --no-floppy --boot-directory=/mnt/BOOT /dev/sde
Installing for i386-pc platform.
Installation finished. No error reported.
└─$ ls /mnt/BOOT/grub
fonts  grubenv  i386-pc  locale  themes  x86_64-efi
```

L'últim paràmetre (/dev/sde) ha indicat a l'instal·lador on volem que estigui el bootloader del grub, o sigui el pen drive. Com que l'instal·lador ha detectat que aquest pendrive no té MBR funcional, ha buscat una partició que tingui l'etiqueta "BIOS Boot partition", i ha instal·lat el grub directament a aquella partició. Ho podem comprovar amb el gparted:

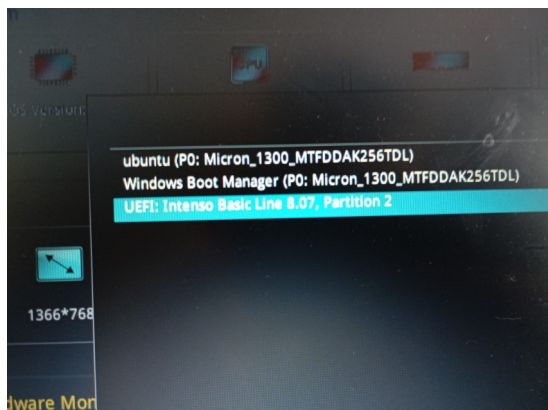


Un cop feta la instal·lació tant per BIOS com per UEFI, el pen drive ja serà bootable per sistemes que utilitzin BIOS i també pels que utilitzin UEFI com a firmware.

Ara, podem fer la prova d'intentar arrencar el pendrive a una màquina real (firmware UEFI). Per tant, desmuntarem el pendrive i les imatges ISO, ja que ja no les necessitem:

```
Pau Morales
└─$ sudo umount /mnt/*
```

Al no tenir cap fitxer **.cfg**, hauríem d'arrencar des d'una shell del grub:



Començarem per l'entrada de boot més senzilla de fer, la de l'Ubuntu:

Simplement haurem de carregar el fitxer **/boot/grub/grub.cfg** de la partició BOOT, mitjançant la comanda "configfile". També haurem de fer que el grub vagi a buscar el fitxer .cfg a la partició correcta, per tant, necessitarem l'UUID del sistema de fitxers de la partició "UBUNTU", que es pot obtenir amb **blkid**:

```
Pau Morales
└─$ /mnt/UBUNTU/boot/grub$ blkid
/dev/sda1: LABEL="BACKUP_SMX" BLOCK_SIZE="512" UUID="4CBE2307BE22E0E2" TYPE="ntfs" PARTUUID="f44ae470-01"
/dev/sdb1: LABEL_FATBOOT="ESP" LABEL="ESP" UUID="A0DC-283A" BLOCK_SIZE="512" TYPE="vfat" PARTLABEL="Basic data partition" PARTUUID="932b0b00-8b50-01da-c0c5-9549a845ed00"
/dev/sdb2: BLOCK_SIZE="512" UUID="01DAB850B31ABF10" TYPE="ntfs" PARTLABEL="Basic data partition" PARTUUID="7c150818-1c1d-4a15-b29a-be1fc070101b"
/dev/sdb3: BLOCK_SIZE="512" UUID="01DAB850B52ED200" TYPE="ntfs" PARTUUID="2a006872-b9bb-4ecb-874e-5d5d0db066d3"
/dev/sdb4: LABEL="ROOT" UUID="9913c397-a061-4727-bb22-aa3ffd5dfc34" BLOCK_SIZE="4096" TYPE="ext4" PARTUUID="8b439f31-771e-4d9d-87cc-0a309cff18f0"
/dev/sdc2: LABEL="4 TB" BLOCK_SIZE="512" UUID="01D884F0444F3A10" TYPE="ntfs" PARTLABEL="Basic data partition" PARTUUID="e008759b-58c9-11ee-8b00-806e6f6e6963"
/dev/sdd2: LABEL="15MXC Pau Morales" BLOCK_SIZE="512" UUID="01DA86DA7627C000" TYPE="ntfs" PARTLABEL="Basic data partition" PARTUUID="d393be6c-5ae0-45fa-b98b-b0b3671cb4e4"
/dev/sde2: LABEL_FATBOOT="ESP" LABEL="ESP" UUID="3534-60E4" BLOCK_SIZE="512" TYPE="vfat" PARTUUID="e9516b05-8ba5-4583-8696-54251ab851af"
/dev/sde3: LABEL="BOOT" UUID="f784e95e-256f-4aa1-88de-8c833e823dcc" BLOCK_SIZE="1024" TYPE="ext4" PARTUUID="b469f02f-501f-41b4-b626-f937e51d38a2"
/dev/sde4: LABEL="Ubuntu 22.04 LTS" UUID="efc1315a-69b4-4bfa-95c4-0d2bf0650e14" BLOCK_SIZE="4096" TYPE="ext4" PARTUUID="142e139a-c200-49c5-b4c4-8b406bfa027e"
/dev/sde5: LABEL="Win10 22H2" BLOCK_SIZE="512" UUID="685A68D65BB503FB" TYPE="ntfs" PARTUUID="a845dc75-4731-4d93-9e1e-b524ea41bea1"
└─$ /mnt/UBUNTU/boot/grub$
```

Ens interessan els següents IDs de partició:

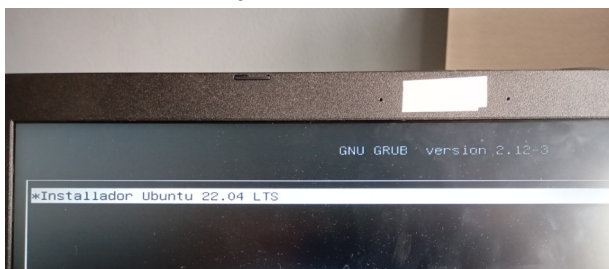
- BOOT: **f784e95e-256f-4aa1-88de-8c833e823dcc**
- UBUNTU: **efc1315a-69b4-4bfa-95c4-0d2bf0650e14**
- WINDOWS: **685A68D65BB503FB**

Ara, crearem la primera *boot entry*:

```
1 # Fitxer grub.cfg per la practica de dual boot SMX-MP01-UF6
2 # Pau Morales i Pau Olmo 15MXC
3 set timeout=-1
4 set default=0
5
6 menuentry "Instal·lador Ubuntu 22.04 LTS" {
7     search --no-floppy --fs-uuid --set=root efc1315a-69b4-4bfa-95c4-0d2bf0650e14
8     configfile /boot/grub/grub.cfg
9 }
```

He fet que la variable d'entorn "**timeout**" estigui posada a "-1" per a que no hi hagi temps d'espera fins que es carregui automàticament l'entrada seleccionada. He posat la variable "**default**" a "0" per a que l'entrada seleccionada per defecte sigui la primera. Després, he afegit una *menu entry* amb nom "**Instal·lador Ubuntu 22.04 LTS**", que si es carrega cerca un sistema de fitxers amb l'UUID de la partició UBUNTU, i el posa com a root, per després carregar el fitxer **grub.cfg** de la partició UBUNTU (instal·lador).

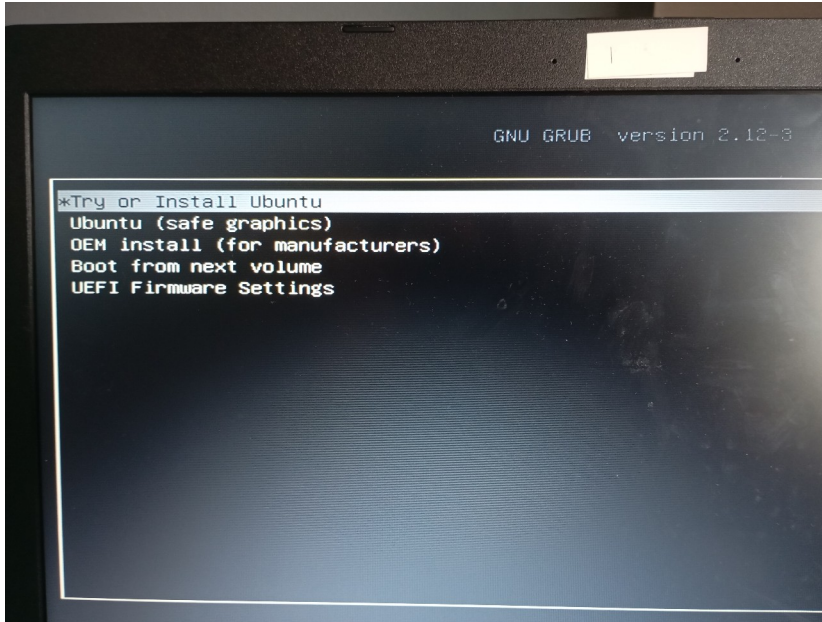
I la provem, bootejant el pendrive a la mateixa màquina d'abans:



Com podem veure, la menu entry ha estat detectada correctament. Això vol dir que el grub ha carregat el fitxer **grub.cfg** de forma automàtica.

El símbol "." no ha estat detectat correctament, segurament degut a que el grub no el suporta. El treurem la següent vegada que modifiquem el fitxer **grub.cfg**.

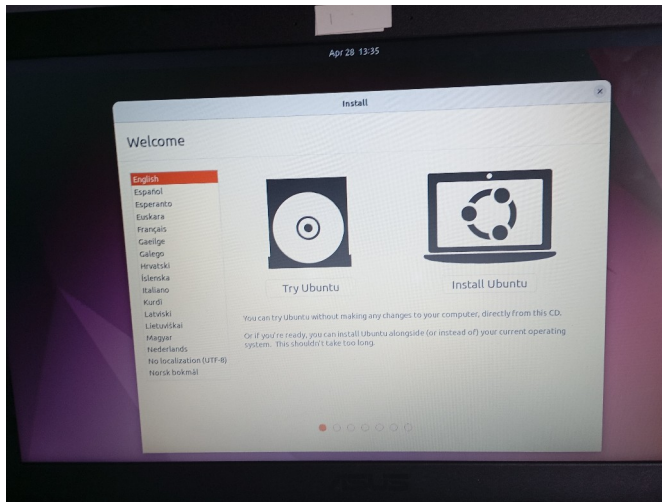
Si carreguem l'entrada que hem configurat, obtindrem el següent:



Que ja és el que volíem, ja que hem configurat que carregui el fitxer de configuració de l'instal·lador. Ara carregarem l'opció "Try or Install Ubuntu":



Sembla que està carregant...



Després d'esperar una estona, ja tindrem l'instal·lador de l'Ubuntu carregat. Fa un temps ho vaig intentar fer muntant les imatges ISO com a dispositius loop (mitjançant *loopback*), com es menciona a [aquesta web](#), però em va portar molts problemes, sobretot a l'instal·lador de Windows. Segurament degut a que no suporta ser arrencat si el muntem com a imatge ISO des del grub.

Ara, configurem la *boot entry* per al Windows. En aquest cas, haurem de tenir en compte si el sistema té firmware BIOS o UEFI, ja que el grub arrencat per BIOS no suporta carregar aplicacions EFI, i el grub arrencat per UEFI no suporta l'NTLDR (New Technology LoaDeR), que és el bootloader que utilitza el Windows per sistemes BIOS. Podem comprovar quin és el firmware de la màquina llegint la variable d'entorn del grub "\$grub_platform", que retorna "efi" si el sistema és UEFI, i retorna "pc" si el sistema és BIOS:

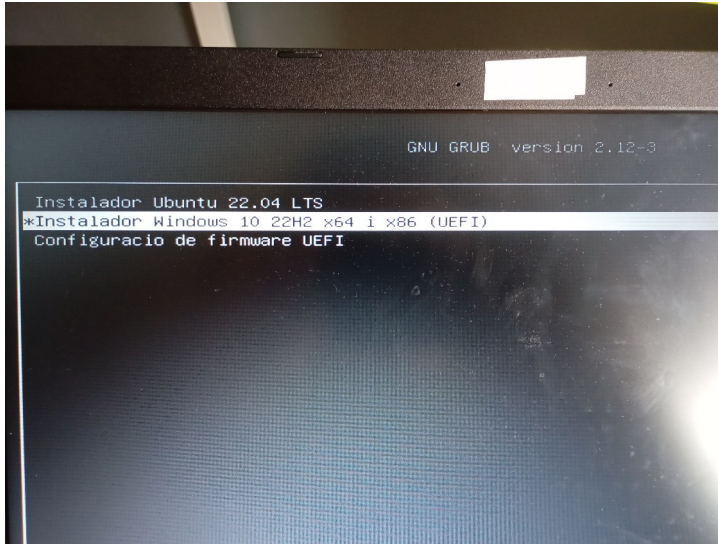
```

1 # Fitxer grub.cfg per la practica de dual boot SMX-MP01-UF6
2 # Pau Morales i Pau Olmo ISMXC
3 set timeout=-1
4 set default=0
5
6 menuentry "Instalador Ubuntu 22.04 LTS" {
7     search --no-floppy --fs-uuid --set=root efc1315a-69b4-4bfa-95c4-0d2bf0650e14
8     configfile /boot/grub/grub.cfg
9 }
10
11 # Windows i configuracio UEFI
12 if [ "$grub_platform" = "efi" ]; then
13     menuentry "Instalador Windows 10 22H2 x64 i x86 (UEFI)" {
14         insmod ntfs
15         insmod ntfscomp
16         insmod chain
17         search --no-floppy --fs-uuid --set=root 685A68D65BB503FB
18         chainloader /efi/boot/bootx64.efi
19         boot
20     }
21     menuentry "Configuracio de firmware UEFI" {
22         fwsetup
23     }
24 # Windows BIOS
25 elif [ "$grub_platform" = "pc" ]; then
26     menuentry "Instalador Windows 10 22H2 x64 i x86 (BIOS)" {
27         insmod ntfs
28         insmod ntfscomp
29         insmod chain
30         search --no-floppy --fs-uuid --set=root 685A68D65BB503FB
31         ntlldr /bootmgr
32         boot
33     }
34 fi

```

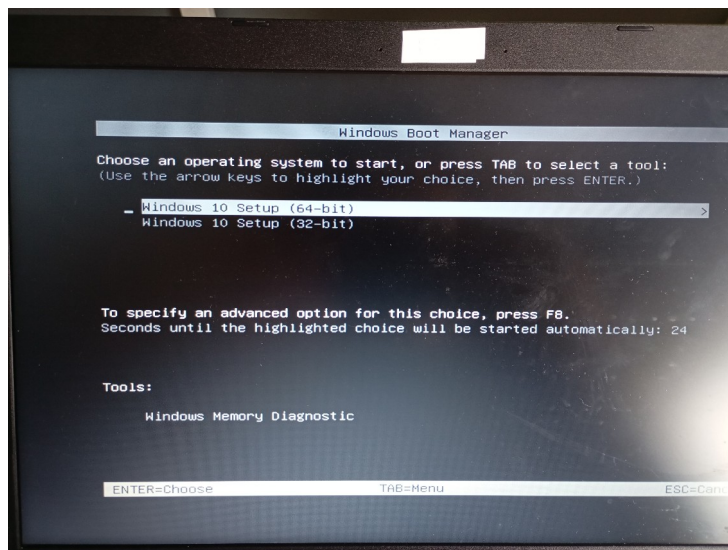
He afegit entrades específiques per sistemes UEFI i BIOS. Si el sistema és UEFI, es mostra una entrada per carregar l'instal·lador del Windows mitjançant el mòdul **chain.mod**, i es mostra una entrada per accedir a la configuració de la UEFI. En canvi, si el sistema és BIOS, es mostra una entrada per carregar l'instal·lador de Windows mitjançant NTLDR.

Ara, provem que tot funcioni correctament:



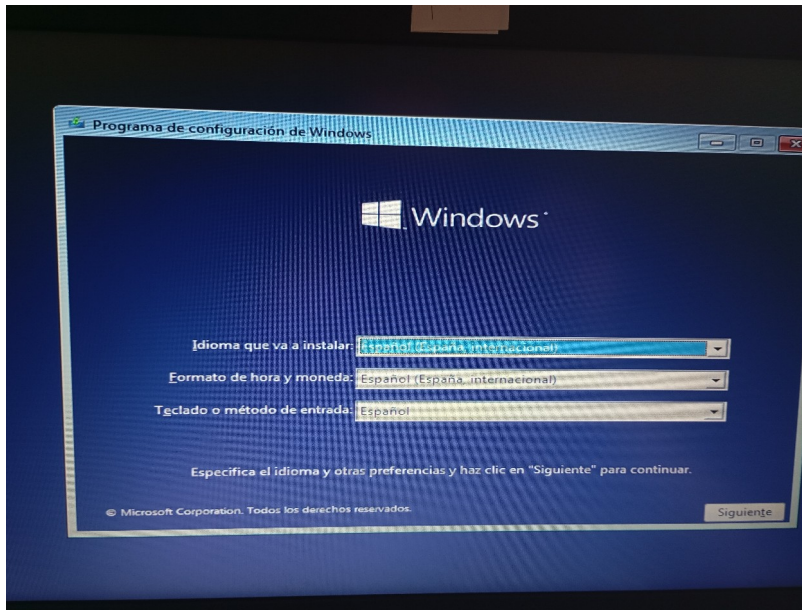
Les entrades de menú han estat carregades correctament, i no s'ha mostrat l'entrada per sistemes BIOS. Després comprovarem que en un sistema BIOS no apareguin les entrades UEFI.

Carreguem l'entrada de l'instal·lador de Windows:



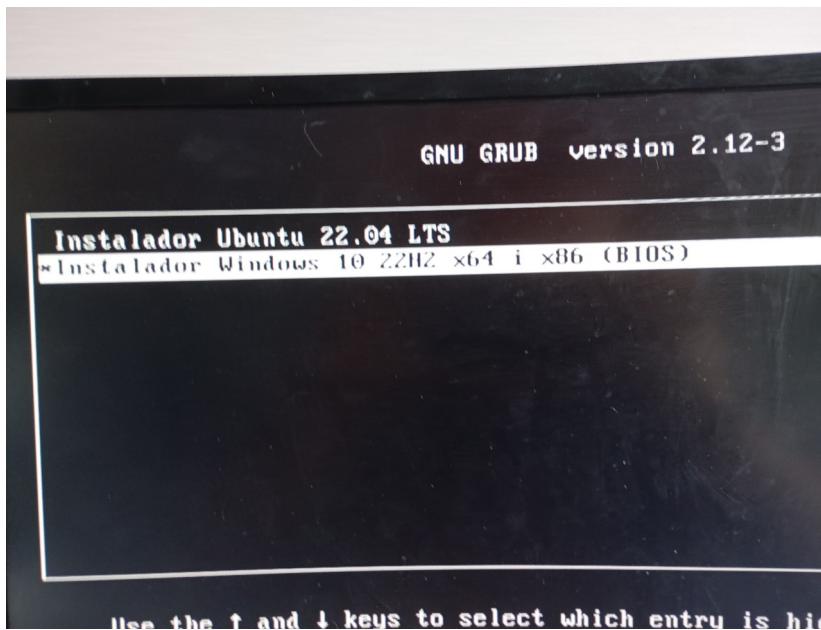
I en el meu cas, apareixerà un menú preguntant si volem arrencar l'instal·lador de 64 bits o el de 32 bits. Això és degut a que la imatge ISO era tant per x64 com per x86.

Escollim l'instal·lador de 64 bits:



I ens carregarà l'instal·lador del Windows correctament.

Ara, provem amb una màquina BIOS:



I veurem que el condicional ha funcionat perfectament; hem obtingut només les entrades d'arrencada compatibles amb BIOS. Encara que al intentar carregar l'instal·lador de Windows per BIOS, en lloc d'arrencar-se, s'ha reiniciat la màquina.

Aquest problema m'ha donat hores de pensar i provar, fins i tot he aconseguit una pantalla molt estranya, intentant provar si no estava carregant el fitxer que havia de carregar amb l'NTLDR:



Després d'una estona d'investigació, hem descobert que l'NTLDR no és compatible amb discs GPT, i per tant, haurem de carregar l'instal·lador d'una altra manera pels sistemes BIOS.

Segons [aquesta](#) pàgina web, es pot carregar un fitxer WIM des del grub, mitjançant una utilitat anomenada WIMBOOT. Un fitxer WIM (Windows IMaging format) és un fitxer que es sol veure a instal·ladors de Windows i conté tots els components del sistema operatiu (Microsoft Edge, Windows Media Player...). També són molt comuns a les versions PE (Preinstallation Environment) de Windows, com ara Gandalf, Sergei, Hiren's Boot, o DLC Boot. El format WIM ha estat reemplaçat per l'ESD (microsoft Electronic Software Download), però la ISO que utilitzem nosaltres té els dos formats (`/x64/sources/boot.wim` i `/x64/sources/install.esd`).

Aquesta utilitat (WIMBOOT) ens permetrà arrencar l'instal·lador des d'un dels fitxers WIM (x64 i x86), com fa la utilitat del DLC Boot que podem trobar com "LOAD ISO IMG IMA" al menú d'arrencada (grub).

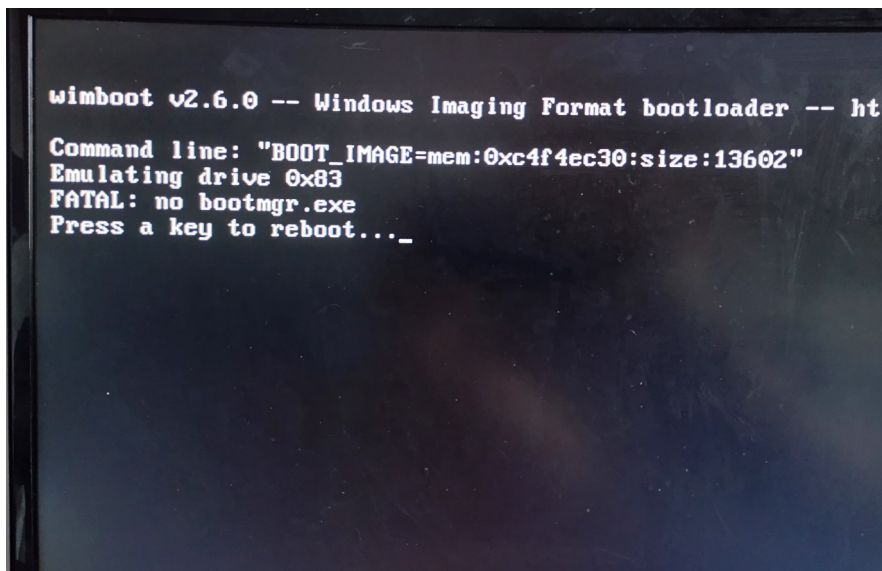
Hem estat investigant sobre la utilitat WIMBOOT i hem arribat a la conclusió de que ens pot ser molt útil, però nosaltres volem fer que s'arrenqui l'instal·lador des del grub, escollint una entrada, i no haguent d'escriure comandes per arrencar l'instal·lador mitjançant WIMBOOT cada cop que volguem instal·lar Windows a una màquina BIOS.

Ara, arranquem des del grub que hem configurat i a la shell integrada intentem carregar el fitxer WIM:



```
GNU GRUB v2.11-20230315-d1942e  
Minimal BASH-like line editing is supported. For the top level shell,  
lists possible command completions. Anywhere else TAB  
device or file completions. ESC at any time exits.  
  
grub> set root=(hd0,gpt5)  
grub> wimboot @:boot.wim:/x64/sources/boot.wim _
```

I obtindrem el següent error:



```
wimboot v2.6.0 -- Windows Imaging Format bootloader -- ht  
Command line: "BOOT_IMAGE=mem:0xc4f4ec30:size:13602"  
Emulating drive 0x83  
FATAL: no bootmgr.exe  
Press a key to reboot..._
```

Això ha estat degut a que el wimboot necessita d'un fitxer anomenat "bootmgr.exe", el qual no disposem.

El "grub2-filemanager" sí que el conté, i està configurat per carregar fitxers WIM, per tant, ens el descarregarem, configurarem i compilarem per fer-lo funcionar amb el nostre grub:

```

~/Desktop git clone https://github.com/a1ive/grub2-filemanager.git &>/dev/null
~/Desktop cd grub2-filemanager
~/Desktop/grub2-filemanager master ./update_grub2.sh &>/dev/null
~/Desktop/grub2-filemanager master nvim release.sh

```

Ens clonem el repositori [grub2-filemanager](https://github.com/a1ive/grub2-filemanager), i amb l'script "update_grub2.sh" ens descarreguem la última versió del repositori que hem mencionat abans, que és la versió modificada del grub amb els mòduls per carregar fitxers WIM.

Ara, amb un editor de text, modifiquem l'script per a que només faci el paquet per la versió anglesa del grub (en_US):

```

1  #!/usr/bin/env sh
2  if [ -e "grubfm*.7z" ]
3  then
4      rm grubfm*.7z
5  fi
6
7  i=0
8  lang="en_US"
9  if [ -d "releases" ]
10 then
11     rm -r releases
12 fi
13 mkdir releases
14 i=`expr $i + 1`
15 echo "${i}" | ./build.sh
16 cp grubfm.iso releases/
17 cp grubfm*.efi releases/
18 cp loadfm releases/
19 cd releases
20 7z a ../grubfm-${lang}.7z *
21 cd ..
22 rm -r releases

```

I ara editem el fitxer boot/grubfm/init.sh, que és un fitxer escrit en el llenguatge del grub (molt similar a bash) que s'executa res més accedir al file manager. El modificarem per a que carregui el fitxer WIM automàticament, per la versió x64 (64 bits):

```

~/Desktop/grub2-filemanager master !1 nvim boot/grubfm/init.sh

```

```

115 else
116     grubfm;
117 fi;
118
119 search --fs-uuid --no-floppy --set=root 055728AE558D6015
120 wimboot @:boot.wim:/x64/sources/boot.wim

```

Ara, compilem aquest grub amb l'script build.sh, que ens generarà un fitxer grubfm.iso, que l'utilitzarem com a initrd, carregant el fitxer loadfm (load file manager) com a kernel:

```

~/Desktop/grub2-filemanager master !2 ./build.sh
checking for gettext ... ok
checking for xorriso ... ok
checking for mtools ... ok
common files
Language
1. Simplified Chinese
2. Traditional Chinese
3. English (United States)
4. Turkish
5. German
6. Vietnamese
7. Russian
8. Hebrew
9. Spanish
10. Polish
11. Ukrainian
12. French
13. Danish
14. Portuguese (Brazil)
15. Arabic
16. Korean
17. Hungarian
Please make a choice: 3
en_US
x86_64-efi
copying afsplitter.mod
copying crypto.mod
copying cryptodisk.mod
copying geli.mod
copying json.mod
copying luks.mod
copying luks2.mod
copying pbkdf2.mod
10472 blocks
i386-efi

```

Copiem el fitxer loadfm i el grubfm.iso al pendrive, com a grubfm-win10_x64.iso:

```

~/Desktop/grub2-filemanager master !2 sudo cp loadfm /mnt/B00T
~/Desktop/grub2-filemanager master !2 sudo cp grubfm.iso /mnt/B00T/grubfm-win10_x64.iso
~/Desktop/grub2-filemanager master !2

```

Ara, compilem el fitxer grubfm.iso per la versió x86 (32 bits) de l'instal·lador:

```

114 configfile ${prefix}/netboot.sh;
115 else
116 grubfm;
117 fi;
118
119 search --fs-uuid --no-floppy --set=root 055728AE558D6015
120 wimboot @:boot.wim:/x86/sources/boot.wi

```

```

~/Desktop/grub2-filemanager master !2 echo "3" | ./build.sh &>/dev/null
~/Desktop/grub2-filemanager master !2 sudo cp grubfm.iso /mnt/B00T/grubfm-win10_x86.iso
~/Desktop/grub2-filemanager master !2

```

També compilarem una versió del grubfm sense modificar, per si ens interessa utilitzar-lo:

```

~/Desktop/grub2-filemanager master !2 nvim boot/grubfm/init.sh
~/Desktop/grub2-filemanager master !1 echo "3" | ./build.sh &>/dev/null
~/Desktop/grub2-filemanager master !1 sudo cp grubfm.iso /mnt/B00T/grubfm.iso
~/Desktop/grub2-filemanager master !1

```

I ara tornem a instal·lar els mòduls del grub que teniem al principi, ja que ja no necessitem la versió modificada:

```

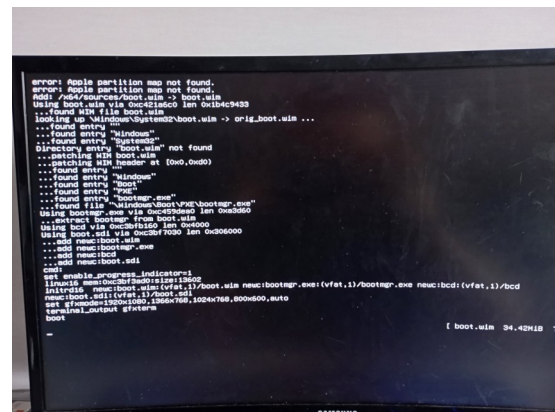
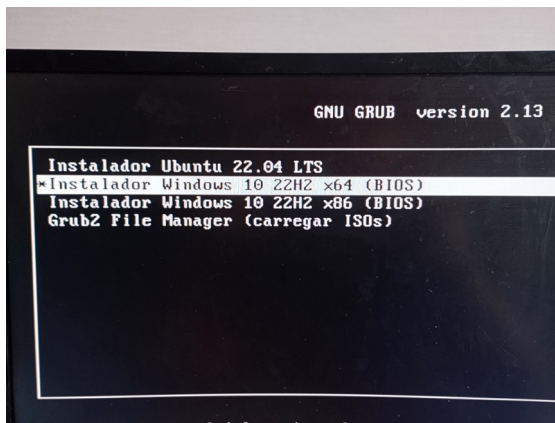
# ~
# /usr/lib/grub/
# /usr/lib/grub sudo rm -rf i386-pc
# /usr/lib/grub sudo tar -zxf i386-pc.gz
# /usr/lib/grub sudo grub-install --target=i386-pc --removable --no-floppy --force --boot-directory=/mnt/B00T /dev/sde
Installing for i386-pc platform.
Installation finished. No error reported.
# /usr/lib/grub
    
```

I editem el fitxer grub.cfg:

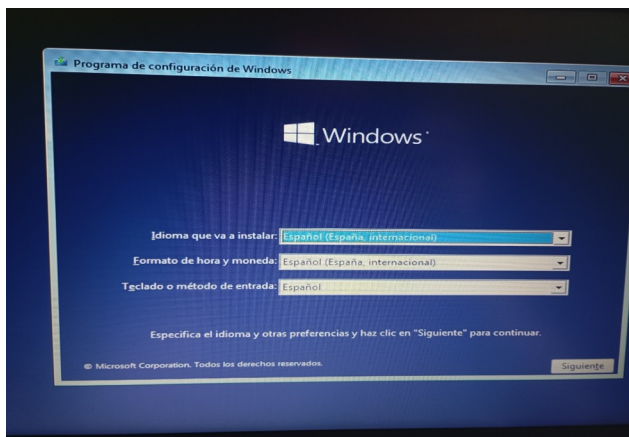
```

24
25 # Windows BIOS
26 elif [ "$grub_platform" = "pc" ]; then
27     menuentry "Instalador Windows 10 22H2 x64 (BIOS)" {
28         linux /loadfm
29         initrd /grubfm-win10_x64.iso
30     }
31     menuentry "Instalador Windows 10 22H2 x86 (BIOS)" {
32         linux /loadfm
33         initrd /grubfm-win10_x86.iso
34     }
35 fi
36
37 # Grub2 File Manager
38 menuentry "Grub2 File Manager (carregar ISOs)" {
39     linux /loadfm
40     initrd /grubfm.iso
41 }
    
```

Ara, provem que tot funcioni, arrencant des del pendrive:



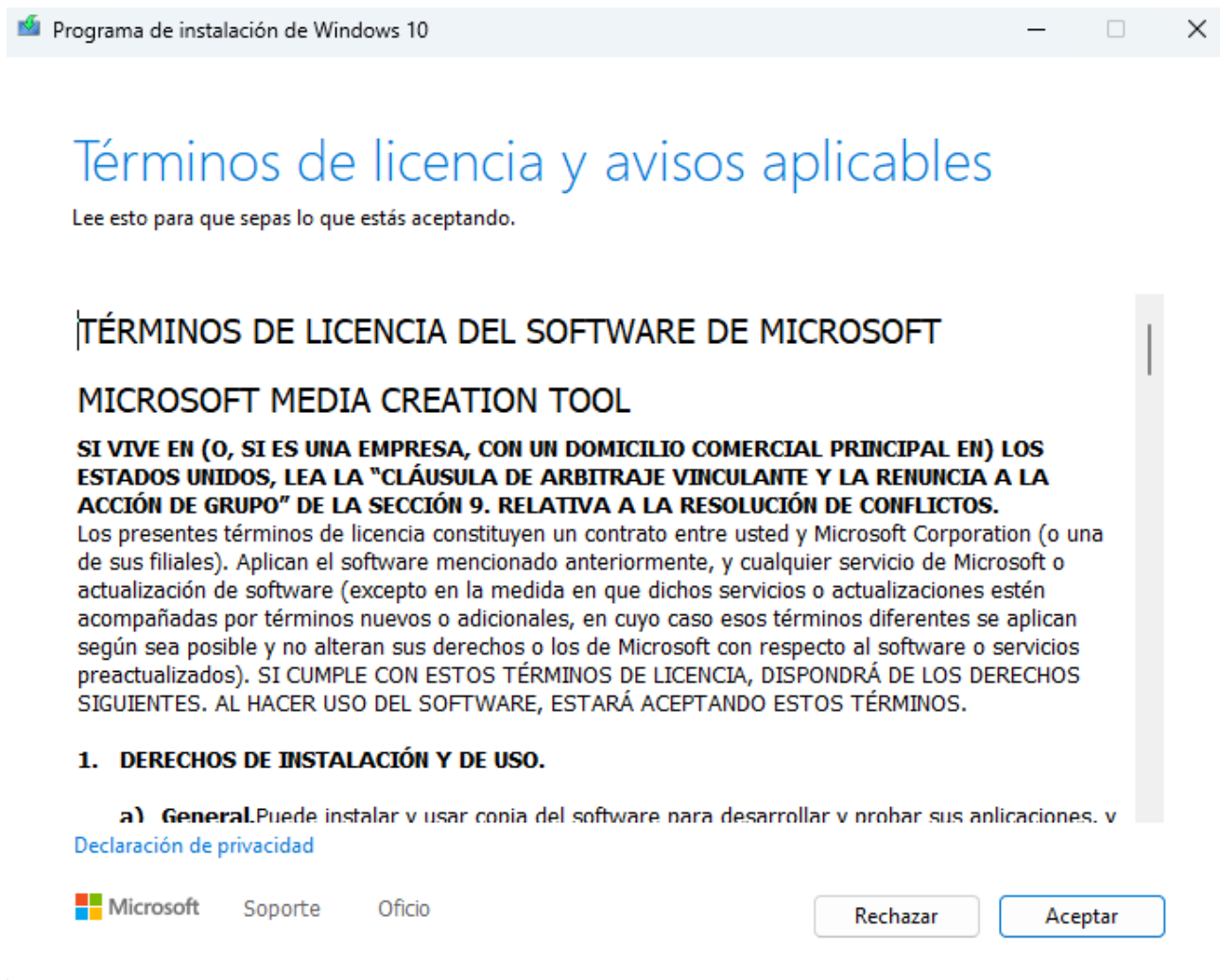
I veurem que tot funciona, fins i tot ens entrarà a l'instal·lador:



En el cas de que volguem instal·lar de la manera que demana l'exercici seria la següent:

Per fer els pen drives instal·lables el que haurem de fer primerament es descarregar les eines necessàries i en el meu cas el que he fet ha sigut instal·lar rufus i el MediaCreationTool.exe per Windows.

En el cas de Windows per fer que sigui un instal·lable el que haurem de fer es executar des de un ordinador Windows 7 o més l'executable .exe MediaCreationTool. En entrar al executable en sortira una pantalla d'acceptar els termes, simplement fem clic a acceptar.



Programa de instalación de Windows 10

Términos de licencia y avisos aplicables

Lee esto para que sepas lo que estás aceptando.

TÉRMINOS DE LICENCIA DEL SOFTWARE DE MICROSOFT

MICROSOFT MEDIA CREATION TOOL

SI VIVE EN (O, SI ES UNA EMPRESA, CON UN DOMICILIO COMERCIAL PRINCIPAL EN) LOS ESTADOS UNIDOS, LEA LA "CLÁUSULA DE ARBITRAJE VINCULANTE Y LA RENUNCIA A LA ACCIÓN DE GRUPO" DE LA SECCIÓN 9. RELATIVA A LA RESOLUCIÓN DE CONFLICTOS.

Los presentes términos de licencia constituyen un contrato entre usted y Microsoft Corporation (o una de sus filiales). Aplican el software mencionado anteriormente, y cualquier servicio de Microsoft o actualización de software (excepto en la medida en que dichos servicios o actualizaciones estén acompañadas por términos nuevos o adicionales, en cuyo caso esos términos diferentes se aplican según sea posible y no alteran sus derechos o los de Microsoft con respecto al software o servicios preactualizados). SI CUMPLE CON ESTOS TÉRMINOS DE LICENCIA, DISPONDRÁ DE LOS DERECHOS SIGUIENTES. AL HACER USO DEL SOFTWARE, ESTARÁ ACEPTANDO ESTOS TÉRMINOS.

1. DERECHOS DE INSTALACIÓN Y DE USO.

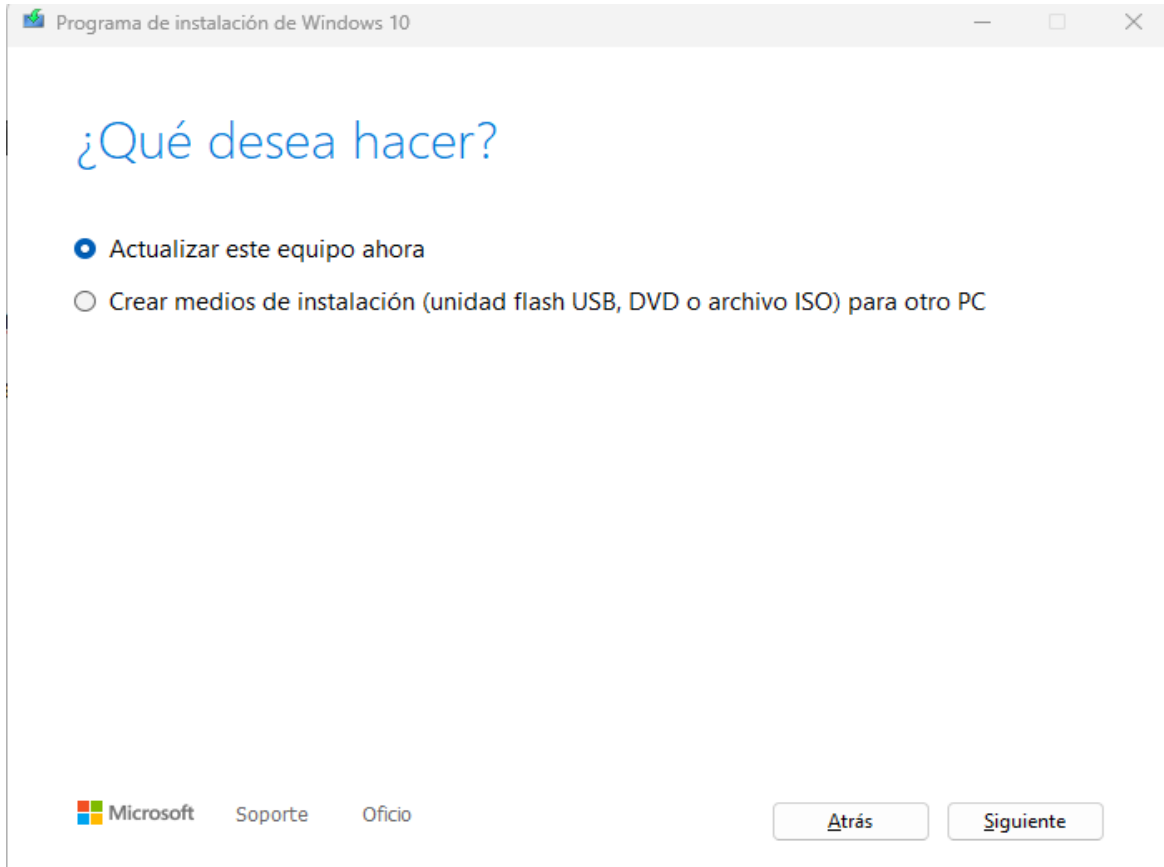
a) General. Puede instalar y usar copia del software para desarrollar y probar sus aplicaciones. y [Declaración de privacidad](#)

Microsoft Soporte Oficio

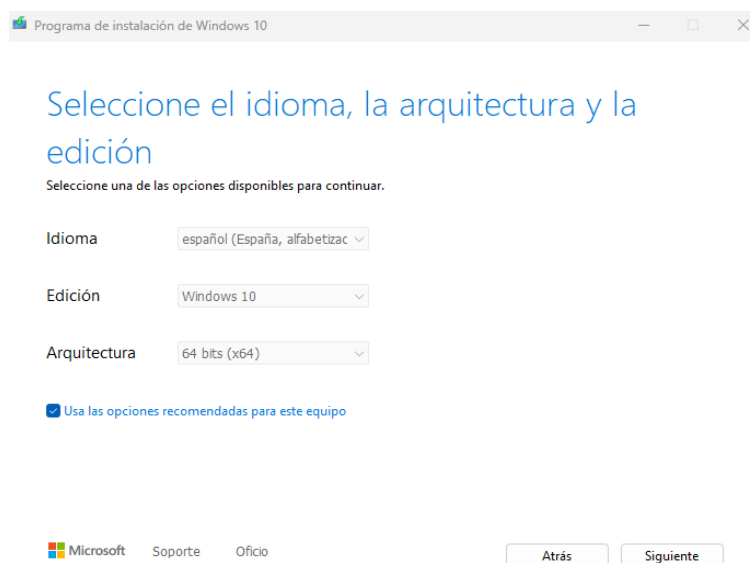
Rechazar Aceptar

Seguidament un cop hagui finalitzat la pantalla de carrega el que haurem de fer es seleccionar l'opció crear medis d'instal·lació.

En la següent pantalla fem clic a següent

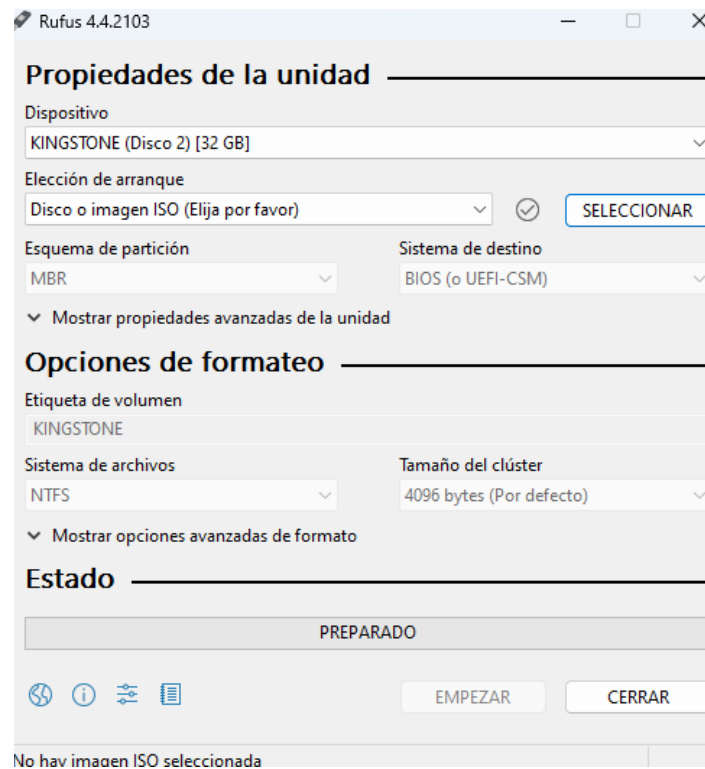


I finalment el que farem es seleccionar la primera opció ja que ho farà directament al pen drive, en el cas de que vulguem fer-ho amb l'aplicatiu rufus seleccionarem la opció iso



Rufus:

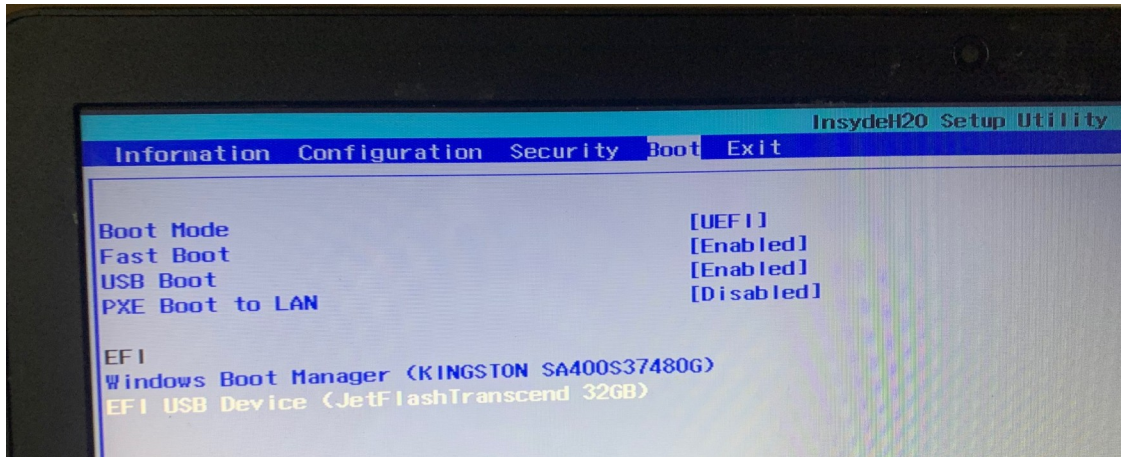
Per instal·lar Linux a un pen Drive en aquest cas faré servir rufus. Un cop posat el pen drive en sortirà una opció de seleccionar iso i simplement seleccionarem la iso de Linux que vulguem prèviament ja descarregada



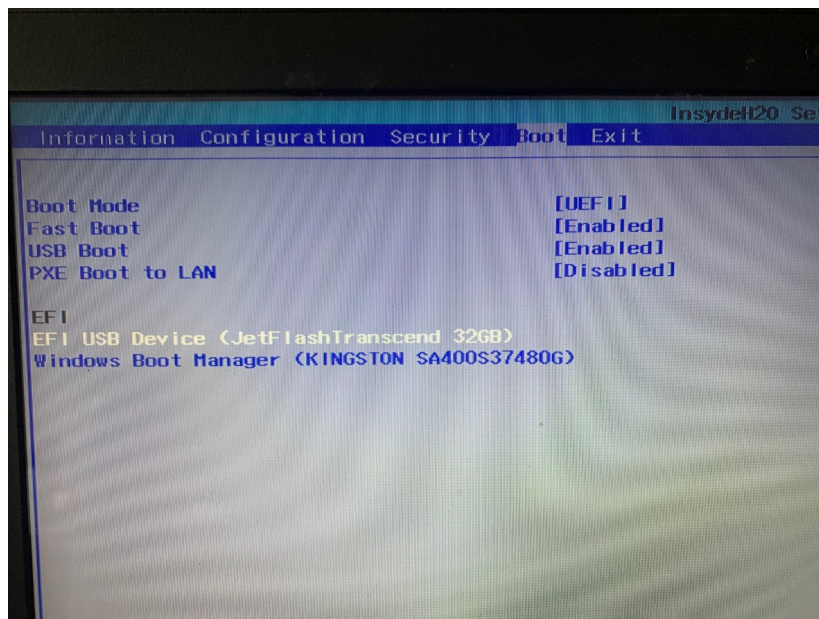
Ens sortiran les opcions tal que així, simplement fem clic a començar i esperarem.

1.5. Instal·lar Windows a l'ordinador de pràctiques

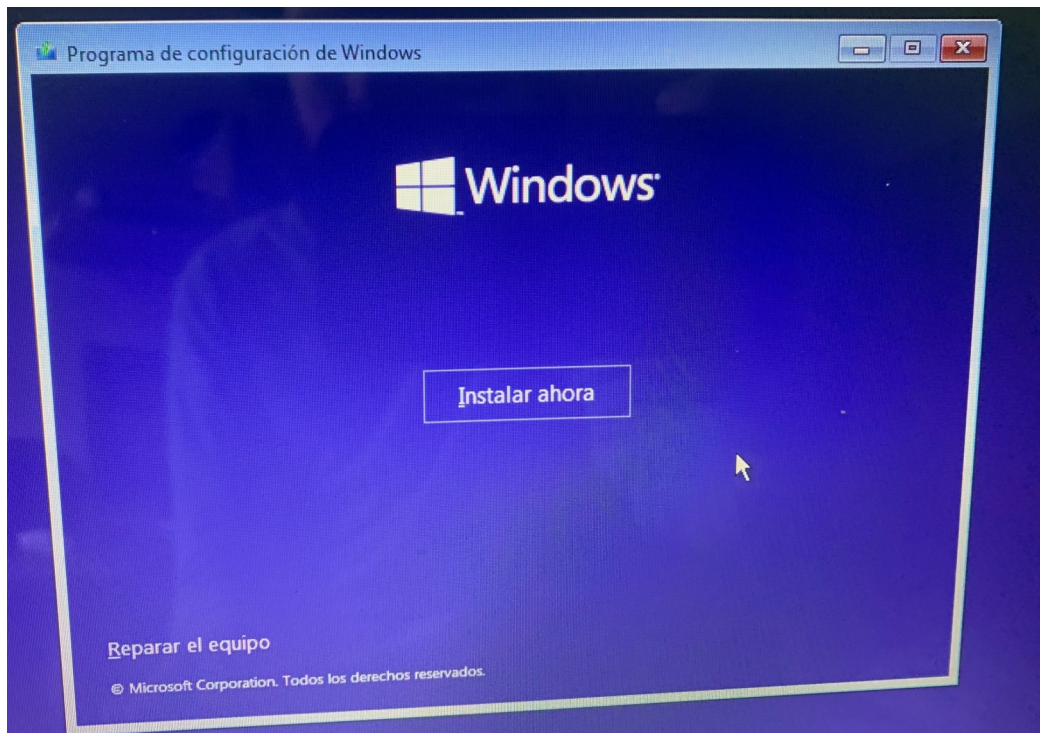
Per instal·lar Windows el que haurem de fer simplement es agafar el pen Drive insertar-lo al USB del ordinador en aquest cas un portàtil personal, i seguidament a les opcions de UEFI seleccionar el pen drive com a opció principal de boot:



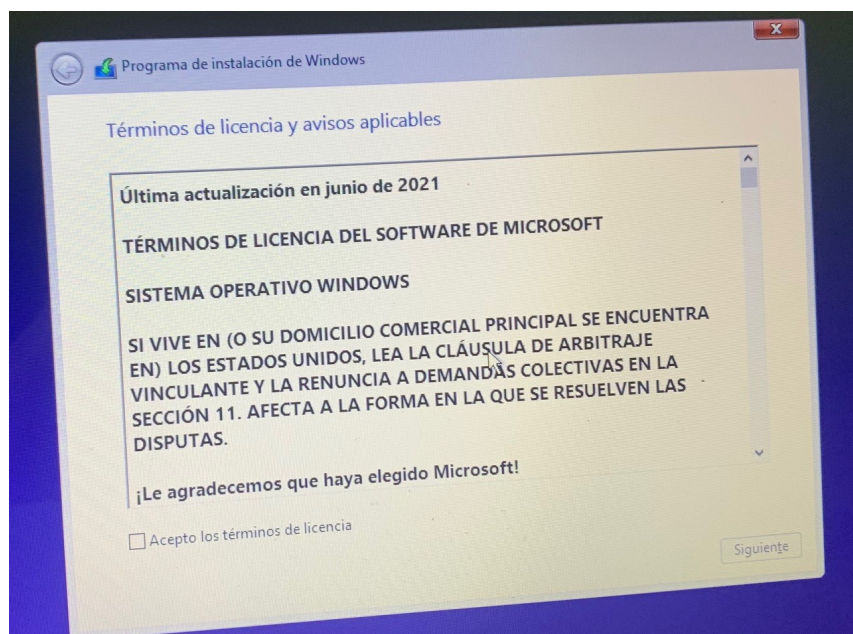
Seguidament passem la opció de l'USB a primera línia amb el símbol de més. Finalment guardem la configuració fent clic a save and exit.



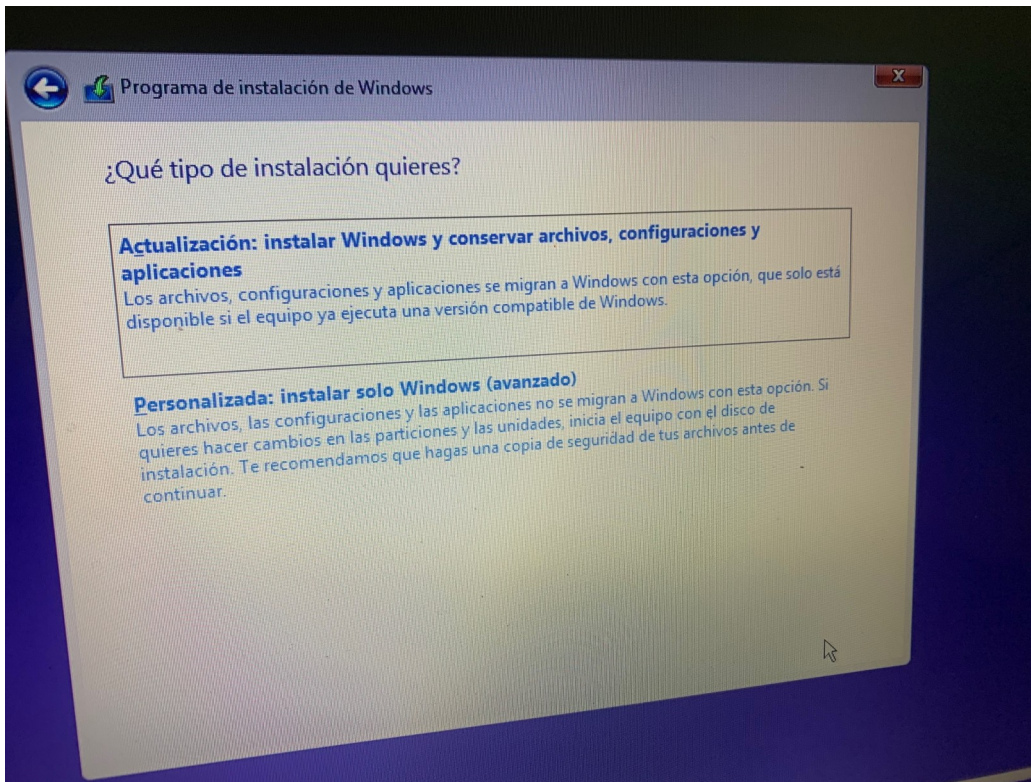
Per començar l'instal·lador, fem clic a "instalar ahora":



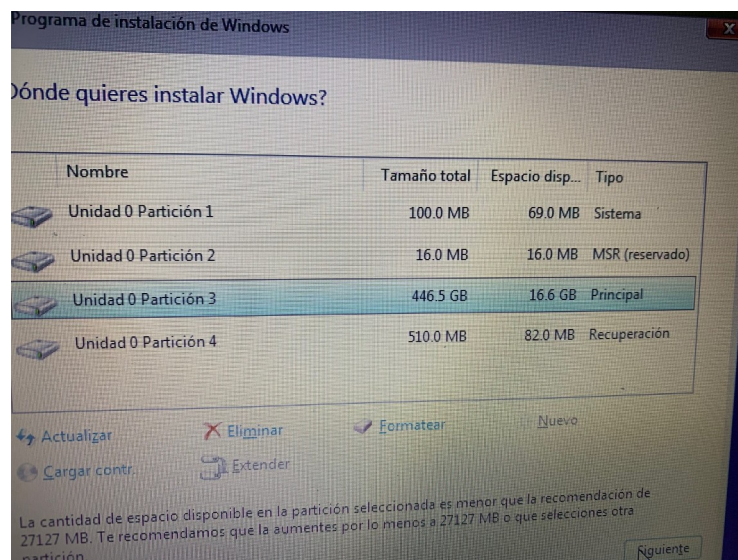
Marquem la casella "Acepto los terminos de llicencia" i seguidament fem clic a següent



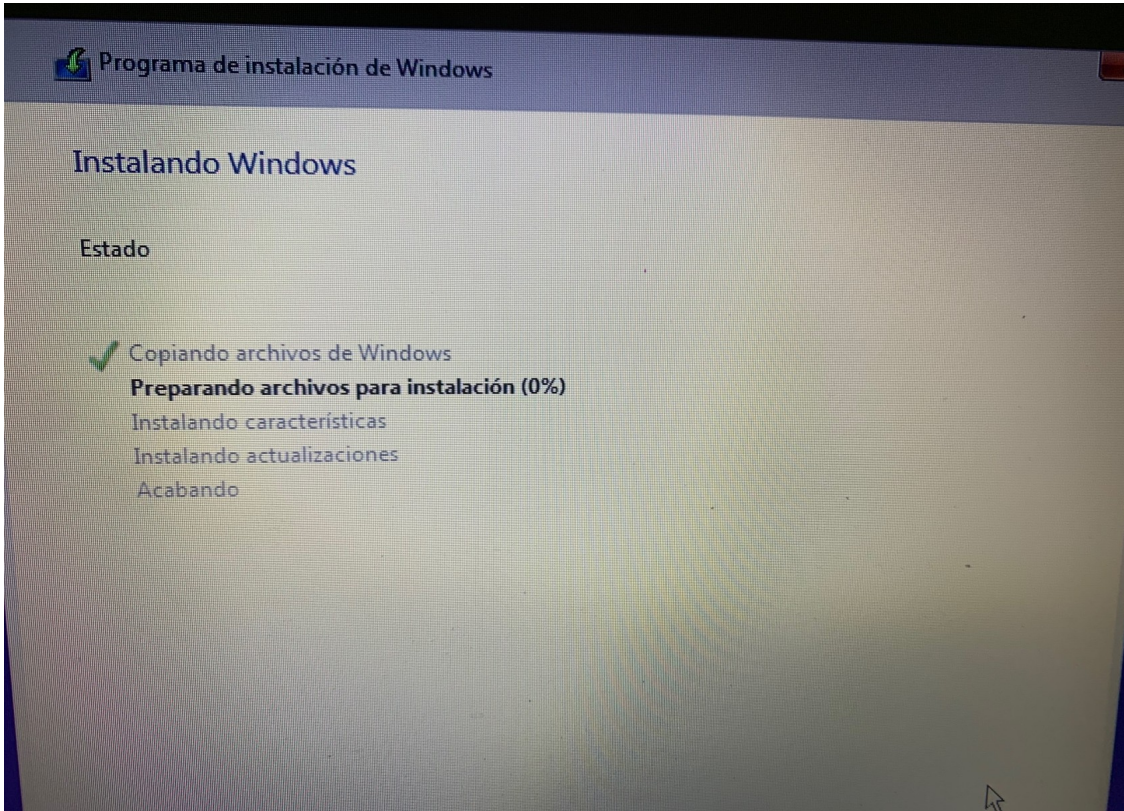
Seleccionem Personalitzada per seleccionar la partició i el disc en el que vulguem instal·lar Windows. El disc seleccionat el contingut serà eliminat totalment amb una formatació.



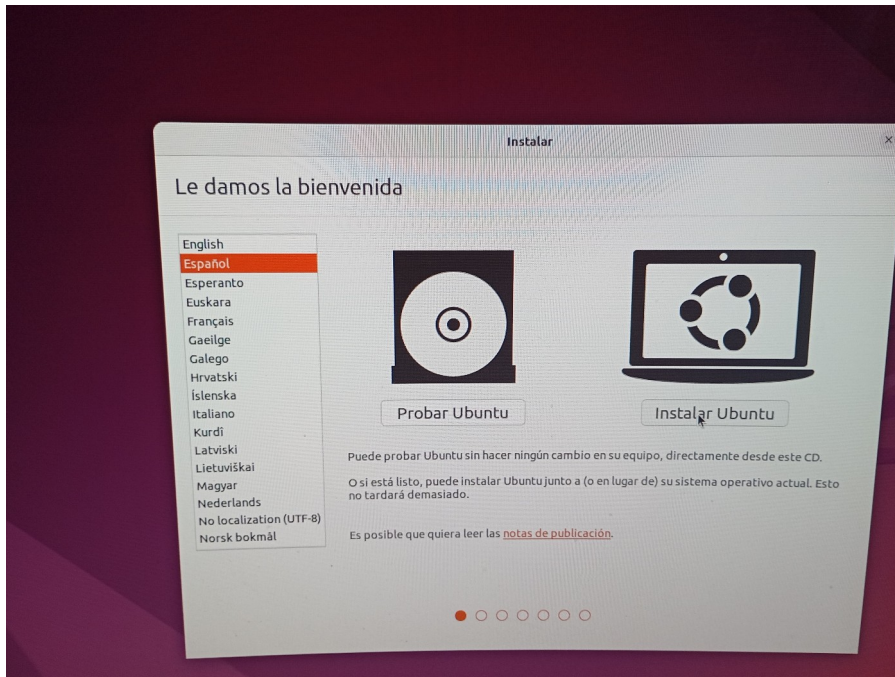
Seleccionem la partició a la que vulguem instal·lar Windows



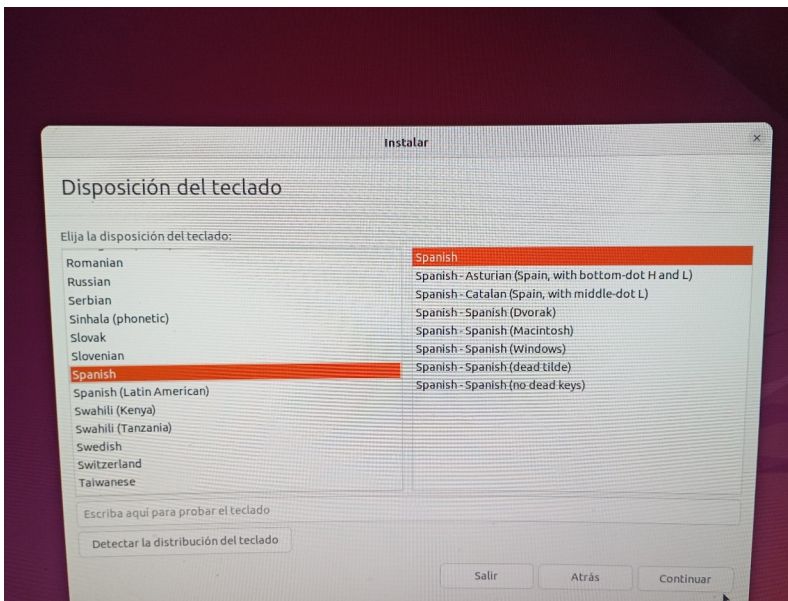
Finalment esperarem a que s'instal·li Windows que aproximadament ens ha trigat uns 15 20 minuts (depene de la màquina, el disc dur i el medi d'instal·lació), aplicarem diferents configuracions de recomanació i publicitat de Windows i ja tindrem el sistema operatiu completament instal·lat i configurat:

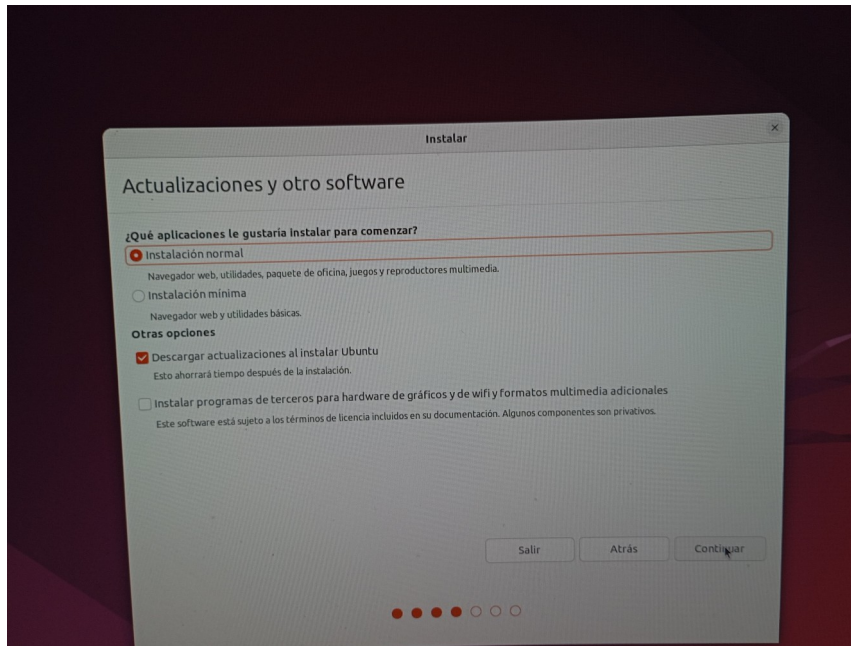


1.6. Instal·lació Ubuntu i creació dual boot

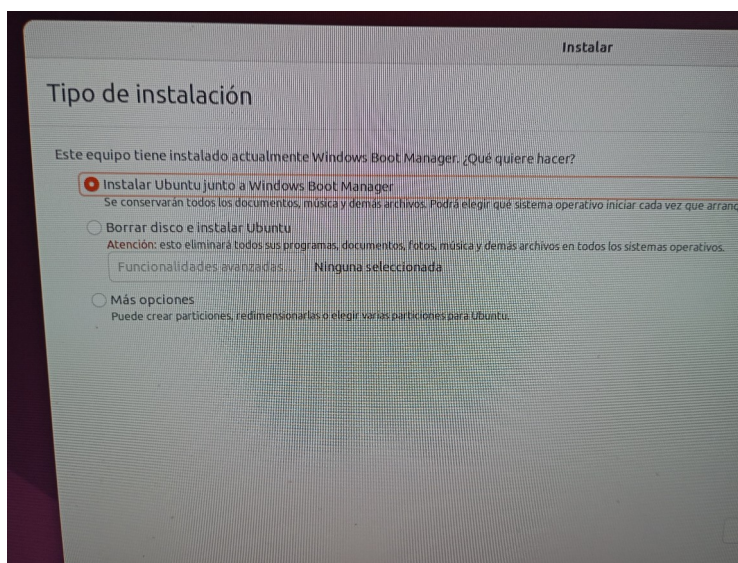


Amb el pen que hem creat, comencem la típica instal·lació de l'Ubuntu:



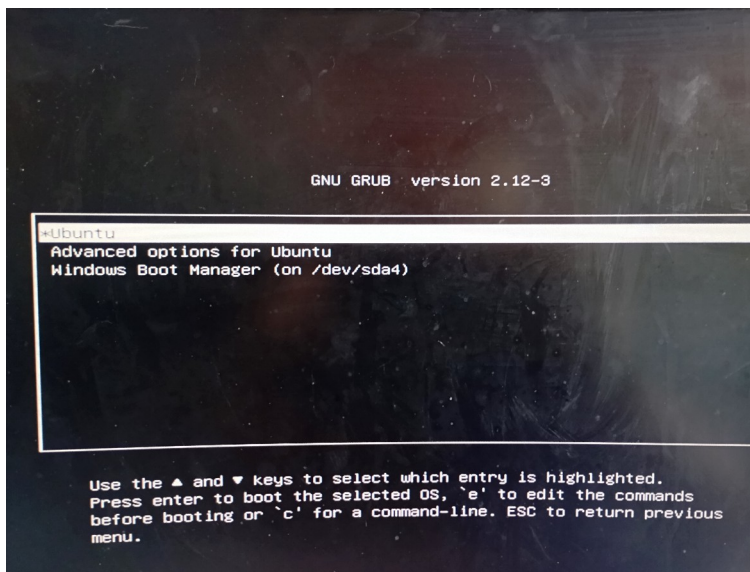


I en el moment de configurar les particions, escollim la opció per instal·lar l'Ubuntu junt amb el "Windows Boot Manager", per fer el dual boot:



Ara, seguim amb la instal·lació.

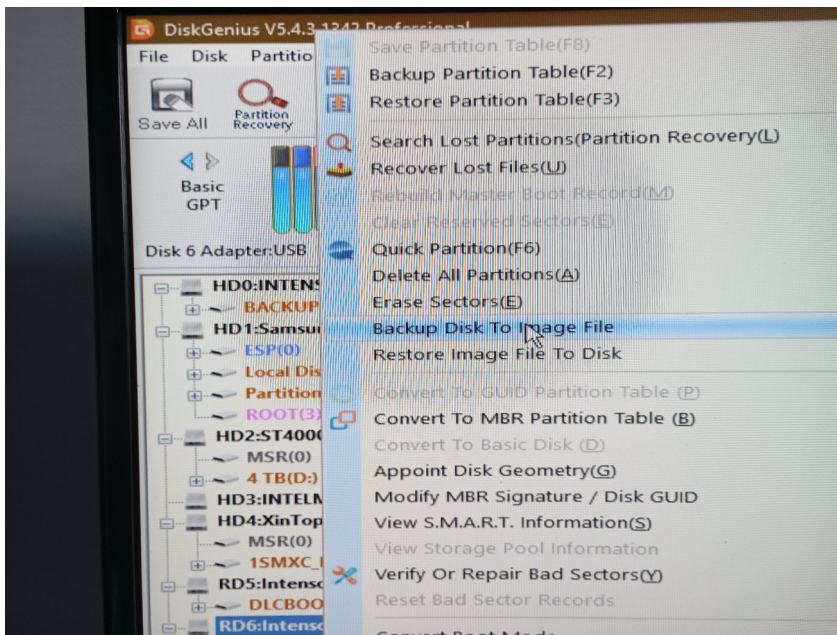
I un cop acabada, se'ns reiniciarà la màquina. El grub que se'ns haurà instal·lat es veurà així:



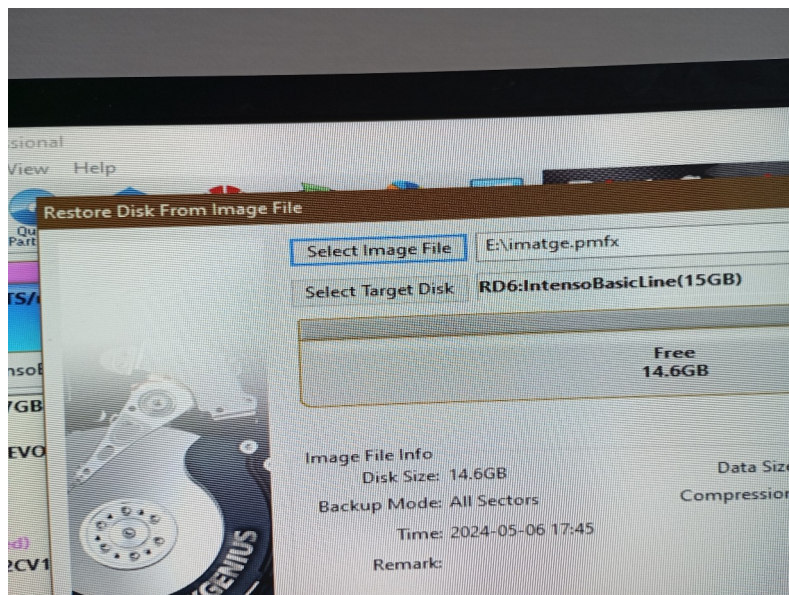
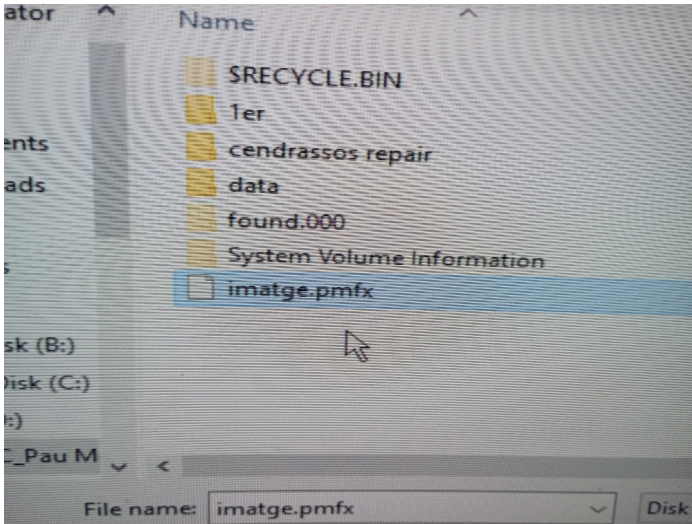
Amb una opció per arrencar el Windows.

2. Imatges de Particions

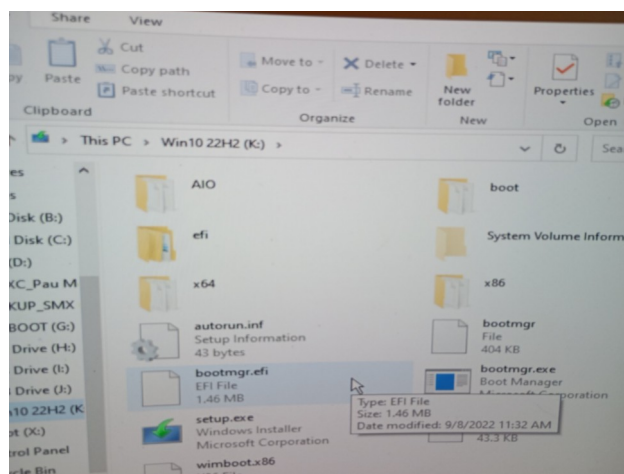
Amb el DLC Boot, creem una imatge d'un disc amb el DiskGenius:



Després, esborrem el disc dur i restaurem la imatge:



I veurem que tot ha tornat a la normalitat, encara que haguem formatat el disc dur:



3. DLC Boot CD

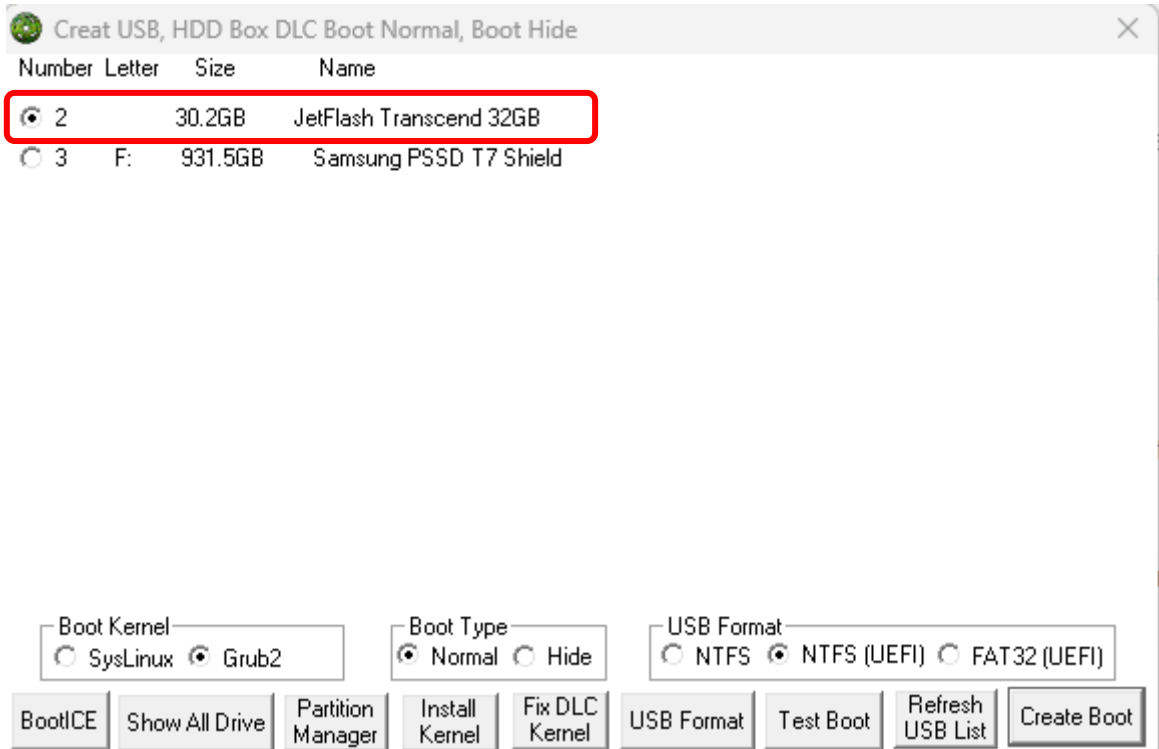
Fer el pen drive instal·lable DLC

- Utilitzant la mateixa eina que porta el DLC fer un pen drive bootable amb aquesta eina. Si ja el tens no et cal fer aquesta part. 1 Punt.

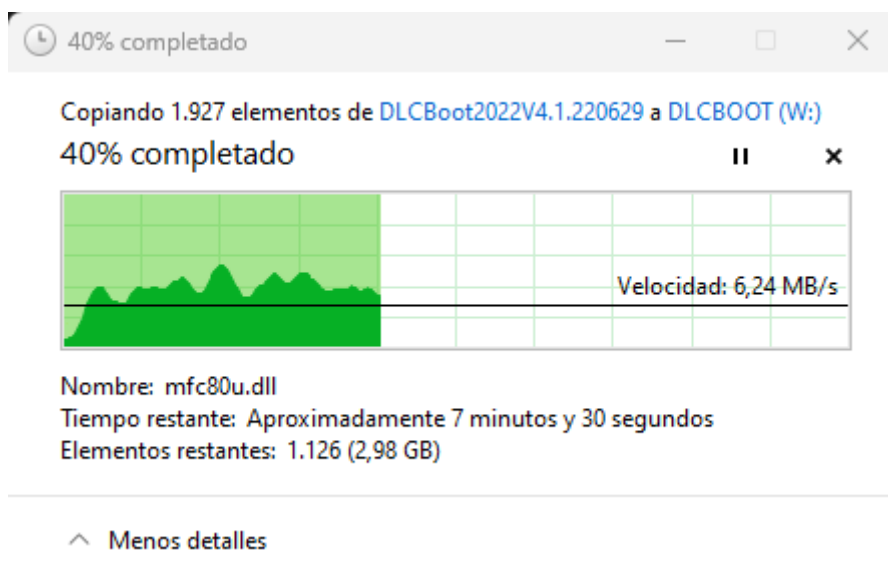
Per instal·lar DLC boot a un pen drive el que haurem de fer un cop dintre del aplicatiu serà ingressar al apartat del símbol del pen drive senyalat a la imatge.



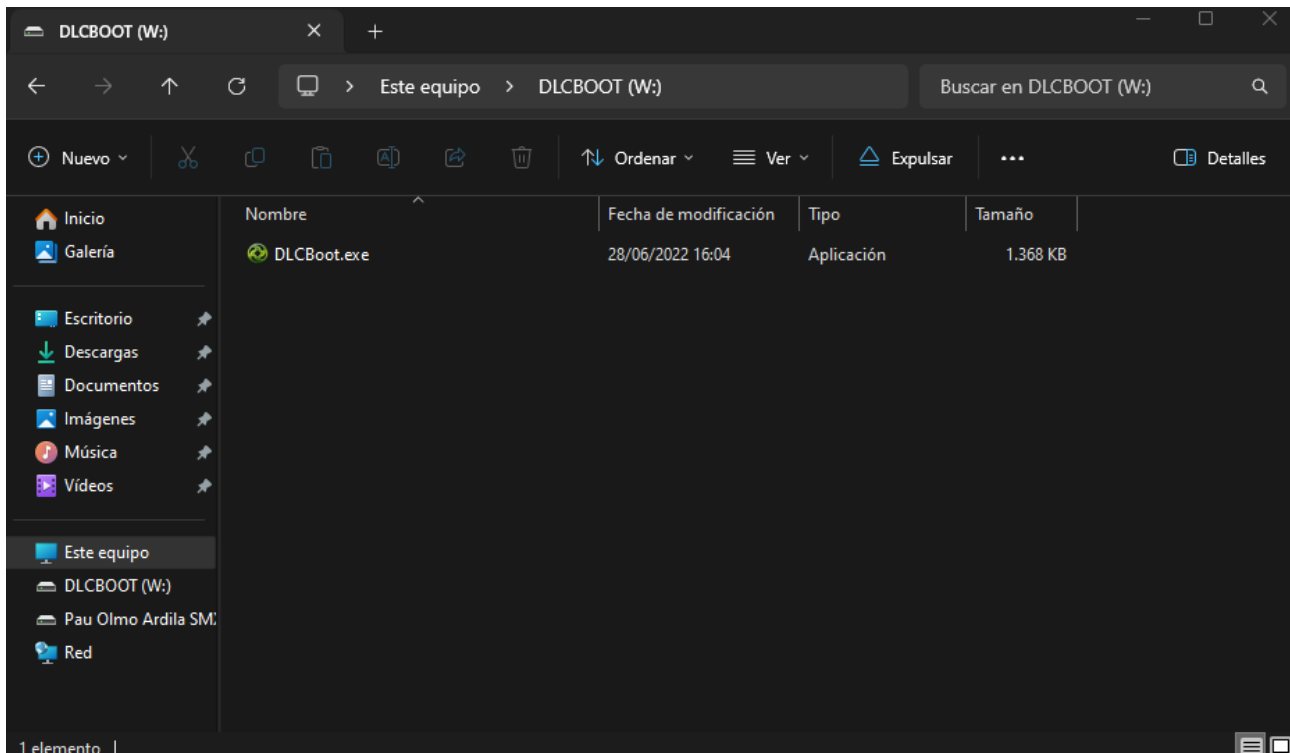
Ens sortirà un apartat com el següent el qual haurem de seleccionar en el dispositiu que vulguem instal·lar el DLC boot. Aquesta eina es molt útil per casos en els que els ordinadors per part de software s'espantlen i el que podem fer amb el DLC boot es ingressar des de la bios i seleccionar la opció d'arrancada del pen drive.



Seguidament esperarem a que la operació sigui copiada al pen drive, aproximadament dura uns 10 minuts depenent de la velocitat del teu pen drive.



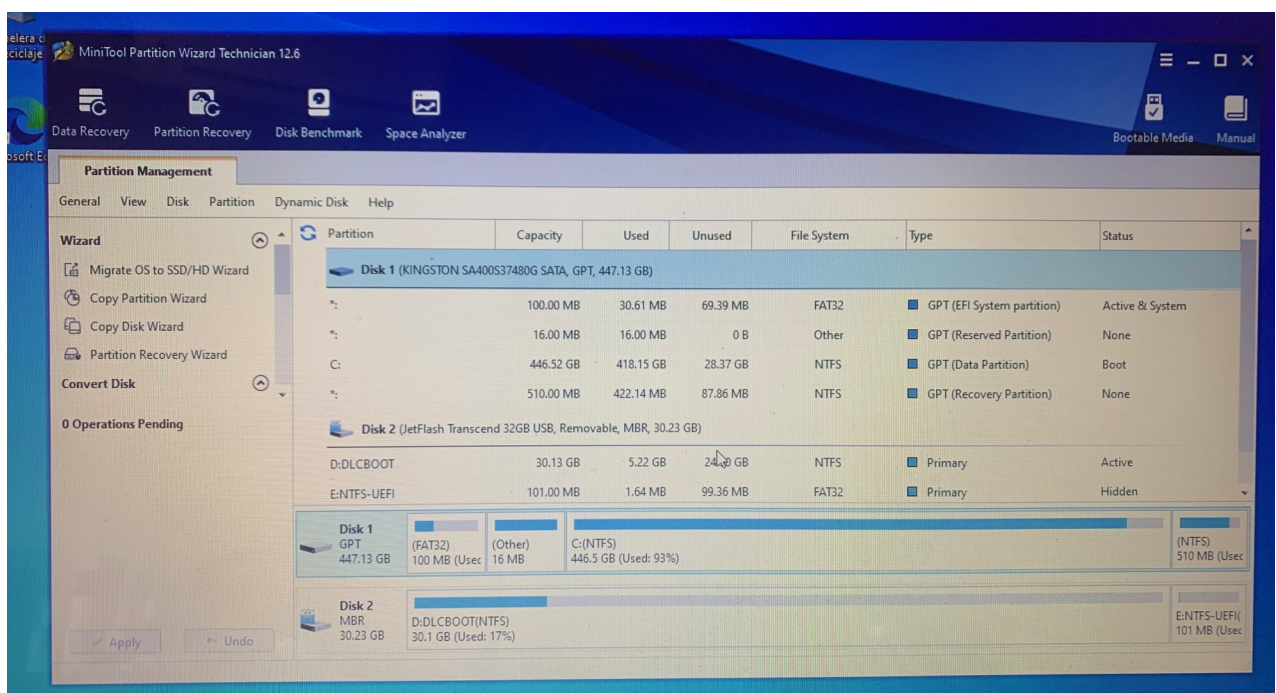
Ja estarà descarregada la eina al pen drive. Ens sortira tal que així:



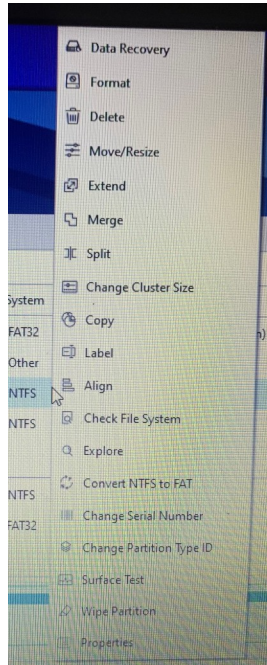
4.Modificar les particions amb el DLC boot CD 2 punts

Per modificar les particions amb el DLC boot es pot fer de dues maneres diferents una iniciant el pen drive com a primera opció de bootable al ordinador i la segona es un cop instal·lat Windows simplement insertem el pen al ordinador i executem la aplicació en mode administració. Pot ser que es necessiti desactivar l'antivirus perquè aquesta aplicació requereix d'edició de programari.

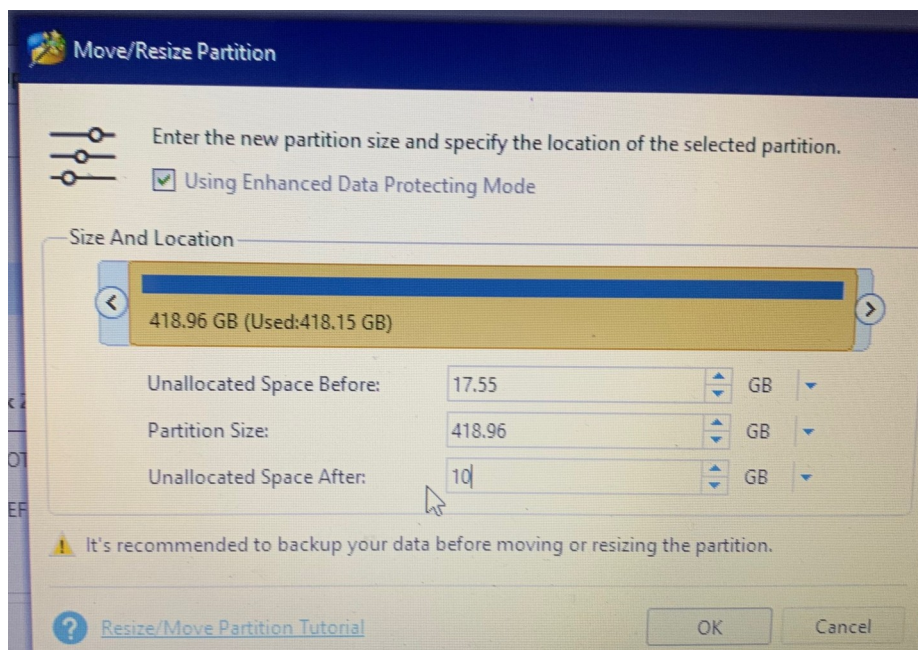
En el nostre cas hem escollit el partition wizard.



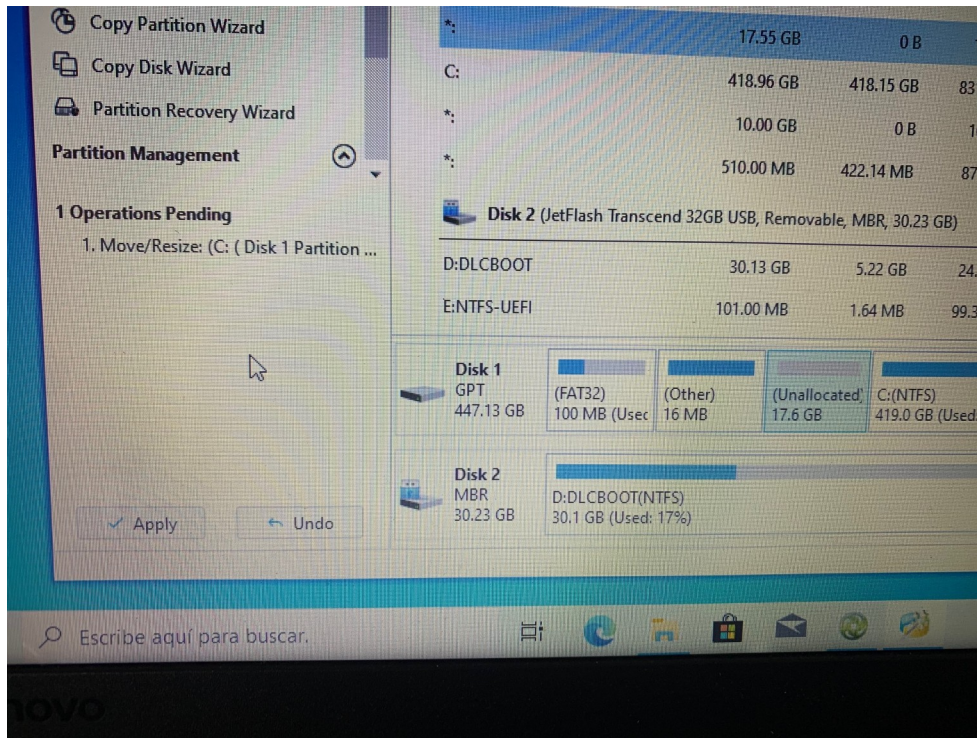
Per redimensionar o reduir la partició el que haurem de fer es clic dret sobre la partició que vulguem reduir l'espai i seguidament fer clic a l'opció move/resize.



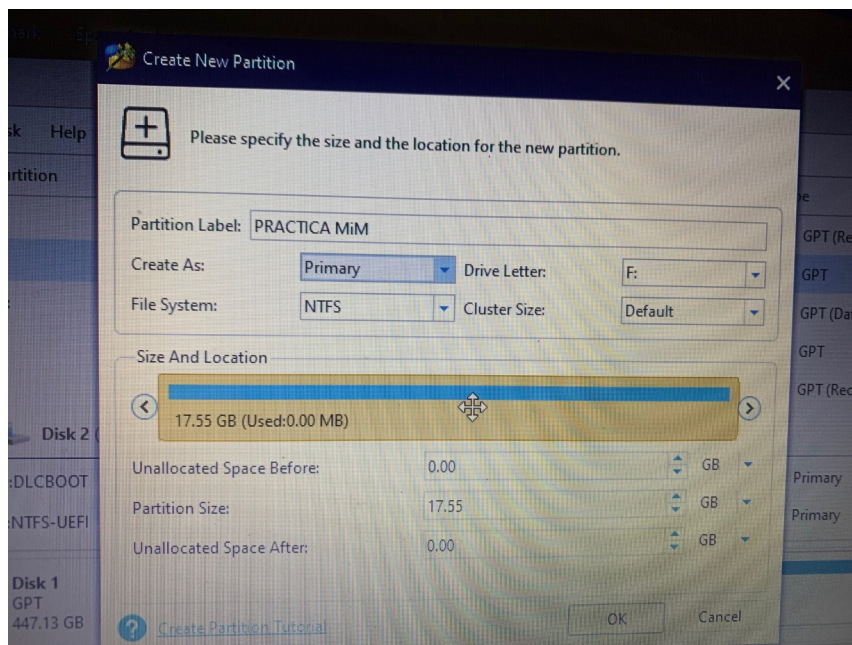
Un cop aquí el que haurem de fer es reservar la part desitjada per crear una nova partició



Un cop tinguem la partició el que haurem de fer es editar-la per posar-li un format i seguidament poder crear la partició. Simplement fem clic dret i a crear.

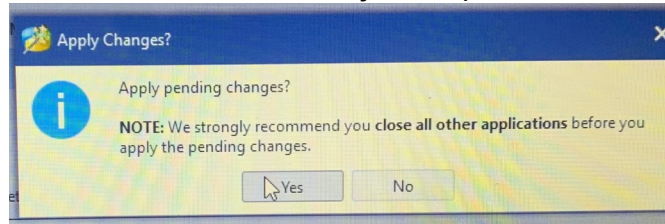


Editem la partició i li posem un nom i un format.

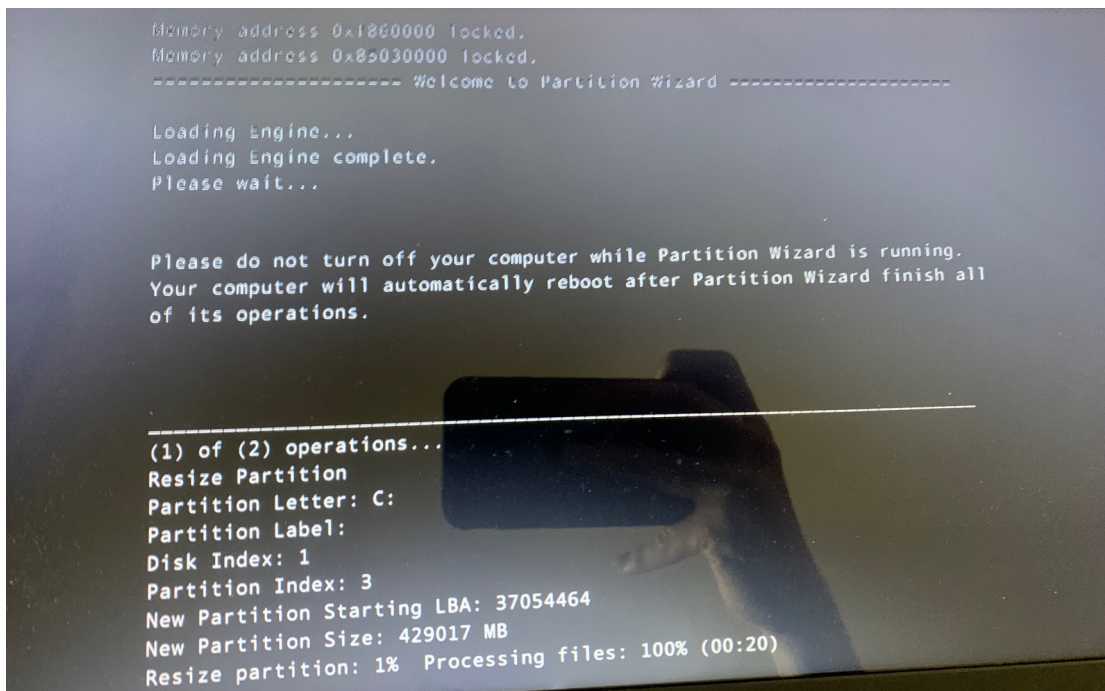


Finalment apliquem els canvis

Seguidament ens reiniciara l'ordinador i començara un procés com el següent i esperarem



a que finalitzi, un cop finalitzat ja tindrem la partició.



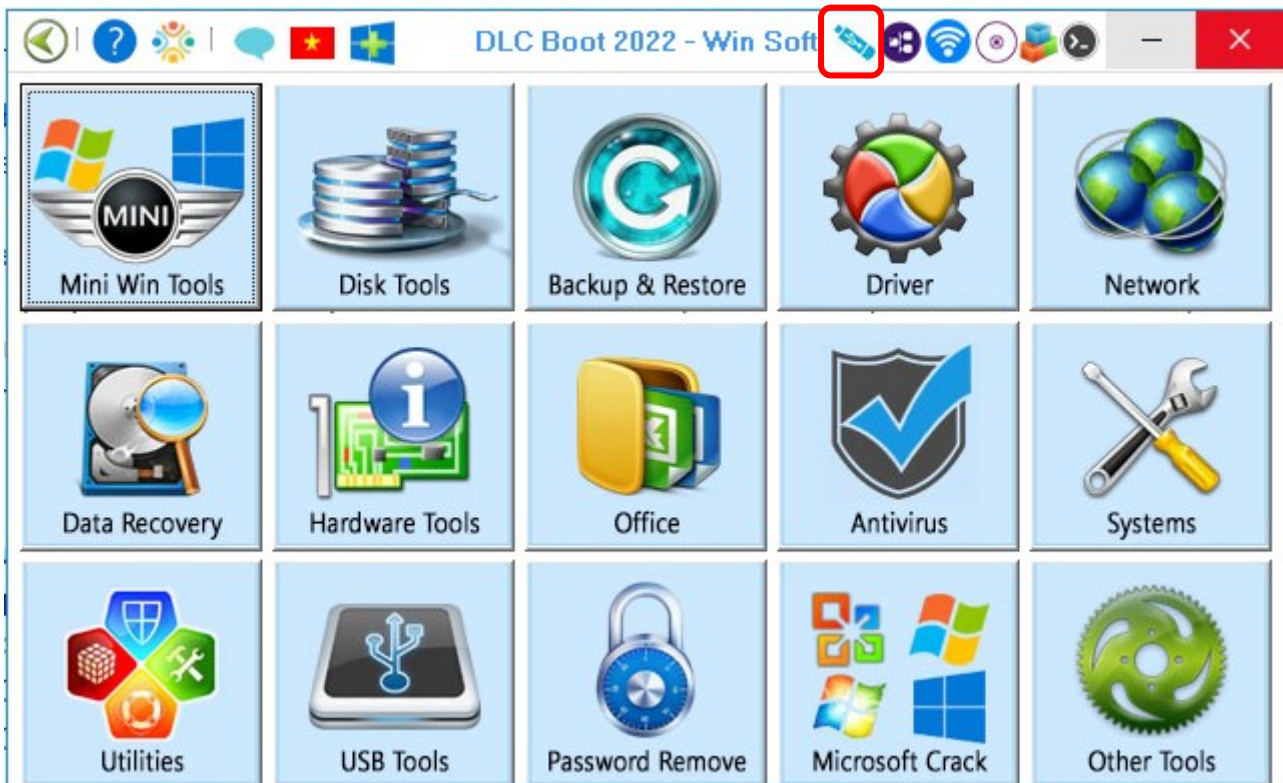
4. Fer el pen drive instal·lable DLC

4.1. Utilitzant la mateixa eina que porta el DLC fer un pen drive bootable amb aquesta eina. Si ja el tens no et cal fer aquesta part. 1 Punt.

5.

6. Per instal·lar DLC boot a un pen drive el que haurem de fer un cop dintre del aplicatiu serà ingressar al apartat del símbol del pen drive senyalat a la imatge.

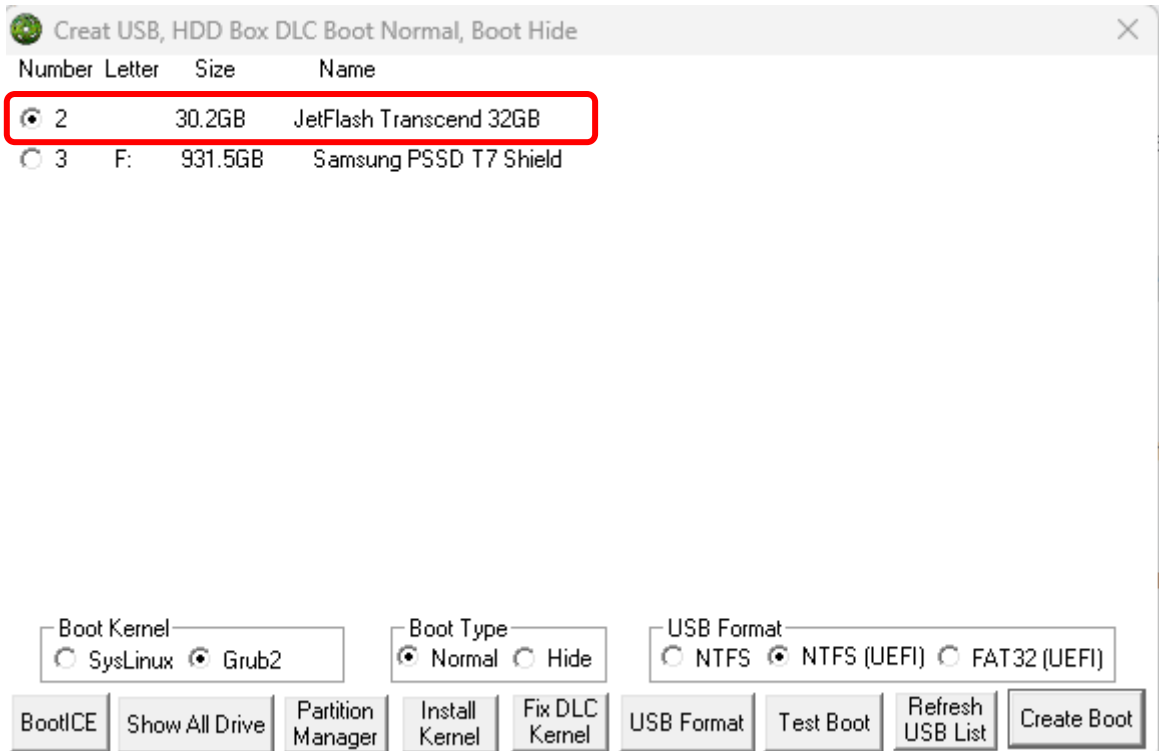
7.



8.

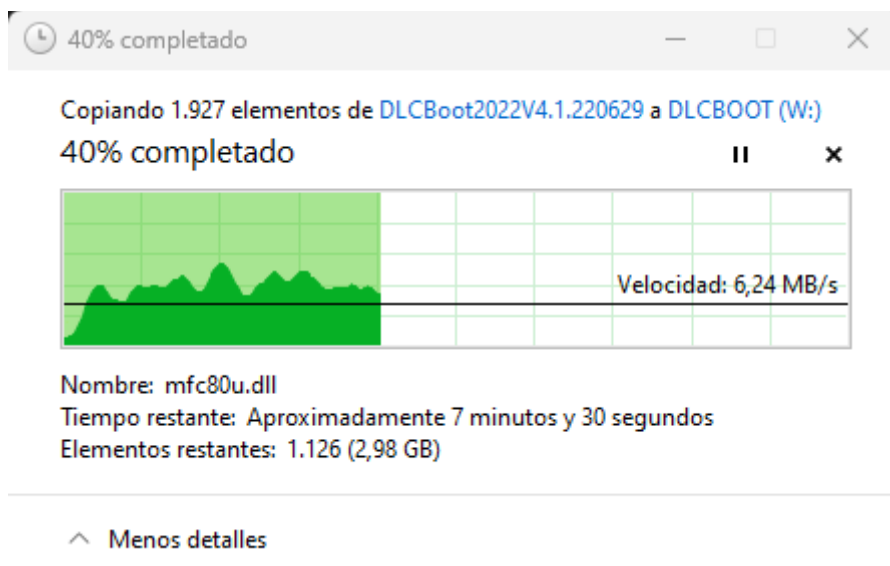
9.

10. Ens sortirà un apartat com el següent el qual haurem de seleccionar en el dispositiu que vulguem instal·lar el DLC boot. Aquesta eina es molt útil per casos en els que els ordinadors per part de software s'espantlen i el que podem fer amb el DLC boot es ingressar des de la bios i seleccionar la opció d'arrancada del pen drive.

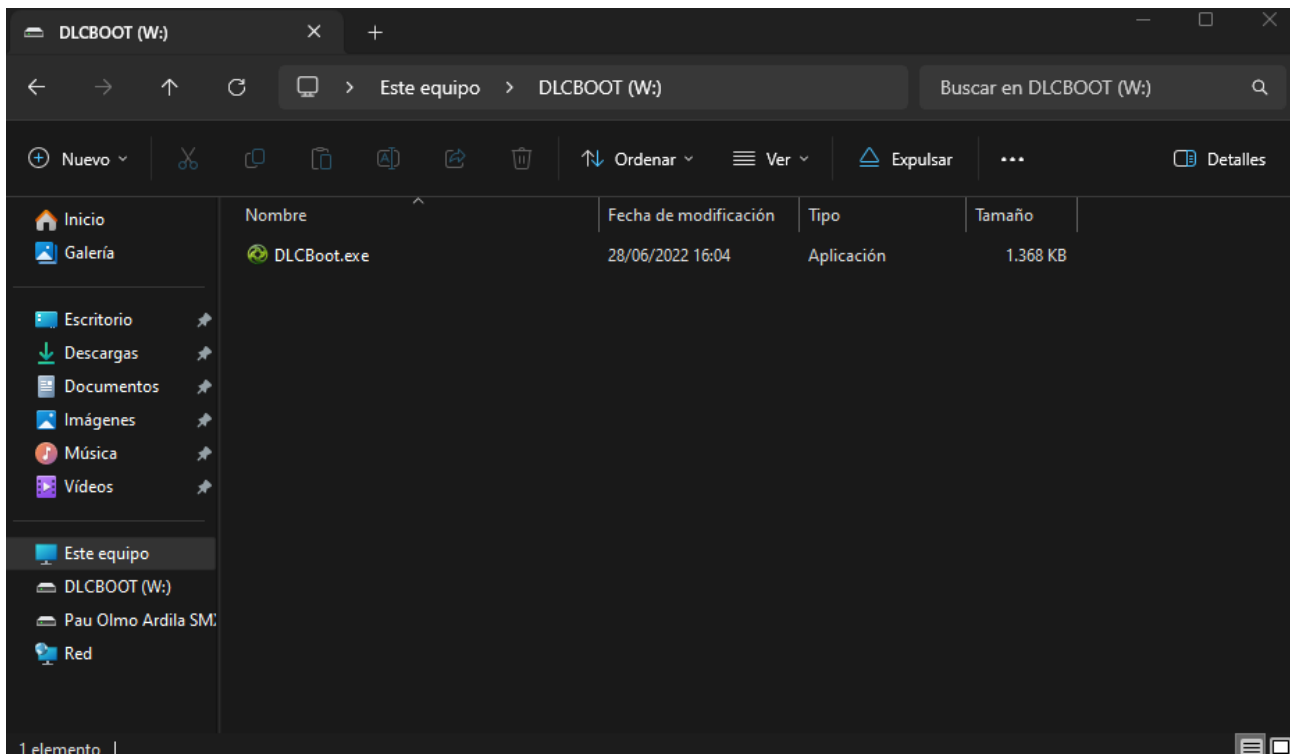


11. Seguidament esperarem a que la operació sigui copiada al pen drive, aproximadament dura uns 10 minuts depenent de la velocitat del teu pen drive.

12.



13. I ja estarà descarregada la eina al pen drive. Ens sortira tal que així:



14.

15.

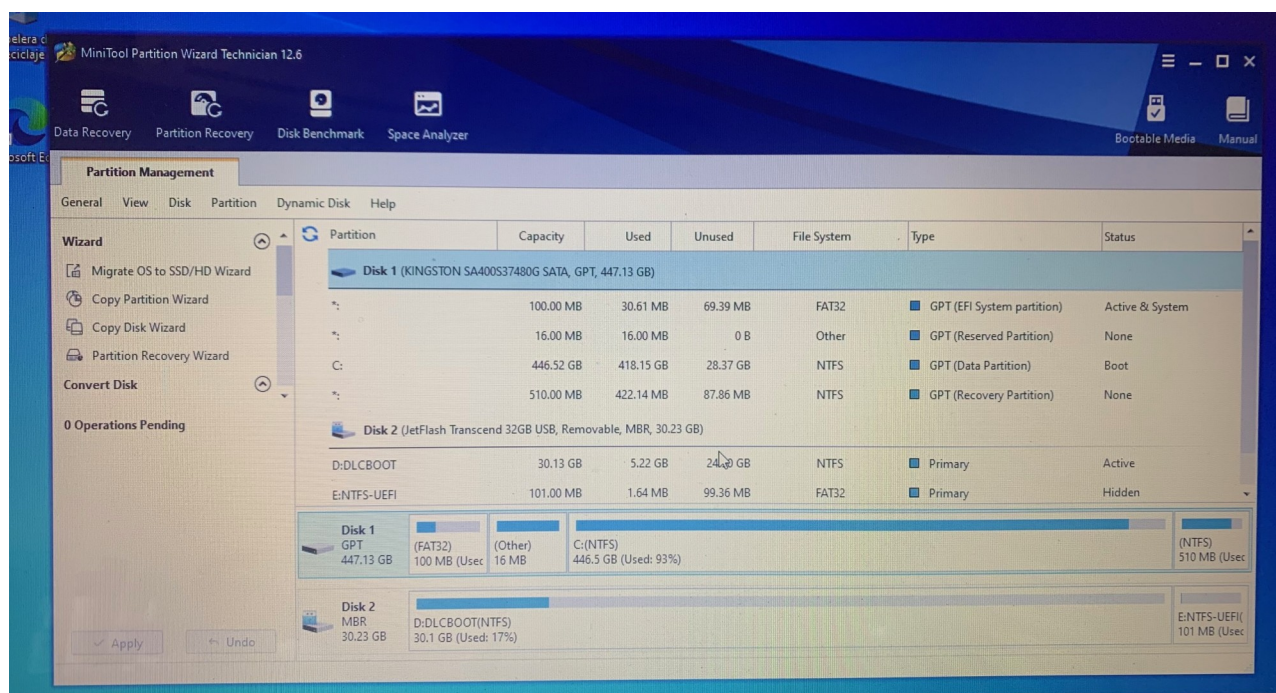
16. 4.Modificar les particions amb el DLC boot CD 2 punts

17.

18. Per modificar les particions amb el DLC boot es pot fer de dues maneres diferents una iniciant el pen drive com a primera opció de bootable al ordinador i la segona es un cop instal·lat Windows simplement insertem el pen al ordinador i executem la aplicació en mode administració. Pot ser que es necessiti desactivar l'antivirus perquè aquesta aplicació requereix d'edició de programari.

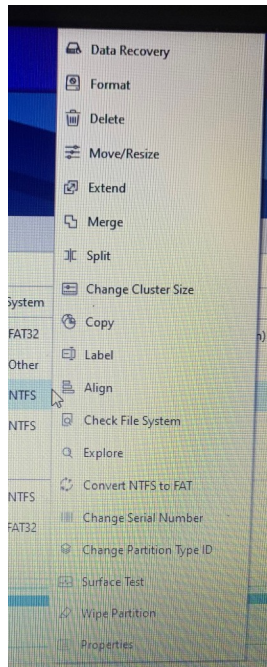
19.

20. En el nostre cas hem escollit el partition wizard.

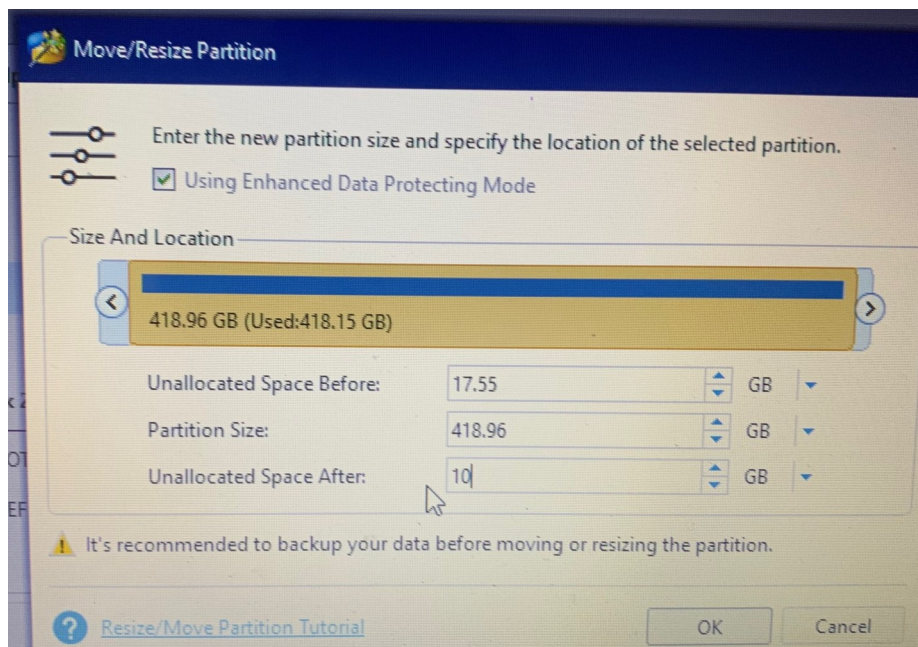


21.

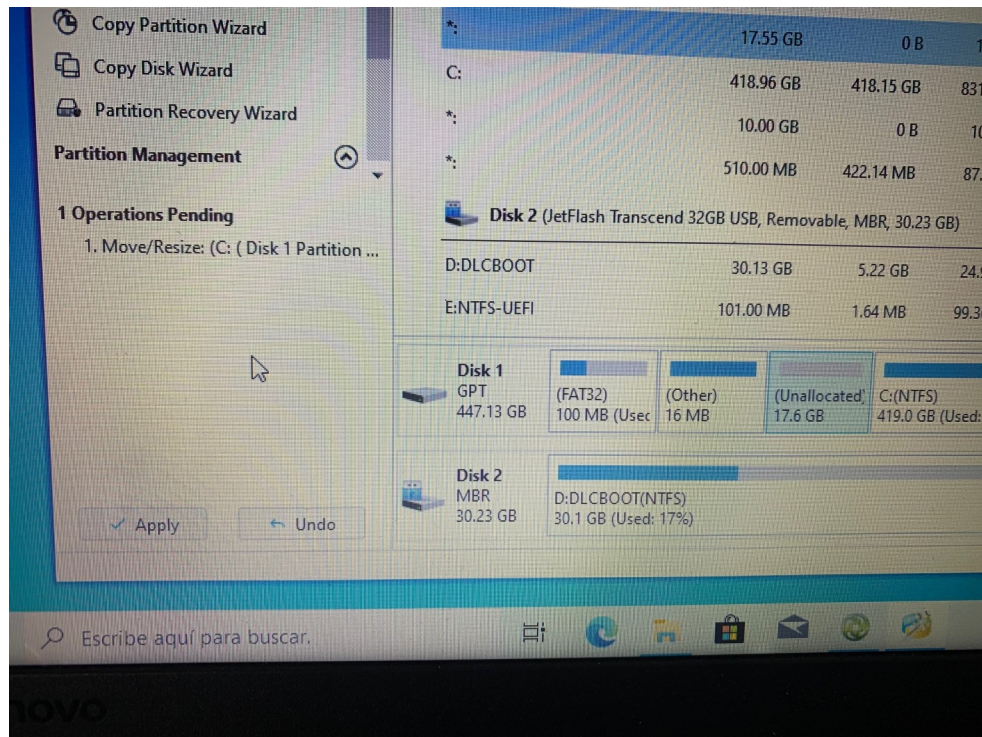
22. Per redimensionar o reduir la partició el que haurem de fer es clic dret sobre la partició que vulguem reduir l'espai i seguidament fer clic a l'opció move/resize.



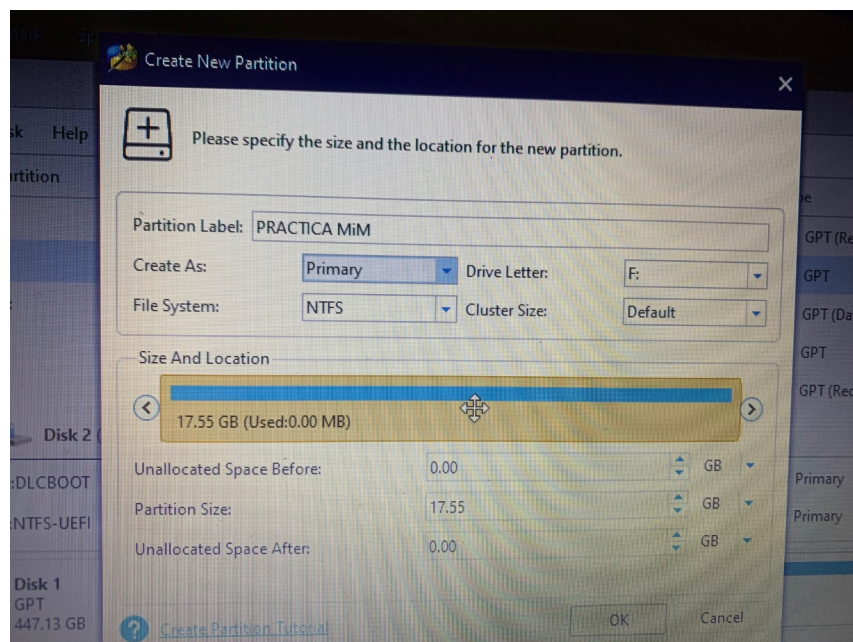
23. Un cop aquí el que haurem de fer es reservar la part desitjada per crear una nova partició



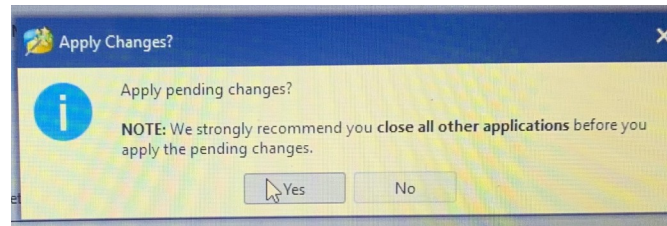
24. Un cop tinguem la partició el que haurem de fer es editar-la per posar-li un format i seguidament poder crear la partició. Simplement fem clic dret i a crear.



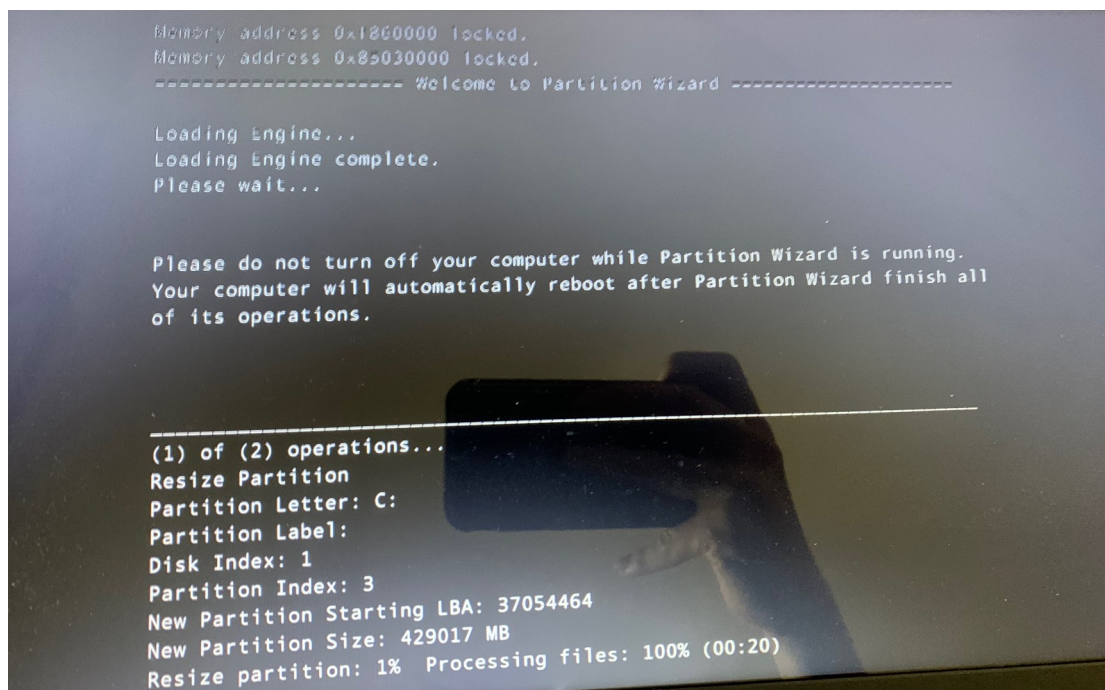
25. Editem la partició i li posem un nom i un format.



Finalment apliquem els canvis



Seguidament ens reiniciara l'ordinador i començara un procés com el següent i esperarem a que finalitzi, un cop finalitzat ja tindrem la partició.



Webgrafia

[Understanding the various grub modules](#)

[WimBoot for Grub2](#)