

Replicación del entorno de desarrollo del laboratorio de AEV

por Frederic García
Curso 2k18/2k19

El propósito del presente anexo es el de replicar el entorno de desarrollo existente en el laboratorio para la realización de las prácticas y del trabajo final de la asignatura *Arquitectura y Entornos de desarrollo para videoconsolas*. Esta ha sido la secuencia de pasos realizados.

1 Instalación de un gestor de máquinas virtuales

De entre los gestores de máquinas virtuales (MV) existentes se ha escogido el de VirtualBox de Oracle dado que es gratuito y multiplataforma (Linux, Windows y MacOS). Puede descargarse desde www.virtualbox.org. El sistema operativo sobre el que se ha instalado para la realización del anexo ha sido Windows. El proceso es análogo al descrito independientemente del sistema operativo anfitrión. La versión instalada de VirtualBox ha sido la 6.06. La *Figura 1* muestra la ejecución del gestor tras su instalación.

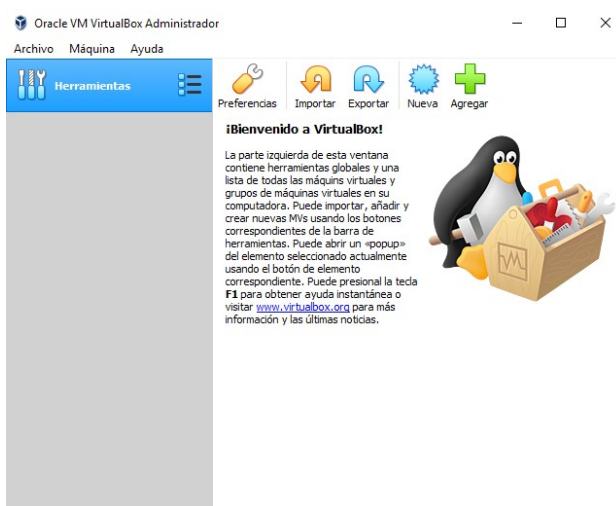


Figura 1. Ejecución de VirtualBox 6.06

2 Creación de la máquina virtual con la distribución de Linux existente en el laboratorio

El primer paso ha sido descargar la ISO de GNU/Linux Kubuntu 18.04¹ desde <https://kubuntu.org/getkubuntu/> ya que era la versión instalada en el laboratorio.

¹ A la fecha de realización de este anexo, la versión de descarga de Kubuntu 18.10 existente en <https://kubuntu.org/getkubuntu/> era la de Kubuntu 18.04.2 LTS.

El segundo paso ha sido el de crear la máquina virtual. Para ello, desde VirtualBox (opción de menú *Máquina/Nueva*) se ha escogido nombre para la máquina virtual, seleccionado la ruta donde se ubicará, el tipo de sistema operativo y la versión de este. La *Figura 2* ilustra este paso.

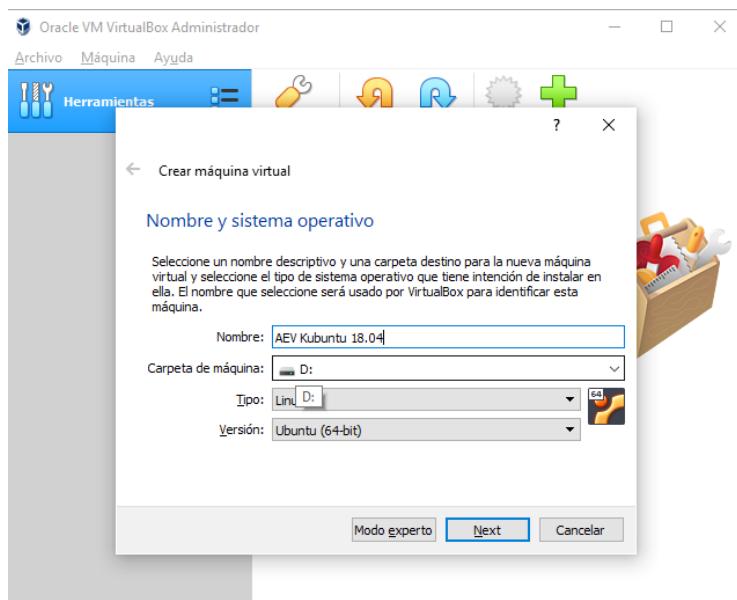


Figura 2. Creando la máquina virtuales

Tras darle al botón de *Next*, se escoge el tamaño de memoria recomendado (1024 MB) pues no se requiere de mayor tamaño dado que sólo instalaremos en ella las herramientas necesarias para el desarrollo de la asignatura. Se aumenta el tamaño del disco duro a 20 GB (necesarios para crear posteriormente una imagen ISO de la instalación) y *Crear un disco duro virtual ahora*. Se navega por el resto de ventanas del proceso de instalación aceptando las opciones que vienen por defecto hasta crear la máquina virtual. La *Figura 3* muestra las opciones finales tras el proceso de creación de la máquina virtual.

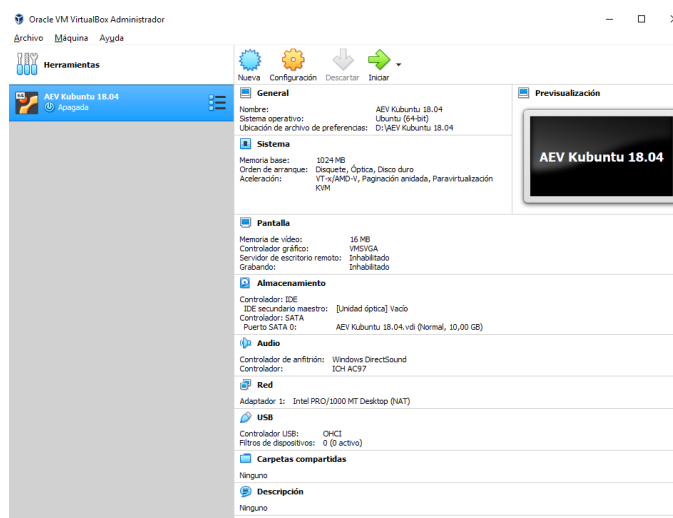


Figura 3. Resultado final tras la creación de la MV.

Dado que tenemos el sistema operativo huésped en un fichero de imagen ISO, le indicamos a VirtualBox dónde se encuentra y cuál es. Para ello, en el apartado de *Almacenamiento*, *Controlador IDE*², cambiamos la fuente de *unidad óptica* por *Seleccionar imagen de disco* e iniciamos la instalación. Las figuras 4 y 5 ilustran este paso.

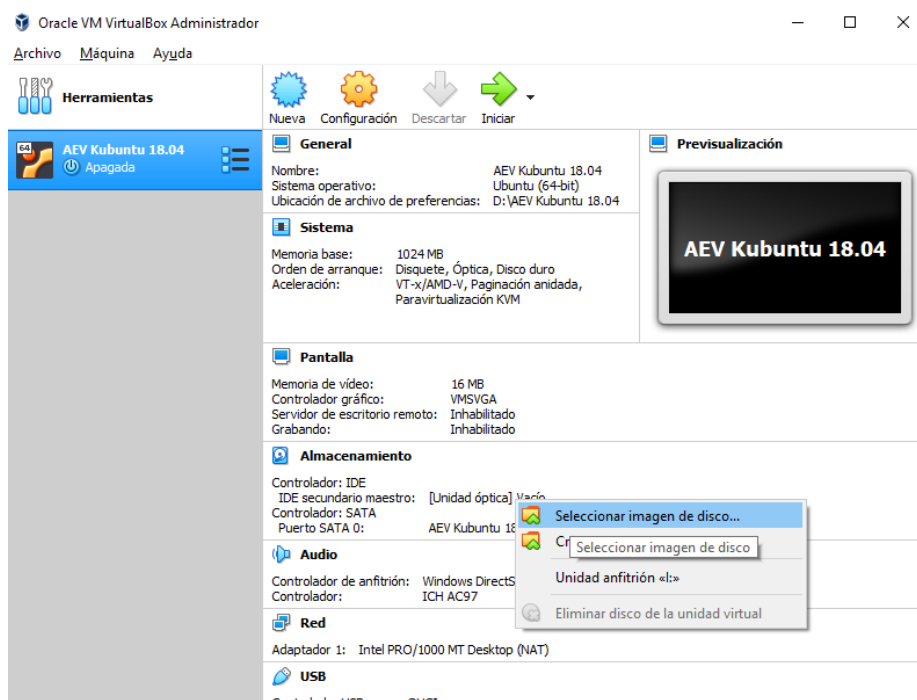


Figura 4. Seleccionando la imagen de disco de Kubuntu 18.04

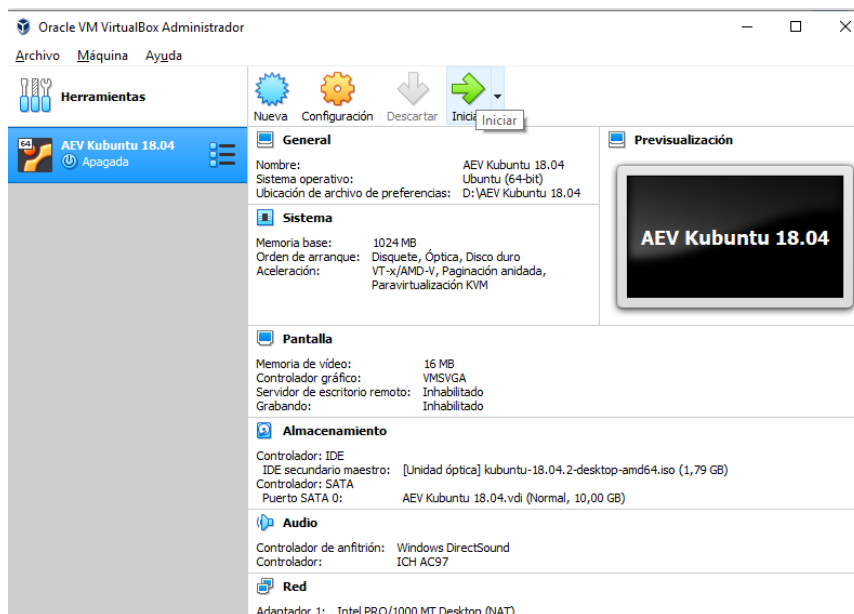


Figura 5. Seleccionada la imagen de disco de Kubuntu 18.04 e inicio de la instalación.

² Esta opción puede variar dependiendo de la configuración del sistema anfitrión. Habrá que ajustarla para indicarle que la fuente es un imagen de disco en lugar de una unidad óptica, usb...

Instalamos Kubuntu 18.04, seleccionando el idioma deseado para la instalación, escogemos la opción de *Instalación mínima*, en configuración de disco, seleccionamos *tipo de instalación guiada* y acabamos de configurar la instalación según nuestras preferencias. La *Figura 6* muestra el resultado final tras la instalación.

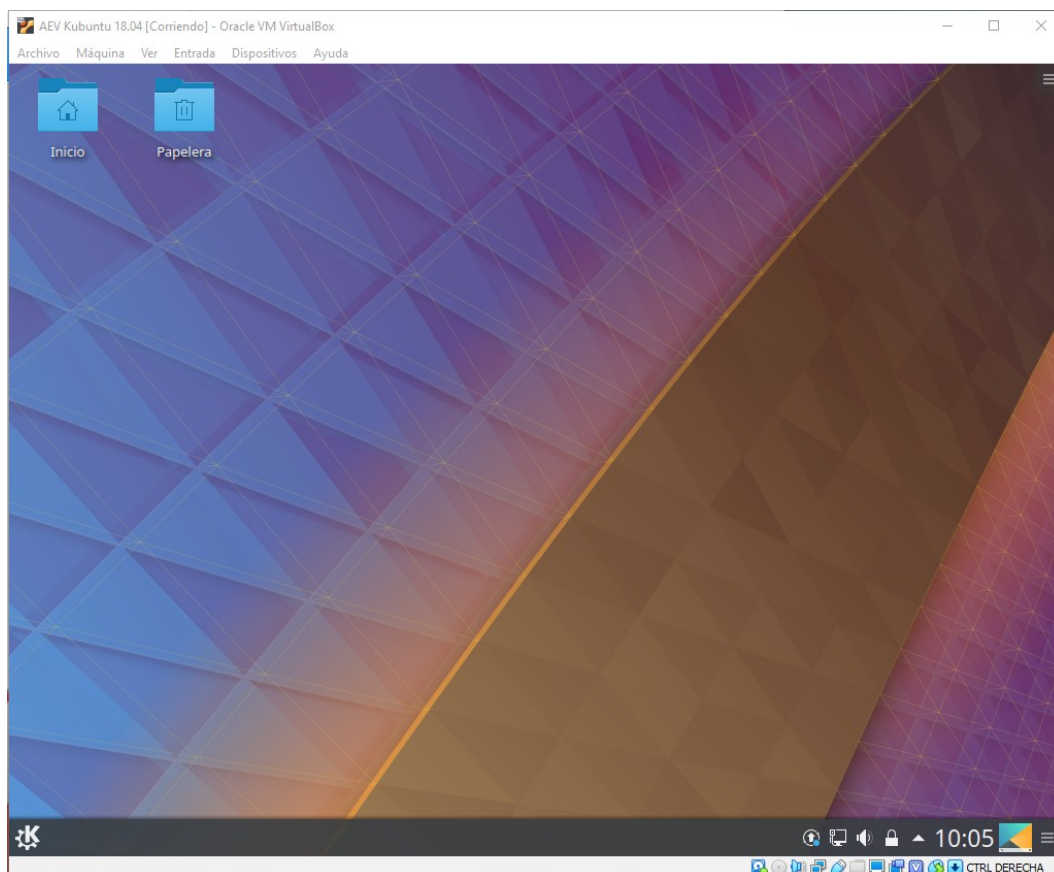


Figura 6. Máquina Virtual con la instalación de Kubuntu 18.04 corriendo.

3 Instalación de DevkitPro

Se sigue la secuencia de pasos del ANEXO A de la práctica 1 realizando mínimos modificaciones. De forma resumida, la secuencia de pasos ha sido:

1. Abrir un terminal (Konsole), ubicarnos en el directorio `/tmp` e instalar el instalador de paquetes Arch Linux *pacman*:

```
$ wget https://github.com/devkitPro/pacman/releases/download/devkitpro-pacman-1.0.1/devkitpro-pacman.deb
```

```
$ sudo dpkg -i /tmp/devkitpro-pacman.deb
```

Y aparecerá en /opt

```
$ ls -l /opt/devkitpro/
```

2. Ahora ya es posible instalar la última versión de devkitPro, para lo que hay que ejecutar:

```
$ sudo dkp-pacman -U https://downloads.devkitpro.org/devkitpro-keyring-r1.787e015-2-
```

```
any.pkg.tar.xz
```

```
$ sudo dkp-pacman -Sy
```

```
$ sudo dkp-pacman -Syu
```

```
$ sudo dkp-pacman -Sl | while read linia; do cut -d" " -f2; done >
totesLesLliberies.txt
```

```
$ sudo dkp-pacman -S `cat totesLesLliberies.txt`
```

4 Instalando el emulador de NDS DeSmuMe

Al igual que en el proceso de instalación de devkitPro, siguiendo los pasos del ANEXO A de la práctica 1, instalamos el emulador DeSmuME. Desde la misma terminal ejecutamos los comandos:

```
$ sudo apt-get install desmume
```

Para comprobar el funcionamiento, deberían estar declaradas las variables durante la instalación, pero habría que actualizar las variables de entorno ejecutando la orden

```
$ source /etc/profile.d/devkit-env.sh
```

```
$ kate ~/.bashrc
```

e incluir al final del fichero la orden:

```
source /etc/profile.d/devkit-env.sh
```

Abrimos un nuevo terminal y comprobamos si las variables de entorno están definidas ejecutando el comando:

```
$ set | grep devkitpro
```

Ha sido necesario instalar el comando *make* para poder compilar los ejemplos de NDS y 3DS. Por ello.

```
$ sudo apt-get install make
```

5 Instalando el emulador de 3DS Citra

De la página web del emulador Citra descargamos la versión instalada en el laboratorio `citra-linux-20190126-bad2e08.tar.xz` y **lo descomprimos en el directorio `/opt`** (esto es importante, para poder realizar posteriormente nuestra distribución portable).

```
$ sudo tar xvf citra-linux-20190126-bad2e08.tar.xz -C /opt/
```

Ahora añadimos al `PATH` la ubicación donde está instalado citra. Al igual que se hizo con las variables de entorno de devkitPro, ejecutamos desde la consola el comando:

```
$ kate ~/.bashrc
```

e incluimos al final del fichero la orden:

```
export PATH=$PATH:/opt/citra-linux-20190126-bad2e08
```

Podremos

6 Haciendo nuestro entorno de desarrollo portable

Para ello haremos uso del software *respin* que podemos o bien descargar desde:

https://launchpad.net/~sergiomejia666/+archive/ubuntu/respin/+files/respin_2.0.6-0-76~ubuntu18.04.1_amd64.deb
o desde la consola.

```
$ wget https://launchpad.net/~sergiomejia666/+archive/ubuntu/respin/+files/respin_2.0.6-0-76~ubuntu18.04.1_amd64.deb
```

```
$ sudo dpkg -i respin_2.0.6-0-76~ubuntu18.04.1_amd64.deb
```

La consola nos informará que no se ha podido instalar por no poder resolver las dependencias. Ejecutamos el siguiente el comando que instalara todas las dependencias no resueltas, entre ellas las del *respin*.

```
$ sudo apt-get -f install
```

Para hacer una imagen ISO de nuestra distribución, ejecutamos el comando *respin* con la siguiente opción.

\$ sudo respin dist

Es un proceso que se tomará su tiempo. Paciencia. La imagen resultante se ubica en /home/respin/respin que podremos volcar a un CD o un *pendrive* USB.