



Crear mapas de Quake 1 desde GNU/Linux

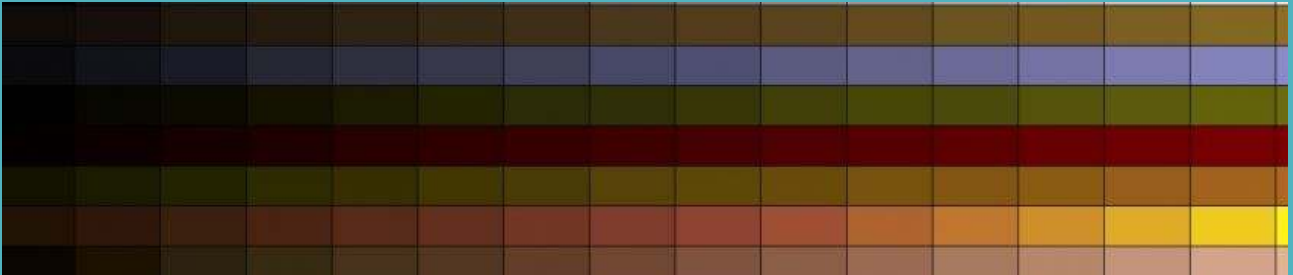
fanta <fanta@56k.es>



Sumario

Crear mapas de Quake 1 desde GNU/Linux.....	1
Paleta de colores en GIMP.....	2
Motor para ejecutar Quake 1.....	4
Paso 1: Compilar el motor Quakespasm.....	4
Paso 2: Obtener los archivos pak del juego y crear un lanzador.....	5
Probar a cargar mapas.....	7
Los archivos PAK.....	7
Extraer el contenido de los archivos PAK.....	8
La forma de cargar mapas.....	9
Configurando el editor de mapas.....	10
Configuración antes de Crear un Nuevo mapa.....	11
Nuestro primer mapa sin nada que hacer.....	14

Paleta de colores en GIMP



Hola amigo/e/a . La idea es dejar descrito la forma en la que podemos crearnos nuestros propios mapas de Quake 1 y hacerlo todo con software libre.

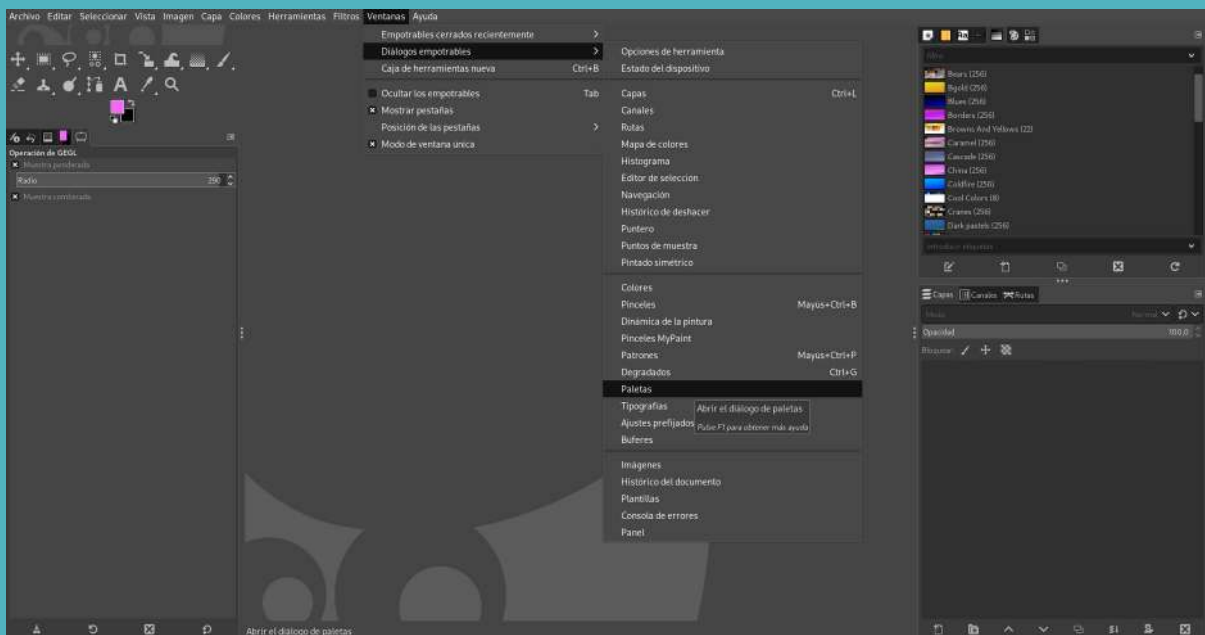
Para empezar vamos a necesitar disponer de una distro GNU+Linux (Arch, Debian, Ubuntu, ...) en la que tengamos instalado GIMP (el editor de imágenes).

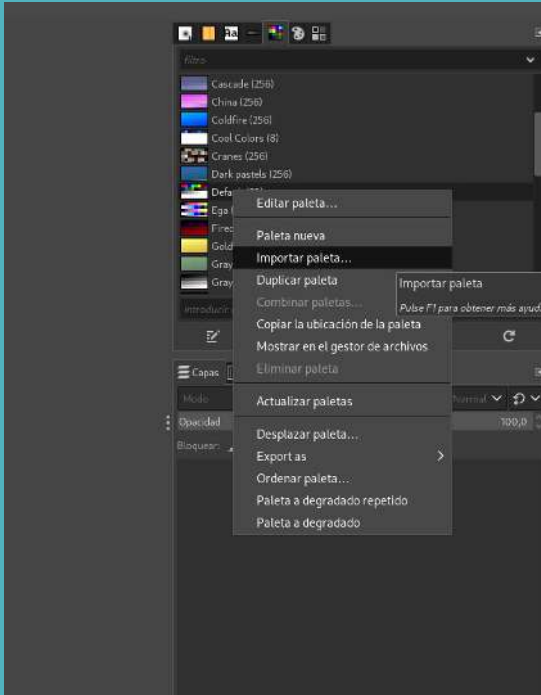
1. **Descargamos y descomprimos** la paleta de colores quake 1.

```
$ wget "https://quakewiki.org/w/images/d/d2/quake_palette.zip"
$ unzip quake_palette.zip
```

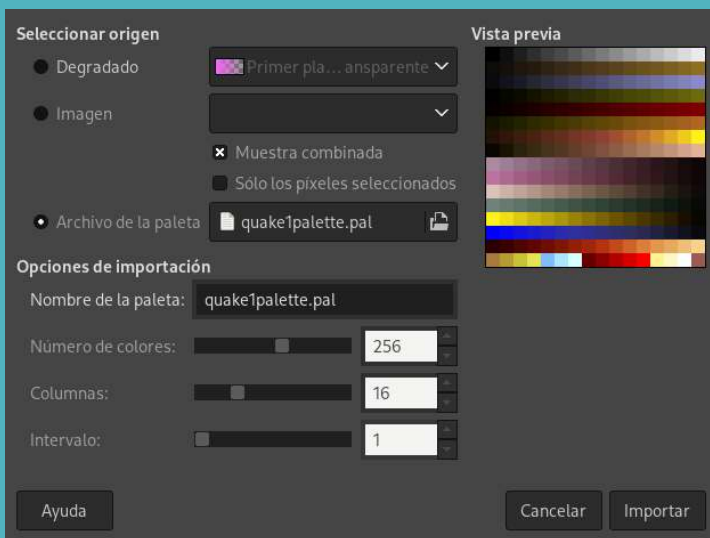
2. **Abrimos GIMP** y nos vamos a: Ventanas → Diálogos empujables → Paletas

Una vez abierta vamos a hacer click derecho para importar una paleta nueva.





Seleccionaremos la paleta llamada: **quake1palette.pal**



Con esto ya tendríamos cargada la paleta de colores y podemos por ejemplo crear un nuevo archivo con 64×64 pixeles de ancho (por ejemplo), pinchar en imagen → modo → indexado y elegir la paleta quake1 :D.

Nos podríamos poner a pintar y guardar por ejemplo en bmp o jpg. El motivo de guardar mejor a bmp es que no estaremos comprimiendo y así se desvirtuará lo menos posible la imagen cuando toque empaquetarla como textura a usar en un archivo WAD.

Motor para ejecutar Quake 1

```
LANGUAGE_INITIALIZATION
COULDN'T LOAD 'LOCALIZATION/LOC_ENGLISH.TXT'
FROM '.'

===== QUAKE INITIALIZED =====

EXECING QUAKE.BC
EXECING DEFAULT.CFG
EXECING CONFIG.CFG
COULDN'T EXEC AUTOEXEC.CFG
3 DEMO(S) IN LOOP
MAP NOMBREMAPA
COULDN'T SPAWN SERVER MAPS/NOMBREMAPA.BSP
1
```

Hemos visto como se puede añadir una paleta de colores con los 256 que usa Quake para las texturas al editor de imágenes GIMP.

Vamos a ver un motor que podemos usar para probar los mapas de quake (Quake 1) que realizaremos.

Un motor que está disponible para GNU+Linux y que tanto sirve para ejecutar el juego original (si disponemos de los datos del juego original o de una demo de este) como para simplemente cargar el mapa que realizaremos nosotros/as/es .

Paso 1: Compilar el motor Quakespasm

Actualmente (29 Noviembre 2021) se puede obtener el código fuente de **quakespasm en su versión 0.94.2** desde aquí:

sourceforge.net/projects/quakespasm/files/Source/quakespasm-0.94.2.tar.gz/download

Una vez descargado vamos a proceder a descomprimir el tar.gz y a **compilar el código fuente**.

```
$ tar xfvz quakespasm-0.94.2.tar.gz
$ cd quakespasm-0.94.2/Quake/
$ bash detect.sh
$ make -j$(nproc)
# make install
$ /usr/local/games/quake
```

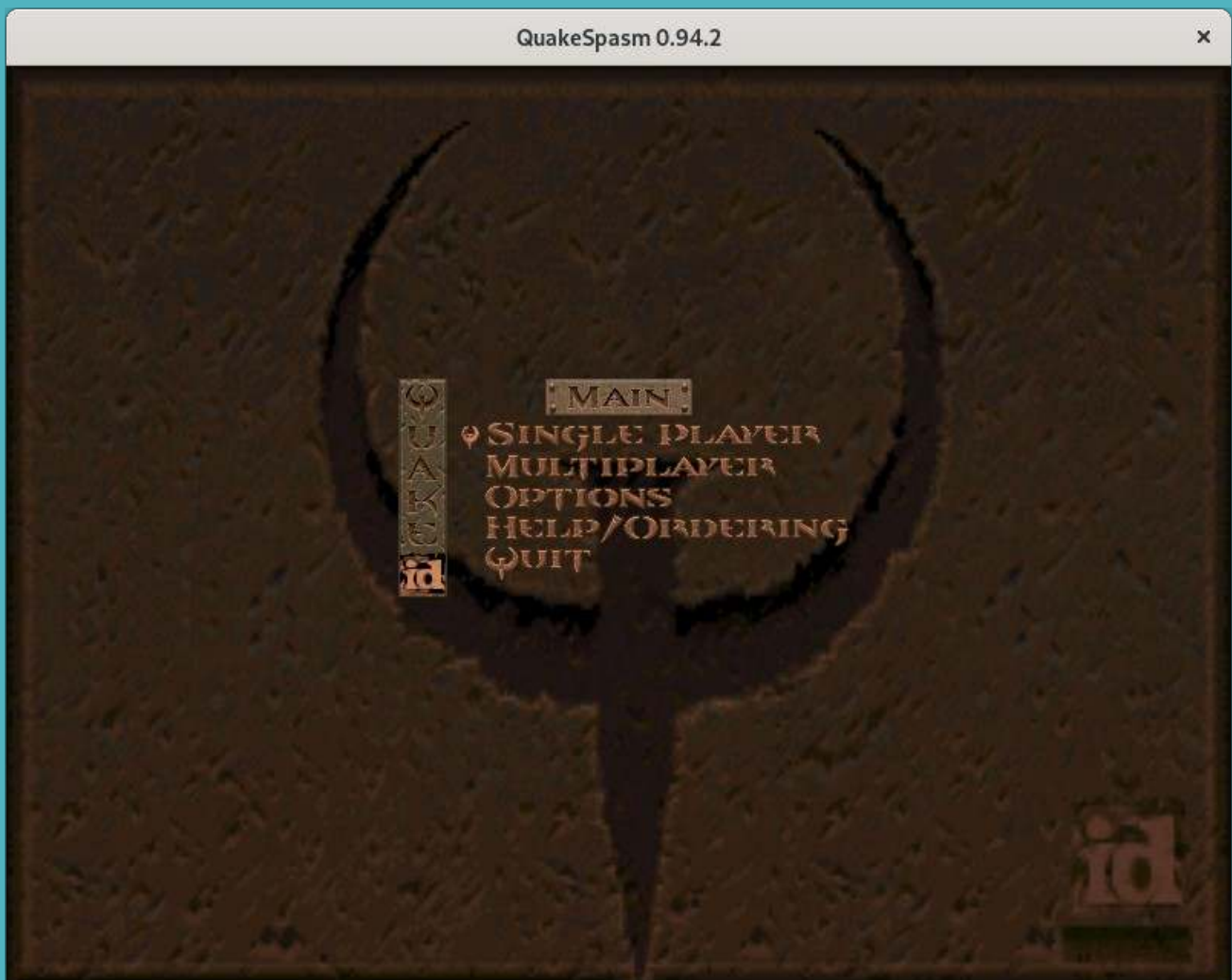
Si todo va bien y disponemos de las dependencias **tendríamos que disponer ya en /usr/local/games/quake de quakespasm (llamado el binario quake a secas)**.

Paso 2: Obtener los archivos pak del juego y crear un lanzador

```
$ wget http://fanta.56k.es/games/1990/1996-Quake/quake1.tar.gz
$ tar xfvz quake1.tar.gz
$ cd quake1
$ echo "/usr/local/games/quake -basedir ." > start.sh
$ chmod +x start.sh
$ bash start.sh
```

Ahora cada vez que queramos lanzar el juego podemos simplemente entrar en el directorio en el que lo tenemos y escribir:

```
$ bash start.sh
```



Disponemos por tanto ya de un engine para ejecutar quake 1. Podemos disfrutar del juego, de una demo o de mapas que hagamos nosotros.

Existen otros motores (darkplaces por ejemplo) pero esta vez **nos vamos a centrar en quakespasm** y es el que vamos a utilizar para testear los mapas que realizaremos.

La forma de cargar mapas será la siguiente.

Con quakepasm ejecutado nos vamos a OPTIONS y luego a GOTO CONSOLE

```
UDP INITIALIZED
EXE: 16:43:04 Nov 29 2021
256.0 MEGABYTE HEAP
VIDEO MODE 800X600X32 60HZ (24-BIT
Z-BUFFER, 0X FSAA) INITIALIZED
GL_VENDOR: INTEL
GL_RENDERER: MESA INTEL(R) IRIS(R)
PLUS GRAPHICS (ICL GT2)
GL_VERSION: 4.6 (COMPATIBILITY
PROFILE) MESA 21.3.0
FOUND: ARB_VERTEX_BUFFER_OBJECT
FOUND: ARB_MULTITEXTURE
GL_MAX_TEXTURE_UNITS: 8
FOUND: ARB_TEXTURE_ENV_COMBINE
FOUND: ARB_TEXTURE_ENV_ADD
FOUND: SDL_GL_SWAP_CONTROL
FOUND: EXT_TEXTURE_FILTER_ANISOTROPIC
FOUND: ARB_TEXTURE_NON_POWER_OF_TWO
FOUND: GLSL
INTEL DISPLAY ADAPTER DETECTED,
ENABLING GL_CLEAR

SOUND INITIALIZATION
SDL AUDIO SPEC : 44100 HZ, 512 SAMPLES, 2 CHANNELS
SDL AUDIO DRIVER: PULSE, 32768 BYTES BUFFER
AUDIO: 16 BIT, STEREO, 44100 HZ
SDL DETECTED 0 CD-ROM DRIVES

LANGUAGE INITIALIZATION
COULDN'T LOAD 'LOCALIZATION/LOC_ENGLISH.TXT'
FROM ' '.

===== QUAKE INITIALIZED =====

EXECING QUAKE.RC
EXECING DEFAULT.CFG
EXECING CONFIG.CFG
COULDN'T EXEC AUTOEXEC.CFG
? DEMO(S) IN LOOP
IMAP NOMBREMAPA
COULDN'T SPAWN SERVER MAPS/NOMBREMAPA.BSP
]
```

El comando a escribir para cargar un mapa y disfrutarlo es: **map nombremapa**

Si por ejemplo el mapa se llama prueba escribiremos: **map prueba**

En la captura se ve que no existe el directorio maps dentro de id1 ni tampoco un archivo de mapa llamado nombremapa.bsp (**bsp será la extensión de los mapas una vez compilados**).

Por tanto para finalizar vamos a crear un directorio llamado maps dentro de id1 y podemos dar por terminada esta entrada. **Ya tenemos el engine listo para probar mapas.**

```
$ mkdir -p id1/maps
```


Probar a cargar mapas



Podemos empezar a probar la carga de mapas ya :).

Los archivos PAK

Los archivos PAK seguramente los conocerás de juegos como Quake, Hexen, Half life, ... Son archivos que llevan en su interior gráficos, objetos, texturas, sonidos, ...

Es posible **ver el contenido de los archivos PAK** en GNU/Linux usando el programa unmass no obstante nos interesa mucho más poder extraer contenido y empaquetar de nuevo esos archivos.

Para extraer contenido de los archivos PAK vamos a utilizar pakextract . Se puede compilar así:

```
$ git clone https://github.com/yquake2/pakextract
$ cd pakextract/
$ make
# cp -pRv pakextract /usr/local/bin/
```

```
[fanta@terminator pakextract]$ pakextract
Extractor for Quake/Quake2 (and compatible) and Daikatana .pak and Sin .sin files
Usage: pakextract [-l] [-dk] [-o output dir] pakfile
-l      don't extract, just list contents
-dk     Daikatana pak format (Quake is default, Sin is detected automatically)
-o      directory to extract to
[fanta@terminator pakextract]$
```

Ya disponemos por tanto de pakextract en el sistema.

Extraer el contenido de los archivos PAK

Utilizando pakextract vamos a desempaquetar y extraer el contenido del archivo pak0 (también se puede hacer del archivo pak1) del juego quake 1.

```
$ cd quake1
$ mkdir pak0-descomprimido
$ pakextract id1/pak0.pak -o pak0-descomprimido/
```

Si nos fijamos se ha desempaquetado y en su interior existe un directorio llamado maps. Vamos a **copiar el contenido de maps al directorio maps dentro de id1.**

```
$ cp -pRv pak0-descomprimido/maps/* id1/maps/
```

```
[fanta@terminator quake1]$ cp -pRv pak0-descomprimido/maps/* id1/maps/
'pak0-descomprimido/maps/b_batt0.bsp' -> 'id1/maps/b_batt0.bsp'
'pak0-descomprimido/maps/b_batt1.bsp' -> 'id1/maps/b_batt1.bsp'
'pak0-descomprimido/maps/b_bh100.bsp' -> 'id1/maps/b_bh100.bsp'
'pak0-descomprimido/maps/b_bh10.bsp' -> 'id1/maps/b_bh10.bsp'
'pak0-descomprimido/maps/b_bh25.bsp' -> 'id1/maps/b_bh25.bsp'
'pak0-descomprimido/maps/b_explob.bsp' -> 'id1/maps/b_explob.bsp'
'pak0-descomprimido/maps/b_nail0.bsp' -> 'id1/maps/b_nail0.bsp'
'pak0-descomprimido/maps/b_nail1.bsp' -> 'id1/maps/b_nail1.bsp'
'pak0-descomprimido/maps/b_rock0.bsp' -> 'id1/maps/b_rock0.bsp'
'pak0-descomprimido/maps/b_rock1.bsp' -> 'id1/maps/b_rock1.bsp'
'pak0-descomprimido/maps/b_shell0.bsp' -> 'id1/maps/b_shell0.bsp'
'pak0-descomprimido/maps/b_shell1.bsp' -> 'id1/maps/b_shell1.bsp'
'pak0-descomprimido/maps/e1m1.bsp' -> 'id1/maps/e1m1.bsp'
'pak0-descomprimido/maps/e1m2.bsp' -> 'id1/maps/e1m2.bsp'
'pak0-descomprimido/maps/e1m3.bsp' -> 'id1/maps/e1m3.bsp'
'pak0-descomprimido/maps/e1m4.bsp' -> 'id1/maps/e1m4.bsp'
'pak0-descomprimido/maps/e1m5.bsp' -> 'id1/maps/e1m5.bsp'
'pak0-descomprimido/maps/e1m6.bsp' -> 'id1/maps/e1m6.bsp'
'pak0-descomprimido/maps/e1m7.bsp' -> 'id1/maps/e1m7.bsp'
'pak0-descomprimido/maps/e1m8.bsp' -> 'id1/maps/e1m8.bsp'
'pak0-descomprimido/maps/start.bsp' -> 'id1/maps/start.bsp'
[fanta@terminator quake1]$
```

Eso copiará unos cuantos mapas que podemos cargar manualmente desde la consola de quake.

En realidad no es necesario desempaquetarlos para cargarlos porque cuando ejecutamos el engine ya lo hace internamente leyendo los archivos pak. No obstante es para que quede claro que cuando queramos cargar los nuestros lo ideal es meterlos en ese directorio.

La forma de cargar mapas

Con quakeaspasm ejecutado nos vamos a OPTIONS y luego a GOTO CONSOLE



Escribimos en este caso: map e1m2 y cargará ese mapa. Si ponemos map e1m1 cargará ese otro.

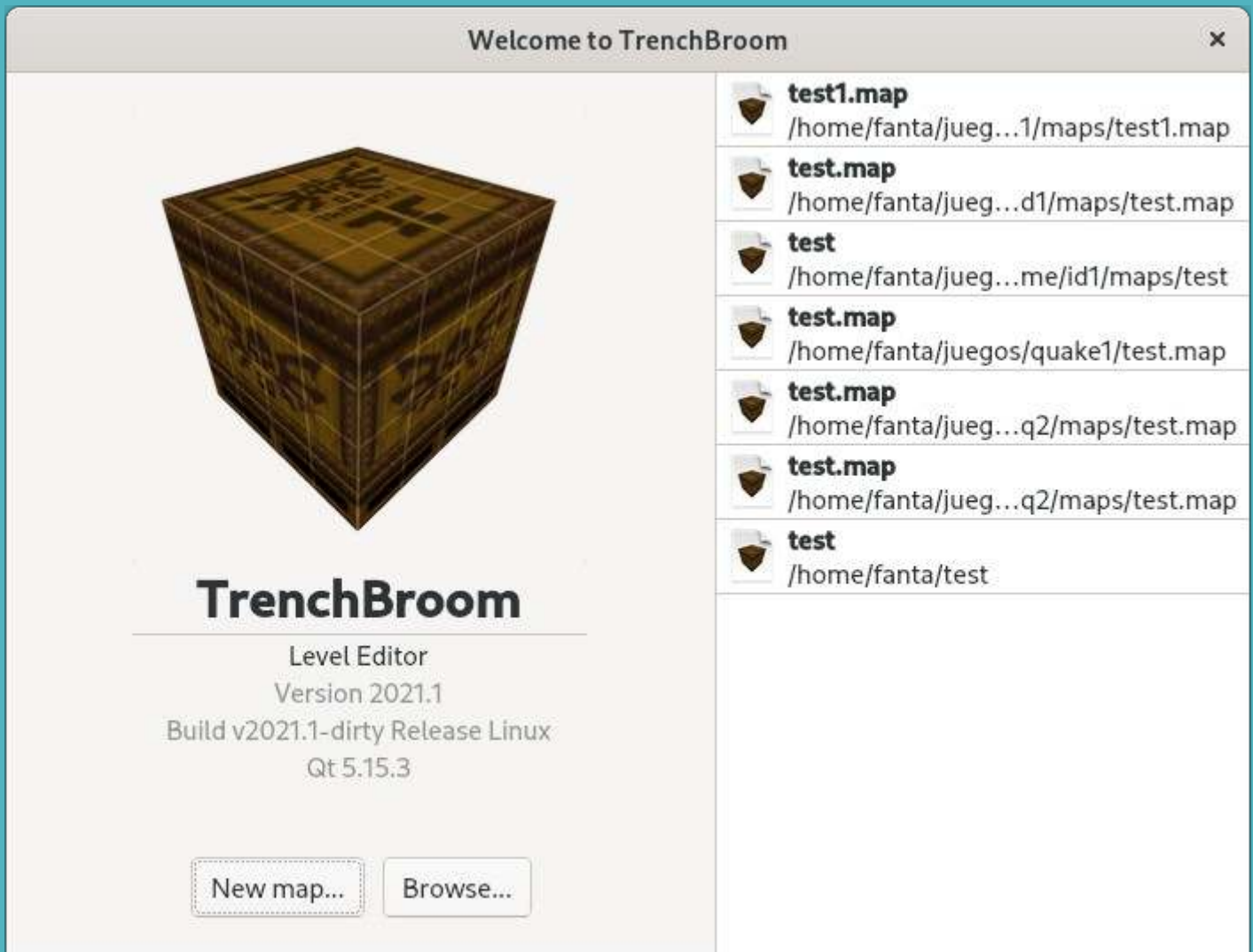
Insisto en que no es vital desempaquetar estos mapas pero si que es recomendable para que dispongamos de la herramienta a mano.

Una vez hemos probado a cargar algunos mapas ya sabemos como va a funcionar esto y es el momento de empezar a crear nuestro primer mapa para Quake 1.

***Nota:** Los archivos *.map serán los que utilizaremos para abrirlos con el editor de mapas mientras que los *.bsp serán los compilados que podemos ejecutar con el motor del juego y probarlos.*

Configurando el editor de mapas

Lo importante ahora es hacernos con el programa: [TrenchBroom](#)

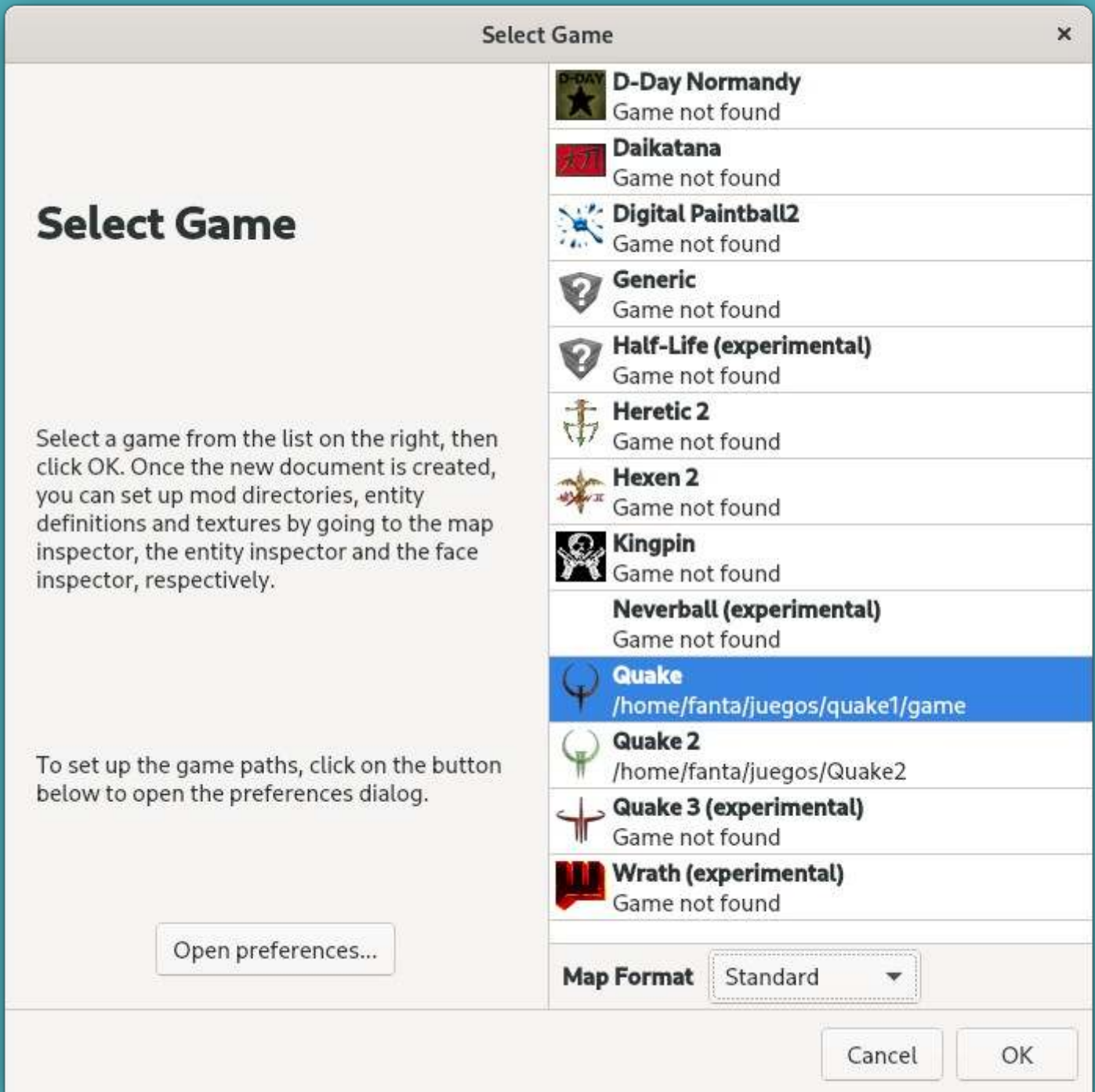


Es posible que esté en los repositorios de tu distro GNU+Linux y si no puede que lo encuentres tanto utilizando flatpak como snap.

Lo importante es disponer del editor en nuestro sistema porque **vamos a empezar a configurarlo para crear nuestro primer mapa de Quake 1.**

Configuración antes de Crear un Nuevo mapa

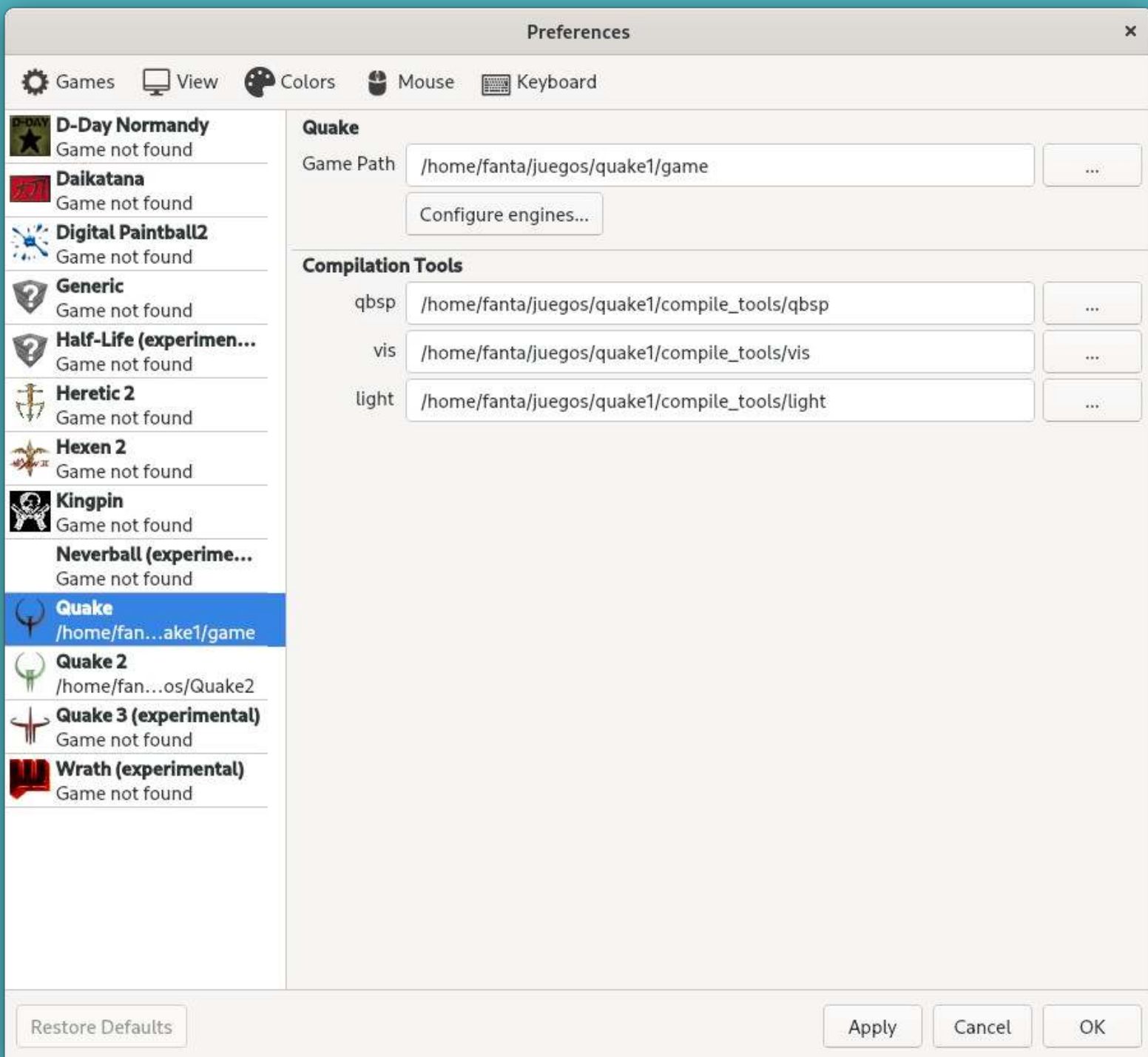
Al lanzarlo veremos la ventana de bienvenida en la que saldrá un botón «New map...» que si presionamos saldrá esto:



Vamos a crear un nuevo mapa de Quake1. Olvidaros de la ruta que sale en la captura puesto que hemos de indicar la ruta en la que están los archivos del juego y el engine.

En realidad vamos a tener que indicar unas cuantas rutas antes de empezar siquiera a crear el mapa.

Si nos fijamos sale un botón a la izquierda llamado «Open preferences ...». Pinchamos sobre las preferencias y se abre una ventana.



Tenemos que indicar donde disponemos de los archivos del juego (**Game path**).

Tenemos que indicar en «**configure engines...**» la ruta en la que tenemos quakespasm (/usr/local/games/quake).

Y finalmente indicar la ruta en la que tenemos los binarios de las herramientas de compilación de mapas.

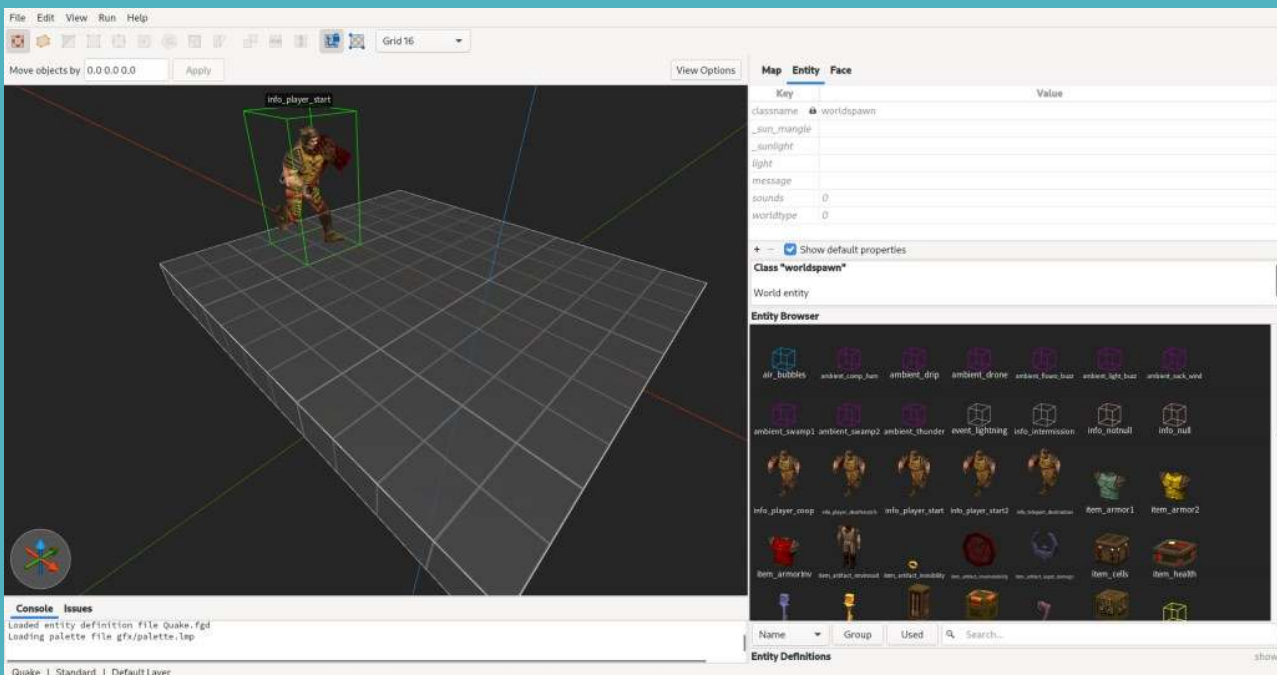
Descargar aquí los binarios de las compile tools: [compile_tools](#)

Recomiendo tenerlo todo bien ordenado. Para ello he creado un pequeño zip que lleva tanto el engine, como las compile_tools así como los pak del juego. Se puede obtener aquí:

http://fanta.56k.es/games/1990/1996-Quake/quake1_crear_mapas.zip

Una vez todo esté bien configurado comenzaremos con la creación de nuestro primer mapa de Quake 1.

Nuestro primer mapa sin nada que hacer



En la pestaña Entity tendríamos que ver entidades que podemos añadir al mapa. La necesaria y vital es añadir «info_player_start».

El suelo que viene por defecto podemos escalarlo para que sea algo más grande. Y luego guardamos el archivo dentro del directorio del juego id1/maps/ como prueba.map

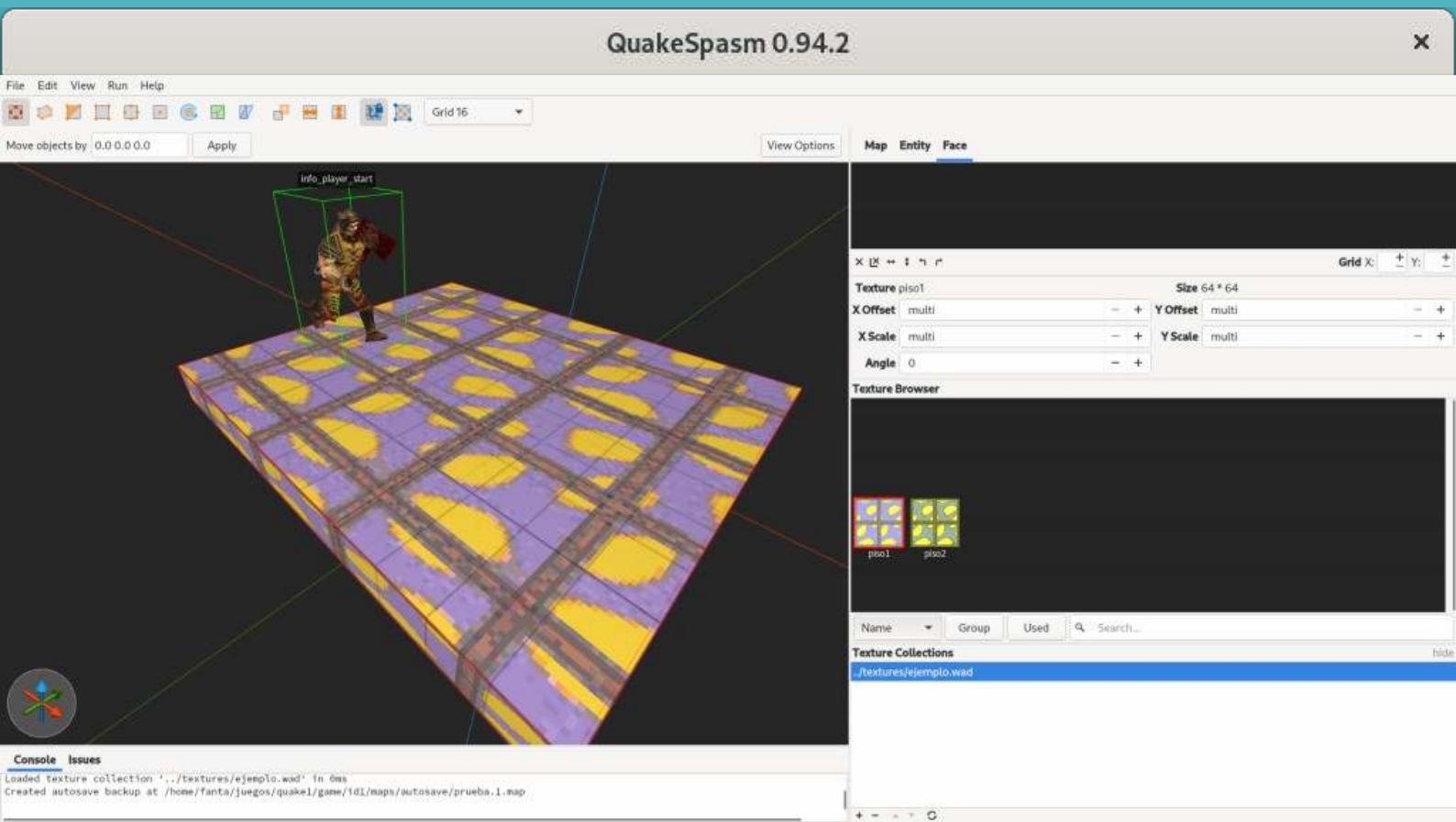
La idea es ahora mismo solamente disponer del player1 y un poco de suelo para probar a compilar el mapa.

El archivo tendrá la extensión .map y por tanto hemos de compilarlo para crear un archivo en el mismo directorio llamado prueba.bsp . Esto se puede hacer así:

```
$ ./qbsp ../game/id1/maps/prueba.map
```

Es decir, vamos a usar qbsp para generar el archivo prueba.bsp dentro de /id1/maps/

Y ejecutamos quake, vamos a la consola y escribimos: **map prueba**



Digamos que ya tenemos nuestro primer mapa sin enemigos y sin texturas.

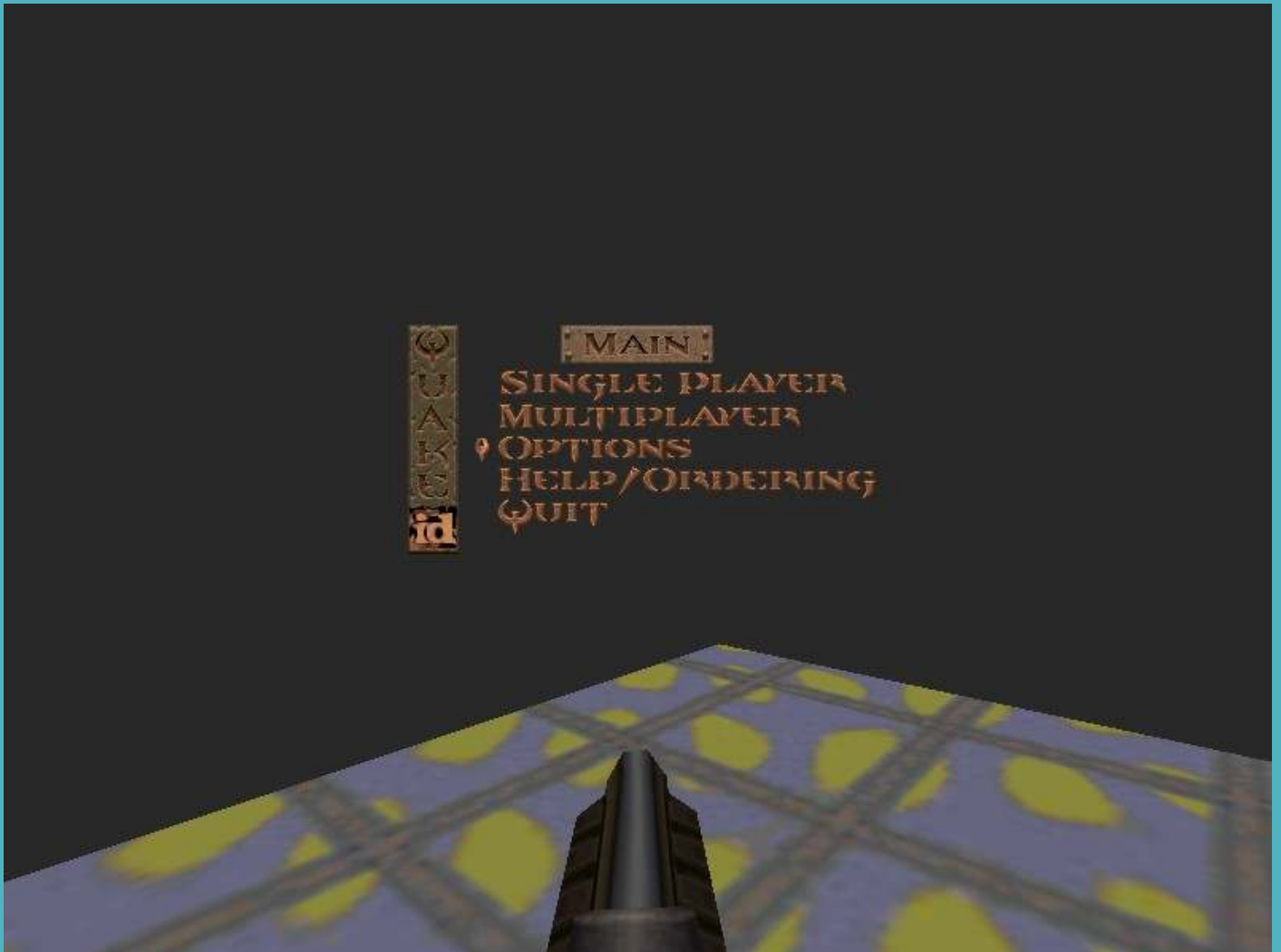
Regresamos al editor y podemos añadir un archivo WAD que contenga texturas.

Esto se hace pinchando en la pestaña «**Face**» (a la derecha de Entity») y luego abajo en **Texture Collections**.

Pinchando en el icono + añadimos el archivo ejemplo.wad

Podemos poner al suelo una de las texturas, guardar y compilar de nuevo el mapa.

Si todo va bien ya tendremos nuestro primer mapa sin sentido pero con texturas y un player que puede moverse (y caerse).



Con -basedir podemos indicar la ruta en la que tenemos el juego.

```
$ ./quakepasm -basedir ../game/
```

Y más o menos eso es todo por el momento.



[TWITCH](#) [PEERTUBE](#) [FEDIVERSE](#) [YOUTUBE](#) [TWITTER](#)