

El Mapeado en QERadiant

por Jorge Pérez Alvarez (jperzalv@nexo.es)
C/ José García Nieto 24-F. 33008 Oviedo (España)

La distribución de este tutorial se realizará incluyendo todos los archivos que pertenecen al mismo, y mencionando el nombre del autor.

En este tutorial no explicaré todo el funcionamiento de QERadiant, ni todos los tipos de entidades existentes, ni cosas por el estilo. Este será un tutorial al estilo puro, aprendiendo poco a poco, cosas determinadas que nos sirvan de ejemplo para otros procesos. Por tanto, si surgiera alguna duda durante la lectura del tutorial, estaré dispuesto a ayudarles. Comencemos.

Indice

Capítulo 1. Primer Mapa QERadiant

- 1.1 Iniciando el Mapa
- 1.2 Nuestro Primer Sólido
- 1.3 La Posición de Inicio
- 1.4 Una Fuente de Luz
- 1.5 La Primera Prueba

Capítulo 2. Añadiendo un Pasillo y una Puerta

- 2.1 Comenzando
- 2.2 Añadiendo una Nueva Habitación
- 2.3 Añadiendo el Pasillo
- 2.4 Habilitando el Pasillo
- 2.5 Construir la Puerta

Capítulo 3. Plataformas

- 3.1 Empezando la Construcción
- 3.2 El Sólido Plataforma
- 3.3 La Entidad Plataforma
- 3.4 Probando Valores

Capítulo 4. Ascensores y Botones

- 4.1 Puertas como Ascensores
- 4.2 Un Botón
- 4.3 Un Segundo Botón
- 4.4 Cambiando Parámetros
- 4.5 Pequeña Lista de Keys

Capítulo 5. Manipulación de Vértices

Capítulo 6. El Corte por un Plano

Capítulo 7. Explotando Objetos

Capítulo 8. Regiones

- 8.1 Funciones de las Regiones
- 8.2 Cargando el Mapa
- 8.3 Región Set XY
- 8.4 Región Set Tall Brush

- 8.5 Región Set Brush
- 8.6 Región Set Selected Brush
- 8.7 Compilando Regiones

Capítulo 9. Consejos Finales

Capítulo 1

Primer Mapa QERadiant

El archivo .map sobre este capítulo se llama capt1.map

Entonces empezaremos, ¿no?

1.1 Iniciando el Mapa

Lo primero que debes hacer es crear un proyecto nuevo. Para ello pulsa en el menú "File" y selecciona New Project. Esto abrirá una ventana para que teclees el nombre del nuevo proyecto. Puedes nombrarlo como quieras.

Después de crear el nuevo proyecto cargará un mapa por defecto que tendrá una habitación y algunas entidades y cosas así en él. Ignoraremos este mapa por ahora y haremos nuestro propio mapa.

Primero crea tu propio mapa pulsando el menú "File" y seleccionando "New Map". Esto mostrará una ventana que dirá "All changes to map will be lost". Pulsa OK.

Ahora debemos hacer un par de cosas antes de crear nuestro primer mapa.

Debemos pulsar en el menú "Textures", tras lo que aparecerá un submenú con una lista de las texturas disponibles para usar. Comúnmente verás E1U1, E1U2, y así. Pulsaremos por ahora sobre E1U1, lo que cargará las texturas del Episodio 1, Unidad 1.

Ahora QERadiant leerá las texturas y las mostrará, dependiendo del tipo de vista que hayas seleccionado en la ventana de Preferencias. Si seleccionaste la QE4 original o la nueva vista de QERadiant entonces las texturas se mostrarán sin tener que usar ninguna tecla. Si seleccionaste el modo de 4 vistas entonces deberás hacer aparecer la ventana de texturas pulsando la letra "T" en tu teclado.

Ahora que las texturas están cargadas ya podemos comenzar a crear nuestro mapa.

Primero busca en las texturas una que te gusta. Para desplazarte por las texturas deberás mantener pulsado el botón "derecho" del ratón y mover el ratón arriba o abajo. Una vez que hayas encontrado una textura que desees debes pulsar el botón "izquierdo" del ratón para que una línea roja rodee a la textura seleccionada.

1.2 Nuestro Primer Sólido

Para hacer esto vete a una vista 2D, preferiblemente la XY. Mantén pulsado el botón izquierdo del ratón y arrastra el ratón. Deberías ver aparecer una caja roja. Ese es el sólido dibujándose. Ahora haz que el sólido mida 256x256 en la vista XY

(superior), y 192 de alto en la vista XZ (lateral). Si necesitas ajustar el factor del zoom sólo pulsa la tecla "Insert" para alejar y la tecla "Supr" para acercar.

Si vas a cualquier vista (XY, XZ, o YZ) y quieres mover la vista del sólido puedes mantener pulsado el botón derecho del ratón en esa vista, y mover el ratón a la vez que mueves la vista, por lo que puedes centrar el objeto en el que estás trabajando.

Ahora acabamos de finalizar la creación de un sólido. Claro está que nosotros no podemos movernos por un sólido compacto, por lo que necesitamos hacerlo hueco. QERadiant tiene una opción "hollow" que convertirá un sólido compacto en un objeto formado por múltiples sólidos con una zona interior hueca, por la que el jugador se puede mover (lo ahuecará). Esto también se conoce como el espacio del jugador.

Por tanto para hacer este sólido hueco, puedes pulsar en el menú "Selection" y bajar hasta "CSG", y entonces seleccionar "Make Hollow", o puedes pulsar en el botón "Hollow" en la barra de herramientas.

Una vez realizadas uno de estos dos procesos tu sólido simple será ahora un cuerpo de 6 sólidos agrupados que forman una habitación hueca que ahora es un rea por la que el jugador se puede desplazar. También notarás que en la vista 3D puedes ahora ver la habitación por el interior.

Estoy seguro de que estas ansioso por jugar en esa habitación por lo que debemos hacer que esto sea posible.

1.3 La Posición de Inicio

Obviamente no podemos jugar sin una posición de inicio del jugador.

Lo que necesitamos hacer es pulsar con el botón izquierdo en la vista 2D para asegurarnos de que el área 2D es la ventana activa. También puedes deseleccionar la habitación, y para ello debes pulsar CTRL+SHIFT+botónIzquierdo del Ratón. Ahora la habitación está deseleccionada.

Ahora, en la vista 2D pulsa el botón "derecho" del ratón. Una pequeña lista debería aparecer. Esta lista es una lista de todas las entidades que tu puedes usar independientemente de los sólidos, y es conocida como "Drop Entity" list. Por ahora sólo queremos una posición de inicio, por lo que mueve el ratón hasta "Info", y aparecerá otro submenú con otra pequeña lista de entidades. Pulsa sobre info_player_start.

Esto debería haber puesto una entidad info_player_start en tu mapa, donde estaba el cursor del ratón. Si la entidad no apareció en el punto deseado entonces pulsa el botón "izquierdo" del ratón mientras mantienes pulsada la tecla "shift" dentro de esta nueva entidad. Ahora la entidad debería estar seleccionada. Suelta el botón y la tecla. Ahora mientras mantienes pulsado el botón "izquierdo" del ratón con el cursor dentro del área de la caja de la entidad que acabas de seleccionar, simplemente mueve el cursor y la entidad lo seguirá. Mueve la entidad a donde desees por ahora. También asegúrate de ir a las vistas XZ o YZ y colocar al jugador sobre el suelo o un poco por encima.

Ahora queremos ajustar el ángulo del jugador (a dónde mira). Con la entidad del jugador todavía seleccionada pulsa la tecla "n", lo que mostrará la lista de entidades y nos dará la información actual de la entidad seleccionada. En este menú cambiaremos la posición actual del jugador. Para hacer esto puedes pulsar sobre la dirección en la esquina inferior izquierda de este menú, que debería tener "135,

90, 45, 0, 315, 270, 225, y 180. Pondremos "90" en este ejemplo, lo cual hará que el jugador se coloque mirando hacia el norte del mapa que estamos construyendo. Una vez que hayas seleccionado la dirección a la que quieres que el jugador mire, únicamente pulsa la tecla "n" otra vez para ocultar esta ventana en la vista actual.

1.4 Una Fuente de Luz

Bien. Ahora... estoy seguro de que queremos un poco de luz para esta habitación. Por tanto deselecciona la entidad `info_player_start` pulsando a la vez `Ctrl+Shift+Botón Izquierdo del Ratón`, y soltándolos después.

Ahora que no hay nada seleccionado vete a la vista 2D y pulsa con el botón "derecho" en algún lugar de la habitación para mostrar de nuevo la lista "drop entity". Esta vez seleccionaremos la primera opción "light" del menú. La que queremos es la única que no tiene una flecha a su derecha. Debería ser sólo "light", sin nada más. Por eso pulsa en "light" y esto debería colocar una nueva entidad en el mapa. Esta entidad debería ser una entidad de color verde con forma de caja. Selecciona la entidad (`Shift+Botón Izquierdo` dentro de la entidad) y muévela a la posición deseada. No olvides comprobar la altura en las vistas XZ o YZ. Para este tutorial la colocaremos cerca del techo y en el centro de la habitación.

Ahora deberíamos ajustar el brillo, y por supuesto el color de esta luz. ¿Por qué?. Simplemente porque podemos, je,je. (El cambio de color sólo es posible si se dispone de una tarjeta aceleradora 3D para jugar a Quake II).

Para ajustar el brillo de esta luz necesitamos seleccionar la entidad luz y pulsar "n" para mostrar la información de la entidad. En esta ventana de información deberíamos ver una caja de edición que dice "key" y debajo de ésta "value". En "key" escribiremos "value" (sí, ya sé que parece raro), y en "value" pondremos el nivel de iluminación en números tales como 300. Entonces simplemente pulsa Intro y esto insertará esa información en las propiedades de la entidad. Yo puse 300 para esta luz.

Ahora cierra esa ventana pulsando de nuevo la tecla "n".

Ahora daremos algo de color a la luz. Esta parte es muy fácil. Todo lo que debes hacer es seleccionar la entidad luz y pulsar la tecla "k". Aparecerá la típica paleta de windows con la opción de ajuste de color. Sólo elige un color básico de uno de los cuadrados para que esté resaltado y pulsa "ok". Entonces la información se añade automáticamente a la información de la entidad. Recuerda también que puedes definir tus propios colores si decides hacerlo.

Ahora ya está todo dispuesto para ver nuestra primera habitación con una luz coloreada y una posición de inicio del jugador.

1.5 La Primera Prueba

Ahora pulsa en "File" y selecciona "Save As" y guarda el mapa con cualquier nombre que te guste, y guárdalo en el directorio "maps" que se encuentra dentro de un directorio que se llama igual que el proyecto que creaste al principio. Por ejemplo: `c:/quake2/nombre/maps`

Pues ahora lo único que falta es compilar el mapa. Pulsa en el menú "Bsp". Selecciona la opción que dice "Bsp_FullVis (grad -extra)". Ahora el nivel se

compilará. Una vez que el nivel esté compilado se cargará automáticamente (si seleccionaste esta opción en la ventana Preferences). Si este no es el caso deberás salir a MS-DOS (desde Windows) y cargar Quake2 así:

```
quake2 +set game nombrepryecto +map nombremapa
```

donde nombrepryecto sería el nombre del directorio-proyecto, y nombremapa sería el nombre del mapa creado.

Con esto finalizamos el primer capítulo de este tutorial, sobre cómo hacer mapas en QERadiant.

Capítulo 2

Añadiendo un pasillo y una puerta

Bien. Ya estamos en el segundo capítulo. En este capítulo vamos a aprender cómo añadir otra habitación y un pasillo para conectar las dos habitaciones. También aprenderemos cómo crear una puerta estándar que se abre cuando caminas cerca de ella y colocarla en el nuevo pasillo.

El archivo .map de este capítulo se llama capt2.map.

2.1 Comenzando

Primero cargaremos QERadiant. Ahora si has escogido en las preferencias que QERadiant cargue el último proyecto usado, entonces no debes preocuparte. Si no, entonces necesitarás ir al menú "File" y seleccionar "Load Project" y entonces localizar tu script del proyecto en tu directorio /quake2/nombrepryecto/scripts , y seleccionar el archivo nombrepryecto.qe4.

Bien. Ahora también necesitamos cargar el mapa que usamos en el capítulo anterior. Para hacer esto sólo pulsa en el menú "File" otra vez, y elige "Load". Entonces localiza tu archivo nombremapa.map en el directorio /quake2/nombrepryecto/maps y selecciónalo.

*NOTA: en el QERadiant versión 038 tu puedes hacer que el programa cargue también el último mapa utilizado, lo que simplifica el proceso en gran medida.

Ahora debes cargar las texturas (ya aprendimos esto en el capítulo anterior). Asegúrate de que cargas las texturas ElU1 otra vez.

2.2 Añadiendo una Nueva Habitación

Ahora que todo está cargado y listo hagamos nuestra nueva habitación. Ve a la vista XY y muévela hasta que veas alguna zona libre donde colocar un nuevo sólido. Haz esto manteniendo pulsado el botón "derecho" del ratón y moviendo el cursor.

Ahora que podemos ver el área deseada, colocaremos un nuevo sólido en el mapa. Para hacer esto necesitas mantener pulsado el botón "izquierdo" del ratón y arrastrar el ratón de manera que aparezca la caja roja. Dale a este sólido un tamaño similar al de la otra habitación (256x256 de ancho y 192 de alto). Ahora hagamos hueca la habitación. Usa el botón "hollow" como en el capítulo anterior.

2.3 Añadiendo el Pasillo

Ahora necesitamos crear el sólido que funcionará como pasillo entre estas dos habitaciones. Deselecciona tu nueva habitación pulsando la tecla "ESC" en tu teclado.

Crea ahora un nuevo sólido entre estas dos habitaciones y ajústalo para que tenga la dimensión aproximada que deseas para el pasillo.

2.4 Habilitando el Pasillo

Lo que debes hacer ahora es deseleccionar el sólido que acabas de crear pulsando la tecla "ESC".

Acto seguido crea otro sólido dentro de este último y hazlo más pequeño, en cuanto a techo, suelo y paredes laterales. También haz que sea más largo, introduciéndose en ambas habitaciones.

Ahora, con el sólido más pequeño seleccionado, vete al menú "Selection" y elige "CSG" y dentro de este nuevo submenú la opción "CSG Subtract".

Ahora con el sólido pequeño todavía seleccionado pulsa la tecla "Retroceso" para eliminar el sólido que empleamos en la resta.

Después de hacer esto deberías ser capaz de ver dentro de la otra habitación desde la vista 3D.

Hemos finalizado la creación de una nueva habitación y un pasillo en nuestro mapa. ¿Por qué no añades una fuente de luz a la otra habitación y compilas el mapa para echarle un vistazo?

2.5 Construir la Puerta

Ahora que has visto tu nueva habitación y el pasillo, hagamos una puerta en él.

Primero encuentra una buena textura para tu puerta y selecciónala. Ahora dibuja un sólido en la vista XY en el pasillo, y asegúrate de construirlo del mismo tamaño que el pasillo en la vista XZ.

La textura que escogiste puede no aparecer como tu deseas y deberás editar los parámetros de la textura. Esto lo analizaremos más adelante.

De momento haremos que este nuevo sólido sea la puerta. Con el sólido-puerta seleccionado pulsa la tecla "n" para mostrar la ventana de información de entidades para este sólido. En ella debería aparecer "worldspawn". Debemos desplazarnos por la lista de entidades y seleccionar la llamada "func_door". Haz doble click sobre "func_door" y las propiedades del sólido deberían cambiar a una "func_door" y eliminar el "worldspawn". Ahora la puerta se abre por defecto con un ángulo de 0 grados. Si quieres cambiar la dirección deberás ir al panel de ángulos de la esquina inferior izquierda (al igual que hacíamos con la posición de inicio del jugador). Aquí también podemos pulsar los botones "Up" y "Down" para que la puerta se abra hacia arriba o abajo, respectivamente.

Otra opción para especificar el ángulo es incluir el valor "angle" en la opción "key" y el ángulo deseado en la opción "value", y pulsar Intro.

Y aquí está. Un nuevo pasillo y puerta para tu mapa. Mencionaremos otras posibilidades con las puertas y los ángulos de apertura de las mismas más adelante.

Capítulo 3

Plataformas

El archivo .map de este capítulo se llama capt3.map.

3.1 Empezando la Construcción

Ahora trataremos las plataformas. Son una parte muy útil en los niveles, ya que nos ofrecen la posibilidad de incluir muchos pisos diferentes en un nivel; pero aumentan considerablemente la ZPV (Zona Potencialmente Visible). La ZPV controla cuántos polígonos se dibujan en cada fotograma del juego. Por esto, debes tener cuidado con la cantidad de nivel que se puede divisar desde cualquier punto (si la ZPV es muy grande, ralentizará mucho el nivel).

Las plataformas son entidades que, cuando se activan (normalmente caminando sobre ellas), se elevan a una nueva altura. Ellas sólo van hacia arriba cuando se activan, y vuelven a la posición inicial después de un cierto tiempo. Por defecto, si un jugador permanece en la plataforma, ésta no bajará (esto se puede modificar, pero lo veremos más tarde). Es importante recordar esto cuando se crea un nivel. Si quieres una plataforma que pueda activarse para ir arriba y abajo, deberás dirigirte al capítulo de Ascensores. Explicaré más cosas por el camino. De momento, inicia QERadiant, y carga el mapa que creamos en los capítulos anteriores. No queremos eliminar el trabajo que hiciste antes, por lo que sálvalo con otro nombre distinto usando la opción "Save As.." del menú "File".

Ahora, en el muro norte (debería ser el muro situado en la parte superior de la vista XY), crearemos un sólido delgado y lo situaremos a medio camino entre el suelo y el techo. Para hacer esto, asegúrate de que no hay sólidos seleccionados, mantén pulsado el botón "izquierdo" del ratón y arrastra el ratón desde la esquina superior izquierda de la habitación (en la vista XY) hasta el lado derecho de la habitación, y un poco hacia la parte inferior de la habitación (en la vista XY, sólo un poco). Necesitamos que el sólido sea delgado. Comprueba la ventana del eje Z y ajusta el tamaño manteniendo pulsado el botón "izquierdo" del ratón FUERA del nuevo sólido (arriba o abajo). Arrastra el ratón arriba o abajo para redimensionar el sólido. Cuando tenga un grosor de uno o dos cuadrados de la rejilla, puedes soltar el botón "izquierdo". Entonces, mantén pulsado el botón "izquierdo" del ratón DENTRO del sólido delgado y muévelo arriba o abajo para que descansa en la coordenada 80 (en la vista del eje Z). Esto debería permitirnos espacio suficiente para que un jugador se coloque encima y debajo del sólido (teniendo en cuenta que un jugador mide 64 unidades de alto y 32 de ancho).

Después, crearemos la entidad plataforma. Es importante anotar que hay dos tipos de entidades: algunas tienen tamaño fijo (luces, monstruos, jugadores, etc.), y el otro tipo no tiene tamaño específico (puertas, plataformas, ascensores, botones, y algunas otras que también caen en este grupo). Las plataformas se encuentran en el grupo de tamaño variable, por lo que se debe crear primero un sólido que defina su tamaño, y la textura a usar. Además, las plataformas siempre se crean en el editor en su posición EXTENDIDA (para que la luz se aplique apropiadamente). Por tanto,

deselecciona todo pulsando CTRL+SHIFT+Botón Izquierdo del Ratón y haz click con el botón "izquierdo" en la vista XY.

3.2 El Sólido Plataforma

Ahora, en la vista XY construye otro sólido de la misma manera que el primero, pero que sea del tamaño suficiente para que quepa en él el jugador (64x64 está bien). Este sólido será el segundo piso. Mantén pulsado el botón "izquierdo" del ratón DENTRO del nuevo sólido y muévelo dentro de la habitación de manera que esté junto al sólido que creamos antes (pero al "sur" de éste). Ahora, en la ventana del eje Z necesitamos redimensionar algo el sólido. Esto es para que la plataforma se eleve hasta el sólido que conforma el segundo piso. Para hacer esto, (en la vista XY), mantén pulsado SHIFT+Botón Derecho del Ratón, y arrastra el comprobador-Z DENTRO del sólido del segundo piso. (Deberían aparecer tres bloques - uno el suelo, otro el techo, y otro en el medio representando al sólido del segundo piso... El Comprobador-Z destaca todos los sólidos sobre los que se encuentra).

Deberías ver un nuevo bloque coloreado (puede que negro) en el eje Z. Este es el sólido del segundo piso. Posiciona el cursor del ratón sobre el nuevo sólido (que será nuestra plataforma), y mantén pulsado el botón "izquierdo" del ratón (en la ventana del eje Z). Arrastra el ratón arriba o abajo hasta que el nuevo sólido esté alineado con la parte de arriba del sólido del segundo piso. Ahora alinearemos la parte de abajo del sólido con el suelo. Haz esto manteniendo pulsado el botón "izquierdo" del ratón cuando el cursor está DEBAJO del sólido "plataforma" (en la ventana del eje Z), y arrastra el ratón arriba o abajo hasta que la parte inferior del sólido "plataforma" descansa sobre el suelo (el bloque negro más inferior en la ventana del eje Z). Es muy importante que este sólido esté completamente dentro de la habitación. Si esto no es así, habrá errores con el nivel cuando se compile, y es posible que no funcione correctamente.

3.3 La Entidad Plataforma

Bien, ahora la única cosa por hacer es que el sólido sea una entidad plataforma... Por tanto, asegúrate de que está seleccionado (y sólo él). Coloca el cursor sobre él en la vista XY y pulsa el botón "derecho" del ratón. La lista de entidades aparecerá. Elige "Func" y se mostrará otro submenú. Aquí selecciona "func_plat" (si no tienes el sólido seleccionado, se creará una plataforma por defecto junto al cursor). Reselecciona nuestra plataforma y llama a la ventana de entidades pulsando la tecla "n" del teclado (en la parte superior de la ventana de entidades, podrías desplazarte arriba o abajo hasta encontrar la entidad "func_plat" y hacer doble-click sobre ella como camino alternativo a usar el botón derecho; pero ¿por qué complicarse la vida?). En la parte media de la ventana de entidades deberías ver el área de Flags y Keys. Los Flags son propiedades o características que puedes activar o desactivar; y las Keys son propiedades a las que puedes asignar más de una configuración. De hecho, en este caso necesitamos definir cómo de rápido se moverá la plataforma. Por defecto es 150. Este valor está bien, pero para nuestro tutorial diremos que queremos una velocidad de 200. En la caja "Key", escribe "speed" (sin las comillas). Entonces, en la caja "value", escribe "200" (sin las comillas). Pulsa Intro, y verás que esta pareja key/value ha sido añadida a la entidad actual (nuestra plataforma). Ahora cierra la ventana de entidades pulsando "n".

3.4 Probando Valores

Todo lo que nos queda por hacer es comprobar el funcionamiento en Quake2. Ve al menú "Bsp" y selecciona "bsp_FullVis (grad -extra)". Esto compilará por completo nuestro nivel. Si has seleccionado "run Q2 after compiling" en las preferencias, Quake 2 se iniciará sólo. Si no, entonces carga Quake 2 así:

```
quake2 +set game nombreproyecto +map nombremapa
```

donde nombreproyecto es el nombre del directorio que tiene el nombre del proyecto, y nombremapa el nombre con el que grabaste el mapa. La manera más simple de hacer esto en Windows 95 es desde el menú Inicio (mediante la opción "Ejecutar...". Esperemos que todo funcione. Deberías poder entrar en el nivel y caminar sobre la plataforma bajada (debería estar sólo un poco sobre el suelo), y ascender.

Observa que no descenderá hasta que salgas de ella. Podemos modificar esto con facilidad. Cuando hayas terminado de probarlo, sal de Quake 2 y vuelve a QERadiant.

Ahora deberías deseleccionar todo pulsando la tecla "ESC". Entonces pulsa Shift+Botón Izquierdo del Ratón sobre la plataforma para seleccionarla. Llama a la ventana de entidades pulsando la tecla "n" del teclado. Verás que hay un Flag llamado "PLAT_LOW_TRIGGER". Lo que esto hace es cambiar el efecto de activación de la plataforma. Normalmente, la plataforma se elevará cuando un jugador pase sobre ella (no es necesario que la toques, prueba saltando por encima de ella en Quake 2 y notarás cómo asciende). Con este Flag activado, la plataforma sólo subirá si alguien la toca (y sólo cuando la plataforma se encuentre en su posición más inferior). Prueba a activar este Flag, cerrar la ventana de entidades y recompilar tu mapa. Observarás que el comportamiento de la plataforma en Quake 2 varía.

Capítulo 4

Ascensores y Botones

El nombre del archivo .map de este capítulo es capt4.map.

4.1 Puertas como Ascensores

Podríamos decir que esta es una parte de un capítulo doble. ¿Por qué?, porque hay tres tipos de objetos que se mueven verticalmente. Plataformas (func_plat, que ya vimos en el capítulo anterior), Ascensores (usando la entidad trigger_elevator), y Puertas que funcionan como Ascensores (empleando la entidad func_door). En este capítulo hablaremos de este último tipo. Aquí lo explicaremos suponiendo que se está utilizando el sistema de ventanas por defecto, con una gran vista XY y el eje Z en una miniventana (si este no es el caso, es fácil realizarlo en cualquiera de los demás sistemas de ventanas). El asunto de los ascensores es bastante complejo, pero es debido al montón de opciones que ID Software permite... Se analizará con más detalle en el siguiente capítulo, con la entidad "trigger_elevator" (pero os advierto de que es difícil).

El uso de puertas como ascensores se remonta ya a los días de Quake 1; por lo que yo sé, todavía es aplicable. La idea básica es crear una puerta (ya vimos en el capítulo dos la entidad "func_door") como una plataforma, y abrirla cuando se activa. Normalmente, una puerta se activa cuando un jugador la toca. Según esto, si un jugador camina sobre ella, se elevará como si fuera una plataforma (si sólo fuera esto, la entidad func_plat sería más cómoda). Pero esto nos abre un abanico de posibilidades como botones para activar, y todos los demás trucos que se pueden

realizar con las puertas, además de poder construir ascensores de estructura compleja. Lo que vamos a hacer ahora será un sólido que se mueva entre dos pisos pulsando botones.

NOTA: Aquí también aprenderemos el funcionamiento básico de los botones.

Para comenzar, abriremos el mapa creado a lo largo de todos los capítulos, y lo grabaremos con otro nombre. Supongo que a estas alturas ya estás familiarizado con el manejo de QERadiant, por lo que no me explicaré tanto en los pasos básicos. Primero, eliminaremos la entidad `func_plat` existente (pero podemos aprovechar el sólido). Por tanto, pulsa SHIFT+Botón Izquierdo sobre la entidad `"func_plat"` para seleccionarla. Entonces vete al menú "Selection", y dentro de éste a la opción "Ungroup Entity". Esto debería convertir la `"func_plat"` en un sólido normal y corriente. Después de esto, haremos que este sólido sea una entidad móvil `"func_door"`. Para hacer esto, asegúrate de que el sólido todavía está seleccionado (debería estarlo), y pulsa el botón "derecho" del ratón sobre él. Cuando el menú aparezca, escoge `"func"` y entonces `"func_door"`. Deberías ver ahora el sólido etiquetado `"func_door"` en la vista superior (XY).

Necesitamos editar las propiedades de esta puerta para que se mueva correctamente. La entidad `"func_door"` funciona al revés que la `"func_plat"` en tanto que la entidad `"func_door"` se debe crear en la posición inferior o cerrada, mientras que la `"func_plat"` se debe crear en su posición elevada. Es importante recordar esto. Una vez que el sólido tenga la altura adecuada, haremos que baje. Así que pulsa "n" para mostrar la ventana de entidades. En el área de selección de ángulos (esquina inferior izquierda), elige "Dn" (down=abajo). Ahora, esto moverá la entidad `"func_door"` hacia abajo, hacia el piso inferior, cuando se active; pero si tú compilas esto ahora no funcionará correctamente. La razón es la siguiente: una entidad `"func_door"` se activa por defecto al tocarla (como una `"func_plat"`), por lo que si vas a Quake 2 y corres sobre el ascensor `"func_door"`, descenderá, pero ahí se quedará. Si te fijas, al permanecer sobre él, lo estás tocando, y en consecuencia estás activando la puerta constantemente para que baje. Para evitar esto, podemos hacer dos cosas. La primera, prueba activando el Flag `"start_open"` en la ventana entidades. Esto invierte el comportamiento de la entidad `"func_door"`. En lugar de "abrirse" hacia abajo, empezará abajo y se "cerrará" (subirá) cuando se active. Pruébalo. Pulsa "n" para cerrar la ventana de entidades. Vete al menú "Bsp" y selecciona la opción `"bsp_FullVis (grad -extra)"`, y como ya explicamos en los capítulos anteriores, deberás iniciar Quake 2 si la opción de autoarranque no está activada. Cuando pruebes el nivel, notarás cómo la puerta permanece activada cuando tu la tocas, y además, ¿notaste cómo se activa ANTES de que la toques? Los chicos de ID no querían que las puertas retrasaran el avance del jugador, por lo que se activan cuando te encuentras a una pequeña distancia de ellas. Pero esto va a cambiar.

4.2 Un Botón

Vuelve a QERadiant, porque vamos a enlazar esta puerta a algunos botones (entidades `"func_button"`). Cuando una `"func_door"` es apuntada por un botón u otro objeto, ya no se activa cuando la tocas. Por tanto, comenzaremos haciendo un botón. Primero haremos un muro en el que montar el botón. No es necesario, pero queda mejor...

Crea un sólido de 1 cuadrado de ancho (de izquierda a derecha en la vista superior XY, con una rejilla de 8 unidades x cuadrado), y tan largo (hacia arriba y hacia abajo de la pantalla en la vista superior XY) como la `"func_door"`. Dale una altura similar al espacio entre la parte superior del sólido que forma el piso inferior y la parte inferior del sólido que conforma el segundo piso. Usa el comprobador-Z si es necesario. Mueve este sólido de manera que se encuentre un cuadrado al "Este" (a

la derecha en la vista superior XY) de la "func_door"), esto es, que haya un cuadrado vacío entre el lado derecho de la "func_door" y el lado izquierdo del nuevo sólido.

Ahora haremos nuestro botón. Crea otro sólido con un ancho de 8 unidades (un cuadrado, a izquierda o derecha en la vista XY), 24 unidades de largo (tres cuadrados, arriba o abajo en la vista XY), y 24 unidades de alto (en el eje Z). Mueve este sólido de manera que se encuentre entre la "func_door" y sólido-muro que acabamos de crear (de ahí que hayamos dejado un cuadrado vacío). No olvides comprobar el eje Z. Un jugador normal tiene un alto de 64 unidades, por lo que coloca el sólido lo bastante cerca del suelo como para que el jugador pueda activarlo. Yo coloqué el sólido-botón de manera que su parte inferior se encontraba en la coordenada 32 del eje Z. Ahora hagamos que este sólido sea un "func_button". Asegúrate de que es el único sólido seleccionado (Shift+Ctrl+Botón Izquierdo en la vista XY, y entonces Shift+Botón Izquierdo sobre el sólido en cualquier vista en la que se vea el sólido. La manera más fácil es seleccionarlo en la vista 3D). Pulsa el botón "derecho" del ratón para desplegar el menú de entidades. Escoge "func" y dentro de ésta "func_button". Vayamos a la ventana de entidades pulsando la tecla "n", ya que necesitamos especificar el ángulo en el que se moverá el botón. Debemos ir a la esquina inferior izquierda y escoger un ángulo adecuado. Si has construido el nivel tal y como hemos dicho en este capítulo, el botón debería llevar un ángulo de "0": o sea, hacia el pequeño muro que está pegado al botón. Cierra la ventana de entidades.

Es hora de "enlazar" el botón a la "func_door". En otros editores esto supone preocuparse de que las keys "target" (en el botón) y "targetname" (en la "func_door") tengan valores iguales (en la ventana entidades, "target" o "targetname" se colocan en la opción "key", y un nombre debe colocarse en la opción "value". Esto todavía se puede utilizar si el método explicado a continuación no funciona correctamente). En QERadiant, podemos usar la opción "connect entities". Esto simplifica enormemente las cosas, y realiza las conexiones por nosotros. Para enlazar el botón a la entidad "func_door", asegúrate de que el botón todavía está seleccionado. El primer objeto seleccionado es siempre el activador o "trigger", mientras que el segundo objeto es siempre el objetivo del activador. Por tanto, selecciona ahora la "func_door" pulsando Shift+Botón Izquierdo sobre ella. Ahora ambas entidades deberían estar seleccionadas, y en el orden apropiado (nada indica el orden, por lo que es cosa tuya recordar cuál seleccionaste primero). Pulsa Ctrl+K (o vete al menú "Selection", opción "Connect entities"), y deberías ver un cambio en la pantalla. Es difícil de ver con entidades pequeñas, pero debería haber una flecha coloreada apuntando desde el botón a la "func_door" (Habría dos flechas, una pequeña indicando la dirección en la que se mueve la "func_door", y otra mucho mayor que muestra la conexión entre el botón y la "func_door"). Antes de ejecutar esto, permitámonos poder decir que nuestro botón es un botón. Para ello, cambiaremos la textura de la superficie del botón. Deselecciona todo, y entonces selecciona sólo tu botón. Vete al menú "Textures" y pulsa sobre "ElU1". Cuando el programa termine de cargar las texturas de ese nivel, colócate en la ventana de texturas y desplázate hacia abajo (mantén pulsado el botón "derecho" del ratón y muévelo hacia arriba o abajo) hasta que encuentres una textura apropiada. Pulsando con el botón "izquierdo" sobre la textura debería cambiar la superficie de tu botón automáticamente. Prueba a ejecutar este mapa (escoge "bsp_FullVis (grad -extra)" en el menú "Bsp"). Advierte el comportamiento de los objetos. Cuando continuemos, añadiremos un segundo botón y cambiaremos un poco las cosas. Este podría ser un buen momento para tomarnos un descanso y relajarnos. Hay MUCHO que aprender sobre los ascensores...

¿Has notado cómo la puerta vuelve sola hacia abajo tras un cierto tiempo, y cómo el botón se recoloca después de unos segundos?. Podemos ajustar la cantidad de tiempo que algo está activado con la key "wait". Introduciendo esto en la opción "key" de

la ventana de entidades, y especificando un número en la opción "value" se cambia el número de segundos que deben pasar hasta que una entidad se pueda reactivar.

4.3 Un Segundo Botón

Y ahora vamos con otro experimento: añadir otro botón. Podríamos simplemente copiarlo seleccionándolo y pulsando la opción "Copy Brush" del menú "Edit", para luego pegarlo con la opción "Paste Brush" del mismo menú, pero vamos a hacer otro manualmente para coger práctica. Haz un sólido con las siguientes dimensiones (puedes mirar en la barra de estado situada en la parte inferior de la pantalla), X:24 Y:24 Z:8.

Este será un botón que se encuentre en el suelo, sobre el segundo piso. Mueve el sólido de manera que descansa sobre el sólido que funciona como segundo piso, y cerca del sitio donde sube la "func_door"; pero no frente a ella (debemos dejar paso al jugador, para que no active el botón cuando salga del ascensor). Debería crearse con la última textura seleccionada, por lo que debería ser la misma que le dimos al primer botón (sólo para que quede bien). Transforma este sólido en un botón seleccionándolo y usando el botón "derecho" del ratón. Escoge de nuevo "func" y "func_button". Usa la tecla "n" para llamar a la ventana de entidades, y elige un ángulo de "Dn". Cierra la ventana de entidades y selecciona de nuevo la "func_door". Deberías tener dos objetos seleccionados, primero el nuevo botón y segundo la "func_door". Podemos conectar otra vez las entidades usando la función Ctrl+K.

NOTA: Varias entidades pueden apuntar al mismo objeto, al igual que una entidad puede apuntar a varios objetos.

Ahora ya casi estamos listos para intentar esto, excepto por una cosa. Cada vez que activamos la "func_door", siempre vuelve a su posición original tras un cierto tiempo. Podemos cambiar esto, por supuesto. Deseleccionalo todo y reselecciona la "func_door". Pulsa "n" para mostrar la ventana de entidades, y mira al área de Flags. Verás uno llamado "toggle". Lo que este flag hace es que la puerta esté "abierta" cuando se activa por primera vez, "cerrada" cuando se activa por segunda vez, y así... Activa este flag. Ahora cierra la ventana de entidades y recompila tu mapa para probarlo (escoge "bsp_FullVis (qrad -extra)" en el menú "Bsp"). Notarás de nuevo cómo el comportamiento cambia. Está un poco facilitado con el botón en el suelo. Hubiera quedado mejor si lo hubiéramos montado sobre otro muro, pero esto sólo tiene propósitos demostrativos.

4.4 Cambiando Parámetros

La última parte a tratar es la equivalente en la "func_door" a la key "height". La key "height" se usa en la entidad "func_plat", y se puede usar para cambiar la cantidad de movimiento de la "func_plat". Desafortunadamente, la "func_door" no tiene esta key. Ambas entidades tienen una key llamada "lip". El valor de "lip" es un número que se resta de la cantidad de movimiento de un botón, plataforma, o puerta. Esto permite que estas entidades sobresalgan siempre un poco (como habréis observado en el juego, las puertas no se abren completamente, ni los botones se pulsan por completo (siempre sobresalen un poquito). La ecuación de esto es [altura del sólido] - [valor lip] = cantidad total de movimiento. Normalmente Q2 da una valor por defecto a "lip", pero podemos cambiarlo positiva o negativamente, y en cantidades muy grandes si es necesario. Entonces volvamos al editor, y te explicaré cómo va. Esto es una idea bastante abstracta, por lo que pon los cinco sentidos en ella. Iremos despacio.

Selecciona la "func_door", y en el eje Z, estrújala hasta que sea delgada (1 cuadrado de alto). Muévela en el eje Z hasta que esté al mismo nivel que el sólido del segundo piso. Ahora, recuerda que una "func_door" sólo se mueve tanto como alta es, por lo que en este caso sólo bajará/subirá 8 unidades, por lo que es bastante difícil que éste baje hasta el piso inferior (puedes comprobarlo si tú quieres). Claro está, podemos alterar esto, usando la key "lip". Primero, echa un vistazo a la distancia existente entre la parte inferior del sólido "func_door" y la parte superior del sólido "piso inferior". Esta es la distancia que necesitamos cambiar. ¿Recuerdas tus clases de matemáticas?. Bien, por que si alguna vez te preguntaste para qué podían servir, aquí está la respuesta.

La ecuación es: [altura del sólido] - [valor lip] = [movimiento total]

Si transformamos esto obtenemos:

[altura sólido] - [movimiento total] = [valor lip]

Por tanto, en nuestro caso comenzamos en el suelo con una coordenada Z de "8", y acabamos en la coordenada Z de "80" (la parte inferior de la "func_door"). La diferencia entre ellas es nuestro valor de [movimiento total]. En este caso, es "72". Insertando esto en nuestra ecuación, obtenemos que:

[altura del sólido] - 72 = [valor lip]

Como sabemos que el sólido tiene 8 unidades de alto:

8 - 72 = [valor lip]

Por tanto: [valor lip]= -64

Ahora que conocemos el valor, abriremos la ventana de entidades (asegúrate de que sólo la "func_door" está seleccionada). En la opción "key", escribe "lip" (sin comillas). En la opción "value", escribe "-64" (de nuevo, sin comillas). Pulsa Intro y deberías ver la pareja key/value añadida a la lista en el centro de la ventana de entidades. Ahora cierra la ventana de entidades y recompila el mapa. Experimenta en el juego y observa cómo funciona esto. Debería funcionar igual que antes, pero con la diferencia de que nuestro sólido "func_door" es mucho más delgado.

Prueba cambiando el valor de "lip" para ver cómo afecta al movimiento de la "func_door". Prueba también colocando una key llamada "speed" con diferentes valores, tanto en la "func_door" como en el "func_button", para ver cómo afecta a su movimiento.

Hay muchas cosas más que puedes hacer, recordando que una "func_door" se mueve entre dos puntos (según un valor en la key "angle"), por lo que no tiene que ir arriba o abajo necesariamente, puede ir de izquierda a derecha (cambiando el valor de la key "angle"), etc. La versatilidad es la madre del juego. Sólo asegúrate de que entiendes cómo cada entidad funciona, y entonces podrás dedicarte a cambiar el comportamiento de la misma.

4.5 Pequeña Lista de Keys

Podemos ver ahora una pequeña lista de las keys empleadas por las "func_door" y los "func_button", y puede que también por otras entidades:

TARGET : debe tener el mismo valor que el Targetname de otra entidad. (uso: target nombre_objetivo).

TARGETNAME : se usa conjuntamente con Target (uso: targetname nombre_objetivo).

(Por ejemplo, un botón con "trgt1" en Target y una puerta con "trgt1" en Targetname. Cuando pulses el botón la puerta se abrirá.)

DMG : significa daño, e indica la cantidad de daño que un jugador (o monstruo) recibirá cuando la puerta, plataforma o entidad le golpee. (uso: dmg #, siendo # un número).

SPEED: Indica la velocidad de movimiento de la puerta, botón, plataforma, etc. (uso: speed #).

WAIT: Indica el tiempo que una entidad esperará antes de volver a su posición original. (uso: wait #).

MESSAGE: Muestra un mensaje al jugador cuando se activa (las puertas sólo dicen el mensaje si funcionan como objetivos de un botón, etc; y se intentan abrir por el método tradicional: tocándolas). (uso: message mensaje).

SOUNDS: para qué explicarlo. La siguiente lista es para puertas, botones, plataformas y worldspawn (uso: sounds #).

Puertas

- 1) ninguno
- 2) ligero
- 3) medio
- 4) fuerte

Botones

- 1) ninguno
- 2) metal
- 3) choque de madera
- 4) choque de metal
- 5) dentro-fuera

Plataformas

- 1) rápido
- 2) lento con cadenas

Worldspawn

Sólo selecciona un número de pista (ej. 10)

La mayoría de estos se muestran en la caja de información de la ventana de entidades.

Capítulo 5

Manipulación de Vértices

Este es un capítulo bastante simple, ya que no hay demasiadas cosas de las que hablar.

Primero crearemos un simple sólido cuadrado. No vamos a construir ni habitaciones ni cosas así, ya que la manipulación de vértices sólo necesita un sólido.

Ahora iremos al modo de manipulación de vértices. Simplemente selecciona el sólido (si no está ya seleccionado) y pulsa la tecla "E". Cuando pulses la tecla "e" deberían aparecer unos puntos en diferentes áreas del sólido que creamos.

Ahora asegúrate de que estás en la vista superior (vista XY). Coloca el cursor del ratón sobre el punto azul que se encuentra en el centro de la parte de arriba. Ahora con el cursor sobre el punto azul, mantén pulsado el botón "izquierdo" del ratón y arrastra el cursor al centro del sólido. Lo que hemos hecho ha sido arrastrar ese vértice del sólido al centro, con lo que creamos un sólido semipicudo.

Parece obvio que ese no es el único método de usar la manipulación de vértices. Ahora pulsa en el punto azul que acabamos de mover (que debería estar en el centro del sólido) y arrástralo otra vez a su posición original, para que tengamos de nuevo un sólido cuadrado.

Ahora que tenemos otra vez el sólido cuadrado, vete a una vista lateral. Ahora mira al punto azul de la parte inferior izquierda. Esta vez arrastra ese punto azul de nuevo al centro del sólido. Ahora deberías haber hecho un triángulo perfecto (visto desde un lateral). Recuerda que si haces este proceso, y acaba siendo un triángulo perfecto, será casi imposible que consigas recolocar el punto azul para que el sólido sea de nuevo un cuadrado.

Y esto es todo. Puedes alterar también los vértices (puntos azules) en la vista 3D, pero esto podría ser muy complicado, además de inexacto, pero es bueno tener esa posibilidad. Ahora lo que puedes hacer es probar con diferentes sólidos de distintos tamaños.

Capítulo 6

El Corte por un Plano

Aquí hablaremos del corte por un plano. Si la manipulación de vértices es una buena herramienta y la usamos mucho, el corte por un plano lo usaremos aún más. Esta es una herramienta versátil y te mostraré para qué sirve, y cómo emplearla.

Lo primero que haremos será crear un nuevo mapa porque estamos de nuevo en un caso en el que lo único que necesitamos será un sólido simple. O sea que una vez que hayas iniciado tu nuevo mapa, crea un sólido medianamente grande para trabajar con él, por ejemplo con un tamaño de 256x256x256. También puedes imaginar que no importa la textura que lleve. No deselecciones el sólido, porque necesitamos que esté seleccionado para cortarlo.

Primero vamos a comenzar con un objeto usado comúnmente y fácil de crear con el corte por un plano. Este objeto será una rampa, o una superficie inclinada. Seleccionaremos el sólido si no lo dejamos seleccionado cuando lo creamos. Para hacer esto pulsa Shift+Botón Izquierdo sobre él. Ahora iremos al modo de corte. Hay 3 formas de hacer esto:

1. Pulsar la tecla "x"
2. Pulsar en el botón de la barra de herramientas que parece tres puntos azules pequeños y líneas formando un triángulo angulado.
3. Ir al menú "Selection" y pulsar en la opción "Toggle clipper" dentro del submenú "Clipper".

Una vez que hayas hecho uno de estos 3 pasos para ir al modo de corte, lleva el cursor del ratón a la parte más superior izquierda del sólido, justo en la esquina, y pulsa el botón. Deberías ver aparecer un punto azul con el número "1" junto a él.

Ahora pulsa en la parte más inferior derecha del sólido, asegurándote de que está exactamente en la esquina. Una vez más aparecerá un punto azul, pero éste tendrá el número "2" junto a él, y estará unido al primero por una línea amarilla, y un par de líneas amarillas bordearán un área del sólido.

Seguidamente cortaremos el sólido. Todo lo que debemos hacer es pulsar la tecla Intro. Lo que esto debería hacer sería cortar el área del sólido que está rodeada por las líneas amarillas.

Ahora puedes hacer lo que quieras con el sólido, pero asegúrate de que has salido del modo de corte pulsando la tecla "esc". Haz experimentos con este tipo de corte porque puedes colocar los puntos azules en el lugar del sólido que quieras para obtener diferentes formas.

Otro punto a favor para el corte es que no tienes que eliminar necesariamente la parte rodeada por líneas amarillas. Puedes también dividir el sólido entre esos dos puntos. Veamos cómo.

Primero eliminemos este sólido cortado pulsando la tecla "Retrosceso". No debería entonces haber ningún otro sólido en el mapa. Ahora crea otro sólido cuadrado de 256x256x256.

Ahora haremos lo mismo que antes: con el sólido seleccionado, vete al modo de corte, coloca un punto azul sobre la esquina superior izquierda y otro sobre la esquina inferior derecha.

Pero esta vez no pulses la tecla Intro. Esta vez, para evitar que el sólido rodeado por líneas amarillas desaparezca, y así quedarnos con 2 sólidos, todo lo que debes hacer es mantener pulsada la tecla "shift" y pulsar "Return" al mismo tiempo. Ahora el sólido debería estar dividido en dos. Ahora tenemos dos sólidos separados. Y ahora ya conoces lo básico sobre el corte por un plano en QERadiant. Muy simple y muy útil. No olvides que puedes obtener un montón de objetos cortando, y ésto era sólo una lección para iniciarte, así que experimenta todas las cosas que puedes hacer con el corte por un plano.

Capítulo 7

Explotando Objetos

Aquí te enseñaré cómo crear un objeto explosionable. Parece difícil pero realmente es muy simple.

Primero crearemos un par de habitaciones conectadas por un pasillo (podemos emplear las habitaciones que creamos en el capítulo 2) con un tamaño de 300x300x300 y colocaremos en ellas la entidad "info_player_start" y unas luces. Después colocaremos en el pasillo un sólido que será la puerta. Con la puerta seleccionada llamaremos a la ventana de entidades pulsando la tecla "n" y buscaremos la entidad "func_explosive", haciendo doble-click sobre ella, para que nuestra puerta sea ahora una entidad del tipo "func_explosive". Ahora, en la ventana de entidades añadiremos ciertas keys y sus correspondientes values:

<u>key</u>	<u>value</u>
mass	800
health	10
dmg	50

Para que la explosión parezca realista colocaremos un par de explosiones adicionales.

Coloca cerca de la puerta una entidad "target_explosion" (pulsando con el botón "derecho" del ratón para mostrar la lista de entidades, y después dirigiéndote a "target", y dentro de ésta a "target_explosion").

Ahora deselecciona la entidad "target_explosion", ya que debemos conectar la puerta y esta entidad. Para ello seleccionaremos la puerta, y acto seguido (sin deseleccionar la puerta) seleccionaremos la entidad target_explosion. Con ambas seleccionadas pulsa Ctrl+K, con lo que enlazaremos las dos entidades, para que cuando la puerta explote, se produzca la otra explosión. Si ahora compilas el mapa, verás cómo la puerta explota cuando se le dispara.

Las explosiones son un efecto interesante, y no tienen que necesariamente producirse cuando el jugador dispara al objeto, sino que se puede enlazar el objeto que explotará a un botón, o a otra circunstancia activadora. Ejercítate cambiando los valores para las keys explicadas antes, para ver lo que resulta.

Capítulo 8

Regiones

El archivo.map de este capítulo se llama capt8.map.

8.1 Funciones de las Regiones

Lo primero, creo que debería explicar qué es una región en QERadiant. Una región es simplemente un área del mapa, o un conjunto de sólidos, sobre los que trabajas sin otros sólidos en medio. Las regiones permiten la omisión de partes del mapa en las que no estás trabajando (omisión, no eliminación), para facilitar la visión y la construcción de objetos.

Las regiones tendrían su equivalencia en los VisGroups de WorldCraft (un mapeador de Quake 1), los cuales permitían la agrupación de objetos que podían ocultarse y mostrarse cuando fuera necesario.

8.2 Cargando el Mapa

Entonces carguemos el mapa de ejemplo llamado capt8.map. Como puedes observar, hay mucho lío en él, con objetos a los que no se puede acceder con comodidad desde las vistas 2D. Aquí es donde entran las regiones.

Hay varios tipos de regiones, y todas ellas tienen diferentes usos, por lo que las iremos viendo de una en una.

8.3 Región Set XY

Este tipo de región toma los sólidos que se encuentran en la vista XY, y crea una región fuera de ellos. Es un mapa bastante pequeño, por lo que acercaremos el zoom (tecla "supr", en la configuración por defecto) para poder ver cómo funciona.

Acercas mucho el zoom, y seleccionas la opción "set XY" del menú "Region". Ahora, si alejas el zoom (tecla "insert") notarás que la rejilla sólo cubre aquella zona sobre la que habías acercado el zoom, y que los sólidos que no se encontraban en esa zona han desaparecido. Esa zona es tu región (volveremos a la rejilla en breves instantes...).

Otra cosa que habrás observado es que, mientras la rejilla es pequeña en la vista XY, se extiende hasta las +/- 4096 unidades en las vistas laterales y frontales. Esto es debido a que has escogido una región "alta". Las regiones altas son las mejores para destacar zonas completas de un mapa, como una habitación que no tiene otros objetos sobre o debajo de ella.

Pulsa en "Off" en el menú "Region" para eliminar la región, y volver a una vista completa del mapa.

8.4 Región Set Tall Brush

Bien. Ahora imaginemos que no quieres hacer zoom, pero que aun así quieres crear una región "alta". Fácil. Crea un sólido que cubra el área que quieres que forme tu región (la altura del sólido no importa, sólo el tamaño XY) y selecciona "Set Tall Brush" en el menú "Region" (asegúrate de que sólo ese sólido está seleccionado). Más o menos como la región "Set XY", pero con algo más de control sobre tu región.

Selecciona "Off" en el menú "Region" para eliminar la región, y volver a una vista completa del mapa.

8.5 Región Set Brush

Supongamos que deseas seleccionar una zona, pero no deseas que tenga una altura ilimitada. Crea un sólido que cubra el área que deseas seleccionar (en las tres vistas, incluidas las laterales), y asegúrate de que es el único sólido seleccionado. Ahora ve al menú "Region" y selecciona "Set Brush".

Esto es útil especialmente cuando quieres trabajar dentro de una habitación sin que te estorben las paredes y el techo, etc.

Pulsa sobre "Off" en el menú "Region" para eliminar la región, y volver a una vista completa del mapa.

8.6 Región Set Selected Brushes

Esta es otra bastante útil, permitiéndote especificar varios sólidos en tu mapa para crear la región. Intentemos usar dos métodos de regiones a la vez.

Primero crearemos una región "alta" que no incluya a los muros. Ahora que los muros están fuera de nuestro camino, cambiaremos a una de las vistas laterales, y seleccionaremos dos de las luces. A continuación, escoge la opción "Set selected brushes" del menú "Región". Y ya está, dos luces, nada más.

Selecciona "Off" en el menú "Region" para eliminar la región, y volver a una vista completa del mapa.

Una forma de emplear este tipo de región es la selección de todos los objetos de una habitación. Para esto, creamos primero un sólido que ocupe todo el espacio de la habitación, sin incluir los muros, techo o suelo, y lo dejamos seleccionado. Ahora vamos al menú "Selection", y escogemos la opción "Select Inside" que se encuentra dentro del submenú "Selected". Esto seleccionará todos los objetos de la habitación, y ahora sólo nos queda ir al menú "Region" y pulsar sobre "Set Selected". De esta forma ya tenemos una región exclusivamente de la habitación.

8.7 Compilando Regiones

Una anotación rápida. Si compilas una región, QERadiant salvará los sólidos de la región en un archivo *.reg (tiene sentido, ¿no?) y compilará sólo los sólidos y entidades de esa región. De esta manera podemos ahorrar el tiempo de compilar el mapa completo cada vez que queremos echar un vistazo a algo.

Hay, claro está, algunas cosas que deberías saber:

-No hace falta que haya una entidad `info_player_start` en la región. QERadiant usará la colocación de la cámara, por lo que si la cámara se encuentra fuera de la región, dará errores (leak).

-La compilación de regiones se calculan sobre las bases de un sólido alto. Lo que esto significa es que si no tienes techo o suelo en la región, te dará error (leak). También significa que si la región llega hasta fuera de los muros, y el suelo del otro lado del muro no está incluido en la región, te dará error (leak).

-Básicamente, cuando compilas regiones, debes tener mucho cuidado con dónde están definidas tus regiones, dónde está colocada la cámara, y qué sólidos están incluidos.

Una vez que hayas entendido las regiones, te ahorrarán muchas horas de tiempo de compilación.

Capítulo 9 Consejos Finales

Puede que no hagan falta, pero los daré de todas formas:

- Preocúpate de que los objetos no se toquen si no es necesario. Si dos objetos están ocupando el mismo espacio, el tiempo de compilación aumenta, ya que el compilador divide esos objetos en varios objetos, de manera que no existan objetos que se toquen. Además, en Quake no se permiten un número ilimitado de objetos, por lo que esto podría ser un problema.

- Sería recomendable que consigieras librerías con Prefabs, para ahorrarte mucho esfuerzo.

- Si el compilador te da como error la palabra --leaked-- significa que el mapa tiene algun punto "abierto", esto es, que en alguna zona del mapa, el mapa se conecta con el espacio vacío.
- No hay mejor aprendizaje que el análisis de otros mapas, para ver efectos, etc.
- Si tienes cualquier tipo de dudas escíbeme (dirección al principio del tutorial).

Y nada más, que el mapeador te acompañe.

Jorge Pérez Alvarez (jperezalv@nexo.es)