

Jerome Grasydyke sobre los mods

Publicado por CeltíberoRamírol - 19 Mar 2009 23:51

Esta es una traducción más o menos acertada del mensaje de Jerome Grasydyke, miembro de CA, sobre la construcción de mods: <http://www.twcenter.net/forums/showthread.php?t=239487>

Ya que la comunidad de modders parece moverse hacia delante a toda velocidad y ya se dispone de algunos mods básicos lanzados, he pensado en explicar la manera en que se supone que debe funcionar el sistema de mods del Empire. Esto debería permitir trabajar con él, y no contra él, y de esta forma evitar “romper” en el futuro lo que CA ha lanzado.

En primer lugar, el data ETW está pensado para permanecer empaquetado. El sistema de ficheros pack y el sistema de ficheros virtual que construye internamente durante el juego son centrales en los sistemas de parches y mods. Por eso, por favor, no pidáis a los usuarios desempaquetar sus propios data para usar los mods.

Entonces ¿cómo moddear el juego? Bien, esto funciona así. Empire carga todos los ficheros de tipo pack en el directorio data en un orden determinado por el tipo de pack. Hay tres tipos fundamentales, lanzamiento, parche y mod. El quinto byte en la cabecera del fichero pack es un identificador numérico que le dice al juego qué tipo de pack es, lanzamiento son 0 o 1, parche son 2 y mod son 3. El juego carga los packs por ese orden: lanzamiento, parches y mods.

Al comienzo del juego, las entradas de ficheros de todos estos packs son montados en un sistema de ficheros virtual, y si el juego encuentra una entrada en un pack posterior que tiene el mismo nombre y localización que una entrada ya existente, la nueva reemplaza la vieja. Esto significa que un pack de mod puede reemplazar cualquier fichero en el sistema de ficheros virtual y añadir nuevas también, sin preguntar a la persona que cargó el mod por sobrescribir los ficheros del disco ni destruir datos.

Entonces, básicamente se pretende que un mod sea un solo fichero “mod pack”, y que instalar mods sea solo colocar ese pack dentro del directorio data del juego, y que desinstalarlo sea eliminarlo de esa ubicación. Hay un poco más que eso, por ejemplo un mecanismo que controla el orden de carga del mod pack, pero...

Ahora, una nota de cómo trabajar con las bases de datos. Habréis notado que cada tabla en las bases de datos (carpeta db) es a la vez una carpeta y un fichero binario. Cuando el juego toma las bases de datos, lo que ocurre cuando el sistema de ficheros virtual es construido, es que busca cada directorio y coloca todos los ficheros binarios que encuentra. Esto significa que un mod pack puede añadir un nuevo fragmento binario dentro de la carpeta db/unit_stats_land_table que añade varias unidades nuevas, y el juego las añadirá a su propia base de datos de unidades terrestres. Los fragmentos son procesados en

el mismo orden que los ficheros pack, y cualquier nueva entrada de base de datos con el mismo nombre que una vieja producirá una sustitución. Esto significa que se puede reemplazar cualquier entrada en una base de datos sin tocar el resto de los datos o modificar los ficheros suministrados por CA en el disco.

Por ejemplo, digamos que los packs CA contienen una entrada para la unidad "french_dragoons", una entre cientos dentro de los datos incluidos en unit_stats_land_table, usada para construir la tabla de unidades del juego "vanilla". Ahora digamos que se crea un mod pack que coloca un nuevo fragmento de datos, digamos que se llama my_mod_unit_stats_land, dentro de db/unit_stats_land_table, que contiene sólo una entrada de datos para "french_dragoons". Lo que ocurrirá es que la nueva entrada sustituirá a la actual para esa unidad del juego, pero el resto quedará como en el original.

Ahora espero que tendréis una buena idea de cómo el sistema de mods ha sido diseñado, y cómo se deberían construir herramientas para trabajar con él. Hay también otra interesante consecuencia de cómo trabaja el sistema, lo que abre la puerta para la construcción de mods sobre mods.

Una cosa más a tener en cuenta es que no hemos terminado todavía de probar el soporte de mods, eso es parte de lo que está pasando aquí antes de que se distribuya el paquete de desarrollo de mods. También somos conscientes de que hay actualmente alrededor de 6000 ficheros en un solo pack. Y deberíais saber que hay algunos requerimientos asociados con la construcción de un pack, y un directorio data lleno con ellos hará que el tiempo de carga del juego se alargue considerablemente.

La otra cosa que debería mencionar, ya que se está ya experimentando con ella, es que el juego comprueba la integridad de la base de datos final tras cargar los mods. Por tanto si se incluye un pack con una entrada a una nueva unidad, que no incluye un modelo o que es invalidada de alguna otra manera, el juego no cargará el mod pack (aunque mantenga el resto). Este no es el mecanismo principal de detección de errores en el flujo de datos de ETW, eso se hace en otro momento, pero los modders debíais saberlo. Desafortunadamente en la actualidad no se genera ningún log con las razones por las que un mod falla al cargarse, y estamos intentando añadir esta función cuando se lance el kit de desarrollo de mods.

Armados con esta información podéis al menos ser capaces de trabajar con el sistema tal y como se concibió, sin tener que esperar al lanzamiento del paquete de desarrollo de mods. La herramienta de inspección de pack y bases de datos es un gran comienzo, y creo que augura un buen futuro para la escena de modding de Empire.

Por tanto, mi recomendación de amigo... NO INSTALÉIS NINGÚN MOD QUE EXIJA SUSTITUCIÓN DE ARCHIVOS. Eso podría impedir que se os instale bien el próximo parche y os deje el juego "capado". Sólo aquellos que incluyan un pack nuevo, que no sustituya a ninguno original, son seguros a este respecto.

Seguiremos informando. :D

=====