

Anexo 1

Bencoding

Bencoding sigue el siguiente formato de datos.

- Los “strings” se guardan con un prefijo que indica su longitud (en base decimal) seguido de dos puntos. Por ejemplo '4:spam' corresponde a 'spam'.
- Los números enteros son representados por una 'i' seguida por el número en base 10 y una 'e' indicará el final. Por ejemplo 'i3e' corresponde a 3 y 'i-3e' corresponde a -3. Los números enteros no tienen ninguna limitación del tamaño. No se admiten números con ceros a la izquierda y la única manera de escribir el cero es 'i0e'.
- Las listas se codifican como 'l' seguida por sus elementos (también bencoded) finalizados con una 'e'. Por ejemplo 'l4:spam4:eggse' corresponde a ['spam', 'eggs'].
- Los diccionarios se codifican como 'd' seguida por una lista de palabras clave y de sus valores correspondientes finalizados por una 'e'. Por ejemplo, 'd3:cow3:moo4:spam4:eggse' corresponde a {'cow':'moo', 'spam':'eggs'} y 'd4:spaml1:a1:bee' corresponde a {'spam': ['a', 'b']}.

Los archivos de Metainfo son diccionarios bencoded con las siguientes palabras clave:

'announce'

La URL del tracker.

'info'

Otro diccionario, este tiene las siguientes palabras clave:

'name'

Contiene el nombre del archivo.

'piece length'

Número de bytes de cada trozo en que esta partido el archivo. Para hacer la transferencia los archivos se parten en trozos de longitud fija exceptuando el último que puede ser más corto, aunque puede ser truncado. La longitud de un trozo es siempre potencia de 2, normalmente 2^{20} .

'pieces'

Es un “string” de longitud múltiplo de 20. Esta subdividido en “strings” de longitud 20. Cada uno es el sha1 hash del trozo en el índice correspondiente.

'length' y 'files'

Siempre tiene que haber uno de los dos, pero nunca los dos a la vez. Si esta 'length' la transferencia será de un solo archivo, si no se refiere a un conjunto de archivos en un directorio.

En el caso de un solo archivo, 'length' indica la longitud del archivo en octetos.

El caso de varios archivos es tratado como si solo hubiera un archivo, concatenando los archivos en el orden en que aparecen en la lista de archivos. La lista de archivos es el valor 'files' y es una lista de diccionarios que contiene las siguientes palabras clave.

'length'

La longitud del archivo en bytes.

'path'

Una lista de "strings" que corresponden a los nombres de los subdirectorios. El último de ellos es el nombre de archivo actual ('length=0 es un error) .

Cuando los clientes se conectan al tracker (servidor principal) le hacen las peticiones de la siguiente manera:

'info_hash'

Identifica la descarga.

'peer_id'

Identifica al cliente.

'ip'

Un parámetro opcional es dar la IP (o nombre dns).

'port'

El número de puerto por el que se esta escuchando. Normalmente el cliente intenta escuchar por el puerto 6881, sino por el 6882 y así hasta el 6889.

'uploaded'

La cantidad total uploaded hasta ahora, codificado en ASCII y base diez

'downloaded'

La cantidad total descargada hasta ahora, codificado en ASCII y base diez.

'left'

El número de octetos que le quedan por descargar a este cliente, codificado en ASCII y base diez.

'event'

Es un parámetro opcional que indica 'started', 'completed', or 'stopped'.

'started'

Se envía cuando se comienza una descarga.

'completed'

Se envía cuando se termina una descarga.

'stopped'

Se envía cuando el cliente no puede descargar el archivo.

Anexo 2

Mensajes

Hay dos tipos principales de mensajes:

- **Mensajes “keepalives”**: son los mensajes para comprobar que la comunicación sigue viva. Su longitud es cero y son ignorados. Estos mensajes con enviados, generalmente, una vez cada dos minutos (pero puede cambiar de tiempo dependiendo de los datos que se envíen).
- **Mensajes “non-keepalives”**: son los que contienen información y permiten realizar funciones. Su primer byte identifica los tipos de mensajes que hay:
 - 0 – choke
 - 1 – unchoke
 - 2 – interested
 - 3 – not interested
 - 4 – have
 - 5 – bitfield
 - 6 – request
 - 7 – piece
 - 8 – cancel

Los mensajes *choke*, *unchoke*, *interested* y *not interested* no tienen payload.

Los mensajes *choke* y *unchoke* son los que permiten la transmisión de información. *Choke* es una notificación que indica que ningún dato se enviará hasta que no ocurra un *unchoke*.

Los mensajes *interested* y *not interested* indican el interés del cliente para bajar un archivo. Para poder bajar cualquier archivo, el estado de *interested* ha de estar activo durante todo el tiempo (enviando mensajes *interested*).

Para que tenga lugar una transferencia de datos, en un lado ha de estar *interested* y en el otro lado de la comunicación ha de estar *unchoke*.

El payload del mensaje *have* es sólo un número, un índice que indica que el cliente ha acabado y que se ha de limpiar su ‘hash’ (que identifica su descarga).

El mensaje *bitfield* se envía siempre (y sólo) como el primer mensaje. Su payload es un campo de bits con cada índice que el cliente ha enviado puesto a uno, y el resto a cero. Los clientes que no tienen ninguna cosa, todavía, que enviar se pueden saltar el mensaje de *bitfield*. El primer byte del *bitfield* corresponde a los índices del 0 al 7, del bit alto al bit menor. Los siguientes son del 8 al 15 y así sucesivamente. Los bits del final se fijan a cero.

Los mensajes *request* contienen un índice, el principio y la longitud; éstos dos últimos son bytes offsets. Generalmente, la longitud es una potencia de 2 a menos que sea final de fichero que será truncada. Todas las implementaciones actuales usan 2^{15} , y el fin de conexión los cuales requieren una cantidad mayor que 2^{17} .

Los mensajes *piece* contienen un índice, el comienzo y el pedazo. Observar que estos mensajes están correlacionados con los mensajes *request* implícitamente. Es posible que un pedazo llegue inesperadamente si los mensajes *choke* y *unchoke* son enviados en sucesión rápida, y/o la transferencia va muy lentamente.

Los mensajes *cancel* tienen el mismo payload que los mensajes *request*. Generalmente, son enviados solamente hacia el final de una transferencia, durante el periodo que se llama 'endgame mode' (final del juego). Cuando una transferencia está casi acabada, hay una tendencia para los últimos pedazos para ser bajados de una sola línea regulada por módem, tomando un tiempo muy largo. Para realizar rápidamente la bajada de los últimos pedazos, la solución es enviar mensajes *cancel* a todos los que comparten el archivo con nosotros si paran de llegar pedazos o llegan muy lentos.