

# Reading Report: Cohen03

**Eric Casanovas**

May 5, 2019

Upload your report in PDF format.

Use this LaTeX template to format the report, keeping the proposed headers.

The length of the report must not exceed **5 pages**.

## 1 Summary

Aquest document es un paper o un extracte d'un llibre escrit per Bram Cohen el 22 de maig del 2003.

### 1. What BitTorrent Does

Quan un fitxer està disponible fent servir HTTP tot el cost recau sobre el host. En canvi amb BitTorrent, quan algú descarrega quelcom, també pugen parts d'aquest fitxer, fet que fa que es redistribueixi el cost de pujada. Es complica fer-ho a gran escala degut a que provocaria overheads molt grans, només el calcular quines parts van a quins peers ja es molt costós. També trobem problemes com que els peers estan connectats durant poques hores o minuts, i hem d'aconseguir que la suma de pujades dels peers sigui igual a la de baixada, cosa que intentarem aconseguir limitant la baixada en funció de la pujada aportada a la xarxa.

#### 1.1 BitTorrent Interface

Es tan fàcil com clicar un link i guardar el fitxer a un lloc del sistema i bittorrent ja descarregarà i anirà pujant parts del fitxer als altres peers.

#### 1.2 Deployment

El fet de fer servir bitTorrent ho decideix qui té el fitxer. Freqüentment els que el descarreguen tanquen l'aplicació un cop finalitzada la descarrega i això no és una forma política de fer servir BT.

## 2. Technical Framework

2.1 Publishing Content Cal publicar un fitxer .torrent amb informació del fitxer (mida, hash, nom i URL on es troba el fitxer). Trackers ajuden als downloaders a trobar-se i aquests es comuniquen via HTTP per demanar informació sobre la seva descarrega. Els requisits de l'amplada de banda del rastrejador i del servidor web són molt baixos, mentre que la seed ha d'enviar almenys una còpia completa del fitxer original.

- 2.2 Peer Distribution Tots els problemes logístics sobre el fitxer han de ser arreglats pels peers, ja que els trackers l'únic que fan es comunicar els peers. Però els trackers són la única forma de que els peers trobin altres peers. Per saber que té cada peer BT talla els fitxers en paquets de mida definida (típicament 256KB) i per verificar la integritat del paquet fa servir SHA-1. Els peers continuament descarreguen parts dels peers que poden, però a vegades no poden continuar descarregant perquè els peers que te a l'abast no tenen el que necessita.
- 2.3 Pipelining Mentre transferim per TCP es important tenir varis requests pendents per evitar un delay entre peces. BT facilita això pertint els paquets en subpaquets, normalment de 16KB, i cada vegada que arriba una peça se'n demana la següent.
- 2.4 Piece Selection Es important seleccionar bé les peces per tenir una bona performance. 4 exemples d'algoritmes:
- Strict Priority Si es demana un subpaquet, es demanaran els subpaquets que faltin del mateix paquet abans que els d'un altre paquet.
  - Rarest First Selecciona el paquet que menys peers tenen primer. Això ens permet evitar el risc de que una part del fitxer deixi d'estar available pels peers.
  - Random First Piece Selecciona el primer paquet de forma aleatòria i després fer servir el Rarest First. Això permet que la primera peça sigui més ràpida d'obtenir i pugui començar la pujada abans.
  - Endgame Mode Pot donar-se que una peça tingui un transfer rate força baix i això no es un problema al mig del fitxer però sí al final. Per tant el que fem es demanar a tots els peers els subpaquets que falten al final i descartar els subpaquets que triguin més a arribar (perquè arribaran repetits) així evitem un llarg temps al final de la descarrega.

### 3. Choking Algorithms

Choking es una aturada temporal de la pujada tot i que pot donar-se una baixada. Choking Algorithm no es un protocol de xarxa de BT però sí necessari per una bona performance.

- 3.1 Pareto Efficiency Algoritme d'optimització local que intenta millorar la pujada/baixada entre dos parells. Per exemple 2 parells tenen mala pujada/baixada entre ells buscar-se uns altres per millorar la seva pujada/baixada.
- 3.2 BitTorrent's Choking Algorithm Fa que cada 10 segons es calculi el download rate i s'assignin peers per evitar una pèrdua de recursos. 10 segons es suficient per a que TCp pugui augmentar el seu temps de transferència.
- 3.3 Optimistic Unchoking Pujar als parells que ofereixen el millor download rate no tindriem cap mètode per descobrir si les connexions actuals no utilitzades són millors que les que s'utilitzen. BT fa servir un parell que no conta pel download rate i així saber si s'ha de canviar de parells o no.

- 3.4 Anti-snubbing De vegades un peer serà "ofegat" per tots els seus peers cosa que farà que disminueixi la seva download rate fins que faci efecte l'Optimistic Unchoking. Si no es rep cap paquet d'un peer durant un minut tampoc se li enviara cap paquet fins que l'Optimistic Unchoking faci la seva feina.
- 3.5 Upload Only Quan s'ha finalitzat la descarrega ja no ens calen els download rates, llavors únicament es centra en fer upload als peers que ho necessitin preferint als peers que ningú està fent upload.

#### 4. Real World Experience

BitTorrent no només ja està implementat, sinó que és àmpliament desplegat. Els desplegaments més grans coneguts han tingut més de mil descarregadors simultanis. El coll de botella d'escala actual (que realment no s'ha assolit) sembla ser l'amplada d'amplada de banda del seguidor. Actualment, es tracta d'una mil·lèsima part de l'ample de banda que s'utilitza, i algunes extensions de protocol menors el faran fins a deu mil·lèsima

## 2 Assessment

En la meua opinió el text ha sigut de fàcil lectura i fàcil de comprendre, a més a més es un text molt interessant i s'hauria de seguir ficant els següents cursos.