

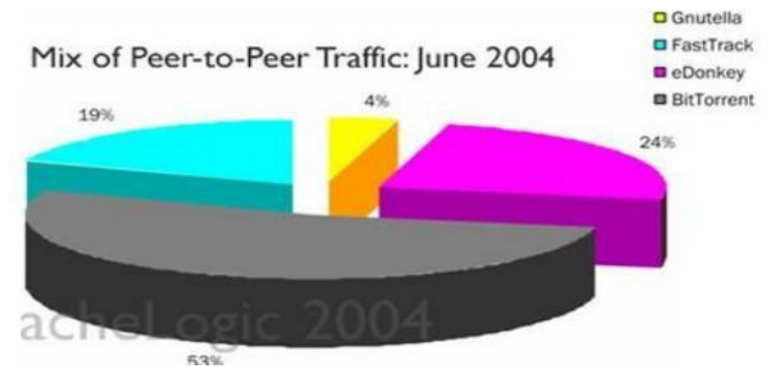
Számítógép hálózatok, osztott rendszerek

BitTorrent

BitTorrent

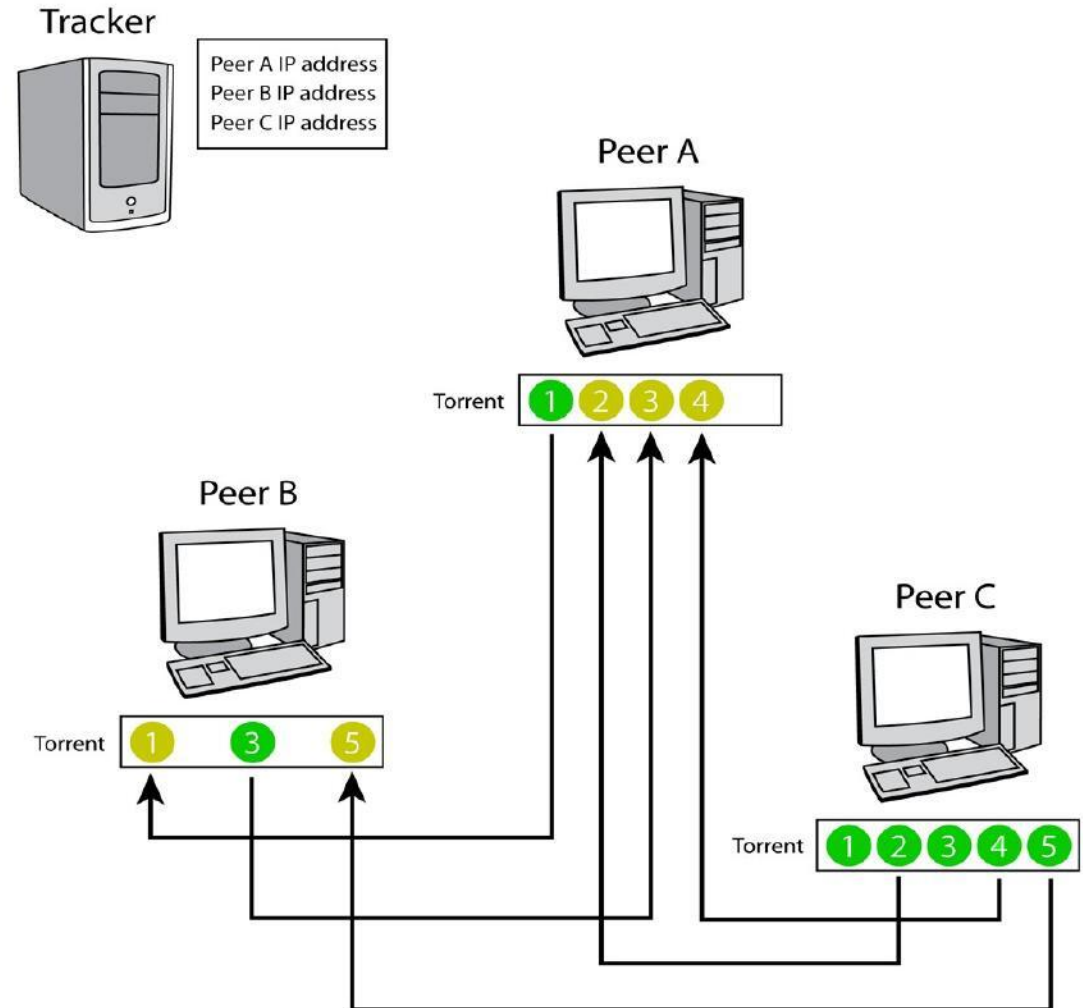
- Hatékony tartalomelosztó/fájl cserélő rendszer
- Nem strukturált peer-to-peer rendszer
- Kezdetek: 2001 – Bram Cohen – BitTorrent Inc.
- 2009-ben a teljes Internet forgalom 27-55%-át generálta

- Alapelvek:
 - Tit-for-tat (valamit valamiért)
 - Ösztönző
 - Blokkok-részblokkok



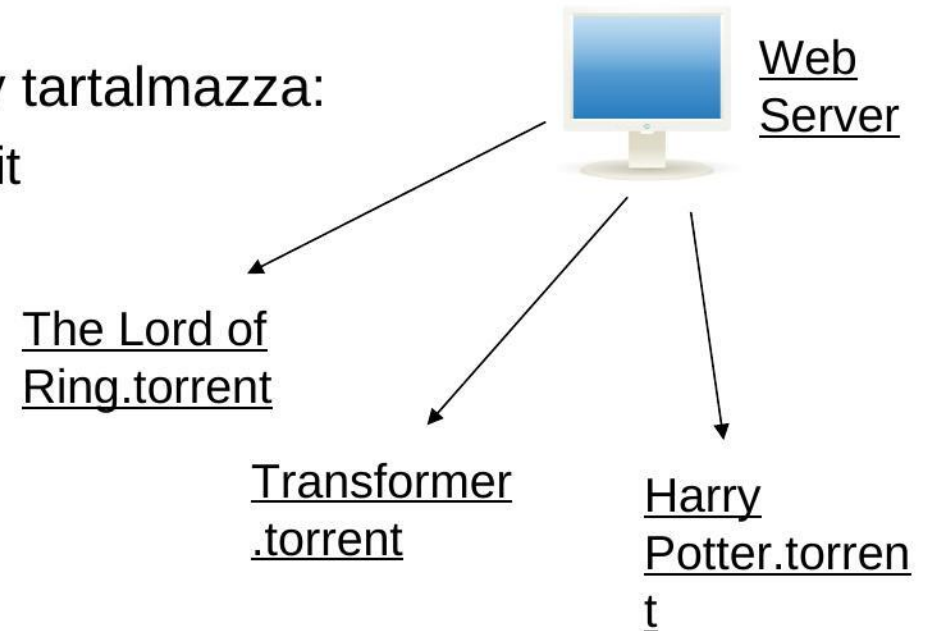
BitTorrent

- A file blokkokban (piece) kerül átvitelre
- A peer-ek miközben egymástól letöltik a file blokkjait, a meglévő blokkokat feltöltik egymásnak



Fájl megosztás – File Sharing

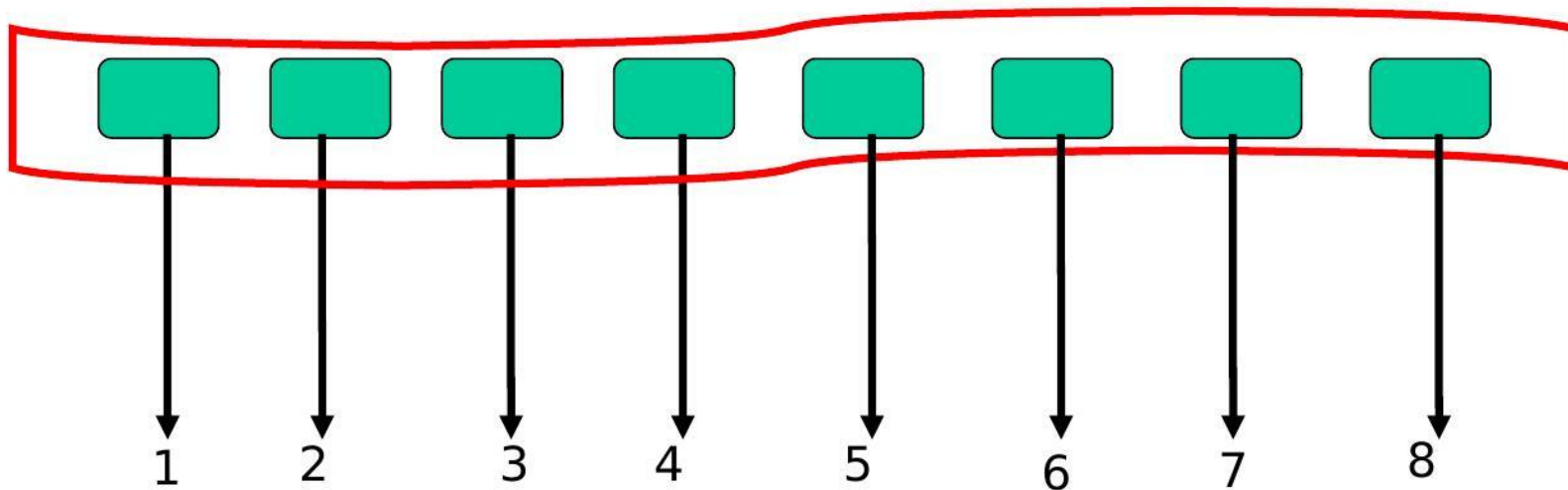
- Megosztás:
 - a peer egy .torrent fájlt készít, mely tartalmazza:
 - (1) a megosztandó fájl metaadatait
 - (2) információkat a tracker-ről



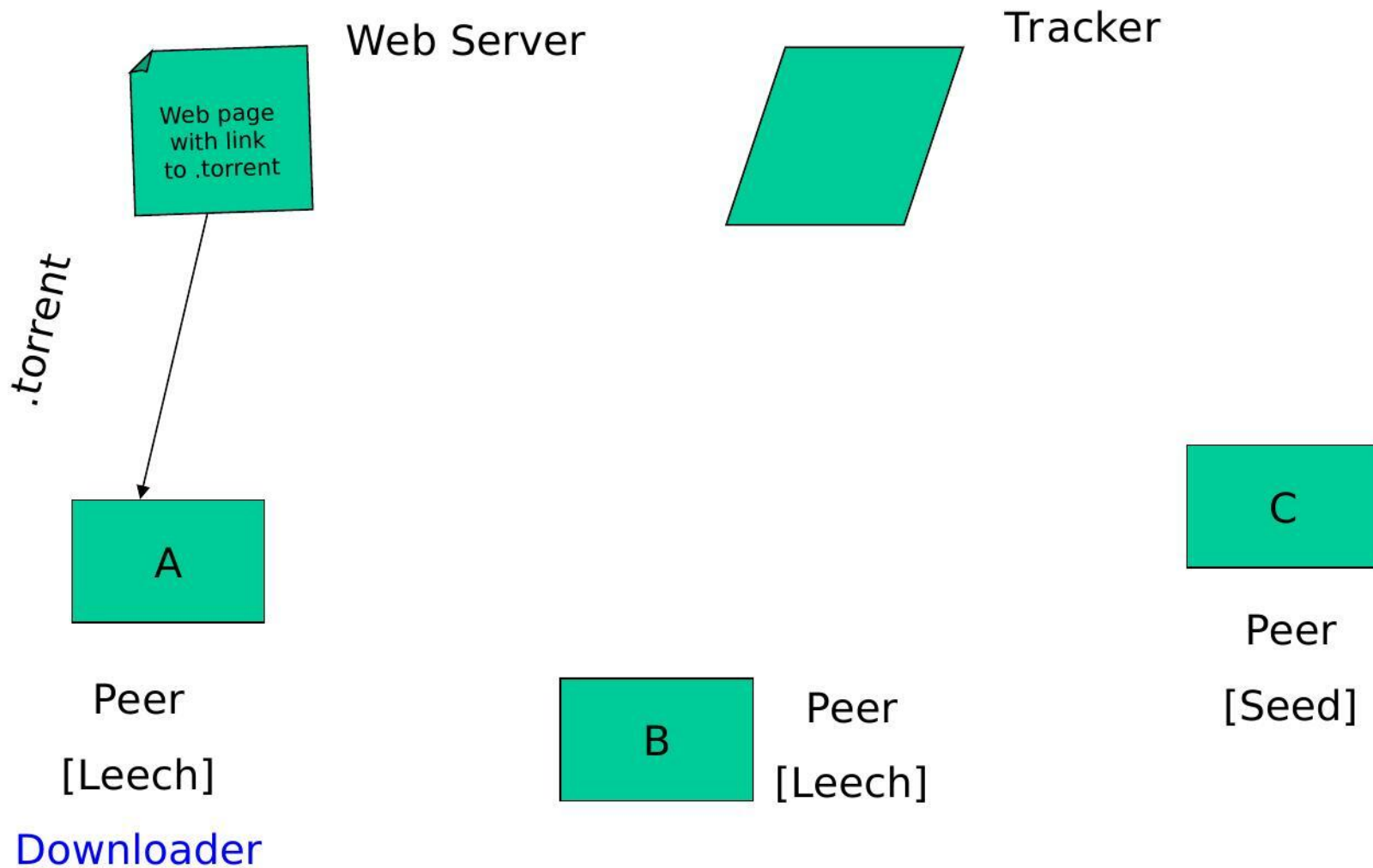
- Letöltés:
 - A peer először egy .torrent fájlt kap
 - Csatlakozik a megfelelő tracker-hez,
 - ami megadja, hogy mely peer-ektől tudja letölteni a fájl darabjait

Fájl megosztás – File Sharing

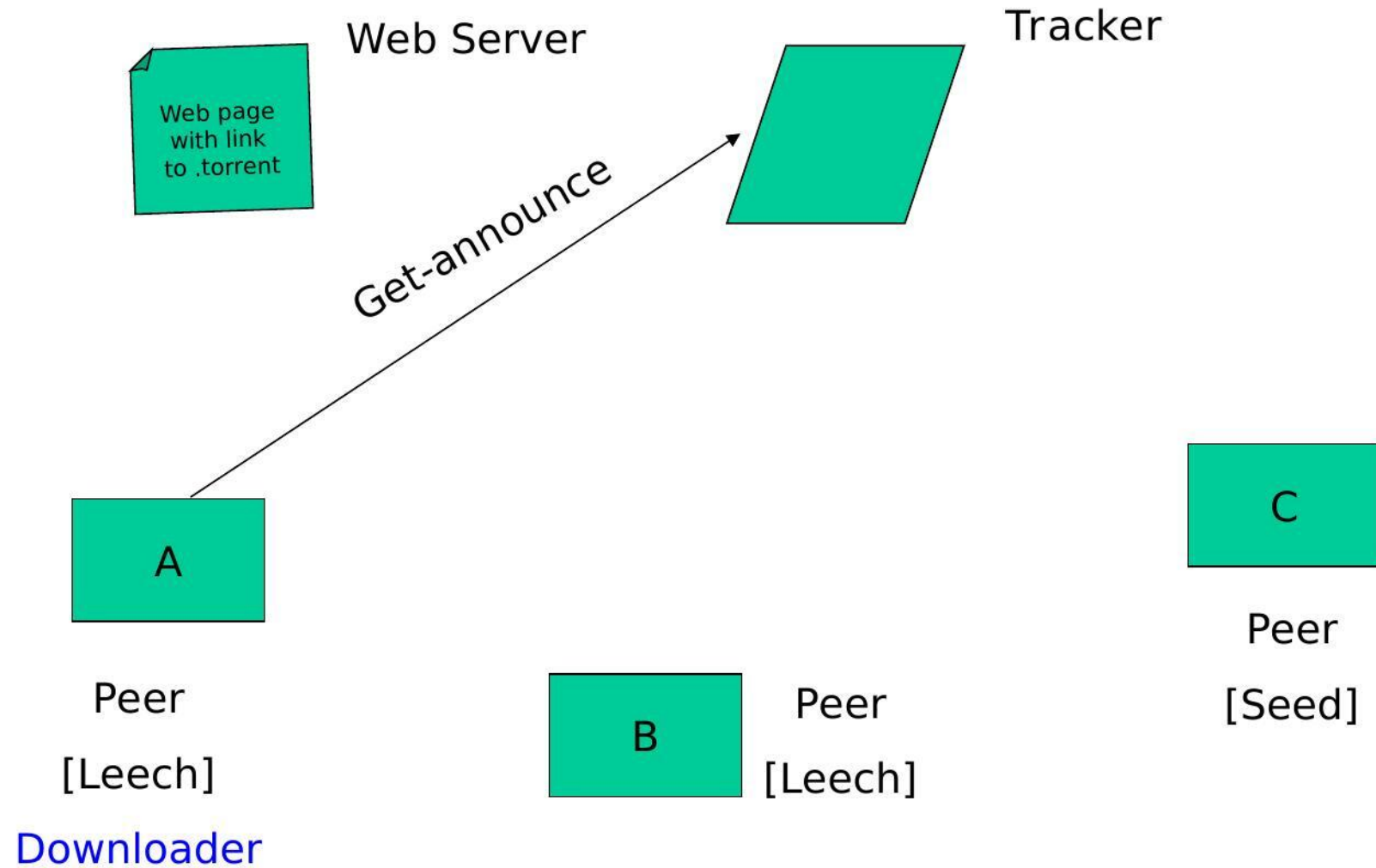
- A nagy fájlokat 64 KB és 1 MB méret közötti blokkokra (pieces) bontjuk



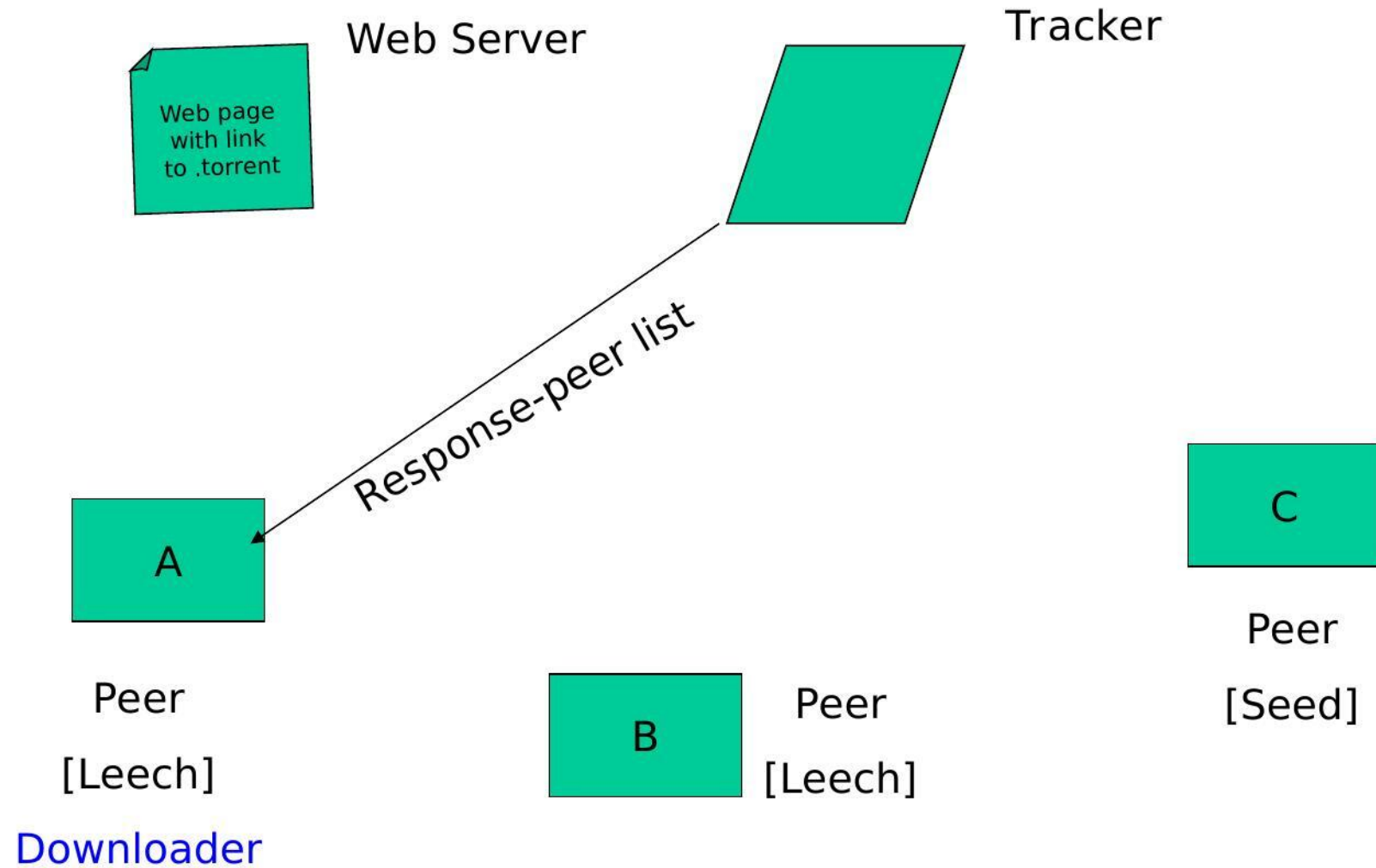
Architektúra



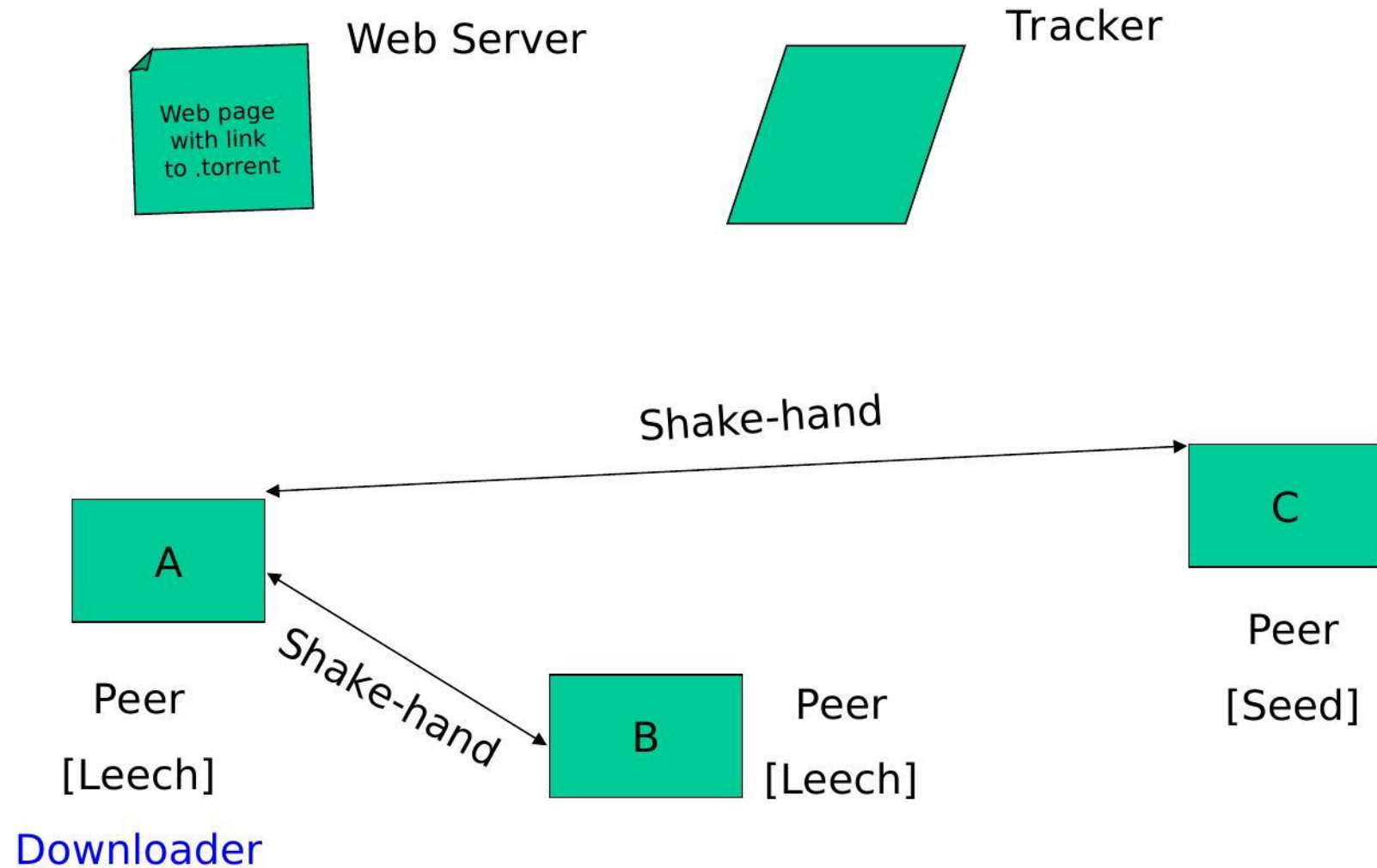
Architektúra



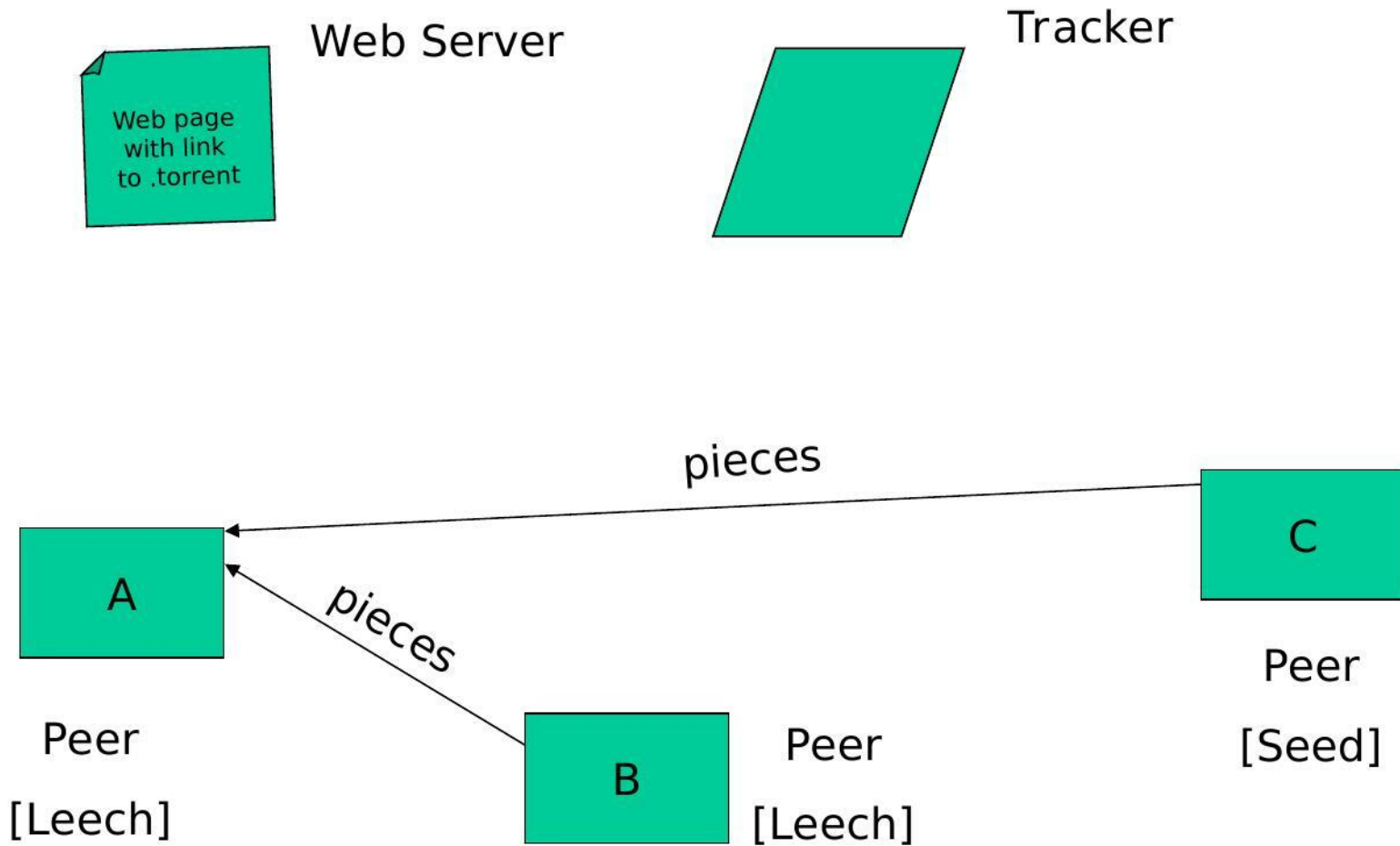
Architektúra



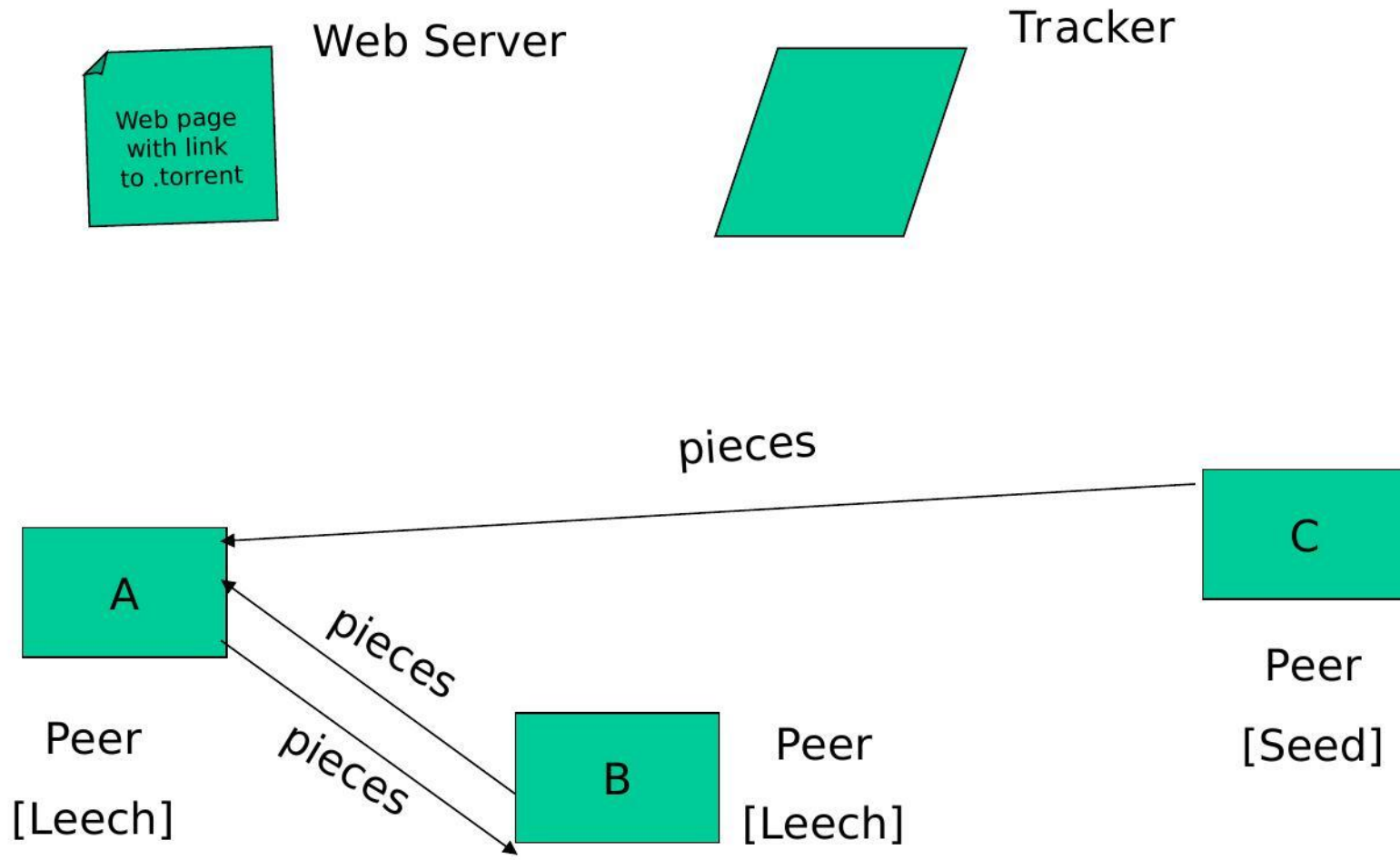
Architektúra



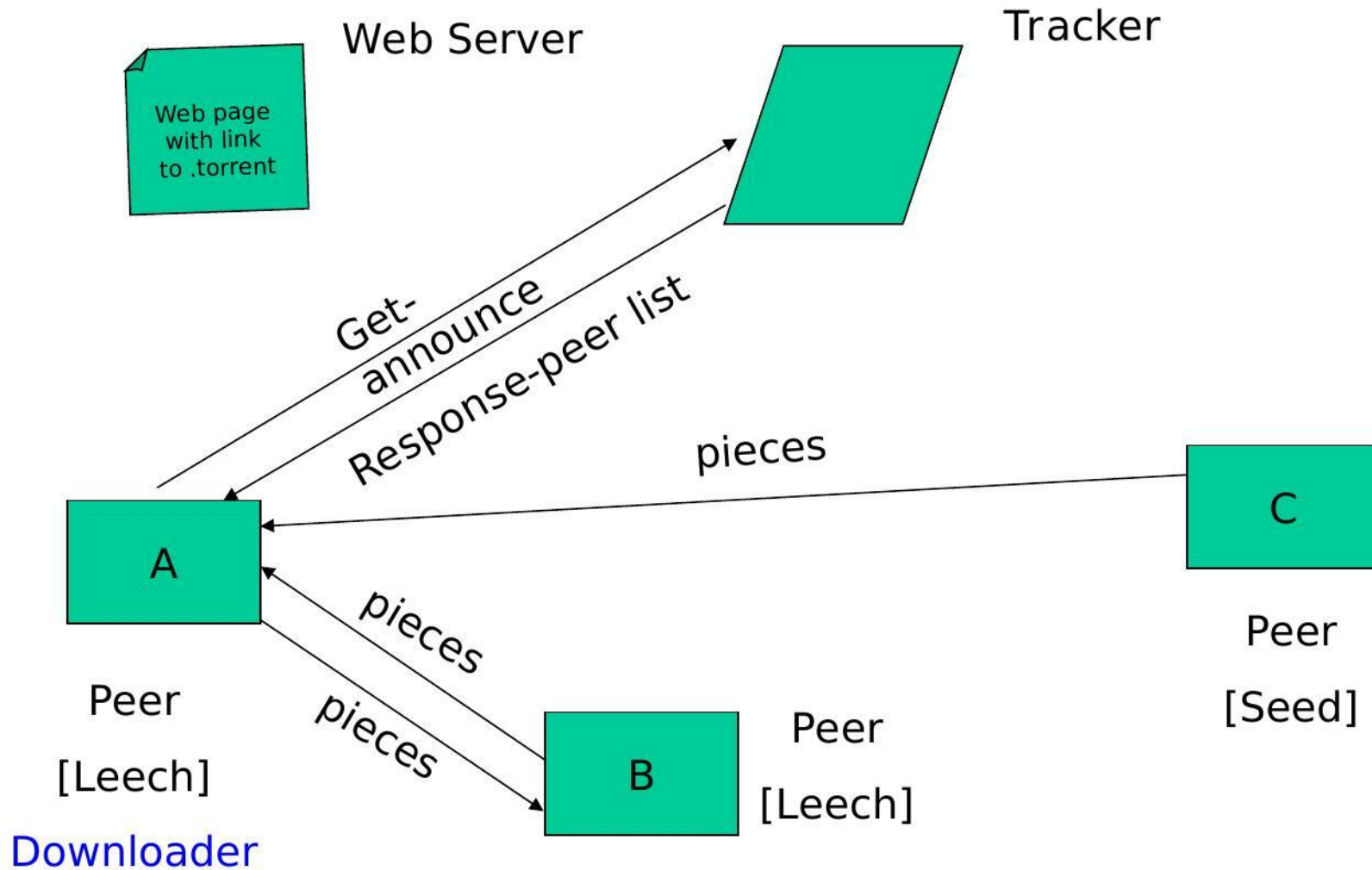
Architektúra



Architektúra



Architektúra



Egy .torrent fájl

- A tracker elérhetősége (URL)
- Blokkok (Pieces) <hash1, hash 2,..., hash n>
- Blokk (Piece) hossza
- Fájl(ok) neve
- Fájl(ok) mérete

```
d8:announce39:http://torrent.ubuntu.com:6969/announce13:announce-list1139:http://
/torrent.ubuntu.com:6969/announcee144:http://ipv6.torrent.ubuntu.com:6969/announ
ceee7:comment30:kubuntu CD releases.ubuntu.com13:creation datei1286701667e4:info
d6:lengthi729368576e4:name31:kubuntu-10.10-desktop-amd64.iso12:piece lengthi5242
88e6:pieces27840:^XÜ>,Ij26ML0^["]ŽWÚŮP6óAt§e~→śśž,^ZÄJ-4SM^N^Dähn@w0q;CP
ŽĚD1#Á-ŽĚäOCöPñŮ^O^[\â^ Ūüv|=*\ŰtEH?f@wY ō[?E^Fä§ôô^^[\ ^\zâ~Xd^W$^@7.d<P^E$Uáx'
CaúLuí,1'^[■#U3i;<<^Í^[\^T^_â,■Ý■Ī_ĚB4^]Ó\■Ů^E*Ŧ,ĀžŪŽŮ...>^"U*ŝ1řúgîP}Ī"Ç-îx~ž>Ā>Ÿ--
2A,,#©fš;yž^ZŽ■Ěäš-U^Xžbc^_yŭ>'ŮŽ1ÁŮŝUĀx"Ů^Gu&1#zŦŦD-î?ELĹŮS62ăé<<Ůz^]^[ '■q'»S■~/''
Ĉ^\
    ^U,Ī^S^B3·
    @RŮôBěö  ·Ū;Ĉa^Y?ŽShôzYL^U^@k^F^XD-^@^K°ymL"gsĚĪŮ^BĀpi6ŮĚ
^G■^C^Aq~"ñ?b^Q^@@_žěśś^L1"Ā^ŌŦŮ^CUMŌŮŘ^LĀ!Ě#uLgÍÉ[%-ŪŦŮtĪlŭvž  -MžĀ%rM^D*!qA\^
R
    #xtñE?U^@qēŮ]ŦyĪĀĈ<âcÇC^\ŮP!ĀáScZcž'^CđRăžš'ž'xqñ`#^_šĈ^E!ă'ñĚĀ
^[Ě]ĪĀ<■;ju^[C60~kŦ'^L^N(Āt"ăö;îD
    ;!$,^U2ŮŸ^CtK~tĪŮ}>:ĚrUw^Mμ6ž"šŦĀK%^]~]p-
^CŮĚ!Kă^P@<^T^DWI^Y^LŪh[ĸŮ>>Ašš1μŮř^\ś19\^PmžjG^'Ŧmôž^N^ř-ŦXb*p_Ěxqμç'ĪŮĚ;ô.-yx'^
O^@mKž^\HřŮž^Mňq^L;aqŮ.'áĪ-[ĚhJ  ■žbJ^"ĪŮ^GF-z7Ů2jšZ-■~uĉĀ&uie-[LŮRšg"ž■^B~#TTî8
^RŮ4%~p?ŝŮ^X^Rcš`^ŮS^Rđ~)[-^SžŦ>Ī#>Ÿ7^No^YS1^^ř^_>}>ú^G2^Lđq^A>Ů^]ēĪ ±ējĚkJž>^^
^
-UJ^RžH#šg^XBA
iĀx'«■~ĪPcx■šžX
hŦkôTždšĪ^Tž,■ĀŮ^MěxĀžžžŮĪžq\ŮĀĚĈk†|Ů+šřŸ?^L^[~w±^Kăéqîz^CYĀI1fĚC"Ĉ~fU^Pü(d^TA_
T-žRĀĚŮŮ■Ě&ăéĪ^WŮxŮ■x<Ů^Cx^F^E^K  Ç<<šŦ■[KpY7,Ÿś^CŘēô^@,@Ÿ-ĚŮtă3■1š>1%  RŮv~»■š4ăô%
^O@<<
ŮbĪĀŮArk-%^GHšĚF^D†G^S#<<ö_]^K",^M^Zx- 7%~±žš,ňWqĪŮ^·^FUŮWŦX%šy#R^Ī#-Ů{Ůđ#X■kJ,D^R
^SŮŮ÷tŘŮ^Zq^XĀ!ŮĚĪx^O-'-žĀř^]ödüĈTĚZ^YVG^B^FĪ' -ĈŮžă)~(@i2pŦ+>^T;Awy■^ž^ĈKbé"Ā4ĀĪ^
NŸĪřž4^YH_Ŧ#&^D^ăăXH  Tn%ĪŸ|
```

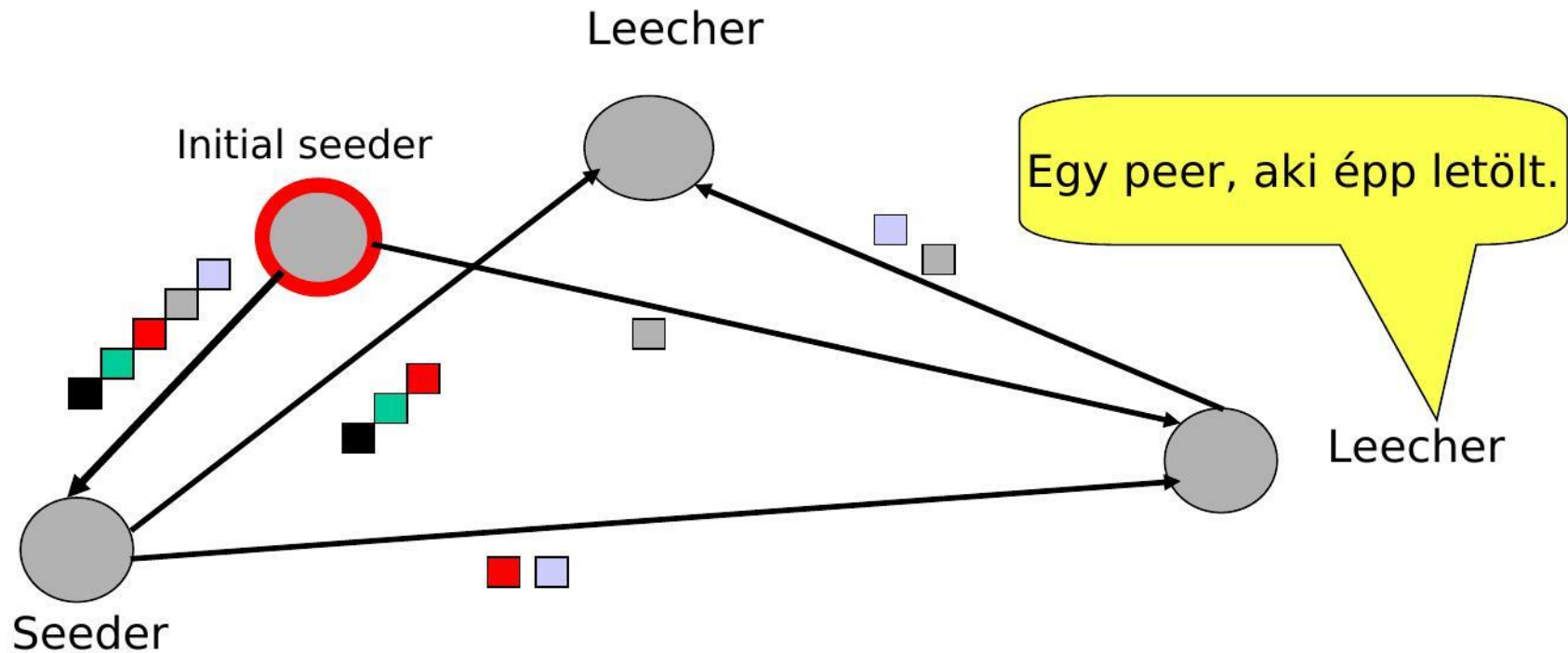
A Tracker

- Nyilvántartást vezet a peer-ekről
- IP cím, port, peer id
- Állapot információ (Completed or Downloading)
- Random peer-listát ad vissza

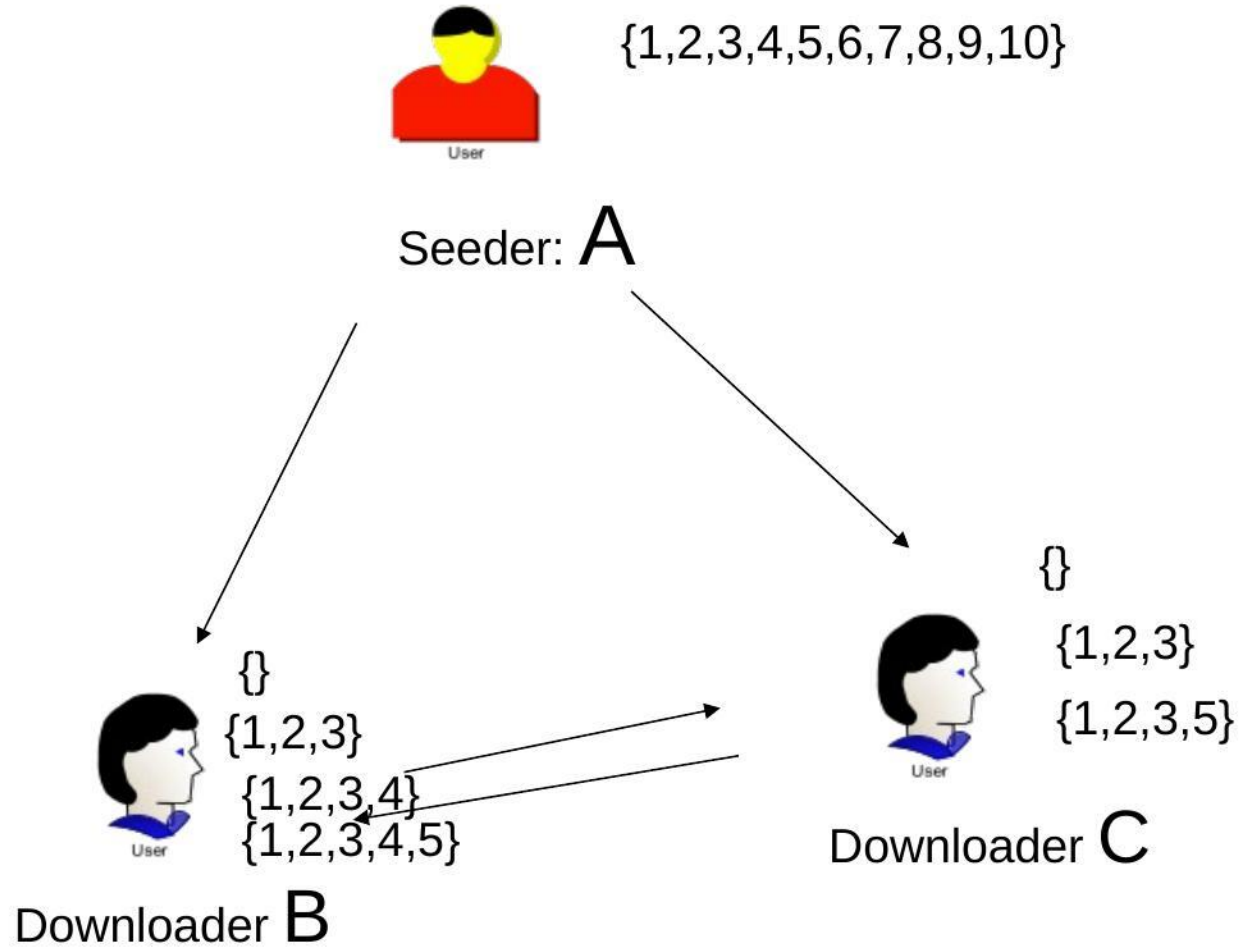
BitTorrent terminológia

Seeder = egy peer, akinél a teljes fájl elérhető

Initial seeder = egy peer, aki a kezdeti másolatot nyújtja



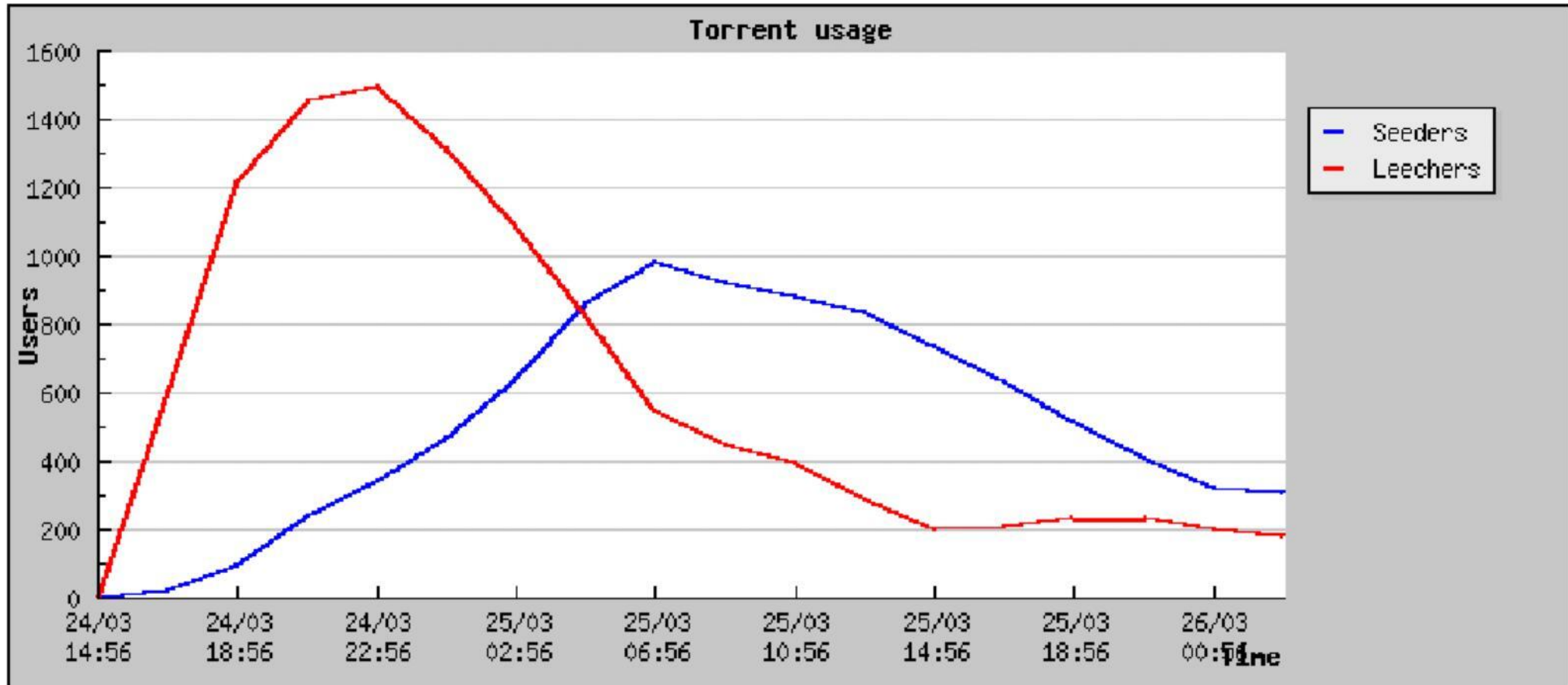
Egy példa



Alapelvek

- Initial seeder: blokkokra osztja a fájlt
- Seeder: akinél a teljes fájl elérhető
- Leecher
 - A .torrent fájl alapján megtalálja a tracker-t
 - Más, épp letöltő peerek listája a trackertől
 - Amint elkészül egy blokk letöltésével, azt meg tudja osztani más peerekkel
 - Az összes blokk megszerzése után, összeállítja a fájlt. Ezzel seeder-ré válik.
 - Minél több letöltés annál több replikátum a rendszerben

Működés



Blokkok és részblokkok

- Egy blokk tipikusan 16 KB méretű részblokkokra bomlik
- Amíg egy blokk nincs kész, addig annak a részblokkjai nagy prioritással töltődnek
- Cél, mielőbb előálljon a teljes blokk

Pipelining

- Adatátvitel TCP felett
- Több kérés kiszolgálása egyidőben
 - Tipikusan 5 kérés
- Mikor egy blokk vagy egy részblokk érkezik, új kérés kerül kiküldésre

Blokk Kiválasztás – Piece Selection

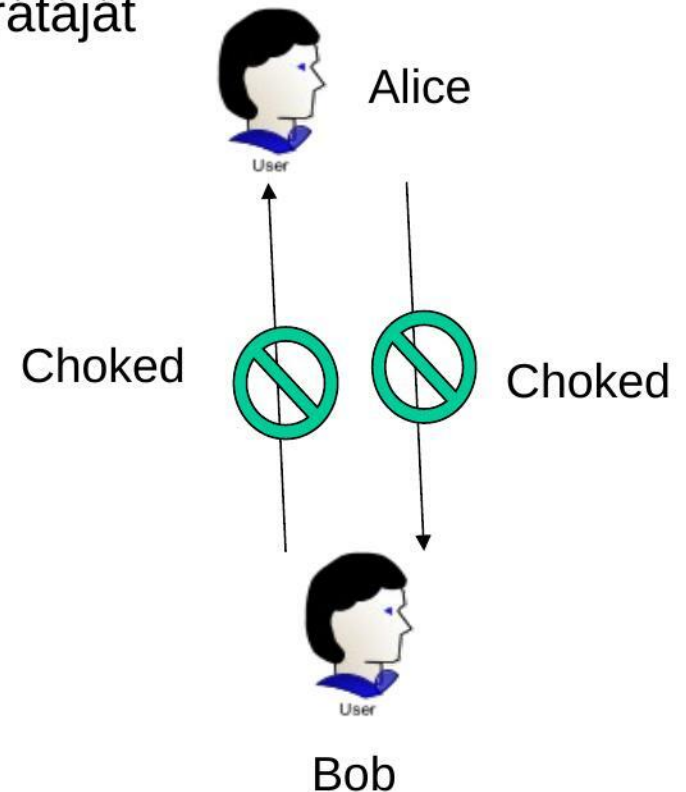
- A jó sorrend FONTOS a jó teljesítmény eléréséhez
- Ritka blokk esete
 - A legrosszabb eset, amikor néhány blokk senkinek sincs meg
 - pl. a peer-ek, ahol megvolt, eltűntek
 - Ilyen helyzetben ha az initial seeder leáll, akkor nem rakható össze a fájl!
- Mi a jó stratégia?

Blokk Kiválasztás

- Strict Priority
 - Miután egy blokkhoz tartozó részblokk megérkezett, utána az ahhoz tartozó részblokkokat a legnagyobb prioritással töltjük
 - Cél: Teljes blokk a lehető leghamarabb
 - Elsődleges szabály
- Rarest First (általános szabály)
 - Mindig a lokálisan legritkább blokkot kérjük
- Random First Piece
 - Az első blokk letöltéséig ez a policy. Random blokk, mely részblokkjai akár több szomszédtól egyszerre tölthetők.
 - Cél, hogy legyen saját blokk!
- Endgame Mode
 - Már csak pár blokk van hátra. Mindenkitől kérjük a megfelelő részblokkokat. Ha megvan egy részblokk, akkor a többi kérést leállítjuk.

Choking

- Nincs központi erőforrás elosztás
- Minden peer próbálja maximálni a saját letöltési rátáját
- Tit-for-tat:
 - feltöltés
 - Choking
- Chocking
 - A feltöltés átmeneti megtagadása, leállítása
 - A rendszer potyautasainak kezelése
 - csak letöltenek, de fel nem
 - Free riders



Choking algoritmus

- Choking: A peer lefojtja B peert
 - ha A úgy dönt, hogy nem tölt fel B-nek
- Rechoking periódus ~10 sec
 - Letöltési ráta alapján
- Minden peer legfeljebb 4 másik érdeklődő peernek tölt fel egy időben
 - Ezek közül 3 a legnagyobb feltöltési rátával rendelkezik A-hoz
 - Tit-for-tat
 - A 4. random (**Optimistic Unchoke**)
 - A legrosszabb periódikusan cserélődik

Optimistic unchoke

- Új kapcsolatok kipróbálása, hátha talál jobbat
- Új peereknek is kell blokkot juttatni
- Minden 3. rechoke (~30 sec) periódusban

Upload only mode

- A letöltés befejeződött
- Kinek töltsünk vissza?

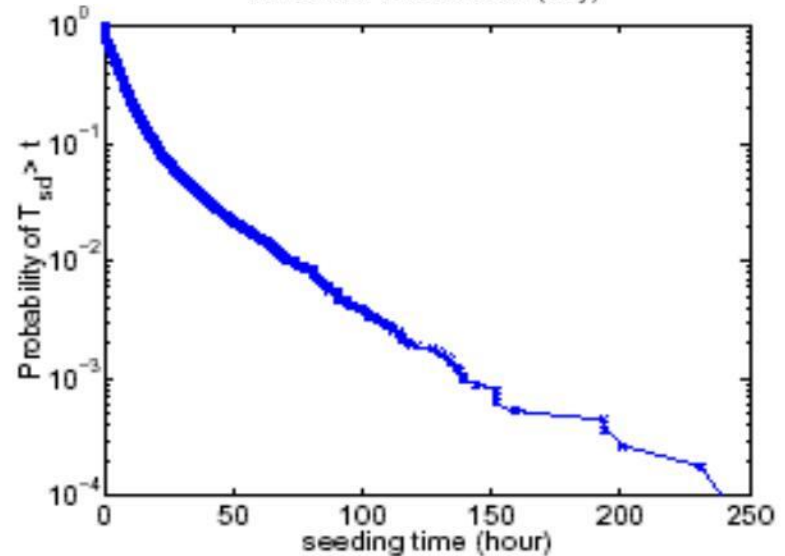
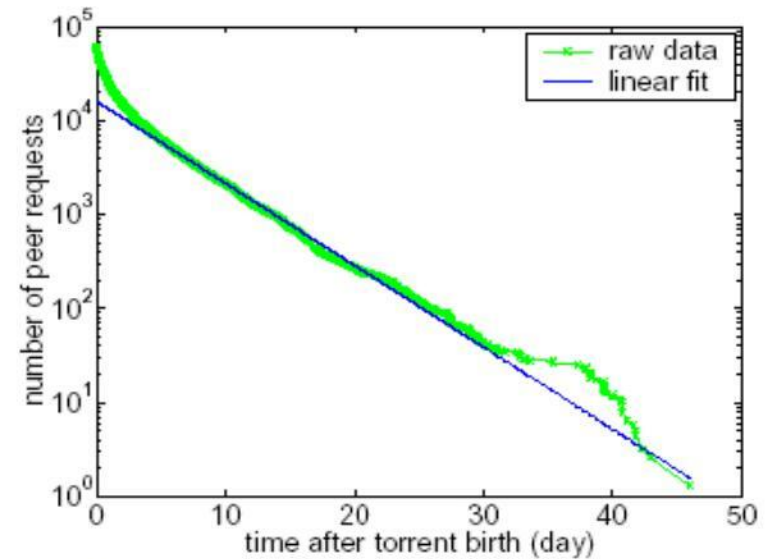
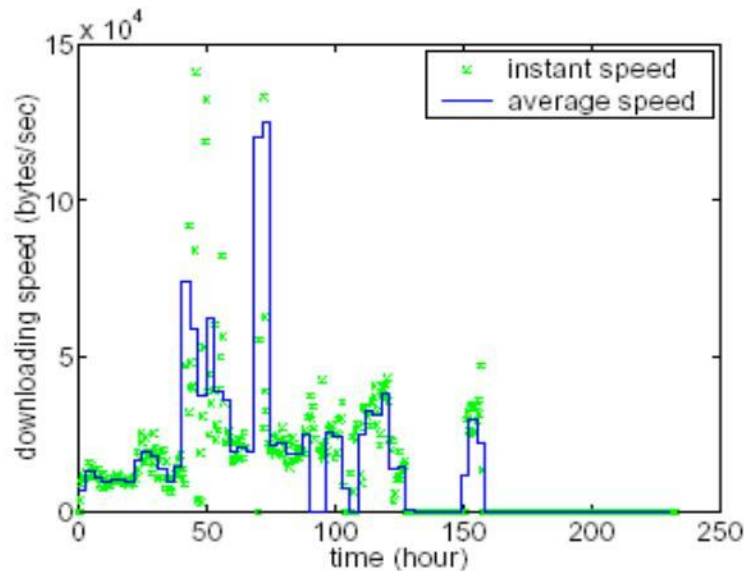
- Akihez a legjobb a feltöltési ráta
 - Cél, hogy minél előbb minél több seeder legyen

Kísérleti eredmények – 2005 (2003-2004)

L. Guo et al.
Measurements, Analysis, and Modeling of BitTorrent-like Systems
IMC 2005

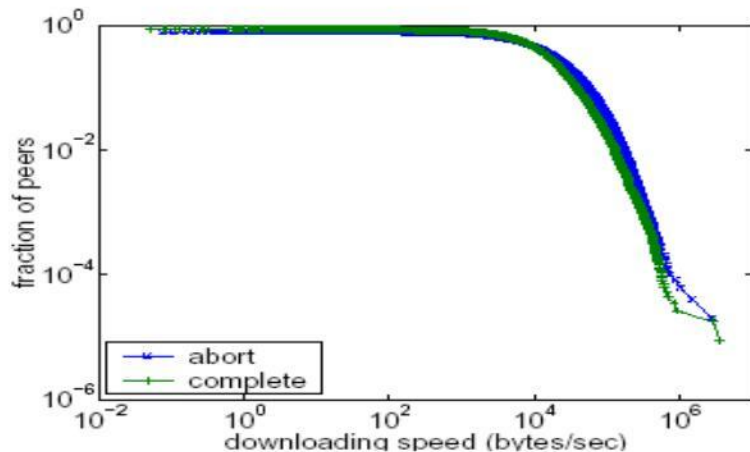
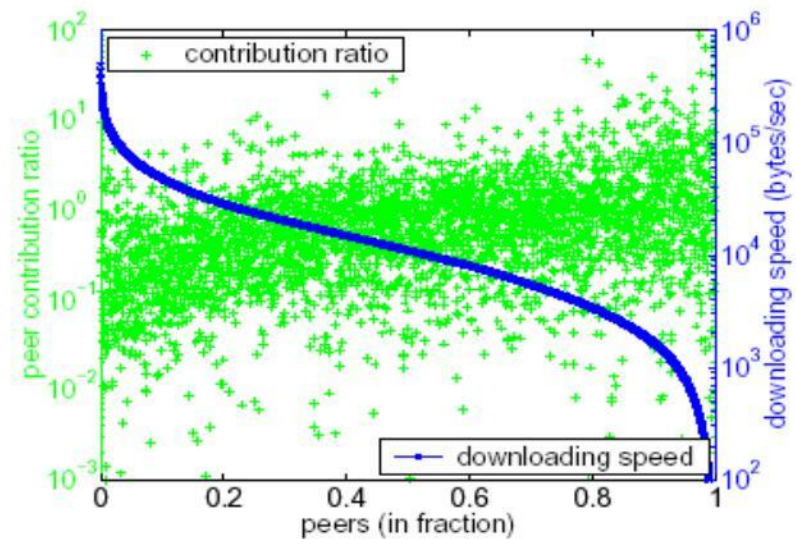
Egy torrent életútja

- Exponenciális lecsengés új kérésekre
- Kezdeti fellángolás, majd jelentős visszaesés
- A leecher-ek letöltési rátája elég nagy szórást mutat különböző időintervallumokban

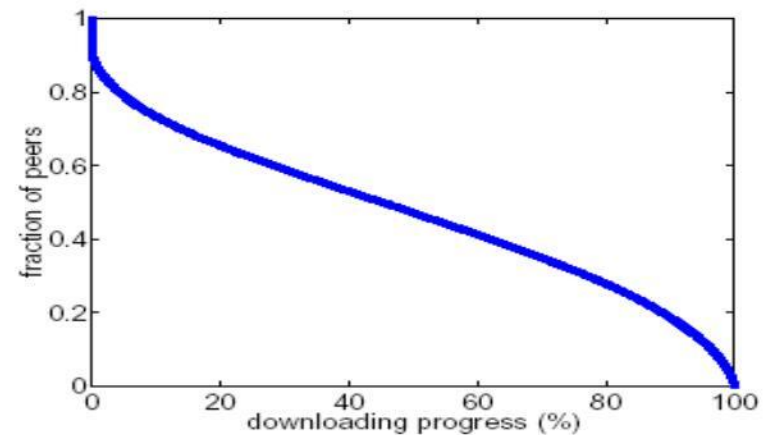


Peerek viselkedése

- Egy peer hozzájárulása a hálózathoz csökken, ha nő a letöltési rátája
- Annak a valószínűsége, hogy egy peer megszakítja a letöltést lényegében független a letöltési sebességétől, és a letöltés állapotától.



(a) The downloading speed distribution (complementary CDF, in log-log scale)



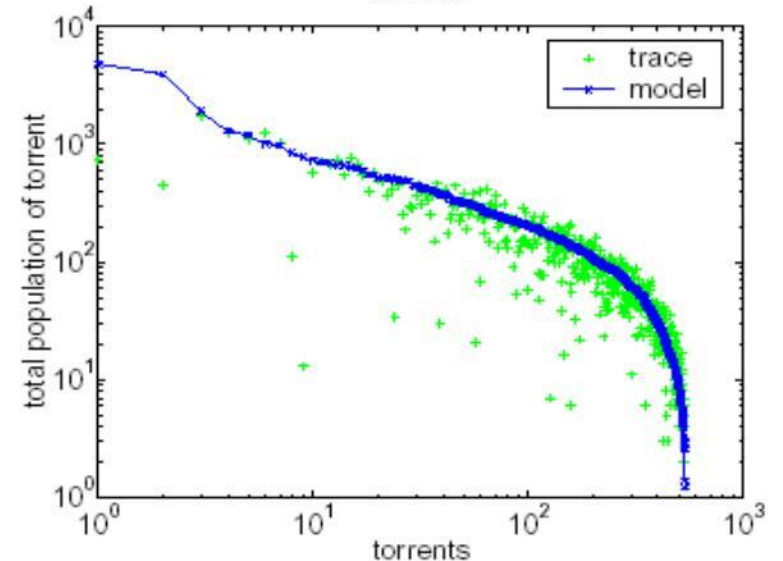
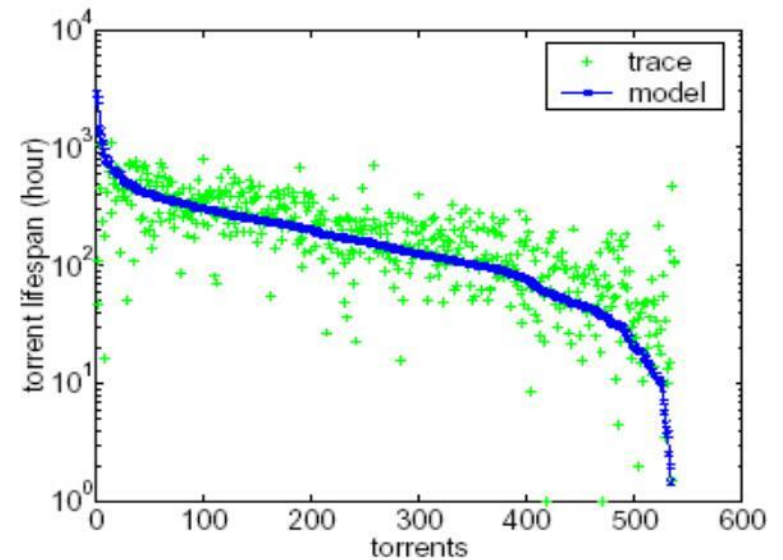
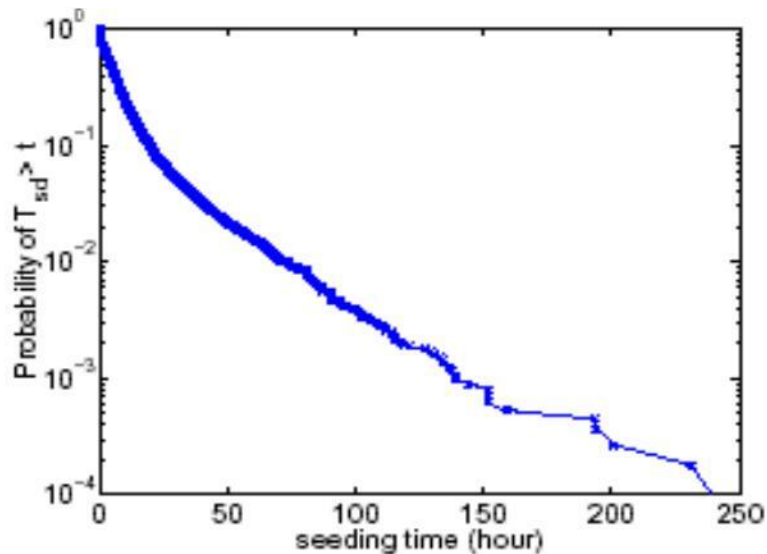
(b) The downloading progress distribution (complementary CDF)

Peerek viselkedése

A legtöbb torrent élettartama 30-300 óra között van, az átlag 8.5 óra

Az átlagos populáció ~102 peer, ami nem sok

A seed-elés átlagos hossza 8.42 óra



Teljesítmény elemzés

D. Arthur, R. Panigrahy:
Analyzing BitTorrent and Related Peer-to-Peer Networks,
SODA 2006, 961-969.

Modell

- G gráf n csomóponttal b megosztandó adatblokk
 - Kezdetben egy csomópontnál van az összes blokk (initial seeder)
- Diszkrét lépések:
 - Minden csomópont felajánlja egy szomszédjának a feltöltést
 - Random választás
 - Minden csomópont elfogad egy felajánlást és letölt egy blokkot.
 - Tetszőleges választás
- T lépésszám, mire minden peernek meglesz minden blokk
 - Mennyi ez a T ? korlátok...

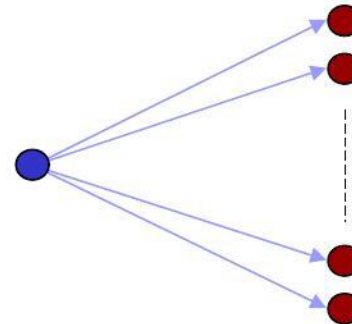
Korlátok

Legjobb eset:

- $T = \Omega(b + \log n)$
 - egy csomópont minden blokkot megkapjon: $\Omega(b)$
 - egy blokk minden csomóponthoz eljusson: $\Omega(\log n)$

Legrosszabb eset:

- $T = O(bn)$
 - Legalább egy letöltés minden lépésben történik
 - Példa:



BitTorrent gráf

- BitTorrent hálózat egy speciális gráffal modellezhető:
 - Csomópontokat egymás után adjuk hozzá
 - Minden csomópont C darab véletlenszerűen választott megelőző csomóponthoz csatlakozik (ha nincs még C , akkor minden megelőzőhöz)
 - Ez a BitTorrent- C gráf
- A BitTorrent egy BitTorrent-50 gráfot használ
- Milyen tulajdonságai vannak ennek a gráfnak?
 - Fokszám
 - Átmérő

BitTorrent-C gráf – medián mélység

- Minden i -re, tekintsük j egészek halmazát, melyre létezik v_j -ből v_i -be vezető él. Legyen $\mathbf{m}(i)$ ezen halmaz medián eleme.
- $m^k(i) = m(m^{k-1}(i))$, ahol $k > 0$ és $m^0(i) = i$
- v_i **medián mélysége**: az a legkisebb k , amire $m^k(i) = 1$
- Látható, hogy $v_1 \rightarrow v_i$ távolság legfeljebb v_i medián mélysége

BitTorrent-C gráf tulajdonságai

- Nagy valószínűséggel:
 - Átmérő $d = O(\log n)$
 - Max. fokszám: $D = O(C \log n)$

Lemma: Egy BitTorrent-C gráfban $1-2/n$ valószínűséggel, bármely pont medián mélysége $\leq 3 \log n$ és a maximális kifoka a gráfban $\leq 3C(1 + \ln n)$.

BitTorrent-C gráf

- BitTorrent gráfok:
 - Sorban érkező peerek
 - Minden peer C korábbi peerhez kapcsolódik
- Módosítás: Smoothed BitTorrent-C:
 - A peerek sorban, egymás után érkeznek
 - Minden peer $2C \log n$ korábbi peert választ szomszédjául véletlenül, függetlenül, egyenletes eloszlás szerint és ezek közül a $C \log n$ legújabbhoz kapcsolódik
- A Smoothed BitTorrent-C a gráfon: $T = O(b + \log^2 n)$
 - Megj.: az optimum: $O(b + \log n)$

Referenciák

- B. Cohen: Incentives Build Robustness in BitTorrent. NetEcon 2003.
- L. Guo et al.: Measurements, Analysis, and Modeling of BitTorrent-like Systems. IMC 2005, pp. 35-48.
- D. Arthur, R. Panigrahy: Analyzing BitTorrent and Related Peer-to-Peer Networks. SODA 2006, pp. 961-969.