Peer-to-Peer File Sharing: Past!–Present–Future? a Delft View

11th IEEE Int'l Conference on Peer-to-Peer Computing
Kyoto, Japan

Dick Epema
Parallel and Distributed Systems Group
Delft University of Technology
Delft, the Netherlands

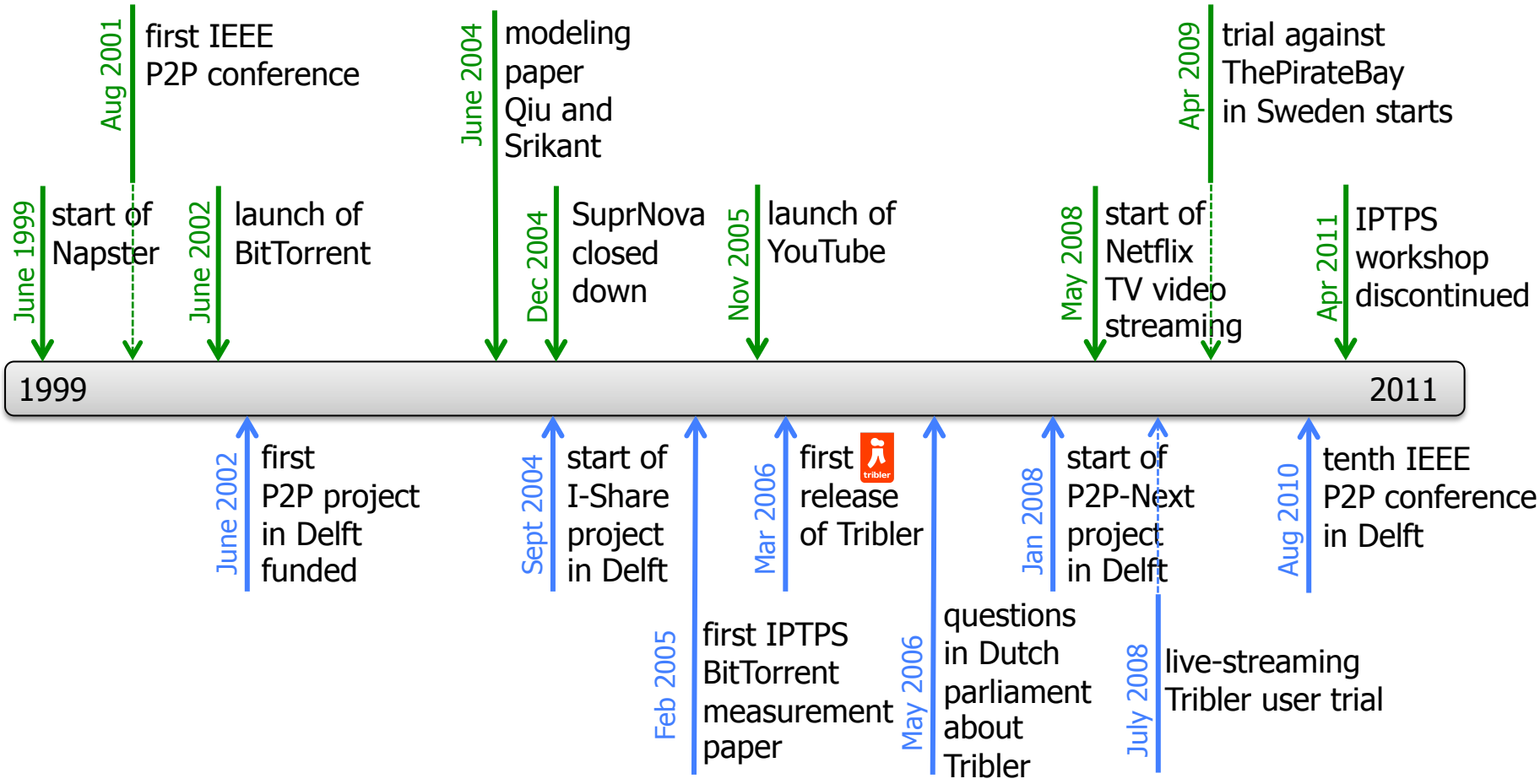
August 31, 2011

TU Delft
Delft University of Technology

P2P file sharing: a mixed bag

TU Delft

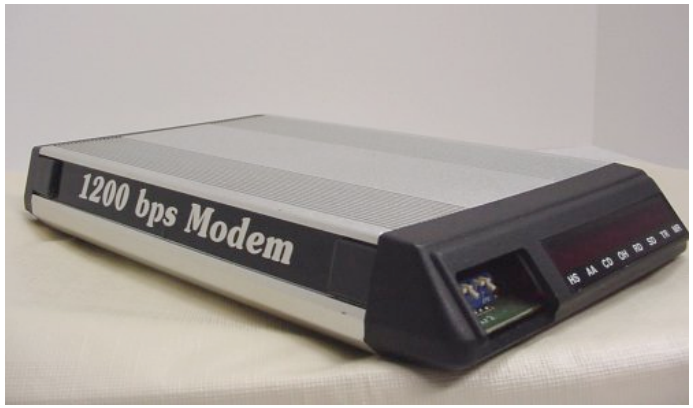
A little history



TU Delft

Bandwidth (1/3): 3000-fold increase

1200 bps dial-up modem



1988

500 MB movie: **40 days**

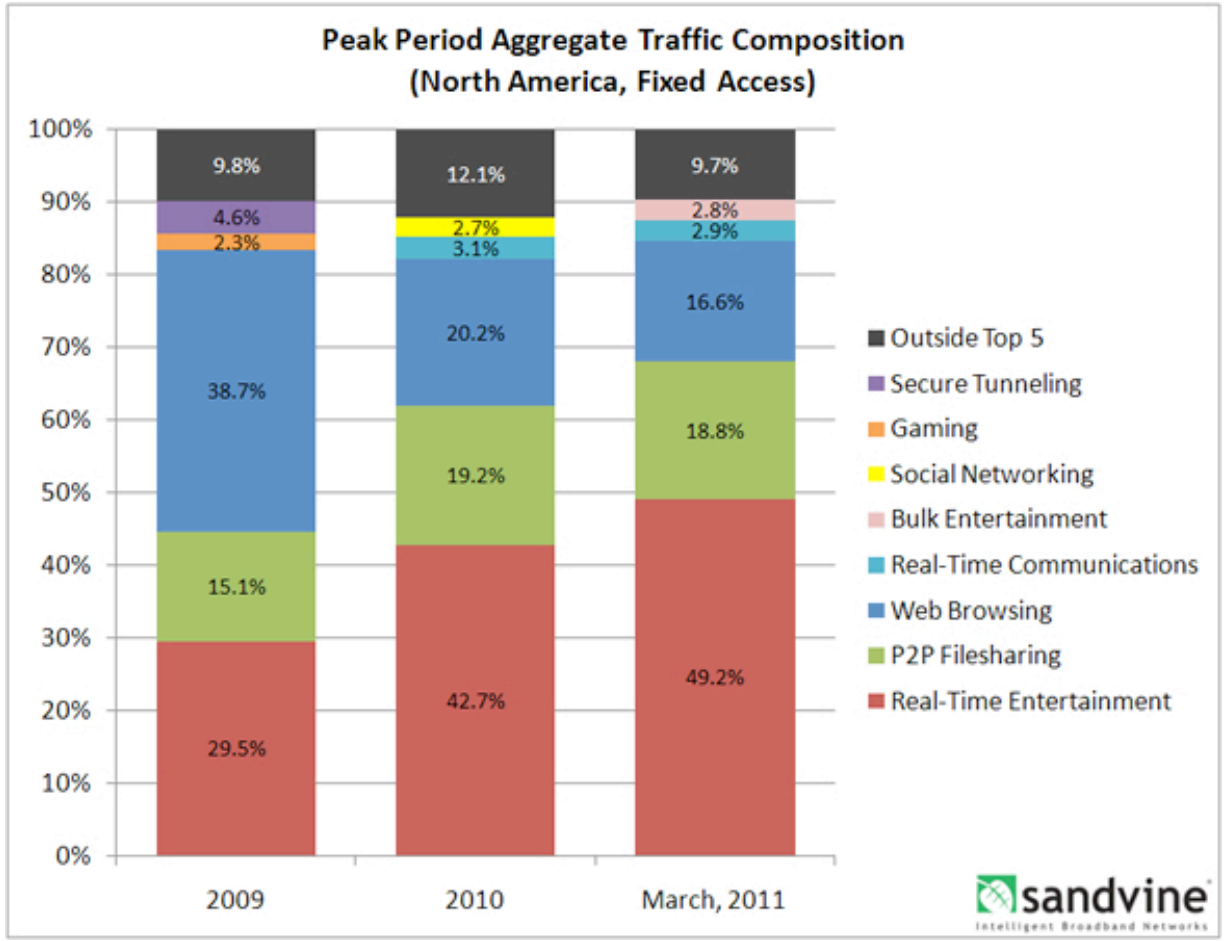
4 Mbps/1Mbps ADSL

2011

20 minutes

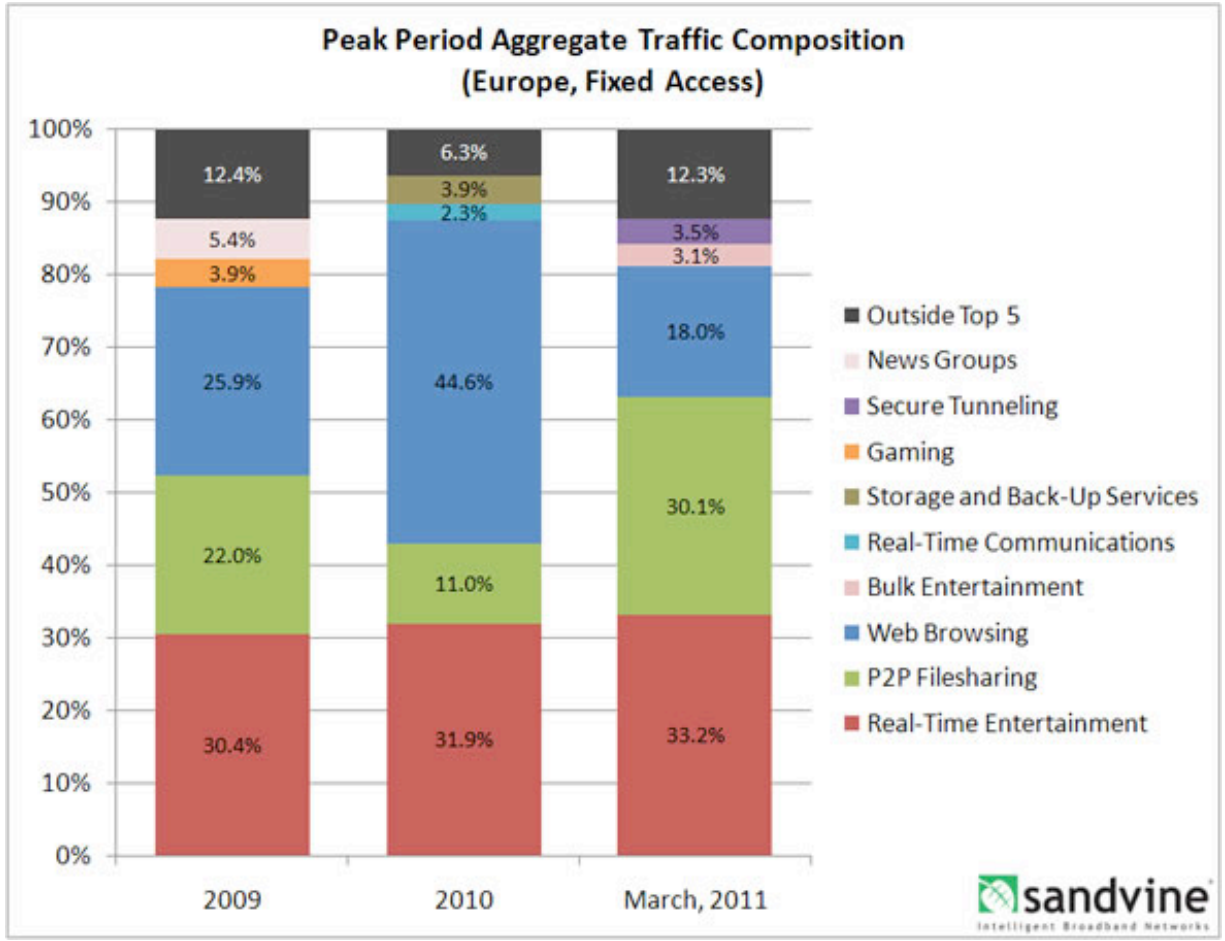
TU Delft

Bandwidth (2/3): backbone use in North-Am.



TU Delft

Bandwidth (3/3): backbone use in Europe



TU Delft

Video-on-demand vs file sharing (US)

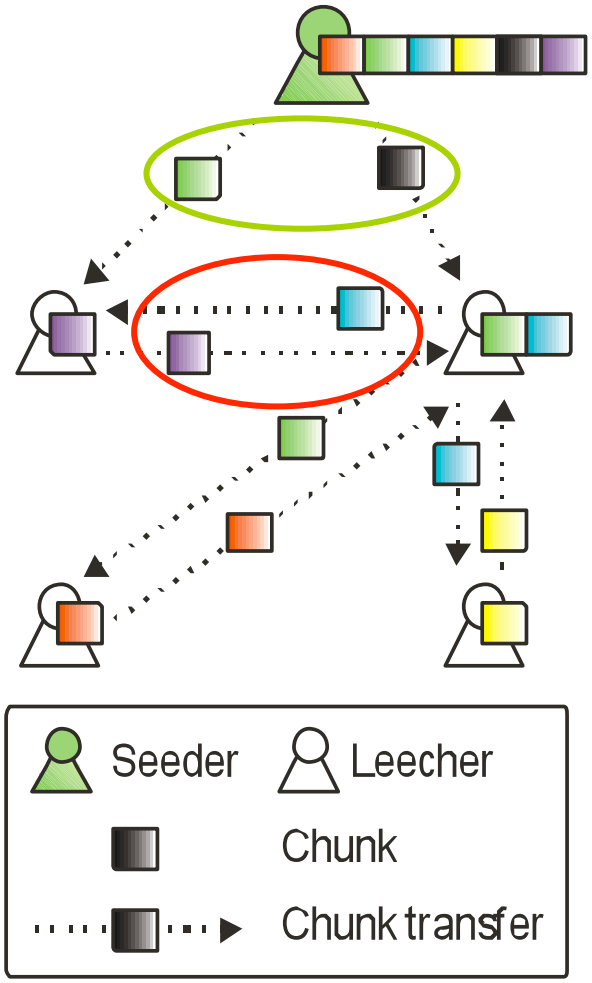
	Upstream Traffic		Downstream Traffic		Total Traffic	
Rank	Application	Share	Application	Share	Application	Share
1	BitTorrent	52.01%	Netflix	29.70%	Netflix	24.71%
2	HTTP	8.31%	HTTP	18.36%	BitTorrent	17.23%
3	Skype	3.81%	YouTube	11.04%	HTTP	17.18%
4	Netflix	3.59%	BitTorrent	10.37%	YouTube	9.85%
5	PPStream	2.92%	Flash Video	4.88%	Flash Video	3.62%

SOURCE: SANDVINE NETWORK DEMOGRAPHICS

sandvine

TU Delft

BitTorrent design (1/3): swarms



File divided into **chunks** (size 256 KB, 1000 per file)

Swarm – the set of peers downloading the same file

Seeders – online peers with the complete file who upload **for free**

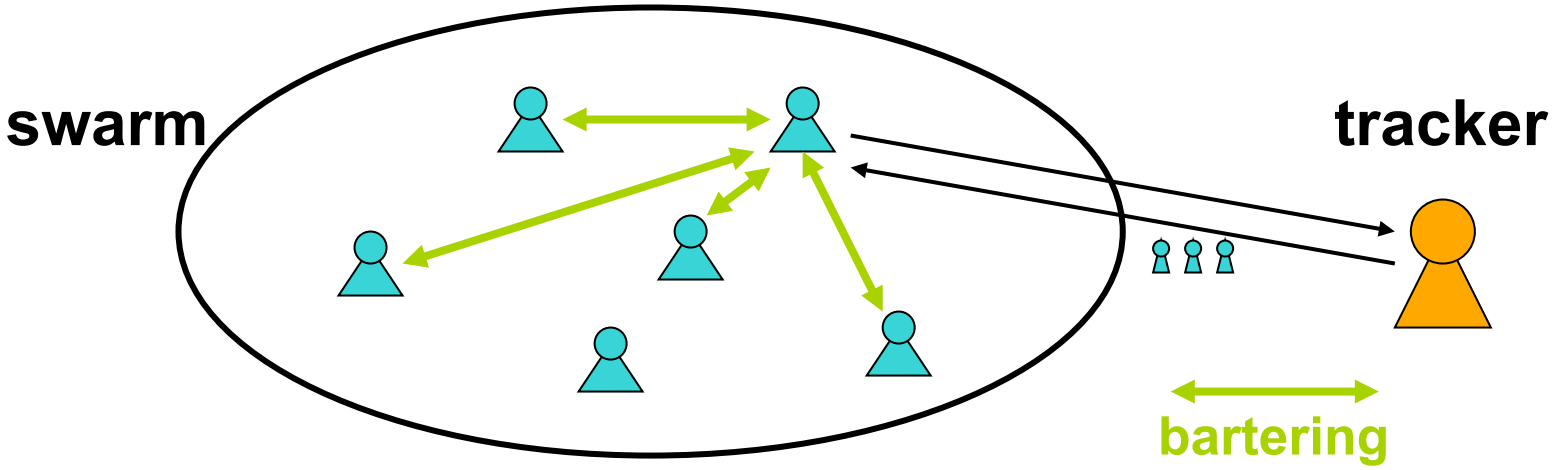
Leechers – peers whose download is in progress

Chunks exchanged between peers according to **tit-for-tat** strategy

TU Delft

BitTorrent design (2/3): trackers

- **Torrent metadata files** contain tracker URL(s)
- Trackers **centrally manage** swarms:
  - a peer indicates its interest in a file to a **tracker**
  - peers periodically contact a **tracker** to obtain the IP numbers of other peers downloading the same file
  - a peer selects the best other peers as **bartering** partners



TU Delft

BitTorrent design (3/3): downloading

- Peers in a swarm exchange the ids of the chunks they possess wit

[have

f file

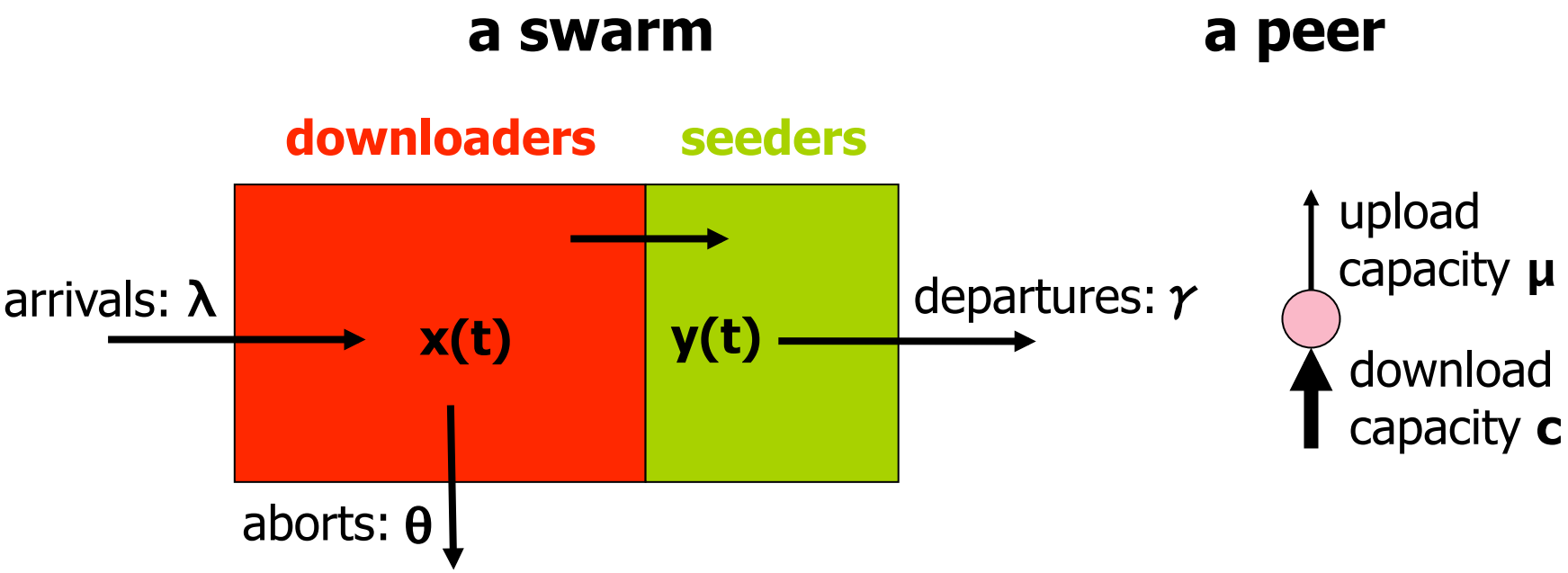
- Le
- An
- Chunk selection uses the **rarest-first policy**
- **Periodic optimistic unchoke** to find better bartering partners

- Whole BitTorrent design in three slides!!
- But:
  - tens of millions of users
  - large fraction of Internet bandwidth consumed
- Reasons:
  - simplicity
  - effectiveness
  - timeliness

TU Delft

BitTorrent analysis (1/4)

- A **fluid model** of the operation of BitTorrent for a single file

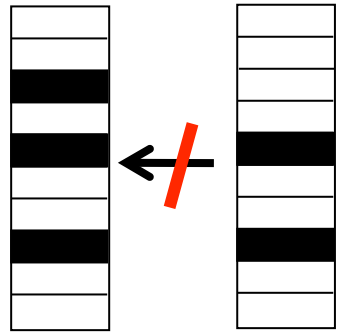


D. Qiu and R. Srikant, "Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks," *ACM SIGCOMM'04*, 367-377, 2004.

TU Delft

BitTorrent analysis (2/4)

- The **file-sharing effectiveness** η :
  - accounts for the **reduced download speed** of downloaders when they cannot find suitable bartering partners
  - turns out to be **close to 1**

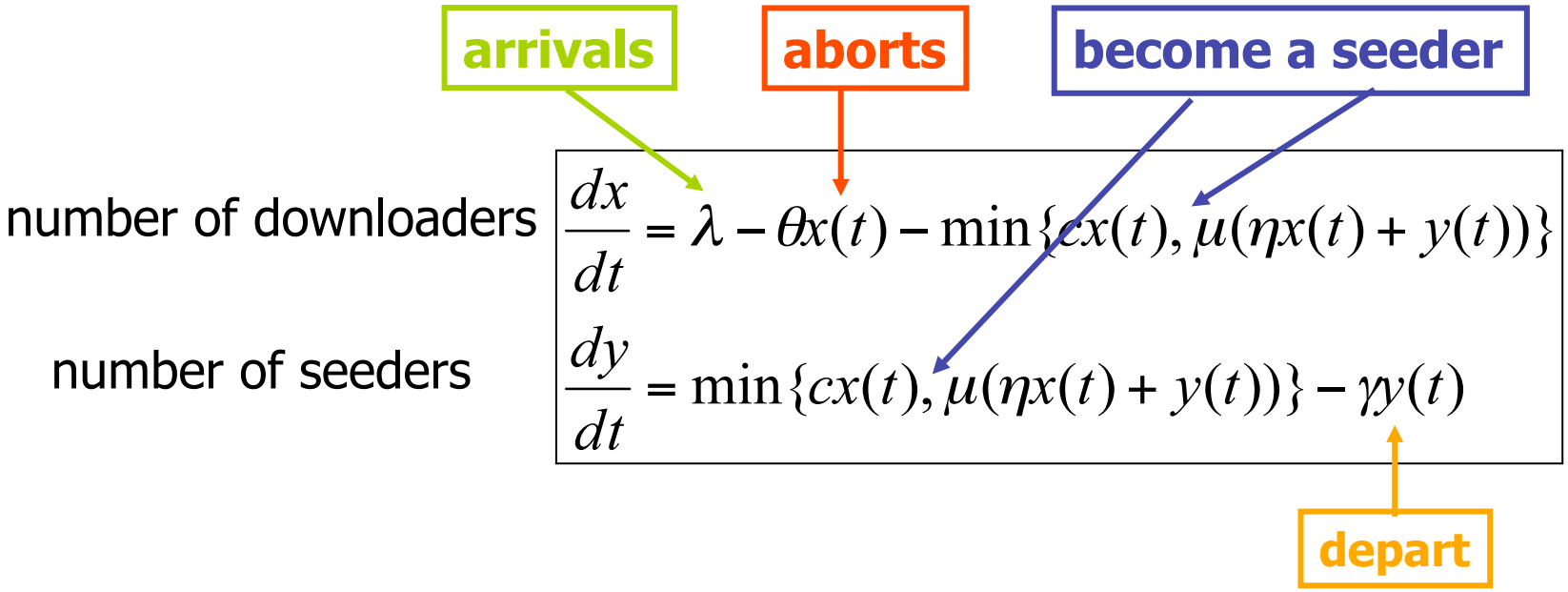


- Maximum **download capacity** in the system: **cx(t)**
- Maximum **upload capacity** in the system: $\mu (\eta \mathbf{x(t)} + \mathbf{y(t)})$
- Total **system transfer rate**: $\mathbf{min(cx(t),\ } \mu (\eta \mathbf{x(t)} + \mathbf{y(t)})$

TU Delft

BitTorrent analysis (3/4)

- Assume the **file size** to be **1**
- Assume the arrival, abort, and departures processes to be Poisson
- **System evolution** governed by two differential equations:



TU Delft

BitTorrent analysis (4/4)

except for η all system constants

- Very strong assumptions in mathematical model:
  - Poisson distributions
  - steady state
- Still this analysis reveals important insights
- Mathematical analysis may be (in)validated by simulations and measurements, and may steer design

TU Delft

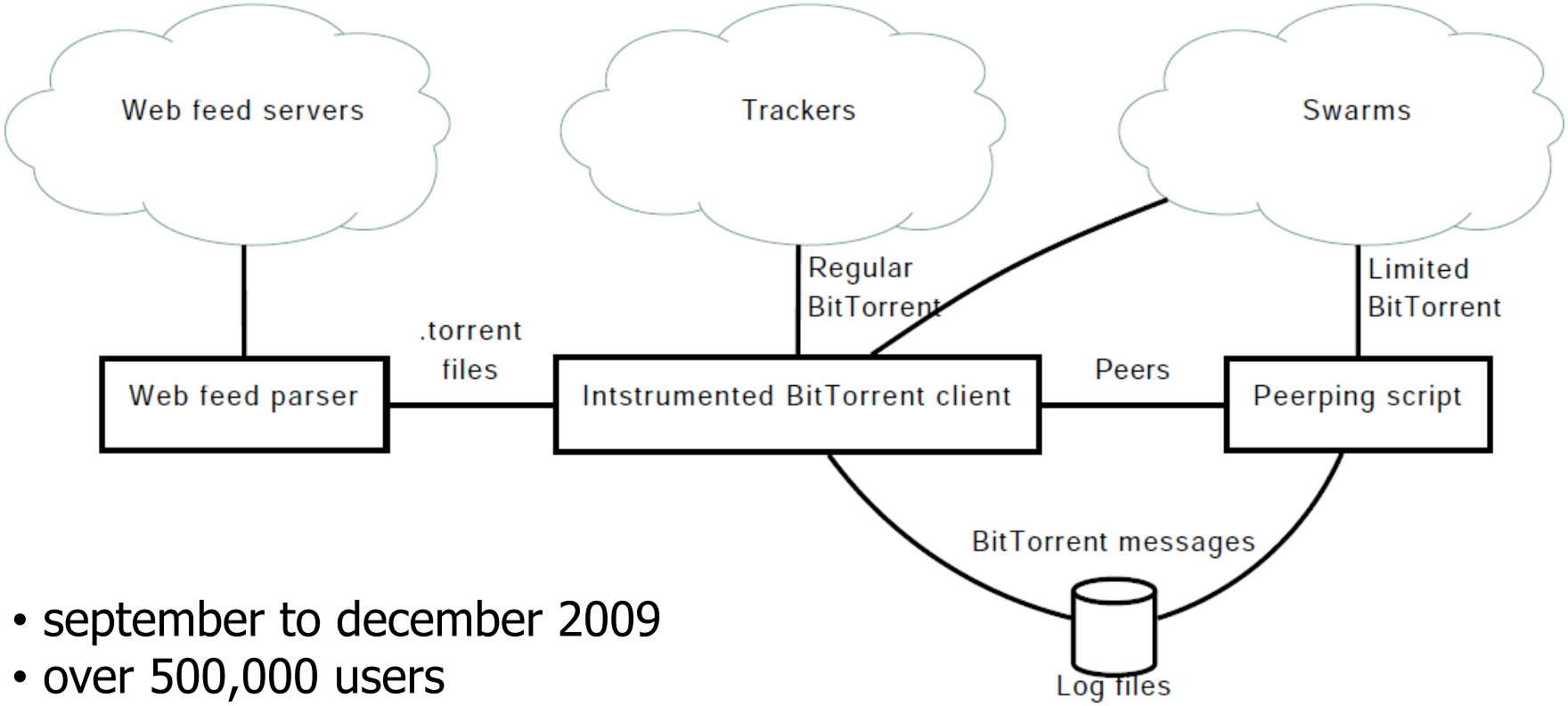
BT public vs private communities (1/5)

- **Private communities**
  - require membership
  - are difficult to measure
  - employ different forms of **sharing-ratio enforcement**
  - and so provide an incentive for seeding

- **Questions:**
  - are there **differences in performance** between public and private communities?
  - **how has BT evolved** since our measurement study in IPTPS 2005?

M. Meulpolder, L. D'Acunto, M. Capotă, M. Wojciechowski, J.A. Pouwelse, D.H.J. Epema, and H.J. Sips, "Public and Private BitTorrent Communities: A Measurement Study," *9-th Int'l Workshop on Peer-to-Peer Systems (IPTPS'10)*, 2010.

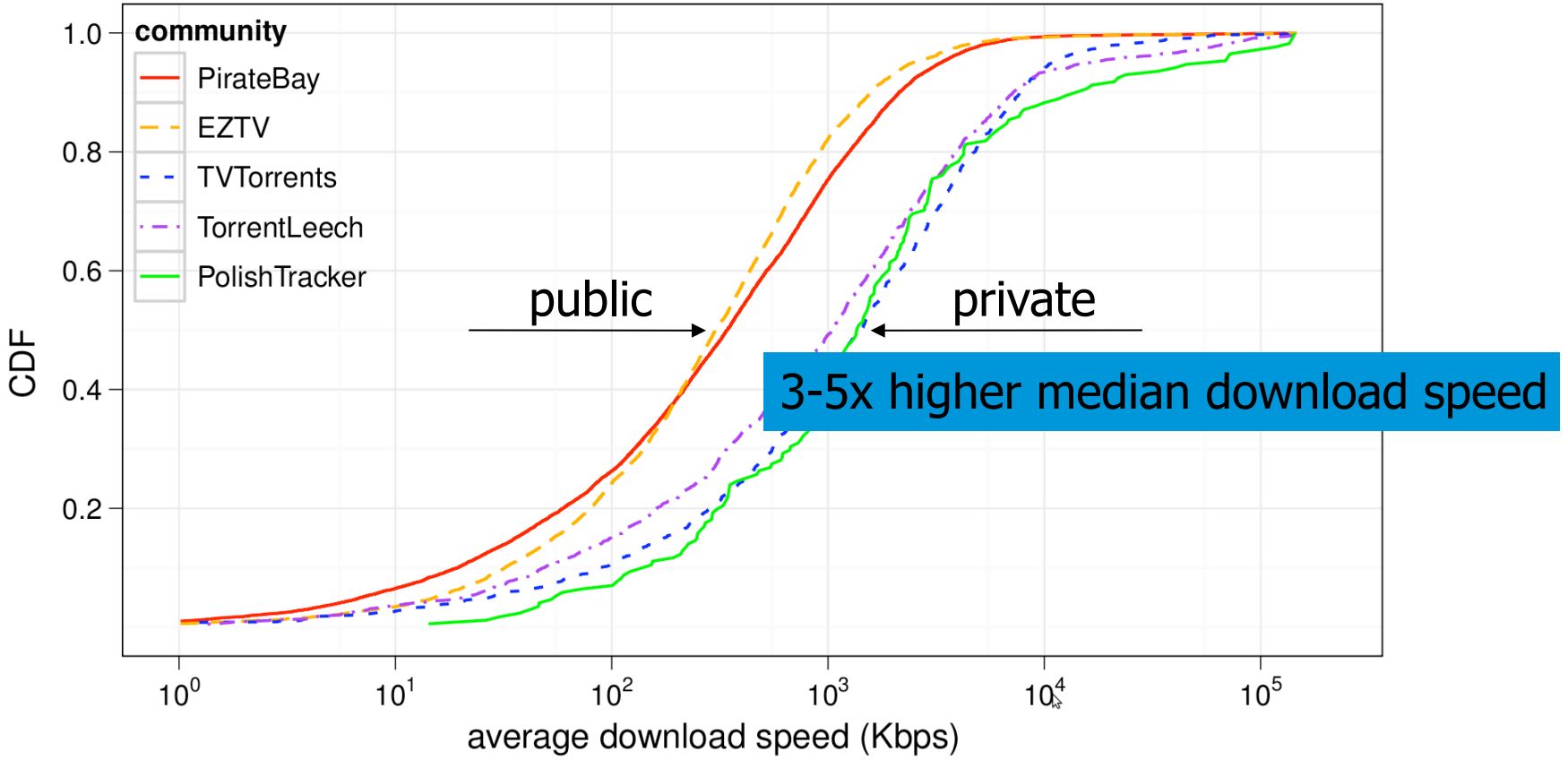
Public vs private (2/5): experimental setup



- september to december 2009
- over 500,000 users
- over 400 torrents
- over 20 million bitfields

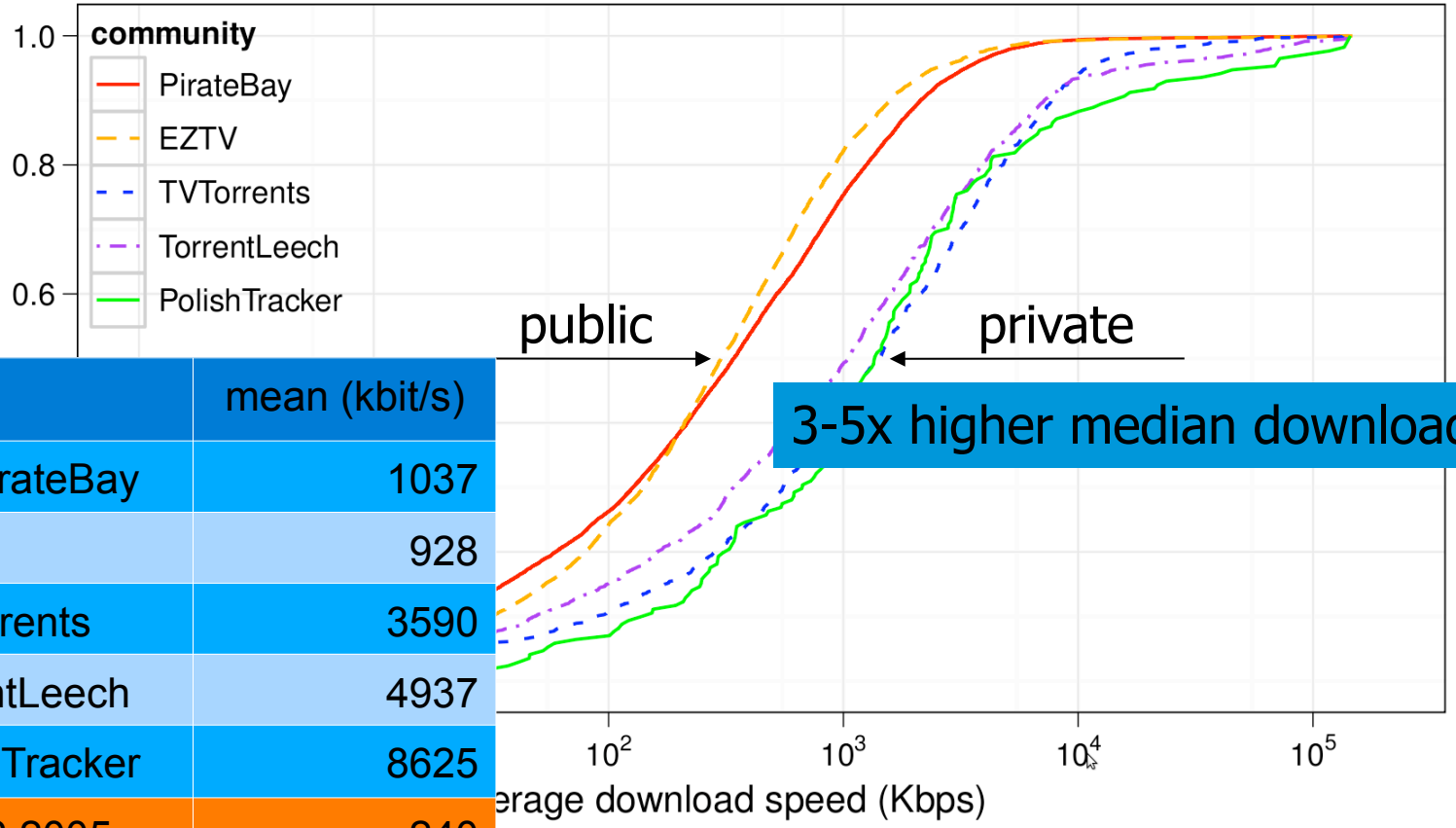
TU Delft

Public vs private (3/5): download speed



TU Delft

Public vs private (3/5): download speed

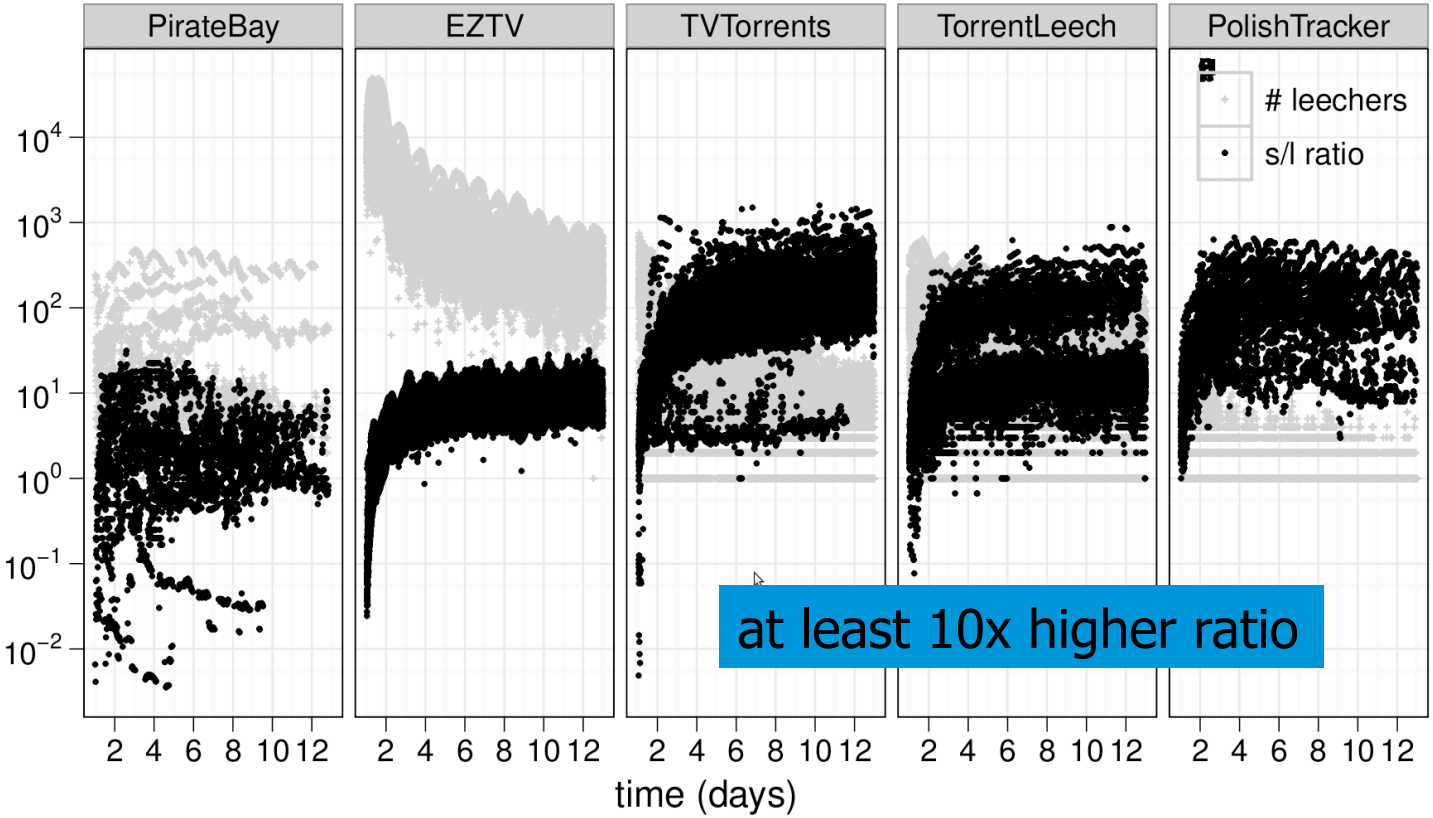


	mean (kbit/s)
ThePirateBay	1037
EZTV	928
TVTorrents	3590
TorrentLeech	4937
PolishTracker	8625
IPTPS 2005	240

3-5x higher median download speed

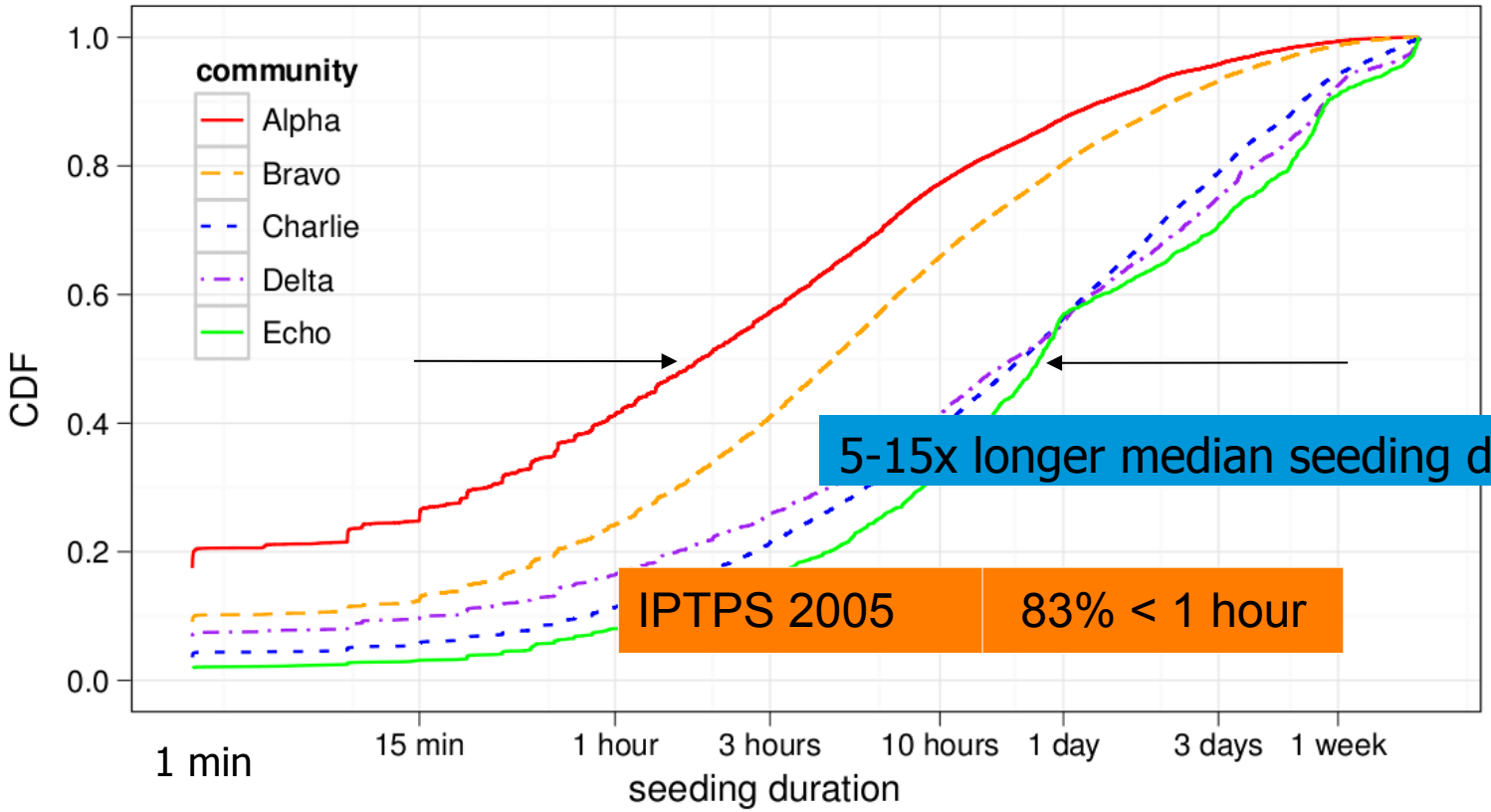
TU Delft

Public vs private (4/5): seeder/leecher ratio



TU Delft

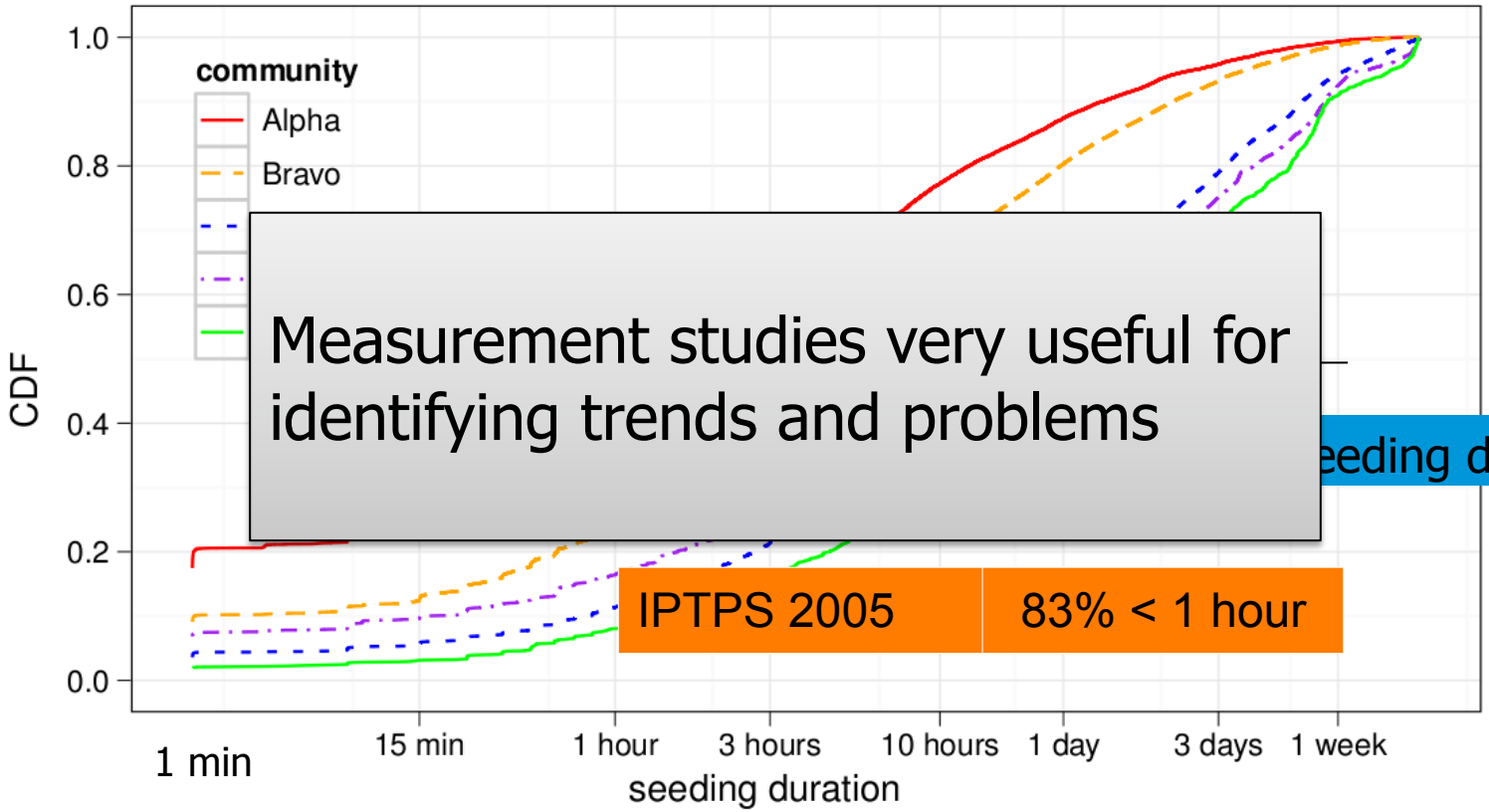
Public vs private (5/5): seeding duration



So what is still the value of tit-for-tat?

TU Delft

Public vs private (5/5): seeding duration



Measurement studies very useful for identifying trends and problems

eeding duration

IPTPS 2005 | 83% < 1 hour

So what is still the value of tit-for-tat?

TU Delft

BitTorrent: tweaks

- Many efforts to **improve/extend** BitTorrent:
  - peer exchange (PEX)
  - enforce locality
- In essence, the protocol has remained the same
- But it is very fruitful for extensions,
- And for modeling
- can better be modeled as an **auction**
- In BitTorrent, peers with similar upload capacities tend to **cluster** (=connect to each other)

TU Delft

Tribler (1/5): initial main features

Tribler

- is based on the **BitTorrent** P2P file-sharing system
- considers peers as really representing **actual users**
- adds **social-based** functionality (e.g., taste buddies)
- uses an **epidemic protocol** for peer and content discovery
- peers keep a **MegaCache** with information on the whole system
- was **first released** on 17 March 2006 (1,000,000+ downloads)
- is our **research vehicle** for P2P research

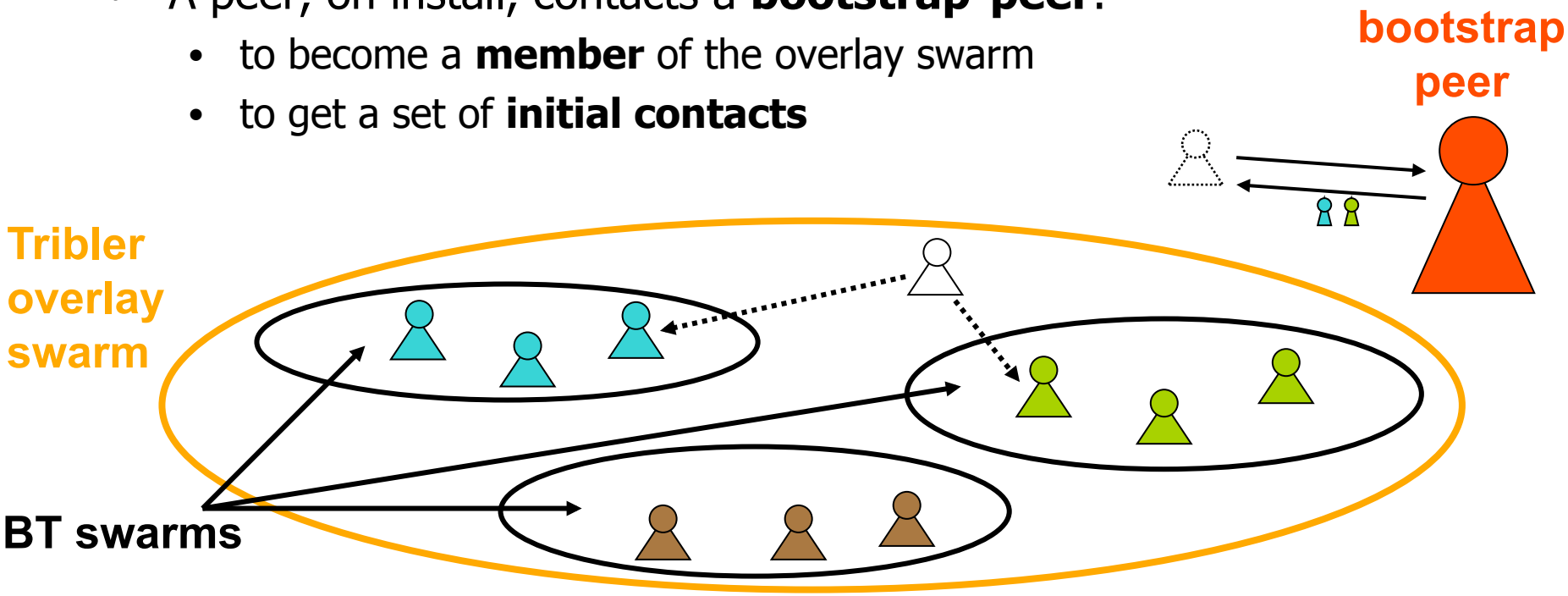
J.A. Pouwelse, P. Garbacki, A. Iosup, D.H.J. Epema, H.J. Sips, M. van Steen, et 4 al., "Tribler: A Social-Based Peer-to-Peer System," *Concurrency and Computation: Practice and Experience*, Vol. 20, pp. 127-138, 2008.

Tribler (2/5): features added later

- **Video-on-Demand**
- **Live Streaming**
- **Decentralized content discovery** through keyword search
- **Channels**: identified sets of torrents injected by a single user
- The BarterCast **reputation mechanism**
- Web-based **SwarmPlayer**

Tribler (3/5): the overlay swarm

- In Tribler we want all peers to communicate, so we define a single **overlay swarm** that contains **all peers**
- The overlay swarm is used for **decentralized** peer and content discovery
- A peer, on install, contacts a **bootstrap-peer**:
  - to become a **member** of the overlay swarm
  - to get a set of **initial contacts**

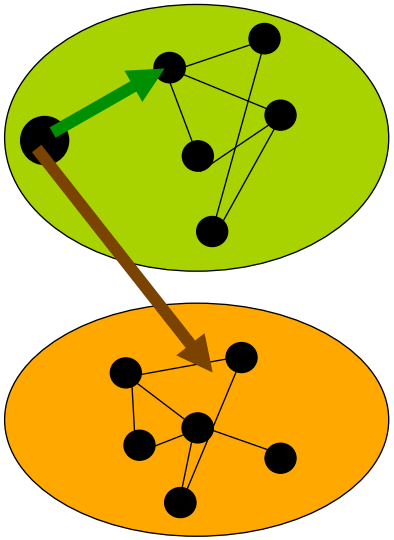


TU Delft

Tribler (4/5): BuddyCast—basic idea

- Buddycast is an **epidemic protocol** for peer and content discovery and recommendation
- Peers maintain lists of **buddies** and **random** peers
- Buddycast switches between sending a buddycast message to
  - a buddy (**exploitation**) and
  - a random peer (**exploration**)

Exploitation
finding **similar peers**
and discover their files



social network (your buddies)

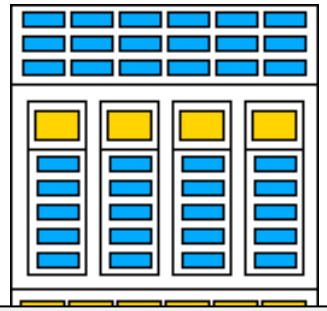
Exploration
discover **new peers**

other (random) peers

TU Delft

Tribler (5/5): BuddyCast—messages

- **Message contents**
  - 50 my preferences (torrents)
  - 10 taste buddies
    + 10 preferences per taste buddy
  - 10 random peers



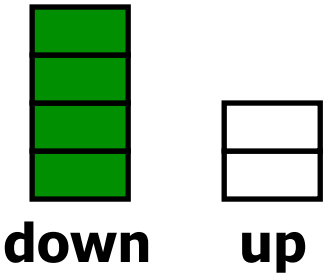
For decentralized, large-scale dynamic distributed systems, epidemic protocols are essential

- **Buddycast**:
  - every peer sends one buddycast message every 15 seconds
  - target a **buddy** or a **random peer** with some probability
  - communicating peers **merge** their buddy lists

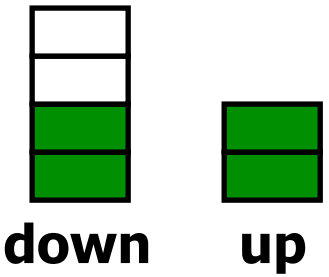
TU Delft

Collaborative downloading (1/3): motivation

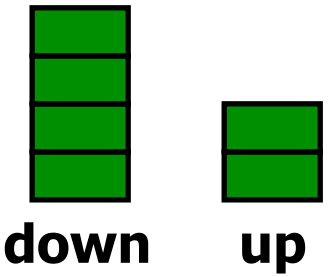
- In **early P2P systems**
  - no incentives for bandwidth sharing
  - poor utilization of upload bandwidth



- In **BitTorrent (BT)**
  - tit-for-tat enforces **intra-session** fairness
  - cannot handle **asymmetric links** very well
  - poor utilization of download bandwidth



- **2Fast: BT+collaborative downloads**
  - **cross-session** bandwidth sharing
  - full utilization of upload AND download links

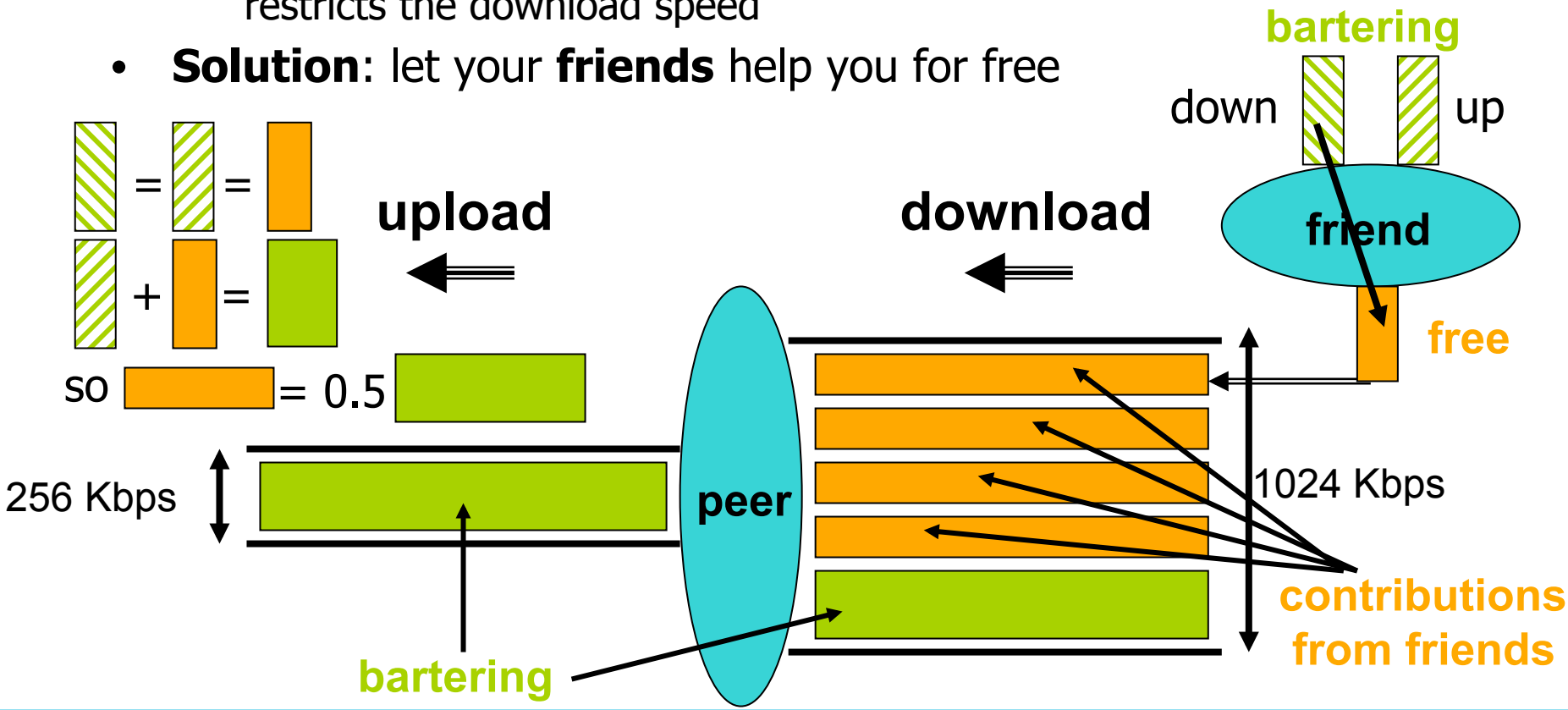


P. Garbacki, A. Iosup, D.H.J. Epema, and M. van Steen, "2Fast: Collaborative Downloads in P2P Networks," *6-th IEEE International Conference on Peer-to-Peer Computing*, 2006 (**best-paper award**).

TU Delft

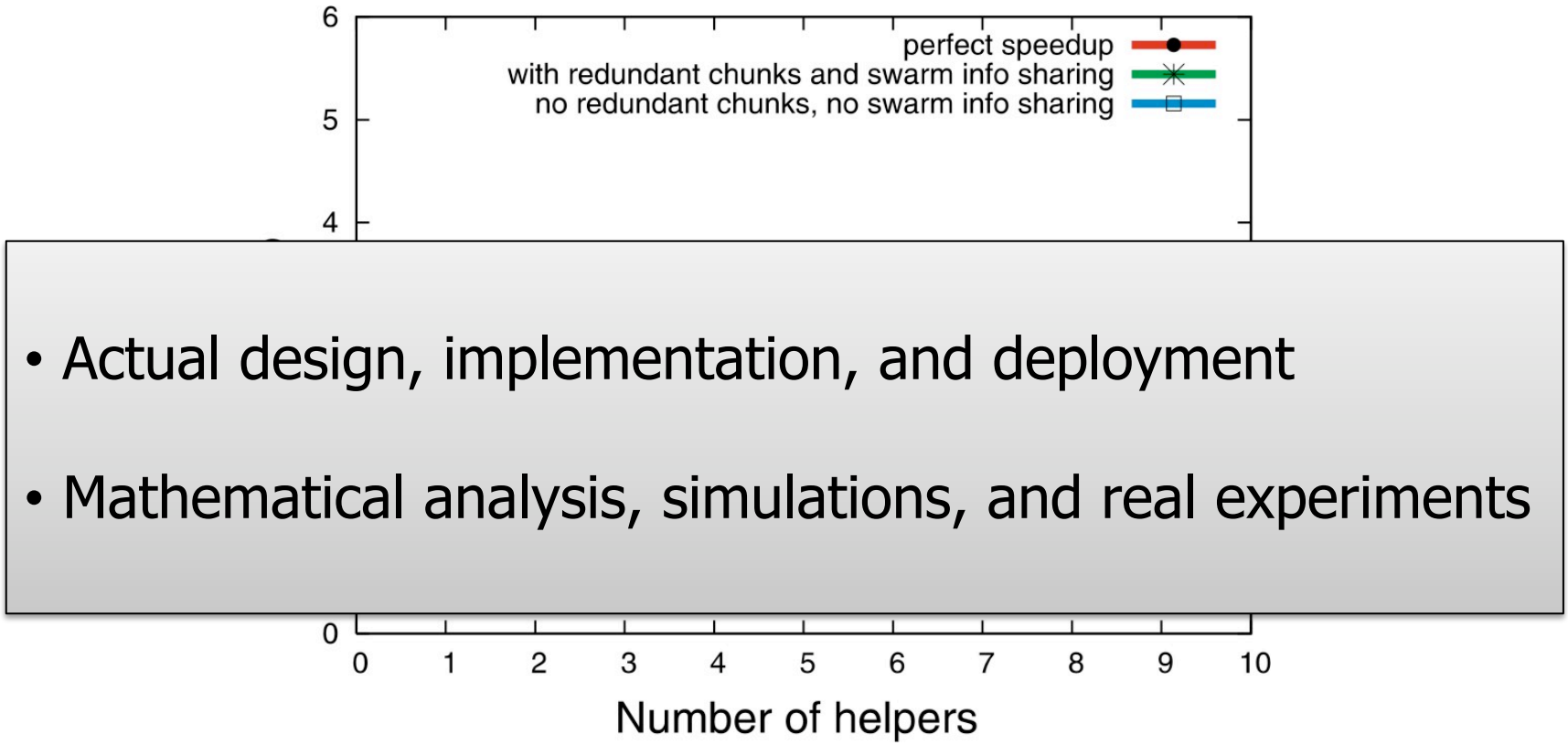
Collaborative downloading (2/3): basic idea

- **Problem**:
  - most users have **asymmetric** upload/download links
  - because of the **tit-for-tat** mechanism of Bittorrent, this restricts the download speed
- **Solution**: let your **friends** help you for free



TU Delft

Collaborative downloading (3/3): speedup

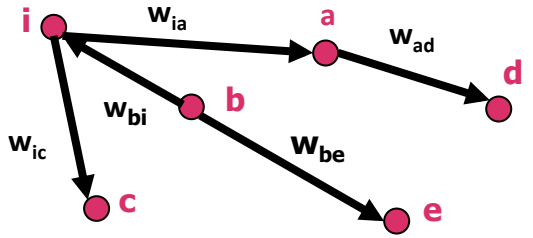


- Actual design, implementation, and deployment
- Mathematical analysis, simulations, and real experiments

TU Delft

BarterCast (1/6): basic concepts

- **Goal**: Each peer computes **locally** the **subjective reputations** of other peers in the system
- Peers maintain a **history** of their own data transfer actions + the interactions among other peers through an **epidemic protocol**
- Each peer creates a directed, weighted **local subjective graph:**
  - **vertices:** the peers whose activities are known to it
  - **weighted edges:** the amounts of the transferred data between two peers

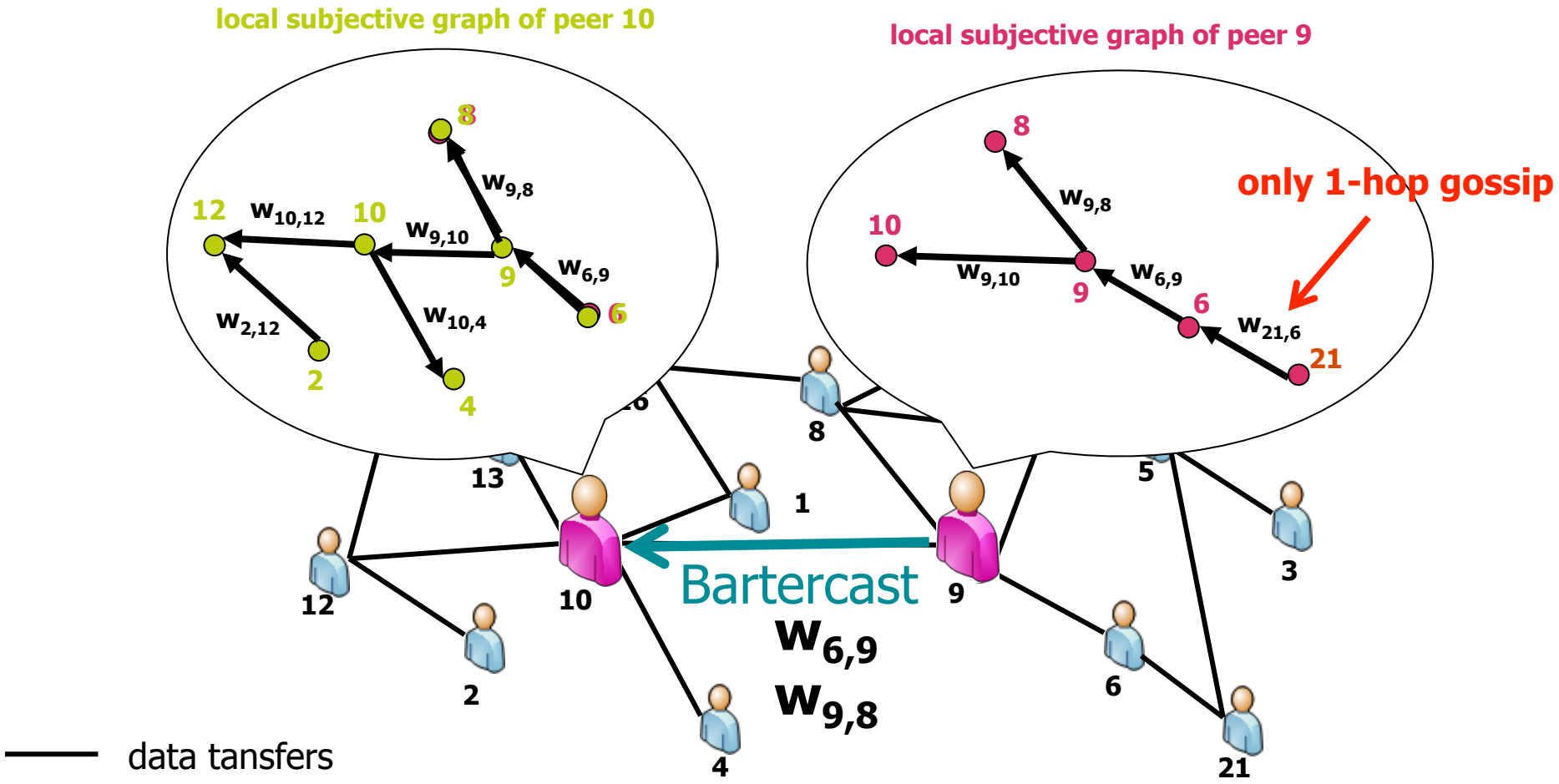


local subjective graph of peer i

M. Meulpolder, J.A. Pouwelse, D.H.J. Epema, and H.J. Sips, "BarterCast: A Practical Approach to Prevent Lazy Freeriding in P2P Networks," *Hot-P2P*, 2009.

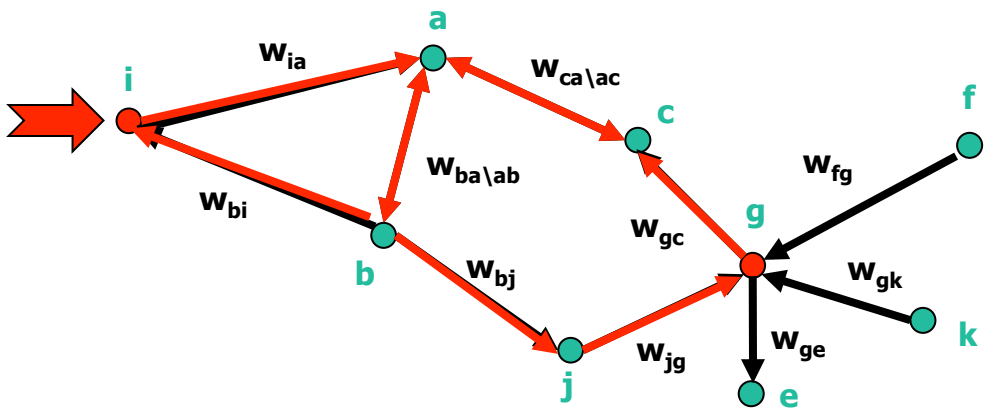
TU Delft

Bartercast (2/6): information exchange



Bartercast (3/6): computing reputations

- A **peer i** considering to upload to **peer g:**
  - considers the amounts of transferred data in its local subjective graph as **flows**
  - uses the **max-flow algorithm** to compute the total flows $\mathbf{f_{gi}}$ and $\mathbf{f_{ig}}$
  - **computes the reputation** of peer **g** as *arctan*($\mathbf{f_{gi}} - \mathbf{f_{ig}}$) / (π/2)

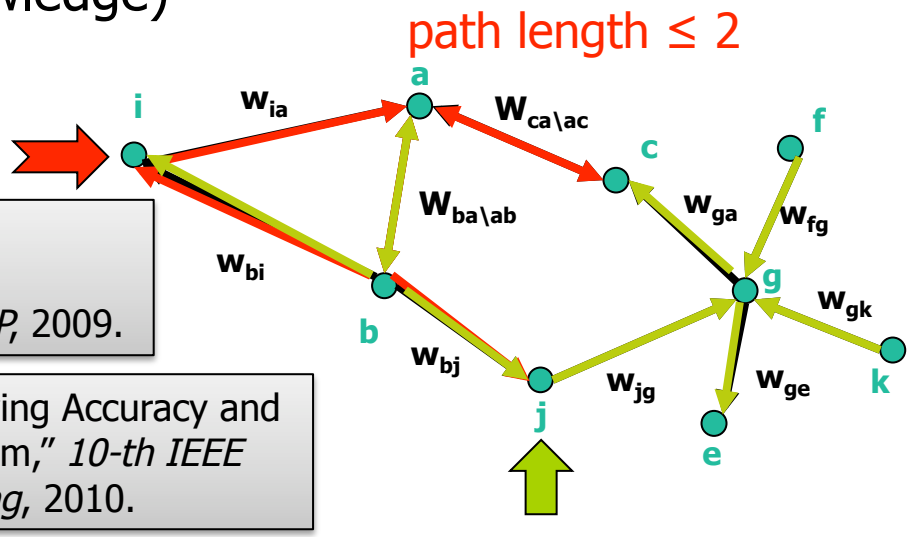


local subjective graph of peer i

TU Delft

Bartercast (4/6): three improvements

1. Restrict max-flow to a **specific number of hops** to reduce computational complexity (leads to loss of coverage)
2. Local peer may not be very central in its own subjective graph:
   - so compute maxflow starting from the **most central node** (with the highest **betweenness centrality**) in the graph
3. Use **full gossiping** instead of **1-hop gossiping** (everyone in the end has complete knowledge)



D. Gkorou, J.A. Pouwelse, D.H.J. Epema, "Betweenness Centrality Approximations for an Internet Deployed P2P Reputation System," *Hot-P2P*, 2009.

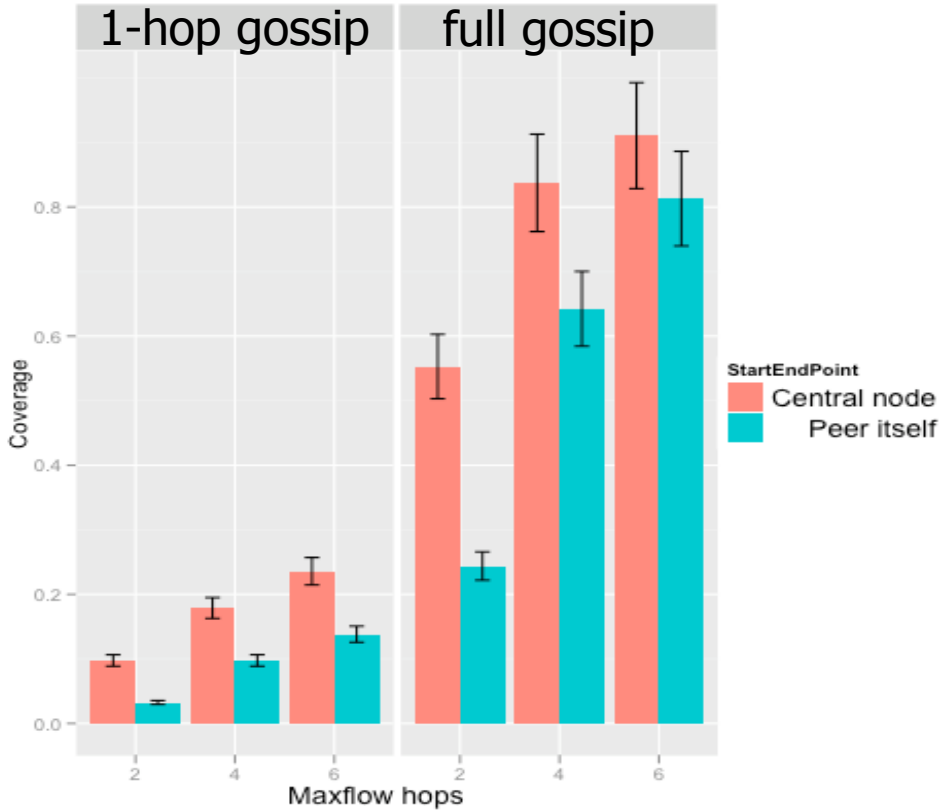
R. Delaviz, N. Andrade, and J.A. Pouwelse, "Improving Accuracy and Coverage in an Internet-Deployed Reputation System," *10-th IEEE International Conference on Peer-to-Peer Computing*, 2010.

TU Delft

Bartercast (5/6): coverage

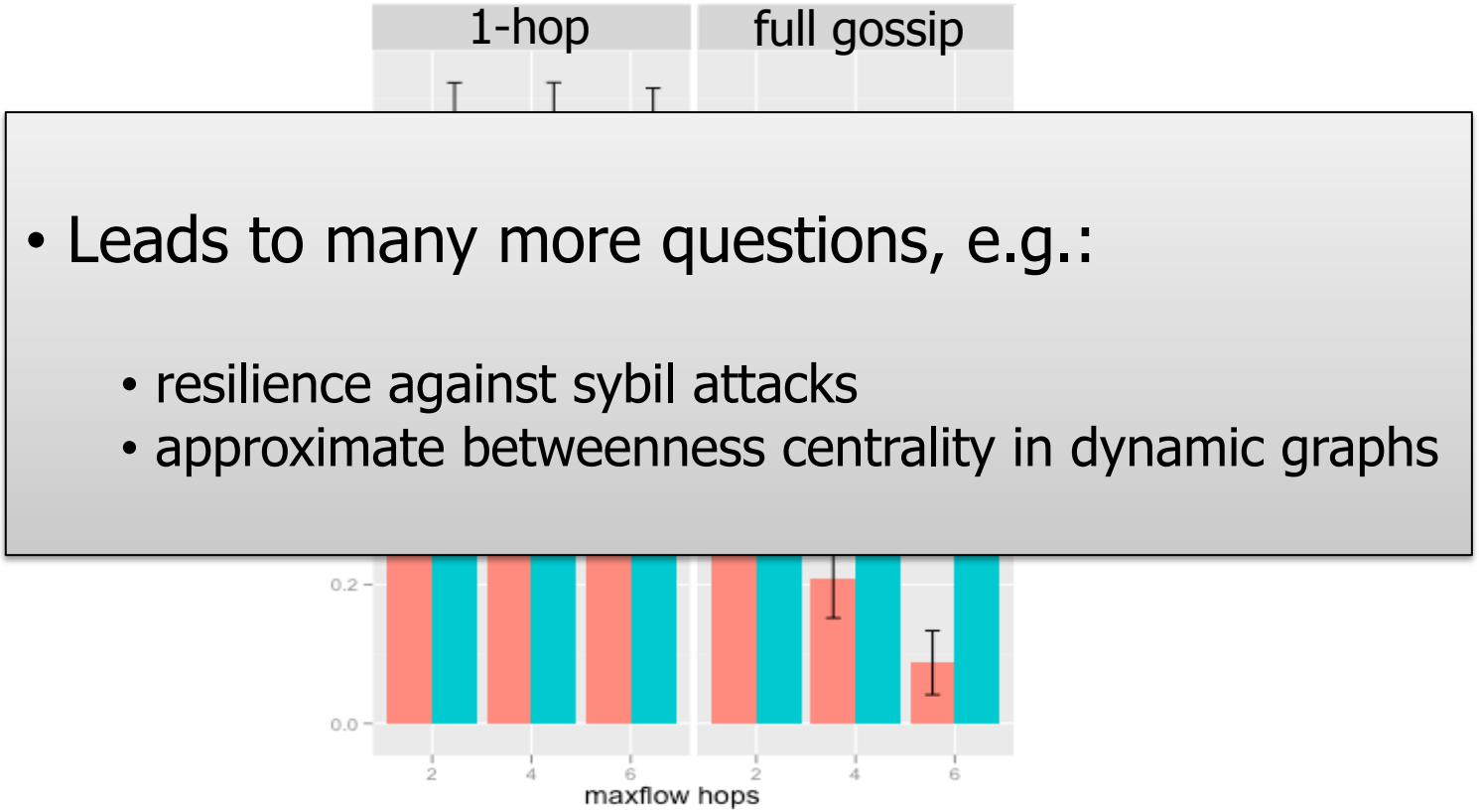
Experiments based on replaying data obtained from Tribler crawl

Coverage = average fraction of peers with non-zero reputation



TU Delft

Bartercast (6/6): median error in reputation



- Leads to many more questions, e.g.:
  - resilience against sybil attacks
  - approximate betweenness centrality in dynamic graphs

TU Delft

Tribler: valorization

- **Deployment** of Tribler:
  - Tribler SwarmPlayer is used in Wikipedia
  - Tribler SwarmPlayer deployed on the Pioneer set-top box in the NextShare platform (P2P-Next)
  - Stable Tribler user base of 5,000-10,000 user
- **Lessons** of valorization efforts:
  - *The market, even in presumably innovative sectors, is only open to **mature products and technologies***
  - ***Don't make promises** about new products or features that you are not sure you can keep*
  - *Beware of having a very broad product with many features, however interesting – rather focus on the **few unique selling points***

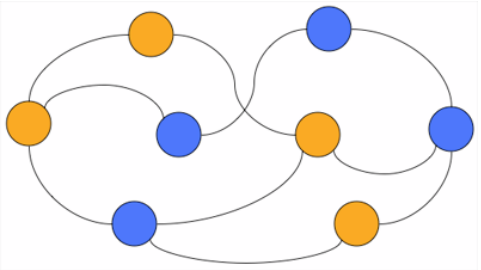
The P2P Trace Archive

- **Motivation**: limited knowledge about real P2P use
  - no P2P traces workloads available
  - no standard way to share them

- The **Peer-to-Peer Trace Archive** facilitates sharing P2P traces and associated research:
  - **understand** how real P2P systems operate
  - **address challenges** in P2P system development
  - **develop and test** solutions (simulations, experiments)

- P2PTA has unified trace format and 21 traces (BT, but also Gnutella, Skype, PP Live, …)

Boxun Zhang, Alexandru Iosup, and Dick Epema, *The Peer-to-Peer Trace Archive: Design and Comparative Trace Analysis*, Technical Report PDS-2010-003, Delft University of Technology.

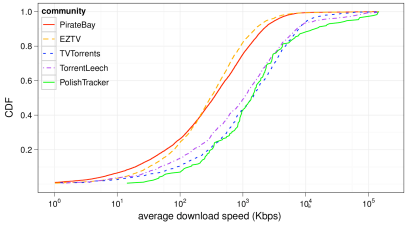


TU Delft

Current research in the Delft P2P group

- Four paper presentations here in IEEE P2P11:
  - Lucia D'Acunto et al., *Bandwidth Allocation in BitTorrent-like VoD Systems under Flashcrowds*
  - Adele Lu Jia et al., *Fast Download but Eternal Seeding: The Reward and Punishment of Sharing Ratio Enforcement*
  - Mihai Capotă et al., *Inter-swarm Resource Allocation in BitTorrent Communities*
  - Boxun Zhang et al., *Identifying, Analyzing, and Modeling Flashcrowds in BitTorrent*
- New IP multiparty transport protocol (do away with TCP)
- Reputation mechanisms
- Tribler-**G**: Playing **G**ames by means of Tribler

Research issues in P2P file sharing



1. Achieve private-like performance in public communities

2. Improve quality of offered content

3. Mobile

4. Integration with (decentralized) general online social networks

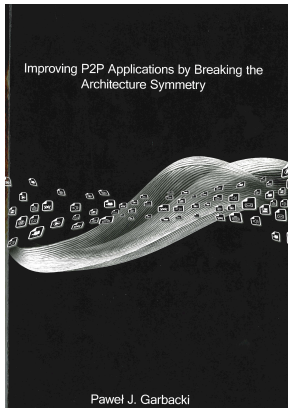
Actual implementation and deployment needed to make real progress

TU Delft

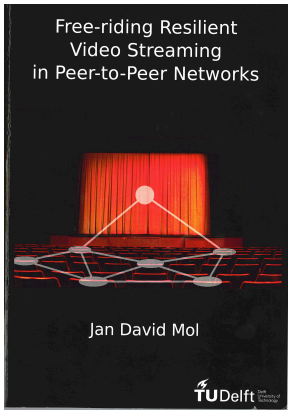
Thanks to

Johan Pouwelse

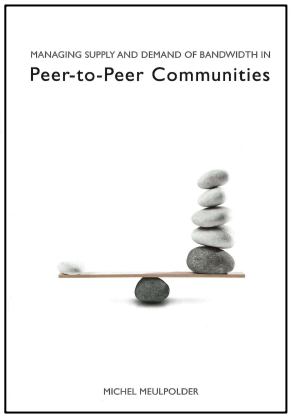
Henk Sips



Paweł Garbacki



Jan David Mol



Michel Meulpolder

And:

- Nazareno Andrade
- Arno Bakker
- Mihai Capotă
- Lucia D'Acunto
- Rahim Delaviz
- Dimitra Gkorou
- Alexandru Iosup
- Adele Lu Jia
- Riccardo Petrocco
- Rameez Rahman
- Boudewijn Schoon
- Tamas Vinko
- Boxun Zhang

TU Delft

More information

- **Publications**
  - see PDS publication database at www.pds.ewi.tudelft.nl
- **Home page**:
  - www.pds.ewi.tudelft.nl/~epema
- **Web sites:**
  - Tribler: www.tribler.org
  - P2P trace archive: p2pta.ewi.tudelft.nl
  - P2P-NEXT: p2p-next.org

TU Delft