Users want P2P, we make it work

Stanislav Shalunov <shalunov@bittorrent.com>, BitTorrent Inc

Position paper, IETF P2P Infrastructure Workshop, Boston, May 28, 2008

P2P improves the reliability of service and lowers costs for the content provider. With more content available, users consume more bits, creating more demand for Internet service. P2P should, thus, be good for content providers, users, ISPs, and vendors of network equipment. Currently, it is good for the content provider; its shortcomings are tolerated by the users because of increased content availability; and it is essentially a liability for the ISPs and, largely, a lost opportunity for the network equipment vendors.

Problems with P2P today

For the **users**, the main shortcomings of current P2P are:

1. Increased delays for other applications while P2P is in use;
2. Problems using P2P because of ISPs throttling and blocking.

When a user starts a typical implementation of BitTorrent today, multiple uploading TCP connections entirely saturate the uplink and fill the buffer in the bottleneck device, typically cable or DSL modem. This imposes an additional delay on all traffic, equal to the size of this buffer divided by the uplink bitrate. In typical home usage cases, this additional delay can range from a second to four seconds or so. An increase in RTT of this magnitude not only starves out other TCP connections, it quickly makes real-time communication, such as VoIP and games, entirely impossible.

For **ISPs**, P2P poses related problems, and more:

1. Increased delays experienced by the user running P2P would be bad enough, but on shared uplinks, as with cable, extra delays may also be experienced by others in the same neighborhood, who derive no benefit from P2P;
2. Increased consumption of bits by users on flat monthly pricing plans increases transit costs and necessitates earlier upgrades of internal networks without additional revenues.

For **content providers**, while P2P brings substantial benefits, the benefits could be larger if the ISPs did not consider P2P a problem and did not take technical measures against it.

For network equipment **vendors**, P2P has been an essentially lost opportunity. Equipment was created to limit the amount of P2P traffic. This equipment's purpose is to act against a service that users, who ultimately pay, want. It also appears to be against regulation for some ISPs. Instead of making devices that delay, drop, and forge users' packets, vendors could be making caches and trackers, which would enable a useful service.

P2P solution space

The problems experienced by the users and the ISPs can be addressed in many ways, most beneficial of which are:

1. Caching;
2. Smarter peer selection;
3. Better congestion control.

Scavenger service class and routers providing end nodes with **more explicit information about congestion** than loss and delay may also be potentially worth considering. They are sound conceptually, but perhaps less practical in the short term because they require modifications in the device at the head of the bottleneck link, and many of these devices, such as cable and DSL modems, are hard to modify *en masse*.

Caching would allow the ISPs to

- lower transit costs by reducing the number of times each file is downloaded *and uploaded;*
- lower internal link use and delay upgrades by distributing the caches closer to users;
- lower uplink use on last-mile connections by letting peers download quickly from the cache and leave the swarm.

For users, caching would mean more reliable and faster downloads.

For network equipment vendors, caching would create new product opportunities.

Caching radically reduces the total number of bit-miles[1] consumed by P2P cost-effectively and is, therefore, a critical piece of any solution.

The most natural way to implement caching would be to run a special sort of a well-provisioned peer in the middle of the network. This cache might be an appliance or might even reside inside a router. Its function does not require any tight integration with packet forwarding or routing, although some forms of integration might be useful: e.g., quickly learning what torrents users download. Some way to find caches is required, but no further modifications to client software are necessary to prefer caches to other peers as long as caches can provide substantially faster download rates than peers behind narrow uplinks.

Cache sizes needed to provide significant hit rates are practical, and depending on the desired hit rate and the data set, have been estimated to range from a bare minimum of 1 terabyte to 100 terabytes in aggregate. The top end of the range is entirely practical to achieve in aggregate and the bottom end of the range can fit into a single device.

Smarter peer selection is required for cache discovery, but could extend further. Today, most BitTorrent clients try peers essentially randomly. This ensures nearly optimal

---

[1] Bit-miles, the product of the number of bits transferred by the distance, is a very rough proxy of cost. Better and more precise metrics that take into account path cost could be used in an actual specification.

dissipation of pieces and great performance of the rarest-first algorithm, but provides nearly *pessimal* network efficiency in terms of bit-miles consumed. Improvements to peer selection could include, starting from the most trivial:

- preferring peers from the local AS or peers with many leading bits of IP address coinciding with the local address;
- preferring peers from a given list of ASes or from a given list of IP prefixes;
- communicating a notion of cost associated with ASes or IP prefixes to the peers.

Selecting peers in a way more informed than random would reduce the number of bit-miles consumed and, perhaps even more importantly, direct the traffic towards links that the ISP prefers. Any solution that includes caching requires a rudimentary form of smarter peer selection — cache discovery, but smarter peer selection is useful regardless of caching.

In one small example, we collected the IP numbers of peers for four most popular torrents on the most popular open tracker. Swarm size ranged from 2,000 to 10,000 peers. For each of the four swarms, about half of the peers were in just 40 ASes. Of these, 22 ASes appeared in the top 40 in all four swarms. We expect that any AS with 20 peers or more in a given swarm could essentially remove interdomain traffic with smarter peer selection even before caching. In this example, 57% of peers were in ASes with 20 peers or more. While in general we expect this fraction to be lower on average than for these four very popular swarms, interdomain traffic could be reduced as soon as there are 2 peers in an AS.

Better congestion control goes well beyond TCP-friendliness and fairness per flow. Delay experienced by other apps must be minimized. This is a new design goal, and not a goal of TCP's AIMD congestion control. AIMD must experience loss before backing off. Typical cable and DSL modems at heads of bottleneck links going up from users' homes use FIFO queuing without AQM. Thus, on these links, TCP must fill the buffers. These buffers happen to be provisioned overly generously, in the seconds. Because of the huge number of deployed devices with these buffers, replacing or upgrading them is not practical.

BitTorrent DNA, our product for P2P-assisted content delivery, implements a novel form of end-to-end congestion control that, in addition to being TCP-friendly in the usual sense, maintains a low queuing delay on the bottleneck link.

We have tested this congestion control in a variety of circumstances and have deployed it in the wild. About 7M users have by now used this congestion control mechanism and data were collected about the usage.

In one example, BitTorrent DNA was used to download and seed game updates while an online multiplayer game was being played. With TCP used for transport the way it is usually used in BitTorrent, ping times shot up to 2000 milliseconds and beyond and stayed there while seeding. With the novel congestion control, ping times were in the 50-100 millisecond range, while the upload rate remained essentially unchanged and, therefore, close to the uplink capacity minus game traffic. This is an improvement from a network nonfunctional by modern standard where users expect interactive apps to work, to a network essentially undistinguishable from absence of P2P.

BitTorrent Inc is interested in working with the IETF, congestion control researchers, network equipment vendors, and other P2P vendors to standardize an approach to congestion control that would enable other P2P applications to achieve the same objective of not increasing delay significantly for other apps.

The role of the IETF

The IETF could help solve the problem P2P apps currently pose for ISPs and users in these ways:

1. Standardize a mechanism for BitTorrent cache discovery. This is the lowest-hanging fruit.
2. Standardize a mechanism for conveying information about the network to P2P software, which would enable smarter peer selection, and standardize ways to use this information. Cache discovery could be a part of this potentially more general mechanism or could be separate.
3. Document an experimental approach to end-to-end congestion control that meets the objective of not increasing delay substantially even if bottleneck device buffers are large relative to the bitrate.
4. Publish an informational RFC that documents caching, smarter peer selection, and better congestion control as the best current practices in solving problems caused by P2P.

We look forward to contributing to these goals within the IETF. While we do not have access to network use statistics at relevant ISPs, it is our understanding that BitTorrent comprises a substantial fraction of total P2P traffic. Our µTorrent client, with 35M active installs, is one of the most popular and probably the most popular is the U.S. We're looking forward to implementing standard ways of making it work better for the users and the ISPs and we believe that the vendors of other popular BitTorrent clients would follow because this would improve the experience of the users of their clients.