Network-Friendly Peer-to-Peer Services

Michael Merritt (AT&T Labs Research), Doug Pasko (Verizon), Laird Popkin (Pando)

MOTIVATION:

The success of the Internet and the supporting community of ISPs, content providers, hardware vendors and application developers has depended heavily upon the voluntary adoption of network-friendly protocols such as TCP. This voluntary collaboration between application and network service providers has been a remarkable enabler of today's rich Internet ecosystem.

Today this infrastructure is supporting a rapidly growing quantity of multimedia traffic. The increasing popularity of such content and its expected migration to higher definition formats presages an explosion in total multimedia data volumes. As libraries and access bandwidth requirements grow to meet this demand, content providers are exploring peer-to-peer (P2P) content distribution architectures that scale content-delivery system capacity and access bandwidth with growing demand, by utilizing consumer storage and bandwidth as peers work together to cooperatively download and share popular content. While they can be effective in reducing server capital and access expense, today these architectures are wasteful of network capacity compared to classical CDN architectures--these classical CDN architectures are designed to simultaneously maximize latency-dependent performance and network efficiency by serving content from a site close to the requesting consumer.

The radically larger number of nodes in a P2P network, as compared to a traditional CDN, allow for even greater performance and scalability of data delivery. However, there is a challenge, in that the distributed and dynamic nature of P2P architectures makes the coordination required to provide for network and performance efficiencies a challenge. But without this efficiency, are P2P architectures merely a means of shifting CDN costs to the network provider? And must they depend upon complex adaptive mechanisms and/or duplicative Internet instrumentation (with multiple P2Ps reverse-engineering the same ISPs) to guide peer selection for efficient network utilization?

Recent theoretical work from Yale and proof-of-concept trials of their scheme by Pando, Verizon, and Telefonica demonstrate that cooperative mechanisms can indeed guide peer selection to be network-aware, resulting in simultaneous improvements in network efficiency and application performance. In the tradition of TCP, these results strongly suggest that there is a new opportunity for collaborative mechanisms to support global efficiencies in multimedia content delivery. But this efficiency can best be achieved if common standards can be established on an industry-led, collaborative basis so that P2P providers and ISPs can effectively and efficiently coordinate.

While this work has focused on proximity-based peer selection, standards should generally afford an opportunity for these parties to implement a broad range of policies for adapting P2P traffic and ISP resources, preserving application performance and safe co-existence with other network services.

Another major concern of ISPs is the risk of adverse impact of P2P-CDN Clients' operations on the performance of subscriber's computing endpoints (PCs/Laptops) and Home Network elements (Residential Gateways, Home Routers). As P2P-CDNs proliferate in the current unmanaged fashion there will likely be multiple P2P Clients concurrently running on each of several active PCs/Laptops on a Home Networks, potentially consuming PC/Laptop/RG/Home Router resources and thus causing performance degradation of other networked applications (Web surfing, VoIP, IPTV) which could be perceived by subscribers as ISP's network/service problems – resulting in costly Customer Care costs for ISPs. This points to the need for home-computing as well as home-network resource management standards and tools.

APPLICATION MOTIVATION

There are two general motivations for application developers to work with ISPs to optimize P2P traffic: to improve delivery speed and performance to end users, and to encourage ISPs to work collaboratively on shared solutions for managing the bandwidth demands of P2P. In addition, there is the opportunity to mitigate the potential negative impacts of P2P applications on end-users' other networked applications.

All P2P networks attempt to maximize download speed to their users. P2P software generally includes strategies for optimizing data throughput. If working with ISPs will allow P2P networks to provide superior performance, this is a strong incentive toward partnership. As some P2P platforms start implementing open standards that provide superior performance, competitive pressures will drive other P2P companies to follow suit.

P2P applications are extremely popular with users, which leads to a large and growing volume of data being exchanged using P2P protocols. In order to balance traffic and limit the infrastructure costs of supporting this demand, it has been reported that some ISPs have applied traffic shaping technologies that focus on high bandwidth-consuming P2P communications. This could lead to a counter-productive "cat and mouse" contest between P2P applications and ISPs, where ISPs deploy a management technology, then P2P applications deploy counter-measures (e.g. encryption) to avoid that management, the ISPs deploy new measures, and so on. This could result in both P2P networks and ISPs expending significant resources in efforts that are not ultimately productive. If P2P networks instead can work with ISPs to reduce the resource costs of P2P applications to ISPs, this could in turn lessen the need for ISPs to deploy technologies that could have the effect of limiting P2P application performance in order to ensure overall customer performance. Such an outcome ultimately would be most beneficial to the users that are the common customers of the ISPs and the P2P platforms.

APPLICATION REQUIREMENTS AND REQUESTS

P2P applications could benefit from information such as:

- If a P2P platform could accurately identify network capabilities (e.g. Dial, DSL, FTTH, Cable, Business-Grade). With this information, the application could make more intelligent network capacity decisions, allowing P2P networks to shift load away from links that cannot support it.
- P2P caching servers can potentially reduce the network resource consumption of P2P applications. If ISPs deploy P2P caching servers, a standard means of locating and working with such servers would allow P2P platforms to most efficiently take advantage of them, reducing infrastructure costs to ISPs.
- If ISPs can communicate their network policies (e.g. pricing, quotas, current consumption), P2P users can make informed choices about participation in P2P networks, eliminating a current source of uncertainty in the marketplace.
- Can ISPs tell P2P applications about optimal low-latency routes between peers? These could be used for control messaging, video game interaction, VOIP, etc.
- Can ISPs tell P2P applications about optimal high-throughput/low cost routes between peers? These could be used for bulk data delivery.
- How can P2P applications reliably determine the network distance between peers within an ISP? Many ISPs operate infrastructures in which 'traceroute' does not return meaningful information, so an alternative mechanism would be useful.
- What routing policies should P2P applications be aware of? Which types of packets or protocols are routed or dropped, based on what rules?

- How should P2P applications mark traffic to provide an appropriate QoS? If P2P applications mark their traffic as "bulk data," how would that be handled by ISPs?
- Do ISPs prefer any particular port range or protocol type over others?
- What kind of P2P traffic or activity are ISPs most concerned about? Are there preferable alternatives that P2P applications should use instead?
- What does a P2P system need to do to be considered a network-aware or networked-optimized?

Most P2P networks apply a variety of proprietary techniques in an attempt to determine the answers to many of the above questions. For example, the BitTorrent protocol optimizes data sources, so peers try to download data from peers that provide the fastest data supply over time. So while a peer may start with a small set of known peers, downloading slowly, as peers gradually discover new peers and test communications with them they gradually find better and better data sources, leading to the gradually growing "torrent" of data after which the protocol is named.

But if the ISPs could provide knowledge, this would be more accurate than application reverse-engineered knowledge. It would also be more timely. For example, a p2p protocol that optimizes data flow between known peers based on observed throughput cannot fully optimize data flow between all peers in a large swarm until all peers have exchanged data with all other peers. But if ISPs can provide additional guidance to assist P2P networks in identifying likely good peers (e.g. close to each other in the network) that could allow the P2P network to immediately establish high-quality connections between peers.

In addition, the ISPs could benefit from providing information to the P2P networks that cannot be determined by network inspection, because they can include additional useful information that might not otherwise be apparent to the P2P network. For example, a P2P network node in NY might see a fast peer in Hong Kong, and use it heavily, at high cost to the ISP, because the application does not have the information that could allow it to differentiate between a fast peer that is a neighbor and a fast peer that is across the Pacific Ocean. While both are fast data sources, so the P2P network does not (in principle) care which it uses, the difference between the two is extremely important to the ISP.

Applications also have requirements for the standard in this area, including:
- Implementation effort should be minimized.
- The performance impact of the standard should generally be positive, and should never be worse than not using it.
- The standard should protect the privacy of the P2P network and its users.
- The standard should be optional, providing the P2P network with operational independence. That is, if the service becomes unavailable, the network should be no worse off than it is currently.

ENTITIES

The entities that relate include:

P2P Clients: these are the programs that run on user computers, communicating directly with each other to coordinate and transfer data. There are a wide variety of P2P clients, utilizing a wide range of protocols and user interaction models (e.g. live streaming, stream on demand, VOIP, download on demand, subscription). Any standard should support as wide a range of P2P clients as possible.

P2P Control Servers: for some P2P networks (e.g. BitTorrent, eDonkey) there are servers that coordinate communications between the P2P Clients. Note that some P2P networks (e.g.

BitTorrent with DHT's) consist only of P2P Clients, and do not have centralized control servers. Any standard should ideally support both P2P networks with control servers and without.

Internet Service Providers: The ISPs operate networks that connect end users (and thus P2P Clients) to the rest of the internet.

Proximity Servers: The ISP's topological data could allow P2P networks access to guidance that allows them to optimize network traffic. This information could be exposed by proximity servers. The ISP's topology and policy data could be processed and collected into a centralized proximity server, operated by a trusted third party (similar to how ARIN administers IP addresses) or individual ISPs may wish to operate their own proximity servers. Indeed, they may choose to abstract their topology almost completely, and implement "policy servers" that provide very coarse topology-dependent guidance, but afford the opportunity to impact major traffic flows under periods of peak demand and/or during catastrophic events. The tradeoffs between centralized and distributed proximity servers should be explored further.

ARCHITECTURE

Proximity-based peer selection, or more general network-aware peer selection is driven by topology and policy data to produce a proximity database. The state of the art today requires P2P applications to inefficiently query the network itself to collect network performance and connectivity information to manage application performance.

Proximity databases can be built cooperatively by clients, either implicitly by performance-driven measurements, or explicitly, in a fully distributed architecture. Alternatively, detailed proximity databases…even taking into account anticipated demands…are maintained by individual ISPs and could be queried by P2P entities via a standard interface. Finally, proximity servers could be supported as a third party service, potentially aggregating data provided by cooperative ISPs together with reverse-engineered topology and measurement data. In addition, any of these alternatives can be extended to include policy and business information to guide peer selection.

In turn, proximity routing (or more generally, "network-friendly" routing) can be done by clients using their own topology database, can be done by clients in consultation with one or more proximity servers, or done by trackers in consultation with one or more proximity servers.

TRADEOFFS

A successful architecture and accompanying standard for P2P optimization should include a reasonable tradeoff between implementation complexity and optimization potential. In the traditions of the Internet, this collaborative technology effort should provide value to all parties involved. The objectives of optimizing carrier resources while simultaneously increasing P2P performance should be considered equally valuable.

At one extreme, providing a real-time view of an ISP's network topology, routing tables and current link loads could provide the maximum possible potential for optimization. The other extreme can best be characterized as the current condition. ISPs provide little or no visibility into their network topology and P2P companies are left to attempt reverse engineering. The middle ground has a number of tradeoffs that must be considered.

Tradeoff: Carrier Security vs. Enough Detail for Optimization

While the technology for sharing topological data may be straightforward, the business drivers and security concerns can be more complicated. An appropriate standard should provide the flexibility for a carrier to summarize or even abstract the topological data it provides for external use. As was seen in the recent field trial of the Yale pre-standard implementation, simple abstracted topology data was very effective in providing optimization. Allowing for various levels of topology detail to combine and form an effective traffic direction protocol will be a necessary result.

Tradeoff: Real-time Capacity Information vs. Complexity

The possible inclusion of capacity information within the optimization directives should be addressed. In the simplest of models, a static relationship between nodes might be specified. This relationship could include a summary level of carrier desires for traffic flow. In the most complex implementation, a real time feed of individual circuit capacity could provide the maximum opportunity to direct P2P peer selection. This area again has a mixture of technology issues and business issues. The complexity for a P2P provider to accept such real-time data and use it effectively must be balanced with having enough detail to make an optimal decision. As with the Security/Detail tradeoff, the granularity of dynamic updates could vary from constant incremental network-wide link performance, to occasional but real-time revisions to deal with catastrophic internal or intra-ISP outages.

Tradeoff: Carrier Preference vs. User Experience

Existing mechanisms used by P2P providers to provide the optimal user experience must also be balanced with the new abilities for carriers to offer direction to manage network capacity. The additional feature-sets that come as a result of the cooperative efforts with carriers should be integrated and balanced with existing user experience management techniques already being employed. These new features should not be considered as a replacement for peer selection criteria, but rather an augmentation. Finding a balance between them is a requirement. In some ways this mirrors the BGP route selection decision sequence to break ties. Tradeoffs in this space must be transparent and the root cause of performance impairment easily diagnosable…one ISP concern about P2P applications is the potential for increased customer care costs.

Tradeoff: Centralization vs. Distributed

Amongst the above tradeoff topics is the variable of topological data warehousing. This tradeoff topic area can greatly impact the complexity issues as well as carrier security. If the data were to be centrally stored with a trusted 3rd party (ARIN and/or RIPE for example), the complexity for implementation amongst P2P providers might be significantly reduced. However, centrally stored data may re-raise carrier security concerns.

These topics are only some of the areas already being discussed within the DCIA led P4P Working Group and sub-committees. We would like to bring these topics to the IETF for broader discussion and for working within the auspices of the IETF to examine and standardize protocols in this area.