

# Performance Evaluation of the Advanced Network Tracker for BitTorrent

Shakir James, [sjames@wustl.edu](mailto:sjames@wustl.edu) (A project report written under the guidance of [Prof. Raj Jain](#))



---

## Abstract:

BitTorrent, a popular Peer-to-Peer (P2P) protocol, has recently engendered considerable controversy. Many P2P applications implement a variant of the protocol due to its cost-effective scalability. However, for Internet Service Providers (ISPs) the protocol is notorious for its high network cost. To reduce these costs, ISPs have deployed network devices to detect and throttle BitTorrent traffic. In response, application developers have begun encrypting the protocol's traffic to avoid ISP-detection. These actions perpetuate a "cat and mouse" game between ISPs and developers. Recent proposals to reduce BitTorrent's network cost involve either changing the protocol or deploying ISP-managed services. These proposals require broad adoption by developers to significantly benefit ISPs. In this paper, we propose a solution that is universally applicable to all BitTorrent clients and provides savings for ISPs out of the box.

We have developed a high performance network device, called the Advanced Network Tracker (ANT), which reduces BitTorrent network cost without changing the protocol. Moreover, ANT requires no explicit communication between client applications and ISPs. It inspects packets on the wire to identify BitTorrent traffic to discover local peers sharing similar files. To reduce network costs, ANT localizes traffic to an ISP's network by informing local peers about each other. In this work, we implement a working prototype of ANT using Network Processors (NP) for high throughput packet processing. In addition, we evaluate our implementation on the Open Network Lab's [\[ONL\]](#) NP-based programmable routers. Our evaluation shows ANT has a considerable impact on cross-ISP traffic. And equally important, it has a negligible impact on client performance.

---

## Keywords:

Network architecture, network processors, traffic management, performance evaluation, peer-to-peer (P2P), BitTorrent

---

## Table of Contents

- [1. Introduction](#)
- [2. Background](#)
- [3. Design](#)
  - [3.1 Component Architecture](#)
  - [3.2 Packet Filter](#)
  - [3.3 Local Tracker](#)
  - [3.4 Proxy Cache](#)

## [4. Implementation](#)

### [4.1 Network Processors](#)

### [4.2 Open Network Lab](#)

### [4.3 ONL Network Processor Router](#)

### [4.4 ANT Plugin](#)

## [5. Evaluation](#)

### [5.1 Goal and Metrics](#)

### [5.2 Factors](#)

### [5.3 Evaluation technique and Experimental Design](#)

### [5.4 Results](#)

## [6. Related Work](#)

## [7. Future Work](#)

## [8. Concluding Remarks](#)

## [Acknowledgements](#)

## [References](#)

## [List of Acronyms](#)

---

# 1. Introduction

Peer-to-Peer (P2P) systems promise cost-effective file distribution for content providers on the Internet. In a P2P system, each peer downloads as well as uploads data to other peers. Thus, P2P systems are self-scaling because their bandwidth capacity increases as the number of peers increase. Compared to the client-server model or content distribution networks, P2P systems are cheaper for content providers since they do not require an investment in dedicated servers. Furthermore, P2P systems are better able to recover from failures since each peer holds some piece of the file. This paper focuses on the BitTorrent P2P protocol. Researchers have shown that the BitTorrent protocol is self-scaling [Yang04][Qiu04], robust [Legout06], and provides near optimal performance in terms of peer download time [Bharambe06].

Although the BitTorrent protocol provides cheap scalability for content providers, it is expensive for Internet Service Providers (ISPs). The protocol is oblivious to the underlying network topology, so its traffic traverse routes without concern for ISP economics. In fact, the protocol increases traffic on costly, cross-ISP link [Bindal06]. At the same time, P2P applications like BitTorrent increase revenue for ISPs because they encourage more broadband subscriptions. As a result, ISPs are left in a quandary: throttle BitTorrent traffic to reduce network cost at the risk of losing revenue from broadband subscriptions. ISPs have deployed traffic shaping devices [P-Cube][Sandvine] to "throttle" or limit BitTorrent traffic. Although the legality of traffic shaping is debatable, the practice degrades P2P application performance and can turn away potential customers.

Instead of reducing BitTorrent network cost, traffic-shaping devices have sparked an arms race between BitTorrent developers and ISPs. To evade identification by the first generation of these devices [P-Cube], some BitTorrent applications added header encryption to peer-to-peer messages. To counter, ISPs introduced new devices [Sandvine] that targeted other protocol (peer-to-tracker) messages. Recently there have been signs of an armistice, as ISPs have begun supporting initiatives to cooperate with developers to reduce BitTorrent network cost [Xie08]. Some authors [Bindal06][Xie08][Choffnes08] have proposed changes to the de facto protocol. Unfortunately, these collaborative initiatives have not been widely adopted.

As a new solution, we propose a network device, called the Advanced Network Tracker (ANT), which ISPs install on their network beside edge routers. What separates ANT from other proposals, is that ANT works with all BitTorrent clients and requires neither protocol changes nor direct application-ISP communication. Thus, ANT's novel contribution is solving ISPs' P2P quandary without requiring any extensions to the

protocol. To reduce network costs, ANT keeps BitTorrent traffic within an ISP's local network.

We have implemented ANT on Intel's Internet eXchange Architecture (IXA), which uses a Network Processor (NP) called the IXP [IXP] for high performance and programmability. NPs are a programmable alternative to Application Specific Integrated Circuits (ASIC) for rapid development of high performance network devices. We used the specialized hardware units on the IXP, such as the hash unit, a Ternary Content Addressable Memory (TCAM), and high performance memory interfaces to enable high performance packet processing. We also used the hardware multithreading capability of the IXP's Micro-Engines (ME) to hide the latency of accessing of chip memory. To evaluate our implementation, we deployed ANT on the Open Network Lab's [ONL] IXP-based routers. Our evaluation shows that ANT has a meaningful impact on cross-ISP link utilization and an insignificant impact on client download time.

The remainder of the paper is organized as follows. Section 2 provides a brief background on relevant aspects of the BitTorrent protocol. Section 3 describes the design of ANT. Section 4 discusses interesting aspects of ANT's implementation on the IXP 2800, and on ONL in particular. Our evaluation results are presented in Section 5. In Section 6, we survey related work. We consider future work in Section 7 and conclude in Section 8.

---

## 2. Background

BitTorrent is an extremely popular P2P file distribution protocol that is not formally standardized. The original BitTorrent client, called the mainline client, was open-source. Other clients interoperated with the original client and each other by conforming to aspects of the "protocol" gleaned from its source code. A de facto standard consisting of dominant protocol features eventually emerged [Cohen08][BitTorrent]. However, many clients make non-standard extensions to the protocol such as Ono [Choffnes08], peer ISP-cache support, peer exchange protocol, and multitorrents. Interestingly, the mainline client is no longer open source and is now owned by a company called BitTorrent. Nonetheless, other existing clients have no incentive to adopt any new, official protocol changes. ANT is based on aspects of the de facto protocol, so it works with all clients.

Although many different clients exist, at a high level, they all follow the same basic steps when downloading a file. To begin downloading a file, a peer procures a metainfo file with a .torrent extension from a website. Among other fields the metainfo file contains the infohash field that uniquely describes the file and the Uniform Resource Locator (URL) that represents the address of the tracker, a server that aids in peer discovery. Using the metainfo file, the peer queries the tracker's URL with the infohash as a parameter. In its request, the peer also includes the TCP port number it is listening on for incoming peer connections. Then the tracker responds with a random list of IP addresses and port numbers of other peers that are sharing the file identified by the infohash. The peer attempts to connect to these other peers to download pieces of the file. Finally, once a peer obtains a piece of the file, it uploads that piece to other peers that need it. In BitTorrent jargon, peers sharing the same file are connected to the same torrent or participating in the same swarm. Also, a peer that has the complete file is called a seed and other peers without the complete file are called leechers.

As might be expected, the de facto protocol relates to the basic operation of the protocol. It specifies the content of the torrent file and the formats of the peer-tracker and peer-peer messages [Cohen08]. It also makes recommendations for a peer selection strategy, but it only mandates that the strategy selected "should" work with the strategy used by the mainline client and other peers using a similar strategy. ANT operates on the peer-tracker control messages. The control messages use the Hypertext Transfer Protocol (HTTP), which is layered above the Transmission Control Protocol (TCP). A peer's HTTP GET request embeds parameters in the URL, so ANT can capture parameters such as the infohash from a single peer-tracker request. The tracker responds to a peer requests with a plain text document containing a list of peer addresses and port numbers.