

# Aspera FASP 고속 데이터 전송



TCP 기반의 전송 기술 대안과 중요 기술 비교

---

## 목차:

- 2 서론
  - 3 고속 TCP 개요
  - 5 UDP 기반의 고속 솔루션
  - 10 Aspera® FASP® 솔루션
- 

## 요점

### 과제

전송 제어 프로토콜(TCP)은 이상적 조건에서는 신뢰성 높은 데이터 전송을 보장하지만 장거리 WAN에서 패킷 손실 및 응답시간이 증가함에 따라 전송 속도 병목 현상이란 심각하고 고질적인 문제가 두드러지게 나타납니다. 대역폭을 추가해도 유효 처리 속도에는 변화가 없습니다. 파일 전송 속도는 개선되지 않고 값비싼 대역폭은 제대로 활용되지 못합니다.

### 솔루션

TCP와 달리, FASP의 처리 속도는 네트워크 지연의 영향을 받지 않으며 극심한 패킷 손실에 대한 내성도 뛰어납니다. FASP 전송은 네트워크 상태에 관계없이 가장 빠른 속도(표준 FTP의 최대 1,000배)와 우수한 안정성을 유지합니다. 최대 전송 속도는 오직 엔드포인트 컴퓨터의 인프라(일반적으로 디스크의 전송 속도)에 의해서만 제약을 받습니다.

### 혜택

- 가장 빠른 속도와 신뢰성
- 놀라운 대역폭 통제
- 자체 보안
- 유연한 개방형 아키텍처



## 서론

오늘날의 디지털 세계에서는 대용량 파일이 포함된 디지털 데이터를 전 세계 어디로든 빠르고 안정적으로 전송하는 일이 사실상 모든 산업에 중사하는 기업들의 성공에 결정적인 영향을 미칩니다. 지금까지 이와 같은 데이터 이동의 핵심 역할을 해온 TCP는 성능 면에서 고질적인 병목 현상(그림 1)을 유발합니다. 긴 왕복시간(RTT)과 패킷 손실이 발생하는 네트워크에서 이런 문제가 특히 두드러지는데, 그 중에서도 고대역폭 네트워크에서 가장 심각하게 나타납니다. TCP의 AIMD(Additive-Increase-Multiplicative-Decrease) 혼잡 회피 알고리즘이 이와 같은 고질적인 '소프트' 병목 현상을 유발하는 것으로 널리 알려져 있습니다. AIMD 알고리즘은 느린 속도로 네트워크 가용 대역폭을 탐색하고, 전송 속도를 늘렸다가 패킷 손실이 감지되면 전송 속도를 기하급수적으로 줄이기 때문입니다. 그러나 네트워크 혼잡과 관련 없는 다른 패킷 손실 원인(예: 물리적 네트워크 매체로 인한 손실) 역시 그와 동일하게 전송 속도를 저하시킨다는 사실을 아는 사람은 상대적으로 적습니다. 실제로 TCP AIMD 그 자체가 패킷 손실을 유발하고 다른 요인과 동일하게 병목 현상의 원인으로 작용합니다. 패킷 손실이 발생할 때까지 급격히 전송 속도를 높인다는 점에서 AIMD는 본질적으로 가용 대역폭을 낭용하는 셈입니다. 경우에 따라 이와 같이 자체적으로 유발한 손실이 오히려 다른 원인(예: 물리적 매체나 교차 트래픽의 버스트 현상)으로 인한 손실을 능가하고, 그로 인해 손실 염려 없던 통신 '채널'이 손실률을 예측할 수 없는 불안정한 '채널'로 탈바꿈하기도 합니다.

손실이 불가피한 TCP AIMD의 혼잡 통제 방식은 전송 속도에 결정적인 악영향을 미칩니다. 패킷이 유실될 때마다 재송신이 이뤄져야 하고, 재송신이 이뤄질 때까지 수신용 애플리케이션으로 데이터를 전송하는 일은 중단됩니다. 이와 같은 상황은 네트워크 애플리케이션의 성능 저하를 유발하는 점도 문제이지만 더 큰 문제는 대용량 데이터의 안정적인 전송(예: 순차 전송(바이트 스트림)을 필요로 하지 않는 파일 전송)이 근본적으로 위협 받는다는 것입니다.

FTP, HTTP, CIFS, NFS처럼 TCP를 토대로 개발된 기존의 파일 전송 프로토콜의 미흡한 성능에서 명백히 드러나듯이, 신뢰성 문제(재송신)가 TCP의 혼잡 제어 방식 문제와 맞물려서 불필요하게 극심한 파일 전송 속도 저하 현상이 발생합니다. 하드웨어 장치나 TCP 대안에 응용된 TCP Acceleration과 같은 프로토콜 최적화는 왕복시간과 패킷 손실률이 적당한 수준일 때 어느 정도 파일 전송 속도를 개선하지만 전송 거리가 국가를 벗어나는 경우 그 효과가 크게 감소합니다.

게다가 이 백서 후반에 설명하고 있듯이 TCP 또는 UDP 블래스팅 기술은 눈에 띄게 더 우수한 전송 속도를 발휘할 수 있는 대체 수단으로 거론되지만 대역폭 비용 부담이 크다는 단점이 있습니다. 이런 방식은 이미 전송 중이거나 수신한 데이터와의 중복으로 인해 상당한 양의 불필요한 파일 데이터, 때때로 엄청난 양의 데이터를 재전송하기 때문에 파일 데이터 전송 시간이 실제 필요한 것보다 몇 배 더 걸려서 막대한 대역폭 비용이 발생합니다. 특히 재전송된 데이터 패킷을 제외하면 유효 비트(goodput)의 전송 속도는 대단히 형편없습니다. 언뜻 보기에 이런 방식은 네트워크 대역폭 활용도를 개선하는 것처럼 보이지만 실상은 파이프가 '쓰레기 투성이'인 데다 전송 시간도 여전히 느립니다.

TCP 최적화나 단순한 블래스터가 네트워크 중심 프로토콜처럼 우수한 '양질의 데이터' 전송 속도를 발휘하는 협소 네트워크 상태를 유지하기 위해 스토리지 시스템의 데이터 유출입 과정에서 발생할 수 있는 소프트 병목 현상을 극복하는 데 초점이 맞춰져 있습니다.

최대 속도로 대량의 데이터를 옮기려면 아직 전송되지 않거나 수신하지 않은 데이터, 즉 '양질의 데이터'를 전송할 때 원천지부터 데이터 대상지까지의 전체 전송 경로 내내 가용 대역폭을 최대한 활용하는 종합적 접근법이 필요합니다. 오늘날 상용 WAN 환경에서 발생하는 다양한 범위의 네트워크 왕복시간, 손실률, 그리고 대역폭 용량 특성에 관계없이 이런 목적을 달성하려면 새롭고 혁신적인 대량의 데이터 이동 방식, 특히 신뢰성과 속도 제어 간의 상충 작용을 철저히 배제한 접근법이 필요합니다. 신뢰성 메커니즘 측면에서 이러한 접근법은 양질의 데이터 전송 속도가 100%를 유지할 수 있도록 필요한 데이터만 재전송해야 합니다. 그리고 공유형 인터넷 네트워크에 보편적으로 구축된 속도 제어 측면에서 이러한 접근법은 대역폭 공평성의 원칙을 유지하고 다른 전송 작업과 다른 네트워크 트래픽이 존재하더라도 혼잡을 피하는 한편, 필요한 경우 우선순위가 높은 전송 작업에 대역폭을 할당할 수 있는 옵션을 제공해야 합니다.

Aspera FASP는 이런 핵심 원칙을 토대로 공개 및 비공개 IP 네트워크를 통해 파일을 전송하는 기존의 TCP 기반 전송 기술의 최적의 대안을 제시하고자 개발된 혁신적인 데이터 대량 전송 기술입니다. Aspera FASP는 표준 네트워크를 그대로 유지한 채 애플리케이션 계층에 엔드포인트 애플리케이션 프로토콜로 구현됩니다. FASP는 네트워크 지연 및 패킷 손실에 관계없이 100%의 대역폭 효율성을 유지한 채 각종 IP 네트워크 상에서 대량의 데이터를 전송하도록 설계된 최고의 고성능 차세대 데이터 대량 전송 방식입니다.

이 백서에서는 대역폭 활용도, 네트워크 효율성, 전송 시간 측면에서 상업적으로나 학술적으로 파일 기반 전송을 가속화하는 새로운 접근법을 설명하고 각 접근법의 성능 및 실제 대역폭 비용을 Aspera FASP와 비교합니다.

네트워크 거리 증가에 따른 최대 TCP 속도 변화

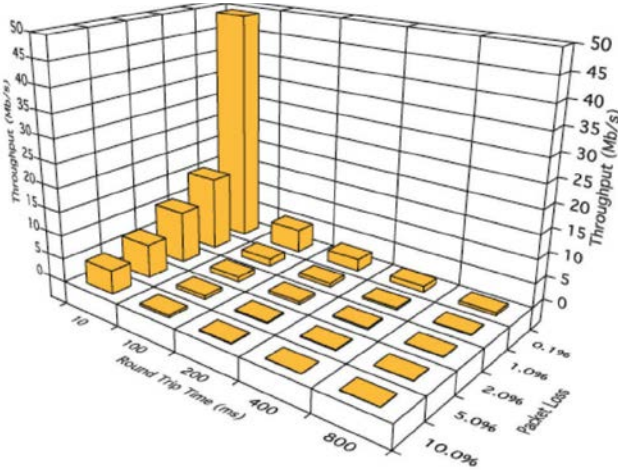


그림 1: 이 막대그래프는 다양한 패킷 손실 및 네트워크 지연 조건에서 TCP를 이용하는 파일 전송 기술(황색으로 표시)용 OC-3(155Mbps) 네트워크를 통해 달성할 수 있는 최대 전송 속도를 보여주고 있습니다. 전송 속도의 제약은 이론적으로 네트워크 RTT와 패킷 손실에 따라 달라집니다. 대역폭을 추가하더라도 유효 전송 속도에는 변화가 없다는 점을 유의하십시오. 파일 전송 속도는 개선되지 않고 값비싼 대역폭의 활용도만 저하될 뿐입니다.

### 고속 TCP 개요

최근 몇 년 새에 이런 변화에 대응한 다수의 고속 TCP 프로토콜과 TCP 가속 장비가 개발되었습니다. 고속 TCP 프로토콜은 AIMD의 근본적인 결함을 인지하고 이와 같은 프레임 기반의 혼잡 제어 알고리즘을 개량한 덕분에, 알고리즘으로 인해 인위적으로 유발되는 병목 현상이 줄고 장시간의 평균 전송 속도가 향상됩니다. 일반적으로 가장 진화된 버전의 프로토콜은 손실 사고가 발생할 때까지 전송 속도를 높이는 방식 대신, 네트워크 대기 행렬 지연과 같은 더욱 다양한 신호를 측정함으로써 혼잡 감지 능력을 개선하는 데 초점이 맞춰져 있습니다. 이런 장점은 TCP 흐름이 패킷 손실을 유발하여 인위적으로 혼잡 회피 모드에 돌입하는 것을 방지하는 데 유용하고 손실이 거의 없는 네트워크에서 장시간의 전송 속도를 개선하는 데도 효과적입니다.

그러나 물리적 매체 오류나 교차 트래픽 버스터 현상으로 인한 버퍼 오버플로우 때문에 발생하는 패킷 손실이 무시할 수 없는 수준인 WAN에서는 이런 장점이 제대로 빛을 발하지 못합니다. WAN에서는 단 하나의 패킷 손실도 TCP 전송 프레임이 극심한 감소를 유발하며 다수의 패킷 손실은 데이터 전송 속도에 막대한 악영향을 미치게 됩니다. 프레임당 복수의 패킷 손실이 발생할 경우 일반적으로 전송 시간 초과로 인해 발신 시스템부터 수신 시스템까지의 대역폭 지연 곱(bandwidth-delay-product) 파이프라인이 고갈되어 데이터 전송 속도가 0으로 급락합니다. 그리고 발신 시스템은 필연적으로 데이터 전송을 느리게 재시작해야 합니다.

그와 대조적으로 Aspera FASP에서는 전송 속도가 패킷 손실 사고의 영향을 받지 않습니다. 유실된 데이터는 바람직한 종단간 대역폭에 상응하는 속도로 재전송됩니다. 재전송은 사실상 이상적인 대역폭 효율성을 유지하면서 이뤄집니다. 즉, 중복으로 전송되는 데이터가 전혀 없고 총 목표 용량을 완벽하게 활용합니다.

그림 2에 보이는 것처럼 패킷 손실률이 1%인 네트워크에서 (CUBIC, H-TCP, BIC 등의 이형 버전을 포함한) 상용 버전의 고속 TCP인 FAST TCP의 전송 속도는 지연이 적은 네트워크의 표준 TCP Reno보다 빠르지만 국가간 혹은 대륙간 통신 네트워크처럼 상대적으로 왕복시간이 긴 조건에서는 그 격차가 급격히 감소합니다. 반면에 FASP의 전송 속도는 네트워크 지연 시간이 증가하더라도 전혀 저하되지 않으며 95% 이상의 대역폭 용량에서 최대 100%의 유효 전송률과 유효 파일 전송 속도를 발휘합니다. 이와 유사하게 패킷 손실이 증가(예: 손실률 5% 이상)할 경우 FASP의 전송 속도는 그와 동일한 비율만 저하됩니다. 더 높은 손실률에서 가속된 TCP의 전송 속도는 Reno와 비슷합니다.

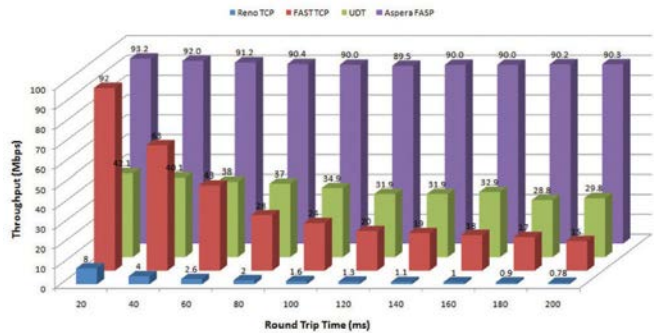


그림 2: 패킷 손실률이 1%인 네트워크에서 상용 버전의 고속 TCP인 Reno TCP, UDT, 그리고 Aspera FASP의 1GB 파일 전송 속도를 비교한 결과입니다. 지연이 적은 네트워크에서 가속된 TCP의 전송 속도는 지연이 적은 네트워크의 표준 TCP Reno보다 우수하지만 국가간 혹은 대륙간 통신 네트워크처럼 상대적으로 왕복시간이 긴 조건에서는 그 격차가 급격히 감소합니다. 반면에 FASP의 전송 속도는 네트워크 지연 시간이 증가하더라도 전혀 저하되지 않습니다. 패킷 손실이 증가(예: 손실률 5%)할 경우 FASP의 전송 속도는 그와 동일한 비율만 저하되며 더 높은 손실률에서 TCP의 전송 속도는 Reno와 비슷합니다.

패킷 손실에 대한 표준 및 고속 TCP의 대응 때문에 발신 시스템은 어쩔 수 없이 전송 프레임이 줄이느라 전송 속도가 불규칙해 집니다. 뿐만 아니라 TCP의 순차적 전송을 유지하기 위해 재전송된 패킷과 새 패킷이 동시에 발신 프레임을 점유하게 됩니다. 이처럼 새로 전송되는 패킷과 재전송된 패킷이 동일한 TCP 전송 프레임에 공존하게 되면 교착 현상으로 말미암아 전송 무결성을 보증하는 TCP의 신뢰성 메커니즘과 혼잡 제어가 제 기능을 발휘하지 못하게 되고 대용량 데이터처럼 순차 전송을 필요로 하지 않는 애플리케이션의 전송 속도가 불필요하게 저하됩니다.

TCP의 신뢰성 덕분에 데이터가 전혀 유실되지 않으며(수신 시스템이 패킷 손실을 감지하고 차후에 발신 시스템이 유실된 패킷을 재전송합니다) 수신된 데이터는 애플리케이션에 순차적으로 전달됩니다. 이 두 가지 장점을 보장하기 위해 TCP는 유실된 패킷을 재전송할 뿐 아니라 유실된 패킷이 도착할 때까지 먼저 도착한 비순차적 패킷(커널 메모리에 임시 저장)을 보류합니다. 이렇게 해야 수신된 데이터를 애플리케이션 계층에 순차적으로 전달할 수 있습니다. 유실된 패킷이 수신될 때까지 수신 시스템이 유입되는 패킷을 지속적으로 RAM에 저장해야 한다는 점을 고려하면, 재전송이 가장 시급한 선결과제이므로 새 데이터 수신은 일제히 늦춰야 합니다. 특히, 패킷 유실 사고가 발생할 때마다 새 패킷의 전송 속도를 늦추고(일반적으로 유실된 패킷이 수신 시스템에 재전송됐음이 확인될 때까지 전송 프레임이 정지합니다) 재전송 패킷이 수신부의 바이트 스트림에 난 '구멍'을 메울 때까지 기다려야 합니다. 본질적으로 TCP의 신뢰성 제어와 흐름 제어(또는 혼잡 제어)는 철저히 연동하도록 설계되어 있습니다.

이와 같은 유형의 메커니즘을 토대로 TCP는 다수의 애플리케이션이 필요로 하는 엄격한 순차적 바이트 스트림 전송을 실현하지만, 파일 전송처럼 엄격한 바이트 순서를 굳이 필요로 하지 않는 애플리케이션에는 오히려 엄청난 악재로 작용하여 보이지 않는 인위적 병목 현상이 발생하고 그에 상응하는 데이터의 전송 속도가 제한됩니다.

이를 명확히 이해할 수 있도록 간단한 예를 들어 손실 한 건당 프레임이 1/8 감소하는 조건에서 혼잡과 무관한 고속 TCP의 패킷 손실 한 건으로 인한 전송 속도 저하를 계산해보겠습니다. 패킷 손실률이 1%이고 왕복 시간 지연이 100ms인 기가비트 네트워크의 경우, 패킷 손실이 발생할 때마다 (TCP Reno은 1/2인데 반해) 속도가 1/8씩 감소합니다.

수신 시스템이 패킷 손실 사고 이전의 원래 전송 속도를 회복하는 데 걸리는 시간은  $1\text{Gbps} \div 8(\text{비트}/\text{바이트}) \div 1024(\text{바이트}/\text{패킷}) \times 100\text{ms} \times 0.125(\text{저하율}/\text{손실}) \times 100\text{ms} \approx 152.6\text{초}$ 입니다. 이 회복 기간 동안 고속 TCP는 단 한 건의 패킷 손실 때문에  $152.6\text{초} \times 1\text{Gbps} \times 0.125/2 \approx 8.9\text{GB}$ 의 전송 속도를 손해 봅니다. 실제 WAN에서 네트워크 대기 행렬, 물리적 매체 접근, 일정 예약 및 복구 등으로 인해 RTT가 더 커질 수 있기 때문에 실제 손실 값은 훨씬 더 커집니다. 따라서 일반적으로 발신 시스템이 회복하는 데 152.6초 이상의 시간이 소요됩니다. 수 차례의 연이은 패킷 손실은 재앙이나 다름없습니다. IETF(Internet Engineering Task Force)의 설명도 같은 맥락으로 해석됩니다.

"LFN(Long Fat Network)의 용량에 맞춰서 프레임 크기를 늘리면 프레임당 하나 이상의 패킷이 유실될 가능성도 그만큼 커집니다. 이런 상황은 LFN을 기반으로 한 TCP의 전송 속도에 치명적인 영향을 미칠 수 있습니다. 게다가 RFC 1323 발표 이후 무작위 폐기 방식을 토대로 한 혼잡 제어 메커니즘이 등장하고 무작위 간격의 패킷 폐기가 보편화됐습니다. 그로 인해 프레임당 하나 이상의 패킷이 유실될 가능성도 커졌습니다."<sup>1</sup>

엄격한 순차 전송이 필수인 바이트 스트림 애플리케이션에는 때때로 이와 같은 속도 저하나 전송 속도 손실이 오히려 필요하다는 사실에 주목할 필요가 있습니다. 그렇지 않을 경우 앞서 소개한 예처럼 최소한 1RTT를 유지하기 위해 TCP 연결 라인당 하나의 유실된 패킷이 수신되길 기다리느라 최소한  $1\text{Gbps} \times 100\text{ms} \times 0.125 \approx 1.5\text{MB}$ 의 데이터를 추가로 RAM에 저장해야 합니다.

그러나 파일 전송 애플리케이션은 유실된 패킷을 기다리지 않고 비순차적으로 데이터를 디스크에 기록하기 때문에 이와 같은 지연이 불필요합니다. 따라서 이런 상황에서는 고급 속도 제어 메커니즘으로 찾아낸 네트워크 내부의 가용 대역폭과 정확히 일치하는 속도로 항상 재전송할 수 있습니다.

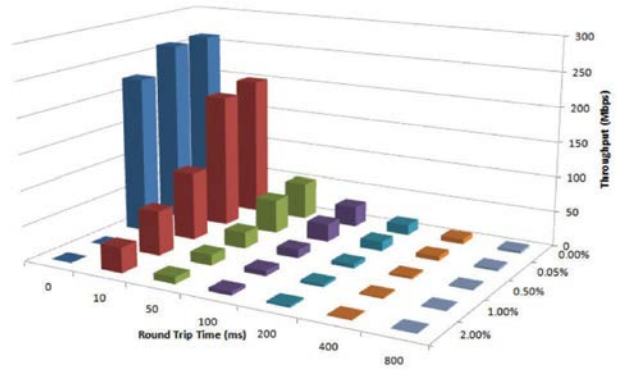
사실상 TCP 자체적으로는 신뢰성 메커니즘과 혼잡 제어가 을 분리할 수 없어서 신뢰할 수 있는 바이트 스트림 전송을 실현한다는 TCP의 목적을 IETF가 재정립하지 않는 한 이와 같은 인위적인 병목 현상은 해소되지 않을 것입니다.<sup>2</sup> 신뢰성 높은 스트리밍 애플리케이션과 기타 애플리케이션용 전송 제어 프로토콜에 대한 전통적인 의존도는 감안하면 이런 프로토콜이 두 가지 분야 모두의 실질적인 차선책임이 입증된 셈입니다.

## UDP 기반의 고속 솔루션

TCP가 보장하는 신뢰성 때문에 네트워크 전송 속도가 저하되고 평균 지연 시간이 증가하며 지연 편차가 악화됩니다. 신뢰성과 혼잡 회피의 상충 관계를 극복하려는 노력은 수년간 계속돼 왔습니다. TCP 그 자체의 수정에 수반되는 복잡성 때문에 최근 몇 년간 학계와 업계는 속도와 신뢰성을 각기 따로 관리할 수 있는 애플리케이션 계층 프로토콜을 추구해 왔습니다. 이런 접근법은 전송 계층에 UDP를 TCP의 대안으로 사용해서 애플리케이션 계층에서 신뢰성을 확보합니다. 그와 같은 접근법 중 대다수는 UDP 블래스터입니다. **그림 3**은 전형적인 WAN 조건(RTT 및 패킷 손실 증가)에 300Mbps 연결 상태에서 실행되는 상용 UDP 데이터 블래스터 Rocketstream의 전송 속도를 보여주고 있습니다.

(Signiant, File Catalyst, Sterling Commerce®와 같은 제품에 의해 사용되는) 오픈 소스 구현 기술인 Tsunami와 UDT를 비롯한 UDP 솔루션에서는 패킷이 유실된 경우 전송 속도를 줄이는 단순한 알고리즘을 통해 UDP 블래스터의 혼잡 제어 방식이 강화됐습니다. 각 이용 사례, 즉 대역폭, 왕복시간 지연, 패킷 손실, 그리고 경로 개수의 조합에 따라 특정 네트워크 경로의 타당한 성능을 확보하기 위해 백오프 알고리즘을 '조정'할 수 있는 반면, 설계 구조 자체는 실제 인터넷 네트워크의 네트워크 RTT 및 패킷 손실 조건 범위와 동시 전송 경로에 맞게 조정할 수 없습니다. 결과적으로, 이러한 접근법에서는 단순히 '파이프라인을 채우느라' 가용 대역폭의 활용도가 저하될뿐더러, 그 과정에서 데이터 중복 전송으로 네트워크를 낭용(전형적인 네트워크 조건에서 중복 데이터 50%)하여 1차적으로 대역폭이 낭비되고 2차적으로 유효 파일 전송 속도('goodput')가 급락하게 됩니다. 게다가 이러한 접근법에서는 대역폭 낭용으로 인해 다른 TCP 애플리케이션의 패킷 손실이 유발되고 유효 데이터 전송이 중단되면서 다른 트래픽의 네트워크 사용을 방해할 수 있습니다.

RocketStream 전송 속도(Mbps)



**그림 3** 이 막대그래프는 300Mbps 연결 상태의 다양한 패킷 손실 및 네트워크 지연(WAN) 조건에서 RocketStream의 전송 속도를 보여주고 있습니다. 높이가 0인 이 막대그래프는 발신 시스템과 수신 시스템을 서로 연결하는데 실패한 경우입니다. RTT나 패킷 손실률이 큰 경우, RocketStream에서는 이런 현상이 흔히 나타납니다.

이 연구 사례에서는 이와 같은 문제를 입증하기 위해 가장 진화된 (NACK 기반의) 재전송용 UDP 전송 솔루션으로 손꼽히는 UDT(솔루션 제공업체에 따라 다름)를 선택했습니다. UDT의 특징은 다음과 같습니다.

- 미흡한 혼잡 회피 기능: UDT에 채용되는 D-AIMD(dynamic 'AIMD') 알고리즘은 AIMD와 유사하게 기능하지만 전송률이 증가할 때 AI(Additive Increase) 매개변수로 전송률 증가 페이스를 줄인다는 점이 다릅니다. 이와 같은 접근법은 앞서 언급한 TCP의 맹점(신뢰성 및 전송률 제어의 상관관계)을 해결하지 못하며, 대신 매개변수 한 개를 조정하면 AIMD와 심지어 TCP의 성능 저하 문제를 해결할 수 있는 것으로 알려져 있습니다. 실제로, 특별히 조정된 D-AIMD는 특정 이용 환경에서 TCP보다 우수한 성능을 발휘합니다. 그러나 또 다른 이용 환경에서는 TCP의 성능에 미치지 못합니다. 따라서 다수의 전형적인 WAN에서 UDT의 성능은 사실상 TCP보다 떨어집니다.

- UDT의 공격적인 데이터 전송 메커니즘은 대폭적인 전송률 편차와 패킷 손실을 유발해서 자체적인 전송 속도를 떨어뜨릴 뿐 아니라 다른 트래픽마저 위태롭게 하고 전반적인 네트워크 성능을 저하시킵니다. 일반적인 TCP 흐름(예: 웹 클라이언트의 HTTP 세션)이 UDT 전송과 대역폭을 공유하는 전형적인 WAN에서는 공격적인 UDT 흐름 때문에 TCP 흐름에 서비스 거부(DoS) 사태가 발생할 소지가 있습니다(그림 4 참조). TCP의 극도로 불량한 친화력은 독창적인 2005년 UDT 논문 "UDT 기반의 프로토콜 구현 최적화(Optimizing UDP-based Protocol Implementations)"에서 이론적으로 연구됐는데 저자들은 이와 같이 극도로 불량한 친화력을 방지하기 위해 반드시 충족해야 할 조건을 구체적으로 제시한 바 있습니다.<sup>2</sup> 그러나 현실적으로 대단히 전형적인 WAN(예: RTT 100ms, 패킷 손실률 0.1%인 WAN)에서는 이 조건을 충족할 수 없으며 TCP의 극도로 불량한 친화력은 불가피합니다. 다시 말해서, UDT 기반의 데이터 전송 솔루션을 사용하려는 일반 고객은 UDT가 전체 네트워크 환경(예: 웹, 이메일, VOIP, 네트워크 관리)의 운영 구조를 저해하지 못하도록 특정 유형의 QoS 체제에 더 많은 시간과 비용을 투자해야 합니다.

- UDT의 공격적인 전송 및 불안정한 재전송 방식은 소중한 대역폭의 효율성 저하로 이어지며 그로 인해 고객이 더 많은 대역폭을 불필요하게 구매해야 하는 상황이 발생하기도 합니다. (값비싼 대역폭을 더욱 효과적으로 활용할 목적으로 개발된 솔루션이 오히려 대역폭을 낭비하는 주범이 된 것입니다.) 몇 가지 실험에서 나타난 전송률, 수신율, 그리고 유효 파일 전송 속도 간의 큰 격차(그림 6 및 7)은 주로 UDT의 지나치게 공격적인 데이터 삽입과 불안정한 재전송 메커니즘 때문에 라우터와 수신 시스템에서 상당한 데이터 누락을 초래합니다. 일부 전형적인 WAN에서는 효율성('goodput')이 20% 이하로 떨어지는 것으로 측정됐습니다. 다시 말해서, UDT의 네트워크 활용도가 100%인 경우, 수신 시스템에 중복 데이터를 전송하거나 (UDT 자체의 대역폭 남용으로 인해) 오버플로우가 발생한 버퍼에 유효 데이터를 전송하는 데 대역폭 용량의 80%가 사용됩니다.

일반적인 사용자가 UDT를 사용할 경우의 '이득'과 '비용'은 정확한 비교를 위해 수치화할 수 있습니다. '이득'은 필요한 데이터를 전송(goodput)하는 데 사용된 대역폭의 효율성을 기준으로 측정하여 바로 전송 시간으로 산정할 수 있는 반면, '비용'은 하나의 필요한 데이터 패킷을 전송하는 데 들인 수고, 즉 하나의 필요한 패킷을 다른 네트워크 종단부의 애플리케이션 계층에 성공적으로 전달하기 위해 전송한 중복 데이터 전송 횟수로 도출할 수 있습니다. 이런 비용에는 다른 전송에 유발한 비용, 즉 공정한 대역폭 공유 기회 상실(그림 4)과 그로 인해 저하된 전송 속도도 반영됩니다. 특히, 그림 5에 이미 부분적으로 반영됐듯이 다양한 WAN 조건에서 UDT가 (느린 전송으로 인해) 유효 전송 속도를 저하시키므로 사용자에게는 거의 이득이 없는 셈입니다. 그리고 대역폭 남용과 중복 재전송 때문에 발생하는 대역폭 비용은 다른 워크플로우에 지대한 영향을 미칩니다.

그림 5는 각기 다른 RTT와 패킷 손실률에서 T3(45Mbps) 연결을 통한 UDT 전송으로 패킷 하나를 전송하는 데 소요되는 총 비용을 보여주고 있습니다. 대부분의 전형적인 WAN에서 하나의 패킷 전송에는 8~10회의 재전송이 필요합니다. 다시 말해서, UDT 발신 시스템이 1GB 파일을 전송하기 위해 결국 9~11GB를 네트워크에 낭비하는 것입니다. 따라서 전송을 완료하는 데 필요한 시간보다 9~11배 많은 시간이 소요되며 다른 흐름에 대규모 패킷 손실도 유발합니다.

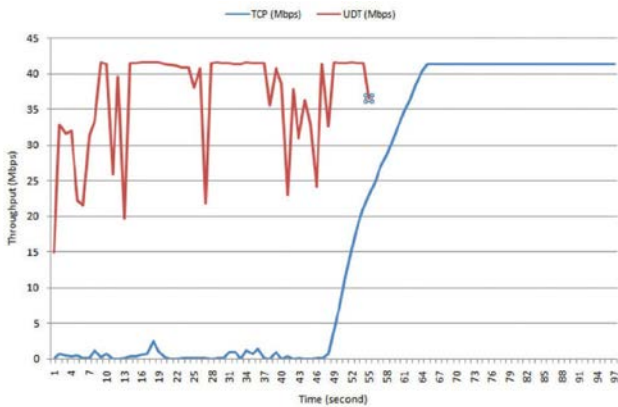


그림 4. 패킷 손실률이 0%이고 RTT가 50ms인 전형적인 T3 연결 상태에서 UDT 전송 방식의 파일 전송 속도와 UDT 전송 방식 일반적인 TCP 흐름에 미치는 영향을 보여주고 있습니다. UDT 흐름이 종료될 때까지 대부분의 시간 동안 TCP 흐름은 확인할 수 없습니다.

그림 5에서 비용은 지나치게 공격적인 UDT 발신 시스템의 데이터 삽입률과 UDT 수신 시스템 데이터 누락으로 인한 중복 재전송 때문에 발생합니다. 좀 더 구체적으로 말해서, 전송 비용에는 지나치게 공격적인 UDT 발신 시스템의 데이터 삽입률과 라우터의 패킷 누락으로 인한 손실이 반영되며 수신 비용에는 수신 시스템 데이터 누락으로 인한 중복 재전송이 반영됩니다.

좀 더 정확히 하자면 전송 비용은 다음과 같이 산정됩니다.

$$\text{전송 비용} = \frac{\text{전송된 총 바이트} - \text{실제 수신된 바이트}}{\text{실제 수신된 바이트}}$$

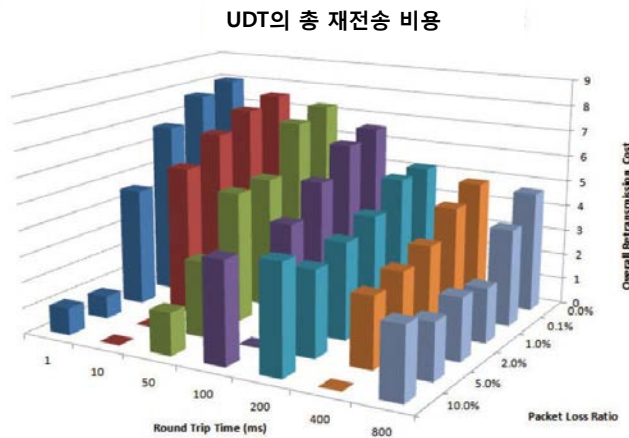


그림 5. 이 막대그래프는 각기 다른 RTT와 패킷 손실률 조건에서 UDT 전송의 재전송 비용을 보여주고 있습니다. '전송 비용'을 의미하는 각 막대그래프의 높이는 1GB의 파일이 전송될 때 기가바이트 단위로 재전송된 데이터의 양입니다. 높이가 0인 이 막대그래프는 발신 시스템과 수신 시스템을 서로 연결하는 데 실패한 경우입니다. RTT나 패킷 손실률이 큰 경우, UDT에서는 이런 현상이 흔히 나타납니다. 원래 파일 크기의 최대 9배가 전송되어 불필요한 재전송에 낭비된다는 점에 유의하십시오.

수신 비용은 다음과 같이 산정됩니다.

$$\text{수신 비용} = \frac{\text{수신된 총 바이트} - \text{실제 유효 바이트}}{\text{실제 유효 바이트}}$$

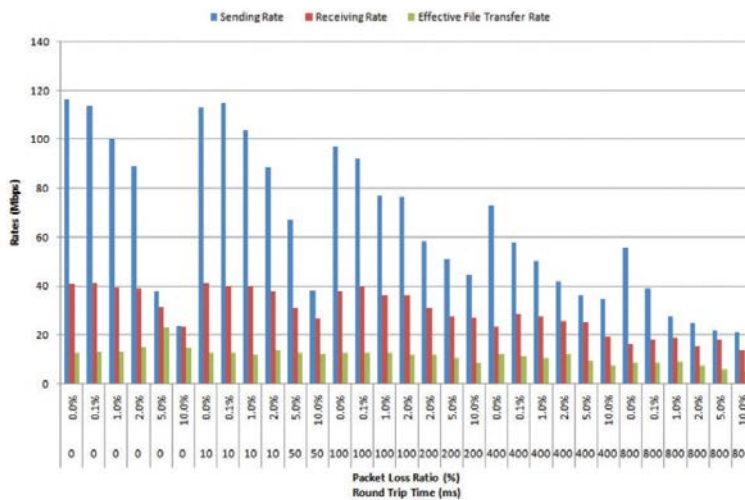
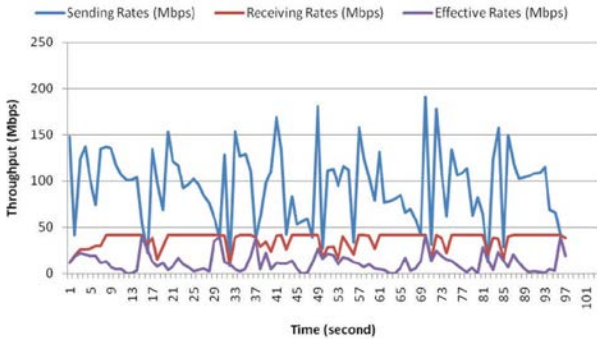
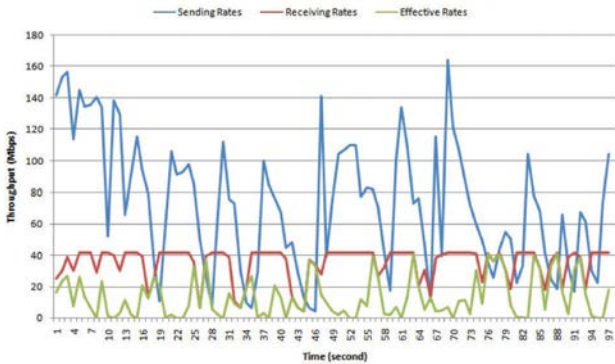


그림 6. 이 막대그래프는 각기 다른 RTT와 패킷 손실률에서 T3 연결을 통한 UDT 전송의 전송률, 수신율, 그리고 유효 수신율을 보여주고 있습니다. 전송률과 수신율의 큰 격차는 중간 네트워크 경로에서 대량의 패킷 손실이 발생했음을 시사합니다. 그리고 수신율과 유효 수신율의 큰 격차는 다수의 중복 데이터 재전송이 이뤄졌음을 의미합니다.

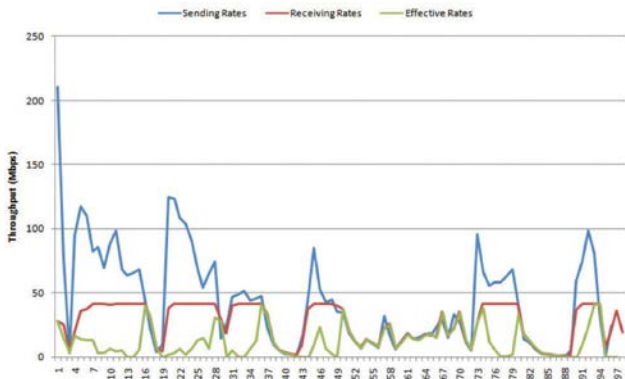
그림 7. 패킷 손실률이 각각 0%, 1%, 5%이고 RTT가 각각 100ms, 200ms, 200ms인 T3 연결에서 UDT 전송의 전송률, 수신율, 유효 수신율을 보여주고 있습니다. 전송률과 수신율의 격차는 라우터에서 대량의 데이터 손실이 발생했음을 의미하며 수신율과 유효 수신율의 격차는 UDT 수신 시스템에서 다수의 중복 재전송 누락이 발생했음을 시사합니다.



(a) 패킷 손실률이 0%이고 RTT가 100ms인 T3 네트워크에서 UDT 전송



(b) 패킷 손실률이 1%이고 RTT가 100ms인 T3 네트워크에서 UDT 전송



(c) 패킷 손실률이 5%이고 RTT가 200ms인 T3 네트워크에서 UDT 전송

전송 비용이 높아질수록 라우터에서 누락되는 패킷이 많아지고, 수신 비용이 높아질수록 수신 시스템에서 누락되는 패킷이 많아집니다. 그림 7은 각기 다른 RTT와 패킷 손실률에서 T3 연결을 통한 UDT 전송의 전송률, 수신율, 그리고 유효 수신율을 보여주고 있습니다. 전송률 대 수신율의 비율과 수신율 대 유효 수신율의 비율에 따라 위에 정의한 비용이 결정됩니다. 전송률이 한결같이 수신율보다 높고 수신율은 모든 네트워크 구성에서 유효 수신율보다 높은 것으로 관찰됐습니다. 이와 같은 비용 때문에 네트워크의 운영 점수에서 (전송 속도를 대역폭으로 나눠서 산출되는) 네트워크 활용도는 1에 가깝지만 (유효 파일 전송 속도를 대역폭으로 나눠서 산출되는) 네트워크 효율성은 15%에 불과합니다. 결과적으로 파일 전송 속도는 목표한 것보다 6배 이상 느립니다.

이해를 돕고자 다음과 관련 성능 관련 문제에 대한 답을 구함으로써 WAN 조건이 각기 다른 간단한 파일 전송 사례를 통해 위의 비용을 산정할 수 있습니다.

- 몇 바이트를 전송해야 하는가?
- 실제로 몇 바이트가 전송됐는가?
- 실제로 몇 바이트가 수신됐는가?
- 전송에 얼마나 걸렸는가?
- 유효 파일 전송 속도는 얼마인가?

답변은 표 1에 정리되어 있으며 동일한 조건에서 Aspera FASP와 비교해 봤습니다.



대역폭 (Mbps)	RTT (ms)	패킷 손실률 (%)	전송하려는 용량 (MB)	전송해야 할 용량(실제 데이터 + 저장 매체로 인한 불가피한 손실)(MB)	실제 전송한 데이터 용량 (MB)	전송 비용 (발신자의 경비) (%)	실제 수신한 데이터 용량 (MB)	수신 비용 (수신자의 경비) (%)	소요 시간(s)	유효 파일 전송 속도 (Mbps)	관측된 네트워크 활용도	네트워크 효율성(유효 활용도) (%)
45	0	0	953.7	953.7	9093.2	314.1%	2195.8	130.2%	625.0	12.8	66.2%	28.4%
45	100	1	953.7	963.2	5941.8	234.9%	1774.0	86.0%	618.0	12.9	54.1%	28.8%
45	400	5	953.7	1001.4	3764.1	150.6%	1501.8	57.5%	830.0	9.6	34.1%	21.4%
45	800	5	953.7	1001.4	3549.9	152.9%	1403.9	47.2%	1296.0	6.2	20.4%	13.7%
100	100	1	953.7	963.2	1413.0	14.0%	1239.8	30.0%	239.0	33.5	44.0%	33.5%
100	200	5	953.7	1001.4	2631.2	19.6%	2200.1	130.7%	571.8	14.0	32.6%	14.0%
300	100	1	953.7	963.2	1060.0	2.1%	1038.4	8.9%	232.0	34.5	12.6%	11.5%
300	200	1	953.7	963.2	1083.0	2.3%	1059.1	11.1%	273.0	29.3	11.0%	9.8%
500	200	1	953.7	963.2	1068.9	1.7%	1051.5	10.3%	252.0	31.8	7.1%	6.4%
500	200	5	953.7	1001.4	1660.9	5.3%	1576.7	65.3%	539.1	14.8	5.0%	3.0%

표 1: 전형적인 WAN을 이용한 UDT 파일 전송 - 높은 대역폭 비용과 느린 전송 속도

대역폭 (Mbps)	RTT (ms)	패킷 손실률 (%)	전송하려는 용량 (MB)	전송해야 할 용량(실제 데이터 + 저장 매체로 인한 불가피한 손실)(MB)	실제 전송한 데이터 용량 (MB)	전송 비용 (발신자의 경비) (%)	실제 수신한 데이터 용량 (MB)	수신 비용 (수신자의 경비) (%)	소요 시간(s)	유효 파일 전송 속도 (Mbps)	관측된 네트워크 활용도	네트워크 효율성(유효 활용도) (%)
45	0	0	953.7	953.7	953.7	0.0%	953.7	0.0%	185.4	43.1	98.5	95.9%
45	100	1	953.7	963.2	963.3	1.0%	953.7	0.0%	187.8	42.6	97.1	94.6%
45	400	5	953.7	1001.4	1002.1	5.0%	954.3	0.1%	197.0	40.6	92.1	90.3%
45	800	5	953.7	1001.4	1003.5	5.1%	955.2	0.2%	197.0	40.6	91.6	90.3%
100	100	1	953.7	963.2	963.3	1.0%	953.8	0.0%	85.0	94.1	96.3	94.1%
100	200	5	953.7	1001.4	1002.4	5.0%	954.5	0.1%	88.9	90.0	91.9	90.0%
300	100	1	953.7	963.2	964.0	1.0%	954.4	0.1%	29.3	273.4	92.6	91.1%
300	200	1	953.7	963.2	964.7	1.0%	955.1	0.1%	29.2	274.3	91.9	91.4%
500	200	1	953.7	963.2	963.0	1.0%	953.9	0.0%	181.6	440.6	90.6	88.1%
500	200	5	953.7	1001.4	1001.8	5.0%	954.2	0.0%	186.9	428.0	88.0	85.6%

표 2: 전형적인 WAN을 이용한 Aspera FASP 파일 전송 - 0에 가까운 대역폭 비용과 빠른 전송 속도

표 1에 보이는 UDT 파일 전송 속도의 직접적인 결과는 유효 데이터가 네트워크를 전혀 통과하지 못한다는 것입니다. 또한 (표 1의 8번째 열에 보이는) 네트워크 효율성의 가치 측면에서도 그 결과는 별반 다르지 않아서 미흡한 성능을 더욱 악화시킬 뿐 아니라 대역폭을 포화시켜서 다른 네트워크 트래픽의 서비스 거부(DoS) 현상을 유발합니다. 일부 UDT 및 TCP 솔루션에 채용된 것처럼 전송 속도와 네트워크 활용도 개선을 목적으로 병렬 전송 구조를 구축하면 대역폭 낭비만 가중되고 고객이 어쩔 수 없이 너무 이른 시점에 더 많은 대역폭을 확보하는 데 투자해야 합니다. 병렬 전송 구조를 통한 네트워크 활용도 및 데이터 전송 속도 개선은 거의 찾아보기 힘든 데다 그로 인한 비용(그림 8)은 대폭 상승합니다. 게다가 2개의 UDT 세션으로 인해 재전송률이 40% 추가로 증가합니다.

같은 예(표 1)에서 UDT는 1GB 미만의 파일을 성공적으로 전송하는 데 13GB~15GB의 데이터를 네트워크에 낭비합니다. 병렬 TCP 또는 UDT 전송 방식이 사용되는 솔루션의 성능은 그림 8에 보이는 것과 비슷하거나 심지어 더 떨어집니다.

UDT의 총 재전송 비용

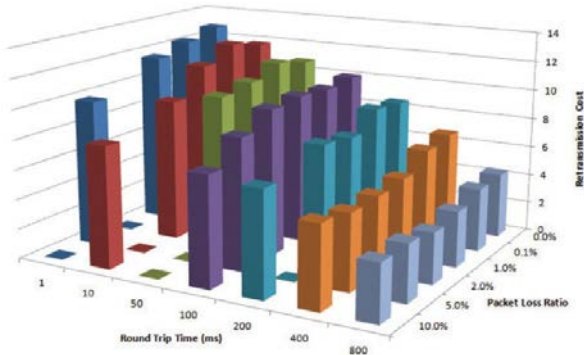


그림 8. 이 막대그래프는 각기 다른 RTT와 패킷 손실률에서 T3 네트워크를 통한 1GB 파일 한 개를 전송하는 데 사용되는 2개의 병렬 UDT 세션의 재전송 비용을 보여주고 있습니다. 전송 비용을 의미하는 각 막대그래프의 높이는 1GB의 파일이 전송될 때 기가바이트 단위로 재전송된 데이터의 양입니다. 높이가 0인 이 막대그래프는 발신 시스템과 수신 시스템을 서로 연결하는 데 실패한 경우입니다. RTT나 패킷 손실률이 큰 경우, UDT에서는 이런 현상이 흔히 나타납니다. 이 과정에서 실제 파일 크기의 거의 14GB(14배)가 재전송된다는 점에 주목할 필요가 있습니다.

## Aspera FASP 솔루션

Aspera FASP는 바이트 스트림 전송을 필요로 하지 않는 애플리케이션에 안정적인 전송을 지원하느라 TCP가 남긴 취약점을 해소하고 상층 관계에 있는 신뢰성과 속도 제어를 완벽하게 분리합니다. Aspera FASP는 전송 계층에서 표준 UDP를 사용하는 한편, 채널에서 유실된 실제 패킷을 정확히 재전송하는 최적의 접근법을 통해 애플리케이션 계층에서 신뢰성과 혼잡 제어의 상층 관계로 인해 파급된 병목 현상을 해소합니다.

신뢰성과 속도 제어의 상층 관계를 해소했기 때문에 TCP 기반의 바이트 스트리밍 애플리케이션에서처럼 유실된 패킷을 재전송하느라 새로운 패킷의 속도를 늦출 필요가 없습니다. 전송 과정에서 유실된 데이터는 종단간 경로의 가용 대역폭에 상응하는 속도나 설정되어 있는 속도로 재전송되는데, 중복 데이터가 재전송되는 일이 전혀 없기 때문에 수신 비용이 전혀 소요되지 않습니다.

경로 안의 가용 대역폭은 지연 기반의 속도 제어 메커니즘으로 찾아내므로 전송 비용이 0에 가깝습니다. 특히, FASP의 적응식 속도 제어는 네트워크에서 적고 안정적인 양의 '대기 행렬'을 유지하는 데 중점을 둔 채, 측정된 대기 행렬 지연을 네트워크 (또는 디스크로 인한) 혼잡의 1차 지표로 사용합니다. 측정된 대기 행렬이 목표 수치 이하(일부 대역폭이 미사용 상태여서 전송 속도를 높여야 한다는 의미입니다)로 떨어지면 전송 속도를 상향 조정하고 측정된 대기 행렬이 목표 수치 이상(대역폭이 모두 사용되고 있어서 혼잡이 임박했다는 의미입니다)으로 증가하면 전송 속도를 하향 조정합니다. FASP는 탐색 패킷을 네트워크에 주기적으로 전송함으로써 전송 경로에 따른 더욱 정확하고 시기적절한 대기 행렬 지연 시간 측정 결과를 확보할 수 있습니다. 대기 행렬 지연 시간이 상승한 것으로 감지되면 FASP 세션이 대상 대기 행렬과 현재 대기 행렬의 격차에 비례한 만큼 전송 속도를 줄이므로 네트워크 과부하를 막을 수 있습니다. 네트워크 혼잡이 완화되면 FASP 세션이 대상 대기 행렬의 비율에 따라 재빨리 전송 속도를 높이므로 가용 네트워크 용량 중 거의 100%를 다시 활용하게 됩니다.

TCP의 속도 제어와 달리, FASP의 적응식 속도 제어는 여러 가지 큰 장점이 있습니다. 첫째, 네트워크 대기 행렬 지연을 1차 혼잡 신호로 사용하고 패킷 손실률을 2차 혼잡 신호로 사용하여 데이터 매체로 인한 패킷 손실이 발생했을 때 인위적으로 네트워크에서 속도를 늦추는 게 아니라 네트워크 혼잡을 정확히 예측할 수 있습니다. 둘째, FASP에 내장된 신속 대응 메커니즘 덕분에 고속 파일 전송 도중 자동으로 속도를 늦춰서 다수의 동시 전송이 이뤄질 때는 안정적이고 높은 전송 속도를 유지하다가 보다 효율적인 전송 시간대에는 전송 속도를 대폭 올려서 미사용 대역폭을 최대한 효율적으로 활용할 수 있습니다. 셋째, 고급 피드백 제어 메커니즘으로 FASP 세션의 속도가 더욱 빠른 시간 내에 안정적인 균형 속도에 수렴하여 혼잡한 라우터의 버퍼에 목표량의 대기 행렬 비트를 삽입할 수 있습니다. 안정적인 전송 속도와 대기 행렬 지연 시간은 QoS 하드웨어나 소프트웨어에 추가로 투자하지 않고서도 최종 사용자에게 QoS 경험을 선사하는데 도움이 됩니다. 그리고 데이터 전송 시간을 예측할 수 있어 동일한 네트워크를 공유하는 다른 애플리케이션으로 데이터가 이동하는 과정이 투명하게 공개됩니다. 넷째, NACK 기반의 UDP 블래스터와 달리, 완벽한 대역폭 활용으로 네트워크 관련 비용이 사실상 전혀 소요되지 않으며 100%를 넘나드는 네트워크 효율성이 유지됩니다.

Aspera FASP 전송 속도(Mbps)

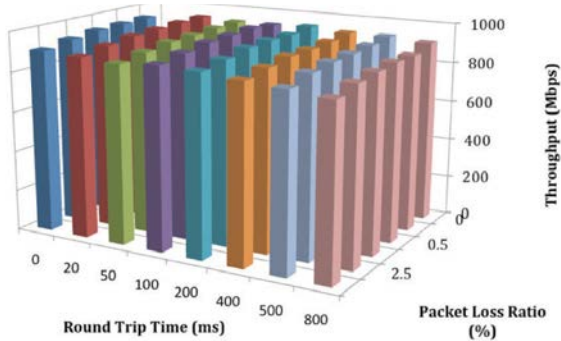


그림 9. 이 막대그래프는 혁신적인 전송 기술인 FASP를 이용하는 파일 전송 기술용 1Gbps 네트워크로 다양한 패킷 손실 및 네트워크 지연 조건에서 달성할 수 있는 전송 속도를 보여주고 있습니다. 네트워크 지연과 패킷 손실이 발생하더라도 대역폭 효율성은 저하되지 않습니다.

가용 대역폭을 효율적으로 활용한다는 점 외에도 FASP의 적응식 속도 제어는 지연 시간을 근거로 하기 때문에 애플리케이션이 전송 계층 서비스에서 계획적인 우선순위를 정할 수 있습니다. 네트워크 대기 행렬에 대한 체계적 응답 덕분에 애플리케이션 목표(예: 동시 FASP에 차별화된 대역폭 우선순위 적용)를 충족하기 위해 개별 전송의 우선순위를 지정하거나 해제할 수 있는 가상의 지표가 마련되어 있는 셈입니다.

Aspera FASP의 적응식 속도 제어

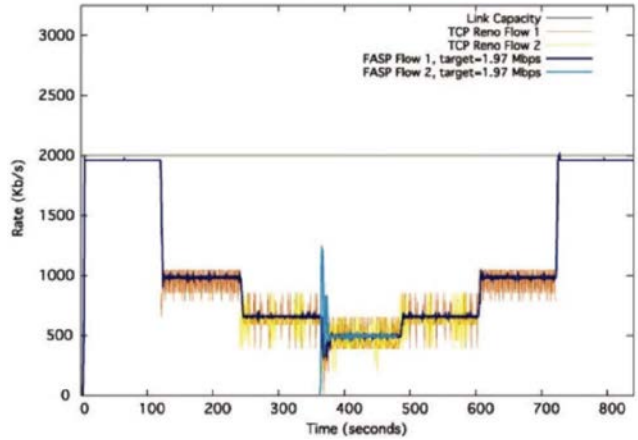


그림 10. FASP는 내부 프로토콜 및 프로토콜 간 공정성을 유지하면서 다른 FASP 및 표준 FASP 트래픽과 네트워크 대역폭을 공유합니다.

Aspera FASP의 속도 제어

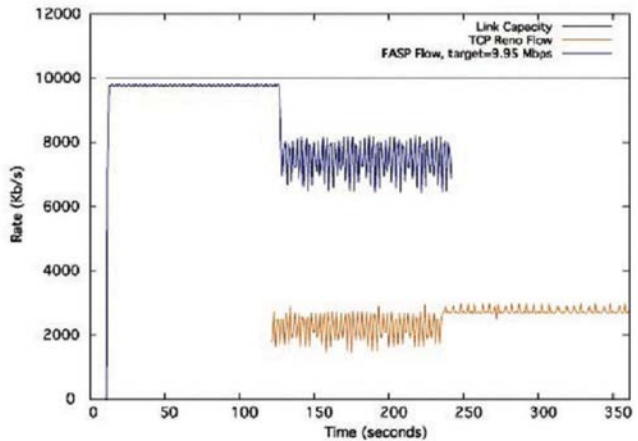


그림 11. TCP가 네트워크 상태의 제약을 받는 경우라도 FASP는 FASP 흐름 간, 혹은 다른 (TCP) 트래픽과의 철저한 공정성을 유지하면서 가용 대역폭을 사용합니다.

FASP 전송 방식은 네트워크 전송 경로의 인위적인 병목 구간을 해소하고 네트워크 대역폭을 마음껏 사용할 수 있는 환경을 최종 사용자에게 선사함으로써 이따금 디스크 IO, 파일 시스템, CPU 스케줄링 등에서 발생하는 새로운 병목 현상을 해소합니다. 이런 요인들은 특히 기가비트 네트워크에서 전송 속도를 최대한 올리는 데 새로운 장애물로 작용합니다. FASP의 적응식 속도 제어 방식에는 상대적으로 느린 스토리지에 기록하는 동안 고속 파일 전송에서 발생할 수 있는 데이터 유실을 방지하기 위해 디스크 흐름 제어 기능도 추가되었습니다. 그리고 이와 유사한 디스크 버퍼용 지연 시간 기반의 모델(특히 출원 중)도 개발되었습니다. 네트워크와 디스크 역학의 다른 시간 척도 때문에 대역폭 변화와 디스크 속도 변화를 모두 수용할 수 있도록 설계된 두 가지 시간 척도 방식이 동시에 채용되었습니다. 세밀한 고속 시간 척도 측면에서는 이를테면, 운영체제 스케줄링으로 인한 주기적인 디스크 속도 저하를 감당할 수 있도록 정교하고 신속한 시간 수신부에 로컬 피드백 메커니즘이 도입된 한편, 개괄적인 저속 시간 척도 측면에서는 대역폭 제어와 디스크 제어를 목적으로 지연 시간 기반의 통합 혼잡 회피 기법이 구현되었습니다. 덕분에 FASP 전송 방식은 가용 네트워크 대역폭과 디스크 속도에 동시에 적응할 수 있습니다.

파일 시스템 병목 현상은 다양한 양상으로 명확히 드러납니다. 실제로 많은 고객들이 작은 파일 한 개를 전송할 때에 비해 그와 동일한 크기의 파일 여러 개를 전송할 때 속도가 크게 저하되는 것을 경험합니다. FASP는 새로운 파일 스트리밍 기법을 이용해서 파일 시스템이 유발한 인위적 병목 현상을 해소하고 다수의 작은 파일을 전송하는 데 이상적인 효율성을 실현합니다. 예를 들어, OC-3를 가득 채운 채 155Mbps의 유효 전송 속도로 2MB 파일 천 개를 미국에서 뉴질랜드로 전송할 수 있습니다.

결과적으로, FASP는 FTP 및 UDT와 같은 TCP 또는 UDP 기반의 파일 전송 기술의 근본적인 병목 현상을 해소하고 공공 및 사설 IP 네트워크를 통한 전송 속도를 크게 개선합니다. FASP는 불완전한 혼잡 제어 알고리즘, (물리적 매체, 교차 트래픽 버스트 혹은 조악한 프로토콜에 의한) 패킷 손실, 그리고 신뢰성과 혼잡 제어의 상충 관계로 인해 파급된 인위적 병목 현상을 해소합니다. 또한 혁신적인 FASP는 디스크 IO, 파일 시스템, CPU 스케줄링 등이 유발하는 새로운 유형의 병목 현상을 해소하고 심지어 가장 길고 가장 빠른 WAN에서도 최대 속도를 발휘합니다. 따라서 FASP는 상용 네트워크를 통해 장거리로 대량의 파일 기반 데이터를 전송하는 TCP로 말미암아 갈수록 커지는 취약점을 해소하고 매일 전 세계로 대량의 디지털 데이터를 옮길 수 있는 차세대 고성능 전송 프로토콜로 손색없습니다.

## IBM 계열사인 Aspera 소개

IBM 계열사인 Aspera는 파일 크기, 전송 거리, 네트워크 상태에 관계없이 가장 빠른 속도로 전 세계 데이터를 옮기는 차세대 전송 기술을 개발해 왔습니다. 특허를 획득하고 에미상 수상 경력에 빛나는 FASP™ 프로토콜을 기반으로 하는 Aspera 소프트웨어는 기존의 인프라를 십분 활용하여 가장 빠르고 가장 안정적인 파일 전송 경험을 선사합니다. Aspera의 핵심 기술은 대역폭에 대한 전례 없는 통제력, 완벽한 보안, 그리고 타협 없는 신뢰성을 보장합니다. 전 세계의 다양한 산업에 종사하는 기업들은 자사의 비즈니스 크리티컬 디지털 자산을 전송하는 데 Aspera 소프트웨어를 적극 활용하고 있습니다.

## 추가 정보

Aspera 솔루션에 대한 추가 정보를 확인하려면 <http://www.asperasoft.com>을 방문하시거나 트위터 [@asperasoft](https://twitter.com/asperasoft)를 통해 팔로우하십시오.



---

© Copyright IBM Corporation 2014

IBM Corporation  
Route 100  
Somers, NY 10589  
U.S.A.

Produced in the United States of America  
August 2014

IBM, IBM 로고, [ibm.com](http://ibm.com) 및 Aspera는 미국 또는 기타 국가에서 사용되는 International Business Machines Corporation의 상표 또는 등록상표입니다. 이와 함께 기타 IBM 상표가 기재된 용어가 상표 기호(® 또는 ™)와 함께 이 정보에 처음 표시된 경우, 이와 같은 기호는 이 정보를 발행할 때 미국에서 IBM이 소유한 등록상표 또는 일반 법적 상표입니다. 또한 이러한 상표는 기타 국가에서 등록상표 또는 일반 법적 상표입니다. 현재 IBM 상표 목록은 웹 "저작권 및 상표 정보"([ibm.com/legal/copytrade.shtml](http://ibm.com/legal/copytrade.shtml))에 있습니다.

Sterling Commerce®, Sterling Information Broker®, 및 Sterling Integrator®는 IBM 회사인 IBM International Group B.V.의 상표 또는 등록상표입니다.

기타 제품, 회사 및 서비스 이름은 해당 회사의 상표 또는 서비스 표입니다.

본 문서는 발행일 기준으로 최신이고 IBM은 이를 통지없이 변경할 수 있습니다. 본 문서에서 언급된 모든 오퍼링이 IBM이 영업하고 있는 모든 국가에서 제공된다는 것을 의미하지는 않습니다.

본 문서에 언급된 성능 데이터 및 인용된 고객 예제는 설명의 목적으로 표시되었습니다. 실제 성능 결과는 특정 구성 및 운영 환경에 따라 다를 수 있습니다. IBM 제품 및 프로그램과 함께 사용된 모든 제품 또는 프로그램의 운영에 관한 평가 및 검증은 전적으로 고객의 책임입니다. 본 문서의 모든 정보는 타인의 권리 침해, 상품성 및 특정 목적에의 적합성에 대한 명시적 보증을 포함하여 명시적이든 명시적이든 어떠한 종류의 보증 없이 "현상대로" 제공됩니다. IBM 제품은 제공된 제품에 적용된 계약의 이용 약관에 따라 보증됩니다.

- 1 Jacobson, V., Braden, R., and Borman, D., *TCP Extensions For High Performance*, The Internet Engineering Task Force (IETF®), May, 1992, <http://www.ietf.org/rfc/rfc1323.txt>
- 2 Ibid, 1992.
- 3 Gu, Yunhong and Grossman, R., *Optimizing UDP-based Protocol Implementations*, 2005, <http://udt.sourceforge.net/doc/pfdnet2005-v8.pdf>

