

# 모바일 기기와 대용량 스토리지간의 I/O latency를 줄이기 위한 iSCSI 중개 서버 시스템

양연<sup>o</sup> 최원일 박명순  
고려대학교 인터넷 컴퓨팅 연구실  
{yy<sup>o</sup>, wonil22, myongsp}@lab.korea.ac.kr

## iSCSI Intermediate Target System for Reducing I/O Latency between Mobile Device and Mass Storage

Yuan Yang<sup>o</sup> Wonil Choi and Myong-Sun Park  
Internet Computing Laboratory, Korea University

### 요 약

모바일 기기가 널리 보급되면서 모바일 기기에서 멀티미디어 데이터나 데이터베이스 같은 대용량 데이터 사용에 대한 요구가 커지고 있다. 따라서 모바일 기기의 저장 공간의 한계를 극복하기 위한 노력의 하나로 iSCSI를 이용해 모바일 기기에 원격 스토리지 서비스를 제공하는 연구가 진행되었다. 그러나 모바일 기기에 iSCSI를 적용했을 때, 지연 시간에 민감한 iSCSI의 구조적인 특성 때문에 iSCSI 클라이언트가 서버에서 멀리 떨어진 위치로 이동하게 되면 iSCSI I/O 성능이 급격히 떨어지는 문제가 발생한다. 본 논문에서는 모바일 기기가 스토리지 서버로부터 거리가 멀어졌을 때, 전송 지연시간이 증가함에 따라 iSCSI 성능이 급격하게 떨어지는 단점을 개선하기 위해 중개서버(Intermediate Target)를 이용해 iSCSI Target을 지역화하여 성능을 높이는 방안을 제안한다.

### 1. 서 론

모바일 기기 시장의 폭발적인 성장과 함께 모바일 기기를 이용한 서비스에 대한 요구가 증가하게 되었다. 유선 환경에서 제공되었던 멀티미디어, 데이터베이스와 같이 대용량의 저장 공간을 필요로 하는 서비스를 무선 환경에 적용하고자 하는 노력이 진행되었지만, 모바일 디바이스의 경우 이동성을 지원하기 위해 크기가 작고 무게가 가벼워야 하기 때문에, 대용량의 하드디스크 대신 플래쉬 메모리를 이용하여 데이터를 저장한다. 이러한 모바일 기기의 제한된 저장 공간은 유선 환경의 다양한 서비스를 모바일 기기에 적용하는데 있어 제약사항이 되었다. 이와 같은 모바일 디바이스의 제한된 저장 공간을 극복하고 대용량 데이터를 저장하거나 다양한 어플리케이션 서비스를 제공하기 위해 모바일 디바이스를 위한 원격 스토리지 서비스의 필요성이 대두되었다[1].

본 논문은 원격 스토리지 서비스에서 전송 지연시간을 줄임으로써 효과적으로 원격 스토리지 서비스를 실현하는 방안에 대하여 제안한다.

### 2. 연구 배경

데이터 스토리지 장비들을 묶어주기 위해 스토리지 네트워크 표준에 기반하여 Internet Engineering Task Force(IETF)에 의해 개발된 iSCSI는 원격 스토리지 서비스에 대한 요구를 충족시켜 주었다. iSCSI 표준으로 지정된 SAM-3(SCSI Architecture Model-3)에서 연속된 SCSI Command들은 순차적으로 처리된다[2]. 이전 커맨드 처리가 완료된 후에 그 다음 커맨드가 처리되는 방식으로, 이러한 순차 처리 특성은 SCSI 프로토콜이 Latency에 민감한 원인이 된다. Initiator는 SCSI 커맨드를 생성해 Target으로 보낸 뒤 그 결과를 받을 때까지 Waiting시간이 존재한다. 여기서 Initiator와 Target의 거리가 멀어질수록 전송 지연에 의한 Latency가 커지게 되어 그만큼 Waiting시간도 길어지게 된다.

특히 데이터 크기가 큰 경우에는 대역폭에 영향을 받는 전송 지연이 커지게 되고, 데이터 크기가 작은 경우에는 전송 지연 보다

는 물리적 네트워크의 거리에 영향을 받는 전송 지연이 iSCSI 지연에 큰 비중을 차지하게 된다. 모바일 기기의 경우 보통 작은 크기의 블록 단위로 I/O를 수행하기 때문에 대역폭에 의한 지연 보다는 패킷이 이동하는 물리적 네트워크 길이에 의한 지연이 iSCSI 성능을 저하시키는 주된 원인이라 할 수 있다[3].

### 3. 관련 연구

디스크 I/O성능을 높이기 위해 캐시를 이용하는 방안은 이미 오래전부터 연구되어왔다. 일반적으로 로컬 디스크의 성능을 높이기 위해 메모리의 일부 공간을 버퍼캐시로 이용하는 연구들이 대부분이었다. 하지만, 최근들어 iSCSI가 표준으로 자리잡고 iSCSI를 지원하는 제품들이 등장하면서 iSCSI의 성능을 높이기 위해 캐시를 이용하는 연구들이 활발히 진행되었다.

iCache[4]는 클라이언트 시스템의 로컬 캐시를 이용해 iSCSI 성능을 높이기 위한 연구이다. iCache는 로컬 시스템에 iSCSI 데이터를 위한 캐시 공간을 두고 iSCSI 블록 데이터를 캐싱해 iSCSI의 네트워크 블록 I/O 회수를 최소화한다. 따라서, 디스크 I/O가 발생할 때마다 네트워크를 통해 I/O를 수행하는 것이 아니라 캐시된 데이터를 읽거나, LogDisk에 담았던 쓰기 블록들을 한꺼번에 전송함으로써 iSCSI 성능을 높인다.

그러나 iCache는 로컬 캐시를 구현하기 위해 추가적인 메모리와 하드디스크 공간을 필요로 하기 때문에 저장 공간이 부족한 모바일 디바이스에 적용하기 어렵다.

### 4. 중개 서버 (Intermediate Target)

본 장에서는 모바일 클라이언트가 스토리지 서버와 거리가 멀어졌을 때 iSCSI 성능이 떨어지는 문제를 해결하기 위해 중개 서버를 이용한 iSCSI Target의 지역화 하는 방안을 제안한다. iSCSI 모바일 클라이언트가 사용할 블록들을 가까운 중개 서버에 미리 읽거나 쓰기 블록에 대한 응답을 중개 서버에서 전송함으로써 iSCSI 응답 지연 시간을 최소화 한다. 문제를 간단히 하기 위하여

제한된 방안에서는 다음과 같은 3가지 사항을 가정한다.

- 분산된 스토리지 서버 환경을 가정한다. 대용량 스토리지 서버를 중심으로 iSCSI 중개 서버는 여러 지역에 걸쳐 분산되어 있으며 iSNS(Internet Storage Name Server)[5]를 통해 모바일 클라이언트는 가장 가까운 위치의 중개 서버와 연결된다.
- 전송 거리가 늘어날수록 물리적 매체의 전송 지연, 중간 라우터들의 큐잉 지연의 합이 증가한다.
- iSCSI 지연은 거리에 의한 전송 지연, 큐잉 지연과 대역폭에 의한 전송 지연만을 고려하고, 중단 노드의 처리 지연은 고려하지 않는다.

#### 4.1 iSCSI 중개 서버

그림 1에서 모바일 노드에 iSCSI 서비스를 제공할 때 A지역에서 서비스 받던 모바일 노드가 C지역으로 이동하는 경우 서비스를 제공하는 서버와 모바일 노드간의 거리가 멀어지게 되어 패킷 이동 지연 시간이 길어지게 된다. iSCSI 프로토콜은 크기가 작은 제어 패킷들이 응답 지연 시간에 영향을 주기 때문에 서버와 클라이언트의 거리가 멀어지게 되면 그만큼 링크 활용율이 떨어지면서 iSCSI 성능이 낮아지게 된다.

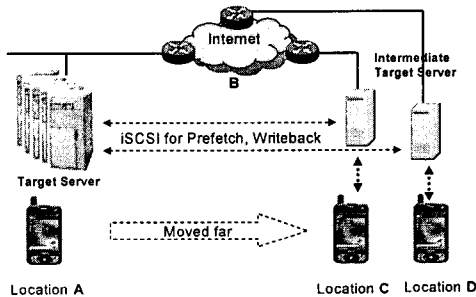


그림 1 중개 서버를 이용한 iSCSI Target 지역화

서버와 클라이언트간의 패킷 전송 지연 시간을 줄이고, 네트워크 대역폭 활용율을 높이기 위해 iSCSI 중개 서버를 도입한다. 중개 서버를 이용해 iSCSI 읽기 블록을 미리 읽거나 쓰기 요청에 대한 응답을 빠르게 줄 수 있다. iSNS에 의해 모바일 클라이언트로부터 가장 가까운 중개 서버가 선택되고, 클라이언트는 중개 서버와 iSCSI 프로토콜을 수행하게 되고 중개 서버는 다시 스토리지 서버와 iSCSI 프로토콜로 연결된다. 따라서, 클라이언트는 원거리의 스토리지 서버와 직접 통신하는 것이 아니라 가까운 중개 서버로부터 원격 스토리지 서비스를 수행하게 되기 때문에 I/O요청에 대한 응답 지연 시간이 짧아진다.

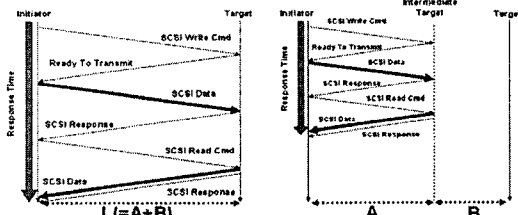


그림 2 Initiator/Target 직접통신과 Intermediate Target을 통한 통신 비교

그림 2는 각각 iSCSI Initiator와 Target이 직접 통신 하는 경우와 iSCSI 중개 서버를 두었을 때, 응답 지연 시간이 차이가 나는 것을 보여준다. Write 커맨드를 처리하기 위해서는 총 3개의 제어 패킷을 주고받아야 하고, Read 커맨드 또한 2개의 제어 패킷을 주고받는다. 제어 패킷들은 크기가 작지만 여전히 이동 지연 시간(propagation, queuing delay)을 갖기 때문에 Initiator와 중개 서버의 거리가 짧다면 I/O 응답 시간은 그만큼 짧아지게 되는 것이다. 중단 노드인 Initiator와 Target의 패킷 처리 지연시간을 무시한다면 중개 서버를 이용했을 때와 이용하지 않았을 때의 응답 지연 시간은 A/L의 비율이 될 것이다.

#### 4.2 System Architecture

그림 3은 중개 서버 시스템의 모듈 구성도이다. 중개 서버 시스템은 Initiator모듈과 Target모듈 그리고 블록 관리 모듈로 구성되어 있다. iSCSI 중개 서버는 두개의 iSCSI 연결을 갖는다. 하나는 중개 서버와 클라이언트와의 연결이고 다른 하나는 중개 서버와 스토리지 서버와의 연결이다. Target모듈은 모바일 클라이언트의 iSCSI Initiator와 연결되고 Initiator모듈은 스토리지 서버의 iSCSI Target 모듈과 연결된다. 두 모듈은 일반적인 iSCSI Target / Initiator 모듈과 동일한 역할을 수행한다. 하지만, 전자는 모바일 클라이언트와 I/O를 수행하기 위한 iSCSI 프로토콜을 수행하고, 후자는 iSCSI 블록을 미리 읽거나 쓰기 블록을 서버로 전달하기 위해 이용된다. 중개 서버의 Target / Initiator 모듈은 블록 관리 모듈에 의해 제어된다. iSCSI 버퍼는 가장 최근에 사용한 블록을 버퍼에 남겨두기 위해 FIFO 방식으로 관리된다.

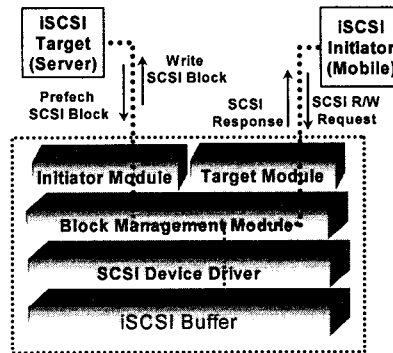


그림 3 Intermediate Target System Architecture

그림 4는 클라이언트 I/O 요청에 대한 블록 관리 모듈의 처리 알고리즘을 보여준다. 읽기 요청시 Target모듈은 iSCSI 버퍼에 블록이 존재하는지 검색한 후 존재하면 클라이언트로 전송해주고 존재하지 않으면, 스토리지 서버와 연결되어 있는 Initiator모듈을 통해 해당 블록을 요청해 패킷을 가져와 클라이언트에 서비스 해준다. 이때, 클라이언트가 요청한 블록의 논리적인 주소 다음 블록을 미리 읽게 된다. 쓰기 요청의 경우에는 중개 서버가 쓰기 데이터 블록을 받은 후 클라이언트에 바로 응답을 주고 그와 동시에 쓰기 블록을 스토리지 서버로 전송해 준다. 따라서 모바일 클라이언트의 iSCSI Initiator는 근거리의 중개 서버와 iSCSI프로토콜을

```

When Intermediate Target Receives I/O Request
i: Requested Block
{
  if(Read Op)
  { //Client's I/O Request is Read Operation
    if(i exists in iSCSI Buffer) {
      Send i and Response to Initiator
    }
    else {
      Send Read Request of i to Target
    }
    Send Read Request of i+1 from Target
    and Put at Head of iSCSI Buffer
    //Prefetch next block from Storage Server
  }
  else (Write Op)
  { //Client's I/O Request is Write Operation
    Put i at Head of iSCSI Buffer
    Send Response to Initiator
    Send Write Request of i to Target
  }
}
    
```

그림 4 Block Management Algorithm

수행하는 효과를 얻게 된다.

5. 시뮬레이션

시뮬레이션에서는 iSCSI Target과 Initiator의 거리 및 중개 서버의 위치에 따른 iSCSI 성능의 변화를 측정하였다. 또한 읽기 오퍼레이션의 경우 미리 읽은 블록의 히트율에 따른 성능의 변화를 측정하였다.

5.1 시뮬레이션 환경

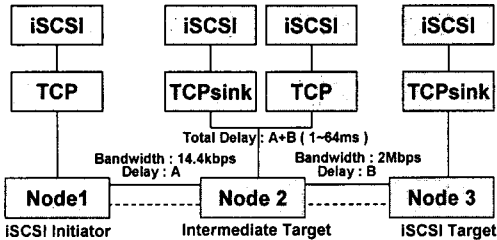


그림 5 NS2 네트워크 구성

Network Simulator 2.27를 이용해 iSCSI Initiator, Intermediate Target 그리고 iSCSI Target간의 네트워크를 구성 하였다. 그림 5는 시뮬레이션에 사용된 네트워크 구성을 나타낸다. 대역폭의 경우 모바일 노드가 일반적으로 사용하는 무선 환경(CDMA 2000 1x)을 고려하여 Node1과 Node2 사이를 14400bps로 제한하였다. iSCSI Initiator와 Intermediate Target의 거리에 따른 성능 변화를 측정하기 위해 지연 시간은 노드간의 거리에 비례한다는 가정을 두고 지연 시간을 1~64ms로 변화시키면서 iSCSI 성능 변화를 측정하였다. 데이터 크기는 대표적인 모바일 기기의 하나의 Windows CE기반의 PDA에서 사용되는 SCSI 블록 크기인 512바이트 크기로 지정하였다.

5.2 실험결과 및 분석

그림 6은 iSCSI 읽기 동작에 대해 iSCSI Target과 Initiator간의 지연시간이 증가될 때, 데이터 타입에 따라 중개 서버를 사용했을 때와 하지 않았을 때 성능 차이를 보여준다. 지연 시간 A와 지연 시간 B의 비율은 7:3 으로 지정하여 실험한 결과 전체 지연 시간이 커질수록 즉 iSCSI Initiator와 Target의 거리가 멀어질수록 성능 차이가 점점 크게 나타났다. 동영상, 텍스트, 애플리케이션 데이터의 경우 지연 시간이 64ms인 경우에 각각 30%, 27%, 17%의 성능 향상이 발생하였다. 데이터 타입에 따라 성능 차이가 다르게 나타나는 것은 읽기 블록에 대한 iSCSI 버퍼 히트율에 차이가 발생하기 때문이다. 순차 접근이 이루어지는 동영상의 경우 90%의 히트율을 가정하였고, 텍스트, 애플리케이션 데이터의 경우 각각 80, 50%의 히트율을 가정하였다.

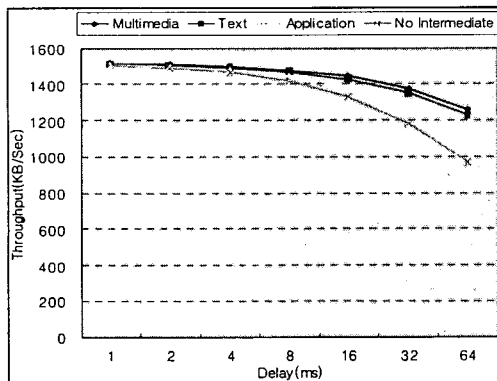


그림 6 데이터 타입에 따른 처리량의 변화(Read Operation)

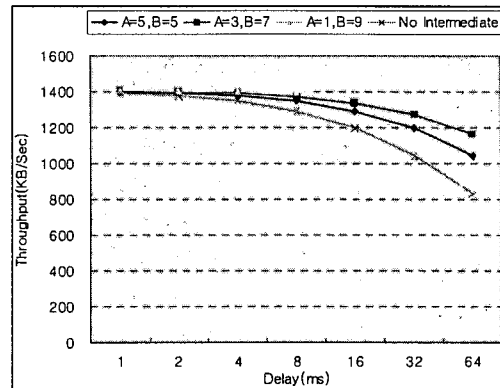


그림 7 거리 비율에 따른 처리량의 변화(Write Operation)

그림 7은 iSCSI 쓰기 작업에 대해 지연 시간에 따른 처리량의 변화를 보여준다. 쓰기 요청의 경우 데이터 타입에 따라 성능이 달라지는 것이 아니기 때문에, 중개 서버의 위치에 따라 처리량의 어떻게 달라지는지 알아보았다. 쓰기 작업도 읽기 작업과 마찬가지로 거리가 짧을 때는 성능 차이가 크지 않았지만, 거리가 멀어지게 되면 성능 차이가 점점 커지게 된다. 또한 iSCSI Initiator와 중개 서버의 지연 시간이 작을수록, 즉 거리가 가까울수록 성능이 높게 나타나는 것을 볼 수 있었다. 읽기 작업의 경우와 마찬가지로 전체 지연 시간이 64ms인 지점에서 iSCSI 성능 차이가 가장 컸으며, A:B=5:5, A:B=3:7, A:B=1:9인 경우에 각각 25%, 40%, 59%의 성능 향상이 발생하였다. 따라서 제안된 기법은 중개 서버가 initiator에 가까울수록 iSCSI 성능이 매우 높아지는 것을 알 수 있다.

6. 결론 및 향후 과제

본 논문에서는 모바일 기기에 iSCSI기반의 원격 스토리지 서비스를 적용했을 때 성능이 저하되는 문제를 개선하기 위해 iSCSI Intermediate Target을 도입하는 방안을 제안하였다. 제안 방안을 통해 iSCSI Target을 지역화 시킴으로써 iSCSI Initiator와 Target이 멀어질 때 링크 활용율이 떨어지고 iSCSI 성능이 급격히 떨어지는 문제를 개선하였다. 시뮬레이션 결과를 통해 iSCSI 프로토콜은 Initiator와 Target의 거리(지연시간)가 멀어지게 되면 iSCSI 성능이 급격히 저하되는 것을 알 수 있었고 중개 서버가 Initiator로부터 가까울수록 iSCSI 성능이 높아지는 것을 볼 수 있었다. 향후 연구 과제는 데이터 타입에 따라 다른 캐싱 알고리즘을 적용하거나, 현재의 FIFO 버퍼 관리 알고리즘을 개선해 iSCSI 버퍼 히트율을 높이는 것이다. 그러기 위해서는 데이터 타입에 대한 분류 기준과 타입 정보를 어떤 방법으로 알려줄 것인지 그리고 버퍼 관리 알고리즘에 대한 연구가 이루어져야 할 것이다.

References

[1] Sula Park, Bo-Seok Moon and Myong-Soon Park, "Design, Implementation and Performance Analysis of the Remote Storage System in Mobile Environment", International Conference on Information Technology & Applications(ICITA 2004), pp.8-11, Jan. 2004  
 [2] SAM-3 : Information Technology - SCSI Architecture Model 3, Working Draft, T10 Project 1561-D, Revision7, 9 May, 2003.  
 [3] Yingping Lu and David H.C. Du, University of Minnesota, "Performance Study of iSCSI-Based Storage Subsystems", IEEE Communication Magazine, pp.76-82, August, 2003  
 [4] Xubin He, Qing Yang and Ming Zhang, "A caching strategy to improve iSCSI performance", LCN 2002 27th Annual IEEE Conference, pp.278-285, Nov. 2002