

## 분산된 웹 서버 환경을 이용한 개선된 모바일 서비스 알고리즘

김선호 박진홍\* 신용태  
숭실대학교 컴퓨터학과  
[shkim98@dongduk.ac.kr](mailto:shkim98@dongduk.ac.kr)

### A Advanced Mobile Service Algorithm on the Distributed Web Server

Seonho Kim Jinhong Park\* Yongtae Shin  
Dept. of Computing, Soongsil University

#### 요 약

최근 인터넷상에서 사용자와 대용량 콘텐츠의 급증으로 서버와 네트워크의 부하가 가중되고 서비스의 질이 떨어지는 문제를 해결하기 위해 콘텐츠를 다수의 지역 서버에 분산하여 서비스 하는 기술들이 대두되고 있다. 이러한 환경에서는 클라이언트의 요청에 대해 다수의 웹 서버 중 가깝고 트래픽이 적은 서버로 요청을 재 전달하여 빠르고 안정된 멀티미디어 서비스를 기대할 수 있다.

그러므로 본 연구에서는 이러한 환경을 이동 환경에 적용하여 이동하는 클라이언트의 요청에 대해 적절한 서버가 응답함으로써 이동 호스트에서도 계속적으로 증가하는 대용량, 실시간 멀티미디어 콘텐츠의 빠르고 안정적인 서비스를 가능하게 할 수 있는 알고리즘을 제안하고자 한다.

#### 1. 서론

본 연구의 목적은 이동 환경에서의 효율적인 대용량 콘텐츠의 전송을 위해 기존 유선망의 분산된 웹 서버 환경을 이용하여 가깝고 트래픽이 적은 서버로부터 빠르고 안정적인 서비스를 제공할 수 있는 서버 선택 알고리즘을 제시하고자 한다.

최근 인터넷 사용자 수와 인터넷상에서 서비스되는 콘텐츠가 급격히 증가함에 따라 서버와 네트워크에 엄청난 부하가 발생하고 있다. 이와 같은 서버의 부하와 네트워크의 지연을 인터넷상에 콘텐츠를 분산시킴으로써 해결하기 위한 방안으로 프락시 캐쉬, CDN(Content Distribution Network) 등 같은 방법들이 제안되어 사용되고 있다. 콘텐츠를 인터넷상에 분산하여 설치한 다수의 서버에 복제하여 클라이언트와 가까운 서버로부터 서비스하게 함으로써 서비스의 질을 높이도록 하는 서버와 콘텐츠의 분산기술은 데이터 저장 비용이 네트워크 회선 비용보다 저렴한 현실을 감안할 때 매우 적절한 대안이라 할 수 있다[1].

반면, 랩탑, PDA 등 개인용 휴대 컴퓨터의 사용이 늘어나고, 컴퓨터 하드웨어의 크기와 성능이 크게 성장함에 따라 클라이언트의 물리적 위치에 관계없이 어느 곳에서나 원하는 정보를 손쉽게 접근하고자 하는 요구는 인터넷과의 연동을 통해 기존 유선망에서 제공해 왔던 다양한 서비스를 이동 환경에서도 가능케 하도록 추구

하고 있으며 차세대 인터넷은 유선과 무선이 혼합된 초고속 네트워크 기반 하에 대용량 멀티미디어 데이터 전송을 가능하게 할 것이다.

이에 본 논문에서는 유선망에서의 분산된 서버 환경을 이동 환경에서 이용할 경우 적절한 시스템 모델과 서버 선택 알고리즘을 제안한다.

#### 2. 관련연구

분산된 서버 환경에서 가장 중요한 이슈는 분산된 서버 중 가장 적절한 서버로 클라이언트의 요청을 전달하여 빠른 응답을 받도록 하는 것이라 할 수 있다.

요청을 전달 받은 요청 리다이렉터가 적절한 서버를 선택하여 응답하는 방식의 기존 연구는 다음과 같다.

##### 2.1 HTTP Modification

요청 라다이렉터가 웹 서버로 동작하여 클라이언트 요청시 HTTP 프로토콜의 302 응답코드(Moved Temporarily)와 함께 인접 서버의 위치를 반환하고 클라이언트는 302 응답 메시지에 포함된 서버 주소로 요청을 한다. 이 경우 클라이언트가 바로 요청 리다이렉터에게 요청을 보내기 때문에 요청한 클라이언트의 정확한 위치를 파악할 수 있고 구현하기가 매우 간단하지만 모든 수신되는 패킷에 대해 적절한 서버로 라우팅 되도록 IP 주소를 변경해 주어야 하므로 부가적인 오버헤드와 지연이 발생한다[2].

## 2.2 DNS-based Redirection

클라이언트의 DNS 질의에 대해 분산 서버 중 하나의 IP 주소를 반환한다. 클라이언트 측의 아무런 수정 없이 기존 인터넷상의 DNS 인프라를 그대로 사용할 수 있고 콘텐츠 제공자 입장에서도 단지 콘텐츠가 저장되어 있는 분산 서버들의 위치 목록만을 DNS에 추가하면 된다. 그러나, 대부분의 경우 클라이언트는 TCP/IP에 설정된 지역 DNS에게 질의를 맡기게 되는데 지역 DNS에 의해 질의를 받은 요청 리다이렉터는 클라이언트의 정확한 위치를 알 수 없기 때문에 인접한 서버 선택을 보장할 수 없으며 하나의 지역 DNS에 의한 요청은 같은 서버로 리다이렉션 되므로 같은 지역 DNS에 접근한 클라이언트가 많은 경우 특정 서버에 부하가 집중될 수 있다[3].

## 3. 제안하는 시스템 모델

### 3.1 개요 및 목표

이동 환경은 유선 환경에 비해 상대적으로 신뢰성이 떨어지고 낮은 대역폭을 지원하며, 이동 호스트 또한 제한된 자원을 갖고 있으므로 복잡한 연산을 수행하기가 곤란하다[4].

그러므로 이동 환경에서의 서버 선택은 통신 제어를 위한 오버헤드 메시지를 가능한 줄이고 시스템 부하가 많은 부분들은 되도록 고정 호스트에서 처리 할 수 있도록 고려하여 설계하여야 한다.

### 3.2 시스템 모델

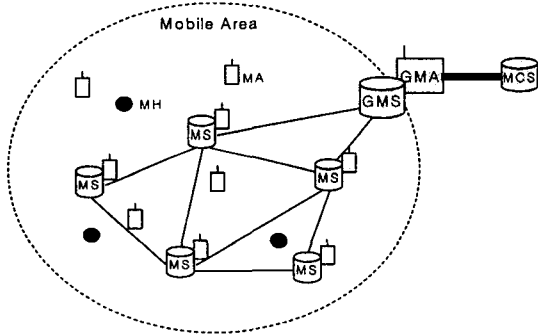


그림 1. 시스템 모델

제안하는 시스템 모델은 그림 1과 같은 구조를 가지며 모든 서버들 간에는 콘텐츠의 동기화가 이루어지고 있음을 가정한다. 시스템 모델의 각 노드들의 기능은 다음과 같다.

- 1) 이동 콘텐츠 서버(MCS : Mobile Content Server) : 웹 서버 중 Mobile 클라이언트를 위한 콘텐츠를 포함하고 있는 서버
- 2) 게이트웨이 이동 서버 (GMS : Gateway Mobile Server) : 이동 콘텐츠 서버와 일관된 콘텐츠를 보관하고 있는 고정된 데이터베이스 서버로 상대적으로 용량이 적은 MS를 지원한다.
- 3) 게이트웨이 이동 에이전트 (GMA : Gateway Mobile Agent) : 유선망에 유선으로 연결되어 있으면서 무선을 지원하는 고정된 지역 라우터로 지역 내의 이동 에이전트와 무선 통신 채널을 통해 통신하고 관리한다.
- 4) 이동 서버(MS : Mobile Server) : GMS에서 프리패치한 콘텐츠를 보관하고 있는 고정된 데이터베이스 서버
- 5) 이동 에이전트 (MA : Mobile Agent) : 이동 호스트와 통신할 수 있는 인터페이스를 가지고 있는 고정 라우터로 호스트의 이동성을 지원한다.
- 6) 이동 호스트(MH : Mobile Host) : 무선 통신 채널을 통해 하나의 MA와 통신하며 이동성을 갖고 있다.

## 4. 이동 환경에서의 서버 선택 알고리즘

### 4.1 개요

가장 적절한 서버라는 것은 서버의 부하가 적으며 이동 호스트와 지리적으로 가깝고 이동 호스트와의 사이에 트래픽이 적은 서버라고 할 수 있다.

본 논문에서는 게이트웨이 이동 에이전트가 관할하고 있는 영역 내의 이동 서버의 부하 정보를 주기적으로 갱신하고 있다가 이동 호스트의 요청에 대해 전체 서버 중 부하가 적은 서버  $k$ 개를 선택하여 이동 호스트에게 보내면 이동 호스트가  $k$ 개의 서버로 신호 메시지를 보내어 가장 빠른 응답을 한 서버로 요청하는 메커니즘을 제안한다. 이것은 각 이동 서버가 그들의 부하 정보를 주기적으로 게이트웨이 이동 에이전트에게 보냄으로써 구현한다.

이것은 서버 측면과 클라이언트 측면에서의 선택을 모두 고려한 2단계 서버 선택으로 부하가 적고 트래픽이 한가하며 거리가 가까운 서버를 선택하여 서비스 요청을 함으로써 빠른 응답 시간을 기대할 수 있다.

이동 호스트의 서버 선택 과정은 그림 2에서와 같이 다음 단계를 따른다.

- 1) MA는 에이전트 광고 메시지를 주기적으로 방송한다.
- 2) MH는 MA를 통해 GMA에 자신의 존재를 알리고

응답 메시지를 받는다.

- 3) GMA로 서비스를 요청한다.
- 4) GMA가 영역 내에서 부하가 적은 이동 서버  $k$ 개를 후보 서버 set으로 응답한다.
- 5) 이동 호스트가 다시  $k$ 개의 서버로 신호 전달한다.
- 6) 가장 먼저 응답이 온 서버에게로 요청을 보내어 서비스를 받는다.
- 7) 이동
- 8) MH는 MA의 광고 메시지에 의해 이동 사실을 안다.
- 9) 새로운 MA, GMA에 이동을 알리고 이동 후의 서버 선택을 다시 한다.

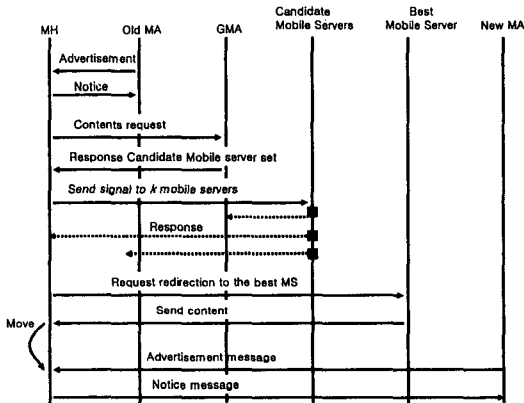


그림 2. 서버 선택 과정

#### 4.2 서버 선택 알고리즘

서버와 클라이언트에 의한 2단계 서버 선택 알고리즘을 제안한다.

##### 4.2.1 이동 게이트웨이 서버의 서버 선택 알고리즘

게이트웨이 이동 에이전트가 이동 서버들로부터 부하 정보에 관한 피드백을 주기적으로 받아, 사용 가능하며 (active) 부하가 적은 서버  $k$  개를 선택한다. 이때 서버의 부하 메트릭으로는 CPU 점유율, 인터페이스 수, 연결된 서비스 수, 기존 부하 정보가 고려된다.

##### 4.2.2 이동 호스트의 서버 선택 알고리즘

이동 호스트가 게이트웨이 이동 에이전트에서 부하에 기반하여 선택한  $k$  개의 후보 서버에 신호 메시지를 보내어 응답시간이 가장 빠른 서버를 선택하여 요청을 전달한다. 이때, 모든 서버들의 응답을 기다리지 않는다. 응답시간은 거리, 홑 수, 트래픽에 의해 영향을 받는다.

#### Server-side Server Selection Algorithm

##### Select Candidate Servers (CSs)

```

candidate_servers (s, S) {
  for each server si in server set S,
    Loadi = MeasureLoad(i);
  sort Load ;

```

```

Scandidate <-- least loaded server of top k servers;
Candidate Server Set <-- Scandidate ;

```

##### MeasureLoad(j)

```

Loadrate = Loadprevious*k + Loadcurrent*(1-k) for some k < 1
Loadcurrent = (CPUload + Bufferjob + Servicecnt) * Loadrate
Loadprevious = Loadcurrent
return Loadcurrent

```

#### Client-side Server Selection Algorithm

##### Select Best Server

```

Best_server (s, CSs) {
  for each candidate server i ∈ CSs
    Syn to i ;
  if response i arrived
    forward request to server i;

```

#### 5. 결론 및 향후 연구 계획

이동 호스트의 적절한 서버 선택을 위한 모델 정의 및 알고리즘 제안은 계속적으로 증가하는 대용량 콘텐츠의 이동 호스트 서비스를 위해 매우 의미 있는 일이라고 할 수 있다. 향후 이동 호스트의 이동에 따른 서버 선택 알고리즘을 덧붙여 제안하고 실험하여 검증할 것이다.

#### 6. 참고문헌

- [1] G. Peng, "CDN: Content Distribution Network", in *Stony Brook University Tech. Reports*, TR-125, 2003.
- [2] M. Rabinovich and O. Spatscheck, "Web Caching and Replication", published by Addison Wesley Professional, 2002.
- [3] A. Shaikh, R. Tewari and M. Agrawal, "On the Effectiveness of DNS-based Server Selection," in *Proceedings of IEEE INFOCOM 2001*, 2001.
- [4] S. K. Madria and B. Bhargava, "On the Correctness of a Transaction Model for Mobile Computing", in *Proceedings of DEXA '98*, 1998.