

# CDN에서 서비스 호의 수 및 응답 시간을 고려한 서버 선택 알고리즘

한상희\*, 김상식\*, 김상하\*

\* 충남대학교 컴퓨터과학과

## A Server Selection method with consideration about Round-Trip Time and Actual Serving Calls in CDN

Sang-hee Han\*, Sang-sik Kim\*, Sang-ha Kim\*

\* Dept. of Computer Science, Chungnam National University

{shhan, sskim, shkim}@cclab.cnu.ac.kr

### 요 약

CDN (Content Delivery Network) [1]은 분배서버와 네트워크의 상태를 고려하여 다양한 서비스의 콘텐츠를 보다 효율적으로 전송할 수 있도록 설계된 네트워크이다. 이러한 CDN을 구성하기 위해서는 네트워크 계획, 콘텐츠 할당, 분배서버 선택 등을 고려해야 한다. 특히, 분배서버 선택을 위한 기존의 연구들은 분배서버 선택 시 모든 분배서버들에 대한 측정된 값 (응답 시간, 지연)이나 실시간 정보를 기준으로 분배서버를 선택하게 되므로 호 봉쇄 확률을 고려하지 않게 되고 제어 프로토콜로 인한 오버헤드가 많아지게 된다. 이를 해결하기 위하여 본 논문에서는 분배서버에서 서비스하고 있는 호의 수와 응답 시간을 고려한 새로운 분배서버 선택 알고리즘을 제안한다. 제안된 알고리즘에서 CDN의 분배서버들은 서비스하고 있는 호의 수에 따라 논리적 최소 힙(heap) 구조로 구성되며 이를 이용하여 각 호스트는 자신에게 적당한 분배서버들의 서브 서버 집합을 구성함으로써 오버헤드를 줄일 수 있는 메커니즘을 제안한다.

### 1. 서 론

CDN은 다양한 웹 콘텐츠, 그 중에서 대역폭이 큰 콘텐츠들의 전송을 향상시키기 위하여 여러 분배서버들을 적절히 배치하고 이들을 통하여 서비스를 제공할 수 있게 하는 네트워크를 일컫는다. 이러한 CDN을 구성하기 위해서는 기본적으로 각 분배서버들을 네트워크의 어느 부분에 배치할 것인가의 문제, 각 서버에 어떠한 콘텐츠를 할당 할 것인가의 문제, 그리고 마지막으로 사용자들이 어떤 분배서버를 선택하게 할 것인가에 대한 세가지 큰 이슈가 존재하게 된다. 앞에서 살펴본 이슈 중에서 앞의 두 가지 문제는 서비스 제공자 측면에서의 성능 향상에 관한 문제인 반면 마지막 문제는 사용자 측면에서의 효율적인 알고리즘을 통하여 성능 향상을 도모할 수 있는 문제로 인식되고 있다.

이러한 효율적인 분배서버 선택을 위한 다양한 메커니즘이 제안되고 있다. 대표적인 메커니즘으로는 DNS(Domain Name System) 리디렉션[2]이 있다. [2]는 많은 상용 CDN 서비스에서 사용하고 있는 기술로써 사용자의 지역 DNS 서버로부터 DNS 서비스 요청을 받은 권위 있는 DNS 서버가 가장 가까운 분배서버의 정보를 알려주게 된다. 그러나, 분배서버 선택 시에 라우터 홉수나 응답 시간과 같은 단순한 기준치를 이용하므로 지속적으로 서비스 질을 최적화하는 분배서버 선택을 할

수 없게 되며 호 봉쇄 확률을 고려하지 않는 문제점이 생겨나게 되었다. 이를 해결하기 위하여 실시간 정보를 이용하는 새로운 메커니즘인 [3]이 제안되었다. 이 메커니즘은 요청이 있거나 또는 주기적으로 모든 분배서버들과 연결된 링크의 상태를 파악하여 최적의 분배서버를 선택하는 방법이다. 그러나 이 메커니즘 또한 정확한 실시간 정보를 얻기 위해서는 비용이 많이 들고, 실시간 정보 수집으로 인해 최대 부하 시에 혼잡을 발생시키는 원인이 될 수 있는 문제점을 남기게 되었다.

즉, 현재의 분배서버 선택 알고리즘의 대표적인 문제점으로는 하나의 메트릭에 의존함으로써 인하여 호의 서비스 확률이 낮아지거나 또는 이러한 서버 선택을 위한 제어 프로토콜로 인한 오버헤드가 급격히 증가한다는 것이다. 이를 해결하기 위하여 본 논문에서는 각 CDN을 구성하는 분배서버들을 현재 서비스 하고 있는 호의 수에 따라 논리적 최소 힙 구조로 구성하며 이를 통하여 적절한 서브 서버 집합을 구성하는 방법을 제안한다. 논리적인 최소 힙 구조를 통하여 각 호스트는 현재 가장 적은 수의 서비스를 제공하고 있는 분배서버를 빠르고 쉽게 찾을 수 있다. 또한 이들에 대한 응답 시간을 측정하여 네트워크 상태의 변화에 동적으로 대처할 수 있다. 두 파라미터의 값을 이용하여 정해진 기준값 보다 메트릭이 작은