

지능형 디스크 기술에 관한 조사 연구

조인순*, 김기영*, 엄현영*

*서울대학교 컴퓨터공학부

e-mail : {ischo, kykim, yeom}@dcslab.snu.ac.kr

A Survey of Intelligent Storage Subsystems

In-Soon Cho*, Ki Young Kim*, Heon Young Yeom*

*Dept. of Computer Science & Engineering, Seoul National University

요 약

지능형 디스크 시스템은 디스크 내부에 CPU, 메모리 등을 내장한 스토리지 서브시스템으로서, 디스크 자체에서 프로세싱이 가능한 장점을 가진다.

1. 서론

지능형 디스크 시스템은 디스크 내부에 CPU, 메모리 등을 내장한 스토리지 서브시스템으로서, 디스크 자체에서 프로세싱이 가능한 장점을 가진다. 이러한 개념은 1970년대 후반부터 제시되었으나, 서브시스템을 구성하기 위한 하드웨어 가격 문제로 인해 크게 주목 받지 못했다. 그러나 최근 CPU나 메모리 등의 지속적인 가격 하락으로 인해 이러한 접근 방법에 대한 관심이 고조되고 있다.

2. 지능형 디스크 시스템

Processing을 disk 근처로 옮기자는 개념은 새로운 것이 아니며, 실제로 1970년대 후반과 1980년대 초반에 hardware database machine에 대한 활발한 연구가 있었다. 이러한 연구들은 4가지 카테고리로 나눌 수 있는데, processor per head, processor per track, processor per disk, multi-processor cache 등을 이용하자는 것이었다. 이러한 시스템들은 실제로 사용되지는 않았는데 가장 주요한 이유는 가격 대비 성능 때문이다. 즉, 대부분의 database machine들이 associative disk, associative CCD device, magnetic bubble memory 같은 non-commodity hardware를 사용했는데, 이런 extra cost를 들일 만큼의 성능 향상을 보여주지 못했기 때문이다. Select와 같은 simple operation에서만 가시적인 성능 향상을 보여주고, sort나 join 같은 complex operation에 대해서는 별 효과를 보여주지 못했다.

그러나 1990년대 후반에 database machine에 대한 논의가 다시 활발[1]해지는데, 이는 두 가지 요인에 기반한다. 우선 disk에 들어가는 CPU와 메모리 용량이 증가함에 따라 commodity disk로 processing을 하는 일이 더 이상 무리한 일이 아니게 되었다. 실제로 2003년에 출시된 AT91RM9200의 경우, 200MIPS 정도의 성능을 가지고 있고, Seagate barracuda와 같은 경우 16MB~32MB 사이의 cache를 가지고 있다. 두 번째

이유로는 data warehouse에 저장되는 데이터의 용량이 커지고 larger data에 대한 repeated scan 연산이 많아지는 등 database 성능 향상에 대한 필요가 대두되었기 때문이다.

1990년대 후반에 시작된, disk에서 processing을 하고자 하는 연구는 대략 4가지로 분류할 수 있는데, Mercury System [2], active disk project at Carnegie Mellon University [3], [4], [5], active disk project in UC Santa Barbara/Maryland [6], [7], IDISK [8]가 그것이다.

3. 지능형 디스크 시스템의 장점

이들의 공통적인 목표는 computation을 가능한 한 data 근처로 옮기는 것이다. 이를 통해 2가지 장점을 얻을 수 있다.

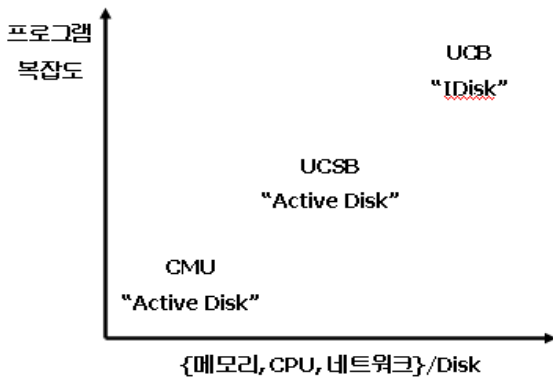
첫 번째로 good scalability를 제공할 수 있다. Processor 하나에 여러 개의 disk를 장착하는 경우, processor나 disk 성능이 향상된다고 해도 이에 비례해서 throughput이 증가하지 않는다. Processor와 disk간의 bus가 bottleneck으로 작용하게 되기 때문이다. 그러나 embedded processor를 통해 disk 내부에서 processing이 이루어지도록 하면, disk와 processor간에 one-to-one coupling이 이루어지므로, disk 개수에 비례해서 throughput이 증가되도록 할 수 있다.

두 번째로 data movement를 감소시켜서 I/O bottleneck을 제거할 수 있다. 데이터 처리를 위해서는, 일반적으로 디스크에서 메모리로 데이터를 읽어 들인 후 작업을 시작하게 된다. 대량의 데이터를 대상으로 소량의 결과를 생성해 내는 작업이라면, 쓸데없는 데이터의 이동이 과도하게 되는 셈이다. Disk 내부에서의 처리를 통해 결과를 생성해내고, 결과만을 host processor에 전송하게 된다면 쓸데없는 데이터의 이동을 최소한으로 줄일 수 있다.

4. 대표적인 지능형 디스크 시스템들의 성능 비교

CMU의 active disk의 경우, nearest neighbor search,

image edge detection 과 같은 scan-based algorithm 에 초점을 두었는데, 이러한 연산들은 CPU 나 메모리를 많이 필요로 하지 않으며, disk 간 통신도 필요로 하지 않는 특징을 가진다. UCSB 의 active disk 도 similar application 을 대상으로 하지만, 좀더 강력한 기능의 CPU 와 메모리를 필요로 하며, 제한적이지만 disk 간 통신을 고려하고 있다는 점이 다르다. IDISK 의 경우, disk 간 통신을 기본 구조로 하고 있기 때문에, 가장 복잡한 구조를 가지지만 higher bandwidth 를 제공할 수 있다.



(그림 1) 지능형 디스크 시스템의 비교

5. 개선점

이제까지의 active disk 시스템들은 주로 단순한 모델들을 사용했다. 즉, host 와 disk 간 통신에 집중하고, disk 쪽 코드를 간단하게 하기 위해, complex operation 은 host computer 에서 수행하고, disk 는 input stream 을 받아서 처리하고 output buffer 에 채우는 식의 단순한 작업만을 수행한다고 가정했다. 따라서 active disk 에서 수행하면 좋은 결과를 얻을 수 있는 작업들은 몇 가지 특징을 가지고 있다. 우선 scan-intensive operation 이어야 하고, selectivity 가 매우 높아야 하며, disk 간 통신이 필요 없어야 한다. 대표적인 연산이 select 로서 simulation 이나 emulation 결과 수십 배까지 성능이 향상되었다. 그러나 sort 나 join 같은 complex operation 의 경우, 성능 향상의 정도가 현저히 낮아지게 된다.

현재의 구조로는 active system 의 실용성을 강조하기가 어렵다. 기본적으로 active system 의 target application 이 data mining 이나 image processing 인데, 이러한 종류의 application 들은 data redistribution 을 필요로 한다. 기존 구조에서처럼 host 를 통해 redistribution 을 하는 방식으로는 성능을 향상시키기가 어려우므로, 효율적인 disk 간 통신 방법이 반드시 추가되어야 한다.

참고문헌

[1] Dina Bitton, and Jim Gray, "The Rebirth of Database Machines", VLDB, 1998.
 [2] Roger D. Chamberlain, Ron K. Cytron, Mark A. Franklin

and Ronald S. Indeck, "The Mercury System: Exploiting Truly Fast Hardware for Data Search", International workshop on storage network architecture and parallel I/Os, 2003.
 [3] Erick Reidel, Garth Gibson, and Christos Faloutsos, "Active Storage For Large-Scale Data Mining and Multimedia", VLDB, 1998.
 [4] Erik Riedel, Christor Faloutsos, and David Nagle, "Active Disk Architecture for Databases", 2000.
 [5] Erik Riedel, Christor Faloutsos, Garth Gibson, and David Nagle, "Active Disks for Large-Sacle Data Processing", 2001.
 [6] Anurag Acharya, Mustafa Uysal, and Joel Saltz, "Active Disks: Programming Model, Algorithms and Evaluation", 1998.
 [7] Mustafa Uysal, Anurag Acharya, and Joel Saltz, "Evaluation of Active Disks for Decision Support Databases," Proc. Sixth Int'l Symp. High-Performance Computer Architecture, pp. 337-348, Jan. 2000.
 [8] Kimberly Keeton, David A. Patterson and Joseph M. Hellerstein, "A case for intelligent disks (IDISKS)", ACM SIGMOD, 1998.