ACM Word Template for SIG Site

Hong Taeha

SungKyunKwan Univ

Software department  
2014313303  
010-8011-4229

xogk369@hanmail.net

**ABSTRACT**

In this paper, we describe the formatting guidelines for ACM SIG Proceedings.

**Keywords**

Keywords are your own designated keywords separated by semicolons (“;”).

# INTRODUCTION

최근 우리 사회에서는 이용 가능한 정보의 양이 넘쳐나게 되면서 정보 자체가 하나의 엄청난 자산이 되고 있다. 그에 따라서 많은 양의 정보를 수집하고 그 정보에서 유의미한 가치를 찾아내는 기술들과 많은 양의 정보를 관리하는 기술들을 필요로 하고 있다.

하지만 많은 양의 데이터를 관리하고 분석하기 위해서는 몇 가지 문제점을 극복해야 한다. 데이터가 많아짐에 따라 그 데이터를 사용하는 사용자들도 많아졌는데 이 사용자들이 동시다발적으로 데이터에 접근 할 때 생기는 문제들을 해결해야 한다. 그리고 보안 관련한 문제도 해결할 수 있어야 대용량의 데이터를 잘 처리할 수 있다.

그렇다면 대용량의 데이터를 관리하기 위해 어떤 방법을 써야 할까? File system을 이용해서 관리하는 방법도 있지만 많은 문제점을 가지고 있다. 이 문제점에 대해서는 후에 알아보도록 한다. 반면에 DBMS는 file system의 문제점들을 극복하고 대용량의 데이터를 관리하는데 효율적이다. 따라서 이 논문에서는 대용량의 데이터를 관리하는데 필요한 점과 한계점, 그 한계점을 극복한 DBMS의 특징들에 대해서 중점적으로 다룰 것이다.

# REQUIREMENTS IN LARGE DATA

대용량의 데이터를 다루기 위해서는 몇 가지 필요한 조건들이 있다.

보통 대용량의 데이터는 디스크나 테이프 같은 저장 장치에 저장한다. 그리고 그 저장 장치에서 메인 메모리로 프로세싱에 필요한 관련 있는 파트를 가져온다. 하지만 32비트 컴퓨터는 데이터를 최대 4기가 바이트씩 조회 할 수 있으므로 데이터 접근 시에 많은 시간이 소요된다. 따라서 필요한 데이터로 한번에 접근해서 참조할 수 있도록 하는 기술이 필요하다.

Data concurrency 문제를 해결하는 것도 중요하다. 다양한 사용자가 동시에 데이터에 접근을 하게 되는데 이 때 data concurrency가 지켜지지 않으면 어떠한 데이터의 수정 중에 발생한 invalid한 데이터를 사용하게 되는 문제가 생긴다. 대용량의 데이터를 다루려면 이러한 문제를 막을 수 있어야 한다.

또, 보안 문제도 중요하다. 어떤 정보는 사용자들에게 제한되고 보호 받아야 한다. 따라서 접근권한관리와 보안 문제를 유연하게 관리해야 한다.

결과적으로, 대용량의 데이터에 쉽게 접근하여 빠르게 사용할 수 있어야 하고, 많은 사용자들의 동시다발적인 접근을 제대로 컨트롤 할 수 있어야 한다. 또한 보안상의 문제와 사용자들의 접근 권한도 잘 관리해야 한다.

## THE LIMITATION OF FILE SYSTEM

우리는 operate system file 에 데이터를 저장해 관리하는 것을 시도할 수 있다. 하지만 이 접근은 많은 문제점을 가지고 있다.

File system을 이용할 경우 디스크나 테이프 같은 저장 장치에 저장된 데이터를 한정된 양씩 읽어온다. 32비트 컴퓨터의 경우 4기가 바이트가 최대이다. 이런 방식으로 데이터를 읽게 되면 데이터를 한번에 읽을 수 없는 경우가 생기고 필요한 데이터로 한번에 접근해서 읽는 방법보다 당연히 시간이 오래 걸린다.

그리고 파일 시스템에서는 유저들이 요청하는 데이터를 처리하는 스페셜한 프로그램을 작성해야 한다. 하지만 찾아야 하는 데이터의 양이 매우 크기 때문에 이 프로그램을 구현하는 것이 매우 복잡하다. 또 data concurrency 문제를 해결하는 것도 어려워진다.

보안과 관련한 부분에서도 문제점이 있다. Operating System 은 패스워드 메커니즘만을 제공하기 때문에 이것은 데이터의 안전성과 사용자들의 접근권한을 관리하는 부분에서 충분히 유연하지 않다.

또한 File system은 시스템 충돌 같은 문제가 일어났을 때 손상된 자료를 복구하는 부분에 대해서도 취약하다.

# OVERCOMING LIMITATIONS

2.1에서 봤듯이 file system은 대용량의 데이터를 다루기에 많은 취약점이 있다. 이 취약점을 극복하고 대용량의 데이터를 다루는 것의 요구사항을 충족시키기 위해 필요한 특징들에 대해서 설명하겠다.

## DBMS concept

DBMS를 사용하여 데이터를 다루는 것에는 많은 이점이 있다.

Data Independence는 어플리케이션이 데이터 표현과 저장에 대한 세부적인 정보를 노출되지 않도록 한다. 그래서 DBMS는 데이터의 세세한 부분을 숨기고 추상적인 모습을 제공한다. 이는 특정한 데이터가 수정되어도 다른 데이터나 다른 schema에 영향을 주지 않는 것을 말한다.

또 데이터를 효율적으로 되찾아오고 저장하기 위해 세련된 기술들을 다양하게 이용한다. 따라서 file system에서 데이터의 일부만을 scan하여 읽어오던 것과 달리 데이터에 접근하여 바로 참조할 수 있도록 하여 데이터 처리 시간을 줄여준다.

DBMS는 여러 사용자가 동시에 어떤 정보에 접근하기 전에 그 정보가 유효한지 미리 확인을 할 수 있다. 미리 확인을 함으로써 data integrity and security를 지킨다.

그리고 DBMS는 transaction을 통해서 여러 사용자가 동시다발적으로 정보에 접근할 때 발생하는 문제를 해결하고 recovery에 대한 부분도 처리한다. Transaction에 대해서는 뒤에서 더 자세히 다루도록 하겠다.

Table 1 은 file system과 DBMS의 차이점에 대해서 보여주고 있다.

Table 1. difference of File System and DBMS.

|  |  |  |
| --- | --- | --- |
|  | **File System** | **DBMS** |
| Data access | It is slow because it has to scan several times. | It is fast because you need to access and reference the data you need. |
| Data concurrency | Implementation is very complex. | Maintain data independence |
| Security | Operating systems provide only a password mechanism for security. So it is not suitable. | It protects the security by checking in advance before the user accesses the data. |
| Recovery | weak. | Protect users from the effects of system failure. |

## Data Independence and Query language

[[1]](#footnote-1)

## Transaction, Concurrency Control, Recovery

# REFERENCES

1. Ramakrishnan, Gehrke, 2003, Database Management System

1. [↑](#footnote-ref-1)