

리눅스 환경에서 파일 시스템들의 블록 할당 성능 분석

최진오*

*부산외국어대학교 임베디드소프트웨어학과

Performance Analysis of Block Allocation of File Systems on Linux Environment

Jin-oh Choi*

*Department of Embedded Software, Busan University of Foreign Studies

E-mail : jochoi@bufs.ac.kr

요 약

임베디드 시스템에 많이 사용되는 리눅스 환경에는 Ext2, FAT, NTFS 등 다양한 파일 시스템이 사용된다. 임베디드 시스템에 탑재된 파일 시스템은 미니 하드 디스크 또는 플래시 메모리를 미디어로 채택하고 있다. 이러한 장치에 구현되는 파일 시스템의 종류는 응용 프로그램의 성능에 많은 영향을 미친다. 동일한 미디어에서 파일시스템의 성능 요인은 블록 할당과 블록 해제 오버헤드이다. 이 중에서 블록 해제 성능은 파일시스템에 따라 큰 차이를 보이지 않는다. 이 논문에서는 임베디드 시스템의 Ext2, FAT, 그리고 NTFS 파일 시스템에서 블록 할당 성능을 비교 분석한다. 그래서 어떤 경우에 어느 파일 시스템이 우수한지에 대하여 고찰한다.

ABSTRACT

Linux environment that is commonly used at embedded systems, supports various file systems as Ext2, FAT, NTFS, etc. The file system that is equipped on the embedded system is mostly implemented on mini hard disk or flash memory. The types of the file system of the system make an effect on the performance of a application programs. The factors of file system performance on a same media are block allocation and block free time. On these factors, block free time is not so different according to the type of file systems. This thesis performs the performance benchmark of a Ext2, FAT and NTFS file systems about block allocation performance. As the result, it is discussed that what file system is better at which case.

키워드

Performance benchmark, File System, Block Allocation, Embedded File System

1. 서 론

현재 리눅스는 모바일 디바이스를 포함하여 임베디드 시스템(Embedded System)의 운영체제로 가장 많이 사용되고 있다[1]. 리눅스 운영체제 환경은 가상 파일 시스템을 이용하여 다양한 파일 시스템들을 지원한다[2]. 리눅스 환경의 임베디드 시스템에서 파일 시스템을 사용하는 응용 프로그램들은 CPU와 주 메모리에 비해 느린 속도의 파일 시스템 성능 때문에 전체 성능이 파일 시스템에 의해 영향을 받는다.

리눅스환경의 임베디드 시스템에서 파일 시스템 구축 미디어로는 광학 CD-ROM, 미니 하드 디스크, 플래시 메모리, SSD(Solid State Drive)

등이 활용된다. 특히 최근 SSD와 같이 고속의 플래시 메모리 기반 파일시스템 구축이 일반화되면서 보조 기억장치에 의존하는 임베디드 시스템이 늘어나고 있다.

리눅스에서 지원하는 파일 시스템은 Ext2, Ext3, Ext4, FAT-16, FAT-32, NTFS 등이다. 각 파일 시스템은 구조와 메커니즘 특성에 따라 장단점이 존재한다. 따라서 리눅스 환경에서 파일 시스템의 종류에 따라 응용 프로그램이 디스크에 데이터를 기록하는 성능, 검색하는 성능, 그리고 삭제하는 성능이 각각 장단점을 가지며 상이할 것이다.

파일 시스템들의 삭제(블록 해제) 성능은 크게 다르지 않다. 대부분 파일시스템의 메타 데이터

(meta data) 영역에 삭제 표시만 한다[3]. 그리고 파일 시스템에서 기록(블록 할당) 성능과 검색 성능에서 기록 성능이 월등히 낮다. SSD의 경우 4 배 가까이 차이가 난다[4]. 따라서 블록 할당 성능이 전체 파일 시스템의 성능에 가장 큰 영향을 미친다고 볼 수 있다.

이 논문에서는 임베디드 시스템을 위한 리눅스 환경에서 파일 시스템의 종류에 따라 블록 할당 성능이 어떠한 차이와 특징을 보이는지 실험하고 그 결과를 분석한다.

II. 기존 파일 시스템

2.2 Ext 파일 시스템

Ext(Extended) 파일 시스템은 파일 시스템을 여러 블록 그룹으로 쪼개어 별도 관리하여 다소 복잡한 구조를 가진다. 각 블록 그룹은 i-node로 파일 정보를 저장하는 방법을 채택하며 Bitmap들을 도입하여 빠른 접근을 지원한다. 특히 블록 할당 시 블록 Bitmap을 이용하여 빈 블록을 빨리 탐색할 수 있다는 장점이 있다. 그림 1은 Ext 파일 시스템의 디스크 구조이다[3].

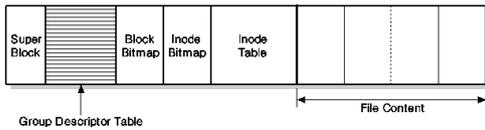


그림 1. Ext2 파일 시스템

2.2 FAT 파일 시스템

FAT(File Allocation Table) 파일 시스템은 단순한 구조를 가지고 있으며 FAT 영역에 별도로 파일의 데이터 블록 리스트를 저장하고 있다. 따라서 FAT 내용만으로 모든 파일의 배치 구조를 알 수 있는 장점이 있다. 특히 블록 할당 시 FAT만 탐색함으로써 빈 블록을 빨리 찾을 수 있다는 장점이 있다. 그림 2는 FAT 파일 시스템의 개념도이다[3].

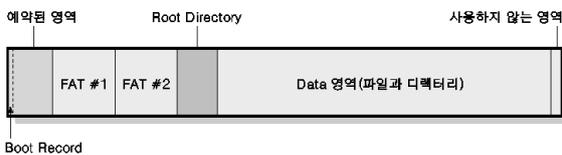


그림 2. FAT 파일 시스템

2.3 NTFS 파일 시스템

NTFS(New Technology File System)는 FAT의 한계를 극복하고 서버용으로 개발되어 FAT보다 복잡한 구조를 가지고 있으며 MFT(Master File Table)로 모든 개체를 저장하며 내부에 다양한 속성(attribute)들을 가지고 있다. 블록 할당에는 내장 MFT 파일 정보를 이용한다. 그림 3은 NTFS

의 디스크 구조이다[3].

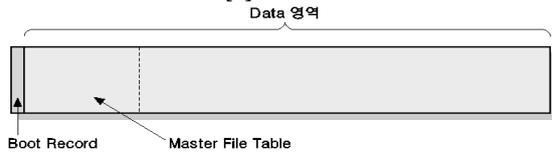


그림 3. NTFS 파일 시스템

III. 파일 시스템의 블록 할당 성능 비교

3.1 파일 시스템별 블록 할당 메커니즘 비교

Ext 파일 시스템은 새로운 파일을 생성하기 위해 먼저 빈 i-node를 i-node Bitmap에서 찾아 할당 받고 빈 블록들을 블록 Bitmap에서 찾아 할당 받는 과정을 거친다. 할당받은 블록들은 i-node에 목록을 저장한다. 마지막으로 파일 이름과 i-node 번호를 저장할 디렉터리 엔트리(directory entry)를 할당받아 저장한다.

FAT 파일 시스템은 빈 디렉터리 엔트리(directory entry)를 할당 받고 FAT 정보를 참조하여 빈 블록들을 할당 받는다. 할당 받은 블록들은 FAT에 체인(chain)으로 연결 정보를 기록한다.

NTFS는 빈 MFT 엔트리를 하나 할당 받고 빈 블록들은 \$Bitmap 파일을 참조하여 할당 받는다. 할당받은 블록들은 \$DATA 속성에 저장한다.

3.2 블록 할당 시나리오

파일 시스템의 블록 할당 성능은 자체 접근 메커니즘에 따라 차이가 나겠지만 특정 시점의 파일 시스템 상태에 따라 다른 특성을 보일 수도 있다. 따라서 실험을 가능한 몇 가지 상태로 나누어 실시한다.

첫째, 파일 시스템이 비어 있는 경우이다. 순수하게 블록을 순차 할당받아 지속적으로 파일을 기록하는 시나리오이다.

둘째, 파일 시스템이 중간 중간에 빈 공간을 가지는 시나리오이다. 빈 공간의 크기도 랜덤하게 설정한다. 즉 약 50%의 빈 공간이 있는 경우이다.

3.3 할당 블록 크기

할당받을 블록의 크기는 일정한 크기의 경우와 랜덤한 경우로 나누어 실험한다. Ext 파일 시스템은 1Kb, 2Kb 또는 4Kb 블록 크기를 지원하며 FAT 파일 시스템은 16Kb, 32Kb, 64Kb의 cluster 크기를 지원한다.

할당 받는 블록의 크기에 따라 비례적으로 성능이 변화될 것으로 예측된다. 작은 블록의 할당에는 여러 번의 작업이 필요하기 때문이다.

IV. 결론

이 논문에서는 임베디드 시스템을 위한 리눅스

환경에서 파일 시스템의 블록 할당 성능에 대하여 고찰하였다. 관찰 대상은 Ext, FAT, 그리고 NTFS 파일 시스템으로서 다양한 시나리오에서 블록 할당에 대한 성능을 비교 분석하였다. 그리고 이를 비교 분석하면 리눅스 기반 임베디드 시스템에서 주된 작업에 적합한 파일 시스템을 채택할 수 있다.

향후 실험에 의한 정확한 결과 도출로 추론과 비교 분석이 필요하며 보다 다양한 성능 비교 인자를 개발하여 비교 분석하는 실험과 연구가 필요하다. 또한 표준 벤치마크 프로그램에 의한 성능 비교도 필요하다.

참고문헌

- [1] http://www.itdaily.kr/atl/view.asp?a_id=44297
- [2] 백승재, 최종무, “리눅스 커널 내부구조“, 교학사, 2008
- [3] 정준석, 정원용, “임베디드 개발자를 위한 파일시스템의 원리와 실습“, 한빛미디어, 2007
- [4] http://www.ebuzz.co.kr/news/hotissue/2283716_4997.html
- [5] 박홍진, “리눅스에서 커널 기반 벤치마킹 도구글 이용한 파일 시스템 성능 평가 연구,” 한국정보기술학회지 제 10권 제12호, 2012.12, pp119-125
- [6] 류성태, 김현준, 한환수, “비휘발성 메모리 파일시스템을 위한 신뢰성 보장 기법의 파일 쓰기 성능 분석,” 정보과학회논문지 제19권 제11호, 2013
- [7] 최민석, “리눅스 환경에서 SSD를 위한 파일 시스템 성능 분석”, 한양대학교 전자통신컴퓨터공학 석사학위논문, 2009