

<http://dx.doi.org/10.7236/IIBC.2015.15.6.181>

IIBC 2015-6-25

초경량 사물인터넷을 위한 비휘발성램 스토리지 성능평가 및 분석

Performance Evaluation and Analysis of NVM Storage for Ultra-Light Internet of Things

이은지*, 유승훈**, 반효경***

Eunji Lee*, Seunghoon Yoo**, Hyokyung Bahn***

요 약 최근 통신 및 반도체 기술의 급격한 발전과 함께 소규모 기기에도 컴퓨팅 기능을 탑재하는 사물인터넷 시장이 부상하고 있다. 사물인터넷을 위한 저장장치는 전력소모와 물리적 크기에 제한이 있어 기존 HDD나 SSD 대신 NVRAM 기반의 스토리지가 사용될 것으로 전망되고 있다. 그러나 현재 사물인터넷 플랫폼 기술은 기존의 전통적인 스토리지를 타겟으로 설계되어 NVRAM 스토리지에서는 다양한 비효율성을 초래할 수 있다. 본 논문은 현재의 다양한 운영체제의 I/O 기법들의 효율성과 성능을 NVRAM 스토리지 환경에서 평가하고 분석하여 향후 사물인터넷을 위한 스토리지 기술에 대해 방향성을 제시한다.

Abstract With the rapid growth of semiconductor technologies, small-sized devices with powerful computing abilities are becoming a reality. As this environment has a limit on power supply, NVM storage that has a high density and low power consumption is preferred to HDD or SSD. However, legacy software layers optimized for HDDs should be revisited. Specifically, as storage performance approaches DRAM performance, existing I/O mechanisms and software configurations should be reassessed. This paper explores the challenges and implications of using NVM storage with a broad range of experiments. We measure the performance of a system with NVM storage emulated by DRAM with proper timing parameters and compare it with that of HDD storage environments under various configurations. Our experimental results show that even with storage as fast as DRAM, the performance gain is not large for read operations as current I/O mechanisms do a good job hiding the slow performance of HDD. To assess the potential benefit of fast storage media, we change various I/O configurations and perform experiments to quantify the effects of existing I/O mechanisms such as buffer caching, read-ahead, synchronous I/O, direct I/O, block I/O, and byte-addressable I/O on systems with NVM storage.

Key Words : Operating Systems, Embedded Systems, Internet of Things, Non-volatile Memory

I. 서 론

최근 사물인터넷에서는 저전력성과 안정성이 메모리

및 스토리지의 주요 요구사항으로 부상하며 기존의 플래시 메모리와 DRAM과 같은 전자 충전 방식의 저장매체 대신 PRAM, STT-MRAM과 같은 저항 기반의 저장매

*정회원, 충북대학교 소프트웨어학과

**준회원, 서울대학교 컴퓨터공학부

***정회원, 이화여자대학교 컴퓨터공학과(교신저자)

접수일자: 2015년 10월 21일, 수정완료: 2015년 11월 21일

게재확정일자: 2015년 12월 11일

Received: 21 October, 2015 / Revised: 21 November, 2015 /

Accepted: 11 December, 2015

***Corresponding Author: bahn@ewha.ac.kr

Dept. of Computer Engineering, Ewha University, Korea

체 기술이 급속도로 발전하고 있다. 차세대 비휘발성 메모리(Non-volatile RAM, NVRAM)로 통칭되는 이러한 저장매체들은 비휘발성의 특징을 가지면서도 기존 메모리에 근접한 빠른 성능을 지니기 때문에 Wearable 장치, IoT (Internet of Things) 등 다양한 시스템에서 널리 사용될 것으로 예측되고 있다^[10-12]. 현재 PRAM, MRAM, FeRAM과 같은 다양한 차세대 비휘발성 메모리가 연구되고 있으며 일부는 시제품이 출시되기도 하였다^[1]. 그 중에서도 PRAM (Phase Change Memory)은 가장 활발하게 연구되고 있는 매체로서 이미 상용화 단계에 들어섰다. PRAM의 가장 큰 장점은 집적도가 매우 높아 대용량화가 용이하다는 것이다. PRAM의 집적도는 이미 22nm급 수준에 이르렀으며 ITRS에서는 PRAM의 집적도가 5~9nm까지 향상될 수 있을 것이라고 예상했다^[2]. 현재 DRAM의 집적도 한계를 40nm급으로 예측하는 것을 감안할 때 PRAM이 가진 잠재력을 충분히 예상할 수 있다. 현재 오래전부터 PRAM 개발에 주도적인 역할을 하고 있는 삼성전자에 의해서 512Mbit의 PRAM 제품이 상용화되었으며 또 다른 선두업체인 뉴모닉스에서는 1Gbit PRAM의 시제품을 개발하였다^[3].

본 논문은 비휘발성 메모리의 등장과 함께 고속화되고 있는 스토리지 시스템을 효율적으로 사용하기 위하여, 운영체제 입출력 기법의 전반적인 요소를 분석하고 설계하는 연구를 수행하였다. 이를 위하여 입출력 시에 거치는 하드웨어 및 소프트웨어 스택 구조를 분석하고 각 계층에서 사용되는 요소 기술에 대하여 스토리지 속도 향상에 따른 효율성을 분석하였다. 그림 1은 입출력 요청부터 실제 디바이스에 수행되기까지의 전 소프트웨어 계층 구조 및 관련 기술을 나타낸다. 그림에서 보는 바와 같이 입출력 요청은 System call이나 Memory mapped I/O 방식을 통해 이루어지며 이것은 파일시스템 인터페이스를 통해 버퍼캐쉬 및 블록 계층에 전달되어 최종적으로 물리 디바이스에서 수행된다. 각 계층에서는 입출력 성능을 향상시키기 위해 캐싱, 미리읽기(read-ahead), 스케줄링, 블록 단위의 입출력 수행 등 다양한 기법이 적용되고 있다.

본 논문에서는 하드디스크 환경에서 전형적으로 사용되는 비동기적 버퍼 I/O (Asynchronous Buffered I/O) 기법을 차세대 메모리 기반 스토리지 환경에 적용했을 때의 성능을 측정하기 위하여 램 기반의 스토리지를 구성하고 다양한 벤치마크를 통해 해당 I/O 방식의 성능을

측정하였다. 또한 버퍼캐쉬의 효율성 측정을 위해서는 버퍼캐시를 사용하는 Buffered I/O와 캐시를 사용하지 않는 Direct I/O 방식의 성능을 비교하여 메인메모리의 스토리지 캐시가 스토리지 속도변화에 따라 어떠한 성능 기여를 나타내는지 측정하였다. 미리읽기(Read-ahead) 기법의 성능 측정을 위해서는 미리읽기의 기능을 hdparm 명령어를 통해 설정/비설정 및 윈도우 사이즈를 변경하면서 각 경우의 성능을 측정하였다. I/O 단위 변화에 따른 성능평가를 위해서는 바이트 단위의 접근을 허용하는 인터페이스에 비휘발성램을 저장장치로 연결하고 이를 기존의 블록 저장장치와 비교하여 성능평가를 수행하였다.

본 논문의 구성은 다음과 같다. II장에서는 NVRAM과 HDD의 성능평가를 위한 실험환경에 대해 기술한다. III장에서는 다양한 시나리오에 따라 측정된 저장장치의 성능을 평가하고 분석한 결과에 대해 기술한다. IV장에서는 본 논문의 관련 연구에 대해 기술한다. 끝으로 V장에서는 본 논문의 결론을 제시한다.

II. 실험환경

연구 실험 플랫폼 환경은 아래와 같다. Intel Core i5-3570 CPU 3.4GHz를 사용하였으며 32GB의 DDR3-1600 memory와 1TB of SATA 3Gb/s HDD를 스토리지로 사용하였다. 운영체제로는 Linux kernel 3.10.20을 사용하였으며 파일시스템으로는 디폴트로 사용되는 ext4를 사용하였다. 비휘발성 메모리 기반 스토리지로는 PCM이나 STT-RAM 등이 아직 초기 시판 단계에 있기 때문에 비슷한 성능을 가지는 DRAM을 디스크 형태로 설정하여 실험을 수행하였다. DRAM과 비휘발성램의 성능차이를 반영하기 위하여 우리는 NVM 용 디바이스 드라이버를 개발하여 DRAM 대비 NVM의 성능을 읽기/쓰기의 경우 각각 4.4배와 10배 느린 것으로 설정하여 실험하였다. 성능평가를 위해서는 대표적인 벤치마크인 Iozone과 filebench를 사용하였다.

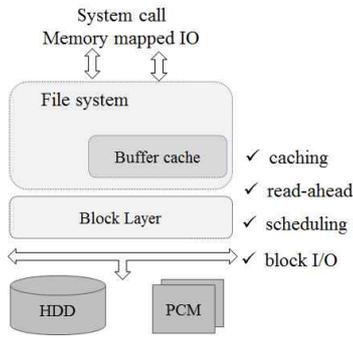


그림 1. 입출력을 위한 운영체제 소프트웨어 스택 구조 및 계층별 수행 기법
 Fig. 1. Software stack architecture in Linux I/O systems

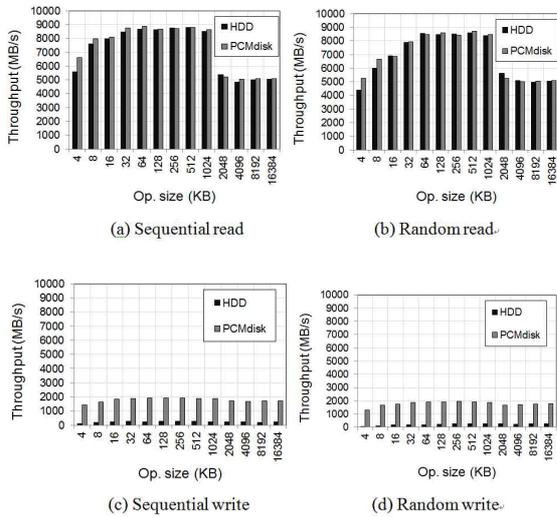


그림 2. 스토리지별 비동기식 버퍼 I/O 기법의 성능 비교
 Fig. 2. Comparison of buffered I/O for different storage devices

II. NVRAM 스토리지 성능평가 및 분석

그림 2은 전형적인 I/O 수행방식인 비동기식 버퍼 I/O 기법의 성능을 스토리지별로 보여주고 있다. 읽기의 경우 캐쉬의 효과로 인해 스토리지의 성능이 거의 비슷하게 나타났으며 쓰기의 경우에는 캐쉬를 스토리지에 주기적으로 반영하기 때문에 스토리지 성능 차이가 실제 I/O 성능에 반영되어 나타났다.

그림 3은 메인메모리의 크기를 변화시키며 측정하는 스토리지별 성능을 보여준다. 하드디스크와 같이 느린 스토리지에서는 캐쉬 사이즈의 감소에 따라 성능이 급격히 감소되는 반면 고속의 램 기반 스토리지에서는 캐쉬 크

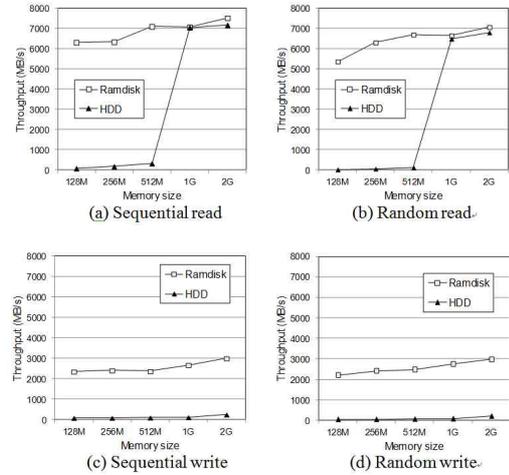


그림 3. 메인메모리 사이즈의 변화에 따른 성능 평가
 Fig. 3. I/O Performance for varying memory sizes

기가 성능에 크게 영향을 끼치지 않는 것으로 나타났다. 그러나 스토리지의 물리적 속도가 메인메모리와 비슷하더라도 I/O 스택 오버헤드로 인해 캐쉬 사이즈 감소에 따라 약간의 성능저하 현상은 나타났다.

그림 4는 스토리지별로 I/O 방식 변화에 따른 성능변화를 보여주고 있다. Baseline는 비동기식 버퍼 I/O 방식이며 Sync는 매 입출력마다 스토리지에 반영하는 방식이다. direct는 버퍼캐쉬를 거치지 않고 직접 스토리지와 사용자 프로그램이 I/O를 수행하는 방식이다. 하드디스크의 경우 캐쉬를 사용하지 않는 경우의 성능이 매우 낮은 것으로 나타난 반면 비휘발성램 기반 스토리지의 경우 캐쉬를 사용하고 스토리지에 동기화를 시키는 Sync 방식이 Direct 방식보다 성능이 저하되는 것으로 나타났다. 현재 DRAM과 HDD로 구성된 메모리 계층구조에서는 데이터를 캐쉬하고 주기적으로 스토리지에 반영하여 데이터의 손실을 막는데, 비휘발성램 스토리지에서는 이러한 방식 대신 Direct I/O 방식을 활용하는 것이 성능에 도움이 될 것으로 예상된다.

그림 5는 미리읽기 기법이 스토리지의 속도 변화에 따라 성능에 끼치는 효율성을 평가한 그래프이다. 미리읽기가 설정된 경우 특정 페이지에 접근이 발생하면 인접한 32개의 페이지를 함께 메인메모리로 적재한다. 하드디스크의 경우 평균 21%의 성능향상이 있었던 반면 비휘발성램 스토리지에서는 미리읽기 기법이 성능을 향상시키지 못하거나 오히려 성능을 저하시켰다. 이것은 기존의 미리읽기 방식이 초기접근 비용이 비싼 하드디스크

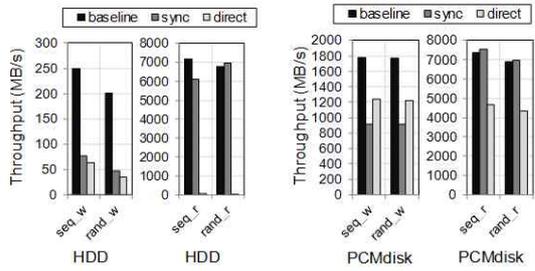


그림 4. 입출력 기법 변화에 따른 스토리지별 성능비교
Fig. 4. Comparison of storages for different I/O policies

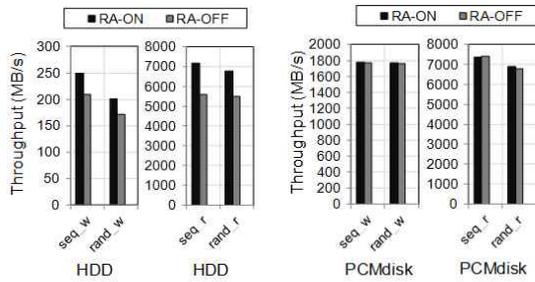


그림 5. 미리읽기 기법의 효율성 분석
Fig. 5 Storage performance with and without read-ahead

에 최적화되어 적극적으로 데이터를 메인메모리에 적재시키는데 메인메모리에서 적중이 되지 않는 경우 발생하는 패널티가 비휘발성램 스토리지 환경에서는 상대적으로 크기 때문이다. 이에 우리는 히스토리 버퍼를 사용해 2차 접근이 발생한 경우 캐시에 적재하고 이 때 인접한 데이터 블록들이 히스토리 버퍼에 존재하는 경우에만 미리읽기를 수행하는 방식을 향후 설계해보고자 한다.

그림 6은 동일한 양의 I/O를 수행하는데 각각을 I/O 접근횟수를 증가시키며 스토리지의 성능 변화를 측정하였다. x축에서 오른쪽으로 갈수록 I/O 사이즈는 작아지고 접근 빈도가 증가한다. 하드디스크의 경우 접근 횟수가 늘어남에 따라 성능이 급격히 저하되는 양상을 나타낸 반면 비휘발성램 기반 스토리지는 빈도 증가에 따른 성능저하가 완만하게 나타났다. 한편 그림 7은 I/O를 동일 횟수로 수행하는데 크기를 증가시키며 이에 따른 성능평가를 측정한 그래프이다. 하드디스크의 경우 1회 I/O에서 수행하는 실제 I/O 양은 I/O 수행시간에 거의 영향을 끼치지 않았으나 비휘발성램 스토리지의 경우 I/O 크기가 증가할 때 거의 비례하게 I/O 수행시간이 증가함을 알 수 있었다.

이러한 결과는 기계적으로 다른 동작 방식을 갖는 하

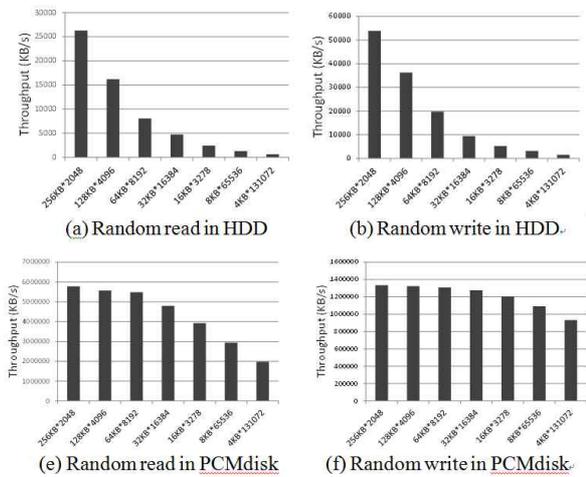


그림 6. I/O 횟수 변화에 따른 성능 변화
Fig. 6. Storage performances for varying the number of accesses

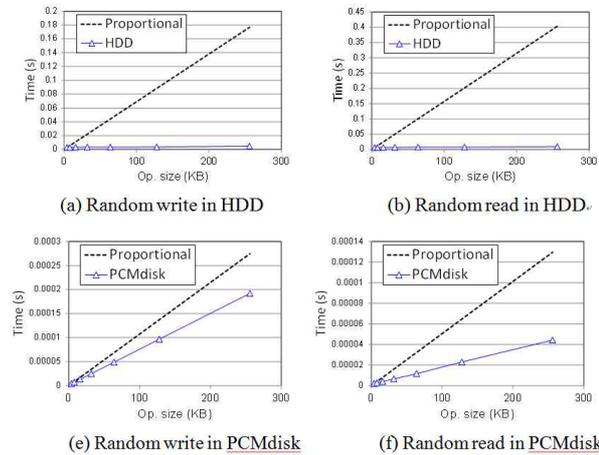


그림 7. 단일 I/O 크기 변화에 따른 성능비교
Fig. 7. Storage performance for varying the I/O unit size

드디스크와 비휘발성 메모리에서 병목현상을 발생시키는 원인을 규명하여, 새로운 스토리지 시스템에 맞는 I/O 방식 설계에 활용될 수 있다.

그림 8은 HDD와 NVRAM의 접근 시간을 단계별로 분할하여 보여주고 있다. 그림에서 보여주는 바와 같이, 하드디스크의 경우 요청된 데이터가 존재하는 위치까지 헤드가 움직여야 하는 탐색시간을 포함하기 때문에, 초기 스토리지 접근 비용이 총 IO 시간에서 많은 부분을 차지한다. 따라서 하드디스크 환경에서는 스토리지의 접근 횟수를 최소화하는 것을 목표로 IO 매커니즘이 발전해왔다. 그러나 비휘발성램 환경에서는 스토리지의 초기

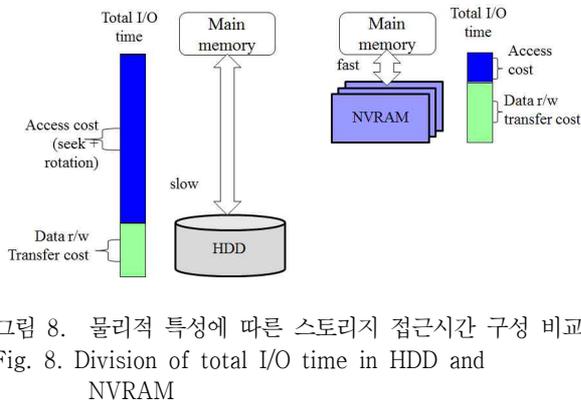


그림 8. 물리적 특성에 따른 스토리지 접근시간 구성 비교
 Fig. 8. Division of total I/O time in HDD and NVRAM

접근 시간보다 데이터를 실제 읽고 전송하거나 프로그램 하는 시간이 IO에서 큰 비중을 차지한다. 따라서 하드디스크를 가정하여 I/O 접근을 최소화하는 방향으로 진화해 온 현재의 운영체제 I/O 기법이 비휘발성램 기반 스토리지에서는 I/O 양을 절감하는 방향으로 발전해 나가야 함을 보여준다고 하겠다.

IV. 관련 연구

최근 컴퓨팅 환경에서 전력소모와 신뢰성이 중시되면서 저전력과 비휘발성 특성을 모두 지닌 NVRAM을 스토리지로 활용하고자 하는 연구가 활발히 이루어지고 있다. [4, 5, 6, 7, 8, 9]

초기에는 소량의 NVRAM을 스토리지의 일부로 활용하는 연구가 이루어졌다. Pramfs는 빈번하게 접근되거나 전력끊김에도 손상되지 않아야 하는 데이터를 저장하는 용도로 NVRAM을 유지함으로써 크래쉬 발생 시 부팅속도와 신뢰성을 높이기 위해 개발된 파일시스템이다^[6]. MRAMFS^[7]와 NEB file system^[8] 역시 메타데이터와 같은 파일시스템의 일부 데이터를 NVRAM에 유지하기 위한 기법에 대해 연구하였다. 용량이 제한적이고 최대 쓰기 횟수에 제한이 있는 NVRAM의 특성을 고려하여 압축기법과 extent 기반으로 데이터를 관리하는 방식에 대해 제안하였다.

NVRAM의 집적도가 급속도로 향상됨에 따라 최근에는 파일시스템 전체를 NVRAM에 유지하는 기법에 대해 연구가 진행되고 있다. Baek et al.은 NVRAM은 메모리와 스토리지 특성을 모두 가지고 있기 때문에 파일 객체와 메모리 객체를 통합된 계층에서 효율적으로 지원하는

기법에 대해 연구를 수행하였다^[9]. Condit et al.은 copy-on-write 파일시스템을 바이트 단위의 접근이 가능한 NVRAM에 최적화 시켜 BPFs라고 하는 파일시스템을 개발하였다^[4]. BPFs는 원자적 업데이트가 가능한 단위까지 트랜잭션의 크기가 감소하면 제자리 쓰기를 통해 재귀적으로 쓰기가 발생하는 오버헤드를 감소시켰다. Wu et al.은 NVRAM 스토리지가 CPU에 의해 직접 접근되는 메모리 버스에 놓여있는 구조에서 데이터 접근을 효율적으로 지원하는 파일시스템을 제안하였다^[5].

V. 결론

본 논문에서는 하드웨어 제약이 심한 초경량 시스템을 위한 NVRAM 스토리지 활용방안을 운영체제 관점에서 살펴보았다. 구체적으로는 기존 운영체제에서 사용되고 있는 I/O 기법의 효율성을 고속의 바이트 단위의 접근이 가능한 NVRAM 기반 스토리지에서 평가 및 분석하였다. HDD를 타겟으로 발전해온 미리읽기, 버퍼 I/O의 기법을 NVRAM에 적용했을 때의 성능변화를 관찰하고, HDD와는 물리적 특성이 다른 NVRAM에서 성능상의 병목을 발생시키는 요소 등에 대해 분석하였다. 본 논문에서 수행된 연구는 향후 널리 활용될 가능성이 높은 NVRAM을 효율적으로 활용하기 위한 시스템 소프트웨어 설계에 큰 도움이 될 것이다.

본 논문에서 분석한 결과를 바탕으로 향후 NVRAM을 위한 I/O 기법에 대한 연구를 수행하고자 한다. 소프트웨어 오버헤드를 최소화하고 I/O 트래픽을 절감할 수 있는 기법을 설계하여 NVRAM 스토리지를 효율적으로 관리하는 시스템 소프트웨어를 개발하고자 한다.

References

- [1] https://en.wikipedia.org/wiki/3D_XPoint
- [2] E. Ipek, J. Condit, E. B. Nightingale, D. Burger and T. Moscibroda, "Dynamically Replicated Memory: Building Reliable Systems from Nanoscale Resistive Memories," Proceedings of the ACM ASPLOS'10, March, 2010.
- [3] G. Atwood, "Phase Change Memory Landscape",

Flash Memory Summit, 2011.

- [4] J. Condit, E.B. Nightingale, C. Frost, E. Ipek, B. Lee, D. Burger, and D. Coetzee, "Better I/O through byte-addressable, persistent memory," ACM Symp. Operating Systems Principles (SOSP), 2009.
- [5] X. Wu and A. L. N. Reddy, "SCMFS: A File System for Storage Class Memory," Proc. Int'l Conf. Supercomputing (SC), 2011.
- [6] PRAMFS, available at <http://pramfs.sourceforge.net>
- [7] N. K. Edel, D. Tuteja, E. L. Miller, and S. A. Brandt, "MRAMFS: a compressing file system for non-volatile RAM," Proc. 12th IEEE Int'l Symp. Modeling, Analysis, and Simulation of Computer and Telecommun. Systems (MASCOTS), 2004.
- [8] S. Baek, C. Hyun, J. Choi, D. Lee, and S. H. Noh, "Design and analysis of a space conscious nonvolatile-RAM file system," Proc. IEEE Region 10 Conf. (TENCON), 2006.
- [9] S. Baek, K. Sun, J. Choi, E. Kim, D. Lee, and S. H. Noh, "Taking advantage of storage class memory technology through system software support," Proc. Workshop on Interaction between Operating Systems and Computer Architecture (WIOSCA), 2009.
- [10] Emerging Non-Volatile Memory (NVM) Technologies & Markets 2015, Yole Development
- [11] J. Kim, N. Kang, "Secure Configuration Scheme of Pre-shared Key for Lightweight Devices in Internet of Things," The Journal of The Institute of Internet, Broadcasting and Communication (JIIBC), Vol. 15, No. 3, pp. 1-6, 2015
- [12] J. Park, N. Kang, "Design of Smart Service based on Reverse-proxy for the Internet of Things," The Journal of The Institute of Internet, Broadcasting and Communication(JIIBC), Vol. 14, No. 6, pp. 1-6, 2014

저자 소개

이 은 지(정회원)



- 2005년 2월 : 이화여자대학교 컴퓨터 공학과 학사
- 2012년 2월 : 서울대학교 컴퓨터공학부 박사
- 2014년 3월 ~ : 충북대학교 소프트웨어학과 조교수

유 승 훈(준회원)



- 2008년 2월 : 공군사관학교 컴퓨터공학과 학사
- 2010년 2월 : 서울대학교 컴퓨터공학부 석사
- 2015년 3월 ~ : 서울대학교 컴퓨터공학부 박사과정

반 효 경(정회원)



- 1997년 2월 : 서울대학교 계산통계학과 학사
- 1999년 2월 : 서울대학교 전산학과 석사
- 2002년 2월 : 서울대학교 컴퓨터공학부 박사.
- 2002년 9월 ~ : 이화여자대학교 컴퓨터공학과 교수.

<주관심분야: 운영체제, 스토리지 시스템, 임베디드 시스템>

※ 이 논문은 2014년도 충북대학교 학술연구지원사업의 연구비 지원에 의하여 연구되었음
(This work was supported by the research grant of Chungbuk National University in 2014)