2011년 공개소프트웨어 커뮤니티 지원사업

PVFS 활용을 통한 클라우드 기반 대용량 데이터 처리 플랫폼 개발

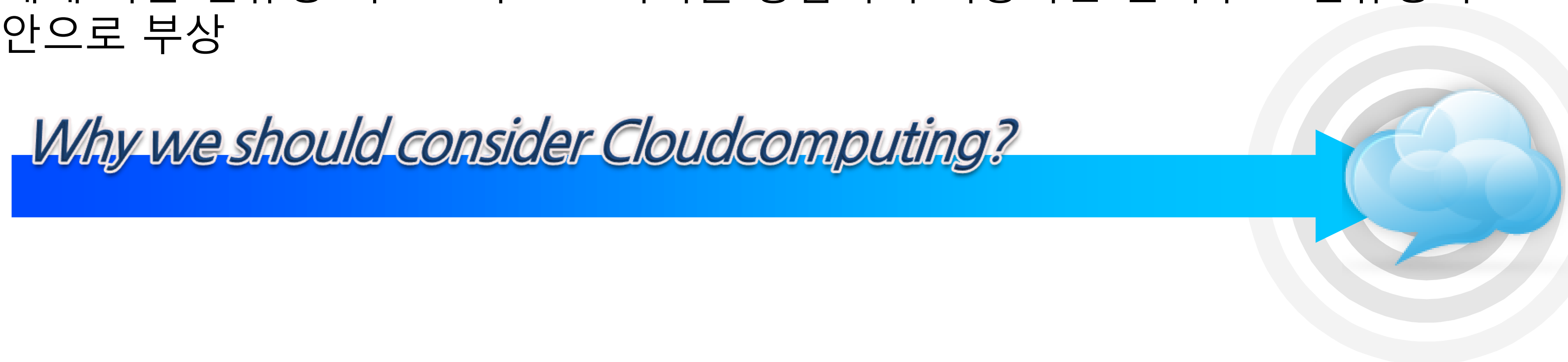
서울시 강남구 삼성동 144-25 애플트리타워 11층 TEL 02.516.5990 FAX 02.516.5997
COPYRIGHT © 2011 INNOGRID. ALL RIGHTS RESERVED.

사업 개요

과제명	PVFS 활용을 통한 클라우드 기반 대용량 데이터 처리 플랫폼 개발
주관기관	㈜ 이노그리드
총괄책임자	손호준 연구소장
참여기관/책임자	(사)한국클라우드서비스협회 / 민영기 사무국장
과제 수행기간	2011. 4. 20 ~ 2011. 12. 9 (8개월)

클라우드 기반 대용량 데이터 처리의 필요성

대용량 컴퓨팅에 대한 수요 증가, 데이터 크기의 증가에 따른 기존 컴퓨팅 패러다임의 한계에 따른 컴퓨팅 리소스와 스토리지를 통합하여 이용하는 클라우드 컴퓨팅이 대안으로 부상



설치/유지보수 용이

- 사용자가 복잡한 하드웨어나 소프트웨어에 대한 세팅 작업 최소화
- 사용자 스토리지 자원 추가 요구에 대한 빠른 대응 가능

빠른 서비스 제공

- 기존 방식에 비해 빠른 데이터 전송속도 가능
- 이론적으로 스토리지 용량에 대한 무한한 확장 가능

데이터 효율적 관리

- 컴퓨팅 리소스 및 스토리지에 대한 유연한 확장
- 대용량 데이터 분산 처리를 통한 효율적 관리

오픈소스 활용

- 오픈소스 활용을 통한 확장성 증대
- 복잡한 라이선스에 대한 문제 불필요
- 국제표준 준수를 통한 활용 가능성 증대

1. PVFS2 소개 및 개발 내용

innogrid

1.1 PVFS2 개요

1.2 PVFS2 구성도

1.3 PVFS2 특징

1.4 PVFS2 테스트 결과

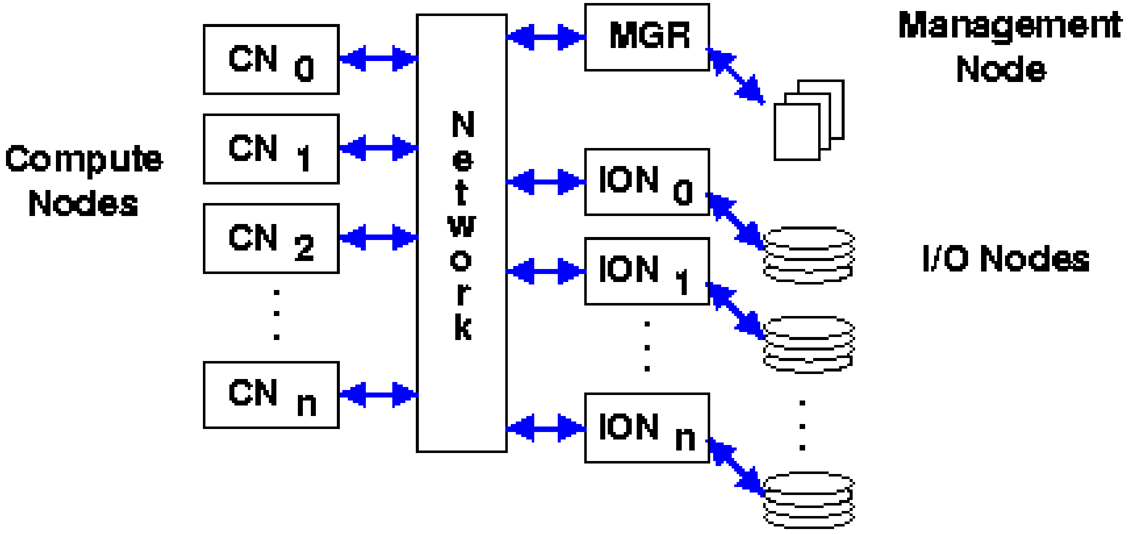
1.5 개발내용

1. PVFS2 개요

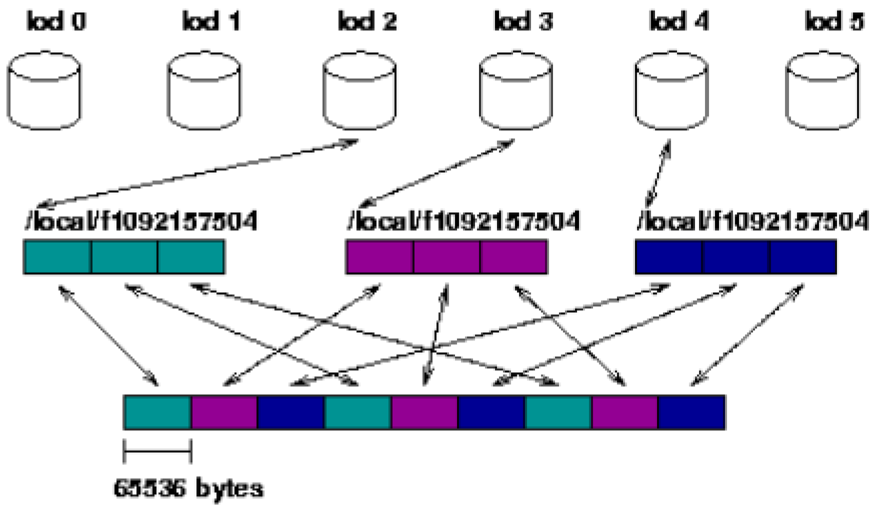
- Parallel Virtual File System 2
- 분산 파일 시스템
  - 다중 서버의 로컬 디스크를 하나의 디스크처럼 서비스
  - 대용량 스토리지 기반 디자인
- 병렬 파일 시스템
  - 다중 디스크에 쪼개져 있는 하나의 파일에 동시 접근
  - Parallel applications을 위해 디자인
  - MPI-IO 최적화
- C 기반 오픈 소스

2. PVFS2 구성도

구성요소	내용
Metadata Server(mgr)	▪ 파일 시스템의 네임스페이스 관리 (파일 시스템 트리 및 모든 파일과 디렉터리에 대한 메타데이터 유지) ▪ 파일의 분산 정보 관리
IO Node (iod)	▪ 데이터 청크들을 저장 및 탐색 제어
PVFS native API (libpvfs)	▪ 사용자 레벨의 PVFS 서버들의 접근 제공 API (예) PVFS_sys_create, PVFS_sys_read, PVFS_sys_write 등등

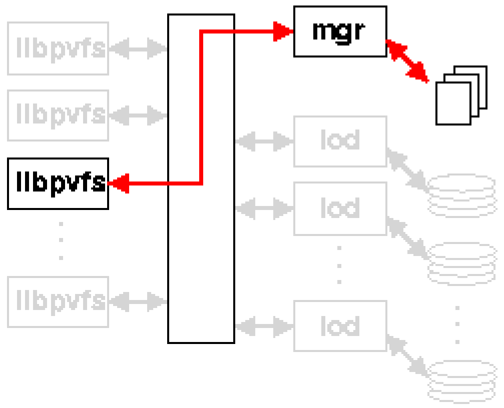


2. PVFS2 구성도 - 계속

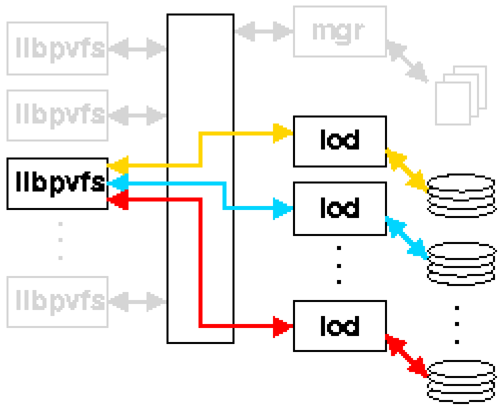


Data 처리

- 하나의 파일은 64KB로 stripe 하여 저장
- 설정 값에 따라 동시 Access



Metadata Access



Data Access

Data Read & Write 처리 순서

1. Metadata server에 데이터 위치 정보 취득
2. 각 io node에 직접 연결
3. Stripe 데이터 read & write

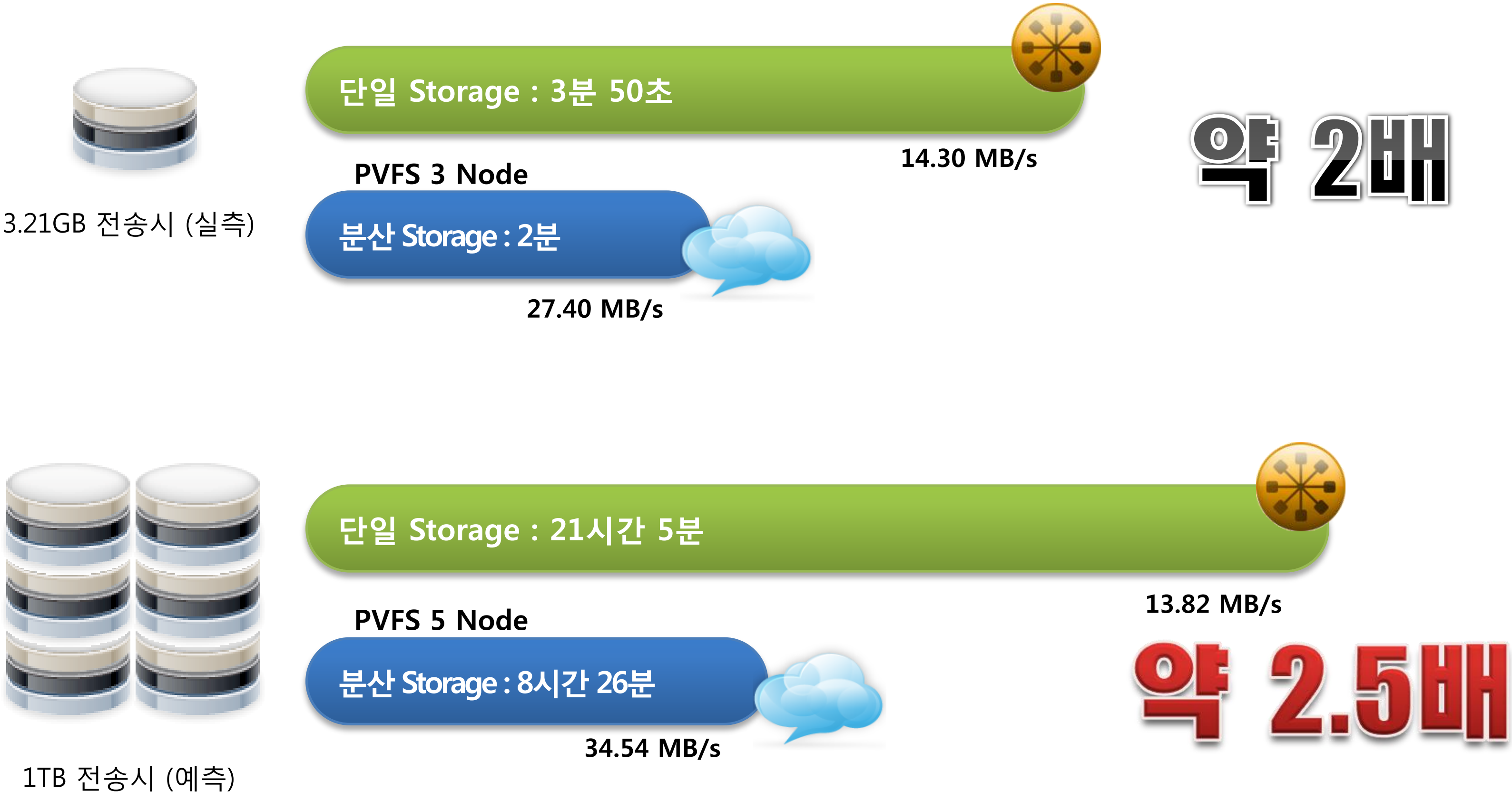
3. PVFS2 특징

특징	내용	장단점
다양한 파일 시스템 접근 방법	1. API 이용 – libpvfs 2. 커널모듈 이용 (UNIX file system API, POXIS)	1. 응용 프로그램 수정 2. 응용 프로그램 수정 불필요 (속도 떨어짐)
단일 파일 병렬 접근	쪼개진 파일을 동시에 접근하여 취득 가능	▪ 장점: 접근 속도 빠르다. ▪ 단점: 처리량 증가
메타 데이터 서버 및 클러스터는 초기 확정	메타데이터 영역의 설정이 초기에 정적으로 확정 클러스터 추가 삭제 시 모든 서버 재 시작	▪ 단점: 동적 확장 및 축소 불가
메타데이터 서버 다중 설정 가능	옵션에서 메타데이터 서버를 다중으로 설정	▪ 장점: 메타 정보가 분산되어 처리 되므로 병목현상 해결 가능 ▪ 단점: 각기 다른 메타 정보를 가짐으로 병목은 해결되지만 복제가 불가능하여 손실 가능
데이터 복제 배치 불가	같은 내용의 데이터 청크가 복제되지 않음	▪ 장점: write 시 응답속도가 빠르다. ▪ 단점: 클러스터 장애 시 데이터 손실 가능 RAID-5등의 하드웨어적 방법 필요

3. PVFS2 특징 - 계속

특징	내용	장단점
클라이언트 캐시 없음	모든 Write를 IO 노드로 직접 보낸다.	▪ 장점: 복잡한 cache consistency protocol 불필요 (create는 있으나 close 없음) ▪ 단점: small write가 많으면 성능 저하
단일 파일의 2 Write 허용	단일 파일의 다른 데이터 청크에 동시 Write 가능	▪ 장점: 빠른 속도 ▪ 단점: 같은 청크의 2 Write 발생할 가능성 있음 - 데이터 무결성을 보장 불가
Round robin 데이터 배치	데이터 저장 시 io node는 모두 같은 level로 인식	▪ 장점: 단순한 알고리즘 ▪ 단점: 네트워크 토폴로지를 이용하지 않아 성능 저하될 수 있음
확장 된 속성 제공	1. Metadata type에 End-of-File(EoF) offset 이라는 확장 된 속성 제공 2. 파일의 분산 정보를 저장	1. 장점: 클라이언트 상에서 File appending 개발 시 EoF 속성을 읽어와 그 이후부터 파일 내용을 Add 하도록 개발 가능 2. API로 제공하지는 않으나 소스 분석으로 개발에 사용 가능
설치의 편리	5단계 정도의 간단한 설치 과정	▪ 장점: 단순한 설치과정으로 편리 ▪ 단점: 다양한 옵션이 제공되지 않아 관리의 한계가 있을 수 있음

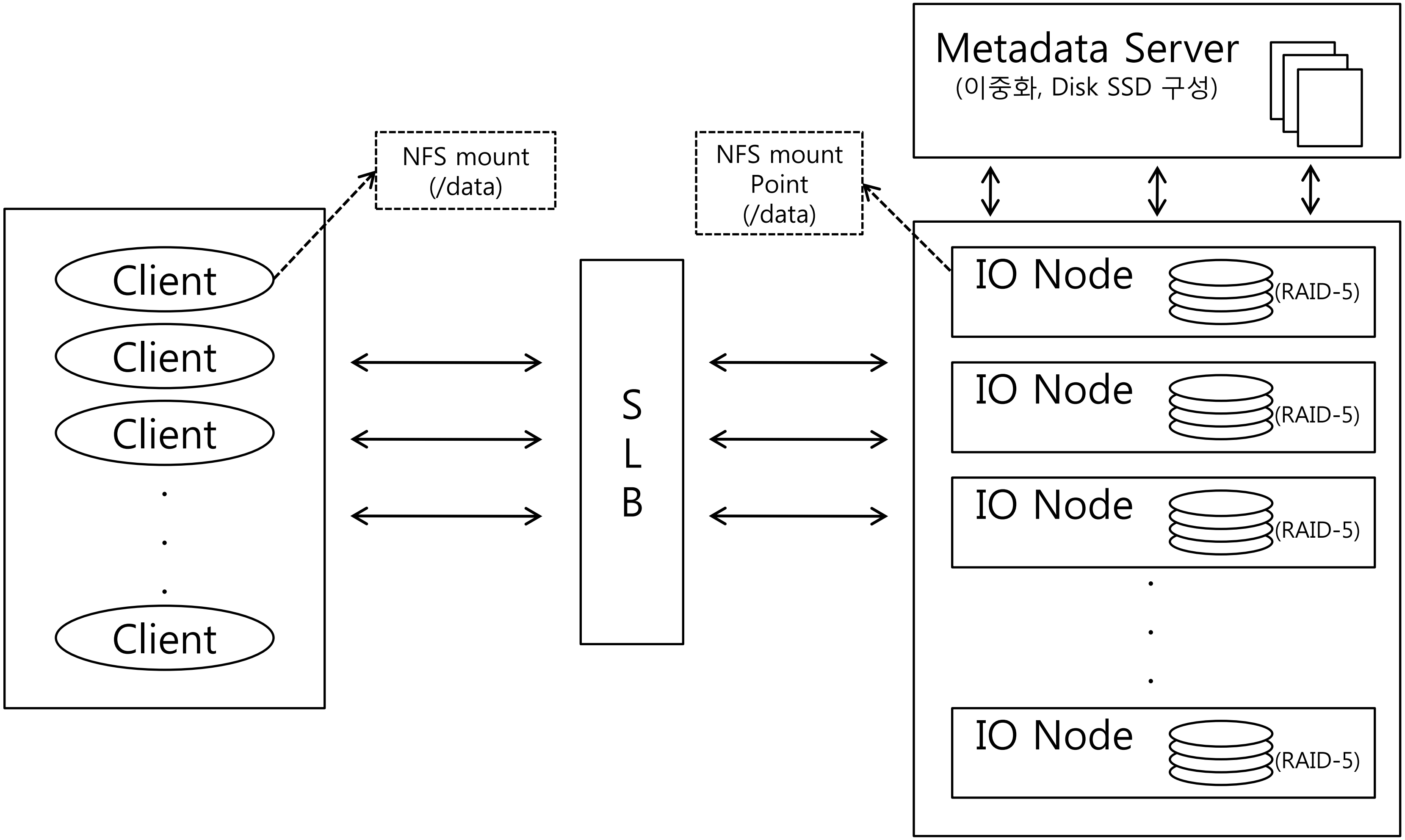
4. PVFS2 테스트 결과



5. 개발 내용

1. Metadata Server 이중화
2. Scale 보장을 위한 동적 확장 지원
3. Small Write 처리 속도 성능 개선
4. 개발자 가이드 및 API 문서

5. 개발 내용 – 개발 시스템 구성도



2. 기대효과

2.1 기대효과

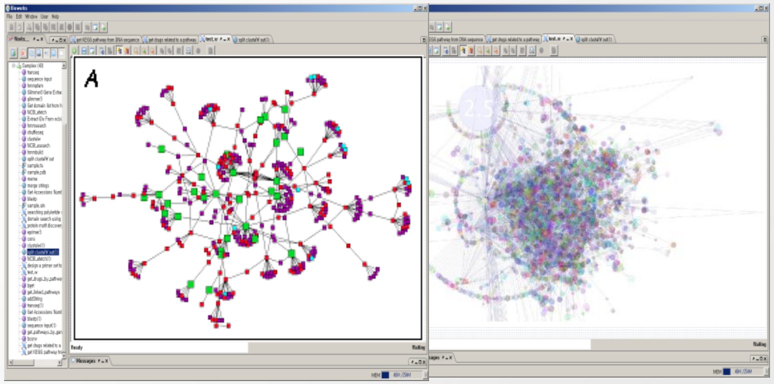
2.2 향후 응용 및 활용분야

기대효과

1. 공개SW 기반 대규모 시스템 구현 시범 사례로 공개 SW에 대한 신뢰성 확보
2. 분산 컴퓨팅 기술 및 클라우드 컴퓨팅 활성화에 기여
3. 시스템의 효율성을 제고하고 관리 비용 절감 기대
4. 타 산업과의 연계로 확대되는 산업적인 파급 효과

향후 응용 및 활용 분야

생명공학 정보 데이터 분석



대용량 멀티미디어 데이터 처리

감사합니다