

GlusterFS 분산 파일 시스템 모니터링 설계

이정현*

**고려대학교 컴퓨터정보통신공학과

e-mail : jhyunlee@korea.ac.kr

Monitoring Design for Distributed File System GlusterFS

Jeong-Hyun Lee *

** Dept. of Computer Engineering, Korea University

요약

최근 Social, Mobile, IoT 등에 기반한 비즈니스 데이터의 폭증과 함께 이를 저장하고 처리하기 위한 Big Data 플랫폼, 분산 스토리지 기술 등이 사용되고 있다. 최근 제안된 분산 스토리지들은 클라우드 기반 기술과 Scale-Out 아키텍처를 적용하여 데이터의 증가에 대응할 수 있는 구조를 갖추고 있다. 분산 스토리지의 노드가 수백 대 이상으로 증가하는 경우 수작업을 통한 관리방법으로는 운영관리는 불가능하며 자동화된 운영관리와 모니터링 방법이 필요하다. 본 논문에서는 GlusterFS 분산 스토리지를 대상으로 네트워크, 서버, 디스크, 스토리지 서비스 등 시스템 상태를 구간별로 모니터링 할 수 있도록 설계하였다. 이를 통해 분산 스토리지 전체 인프라에 대한 모니터링과 스토리지 서비스 수준을 모니터링 할 수 있도록 하였다.

1. 서론

인터넷 통신망의 발달과 이와 관련된 서비스가 증가하면서 정보의 양이 급격하게 많아지고 그 크기도 점점 커지게 되었다. 또한 클라우드 산업의 발달로 대용량 데이터의 저장과 그 처리의 요구가 증가하고 있다[1].

빅데이터를 처리하기 위해서는 기본적으로 많은 양의 데이터를 효율적으로 저장하고 관리하는 방법이 필요한데, 이에 따라, 대용량 데이터를 저장하기 위한 분산 파일시스템의 중요성이 높아지고 있다. 대표적으로 널리 사용되는 분산 파일 시스템 중에는 GlusterFS 파일 시스템이 있으며 수평적 확장(Scale-out)이 가능한 환경을 제공하고 있다[2].

대용량의 데이터를 처리할 때 스토리지 시스템은 병목지점이 될 수 있으므로 스토리지 시스템의 I/O 성능을 개선하고 보다 효율적인 시스템 운영을 위해서 워크로드의 다양한 I/O 특성 및 상호 시스템간 연관성을 분석해야 하며 이를 모니터링 할 수 있는 도구가 필요하다[3].

다수의 분산 스토리지 노드로 구성된 GlusterFS 분산 파일 시스템을 안정적으로 운영하기 위해서는 네트워크를 포함하는 하드웨어적 구성요소뿐 만 소프트웨어 프로세스와 처리량/응답시간 등 통합적 모니터링이 필수적이다.

본 논문에서는 분산 스토리지 노드들을 통합적으로 모니터링하고 관리할 수 있는 방안을 설계 제안한다. 이를 이용하여 GlusterFS 분산 파일 시스템의 장애 관리뿐만 아니라 성능 분석을 통해 서비스 수준에 대한 관리가 가능하게 한다.

2. 관련연구

2.1. 분산 시스템 모니터링

전통적으로 고성능 시스템들은 시스템의 확장성에 시스템 기본 설계의 초점을 맞추고 있었다. 시스템이 점점 더 분산되고 느슨하게 결합하는 아키텍처 변화는 새로운 문제를 발생시키고 있다. 이러한 문제들은 물리적 분산의 증가, 장시간 분산처리 서비스, 시스템의 확장과 진화 등으로 인해 발생한다.

시스템의 물리적인 분산은 중복 구성, 독립적 실패, 신뢰성이 없는 컴포넌트를 의미한다. 이러한 시스템의 분산은 노드 수의 증가와 함께 어플리케이션의 관리 오버헤드가 증가하는 어플리케이션 설계가 필요하다. 장시간 분산처리 서비스는 고가용 서비스를 의미한다. 이것은 다양한 유형의 시스템 실패에 대해서 대응할 수 있는 어플리케이션을 의미한다. 시스템의 확장과 진화는 시간이 지남에 따라 하드웨어와 소프트웨어가 변경되는 것을 의미한다.

분산 모니터링 시스템을 위한 핵심 설계 과제는 다음과 같다.

- 확장성: 모니터링 시스템은 노드 수의 증가에 따라 자연스럽게 확장 증가될 수 있어야 한다.
- 견고성: 네트워크와 노드의 다양한 장애에 대응 할 수 있어야 한다. 수많은 노드로 구성되기 때문에 장애 현상은 피할 수 없으며 일상적인 현상으로 발생하기 때문이다.
- 유연성: 수집하려는 데이터의 종류에 대해서 유연하게 확장할 수 있어야 한다. 모니터링 대상을 미리 확정하는 것은 불가능하기 때문이다. 새로

- 운 데이터를 수집하는 것이 가능해야 한다.
 - 관리성: 시스템 노드의 증가는 시스템관리의 오버헤드의 증가를 발생시킨다. 수작업에 의한 시스템 관리를 최소화 해야 한다.
 - 이식성: 다양한 운영체제와 CPU 아키텍처에 이식 할 수 있어야 한다. 다양한 이 기종 시스템에 사용이 가능해야 한다.
 - 오버헤드: 모니터링 시스템은 네트워크, I/O, 메모리, CPU 를 포함하는 시스템 자원에 대해서 오버헤드를 발생시킨다. 고성능이 요구되는 HPC 환경에서는 특히 중요하다[4].

2.2. 분산 파일 시스템 모니터링 기술

스토리지 클라우드의 고품질의 서비스를 위해서는 모니터링 시스템이 필수적이며 클라우드 스토리지에 대한 정보, 응답시간에 대한 정보, 데이터 전송에 대한 정보 등을 모니터링 해야 한다.

모니터링 필요한 항목은 스토리지 사용량, 스토리지 사용실패, 요청 시 지연시간 통계, 요청 별 평균 전송 속도, 전송 실패 빈도, 평균 재전송 횟수, 최대 트래픽 사용량, 최대 동시 요청 수 등이다[5].

이러한 클라우드 스토리지와 분산 파일 시스템을 모니터링 하기 위한 시스템으로는 Ganglia, Nagios, Chukwa 등을 사용할 있다.

- Ganglia

Ganglia 는 클러스터나 그리드와 같은 고성능 컴퓨팅 시스템의 상태를 모니터링해주는 분산 시스템 모니터링 도구이다. 모니터링하고 있는 시스템에 대한 정보를 실시간으로 제공하거나 누적된 통계정보를 제공한다. 각 클러스터 노드가 전송해주는 정보를 모니터링 서버가 저장하고 웹으로 그 정보를 보여줌으로써 각 클러스터 노드의 상태를 쉽게 볼 수 있다. 데이터 표현을 위해서 XML, 이기종 간의 데이터 전송을 위한 XDR, 데이터 저장과 가시화를 위한 RRptool 등을 이용한다[5].

- Nagios

Nagios 는 컴퓨터 시스템과 네트워크를 모니터링 할 수 있는 리눅스 기반 어플리케이션이다.

주요 시스템 구조는 Nagios 코어라는 중앙 애플리케이션이 있고, 추가 기능을 이용할 수 있는 Nagios 플러그인, 프론트엔드, 구성 도구가 지원되는 형태다.

Nagios 는 시스템 모니터링 프로그램으로 호스트, 서비스, 네트워크를 모니터링할 수 있다. 호스트 자원 관리, 성능, 웹페이지 표시등의 기능이 존재하며, CPU 사용률, 디스크 사용률, 로드 평균 시간, process 개수, users 개수, 파일 개수, 파일 크기 등을 모니터링할 수 있다. 모니터링 결과 문제가 있으면 메일과 SMS 를 이용하여 문제에 대한 경고를 통해 빠른 대처가 가능하도록 한다.

서버의 개수가 많은 경우에는 Distributed Monitoring 을 이용하여 여러 대의 서버에서 분산 모니터링 할 수 있으며, 모니터링 결과를 Database 에 저장할 수 있다.

[3].

[...] 분산 시스템을 모니터링하기 위한 다양한 기술과 방법이 제공되고 있지만, 분산 파일 시스템에 고유한 요구사항을 기반으로 대한 진행된 실험과 연구가 제한적이다.

본 연구에서는 수백 노드 이상으로 구성된 분산화
일 시스템을 모니터링하기 위한 요구사항을 기반으로
모니터링 방법을 설계 제안한다.

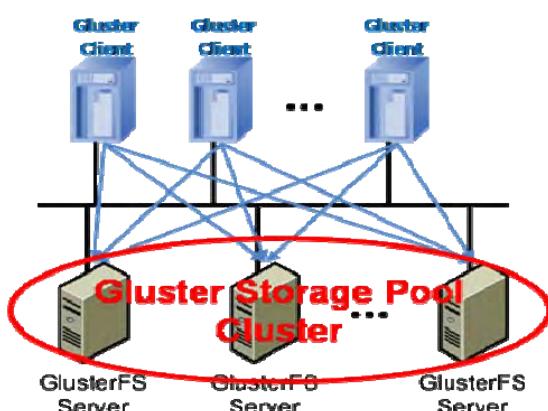
3. GlusterFS 분산 파일 시스템 모니터링 설계

3.1. GlusterFS

본 논문에서 제안하는 분산 파일 시스템 모니터링 설계는 파일단위 저장 서비스를 제공하는 GlusterFS 대상으로 한다.

GlusterFS 는 2005년 Gluster라는 회사에 의해 개발되었으나, 2011년 RedHat에 인수되어 RedHat Enterprise Linux와 함께 Gluster Storage로 상업용 서비스를 제공하고 있다.

GlusterFS 는 [그림 1]과 같이 일반 상용 하드웨어를 이용하여 Gluster Storage Pool Cluster 를 구축할 수 있는 Scale-out 네트워크 파일 시스템이다.



(그림 1) GlusterFS 구성도

데이터와 대역폭이 집중적으로 필요한 작업, 미디어 스트리밍을 위한 대용량 분산 스토리지를 구축할 수 있는 GPL 라이선스로 배포되는 오픈 소스 소프트웨어이다.

GlusterFS 는 분산 파일 시스템의 하나로, 데이터를 분산 저장을 위해서 메타 데이터 서버를 이용하지 않고 DHT(Distributed Hash Table)를 이용한다. 따라서 메타 데이터 서버에 메타 데이터 요청이 집중되는 문제로 성능이 병목 되지 않는 장점이 있다. GlusterFS 는 일반적으로 클라이언트 쪽에서 FUSE 를 이용해서 마운트(Mount)해서 사용한다[2].

GlusterFS의 Scale-out 확장성을 지원하는 수평적 구조의 스토리지 솔루션이다. 단순하게 리소스를 추가하는 것 만으로 스토리지의 용량과 성능을 증가시킬 수 있고, 디스크, 컴퓨팅, I/O 리소스는 독립적으로 증가시킬 수 있다. 그리고 10GbE와 인피니밴드 고속 네트워크를 이용하여 클러스터를 구성할 수 있

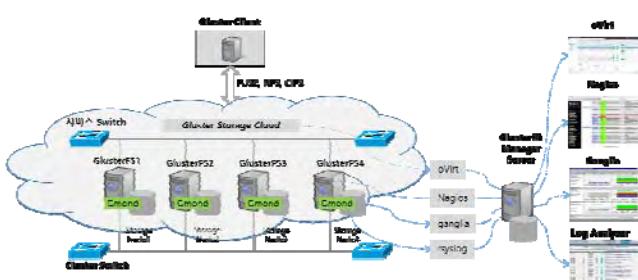
다.

메모리를 확장을 통해 수 PB 스토리지 용량과 수 천대의 클라이언트 서비스 요청을 처리할 수 있다. GlusterFS는 단일 글로벌 네임스페이스를 사용하여 다양한 유형의 데이터 처리에 대해서 우수한 성능을 제공한다. GlusterFS는 가상화된 스토리지 풀과 스토리지 용량은 TB/PB 규모까지 확장시킬 수 있는 확장성과 신뢰성을 가지고 있다[6].

GlusterFS 분산 스토리지는 파일 레벨 스토리지에 기반을 두고 있으며, 웹 사이트, IPTV, 소셜, 앱 등의 대용량 파일들을 저장하고 스토리지 노드의 제한 없는 확장이 가능하며 특히, Read I/O에 집중하여 공유하는 파일 서비스에 적합하다[7].

3.2. 시스템 설계 방안

[그림 2]는 GlusterFS를 모니터링하기 위한 시스템 구성도를 보여주고 있다. GlusterFS 분산 노드를 하드웨어 계층, 프로세스, 서비스 계층 별로 모니터링하고 각각의 로그를 중앙 저장소에 취합하여 분석 활용할 수 있도록 설계하였다.



(그림 2) GlusterFS 모니터링 구성도

(1) syslog 와 GlusterFS log 통합 저장 및 분석

노드 별 시스템 로그는 rsyslog를 이용하여 중앙 서버에 수집하여 RDB에 저장한다. RDB에 저장된 내용은 log 분석 도구를 이용하여 분석할 수 있도록 한다. 분석 도구는 LogAnalyzer를 사용한다. 스토리지 노드에서 발생된 GlusterFS log는 통합 저장하여 검색 및 분석 할 수 있도록 하고 지속적인 로그의 증가를 고려하여 저장주기에 따라 일정기간만 보존하고 삭제하도록 한다.

(2) ganglia 를 통한 시스템 정보 통합

GlusterFS 스토리지 노드의 하드웨어 동작 상태를 모니터링 하기 위해서 각 노드에 gmond 앤이전트를 설치한다. gmond에서 전송하는 시스템 정보는 Gluster Manager Server의 gmetad 데몬으로 전달된다. 수집된 정보는 PHP 언어로 작성된 ganglia 웹 인터페이스를 통해서 브라우저로 표시된다.

(3) nagios 를 통한 주요 프로세스 상태 점검

Gluster Manager Server에서 스토리지 서버 노드에게 시스템 상태와 프로세스 상태 정보를 요청하여 주요 프로세스의 동작 상태를 점검한다. 또한 스토리지 노드에 정의한 이벤트가 발생할 경우 관리 서버로 보고

한다. 또한 사전에 정의한 횟수 만큼 연속적으로 오류가 발생하는 경우 정해진 관리자에게 알림 경보 기능 제공한다.

(4) oVirt 를 통한 GlusterFS Cluster 관리

oVirt는 KVM, Xen, VirtualBox 등과 같은 가상 머신을 관리하는 오픈소스 플랫폼 가상화 관리도구이며, 레드햇 엔터프라이즈 가상화(RHEV)의 커뮤니티 버전이다. oVirt는 레드햇 스토리지 매니지먼트 콘솔을 포함하고 있으며, 가상화 및 프라이빗 클라우드 관리 기술을 포함한다.

oVirt의 운영관리 모드를 GlusterFS 전용모드로 설정하여 운영한다. 클러스터 관리 기능을 통해 GlusterFS Cluster 생성하거나 제거할 수 있으며, 볼륨 관리를 통해서 각각의 볼륨을 생성/기동/종료/삭제/Re-Balancing 작업을 수행할 수 있다.

3.3. 스토리지 클라우드 모니터링 요구사항

수천 노드 이상에서 사용하는 스토리지 클라우드를 모니터링하기 위해서는 모니터링 기술에 추가적으로 고려해야 할 요구사항이 존재한다.

스토리지 클라우드의 수천 노드에 대한 I/O 정보를 수집하게 되면 로그의 버스트 쓰기가 발생하므로 이러한 오버 헤드가 허용 범위 내에 있어야 한다. 모니터링 시스템이 중단되어도 슈퍼컴퓨터 시스템에 영향을 미치지 말아야 한다. 특정 파일시스템에 종속되지 않아야 한다. 모니터링 시스템의 오버 헤드 또한 허용 범위 내에 있어야 한다. 대량의 I/O 정보를 저장 및 분석하기 위해 모니터링 서버는 분산된 형태를 지원해야 하며, 저장해야 할 I/O 데이터 크기를 최소화 시켜야 한다. 사용자에게 수집된 I/O 정보를 다양한 형태의 그래프 및 수치로 데이터를 표현해 주어야 한다[3].

3.4. GlusterFS 모니터링 항목

분산 스토리지의 안정적인 서비스를 위해서는 네트워크, 서버 하드웨어, GlusterFS 프로세스, 분산 파일 서비스 등에 대한 서비스 계층별 모니터링이 필요하다.

모니터링 방법은 데이터 노드에 설치된 애이전트에서 정보를 전송해주는 방법과 중앙관제 노드에서 주기적으로 정보를 요청하는 방법이 있다. GlusterFS 노드는 Client에서 분산 저장된 스토리지 노드에 개별적으로 접속하여 서비스를 제공하기 때문에 전체 클러스터 노드에서 제공하는 서비스 용량을 파악하는 것이 중요하다.

GlusterFS 시스템 구성 요소 별 모니터링 항목은 다음과 같다.

- 서버노드: CPU 사용량, 메모리 사용량, 스왑, 네트워크 트래픽, DISK I/O
- 서버 syslog
- GlusterFS log: glusterd log, bricks log, rebalance log, self heal deamon log, quota log, gluster NFS log, Geo-replication 등

- GlusterFS 프로세스 : Gluster Management (glusterd), Self-Heal, Quota (Quota daemon)
- GlusterFS Cluster : Volume Status, Quota, Replication, Volume Self Heal
- GlusterFS 볼륨 및 Brick 상태
- Client 접속 및 서비스 지원 상태, 데이터 전송 및 전달 상태

3.5. 설계 검증

본 논문에서 제안한 설계를 검증하기 위해 Oracle VM VirtualBox 가상 서버 환경에서 시스템 구성을 진행하였다. VirtualBox 를 설치한 서버의 하드웨어 제약으로 인해 시스템 성능시험은 진행하지 않고 시스템 구성요소간 연동과 설계 항목의 기능적 동작 상태를 점검하였다.

GlusterFS 클라이언트에서 IOzone 명령을 통해 간단한 스토리지 부하 발생 시키고 시스템 Fail 등을 발생시켜 시스템 모니터링이 정상적으로 동작되는 것을 확인하였다.

[표 1]은 설계 검증 환경으로 6 개의 가상 리눅스를 설치하였으며, 리눅스 패키지는 GlusterFS 를 설치 운영하기 위한 최소한으로 설치하여 모니터링 설계를 검증하였다.

<표 1> 설계 검증 환경

구분	구성	비고
GlusterFS 서버	1core, 512MB CentOS 7.1	Virtual Box 4 대
	Gluster 3.6, rsyslog, ganglia3.7.1, oVirt	
Gluster Manager 서버	1core, 512MB CentOS 7.1	Virtual Box 1 대
	Ganglia 3.7.1, nagios4.0.8, rsyslog, MySQL, httpd, oVirt	
GlusterFS Client	1core, 512MB CentOS 7.1	Virtual Box 1 대
	Gluster FUSE, IOzone	

4. 결론

Gluster FS 는 비정형 데이터 보관, 아카이빙, 재해 복구 시스템, 가상머신 이미지 저장, 컨텐츠 클라우드, 빅데이터 분야에 활용되고 있는 분산 파일 시스템이다.

다수의 스토리지 노드가 네트워크를 이용한 클러스터 형태로 서비스를 제공하기 때문에 스토리지 노드에 대한 모니터링과 관리는 매우 중요하며, 이러한 관리는 클러스터를 구성하는 하드웨어와 소프트웨어 요소에 대해서 각각 모니터링 관리가 수행되어야 한다.

본 논문에서는 GlusterFS 의 스토리지 서버, 시스템 로그, 프로세스 데몬, 볼륨 등 시스템 구성 요소 별로 모니터링 할 수 있는 설계를 진행하였다. 이를 통해서 고장 노드를 자동 식별하고 관리자에게 통보하게 하여 장애관리가 가능하게 하고, 스토리지 노드의 DISK IO 와 네트워크 트래픽 모니터링을 통해 성능과 품질을 관리할 수 있도록 하였다. 또한 시스템 로그를 DB 에 저장하여 장애 로그를 분석하고 사전 정후

와 관련된 연관성 등을 향후 분석 할 수 있도록 하였다.

향후 연구에서는 실제 x86 서버에서 GlusterFS 서버와 모니터링 설계를 구현하여 설계 검증하고, 다수의 클라이언트가 동시에 접속하여 서비스를 요청하는 상황에서도 클라이언트에 대해서 균등하고 안정적 서비스를 제공할 수 있는 방안에 대해서 연구할 계획이다.

참고문헌

- [1] 최대순, “대용량 분산파일 시스템의 복제배치기법 분석”, 한국컴퓨터종합학술대회논문집, 2012, p. 373
- [2] 김덕상, “SSD 환경아래에서 GlusterFS 성능 최적화”, 한국컴퓨터정합학술대회, 2015, p. 89
- [3] 원유집, “슈퍼컴퓨터 스토리지 시스템을 위한 위크로드 모니터링”, 정보과학회지, 2013, pp 51-55
- [4] Matthew L. Massie, “The ganglia distributed monitoring system: design, implementation, and experience”, Parallel Computing, 2004, pp. 819-820
- [5] 양종원, “QoS 모니터링을 위한 스토리지 클라우드 모니터링 설계”, 한국엔터테인먼트산업학회논문지, 2010년, pp. 58-59
- [6] Dawei XIAO, Cheng ZHANG, Xiaodong LI, “The Performance Analysis of GlusterFS In Virtual Storage”, AMEII , 2015, pp. 199- 200
- [7] 박정수, “클라우드 컴퓨팅을 위한 클라우드 스토리지 기술 분석”, 한국정보통신학회, 2012, p. 1134