DSV 기반 서버 고가용성을 위한 적응적 서버 선정 알고리즘

김현우*, 변휘림*, 송은하**, 정영식*
*동국대학교 멀티미디어공학과
**원광대학교 후마니타스 칼리지
e-mail : hwkim@dongguk.edu

Adaptive Server Selection Algorithm for High Availability based on DSV

Hyun-Woo Kim*, HwiRim Byun*, Eun-Ha song**, Young-Sik Jeong*
*Dept. of Multimedia Engineering, Dongguk University
**Dept. of Humanitas Collage, Wonkwang University

요 약

현재, 급격한 IT 기술 발전의 산물인 스마트 디바이스에서 발생되는 데이터의 양이 기하급수적으로 증가함에 따라 효율적으로 저장하기 위한 다양한 연구 및 기술 개발이 진행되고 있다. 이 중에 Desktop Storage Virtualization(DSV)은 비가용 데스크톱 스토리지 자원을 하나의 통합된 스토리지로 만들고 이를 스토리지 사용자 니즈에 따라 스토리지를 제공한다. DSV 는 분산된 데스크탑의 비가용 자원을 통합함에 따라 신뢰성이 매우 중요시된다. 그러나 DSV 환경에서 산발적 서버다운 및 기타 장애시 대응체계가 미흡하다. 본 논문에서는 이러한 상황을 고려한 Server Selection Algorithm(SSA)를 제시한다. SSA 는 서버 장애 발생시 대체 서버를 분산된 데스크탑이 적응적 인지 및 서버를 변경함으로써 고가용성을 제공한다.

1. 서론

최근, 스마트 디바이스의 고기능성, 고컴퓨팅 및 휴대의 용이로 인해 다양한 서비스가 융합되고 있다. 이에 스마트 디바이스 및 다양한 서비스에서 생성되는 대용량 데이터 저장을 위한 연구 및 기술 개발이 활발하게 진행되고 있다. 이 중에 Desktop Storage Virtualization(DSV)는 분산된 데스크톱의 비가용 스토리지 자원을 단일 통합 스토리지로 가상화한다. 가상화된 스토리지는 사용자 요청에 따라 스토리지를 제공한다. 그러나 서버다운 및 예기치 못한 장애 발생시 데이터 접근 불능 및 데이터 손실 가능성이 매우 높다. 이에 DSV 환경에서 사용자에게 보다 높은 신뢰성 제공을 위한 스토리지 영속성, 지속성, 무결성이 매우 중요시된다[1, 2, 3, 4, 5].

본 논문에서는 DSV 환경에서 스토리지 서버의 고가용성을 위해 장애 발생시 적응적 서버 선정 및 운용이 가능한 SSA(Server Selection Algorithm)을 제안한다. SSA 가 적용된 DSV 환경에서는 데스크탑의 추가 및 주기적으로 연결된 데스크탑에게 IP 기반의 서버 리스트를 보낸다. 서버 장애 발생시 적응적으로 연결된 데스크탑은 사전에 동기화한 IP 기반 서버 리스트를 이용해 장애 서버 대신 대체 서버가 수행한다. 각 데스크탑이 동일한 서버 리스트를 가짐으로써 수초 내에 장애 복구가 가능하다.

* 이 논문은 2014 년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. NRF-2014R1A1A2053564). 또한 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 대학 ICT 연구센터육성 지원사업의 연구결과로 수행되었음 (IITP-2015-H8501-15-1014).

2. ZooKeeper 와 GSLB

2-1. ZooKeeper

Zookeeper[6]는 서버간 데이터를 복제하는 방법으로 고가용성을 제공한다. 분산된 컴퓨팅 환경에서 내부적으로 단일 서버의 매 작업 처리 결과를 지정된 대체 서버에 동기화시킨다. 이때 관리하는 Name Server 의 데이터 처리 불능 및 기타 장애 발생시 대체 서버가 Name Server 의 역할을 대신함으로써 지속적인 서버 동작이 가능하다. 그러나 서버의 운용보다 데이터 저장을 위한 분배 처리에 비중이 높게 설계되어 연속적인 서버 장애 대처가 미흡하다.

2-2. GSLB (Global Server Load Balancing)

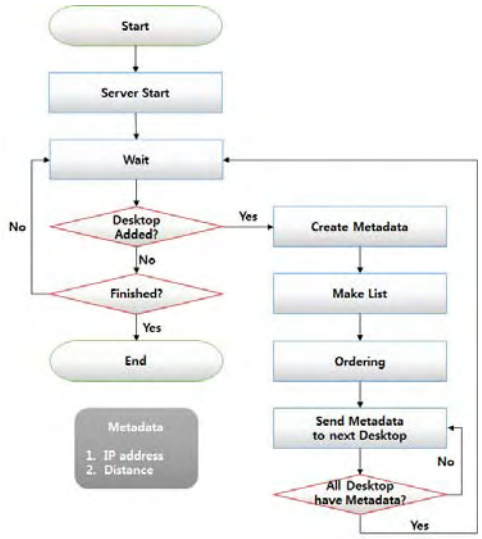
GSLB[7]는 데이터센터가 분산된 환경에서 효율적으로 서버를 선택하기 위해 서버의 가용 상태를 모니터링 한다. 서버의 가용상태 및 네트워크의 상태 모니터링를 통해 작업 요청시 로드밸런싱하여 자원의 가용성을 증대시킨다. 이러한 기법은 각 데이터센터가 동일 데이터를 내재해야만 장애 대응시 데이터 손실을 막을 수 있다. 또한 서버 인프라가 분산된 환경

이라는 가정하에 일정하지 않은 업로드 속도로 인해 데이터 동기화 문제가 야기된다.

3. Server Selection Algorithm

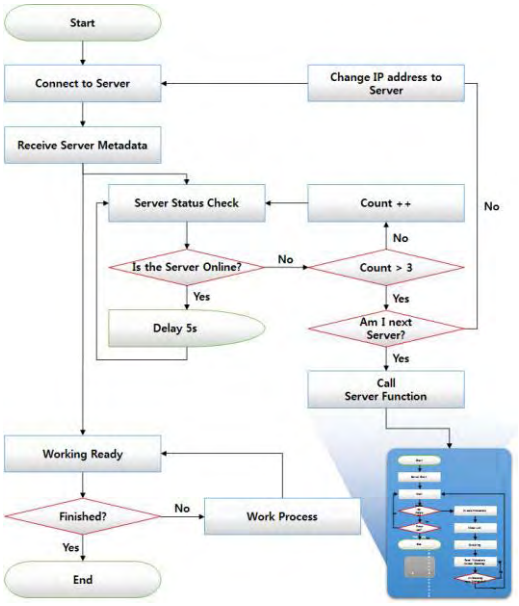
3.1 Server Side

DSV 환경에서 SSA 가 적용된 서버는 새로운 데스크탑이 추가되는 지 모니터링한다. 데스크탑 추가시 동작 중인 서버는 연결된 데스크탑의 IP 기반 정렬을 수행하고 연결된 데스크탑에게 전송한다. 이러한 IP 기반 데스크탑 리스트는 동작중인 서버의 장애 발생 시 대체 서버 선택을 위해 독립된 데스크탑에서 사용된다. 데스크탑이 추가 될 경우에 이러한 동작 과정을 동일시하게 수행되며, 각 데스크탑에게 IP 기반 서버 대체 리스트를 전송한다.



(그림 1) Server-Side Algorithm

3.2 Client Side



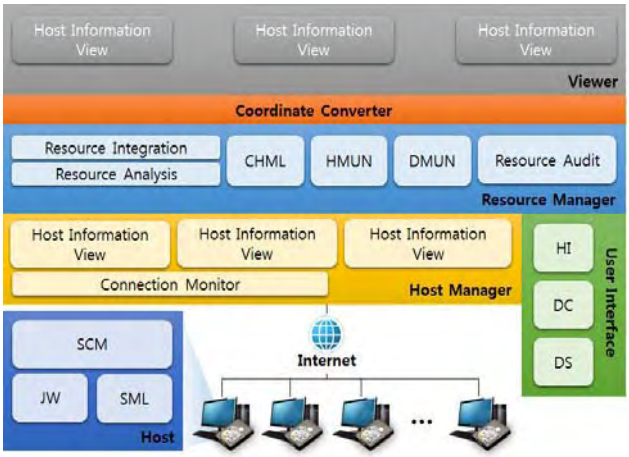
(그림 2) Client-Side Algorithm

클라이언트는 서버에 연결되어있는 독립된 데스크탑으로 주기적인 heartbeat를 서버 측으로 전송하여 생존 여부를 판별한다. 만일 서버의 장애를 감지하는 경우에 기존에 동작 중이던 서버 측에서 전송한 IP 기반 대체 서버리스트를 통해 대체 서버로의 연결을 시도한다. 만일 대체 서버가 자기 자신이라면 Server Side 프로세스를 수행한다. 이외의 다른 데스크탑은 IP를 통해 대체 서버로 기존과 동일한 작업을 수행한다.

4. SSA 의 구조

본 논문에서 제안하는 SSA 는 Server-Side 와 Client-Side 로 나뉜다. Server-Side 는 호스트의 Storage Information 과 Status 를 설정 및 저장하는 User Interface, 연결된 Desktop 을 분석 및 관리하는 Host Manager, Metadata 를 생성하고 생성된 데이터에 대해 List Up 과 Ordering 을 수행하는 Resource Manager, 시각화된 시스템 정보를 제공하기 위한 Coordinate Converter, 마지막으로 시스템 정보를 시각적으로 나타내주는 Viewer 로 구성된다.

각각의 Client-Side 는 주기적으로 서버의 상태를 확인하는 Server Check Manager 와 작업을 수행하는 Job Worker, Metadata 를 저장 및 관리하는 Server Metadata List 로 구성된다.

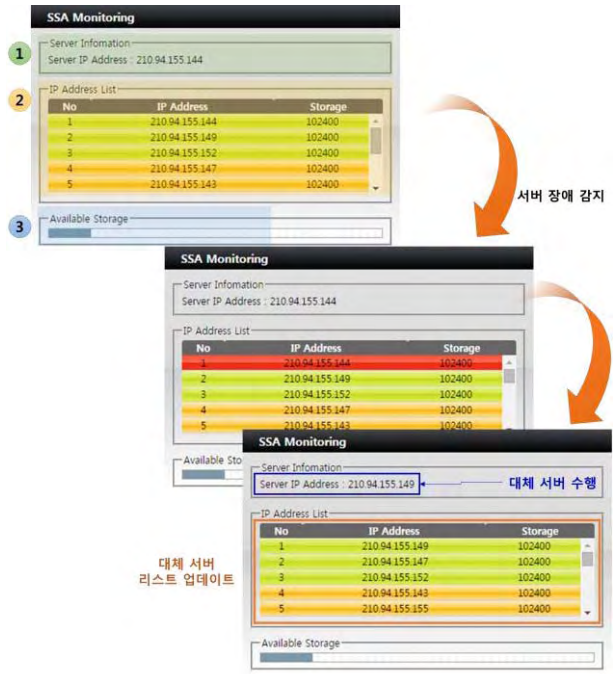


(그림 3) Architecture of SSA

5. SSA 모니터링

그림 4 는 SSA 에 호스트 70 개를 연결한 상태의 동작 모니터링 화면을 나타낸다. 그림 4 내의 ①은 Server Information 으로 현재 서버 IP 주소를 나타낸다. 그림 4 내의 ②는 IP Address List 로 IP 기반 정렬된 리스트를 나타낸다. IP 리스트내의 녹색은 IP 기반 대체 서버의 상위 데스크탑을 의미한다. Available Storage 는 DSV 환경에서 통합된 데스크탑 스토리지의 총 가용 상태를 보인다.

그림 4 내의 적색은 서버 장애를 감지한 것으로 대체 서버가 수행된다. 모든 데스크탑은 장애 서버로의 연결을 제거하고 대체 서버로 연결함으로써 스토리지 서버의 고가용성을 제공한다..



(그림 4) SSA Monitoring

6. 결론 및 향후 연구

본 논문은 DSV 환경에서 분산된 데스크톱 자원의 효율적인 운영을 위한 Server Selection Algorithm(SSA)을 제시하였다. 서버 장애 발생시 동기화된 IP 기반 서버 대체 리스트를 통해 독립된 데스크탑이 차선의 서버를 적응적으로 선택, 대체하여 동작하였다. 이러한 방법은 서버의 고가용성 및 질 높은 QoS 를 보장한다. 또한, 데스크톱 추가 및 제거가 용이하고 수천대로 연결되어 있는 데스크탑 환경에서 최소 2 개까지 적용가능하기 때문에 인위적인 서버의 전체 다운이 아닌 이상 영속적인 서비스 제공이 가능하다.

향후에는 각 데스크톱의 통계기반 실제 가용성, 스토리지 여유 공간 등을 고려한 메타데이터 목록을 생성하여 차선의 서버 선택에 반영하는 연구를 진행하고자 한다. 이를 통해 더욱 효율적이고 안정적인 DSV 를 구현하고자 한다.

참고문헌

[1] Patrick Wendell, Joe Wenjie Jiang, Michael J. Freedman, Jenifer Rexford, “DONAR: decentralized server selection for cloud services,” ACM SIGCOMM Computer Communication Review, Vol. 40, Issue 4, pp. 231-242, Oct. 2010.

[2] Hyun-Woo Kim, Jong Hyuk Park, Young-Sik Jeong, “Human-centric storage resource mechanism for big data on cloud service architecture,” Journal of Supercomputing, online published, Feb. 2015.

[3] Jong Hyuk Park, Hyun-Woo Kim, Young-Sik Jeong, “Efficiency Sustainability Resource Visual Simulator for Clustered Desktop Virtualization Based on Cloud Infrastructure,” Sustainability, Vol. 6, No. 11, pp. 8079-8091, Nov. 2014.

[4] Hu Yuanyuna, Wang Lu, Zhang Xiaodong, “Cloud Storage Virtualziation Technology and Its Architecture,” Applied Mechanics and Materials, Vol. 713-715, No. 2015, pp. 2435-2439, Jan. 2015.

[5] Minseok Kwon, Zuochao Dou, Wendi Heinzelman, Tolga Soyata, He Ba, Jiye Shi, “Use of Network Latency Profiling and Redundancy for Cloud Server Selection,” 2014 IEEE 7th International Conference on Cloud Computing, Anchorage, AK, pp. 826-832, 27. Jun. 2014.

[6] ZooKeeper, http://zookeeper.apache.org/

[7] Lori MAcVittie, Cloud Balancing: The Evolution of Global Server Load Balancing, F5 Network, Inc. Oct. 2012.