고가용성 데이터베이스 구축을 통한 장애 극복 분류 및 관리 기법

Fail Over Analysis and Management for the Database Implement of the High Availability Solution

이병엽
배재대학교 전자상거래학과

Byoung-Yup Lee(bylee@pcu.ac.kr)

요약

인터넷 환경의 급속한 발전과 더불어 국내외 미션 크리티컬한 비즈니스 환경이 온라인에 의해 서비스 되고 있다. 따라서 웹 환경을 통해 처리되어야 할 정보의 양의 급증과, 이의 처리를 위해 여러 개의 단일 서버를 고속의 네트워크로 연결한 고가용성 구현이 가능한 클러스터 컴퓨팅 시스템이 등장하게 되었다. 그 결과 클러스터 기반 DBMS에 관한 연구와 상용화된 솔루션을 통해 국내외적으로 활발히 진행 중이며, 이에 따라 클러스터 기반 DBMS를 효율적이고 최적화된 상태의 관리 연구는 미흡한 실정이다. 따라서 본 논문에서는 클러스터 기반 DBMS를 위한 고가용성 클러스터 솔루션의 최적화된 관리 기법과 이론에 대해 알아본다.

■ 중심어 : | 클러스터 DBMS | 고가용성 |

Abstract

In these days, Internet environment are very quickly development as well On-line service have been using a online for the mission critical business around the world. As the amount of information to be processed by computers has recently been increased there has been cluster computing systems

developed by connecting workstations server using high speed networks for high availability. As a result, this study on a cluster based DBMS and common solution of DBMS venders has been studying with a wide range, as well as It is not good study a management skill for the cluster-based DBMS efficiently and optimization. accordingly, This study find out optimization managements skill and theory of he High availability solution on cluster-based DBMS.

■ keyword : | Cluster DBMS | High Availability |

I. 서 론

2000년대 초반부터 급격하게 보급된 인터넷의 보급과 더불어 최근 금융업무, 온라인 쇼핑 등 미션 크리티컬한 비즈니스가 일반 생활에 깊숙이 사용되고 있다. 이에 인터넷 환경에서 급속히 증대되는 24시간 무정지 서비스 요구를 보다 효율적으로 처리하기 위하여, 저비용 고효율 시스템 성능 및 시스템 확장을 유기적으로 용이하게 하는 클러스터 컴퓨팅 시스템이 필요하게 되었다[1][2]. 그 이유는 인터넷 환경에서 기존의 단일 대용량 데이터베이스 서버를 사용하여 고성능(High Performance)과 고가용성(High availability)의 서비스

접수번호 : #100413-002
접수일자 : 2010년 04월 13일
심사완료일 : 2010년 05월 31일
교신저자 : 이병엽, e-mail : bylee@pcu.ac.kr

를 제공하는 것이 한계에 도달하였기 때문이다. 이러한 한계를 극복하고 더욱 강력한 컴퓨팅 파워와 시스템의 안정적 서비스를 제공하기 위해, 클러스터 기반 DBMS에 대한 연구 및 개발이 활발히 이루어지고 있다. 실제로 Oracle 11g Real Application Server, Informix Extended Parallel Server, IBM DB2 Universal Database EEE 등은 이러한 클러스터 기반의 상용 DBMS이다. 이러한 클러스터 기반 DBMS는 대규모 데이터를 여러 노드에 분산 저장하고 일관성 있게 접근할 수 있는 메커니즘을 제공하며, 또한 클러스터 시스템의 특징인 고성능, 고가용성, 고확장성을 지닌다[10].

이러한 클러스터 기반 DBMS를 효율적으로 관리하기 위해서는 다음과 같은 기능을 가지는 관리 도구가 필요하다. 첫째, 다수의 노드로 구성된 클러스터 시스템을 단일 시스템처럼 인식할 수 있는 환경이 제공되어야 한다. 즉, 시스템 내의 어떠한 노드에서도 클러스터 내의 모든 시스템 자원과 행위를 제공 받을 수 있어야 한다. 둘째 전체 시스템 구성과 각 노드의 부하 및 CPU, 메모리, 디스크 등 자원의 활용 상태의 파악이 용이 하여야 한다. 셋째, 모든 노드의 성능을 최대한 발휘하기 위해 사용자의 요구를 적절히 분산시키는 스케줄링 방법이 필요하다. 마지막으로, 고가용성을 위해 노드 장애가 발생한 상황에 즉시 대응 할 수 있는 장애극복(fail-over)을 위한 방법이 필요하다[4][5].

이를 위해 본 논문에서는 클러스터 기반 DBMS를 위한 고가용성 클러스터의 관리적인 상용 솔루션에 대한 고찰을 한다.

II. 본 론

2. 본론

고가용성 DBMS 구현의 80% 이상은 복구 클러스터링 구조를 기초로 하고 있으며, 이 구조는 미션 크리티컬한 비즈니스에 중요한 데이터베이스의 고가용성을 보장하는, 업계가 인정한 가장 신뢰성 있는 솔루션의 자리를 유지하고 있다. 오라클은 노드 오류가 발생하는 경우에도 데이터베이스의 중단을 최소화하는데 중점을 둔 RAC(Real Application Cluster)를 10g 버전 이상부터 제공 하고 있다. 일부 기업은 고가용성을 위해 복구 클러스터링과 RAC 외에 데이타베이스 복제를 사용하기도 하지만 이 방법은 종종 관리 효율성에 문제를 유발할 수 있다. Microsoft와 IBM도 앞으로 수년 내에 간단하고 향상된 통합 데이터베이스 가용성 솔루션에 중점을 둔 강화된 HA 제품을 제공할 것으로 사료된다.

복구 클러스터링, RAC 및 데이터베이스 복제가 사용 가능한 HA 솔루션이긴 하지만 성공적인 배포와 지속적인 가용성을 보장하려면 모두 (1) 신중한 계획, (2) 추가 관리 노력, (3) 운영 정책 및 절차, (4) 엔드-투-엔드 통합 테스트가 필요하다. 따라서 데이터베이스 애플리케이션의 요구 사항이 많아지면 서비스 수준 계약(SLA)의 요구 사항도 늘어나는 것을 알 수 있고, 무엇보다도 SLA의 요구수준을 준수하여야 한다. 최근에 대부분의 DBMS 작동 중단의 가장 큰 원인은 하드웨어 문제와 관련이 있는 것으로 나타났다.

하드웨어 문제는 디스크 고장, 네트워크 카드 고장, 운영 체제 고장, 리소스 할당 실패 등 많은 원인에서 기인한다. DBMS용 복구 클러스터링 솔루션은 데이터베이스 인스턴스를 다른 서버로 복구하여 대부분의 하드웨어 관련 문제를 극복하고, 복구 구조에 대한

일반적인 복구 클러스터링 솔루션은 (1) 하드웨어 서버, (2) 운영 체제, (3) 패치 레벨, (4) 네트워크 카드로 구성되며, SAN과 같은 공유 기억 장치에 연결되어 있습니다. 하지만 성공적인 HA 구현을 위해서는 마찬가지로 신중한 계획, 실행 및 관리가 아주 중요하다[9].

2.1 고가용성 관련 연구

(1) 오라클 RAC

오라클에서는 100% 무정지를 구축하고자 하는 IT환경의 요구에 따라 오라클 7부터 OPS(Oracle Parallel Server)를 지원하였으며 오라클 8, 오라클 8i로 발전하여 오라클 9i에서는 각 노드간 캐시의 일치성을 보장하기 위한 서버간의 통신 방식을 디스크를 이용한 방식에서 초고속 인터커넥트를 이용한 캐시 퓨전(Cache Fusion)으로 변경하면서 고가용성 구현의 완성도를 높였다. 현재 오라클 10g에서 그리드 컴퓨팅을 지원하는

더욱 발전된 RAC(real application cluster)구조를 상용화 하였다. Oracle RAC는 [그림 1]과 같이 다중 노드를 지원하는 공유 디스크 구조를 사용하여 한층 강화된 HA 솔루션을 Oracle Database에 제공 한다.

RAC의 실질적인 이점은 어느 한 노드에 오류가 발생하더라도 나머지 노드는 계속해서 연결(Connection)을 허용하는 한편 오류가 발생한 노드의 연결(Connection)을 이어 받으므로 그 애플리케이션이 계속 작동 가능하다는 것 이다. 대부분의 RAC 구성이 다소 복잡하긴 하지만 오라클은 Oracle 10g 또는 11g를 통해 설정을 보다 쉽고 간단하게 관리하고 더 많은 수의 노드를 지원할 수 있다. RAC를 사용하면 확장성과 가용성을 모두 달성할 수 있으므로 규모가 큰 업무에 또는 중요한 데이터베이스에 유용하다. RAC는 동일 데이터베이스 또는 스토리지를 여러 인스턴스에서 동시에 엑세스 할 수 있는 장점을 가지고 있으며, 시스템 확장 즉 유기적으로 인스턴스 노드의 추가가 가능하기 때문에 탁월한 로드밸런싱 및 향상된 성능을 구현 할 수 있다. 또한 RAC 구조는 모든 노드가 동일한 데이터베이스를 액세스하기 때문에 한 인스턴스에서 장애가 발생해도 데이터베이스에 대한 액세스가 손실되지 않는 장점을 가지고 있다.

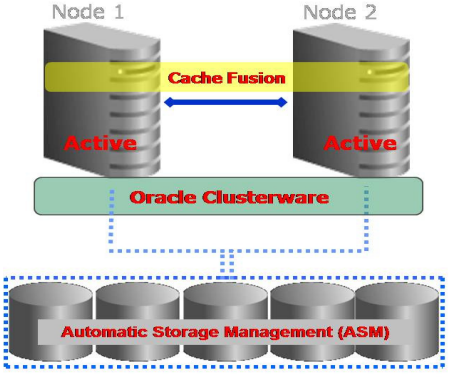

그림 1. 오라클 RAC 서버 구조

(2) Microsoft Cluster

Microsoft는 [그림 2]와 같이 비공유 구조(shared nothing architecture)를 하여 데이터 분할을 사용하는 클러스터링을 제공한다. 이 방법은 특히 데이타웨어하우징 애플리케이션의 확장성에 유용하다. 하지만 이 경우 하나의 노드가 전체 애플리케이션에 영향을 미칠 수 있으므로 HA를 위해서는 모든 노드에 데이터가 균일하게 분산되도록 해야 한다. 또한 각각의 노드의 장애시 데이터의 사용이 불가능하며, 새로운 노드가 추가시 데이터의 재분배(repartition)을 반드시 수행하여야 한다. 따라서 독립서버구조(federated)의 구성은 자동적인 장애극복을 위해서는 추가적인 시스템(witness)이 필요하며, 단하나의 미러서버(mirror server)만을 허용한다.

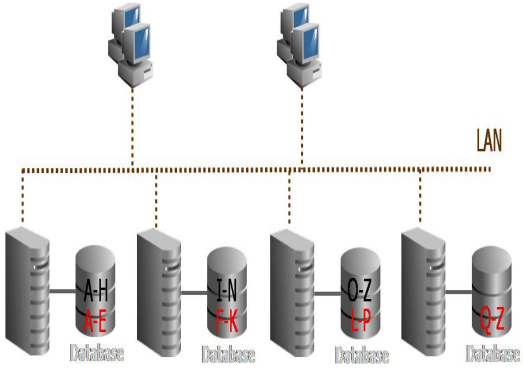

그림 2. 마이크로소프트 클러스터 서버 구조

(3) IBM pureScale

그림 IBM은 최근 애플리케이션과 데이터베이스의 고가용성 및 확장성을 지원하는 클러스터링 기술인 DB2 퓨터스케일(pure scale)을 구현한다. DB2의 퓨어스케일은 기존 메인프레인의 디스크 클러스터링 기술인 시스플랙스를 유닉스 환경에서도 이용할 수 있도록 구현한 것으로 중요하고 민감함 업무를 진행시 시스템 확장을 통해 비즈니스 요건을 충족시킬 수 있는 솔루션이다[6].

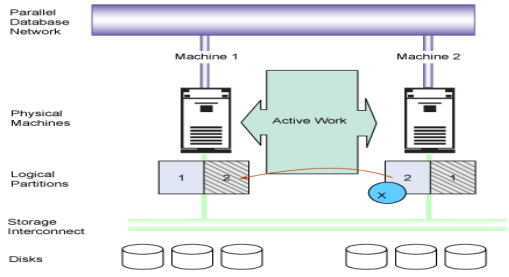

그림 3. IBM pureScale 클러스터 서버 구조

따라서 본 논문에서는 클러스터 기반 DBMS를 위한 고가용성 클러스터기반의 검증된 상용 솔루션에 대한 구현 메커니즘을 고찰하고자 한다.

3. 클러스터 서버의 구성

클러스터 가상서버(clustering virtual server)는 하나의 시스템처럼 행동하도록 하기 위한 독립적인 컴퓨터들의 집합을 의미한다. 웹서비스와 같이 대량의 트래픽을 처리해야 하는 경우, 많은 서버가 필요하지만 서버가 많은 시스템으로 구성되었을 지라도, 각 서버가 IP를 가지고 서비스를 수행한다면 하나의 서버가 고장을 일으켰을 경우, 그 서버에 대한 서비스는 중단되어 전체 서버에 대한 고가용성 및 고성능 서비스를 수행하지 못할 것이다. 이에 반해 클러스터링 기반의 서버는 사용자에게는 단일 서버 또는 단일 서버 이미지를 갖게 하여 클러스터를 하나의 서버인 것처럼 간주하게 하기 때문에 고가용성이 높은 서비스를 수행할 수 있다. 또한 클러스터링 서버는 구조의 특성상 서버의 용이한 확장이 가능하여 사용자 요청이 증가하거나 복잡한 처리 작업을 위하여 필요에 따라 여분의 시스템이 추가될 수 있다. 클러스터의 한 시스템이 에러를 일으키면, 이 시스템의 작업은 자동적으로 다른 시스템에 분산되며, 클러스터는 사용자에 대해 투명한 단일화 서비스를 제공한다. 이러한 클러스터링 기술은 저가의 PC를 이용하여 고가용성 및 고성능의 시스템을 위해 널리 활용되고 있는 실정이다.

본 연구에서는 [그림 4]와 같은 구조의 클러스터링 가상 서버를 고려한다. 클러스터 시스템은 다양한 아키텍처로 구성될 수 있으나, 일반적으로 앞단에 부하를 분산해 주는 로드 밸런스 역할의 스위치와 그에 딸린 여러 대의 실 서버들로 구성되어 있다. 클라이언트가 서비스 요청을 하면 스위치는 모든 요청을 받아들이고 적절한 스케쥴링 알고리즘에 의해 클라이언트의 서비스 요청을 실 서버에 분산 처리 한다. 물론 클라이언트는 최종적으로 연결된 실 서버와 직접적으로 연결된 것으로 판단하게 된다. 이러한 작업을 수행하는 과정에서 스위치는 각 실 서버의 상태를 파악하고 있어야 하고 이러한 정보는 실제 클러스터링 서버에 중대한 영향을 미친다. 특히, 급격한 부하증가, 시스템의 오동작, 통신 장애 등 열악한 서버 환경에서는 고장 발생률이 매우 높아지므로 고장 났을 경우 전체 시스템의 가용성에 치명적인 영향을 미치는 로드밸랜스의 고장복구 문제는 매우 중대한 사안이다. 이러한 문제를 해결하기 위해 메인 로드밸랜스와 더불어 백업 로드밸랜스를 hot-standby 상태로 두어 백업 로드밸랜서가 메인 로드밸랜스의 작업을 대신 수행하는 방안이 이용되고 있다.

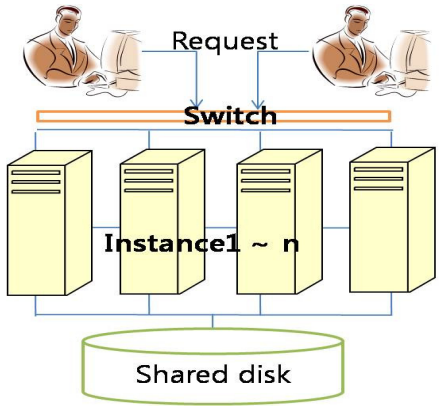

그림 4. 클러스터 서버 구조

3.1 로드밸런스의 장애극복

하나의 노드가 증폭되는 트래픽의 부담으로 인하여 고장이 발생했을 경우 이를 자동으로 감지하고 복구 즉 장애극복하는 작업을 고장포용(fault-tolerant)이라고 한다. 고장포용의 기법은 대표적으로 여분(redundancy)을 두는 공간적인 방법과 restart, rollback 과 같은 시간적으로 수행되는 방법이 있다[3]. 웹서버와 같이 고가용성을 요구하는 경우에 고장포용을 위해 Heartbeat, Fake, Checkpoint 등의 고장포용 기법들이 적용되고 있다. 또한 로드밸랜서를 위해 백업 로드밸랜서를 적용하고 있다, 로드밸랜서는 메인과 백업의 두 동일한 시스템으로 구성된다. 이는 고장포용의 공간여분으로서 메인 로드밸런서가 장애가 발생했을 경우 일정한 메커니즘에 의해 백업 로드 밸랜서가 작업을 이어받게 된다. 따라서 백업 로드 밸랜서는 메인 로드밸런서의 상태를 실시간 확인하고 있어야 한다. 이러한 메인 로드밸랜서의 장애검출을 위핸 Heartbeat이라는 방법이 적용되는

데 앞의 [그림 1]과 같이 Heartbeat은 메인과 백업 사이에 두 개의 네트워크 인터페이스 카드(Network Interface Card)와 하나의 Serial line을 통해 수행된다[4]. 즉 백업은 메인데 3가지 네트워크 통로를 통해 주기적으로 신호를 전달하고 이에 대한 응답을 요구하게 된다. 이를 위해 Fake 라는 기법이 적용되는 이는 Heartbeat을 통해 메인의 장애를 확인한 백업이 메인의 IP주소를 가져오게 된다. 따라서 클라이언트들이 메인의 IP 주소로 연결하더라도 백업이 이를 받게 되는 것이다[3].

3.2 인스턴스의 장애극복

가용성을 지원하는 클러스터 구조의 RAC은 돌발 장애 발생 시의 복구 요구 시간에 따란 인스턴스의 복구 시간을 만족할 수 있어야 한다. 즉 하나의 노드에 장애가 발생 하였을 때 얼마나 빠른 시간 내에 서비스를 재개할 수 있는가에 대한 것이다. 따라서 인스턴스 복구 시간을 고려하여 돌발 장애 시에 사용자가 모르는 사이 복구 되는 두 가지의 모드를 지원하고 있다. 이 기능은 TAF(Transportation Application Failover)와 CTF (Connection Time Failover) 기능이다. TAF는 클라이언트가 인스턴스의 한쪽 노드에 접속하여 사용하는 중에 접속한 노드에 장애가 발생한 경우 가용한 다른 노드로 접속하여 작업을 계속할 수 있도록 하는 기능이다. 즉 현재의 버전에서 조회를 하는 애플리케이션인 경우에만 TAF기능을 이용하여 애플리케이션의 수정 없이 Failover를 구현할 수 있고, CTF는 클라이언트가 데이터베이스로 접속을 시도할 때 접속하고자 하는 서버가 장애가 발생하여 접속하지 못할 경우 다른 서버로 접속할 수 있도록 하는 기능이다. 이와 같이 TAF와 CTF를 통해 돌발 장애시 서비스의 정지 없이 혹은 수초 내에 가용한 노드로 접속을 넘김으로서 시스템의 가용성을 극대화 시킬 수 있다[7].

3.3 진단

RAC에서의 진단은 즉 RAC의 아키텍처에 대한 진단으로 귀결될 수 있다. 이유는 초고속 인터커넥트를 통한 데이터의 캐시퓨전 구조를 가지고 있으므로 노드간 연결을 해주는 인터커넥트와 관련된 진단이 주가 된다. 따라서 인터커넥트의 사용량과 부하를 진단하여 문제점을 해결하는 것이 RAC 진단의 핵심이다. 인터커넥터의 진단을 위한 방식은 DBMS내의 통계정보를 이용한 분석, RAC 관련된 대기 정보를 이용한 분석, 각 종 오브젝트의 핑 정보 들을 통해 현 시스템의 적정 여부에 대한 진단이 가능하다. RAC환경에서 핑 오퍼레이션이 많을수록 시스템의 부하는 가중되어 성능의 저하를 발생하는 요인으로 꼽을 수 있다. 따라서 이러한 핑 현상이 얼마나 발생하는지 확인하는 진단이 필요하고 만약 많이 발생한다면 적절한 조치를 취해 성능이 최적화될 수 있도록 해야 한다.

III. 결 론

4. 결론 및 향후 연구

최근 인터넷 환경에서 요구되는 24*7시간 무정지 시스템 즉 고가용성 인터넷 서비스 요구를 효과적으로 처리하기 위해, 여러 대의 단일 서버를 고속의 네트워크로 연결한 클러스터 기반 DBMS의 본격적인 상용화 및 국내외적으로 활발히 진행 중이다. 더불어 최근 클러스터 기반의 발전된 모습으로 그리드컴퓨팅 기술을 통해 보다 성숙되어 가고 있다. 이는 클러스터 DBMS의 공유화 환경을 그리드 인프라 스트럭쳐로 구현 가능하며 이러한 자원 공유의 기술들은 최근 클라우드 컴퓨팅(cloud computing)기술의 근간이 되고 있다. 서버, 스토리지, 네트워크를 가상화 환경으로 만들어 필요에 따라 인프라 지원을 사용할 수 있게 서비스를 제공하는 IaaS(Infrastructure as a service), 최근 구글이나 네이버, 다음 등에서 제공하는 오픈 API들을 통해 직접 온라인 서비스를 개발에서 배포, 관리까지 가능한 플랫폼을 서비스 하는 PaaS(Platform as a Service), 기존의 ASP를 확장한 개념으로 차세대 ASP로 볼 수 있는 Saas(Software as a Service) 로 현실화 되고 있다. 더불어 서버의 가상화를 통해 인프라플랫폼 즉 데이터베이스부터 어플리케이션 부분까지의 통합 환경에 따른 기술들로 발전되어 가고 있다[8]. 따라서 본 논문을 통

해 고찰된 DBMS의 고가용성 솔루션을 보다 효율적으로 운영 관리하는 방안을 필두로 보다 진보된 다양한 그리드 기술을 이용한 메모리 기술 및 캐싱 방법 또는 서버 및 소프트웨어의 서비스 가상화, 어플리케이션, 데이터베이스의 가상화에 따른 기술들에 대한 연구가 필요하다.

참 고 문 헌

[1] 김진미, 은기원, 김학영, 지동해, "클러스터링 컴퓨팅 기술," 1999.

[2] R. Buyya, High Performance Cluster Computing Vol 1&2, Prentice Hall, 1999.

[3] 홍태희, 구본준, 김학배, "고가용성 클러스터링 가상서버의 로드밸랜서를 위한 고장극복 기법에 관란 연구," 대한전기학회 하계학술대회, pp.17-20, 2000.

[4] Gregory, F.Pfister, In Search of Cluster 2nd Edtion, Prentcd-Hall, 1998.

[5] http://dpnm.postech.ac.kr/cluster/index.htm

[6] http://www.imaso.co.kr/?doc=bbs%2Fgnuboard_pdf.php&bo_table=article&page=1&wr_id=34395&publishdate=20100101

[7] 마이크로소프트웨어 특집 2-4부, pp.231-234, 2008.

[8] http://ko.wikipedia.org/wiki/%ED%81%B4%EB% 9D%BC%EC%9A%B0%EB%93%9C_%EC%BB%B4%ED%93%A8%ED%8C%85

[9] 김영창, 장재우, 김홍연, "클러스터 기반 DBMS를 위한 고가용성 클러스터 관리기의 설계 및 구현," 정보과학회, 제12권, 제1호, pp.21-30, 2006.

[10] 최재영, 황찬석, "클러스터를 위한 소프트웨어 도구," 정보과학회지, 제18권, 제3호, pp.40-47, 2000.

저 자 소 개

이 병 엽(Byoung-Yup Lee) 정회원

- 1991년 2월 : 한국과학기술원 전산학과(공학사)
- 1993년 2월 : 한국과학기술원 전산학과(공학석사)
- 1997년 2월 : 한국과학기술원 경영정보공학(공학박사)
- 1993년 1월 ~ 2003년 2월 : 대우정보시스템 차장
- 2003년 3월 ~ 현재 : 배재대학교 전자상거래학과 부교수

<관심분야> : XML, 지능정보시스템, 데이터베이스시스템, 전자상거래학