

고가용성 데이터베이스 구축을 위한 솔루션

the Database implement of the High Availability solution

이병엽

배재대학교

Byoung-Yup Lee

Pai-Chai University

요약

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.

I. 서론

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 용이하여야 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[1,2]

II. 본론

고가용성(High availability) DBMS 구현의 80% 이상은 복구 클러스터링 구조를 기초로 하고 있으며, 이 구조는 미션 크리티컬한 비즈니스에 중요한 데이터베이스

의 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 고가용성(high availability)를 위해 복구 클러스터링과 RAC 외에 데이터베이스 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[4].

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(high availability) 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.

가용성을 지원하는 클러스터 구조의 RAC은 돌발 장애 발생 시의 복구 요구 시간에 따라 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 수 초 내에 가용한 노드로 접속을 넘김으로서 시스템의 가용성을 극대화 시킬 수 있다.

III. 결론

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 가상화(virtualization)를 통해 인프라플랫폼(Infra platform) 즉 데이터베이스부터 어플리케이션 부분까지의 통합 환경에 따른 기술들로 발전되어 가고 있다[3]. 따라서 본 논문을 통해 고찰된 DBMS의 고가용성 솔루션을 보다 효율적으로 운영 관리하는 방안을 필두로 보다 진보된 다양한 그리드 기술을 이용한 메모리 기술 및 캐싱(caching) 방법 또는 서버 및 소프트웨어의 서비스 가상화, 어플리케이션, 데이터베이스의 가상화에 따른 기술들에 대한 연구 및 고찰이 필요하다.

■ 참고 문헌 ■

- [1] Gregory, F.Pfister, In Search of Cluster 2nd Edition, Prented-Hall, 1998.
- [2] <http://dpmn.postech.ac.kr/cluster/index.htm>
- [3] <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

- [4] 클러스터 기반 DBMS를 위한 고가용성 클러스터 관리의 설계 및 구현, 정보과학회, 제 12권 제1호, 2006, p21-30