

Linux 상에서의 RAC를 이용한
데이터베이스 확장성

Oracle 백서
2003년 1월

Linux 상에서의 RAC를 이용한 데이터베이스 확장성

목차

개요	3
서론	3
Linux 상에서의 RAC의 장점	4
동적 클러스터 재구성	4
투명한 애플리케이션 복구(TAF)	4
복구 유형	4
복구 방법	5
Tnsnames.ora 예제	5
Cache Fusion	6
애플리케이션 재디자인의 불필요	6
Linux 상에서의 RAC 설치	6
저장영역	6
NAS	6
네트워크 기기	7
DAS/SAN	8
OCFS	9
RAW	9
상호 연결	10
관리 비용 절감	10
고객 성공 사례	10
구축 사례	10
결론	11
참고 자료	12

Linux 상에서의 RAC를 이용한 데이터베이스 확장성

개요

오늘날의 시장 상황에서 IT 부문은 비용을 절감하려 애쓰고 있으며, 기업들은 고효율의 독점 시스템 뿐 아니라 저렴하면서도 제 역할을 할 수 있는 비즈니스의 성장에 부합하는 확장성 있는 솔루션을 기대하고 있습니다. Linux에서 실행되는 Oracle9i RAC(Real Application Cluster)는 저렴한 비용의 상용 하드웨어에 기반하고 있으므로 이런 면에서 완벽한 솔루션이라 할 수 있으며, 또한 기존의 클러스터에 저렴한 새 서버를 추가함으로써 손쉽게 확장할 수가 있습니다.

서론

오라클의 확장성 있는 RAC 아키텍처는 Linux 운영 체제를 실행하는 인텔 기반 서버상에서 확장성 있는 데이터베이스 시스템 성능을 제공하기 위해 전통적인 클러스터화(비공유 대 공유) 방식의 최적 요소를 결합시킵니다. 이 글은 Linux에서의 Oracle RAC 확장성의 개요와 이러한 플랫폼에서의 Oracle 아키텍처에 대해 설명합니다.

증가하는 수요에 대응하기 위해서는, 사업상 중요한 데이터베이스를 실행하는 시스템을 쉽게 확장할 수 있어야만 합니다. 하드웨어 교체는 시스템 관리자와 데이터베이스 관리자가 수행해야 하는 눈에 보이지 않는 수고와 하드웨어 교체에 따른 지출 부담 때문에 좋은 선택이 아닙니다.

뿐만 아니라 수요 증가에 대응하기 위해 하드웨어를 교체함으로써 수반되는 다운타임, 보다 앞선 계획, 다른 요소들은 말할 것도 없습니다. 많은 경우에 하드웨어를 교체해 모든 프로세스가 다시 가동되기까지 1년 동안의 수요를 충분히 감당할 수 있다는 보장이 없습니다.

Oracle 9i Real Application Clusters 아키텍처를 사용하면 다운타임과 관련된 비용과 시스템 및 데이터베이스 관리자의 수고를 필요로 하는 모든 것들이 최소한으로 줄어들 수 있습니다. 뿐만 아니라 이 솔루션은 몇몇 노드가 문제를 일으킬 경우에도 확장성과 가용성을 제공합니다.

Linux 상에서의 RAC의 장점

Oracle9i Real Application Clusters를 도입하면, 증가된 수요에 대응하여 규모를 변경할 필요가 있을 경우, 새 노드를 사용하기 위해 애플리케이션을 변경하지 않고도 기존 클러스터에 노드를 추가함으로써 이를 간단히 해결할 수 있습니다. 또한 하드웨어 고장 시, 수동 조작에 의하지 않고 자동으로 로드 밸런싱되어 다른 노드로 분산됩니다.

동적 클러스터 재구성

Oracle9iR2 RAC(9.2.0.2.0)로 시작하면, 전체 클러스터의 중단 없이 Oracle 클러스터 관리자는 사용자가 새로운 노드를 추가할 수 있게 해줍니다. 이러한 새로운 방식은 시스템의 관리를 용이하게 합니다.

예를 들어 노드1이 클러스터 관리자를 실행시키고 그것이 cmcfig.ora 목록에 오직 하나의 노드 이름만을 가졌다고 가정해 봅시다. 그리고 노드2는 RAC 소프트웨어가 설치되어 있고 클러스터를 연결할 준비가 되어 있습니다.

노드2를 클러스터에 연결하려면 다음의 과정을 거쳐야 합니다.

1. 노드 1에서: 두 번째 노드 즉, 노드2를 포함시키기 위해서 cmcfig.ora를 수정합니다.
2. 노드 2에서: 노드1 및 노드2를 목록에 추가하기 위해 cmcfig.ora를 수정합니다.
3. 노드 2에서: CM을 시작합니다.
4. 노드 1에서: 노드2 상의 새로운 인스턴스에 사용될 새로운 온라인 리두 로그 스레드(redo log thread)를 추가하고 작동시킵니다.
5. 노드 2에서 전 단계에서 만들어진 새로운 리두 로그 스레드를 사용하여 클러스터 관리자와 새로운 인스턴스를 실행합니다.

클러스터 관리자와 노드1상의 Oracle 인스턴스는 계속 가동 중입니다. 노드2를 추가하기 위해서 시스템을 재시작할 필요가 없습니다.

노드2가 마스터(Master)인 노드1을 기존의 클러스터에 결합시키는 것에 주목하십시오.

투명한 애플리케이션 복구(TAF)

이 기능을 사용하면, 원하지 않은 장애를 차단할 수 있어 애플리케이션을 재시작하고 데이터베이스에 재접속할 필요 없이 사용자는 트랜잭션만 다시 시도하기만 하면 됩니다.

두번째 인스턴스에서 클라이언트 접속과 질의 복구가 허용됩니다. 이것은 복구(failover)의 영향을 최소화하기 때문에 DBA들을 위한 매우 유용한 기능이며, 이 기능을 통해 연결 로드 밸런싱이 가능하고 사용자에게 영향을 주지 않으면서 노드 유지관리를 할 수 있습니다.

복구 유형

세 가지 유형의 투명한 장애 복구 옵션을 사용할 수 있습니다. 클라이언트의 복구 기능은 SqlNet의 연결 문자열에서 "TYPE"이라는 키워드로 정의됩니다.

다음과 같은 선택 사항이 있습니다.

SESSION - 만약 사용자가 연결에 실패한다면, 새로운 세션은 남아있는 백업 인스턴스에서 자동으로 생성됩니다. 사용자 질의 복구를 위해 어떤 시도도 할 필요가 없습니다.

참고 - 활성 트랜잭션은 복구되지 않습니다.

SELECT - 만약 사용자가 연결에 실패한다면, open-cursor가 있는 사용자는 실패 후 계속 open-cursor 명령을 가져옵니다. 사용자들의 상호작용을 위한 어떤 노력도 필요 없이 질의는 자동적으로 계속됩니다. 또한 내부적으로, SELECT 명령은 남아 있는 인스턴스에서 재실행됩니다. 이 모드에서는 일반 사용자가 작업을 선택하는 동안 오버헤드가 발생합니다.

NONE - 이것은 복구 기능이 사용되지 않는 곳에서의 기본적인 유형입니다. 설정 유형 = none은 복구가 필요하지 않음을 명시적으로 지정합니다.

복구 방법

복구 방법은 백업 인스턴스에서 복구 연결 설정의 시기를 결정합니다. METHOD 키워드는 SqlNet 연결 문자열에서 BASIC 또는 PRECONNECT 실행 옵션을 구성하는 데 사용됩니다.

BASIC - 복구 동안 연결이 설정됩니다. 이 옵션은 복구 시간까지 백업 인스턴스의 시스템 리소스를 거의 소모하지 않습니다.

PRECONNECT - 연결은 1차 인스턴스에서 설정된 연결과 동시에 백업 인스턴스로 설정됩니다. 이 방법은 백업 서버에서 정지된 연결 프로세스의 일정한 오버헤드 뿐만 아니라 계속되는 예비 연결의 핑(ping)과 지속적인 네트워크 트래픽을 요구합니다. 그러나, 만약 중요한 시스템에 장애가 발생한다면 이 옵션은 필수 소요 시간을 줄여 줍니다. 만일 복구 중 예비 연결을 이용할 수 없다면 SqlNet 연결 문자열이 새로운 연결을 시도하기 위해 사용됩니다.

Tnsnames.ora 예제

전형적인 tnsnames.ora 파일은 아래 예와 비슷한 TAF를 사용하도록 구성됩니다:

```
SID1 = (DESCRIPTION=
        (ADDRESS=(PROTOCOL=TCP)(HOST=host1)(PORT=1521))
        (CONNECT_DATA=(SID=sid1)(SERVER=DEDICATED)
        (FAILOVER_MODE=(TYPE=select)(METHOD=preconnect)
        (BACKUP=sid2))
        )
    )
```

```
SID2 = (DESCRIPTION=
        (ADDRESS=(PROTOCOL=TCP)(HOST=host2)(PORT=1521))
        (CONNECT_DATA=(SID=sid2)(SERVER=DEDICATED)
        (FAILOVER_MODE=(TYPE=select)(METHOD=preconnect)
        (BACKUP=sid1))
        )
    )
```

Cache Fusion

인스턴스가 블록을 업데이트할 때마다 클러스터 내에 있는 다른 인스턴스가 동일 블록을 업데이트할 수 없도록 언제나 잠금(Lock)을 걸어야 합니다. 이 문제를 해결하기 위해 Oracle은 디스크로부터 데이터를 읽기 전에 특정한 블록 상태를 만드는 데이터 블록 핑 메커니즘을 행합니다.

캐시 퓨전은 오라클 데이터베이스 노드에서 데이터 블록 읽기/읽기, 읽기/쓰기, 쓰기/쓰기 충돌 문제를 고성능 상호 연결 네트워크를 통해 이전 릴리스에서 사용된 더 느린 물리적 디스크 작업을 우회하여 해결합니다. 클러스터에 노드를 추가할 때 Oracle 9i Real Application Clusters의 캐시 퓨전 기능을 사용하여 선형에 가까운 데이터베이스 성능 확장을 이룰 수 있습니다. 오라클은 더욱 뛰어난 데이터베이스 용량 계획을 실현하며 자본 투자를 보호합니다.

애플리케이션 재디자인의 불필요

데이터 액세스 패턴이 데이터 블록 핑을 감소 또는 어렵게 하더라도 애플리케이션은 더 이상 분할될 필요가 없습니다. 단일 노드의 Oracle9i 서버에서 확장성 있는 애플리케이션은 멀티 노드의 Oracle9i Real Application Clusters상에서도 확장성이 있습니다.

Linux 상에 RAC 설치

공식적으로 Linux 기반 Oracle9i Real Application Clusters는 RedHat Linux Advanced Server 2.1이 지원하고 있습니다. 오라클은 이 솔루션에 대해 모든 필수적인 지원을 제공합니다. 이는 운영 체제 또는 오라클 문제를 해결하기 위해 사용자는 단지 요청만 하면 된다는 것을 의미합니다.

RAC 구성 요소 중 일부는 각각의 RAC 구현이 얼마나 확장성이 있는가를 설명하는 중요한 것들입니다. Oracle9i Real Application Clusters는 대부분의 경우 확장성이 있지만, 때로는 네트워크 또는 하드웨어 아키텍처와 같은 물리적 및 구조적 제한을 극복하지 못하는 경우가 있습니다. 그러므로 수요 증가와 구성 요소에 대한 신중한 분석이 매우 중요합니다.

저장 영역

Oracle9i Real Application Clusters는 공유 SCSI, NAS 및 SAN 저장 영역 시스템을 지원합니다. 모든 저장 시스템에서 RAC가 실행 가능하다고 보증할 수는 없으나 올바른 실행을 위한 아키텍처 및 표준들을 준수합니다.

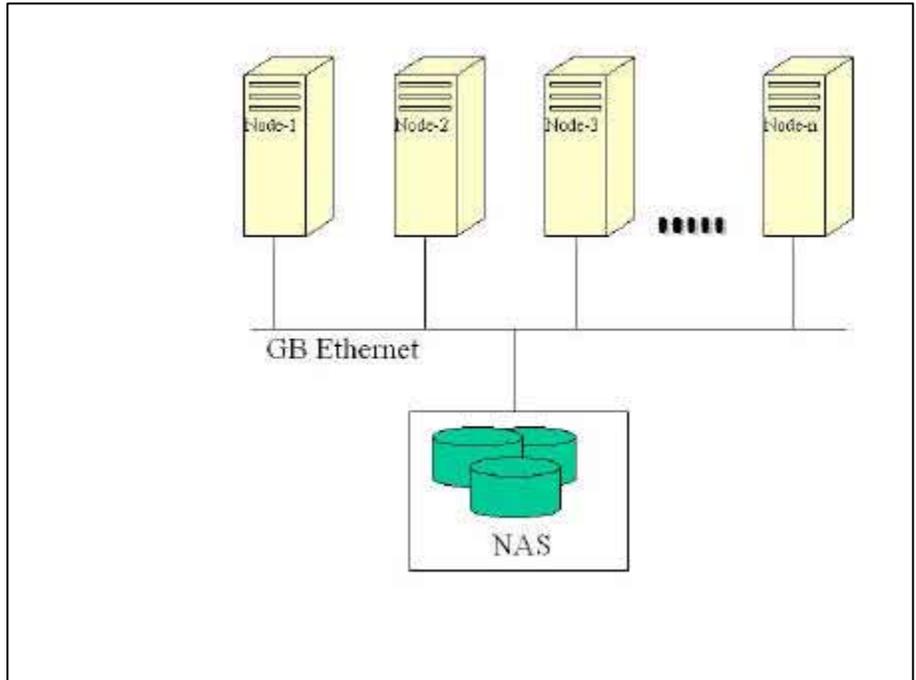
NAS

일반적으로 파일 서버로 사용되는 NAS 아키텍처는 NFS를 사용하는 네트워크 서버로부터 사용자가 특정한 양을 마운트할 수 있게 합니다. Oracle은 데이터베이스를 생성하기 위해서 NAS 저장 영역 시스템을 사용할 수 있으나, 그러기 위해서는 데이터 무결성을 보장하기 위한 특별한 조건이 필요한데, NFS 볼륨은 UDP가 아닌 TCP/IP를 이용하여 마운트되어야 합니다. UDP를 이용하는 것은 패키지 전달을 검증할 수 없기 때문에 가능하지 않지만, TCP/IP를 이용하면 가능합니다. 이와 같은 경우에 NAS 서버는 TCP/IP를 지원해야 합니다.

NAS를 이용한 Oracle9i Real Application Clusters의 구현에 있어서 실행상의 문제로 인해 Oracle은 저장 영역에 액세스하는 데 사용되는 NIC(Network Interface Card)가 고성능 네트워크 장치(Gb Network)일 것을 권장합니다. 또한, 하나는 온라인 로그 파일용으로 나머지 하나는 데이터 파일용으로 사용되도록, 즉 적어도 하나 이상의 인터페이스가 사용될 것을 권장합니다.

볼륨이 보통의 파일 시스템으로 제공되기 때문에 NAS를 사용한 구현은 상대적으로 손쉽고 그 관리 또한 매우 쉽고 단순합니다.

이와 같은 예제는 공용 및 개인용 네트워크(상호 연결된)에서는 볼 수 없고, 또한 각각의 노드가 3 개 이상의 인터페이스를 갖는 것을 전제로 합니다.



네트워크 기기

NetApp 필터를 구성하는 과정에서 NFS를 구성할 때 NFS Enabled, NFS Over TCP 및 NFS Version 3을 예로 설정했는지 확인하십시오. 또한 진행되기 적용 단추를 클릭해야 합니다. Oracle 9i Real Application Clusters에서 사용되는 NFS 볼륨을 전송할 때 서버가 마운트하고 볼륨에 기록할 수 있도록 올바른 권한을 승인해야 합니다.

하나의 예로서 특정 네트워크 기기를 사용합니다. 다른 NAS 업체의 경우에도 유사하게 진행됩니다. 오라클은 특정 업체를 추천하지 않고 다만 표준을 제시할 뿐입니다.

문제를 피하기 위해서는 NFS 볼륨을 특정 매개 변수 집합을 사용해 서버에 마운트해야 합니다. 볼륨을 항상 정확하게 마운트하기 위한 가장 쉬운 방법은 부팅할 때 행들을 자동으로 마운트하기 위해서 /etc/fstab 파일에 행(line)을 추가하는 것입니다. 예를 들어 /vol/rac00 볼륨을 내보내는 NetApp filer를 가지고 있고 또한 서버에서 마운트 위치 /mnt/netapp0 를 사용하고 있다면 행은 다음과 같아야 합니다.

```
filer:/vol/rac00 /mnt/netapp0 nfs
rw,bg,hard,intr,vers=3,tcp,rsize=32768,wsiz=32768,noac
```

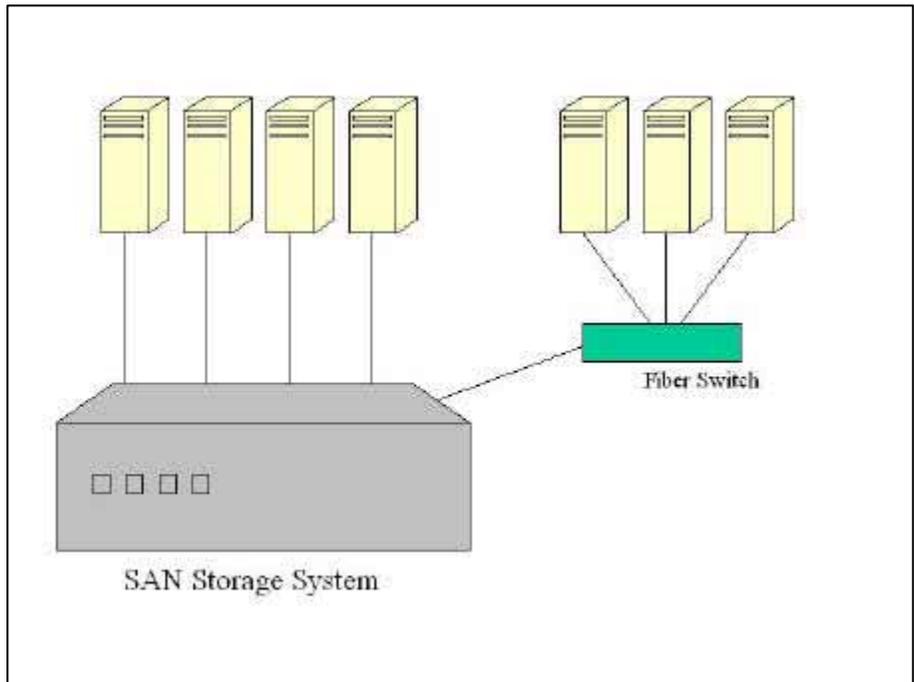
NetApp Filer를 설정하는 방법에 대해 질문이 있으면 네트워크 기기 기술 지원 센터(Network Appliance Technical Support.)로 문의하십시오.

DAS/SAN

DAS(Direct Attached Storage) 또는 SAN(Storage Area Network)은 보통 빠른 솔루션이지만 가격은 매우 비싼 편입니다. 이들은 네트워크 트래픽에서 자유롭고 우수한 성능으로 높은 가용성을 제공하며, 보통 80Mb/s이상의 전송율의 매우 빠른 액세스 속도를 가진 HBA를 이용하여 노드에 접속됩니다.

이들은 운영 체제에 의해서 기본적으로 SCSI 장치로 인식되기 때문에 모든 노드는 물리적으로 디스크를 공유하고 동일한 권한을 갖고 있습니다. 두 가지 관리 작업은 동시에 같은 장치의 다른 노드에서 수행되기 때문에 항상 관리가 필요합니다. 조정과 구성의 합당한 수준 설정은 심각한 문제를 피하기 위한 필수 사항입니다.

이 예제는 4개의 노드가 SAN 저장 영역과 직접 연결되고 3개의 노드가 스위치를 사용하고 있음을 보여주고 있습니다. 오라클은 두 솔루션 모두 지원하고 있습니다.



OCFS

OCFS는 보통의 파일 시스템과 동일한 기능에 더하여 클러스터 내에 있는 노드 사이에서 파일과 디렉토리의 일관적인 뷰를 가능하게 하는 메커니즘으로 클러스터 관리 역량을 배가 시킵니다. Unix 파일 시스템에서 활용 가능한 파일 시스템 모든 능력은 OCFS에서도 마찬가지입니다. 이 경우에는 각각의 데이터베이스의 백업/복구 과정을 세부적으로 분할할 필요가 없습니다. OCFS는 파일 시스템의 단순성과 관리 용이성 측면에서 원시 장치를 이용한 것에 가까운 성능을 나타냅니다. 데이터 파일은 보통의 파일로서 생성되고, Oracle은 Direct I/O를 이용하여 그에 액세스하는데 Direct I/O는 운영 체제 캐시 메커니즘을 무시하고 보다 나은 성능을 제공합니다.

OCFS는 쉽게 구성할 수 있으며 성능을 향상하기 위해 어떠한 튜닝도 필요하지 않습니다. 다만, 설치 후 초기 구성은 필수 사항입니다. 이 초기 구성은 `Ocfstool` 명령을 호출하고 CTRL-G 키를 누름으로써 간단히 끝낼 수 있습니다. 이 과정은 클러스터 내의 모든 노드에서 행해져야 하고 그러면 `/etc/ocfs.conf`라는 구성 파일이 생성됩니다. `ocfs.conf`의 표준은 다음과 같습니다:

```
#
# ocfs config
# Ensure this file exists in /etc#
    node_name = serena
    node_number =
    ip_address = 130.35.149.132
    ip_port = 7000
    guid = 390497153C4039EC1E2100D0B7E4D526
```

OCFS의 설치 및 설정에 관한 자세한 내용은 “Oracle Cluster File System Installation Notes Release 1.0 for Red Hat Linux Advanced Server 2.1 - PN B10499-01”을 참조하십시오.

RAW

오라클은 RAC를 위해 OCFS를 사용할 것을 권장하지만 원시 장치 역시 저장 영역을 위한 또 다른 옵션입니다. 테이블스페이스 데이터파일과 물리적인 원시 파티션 간의 매핑은 손상 방지를 위해 클러스터 내 노드 사이의 완벽한 조화를 요구하기 때문에 원시 장치의 구축은 배포를 위해 좀 더 세심한 계획이 요구됩니다. 또한 Linux에서는 원시 장치를 각각의 파티션과 묶는 것이 필수적입니다. 이것이 바로 Linux상의 물리적인 파티션이 블록 장치인 이유이고, Oracle이 문자 장치에 기록하는 이유입니다. 이 바인딩 작업은 영구적인 작업 상태가 아닌 데이터베이스가 시작하기 전 부팅 시마다 이루어져야 합니다.

Oracle9i Real Application Clusters의 전체적인 성능과 확장성에 영향을 미칠 수 있기 때문에 GB 상호 연결 사용이 적극 권장됩니다.

상호 연결

오라클은 산재한 모든 잠금 관리와 캐시 퓨전 작업을 수행하기 위해서 상호 연결 네트워크를 이용합니다. 이런 이유 때문에 Oracle은 RAC에서 모든 노드 연결 전용의 고성능 상호 연결 네트워크(Gb Network)를 추천합니다. 고성능 상호 연결 네트워크를 사용하면 클러스터 내의 새로운 노드의 충격을 완화할 수 있으며 성능 저하 없이 더 높은 수준의 확장성을 얻을 수 있습니다.

관리 비용 절감

Oracle9i Real Application Clusters는 클러스터 내의 어떠한 노드에서건 데이터베이스의 전체적인 뷰를 얻을 수 있고, 노드를 추가하는 데 관리상의 노력이 요구되지 않기 때문에 손쉽게 관리할 수 있습니다. 또한 새로운 리소스를 관리하기 위한 직원을 따로 채용하지 않고도 확장성을 증대시킬 수 있습니다.

이와 같은 e-business 시스템의 요구 사항과 업무 부담이 빈번하게 변화함에 따라 Oracle9i Real Application Clusters에 의한 관리 비용 절감과 유연성으로 인하여 그와 같은 시스템을 위해서는 대단히 관심을 끌고 있으며 또한 저 비용과 확장성 때문에 Oracle9i Real Application Clusters는 Linux 상에서 더욱 매력적인 솔루션입니다. 많은 고객들은 실용적이고 확장성이 있으며 믿을 수 있는 솔루션으로서 Linux 상에 Oracle9i Real Application Clusters를 이미 선택했거나 바꿀 것을 심각하게 고려하고 있습니다.

고객 성공 사례

구축 사례

이 사례는 제조 분야의 많은 오라클 고객의 배포 아키텍처를 보여주고 있습니다. 그들은 Dell Servers와 Red Hat Linux Advanced Server 상에서 Oracle9i Real Application Clusters를 이용하여 Oracle E-Business Suite (11i)를 성공적으로 구축하였습니다. 새로운 솔루션의 구축 후 중요한 일괄 작업 및 온라인 응답 시간을 50% 이상 줄일 수 있었습니다. Dell/RedHat/Oracle9i Real Application Clusters의 도입으로 오라클 데이터베이스와 애플리케이션의 전반적인 성능 또한 향상되었습니다.

현재 구성은 다음과 같습니다.

Intel 컴퓨터상에서 실행하는 중간 계층 서버를 가진 다섯 개의 APPS

iAS와Form Server를 실행하는 네 개의 노드(2X1.2GHz processors 와 1, 2, 4GB RAM의 혼합)

Concurrent Manager and Report Server를 실행하는 한 개의 노드 (8X550MHz Processors 와 8GB RAM)

Oracle9i Application Server Portal/Discoverer 4i를 실행하는 한 개의 서버(2x1.2GHz processors 와 4Gb RAM)

Oracle9i Real Application Clusters를 실행하는 네 개의 노드(4X900MHz processors 와 8GB RAM)

Oracle9i Real Application Cluster가 사용하는 공유 저장 영역은 RAID 10를 가진 DELL 4700 입니다.

부하는 다음과 같이 4개의 노드에 할당됩니다.

Node1

- 셀프 서비스
- BOM
- AOL
- Misc. (DBA, 개발자, etc).

Node2

- 재무
- 재고

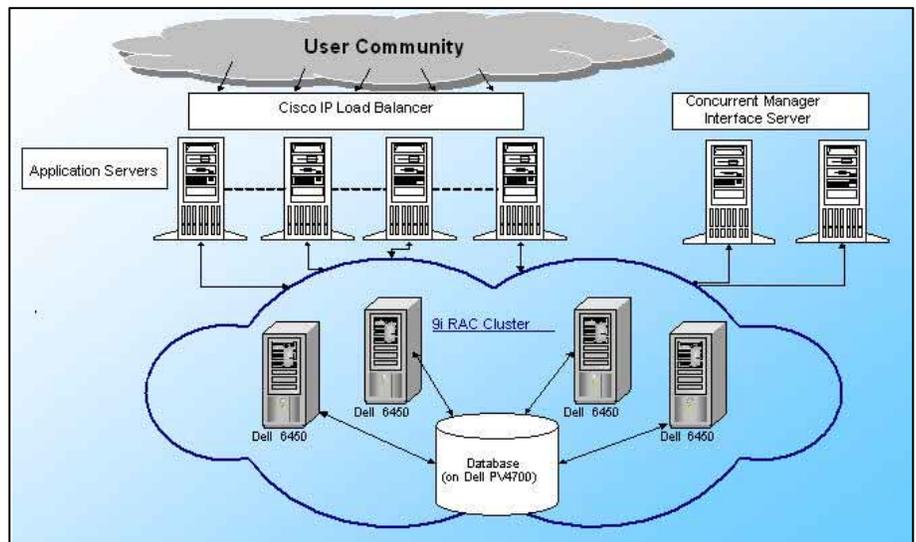
Node3

- HR
- AP
- Discover 4i
- ADI
- MS-Access

Node4

- 동시 관리자
- 구매

하단의 다이어그램은 아키텍처를 설명합니다 :



결론

Linux 상의 Oracle9i Real Application Clusters는 독점 시스템에 비해 저렴하기 때문에 대단히 매력적인 솔루션일 뿐 아니라 높은 가용성을 가진 확장 가능성이 매우 좋은 솔루션입니다.

참고 자료

- Oracle9i Real Application Clusters - 데이터시트
- Oracle9i Real Application Clusters Manuals - Oracle 문서
- Oracle Cluster File System Installation - Oracle 문서
- Oracle9i Real Application Clusters Cache Fusion Delivers Scalability
- 백서
- Oracle Cluster File System Installation Notes Release 1.0 for Red Hat Linux Advanced Server 2.1 - PN B10499-01



한국오라클(주)

서울특별시 강남구 삼성동 144-17
삼화빌딩
대표전화 : 2194-8000
FAX : 2194-8001

한국오라클교육센터

서울특별시 영등포구 여의도동 28-1
전경련회관 5층, 7층
대표전화 : 3779-4242~4
FAX : 3779-4100~1

대전사무소

대전광역시 서구 둔산동 929번지
대전둔산사학연금회관 18층
대표전화 : (042)483-4131~2
FAX : (042)483-4133

대구사무소

대구광역시 동구 신천동 111번지
영남타워빌딩 9층
대표전화 : (053)741-4513~4
FAX : (053)741-4515

부산사무소

부산광역시 동구 초량동 1211~7
정암빌딩 8층
대표전화 : (051)465-9996
FAX : (051)465-9958

울산사무소

울산광역시 남구 달동 1319-15번지
정우빌딩 3층
대표전화 : (052)267-4262
FAX : (052)267-4267

광주사무소

광주광역시 서구 양동 60-37
금호생명빌딩 8층
대표전화 : (062)350-0131
FAX : (062)350-0130

고객에게 완전하고 효과적인
정보관리 솔루션을 제공하기 위하여
오라클사는 전 세계 145개국에서
제품, 기술지원, 교육 및
컨설팅 서비스를
제공하고 있습니다.

<http://www.oracle.com/>
<http://www.oracle.com/kr>