

Microsoft®
SQL Server™ 2005

**고가용성을 위한 데이터베이스의 선택:
SQL Server 및 Oracle에 대한 분석**

요약

SQL Server 2005는 Oracle 10g보다 낮은 비용으로 엔터프라이즈급 고가용성 기능을 제공합니다. 본 문서에서는 계획된/예상치 못한 시스템 정지 시간을 해결하는 SQL Server 2005와 Oracle 10g의 고가용성 기능을 비교하여 설명합니다. 또한 고가용성의 주요 장애 요인을 해결하는 방법도 제시하고 있습니다.

본 문서는 예비 문서이며 여기에서 설명된 소프트웨어의 최종 상용 버전이 출시되기 전에 상당 부분 변경될 수 있습니다. 이 문서에 포함된 정보는 문서 발행 시에 논의된 문제들에 대한 Microsoft Corporation의 당시 관점을 나타냅니다. Microsoft는 변화하는 시장 상황에 부응해야 하므로 이를 Microsoft 측의 공약으로 해석해서는 안되며 발행일 이후 소개된 어떠한 정보에 대해서도 Microsoft는 그 정확성을 보증하지 않습니다.

이 문서는 오직 정보를 제공하기 위한 것입니다. Microsoft는 이 설명서에서 어떠한 명시적이거나 묵시적인 보증도 하지 않습니다. 해당 저작권법을 준수하는 것은 사용자의 책임입니다. 저작권에서의 권리와는 별도로, 이 설명서의 어떠한 부분도 Microsoft의 명시적인 서면 승인 없이는 어떠한 형식이나 수단(전기적, 기계적, 복사기에 의한 복사, 디스크 복사 또는 다른 방법) 또는 목적으로도 복제되거나, 검색 시스템에 저장 또는 도입되거나, 전송될 수 없습니다.

Microsoft가 이 설명서 본안에 관련된 특허권, 상표권, 저작권 또는 기타 지적 재산권 등을 보유할 수도 있습니다. 서면 사용권 계약에 따라 Microsoft로부터 귀하에게 명시적으로 제공된 권리 이외에, 이 설명서의 제공은 귀하에게 이러한 특허권, 상표권, 저작권 또는 기타 지적 재산권 등에 대한 어떠한 사용권도 허여하지 않습니다.

특별한 언급이 없는 한, 용례에 사용된 회사, 기관, 제품, 도메인 이름, 전자 메일 주소, 로고, 사람, 장소, 이벤트 등은 실제 데이터가 아닙니다. 어떠한 실제 회사, 기관, 제품, 도메인 이름, 전자 메일 주소, 로고, 사람, 장소 또는 이벤트와도 연관시킬 의도가 없으며 그렇게 유추해서도 안됩니다.

© 2005 Microsoft Corporation. 전권 보유.

Microsoft와 ActiveX는 미국, 대한민국 및/또는 기타 국가에서 Microsoft의 등록 상표 또는 상표입니다. 여기에 인용된 실제 회사와 제품 이름은 해당 소유자의 상표일 수 있습니다.

Contents

전체 개요	3
들어가는 글	3
고가용성 및 99.999%(Five 9)에 대한 정의	4
가용성에 영향을 미치는 요인	5
계획된 시스템 정지 시간	6
서버 유지 관리를 위한 데이터베이스 고가용성	6
하드웨어 업그레이드 및 유지 관리	6
데이터베이스 유지 관리	6
예기치 못한 시스템 정지 시간	8
데이터베이스 복구를 위한 데이터베이스 고가용성 솔루션	8
SQL Server 2005-백업 및 트랜잭션 시점 시간(Point-in-Time) 복구	8
SQL Server 2005-로그 전달과 지연	9
SQL Server 2005-파일 그룹 복원	9
SQL Server 2005-빠른 복구	9
SQL Server 2005-데이터베이스 스냅샷	10
Oracle 10g-RMAN(Recovery Manager)	10
Oracle 10g-플래시백(Flashback)	11
서버 오류를 막는 데이터베이스 고가용성 솔루션	11
SQL Server 2005 N-Way 클러스터링	12
Database Mirroring	15
SQL Server 2005 로그 전달	17
SQL Server 2005-복제	17
Oracle 10g-Real Application Clusters(RAC)	18
Oracle 10g Data Guard	19
결론	20
부록 A-고가용성의 장애 요소	20
인적 장애 요소	20
프로세스 장애 요소	21
기술 장애 요소	22
인프라 오류	22
사이트 장애	23



전체 개요

Microsoft® SQL Server™ 2005는 Oracle 10g보다 저렴한 비용으로 고가용성에 대한 고객의 요구 사항을 충족할 수 있는 제품인 것으로 입증되고 있습니다. SQL Server 2005는 추가 비용 없이 SQL Server 2005 Standard Edition과 SQL Server 2005 Enterprise Edition에서 지원되는 MSCS(Microsoft Clustering Services), Database Mirroring, 데이터베이스 스냅샷, 로그 전달, 복제 기능을 모두 포함하고 있습니다.

Oracle의 RAC(Real Application Clusters)는 고가용성 구성으로 구현할 수 있지만 RAC는 Oracle 10g Standard Edition에만 포함되어 있으며 Oracle 10g Enterprise Edition에는 포함되어 있지 않습니다. RAC는 자동 장애 조치를 지원할 수는 있지만 SQL Server 2005 Database Mirroring에서와 같은 5초 이하의 장애 조치 속도에는 미치지 못합니다. 이와 마찬가지로 Oracle 10g의 플래시백 및 데이터 가드 기능도 Oracle 10g Standard Edition에서 지원되지 않습니다. 이러한 기능을 이용하기 위해 고객들은 훨씬 고가 제품인 Oracle 10g Enterprise Edition을 구입해야 합니다.

또한 SQL Server 2005 Enterprise Edition은 여러 서버 간의 데이터 파티셔닝을 통해 가용성을 향상시킬 수 있습니다. Oracle 10g에 이 기능을 추가하려면 Oracle Partitioning 제품(옵션)을 구입해야 합니다. 고가용성을 실현하는 데 반드시 많은 비용을 투자해야 하는 것은 아니며 SQL Server 2005는 Oracle 10g보다 훨씬 저렴한 가격으로 고가용성에 대한 고객의 요구 사항을 충족합니다.

들어가는 글

기업 내 컴퓨팅 자원의 지속적인 가용성을 보장하는 것은 오늘날 DBA(Database Administrator)의 가장 중요한 목표입니다. 사용자와 고객이 업무를 수행하는 데 이용하는 자원을 액세스할 수 없다면 세상의 모든 컴퓨팅 성과와 보안/방어 응용 프로그램 코딩은 무의미하게 될 것입니다. 응용 프로그램을 지원하는 데이터베이스와 서버를 사용할 수 없다면 기업은 금전적인 손실은 물론 명성과 신용 측면에서도 타격을 입을 수 있습니다.

기업의 특성과 응용 프로그램의 유형에 따라 시스템 정지 시간으로 인해 엄청난 대가를 치를 수 있습니다. Forrester Research 가 실시한 연구에 따르면 온라인 브로커 업체인 eTrade의 경우 1시간의 서비스 중단 사태로 인해 8백만 달러의 손실을 입은 것으로 나타났습니다. 또한, 델(DELL)은 서비스 중단 10시간 만에 8천3백만 달러를 잃었으며 인텔은 33시간 만에 2억7천5백만 달러의 손실이 발생했습니다. 이들 수치는 수익 손실만을 나타내는 것으로 고객 만족도 저하나 기업 평판 훼손 등과 같은 무형적인 비용은 포함되지 않았습니다.

미션 크리티컬 응용 프로그램의 경우, 사용자가 응용 프로그램을 필요로 하는 시간 동안 해당 응용 프로그램을 지원하는 데이터베이스와 서버를 사용할 수 있어야 합니다. 오늘날 기업에서 필요로 하는 가용성 수준을 실현하기 위한 요건은 기존 백업 및 복구 기술이 제공하는 단순한 데이터 보호 기능 이상으로 확장됩니다. 오늘날 기업 응용 프로그램은 그 어느 때보다도 우수한 가용성 시간을 필요로 하며, 많은 기업들이 하루 24시간, 주7일, 일년 365일 가용성을 요구합니다. 비즈니스 연속성을 위한 고가용성 환경을 구축하는 것은 기업 내 매우 다양한 영역을 아우르기 때문에 매우 복잡할 뿐만 아니라 기술 과제와 역량은 물론, 순수 기술 영역을 넘어 운영 부문으로 확장되는 인적/조직적 요인 등을 비롯하여 매우 많은 요인의 영향을 받습니다. 본 백서에서는 SQL Server 2005와 Oracle 10g가 제공하는 고가용성 기술을 비교하여 설명합니다. 고객들은 이러한 정보를 사용하여 두 시스템의 각기 다른 기능을 평가하고 각자의 요구에 가장 부합하는 데이터베이스를 선택할 수 있습니다.

고가용성 및 99.999%(Five 9)에 대한 정의

가용성은 시스템이나 자원을 사용할 수 있는 시간으로 정의할 수 있습니다. 고가용성에 대한 정의는 일반적으로 절대 가용성에서 차지하는 비율에 따라 측정됩니다. 여기에서 100%는 자원을 항상 사용할 수 있으며 시스템 정지 시간이 없다는 것을 의미합니다. 그러나 100% 가용성을 실현하는 것은 사실 매우 어렵습니다. 그 대신, 달성할 수 있는 최고 수준의 가용성을 가장 근접하게 측정한 실질적인 수치가 파이브 나인(Five 9), 즉 99.999%입니다. 수학적으로 가용성을 다음과 같이 정의할 수 있습니다.

가용성 비율 = (총 경과 시간 - 시스템 정지 시간 합계) / 총 경과 시간

시스템 가용성 비율은 총 경과 시간에서 시스템을 사용할 수 없는 총 시간을 뺀 값을 총 경과 시간으로 나눈 것입니다. 매년 시스템을 사용할 수 있는 총 가동 시간은 8,760시간입니다(하루 24시간 x 일년 365일). 총 8,760시간의 가동 시간은 해당 연도에 시스템을 사용할 수 있는 가동 시간의 100%가 됩니다. 그러나, 정기적인 시스템 유지 관리를 수행하거나 예기치 못한 여타 이벤트로 인해 전체 8,760시간의 가동 시간을 실현한다는 것은 일반적으로 불가능합니다. 예를 들어, IT 직원이 하드웨어 업그레이드를 수행하고 시스템 패치를 적용하거나 다른 일상적인 유지 관리 작업을 수행할 수 있도록 시스템이 매월 유지 관리를 위해 하루(8시간) 동안의 예정된 시스템 정지 시간을 갖는 것이 일반적입니다. 이 시나리오에서 시스템 가용성 비율은 다음과 같은 식으로 나타낼 수 있습니다.

$$98.9 = ((8760 - (8 \times 12)) / 8760)$$

즉, 한 달 동안 시스템 정지 시간이 하루 발생하면 전체 가용성 수치는 98.9%가 됩니다. 다음 공식은 시스템 가용성에서 9의 개수를 계산하는 방법을 설명합니다. 99.9%는 쓰리 나인(three 9), 99.999%는 포 나인(four 9)과 같이 9의 개수가 표현됩니다.

한달 동안 시스템 정지 시간이 하루 발생하면 전체 가용성은 98.9%(파이브 나인 중 9 한 개의 비율)입니다. 한달 간 하루는 그리 크지 않은 정지 시간이며 많은 기업들에게 이 정도 수준의 가용성은 적당합니다. 그러나 이 정도의 가용성으로는 충분하지 않은 환경들이 많습니다. 아래 표는 가용성 수준의 연속적인 증가에 따른 시스템 정지 시간을 설명하고 있습니다.

9의 개수	가용성 비율	연간 시스템 정지 시간
1	98.9%	3일, 18시간, 20분
2	99.0%	3일, 15시간, 36분
3	99.9%	8시간, 46분
4	99.99%	53분
5	99.999%	5분

하루 15분, 즉 연간 92시간(3일, 18시간, 20분)의 시스템 정지 시간으로 90% 이상의 가용성을 실현할 수 있습니다. 그러나 연속되는 보다 높은 수준의 가용성을 실현하는 것은 점차 엄격해집니다. 가장 높은 가용성 수준(99.999%의 가용성)에서는 연간 시스템 정지 시간이 .09시간, 즉 약 5분 미만이어야 합니다. 현재의 데이터베이스 플랫폼으로 보다 높은 수준의 정지 시간을 달성하는 것이 가능하지만 기술만으로는 이를 실현할 수 없습니다. 즉, 사람, 프로세스 및 기술 요소들을 유기적으로 활용해야 이러한 최고 수준의 가용성을 실현할 수 있습니다.



사용자에 대한 시스템 가용성의 관점에서 시스템 정지 시간을 추적해야 한다는 점에 주목해야 합니다. 일부 공급업체들이 서버 가용성을 기준으로 가용성을 주장하고 있지만 서버 수준의 가용성 추적에 실제 사용자의 가용성 수준이 고려되지 않는 경우가 많습니다. 서버가 가동 중이지만 다른 일부 요인에 의해 최종 사용자의 시스템 액세스가 저지되는 경우, 해당 시스템은 사용할 수 없는 것으로 간주되어야 합니다.

가용성에 영향을 미치는 요인

1년 365일 하루 24시간 항상 사용할 수 있도록 해야 한다는 정보에 대한 요구는 여러 요인들의 영향을 받습니다. 가용성이 뛰어난 운영 환경 구축에 장애가 될 수 있는 주된 요인은 다음과 같습니다.

- 사람
- 프로세스
- 기술

지속적인 정보 가용성을 달성하는 데 영향을 미치는 것은 기술만이 아닙니다. 본사와 원격지에 있는 직원 모두를 포함한 기업 직원도 중요합니다. 기업이 최고 수준의 가동 시간을 유지하기 위해서는 기업 활동을 원활하게 운영할 수 있는 최고의 인재를 채용하는 것이 중요합니다. 여기에 효율적인 운영 절차와 훌륭한 정책을 실행하는 프로세스가 결합된다면 직원들은 최상의 전문성을 발휘하며 업무를 수행할 수 있도록 보장하게 됩니다.

가용성이 매우 높은 환경을 구축하는 데 있어서 기술의 역할은 여러 수준으로 나눌 수 있습니다. 시스템의 관점에서 기술 솔루션은 일상적인 유지 관리는 물론 사이트 장애, 서버 장애, 데이터베이스 손상 등과 같이 발생 가능한 다양한 장애 유형을 처리합니다. 비즈니스의 관점에서 기술은 기업 환경 내 사람, 정책 및 프로세스에도 영향을 미칩니다. 예를 들어, 기업이 선택하는 하드웨어 및 소프트웨어 솔루션에 따라 직원이 필요로 하는 직무 역량과 기업이 해당 기술을 관리하기 위해 규정해야 하는 특정한 프로세스가 결정됩니다. 해당 직무 역량을 유지하는 것은 시스템 가용성에 영향을 미칠 수 있는 매우 중요한 요인입니다. 운영 인력이 최신 기술을 유지하여 루틴 및 비상 프로시저를 정확하고 효율적으로 수행할 수 있도록 보장하기 위해서는 지속적인 교육이 필수적입니다.

계획된 시스템 정지 시간

일상적인 시스템 유지 관리를 수행하는 동안 데이터베이스 가용성을 제공하고 응용 프로그램 장애 조치나 다른 이벤트에 의해 야기된 데이터 손실을 복구하는 과정에서 SQL Server 2005와 Oracle 10g는 각각 고유한 고가용성 기능을 제공합니다. 다음 섹션에서는 오늘날 엔터프라이즈 데이터베이스 플랫폼에서 운영 지속성을 향상시키고 데이터 훼손을 방지하도록 설계된 다양한 데이터베이스 기능을 설명합니다.

서버 유지 관리를 위한 데이터베이스 고가용성 솔루션

오늘날 서버 하드웨어는 매우 안정적이며 모든 최상급 하드웨어 공급업체들은 주요 시스템 구성 요소들을 위해 많은 이중화 기능을 제공하고 있지만 하드웨어 및 운영 체제에 대한 유지 관리와 업그레이드는 여전히 피할 수 없는 현실입니다.

하드웨어 업그레이드 및 유지 관리

Microsoft Windows Server™ 2003의 핫 스왑 RAM 및 RAID 드라이브 지원 기능은 시스템에 메모리와 디스크 저장소를 추가해야 하는 것과 같은 가장 일반적인 하드웨어 업그레이드 시나리오를 해결합니다. 이러한 기능을 지원하는 하드웨어를 사용하면 가용성의 중단 없이 시스템이 실행되는 동안 RAM 및 RAID 드라이브를 동적으로 추가할 수 있습니다. 이러한 경우에도 일상적인 하드웨어 유지 관리 작업을 수행해야 하는 때가 있기 마련입니다. 예를 들어, 시스템 로그에 있는 이벤트는 장애가 발생한 시스템 구성 요소를 나타내거나 시스템 재부팅이 필요한 운영 체제 또는 응용 프로그램 수준 서비스 팩을 적용해야 할 수도 있습니다. 단일 서버 환경에서 이러한 유형의 이벤트는 계획된 시스템 정지 시간을 발생시키게 됩니다. 계획된 시스템 정지 시간은 시스템에 최소한의 영향을 미치는 시점에 실행되도록 일정을 지정할 수 있기 때문에 계획되지 않은 시스템 정지 시간보다 업무 운영의 연속성에 큰 타격을 입히게 됩니다.

그럼에도 불구하고, 최대 수준의 가용성을 요구하는 기업들에게는 계획된 시스템 정지 시간조차도 지나치게 긴 시간일 수 있습니다. 일상적인 유지 관리 및 시스템 정지 시간에 대한 요구를 해결하기 위해 오늘날 출시되어 있는 모든 엔터프라이즈 데이터베이스 플랫폼은 IT 직원들이 롤링 업그레이드를 수행할 수 있도록 하는 멀티서버 클러스터링 및 기타 가용성 기능을 지원합니다. Microsoft Windows Clustering Services 및 Oracle RAC는 일상적인 유지 관리를 수행하기 위해 클러스터 내 한 시스템 또는 시스템 그룹을 수동으로 오프라인으로 전환할 수 있는 롤링 업그레이드 기능을 지원합니다. 예를 들어, 1년 365일 하루 24시간(24x7) 지속적인 가용성을 필요로 하는 응용 프로그램이 있을 경우, 이러한 다중 서버 클러스터링 또는 기타 가용성 기술 중 하나를 사용해 데이터베이스 플랫폼상에 해당 응용 프로그램을 구현할 수 있습니다. 이후 유지 관리가 이루어져야 할 때 수동 장애 조치를 시작하여 유지 관리가 필요한 노드에서 워크로드를 차단합니다. 그런 다음 오프라인 상태에 있는 동안 노드를 수리하고 업그레이드하거나 패치를 적용합니다. 나머지 클러스터 노드 또는 대기 서버는 노드가 유지 관리되고 있는 동안 워크로드를 사용할 수 없다고 가정할 것입니다. 따라서 응용 프로그램 가용성은 전혀 손상되지 않습니다. 프로시저가 완료되면 노드를 클러스터로 복원하고 정상적인 작업으로 대체할 수 있습니다. 필요하다면 이 프로세스를 다른 클러스터 노드에 반복할 수 있습니다. 롤링 업그레이드를 실행할 수 있는 이러한 기능으로 일상적인 유지 관리와 관련된 계획된 시스템 정지 시간을 제거할 수 있습니다.

데이터베이스 유지 관리

데이터베이스 유지 관리는 효율적이고 가용성이 뛰어난 데이터를 위해 필수적인 요소입니다. 인덱스를 유지하고 튜닝을 수행하며 백업을 실행해야 합니다. 효과적인 작업 계획을 수립하기 위해서는 어느 정도의 데이터베이스 유지 관리 시간을 감안해야 합니다. 그러나 데이터베이스 유지 관리 시간이 데이터베이스 시스템 정지 시간과 반드시 일치해야 할 필요는 없습니다. DBA는 몇몇 도구들을 사용하여 성능 및 시스템 정지 시간에 미치는 부정적인 영향을 없애고 루틴 데이터베이스 유지 관리 작업 동안의 가동 시간을 개선할 수 있습니다.

SQL Server 2005에서는 보다 용이하게 온라인 서버와 데이터베이스 유지 관리 업무를 수행할 수 있도록 대부분의 SQL Server 시스템 속성을 동적 구성할 수 있습니다. 다음 표에는 온라인으로 변경할 수 있는 서버 속성이 나열되어 있습니다.

allow updates	network packet size
cost threshold for parallelism	query governor cost limit
cursor threshold	query wait
default full-text language	recovery interval
default language	remote login timeout
index create memory	remote proc trans
max degree of parallelism	remote query timeout
max server memory	show advanced options
max text repl size	two digit year cutoff
min memory per query	user options
min server memory	network packet size
using nested triggers	query governor cost limit

SQL Server 2005는 테이블 및 뷰에 있는 클러스터된/클러스터되지 않은 인덱스의 온라인 조각 모음을 수행할 수 있도록 인덱스 조각 모음 문을 제공하는 것이 특징입니다. DBCC INDEXDEFRAG 문은 오랫동안 잠금을 유지하지 않기 때문에 실행 중인 쿼리나 업데이트를 차단하지 않습니다. SQL Server 2005 및 Oracle 10g는 모두 데이터 가용성, 응답 시간, 디스크 공간 활용도, 데이터베이스 성능을 향상시키는 온라인 인덱싱 작업을 포함합니다. 기본 테이블 데이터가 쿼리되거나 업데이트되는 동일한 시점에 조각 모음을 실행하거나 인덱스를 추가 또는 재작성할 수 있습니다.

또한, 향후 사용자 로드와 쿼리가 변경되더라도 성능을 향상시킬 수 있도록 각 데이터베이스 솔루션에는 종합적인 데이터베이스 튜닝 도구가 포함됩니다. 이러한 도구를 통해 DBA는 쿼리 문 작업을 모니터링 및 분석하고 디스크 I/O 하위 시스템 성능을 모니터링함으로써 구성 설정을 활용하는 최적의 방법에 대한 조언을 얻을 수 있습니다.

예기치 못한 시스템 정지 시간

여러 다양한 엔터프라이즈급 데이터베이스 플랫폼은 각각 고유의 메서드 및 기술을 약간씩 다른 방식으로 조합하여 비슷한 수준의고가용성을 제공할 수 있습니다. 다양한고가용성 솔루션들은 서버 장애 및 데이터 손상을 방지하는 데 각기 다른 수준의 보호 기능을 제공하며 어느 메서드를 사용하느냐에 따라 솔루션 구입에 필요한 기술 뿐만 아니라 솔루션 구현 및 운영을 위해 필수적인 인건비 측면에서 차이가 발생합니다.

데이터베이스 복구를 위한 데이터베이스고가용성 솔루션

응용 프로그램의 오류와 데이터 손상을 수정해야 하는 경우,고가용성 환경을 구축하는 능력에 영향을 미칠 수 있습니다. 사용자와 직원들이 데이터에 액세스할 수 있도록 할 계획이라면 정확한 데이터를 사용할 수 있도록 하는 것이 중요합니다. 데이터베이스의 데이터가 훼손된 경우(예를 들어, 부정확한 정보로 데이터베이스를 업데이트하거나 부주의하게 정보를 삭제하는 등 사용자의 실수로 인해) 해당 데이터를 신속히 파악하여 원래 상태로 복구하는 프로시저를 마련해야 합니다

SQL Server 2005-백업 및 트랜잭션 시정 시간(Point-in-Time) 복구

탄탄한 백업 및 복구 계획은고가용성의 필수 구성 요소입니다. SQL Server 2005의 다양한 복구 모델은 로깅 오버헤드와 완벽한 데이터 복구 기능 간의 균형을 조절합니다. SQL Server 2005는 단순(Simple), 전체(Full) 및 대량 로그(Bulk-Logged)라는 3가지 복구 모델을 제공합니다.

- **단순 복구(Simple Recovery) 모델** 이 모델은 최저 수준의 로깅 오버헤드를 제시하지만 최종 백업이 완료된 이후의 데이터는 전혀 복구할 수 없습니다. 단순 복구 모델을 사용하면 최종 백업 이후에 이루어진 모든 데이터 수정 사항이 손실됩니다.
- **전체 복구(Full Recovery) 모델** 이 모델은 단순 복구 모델과 반대로 모든 데이터 수정 사항을 기록합니다. 전체 복구 모델을 사용하면 모든 데이터를 장애 발생 시점까지 복구할 수 있습니다. 기본적으로, SQL Server는 전체 복구 모델을 사용합니다.
- **대량 로그 복구(Bulk-Logged Recovery) 모델** 이 모델은 대량 복사 및 SELECT INTO와 같은 대량 작업을 제외한 모든 트랜잭션을 기록하는 방식으로, 위에서 설명한 2가지 모델의 중간적인 특징을 가지고 있습니다. 복구 상황에서는 이들 작업들이 손실됩니다. 반면, 대량 로그 모델은 최종 데이터베이스 또는 로그 백업 완료 시점까지 복구할 수 있습니다.

일단 복구 모델을 선택하면 기업은 데이터베이스 백업 계획을 결정해야 합니다. 백업은 디스크, 테이프 또는 기타 미디어를 이용해 이루어질 수 있습니다. 디스크 백업 수행은 데이터를 백업 및 복원하는 가장 빠른 방법입니다. 드라이브 오류를 방지하기 위해 항상 데이터베이스 데이터와는 별도의 드라이브 및 컨트롤러로 백업해야 합니다. SQL Server는 전체, 차등 및 로그 백업 등 3가지 기본 유형의 백업을 지원합니다.

- **전체 데이터베이스 백업** 이 백업 유형은 데이터베이스 전체를 복사하는 것으로 복원 프로세스를 시작해야 하는 알려진 시점을 제공합니다.
- **차등 백업** 이 백업은 최종 전체 데이터베이스 백업 이후 수정된 데이터베이스 페이지만 복사합니다. 빈번한 주기로 차등 백업을 실행하여 최종 현재 트랜잭션까지 데이터베이스를 복구하기 위해 적용해야 하는 트랜잭션 로그 개수를 최소화합니다.
- **로그 백업** 이 백업은 트랜잭션 로그만 복사하며 트랜잭션 로그 백업은 최종 차등 백업이 복원된 후 적용될 수 있습니다.



트랜잭션 지정 시점(point-in-time) 복구 기능을 이용하면 전체 데이터베이스를 지정된 모든 시점으로 복구할 수 있습니다. SQL Server 2005 트랜잭션 로그는 트랜잭션 로그가 마지막으로 백업된 이후의 모든 변경 사항에 대한 일련의 레코드입니다. 트랜잭션 로그 백업을 사용하면 SQL Server 데이터베이스를 지정된 시간대로 복구할 수 있습니다. 예를 들어, 데이터베이스와 함께 데이터 손상을 야기한 응용 프로그램 오류가 새벽 4시에 발생했다면 SQL Server 2005 트랜잭션 로그 백업을 이용해 데이터 손상이 발생하기 바로 직전인 새벽 3시 59분 시점의 데이터베이스로 복구할 수 있습니다.

SQL Server 트랜잭션 로그가 복원되면 로그에 포함된 모든 트랜잭션이 롤 포워드(roll forward)됩니다. 데이터베이스를 업데이트하는 데 사용되는 데이터 이외에도, 각 트랜잭션 로그 레코드에는 트랜잭션 발생 시점을 표시하는 타임스탬프가 포함됩니다. 트랜잭션 로그의 완료 시점에 도달하거나 복원 작업 도중에 지정된 시간에 이르면 데이터베이스는 최종 트랜잭션 시점과 정확하게 일치하는 상태로 돌아가기 때문에 데이터 손상 오류를 신속하게 복구할 수 있습니다.

이러한 표준 백업 옵션 뿐 아니라, SQL Server 2005는 페이지 또는 페이지 그룹을 복원할 수 있는 매우 세분화된 페이지 단위 복원 기능을 지원합니다.

SQL Server 2005-로그 전달과 지연

“대기 시간대(window in time)”를 이용해 데이터 손상 시 쉽게 복구할 수 있도록 로그 전달을 구성할 수 있습니다. 로그 전달은 하나의 데이터베이스 서버에서 하나 이상의 백업 서버로 트랜잭션 로그를 보내는 데이터베이스 가용성 기술입니다. 주 서버 또는 데이터베이스 오류 발생 시 이러한 트랜잭션 로그를 백업 서버에 적용할 수 있습니다. 주 서버의 트랜잭션 로그 백업은 보조 서버로 전송되기 전에 지정된 시간 동안 지연될 수 있습니다. 예를 들어, 트랜잭션 로그를 보조 서버에 바로 기록하는 대신, 5분마다 보조 서버에 기록하도록 로그 전달을 구성할 수 있습니다. 데이터 오류가 5분 간의 대기 시간대 중에 발생했다면 보조 서버의 트랜잭션 로그를 사용해 5분 전의 상태로 데이터를 복구할 수 있습니다.

SQL Server 2005-파일 그룹 복원

SQL Server 2005에서는 지정 시점(Point-in-time) 복구 및 로그 전달 복구 기능이 지원되며 트랜잭션 로그 백업을 적용하면 특정 시간대로 데이터를 복구할 수도 있습니다.

SQL Server 2005의 새로운 기능을 통해 손상된 객체만 쉽게 복구할 수 있으며 세분화된 복원 기능을 이용하여 데이터베이스에서 선택한 파일 그룹을 복원할 수 있습니다. SQL Server 2000에서는 가용성의 단위가 데이터베이스이기 때문에 데이터베이스를 사용할 수 있기 전까지 해당 데이터베이스의 모든 구성 요소들을 변경해서는 안되며 그대로 두어야 합니다. SQL Server 2005의 가용성 단위는 파일 그룹이어서 현재 복원 중인 데이터만 사용할 수 없기 때문에 가용성이 향상되며 다른 파일 그룹에 포함된 나머지 데이터베이스 데이터는 그대로 액세스할 수 있습니다. SQL Server 2005는 파일 그룹을 한꺼번에 복원하거나, 주 파일 그룹이 준비되었을 때 페이지 또는 페이지 그룹을 한 번에 복원할 수도 있습니다.

SQL Server 2005-빠른 복구

Oracle의 FastStart Fault Recovery와 마찬가지로 SQL Server 2005의 빠른 복구 기능은 트랜잭션 로그가 롤 포워드 되는 즉시 사용자가 복구 중인 데이터베이스에 다시 연결할 수 있도록 하여 데이터 가용성을 높여줍니다. 초기 버전의 SQL Server에서는 사용자가 해당 데이터베이스의 관련 영역에 액세스할 필요가 없더라도 불완전한 트랜잭션이 롤백할 때까지 대기해야 했습니다.

SQL Server 2005-데이터베이스 스냅샷

SQL Server 2005에는 손상된 데이터를 신속하고 쉽게 복원할 수 있는 데이터베이스 스냅샷 기능이 포함되어 있습니다. 데이터베이스 스냅샷은 데이터베이스와 이에 수반된 저장소의 전체 복사본을 만드는 데 따른 오버헤드 없이 데이터베이스의 읽기 전용 뷰를 생성하는 기능을 제공합니다. 그림 1에서 데이터베이스 스냅샷의 개요를 확인할 수 있습니다.

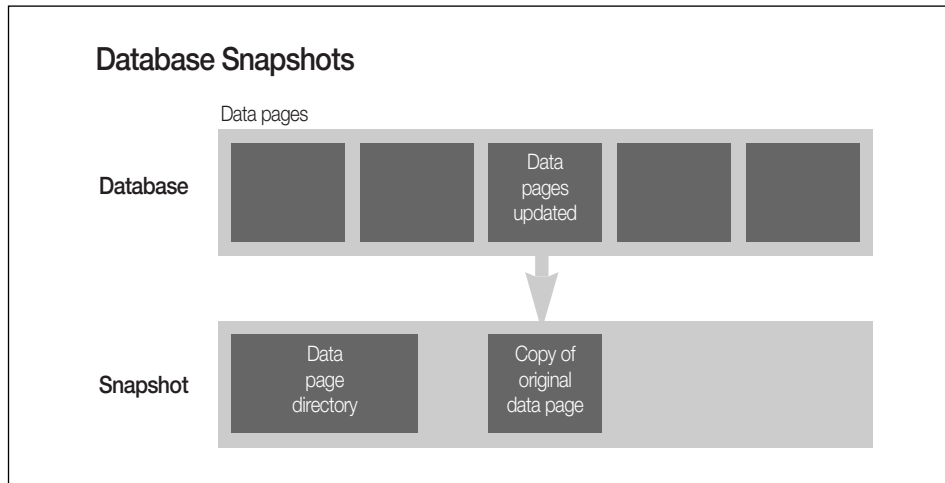


그림 1: 데이터베이스 스냅샷

데이터베이스 스냅샷은 데이터베이스 복사본과는 다릅니다. 데이터베이스 스냅샷은 데이터베이스 정보의 변경 사항을 포함하는 데 필요한 공간만 차지합니다. 데이터베이스가 변경되면 스냅샷은 데이터베이스로부터 원본 페이지의 고유 복사본을 수신합니다. 원치 않는 데이터베이스 변경 작업을 무효화하고 원래대로 복구하기 위해 스냅샷의 원본 페이지를 데이터베이스에 다시 적용할 수도 있습니다.

Oracle 10g-RMAN (Recovery Manager)

Oracle 10g에는 백업 생성 및 복원 프로세스를 관리하는 RMAN(Recovery Manager)이라는 도구가 포함되어 있습니다. RMAN 도구는 RMAN 실행 파일 기능, 백업할 대상 데이터베이스, 복구 카탈로그로 구성됩니다. 복구 카탈로그가 지정되지 않은 경우 백업 세부 내용은 대상 데이터베이스의 controlfile에 저장됩니다. Recovery Manager의 일부 기능은 손상되거나 원치 않는 데이터 변경을 복구하는 데 사용할 수 있습니다. controlfile은 데이터 파일 생성 시점부터 복구 시점까지 데이터 파일 및 아카이브 로그 파일에 대한 정보를 포함하고 있습니다.

표준 RMAN 백업은 특정 데이터 파일에 대한 데이터 블록으로 이루어진 백업 단위를 포함하고 있습니다. 데이터 블록은 특수한 압축 형식으로 저장됩니다. 데이터 파일을 다시 저장해야 하는 경우, 백업 단위의 블록에서 전체 데이터 파일이 다시 작성되어야 합니다. 이제 Oracle 10g를 사용하면 이미지 복사본을 데이터베이스, 테이블스페이스 또는 데이터 파일 수준으로 생성할 수 있습니다. 데이터 파일의 실제 구조가 이미 존재하기 때문에 데이터 파일의 이미지 복사본을 더욱 신속하게 복구할 수 있습니다. Incrementally Updated Backup이라는 RMAN 기능을 이용하여 특정 시간대로 롤 포워드하기 위해 데이터 파일 이미지 복사본 백업에 증분 데이터베이스 변경을 적용할 수 있습니다. 때때로 데이터 파일 이미지 복사본을 증분 백업으로 업데이트함으로써 데이터 파일의 이미지 복사본은 점차 최신 상태와 근접해집니다. 이를 통해 데이터 복구 시간을 단축할 수 있습니다.



변경 내용 추적(Change Tracking)은 증분 백업의 성능을 향상시킬 수 있는 Oracle 10g의 옵션 기능입니다. 이전 버전의 Oracle에서는 최종 증분 백업 이후 데이터 파일의 모든 블록에 대해 변경 여부를 검사해야 했습니다. 변경 내용 추적 기능이 실행되면 변경된 블록의 ID가 변경 추적 파일에 기록되기 때문에 최초의 증분 백업 블록만 완벽하게 검사하면 됩니다. 이후의 증분 백업은 변경된 블록을 백업해야 할지 여부를 파악하기 위해 변경 내용 추적 파일을 검사합니다.

Oracle 10g-플래시백(Flashback)

Oracle 10g의 Flashback은 SQL Server 2005 데이터베이스 스냅샷과 매우 유사한 기능을 제공합니다. Flashback 데이터베이스는 표준 백업 미디어 대신 Flash Recovery Area을 사용해 데이터베이스를 특정 시간대로 복구할 수 있도록 합니다. Flashback 기능은 데이터의 대규모 블록을 복구하는 데 가장 적합한 RMAN 기능과 달리, 손상된 단순한 테이블 및 행 데이터를 복구하는 데 적합합니다. 이 기능을 사용하려면 DBA는 Flashback 데이터베이스 로그, redo archive 로그 및 RMAN 백업을 포함하는 Flash Recovery Area을 구성해야 합니다. 블록 변경의 복사본은 Flashback 로그에 기록되며 데이터에 사용자 또는 응용 프로그램 오류가 발생한 경우 복구할 수 있습니다. Flashback은 Flash Recovery Area에서 데이터를 쿼리 및 복구하는 데 사용되는 특정 SQL 문을 포함하기 때문에 객체에 액세스하려면 사용자에게 권한이 부여되어야 합니다.

주의

Flashback 기술의 중요한 한계는 참조 무결성 지원 기능을 기본적으로 제공하지 않는다는 것입니다. Flashback을 사용해 종속성을 지닌 테이블을 복구하고 이러한 종속 객체들이 변경됐을 경우 데이터베이스[0]에 불일치를 야기할 수 있습니다.

Flashback 테이블, 데이터베이스 및 트랜잭션 쿼리 기능은 Oracle Enterprise Edition에서만 제공됩니다.

서버 오류를 막는 데이터베이스 고가용성 솔루션

고가용성 환경에서 가장 중시되는 기술 문제는 바로 서버 오류를 막는 것입니다. 서버 오류는 사용자들이 서버 시스템을 액세스할 수 없도록 만드는 예기치 못한 이벤트로 정의할 수 있습니다. 여러 다양한 요인들로 인해 서버 오류가 야기할 수 있으며 다음과 같이 다양한 하드웨어 및 소프트웨어가 그 원인이 될 수도 있습니다.

- 하드웨어 오류(CPU, RAM, 저장소, I/O 또는 전원 공급 장치)
- 운영 체제 또는 장치 드라이버 오류
- 데이터베이스 서버 오류

하드웨어 오류로부터 보호하는 첫 번째 방법은 핵심 구성 요소의 중복을 제공하는 하드웨어 플랫폼에 투자하는 것입니다. 예를 들어 오늘날 최상급 하드웨어 공급업체들이 제공하는 대부분의 서버는 중복 전원 공급장치, 기본 제공 UPS(무정전 전원 공급 장치)는 물론 핫 스왑 가능한 RAM 및 RAID 드라이브를 제공합니다.

소프트웨어 측에서, 고가용성을 보장하는 가장 중요한 단계는 최신 서비스 팩을 이용해 운영 체제, 장치 드라이버 및 응용 프로그램 소프트웨어를 최신 상태로 유지하는 것입니다. 이러한 방식으로 시스템에 최신 소프트웨어 업데이트 및 보안 업데이트가 설치되도록 보장할 수 있습니다. Microsoft는 시스템 업데이트 문제를 해결하는 많은 기술을 제공하고 있습니다. SMS(System Management Server) 제품은 엔터프라이즈급 소프트웨어 업데이트 및 인벤토리 기능을 제공합니다. 중기업의 경우, 새로운 WSUS(Windows Server Update Services)는 조직 전반에 걸쳐 Microsoft Windows® 및 Microsoft Windows Server 업데이트를 배포하는 기능을 제공합니다. 또한 Windows Update 사이트로 소기업들에게 시스템 업데이트를 지원합니다.

최신 패치를 계속 설치하는 것은 중요한 단계이며 모든 소프트웨어 업데이트를 프로덕션 환경에 설치하기 전에 테스트 환경에서 엄격하게 테스트하는 품질 보증 절차를 갖추어야 합니다.

이와 같은 기본적인 시스템 필수 전제 조건을 해결하는 것은 데이터베이스 및 서버 가용성 개선을 위한 필수적인 단계이지만 이러한 방법만으로는 고가용성을 보장할 수 없습니다. 서버 오류 발생을 방지하고 데이터베이스 고가용성을 보장하기 위해 경쟁 데이터베이스 플랫폼이 지원하는 고가용성 기술 중 하나를 채택해야 합니다. 이들 기술은 시스템 오류 방지 기능을 한층 강화하고 서버 오류 발생 시 보다 신속하게 복구할 수 있도록 한다는 데 목적을 두고 있습니다.

클러스터링 기술과 로그 전달 기능의 활용은 더욱 우수한 고가용성 데이터베이스 플랫폼을 구축하는 데 있어 중요한 기술적 의미가 있습니다. 클러스터링은 기본적으로 하나 이상의 백업 서버로 장애가 발생한 주 서버의 워크로드를 중단없이 위임받을 수 있는 환경에서 다중 서버를 사용하는 것입니다. 클러스터링 이외에도 오늘날의 모든 경쟁 엔터프라이즈 데이터베이스 플랫폼은 서버 오류를 방지할 수 있는 여타 많은 기술을 지원합니다. 로그 전달 및 복제도 여기에 포함됩니다. 이 섹션에서는 이러한 대안을 각각 검토하고 각 엔터프라이즈 데이터베이스 제품에서 이들 기능이 어떻게 구현되는지 설명합니다.

SQL Server 2005 N-Way 클러스터링

Windows Server 2003의 향상된 클러스터링 지원을 활용하는 SQL Server 2005는 Windows Server 2003 Datacenter Edition에서 최고 8개의 노드 클러스터, Windows Server 2003 Enterprise Edition에서 4개의 노드 클러스터, 그리고 Windows 2000 Advanced Server에서 2개의 노드 클러스터를 지원합니다. 설치 프로세스와 관리 도구는 모두 클러스터를 인식합니다.

Microsoft Windows Clustering Services는 서버 오류 시 데이터베이스 플랫폼을 보호하는 중요한 기술입니다. Windows Clustering Services는 SQL Server, Oracle 및 DB2 등의 모든 엔터프라이즈 데이터베이스 응용 프로그램에 사용할 수 있습니다. Windows Server 운영 체제의 버전에 따라 지원하는 노드의 기능이 달라집니다. 다음 표는 서로 다른 Windows 2000 Server 및 Windows Server 2003 에디션의 기본적인 클러스터링 기능을 설명합니다.

운영 체제	노드
Windows 2000 Server	0
Windows 2000 Advanced Server	2
Windows 2000 Datacenter Server	4
Windows Server 2003 Standard Edition	0
Windows Server 2003 Enterprise Edition	4
Windows Server 2003 Datacenter Edition	8(SQL Server Enterprise Edition 64비트만 해당)

Windows 2000 Server 및 Windows Server 2003 Standard Edition은 장애 조치 클러스터링을 지원하지 않습니다. Windows 2000 Advanced Server는 2-노드 클러스터를 지원합니다. Windows Datacenter Server 2000 및 Windows 2003 Enterprise Edition은 4-노드 클러스터를 지원하며, Windows Server 2003, Datacenter Edition은 최대 8-노드 클러스터를 지원합니다. 지원되는 노드 수는 호스트 운영 체제의 기능에 따라 달라집니다.

Windows Clustering를 사용할 때 클러스터 내 각 물리적 서버를 노드라고 하며 노드들이 서로 연동되어 클러스터를 형성합니다. 클러스터 내 모든 노드는 지속적인 통신 상태를 유지합니다. 클러스터 내 노드 중 하나가 사용할 수 없게 되면 다른 노드가 자동으로 그 노드의 임무를 맡고 사용자에게 장애가 생긴 노드와 동일한 서비스를 제공하기 시작합니다. 이 프로세스를 장애 조치라고 합니다. 서비스 도중에 중단되는 일이 없는 전문 내결함성 타사 하드웨어 솔루션과 달리, Windows 클러스터에 대한 장애 조치 프로세스에 설치된 하드웨어에 따라 약 20초의 짧은 시간차가 발생합니다. 뿐만 아니라, 트랜잭션 일관성을 유지하기 위해 장애 조치 노드에 있는 데이터베이스를 복구해야 합니다. 이러한 복구 기간의 길이는 장애 조치 시 수행되고 있던 데이터베이스 작업의 수준과 사용되는 하드웨어의 유형에 크게 좌우됩니다. 장애가 발생한 노드에 연결된 클라이언트는 연결이 종료되며 다시 연결을 시도하면 백업 노드에 있는 클러스터 자원에 액세스할 수 있습니다. Windows Clustering은 다음과 같은 이점을 제공합니다.

- **자동 장애 조치** 오류가 감지되면 클러스터는 자동으로 주 노드에서 보조 노드로 전환합니다.
- **클라이언트에 대한 투명성** 장애 조치가 완료된 후 클라이언트는 동일한 가상 이름 및/또는 IP(Internet Protocol) 주소를 사용해 클러스터에 다시 연결할 수 있습니다.
- **트랜잭션 무결성** 장애 조치 프로세스가 완료된 후 모든 커밋된 트랜잭션이 저장되고 사용할 수 있게 됩니다.
- **신속한 장애 조치** 대개의 경우 시스템에 대한 장애 조치 프로세스는 약 30초 내에 완료될 수 있습니다. 이후 데이터베이스 가용성은 롤 포워드 또는 롤 백을 실행해야 하는 트랜잭션 수에 따라 달라집니다.

그림 2에서 Windows Clustering Services의 기본적인 개요를 확인할 수 있습니다.

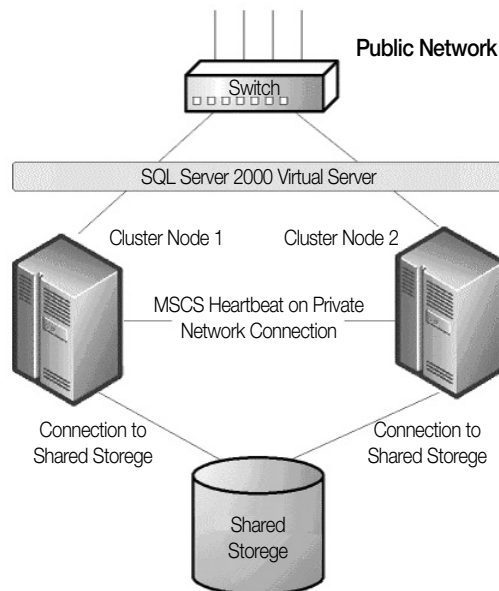


그림 2: 기본적인 클러스터링

각 클러스터 노드는 다음 하드웨어를 필요로 합니다.

- Windows Server 운영 체제를 위한 하드 디스크. 이 디스크는 공유되지 않으며, 공유 저장소를 연결하는 데 사용되는 컨트롤러로 연결되지 않습니다. 대신 이 디스크는 자체 컨트롤러를 사용하여 가용성 향상을 위해 미러링되어야 합니다.
- 클러스터의 공유 디스크 저장소에 연결하는 SCSI 또는 Fibre Channel 어댑터
- 2개의 NIC(network interface card). NIC 1개는 클러스터 노드를 외부 네트워크에 연결하는 데 사용되며, 2번째 NIC는 클러스터의 “하트비트”-노드 가용성을 알리는 신호를 유지하는 전용 클러스터 네트워크에 사용됩니다.

클러스터 내 노드는 공유 저장소 하위 시스템을 사용하기 때문에 일반적으로 비교적 서로 가까이 위치해야 합니다. 노드 간 거리는 노드가 저장소 하위 시스템에 사용하는 연결에 따라 좌우됩니다. 공유 SCSI 연결을 사용하는 클러스터는 비교적 근접해야 하지만(수 미터 이내) 파이버 채널에 의해 연결된 노드는 수 마일 떨어져 있을 수 있습니다. 이 솔루션은 서버 오류로 인해 서비스 정지가 발생할 가능성을 줄이지만 여전히 전체 위치에 영향을 미치는 이벤트에 취약합니다. Geo-Cluster(멀티-사이트 클러스터)는 클러스터 노드를 지리적으로 분리하여 이 문제를 해결합니다. 각각 다른 위치 사이에 퀵림 디스크를 미러링하는 방법으로 이를 수행합니다. 클러스터는 본질적으로 노드 간의 지리적 거리를 인식하지 못하기 때문에 이들 솔루션은 기업 인프라의 네트워크 및 저장소 수준으로 구현되어야 합니다.

Windows Clustering 솔루션을 구현하기 위해 Windows Clustering 소프트웨어와 함께 사용할 Microsoft 인증 서버 시스템을 사용해야 합니다. 지원되는 하드웨어 플랫폼 목록은 Windows HCL Home 웹 페이지에서 확인할 수 있습니다. 인증된 클러스터 시스템을 사용하고 기존 부품에서 자체 클러스터를 조합하지 않도록 해야 합니다. 그 이유는 하드웨어 공급업체들이 Microsoft가 정의한 요구 사항을 이행하고 시스템을 전체적인 측면에서 증명하기 위해 자체 시스템에 대한 엄격한 테스트를 실시하고 있기 때문입니다. 증명된 시스템이 아닌 기존 부품에서 “자체 구축한(Home-grown)” 클러스터는 지원되지 않는 구성입니다.

N+1 구성(N개의 활성 노드 + 1개의 예비 노드)을 사용해 SQL Server 2005와 Windows Server 2003를 조합하면 고가용성을 갖춘 응용 프로그램 실행하는 매우 유연하고 비용 효율적인 클러스터링 환경을 구축할 수 있습니다. 예를 들어 8-노드 클러스터를 N+1 구성으로 조합하면 노드 8개 중 7개는 서로 다른 서비스를 능동적으로 제공하도록 설정할 수 있으며 8번째 노드는 서버 오류 시 활성 노드 7개의 서비스를 대신 위임받게 되는 수동 노드가 됩니다. 그림 3은 활성 노드가 7개인 8-노드 클러스터를 나타내며 나머지 노드 1개는 7개의 활성 노드 중 하나라도 장애를 일으킬 경우 실행될 수 있도록 대기 상태에 있습니다.

Eight-node Cluster Support

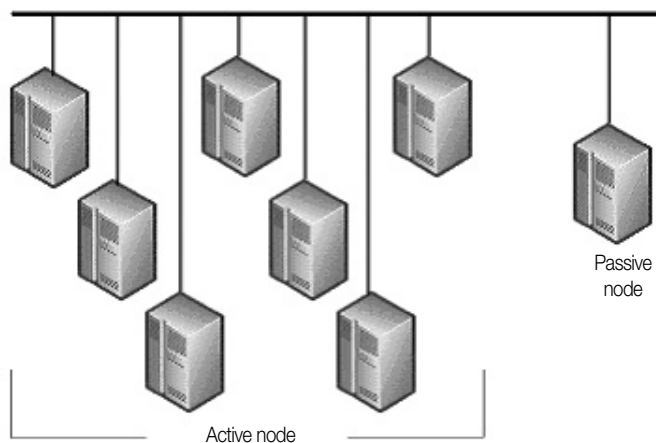


그림 3: N-Way 클러스터

Database Mirroring

SQL Server 2005의 새로운 Database Mirroring 기능은 서버 또는 데이터베이스 오류로 인해 발생한 예기치 못한 시스템 정지 시간에 대비할 수 있는 또 다른 중요한 옵션입니다. 그 이름에서 알 수 있듯이 Database Mirroring은 데이터베이스 수준의 장애 조치 기능을 제공하는 것으로 주 데이터베이스에 장애가 발생하면 보조 SQL Server 시스템에 위치한 두번째 대기 데이터베이스를 거의 즉시 활성화합니다. Database Mirroring은 단일 데이터베이스를 지원하거나 동일 서버상의 여러 데이터베이스를 지원할 수도 있습니다. 이 기능을 사용하면 데이터 손실이 거의 발생하지 않으며 보조 데이터베이스는 항상 주 데이터베이스 서버에서 처리되고 있는 현재 트랜잭션으로 업데이트됩니다. Database Mirroring 실행으로 트랜잭션 처리량에 미치는 영향은 거의 없습니다.

서버 수준에서 작동하는 Windows Clustering Services와 달리, Database Mirroring은 데이터베이스 수준에서 구현됩니다. Database Mirroring은 단 수 초에 불과한 거의 즉각적인 장애 조치를 제공하지만 클러스터링은 일반적으로 장애 조치 시간이 약 30초 정도에 이르며 데이터베이스 작업의 수준과 장애가 생긴 데이터베이스의 규모에 따라 더 많은 시간이 걸리기도 합니다. Database Mirroring은 클러스터링 솔루션에서처럼 공유 쿼럼 디스크가 없기 때문에 디스크 오류로부터 보호하는 추가 보호 기능을 제공합니다. 뿐만 아니라, Database Mirroring에는 거리 제한이 거의 없지만 클러스터링을 사용하는 고가용성 솔루션은 약 100마일 범위 내에서만 클러스터 노드 간 하트비트를 전송할 수 있습니다. 특정 하드웨어 구성을 필요로 하는 클러스터링과 달리, Database Mirroring은 SQL Server를 지원하는 모든 표준 하드웨어와 연동됩니다. 그림 4에서 새로운 Database Mirroring 기능의 작동 방식에 대한 개요를 확인할 수 있습니다.

Database Mirroring

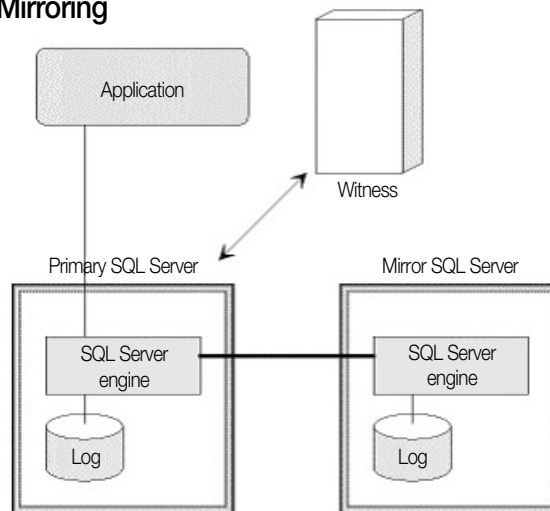


그림 4: Database Mirroring

Database Mirroring은 주 서버, 보조 서버 및 감시 서버라는 3가지 시스템을 사용하여 구현됩니다.

주 서버는 현재 데이터베이스 서비스를 제공하고 있는 SQL Server 시스템입니다. 기본적으로, 모든 수신 클라이언트 연결은 주 서버를 대상으로 이루어지며 보조 서버의 임무는 주 서버의 미러 데이터베이스의 복사본을 유지 관리하는 것입니다. 보조 서버는 백업 서비스만을 제공하도록 제한되지는 않으며 보조 서버의 다른 데이터베이스는 관련 없는 다른 응용 프로그램을 능동적으로 지원할 수 있습니다. 감시 서버는 본질적으로 어떤 시스템이 주 서버의 역할을 맡게 될 것인지 결정하는 책임을 가진 독립적인 제3자의 역할을 수행합니다.

Database Mirroring은 주 서버와 보조 서버 사이에 트랜잭션 로그를 전송하는 방식으로 실행되기 때문에 Database Mirroring은 실시간 로그 전달 응용 프로그램이 됩니다. 클라이언트 시스템이 트랜잭션을 주 서버에 기록하면 해당 요청은 데이터 파일에 기록되기 전에 주 서버의 로그 파일에 기록됩니다. 그런 다음 해당 트랜잭션 레코드는 보조 서버에 전송되어 보조 서버의 트랜잭션 로그에 기록됩니다. 보조 서버는 레코드를 로그에 기록한 후 주 서버에 승인을 전송합니다. 이를 통해 양 시스템 모두 레코드가 수신되었으며 이제 동일한 데이터가 각 서버의 로그 파일에 존재한다는 것을 인식할 수 있습니다. 커밋 작업의 경우 주 서버는 미러링 서버에서 승인을 받고 클라이언트에게 응답을 다시 전달하여 작업이 완료되었음을 알릴 때까지 대기합니다. 보조 서버는 기본적으로 지속적인 복구 상태에 있으며 수신 트랜잭션 로그 데이터를 이용해 데이터 파일을 최신 상태로 유지합니다.

클라이언트 응용 프로그램에 대한 고가용성을 높이기 위해 Database Mirroring은 TCR(Transparent Client Redirection)로 알려진 MDAC(Microsoft Data Access Components) 레이어의 업데이트와 함께 연동됩니다. TCR은 주 서버의 데이터베이스를 사용할 수 없는 경우 최종 사용자 시스템을 자동으로 보조 서버로 재전송합니다. 새로운 TCR 기능이 MDAC 레이어에 구현되어 있으므로 이 기능을 사용하기 위해 클라이언트 응용 프로그램을 변경할 필요가 없습니다. MDAC 소프트웨어 레이어는 주 서버와 보조 서버를 모두 인식하며, 주 서버에 처음 연결할 때 보조 서버의 이름을 입수합니다. 클라이언트가 주 서버와의 연결이 끊기면 MDAC가 주 서버로 한 번 재연결을 시도합니다. 연결 시도가 실패할 경우 MDAC는 자동으로 다음 연결 시도를 보조 서버로 전달합니다.

Database Mirroring은 SQL Server 2005 데이터베이스 스냅샷과 결합되어 미러 서버에 있는 데이터를 사용하는 보고 서버를 구축할 수 있습니다. 데이터베이스 스냅샷은 지정된 시간대 데이터베이스의 읽기 전용 스냅샷을 제공합니다. 그림 5에서 Database Mirroring과 데이터베이스 스냅샷을 결합하여 읽기 전용 보고 데이터베이스를 구축하는 예제를 확인할 수 있습니다.

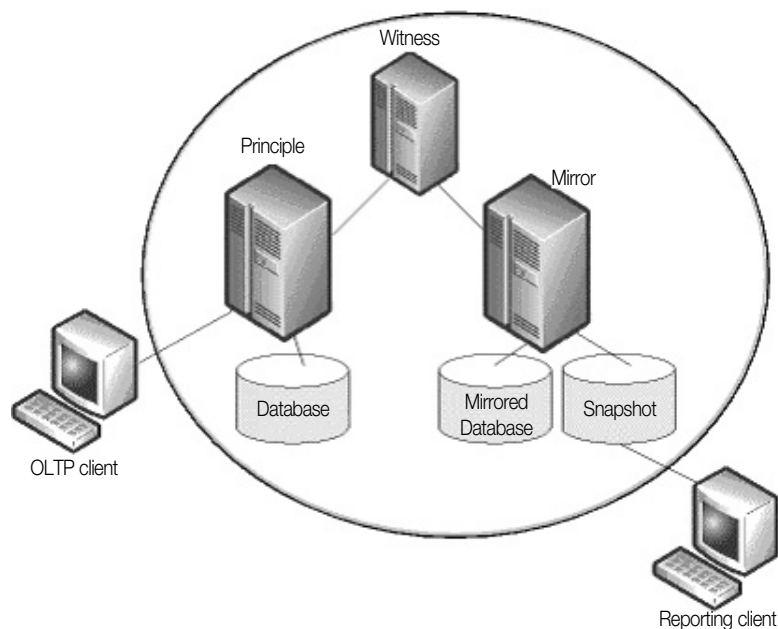


그림 5: 보고 서버

일반적으로 미러 서버에 있는 데이터는 항상 복구 모드에 있기 때문에 응용 프로그램이 액세스할 수가 없습니다. 그러나 미러 데이터베이스의 읽기 전용 복사본을 생성하는 미러 데이터베이스에 대해 데이터베이스 스냅샷을 생성할 수 있습니다. 해당 데이터베이스 스냅샷은 보고 응용 프로그램을 위한 읽기 전용 모드로 자유롭게 액세스할 수 있습니다.

SQL Server 2005 로그 전달

로그 전달은 주 서버의 오류로부터 데이터베이스를 보호하는 데 사용할 수 있는 고가용성 및 장애 복구 솔루션입니다. SQL Server 2005는 물론 이전 버전의 SQL Server에서도 지원되는 로그 전달은 서버 오류에 대한 저렴한 보호 수단으로 설계되었습니다. 로그 전달은 SQL Server를 실행할 수 있는 하드웨어 플랫폼에서 구현할 수 있으며, 모든 SQL Server 에디션과 연동하도록 구성할 수 있습니다. 로그 전달은 먼저 주 데이터베이스의 전체 백업을 대기 서버에 복구합니다. 그 시점부터 주 서버의 데이터베이스에서 트랜잭션 로그가 전송되고 대기 서버의 데이터베이스에 자동으로 적용됩니다. 대기 서버에서 트랜잭션 로그를 적용하기 위한 시간 지연과 함께 로그 전달을 구성하여 사용자 오류로부터 보호할 수 있습니다. 사용자가 지정한 시간 지연은 실수에 의한 삭제, 부정확한 데이터 입력, 응용 프로그램 오류, 기타 데이터 관련 문제와 같은 사용자 오류가 확산되는 것을 방지할 수 있는 시간적 여유를 제공합니다.

로그 전달의 구성 요소는 다음과 같습니다.

- **주 서버** 이 서버에는 프로덕션 데이터베이스가 포함됩니다. SQL Server Agent 작업은 프로덕션 데이터베이스의 변경 사항을 포착하기 위해 프로덕션 데이터베이스의 주기적인 트랜잭션 로그 백업을 만듭니다.
- **대기 서버** 대기 서버에는 주 데이터베이스의 복구되지 않은 복사본이 포함되어 있습니다. 대기 서버의 SQL Server Agent 작업은 주 서버에서 트랜잭션 로그 백업을 주기적으로 복사하여 대기 데이터베이스에 복구합니다.
- **모니터링 서버** 이 서버는 주 서버 및 대기 서버의 상태를 모니터링합니다.

Windows Clustering Services 또는 Database Mirroring과 달리, 로그 전달은 주 서버와 대기 서버의 역할을 전환시키는 자동 프로세스가 없습니다. Windows Clustering Services와 함께 로그 전달을 사용하여 사이트 수준 장애 뿐만 아니라 로컬 서버 오류로부터 보호할 수 있습니다. 로그 전달을 이용하여 하나 이상의 보조 서버에 있는 프로덕션 데이터베이스의 복사본을 유지 관리하고 서버나 사이트 오류 발생 시 보조 서버 중 하나를 새로운 주 서버로 전환시킬 수 있습니다.

SQL Server 2005-복제

트랜잭션 복제는 서버 오류 문제를 해결하는 데 사용할 수 있는 또 다른 기술 도구입니다. 복제는 주 데이터베이스에서 보조 데이터베이스로 트랜잭션을 전달할 수 있기 때문에 고가용성 솔루션으로 계획된 것은 아니지만, 트랜잭션 복제를 저렴한 데이터베이스 서버 백업 메커니즘으로 사용할 수 있습니다. 그림 6에서 SQL Server 2005 트랜잭션 복제의 개요를 확인할 수 있습니다.

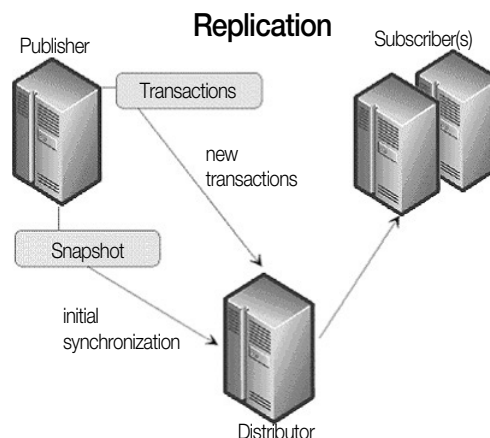


그림 6: 복제

SQL Server 트랜잭션 복제의 3가지 주요 구성 요소는 다음과 같습니다.

- **Publisher** Publisher는 복제되고 있는 데이터의 원본입니다.
- **Subscriber** Subscriber는 복제되는 데이터의 대상입니다. 하나 이상의 Subscriber가 존재할 수 있습니다.
- **Distributor** Distributor는 Publisher에서 Subscriber로의 데이터 전송을 처리합니다.

트랜잭션 복제는 원본 데이터베이스의 스냅샷을 사용해 Publisher 및 Subscriber에서 데이터베이스를 처음으로 동기화합니다. 트랜잭션이 Publisher에서 커밋되면 Subscriber로 포착 및 전송됩니다.

복제를 사용할 때의 이점은 보조 서버가 계속 사용 가능하며 보조 서버를 보고 서버로 바로 사용할 수 있다는 점입니다. 그러나 트랜잭션 복제는 고가용성을 위해 개발된 것이 아니기 때문에 보조 서버가 주 서버의 역할을 맡게 하는 프로세스는 자동이 아니라 수동입니다. 또한 오류 발생 이후 주 서버를 원래 역할로 복귀시키려면 전체 데이터베이스 복구가 필요합니다. 로그 전달과 같이 Windows Clustering Services와 함께 트랜잭션 복제를 사용하면 트랜잭션을 보조 사이트에 있는 서버로 복제함으로써 사이트 오류에 대비할 수 있습니다.

Oracle 10g—Real Application Clusters (RAC)

Oracle은 SQL Server 2005와 매우 유사한 고가용성 옵션들을 지원합니다. Oracle Fail Safe라는 기능을 사용하면 최대 8개의 노드와 함께 Windows Clustering Services를 사용할 수 있습니다. 또한 Oracle은 느슨하게 결합된 클러스터는 물론 Oracle의 트랜잭션 복제를 지원하여 서버 오류로부터 조직을 보호합니다. Oracle 9에서 Oracle 10g에 이르기까지 Oracle은 고가용성 컴퓨팅을 위한 또 다른 옵션인 Oracle Real Application Clusters(RAC)를 제공합니다. Oracle RAC는 Oracle 10g Standard Edition는 물론 10g Enterprise Edition에서도 제공됩니다. Oracle 10g Standard Edition의 RAC는 최대 CPU 4개로 제한됩니다. 관리 팩, 모니터링 팩, 파티셔닝 같은 향상된 RAC 관리 기능은 Enterprise Edition에서만 제공됩니다. Oracle 10g Standard Edition에서 Enterprise Edition으로 전환하는 비용은 매우 높습니다. Standard Edition은 CPU당 1만5천 달러이지만 Enterprise Edition의 경우 CPU당 4만 달러로 가격이 상승합니다.

Oracle RAC는 노드라고 하는 여러 개의 상호 연결된 컴퓨터로 구성됩니다. Oracle RAC 소프트웨어를 사용하면 연결된 노드들이 단일 컴퓨팅 환경으로 작동할 수 있습니다. Windows Clustering Services와 상당히 유사하게 Oracle만이 제한된 하드웨어 플랫폼에서 RAC를 지원합니다. 지원되는 하드웨어와 운영 체제 플랫폼 목록은 <http://metalink.oracle.com>에서 확인할 수 있습니다. Oracle은 최대 64개의 노드와 함께 RAC를 지원합니다. 상호 연결할 수 있는 인스턴스의 최대 개수는 호스트 운영 체제 플랫폼에 따라 달라집니다. 그림 7에서 Oracle RAC의 개요를 확인할 수 있습니다.

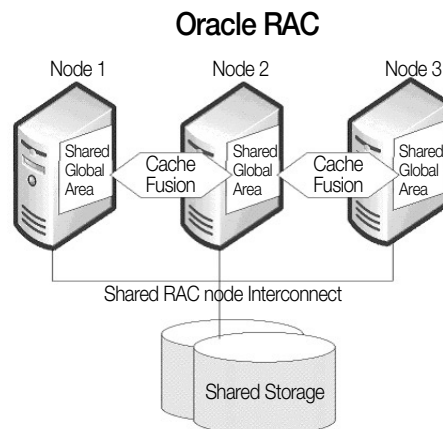


그림 7: Oracle RAC

노드 오류 발생 시 시스템 내 잠금이 다시 마스터(remaster)되고 RAC 노드가 다시 동기화(resync)되는 동안 클라이언트 연결이 일시 중단되는 짧은 기간이 있습니다. Oracle RAC는 공유 디스크 아키텍처를 사용하기 때문에 디스크 오류로부터 보호하려면 Enterprise Edition에서만 제공되는 Oracle의 Data Guard 기능을 사용해야 합니다.

Oracle RAC 시스템은 가용성이 높은 클라이언트 액세스를 위해 다음 2가지 연결 장애 조치 방법을 제공합니다.

- **연결 장애 조치** 처음 연결하는 동안 연결 오류가 발생하면 응용 프로그램은 동일한 가상 서버 이름을 사용해 클러스터에 있는 또 다른 활성 노드로의 연결을 재시도할 수 있습니다.
- **TAF(Transparent Application Failover)** 연결이 설정된 후 통신 오류가 발생하면 해당 연결은 다른 활성 노드로 장애 조치를 취할 수 있습니다. TAF는 현재 트랜잭션의 상태를 저장하기 때문에 연결 장애 조치보다 더 많은 시스템 오버헤드가 필요합니다. TAF를 사용하려면 응용 프로그램이 최신 OCI(Oracle Call Interface)를 사용하도록 수정되어야 하며 분실 세션 상태를 처리하는 코드를 포함해야 합니다. 그 뿐 아니라 업데이트 트랜잭션이 롤백되어야 하며 서버 상태 정보에는 장애 조치가 이루어지지 않습니다.

Windows Clustering과 매우 유사한 RAC 장애 조치를 이용하려면 클러스터 노드에 인스턴스 모니터링 또는 하트비트 메커니즘이 필요합니다. 이 노드 모니터 기능을 통해 RAC 클러스터는 장애 조치 중에 자원을 신속히 동기화할 수 있습니다. Oracle RAC는 신속한 서버측 장애 조치 기능을 제공합니다. 이 기능은 RAC(Real Application Clusters)의 동시 활성-활성(active-active) 아키텍처에 의해 실현됩니다. 즉, 여러 Oracle 인스턴스가 여러 노드에서 동시에 활성화되며 이들 인스턴스는 동일 데이터베이스에 대한 액세스를 동기화합니다. 또한 모든 노드는 모든 저장소에 대한 동시 소유권 및 액세스 권한을 가집니다. 한 노드에 장애가 생기는 경우에도 클러스터 내 다른 모든 노드들은 여전히 저장소에 액세스할 수 있습니다. 전달할 디스크 소유권이 없으며 데이터베이스 서버 코드는 이미 메모리에 로드되어 있습니다. 장애 조치에 이어 RAC 노드를 동기화하는 프로세스는 먼저 클러스터에서 장애가 생긴 노드를 순서대로 제거한 다음 해당 노드가 소유했던 자원의 제어 임무를 대신 맡는 식으로 진행됩니다. 장애 조치가 완료된 후 모든 진행 중인 쿼리는 처음부터 다시 실행됩니다.

Oracle 10g Data Guard

SQL Server 2005 Database Mirroring과 매우 유사한 Oracle의 Data Guard 기능은 프로덕션 데이터베이스 트랜잭션 로그 데이터를 사용해 대기 서버에서 프로덕션 데이터베이스의 일관된 복사본을 유지 관리합니다. Data Guard 기능은 프로덕션 서버에서 서버 오류가 발생한 경우 대기 데이터베이스를 프로덕션 데이터베이스로 자동으로 전환할 수 있습니다. Oracle의 Data Guard 기능은 프로덕션 데이터베이스의 서로 다른 백업 복사본을 최대 9개까지 유지 관리할 수 있습니다. Data Guard는 다음 3가지 모드 중 하나로 실행됩니다.

- **최대 보호** 최대 보호 모드에서 데이터는 주 데이터베이스에서 대기 데이터베이스로 동기적으로 전송됩니다. 대기 데이터베이스에서 재실행 데이터를 사용할 수 있을 때까지 주 데이터베이스에서 트랜잭션이 커밋되지 않습니다. 재실행 데이터가 대기 서버에 기록될 수 없는 경우 주 서버에서는 처리가 중단됩니다.
- **최대 가용성** 최대 가용성 모드는 최대 보호 모드와 유사하게 작동합니다. 그러나 재실행 데이터가 첫 번째 대기 서버에 기록되자마자 주 데이터베이스에서 처리가 계속됩니다. 백업 서버를 사용할 수 없는 경우에도 주 서버의 처리가 중단되지 않습니다.
- **최대 성능** 최대 성능 모드에서 재실행 데이터가 비동기적으로 대기 데이터베이스로 전달되는 동안, 주 데이터베이스는 트랜잭션을 계속 처리합니다. 대기 서버가 재실행 데이터 수순을 승인할 때까지 기다리지 않고 주 데이터베이스에서 트랜잭션이 커밋됩니다.

참고

Data Guard는 Oracle Enterprise Edition에서만 제공됩니다.

결론

고가용성은 저절로 이루어지는 것이 아니라 인력, 프로세스 및 기술의 조합이 한층 강화되었을 때 실현되는 결과입니다. 기술에만 초점을 맞춘 계획은 매우 높은 수준의 가용성을 달성하지는 못합니다. 왜냐하면 가용성에 영향을 미치는 여러 중요한 요인들이 인력과 프로세스의 상호 작용에서 비롯되기 때문입니다. 적합한 하드웨어 및 소프트웨어 플랫폼을 준비하는 것은 출발점에 불과합니다. 고가용성은 이와 더불어 훌륭한 계획 및 실행 방안과 적합한 기술이 결합된 결과입니다.

고가용성은 특정 기술의 출현 때문이 아니라 비즈니스 요구 사항으로 인해 그 필요성이 더욱 높아지고 있습니다. 고가용성 환경을 구축하는 것은 항상 바람직한 요건이지만, 필요로 하는 가용성 수준이 높아질수록 관련 비용도 높아진다는 사실을 유념해야 합니다. 따라서, 비즈니스가 필요로 하는 가용성 수준을 제대로 이해하는 것이 정말 중요합니다. SQL Server 2005 및 Oracle 10g는 모두 매우 높은 수준의 가용성을 제공하기 위해 사용할 수 있는 기능을 갖추고 있습니다. 그러나 이 2가지 데이터베이스 플랫폼의 모든 기능이 비용과 사용 용이성 면에서 동일하지는 않습니다. Microsoft SQL Server 2005는 Oracle 10g보다 복잡하지 않으며 저렴한 가격으로 엔터프라이즈급 고가용성 기능을 기업에 제공합니다.

부록 A-고가용성의 장애 요소

인적 장애 요소

어떤 환경에서든 시스템 정지 시간을 야기하는 가장 큰 원인 중의 하나는 인적 오류입니다. 인적 오류로부터 신속하게 복구할 수 있다면 데이터베이스 가용성 요건에서 우위를 선점할 수 있습니다. 데이빗 패터슨(David Patterson)이 시행한 연구 조사(A Simple Way to Estimate Downtime)에 따르면 시스템 정지 시간의 53%는 인적 오류의 결과입니다. "Disaster Recovery Journal"에 게재된 자료와 같은 기타 연구 조사에 따르면 인적 오류가 조직에서 발생하는 데이터 손실의 36%를 차지한다는 결과가 밝혀진 바 있습니다. 인적 장애 요소를 극복하는 것은 보다 높은 가용성을 실현하기 위한 가장 중요한 수단 중의 하나임이 틀림없습니다. 운영자의 오류는 단 몇 분만에 데이터베이스를 손상시킬 수 있지만, 데이터를 복구하거나 복원하려면 몇 시간 또는 몇 일이 걸릴 수 있습니다. 복구 시간은 많은 비용을 유발하지만 방지할 수 있습니다.

사람들은 실수를 저지르지만 시스템 정지 시간을 최소화하고 이러한 오류로부터 복구하기 위해 사전 대처적으로 취할 수 있는 방법들이 있습니다. 인적 오류의 주요 2가지 원천은 사용자 오류와 운영자 오류입니다. 만일 허용된다면 사용자는 부주의하게 중요한 데이터를 삭제한 다거나 데이터베이스를 잘못된 정보로 업데이트한다거나 하여 기업 정보를 마구 파괴할 수 있습니다. 교육, 적절한 응용 프로그램 문서 작성, 정기적인 프로시저 확립은 사용자 오류를 방지하는 최적의 방어 수단입니다. 사용자 오류로 인한 시스템 정지 시간을 단축시키는 가장 중요한 방법 중 하나는 각 사용자의 데이터 및 서비스 액세스를 반드시 필요한 항목으로 제한하는 것입니다.



운영자 또는 응용 프로그램 개발자 오류는 데이터베이스 및 응용 프로그램 가용성에 엄청난 영향을 미칠 수도 있습니다. 예를 들어, 데이터베이스에서 부정확한 테이블을 삭제하거나 데이터베이스에 부정확한 데이터를 기록시키는 오류를 코딩하면 응용 프로그램 가용성에 부정적인 영향을 미칠 수 있습니다. 이러한 유형의 오류를 방지하는 훌륭한 방법은 직원의 인식 수준을 높이는 것입니다. 특히 상급 관리인들은 지속적인 정보 가용성에 관한 복잡성 및 책임에 대해 인식 수준을 높여야 합니다. 즉, 교육 예산을 늘리고, 운영 지침을 개발하기 위해 시간과 자원을 투자할 뿐만 아니라 장애 복구 계획을 개발 및 구현해야 합니다.

데이터베이스 관리자(DBA)는 점차 데이터베이스 무결성을 유지 관리해야 하는 어려움에 직면하고 있습니다. DBA는 전체 개발 프로세스의 모든 단계에 참여해야 하는 경우가 많습니다. 아키텍처 및 데이터베이스 설계, 응용 프로그램 개발, 데이터베이스 관리, 장애 및 복구 시나리오 구현 등이 이러한 단계에 포함됩니다. DBA는 문제를 해결할 수 있도록 교육을 받아야 하며 데이터 복구 계획을 효율적으로 실행하기 위한 필수 도구를 확보해야 합니다.

프로세스 장애 요소

고가용성 환경을 구축하는 능력에 상당한 영향을 미칠 수 있는 또 다른 요인은 내부 프로세스입니다. 적절한 프로세스를 확립하면 불필요한 시스템 정지 시간을 제거할 수 있을 뿐만 아니라 서비스 중단 시 보다 신속한 복구를 수행할 수 있습니다.

고가용성에 대한 가장 중요한 장애 요소 중 하나로 문서화된 운영 프로시저의 부재를 꼽을 수 있습니다. 조직은 루틴 운영 작업을 수행하기 위해 작성된 프로시저를 개발할 뿐만 아니라 다양한 유형의 장애로부터 복구하는 데 필요한 방법을 문서화해야 합니다. 이 문서화된 프로시저를 일반적으로 런 북(run book)이라고 합니다. 적합한 문서가 결여되면 서비스 중단 시 부정확한 복구가 초래되고 필수 단계 또는 프로시저를 생략할 가능성이 커져 전체 복구를 구현하는 데 필요한 전체 시간이 증가됩니다. 이와 마찬가지로 루틴 운영 프로시저의 적절한 문서가 없다면, 특히 질병이나 책임의 재할당과 같은 기타 요인으로 인해 인력 변동이 있는 상황에서 운영자 오류가 발생할 가능성이 높아집니다. 운영 프로시저가 없다면 문제 해결 기능도 저하됩니다. 표준화된 프로시저가 없는 경우, 사람들마다 다양한 작업을 제각각 수행하는 경향이 있기 때문에 특정한 상황을 초래하게 된 일련의 단계를 정확히 알아내는 것이 어렵습니다. 효과적인 런 북을 이용하면 신입 DBA라도 숙련된 팀 구성원처럼 효과적으로 루틴 업무를 수행할 수 있습니다.

적합한 프로세스의 결여가 가용성에 부정적인 영향을 미칠 수 있는 또 다른 영역은 부적절한 문제 해결 문서입니다. 표준 문제 해결 프로시저를 확립하면 운영 직원들이 일반적인 문제 시나리오를 파악하고 다양한 상황을 보다 신속하게 진단 및 해결할 수 있도록 지원할 수 있습니다. 표준화된 문제 관리 프로시저가 없는 경우, 데스크 및 운영 인력이 일반적인 문제 상황이 나타날 때마다 해결 방법을 다시 개발해야 합니다. 이렇게 되면 알려진 상황으로부터 복구하는 시간이 길어지며 문제를 부정확하게 진단할 가능성 역시 높아집니다. 궁극적으로, 시스템 정지 시간이 증가하고 다른 문제까지 시작될 수 있습니다.

또한 부적절한 변경 관리 프로시저는 고가용성에 중대한 장애 요소가 될 수 있습니다. 변경 관리 프로시저를 통해 조직은 응용 프로그램 수정 동안 발생하는 응용 프로그램 및 데이터베이스 스키마 변경을 모두 추적할 수 있습니다. 소스 코드 및 데이터베이스 스키마 변경을 추적하기 위한 표준 메커니즘을 제공하는 것 이외에도, 적절한 변경 관리 프로시저를 개발하면 변경 사항을 프로덕션 환경에 적용하기 전에 QA(quality assurance) 설정에서 테스트할 수 있는 품질 보증 환경을 구축할 수 있습니다. 변경 관리 프로시저가 없다면 최신 업데이트를 포함하지 못한 이후 변경에 의해 데이터베이스 스키마 및 응용 프로그램 업데이트가 손실 또는 무시될 수 있는 전체 복구 오류가 초래될 수 있습니다. 이와 마찬가지로 QA 환경의 부재로 응용 프로그램을 사용할 수 없게 되는 응용 프로그램 및 데이터베이스 배포 오류가 발생할 수 있습니다.

고가용성에 대한 그 외 2가지 프로세스 장애 요소는 표준화된 하드웨어와 소프트웨어 구성의 결여입니다. 가능할 때마다 데이터 센터에서 사용되는 하드웨어 및 소프트웨어 구성을 모두 표준화합니다. 하드웨어 구성 요소가 표준화되면 하드웨어 오류 발생 시 시스템 수리 또는 교체를 더욱 쉽게 수행할 수 있습니다. 마찬가지로, 소프트웨어 구성이 표준화되면 루틴 작업이 쉬워지고 운영자 오류 가능성이 줄어듭니다. 예를 들어, 가능한 경우 모든 서버는 표준 명명 구성표를 사용해야 하며 표준화된 드라이브 문자, 매핑된 디렉터리 및 공유 이름을 가져야 합니다. 또한 모든 데이터베이스 서버는 운영 체제, 데이터베이스 및 미들웨어 데이터 액세스 레이어에 대해 동일한 수준의 서비스 팩으로 실행되어야 합니다.

표준 하드웨어 및 소프트웨어 구성이 없는 경우, 가능한 오류 지점이 증가되어 문제 해결 시간이 더욱 길어지고 운영 및 복구 오류를 일으킬 가능성이 높아집니다.

고가용성 환경 구축을 방해할 수 있는 마지막 프로세서 장애 요소는 부정확하거나 오래된 기업의 정보입니다. 모든 IT 직원을 위한 정기적인 교육 일정을 마련하면 조직의 기술력을 최신 상태로 유지하도록 보장할 수 있을 뿐 아니라 조직이 가장 효과적인 기술 및 도구를 채택할 수 있는 가능성이 높아집니다..

기술 장애 요소

기술 부문에서는 최고의 가용성 수준을 실현하기 위해 몇 가지 문제를 해결해야 합니다. 서버 단위의 거의 모든 구성 요소에 하드웨어 오류가 발생할 수 있습니다. 응용 프로그램 오류가 데이터베이스 액세스에 영향을 미칠 수도 있습니다. 데이터가 손상될 경우 정보를 복구하기 위해 적절한 데이터베이스 복구 메커니즘이 배치되어야 합니다. 또한 일정이 예정된 하드웨어 업그레이드 및 데이터베이스 유지 관리는 시스템 가용성을 감소시킬 수 있는 요인들입니다. 사용 가능한 데이터베이스 기술을 활용하면 계획된 유지 관리와 관련된 시스템 정지 시간을 줄이거나 없앨 수 있습니다. 마지막으로, 인프라 오류 또는 사이트 장애는 데이터베이스 가용성에 상당한 영향을 미칠 수 있습니다.

인프라 오류

오늘날의 데이터베이스 응용 프로그램은 클라이언트 컴퓨터를 위해서는 물론 응용 프로그램 서버를 데이터베이스에 연결하고 많은 경우 여러 데이터베이스 서버를 함께 연결하기 위해 네트워크 연결을 활용하고 있습니다. 네트워크 인프라 오류는 구현된 데이터베이스 플랫폼에 관계 없이 이러한 서로 다른 데이터베이스 계층의 가용성에 부정적인 영향을 미칠 수 있습니다.. 네트워크 인프라 오류는 DNS 서버를 비롯한 수많은 네트워킹 구성 요소, 응용 프로그램 오류, 그리고 스위치, 허브, 라우터 및 NIC와 같은 장치에서의 네트워크 하드웨어 오류에 의해 야기될 수 있습니다.

하드웨어 오류에 대처하는 것은 이러한 문제 중에서 가장 간단합니다. 다중 자원 액세스 경로 생성은 네트워크 하드웨어 오류 가능성을 해결하는 데 있어 열쇠입니다. 데이터베이스 서버 수준에서 데이터베이스 및 응용 프로그램 서버 모두에 중복 네트워크 카드를 장착하여 네트워크 인프라 오류에 대비함으로써 NIC에 장애가 생기더라도 서버에 대한 네트워크 연결이 계속 사용 가능하도록 보장할 수 있습니다. 하나의 NIC가 잘못되더라도 네트워크는 계속 작동하기 때문에 하드웨어 오류 이벤트 메시지를 확인하고 수리하려면 시스템 이벤트 로그의 정기적인 모니터링이 필요합니다.



데이터 센터 수준에서는, 네트워크 구성 요소 또는 네트워크 세그먼트의 오류로 인해 야기되는 네트워크 중단에 대비하기 위해 다중 라우터 및 스위치를 설치함으로써 데이터베이스 서버에 대한 다중 자원 액세스 경로를 생성할 수 있습니다. 데이터베이스 서버에 대한 모든 네트워크 연결이 단일 스위치를 통해 경로 지정되고 해당 스위치에 장애가 생기는 경우 데이터베이스 서버와 응용 프로그램이 문제 없이 실행되고 있다 하더라도 데이터베이스는 사용할 수 없게 됩니다. 스위치 같은 중복 네트워크 장치를 구현하고 각 장치가 자체 UPS를 보유하도록 하면 데이터베이스에 대한 다중 네트워크 경로를 설정할 수 있습니다. 그러면 네트워크 하드웨어 오류가 시스템 가용성에 부정적인 영향을 미치지 않게 할 수 있습니다.

NLB(Network Load Balancing) 클러스터를 설치하면 응용 프로그램 오류 또는 웹 서버 오류로 인해 야기되는 인프라 오류로부터 네트워크 인프라를 보호할 수 있습니다. NLB 클러스터는 확장성 뿐만 아니라 향상된 가용성을 제공합니다. NLB는 웹 서버가 데이터베이스 시스템에 대한 중복 및 확장 가능한 프런트 엔드를 생성하는 데 종종 사용됩니다. Windows NLB는 가상 서버 이름 및 IP 주소를 사용해 액세스되는 물리적 서버를 결합할 수 있도록 하는 기본 제공 Windows 서비스입니다. 클라이언트는 물리적 서버의 현재 자원 활용도와 같은 사전 구성된 일련의 기준에 따라 다양한 물리적 서버에 대한 라우팅 연결을 책임지고 있는 NLB 가상 서버에 연결합니다. NLB는 여러 물리적 서버 간에 수신 연결을 확산시킴으로써 확장성을 향상시킵니다. 또한 NLB는 NLB 클러스터에 포함되는 하나 이상의 장애가 생긴 물리적 서버를 피해 요청의 경로를 지정함으로써 네트워크 인프라 가용성을 향상시킵니다.

네트워크의 DNS는 데이터베이스 가용성에 상당한 영향을 미칠 수 있습니다. 네트워크 클라이언트 시스템 및 응용 프로그램 서버는 자원을 액세스하기 위해 네트워크에서 데이터베이스 서버를 찾을 수 있어야 합니다. 네트워크의 DNS 서버는 이 가상 네트워크 기능을 제공합니다. DNS 서버 오류가 발생하면 사용자는 응용 프로그램이 필요로 하는 데이터베이스 자원이 포함된 서버를 찾지 못할 수 있습니다. DNS 가용성을 향상시키는 최선의 방법은 네트워크에 다중 DNS 서버를 구현하는 것입니다. 이처럼, 1개의 DNS 서버에 장애가 발생하더라도 하나 이상의 보조 DNS 시스템에서 여전히 네트워크 이름 확인 기능을 제공할 수 있습니다.

네트워크 사용자가 네트워크 인증 서비스를 이용하도록 하려면 다중 도메인 컨트롤러가 필요합니다. 중복 도메인 컨트롤러를 통해 네트워크 사용자는 하나 이상의 도메인 컨트롤러가 사용할 수 없게 되더라도 네트워크의 인증을 계속 받을 수 있습니다.

사이트 장애

고가용성에 대한 마지막 장애 요소는 전체 사이트를 사용할 수 없게 되는 장애에 대비하는 것입니다. 사이트 장애에서 조직을 보호하는 단계는 모든 주요 엔터프라이즈 데이터베이스 플랫폼에서 유사합니다. 사이트 장애로부터 총체적인 보호를 제공하려면 중복 데이터 센터 설비를 구축해야 합니다. 중복 데이터 센터를 구축하면 주 데이터 센터의 작동을 불가능하게 만드는 화재, 지진 및 기타 예측 불가능한 사건 등에 의해 야기되는 전체 사이트 중단 사태로부터 조직을 보호할 수 있습니다. 백업 데이터 센터가 주 데이터 센터의 미래 이미지여야 할 필요는 없지만 주 데이터 센터의 처리 요구 사항을 해결할 수 있도록 설계되어야 합니다. 많은 경우, 주 데이터 센터는 미래의 확장을 처리할 수 있는 용량을 갖추도록 설계됩니다. 백업 데이터 센터는 이 초과 용량을 필요로 하지 않지만 주 데이터 센터의 최대 현재 워크로드를 처리할 수 있어야 합니다.

백업 데이터 센터를 구현하는 경우 주 데이터 센터는 물론 백업 데이터 센터를 위해 여러 인터넷 통신 회사를 구현하는 것이 중요합니다. 이를 통해 대규모 지역에 영향을 미치는 전기 배선 손실과 같은 사건이나 인터넷 통신 회사 중 하나의 서버를 중단시키는 사건이 발생한 경우에도 서비스 가용성이 보장됩니다. 한 통신 회사가 서비스를 손실하면 대체 통신 회사를 사용하여 데이터베이스 서비스에 계속해서 액세스할 수 있습니다. 중복 데이터 센터 구현은 분명히 막대한 비용을 감수해야 하는 해결책이지만 많은 기업들에게 있어서 중복 데이터 센터와 인터넷 통신 회사를 구현하는 데 필요한 비용은 오랜 시스템 정지 시간으로 인해 야기되는 비용을 능가하는 가치가 있습니다.

서버 가용성의 장애 요소를 해결하는 여러 엔터프라이즈 데이터베이스 기술은 사이트 오류로부터 조직을 보호할 수도 있습니다. 예를 들어 데이터베이스 복제 기능 및 로그 전달 기능은 지리적으로 분산된 하나 이상의 사이트에서 프로덕션 데이터베이스를 미러링하는 데 사용할 수 있습니다.

사이트 장애 요소를 극복하는 또 다른 도구는 IP 기반 SAN 복제입니다. IP 기반 SAN 데이터 복제는 로컬 SAN 저장소 시스템을 미러링하는 원격 SAN 디스크 저장소 시스템을 구축 및 유지 관리함으로써 디스크 중복을 실현할 수 있습니다. IP는 데이터 전송 프로토콜이기 때문에 로컬 및 원격 SAN은 지리적으로 아주 먼 거리에 분산될 수 있으며 인터넷 또는 전용 IP 기반 네트워크 설비를 통해 로컬 트랜잭션을 원격 SAN에 전송할 수 있습니다. 주 사이트에 장애가 발생하는 경우 원격 SAN이 미러링된 데이터 복사본을 보유하고 있으면 데이터 손실은 전혀 없습니다.

필자 소개

마이클 오테이(Michael Otey)는 Windows IT Pro Magazine의 기술 책임자이며, SQL Server Magazine의 수석 기술 에디터(Senior Technical Editor)입니다. 또한 그는 상호 운용성 및 데이터베이스 응용 프로그램 전문 소프트웨어 개발 및 컨설팅 기업인 TECA, Inc.의 사장입니다. 마이클은 Osborne McGraw Hill에서 출판한 SQL Server 2005 New Features Guide의 저자이기도 합니다.

데니얼 오테이(Denielle Otey)는 TECA, Inc.의 부사장이며 C, Visual C++(r), Visual Basic(r) 및 Visual Studio(r) .NET으로 소프트웨어를 설계, 구현 테스트 및 디버깅하는 폭 넓은 경험을 보유한 소프트웨어 컨설턴트입니다. 데니얼은 일부 SQL Server 유틸리티의 개발자이자 Osborne McGraw Hill이 출판한 ADO.NET The Complete Reference의 공동 저자입니다. 본 자료는 A23 Consulting과의 협력 하에 작성되었습니다.

문서 번호, SQL-200509-18

Microsoft®

한국마이크로소프트(유)

서울특별시 강남구 대치동 892번지 포스코센터 | 전화 : 080-985-2000 | 인터넷 : www.microsoft.com/korea