

Microsoft®
SQL Server™ 2005

Microsoft®

SQL Server 2005
Enterprise Edition으로
더욱 강력해지는 기업솔루션

Microsoft®

한국마이크로소프트(유)

서울특별시 강남구 대치동 892번지 포스코센터 서관 5층 | 고객센터 : 1577-9700 | 인터넷 : <http://www.microsoft.com/korea/sql>

Microsoft®
SQL Server™ 2005

Microsoft® SQL Server™ 2005

SQL Server 2005 Enterprise Edition 으로 더욱 강력해지는 기업 솔루션

SQL Server 기술 문서

작성자: Lara Rubbelke, Information Management Service Line Leader, Diginer

참조 도움: Tom Abraham; David Baldauff; Paul Doyle; Gordon D'Souza; Teresa Nowak;
Stan Sajous; Jason Strate; Nelson Williams

초판: 2007 년 2 월

대상: SQL Server 2005 Service Pack 2

요약: Microsoft SQL Server 2005 Enterprise Edition 의 강력한 기능은 기업의 가장 많이 사용되는, 대형의, 미션 크리티컬 분석용 데이터베이스 애플리케이션을 지원하는 가장 완벽한 플랫폼을 제공한다. SQL Server 2005 Enterprise Edition 의 고급 기능이 기업 데이터베이스와 분석용 애플리케이션의 성능, 가용성, 확장성과 관리편의성을 어떻게 높여주는 지에 대하여 확인해보도록 하자.



Copyright

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, email address, logo, person, place or event is intended or should be inferred.

© 2007 Microsoft Corporation. All rights reserved.

Microsoft, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

목차

Copyright	2
소개	5
데이터 관리	5
비즈니스 인텔리전스	6
데이터 관리 기능	9
가용성	9
데이터베이스 미러링: 소개	9
데이터베이스 미러링: 비동기식	10
데이터베이스 미러링: 병렬 실행취소(Parallel Redo)	11
신속한 복구	11
데이터베이스 스냅샷	12
순간 파일 초기화(Instant File Initialization)	14
온라인 페이지와 파일 복원	15
지연 트랜잭션(Deferred Transactions)	16
Highly Available Indexes: 간단한 소개	16
Highly Available Indexes: 온라인 인덱스 작업	17
Highly Available Indexes: 병렬 인덱스 작업(Parallel Index Operations)	19
장애극복 클러스터링	20
확장성과 성능	22
테이블과 인덱스 분할	22
64 비트 지원	25
프로세서 지원	26
메모리: 소개	26
메모리: 순간 메모리 추가(Hot-add) 지원	27
메모리: Dynamic AWE 메모리 관리	27
다중 인스턴스 지원	27
확장성 높은 공유 데이터베이스	28
인덱싱된 뷰(Indexed Views)	29
병렬 DBCC (Database Console Commands)	29
업데이트가능한 분산 분할 뷰	30
고급 스캐닝(Advanced Scanning)	31
VARDECIMAL 저장 형식	32
관리편의성과 보안	33
로그온 트리거	33
일반 조건 호환성 설정(Common Criteria Compliance)	33
백업 미디어 세트 미러링	34

VIA 지원.....	36
복제.....	36
피어 투 피어 트랜잭션 복제	36
오라클 게시(Publishing)	37
비즈니스 인텔리전스기능	39
통합.....	39
유사항목 조회와 유사항목 그룹화 변환	39
용어 추출과 용어 조회 변환	41
차원과 파티션 처리 어댑터	42
SQL Server Integration Services 의 데이터 마이닝 기능 향상	44
분석	44
자동관리 캐싱.....	45
분할된 큐브	46
큐브 뷰(Perspectives)	47
반가산성 측정값(Semiadditive Measures).....	48
계정 인텔리전스(Account Intelligence).....	49
전체 쓰기저장 지원	50
메타데이터 번역	50
데이터베이스 간/서버 간 연결된 차원과 측정값	52
압축된 XML 과 이진 XML 전송	52
Analysis Services 다중 인스턴스 지원.....	53
리포팅	54
검색가능한(Clickthrough) 비정규 보고서	54
데이터 중심의 구독	55
Reporting Services 스케일 아웃 전개	56
데이터 마이닝	57
시계열 교차 예측	57
데이터 마이닝 알고리즘 확장성.....	58
데이터 마이닝 모델의 예측과 처리 능력 개선	58
무제한의 동시 데이터 마이닝 질의	59
연결 규칙을 위한 무제한의 특성.....	59
Notification Services	60
Notification Services 스케일 아웃 전개	60
멀티캐스트 배달	61
조정가능한 알림 일괄처리 크기.....	62
인스턴스 당 증가된 배포자와 생성자 쓰레드	62
결론	63

소개

Microsoft® SQL Server™ 2005 Enterprise Edition 은 가장 중요한 대형 미션 크리티컬 OLTP 와 분석용 데이터베이스 애플리케이션을 완벽하게 지원하는 플랫폼이다.

All-in-One 데이터베이스 관리 플랫폼인 SQL Server 2005 Enterprise Edition 은 엔터프라이즈급 대형 데이터베이스 환경에 필요한 안정성, 확장성, 성능과 보안 기능을 모두 제공한다. 기업은 다재다능한 기능을 사용해서, 서버 구성을 다양하게 구현할 수 있기 때문에 IT 인프라를 제대로 활용할 수 있다. 하나의 큰 데이터 웨어하우스 또는 여러 데이터베이스를 사용하는 통합 서버 모두에 대해, SQL Server 2005 Enterprise Edition 은 최상의 가용성, 확장성과 성능을 제공한다.

비즈니스 인텔리전스 기능이 통합된 Enterprise Edition 은 신뢰성 높은 엔드 투 엔드 비즈니스 인텔리전스와 분석 애플리케이션을 만들 수 있다. SQL Server 2005 Enterprise Edition 은 대규모이며 복잡한 데이터 통합, 분석과 보고서 작성 환경에 이상적이다. SQL Server 2005 Enterprise Edition 을 통해, 기존의 비즈니스 인텔리전스를 뛰어 넘어 궁극의 데이터 통합, 실시간 분석과 내용이 알찬 보고서를 만들어 낼 수 있다.

데이터 관리

가용성

SQL Server 2005 Enterprise Edition 은 이전과 비교할 수 없는 최고의 데이터베이스 가용성을 제공한다.

데이터베이스 미러링(mirroring), 피어 투 피어 복제(peer-to-peer replication), 향상된 장애극복 클러스터링 지원과 같이 새로운 기능으로 언제나 사용할 수 있는 애플리케이션을 만들고 전개할 수 있다. 불가피한 유지보수, 튜닝과 복원 작업 중에도 순간 파일 초기화(instant file initialization), 데이터베이스 스냅샷, 온라인과 병렬 인덱스 작업, 온라인 복원 작업과 같은 기능을 통해 계획된 다운타임을 최소화시킨다.

- **비동기식 데이터베이스 미러링**
비동기식 데이터베이스 미러링을 통해 애플리케이션 성능 저하 없이 데이터베이스를 또 하나 만들 수 있다.
- **피어 투 피어 복제**
피어 투 피어 복제 기능을 사용해서, 구성 데이터베이스 중 어느 하나에 장애가 발생하거나 유지보수를 할 경우에도 높은 가용성을 제공한다.
- **8 노드 까지의 클러스터링 지원**
8 노드 까지 지원하기 때문에 유연하고 비용 효과적인 클러스터링 시나리오를 설계하고 구현할 수 있다.
- **데이터베이스 스냅샷**
데이터베이스 스냅샷을 사용해서 사용자 또는 관리자 실수에서 신속하게 복원합니다. 데이터베이스 미러링으로 데이터베이스 스냅샷이 만들어지면, 사용자는 다른 서버로 리포팅을 옮겨 원(principal) 서버의 자원을 다른 작업에 사용할 수 있다.
- **온라인 인덱스와 병렬 인덱스 작업**
온라인 인덱스 최적화를 통해 동시성(concurrency)에 악영향을 미치지 않으면서 인덱스가 최적의 상태를 유지할 수 있게 한다. 또한 병렬 인덱스 작업을 통해 인덱스를 만들고 다시 빌드하는 시간을 줄인다.
- **온라인 복원**
다른 데이터베이스는 사용자에게 서비스를 제공하는 동안, 한 데이터 페이지 또는 파일그룹을 복원한다.

확장성과 성능

SQL Server 2005 Enterprise Edition 은 대규모 엔터프라이즈에서의 Online Transaction Processing (OLTP), 복잡한 데이터 분석, 데이터 웨어하우징 시스템과 웹 사이트를 지원하는 데 필요한 고성능을 제공하도록 확장된다. Enterprise Edition 은 스케일 업과 스케일 아웃 중 적합한 것을 필요에 따라 선택해서 모든 유형의 애플리케이션 시나리오를 지원할 수 있다.

Windows Server™ 2003 x64 Enterprise Edition 에서는 64 개의 CPU 와 8 테라바이트의 메모리까지 확장된다. 실제로는, 메모리 순간 추가(hot-add) 기능을 사용해서 기존의 데이터베이스를 확장시킬 수 있다. 테이블 분할(table partitioning), 병렬 DBCC 와 인덱싱된 뷰(indexed views)와 같은 기능을 통해 대형 프로젝션과 데이터 웨어하우징 데이터베이스에 도입한 하드웨어의 성능을 높이고 투자효과를 극대화시킨다.

- **테이블과 인덱스 분할**

테이블과 인덱스 분할은 단일 테이블의 무결성을 그대로 유지하면서 큰 테이블을 더 작고 관리하기 쉬운 데이터 집합으로 나눈다. 테이블과 인덱스 분할은 질의 성능을 크게 향상시키고 매우 큰 테이블을 관리하는 경우의 다운타임을 줄여준다.

- **질의를 위한 인덱싱된 뷰**

SQL Server Enterprise Edition 질의 최적화 프로그램(query optimize)은 질의에서 뷰를 직접 참조하지 않는 경우에도 질의 실행에 있어서 인덱싱된 뷰를 사용할 수 있는 지를 자동으로 판단한다.

- **다중 프로세서 지원**

다중 프로세서를 지원해서 DBCC 명령에 대해 데이터를 신속하고도 효율적으로 검증한다.

- **VARDECIMAL 저장 형식**

VARDECIMAL 저장 형식을 사용해서, 데이터 웨어하우스의 공간 사용을 크게 줄여준다.

Enterprise Edition 은 모든 유형의 데이터베이스 애플리케이션을 지원하는 다양한 스케일 아웃 기능들을 제공한다. 확장성 높은 공유 데이터베이스, 피어 투 피어 복제, 분산 분할 뷰(distributed partitioned views)와 같은 Enterprise Edition 기능을 일반 범용 하드웨어에서 사용할 수 있기 때문에 애플리케이션 사용 환경에 따라 비용효과적으로 스케일 아웃할 수 있다.

- **확장성 높은 공유 데이터베이스**

확장성 높은 공유 데이터베이스를 사용해서 리포팅 데이터베이스를 스케일 아웃한다.

- **피어 투 피어 복제**

피어 투 피어 복제를 통해, 질의를 여러 노드에 분산시켜서 애플리케이션의 성능을 높인다.

- **업데이트 가능한 분산 분할 뷰(Updateable distributed partitioned views)**

업데이트 가능한 분산 분할 뷰를 사용해서 트랜잭션 데이터베이스를 여러 서버 간에 투명하게 스케일 아웃한다.

관리와 보안

Microsoft SQL Server 2005 는 보안에서도 업계 최고 수준을 자랑한다. "secure by default" 설정, 데이터베이스 암호화, 향상된 보안 모델과 같이 개선된 기능은 기업 데이터를 가장 높은 수준으로 보호해준다. SQL Server 2005 Service Pack 2 의 새로운 Enterprise Edition 기능인 로그인 트리거(Logon triggers)와 일반 조건 호환성(common criteria compliance) 옵션은 일반적인 기업 표준에 따라 보안 모델을 더 좋게 정의하고, 조정하고, 구현할 수 있는 기능을 제공한다.

비즈니스 인텔리전스

통합

Integration Services 는 데이터 웨어하우징의 추출, 변형과 로딩(ETL) 작업 등의 고성능의 데이터 통합과 워크플로우 솔루션을 구축할 수 있는 플랫폼이다. SQL Server Integration Services (SSIS)의 Enterprise Edition 은 데이터를 발견하고 데이터를 클렌징하고, 실시간에

가까운 Analysis Services 환경을 유지할 수 있는 혁신적인 어댑터와 변환기능으로 기존의 ETL 이상의 서비스를 제공한다.

- **유사항목 조회(Fuzzy Lookup)와 유사항목 그룹화 변환(Fuzzy Grouping transformations)**
유사항목 조회와 유사항목 그룹화 변환을 사용해서 일반적인 데이터 품질 문제를 해결한다. 데이터 표준화, 데이터 수정, 별개 시스템의 데이터 일치, 없는 값 채우기, 중복 레코드 제거와 같은 데이터 클렌징 작업을 쉽고도 안정적으로 수행한다.
- **용어 추출(Term Extraction)과 용어 조회 변형(Term Lookup transformations)**
용어 추출과 용어 조회 변형은 데이터 변형 과정을 텍스트 마이닝 도구로 바꾸어준다. 문서와 텍스트 문자열을 분류하고 검색해서 비즈니스 프로세스를 간결하고 자동화될 수 있게 한다.
- **차원과 파티션 처리 어댑터(Dimension and Partition Processing adapters)**
차원과 파티션 처리 어댑터는 Analysis Services 큐브를 최신 데이터로 업데이트하는 속도를 높여주고 실시간에 가까운 분석 환경을 제공한다.

분석

SQL Server 2005 Analysis Services 는 전통적인 OLAP 기반의 분석과 관계형 기반의 리포팅의 각각의 장점을 결합해서 하나 이상의 물리적 데이터 소스에서 만들어진 단일 데이터 모델을 제공하고 있다. OLAP, 리포팅과 사용자 지정 비즈니스 인텔리전스 애플리케이션을 사용하는 모든 일반 사용자 질의는 관계형 데이터에 대해 단일 비즈니스 뷰를 제공하는 Unified Dimensional Model (UDM)을 통해 기본 데이터 소스의 데이터에 접속한다.

Enterprise Edition 의 SQL Server Analysis Services 는 더 크고 더 복잡한 환경을 지원할 수 있는 가장 완벽한 플랫폼이며 쉽게 구성되는 환경으로 실시간 분석 기능을 제공한다.

- **자동관리 캐싱(Proactive caching)**
자동관리 캐싱은 실시간의, 높은 성능과 가용성의 분석환경을 제공한다.
- **분할된 큐브(Partitioned cubes)**
큐브의 다중 파티션은 더 쉽게 질의하고 처리할 수 있으며, 여러 프로세서, 드라이브와 서버에 질의 또는 처리 부하를 분산할 수 있는 기능을 제공한다.
- **큐브 뷰(Perspectives)**
큐브 뷰는 일반 사용자가 필요로 하는 정보를 간결하게 보여주고 큐브를 쉽게 탐색할 수 있게 한다.
- **차원과 파티션 쓰기저장(Dimension and partition writeback)**
차원과 파티션 쓰기저장은 큐브에 데이터를 업데이트하고 데이터 변경 내용의 영향을 분석해서 what-if 시나리오를 시작할 수 있게 해준다.

리포팅

Microsoft SQL Server Reporting Services 는 기업이 중요한 기업 데이터를 공유 정보로 변형해서 더 낮은 총소유 비용으로 적시에 통찰력있는 의사결정을 할 수 있게 해준다. 기존의 정적인 리포트의 한계를 넘어 직관적이고, 비정규의, 웹 기반의 리포팅 플랫폼이 된다.

- **심층분석 리포트**
심층분석 리포트는 사용자가 보고서를 탐색해서 미처 알지 못했던 관계를 발견하고, 어려운 의문에 대한 답을 발견하고, 비즈니스에 대한 새롭고 상세한 내용을 발견할 수 있게 한다.
- **데이터 중심의 구독**
데이터 중심의 구독은 대규모의 구독을 더 잘 관리하는 동시에 수신자에게 의미있고 사용자 고유의 데이터를 전달한다.

데이터 마이닝

SQL Server 2005 데이터 마이닝은 비즈니스에 대해 귀중한 통찰력을 제공해서 사용자가 이벤트의 원인을 이해하고 향후에 발생할 이벤트를 예측할 수 있게 한다. Enterprise Edition 은

매우 다양한 구성과 설정을 지원하는 확장성 높은 플랫폼으로 사용자의 비즈니스 패턴에 따라 데이터 마이닝 모델의 성능, 정확도와 산출물을 높여준다.

데이터 관리 기능

Microsoft® SQL Server™ 2005 Enterprise Edition 은 가장 중요한 애플리케이션을 위한 이상적인 기업용 플랫폼으로, 가장 많이 요구되는 가용성, 확장성, 성능과 보안 요구사항을 지원하는 다양한 기능을 제공한다.

가용성

확장성과 성능

관리편의성과 보안

복제

가용성

SQL Server 2005 Enterprise Edition 은 가장 많이 요구되는 데이터베이스 가용성 요구사항을 제공하는 최고의 능력을 제공한다.

데이터베이스 미러링, 피어 투 피어 복제, 향상된 장애극복 클러스터링 지원과 같은 새로운 기능으로 고 가용성의 애플리케이션을 구축하고 전개한다. 데이터베이스 스냅샷, 온라인과 병렬 인덱스 작업, 온라인 복원 작업과 같은 Enterprise Edition 기능은 필수적인 유지보수, 튜닝과 복구 프로세스에 의한 계획된 다운타임을 줄인다.

여기에서는 다음과 같은 내용을 설명하고 있다:

데이터베이스 미러링

데이터베이스 스냅샷

순간 파일 초기화

온라인 페이지와 파일 복원

자연 트랜잭션

데이터베이스 미러링: 소개

SQL Server 2005 Service Pack 1 부터 제공된 데이터베이스 미러링은 높은 가용성, 장애극복, 대기 솔루션을 제공한다. 데이터베이스 미러링은 장애극복 클러스터링과 복제를 포함한 기존의 높은 가용성을 제공하는 솔루션을 이상적으로 대체하거나 보완해준다.

SQL Server 2005 Standard Edition 에서도 다양한 데이터베이스 미러링이 제공되지만, SQL Server 2005 Enterprise Edition 만이 대규모 데이터 웨어하우스와 하이엔드 OLTP 데이터베이스 애플리케이션의 데이터베이스 미러링 요구사항을 지원하는 완벽한 옵션을 제공한다. Enterprise Edition 은 데이터베이스 애플리케이션에 대한 데이터베이스 미러링의 모든 악영향을 줄이거나 없애주며 극도의 가용성을 제공해야 하는 데이터베이스 애플리케이션에게 가장 빠른 장애극복 능력을 보장한다.

데이터베이스 미러링은 SAFETY FULL 의 보호 우선(high-safety), SAFETY OFF 의 보호 우선, 고성능이라는 세 가지 모드를 지원한다. 비동기식 데이터베이스 미러링이라고도 부르는 고성능 모드는 Enterprise Edition 에서만 제공된다.

보호 우선 모드는 SQL Server 2005 의 Standard 와 Enterprise Edition 에서 제공되는 동기식 데이터베이스 미러링 형식으로, 보호 우선 모드는 미러링 모니터를 통해 (SAFETY FULL) 자동으로 탐지하거나 장애극복을 하도록 구성하거나 미러링 모니터 없이 (SAFETY OFF) 수작업 장애극복을 지원하도록 구성될 수 있다. 보호우선 모드는 어떤 데이터 손실도 허용되지 않으며 애플리케이션 성능이 영향을 받아도 되는 환경에서 이상적이다.

보호우선 모드의 동기식 성격은 데이터베이스 미러링에 있어서 궁극의 데이터 보호를 하지만 애플리케이션 성능을 저하시키기도 한다. 주 서버가 주 데이터베이스에 트랜잭션을 기록하는 경우, 미러 서버에 로그 기록을 즉시 전송한다. 주 서버는 클라이언트에게 트랜잭션 완료를 알리기 전에

미러 서버의 승인을 기다려야 한다. 그 동안, 미러 서버는 주 서버에 승인을 보내기 전에 미러 로그에 트랜잭션을 기록하게 되고, 승인을 받은 주 서버는 완료를 클라이언트에게 알린다. 데이터베이스 미러링에서 보호우선 모드를 구현하게 되면 기존 애플리케이션의 성능에도 영향을 미친다. 성능에 대한 영향의 정도는 네트워크 대역폭, 주 서버에서 전송된 트랜잭션 수, 미러 서버의 I/O 디스크 서브시스템 등의 다양한 요소에 따라 달라진다.

데이터베이스 미러링: 비동기식

애플리케이션 성능에 영향을 거의 미치지 않으면서도 데이터 중복을 제공하는 고가용성 솔루션이 필요하다면, SQL Server 2005 Enterprise Edition 의 데이터베이스 미러링을 고성능 모드에서 사용하도록 한다. 고성능 데이터베이스 미러링은 주 서버와 미러 서버가 상당히 멀리 떨어져있거나 데이터베이스 애플리케이션이 많은 트랜잭션 양을 처리할 경우의 재난 복구 시나리오에서 효과를 볼 수 있다.

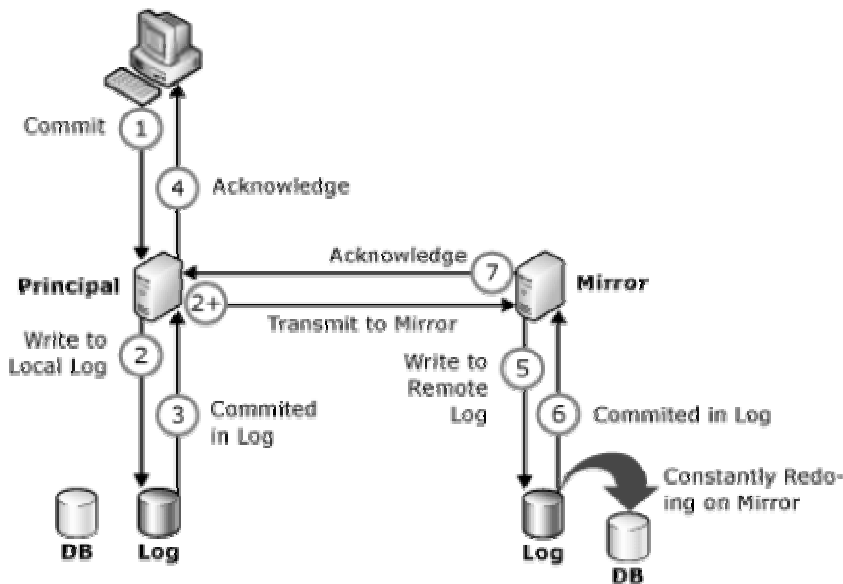


그림 1 고성능 데이터베이스 미러

그림 1의 고성능 모드에서는, 트랜잭션이 미러 데이터베이스에 비동기식으로 기록된다. 주 서버는 로컬 로그에 기록을 하고 로그 레코드를 미러 서버로 전송한다. 주 서버는 미러 서버가 로그를 디스크에 기록하고 승인을 반환하는 것을 기다리지 않고도 클라이언트에 완료 확인을 전송한다. 미러 서버는 주 서버에서 전송된 최신 로그 레코드를 유지하려고 하지만, 미러는 주 데이터베이스에 비해 조금 늦을 수도 있다. 그렇지만 일반적으로는 데이터베이스 간의 격차는 작다. 고성능 모드의 중요한 요소는 미러 서버에서의 작업이 애플리케이션 성능에 영향을 주지 않는다는 것이다.

장점

데이터베이스 미러링은 가장 많이 요구되는 가용성 요구사항을 충족시키는 이상적인 재해 복구 솔루션이다. 비동기식 데이터베이스 미러링은 장애극복 시간이 매우 짧은 데이터베이스 중복성을 제공하지만 보호우선 모드를 사용할 경우에 경험할 수도 있는 성능 문제도 피할 수 있다.

고성능 데이터베이스 미러링은 로그 전달(log shipping)과 장애극복 클러스터링에 비해 여러 가지 장점을 제공한다. 데이터베이스 미러링은 이런 기술과 함께 사용될 수도 있다.

- 데이터베이스 미러링은 로그 전달과 장애극복 클러스터링보다 훨씬 쉽게 구성하고 유지보수할 수 있으며 이런 기술들에 대해 선호되는 대안기술 또는 보완기술이다.
- 데이터베이스 미러링은 장애극복 클러스터링 기능이 아닌 데이터베이스 중복성을 제공한다.

- 데이터베이스 미러링은 고성능 모드에서도, 로그 전달에 비해 데이터 손실 가능성이 낮다
비동기식 고성능 모드는 주 서버가 일부 있을 수도 있는 데이터 손실 위험이 있지만 최소한의 트랜잭션 대기 시간으로 실행할 수 있게 한다. Database Mirroring Monitor 를 사용해서 미러링 성능 값의 한계 값을 정하고 경고하며 동적 관리 뷰를 사용해서 다양한 작업시간 동안 데이터 손실 위험을 파악할 수 있다.

데이터베이스 미러링: 병렬 실행취소(Parallel Redo)

데이터베이스가 미러로 장애 조치를 할 경우에는 매 순간이 중요하다. 데이터베이스의 장애 조치를 하려면 미러(실행취소 대기열)에 아직 적용되지 않은 모든 로그 레코드를 롤 포워드(rolling forward)해야 한다. 데이터베이스의 장애극복 시간은 미러 서버가 실행취소 대기열의 로그를 얼마나 빨리 롤 포워드하는 지에 따라 달라진다. 다시 말해, 시스템 하드웨어와 현재의 작업부하에 따라 크게 달라진다. SQL Server 2005 Enterprise Edition 은 실행취소 단계 중에는 다중 스레드를 사용해서 고급 하드웨어를 활용할 수 있게 한다.

장애극복 중의 롤 포워드 시간을 측정하는 방법은 실행취소 단계 중에 미러 서버가 사용하는 스레드 수에 따라 달라진다.

스레드의 수는 SQL Server 에디션과 CPU 수에 따라 달라진다:

- SQL Server Standard Edition 에서는, 미러 서버는 항상 단일 스레드를 사용해서 데이터베이스를 롤 포워드한다.
- SQL Server 2005 Enterprise Edition 에서는, 5 개의 CPU 미만을 가진 컴퓨터의 미러 서버는 단일 스레드만을 사용한다.
- 5 개 이상의 CPU 를 가진, SQL Server 2005 Enterprise Edition 미러 서버는 장애 극복 중에 다중 스레드 간에 롤 포워드 작업을 분산시키며(병렬 실행취소) 각 4 개의 CPU 에 대해 한 개의 스레드를 사용하도록 최적화된다. 로그 레코드를 병렬로 롤 포워드할 수 있기 때문에 미러 서버의 장애극복 시간은 개선된다.

장점

실행취소 단계에서 다중 스레드를 활용하기 때문에 SQL Server Enterprise Edition 은 대형 데이터베이스를 순식간에 장애 극복 조치를 하고 복원 중에 데이터베이스를 사용하지 못하는 시간을 줄인다. 이 기능은 특히 대형의 미션 크리티컬 데이터베이스 애플리케이션과 가용성이 중요해서 장애극복을 빨리 해야 하는 OLTP 데이터베이스 애플리케이션을 미러링할 때에 특히 도움이 된다.

신속한 복구

Microsoft SQL Server 2005 Enterprise Edition 은 신속한 복구(fast recovery) 기능을 도입하여, 시스템 장애 또는 데이터베이스 미러링 장애극복으로부터 데이터베이스를 복구하는 가용성을 크게 개선하였다. 신속한 복구는 실행취소 단계 중에도 데이터베이스를 사용할 수 있게 해준다.

SQL Server 시스템 장애와 데이터베이스 미러 장애극복은 여러 단계로 진행된다. 첫 번째 단계는 로그된 트랜잭션을 적용시켜서 데이터를 롤 포워드하는 것으로 다시실행(Redo) 단계로 부른다. 다시실행 단계로 모든 로그 트랜잭션을 롤 포워드하게 되면, 데이터베이스는 일반적으로 트랜잭션에서 커밋되지 않은 변경내용을 가지게 된다. 복구 프로세스가 실행취소 단계에서 커밋되지 않은 모든 트랜잭션을 롤백하게 된다. SQL Server 2005 Enterprise Edition 사용자는 이 실행취소 단계(그림 2 참조)에서도 데이터베이스를 사용할 수 있다.

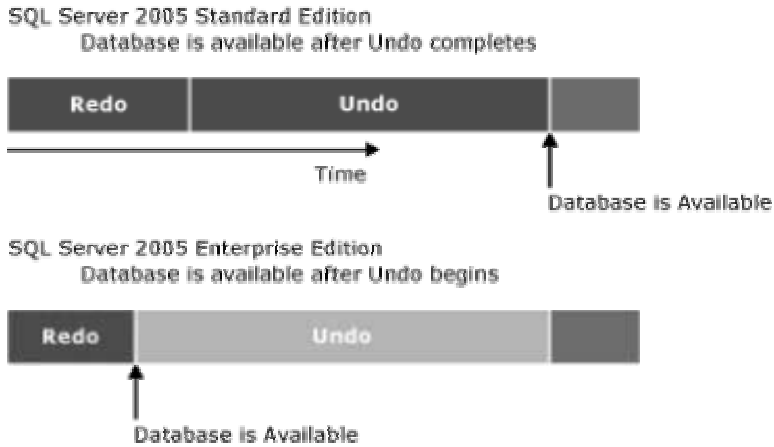


그림 2 에디션 별 복구시간

시스템 장애 또는 장애극복이 발생되었을 때에 커밋되지 않은 트랜잭션이 장애 이전에 가지고 있었던 모든 락을 다시 가지게 되기 때문에 신속한 복구가 가능하다.¹ 이런 트랜잭션이 롤백되는 동안, 사용자가 작업에 영향을 주지 못하도록 잠긴다.

장점

신속한 복구는 실행취소 단계에서도 사용자가 데이터베이스를 이용할 수 있기 때문에, 데이터베이스 서비스 중단 시간을 최소한으로 줄인다. SQL Server와 SQL Server 2005 Standard Edition 이전 버전은 실행취소 단계가 완료되기 전에는 데이터베이스를 이용할 수가 없었다.

데이터베이스 스냅샷

데이터베이스 스냅샷은 SQL Server 2005 Enterprise Edition에서만 제공되는 기능이 풍부한 다목적 기능이다. 데이터베이스 스냅샷은 전체 데이터베이스를 복사하는 작업부하없이 데이터베이스의 읽기전용의 정적 복사본을 효과적으로 제공한다. 초대형 데이터베이스의 경우, 백업을 하고 복원을 해서 특정 시점의 데이터베이스 복사본을 유지하는 것은 많은 시간이 걸리고 막대한 디스크 공간을 사용할 수도 있다. 스냅샷은 데이터베이스 크기와 상관없이 거의 순간적으로 만들어지고 사용할 수 있다. 그리고, 스냅샷을 가지고 있는 파일은 스냅샷 생성 이후에 수정된 데이터만을 가지게 된다.

장점

데이터베이스 스냅샷은 확장성과 관리편의성을 제공하는 다양한 기능들을 제공한다:

- **사용자 또는 애플리케이션 오류로부터 데이터를 보호**

정기적으로 데이터베이스 스냅샷을 만들게 되면, 테이블 삭제와 같은 중대한 사용자 실수로부터 보호받을 수 있다. 더 높은 수준의 보호를 위해서, 일련의 데이터베이스 스냅샷을 만들어서 대부분의 사용자 실수를 인식하고 대처할 수도 있다. 예를 들면, 디스크 자원에 따라, 24 시간 간격으로 6~12 개의 스냅샷을 유지할 수도 있다. 그 후에 새 스냅샷이 만들어질 때마다 가장 처음에 만들어진 스냅샷이 삭제된다.

사용자 실수에서 복구하려면, 실수 바로 직전의 스냅샷으로 데이터베이스를 되돌린다. 되돌리기는 백업에서 복구하는 것보다 훨씬 더 쉬운 작업이다.

¹ 이것은 지연 트랜잭션에 사용되는 것과 같은 로직이다. 지연 트랜잭션에 대한 더 자세한 내용은 이 기술백서의 뒤 부분에 있는 지연 트랜잭션을 참조한다.

다른 방법으로는, 스냅샷의 정보에 따라 삭제된 테이블 또는 다른 데이터를 수작업으로 재구성하는 방법이 있다. 예를 들면, 스냅샷의 데이터를 데이터베이스로 벌크 복사를 하고 수작업으로 데이터를 데이터베이스에 병합시킬 수도 있다.

- **관리 실수로부터 데이터를 보호**
벌크 업데이트와 같이 중요한 업데이트를 하기 전에, 데이터베이스의 스냅샷을 만들면 데이터를 보호할 수 있다. 실수를 하더라도, 스냅샷을 사용해서 데이터베이스를 되돌릴 수 있기 때문에 바로 복구를 할 수 있다.
- **미러 데이터베이스를 사용해서 리포팅 작업 분산**
데이터베이스 미러링과 함께 데이터베이스 스냅샷을 사용하게 되면 미러 서버의 데이터를 리포팅 작업에 사용할 수 있다. 이 밖에, 미러 데이터베이스에서 질의를 실행하여 주 서버의 자원을 여유있게 만들 수 있다. 미러 데이터베이스는 동기화된 상태가 되어야 데이터베이스 스냅샷을 이용할 수 있다.
- **보고서 생성을 위한 연혁 데이터 유지**
데이터베이스 스냅샷은 데이터베이스에 대해 정적인 뷰를 제공하기 때문에, 스냅샷은 특정 시점의 데이터를 접속하도록 확장사용될 수 있다. 예를 들면, 회계 분기와 같은 특정 기간 말에 데이터베이스 스냅샷을 만들어서 나중에 스냅샷에 대해 기간말 보고서를 실행해서 리포팅 용으로 사용할 수 있다. 디스크 공간이 충분하다면, 기간말 스냅샷을 시간제한없이 보관해서, 이 기간의 결과에 대해 질의를 해서 기업의 성과를 조사할 수도 있다.
- **테스트 데이터베이스 관리**
테스트 프로토콜을 반복적으로 실행하는 테스트 환경에서는, 데이터베이스가 각각의 테스트를 시작할 때에 동일한 데이터를 가지는 것이 좋다. 첫 번째 테스트를 실행하기 전에, 애플리케이션 개발자 또는 테스터가 테스트 데이터베이스에 대해 데이터베이스 스냅샷을 만든다. 테스트가 진행된 후에는, 데이터베이스를 스냅샷으로 되돌려서 이전 상태로 빠르게 복구할 수 있게 된다.

소스 데이터베이스에 페이지가 업데이트되면, 원래 데이터 페이지가 스냅샷에 기록된다(그림 3 참조). 이 과정을 쓰기과 복사 동시작업(copy-on-write operation)이라고 한다. 수정된 페이지의 레코드에 대한 이후 업데이트는 스냅샷의 콘텐츠에 영향을 주지 않는다. 처음으로 수정되고 있는 모든 페이지에 대해 같은 프로세스가 반복된다. 이런 식으로, 스냅샷이 만들어진 후에 수정된 모든 데이터 레코드에 대해 원래의 페이지를 스냅샷이 유지한다.

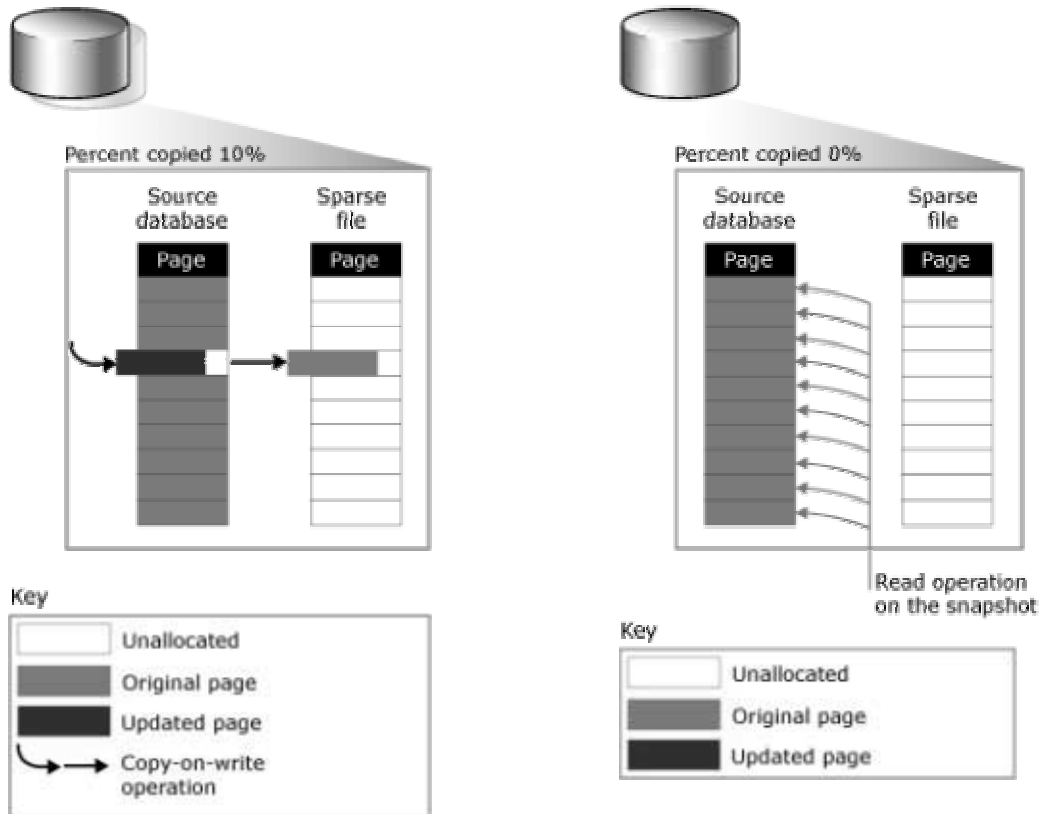


그림 3 데이터베이스 스냅샷의 쓰기와 읽기 프로세스

사용자에게는 데이터베이스 스냅샷이 전혀 바뀌지 않는 것처럼 보이는데, 이것은 데이터베이스 스냅샷에 대한 읽기 작업이 항상 원래의 데이터 페이지를 접속하기 때문이다. 페이지가 소스 데이터베이스에 아직 업데이트되지 않았다면, 스냅샷에 대한 읽기 작업은 소스 데이터베이스의 원래 페이지를 읽는다. 어떤 페이지가 업데이트된 후에도, 스냅샷에 대한 읽기 작업은 여전히 스냅샷의 원래 페이지를 사용한다. 더 많은 데이터 페이지가 데이터베이스 스냅샷에 기록될 수록, 더 적은 읽기 락이 소스 데이터베이스에 대해 사용되기 때문에 소스의 동시성(concurrency)이 향상된다.

순간 파일 초기화(Instant File Initialization)

SQL Server 2005 Enterprise Edition 은 중요한 데이터베이스 유지보수 작업 중에도 최고의 가용성을 제공한다. Enterprise Edition 은 큰 데이터 파일을 만들거나 추가하고 데이터베이스 복구작업을 수행하는데 필요한 시간을 크게 줄여준다.

데이터베이스가 만들어져서, 추가 공간과 추가 파일로 확대되거나 백업에서 복원되면, 새로 할당된 공간은 초기화되어야 한다. Standard Edition 에서는 초기화 과정 중에 디스크가 0 으로 기록되어야 하는데 큰 데이터 파일에서는 시간이 많이 걸리는 작업이 될 수 있다.

SQL Server 2005 Enterprise Edition 데이터 파일은 즉시 초기화될 수 있다. 순간 파일 초기화는 공간을 0 으로 채우지 않고도 사용된 디스크 공간을 회수할 수 있다. 대신에 디스크 콘텐츠는 파일에 기록된 새로운 데이터로 덮어쓰워진다.

SQL Server 2005 Enterprise Edition 에서 이 기능을 활용하려면, Windows Server 2003 또는 Microsoft Windows® XP 가 설치되어 있어야 한다. 그리고, SQL Server 2005 서비스 계정이 Perform Volume Maintenance Tasks 보안 정책에 추가되어 있어야 한다. 이런 조건이 충족되면, SQL Server 2005 Enterprise Edition 은 새 파일이 추가되고, 기존 파일이 확대되거나

데이터베이스 복원이 실행될 때마다 항상 순간 파일 초기화를 사용할 것이다. 이 기능은 1806 추적 플래그를 사용해서 해제될 수 있다.

순간 파일 초기화는 데이터 파일에 대해서만 사용될 수 있다. 로그 파일 초기화는 여전히 0 으로 채워넣기가 필요하다.

장점

순간 파일 초기화는 중요한 데이터베이스 유지보수 작업 중에도 최고의 데이터베이스 애플리케이션 가용성을 보장한다. 가장 많이 사용되는 데이터베이스 애플리케이션은 계획되거나 계획되지 않은 유지보수 작업의 경우에도 오랜 시간의 서비스 중단이 허용되지 않는다. 순간 파일 초기화는 데이터베이스 유지보수 작업시간을 줄여서 데이터베이스가 복원 작업을 훨씬 빨리 완료하게 하고, 데이터 파일 증가가 신속하게 완료되고, 새 데이터베이스가 순식간에 만들어지게 한다. 대부분의 경우, 이런 작업을 수행하느라 몇 분에서 몇 시간까지 걸리던 시간이 몇 초로 단축된다.

데이터베이스는 계속 증가하며, 계획된 확대일지라도 데이터베이스 가용성에 나쁜 영향을 미칠 수 있다. 데이터베이스 복원은 가장 적은 시간 동안 실행되어서 비즈니스를 계속 할 수 있어야 한다. 순간 파일 초기화는 사용자와 애플리케이션에 최소한의 영향만으로 데이터베이스 서비스를 재개할 수 있게 해준다.

온라인 페이지와 파일 복원

SQL Server 2005 Enterprise Edition에서는, 데이터베이스 관리자가 데이터베이스 전체의 서비스를 중단시키지 않고도 어느 특정 부분에 대해 복원 작업을 수행할 수 있다. 온라인 복원 옵션은 복원되고 있는 데이터만 사용할 수 없기 때문에 SQL Server의 가용성을 높여준다. SQL Server Enterprise Edition은 온라인 페이지 복원과 온라인 파일 복원 모두를 지원한다.

데이터 페이지는 SQL Server에서 가장 작은 데이터 저장 단위다. SQL Server가 한 개 이상의 데이터 페이지에 오류가 존재한다고 알리면, 사용자는 전체 데이터베이스를 복원하는 대신에 손상된 페이지만 복원할 수 있다. 대부분의 경우, 몇 개의 페이지를 복원하고 복구하는 것이 전체 데이터베이스 복원 또는 파일 복원보다 더 빠르고 복원 작업 중 사용할 수 없는 데이터 양도 적어진다.

페이지 복원은 온라인 또는 오프라인으로 진행될 수 있지만 Enterprise Edition 만이 온라인 페이지 복원을 지원한다. 대부분의 경우, 페이지가 복원되고 있는 파일그룹을 포함한 데이터베이스가 온라인 상태를 유지하면서도 손상된 페이지가 복원될 수 있다.

예를 들면, 하드웨어 장애로 어떤 데이터 페이지가 손상되면, 페이지 헤더에 저장된 체크섬 값이 페이지를 읽을 때에 계산된 체크섬과 일치하지 않는다는 것을 알려주는 손상 페이지 오류 메시지 824를 보게 된다.² **msdb..suspect_pages** 테이블을 확인하면, 몇 개의 페이지에만 복원작업이 필요하다는 것을 알게 된다. 가장 최근의 전체 백업에서 복원작업이 실행된다. 복원은 의심되는 페이지의 페이지 ID 값을 알려줄 것이다. 이후의 로그가 복원되어서 그 페이지가 현 상태가 되게 한다. SQL Server 2005 Enterprise Edition은 온라인 복원이 기본 값으로 설정되어서, 복원하고 있는 페이지는 사용할 수 없지만 데이터베이스의 나머지 부분은 사용자와 애플리케이션이 사용할 수 있게 해준다.

온라인 파일 복원은 여러 파일과 파일그룹을 가지고 있는 데이터베이스에 적합하다.

데이터베이스는 하나의 주 파일그룹과 여러 개의 부 파일그룹으로 구성될 수 있다. 이 경우는 여러 개의 디스크 드라이브에 데이터베이스는 분산시킬 때에 도움이 된다. SQL Server 2005는 데이터베이스 파일그룹이 단계별로 복원되는 점진적 복원 개념을 지원한다. 데이터베이스의 모든

² SQL Server 2005 데이터베이스는 페이지 일관성을 검증하고 손상된 페이지를 찾기 위해 CHECKSUM을 기본적으로 사용한다. 체크섬은 페이지 헤더에 기록되고 저장된 페이지 각각의 데이터에 대해 계산된다. 체크섬이 저장되어 있는 페이지를 읽어올 때마다 Database Engine은 페이지의 데이터에 대한 체크섬을 다시 계산하고 새로운 체크섬이 저장된 체크섬과 다른 경우에는 오류 824를 발생시킨다.

파일그룹을 복원하지 못했더라도, 복원 단계 마다 데이터베이스를 사용할 수 있게 된다. 복원 후에도 복원되지 못한 파일그룹은 오프라인 상태로 남는다. Enterprise Edition 을 사용하면, 앞에서 복원한 파일의 가용성을 해치지 않고도 부 파일들을 복원할 수 있다. 데이터베이스의 나머지 부분은 온라인 상태를 유지하고 서비스를 제공한다. 복원되고 있는 파일들만 오프라인 상태가 된다. 주 파일그룹의 어떤 파일이라도 복원한다면, 전체 데이터베이스는 사용할 수 없게 된다.

장점

SQL Server 2005 Enterprise Edition 은 시간이 오래 걸리는 복원 작업 중에도 데이터베이스의 가용성을 유지한다. 온라인 페이지와 파일 복원은 재난에서 더 빠르게 복구할 수 있는 능력을 제공하며, 중요한 애플리케이션에게 서비스를 계속 제공할 수 있게 한다. 온라인 복원은 특히 읽기 전용 데이터베이스에 적합하다. 예를 들면, 대형 데이터 웨어하우스의 복구를 점진적으로 수행해서, 관리자가 가장 최근의 데이터를 복구한 후에 시스템을 온라인 상태로 되돌리고, 파일 복원 작업을 계속할 수 있다. 연혁 데이터 복원 작업이 진행되는 동안에도 사용자들은 가장 최근의 데이터를 사용할 수 있다.

지연 트랜잭션(Deferred Transactions)

SQL Server 2005 Enterprise Edition 에서는, 데이터베이스 시작 중에 롤백(실행취소)에서 필요한 데이터를 사용할 수 없게 되면 트랜잭션은 이후로 지연될 수 있다. SQL Server 가 손상된 페이지를 만나게 될 경우에 데이터베이스의 일부를 사용하고 복구 중에도 가용성을 유지하게 된다.

지연 트랜잭션은 롤 포워드 단계가 완료되었을 때에 커밋되지 않고 롤백되지 못하는 오류를 만난 트랜잭션이다. 트랜잭션이 부분 데이터베이스 복원 또는 I/O 오류 때문에 오프라인 상태인 데이터를 사용할 수도 있다. 이런 문제로 트랜잭션 롤백이 안되기 때문에, 지연 상태가 된다. 지연 트랜잭션은 Enterprise Edition 에서만 지원된다. 다른 에디션들은 이런 상황을 만나게 되면 서버가 제대로 시작되지 않는다.

장점

치명적인 오류에 대해, 부분 데이터베이스 복원을 구현하고, 데이터베이스 미러로 장애극복 조치를 하게 되면 비즈니스의 연속성은 최소한만 지장을 받는다. Enterprise Edition 은 지연 트랜잭션에 대한 조치를 취하는 동시에 데이터베이스 애플리케이션에 계속 서비스를 제공할 수 있다.

Highly Available Indexes: 간단한 소개

많은 사용자들은 데이터베이스를 최적의 실행 상태로 계속 유지해야 하는 유지보수 문제의 어려움을 겪고 있다. 365 일 24 시간 내내 실행되어야 하는 애플리케이션을 위해 데이터베이스의 높은 가용성을 유지하는 동시에 매우 촉박한 시간 동안에 데이터베이스를 백업하고, 무결성을 확인하고, 인덱스 최적화를 해결해야 한다. SQL Server 2005 의 Enterprise Edition 은 가장 높은 가용성과 확장성을 제공하기 위하여 향상된 인덱스 작업을 도입하였다.

병렬 인덱스(Parallel index) 작업과 온라인 인덱스 작업은 SQL Server 2005 Enterprise Edition 의 두 가지 중요한 기능으로 CREATE, ALTER, DROP 인덱스 명령 실행 중에 인덱스의 가용성을 높여준다. 읽기와 쓰기 작업의 성능을 높이기 위해서는 인덱스 유지보수 작업이 자주 필요하다. 아마도 읽기 작업을 향상시키기 위해 단편화(fragmentation)를 줄이거나 쓰기 작업을 향상시키기 위해 인덱스 페이지에 대한 채우기 비율(fill factor)을 재설정할 것이다.

애플리케이션의 새 프로세스 또는 기존의 프로세스의 튜닝 요구사항을 지원하기 위해 새 인덱스를 추가해야 할 수도 있다. 그 이유가 무엇이던, 기업 애플리케이션은 이와 같은 다양한 인덱스 작업을 위한 빠르고 가용성이 높은 옵션을 필요로 한다.

SQL Server 2005 Standard Edition 의 클러스터형 인덱스를 변경하면, 테이블을 배타적(exclusive)으로 잠궈서 모든 동시 작업을 허용하지 않는다. 비클러스터형 인덱스를 업데이트하면, 테이블을 공유 잠금을 해서 읽기는 허용하지만 쓰기는 허용하지 않는다. 단일 쓰레드만 처리하도록 허용된다면, 큰 테이블에 대해 인덱스 작업을 수행하는 것은 데이터베이스 가용성에 큰 영향을 미친다. Enterprise Edition 을 사용하지 않는다면, 사용자의 옵션은 제한되고

인덱스 유지보수를 건너뛰게 되며 최적화되지 못한 데이터베이스 상태에서 작업을 하는 위험을 감수하게 된다.

SQL Server 2005 Enterprise Edition 은 중요한 인덱스 유지보수 작업 중에도 데이터를 읽고 쓸 수 있게 하는 ONLINE 인덱스 작업을 지원한다. SQL Server Enterprise Edition 만이 다중 프로세서 시스템을 완벽하게 활용해서 큰 데이터베이스에 대해 필수적이고 중요한 인덱스 작업의 안 좋은 영향을 줄일 수 있다. ONLINE 옵션과 병렬 인덱스 작업을 함께 사용하는 SQL Server Enterprise Edition 은 데이터베이스 가용성을 해치지 않고도 데이터베이스 성능을 관리하고 데이터베이스를 확대할 수 있는 최고의 선택이다.

Highly Available Indexes: 온라인 인덱스 작업

데이터베이스 유지보수는 미션 크리티칼 애플리케이션의 무난한 작업을 위해 필수적이지만, 유지보수 작업은 동시성 작업을 해치고 데이터베이스 애플리케이션의 전반적인 가용성에 영향을 미칠 수 있다. Standard Edition 에서는, 인덱스를 만들고 재구성하게 되면 이 작업이 끝날 때까지 테이블을 잠그고 SELECT, INSERT, UPDATE 작업도 못하는 경우가 많다. 애플리케이션이 하루 24 시간, 365 일 실행되어야 하는 경우에는 큰 테이블에 대해 클러스터형 인덱스를 재구축하는 것은 거의 불가능한 일이다. SQL Server 2005 Enterprise Edition 을 사용하면, ONLINE 옵션으로 인덱스를 만들고, 재구성하고, 삭제할 수 있다. 이 옵션으로, 인덱스 작업 중에도 동시 사용자가 기본 테이블 또는 클러스터형 인덱스 데이터 그리고 관련된 비클러스터형 인덱스를 사용할 수 있다.

ONLINE 옵션을 사용하면, SQL Server 는 인덱스에 대해 두 개의 복사본을 유지한다. 소스는 인덱스 작업 중에 사용자가 사용할 수 있는 원래의 인덱스 또는 테이블이다. 대상은 이면에서 인덱스를 재구성하기 위하여 SQL Server 가 사용하는 인덱스 또는 테이블의 읽기전용 복사본이다. 클러스터형 인덱스를 구성하는 것과 같이 간단한 온라인 인덱스 작업 중에는, 소스와 대상이 준비, 작성 (build)과 마무리(final) 세 단계를 거치게 된다. 짧은 준비와 마무리 단계 외에도, 사용자는 소스 테이블 또는 인덱스의 데이터를 계속 선택하고 변경하게 된다. 이 데이터는 SQL Server 2005 의 새로운 행 버전관리(row versioning) 기능을 사용해서 최신의 상태를 유지된다. 인덱스 작업 중에 테이블 또는 인덱스에서 발생하는 모든 데이터 변경은 원래 인덱스 또는 테이블 뿐만 아니라 대상 인덱스 또는 테이블의 스냅샷 모두에 적용된다. 이 작업이 완료되면, 인덱스 또는 테이블의 오래된 버전은 삭제되고 새 버전으로 교체된다.

그림 4 는 클러스터형 인덱스를 생성하는 온라인 프로세스를 보여준다. 각 단계 별로 소스와 대상 구조 작업이 동시 사용자 SELECT, INSERT, UPDATE, DELETE 연산과 함께 잘 설명되고 있다.

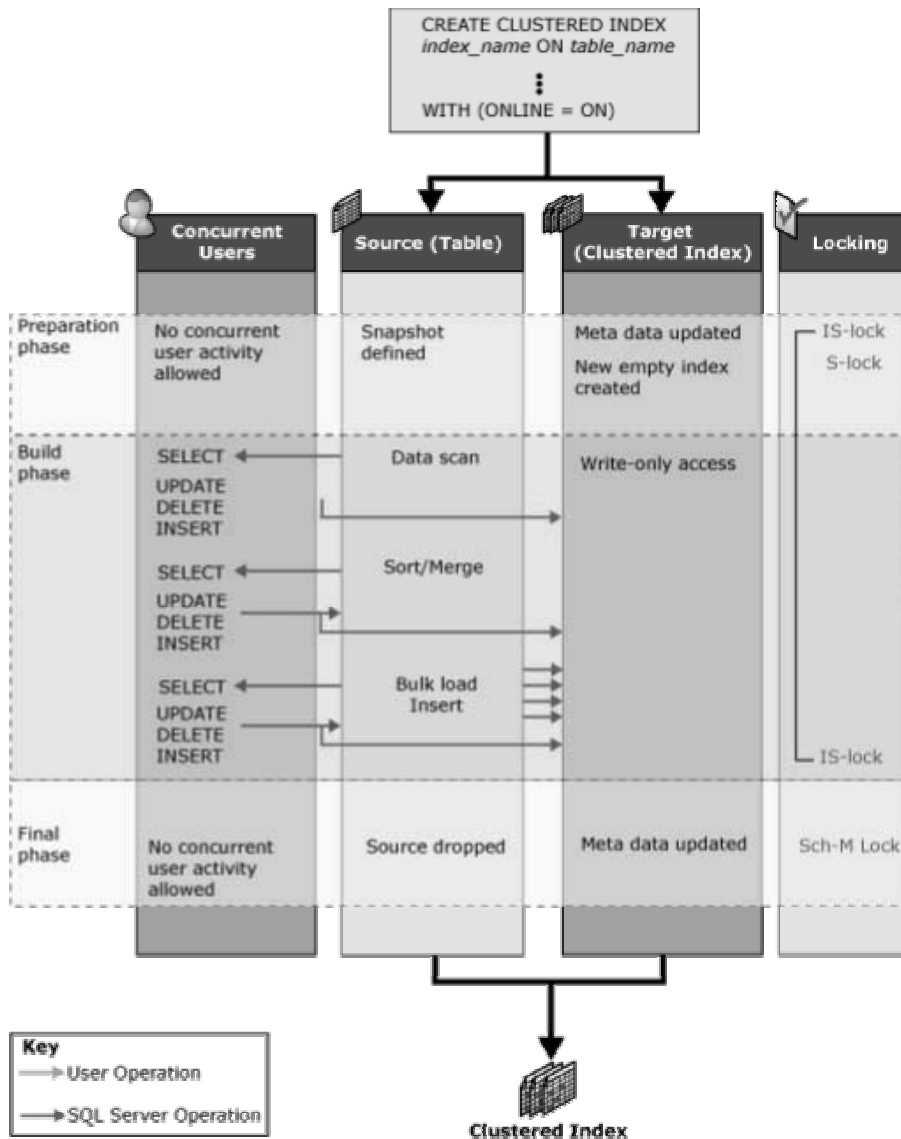


그림 4 온라인 인덱스 작업 프로세스

ONLINE 인덱스 작업은 가용성을 높이기 위한 것이지만 인덱스 작업의 성능을 높이기 위한 것이 아니다. ONLINE 옵션을 사용해서 인덱스를 생성하거나 재작성하게 되면 오프라인 인덱스 작업보다는 일반적으로 더 느리다.

장점

24 시간 365 일에 가까운 가용성을 제공해야 하는 모든 크기의 데이터베이스는 사용자의 업무를 중단시킬 정도의 다운타임없이 제대로 유지보수되고 최적화된 상태를 유지할 수 있다.

큰 인덱스에 대해 인덱스 작업을 완료하는데 얼마나 많은 시간이 걸릴지 계산하기 어렵다. 온라인 인덱스 작업의 또 다른 장점 중 하나는 SQL Server Profiler의 온라인 인덱스 작업 이벤트인 새로운 진행률 보고서(Progress Report)다(그림 5 참조). 온라인 인덱스 작업을 수행하게 되면, 이 새로운 이벤트는 온라인 작업의 상태를 추적한다. 이 이벤트를 통해, 프로세스를 완료하는데 필요한 시간을 보다 잘 관리할 수 있게 된다.

EventClass	EventSubClass	BigIntData1	StartTime	IndexID	Duration	SPID
Trace Start			2007-02-02 00:18:24...			
Progress Report: Online Index Operation	1 - Start		2007-02-02 00:18:28...	1		51
Progress Report: Online Index Operation	2 - Stage 1 execution begin		2007-02-02 00:18:28...			51
Progress Report: Online Index Operation	6 - Inserted row count	2535	2007-02-02 00:18:28...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	24999	2007-02-02 00:18:29...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	47463	2007-02-02 00:18:29...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	67431	2007-02-02 00:18:29...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	87399	2007-02-02 00:18:30...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	109863	2007-02-02 00:18:30...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	209863	2007-02-02 00:18:32...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	309863	2007-02-02 00:18:34...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	409863	2007-02-02 00:18:36...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	509863	2007-02-02 00:18:39...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	609863	2007-02-02 00:18:41...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	709863	2007-02-02 00:18:43...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	809863	2007-02-02 00:18:45...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	909863	2007-02-02 00:18:47...	1		51
Progress Report: Online Index Operation	6 - Inserted row count	1000000	2007-02-02 00:18:49...	1		51
Progress Report: Online Index Operation	3 - Stage 1 execution end		2007-02-02 00:18:49...			51
Progress Report: Online Index Operation	7 - Done		2007-02-02 00:18:31...	1	17562	51

그림 5 SQL Server Profiler 진행률 보고서(Progress Report)

몇 가지 디자인 전략으로 인덱스 단편화를 줄일 수 있지만, 시간이 지나면서 성능이 저하될 수 있다. 대부분의 데이터베이스는 인덱스를 추가하거나 업데이트하기 위해 또는 기존 인덱스의 단편화 또는 채우기 비율을 개선시키기 위해 어느 정도의 유지보수 작업이 반드시 필요하다. 이제는 이렇게 중요한 인덱스 작업을 어쩔 수 없이 생략하거나 인덱스를 추가하고 재조직하기 위해 다운타임을 계획할 필요가 없다.

Highly Available Indexes: 병렬 인덱스 작업(Parallel Index Operations)

SQL Server Enterprise Edition 만이 하드웨어 확장과 함께 스케일 업되어 병렬, 다중쓰레드 인덱스 작업을 지원한다. SQL Server 2005 Standard Edition 은 CREATE, ALTER, DROP 색인 작업에 대해 단일 스레드 질의 계획만 지원한다.

큰 테이블은 인덱스를 생성하거나 재작성하고 클러스터형 인덱스를 삭제하는데 오랜 시간일 걸릴 수도 있다. SQL Server 2005 의 Enterprise Edition 은 이 작업을 여러 개의 스레드로 나누어서 인덱스를 정렬하고 작성하는 시간이 오래 걸리는 단계들은 병렬로 처리한다. SQL Server 는 여러 개의 운영 체제 스레드를 사용해서 인덱스 작업을 병렬로 수행하기 때문에, 작업은 빠르고도 효율적으로 완료될 수 있다. 그리고 정기적인 인덱스 최적화와 튜닝 작업 중에도 업데이트된 클러스터형과 비 클러스터형 인덱스를 사용하지 못하는 시간이 줄어든다.

Database Engine 은 현재의 작업부하와 최대 병렬 처리 수준(max degree of parallelism) 구성에 따라 각각의 인덱스 명령에 얼마나 많은 프로세서를 사용할 것인지를 판단한다.

SQL Server 2005 Database Engine 은 시스템에 부하가 많은지를 판단하고, 명령 실행을 시작하기 전에 인덱스 작업에 대한 병렬 처리를 자동으로 줄인다.

최대 병렬 처리 수준 옵션은 인덱스 작업 명령에서 **server** 구성 값 또는 MAXDOP 중 한 방법을 사용해서 질의 계획 실행에 사용되는 프로세서 수를 제한한다.

최대 병렬 처리 수준 설정은 0 에서 서버의 최대 프로세서 수까지의 값 중 하나로 구성될 수 있다. 0 으로 설정되면, SQL Server 는 필요하다면 서버에서 사용할 수 있는 최대의 프로세서까지 사용할 수 있다.³ 서버가 1 보다 큰 최대 병렬 처리 수준 설정으로 구성되어 있고 Database Engine 이 서버에서 사용할 수 있는 리소스를 발견한다면, Database Engine 은 구성 값까지

³ 최대 병렬 처리 수준의 기본 값은 0 이다.

사용할 것이다. 이 구성 값을 1로 설정하게 되면 특정 명령에 대해 Database Engine 은 하나의 프로세서만을 사용하게 된다.

이전 버전의 SQL Server 에서는, 질의에서 MAXDOP 옵션을 사용해서 명령이 실행될 때의 프로세서 수를 조정했을 것이다. 이제 이 옵션은 인덱스 작업에서도 사용할 수 있게 되었다! 사용자들은 서버 차원에서 병렬 처리를 해제시키는 환경을 유지해야 하는 어려움을 겪으면서 병렬 인덱스 작업을 활용하고 싶어했다. MAXDOP 옵션은 각각의 명령이 실행될 때에 최대 병렬 처리 수준 서버 설정보다 우선하며 Database Engine 이 지정된 프로세서 수 이상을 사용하지 못하게 한다. 사용자는 이제 일반적인 질의작업에 대해서 서버에서 최대 병렬 처리 수준 설정을 구성하고 CREATE, ALTER, DROP 명령에서 직접 프로세서의 수를 줄이거나 늘릴 수 있게 되었다.

장점

인덱스 작업은 데이터베이스 애플리케이션의 신뢰성과 성능을 유지하고 높이기 위해 필요하다. 여러 쓰레드로 인덱스 작업을 처리하게 되면 일반적인 인덱스 작업의 실행 시간을 향상시켜서 사용자에게 대한 영향은 최소화시키는 동시에 새롭고 업데이트된 인덱스를 더 적은 시간으로 사용할 수 있게 된다.

큰 OLTP, 데이터 웨어하우스와 ODS 데이터베이스는 병렬 인덱스 작업으로 큰 효과를 볼 수 있다. OLTP 데이터베이스는 오랜 유지보수 작업을 할 수 없으며 가능한 한 짧은 시간에 인덱스 유지보수를 해야 한다. 웨어하우스 또는 ODS 를 로드하고 최적화시키는 유지보수 시간이 크게 제약을 받는 경우가 대부분이다. 인덱스가 더 짧은 시간에 업데이트되어 최적화된 성능을 사용자에게 제공할 수 있다면, 유지보수 작업을 일정대로 진행할 수 있을 것이다.

장애극복 클러스터링

가장 중요한 애플리케이션은 다운타임이 있어서는 안 된다. 대부분의 경우, 99.999% 가동시간이 일반적인 요구사항이다. Windows Server 2003 Clustering Services 과 SQL Server Enterprise Edition 은 기업 비즈니스의 중심이 되는 애플리케이션의 안정성과 가용성을 완벽하게 지원한다.

장애극복 클러스터링은 데이터베이스 서버 장애 발생 시에 빠르고, 투명하며, 자동으로 장애극복을 하는 고가용성의 솔루션이다. SQL Server 2005 에서는 장애극복 클러스터링이 SQL Server Analysis Services, Notification Services, SQL Server Replication 으로 확대되었으며 SQL Server 2005 Enterprise Edition 에서는 8 개 노드까지 지원하게 되었다.

	SQL Server 2005 Standard Edition	SQL Server 2005 Enterprise Edition
Windows 2000 Advanced Server (SP4)	2	2
Windows 2000 Datacenter Server (SP4)	2	4
Windows Server 2003 Enterprise Edition	2	8
Windows Server 2003 Datacenter Edition	2	8

표 1 SQL Server 2005 장애극복 클러스터링에서 지원되는 노드 수

장점

서버 클러스터에서 노드 수를 늘리게 되면, 관리자는 비즈니스 기대와 위험을 모두 관리하는 장애극복 정책을 지원하고 애플리케이션을 전개할 수 있는 더 많은 옵션을 가지게 된다. 더 큰 서버 클러스터가 지역적으로 여러 사이트에 분산된 클러스터를 구축해서 일반적인 노드 또는 애플리케이션의 장애뿐만 아니라 재난 극복을 할 수 있는 더 많은 유연성을 제공한다.⁴ 여러 사이트의 클러스터를 더 다양한 방법으로 관리할 수 있다. 예를 들면, 각 각의 사이트에 노드들을 가지게 되면, OS 또는 애플리케이션 소프트웨어 업그레이드를 전개하는 동안 다른 노드에서 장애가 발생하면 사이트 간에 장애극복 조치를 할 필요없이 업그레이드할 수 있다. 사이트, 서버 또는 디스크에 장애가 발생되면, 완벽한 시스템과 디스크 중복성 때문에 장애극복 클러스터가 동작하는 사이트에서 이후의 작업을 처리할 수 있게 된다. 이런 구성은 전형적인 장애극복 클러스터 구성에서의 단일 지점 장애라는 문제를 해결한다.⁵

더 큰 클러스터 구성은 또한 데이터 센터와 서버 관리 비용도 줄여준다. 그림 6의 구성 예 1이 보여주는, 전형적인 액티브/패시브 구성을 생각해보자. 이 모델은 분명히 하드웨어 중복성은 제공하지만 예산과 서버 활용도에 집중하는 현재의 데이터 센터에는 적합하지 않다. 전형적인 액티브/패시브 클러스터링은 한 서버가 서비스를 제공하지만 대기 서버는 아무 일도 하지 않으면서 실행 중인 서버에 장애가 발생할 경우만 기다린다. 이런 환경에서는, 3대의 서버가 나름대로 중요한 역할을 하고 있지만 제대로 활용되지 않으면서도 데이터 센터 자원을 필요로 한다. 각각의 물리적인 서버에 대해 서비스 팩, 핫픽스, 보안 업데이트 등을 적용하는 관리 비용을 지불하고 있다. 데이터 센터의 서버에 대해 감가상각 비용뿐만 아니라 온난방, 전력과 공간 비용도 고려를 해야 한다.

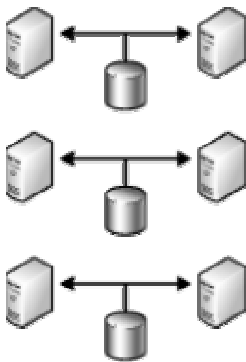


그림 6 구성 예 1: Standard Edition의 다중 2 노드 클러스터 구성

구성 예 2(그림 7)의 같은 환경은 필요한 장애극복 능력을 제공하면서도 서버의 총 숫자를 4 대로 줄인다. 이 예에서는, 3대의 서버가 실행 중이고 4 번째 서버가 다른 주 서버에 대해 동시 서비스를 할 수 있는 충분한 CPU와 메모리를 갖추도록 구성되어 있다.

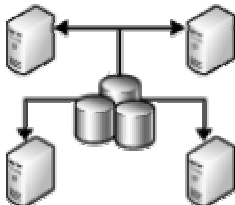


그림 7 구성 예 2: Enterprise Edition의 단일 4 노드 클러스터 구성

⁴ 더 자세한 내용은 [What's New in Clustering Technologies](#)를 참조.

⁵ 더 자세한 내용은 [Server Clusters : Geographically Dispersed Clusters](#)를 참조.

단일 인스턴스 액티브/패시브 구성에만 한정된다고 생각해서는 안 된다. SQL Server Enterprise Edition 에서는, 클러스터의 모든 서버가 안정성에 영향을 주지 않고도 액티브 상태를 유지할 수 있다. 모든 서버는 한 순간에 다른 서버로 자동으로 장애극복 조치를 해서, 가장 중요한 기업용 애플리케이션에 대해 하드웨어를 전부 활용하는 동시에 최상의 성능을 제공한다.

확장성과 성능

Enterprise Edition 은 대형 OLTP, 매우 복잡한 데이터 분석, 데이터 웨어하우징 시스템, 웹 사이트를 지원할 수 있는 최고의 성능 수준까지 확장된다. Enterprise Edition 만이 모든 유형의 애플리케이션 시나리오를 지원하도록 스케일 업과 스케일 아웃 능력을 선택할 수 있는 유연성을 제공한다.

여기에서는 다음과 같은 내용을 설명하고 있다:

테이블과 인덱스 분할

64 비트 지원

프로세서 지원

메모리

다중 인스턴스 지원

확장성 높은 공유 데이터베이스

인덱싱된 뷰

병렬 DBCC

업데이트 가능한 분산 분할 뷰(Updateable Distributed Partitioned Views)

고급 스캐닝(Advanced Scanning)

VARDECIMAL 저장 형식

테이블과 인덱스 분할

기업용 대형 데이터베이스 애플리케이션을 사용하는 경우에는, 높은 가용성과 적은 유지보수라는 모순에 직면하게 된다. 큰 테이블의 가용성, 확장성, 성능, 관리 편의성을 제공하기 위하여, Microsoft 는 SQL Server 2005 Enterprise Edition 에 테이블과 인덱스 분할 기능을 도입하였다. 테이블이 분할되면, 큰 테이블과 인덱스의 하위 데이터 집합을 빠르고도 효율적으로 관리하고 접속할 수 있는 동시에 단일 테이블의 데이터 무결성을 유지할 수 있게 된다. 분할된 테이블과 인덱스의 데이터는 관련 데이터의 더 작은 집합으로 나뉘어져서 데이터베이스의 여러 파일그룹에 분산된다. 질의나 업데이트를 할 경우, 테이블 또는 인덱스는 하나의 논리적 엔터티로 처리된다(그림 8 참조).

Orders table, 24 total partitions, 25 filegroups, 12 Logical disks (E:\throughP:\)

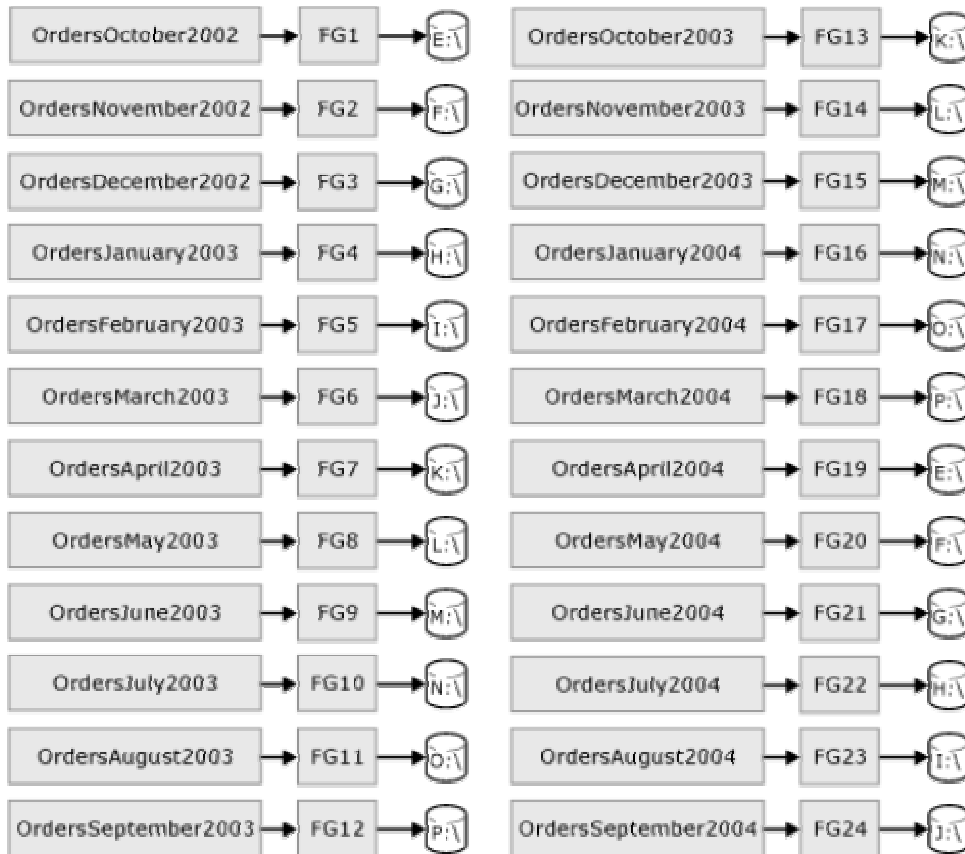


그림 8 분할 전략의 예

SQL Server 분할 뷰는 데이터를 더 작은 청크(chunks)로 구분하는데 도움이 되지만, 많은 물리적 기본 테이블, 데이터 무결성 제약조건과 인덱스를 관리하기 복잡해지고 물리적 데이터베이스 디자인을 제한하기도 한다. 이런 요소들 때문에 질의 최적화 시간에 큰 영향을 받을 수도 있다. SQL Server 2005 분할 테이블과 인덱스는 외래 키 제약조건, 기본 값, ID 와 타임스탬프 속성, 트리거를 포함해, 표준 테이블과 인덱스를 디자인하고 질의하는 것과 관련된 모든 속성과 기능을 지원한다.

테이블과 인덱스 분할은 파티션 함수(partition function)와 파티션 구성표(partition scheme)로 정의된다. 파티션 함수는 데이터를 하나의 열 또는 분할 키(partitioning key)에 따라 각각의 파티션에 매핑된 행 그룹으로 수평 분할된다. 파티션 구성표는 파일그룹 매핑에 따라 각 각의 파티션 저장위치를 정의한다. 집중적으로 사용되는 파티션을 별도의 디스크로 옮기고 데이터를 여러 디스크에 분산시켜서 높은 수준의 무작위 I/O 로 작업부하를 개선시키거나 데이터 접속 경로에 따라 파티션 위치를 디자인해서 저장소 요구사항의 균형을 맞출 경우에, 파티션 구성표를 정의해서 성능을 최적화시킬 수 있다.

테이블과 인덱스 분할은 데이터 웨어하우징 시나리오에 대해 강력한 기능이지만, 스케일 아웃 솔루션은 아니다. 단일 인덱스 또는 테이블의 모든 파티션이 같은 데이터베이스에 있어야만 한다. 단일 서버로 솔루션을 디자인할 경우, 분할된 테이블은 분할된 뷰보다 일반적으로 더 선호되는 솔루션이다. 여러 서버에 스케일 아웃되어야 한다면, 분산 분할 뷰(distributed partitioned views)가 생각해 볼 만한 솔루션이다.

장점

분할(Partitioning)은 큰 인덱스와 테이블 개체를 로딩하고, 보관하고, 삭제하고, 최적화하는 영향을 줄여주기 때문에 데이터 웨어하우스에서는 반드시 구현되어야 한다. 실제로, SQL Server 2005 테이블과 인덱스 분할은 크기 때문에 관리하기 어려운 큰 테이블에 이상적이다. 그림 9 에서와 같이, 회전문(sliding window) 시나리오는 분할의 많은 장점을 보여주는 좋은 예다. 데이터 웨어하우스는 종종 기간별로 데이터를 보관하고 삭제해서 웨어하우스에 최대한의 데이터 양을 유지해야 한다. 회전문 시나리오는 분할된 테이블 또는 뷰로 새로운 파티션을 밀어 넣거나 오래된 파티션을 밀어낸다. 테이블 분할은 이 중요 기간 동안의 다운타임을 최소화시켜준다. 새 데이터가 로딩되는 동안 기존 데이터를 사용할 수 있으며, 오래된 데이터가 보관되는 동안 새 데이터에 질의를 할 수 있다.

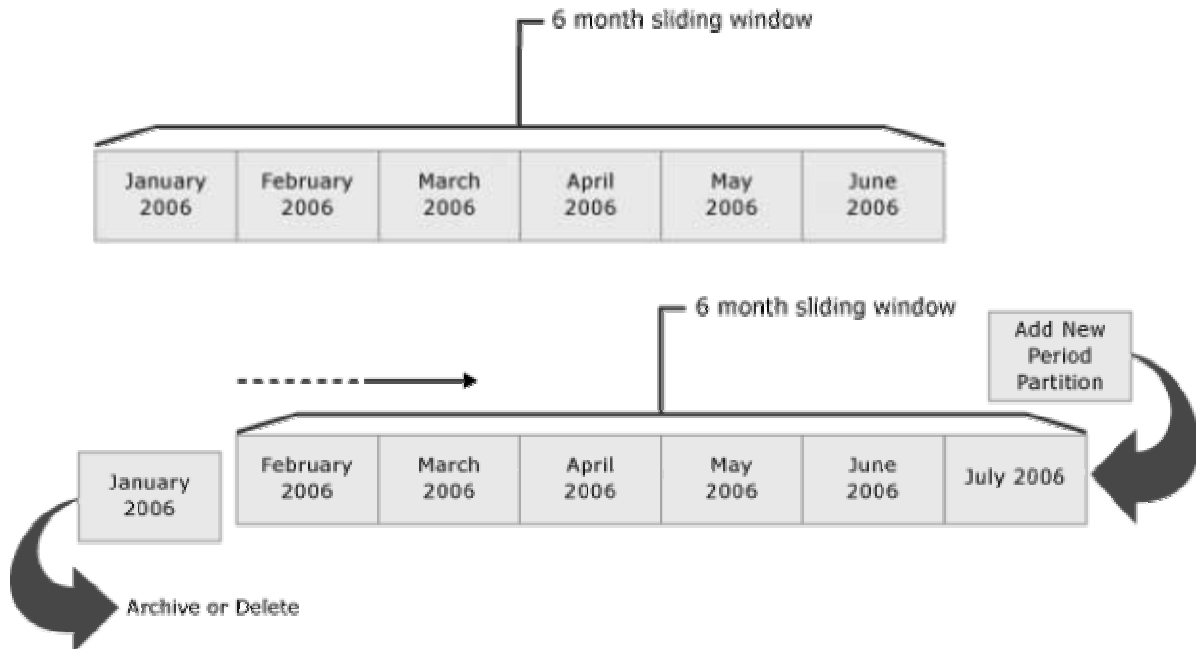


그림 9 회전문 시나리오

SQL Server 2005 Enterprise Edition 을 사용하면, 몇 초만에 기존의 분할된 테이블의 새 파티션에 데이터를 로딩할 수 있다. 분할 방식을 사용하지 않는다면 몇 분에서 몇 시간이 걸릴 작업이다. 분할 없이 큰 테이블에 데이터를 로딩하면 로딩 시간 동안은 전체 테이블을 잠궈야 한다. 로딩 작업이 완료되면, 기존 인덱스는 단편화 때문에 유지보수가 필요할 것이다.

SQL Server 2005에서는 데이터가 빈 새 테이블에 로딩된 후에 분할된 테이블에 새 파티션으로 병합될 수 있다. 인덱스 단편화는 새 파티션으로 분리된 후에 개별적으로 해결될 수 있다. 크고 긴 데이터 로딩 프로세스 중에도, 심각한 서비스 중단없이 기존의 테이블과 인덱스를 사용할 수 있다.

모든 데이터가 하나의 큰, 분할되지 않은 테이블에 있다면, 오래된 데이터를 삭제하려면 행 단위로 조작해야 할 뿐만 아니라 관련된 인덱스도 수작업 해야 한다. 잠금 상태가 더 오래 가고 프로세스는 더 큰 트랜잭션 로그가 필요하다. 테이블이 분할되면, 테이블에서 파티션을 제거하는 것만으로도 훨씬 더 빠르고 최소한의 로그만으로도 데이터를 제거할 수 있다.

단일 파티션도 인덱스 유지보수를 할 수 있다. 많은 OLTP 와 데이터 웨어하우스

데이터베이스에서는, 오래된 데이터는 거의 업데이트되지 않는 반면에 최근 데이터는 더 많은 업데이트와 삽입 작업이 이루어져서 단편화가 발생된다. 분할이 없다면, 큰 테이블에 대한 인덱스 재작성 작업은 많은 양의 로그가 발생하는 긴 프로세스다. 큰 로그 또는 긴 유지보수 시간을 수용할 수 없다면, 최적화되지 못한 테이블을 계속 사용할 수 밖에 없다. 테이블 또는 인덱스가 분할되면, 인덱스 유지보수는 정말로 작업이 이루어져야 하는 최근 파티션에 더 집중될 수 있다. 파티션은 더 큰 테이블의 더 작은 하위집합이기 때문에, 테이블을 최적화시키기 위해 필요한 시간과 로그를 크게

줄일 수 있다. 실제로, 온라인과 병렬 인덱스 작업, 온라인 파일 복원과 같은 다른 Enterprise Edition 기능과 함께 사용하면 가용성을 크게 높일 수 있다.

분할은 다양한 상황에서 병렬처리를 활용할 수 있다. 분할된 테이블은 분할 테이블에 대한 질의가 파티션 키에 따라 필터링되거나 많은 파티션에 걸쳐서 SELECT 작업을 실행하는 경우에, 단일의 분할되지 않은 테이블에 비해 병렬처리의 장점을 이용할 수 있다. 이와 같은 경우, SQL Server 는 시스템의 병렬 처리 능력을 극대화시킬 수 있다.

SQL Server 2005 는 관리, 개발과 사용을 디자인 단계부터 단순화시켰다. SQL Server 2005 분할을 사용해서, 큰 테이블과 인덱스를 쉽고도 일관성있게 관리할 수 있다. Standard Edition 에서도 분할 뷰(업데이트되지 않는)가 지원되지만, 테이블과 분할을 통해 간편하게 관리하고, 성능을 높여준다. 테이블과 인덱스 분할은 특히 데이터 웨어하우스 환경에 적합하며, 대형 OLTP 테이블에서의 보관과 데이터 전략을 지원하는데 도움이 된다.

64 비트 지원

기업 차원의 궁극의 확장성과 성능을 제공하기 위하여 SQL Server 2005 Enterprise Edition 은 Microsoft Windows Server™ Enterprise Edition 와 함께, Intel 과 AMD 의 차세대 64 비트 프로세서의 강력한 능력을 완벽하게 활용한다.

장점

64 비트 프로세서는 가장 많이 요구되는 미션 크리티컬 데이터베이스와 비즈니스 분석 작업에 대해 메모리 용량과 처리 능력을 최대한 확대시켜준다. 기가바이트의 메모리는 잊어도 좋다.

SQL Server 2005 Enterprise Edition 의 고급 Database Engine 는 64 비트 기술을 활용해서 물리적으로 1 테라바이트의 메모리까지 사용할 수 있다. Windows Server Enterprise Edition 다음 버전은 SQL Server 가 8 테라바이트의 물리적 메모리까지 활용할 수 있게 해줄 것이다.

디스크 저장소에 비해 메모리는 비교할 수 없을 정도로 빠르다. 메모리가 더 많을수록 질의결과, 보고서 생성, 데이터 분석과 조작이 더 빨라진다는 것을 의미한다. 서버의 부하가 클수록, 64 비트 기술에서 사용할 수 있는 확장 메모리의 장점도 더 커진다.

SQL Server 2005 Enterprise Edition 의 64 비트 지원은 32 비트 프로세서와 비교해서, 같은 시간에 두 배 더 많은 데이터를 전송하여 더 많은 작업을 더 빨리 할 수 있게 한다. 수치 연산은 더 정확하게 더 빠르게 실행된다.

64 비트 기술의 확장 능력과 처리 능력을 통해, 기존의 여러 데이터베이스 서버를 SQL Server 2005 Enterprise Edition 을 사용하는 하나의 64 비트 서버로 통합할 수 있다. 전산환경에서 서버 숫자를 줄일 수 있다면 막대한 비용을 절감할 수 있다. 물리적인 하드웨어 비용뿐만 아니라 인력, 전력사용, 데이터 센터와 소프트웨어 라이선스 비용까지도 절감할 수 있다. SQL Server Enterprise Edition 은 서버 당 50 개의 독립 서비스를 지원하기 때문에 한 서버에서도 여러 서버에서와 같이 유연하게 구성할 수 있다.

무엇보다도, Microsoft SQL Server 2005 Enterprise Edition 은 지금 당장은 32 비트 서버 플랫폼에서 시작하고, 이후에 기존의 라이선스를 변경하지 않고도 64 비트 플랫폼으로 전환할 수 있다.

실제로, 이미 소유하고 있는 하드웨어에서 고급 64 비트 처리 능력을 사용할 수도 있다. Microsoft Windows Server 2003 x64 Enterprise Edition 은 SQL Server 2005 Enterprise Edition 가 Intel Xeon 과 Pentium 프로세서와 AMD Opteron 과 Athlon 64 프로세서를 포함한 프로세서에서 64 비트 성능을 낼 수 있게 해준다. 현재 지원되는 하드웨어와 Windows Server 2003 x64 Edition 시스템 요구사항에 대해서는, [Overview of Windows Server 2003 R2 Enterprise x64 Edition](#) 을 참조하면 된다. Windows Server 2003 64 비트 에디션은 Intel Itanium 프로세서에서도 사용할 수 있다.

SQL Server 2005 Enterprise Edition 에서 64 비트 기술을 사용할 수 있는 예는 모두 나열할 수 없을 정도다. 고급 데이터 매핑과 GIS 애플리케이션, 대규모 데이터 분석과 조작 등, 사용자의 환경에 의해서만 제약을 받을 뿐이다. 고급 애플리케이션, 더 빨라진 데이터 조회, 서버 통합 등은

SQL Server 2005 Enterprise Edition 과 64 비트 기술을 함께 사용해 얻을 수 있는 장점 중 몇 가지에 불과하다.

프로세서 지원

SQL Server 2005 Enterprise Edition 은 무제한의 스케일 업 확장 능력을 제공하기 때문에, 대규모 통합, 대형 엔터프라이즈 데이터 웨어하우스와 사용이 집중되는 OLTP 애플리케이션에 큰 도움이 된다. Microsoft SQL Server 2005 의 다른 에디션들은 서버의 물리적 프로세서 수에 대해 제한되지만, SQL Server Enterprise Edition 은 이런 제약이 없으며 Windows 운영체제에서 사용할 수 있는 모든 프로세서를 사용할 수 있다. SQL Server 2005 Enterprise Edition 은 Windows Server 2003 Data Center Edition 으로 운영되는 단일 서버에서 64 개의 프로세서까지 지원하는 궁극의 확장성을 제공한다.

장점

추가 프로세서를 설치하는 즉시 데이터 저장소, 조회, 분석에 더 큰 힘을 얻게 된다.

SQL Server 2005 에 내장된 고급 일정관리 시스템은 모든 프로세서 간에 작업부하를 분산시키고 놓고 있는 프로세서 시간을 없애서 다양한 작업부하를 관리한다.

거의 모든 경우에, 서버를 늘리는 것보다 프로세서를 늘리는 비용이 훨씬 적게 든다. 서버 추가는 본체, 전원공급, 메모리, 프로세서, Windows Server 라이선스와 운영비용이 추가되는 반면에, 기존 서버에 프로세서를 추가하는 경우에는 이런 비용이 전혀 들지 않는다.

SQL Server 2005 Enterprise Edition 은 사용자가 추가하는 프로세서를 자동으로 사용한다.

메모리: 소개

기업의 미션 크리티컬 애플리케이션은 빠른 응답시간을 요구한다. SQL Server 2005 Enterprise Edition 의 고급 메모리 기능은 이와 같은 요구를 충족시켜주고 있다. Microsoft SQL Server 2005 Enterprise Edition 은 32 비트 Windows 에서는 64 GB 의 메모리까지 지원하며 Windows Server 2003 x64 Enterprise Edition 에서는 8 테라바이트의 메모리까지 지원해서 고급 메모리 기능을 완벽하게 활용하고 있다.⁵ 최고의 성능과 확장성을 요구하는 미션 크리티컬 애플리케이션에 대해, SQL Server 2005 Enterprise Edition 은 Intel Itanium 프로세서가 장착된 Windows Server 2003 64-bit⁶ 에서 7 테라바이트까지 지원한다.

디스크 저장소는 메모리에 비해 훨씬 속도가 느리다. SQL Server 2005 Enterprise Edition Database Engine 은 Windows 가 데이터를 메모리에 저장하도록 지시해서, 데이터가 메모리에서 디스크로 저장되지 않게 해서 속도를 더욱 최적화시킬 수 있다.

SQL Server 2005 Enterprise Edition 은 재시작 후에 서버를 가동시키는 시간을 줄여준다.

SQL Server 를 재가동시키면, 버퍼 캐시가 관련 메모리를 얻는데 걸리는 시간을 기동준비(ramp-up) 시간이라고 부른다. 이 시간 동안, 읽기 요청이 버퍼를 채우게 된다. Standard Edition 에디션 인스턴스가 메모리를 읽으면, 단일 페이지 읽기 요청이 단일 버퍼 페이지를 채운다.

SQL Server 2005 Enterprise Edition 에서는, 단일 페이지 읽기가 연관된 8 개의 페이지를 메모리로 불러 들인다. Enterprise Edition 은 기동준비 시간을 더 적게 걸리게 해서 재가동이 거의 영향을 미치지 않게 한다.

기업이 어떤 성능 요구사항을 요구할지라도, Windows Server Enterprise Edition 와 함께 구성된 Microsoft SQL Server Enterprise Edition 은 그 어떠한 요구사항도 충족시킬 수 있도록 서버의 메모리 용량을 최대한 사용할 것이다.

⁶ 호스트 운영체제에 따라 제약을 받을 수 있다.

메모리: 순간 메모리 추가(Hot-add) 지원

메모리를 늘려야 하지만 다운타임은 허용되지 않는 환경이라면? 순간 메모리 추가를 지원하는 SQL Server Enterprise Edition 을 사용해서 리부팅없이 메모리를 추가하면 된다.⁷

SQL Server 는 새 메모리를 즉시 활용할 것이다.

순간 메모리 추가 기능으로 서버의 전원을 끄고 메모리 증설 후에 리부팅해야 하는 서버 다운타임을 최소화시킨다.⁸ SQL Server 2005 이전의 SQL Server 버전은 서버가 시작될 때의 메모리 양만을 사용할 수 있었다. SQL Server 2005 에서는, 이런 제약은 사라지고 서버를 재시작하지 않아도 물리적인 메모리를 추가할 수 있기 때문에 가용성이 크게 향상된다.

메모리: Dynamic AWE 메모리 관리

Microsoft SQL Server 2005 는 Address Windowing Extensions (AWE) 기술을 통해 32 비트 플랫폼에서 64GB 의 물리적 메모리까지 활용할 수 있다.

AWE 를 통해, SQL Server 32 비트 에디션은 4GB 의 메모리 이상을 지원할 수 있기 때문에 대형 기업 애플리케이션의 확장성과 성능을 크게 향상시킬 수 있다. AWE 는 SQL Server 가 기존 하드웨어에서 64 GB 까지의 메모리를 활용할 수 있도록 확장시켜준다.

메모리 용량을 더욱 확장시키려면, 64 비트 하드웨어와 Windows 운영체제를 지원하도록 SQL Server 2005 Enterprise Edition 을 사용해야 한다.

SQL Server 2005 Enterprise Edition 은 32 비트 하드웨어도 최대한 활용할 것이다. 더 많은 메모리는 더 빠른 데이터 접속을 의미하기 때문에, AWE 기술을 통해 미션 크리티컬 애플리케이션을 더욱 빠르고 안정적으로 실행하는데 필요한 메모리를 사용할 수 있다.

다중 인스턴스 지원

SQL Server 2005 Enterprise Edition 은 50 개의 완전히 독립적인 Database Engine 인스턴스가 한 서버에서 병렬로 동작할 수 있게 한다. 각 각의 인스턴스는 프로세서와 메모리를 공유한다는 점을 제외하고는 서버의 다른 인스턴스와 완전히 독립적이다.

장점

한 서버에 여러 개의 인스턴스를 설치하면 하드웨어 통합을 통해 IT 운영비용을 크게 줄일 수 있다. 물리적인 하드웨어 구입비용 절감 외에도, 전력사용, 네트워크 지원과 공간 비용 등을 줄일 수 있다. SQL Server 2005 Enterprise Edition 은 최소한의 하드웨어로 가장 복잡한 기업 구성을 유연하게 수용할 수 있다.

각 각의 인스턴스는 자신만의 프로그램 파일, 보안 모델, 데이터베이스 파일, SQL Server Agent 등을 가지고 있기 때문에, 하나의 서버를 사용해서 자신만의 SQL Server 구성을 하는 여러 개의 데이터베이스 애플리케이션을 실행시킬 수 있다. 각 각의 인스턴스는 그것을 사용하는 특정 애플리케이션의 요구에 맞도록 정확하게 구성될 수 있으며 언제 인스턴스가 시작되고 중단되는 지에 대한 독특한 서비스 수준의 요구사항도 지원한다.

한 서버에서 다양한 SQL Server 2005 컴포넌트의 테스트와 전개용 인스턴스를 업무 환경에 맞도록 구성하여, 테스트하고 전개하는 과정을 단순하게 만들 수 있다. 다른 서비스 인스턴스에는 전혀 영향을 주지 않고도 SQL Server 서비스 팩 또는 보안 업데이트를 테스트하고 검증할 수도 있다.

⁷ 하드웨어가 순간 메모리 추가를 지원해야 한다.

⁸ Hot-Add Memory Support in Windows Server 2003(<http://www.microsoft.com/whdc/system/pnppwr/hotadd/hotaddmem.mspx>)
문서를 참조.

확장성 높은 공유 데이터베이스

확장성 높은 공유 데이터베이스(scalable shared database) 기능은 읽기 전용 리포팅 데이터베이스를 스케일 아웃 확장할 수 있는 솔루션을 제공한다. 범용 하드웨어를 사용해서, 여러 개의 SQL Server 인스턴스가 Storage Area Network (SAN)에 저장된 리포팅 데이터베이스의 한 복사본에 연결될 수 있다. 이렇게 되면 리포팅 데이터 파일 복사본 하나만 사용되기 때문에 데이터 저장소에 대한 요구사항이 줄어들게 된다.

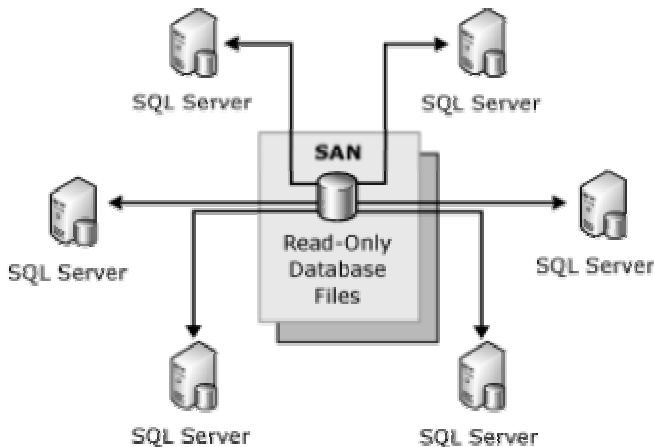


그림 10 확장성 높은 공유 데이터베이스

리포팅 데이터베이스는 주목적이 데이터베이스 호스팅인 읽기 전용 볼륨에 있어야 한다. 리포팅 데이터베이스가 리포팅 볼륨(reporting volumes) 집합에 대해 만들어지면, 그 볼륨은 읽기 전용 표시가 되고 여러 개의 리포팅 서버에 마운트된다. 각 각의 리포팅 서버에서는, 리포팅 데이터베이스가 Microsoft SQL Server 2005 인스턴스에 연결되고 확장성 높은 공유 데이터베이스(그림 10 참조)로 사용할 수 있게 된다. 리포팅 데이터베이스가 확장성 높은 공유 데이터베이스로 설정되면, 서로 다른 리포팅 서버를 사용하는 클라이언트가 공유할 수 있게 된다. 사용자 또는 애플리케이션은 데이터베이스가 연결된 모든 서버 인스턴스에 연결되어 데이터베이스에 질의를 할 수 있다. 특정 버전의 리포팅 데이터베이스에 대해, 서로 다른 서버의 클라이언트들은 리포팅 데이터에 대해 같은 뷰를 가지기 때문에 서버 간의 질의 결과가 일관성을 가진다.

장점

확장성 높은 공유 데이터베이스는 많은 장점을 가지고 있다.

- 범용 서버를 사용하고 있는 리포팅 데이터베이스의 작업부하를 스케일 아웃으로 분산시킨다. 확장성 높은 공유 데이터베이스는 질의를 실행하거나 SQL Server 2005 Reporting Services를 실행하는 등의 리포팅 목적으로 여러 서버 인스턴스가 비용 효과적으로 읽기 전용 저장소 또는 데이터 웨어하우스를 사용할 수 있게 한다.
- 작업부하 격리 기능을 제공한다. 각 각의 서버는 자신만의 메모리, CPU와 **tempdb** 데이터베이스를 사용하게 되어, 큰 질의가 모든 리소스를 독점하고 다른 질의에 영향을 미치지 못하게 한다. 격리 기능은 작업 테이블과 다른 **tempdb** 개체를 집중 사용하는 리포팅 환경에 도움이 된다.
- 모든 서버에서 리포팅 데이터에 대해 같은 뷰를 가지게 한다. 모든 서버 인스턴스가 동일하게 구성된 경우에는 연결된 모든 리포팅 애플리케이션은 같은 데이터 스냅샷을 가지기 때문에, 기업 전체에서 일관성 있는 리포팅을 가지게 된다. 예를 들면, 모든 서버는 단일 정렬(single collation)을 사용한다.

SQL Server Enterprise Edition 만이 사용이 집중되는 대부분의 리포팅 요구사항을 비용효과적으로 스케일 아웃으로 지원하는 솔루션이다.

인덱싱된 뷰(Indexed Views)

인덱싱된 뷰는 다양한 유형의 기업용 데이터베이스의 성능과 확장성을 크게 높여준다. 인덱싱된 뷰는 뷰에 대해 고유한 클러스터형 인덱스(unique clustered index)를 만들어서 계산되고 저장된(구체화된) 뷰다. SQL Server 2005 Enterprise Edition 쿼리 최적화 프로그램은 질의의 FROM 절에서 뷰를 직접 참조하지 않는 경우에서조차도, 인덱싱된 뷰가 질의 실행에 사용될 수 있는지를 자동으로 판단한다. 기존 애플리케이션들은 애플리케이션 코드를 변경하지 않고도 인덱싱된 뷰를 자동으로 활용할 수 있다.

인덱스가 없는 표준 뷰는 가상 테이블이기 때문에 결과는 데이터베이스에 저장되지 않는다. 질의 최적화 프로그램은 뷰 정의의 로직을 표준 뷰를 참조하는 실행 계획에 포함시킨다. 표준 뷰의 경우, 뷰를 참조하는 각 각의 질의에 대해 결과 값을 다이내믹하게 만들어주는 작업 부하는, 많은 데이터를 집계하거나 많은 행을 조인하는 것과 같이, 수 많은 행을 복잡하게 처리하는 뷰에 큰 영향을 준다. 이런 뷰를 질의에서 자주 참조하게 되면 성능 문제가 심각해질 수 있다.

고유한 클러스터형 인덱스가 뷰에 대해 만들어지면, 클러스터형 인덱스와 함께 테이블이 저장되는 것과 같은 방식으로 결과 값이 데이터베이스에 저장된다. 그 다음의 비 클러스터형 인덱스가 만들어져서 특정 명령의 성능을 향상시킬 수 있다. 인덱싱된 뷰를 특정 성능 문제를 해결하기 위한 논리적이고, 자연스럽고, 탈정규화(denormalized)된 디자인을 만들 수 있는 기회로 생각해도 좋다. 인덱싱된 뷰는 기본 테이블에 대해 만들어진 인덱스가 그렇듯이 인덱스가 만들어진 후에 기본 테이블의 데이터가 변경된 것을 자동으로 반영한다. 기본 테이블에서 데이터가 변경되면, 데이터 변경은 인덱싱된 뷰에 저장된 데이터에도 반영된다. Database administrators (DBAs)는 더 집중적으로 업데이트되는 테이블에 대해 인덱싱된 뷰 구현 테스트를 해야 한다. 기본 데이터가 자주 업데이트되면, 인덱싱된 뷰 데이터를 유지하는 비용이 인덱싱된 뷰를 사용하는 성능 장정보다 더 클 수도 있다.

장점

인덱싱된 뷰는 애플리케이션 전체에서 큰 테이블을 조인하거나 같은 조인 로직을 여러 번 사용하는 질의 성능을 향상시킬 수 있다. 예를 들면, OLTP 재고 데이터베이스는 Parts, PartSupplier, Suppliers 테이블을 조인하는 많은 질의를 가질 수 있다. 이런 조인을 하는 각 각의 질의가 많은 행을 처리하지 않을지라도, 수 십만 건의 조인 질의 전체는 막대한 부하를 일으킬 수 있다. 이런 관계가 자주 업데이트되지 않을 것이기 때문에, 조인 결과를 저장하는 인덱싱된 뷰를 정의해서 전체 시스템의 전체 성능을 향상시킬 수 있다.

인덱싱된 뷰를 구현하는 장점은 다음과 같다:

- 가끔 업데이트되는 데이터를 요약하고 집계하는 의사결정 지원과 분석 작업부하 균형
- 큰 테이블을 반복해서 조인하는 데이터 저장소와 데이터 웨어하우스.
- 질의 패턴을 반복하고, 같거나 겹치는 열들을 반복해서 집계하거나 같은 테이블에 조인을 반복하는 OLTP 시스템.
- Online analytical processing (OLAP) 저장소와 소스.
- 데이터 마이닝 작업부하 균형.

SQL Server 2005 Enterprise Edition 의 인덱싱된 뷰는 데이터베이스 애플리케이션 전체에서 동시성을 향상시켜서 집중되어 사용되는 질의 성능을 높인다.

병렬 DBCC (Database Console Commands)

SQL Server 2005 Enterprise Edition 은 다중 프로세서 지원으로 데이터를 빠르고도 효율적으로 검증할 수 있게 한다.

DBCC CHECKDB, DBCC CHECKFILEGROUP, DBCC CHECKTABLE 와 같은 데이터베이스 콘솔 명령은 데이터베이스 개체의 할당과 무결성을 검사한다. 간단하게, 이런 명령들은 페이지가 제대로 연결되고, 데이터가 유효한 상태가 되게 하고, 페이지 오프셋이 올바른지 검사해서 데이터베이스에 손상된 부분이 없게 한다. 안정적인 시스템에 대해 이런 검사를 해서 모든 내부 데이터베이스 문제를 수정하고, 손상을 일으키는 모든 하드웨어 또는 데이터베이스 문제를 찾고 수정하고 데이터베이스가 애플리케이션 요청에 적절하게 응답을 하게 만드는 것이 중요하다.

SQL Server 2005 Enterprise Edition 에 저장된 큰 데이터베이스 또는 테이블에 대해 DBCC CHECKDB, DBCC CHECKFILEGROUP, DBCC CHECKTABLE 를 실행하면, 시스템 자원을 사용할 수 있고 병렬 처리로 작업부하 균형을 맞출 수 있다면, Database Engine 이 병렬로 여러 개의 개체를 확인할 수 있다. 쿼리 프로세서는 각 각의 테이블 또는 테이블 배치를 검사하여 병렬처리를 다시 평가하고 자동으로 조정한다.

병렬 DBCC 는 일반적으로 활성화된 상태이지만, 추적 플래그 2528 을 사용해서 해제될 수 있다.

장점

데이터베이스 개체를 병렬로 검사하면 유지보수와 운영 비용을 줄이고 데이터베이스 가용성을 높여준다. 큰 데이터베이스는 검증 완료에 긴 유지보수 시간이 걸릴 수 있기 때문에 일반 업무 시간 중에는 나쁜 영향을 줄 수 있다. 유지보수 작업에 여러 개의 프로세서를 사용해서, SQL Server 2005 Enterprise Edition 은 데이터베이스 검증을 더 빠르게 완료할 수 있기 때문에 시스템 자원과 **tempdb** 을 풀어주고, 어떤 경우에는 중요한 업무 시간 중에 락을 거는 경우는 줄여준다.

업데이트가능한 분산 분할 뷰

한 서버의 용량 이상의 범위에서 작업을 해야 한다면, 데이터베이스 연합을 설정하면 여러 개의 SQL Server 데이터베이스에 데이터를 수평으로 분할해서 서버 그룹에 처리 부하를 분산시킬 수 있다. SQL Server 2005 Enterprise Edition 만이 분산 분할 뷰에서 데이터 수정을 할 수 있게 지원하기 때문에 OTLP 데이터베이스 애플리케이션을 위한 기능적 스케일 아웃 능력을 제공한다.

테이블의 데이터는 키에 따라 분할되며, 각 각의 파티션은 별개의 서버에 있게 된다. 이 서버들은 독립적으로 관리되지만, 데이터베이스에 대한 요청을 협동으로 처리한다. CHECK 제약 조건은 파티션 키에 적용되고, 최적화 프로그램은 이 키를 사용해서 명령이 어떤 데이터베이스에 영향을 미칠 것인지를 판단한다. 데이터에 접속하려면, 각 각의 서버에서 뷰를 만들고, 각 각의 서버에서 개별 테이블들을 합집합(union)으로 만든다. 뷰에 대해 SELECT, INSERT, UPDATE, DELETE 명령이 실행되면, SQL Server 2005 Enterprise Edition 은 데이터가 분할되는 방법을 이해하고, 적절한 서버의 적절한 테이블에 대해 명령을 수행한다.

분할 키와 관련된 데이터가 분산 분할 뷰의 성공을 좌우한다. 데이터는 균등하게 분산되어서 모든 파티션 간에 최적의 성능을 유지한다. 분할되지 않은 데이터에 대한 조인을 사용하는 분할된 테이블을 정기적으로 업데이트한다면, 여러 파티션에 걸친 질의의 영향을 최소화시킬 솔루션을 분석해야 할 것이다. 여러 기본 테이블을 유지보수해야 하는 어려움때문에, 단일 서버에서 분할 할 경우에는 SQL Server 2005 테이블과 인덱스 분할이 더 선호된다. 그림 11 은 테이블 분할과 분할된 뷰 간의 차이를 보여주고 있다.

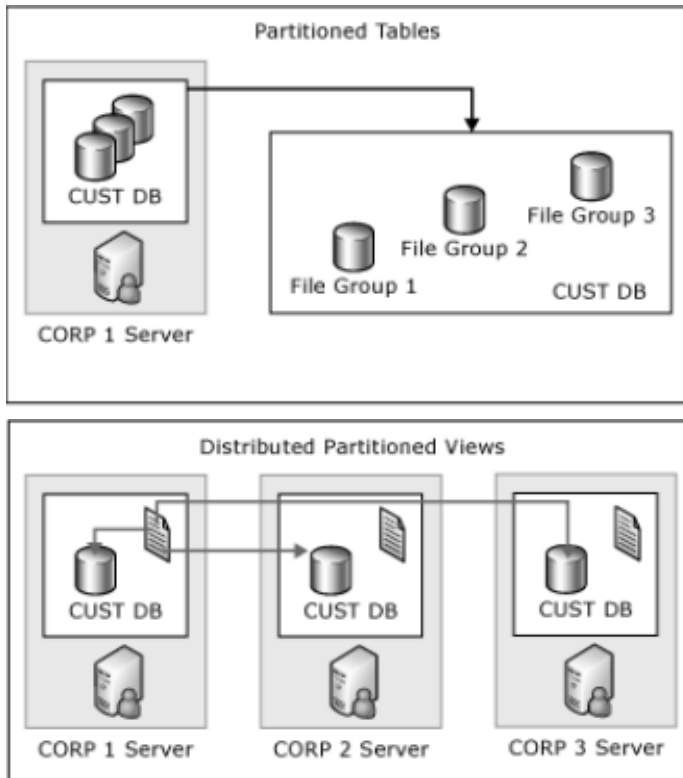


그림 11 분산 분할 뷰

장점

분산 분할 뷰는 큰 데이터베이스에 대해 투명한 스케일 아웃 솔루션이 필요한 경우에 가장 큰 도움이 된다. 작은 양의 쓰기를 수 없이 수행하는 웹 사이트 데이터베이스와 큰 OLTP 애플리케이션 데이터베이스는 단일 서버의 파티션 내에서만 업데이트되는 경우에 여러 서버에 작업부하를 분산시켜서 성능을 향상시킬 수 있다.

고급 스캐닝(Advanced Scanning)

고급 스캐닝은 서버 구성을 변경하거나 애플리케이션 코드를 변경하지 않고도 모든 종류의 데이터베이스의 확장성과 성능 문제를 투명하게 해결하는 SQL Server 2005 Enterprise Edition 의 많은 기능 중 하나다.

고급 스캔 기능은 여러 작업이 전체 테이블 스캔을 공유할 수 있게 한다. 고급 스캐닝을 사용해서, 데이터 페이지는 한 번만 읽어서 여러 명의 사용자가 공유할 수 있다. SQL Server 2005 Standard Edition 은 사용자에 대해 독립적 읽기만을 지원한다. 고급 스캐닝이 없다면, 큰 데이터 웨어하우스, 통합 인스턴스와 트랜잭션 데이터베이스를 위한 서버 자원의 I/O 활동이 늘어나서, 디스크 저장소와 버퍼 캐시에 대한 경합이 심해진다.

Transact-SQL 명령 실행계획이 테이블의 데이터 페이지를 순서없이 스캔하고, Database Engine 은 그 테이블이 이미 다른 실행계획을 위해 검색되고 있다면, Database Engine 은 두 번째 스캔의 현재 위치에서 두 번째 스캔을 첫 번째에 조인시킨다. Database Engine 은 각 페이지를 한번 읽어서 각 페이지의 행을 두 실행계획에 전달한다. 테이블 끝에 도달할 때까지 이 작업이 계속된다.

테이블 끝에 도달하게 되면, 첫 번째 실행계획은 완전한 검색결과를 가지지만, 두 번째 실행계획은 진행 중인 스캔에 조인하기 전에 읽었던 데이터를 여전히 조인하고 있을 것이다. 두 번째

실행계획이 테이블의 첫 번째 페이지로 돌아가고 첫 번째 스캔에 조인했던 부분으로 전진한다(그림 12).

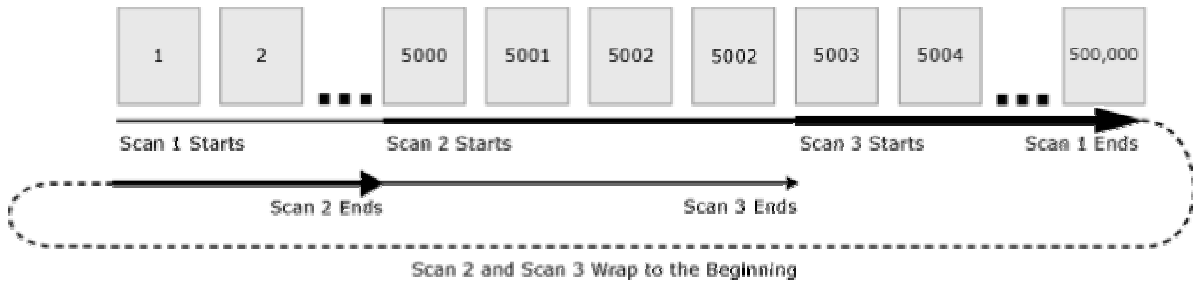


그림 12 고급 스캐닝

Database Engine 은 여러 스캔을 결합하기로 결정할 수도 있다. Database Engine 은 각 그룹의 모든 스캔을 완료할 때까지 데이터 페이지를 계속 순환한다. 이 메커니즘을 회전목마(merry-go-round) 스캐닝이라고도 부르며 SELECT 문에서 반환된 결과 순서를 ORDER BY 절 없이는 보장할 수 없는 한 가지 이유를 보여준다.

VARDECIMAL 저장 형식

VARDECIMAL 저장 형식은 큰 데이터베이스에 분명히 영향을 준다. 이것은 SQL Server 2005 Service Pack 2 Enterprise Edition 에서 제공되는 새로운 기능이다. VARDECIMAL 저장 형식이 테이블에 대해 활성화되면, 데이터, 인덱스와 로그 페이지에 저장된 모든 decimal 데이터는 가변 길이 저장 형식을 사용한다. 이 기능은 앞과 뒤의 0 을 없애고 최소한의 필요한 바이트만을 저장해서 decimal 과 numeric 데이터를 효율적으로 저장한다. 데이터 분산 정도에 따라 이 형식을 사용하면 상당한 공간을 절약할 수 있다.

테이블에 대해 VARDECIMAL 저장 형식을 활성화시키려면 두 단계가 필요하다. 먼저, 데이터베이스에서 VARDECIMAL 을 사용할 수 있도록 구성되어야 한다. 그 다음으로 VARDECIMAL 테이블에 대한 테이블 옵션에 플래그가 설정되어야 한다. 저장 형식 변경은 동시 사용자 접속이 허용되지 않는 오프라인 작업이다. 수정되고 있는 테이블이 작업 시간 동안 배타적으로 잠기며 동시 읽기 또는 쓰기 접속을 할 수 없다.

이것은 데이터 유형이 아니라 저장 형식이라는 것을 기억해야 한다. 이 기능이 활성화되면 numeric 또는 decimal 열의 데이터 유형을 변경할 필요가 없다. 여러 개의 decimal 또는 numeric 열을 한 테이블에 저장하면, 이 새로운 설정이 테이블의 모든 열의 저장 형식에 영향을 미칠 것이다.

장점

Decimal 과 numeric 데이터를 가지고 있는 큰 팩트 테이블을 사용하는 데이터 웨어하우스 시나리오에서, VARDECIMAL 을 사용하도록 팩트 테이블을 변경하면, 데이터베이스 파일에서 상당한 저장 공간을 절약하고 I/O 병목을 줄인다.

VARDECIMAL 저장 형식은, 많은 값이 최대 길이, NULL 또는 0 보다 더 작을지라도 일부 데이터에 대해 많은 자리수가 필요한 경우에 특히 적합하다. 예를 들면, 아주 드문 경우이지만 매우 많은 자리수로 열을 정의하지만, 대부분의 데이터가 최대 자리 수보다 훨씬 작을 수도 있다.

13 개 자리수로 decimal 값을 저장하는 큰 팩트 테이블을 생각해보자. 이 예에서는, 열의 대부분의 값이 13 자리수보다 더 작을지라도 VARDECIMAL 저장 형식은 테이블 공간을 줄일 수 있다. 이 테이블의 많은 행들이 0 값을 저장한다면, 상당히 많은 공간을 절약하게 될 것이다.

물리적 공간 절약은 VARDECIMAL 저장 형식의 한 가지 장점일 뿐이다. 이 새로운 기능은 어려운 디스크 I/O 집중 문제를 해결할 수도 있다. 각 행의 크기가 더 작다면, SQL Server 는 각 각의 테이블과 인덱스 페이지에 더 많은 행을 저장할 수 있으며, 테이블과 인덱스에 필요한 페이지 수를

줄여준다. 전체와 부분 인덱스 스캔은 더 적은 페이지만으로 수행될 수 있기 때문에, 디스크 I/O 를 바로 줄여주며 디스크 I/O 병목을 해소시킨다.

테이블 크기를 줄인다는 것은 데이터베이스 백업과 로그 백업이 더 작아진다는 것을 의미한다. 더 작아진 데이터베이스 백업과 로그 백업은 네트워크를 통해 물리적 백업을 이동시키는 로그 전달 작업 중의 로그 저장장치 비용절감과 네트워크 활용도 개선을 의미한다.

VARDECIMAL 저장 형식은 많은 데이터베이스의 성능을 높이고 공간을 줄일 수 있기는 하지만 모든 테이블에 대해서 좋은 것은 아니다. 테이블의 대부분의 행이 최대 길이의 자리수를 저장한다면, 저장형식은 고정길이로 남아 있어야 한다. 가변 형식으로 데이터를 저장하는 부담이 있기 때문에, 이 신기능을 사용하게 되면 일부 테이블에서는 저장공간을 늘릴 수도 있다.

관리편의성과 보안

Secure by default 설정, 데이터베이스 암호화와 강화된 보안 모델과 같은 기능 향상은 기업 데이터에 대하여 높은 수준의 보안을 제공한다. SQL Server 2005 Service Pack 2 Enterprise Edition 의 새로운 기능인 로그인 트리거(Logon Triggers)와 일반 조건 호환성 설정(common criteria compliance) 옵션은 기업표준과 요구사항에 따라 보안 모델을 보다 잘 정의하고, 조절하고, 구현할 수 있는 기능을 제공한다.

여기에서는 다음과 같은 내용을 설명하고 있다:

로그온 트리거(Logon Triggers)

일반 조건 호환성 설정(Common Criteria Compliance)

백업 미디어 세트 미러링(Mirrored Backup Media Sets)

다중서버 관리

VIA 지원

로그온 트리거

SQL Server 2005 Enterprise Edition Service Pack 2 는 LOGON 이벤트에 동기화된 응답을 하는 로그인 트리거를 새로 도입하였다. 로그인 트리거가 없다면, LOGON 이벤트를 감사하고 처리해야 하는 아키텍트와 어드민들은 비동기식으로만 LOGON 이벤트를 실행할 수 있기 때문에 데이터베이스 인증을 즉시 처리할 수 없게 된다.

로그온 트리거는 인증단계 다음 그리고 사용자 세션이 설정되기 전에 미리 정의된 동작 또는 스토어드 프로시저로 LOGON 이벤트를 처리한다. 에러 메시지와 PRINT 문의 메시지와 같이, 사용자에게 일반적으로 전달되는 트리거 내부에서 나오는 모든 메시지가 SQL Server 서버 로그로 전환한다. 실패한 로그인 시도에 대해서는 로그인 트리거는 동작하지 않는다.

장점

SQL Server 2005 Enterprise Edition 의 로그인 트리거는 데이터베이스에 누가 접속할 수 있고 언제 접속할 수 있는지를 자동으로 결정하는 유연한 솔루션을 제공한다. 로그인 트리거는 SQL Server 세션을 감사하고 조정하는데 사용된다. 이것은 일반 조건 호환성 설정의 요구사항이기도 하다. 로그인 트리거는 일반적인 데이터 로딩 중에 데이터 웨어하우스로의 접속을 제한하는 데이터베이스 애플리케이션 요구사항을 적용할 수 있는 완벽한 솔루션이다. 로그인 트리거는 또한 로그인 동작을 추적하고 일반적인 또는 특정 기간 동안의 사용자 데이터베이스에서 최대 동시 사용자 세션 제한을 적용하는데 도움이 되기도 한다.

일반 조건 호환성 설정(Common Criteria Compliance)

Microsoft 는 SQL Server 2005 Enterprise Edition 인증을 지원하기 위하여 풍부한 기능을 지속적으로 개발하고 있는 것에서 볼 수 있듯이, 일반 조건(Common Criteria)을 강력하게

지원하고 있다. 일반 조건은 보안 제품을 평가할 수 있는 프레임워크를 제공한다. 보안 적용과 정도를 평가하고, 필요한 보안 기능을 명시할 수 있는 언어를 제공한다. 일반 조건 평가는 많은 정부기관의 구매절차 중 일부이며 기업 고객이 선호하고 있는 요구사항이다.

SQL Server 2005 Enterprise Edition Service Pack 2 의 **common criteria compliance enabled** 옵션의 고급구성 기능은 일반 조건 호환성 설정에 필요한 요소들을 준수할 수 있게 해준다.

장점

SQL Server 2005 Enterprise Edition Service Pack 2 는 정부와 산업에서 자주 요구되는 국제 IT 보안 표준을 지원하는 일반 조건 호환성 설정 옵션을 도입하였다. **common criteria compliance enabled** 옵션은 다음과 같이 일반 조건으로 요구되는 3 가지 요소들을 지원한다:

- Residual Information Protection (RIP)은 메모리 공간이 새 자원에 재할당되기 전에 알려진 패턴의 비트로 덮어씌워져야 한다.
RIP 표준을 준수하게 되면 보안은 향상되지만 메모리 공간의 덮어쓰기는 성능을 느리게 할 수도 있다. **common criteria compliance enabled** 옵션을 활성화시키면, 메모리 공간이 덮어씌워진다.
- 로그인 통계(Login Statistics)는 마지막으로 성공한 로그인 시간, 마지막으로 성공하지 못한 로그인 시간과 마지막 성공한 로그인 시간 이후의 성공하지 못한 시도 회수를 추적한다.
common criteria compliance enabled 옵션을 활성화시키면, 특정 사용자 세션의 원래의 로그인과 보안 ID 등의 값을 보여주는 **sys.dm_exec_sessions** 에 5 개의 새로운 열이 추가된다.
- 테이블 수준 접속에 대한 제한된 조정은 테이블에 대해 접속거부된 개인이 테이블의 어떤 열도 사용할 수 없게 한다.
common criteria compliance enabled 옵션을 활성화시키면, 테이블 수준 DENY 가 열 수준 GRANT 보다 우선하게 된다. 활성화되지 않으면, 열 수준 GRANT 가 DENY 보다 우선하게 된다.

이 밖에, **max concurrent connection** 제한 또는 주의 일과 일의 시간 제한에 따라 세션을 거부하는 기능과 같은 일부 일반 조건 호환성 설정은 SQL Server 2005 Service Pack 2 의 로그인 트리거로 적용된다.

백업 미디어 세트 미러링

SQL Server 2005 Enterprise Edition 은 중요한 데이터베이스 백업의 중복성을 제공하기 위하여 백업 미디어 세트의 미러링 기능도 도입하였다. 미디어 세트를 미러링하게 되면 백업 장치의 오작동 영향을 최소화시켜서 백업 안정성을 높여준다. 백업은 데이터 손실에 대한 최후 수단이기 때문에 이런 오작동은 매우 심각한 영향을 준다.

SQL Server 2005 Standard Edition 은 백업 작업 중에는 하나의 백업 복사본 만을 지원한다. 사용자의 요구사항에 따라, Enterprise Edition 은 4 개의 미러링된 미디어 세트까지 만들 수 있다.

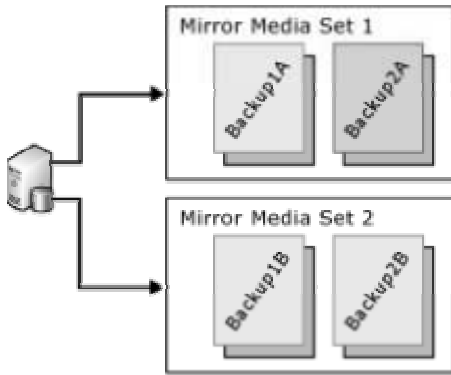


그림 13 미러링된 백업

그림 13 은 2 개의 디스크 파일로 미러링된 미디어 세트를 보여주고 있다. 백업의 디스크 파일은 미러 미디어 세트에 대응 파일이 있어야 한다. 데이터베이스 복원이 필요한 장애가 발생되고, 복구 작업 중에, Backup2A 파일에 중대한 에러가 발생했다. 미러 파일은 동일한 내용을 가지고 있기 때문에, Backup2B 를 복원할 수 있다. 미러링된 파일이 없다면, 복원 작업은 수일 전 또는 몇 주 전의 로그 백업이 필요할 수도 있어서, 복원하는데 오랜 시간이 걸리고 그 과정 중에 더 많은 위험이 발생할 수도 있다.

장점

미러링된 백업 미디어 세트는 복원 작업 중에 다운타임을 최소화시켜서 가용성을 높여준다. 손상된 백업은 더 긴 복원 시간 또는 복원 실패를 가져올 수 있다. 데이터베이스가 커지면, 백업 장치 또는 미디어의 실패로 백업이 복원되지 못할 가능성도 높아진다. 데이터베이스를 복원하는 것은 시간이 중요하며, 미러링된 백업 미디어 세트는 애플리케이션이 완전한 기능으로 더 빨리 복원될 수 있도록 해준다.

다중 서버 관리

대기업은 전체 SQL Server 환경에서 정기적인 작업을 중앙집중시킬 수 있는 도구가 필요하다. SQL Server 2005 는 여러 개의 SQL Server 인스턴스를 간단하게 관리할 수 있는 다중 서버 관리 기능을 지원한다. 다중 서버 관리는 하나의 중앙집중 서버에서 SQL Server Agent 작업과 유지보수 계획을 만들고 이 작업들을 여러 대상 서버에서 동시에 수행하는 것이다. SQL Server 2005 Enterprise Edition 에 다중서버 관리를 전개하면 작업을 대상 서버에 자동으로 전개할 수 있기 때문에 많은 도움이 된다.

주 (마스터) 서버는 대상 서버에서 실행되는 작업에 대해 하나의, 중앙 복사본을 저장한다. 주 (마스터) 서버는 작업을 대상 서버에 배포하고 이벤트를 수신한다. 대상 서버는 정기적으로 주 서버에 연결되어서 자신의 작업 일정을 업데이트받는다. 주 서버에 새 작업이 존재하면, 대상 서버는 그 작업을 다운로드받는다. 대상 서버가 그 작업을 완료하면, 주 서버에 다시 연결되어서 작업 상태를 보고하게 된다.

Multiserver administration configuration

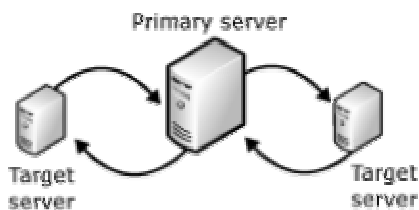


그림 14 다중서버 관리

Enterprise Edition 만이 주 서버에서 대상 서버로 작업을 자동으로 전개한다. Standard Edition 은 새작업 전개, 기존 작업 업데이트, 관련 일정 업데이트를 수작업으로 해주어야 한다.

장점

한 개 이상의 SQL Server 인스턴스가 있는 기업 환경은 SQL Server 2005 Enterprise Edition 을 사용해서 운용과 관리에서 많은 비용을 절감할 수 있다. SQL Server 2005 Enterprise Edition 의 다중서버 관리가 없다면, 대기업 환경에서는 여러 서버에서의 작업을 만들고 유지하는 부담이 클 수밖에 없다. 그리고 작업 전개에서도 일관성 문제가 발생되며 공용 작업이 비즈니스 또는 기술적인 이유로 변경되는 경우에는 서버에 업데이트하느라 많은 시간을 낭비할 수도 있다. 다중서버 관리가 Enterprise Edition 과 함께 사용되면, 관리자들은 한 서버에서 공용 작업을 만들고 업데이트할 수 있으며 모든 대상 서버가 작업 수행에 대해 일관된 정의와 일정을 가지게 만들 수 있다.

다중서버 관리는 각 각의 서버에서 작업 상태를 확인하는 부담과 시간을 줄여준다. 모든 대상 서버는 작업 상태를 주 서버에 보고하기 때문에, 데이터베이스 관리자는 한 서버만 확인해서 전체 환경에서의 작업 상태를 판단할 수 있다. 따라서, 데이터베이스 관리자는 더 많은 서버를 보다 잘 관리할 수 있으며 개별 서버를 추적하는 시간을 줄여 전체 환경을 개선하는데 더 많은 시간을 투자할 수 있게 된다.

VIA 지원

VIA (Virtual Interface Architecture)는 클러스터형 환경에서 상호 커뮤니케이션을 개선하기 위해 특수 하드웨어와 소프트웨어를 필요로 하는 고속 연결 솔루션이다. VIA 는 두 엔드포인트 간에 매우 효율적인 메모리 대 메모리 통신을 제공하도록 디자인되었다. 애플리케이션은 일반적으로 스케일 아웃 환경과 4 개 이상의 노드가 있는 클러스터형 환경에서 VIA 를 활용하여 장점을 얻을 수 있다.

SQL Server 2005 Enterprise Edition 만이 서버와 클라이언트 컴퓨터에 설치된 VIA 를 활용할 수 있다.

복제

이질적인 환경에서 통합 방법을 원하거나 지리적으로 분산된 데이터베이스 아키텍처에 대해 더 높은 가용성과 확장성을 만들어내고 싶다면, SQL Server 2005 Enterprise Edition 의 완벽한 복제 기능을 사용하면 된다.

여기에서는 다음과 같은 내용을 설명하고 있다:

피어 투 피어 트랜잭션 복제(Peer-to-Peer Transactional Replication)

오라클 퍼블리싱(Oracle Publishing)

피어 투 피어 트랜잭션 복제

SQL Server 2005 는 Enterprise Edition 에서만 피어 투 피어 트랜잭션 복제를 지원한다. 피어 투 피어 트랜잭션 복제는 원격지 사이에 데이터를 복제하여, 더 좋은 가용성과 확장성을 제공한다.

피어 투 피어 복제는 확정된 트랜잭션 복제 플랫폼 기반이지만 약간의 차이가 있다. 피어 투 피어 복제는 모든 피어 간에 양방향 복제를 제공해서 기존 트랜잭션 복제를 개선시키고 있다. 피어 투 피어 복제에서는, 모든 노드가 피어가 되어 모든 노드가 게시자인 동시에 구독자가 된다.

피어 투 피어 복제는 복제에 참여하는 모든 데이터베이스의 데이터를 읽거나 수정하는 애플리케이션을 위하여 디자인되었다. 데이터가 한 데이터베이스에서 변경되면, SQL Server 는 가장 빠른 경로를 통해 모든 다른 노드에 업데이트를 보낸다. 특정 노드에 변경 내용이 적용될 때 Replication 이 인지를 해서 변경 내용이 한 번 이상 노드를 순회하지 못하게 한다.

장점

일정에 있거나 예기치 못한 다운타임이 발생하는 환경에서는 피어 투 피어 복제로 가용성을 크게 향상시킬 수 있다. 모든 노드가 게시자와 구독자 역할을 하기 때문에, 애플리케이션은 네트워크에

있는 모든 노드를 지정할 수 있다. 피어 투 피어 복제는 또한 여러 노드에 걸쳐서 시스템 부하가 분산될 수 있는 이상적인 스케일 아웃 솔루션이다.

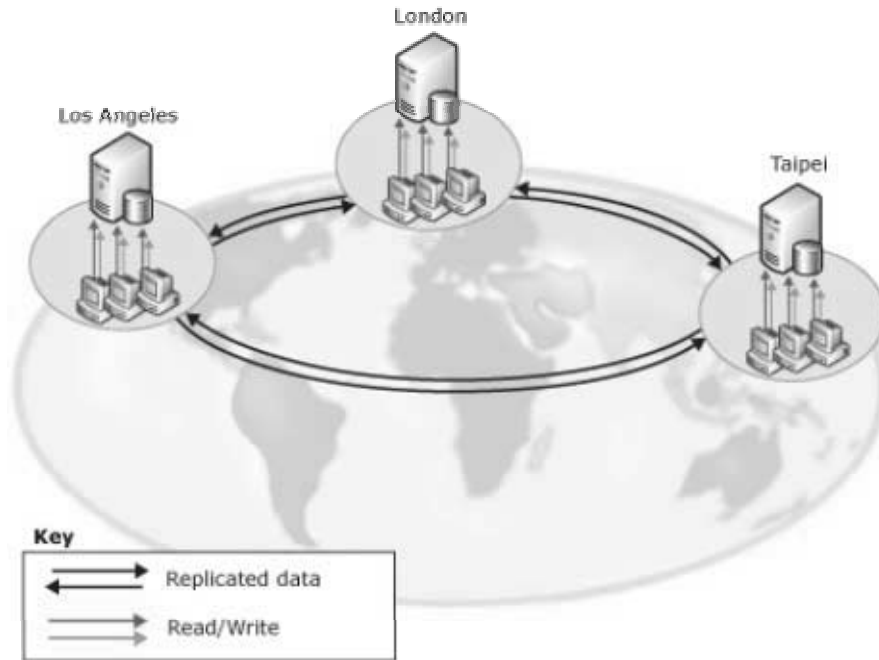


그림 15 피어 투 피어 복제

그림 15 는 Los Angeles, London, Taipei 에 사무실이 있는 전세계 소프트웨어 지원기업의 백엔드를 제공하는 3 대의 데이터베이스를 보여주고 있다. 각 사무실의 지원 엔지니어는 고객의 전화를 받아서 각 각의 고객의 전화에 대한 정보를 입력하고 업데이트한다. 세 사무실의 시간대는 8 시간 차이가 나기 때문에, 작업시간이 중복되지 않는다. Taipei 근무가 끝날 때에, London 근무가 시작된다. 한 사무실의 근무가 끝날 때에도 고객은 언제라도 전화를 할 수 있고, 그 전화는 근무 중인 다음 사무실로 전달된다.

각 지역은 고객 전화에 대해 지원 엔지니어가 정보를 입력하고 업데이트하는데 사용하는 데이터베이스와 애플리케이션 서버가 있다. 네트워크 구성은 시간으로 분할되어 있기 때문에 현재 근무 중인 노드에서만 업데이트가 이루어진다. 그 후에 업데이트된 내용은 다른 데이터베이스로 전달된다.

이런 네트워크 구성은 다음과 같은 장점이 있다:

- 한 데이터베이스에 장애나 유지보수 작업이 발생하여도 높은 가용성을 제공한다.
- 지역적으로 스케일 아웃된 아키텍처로 각 각의 노드의 성능을 높인다.
- 격리되지 않은 상태로 독립성을 유지한다. 각 사무실은 데이터를 독립적으로 삽입, 업데이트 또는 삭제할 수 있지만 다른 모든 데이터베이스에 복제되기 때문에 데이터가 공유된다.

오라클 게시(Publishing)

SQL Server 2005 Enterprise Edition 에서는, 오라클 데이터베이스가 표준 SQL Server 복제에서와 같이 SQL Server 에 바로 복제될 수 있다.

Microsoft SQL Server 2005 Enterprise Edition 을 사용하면, Oracle 8.0.5 이상의 오라클 게시자(Publisher)가 복제 토폴로지 구성(topology)에 포함될 수 있다. 게시하는 서버는 Oracle 을 지원하는 모든 하드웨어와 운영체제에 전개될 수 있다. 오라클 게시 기능은 SQL Server

스냅샷 복제와 트랜잭션 복제 기능에 잘 통합되어 있기 때문에 비슷한 수준의 성능과 편의성을 제공한다.

Oracle 스택샷 게시(publication)는 SQL Server 스택샷 게시와 비슷한 방식으로 구현된다. Snapshot Agent 가 Oracle 게시를 위하여 실행되면, Oracle Publisher 에 연결되어서 게시의 각 테이블을 처리한다. 각 테이블을 처리하면, 에이전트는 테이블 행을 조회하고 스키마 스크립트를 만든 후에, 게시 스택샷 공유에 저장한다. Snapshot Agent 가 실행될 때마다 전체 데이터 집합이 만들어지기 때문에, 트랜잭션 복제의 경우와 같이 변경 추적 트리거 Oracle 테이블에 추가되지 않는다. 스냅샷 복제는 게시 시스템에는 영향을 거의 주지 않고 데이터를 변형시키는 편리한 방법이다.

Oracle 트랜잭션 게시는 SQL Server 의 트랜잭션 게시 구조를 사용해서 구현되지만, 변경 내용은 Oracle 데이터베이스의 데이터베이스 트리거와 Log Reader Agent 를 사용해서 추적된다. Oracle 트랜잭션 게시가 만들어지면, 게시된 각 각의 테이블에 대해 트리거와 추적 테이블이 Oracle 데이터베이스 내에 만들어진다. Oracle 트랜잭션 게시가 만들어지면, 각 테이블의 트리거가 실행되어 수정된 행에 대해 정보를 복제 추적 테이블에 삽입된다. 그 후에 SQL Server 배포자(Distributor)의 Log Reader Agent 가 추적 테이블의 데이터 변경 정보를 배포자의 배포 데이터베이스로 옮긴다. 마지막으로, 표준 트랜잭션 복제와 같이, Distribution Agent 는 배포자의 변경 내용을 구독자로 이동시킨다(그림 16 참조).

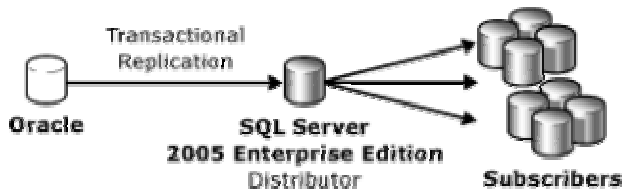


그림 16 Oracle 트랜잭션 게시

장점

이 기능은 단일 데이터 소스 사용이 가능한 이질적인 환경에 이상적이다. 다중 플랫폼 환경은 보다 쉽게 Oracle 데이터를 SQL Server 2005 데이터웨어하우스 또는 트랜잭션 시스템과 통합해서 관리 리포팅, 비즈니스 분석, 기업 전체의 데이터 일관성을 보다 잘 유지할 수 있다. Oracle 게시하기는 SQL Server 또는 Oracle 전환을 고려하거나 레거시 Oracle 데이터베이스를 전환하거나 분리된 데이터를 단일 기업 아키텍처로 통합해야 하는 기업에 적합하다.

Oracle 게시하기 기능은 중소 규모의 Oracle 데이터에 가장 적합하며, 대규모 Oracle 트랜잭션 시스템에는 권장되지 않는다.

비즈니스 인텔리전스기능

Microsoft® SQL Server™ 2005 Enterprise Edition 의 통합 BI 기능들은 신뢰성 높은 엔드 투 엔드 BI 와 분석 애플리케이션을 만들어낸다. SQL Server 2005 Enterprise Edition 은 대규모의 복잡한 데이터 통합, 분석과 리포팅 환경에 적합하다.

통합

분석

리포팅

데이터 마이닝

Notification Services

통합

Integration Services 는 데이터 웨어하우징의 추출, 변형과 로딩 (ETL) 작업을 포함한, 고성능의 데이터 통합과 워크플로우 솔루션을 구축하기 위한 플랫폼이다. SQL Server Enterprise Edition Integration Services 는 데이터를 마이닝하고, 클렌징하고, 실시간에 가까운 Analysis Services 환경을 유지할 수 있는 혁신적인 어댑터와 변형기능으로 기존의 ETL 보다 우수한 결과를 제공한다. 여기에서는 다음과 같은 내용을 설명하고 있다:

유사항목 조회와 유사항목 그룹화 변환(Fuzzy Lookup and Fuzzy Grouping Transformations)

용어 추출과 용어 조회 변환(Term Extraction and Term Lookup Transformations)

차원과 파티션 처리 어댑터(Dimension and Partition Processing Adapters)

SQL Server Integration Services 의 데이터 마이닝 기능 향상

유사항목 조회와 유사항목 그룹화 변환

SQL Server 2005 Enterprise Edition 에서 제공되는 Integration Services 의 유사항목 조회와 유사항목 그룹화 변환은 거의 모든 기업을 괴롭히는 쓸데없는 데이터를 처리할 수 있는 강력한 기능과 유연성을 제공한다. 오늘날의 비즈니스는 중복 레코드, 오타자, 약자, 일관성없는 서식 등의 수 많은 데이터 품질 문제를 가지고 있다. 주문 데이터베이스의 특정 고객을 검색하거나, 여러 소스의 데이터를 데이터 웨어하우스에 통합하거나, 여러 테이블 또는 데이터베이스를 병합하거나, 한 소스에서 중복된 내용을 검색할 경우, 텍스트 데이터를 일치하는지 검색하고 수정하는 전통적인 방식은 시간이 오래 걸리고, 비용이 많이 들며 에러가 발생할 여지가 많다. 유사항목 조회와 유사항목 그룹화는 매우 짧은 시간에 어려운 데이터 품질 문제를 해결할 수 있다.

유사항목 변형은 데이터 표준화, 데이터 수정, 서로 다른 시스템의 데이터 일치, 빠진 값 채워넣기와 같은 데이터 클렌징 작업을 수행한다. 유사항목 조회는 참조 테이블의 간결하고 표준화된 레코드와 입력 레코드를 일치시킨다. 일치 과정은 입력 레코드에 존재하는 에러를 수정한다. 유사항목 조회는 가장 일치하는 값을 반환해서 일치 수준을 알려준다.

유사항목 그룹화 변환은 일치할 가능성이 높은 소스 데이터와 그룹 행을 검사한다. 유사항목 그룹화는 데이터를 표준화시키기 위해 사용할 중복 그룹에서 올바른 가능성이 가장 높은 레코드를 선택한다.

장점

과거에는, 기업들은 일반적인 데이터 품질 문제를 다루기 위해 서드파티 시스템 또는 사용자 구현 알고리즘을 사용할 수 밖에 없었다. 데이터를 일치시키고, 그룹으로 만들고, 수정하는 솔루션을 개발하는데 많은 시간이 걸렸다. 이런 문제들은 이제 유사항목 조회와 유사항목 그룹화 기능을 사용해서 매우 짧은 시간 안에 더 정확하게 해결될 수 있다.

유사항목 조회와 유사항목 그룹화 변환이 이런 일반적인 문제를 어떻게 해결할 수 있는지를 가장 잘 보여주는 예가 별개의 비즈니스 시스템을 병합하는 경우다. 송장 발행 패키지와 CRM 패키지를 각

각 구매할 수 있다. 이 기업은 처음에는 시스템을 연결할 필요가 없었고 각 각의 시스템은 자신만의 고객 데이터를 가지게 된다(그림 17 참조). 어느 시점이 되면, 경영진은 시스템을 연결시키기로 결정하지만 각 시스템의 이름이 똑 같이 입력되지 않았을 수도 있다. 이 회사는 시스템 간에 중복된 고객 그룹과 ID 를 일치시켜야 하는 어려움에 직면하게 된다.

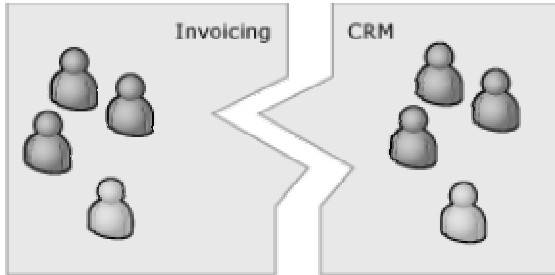


그림 17 단절된 시스템

SQL Server 2005 이전에는, 기업들은 서로 다른 시스템의 고객을 연결하려면 정확하게 이름이 일치해야 하는 경우가 많았다. "John Doe"는 송장발송 시스템에 입력되고 CRM 시스템에는 "John Do"가 입력되는 경우가 있을 수 있다(그림 18 참조). 기업 관점에서는, 동일한 고객도 서로 다른 두 고객이 되며 부정확한 송장발송, 미지급, 고객응대 문제 등이 발생할 수도 있다. 결국 회사는 이런 고객들을 매핑시키기 위해 수 천 줄의 코딩으로 사용자 지정 규칙을 구현하였다. 이런 방법은 개발하고 관리하기도 힘들며, 비슷한 일치 데이터를 수작업으로 검토하고 검증해야 하는 곤란을 겪었다.

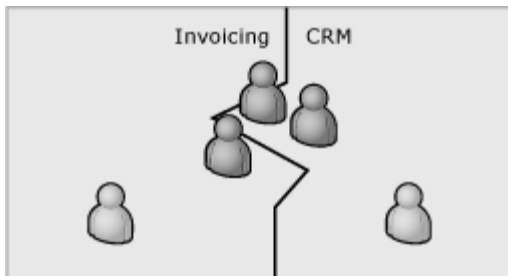


그림 18 일치하지 않는 John Doe 가 있는 연결 시스템

SQL Server 2005 Enterprise Edition 그리고 유사항목 조회와 유사항목 그룹 변환 기능으로, 데이터 일치 작업은 더 적은 비용, 더 짧은 시간 그리고 더 나은 품질로 쉽게 해결된다. 수 천 줄의 코딩을 할 필요도 없다. Integration Services 패키지에서 유사항목 그룹 변환을 사용해서 두 시스템의 고객이 그룹으로 만들어진다. 어느 레코드가 일치할 것인지를 확인할 수 있으며 일치 정보도 확인하고 확신과 유사성 정도에 따라 자동 처리를 하도록 만들 수도 있다. 결국 CRM 과 송장발송 시스템 고객들이 서로 매핑된다(그림 19 참조).

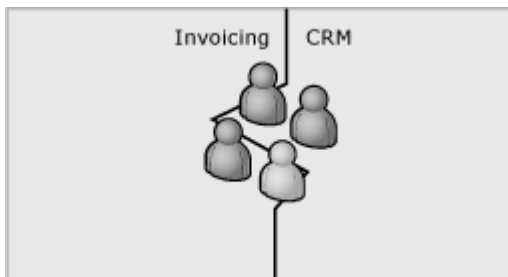


그림 19 유사항목 조회와 유사항목 그룹화를 통한 연결 시스템

유사항목 조회와 유사항목 그룹화는 많은 비즈니스 사례에서 도움이 된다. 다른 예들로는, 도메인 특유의 철자 확인, 비어있는 애플리케이션 양식 채우기, 통합 시스템에서 중복 데이터 삭제하기, 발견하기 어려운 관련 데이터 발견하기 등이 있다.

유사항목 조회와 유사항목 그룹화 변환을 통해 쓸모없는 데이터를 해결하는 훨씬 쉽게 해결할 수 있다. 이 새로운 Enterprise Edition 기능을 사용하는 솔루션을 구현해서 작업에 필요한 개발 시간을 줄이는 동시에 데이터 클렌징 과정도 개선할 수 있다. SQL Server 2005 Enterprise Edition 에 약간의 창조력만 추가하면, 기업의 데이터 클렌징 능력이 제한받을 일이 없게 된다.

용어 추출과 용어 조회 변환

SQL Server 2005 Enterprise Edition 의 Integration Services 는 문서와 텍스트 문자열을 마음대로 분류하고 검색할 수 있는 새롭고 강력한 도구를 제공한다. 용어 추출과 용어 조회 변환은 데이터 변환 과정을 텍스트 마이닝 도구로 만들어준다.

용어 추출 변환은 텍스트 열에 자유롭게 저장된 텍스트 또는 문서 도메인에서 공통 용어를 뽑아내고 그 용어를 참조 테이블에 저장하는데 사용된다.

용어 조회 변환은 변환 입력 열에 있는 텍스트에서 추출된 용어를 참조 테이블의 용어와 일치시킨다(용어 추출 변환을 사용해서 수집되었을 수도 있다). 용어 조회 변환은 조회 테이블의 한 용어가 입력 데이터 집합에 있는 숫자를 집계하고 참조 테이블의 용어와 함께 그 숫자를 변환 출력 열에 함께 기록한다. 내용과 관련된 용어에 따라 분류된 텍스트 결과 또는 문서가 만들어진다.

장점

용어 추출과 용어 조회 변환은 모든 종류의 공통 텍스트 검색과 인덱싱 요구사항을 해결하는데 사용될 수 있다. 예를 들면, 많은 기업들은 정기적으로 수 많은 고객 서비스 이메일 메시지를 받는다. 이런 막대한 양의 메시지에서, 어떻게 고객이 직면하고 있는 비즈니스를 분류하고 중요한 메시지를 놓치지 않을 수 있을까?

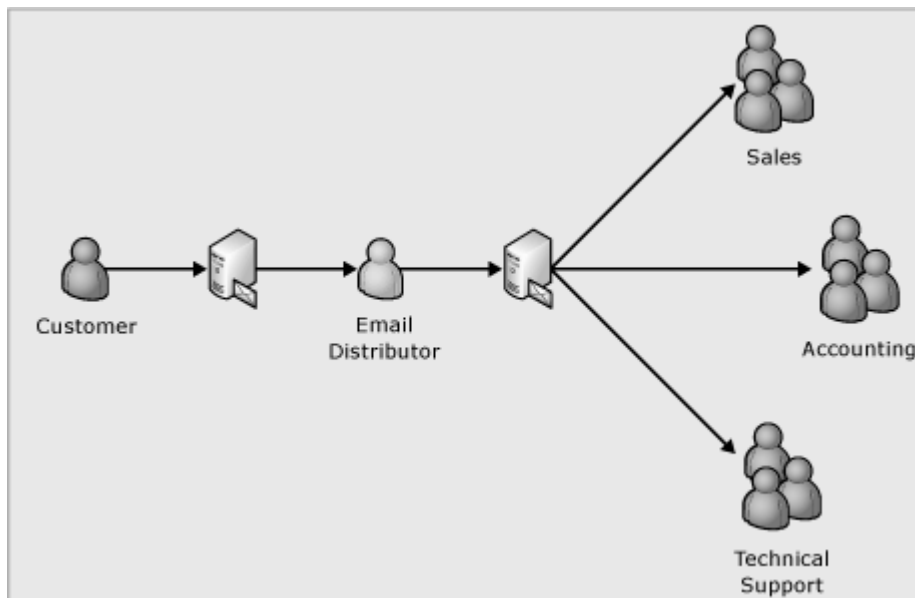


그림 20 수작업의 이메일 처리와 배포

일반적으로 고객 피드백 이메일 메시지에 제대로 응답하기 힘들다. 사람 또는 그룹은 메시지를 필터링하고 그 내용을 관련 부서에 전달해야 하는데, 이것이 많은 시간이 걸리게 된다. 용어 추출과 용어 조회 변환은 이런 고생스러운 작업을 자동화시킬 수 있다.

용어 추출 변환은 다양한 분류로 수신한 일반적인 이메일의 명사와 명사구를 추출한다. 사용자가 수신한 이메일 메시지에서 참조 용어들을 만들게 된다. 참조 용어가 추출되면, 용어 추출 변환은 그 중요도를 표시하는 점수를 부여할 수 있다. 이 참조 집합은 SQL Server 테이블에 저장되는데, 여기에서 용어를 쉽게 추가하거나 제거하고 점수를 조정할 수 있다.

참조 집합이 만들어지면, 이 참조 집합에 대해 밤새 배치 작업이 고객 피드백 이메일 메시지를 처리하고 적절하게 분류하여 예약어에 따라 적합한 개인 또는 부서로 전달한다. 예를 들면, “지불” 또는 “청구”라는 단어에 높은 일치율을 보이는 이메일 메시지는 계산서 발행 부서로 전달된다.

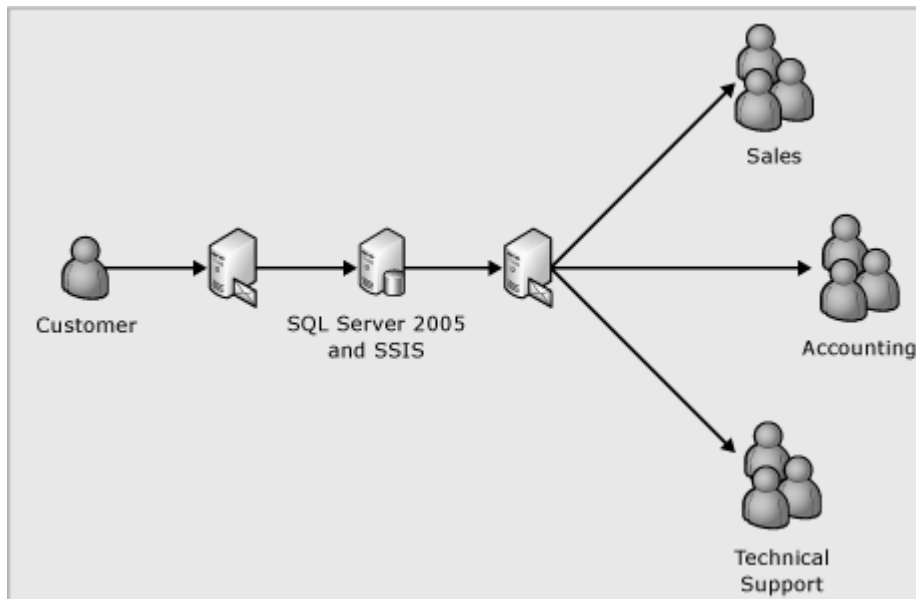


그림 21 용어 추출과 용어 조회로 자동화된 이메일 메시지 분류와 배포

용어 추출과 용어 조회 변환은 모든 종류의 텍스트 데이터에 사용될 수 있기 때문에, 이력서, RSS 피드, 법률 문서 또는 다른 모든 텍스트 마이닝 솔루션을 관리하는 강력한 도구가 된다.

차원과 파티션 처리 어댑터

SQL Server 2005 Enterprise Edition Integration Services 덕분에 차원과 파티션 처리 어댑터는 실시간 데이터 분석에 더 큰 장점을 가지게 되었다. SQL Server 2005 Enterprise Edition Integration Services의 차원과 파티션 처리 어댑터를 사용해서, 데이터를 SQL Server 2005 Analysis Services 파티션 또는 차원에 직접 넣어서 큐브를 최신 데이터로 업데이트할 수 있어서, 실시간 또는 실시간에 가까운 분석 환경을 제공한다.

장점

많은 기업들이 작업 시간은 계속 줄어들고 있는 반면에 차원과 파티션 업데이트는 계속 증가하는 어려움을 겪고 있다. XML 파일, 일반 파일, 레거시 메인프레임 데이터베이스와 OLTP 관계형 데이터베이스의 데이터를 데이터 웨어하우스에 로딩하는 경우에는 이런 문제가 더 악화된다(그림 22 참조).

큐브를 만들기 전에 데이터를 또 클렌징해야 하기 때문에 웨어하우스의 데이터를 다른 준비용 데이터베이스로 옮기는 경우가 있다. 제대로 관리되지 않는 데이터 소스가 Analysis Services 파티션 또는 차원에 데이터를 제공하는 문제가 발생할 수도 있다. 이 경우에도 큐브에 로딩하기

전에 별도의 준비용 데이터베이스가 필요하다. 이 과정에서 모든 단계를 생략할 수 있는 방법이 있다면 어떻게 될까? Analysis Services 환경은 단일의, 신뢰받는 분석 플랫폼으로 더 정확한 정보를 가지고 더 정확한 의사결정을 할 수 있게 한다. 그리고, 더 이상 큐브에 데이터를 로딩하기 전에 별도의 데이터베이스 환경을 관리할 필요도 없어진다.

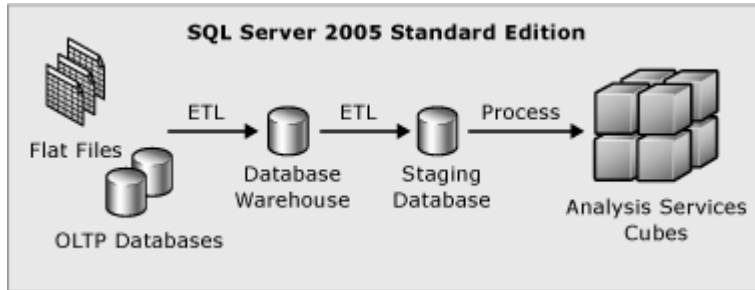


그림 22 Analysis Services 에 데이터를 로딩하는 구성 예

SQL Server Integration Services 의 차원과 파티션 처리는 웨어하우스를 업데이트하는 중간 단계를 완전히 없애고 실시간에 가깝게 유지하는 일을 한다. 차원과 파티션 처리는 데이터 흐름 내에서 데이터를 차원 또는 파티션 처리 선택(destination)으로 바로 보내서, 데이터를 준비용 환경으로 먼저 로딩해야 할 필요가 없다(그림 23 참조). 따라서 차원과 큐브는 SSIS 데이터 흐름과 인라인으로 업데이트되어 훨씬 더 가까워진 진정한 의미의 실시간 분석이 가능해진다.

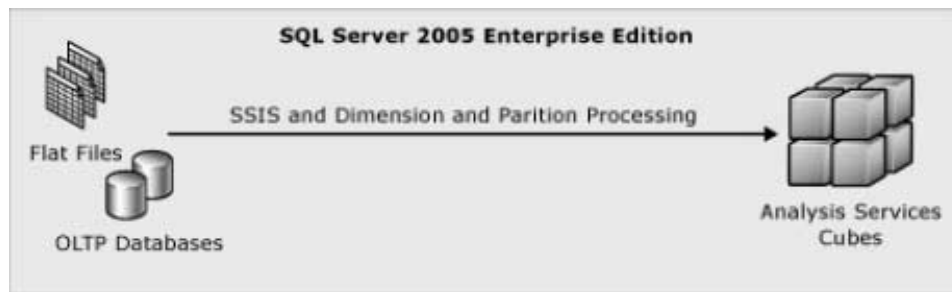


그림 23 SQL Server 2005 Integration Services Enterprise Edition 으로 데이터를 큐브에 보내기

예를 들면, 분산된 여러 센터에 적절한 식료품 공급을 유지하려고 노력하는 온라인 식료품 상점을 운영하고 있다고 가정해보자. 이익을 내기 위해서는 고객이 무엇을 언제 구매하고 있는지에 대해 각 물류 센터에서 실시간에 가까운 정보를 높은 품질로 이용할 수 있어야 한다. 아마도 물류 센터의 일일 구매와 발송 데이터를 복잡한 본사의 관계형 데이터 베이스로 밤새 옮긴 다음에, 데이터를 클렌징하고 큐브를 업데이트하는 시간이 많이 걸리는 작업을 하고 있을 것이다. 이런 상황에서는, 즉시 대응해야 하는 경영 의사결정은 거의 불가능하다.

SQL Server Integration Services 패키지를 사용하면, 클렌징 과정 중에 물류 센터의 데이터베이스의 데이터를 바로 파티션과 차원에 넣을 수 있다. 데이터를 준비 영역으로 이동시키는 오랜 단계를 생략했기 때문에 매 시간 업데이트할 수 있을 정도로 빨라져서 발송과 판매에 대해 가장 최신의 정확한 데이터를 얻을 수 있다. SQL Server 2005 Integration Services Enterprise Edition 의 차원과 파티션 선택을 사용해서 큐브를 더 자주 업데이트할 수 있다. 이제 최신 정보를 바탕으로 더 정확한 의사결정을 내릴 수 있게 된다.

모든 비즈니스 모델은 빠르고, 최신의 정확한 기업 데이터가 반드시 필요하다. 온라인 식료품 상점, 투자 회사, 거래 회사 또는 제조업체와 같은 기업들이 자신의 데이터를 더 자주 이용할 수 있게

된다면, 그들은 더 많은 정보로 더 정확한 의사결정을 할 수 있을 것이고, 경영 상황은 크게 개선될 것이다.

SQL Server Integration Services 의 데이터 마이닝 기능 향상

데이터 마이닝은 정보 시대의 중요한 도구가 되었다. 데이터 마이닝은 막대한 투자를 할 수 있는 대기업들이 개척한 분야이지만, SQL Server 2005 발표와 함께, 데이터 마이닝은 모든 기업이 쉽게 사용할 수 있는 저렴한 도구가 되었다. SQL Server 2005 Integration Services Enterprise Edition 에는 비즈니스와 기업을 지원하는 복잡한 데이터 마이닝 전략을 구현할 수 있는 사용하기 쉬운 도구들이 여러 개 포함되어 있다.

SQL Server 2005 Enterprise Edition 에는 Data Mining Model Training 과 Data Mining Query 가 포함되어 있다. Data Mining Model Training 으로, 데이터 마이닝 모델이 학습되고 다른 Integration Services 데이터 플로우 아키텍처와 함께 최신의 상태를 유지한다. Data Mining Query 는 Analysis Services 에 구축된 데이터 마이닝 모델에 따라 예측 질의를 실행한다. 예측 질의는 마이닝 모델을 사용해서 새 데이터에 대해 예측을 한다.

ETL 아키텍트는 Integration Services 의 데이터 마이닝 지원 기능을 사용해서 데이터 마이닝을 데이터 흐름에 직접 포함시키는 프로세스를 만든다. 한 Integration Services 패키지가 데이터베이스를 업데이트하고, 마이닝 모델을 학습시키고, 데이터 마이닝 질의에서 예측을 하고, 예측 결과에 따라 데이터를 자동화시키는 작업을 모두 한다.

장점

이런 Integration Services 도구들은 고객 특성에 맞는 광고, 제품 조사 그리고 예측에 따라 응답을 자동화하는 다른 의사결정 시나리오 등 다양하게 사용될 수 있다.

전자상거래 사이트의 데이터 마이닝은 SQL Server Integration Services Enterprise Edition 으로 정규 ETL 프로세스 내에 Data Mining Query 와 Data Mining Model Training 을 통합시켜서 빠르고도 쉽게 구현될 수 있다. 고객이 웹 사이트를 브라우징하면, 애플리케이션은 이동, 검색과 구매를 추적하고 기록한다. 하나의 SQL Server Integration Services 패키지가 Data Mining Model Training 을 수행해서 매일 수집된 데이터에 따라 데이터 마이닝 모델을 업데이트한다. 같은 패키지가 모델에 대해 Data Mining Query 를 수행해서 구매될 수 있는 제품 양을 예측하고, 재고 시스템에 반영해서 적절한 제품 양을 유지하게 만든다. 또한 사용자는 같은 패키지를 사용해서 고객에 대한 사용자 지정의 추가 작업을 자동화해서, 고객의 관심을 끌 수 있는 다른 제품으로 추가 구매를 유도할 수도 있다.

Data Mining Model Training 과 Data Mining Query 컴포넌트는 SQL Server 2005 Integration Services 내에서 데이터 마이닝을 지원할 뿐만 아니라 기존의 SQL Server Integration Services 패키지 내에서 직접 사용하기 쉬운 인터페이스를 통해 프로세스 디자인과 구현을 간단하게 할 수 있다.

분석

SQL Server 2005 Analysis Services 는 하나 이상의 물리적 데이터 소스에서 전통적인 OLAP 분석과 관계형 리포팅의 장점을 단일 데이터 모델로 결합하고 있다. OLAP, 리포팅, 사용자 지정 BI 애플리케이션의 모든 일반 사용자 질의는 관계형 데이터에 대해 단일 비즈니스 뷰를 제공하는 Universal Data Model (UDM)을 통해 기본 데이터 소스에 접속한다.

SQL Server Analysis Services Enterprise Edition 은 더 크고 복잡한 환경을 지원할 수 있는 가장 완벽한 플랫폼이고 쉽게 구성되는 환경에서 실시간 분석을 제공한다.

여기에서는 다음과 같은 내용을 설명하고 있다:

자동관리 캐싱(Proactive Caching)

분할된 큐브(Partitioned Cubes)

큐브 뷰(Perspectives)

반가산적 측정값(Semiadditive Measures)

계정 인텔리전스(Account Intelligence)

전체 쓰기저장(Full Writeback) 지원

메타데이터 번역(Metadata Translations)

데이터베이스 간/서버 간 연결된 차원과 측정값(Cross-Database/Cross-Server Linked Dimensions and Measures)

압축된 XML 과 이진 XML 전송(Compressed and Binary XML Transport)

Analysis Services 다중 인스턴스 지원(Multi-Instance Support)

자동관리 캐싱

현재 많은 BI 사용자들은 실시간의, 높은 성능의, 가용성 높은 분석 환경을 요구하고 있다. 경영 의사결정은 최신의 또는 실시간에 가까운 정보를 필요로 하며, 이런 의사결정자들은 긴 시간 동안의 큐브 처리 공백 또는 느린 성능의 관계형 솔루션을 원하지 않는다. SQL Server 2005 Analysis Services Enterprise Edition 은 multidimensional OLAP (MOLAP) 클래스 질의를 실시간 데이터 분석과 결합한 자동관리 캐싱 기능을 제공한다.

자동관리 캐싱 기능은 고성능 질의와 일반 사용자가 관계형 데이터베이스에 큰 부하를 주지 못하도록 하는 목적으로 조직된 데이터 복사본을 투명하게 동기화시키고 업데이트해서 데이터 퇴화(obsolescence)를 관리하는 과정을 단순하게 만든다. 캐싱 구조는 UDM 구조에서 자동으로 만들어지며 성능과 데이터 대기 시간의 균형을 유지하도록 정교하게 조정될 수 있다.

SQL Server 2005 Analysis Services Enterprise Edition 은 성능을 극대화시키고, 대기 시간을 최소화시키고, 처리 일정을 정할 수 있는 여러 가지 자동관리 캐싱 구성 옵션을 제공한다. 자동관리 캐싱은 측정 값 그룹, 각 차원 또는 전체 큐브의 각 각의 파트션에 대해 별도로 구성될 수 있다. 자동관리 캐싱 설정은 multidimensional OLAP 구조(MOLAP 캐시)가 얼마나 자주 재구성될 것인지, 캐시가 재구성되는 동안 질의가 기본 ROLAP 데이터 소스에 접속할 것인지 아니면 이전의 MOLAP 저장소에 접속할 것인지, 캐시가 일정에 따라 재구성될 것인지 아니면 데이터베이스가 변경되는 것에 따라 재구성될 것인지를 결정한다.

• 대기시간 최소화

자동관리 캐싱이 최소 대기시간으로 설정되어 있으면, 데이터가 최근에 변경되었는지 그리고 자동관리 캐싱이 구성되어 있는지에 따라, ROLAP 저장소 또는 MOLAP 저장소에 대해 OLAP 개체 질의를 한다. 질의 엔진은 데이터 소스가 변경될 때까지 MOLAP 저장소의 소스 데이터에 직접 질의를 한다. 대기시간을 최소화하기 위해, 데이터 소스가 변경된 후에는, 캐싱된 MOLAP 개체가 삭제되고 MOLAP 개체가 캐시에서 재구성되는 동안 질의는 ROLAP 저장소로 전환된다. MOLAP 개체가 재구축되고 처리된 후에는, 질의가 자동으로 MOLAP 저장소로 전환된다. 당일에 대해 정의된 최신 파티션과 같이 작은 파티션에 대해서는 순식간에 캐시가 리플레시될 수 있다.

• 성능 극대화

대기시간을 줄이는 동시에 성능을 최고로 높이기 위하여, 최신 MOLAP 개체를 삭제하지 않고도 캐싱이 사용될 수 있다. 그 다음에 데이터가 새 캐시로 읽히고 처리되는 동안에 MOLAP 개체에 대해 계속 질의를 한다. 이 방법은 질의 성능을 더 높이지만 새 캐시가 구성되는 동안에는 이전 데이터를 반환하는 질의가 만들어질 수도 있다.

장점

Analysis Services 환경에서 실시간 또는 거의 실시간에 가까운 정보가 필요하거나 큐브를 리플레시하기 위한 다운타임을 허용할 수 없는 환경은 자동관리 캐싱을 사용해서 놀라운 장점을 얻을 수 있다. 자동관리 캐싱이 없다면, 이런 환경은 관계형 소스에서 리포팅 질의를 할 수 밖에 없으며 두 가지 분명한 문제점을 가진다:

- 일부 유형의 질의분석은 양이 많은 OLTP 데이터베이스에 심각한 성능 문제를 일으킬 수도 있다. OLTP 활동을 위해 디자인된 많은 양의 데이터를 집계하게 되면 데이터베이스와 서버 자원 독점문제를 발생시킨다.
- 리포팅이 더 이상 한 가지 규칙만을 따르지 않는다. 두 번째 시스템이 도입되었지만 요약 요구사항, 통화 변환, 할인과 같이 데이터 웨어하우스에 적용되는 적절한 비즈니스 규칙과 정책이 제대로 적용되지 않을 수도 있다. 이 환경은 리포팅이 일관성을 잃어서 부정확한 데이터에 따라 의사결정을 내릴 수도 있다.

자동관리 캐싱은 세밀하게 구성될 수 있으며, 자동관리 캐싱 설정에 따라 장점도 달라질 수 있다. 모든 설정은 매우 높은 가용성의 분석 환경을 제공할 것이다. 사용자들은 파티션, 차원과 큐브 처리 중에 ROLAP 또는 MOLAP 의 데이터에 항상 접속해서 가용성이 매우 높은 분석 환경을 제공한다. 자동관리 캐싱은 실시간 또는 실시간에 가까운 분석과 리포팅 환경을 제공하는 동시에 막대한 사용자 질의와 백엔드 OLTP 시스템을 분리시킨다. 한 가지 규칙(single version of the truth)은 전통적인 MOLAP 환경의 대기시간 제약이 없이 대화식으로 데이터를 탐색할 수 있도록 디자인되었다. 항상 서비스를 제공하는 단일 소스에 대해 리포팅을 디자인하고, 그 소스에 모든 비즈니스 규칙이 일관적으로 적용되고 대기시간이 줄어들거나 없어지게 된다.

분할된 큐브

다중 큐브는 SQL Server 2005 Analysis Services Enterprise Edition 의 핵심기능으로 큰 측정값 그룹의 성능, 관리, 확장성과 가용성을 개선시켜준다. 여러 파티션을 채택하게 되면 각 각의 측정값 그룹을 더 작은 부분으로 나누어서 질의와 처리를 더 쉽게 만들고, 여러 프로세서, 드라이브와 서버에 질의 또는 처리 작업부하를 분산시킨다.

기본적으로, Analysis Services 큐브의 개별 측정값 그룹은 단일 파티션으로 디자인되었다. 추가 파티션은 데이터 소스의 테이블 또는 뷰, 데이터 소스 뷰의 테이블 또는 지정된 질의(named query)에 따라 측정값 그룹에 정의된다. 파티션 데이터가 저장되는 위치는 데이터 소스 바인딩에 의해 정의된다. 일반적으로, 측정값 그룹을 가로 또는 세로로 분할할 수 있다:

- 가로로 분할된 측정값 그룹에서는, 측정값의 각각의 파티션은 별개의 테이블에 따라 결정된다. 이런 종류의 분할은 데이터가 여러 개의 테이블로 분리되는 경우에 적합하다. 예를 들면 일부 관계형 데이터베이스는 매달의 데이터에 대해 별도의 테이블을 가진다.
- 세로로 분할된 측정값 그룹에서는, 측정값 그룹이 단일 테이블에 따라 결정되며 각 파티션은 파티션에 대한 데이터를 필터링하는 소스 시스템 질의에 따라 결정된다. 예를 들면, 단일 테이블이 여러 달의 데이터를 가지고 있다면, 측정값 그룹은 각 파티션에 대해 각 각의 달의 데이터를 반환하는 Transact-SQL WHERE 명령을 적용해서 월별로 분할될 수 있다.

파티션은 부정확하거나 일관성이 없는 결과를 피하기 위해 제대로 만들어지고 관리되어야 한다. 큐브 데이터의 무결성은 큐브 파티션 간에 분산되고 있는 데이터에 따라 결정되며 어떤 데이터도 파티션 간에 중복되지 않아야 한다. 데이터는 데이터가 파티션으로부터 합계되면, 하나 이상의 파티션에 존재하는 모든 데이터 요소들이 서로 다른 데이터 요소였던 것처럼 합계된다. 이렇게 되면 최종 사용자에게는 부정확한 합계와 오류투성이의 데이터가 전달된다. 예를 들면 제품 X에 대한 매출 트랜잭션이 두 테이블의 팩트 테이블에서 중복되어 있다면 그 제품 매출의 합계는 제품 X를 두 번 포함하게 된다.

장점

큰 측정값 그룹에 여러 개의 파티션을 채택하게 되면 많은 장점을 얻을 수 있다:

- **선택적 처리**
각 파티션은 개별적으로 처리될 수 있기 때문에, 소스 테이블 변경으로 한 두 개의 파티션만이 영향을 받는 경우에는 처리 시간을 단축시켜서 큐브의 가용성을 바로 높여준다. 예를 들면, 큐브는 지난 2년 동안의 매달의 파티션을 가지고 있을 수 있다. 일반적으로 이런 시나리오의 오래된 파티션들은 정적이며 더 이상 처리가 필요하지 않은 반면에, 가장 최근의 파티션들은 데이터가 추가되고 업데이트되면서 자주 처리를 해주어야 한다. 새 정보가 큐브에

업데이트되면 최근 달의 파티션만이 처리될 것이다. Standard Edition 은 각 측정값 그룹에 대해 한 파티션만을 지원하기 때문에 측정값 그룹이 항상 전체 모두가 처리되어야 한다.

- **유일 파티션 구성(Unique partition configuration)**

각 각의 파티션은 자신만의 집계와 저장 모드를 가지고 있다. 일별로 분할되면, 오래된 파티션은 필요한 디스크 공간과 처리시간을 줄이기 위해 매우 낮은 집계로 재처리되거나 사로 다른 저장 모드(ROLAP/HOLAP/MOLAP)를 가지게 된다. 사용 빈도 기반 최적화 마법사(Usage Based Optimization Wizard)는 각 각의 파티션에 대해 오래된 정도에 따라 다양한 집계 전략을 결정한다.

자동관리 캐싱 구성 옵션도 개별 파티션에 대해 정의되어 성능과 가용성은 극대화시키고, 대기시간은 최소화시키고 정기적인 처리 간격을 정할 수 있다.

- **개선된 질의 성능**

분할은 데이터가 단일 파티션 내에서 집계되는 경우에 특히 질의 성능을 높여준다.

- **자동 병렬 파티션 처리(Auto parallel partition processing)**

가장 크게 성능이 향상된 부분이 자동으로 파티션을 병렬로 처리하는 것이다. 기본적으로, SQL Server Enterprise Edition 은 단일 트랜잭션 내에서 처리 작업이 병렬로 진행되게 한다. 병렬 처리는 SQL Server 2005 Enterprise Edition 에서 도입된 기능으로, 처리 시간을 줄이고 새 데이터를 더 빨리 질의할 수 있게 만들어준다. Microsoft SQL Server 이전 버전은 이런 기능이 없었기 때문에, 사용자들이 자신의 유틸리티를 만들어야만 했다.

- **회전식 관리(Sliding window management)**

오래된 파티션은 기본 관계형 데이터 소스의 데이터 보관을 알기 위해 전체 측정값 그룹을 재처리하지 않고 삭제되기 때문에 더욱 관리하기 쉬워졌다. 예를 들면 큐브는 지난 24 개월의 활동을 저장한다고 하자. 이 회전식 관리 상황에서는, 새 달이 추가되면 가장 오래된 달이 삭제된다. 각 각이 분리된 파티션이기 때문에 가장 오래된 파티션을 삭제하여도 남아있는 파티션에는 아무런 영향을 주지 않으며 전체 큐브를 재처리하지 않아도 된다.

- **스케일 아웃 능력**

파티션은 원격 서버에 저장될 수 있다. 이 파티션은 원격 파티션 또는 분산 파티션이라고 부른다. 원격 파티션은 큐브 데이터를 저장하는데 사용되는 서버를 제외하고, 한 서버에 최소한 한 파티션이 있도록 여러 Analysis Services 서버에 파티션들을 전개한다. 원격 파티션이 측정값 그룹에 포함되면, 큐브의 메모리와 CPU 활용은 측정값 그룹의 모든 파티션에 분산된다. 예를 들면, 단독이던 아니면 부모 큐브가 처리될 때에 원격 파티션이 처리되면, 원격 Analysis Services 인스턴스에서 그 파티션을 위해 메모리와 CPU 를 사용하게 된다.

SQL Server 2005 Analysis Services Enterprise Edition 은 대규모 데이터 웨어하우스의 처리 시간, 질의 성능, 스케일 아웃 요구사항, 관리 문제, 파티션 등에서 중요한 솔루션이다.

큐브 뷰(Perspectives)

Microsoft SQL Server 2005 Analysis Services 의 한 큐브가 다중 팩트 테이블과 다중 차원 테이블의 다중 차원을 대표하는 큐브에 여러 측정값 그룹이 있어서 전체 데이터 웨어하우스의 내용을 대표할 수 있다. 이런 큐브는 매우 강력한 기능을 하지만 자신의 BI 와 리포팅 요구사항을 충족시키기에는 작은 부분의 큐브만 있으면 되는 사용자들에게는 매우 불편할 수도 있다. 큐브 뷰는 SQL Server 2005 Analysis Services Enterprise Edition 큐브에 대해 집중된 비즈니스 특유 또는 애플리케이션 특유의 뷰를 제공한다.

SQL Server 2005 Analysis Services 는 Unified Dimension Model (UDM)을 도입하고 있다. UDM 은 관계형과 차원 모델의 장점을 모두 결합하고 있다. 이 모델에서는, 단일 큐브가 다중 팩트 테이블을 대표하는 다중 측정값 그룹을 가질 수 있으며 더 나아가 전체 데이터 웨어하우스를 대표하도록 디자인될 수도 있다.

UDM 이 비즈니스 영역을 넘어서는 데이터를 분석할 수 있는 능력을 크게 확장시켜주는 반면에 훨씬 복잡한 사용자 환경을 만들기도 한다. 일반 사용자들은 모델 전체를 볼 필요가 없기 때문에

모든 종속관계를 이해하지 못하는 사용자들을 당황하게 만들 수 있다. 이런 사용자들은 자신이 수행하는 작업과 관련된 모델의 일부만을 보고 싶어한다.

SQL Server 2005 Analysis Services Enterprise Edition 은 큐브 뷰라는 기능을 도입하여 이런 문제를 해결하고 있다. 큐브 뷰는 UDM 위의 뷰로 정보의 일부만을 공개한다. 큐브 뷰는 서로 다른 사용자들이 같은 모델에 대해, 다양한 사용자들의 작업을 지원하는 부분만을 제공해서 서로 다른 표현을 볼 수 있게 한다. 같은 큐브가 서로 다른 사용자 그룹을 위하여 다중 큐브 뷰를 가질 수 있다.

장점

Analysis Services 의 큐브 뷰는 더 쉬워진 탐색, 질의와 다른 작업 기능을 제공해서 큐브에 접속하는 더 나은 사용자 경험을 제공한다. 크고 복잡한 큐브를 가지고 있는 BI 솔루션은, 큐브 뷰가 사용가 필요로 하는 간결한 정보 뷰만을 제공하고 큐브를 더 쉽게 탐색할 수 있게 하기 때문에 많은 장점을 얻을 수 있다.

예를 들면, Analysis Services 데이터베이스의 단일 Sales 큐브는 여러 개의 측정값 그룹과 서로 다른 많은 수의 큐브 차원, 매출 예상과 회계 데이터를 가지고 있을 수 있다. 클라이언트 애플리케이션은 전체 큐브를 바로 참조할 수도 있지만, 기본적인 영업 예상 정보만을 추출하려는 사용자에게는 너무 힘든 작업이 될 수 있다. 대신에, 같은 사용자가 Sales Targets 큐브 뷰를 사용해서 매출 예상과 관련된 객체들에 대해서만 Sales 뷰를 제한할 수도 있다.

큐브 뷰는 큐브의 개체에 대해 접속을 제한하지 않지만 큐브에 대해서 사용하기 쉬운 시야를 제공한다. 큐브 뷰를 통해 사용자가 보지 못하는 큐브의 개체들도 직접 참조하거나 질의할 수 있다.

반가산성 측정값(Semiadditive Measures)

SQL Server 2005 Enterprise Edition Analysis Services 는 반가산성 측정값을 사용해서 사용자 환경에서 필요한 모든 유형의 집계를 쉽게 개발하고 구성할 수 있다. 사용자들은 전통적인 가산성 성격을 지원하지 않는 데이터를 만나는 경우가 많다. 모든 차원에서 획일적으로 집계하지 않는 반가산성 측정값은 많은 비즈니스 시나리오에서 흔하다. 시간 경과에 따른 잔액 스냅샷에 따라 만들어지는 모든 큐브는 이런 문제를 가지고 있다. 보안, 계좌잔액, 예산, 인적자원, 보험 정책과 클레임, 다른 많은 비즈니스 영역을 다루는 애플리케이션에서 이런 스냅샷을 발견할 수 있다.

가장 일반적인 예로는 시간을 제외한 모든 차원에서 합계를 하는 재고가 있다. 의자 재고의 예를 들어보자.

	USA	Canada	North America
월	재고 수량	재고 수량	재고 수량
1월	5	6	11
2월	3	7	10
3월	2	5	7
분기	2	5	7

재고 수량 측정값은 **LastChild** 반가산성 집계를 구현하였다. 재고는 시간에 나누어진 가산성이 아니며 여러 시간에 걸쳐서 합계되는 것이 아니라 특정 시간에 알려진 마지막 값을 반환하는 것이다. 측정값은 **Geography** 차원에 걸쳐서 합계를 집계한다. 측정값의 반가산성 성격을 고려하지 않는다면, USA 의 분기 재고 수량은 재고에 실제로는 2 개의 테이블을 가지고 있는 경우에도 10 개의 테이블을 가지고 있는 것처럼 보이게 된다.

SQL Server 2005 Enterprise Edition Analysis Services 을 사용하지 않는다면, Transact-SQL 에서 뷰 또는 스토어드 프로시저에 파생 열을 만드는 것처럼, 새롭게 계산된 측정값을 플래그로 만들어야 한다. 이 플래그는 표준 가상 함수(SUM, COUNT, 기타)와 함께 사용해서

반가산성 값을 판단하게 된다. 플래그가 참이라면, 그 항목은 총계에 추가된다. 이 솔루션은 좋은 성과를 보이지 못하면서도 많은 시간이 걸린다.

Analysis Services 를 사용하면, 큐브에 바로 사용할 수 있는 매우 다양한 반가산성 집계 함수를 이용할 수 있다.

FirstChild	첫 번째 자식 멤버의 값을 조회한다.
LastChild	마지막 자식 멤버의 값을 조회한다.
FirstNonEmpty	비어있지 않은 첫 번째 자식 멤버의 값을 조회한다.
LastNonEmpty	비어있지 않은 마지막 자식 멤버의 값을 조회한다.
AverageOfChildren	비어있지 않은 모든 자식 멤버들의 평균 값을 계산한다.
ByAccount	계정 차원의 한 멤버에 대해 계정 유형에 할당된 집계 함수에 따라 집계를 계산한다.
Max	모든 자식 멤버 중 가장 높은 값을 조회한다.
Min	모든 자식 멤버 중 가장 낮은 값을 조회한다.

장점

SQL Server 2005 Enterprise Edition Analysis Services 는 위에서 설명한 것과 같이 반가산성 집계에서 사용할 수 있는 다양한 함수를 제공한다. 이 미리 정의된 함수를 사용하게 되면 여러 장점을 얻을 수 있다:

- **더 적은 개발 노력**
반가산성 집계 유형을 정의하는 것은 비즈니스 인텔리전스 Wizard 를 사용하는 것만큼이나 간단하다. 이 집계 유형들은 비즈니스 인텔리전스 Development Studio 의 측정값 속성에서도 사용할 수 있다. 큐브에서 계정 차원을 정의하면, SQL Server 2005 Analysis Services 는 계정 유형에 따라 적절한 반가산성 동작을 자동으로 설정한다.
- **쉬운 유지보수**
다중 계산 측정값에서 더 이상 많은 코드를 유지보수할 필요가 없다. 반가산성 집계 함수는 각 측정값의 속성일 뿐이다.
- **일관성있는 적용**
오랜 기간에 걸쳐서 여러 개발자들을 가지고 있는 큰 환경에서는 사용자 지정 공식을 일관적으로 적용하기 힘들어진다. 반가산성 함수는 별도의 계산 측정값이 필요하지 않으며, 이런 일반 계산을 지원하기 위해 측정값을 추가하거나 숨길 필요가 없다. 사용자 환경은 측정값 속성 내에서 함수를 일관성있게 적용하여 많은 장점을 얻을 수 있다.

계정 인텔리전스(Account Intelligence)

회계 리포팅에서 시간 경과에 따라 계정을 합계하는 것은 데이터 웨어하우스 기술에서 일반적인 일이다. 기업들은 몇 달 또는 몇 년에 걸쳐서 영업, 비용과 이익이 어떻게 변하는지를 보고 싶어한다. 자산, 부채, 수입 또는 비용과 같은 표준 회계 분류항목을 분석 환경에 어떻게 쉽고도 일관성있게 적용할 수 있을까?

SQL Server 2005 Analysis Services Enterprise Edition 의 계정 인텔리전스는 측정값과 차원에 대해 표준 회계 분류항목을 할당할 수 있도록 안내한다. 향상된 이 기능은 자산과 부채와 같은 계정 유형을 식별해주고 각 계정 유형에 적절한 집계를 할당한다. Analysis Services 는 이 분류항목을 사용해서 필요한 경우에 반가산성과 비가산성 집계 유형을 사용해서 시간 경과에 따른 계정을 적절하게 집계한다.

장점

SQL Server 2005 Analysis Services 큐브에 계정 인텔리전스를 구축하는 것은 마법사 기능을 실행해서 계정 차원에서 일반 계정 범주를 파악하는 것만큼이나 쉽다. Analysis Services 는 표준 계정 분류에 따라 계산을 하고 사용자 지정 롤업(rollup) 공식을 작성하지 않아도 시간 경과에 따른 적절한 집계 형식을 자동으로 할당한다. 필요하다면, 유연한 디자인 환경을 통해 표준 계정 유형의 어떤 기본 집계도 수정할 수 있다.

Enterprise Edition 과 이 기능은 분석 환경의 개발과 관리에 있어서 비용과 시간을 줄여줄 뿐만 아니라 모든 표준 계정 유형이 동일한 계정 공식과 집계 유형을 가지게 만들 것이다.

전체 쓰기저장 지원

SQL Server 2005 Enterprise Edition Analysis Services 의 차원과 큐브 쓰기저장 지원으로, 전력적인 what-if 분석이 가능해졌다. 대화식 what-if 분석은 비즈니스의 전략적 분석의 핵심으로, 데이터에 대한 변경 효과를 분석하고 데이터를 업데이트해서 "what-if" 시나리오를 시작한다. 향후의 분석을 위해 시나리오가 저장되면, 이와 같은 쓰기저장 변경은 향후의 분석을 위해 데이터를 사용하는 큐브로 적용되고 그 큐브에 접속하는 다른 사람들과 공유된다.

큐브의 데이터는 일반적으로 읽기 전용이다. 그렇지만 특정 시나리오의 경우 전체 큐브 또는 큐브의 특정 파티션을 쓰기 가능하게 만들어야 할 때가 있다. 쓰기 활성화된 차원은 서로 다르지만 보완형태를 가진다. 쓰기 저장이 활성화된 파티션은 사용자들이 파티션 셀을 업데이트할 수 있게 하는 반면에 쓰기 활성화된 차원은 차원 멤버를 업데이트할 수 있게 해준다. 사용자는 두 기능을 결합하거나 단독으로만 사용할 수도 있다. 예를 들면, 쓰기 활성화된 큐브 또는 쓰기 활성화된 파티션은 어떤 쓰기 활성화된 차원도 포함하지 않아도 된다.

장점

예산수립, 기획과 예측을 위한 다양한 시나리오를 분석할 경우에는 쓰기저장 기능을 사용하는 What-if 분석이 반드시 필요하다. 분석가들은 한 예산 또는 기획에 대해 여러 수정본을 추적하고, 다양한 조직 모델을 테스트하고, 연간 판매 계획 값을 조정하고 지정된 가중치로 분기와 월에 할당할 수 있게 된다. 사용자들은 원자재 비용을 높이고, 계획된 판매를 줄이거나 재고를 늘리면서 다양한 한계수익 시나리오를 테스트할 수 있다.

메타데이터 번역

SQL Server 2005 Enterprise Edition Analysis Services 는 다양한 언어를 지원해야 하는 글로벌 기업에 대해 단일 정보를 지원한다.

여러 언어와 국가에서 활동하는 기업은 종종 전체 조직을 지원하는 하나의 통합 솔루션을 개발해야 하는 어려움을 겪는다. 기업들은 다양한 사용자가 자신의 언어로 환경을 탐색할 수 있어야 하지만 분석 환경을 여러 버전으로 유지보수하는 위험과 어려움을 피하려고 한다. 번역이라고 하는 Using Microsoft SQL Server 2005 Analysis Services 기능을 통해, 하나의 Analysis Services 환경이 사용자 위치에 따라 다양한 언어를 자연스럽게 지원한다.

Adventure Works [Browse] - Summary
Perspective: Sales Sum Language: English (United States)

Month of Year	Order Count	Order Count	Order Count	Order Count	Order Count
January	1,571	1,390	770	97	2,481
February	1,621	1,620	865	200	2,684
March	1,674	1,635	851	190	2,749
April	1,719				
May	1,952				
June	1,844				
July	1,344				
August	1,482				
September	1,467				
October	1,498				
November	1,553				
December	1,798				
Grand Total	19,523				

Adventure Works [Browse] - Summary
Perspective: Ventes Language: French (Canada)

Mois d'Année	Nombre de Ventes	Nombre de Ventes	Nombre de Ventes
Janvier	1,571	1,390	770
Février	1,621	1,620	865
Mars	1,674	1,635	851
Avril	1,719	1,641	832
Mai	1,952	2,015	964
Juin	1,844	2,006	1,005
Juillet	1,344	1,088	643
Août	1,482	1,262	769
Septembre	1,467	1,208	760
Octobre	1,498	1,186	697
Novembre	1,553	1,399	810
Décembre	1,798	1,908	905
Grand Total	19,523	18,358	9,871

그림 24 큐브 뷰

예를 들면, 단일 큐브가 영어와 프랑스어로 사용자를 지원할 수도 있다. 프랑스 사용자가 프랑스 위치 설정이 되어 있는 워크스테이션에서 큐브에 접속하면, 프랑스어 번역이 존재한다면 그 사용자는 프랑스어로 멤버 캡션과 멤버 속성값을 보게 된다(그림 24 참조). 물리적으로는, 한 언어에 대해서 단일 Analysis Services 객체들을 개발하고 유지보수한다. 번역은 단순히 Analysis Services 개체를 확장시키는 추가 설명자(descriptor)일 뿐이다.

장점

Enterprise Edition Analysis Services 는 조직 전체의 모든 사용자들을 위해 환경을 개발하고, 관리하고, 유지하고, 구축하는 비용을 줄여준다. 번역을 사용하면, 조직 전체의 다국적 사용자를 지원하는 하나의 솔루션을 구현할 수 있다.

SQL Server 2005 Analysis Services Enterprise Edition 의 번역은 많은 장점을 제공한다:

- 단일 솔루션은 저장 장치, 개발과 관리 비용 절감을 의미한다.**
 큐브의 여러 버전을 여러 언어로 저장하지 않아도 되기 때문에 저장장치의 비용을 줄일 수 있다. 단일 개발환경은 필요한 비즈니스 규칙을 저장하기 때문에, 변경된 분석 환경을 전개하는데 필요한 비용과 시간을 크게 줄일 수 있다.
 작은 비즈니스 규칙 변경을 생각해보자. 다양한 언어를 지원하기 위하여 여러 큐브가 개발되고 유지보수되는 경우에는, 비즈니스 규칙을 변경하면 관련된 모든 큐브를 변경해주어야 한다. 여러 큐브에 대한 변경 작업은 비용이 많이 들고 오랜 시간이 걸리며 한 큐브가 제대로 변경되지 않은 위험이 높게 된다.
- 모든 사용자에게 하나의 데이터 집합이 단일 정보로 제공된다.**
 사용자들은 자신의 특정 환경이 만들어진 시간에 따라 데이터 차이를 검증하느라 시간을 낭비하지 않는다. 데이터 차이는 과거의 일이며, 사용자들은 데이터를 믿을 수 있게 된다.
- 단일 큐브는 솔루션을 구축하고 전달하는 시간을 줄여준다.**
 사용자들은 데이터와 솔루션을 더 빨리 사용하게 된다. 단일 큐브와 차원 집합이 만들어지면, 전세계 어디에서도 같은 수준의 서비스 계약을 가질 가능성이 높아진다.

SQL Server 2005 Analysis Services Enterprise Edition 은 조직 전체의 모든 사용자들에게 하나의 통합 정보를 제공하기 위해 필요한 고급 기능을 제공한다.

데이터베이스 간/서버 간 연결된 차원과 측정값

SQL Server 2005 Analysis Services Enterprise Edition에서는, 연결된 차원 또는 연결된 측정값은 같은 서버 또는 다른 서버에 있는 또 다른 Analysis Services 데이터베이스에 저장된 차원 또는 측정값에 따라 결정된다. 같은 측정값 그룹 또는 차원은 여러 Analysis Services 데이터베이스에서 정의되고 저장될 수 있어서, 공간 요구사항을 충족시켜줄 수 있는 동시에 Analysis Services 개체와 속성과 관련된 비즈니스 규칙과 애플리케이션을 로직을 여러 개의 복사본으로 이용할 수 있게 된다. 연결된 차원과 측정값 그룹은 소스 큐브 집계를 사용하며 자신은 데이터 저장소가 필요없다. 연결된 차원과 측정값을 사용해서, 한 Analysis Services 데이터베이스에 차원 또는 측정값을 만들고, 저장하고, 유지할 수 있으며, 데이터를 중복시키지 않고도 이 개체를 여러 데이터베이스의 사용자들이 사용하게 만들 수 있다. 이 연결된 객체는 사용자들에게는 다른 Analysis Services 개체처럼 보인다.

연결된 측정값 그룹은 클라이언트 애플리케이션이 접속할 때에는 다른 측정값 그룹과 같으며, 다른 측정값 그룹과 같은 방식으로 질의된다. 연결된 측정값 그룹에 있는 측정값은 같은 Analysis Services 데이터베이스에서 조회된 연결된 차원과 함께만 조직될 수 있다. 그렇지만, 계산된 멤버(calculated member)를 사용해서, 연결된 측정값 그룹의 정보를 큐브의 다른, 연결되지 않은 차원과 연결시킬 수 있다. 또한 참조 또는 다 대 다 관계와 같은 간접 관계를 사용해서 연결되지 않은 차원을 연결된 측정값 그룹에 연결할 수 있다.

연결된 측정값 그룹이 있는 큐브에 질의를 하면, 대상 큐브의 첫 번째 계산 중에 연결이 되고 확인된다. 이런 동작 때문에, 연결된 측정값 그룹에 저장된 모든 계산은 질의가 평가되기 전에는 확인될 수 없다. 다시 말해서, 계산된 측정값과 계산된 셀은 소스 큐브에서 상속받는 것이 아니라 대상 큐브에서 다시 만들어져야 한다.

장점

분산 애플리케이션을 위해 디자인된 연결된 측정값과 연결된 차원은 분산형 Unified Dimensional Models 간에 데이터와 차원을 효율적으로 공유할 수 있게 한다. 데이터는 물리적으로 한 위치에 저장되고 관리되며, 중복되지 않고도 여러 환경에서 공유된다. 연결된 개체들은 소스 Analysis Services 데이터베이스의 변경을 지속적으로 반영해서, 여러 데이터 저장소 또는 데이터 웨어하우스에서 일반적으로 사용되는 측정값과 차원에서 사용되는 귀중한 저장공간을 줄여준다.

연결된 측정값과 차원은 하나의 정보만을 제공하고, 비즈니스 규칙과 계층구조를 중앙에서 관리할 수 있게 한다. 데이터 소스, 데이터 클렌징, 계층구조, 특성 관계, 개체 처리와 관련된 로직은 한 위치에 저장된다. 연결된 측정값과 연결된 차원은 SQL Server 2005 Analysis Services Enterprise Edition에서만 제공된다.

압축된 XML 과 이진 XML 전송

SQL Server 2005 Analysis Services는 클라이언트와 서버 간의 통신 프로토콜로 텍스트 기반의 XML for Analysis (XMLA)를 사용한다. 실제로는 텍스트 기반의 XML 메시지는 대용량으로 성장할 수 있으며 유무선으로 많은 데이터를 전송할 경우에는 문제를 발생시키기도 한다.

SQL Server 2005 Analysis Services Enterprise Edition은 이진 XML 인코딩을 사용하고 XML 메시지를 압축해서 네트워크로 전송해서 큰 XML 메시지의 영향을 최소화시킨다.

클라이언트가 먼저 서버에 연결되면, 클라이언트는 서버가 통신을 위해 어떻게 구성되어 있는지를 판단한다. 클라이언트와 서버 간의 요청과 응답은 텍스트 또는 이진 XML로 전송될 수 있으며 통신은 압축되거나 그대로 전송된다. 압축되지 않은 텍스트로 요청과 응답을 전송하는 XMLA와 SOAP 프로토콜은 압축된 이진 XML과 비교해서 상당히 크고 느릴 수 있다. Enterprise Edition만이 압축된 이진 XML을 사용해서 대용량 통신의 성능을 높여주는 도구를 제공한다.

이진 XML 형식은 태그, 특성 이름, 일반적으로 반복되는 다른 데이터들이 XML 문서에서 한 번만 나타나게 XML 데이터를 인코딩한다. 앞부분부터, 텍스트 값은 사전(dictionary) 내에서 인코딩된 값으로 교체된다. 수치 값은 문자열로 전환되지 않고 순수 이진형으로 저장되어서 수치 값의 자리

수가 손상되지 않게 하며 통신에서 수치 값의 크기를 줄여서 성능을 높여준다. 이진 XML 인코딩을 사용하면 UTF-8 과 UTF-16 과 같은 인코딩을 통한 XML 문서의 크기를 크게 줄여준다.

SQL Server 2005 는 10 가지 압축 설정으로 압축 수준을 조정한다. 압축 형식은 전용 알고리즘을 사용해서 모든 요청과 응답을 압축한다. XMLA 응답은 일반적으로 큰 용량이며 압축으로 네트워크 성능을 개선시킨다.

장점

압축과 이진 XML 통신은 텍스트 XMLA 통신보다 더 좋은 성능을 제공한다. SQL Server 2005 Enterprise Edition 에서 이런 옵션을 구성하게 되면 일반 요청과 응답 통신 중에 전송된 네트워크 패킷 크기를 줄여주고, XML 파싱 성능을 높여주고, 수치 값 자리 수를 보다 잘 유지할 수 있다.

- **압축과 이진 XML 을 사용하면 네트워크 전송이 더 빨라진다.**

Analysis Services 응답은 일반적으로 크기가 크다. 압축과 이진 XML 형식으로 응답을 전송하면 네트워크 활용도가 매우 좋아진다. 클라이언트 요청은 일반적으로 작으며, 압축과 이진 XML 구성을 사용하여도 큰 성능 향상이 없을 수도 있다. 서버로 대용량의 데이터를 전송하는 클라이언트 요청은 압축과 이진 XML 이 큰 도움이 된다. SQL Server Analysis Services Enterprise Edition 만이 요청과 응답 통신에 대해 압축과 이진 XMLA 메시지를 사용한다.

- **서버에서 이진 XML 을 파싱하는 것이 다른 인코딩을 사용하는 것보다 더 효율적이다.**

이진 XML 을 사용하면, 통신의 모든 문자열이 Unicode 로 인코딩된다. 이진 XML 은 요청과 응답을 처리할 때에 Analysis Services 가 Unicode 문자열로 전환하지 않아도 되기 때문에 CPU 활용도와 응답 시간을 향상시킨다.

- **수치 데이터가 더 작고 더 정확하다.**

수치 데이터는 XML 문서에서 문자열로 전환하지 않고 순수 이진 형식으로 저장된다. 수치 데이터를 문자열로 표현하게 되면 이진 표현보다 크기가 더 커진다. 이 밖에, 수치 데이터를 문자열로 전환하는 작업은 CPU 를 집중적으로 사용하며 가끔씩 자리수가 손상될 수도 있다. 순수 이진 형식을 유지하게 되면, CPU 활용도가 낮아지고, 메시지 크기가 더 작아지고, 문자열로 전환할 경우에도 자리수가 손상되지 않는다.

SQL Server 2005 Analysis Services Enterprise Edition 에서의 압축과 이진 XML 사용은 가장 많이 사용되는 환경에 가장 큰 도움이 된다.

Analysis Services 다중 인스턴스 지원

SQL Server 2005 Analysis Services Enterprise Edition 은 단일 서버에서 50 개까지의 독립 인스턴스를 지원해서 대규모 스케일 업 능력을 제공한다. 한 서버에 Analysis Services 의 여러 인스턴스를 설치하면 CPU 와 메모리 같은 하드웨어 자원을 공유하게 된다. 그렇지 않으면, 각 각의 인스턴스는 독립 실행파일, 속성, 보안 모델, 데이터베이스와 큐브 등으로 완전히 분리된다. 각 각의 Analysis Services 인스턴스는 서버에 설치된 다른 인스턴스에 영향을 주지 않고도 독립적으로 시작되고 중단될 수 있는 서비스로 동작한다.

장점

한 서버에 여러 인스턴스의 Analysis Services 를 설치하면 하드웨어 투자를 극대화하고, 데이터 센터, 관리와 하드웨어 비용을 줄여준다.

여러 인스턴스의 Analysis Services 를 가지면 주 인스턴스에 애플리케이션, 수정, 서비스 팩을 구현하기 전에 같은 컴퓨터의 두 번째 인스턴스에서 테스트할 수 있는 것 등을 포함한, 많은 전략적인 목적을 달성할 수 있다. 또한 서로 다른 서버 구성 속성으로 별도의 인스턴스에서 애플리케이션을 실행할 수 있게 된다. 그렇지만 한 컴퓨터의 모든 인스턴스가 같은 메모리 자원을 공유할 경우, 테스트 목적으로만 존재하는 인스턴스는 주 인스턴스를 위해 메모리를 유보하도록 수작업으로 동작하도록 구성되어야 한다.

리포팅

Microsoft SQL Server Reporting Services 의 강력한 기능을 활용하여 가치있는 기업 데이터를 공유 정보로 변형해서 더 낮은 총소유 비용으로 통찰력있고 시기적절한 의사결정을 내릴 수 있다. 대화식, 비정규의 웹 기반 리포팅 플랫폼으로 다이나믹한 보고서를 만든다.

여기에서는 다음과 같은 내용을 설명하고 있다:

탐색가능한 비정규 보고서

데이터 중심의 구독(Data-Driven Subscriptions)

Reporting Services 스케일 아웃 전개

검색가능한(Clickthrough) 비정규 보고서

SQL Server 2005 Reporting Services Enterprise Edition 에서만 제공되는 탐색가능한 보고서로 보고서 작성기(Report Builder) 리포팅 이상을 사용할 수 있다. 탐색가능한 보고서는 보고서 작성기 보고서의 데이터에 대해 상세한 정보를 제공한다. 대부분의 시스템에서 제공되는 하위 보고서 또는 자식 보고서와 달리, Reporting Services 는 리포팅 모델에서 제공되는 데이터와 관계를 사용해서 실행 시에 탐색가능한 보고서를 디자인하고 구성할 수 있다.

기본적으로 리포트 서버는 각 각의 엔터티에 대해 두 개의 탐색가능한 템플릿 유형을 만든다.

하나는 단일 인스턴스 템플릿이며 다른 하나는 다중 인스턴스 템플릿이다. 보고서 내의 엔터티 간의 관계에 따라, 탐색가능한 보고서는 필드 목록과 값 또는 간단한 테이블로 만들어질 수 있다.

예를 들면, 부서 별로 직원을 나열하는 간단한 보고서 작성기 보고서를 실행한다고 하자. 보고서에 그룹 이름, 부서 이름, 직함, 성과 이름 필드를 추가했다. 포인터를 부서 이름 값 위로 이동시키면, 포인터에서 선택 연결 포인터로 변경된다. Facilities and Maintenance 부서 이름을 클릭하면, 새 보고서가 나타난다. 그림 25 는 Facilities and Maintenance 부서에 대한 정보를 보여주는 단일 인스턴스의 검색 가능한 보고서를 보여주고 있다.

Department	
#Departments	1
Department ID	14
Name	Facilities and Maintenance
Group Name	Executive General and Administration
Modified Date	6/1/1998
#Employee Department Histories	7

그림 25 단일 인스턴스의 탐색가능한 보고서

집계 값이 보고서 작성기에서 선택되면 다중 인스턴스 템플릿이 나타난다. 다중 인스턴스 템플릿은 선택된 집계 값의 모든 상세내용을 반환한다.

앞의 예로 돌아가서, 단일 인스턴스 보고서 내에서 #Employee Department Histories 값을 클릭하면, 새 보고서가 나타난다. 이 보고서는 Facilities and Maintenance 부서에서 일하는 7 명의 직원에 대한 정보를 보여준다 (그림 26 참조). 관련된 데이터가 모두 나올 때까지 계속 관련 데이터를 클릭할 수 있다. 원래 보고서로 돌아갈 준비가 되면, 보고서 실행(Run Report) 툴바의 부모 보고서로 돌아가기 (**Back to Parent Report**) 버튼을 클릭하면 된다.

Employee Department Histories

National ID	End Date ↓↑	Modified Date ↓↑	Start Date ↓↑	Name ↓↑
553069203		1/14/1999	1/15/1999	Facilities and Maintenance
141165819		1/2/2000	1/3/2000	Facilities and Maintenance
552560652		1/21/2000	1/22/2000	Facilities and Maintenance
646304055		2/27/2000	2/28/2000	Facilities and Maintenance
879334904		3/18/2000	3/19/2000	Facilities and Maintenance
28414965		4/4/2000	4/5/2000	Facilities and Maintenance
153479919		4/6/2000	4/7/2000	Facilities and Maintenance

그림 26 다중 인스턴스의 탐색동작

어떤 경우, 사용자는 탐색작업 중에 사용하기 위한 사용자 지정 보고서 템플릿을 디자인하고 싶을 수 있다. 이 사용자 정의의 보고서 디자인은 테이블, 차트와 그래프를 지원하며 보고서 모델의 전체 특성에 할당된다. 사용자 정의 템플릿은 할당된 엔터티 특성이 탐색작업 중에 선택될 때마다 나타난다.

장점

SQL Server 2005 Reporting Services Enterprise Edition 의 탐색가능한 능력을 통해, 사용자는 리포트 모델의 엔터티 간의 관계를 사용해서 엔터티들을 심층분석할 수 있다. 리포트 모델 디자인어는 탐색가능한 보고서에서 어떤 데이터가 나타나게 하거나 어느 리포트 모델 요소가 탐색을 위해 활성화되는지를 완벽하게 조정한다.

탐색가능한 보고서는 데이터 탐색을 위한 무한대의 경로를 제공해서 사용자 경험을 확대시켜준다. 보고서 내에서 관심있는 항목들을 탐색하기 위하여 새 보고서를 작성하는 대신에, 사용자는 관련된 데이터로 연결된 항목을 클릭하기만 하면 된다. 사용자들은 데이터를 탐색해서 미처 알지 못했던 관계를 발견하고, 어려운 질문에 답변을 하고 비즈니스에 대한 새로운 의미있는 상세내용을 발견할 수 있다.

데이터 중심의 구독

다양한 보고서로 수신자 목록을 관리하느라 어려움을 겪고 있는가? 수신자가 다양한 형식과 의미있는 내용으로 조정된 보고서를 보고 싶어하는가? 겉으로 보기에 단순한 시나리오를 해결하기 위해 여러 구독을 만들고 관리해야만 했는가? SQL Server 2005 Enterprise Edition Reporting Services 의 데이터 중심 구독은 IT 전문가들이 직원 또는 고객 데이터베이스와 같은 외부 데이터 소스를 사용해서 수신자 목록, 보고서 매터미터와 보고서 형식을 동적으로 만들 수 있게 한다. 단일 구독을 사용해서, 올바른 수신자에게 올바른 형식으로 올바른 정보를 전송할 수 있다.

데이터 중심 구독은 실행 시에 데이터셋을 생성하여 동작한다. 데이터셋은 모든 보고서 요구사항이 관리되는 한 테이블 또는 여러 테이블에 대한 질의에 따라 수신자 정보를 만든다. 이 데이터셋은 보고서를 만들고 배포할 때에 사용할 구독자 목록과 구독자 고유의 데이터를 반환한다.

장점

구독 관리는 어려운 작업이다. 보고서 수신자는 스태프 변경, 재조직 또는 빠르게 변하는 고객 기나에 따라 자주 변경된다. 비즈니스는 사용자 특유의 데이터만 전달되거나, 특정 형식으로 보고서를 수신하게 만들어야 할 수 있다. 표준 보고서 구독으로 이런 요구사항을 관리하는 것이 불가능하지는 않더라도 어려운 작업이다. 데이터 중심의 구독은 다음과 같은 어려움을 해결한다:

- **동적인 구독자 목록에 따라 보고서를 배포한다.**
데이터 중심의 구독은 구독자가 매달 바뀌거나 기존의 사용자 집합에서 그룹 멤버십을 결정하는 다른 기준을 사용하는 큰 조직 전체에 보고서를 배포하는데 사용될 수 있다. 아마도 단일 구독만을 만들어서 지난 주에 합류한 새 직원 또는 특정 수준 이상을 구매한 고객들에게 자동으로 보고서를 보내고 싶을 수 있다.
- **실행 시에 조회된 보고서 패러미터 값을 사용해서 보고서 결과를 필터링한다.**
영업 스태프는 데이터 중심의 구독을 사용해서 영업 팀이 연락해야 하는 영업 기회 목록을 만들 수 있다. 모든 영업 인력은 자신의 파이프라인에 기회만 포함시키도록 보고서를 사용자 지정할 수 있다. 구독자 특유의 정보를 배달해서, 영업팀 멤버는 보고서의 데이터에 보다 잘 집중할 수 있다.
- **보고서 출력 형식과 보고서 배달 옵션을 다양하게 구성한다.**
일부 수신자는 Microsoft Excel 로 보고서를 받기를 원하는 반면에 다른 사람들은 .pdf 형식으로 보고서를 원할 수도 있다. 모든 수신자가 보고서를 수신하고 싶은 파일 공유장소가 다를 수 있다. 데이터베이스에 쉽게 저장되는 이 모든 정보를 사용해서, 개인의 선호에 따라 보고서를 동적으로 배달할 수 있다. 더 이상 여러 구독을 관리하거나 일반 사용자를 위한 리포팅 기능을 제한하지 않아도 된다.

SQL Server 2005 Enterprise Edition 의 데이터 중심 구독은 구독 관리를 개선시켜주고 수신자에게 의미있는 사용자 특유의 데이터를 배달한다. 표준 구독에서 같은 기능을 복제하면 수십 만개의 구독을 만들어 낼 수 있다. 대신에 데이터베이스의 데이터 집합에 따라 단일 구독을 관리할 수 있다.

Reporting Services 스케일 아웃 전개

Microsoft SQL Server 2005 Enterprise Edition Reporting Services 스케일 아웃 전개를 통해, 가용성과 확장성이 높은 리포트 서버를 설치할 수 있다.

스케일 아웃 전개는 여러 리포트 서버가 공유 리포팅 서비스 데이터를 사용하도록 구성해서 지원된다. Reporting Services 데이터베이스는 클러스터형 서버 환경의 일부로 리포트 서버 중 하나와 함께 또는 원격 SQL Server 인스턴스에 설치될 수 있다. 장애극복 클러스터링은 리포트 서버 데이터베이스에 대해서만 지원된다. 장애극복 클러스터의 일부로 Report Server Windows 서비스 또는 웹 서비스를 실행할 수는 없다.

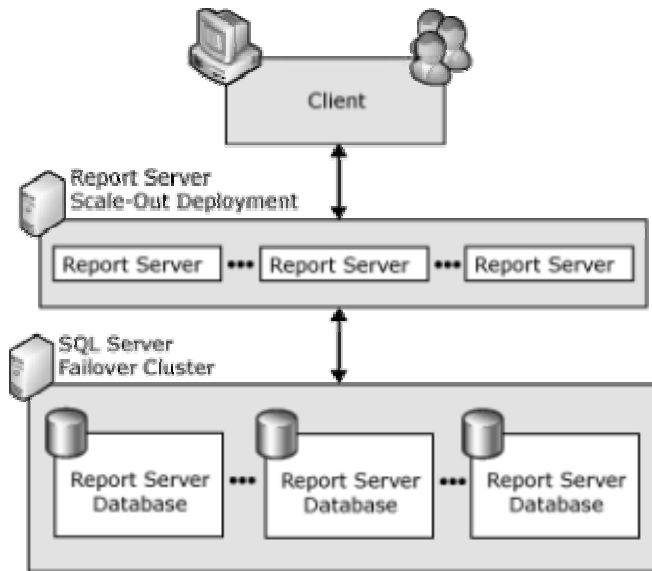


그림 27 SQL Server Reporting Services 스케일 아웃 전개 모델

스케일 아웃 전개로 Reporting Services 를 전개하면, 네트워크 로드 밸런싱으로 온 디맨드 리포팅과 대화식 보고서(보고서 작성기, 탐색가능과 매트릭스 보고서)의 성능을 높일 수 있다. 그렇지만, Reporting Services 는 네트워크 로드 밸런싱을 지원하는 소프트웨어와 도구는 제공하지 않는다.

장점

스케일 아웃 전개 모델은 모델에 더 많은 서버를 추가하면 성능을 크게 높여준다. 이 유연한 모델은 범용 서버를 성능 요구사항에 따라 추가하거나 제거할 수 있다. 리포팅 환경에서 동시사용자 수를 늘리고 성능을 높일 수 있는 유연하고, 관리하기 쉽고, 비용 효과적인 방법을 제공한다.

데이터 마이닝

SQL Server 2005 데이터 마이닝은 비즈니스에 대한 가치있는 통찰력을 제공해서 이벤트가 발생한 원인을 이해하고 미래에 무엇이 발생할 수 있는 지를 예측할 수 있다. Enterprise Edition 은 많은 비즈니스 패턴에 따라 데이터 마이닝 모델의 성능, 정확도와 결과물을 개선시킬 수 있는 매우 다양한 구성과 설정을 지원하는 확대가능한 플랫폼이다.

여기에서는 다음과 같은 내용을 설명하고 있다:

시계열 교차 예측(Time Series Cross Prediction)

데이터 마이닝 알고리즘 확장성(Data Mining Algorithm Extensibility)

데이터 마이닝 모델의 예측과 처리 능력 개선

무제한의 동시 데이터 마이닝 질의

연결 규칙을 위한 무제한의 특성 수

시계열 교차 예측

비즈니스에서는 어떤 일이 어떻게 발생되었는지 이해하고 특정 이벤트에 영향을 미친 모든 변수들을 확인하기 어려운 경우가 많다. 수 많은 이벤트의 여러 연혁들을 분석하고 연결시켜서 미래에 대해 정확하고 의미있는 예측을 하는 것은 더욱 어렵다. 데이터 마이닝은 비즈니스가 내리는 의사결정에 영향을 미칠 데이터의 패턴을 밝혀준다. SQL Server 2005 Enterprise Edition 시계열 알고리즘의 독특한 기능은 교차 예측을 사용해서 고급 예측 능력을 제공한다. 시계열 교차

예측 기능은 독립적인 데이터를 인식하고 분석할 수 있는 능력을 향상시켜주기 때문에 미래의 동작을 보다 잘 예측할 수 있게 된다.

교차 예측 분석은 분리되어 있지만 관련된 시계열을 사용해서 결과로 만들어지는 모델이 다른 계열의 동작에 따라 한 계열의 결과를 예측하는데 사용될 수 있도록 알고리즘을 훈련시킨다. 단일 시계열 모델은 미래 성과에 대해 일반적인 예측을 하는데 도움이 되지만, 시계열은 항상 독립적이지 않다. SQL Server 2005 Enterprise Edition 에서만 여러 시계열로 시계열 모델을 훈련시킬 수 있다.

장점

단일 시계열 모델은 전 해의 제품 판매에 따라 다음 해의 월간 제품 판매를 예측하거나 이전 달에 판매된 수량에 따라 다음 달에 가지고 있어야 하는 최적의 재고를 예측하는데 많은 도움이 된다. 시계열 알고리즘(Time Series algorithm) 예측 계산에서 단일 계열의 연혁(historical) 값을 사용해서 비즈니스 질문에 대한 답변을 얻을 수 있다.

관련되어 있지만 독립적인 여러 시계열 간의 상관관계를 발견하고 발견되지 않은 상관관계에 따라 예측을 하기 위해 시계열 모델이 필요한 경우가 많다. 교차 예측을 통해, 관리 팀은 이제 다양한 제품의 판매가 서로 관련이 있는지에 관심을 가져서 한 제품을 다른 제품의 판매를 예측하는데 사용할 수 있는지 판단할 수 있는 능력을 가지고 있다. 조직 전체의 월간 매출을 예측할 경우에 여러 시계열이 요인일 수도 있다. 교차 예측 모델을 사용해서 이자율과 관련된 가정의 구입 숫자 변경, 실업률 통계의 변화, 과세율의 변경을 보다 잘 예측할 수 있다. SQL Server 2005 Enterprise Edition 은 신뢰성 높은 시계열 교차 예측 알고리즘을 구축하는데 필요한 도구를 제공한다. 이 모델을 사용해서 데이터의 상호연결된 패턴을 밝히고 가장 중요한 비즈니스 의사결정을 내릴 수 있는 재미있는 상관관계를 유추해낸다.

데이터 마이닝 알고리즘 확장성

SQL Server 2005 Enterprise Edition 데이터 마이닝의 핵심 기능 중 하나는 확장성이다. Microsoft SQL Server 2005 Analysis Services 가 제공하는 9 가지 알고리즘 외에도, 데이터 마이닝에 사용할 수 있는 외부 알고리즘이 많이 있다. SQL Server 2005 Enterprise Edition 을 사용하면, 알고리즘 플러그인 API 를 통해 새 알고리즘으로 데이터 마이닝 능력을 확장시킬 수 있다.

개발자는 데이터 마이닝 알고리즘 플러그인 API 를 통해, 특수 데이터 마이닝 알고리즘을 쉽게 만들고 플러그인할 수 있다. 또한 서드파티가 만든 알고리즘도 특정 표준을 따르기만 한다면 Analysis Services 에 추가될 수 있다. 이런 알고리즘은 표준을 따라, 내장된 알고리즘과 똑같이 SQL Server Analysis Services 내에서 동작한다. 일반 사용자 관점에서는 내장된 알고리즘과 다른 것이 없다.

장점

SQL Server 2005 Enterprise Edition 은 데이터 마이닝 솔루션을 확장시킬 수 있는 최고의 유연성을 제공한다. 외부 데이터 마이닝 어댑터를 지원하기 때문에, 데이터를 마이닝하고 비즈니스에 대한 새로운 정보를 발견할 수 있는 가장 완벽한 능력을 가지게 되어, 비즈니스가 다음 단계로 진화할 수 있는 경쟁우위를 가지게 된다. 플러그인 알고리즘의 목적은 데이터에서 특성의 전문화된 패턴을 추출하고 이런 패턴을 디스크에 저장하는 것이다. 그 다음에 알고리즘은 이 패턴을 새 데이터에 적용할 때에 예측을 할 수 있게 된다. 이런 방법을 통해, SQL Server 가 확장되어 무제한의 데이터 마이닝 문제를 해결하는데 필요한 기능을 제공한다.

데이터 마이닝 모델의 예측과 처리 능력 개선

SQL Server 2005 Enterprise Edition 은 데이터 마이닝 알고리즘의 튜닝, 처리와 정확도를 상세하게 조정할 수 있다. SQL Server 2005 Enterprise Edition 은 데이터 마이닝 모델과 마이닝 구조를 병렬로 처리할 수 있는 정교한 조정 기능을 가진 고성능의 처리 엔진을 제공한다.

SQL Server 2005 Enterprise Edition 에서는 결과로 나오는 모델의 성능, 구성과 정확도에 영향을 주는 추가 알고리즘 패러미터도 제공된다.

장점

병렬 처리는 SQL Server 가 큰 데이터세트에 대해 완벽하게 예측하고 더 빨리 데이터 마이닝 모델 작업을 할 수 있게 하여, 마이닝 모델이 질의와 분석에 더 빨리 사용될 수 있게 한다.

데이터 마이닝 알고리즘에 대한 고급 구성과 튜닝 옵션과 함께, Enterprise Edition 은 데이터 마이닝 모델링에서 가장 좋은 성능과 정확도를 제공한다. 알고리즘 패러미터의 기본 값을 변경하면, 더 많은 시스템 자원이 모델 처리에 할당된다. 알고리즘을 완벽하게 통제할 필요가 있는 고급 사용자는 최고의 결과를 만들어낼 수 있다. 예를 들면, 시계열 모델을 교육시킬 때에는, 월 수준에서 파악된 데이터를 사용할 수도 있다. PERIODICITY_HINT 패러미터가 12 로 설정되면, 매년 12 기간(월) 패턴이 반복되는 것을 지정하는 것이다. 이렇게 변경하면 더 정확한 모델을 만들 수 있다.

Server 2005 Enterprise Edition 옵션은 필요한 산출물로 더 좋은 데이터 모델을 디자인해서 정교하게 튜닝된 데이터 마이닝 알고리즘을 얻을 수 있게 해준다.

무제한의 동시 데이터 마이닝 질의

SQL Server 2005 Enterprise Edition 의 데이터 마이닝은 무제한의 동시 데이터 마이닝 질의를 지원하는 기능에서 볼 수 있듯이, 고성능과 확장성을 위해 디자인되었다.

대부분의 데이터 마이닝 프로젝트의 주 목적은 새로운 데이터에 대한 예측을 하는 것이다. 예를 들면, 다음 해 12 월 중에 회사가 얼마나 많은 제품을 판매할 것인지 또는 광고 캠페인으로 인해 잠재 고객이 제품을 구매할 것인지 예측하고 싶을 것이다. 이런 유형의 솔루션은 많은 동시사용 애플리케이션에서 사용될 수 있으며 수 많은 동시 사용자가 필요할 수도 있다.

SQL Server 2005 Enterprise Edition 은 동시 사용자 예측 질의 수를 제한하지 않지만, Standard Edition 은 그 수를 5 개로 제한한다.

장점

SQL Server 2005 Enterprise Edition 은 가장 많이 사용되고 대규모 데이터 마이닝 구현을 지원하도록 디자인되었다. 고성능의 예측이 필요한 경우에는 무제한의 동시 데이터 마이닝 질의가 반드시 필요하다. 예를 들면, 수 많은 동시 고객을 가진 대규모 소매업 웹 사이트는 연결 알고리즘을 사용해서 고객이 이미 구매한 항목에 따라 추가 제품을 실시간으로 추천할 수 있다. Enterprise Edition 은 무제한의 동시 데이터 마이닝을 지원한다. 이 에디션은 수 많은 동시 비정규 질의를 처리해야 하는 애플리케이션을 지원할 수 있는 성능과 확장성을 제공한다.

연결 규칙을 위한 무제한의 특성

Microsoft SQL Server 2005 Analysis Services 에서 제공되는 연결 알고리즘인 Microsoft Association 알고리즘은 추천 엔진 (recommendation engine)에 큰 도움이 된다. 추천 엔진은 고객들이 이미 구매한 항목에 따라 또는 관심이 있다고 표시한 항목에 따라 제품을 제안한다.

Standard Edition of SQL Server 2005 Analysis Service 의 연결 규칙은 5,000 개 특성으로 제한된다. 그렇지만 SQL Server 2005 Enterprise Edition 은 연결 규칙에서 무제한의 특성까지 확장지원되며, 다양한 데이터 마이닝 경험을 달성하려는 더 큰 환경에 적합하다.

장점

SQL Server 2005 데이터 마이닝 연결 규칙은 특히 교차 판매를 촉진시키는 장바구니 분석에 도움이 된다. 예를 들면, 규칙은 고객의 장바구니에 랩톱이 있고 연장 서비스 계약이 있으면, 고객 장바구니에 여분의 배터리가 있을 가능성이 80%라는 것을 알려줄 수도 있다. 이런 정보를 통해 고객이 장바구니에 가지고 있는 항목에 따라 관심을 유발시킬 추가 항목들을 제안하도록 웹 사이트를 디자인할 수 있다. 웹 사이트 환경이 5,000 개 이상의 항목을 가지고 있다면, SQL Server 2005 Standard Edition 은 더 적은 범주로 항목들을 분류할 수 밖에 없고, 데이터를 더 상세하게 마이닝할 수 있는 능력이 제한된다.

SQL Server 2005 Enterprise Edition 은 Microsoft Association 알고리즘을 사용할 경우에는 연결 규칙에 무제한의 특성을 사용할 수 있게 한다. 5,000 개 이상의 제품, 분류, 다른 특성을 가진 사이트에서 추천 엔진을 구현하려고 할 경우에는, Enterprise Edition 이 정답이다. 이 에디션은 사용자의 비즈니스와 함께 성장하며 더 자세한 수준의 분석을 지원한다.

Notification Services

Enterprise Edition of SQL Server 2005 Notification Services 를 사용해서 가장 많이 사용되는 알림(notification) 애플리케이션을 개발하고, 전개하고, 관리할 수 있다. Enterprise Edition 만이 모든 유형의 알림 아키텍처를 지원하는 완벽한 확장성을 가진 솔루션을 구현할 수 있는 유연성을 제공한다.

여기에서는 다음과 같은 내용을 설명하고 있다:

Notification Services 스케일 아웃 전개

멀티캐스트 배달(Multicast Delivery)

조정가능한 알림 일괄처리 크기(Adjustable Notification Batch Size)

인스턴스 당 증가된 배포자와 생성자 쓰레드 수

Notification Services 스케일 아웃 전개

Microsoft SQL Server 2005 Notification Services Enterprise Edition 은 여러 서버에 걸쳐서 Notification Services 엔진 컴포넌트의 스케일 아웃 토폴로지를 지원한다. 생성자, 이벤트 공급자, 배포자는 별개의 서버에 위치해서, 더 큰 애플리케이션과 복잡한 서식을 가지고 있는 애플리케이션의 성능을 크게 높여준다. 이 밖에, SQL Server Enterprise Edition 스케일 아웃 모델은 호스트되지 않은 이벤트 공급자와 구독 관리 애플리케이션을 별도의 서버에서 찾는다. SQL Server 2005 Standard Edition 은 모든 Notification Services 컴포넌트가 한 서버에 설치되어야만 한다.

장점

Notification Services 는 스케일 아웃 전개를 통해, 사용가능한 모든 자원을 활용하고 알림 양을 극대화하도록 설치되고 구성될 수 있다. Enterprise Edition of Notification Services 는 단일 서버의 사용가능한 쓰레드간에 작업 항목들을 분리시키지 않고, 여러 서버에 작업을 분산시켜서 하드웨어를 스케일 아웃해서 자원의 병목현상을 줄인다.

예를 들면, 대용량 애플리케이션에서는 서식 알림이 프로세서를 집중적으로 사용하고, 알림 배포는 배포를 알리는 시스템까지의 충분한 대역폭을 필요로 하기 때문에 배포자가 종종 뒤에 있을 수 있다. 배포 병목현상을 최소화하려면, 애플리케이션 정의에 여러 배포자를 정의해서 여러 서버 간에 배포자를 확장시킬 수 있다. 배포자는 배포 작업 항목들을 사용할 수 있게 되면 배포 작업 항목을 처리한다.

그림 28 은 Notification Services 에 3 대의 서버를 사용하는 시스템을 보여주고 있다. 한 대는 생성자와 이벤트 공급자 그리고 다른 두 대는 배포자를 실행하기 위한 것이다.

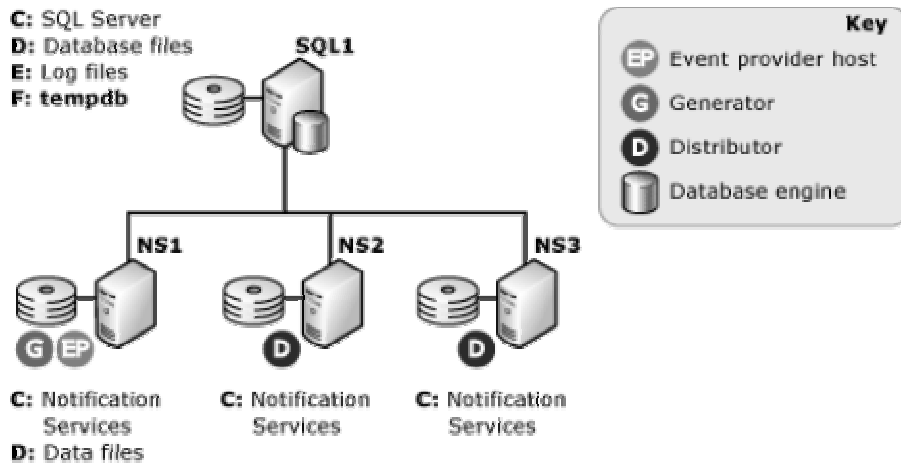


그림 28 SQL Server 2005 Notification Services 스케일 아웃 전개

이 아키텍처에서는, 생성자와 이벤트 공급자가 NS1 에 위치해 있다. 이 두 서버를 스케일 아웃해서, CPU 를 집중적으로 사용하는 콘텐츠 포맷터(formatter)는 CPU 와 네트워크 대역폭을 위해 SQL Server 데이터베이스 또는 생성자와 경쟁하지 않는다. 알림 일괄처리는 서버의 다중 쓰레드와 하드웨어 간의 일괄처리 균형을 통해 병렬로 처리될 수 있다.

Enterprise Edition 에서는, Notification Services 스케일 아웃 전개는 알림 솔루션의 요구에 맞게 Notification Services 구현을 디자인할 수 있는 능력을 제공한다.

SQL Server 2005 Enterprise Edition 은 Notification Services 가 SQL Server 데이터베이스에서 분리된 간단한 아키텍처에서 여러 서버가 솔루션의 요구를 충족시키는 대형 아키텍처에 이르는 모든 환경에 적합한 선택이다.

멀티캐스트 배달

SQL Server 2005 Notification Services Enterprise Edition 만이 같은 내용의 알림을 모아서 멀티캐스트 배달을 통해 여러 구독자에게 하나의 알림을 할 수 있다. 멀티캐스트 배달 기능은 구성되는 알림의 숫자를 줄여서 배포자 성능과 확장성을 높여준다. 멀티캐스트 배달은 기존 하드웨어를 확대하여 사용하고 현재의 전개 시나리오에서 솔루션을 확장하여 서버 사용도를 낮춘다.

장점

배포자가 알림 일괄처리를 하면, 콘텐츠 포맷터에게 전송되는데, 일반적으로 이 작업은 CPU 를 집중적으로 사용하는 작업이다. 콘텐츠 포맷터를 통해 전송된 알림 숫자를 줄인다는 것은 콘텐츠 포맷터가 처리할 작업이 더 적으며, 알림 일괄처리를 처리하는데 필요한 자원을 크게 줄여준다.

멀티캐스트 배달을 사용할 때에는 몇 가지 고려해야 할 것들이 있다:

- 알림은 멀티캐스트와 다이제스트 배달 모두를 활성화시킬 수 없다.
- 서드파티 프로토콜은 멀티캐스트 배달을 지원하지 않을 수도 있다. 배달 프로토콜이 멀티캐스트를 지원하는지 개발자와 협의하여 판단한다. 표준 배달 프로토콜은 멀티캐스트 배달을 지원한다.
- 동일한 알림이 많은 구독자에게 전송되고 있기 때문에, 알림은 구독자 고유의 정보를 가지고 있을 수 없다.

여러 구독자를 위해 동일한 알림 데이터를 만드는 애플리케이션은 멀티캐스트 배달을 사용해서 더 높은 확장성과 성능 향상을 얻을 수 있다. 예를 들면, Notification Services 애플리케이션은 각 각이 수 많은 구독자를 가지고 있는 여러 개의 큰 도시로 기상 예측 업데이트를 전송한다. 동일한 알림을 할 가능성이 높은 이런 전형적인 고용량 시나리오가 멀티캐스트 배달을 사용하는 가장 이상적인 예다.

조정가능한 알림 일괄처리 크기

SQL Server 2005 Notification Services Enterprise Edition 은 알림 일괄처리 크기를 설정할 수 있게 하는데, 이 기능을 통해 알림에 필요한 수준의 서비스 계약을 충족시킬 수 있다.

SQL Server 2005 Standard Edition 에서는, 모든 알림이 한 번의 일괄처리로 생성자를 통해 처리된다. SQL Server 2005 Enterprise Edition 을 사용해서, 규칙 실행 마다 더 많은 일괄처리를 하고 배포자 내에서 병렬로 더 많이 처리할 수 있게 하는 알림 일괄처리 크기를 설정할 수 있다.

알림 일괄처리 설정이 동작하는 것은 다이제스트 배달을 사용해서 알림이 전송되는지에 따라 달라진다. 다이제스트가 아닌 배달 알림의 경우에는, 알림 일괄처리 크기는 각 각의 일괄처리에 있을 수 있는 최대 알림 수를 지정한다. 다이제스트 배달에서는, 알림 일괄처리 크기 설정은 알림 규칙이 동작할 때에 얼마나 많은 일괄처리를 만들어야 하는 지를 지정한다. 규칙이 동작하면, 라운드 로빈(round-robin) 방식으로 각 각의 일괄처리에 알림이 추가된다.

알림 일괄처리가 배포자에게 전송되면, 그 내용은 순차적으로 처리된다. 일괄처리 수를 늘리면 Notification Services 가 배포자에게 전송하는 단위 수를 늘린다. 이 작업 단위는 병렬로 처리되기 때문에 Notification Services 는 멀티쓰레드 배포자뿐만 아니라 스케일 아웃 전개에서의 다중 배포자를 활용할 수 있게 된다.

예를 들면, 구독 규칙이 일반적으로 두 개의 배달 채널에 대해 5,000 개의 알림을 만든다면, Notification Services 의 기본 동작은 하나의 규칙 동작 당 두 개의 작업 항목을 만들 것이다. 이렇게 되면 두 개의 작업 단위 또는 작업 항목만을 만든다. 한 배포자 쓰레드는 각 각의 작업 항목 내에서 모든 알림을 순차적으로 처리한다. 그렇지만, 알림 일괄처리 크기를 1,000 개의 알림으로 제한한다면, Notification Services 는 동일한 5,000 개의 알림을 위하여 5 번의 일괄처리를 만든다. 두 개의 배달 채널에 대해, 규칙 동작 당 10 개의 작업 항목을 만든다.

장점

SQL Server 2005 Notification Services Enterprise Edition 는 모든 유형의 작업에 대해 가장 우수한 성능을 제공하는 환경을 구성한다. 일괄처리 크기를 구성하고 조정하여, 일괄처리의 병렬 처리를 조정해서 시간에 민감한 알림이 지연되지 않고 배달되게 할 수 있다.

인스턴스 당 증가된 배포자와 생성자 쓰레드

SQL Server 2005 Notification Services Enterprise Edition 은 배포자 또는 생성자가 작업 항목을 처리하기 위하여 사용할 수 있는 무제한의 쓰레드 수를 지정할 수 있다. 반대로, SQL Server 2005 Standard Edition 은 배포자와 생성자에 대해 최대 3 개의 쓰레드로 제한된다.

장점

전반적으로, Notification Services Enterprise Edition 에서 배포자와 생성자 쓰레드를 조정하게 되면 애플리케이션의 필요에 따라 알림 솔루션을 확장하거나 축소한다. 확장은 솔루션이 추가 서버 자원을 활용하고 배포자와 생성자 내에서 병렬처리를 늘리게 된다. 축소는 솔루션이 서버 자원 사용을 줄여서 시스템에 대한 요구를 줄이고 다른 프로세스가 방해받지 않게 한다.

쓰레드 사용도가 더 커진다는 것은 더 많은 양의 알림을 병렬로 배포자와 생성자가 모두 처리할 수 있다는 것을 의미한다. 쓰레드 당 한 개의 일괄처리만을 처리할 수 있는 배포자는 더 높은 정도의 병렬처리를 할 수 있게 된다. 사용할 수 있는 생성자 쓰레드 수가 증가하면 동시에 동작할 수 있는 규칙 수가 늘어난다.

SQL Server 2005 Notification Services Enterprise Edition 은 배포자 또는 생성자가 서버에서 사용하고 있는 최대 자원을 통제할 수 있다. 축소는 쓰레드 전환이 발생할 수 있는 큰 쓰레드 풀로 사용 환경에 도움을 줄 수 있다. 배포자가 다른 애플리케이션과 처리 공간을 공유하는 시스템에서 배포자의 프로세서 시간 사용의 최저치를 지정할 수 있다.

결론

사용자의 애플리케이션이 최고의 확장성, 가용성과 성능을 요구한다면, SQL Server 2005 Enterprise Edition 이 최고의 선택이다. 종합적이고 풍부한 기능들은 사용자의 가장 많이 사용되는 데이터베이스 애플리케이션을 지원한다.

더 자세한 내용은:

SQL Server 홈 페이지: <http://www.microsoft.com/sql/default.mspx>

SQL Server Books Online: <http://msdn2.microsoft.com/en-us/library/ms130214.aspx>