

Informix Dynamic Server(IDS)는 높은 신뢰성을 필요로 하는 기간계 시스템의 구축으로부터 대용량을 필요로 하는 대규모 시스템의 구축까지 넓게 대응하는 RDBMS입니다.

IDS V11 에서는 장애에 대비한 고가용성, 고성능, 내장형 데이터베이스의 세 개를 테마로 많은 확장 기능이 추가되어 성공적인 정보 시스템의 구축에 많은 도움을 드리고 있습니다.

IDS V11 에 있어서의 향상된 기능

대용량, 고가용성, 성능 향상

- HDR 부서버의 다중화
Remote standalone secondary (RSS) 서버 기능은 복수 또한 접속 속도가 낮은 원격지에 구축하는 것이 가능하게 되어 있어 데이터센터 재해에 대해서도 계속 실행을 가능하게 하고 있습니다.
- 디스크 공유
Shared disk secondary (SDS) 서버 기능은 여러 대의 서버가 하나의 Disk를 공유하는 것으로, 서버 장애에 있어서의 지속적인 서비스 제공을 가능하게 하고 있습니다.
- HDR의 암호화 통신
HDR에 대해 부서버에 보내는 데이터를 암호화할 수 있게 되었습니다. 이 기능을 실현하기 위해서 새로운 파라미터 ENCRYPT_CDR 가 추가되었습니다.
- 엔진 기동 시간 개선
RTO(recovery time objective) Policy에 근거해 엔진의 장애 복구 시에 있어서의 기동 시간을 지정으로 엔진기동시간이 단축되었습니다.
지정 시간은 새로운 파라미터인 RTO_SERVER_RESTART로 지정한 후, 이 지정된 시간에 맞추어 엔진 내부에서 자동적으로 지정된 시간 내에 엔진을 기동할 수 있도록, 버퍼의 디스크에의 플래시가 자동적으로 조정됩니다.
- Non-Broking Checkpoint
IDS 11.1에서는 퍼지 체크 포인트의 메커니즘을 크게 개선했습니다. 이것에 의해 체크 포인트 발생시에도, 어플리케이션은 그 처리를 계속해 실행하는 것이 가능하게 되었습니다. IDS는 시스템의 작업의 부담량을 감지해 퍼지 체크포인트 시에 어플리케이션이 블록 되는 상황을 자동적으로 제거합니다.

백업 기능의 강화

- Ontape의 디렉토리 백업
ontape에 대해 백업 데이터를 디렉토리 아래에 작성하는 것이 가능하게 되었습니다. 디렉토리 아래에 작성되는 백업 데이터의 명칭은 자동적으로 시스템이 결정되어, 전회 작성된 백업 데이터는 다른 이름으로 자동적으로 변경됩니다. 복원 시, ontape는 자동적으로 디렉토리 아래의 백업 데이터파일로부터 최근의 상태까지 복원합니다.
- 점진적 논리 로그 복구 기능
주서버로 작성된 논리 로그를 부서버로 계속적으로 복원하는 것이 가능하게 되었습니다. 이것에 의해 주서버에 장애 상태가 되었을 경우에는, 부서버가 즉시 주서버가 되어, 어플리케이션에 대해서 계속 실행을 실현합니다.

통합화 된 솔루션 환경

- Deployment 마법사

사용 목적에 따라 필요한 기능만을 상세 선택해 설치하는 것이 가능하게 되었습니다. 인스톨에 필요한 디스크 용량이나 시간을 단축할 수 있게 됩니다. 자사개발의 어플리케이션에 맞추고, 필요 최소한의 기능을 선택해 인스톨 하거나 한 번 인스톨 한 구성을 복수의 거점에 전개하거나 하는 것이 가능하게 되기 위해, 어플리케이션 전개의 효율을 높일 수 있습니다.
- Visual Studio 2005 Add-in

Visual Studio 2005 애드 인(Add In) (IBM Database Add-ins for Visual Studio 2005)를 제공합니다. Visual Studio 2005의 풍부한 개발 지원 기능을 활용해 데이터베이스와 연계한 어플리케이션 개발을 효율적으로 진행하는 것이 가능하게 됩니다.
- 트랜잭션 처리 중 읽기 기능 개선

종래의 COMMITTED READ(확정 읽기)에 비해 여러 어플리케이션의 병렬 처리를 위한 새로운 읽기 방법을 지원합니다. LAST COMMITTED읽기를 지원함으로써 OLAP계 어플리케이션과 OLTP 처리를 공존시켰을 경우의 어플리케이션 성능 개선을 기대할 수 있습니다. 이 읽기 방법은 트랜잭션 진행 중 마지막으로 성공한 데이터를 찾아 읽는 방법입니다.
- 성능의 강화

CPU 가상 프로세서마다 전용의 작업 메모리를 할당하기 위한 파라미터가 준비되었습니다. 유저 세션의 증가에 수반하는 범용 작업 메모리의 경합 상태를 경감해, 프로세서를 보다 효율적으로 이용할 수 있게 되었습니다.
- DataBlade 추가 확장
 - Node Datablade

노드형으로 불리는 새로운 데이터형을 지원합니다. 계층 구조를 가지는 데이터를 RDB로 표현해 고속으로 데이터를 처리할 수 있게 되었습니다
 - Basic Text Search Datablade

BLOB, CLOB 등 비정형 데이터에 대한 단어 및 문장 단위로의 검색 기능을 제공합니다.
- OLTP 기능의 강화

인덱스에 의한 고속 검색 가능한 특수한 바이너리(binary) 표현 데이터형을 제공합니다.
- SQL의 강화

SELECT문의 결과를 다른 SELECT문의 FROM절로서 참조할 수 있게 되었습니다.

테이블이나 뷰에 대한 트리거 설정의 자유도가 향상하고 있습니다.

예를 들면, 하나의 테이블에 대해서 INSERT, DELETE, UPDATE 및 SELECT 트리거를 복수 설정하는 것이 가능하게 되었습니다.

ANSI/ISO 준거의 SELECT문에의 옵티마이저 힌트문의 대응을 강화했습니다. 복수의 데이터베이스 서버에 걸치는 SQL 액세스(분산 액세스)에 대해서, 유저 정의 루틴(UDR)의 사용이 가능하게 되었습니다.

복합 인덱스를 사용한 검색에 대해서, 인덱스를 보다 활용하는 최적화를 실시합니다.
- XML 퍼블리싱 기능

SQL의 실행 결과를, XML 어플리케이션으로 이용할 수 있는 형식의 XML로 변환하는 것이 가능하게 되었습니다.

업계표준의 XML에 준거한 각종의 통과의 제유성이 향상했습니다.

- OGC 준거 Web Feature Service에 의한 위치 정보의 지원
Spatial DataBlade나 Geodetic DataBlade의 presentation layer로서의 기능을 데이터베이스서버내에 DataBlade로서 구현했습니다. Open Geospatial Consortium가 표준화 한 GML(XML 베이스의 위치 정보 마크업 언어)를 사용한, 업계표준에 근거하는 위치 정보 검색이 가능하게 됩니다.
- Advanced access control feature
열 혹은 행에 의한 라벨 기반의 접근 제어합니다. 데이터베이스에 저장되고 있는 오브젝트에 대해서 '라벨'을 할당하는 것과 동시에, 액세스하는 각 유저에게도 적절한 '라벨'을 할당하여 각각의 오브젝트(표, 열, 행 등)에 대한 개별 유저의 액세스권한을 상세하게 설정합니다.

관리 작업의 대폭적인 개선

- SQL 관리 API
SQL 관리 API가 새롭게 내장되어 데이터베이스 서버 관리자는 관리 태스크를 명령어가 아니고, SQL문의 실행으로 실현될 수 있게 되었습니다. SQL를 리모트 쿼리-형식에서 실행하는 것으로, 리모트 서버의 관리를 직접 머신에 로그인하는 일 없이 실현될 수 있습니다.
데이터베이스서버 관리자는, 빌트인 함수인 admin()나 task()에 대해서, 커멘트 라인으로의 관리 태스크와 동등이라고 정의되는 인수를 이것들 함수에게 주어 실행할 수 있습니다. 예를 들면 이하의 함수의 실행은 "oncheck -ce"를 실행한 것과 같은 결과가 됩니다.

```
EXECUTE FUNCTION admin('check extents');
```

- 관리 태스크의 스케줄링
관리 태스크의 스케줄링 기능에 의해서 유지보수 작업, 감시 작업, 및 관리 태스크를 결정할 수 있던 시간이나 시간 간격으로 실행하는 것이 가능하게 되었습니다. 또 이러한 실행 결과의 내용을 확인해, 필요하면 수정하기 위한 액션을 재차 실행할 수도 있습니다.
이러한 관리 기능의 자산은, 새롭게 내장된 sysadmin 데이터베이스의 ph_task 테이블에 등록되어 내용을 수정하거나 새로운 행을 추가하는 것으로 신규의 관리 태스크를 작성할 수 있습니다.
- 실행 SQL문의 모니터링과 트레이스
실행된 SQL문의 성능 상황을 모니터링 할 수 있습니다. 정보는 각각의 SQL문의 실행에 있어서의 통계학적 정보로서 제공되어 버퍼상에 존재합니다.
데이터베이스 관리자는, 이 기능을 인스턴스상의 모든 유저가 실행한 SQL문이나, 어느 특장의 유저가 실행한 SQL만 트레이스할 수 있습니다. 이 기능에 의해서 SQL의 성능 튜닝이 매우 간단하게 실현될 수 있게 됩니다.
- ER의 강화
ER을 위한 구성 파라미터나 환경 변수를, 가동중의 서버로 동적으로 추가, 변경 및 삭제하는 것이 가능해졌습니다. 이러한 기능 실현을 위해 이하의 커멘드가 새로 추가되고 있습니다.

cdr add config: to add a value

cdr change config: to change an existing value

cdr remove config: to remove an existing value

Enterprise replication 실행 중에 복제하고 있는 데이터베이스, 테이블 및 컬럼명을 동적으로 변경하는 것이 가능해졌습니다. 이 변경에는 RENAME문을 사용하여 간단히 작업할 수 있다.

- Web 베이스의 새로운 관리툴

PHP 베이스의 새로운 관리툴인 IDAdmin가 번들 되었습니다. SQL 관리 API를 이 툴로부터 실행하는 것으로 간단하게 복수 인스턴스의 관리를 실현될 수 있게 됩니다.

- 개별 세션에 대한 프로시저 할당

sysdbopen()와 sysdbclose()라는 이름의 새로운 내장된 SPL을 제공되었습니다. 이러한 SPL은 유저가 데이터 베이스에 접속 및 절단 되었을 때에 자동적으로 실행되어 유저 고유의 처리를 지정할 필요가 있는 경우에는, 매우 유용한 기능이 됩니다.

예를 들면 이하의 정의에 대해서는, usr1가 데이터베이스에 접속할 때에 oltp라고 하는 롤을 세트 해, 한편 isolation 레벨을 확인 읽기로서 접속을 시도합니다.

```
create procedure usr1.sysdbopen()
set role to oltp;
set isolation to committed read;
end procedure;"
```

- 자동 통계 정보 수집

자동적으로 통계 정보를 수집하는 것이 가능하게 되어, 수동으로의 UPDATE STATISTICS의 실행의 필요성을 없애는 것이 가능하게 됩니다.이 기능은 TEXT형과 BYTE형을 제외한 모든 빌트인·데이터 타입과 유저 정의형에 대해서 적용됩니다.B-Tree 인덱스가 작성되었을 때에, 인덱스의 최초의 컬럼에 대해서 자동적으로 UPDATE STATISTICS MEDIUM 와 동등의 통계 정보가 수집.

EXPLAIN_STAT 파라미터에 1을 설정하는 것으로,""sqexplain.out"" 파일내에 실행된 문의의 통계 정보를 참조할 수 있게 되었습니다.또, UPDATE STATISTICS 문의 MEDIUM에 대해서 실제로 샘플링 하는 행수의 지정이 가능한, SAMPLING SIZE 옵션이 추가되었습니다.

- 관리 태스크의 정의

관리 태스크를 실행하는 유저를 복수 정의할 수 있게 되었습니다. 이전의 버전에서는 관리 태스크는 인포믹스 사용자나 데이터베이스서버 관리 그룹에 속하는 유저만이 실시 가능했습니다만, ADMIN_MODE_USERS 파라미터에 유저명을 기술하는 것으로, 관리 태스크를 실행 가능한 유저로서의 지정이 가능해집니다.