

Informix Dynamic Server  
2008년 5월



IBM Information Management software



## IDS 11.5 기능 개요



# CONTENTS

<b>1. High Availability</b>	<b>3</b>
1.1 High-Availability 클러스터를 구성하는 SQL Admin API 명령	3
1.2 High-Availability 클러스터 Secondary 서버에서 업데이트 지원	3
1.3 High-Availability 클러스터용 Connection Manager	3
1.3.1 클라이언트 연결 관리	3
1.3.2 오류 해결 조정	4
1.3.3 다중 이중화	4
1.3.4 연결 보안	4
1.3.5 Connection Manager 모니터링	5
1.3.6 High Availability를 위해 오류 해결 스크립트 지원	5
1.4 클러스터 Secondary 서버에서 임시 데이터 형식 지원	5
<b>2. 설치 및 구성</b>	<b>6</b>
2.1 데모 서버의 구성 가능한 옵션	6
2.2 설치 중에 DRDA 프로토콜 구성	8
2.3 개선된 onconfig.std 및 새로운 기본 구성 값	8
<b>3. 성능</b>	<b>8</b>
3.1 버전 컬럼으로 데이터의 최신 여부 결정	8
3.2 Large Integers 및 Serial Types에 대해 개선된 쿼리 성능	9
<b>4. 관리 및 모니터링</b>	<b>9</b>
4.1 OpenAdmin Tool for IDS의 기능 향상	9
4.2 확장된 데이터 서버 클라이언트 세션 정보	10
4.3 XML 형식으로 제공되는 Explain 출력	11
4.4 더욱 정확한 공유 메모리 덤프 파일 크기 제어	13
4.5 향상된 서버 시작	13
<b>5. 새로운 SQL 기능</b>	<b>14</b>
5.1 SPL 루틴을 사용하는 동적 실행	14
5.2 IS [NOT] NULL 술어가 있는 SQL 식	15
5.3 세션의 외부 디렉티브 제어	15
5.4 DRDA(Distributed Relational Database Architecture) 기능 향상	15
<b>6. 보안</b>	<b>16</b>
6.1 Single Sign-on 지원	16
6.2 SSL(Secure Sockets Layer)로 데이터 통신 암호화	16
<b>7. XML 기능</b>	<b>17</b>
7.1 XML 쿼리	17
7.2 XSL Transformation	19
<b>부록 A - 구성 파일 변경사항</b>	<b>20</b>
onconfig.std에 추가된 구성 매개변수	20
onconfig.std에서 업데이트된 구성 매개변수	21
onconfig.std에서 제거된 구성 매개변수	23

## [ Introduction ]

2007년 IBM은 가용성, 성능, 관리 등의 모든 데이터 관리 분야에서 모든 새로운 기능을 포함하는 IDS 11을 출시했습니다. 고급 데이터 관리를 위한 IDS 11의 다음 단계는 IDS 11.5입니다. IDS 11.5는 IDS의 모든 기존 장점과 더불어 더 많은 기능을 추가로 제공합니다. 본 문서는 MACH11에 관련된 새로운 기능을 살펴보는 IDS 11.5의 모든 새로운 기능에 대한 종합적인 백서입니다.

# 1. High Availability

## 1.1 High-Availability 클러스터를 구성하는 SQL Admin API 명령

이제 SQL Administration API를 사용하여 High-Availability 클러스터를 구성할 수 있습니다. API를 사용하여 `onmode -d`와 동일한 기능을 하는 명령을 수행하지만 구문이 약간 다릅니다. 즉 연산 및 매개변수를 지정하기 위해 'HA command string', 'argument'를 사용합니다.

Admin API의 새로운 기능을 사용하여 sysadmin 데이터베이스에서 SQL 명령과 동일한 작업을 수행할 수 있습니다. 이 인터페이스는 Connection Manager가 High-Availability 클러스터 환경을 관리하는 데 사용됩니다.

전체 SQL Administration API 명령 세트는 IBM Informix Dynamic Server Guide to SQL: Syntax에 설명되어 있습니다.

## 1.2 High-Availability 클러스터 Secondary 서버에서 업데이트 지원

클라이언트 애플리케이션이 Secondary 서버에서 데이터를 업데이트할 수 있습니다. 이 기능을 활성화하기 위해 새로운 구성 매개변수가 추가되었습니다.

이 기능은 IBM Informix Dynamic Server Administrator's Guide 및 IBM Informix Dynamic Server Administrator's Reference에 설명되어 있습니다.

## 1.3 High-Availability 클러스터용 Connection Manager

새로운 Connection Manager는 High-Availability 클러스터에 있는 가장 적절한 서버로 클라이언트 애플리케이션 연결 요청을 동적으로 전달합니다. Connection Manager는 클러스터에 있는 각 서버에 연결되어 서버 유형, 미사용 워크로드 용량 및 서버의 현재 상태에 대한 통계를 수집합니다. 이러한 정보를 바탕으로 Connection Manager는 클라이언트 연결을 적절한 서버로 리디렉션합니다. 또한 Connection Manager Arbitrator는 자동 오류 해결(failover) 로직을 High-Availability 클러스터에 제공합니다.

### 1.3.1 클라이언트 연결 관리

Connection Manager는 IDS 인스턴스를 논리적으로 그룹화할 수 있습니다. 각 논리적 그룹을 SLA(Service Level Agreement; 서비스 수준 계약)라고 하며, SLA에는 INFORMIXSERVER 이름 또는 서버 유형 (SDS/RSS/HDR 등)별로 서버 인스턴스의 내역이 포함됩니다.

Connection Manager는 위와 같은 SLA 특성으로 정의된 서비스를 제공하며, 이 서비스를 서버 인스턴스 그룹에 매핑합니다.

Connection Manager는 일반적인 SQL 클라이언트에 대한 연결 지점으로 작동하므로, 자체적으로 sqlhosts 구성 파일에 입력된 값이 있어야 합니다.

```
# Primary server
db_server      onsoctcp      koetsu.lan      9088
db_drda        drsoctcp      koetsu.lan      9089

# SDS servers
db_sds_ro      onsoctcp      koetsu.lan      9098
db_sds_rw      onsoctcp      koetsu.lan      9108

# Connection Manager services
oltp           onsoctcp      koetsu.lan      19188
reports        onsoctcp      koetsu.lan      19189
other          onsoctcp      koetsu.lan      19190
```

클라이언트는 해당 INFORMIXSERVER 환경 변수나 연결 사양의 서버 선언 부분을 reports, oltp 또는 other로 설정합니다.

### 1.3.2 오류 해결 조정

Connection Manager의 또 다른 역할은 클러스터 환경에서 서버 오류 해결을 처리하는 것입니다. 이 작업은 명령행에 -r 플래그를 지정하거나 구성 파일에 FOC(Fail Over Configuration) 매개변수를 사용하여 수행됩니다. 사양은 서버 및 서버 유형의 순서 목록과 타임아웃 값으로 구성됩니다. Connection Manager가 Primary 서버에 고장이 발생한 것을 탐지한 경우, 이후 타임아웃 중에 Primary 서버가 재연결 작업을 수행하지 않으면 FOC 목록에 나열된 그 다음 서버 또는 서버 유형이 Primary 서버의 후보로 선택됩니다.

기본 FOC는 "SDS+HDR+RSS,0"이며, 이는 Primary 서버에 고장이 발생한 경우 SDS 서버가 Primary 서버를 대체하는 첫 번째 후보가 된다는 의미입니다. Connection Manager Arbitrator는 타임아웃 시간(이 경우 0초) 동안 기다린 다음 SDS 서버를 Primary 서버로 전환하는 데 필요한 명령을 생성합니다. SDS 서버를 사용할 수 없는 경우에는 사용 가능한 HDR 서버, RSS 서버 순으로 선택되어 Primary 서버로 전환됩니다.

Connection Manager Arbitrator를 해제하려면 FOC DISABLED를 사용하십시오.

### 1.3.3 다중 이중화

Connection Manager를 임의의 서버 인스턴스에서 로컬로 실행할 필요가 없으며, 서버 컴퓨터에 발생한 고장이 영향을 미치지 않도록 Connection Manager를 다른 물리적 컴퓨터에서 실행할 수 있습니다. 연결 호스트 이름과 포트 번호를 나타내도록 sqlhosts 파일을 올바르게 구성하십시오. 또한 클러스터를 구성하는 IDS 인스턴스와 달리 Connection Manager는 서버 플랫폼이 아닌 다른 하드웨어 플랫폼에서 실행될 수 있습니다.

가용성을 늘리기 위해 여러 개의 Connection Manager를 실행하고 sqlhosts에 있는 서버 그룹의 기본 오류 해결 기능을 사용할 수 있습니다. 또한 이 그룹에 Primary 서버를 추가하면 두 Connection Manager에 고장이 발생하더라도 계속해서 연결 상태를 유지할 수 있습니다.

### 1.3.4 연결 보안

복잡한 네트워크 환경에서 여러 대의 서버를 관리하기 위해서는 Connection Manager가 클러스터 내의 서버에 연결하고, 내부 서버 통계를 모니터링하며, 서버 간의 연결을 처리하고, 오류 해결 상황에서 서버를 관리할 수 있어야 합니다. 이러한 권한 레벨은 Informix 사용자에게만 부여되며, 클러스터의 모든 컴퓨터에 대해 이러한 권한 액세스 레벨을 허용하는 메커니즘이 있어야 합니다.

이러한 메커니즘을 수행하기 위해 클러스터의 서버에 필요한 사용자 및 암호 조합이 암호화된 파일에 저장됩니다. onpassword 유틸리티는 암호화된 파일을 생성하는 데 사용됩니다.

안전한 암호 저장 메커니즘을 사용할 뿐만 아니라 SSL을 사용하여 클라이언트, 서버 및 Connection Manager 간의 모든 연결을 암호화할 수 있습니다. 20페이지의 6.2 SSL(Secure Sockets Layer)로 데이터 통신 암호화 절을 참조하십시오.

### 1.3.5 Connection Manager 모니터링

Connection Manager의 현재 상태가 로그 파일에 기록됩니다. 구성 파일에서 DEBUG 매개변수를 1로 설정하여 메시지의 수를 늘릴 수 있습니다. 기본 설정은 0입니다.

Connection Manager에서 생성된 로그 파일 외에도 Connection Manager의 상태 모니터링에 사용될 수 있는 새로운 onstat -g cmsm 옵션이 있습니다. 모니터링 내역은 sysmaster:syscmsm SMI 테이블을 통해서도 볼 수 있습니다.

Connection Manager는 Client SDK와 함께 설치되며, 주로 IBM Informix Dynamic Server Administrator's Guide 및 IBM Informix Dynamic Server Administrator's Reference에 설명되어 있습니다.

### 1.3.6 High Availability를 위해 오류 해결 스크립트 지원

하드웨어 클러스터 환경에 따라 오류 해결 시 운영 체제의 클러스터 관리 소프트웨어가 특정 작업을 수행해야 합니다. 이것은 디스크 서브시스템을 다시 구성해야 하는 경우에 가장 중요합니다. 새로운 Primary 서버를 시작할 수 있도록 읽기 전용 공유 디스크를 쓰기 가능 디스크로 전환하는 작업을 I/O Fencing이라고 합니다.

O/S 제공 클러스터 관리 소프트웨어를 Connection Manager에 통합하기 위해 새로운 구성 매개변수인 FAILOVER\_CALLBACK이 사용되었습니다. 이 매개변수는 Secondary 서버가 Primary 서버나 표준 서버로 승격될 때마다 호출되어야 하는 스크립트를 지정합니다. 이 스크립트는 Secondary 서버의 상태가 변경되기 전에 호출되며, Primary 서버가 되는 서버 상의 공유 디스크에 대한 쓰기 액세스를 허용하고 고장이 발생한 서버의 공유 디스크에 대한 쓰기 액세스를 해제합니다.

이 기능은 IBM Informix Dynamic Server Administrator's Guide에 설명되어 있습니다.

## 1.4 클러스터 Secondary 서버에서 임시 데이터 형식 지원

Secondary 서버가 읽기 전용이든지 보조 쓰기를 사용하든 상관없이 High-Availability 클러스터 Secondary 서버에서 명명되지 않은 복잡한 임시 데이터 형식(ROW, SET, LIST 및 MULTISSET)을 사용할 수 있습니다.

임시 형식을 사용하는 다음과 같은 종류의 작업이 Secondary 서버에서 지원됩니다.

- 임시 형식을 명시적으로 사용하는 SQL 쿼리
- 파생 테이블, 컬렉션 서브쿼리 및 XML 함수를 사용하는 SQL 쿼리  
(이러한 문에서는 임시 형식을 암시적으로 사용)
- CREATE TEMP 문으로 만들어지고 임시 형식을 사용하는 임시 테이블

이 기능은 대부분 IBM Informix Dynamic Server Administrator's Guide에 설명되어 있습니다.

## 2. 설치 및 구성

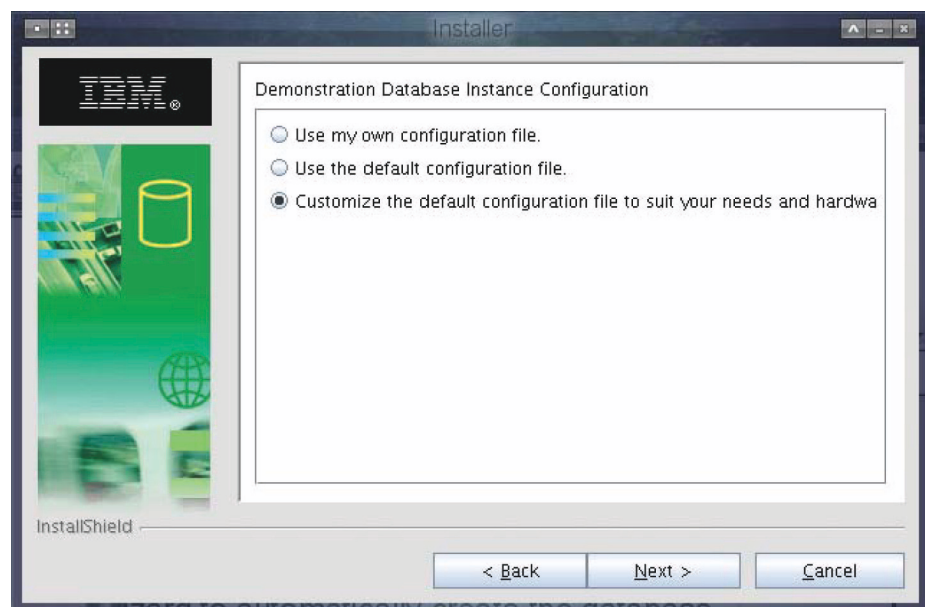
이전보다 더욱 상세한 설치 옵션을 제공하고 데모 서버 구성의 사용자 정의를 허용하기 위해 Java 기반 설치 관리자의 기능이 향상되었습니다.

### 2.1 데모 서버의 구성 가능한 옵션

새로운 인스턴스 구성 마법사를 사용하여 설치 중에 데이터베이스 서버 구성 파일(ONCONFIG)을 자동으로 만들 수 있습니다. GUI 또는 콘솔 설치 모드 중 하나를 사용하여 설치 프로그램의 마법사에 액세스합니다.

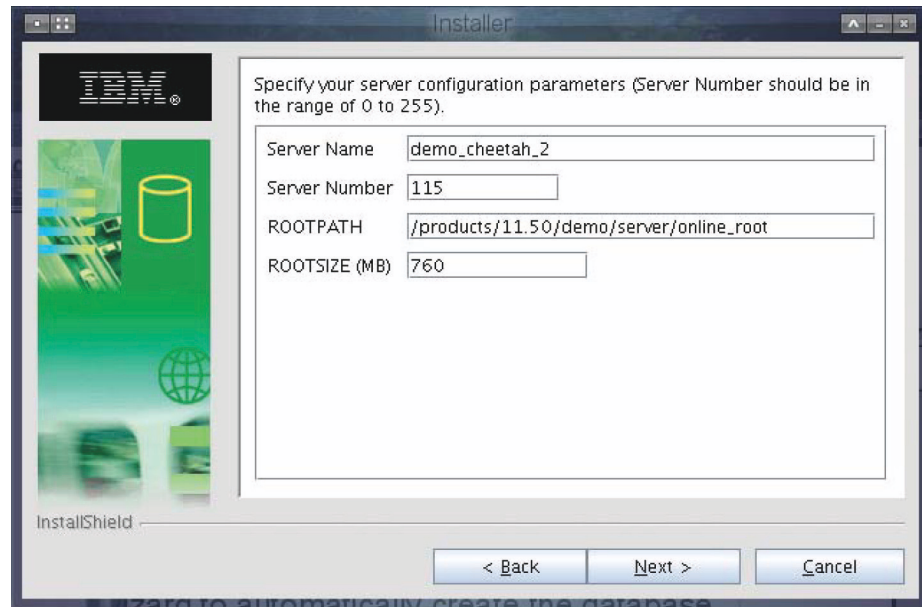
- **UNIX 및 Linux** : 데모 데이터베이스 서버를 만들도록 선택하고 기본 구성 파일을 사용자 정의하도록 선택합니다.
- **Windows** : 사용자 지정 설치 설정을 선택한 다음 "Initialize Server"를 선택하고 하드웨어 및 사용자의 필요에 맞게 기본 구성 파일을 설정하는 옵션을 선택합니다.

예를 들어, 다음은 Linux 설치 GUI의 일부 스크린샷입니다.

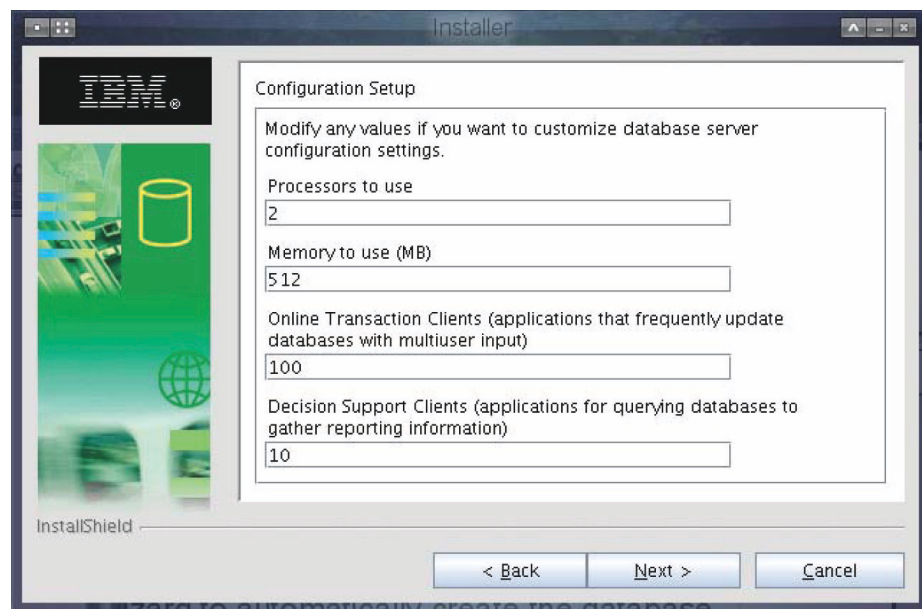


서버 인스턴스의 고유한 식별 정보를 일부 입력한 후 설치 중인 인스턴스에 대해 CPU, 메모리, 디스크 공간, 온라인 트랜잭션 및 쿼리 클라이언트 예측 값과 같은 구성 정보를 제공해야 합니다.





마법사는 설정이 올바른지 확인하고 이 설정에 따라 다른 서버 구성 매개변수의 값을 계산합니다.



사용자 지정 구성 정보는 ONCONFIG 파일에 저장되므로, 제품을 설치한 후 인스턴스를 시작하면 설정에 따라 이 인스턴스가 실행됩니다. 이 서버 인스턴스에 액세스하는 데 필요한 환경 설정은 profile\_settings 스크립트에서 가져올 수 있습니다. 이 스크립트는 서버 데이터 체크 파일 및 로그 메시지와 함께 \$INFORMIXDIR/demo/server 디렉토리에 저장됩니다.

이 기능은 IBM Informix Dynamic Server Installation Guide for Windows 및 IBM Informix Dynamic Server Installation Guide for UNIX, Linux, and Mac OS X에 설명되어 있습니다.

## 2.2 설치 중에 DRDA 프로토콜 구성

이전 릴리스보다 더욱 쉽게 인스턴스를 설정하여 다양한 데이터베이스 클라이언트를 사용할 수 있습니다. IDS 11.5를 설치하면 설치 관리자를 통해 DRDA(Distributed Relational Database Architecture) 프로토콜을 사용하는 클라이언트에 대해 데이터베이스 서버 별칭과 포트를 구성할 수 있습니다. DRDA 지원을 선택하여 취소하지 않는 경우 이러한 항목은 기본적으로 구성됩니다.

이 기능은 IBM Informix Dynamic Server Installation Guide for Windows 및 IBM Informix Dynamic Server Installation Guide for UNIX, Linux, and Mac OS X에 설명되어 있습니다.

## 2.3 개선된 onconfig.std 및 새로운 기본 구성 값

IDS 구성 파일인 onconfig.std가 다음과 같은 두 가지 면에서 개선되었습니다.

주석 및 매개변수가 기능별 영역에 따라 별도로 나열되고 그룹화되기 때문에 onconfig.std 파일을 더욱 쉽게 읽을 수 있습니다.

- 지원되는 대부분의 구성 매개변수가 파일에 포함됩니다.
- 사용되지 않는 구성 매개변수가 제거되었습니다.
- 크기를 지정하는 일부 구성 매개변수가 더 높은 값을 가집니다.
- 파일 위치를 지정하는 일부 구성 매개변수가 \$INFORMIXDIR 디렉토리에 보다 안전하게 저장됩니다.
- 파일 경로를 지정하는 매개변수로 "\$INFORMIXDIR"을 사용할 수 있습니다.

부록 A에는 추가 사항과 변경 사항이 요약된 표가 있습니다. 구성 매개변수의 기본값은 IBM Informix Dynamic Server Administrator's Reference에 설명되어 있습니다.

## 3. 성능

### 3.1 버전 컬럼으로 데이터의 최신 여부 결정

이제 테이블에 버전 컬럼을 추가하여 체크섬과 버전 번호를 모두 포함할 수 있습니다. 버전 컬럼을 사용하면 처음 쿼리된 이후로 행이 업데이트되었는지 여부를 확인할 수 있습니다. 또한 버전 번호를 사용하면, 행이 삭제되고 또 다른 행이 테이블에 다시 삽입된 경우, 그 차이를 확인할 수 있습니다. 버전 컬럼을 추가하려면 ALTER TABLE 문을 사용합니다.

```
ALTER TABLE stock ADD VERCOLS;
```

웹 애플리케이션에서 버전 컬럼을 사용하여 이전에 검색한 개체에 대한 정보가 최신인지 확인할 수 있습니다. 예를 들어, 웹 애플리케이션에 고객의 주문을 표시할 수 있습니다. 그러면 고객이 구매 완료를 결정하고 애플리케이션이 품목의 원래 재고에 대한 버전 컬럼을 확인하여 해당 품목에 관한 내역 정보가 변경되었는지 여부를 알 수 있습니다. 간단한 단일 Select 문을 사용하여 최신 정보를 수집한 후 확인할 수 있습니다. 이때 주문 품목에 있는 모든 행을 검사하고 해당되는 재고 품목과 비교할 필요가 없습니다.



```
SELECT SUM (stock.ifx_row_version)
FROM stock, items
WHERE stock.stock_num = items.stock_num
      AND stock.manu_code = items.manu_code
      AND items.order_num = 1023;
```

버전 컬럼은 또한 업데이트 가능 Secondary 서버(Updatable Secondary 서버)가 활성화된 경우 High-Availability 클러스터의 성능을 개선합니다.

숨겨진 컬럼 ifx\_row\_version 및 ifx\_insert\_checksum을 사용하여 새로운 정보를 개별적으로 얻을 수 있습니다. 또한 테이블에서 VERCOLS가 활성화된 경우 이러한 열은 idx\_row\_id에도 포함됩니다. 해당 열의 형식은 다음과 같습니다.

- partnum:rowid:ifx\_insert\_checksum:ifx\_row\_kversion

이 기능은 IBM Informix Dynamic Server Guide to SQL: Syntax 및 IBM Informix Dynamic Server Administrator's Guide에 설명되어 있습니다.

### 3.2 Large Integers 및 Serial Types에 대해 개선된 쿼리 성능

INT8 및 SERIAL8의 대안으로 ISO 표준 데이터 형식 BIGINT 및 BIGSERIAL이 제공됩니다. BIGINT 및 BIGSERIAL 데이터 형식은 INT8 및 SERIAL8 데이터 형식보다 성능이 우수합니다.

기본 유형 연산이 INT8 연산보다 빠르기 때문에 내부 표현식은 더 신속하게 실행될 수 있습니다. 예를 들어, INT8 대신 BIGINT 데이터 형식을 사용한 경우, 큰 테이블에서 열 값을 비교하는 간단한 테스트 속도가 20% 더 빨라졌습니다.

이 기능은 대부분 IBM Informix Dynamic Server Guide to SQL: Reference에 설명되어 있습니다.

## 4. 관리 및 모니터링

### 4.1 OpenAdmin Tool for IDS의 기능 향상

OpenAdmin Tool for IDS에서 다음과 같은 기능이 향상되었습니다.

- 향상된 High-Availability 클러스터 관리(MACH) 지원 :
  - 클러스터에서 원격 서버 시작 및 종료
  - idsd용 암호화
  - Connection Manager 마법사
  - SD 보조 디스크 설정 마법사
- 새로운 보안 기능 :
  - 읽기 전용 사용자 그룹 지원
  - 사용자 권한 관리
  - 스케줄러 작업을 만들고 수정하는 새로운 작업 스케줄러

- 새로운 서버 관리 도구 :
  - 자동화된 통계 업데이트
  - 구성 매개변수의 동적 업데이트
  - 시스템 무결성 검사
  - 가상 프로세서 관리
  
- 개선된 성능 모니터링 도구 :
  - 개선된 과거 성능 데이터 추적
  - 새로운 시스템 보고서
  - 메모리 풀 통계
  
- 개선된 사용성 :
  - 메뉴 재구성
  - 새로운 모양과 느낌
  - 새로운 상황별 도움말 항목
  - HOWTO.html 작업 안내서

자세한 내용은 OpenAdmin Tool 릴리스 노트를 참조하십시오.

#### 4.2 확장된 데이터 서버 클라이언트 세션 정보

sysmaster 데이터베이스에서 새로운 sysappinfo 테이블을 사용하여 데이터 서버 클라이언트 세션 정보를 볼 수 있습니다. 이 테이블은 클라이언트 세션 ID, 세션 애플리케이션 이름 및 세션 값을 sesapp\_sid, sesapp\_name 및 sesapp\_value 컬럼에 표시합니다.

또한 onstat -g ses 명령을 사용하여 클라이언트 세션 정보를 표시할 수도 있습니다. 여기서 DRDA 프로토콜을 사용하여 연결 중인 Data Studio 세션의 정보를 볼 수 있습니다.

```

Sess      SQL          Current      Iso Lock     SQL  ISAM F.E.
Id        Stmt type   Database     Lvl Mode    ERR  ERR  Vers   Explain
20        SELECT      stores_demo  DR  Not Wait  0    0    -  Off

Current statement name : sql_cursn100c5

Current SQL statement :
SELECT * FROM "simond"."customer" JOIN "simond"."state" ON
"simond"."customer"."state" = "simond"."state"."code" WHERE
"simond"."customer".CUSTOMER_NUM < 105

Last parsed SQL statement :
SELECT * FROM "simond"."customer" JOIN "simond"."state" ON
"simond"."customer"."state" = "simond"."state"."code" WHERE
"simond"."customer".CUSTOMER_NUM < 105

DRDA client info
Userid:
Wrkstname: koetsu.lan
Applname: db2jcc_application
Acctng: JCC03500koetsu.lan
Programid:
Autocommit:
Packagepath:
    
```

또한 sysmaster 데이터베이스에서도 확장된 DRDA 클라이언트 정보를 볼 수 있습니다.

```
select * from sys sesappinfo where sesapp_sid = 20 and sesapp_value is not null;

sesapp_sid  20
sesapp_name Acctng
sesapp_value JCC03500koetsu.lan

sesapp_sid  20
sesapp_name Applname
sesapp_value db2jcc_application

sesapp_sid  20
sesapp_name Wrkstnname
sesapp_value koetsu.lan

3 row(s) retrieved.
```

sys sesappinfo 테이블과 onstat -g ses 명령은 IBM Informix Dynamic Server Administrator's Reference 에 설명되어 있습니다.

#### 4.3 XML 형식으로 제공되는 Explain 출력

EXPLAIN\_SQL()은 쿼리 Explain 출력을 XML 형식으로 얻기 위해 IBM Data Studio Administration Console 에서 사용할 수 있는 새로운 내장 루틴입니다. 공급업체는 이 루틴을 사용하여 IDS용 Visual Explain 정보를 만들 수 있습니다.

예를 들어, 아래와 같은 쿼리 계획이 생성된 일반적인 메커니즘에서는 쿼리를 실행하지 않고 - '{+ AVOID\_EXECUTE}' 계획을 가져오는 디렉티브로 쿼리를 작성한 다음 서버 컴퓨터의 \$HOME 디렉토리에서 sqexplain.out 파일로부터 쿼리 계획을 추출했습니다. 이 방법은 일반적으로 클라이언트-서버 환경에는 부적합합니다.

또한 수집되는 정보가 상당히 부족하며, 각 테이블에서 사용되는 조인 순서 및 액세스 방법을 제외한 다른 내역을 얻을 수 없습니다.

```
QUERY: (OPTIMIZATION TIMESTAMP: 03-10-2008 17:47:21)
-----
SELECT {+ AVOID_EXECUTE }
* FROM customer, state, orders, items
WHERE customer.state = state.code
AND orders.customer_num = customer.customer_num
AND orders.order_num = items.order_num
AND customer.customer_num = 101

DIRECTIVES FOLLOWED:
AVOID_EXECUTE
DIRECTIVES NOT FOLLOWED:

Estimated Cost: 8
Estimated # of Rows Returned: 3

1) simond.customer: INDEX PATH
(1) Index Keys: customer_num (Serial, fragments: ALL)
Lower Index Filter: simond.customer.customer_num = 101
```

```

2) simond.state: INDEX PATH
  (1) Index Keys: code (Serial, fragments: ALL)
      Lower Index Filter: simond.customer.state = simond.state.code

NESTED LOOP JOIN

3) simond.orders: INDEX PATH
  (1) Index Keys: customer_num (Serial, fragments: ALL)
      Lower Index Filter: simond.orders.customer_num = simond.customer.customer_num

NESTED LOOP JOIN

4) simond.items: INDEX PATH
  (1) Index Keys: order_num (Serial, fragments: ALL)
      Lower Index Filter: simond.orders.order_num = simond.items.order_num

NESTED LOOP JOIN
    
```

이제 Data Studio의 Visual Explain Analyzer를 사용한 동일한 쿼리를 살펴보겠습니다. 시각적으로 배치될 경우 정보가 훨씬 명확하게 표시되며, 각 노드 레벨에서 상세 정보를 볼 수도 있습니다.

The screenshot displays the Visual Explain Analyzer interface. On the left, the 'Overview of Diagram' panel shows a tree view of the execution plan. The main canvas shows a detailed execution plan with nodes for Query, Queryblock, NLJoin, Index Scan, and Index. A 'Description of Selected Node' dialog is open, showing details for an NLJoin node. Below the plan, the 'SQL Statement' window shows the following query:

```

SQL Statement 1
SELECT *
FROM
  customer, state, orders, items
WHERE customer.state = state.code
AND orders.customer_num = customer.customer_num
AND orders.order_num = items.order_num
AND customer.customer_num = 101;
    
```

#### 4.4 더욱 정확한 공유 메모리 덤프 파일 크기 제어

DUMPSHMEM 구성 매개변수 및 onstat 유틸리티에 새로운 옵션을 사용하여 덤프 파일에 기록되는 메모리의 크기를 제어할 수 있습니다. 이러한 옵션은 상주 메모리에서 버퍼 풀을 제외하므로, 파일 크기가 훨씬 작아질 수 있습니다.

어설션(assertion)이 실패할 때 DUMPSHMEM 구성 매개변수를 사용하여 덤프 파일을 자동으로 만듭니다. DUMPSHMEM을 2로 설정하여 버퍼 풀이 제외된 공유 메모리 덤프를 만듭니다. DUMPSHMEM 값을 onmode -wm 및 onmode -wf로 동적으로 변경할 수 있습니다.

버퍼 풀이 없는 공유 메모리의 내용을 지정된 파일에 기록하려면 onstat -o 명령에 새로운 nobuffs 옵션을 사용합니다. 파일을 지정하지 않으면 현재 작업 중인 디렉토리의 onstat.out 파일에 내용이 기록됩니다.

옵션 없이 onstat -o를 사용하면 DUMPSHMEM 구성 매개변수 설정에 의해 nobuffs 또는 full 옵션이 제어됩니다.

전체 덤프 파일에서 작동하는 onstat -(대화식)와 같은 여러 onstat 옵션은 버퍼 풀이 제외된 덤프 파일에서도 작동합니다. 버퍼 풀에 한정적으로 액세스하는 일부 옵션(예: -b, -B, -P)의 경우는 작동하지 않습니다.

이 기능은 대부분 IBM Informix Dynamic Server Administrator's Reference에 설명되어 있습니다.

#### 4.5 향상된 서버 시작

oninit 유틸리티에 새로운 -w 옵션을 사용하여 시작 스크립트를 사용자 정의하고 시작을 자동화할 수 있습니다. -w 옵션을 사용하면 서버가 쉘 프롬프트로 돌아가기 전에 성공적으로 초기화되기 전까지 기다립니다. -w 옵션은 시스템이 문제 없이 정상적으로 시작되었는지 확인할 수 있도록 반환 코드를 제공합니다. 구성 가능한 타임아웃 시간 내에 서버가 초기화되지 않으면 유틸리티가 1을 반환하고 메시지를 online.log 파일에 기록합니다. 반환 코드 0은 성공적인 초기화를 나타냅니다.

-w 옵션은 디스크 공간 초기화 및 공유 메모리 초기화 oninit 옵션에서 작동합니다. 예를 들어, 아래 프로세스 목록에서는 데모 서버를 초기화할 때 새로운 설치 관리자 GUI가 이 기능을 사용하여 60초 동안 기다리는 것을 보여줍니다.

informix	15965	1	1	16:33	?	00:00:02	oninit -iyw 60
root	15970	15965	0	16:33	?	00:00:00	oninit -iyw 60
informix	15971	15970	0	16:33	?	00:00:01	oninit -iyw 60
root	15972	15970	0	16:33	?	00:00:00	oninit -iyw 60
root	15973	15970	0	16:33	?	00:00:00	oninit -iyw 60
root	15974	15970	1	16:33	?	00:00:03	oninit -iyw 60
root	15975	15970	0	16:33	?	00:00:00	oninit -iyw 60

High-Availability 환경에서는 oninit -w 명령을 Primary 서버에서만 사용할 수 있으며 Secondary 서버에서는 사용할 수 없습니다.

oninit 유틸리티의 -w 옵션은 IBM Informix Dynamic Server Administrator's Reference에 설명되어 있습니다.

## 5. 새로운 SQL 기능

### 5.1 SPL 루틴을 사용하는 동적 실행

SPL 언어로 작성된 EXECUTE IMMEDIATE 문을 UDR에 포함시킬 수 있으며, 이 경우 유효한 SQL 문의 텍스트인 문자 변수 또는 따옴표 문자열이 IMMEDIATE 키워드 뒤에 나옵니다. 예는 다음과 같습니다.

```
CREATE FUNCTION mk_log_table (base_name VARCHAR(128),
                             period DATETIME YEAR TO DAY,
                             column_definitions LVARCHAR)
    RETURNING VARCHAR(128) AS table_name;

DEFINE create_stmt LVARCHAR;

DEFINE table_name VARCHAR(128);

ON EXCEPTION END EXCEPTION WITH RESUME;

LET table_name = base_name || '_' || year(period) ||
                '_' || month(period) || '_' || day(period);

LET create_stmt = 'CREATE TABLE ' || table_name || ' ' ||
                 column_definitions || ';';

EXECUTE IMMEDIATE create_stmt;

IF (SQLCODE = -310) THEN

    EXECUTE IMMEDIATE 'DROP TABLE ' || table_name;

    IF (SQLCODE = 0) THEN

        EXECUTE IMMEDIATE create_stmt;

    END IF

END IF

IF (SQLCODE != 0) THEN

    LET table_name = NULL;

END IF

RETURN table_name;

END FUNCTION;
```

또한 문의 성공 또는 실패 여부를 테스트하기 위해 새로운 SQLCODE 키워드가 추가되었습니다.

ESQL/C 루틴에서와 달리 SPL 루틴 내에 있는 EXECUTE IMMEDIATE 문의 문 텍스트는 여러 SQL 문을 세미 콜론으로 구분한 목록을 포함할 수 없습니다. 이 방법은 클라이언트-서버 상황에서 네트워크 패킷의 수를 줄이는 데만 유용하기 때문에 SPL 루틴에서는 이 작업을 수행할 필요가 없습니다. 이 기능을 사용하여 동일한 SPL 루틴 내에서 여러 SQL 문을 실행하려면 각 SQL 문의 텍스트에 대해 별도의 EXECUTE IMMEDIATE 문을 제공해야 합니다.

이 기능은 [IDS Guide to SQL: Syntax]에 설명되어 있으며, EXECUTE IMMEDIATE에 의해 지정될 수 있는 SQL 문에 대한 정보를 제공합니다.



### 5.2 IS [NOT] NULL 술어가 있는 SQL 식

SQL 식을 IS NULL 및 IS NOT NULL 술어의 피연산자로 사용할 수 있습니다. 이전에는 TRUE 또는 FALSE 결과를 반환하는 컬럼 이름 피연산자만 IS NULL 및 IS NOT NULL과 사용할 수 있었습니다.

식을 IS NULL 또는 IS NOT NULL과 함께 사용하면 이전에는 NULL이 올바른 숫자 값이 아니기 때문에 계산할 수 없었던 값을 사용할 수 있습니다. 한 함수의 반환 값을 예로 들어보겠습니다.

```
CREATE FUNCTION calculate_value (savings INT,
    rate DECIMAL, years INT, OUT val INT)
RETURNING INT;

    LET val = savings * pow ((1.0 + rate), years);

    RETURN val;

END FUNCTION;

SELECT emp_id, ret_val AS value_in_10_years
FROM savings
WHERE compound_value (savings_in_401k, 0.05, 10,
    ret_val # INT) IS NOT NULL;
```

식을 IS NULL 또는 IS NOT NULL과 함께 사용하는 방법은 IBM Informix Dynamic Server Guide to SQL: Syntax를 참조하십시오.

### 5.3 세션의 외부 디렉티브(External Directives) 제어

SET ENVIRONMENT 문의 새로운 EXTDIRECTIVES 세션 환경 옵션을 사용하여 세션 중에 외부 디렉티브가 사용되는지, 사용되지 않는지, 기본 동작을 수행하는지 여부를 제어할 수 있습니다. EXT\_DIRECTIVES 구성 매개변수 및 클라이언트의 IFX\_EXTDIRECTIVES 환경 변수를 사용하여 기본 디렉티브 동작을 지정합니다.

예를 들어, 세션 내에서 외부 디렉티브를 사용하지 않으려면 다음 명령을 사용하십시오.

```
SET ENVIRONMENT EXTDIRECTIVES 'off';
```

자세한 내용은 IBM Informix Dynamic Server Guide to SQL: Syntax에서 SET ENVIRONMENT 구문 다이어그램과 EXTDIRECTIVES 옵션 절을 참조하십시오.

### 5.4 DRDA(Distributed Relational Database Architecture) 기능 향상

DRDA를 사용하면 서로 다른 플랫폼에 설치된 애플리케이션 및 데이터베이스 시스템 간에 통신이 가능하며, 관계형 데이터를 여러 플랫폼에 분산시킬 수 있습니다. DRDA 세션 정보를 표시하기 위한 지원뿐만 아니라(위에서 설명) 다음과 같이 DRDA 기능이 향상되었습니다.

- ISAM 오류 검색 지원
- 이름 길이가 18바이트가 넘는 데이터베이스에 대한 연결 지원
- 여러 문을 단일 요청으로 일괄 처리
- 식을 IS NULL 또는 IS NOT NULL과 함께 사용하도록 지원
- DRDA 통신에 SSL(Secure Sockets Layer) 프로토콜을 사용하도록 지원
- High-Availability 클러스터에서 Primary 서버 및 공유 디스크 Secondary 서버 간의 DRDA 연결 지원

이 기능은 대부분 IBM Informix Dynamic Server Administrator's Guide에 설명되어 있습니다.

## 6. 보안

### 6.1 Single Sign-on 지원

Single Sign-on에서는 사용자가 한 번만 인증되면 신뢰되는 네트워크와 신뢰되지 않는 네트워크에서 안전하게 인증을 수행할 수 있습니다. 사용자가 클라이언트 컴퓨터에 로그인할 때 올바른 사용자 ID와 암호를 입력하면 다시 로그인하지 않고도 데이터베이스 서버와 다른 SSO 지원 서비스에 액세스할 수 있습니다. 이전에는 사용자가 여러 번 로그인해야 했습니다.

IDS는 GSSCSM(Generic Security Services Communications Support Module)에서 SSO를 지원합니다. GSSCSM은 네트워크 인증 프로토콜인 Kerberos와 함께 작동합니다. IDS와 함께 SSO를 사용하려면 먼저 GSSAPI(Generic Security Services Application Programming Interface)를 지원하는 Kerberos 인증 프로토콜을 설치해야 합니다. SSO를 활성화한 후 중앙집중식 인증 관리 기능을 사용할 수 있습니다.

이 기능은 대부분 IBM Informix Security Guide에 설명되어 있습니다.

### 6.2 SSL(Secure Sockets Layer)로 데이터 통신 암호화

SSL 프로토콜을 사용하도록 IDS를 구성할 수 있습니다. 이 프로토콜은 네트워크의 두 지점 간 TCP/IP 연결에서 데이터를 암호화합니다. SSL 프로토콜은 CSDK 클라이언트용 단순 암호 CSM(Communication Support Module) 및 IDS별 암호화 CSM의 대안으로 널리 사용됩니다.

다음과 같은 연결에 SSL을 사용할 수 있습니다.

- Dynamic Server와의 JDBC 및 SQLJ 연결을 위한 IBM Data Server Driver
- Dynamic Server와의 IBM Informix ESQL/C 연결
- Dynamic Server와의 IBM Informix ODBC Driver 연결
- DB-Access 연결
- Enterprise Replication 연결
- 기본 HDR 서버와 한 대 이상의 Secondary 서버(HDR 보조, SDS 서버 또는 RSS 서버) 간에 HDR(High-Availability Data Replication) 연결
- 여러 개의 데이터베이스 서버에 걸쳐 있는 분산 트랜잭션 연결
- dbexport, dbimport, dbschema 및 dbload 유틸리티 연결
- 클러스터의 서버 간에 Connection Manager 연결
- DRDA 클라이언트

SSL은 암호화 및 서버 인증을 위해 신뢰된 기관에서 발급하는 전자 ID 카드인 디지털 인증서를 사용하여 키를 교환합니다. 디지털 인증서를 발급하는 신뢰된 기관을 인증 기관(CA)이라고 합니다.

IDS 서버는 keystore를 사용하여 디지털 인증서와 이 서버가 연결 중인 다른 모든 서버의 루트 CA 인증서를 저장합니다. 서버 keystore는 \$INFORMIX/ssl 디렉토리에 위치해야 합니다. keystore 파일의 이름은 server\_name.kdb여야 하며, 여기서 server\_name은 DBSERVERNAME 구성 매개변수에 지정된 값입니다. owner & group Informix를 위해 이 파일에 대한 액세스 권한이 600개로 보호되어야 합니다.

클라이언트의 경우 keystore 및 stash 파일(암호화된 암호가 포함된 파일)의 위치가 고정되지 않으며, \$INFORMIXDIR/etc 디렉토리의 conssl.cfg 파일이 이 위치를 지정합니다. 아무 것도 지정하지 않은 경우 기본 값은 \$INFORMIXDIR/etc/client.kdb 및 \$INFORMIXDIR/etc/client.sth입니다.

Dynamic Server 설치와 함께 제공되는 ikeyman GSKit를 사용하여 keystore 암호를 만들고 디지털 인증서를 keystore에 추가합니다. 여기서 iKeyman은 IBM 키 관리 유틸리티입니다.

SSL 기능은 주로 [IDS Security Guide]에 설명되어 있습니다. keystore 내에서 인증서를 관리하기 위해 iKeyman을 사용하는 방법은 IBM Global Security Kit Secure Sockets Layer Introduction and iKeyman User's Guide를 참조하십시오.

## 7. XML 기능

IDS 11에서도 이미 XML 기능을 사용할 수 있었으나, 이를 확장하기 위해 새로운 XML 기능이 추가되었습니다. 이 기능은 XSL Transformation을 추가하고 Basic Text Search 기능을 확장하여 XML 문서를 쿼리하는 기능을 추가합니다. Basic Text Search 기능은 IBM Informix Database Extensions User's Guide에 설명되어 있습니다.

### 7.1 XML 쿼리

XML 문서는 비정형 텍스트 데이터를 저장할 수 있는 모든 데이터베이스 시스템에 쉽게 저장 가능합니다. 이 데이터를 비정형 텍스트로 검색하는 경우 쿼리 기능은 제공되지만 XML 태그, 속성 및 값이 구분되지 않습니다. (11.5 이전의 IDS 버전에서) Basic Text Search 기능을 사용하여 이 데이터에 인덱스를 만든 경우 아주 간단하게 데이터를 쿼리할 수 있습니다.

```
CREATE TABLE albums (id BIGSERIAL, record LVARCHAR);  
  
EXECUTE PROCEDURE IFX_ALLOW_NEWLINE('t');  
  
INSERT INTO albums VALUES (0, '<album>  
<lp>9362-48141-1</lp>  
<artist>The Flaming Lips</artist>  
<title>Yoshimi Battles the Pink Robots</title></album>');  
  
INSERT INTO albums VALUES (0, '<album>  
<lp>PL 75100</lp>  
<artist>The Blues Band</artist>  
<title>Fat City</title></album>');  
  
INSERT INTO albums VALUES (0, '<album>  
<lp>CH-9271</lp>  
<artist>John Lee Hooker</artist>  
<title>The Real Folk Blues</title></album>');  
  
CREATE INDEX albums_bts1 ON albums (record bts_lvvarchar_ops)  
USING bts IN bts_space;
```

제목에 'Blues'가 있는 앨범을 찾으려는 경우, 인덱스를 사용하여 'Blues'가 텍스트에 포함된 모든 레코드를 검색할 수 있지만, 제목에 'Blues'가 나타나지 않았던 레코드는 무시해야 합니다.

```
SELECT record FROM albums WHERE bts_contains (record, 'blues');
record <album>
      <lp>PL 75100</lp>
      <artist>The Blues Band</artist>
      <title>Fat City</title></album>

record <album>
      <lp>CH-9271</lp>
      <artist>John Lee Hooker</artist>
      <title>The Real Folk Blues</title></album>
```

'blues' 값을 포함하는 범주 태그가 여러 레코드에 추가된 경우, 위의 쿼리는 더욱 무의미해지고 원하는 결과를 분류하는 데 상당한 시간 및 노력이 필요합니다.

Basic Text Search 1.10의 새로운 XML 쿼리 기능에서는 XML 태그를 사용하여 인덱스를 만들 수 있습니다. 이렇게 하면 태그가 붙은 데이터의 어떤 부분을 검색할지 쿼리로 지정할 수 있습니다.

```
CREATE INDEX albums_bts2 ON albums (record bts_lvarchar_ops)
USING bts (xmltags='(artist,title)') IN bts_space;

SELECT record FROM albums WHERE bts_contains (record, 'title:blues');

record <album>
      <lp>CH-9271</lp>
      <artist>John Lee Hooker</artist>
      <title>The Real Folk Blues</title></album>

1 row(s) retrieved.
```

XML 쿼리 기능에는 다른 여러 옵션이 있습니다. 예를 들어, 쿼리에서 XML 경로와 XML 네임스페이스를 사용할 수 있도록 인덱스를 만들 수 있습니다.

애플리케이션에서 어떤 태그를 쿼리에 사용할 수 있는지 확인하기 위해 새로운 함수인 `bts_index_fields()`가 추가되었습니다. 인덱스는 'extspace/database/owner/index'로 지정됩니다.

```
EXECUTE FUNCTION bts_index_fields
(' /products/11.50/demo/server/bts_space/bts_1_10/simond/albums_bts2 ');
(expression)  artist
              title
```

참고: 인덱스가 생성된 extspace의 전체 디렉토리 경로를 지정하십시오.

## 7.2 XSL Transformation

XSL(eXtensible Stylesheet Language)은 상위 레벨 XML 요소로 문서의 형식을 나타냅니다. XSLT(XSL Transformation)은 문서 형식을 다른 형식으로 변환하기 위한 언어입니다. 일반적으로 XML 문서를 XML, HTML 또는 일반 텍스트로 변환하지만, 원하는 파일은 모두 다른 파일로 변환할 수 있습니다. XSL, XSLT 및 XPath를 비롯한 여러 W3C 표준에서 이 언어를 사용할 수 있습니다.

XML 문서 검색에 스타일을 적용하기 위해 새로운 SQL 함수인 `xslttransform(document,style)`이 사용되었습니다. 함수 입력은 XML 문서와 스타일시트이고, 출력은 변환된 문서입니다. `varchar`, `clob` 또는 `blob` 유형을 매개변수로 사용할 수 있도록 여러 다른 함수 서명이 생성되었습니다.

```
SELECT xslttransform (record, "<?xml version='1.0' ?>
<xsl:stylesheet xmlns:xsl='http://www.w3.org/1999/XSL/Transform' version='1.0' >
<xsl:template match='artist'><b><xsl:value-of select='.' /></b></xsl:template>
<xsl:template match='title'><i><xsl:value-of select='.' /></i></xsl:template>
</xsl:stylesheet>") FROM albums;

(expression) <?xml version="1.0" encoding="UTF-8"?>
9362-48141-1
<b>The Flaming Lips</b>
<i>Yoshimi Battles the Pink Robots</i>

(expression) <?xml version="1.0" encoding="UTF-8"?>
PL 75100
<b>The Blues Band</b>
<i>Fat City</i>

(expression) <?xml version="1.0" encoding="UTF-8"?>
CH-9271
<b>John Lee Hooker</b>
<i>The Real Folk Blues</i>

3 row(s) retrieved.
```

함수는 "idsxmlvp"라는 외부 VP에서 실행되며, 이것은 ONCONFIG에 포함되거나 `onmode -p`를 사용하여 동적으로 추가됩니다.

이 기능은 XML4C(XML Parser - Xerces) 및 XSLT4C(XSLT 및 XPath processor - Xalan) 패키지를 사용할 수 있는 플랫폼에만 포함됩니다.

## 부록 A – 구성 파일 변경사항

onconfig.std 파일에서 여러 내용이 변경되었습니다. 다음 3개의 표에 주요 차이점이 요약되어 있습니다.

### onconfig.std에 추가된 구성 매개변수

구성 매개변수	값
ADMIN_USER_MODE_WITH_DBSA	None
BTSCANNER	num=1,priority=low,threshold=5000, rangesize=-1,alice=6
BACKUP_FILTER	None
BAR_DEBUG	0
CDR_SUPPRESS_ATSRISWARN	None
DD_HASHMAX	10
DD_HASHSIZE	31
DEF_TABLE_LOCKMODE	page
DS_HASHSIZE	31
DS_POOLSIZE	127
ENCRYPT_CDR	None
ENCRYPT_CIPHERS	None
ENCRYPT_HDR	None
ENCRYPT_MAC	None
ENCRYPT_MACFILE	None
ENCRYPT_SMX	None
ENCRYPT_SWITCH	None
EXT_DIRECTIVES	0
FAILOVER_CALLBACK	None
FASTPOLL	1
HA_ALIAS	None
LOG_INDEX_BUILDS	None
MAX_INCOMPLETE_CONNECTIONS	1024
PC_HASHSIZE	31
PC_POOLSIZE	127
PLCY_HASHSIZE	127
PLCY_POOLSIZE	31
PLOG_OVERFLOW_PATH	UNIX: \$INFORMIXDIR/tmp Windows: None
UPDATABLE_SECONDARY	0
RESTORE_FILTER	None
SBSPACETEMP	None
SDS_ENABLE	None
SDS_PAGING	None
SDS_TEMPDBS	None



SDS_TIMEOUT	20
SECURITY_LOCALCONNECTION	None
SQLTRACE	Commented out: # SQLTRACE level=low,ntraces=1000,size=2,mode=global
SSL_KEYSTORE_LABEL	None
STMT_CACHE	0
STMT_CACHE_HITS	0
STMT_CACHE_NOLIMIT	0
STMT_CACHE_NUMPOOL	1
STMT_CACHE_SIZE	512
TEMPTAB_NOLOG	0
UNSECURE_ONSTAT	None
USRC_HASHSIZE	31
USRC_POOLSIZ	127
VPCLASS	cpu,num=1,noage Commented out: # VPCLASS aio,num=1 # VPCLASS jvp,num=1

### onconfig.std에서 업데이트된 구성 매개변수

매개변수	이전 값	새로운 값
ADMIN_MODE_USERS	1	None
ALARMPROGRAM	UNIX: /usr/informix/etc/alarmprogram.sh Windows: None	UNIX: \$INFORMIXDIR/etc/alarmprogram.sh Windows: \$INFORMIXDIR\etc\alarmprogram.bat
BAR_ACT_LOG	/usr/informix/bar_act.log	UNIX: \$INFORMIXDIR/tmp/bar_act.log Windows: \$INFORMIXDIR\tmp\bar_act.log
BAR_BSALIB_PATH	UNIX: \$INFORMIXDIR/lib/libsad001.so Windows: libbsa.dll	None
BAR_DEBUG_LOG	UNIX: /usr/informix/bar_dbug.log Windows: bar_dbug.log	UNIX: \$INFORMIXDIR/tmp/bar_dbug.log Windows: \$INFORMIXDIR\tmp\bar_dbug.log
BUFFERPOOL	OS with 2K page size: default,buffers=5000,lrus=8, lru_min_dirty=50, lru_max_dirty=60  size=2k buffers=5000,lrus=8, lru_min_dirty=50, lru_max_dirty=60	OS with 2K page size: default,buffers=10000,lrus=8, lru_min_dirty=50.00, lru_max_dirty=60.50  size=2k, buffers=50000,lrus=8, lru_min_dirty=50, lru_max_dirty= 60

onconfig.std에서 업데이트된 구성 매개변수

매개변수	이전 값	새로운 값
BUFFERPOOL	OS with 4K page size: default,buffers=1000,lrus=8, lru_min_dirty=50, lru_max_dirty=60  size=4k,buffers=1000,lrus=8, lru_min_dirty=50, lru_max_dirty=60	OS with 4K page size: default,buffers=10000,lrus=8, lru_min_dirty=50.00, lru_max_dirty=60.50  size=4k,buffers=10000,lrus=8, lru_min_dirty=50, lru_max_dirty=60
CLEANERS	1	8
CONSOLE	UNIX: /dev/console Windows: console.log	UNIX: \$INFORMIXDIR/tmp/online.con Windows: online.con
DB_LIBRARY_PATH	commented out: # DB_LIBRARY_PATH\$INFORMIXDIR/ extend	commented out: # DB_LIBRARY_PATH
DRLOSTFOUND	UNIX: /usr/etc/dr.lostfound Windows: \tmp	UNIX: \$INFORMIXDIR/etc/dr.lostfound Windows: \$INFORMIXDIR\tmp
DUMPDIR	UNIX: /usr/informix/tmp Windows: \$INFORMIXDIR\tmp	UNIX: \$INFORMIXDIR/tmp Windows: \$INFORMIXDIR\tmp
EXPLAIN_STAT	1	0
LISTEN_TIMEOUT	10	60
LOCKS	2000	20000
LOGBUFF	32	64
LOGSIZE	2000	10000
LTAPEDEV	UNIX: /dev/tapedev Windows: \\TAPE1	UNIX: /dev/tapedev (same) Windows: NUL
MIRRORPATH	None	\$INFORMIXDIR/tmp/demo_on.root_ mirror Windows: None
MSGPATH	UNIX: /usr/informix/online.log Windows: online.log	UNIX: \$INFORMIXDIR/tmp/online.log Windows: online.log
NETTYPE	UNIX: None Windows: onsoctcp,drssoctcp,1,NET	UNIX: ipcshm,1,50,CPU Windows: None
PHYSBUFF	32	128
PHYSFILE	2000	50000
RA_PAGES	None	64
RA_THRESHOLD	None	16
ROOTOFFSET	0	UNIX: 4 Windows: 0
ROOTPATH	UNIX: /dev/online_root Windows: None	UNIX: \$INFORMIXDIR/tmp/demo_on.rootdbs Windows: None
ROOTSIZE	30000	200000
SHMVIRT_ALLOCSEG	0	0,3
SHMVIRT_SIZE	8192	32656

SYSALARMPROGRAM	UNIX: /usr/informix/etc/evidence.sh Windows: \$INFORMIXIDR\etc\evidence.bat	UNIX: \$INFORMIXDIR/etc/evidence.sh Windows: Commented out # SYSALARMPROGRAM\$INFORMIXDIR \etc\evidence.bat
TAPEBLK	32	UNIX: 32 (same) Windows: 16
TAPESIZE	10240	0

### onconfig.std에서 제거된 구성 매개변수

매개변수
AFF_NPROCS
AFF_SPROC
NOAGE
NUMCPUVPS
PHYSDBS

#### 사용된 참조 문서

- IBM Informix Dynamic Server Guide to SQL: Syntax, Version 11.50
- IBM Informix Dynamic Server Administrator's Guide, Version 11.50
- IBM Informix Dynamic Server Administrator's Reference, Version 11.50
- Installation Guide — Windows, Version 11.50
- Installation Guide — UNIX, Linux & Mac, Version 11.50
- IBM Informix Dynamic Server Guide to SQL: Reference, Version 11.50
- OpenAdmin Tool 릴리스 노트
- IBM Informix Dynamic Server Security Guide, Version 11.50
- IBM Global Security Kit Secure Sockets Layer Introduction and iKeyman User's Guide,
- IBM Informix Database Extensions User Guide, Version 11.50



© Copyright IBM Corporation 2008

(135-270) 서울시 강남구 도곡동 467-12  
군인공제회관빌딩

한국아이비엠주식회사  
고객만족센터

TEL: (02)3781-7114  
[www.ibm.com/kr](http://www.ibm.com/kr)

2008년 7월

Printed in Korea  
All Rights Reserved