IBM® DB2® Universal Database

릴리스 정보

버전 7 -- *FixPak 7*

IBM® DB2® Universal Database

릴리스 정보

버전 7 -- FixPak 7

목차

DB2 Universal Database 버전 7 FixPak 7 소개

주: 글꼴을 모노스페이스로 설정하면, 이 릴리스 정보를 보다 잘 볼 수 있습니다.

DB2 Universal Database 및 DB2 Connect 지원 사이트는 규칙적으로 갱신됩니다. 최신 정보가 있는지 http://www.ibm.com/software/data/db2/udb/winos2unix/support를 확인하십시오.

이 책에는 인쇄된 DB2 매뉴얼에 포함되어 있지 않은 다음 제품에 대한 정보가 들어 있습니다.

```
IBM DB2 Universal Database Personal Edition, 버전 7.2
IBM DB2 Universal Database Workgroup Edition, 버전 7.2
IBM DB2 Universal Database Enterprise Edition, 버전 7.2
IBM DB2 Data Links Manager, 버전 7.2
IBM DB2 Universal Database Enterprise - Extended Edition, 버전 7.2
IBM DB2 Query Patroller, 버전 7.2
IBM DB2 Personal Developer's Edition, 버전 7.2
IBM DB2 Universal Developer's Edition, 버전 7.2
IBM DB2 Data Warehouse Manager, 버전 7.2
IBM DB2 Relational Connect, 버전 7.2
IBM DB2 Connect Personal Edition, 버전 7.2
IBM DB2 Connect Enterprise Edition, 버전 7.2
```

다음 제품에는 별도의 릴리스 정보 파일 READCON.TXT가 제공됩니다.

```
IBM DB2 Connect Personal Edition, 버전 7.2
IBM DB2 Connect Enterprise Edition, 버전 7.2
```

DB2 Life Sciences Data Connect 제품에 대한 문서는 IBM 소프트웨어 사이트 http://www.ibm.com/software/data/db2/lifesciencesdataconnect/library.html에서 다운로드할 수 있습니다. 이 제품에 대한 정보는 http://www.ibm.com/software/data/db2/lifesciencesdataconnect에서 온라인으로 사용할 수 있습니다.

다음 서적은 FixPak 4용으로 갱신되었으며, http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 최신 PDF 파일을 다운로드할 수 있습니다.

관리 안내서
응용프로그램 빌드 안내서
응용프로그램 개발 안내서
Command Reference
데이터 복구와 고가용성 안내 및 참조서
Data Warehouse Center 관리 안내서
메시지 참조서
SQL 참조서
DB2 Warehouse Manager 설치 안내서

또한 갱신된 참조에 이러한 참고 내용을 추가해 놓았습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

새로운 기능 책에는 버전 7.2의 주요 DB2 향상 내용 중 일부에 대한 개요가 들어 있습니다. 7.2 버전의 새로운 기능 책이 없으면 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 내용을 열람하거나 다운로드할 수 있습니다.

DB2 제품군에 대한 최신 정보를 보려면, "DB2 잡지"를 무료로 구독하십시오. 잡지의 온라인 판은 http://www.db2mag.com에서 사용할 수 있으며 구독 요청을 위한 지시사항도 이 사이트에 게시되어 있습니다.

주: 이 릴리스 정보에서 Windows NT에 대한 참조는 Windows 2000에도 해당됩니다. 다르게 지정하는 경우를 제외하고는 22 페이지의 『Windows ME, Windows XP 및 Windows 2000 Datacenter Edition 플랫폼 지원』에서 나열된 제품 컨텍스트에 있는 Windows XP에 대한 참조도 포함됩니다.

주: 페이지 왼쪽의 갱신 막대(|)는 첫 릴리스 정보가 발행된 이후에 추가되거나 수정된 행을 나타냅니다.

제 1 부 이 책을 읽기 전에

FixPak 7 주요사항

다음과 같은 새로운 정보가 이 FixPak에 추가되었습니다. 이 목록은 완벽하지 않지만, 주요 문서 변경사항을 강조표시합니다. 작업 환경에 관련된 해당 릴리스 정보의 섹션을 살펴서 적절한 모든 갱신사항을 파악할 수 있습니다.

- 9 페이지의 『Veritas AIX 볼륨 관리 프로그램 지원』
- 20 페이지의 『AIX CICS 사용자를 위한 추가 설치 단계』
- 40 페이지의 『UNIX의 Merant 드라이버 관리자 및 DB2 UDB 버전 7 ODBC 드라이버』
- 78 페이지의 『7단계: 서버 작성』
- 110 페이지의 『익스포트 유틸리티가 식별 속성을 제대로 처리하려면 FixPak 7 이상이 필요함』
- 110 페이지의 『국가/지역 코드 및 코드 페이지 지원』
- 132 페이지의 『레지스트리 변수 DB2BPVARS의 새 매개변수』
- 155 페이지의 『ARCHIVE LOG의 추가 컨텍스트에 대한 사용 설명』
- 175 페이지의 『새 아카이브 로깅 동작』
- 176 페이지의 『데이터베이스 복구를 위한 일시중단된 입출력 사용법』
- 182 페이지의 『증분 백업 및 복구 - 추가 정보』
- 185 페이지의 『이제 익스포트 유틸리티에서 확장 식별 값이 완전히 지원됨』
- 185 페이지의 『익스포트, 임포트 및 로드에 의한 LOB 파일 조절 변경』
- 216 페이지의 『제 15 장 로그된 정보』
- 247 페이지의 『경로 재지정 리스토어 선택으로 데이터베이스 리스토어 커미트』
- 261 페이지의 『z/OS 또는 OS/390용 Java 스토어드 프로시저 지원』
- 262 페이지의 『z/OS 또는 OS/390용 SQL 스토어드 프로시저 지원』
- 263 페이지의 『z/OS 또는 OS/390 문서의 Stored Procedure Builder 참조서 갱신』
- 263 페이지의 『결과 세트 등록 정보 설정 지원』

- 264 페이지의『Windows NT의 DB2 데이터베이스에서 프로시저 삭제』
- 359 페이지의『S/390 및 zSeries에서 Linux 지원』
- 360 페이지의『Linux Rexx 지원』
- 361 페이지의『"Java로 작성된 스토어드 프로시저 디버깅"에 대한 수정사항』
- 380 페이지의『SQLGetInfo - 일반 정보 가져오기』
- 394 페이지의『CREATE TABLESPACE 컨테이너 절의 정정사항 및 컨테이너 문자열 정보』

제품 정보

Solaris 운영 환경용 DB2 버전 7에서 지원되는 CPU

UltraSparc 이전의 CPU 버전은 지원되지 않습니다.

Red Flag Linux에서 중국어 로케일 수정

중국어 Red Flag Linux 서버 버전 1.1을 사용하는 경우, 중국어 로케일 수정사항을 받으려면 Red Flag에 접속하십시오. 버전 1.1용 중국어 로케일 수정이 없으면, DB2는 중국어의 코드 페이지가 1386임을 인식하지 못합니다.

일본어 및 중국어 Linux 환경에서 Linux용 DB2에 대한 추가 로케일 설정

추가 로케일 설정은 일본어 또는 중국어 Linux 시스템에서 제어 센터와 같은 Java GUI 도구를 사용하려고 할 때 필요합니다. 일본어 또는 중국어 문자는 이 설정 없이는 올바르게 표시될 수 없습니다. 다음 설정을 사용자 프로파일에 포함시키거나, 제어 센터를 호출하기 전에 항상 명령행에서 이를 수행하십시오.

일본어 시스템의 경우

```
export LC_ALL=ja_JP
```

중국어 시스템의 경우

```
export LC_ALL=zh_CN
```

PTX에서의 일본어 제한사항

PTX 시스템에서 일본어로 DB2 UDB를 실행할 경우, DB2가 사용하는 일부 프로세스가 올바른 로케일 정보를 상속하지 못할 수 있습니다. 이를 방지하려면 DB2CODEPAGE 및 DB2COUNTRY 레지스트리 변수를 로케일에 맞게 수동으로 설정하십시오.

Microsoft Internet Explorer의 제어 센터 문제점

Internet Explorer(IE) 보안 옵션 설정으로 인한 문제점이 있습니다. 제어 센터에서는 서명하지 않은 jar를 사용하므로, 시스템 정보로 액세스하는 경우에는 보안 관리 프로그램이 사용할 수 없습니다.

이 문제점을 제거하려면, 다음과 같이 IE 보안 옵션을 다시 구성하십시오.

1. **보기** 메뉴(IE4) 또는 **도구** 메뉴(IE5)에서 **인터넷 옵션**을 선택하십시오.
2. 보안 페이지에서 **신뢰할 수 있는 사이트**를 선택하십시오.
3. **사이트**를 누르십시오.
4. 제어 센터 웹 서버를 신뢰할 수 있는 사이트 목록에 추가하십시오. 제어 센터 웹 서버가 같은 도메인에 있으면 웹 서버 이름(도메인 이름없이)만을 추가하는 것이 유용할 수 있습니다. 예를 들어, 다음과 같습니다.

   ```
   http://ccWebServer.ccWebServerDomain
   http://ccWebServer
   ```

   주: URL을 입력할 때에는 https:// 접두어를 사용해야 합니다. 사이트에 추가하기 위해서는 **이 영역에 있는 모든 사이트에 대해 서버 확인(https:) 요청** 옵션을 선택 취소합니다.
5. **확인**을 누르십시오.
6. **설정값...**(IE4) 또는 **사용자 정의 수준...**(IE5)을 누르십시오.
7. 화면을 아래로 이동하여 **Java --► Java 허용**으로 가서 **사용자 정의**를 선택하십시오.
8. **Java 사용자 정의 설정...**을 누르십시오.
9. 권한 편집 페이지를 선택하십시오.
10. 아래로 화면 이동하여, 서명하지 않은 컨텐트 --► 서명하지 않은 컨텐트 실행 --► 서명하지 않은 추가 권한 --► 시스템 정보로 가서 **사용**을 선택하십시오.
11. 열린 각 창마다 **확인**을 누르십시오.

제어 센터 기능 손실

DB2 버전 7.2에서, FixPak 6 이전의 버전 6 제어 센터 클라이언트 및 FixPak 2 이전의 버전 7 클라이언트를 DB2 버전 7.2와 함께 사용할 경우 거의 모든 기능을 잃게 됩니다. 이를 수정하려면 버전 6 클라이언트는 FixPak 6 이상으로 업그레이드해야 하고, 버전 7 클라이언트는 FixPak 2 이상으로 업그레이드해야 합니다.

DB2 서버에 FixPak 2를 적용하면 이전 레벨 제어 센터 클라이언트에 대해 소개된 문제점이 없어야 합니다.

DB2 UDB에 포함되어 있지 않은 Netscape CD

이제 DB2 UDB에는 Netscape CD가 더 이상 포함되어 있지 않습니다. Netscape 제품은 http://www.netscape.com에서 사용 가능합니다.

XML Readme 파일 오류

DB2 XML Extender용 버전 7.1 README.TXT 파일의 "고려사항" 부분을 보면 다음과 같은 내용이 있습니다.

> 3. DB2 UDB 기본 버전은 DB2 UDB 버전 7.1입니다. AIX 및 Solaris 시스템에서 DB2 UDB 버전 6.1을 사용하려면, DB2 UDB V6.1 인스턴스 및 DB2 UDB V6.1 라이브러리가 함께 실행되고 있는지 확인해야 합니다.

이는 잘못된 정보입니다. **DB2 XML Extender는 DB2 버전 7.1 및 7.2에만 지원됩니다.**

readme.aix, readme.nt 및 readme.sun 파일에 나타난 소프트웨어 요구사항은 다음과 같습니다.

- FP1_U465423 이상인 DB2 UDB 6.1(AIX)
- FixPak 3이 설치된 DB2 Universal Database 버전 6.1 이상(NT)
- FixPak FP1_U465424 이상인 DB2 UDB 버전 6.1(Sun)

이는 잘못된 정보입니다. **DB2 XML Extender는 DB2 버전 7.1이나 7.2에서 사용 가능합니다.**

DB2 버전 7.2의 새로운 비즈니스 인텔리전스 향상

버전 7.2 새로운 기능 책 및 기타 다른 문서에서는 버전 7.2에 추가된 새로운 비즈니스 인텔리전스를 참조합니다. 이들 개선된 기능은 나중에 사용 가능합니다.

IBM DB2 OLAP Server에서 발생하는 FixPak 2A 이상의 문제점

UNIX에서 IBM DB2 OLAP Server를 사용할 경우 DB2 Universal Database 버전 7의 FixPak 2A를 설치하면 DB2에 OLAP Server 문제점이 발생할 수 있습니다. FixPak 2A 이상은 UNICODE를 지원하는 새 ODBC 드라이버를 설치하지만, DB2 OLAP Server가 이 새 드라이버를 지원하지 않습니다. DB2 OLAP Server에 대한 일시적인 해결책은 비UNICODE ODBC 드라이버를 가리키도록 ODBC 파일을 전환하는 것입니다.

비UNICODE 드라이버의 이름을 바꿉니다. 기존 이름에 ″_36″을 추가합니다. 예를 들어, Solaris 운영 환경에서 드라이버 `libdb2.so`는 `libdb2_36.so`로 이름이 바뀌었습니다. ODBC 드라이버 변경에 대한 자세한 정보는 *OLAP 설치 및 사용자 안내서*의 제 4 장 ″AIX, Solaris 운영 환경 및 HP-UX에서 설치″에서 ″SQL 인터페이스용 ODBC 로드 및 구성″을 참조하십시오.

WebSphere 3.5.5 사용시 세그먼트화 위반

Linux390에서 DB2 V7.2 FixPak 4 이상을 설치한 WebSphere 3.5.5 사용자 프로파일 샘플을 실행 중인 경우, SIGSEGV 11 (*) 세그먼트화 위반이 일어날 수 있습니다. 이 문제점은 JDK에서의 결함과 관련된 것으로, JDK 1.2.2 및 JDK 1.3 모두에서 발생합니다.

이 문제점은 다른 JDBC 응용프로그램에도 영향을 미칠 수 있습니다.

JDK 1.2.2의 11월 서비스 릴리스가 이 문제점을 수정합니다. JDK 1.3은 1월 서비스 릴리스에서 수정됩니다.

이 문제점의 일시적인 해결책은 다음 명령으로 JIT를 종료하는 것입니다.

```
export JAVA_COMPILER=NONE
```

Veritas AIX 볼륨 관리 프로그램 지원

DB2 UDB Enterprise Edition, FixPak 7 이상은 AIX 5.1 ML 2 이상에서 Veritas AIX 볼륨 관리 프로그램 버전 3.2와 함께 사용할 수 있습니다. Veritas AIX 볼륨 관리 프로그램을 Enterprise Extended Edition을 포함한 DB2 UDB의 다른 버전과 함께 사용하는 것은 현재 지원되지 않습니다.

AIX V4에서 Java 응용프로그램에 필요한 수정사항

AIX 4.3.3에서 실행되는 Java 응용프로그램은 커널 파일 세트 bos.mp 또는 bos.up이 4.3.3.77인 경우 예기치 않게 종료될 수 있습니다.

커널 수정 레벨을 판별하려면 **lslpp -l bos.ip bos.mp** 명령을 실행하십시오.

AIX 4.3.3에서 실행하는 모든 Java 고객은 4.3.3.78로 업그레이드하는 것이 좋습니다. 커널을 제안된 레벨로 갱신하는 수정사항이 제공됩니다. APAR(Authorized Problem Analysis Report) IY25282용 수정사항을 구해야 합니다.

IY25282용으로 수정사항을 설치하면 Java 종료 조건이 정정됩니다. 2002년 1/4 분기에는 APAR 번호 IY26149를 사용하여 추가 수정사항이 릴리스됩니다.

AIX 5.1C는 동일한 수정사항이 포함된 IY25377과 함께 제공됩니다.

추가 정보 및 지침은 AIX 지원을 참조하십시오.

온라인 문서(HTML, PDF 및 검색) 정보

Windows 2000 운영 체제에서 지원되는 웹 브라우저

Windows 2000에서는 Microsoft Internet Explorer를 사용하십시오.

Netscape를 사용할 경우에는 다음 사항에 유의하십시오.

- Netscape를 사용하여 Windows 2000에서 DB2 온라인 정보 검색을 완료하는 데는 시간이 아주 오래 걸릴 수도 있습니다. Netscape는 사용 가능한 모든 CPU 자원을 사용하며 무한정으로 수행될 것으로 나타납니다. 검색 요청 후 검색 결과가 표시되는 동안, 다른 창을 눌러 활성화 창을 변경하는 것이 좋습니다. 그러면, 적당한 시간 내에 검색 결과가 리턴됩니다.
- 제어 센터에서 Netscape 브라우저 창에 도움말이 표시되도록 요청한 후, 브라우저 창을 열린 채로 두고 제어 센터의 다른 부분에서 나중에 도움말을 요청하면, 브라우저는 이전 도움말을 표시한 채로 변경되지 않을 수도 있습니다. 브라우저 창을 닫고 도움말을 다시 요청하면, 올바른 도움말이 제공됩니다. 12 페이지의 『Netscape 시작시 발생하는 오류 메시지』의 단계를 수행해도 이 문제점을 수정할 수 있습니다. 제어 센터에 대한 도움말을 요청하기 전에 브라우저 창을 닫아서 문제점을 해결할 수도 있습니다.
- 제어 센터 도움말이나, 정보 센터로부터 주제를 요청할 경우, 오류 메시지가 표시될 수도 있습니다. 이를 수정하려면, 12 페이지의 『Netscape 시작시 발생하는 오류 메시지』의 단계를 수행하십시오.

Solaris 운영 환경에서 DB2 온라인 정보 검색

Solaris 운영 환경에서 DB2 온라인 정보를 검색하는 데 문제점이 있으면, `/etc/system`에서 시스템의 커널 매개변수를 확인하십시오. 다음은 DB2의 검색 시스템인 NetQuestion에서 필요한 최소 커널 매개변수입니다.

```
semsys:seminfo_semmni 256
semsys:seminfo_semmap 258
semsys:seminfo_semmns 512
semsys:seminfo_semmnu 512
```



```
semsys:seminfo_semmsl 50
shmsys:shminfo_shmmax 6291456
shmsys:shminfo_shmseg 16
shmsys:shminfo_shmmni 300
```

커널 매개변수를 설정하려면, /etc/system의 끝에 다음 행을 추가하십시오.

```
set <semaphore_name> = value
```

새로운 값이나 변경된 값이 적용되려면 시스템을 재부트해야 합니다.

TCP/IP를 사용하도록 OS/2용 NetQuestion 전환

OS/2 시스템에서 TCP/IP를 사용하기 위해 NetQuestion의 전환에 관한 지시사항이 완전하지 않습니다. 그 지시사항에 언급된 *.cfg 파일의 위치가 NetQuestion 설치 디렉토리의 데이터 서브디렉토리입니다. 다음 명령 중 하나를 입력하여 NetQuestion 설치 디렉토리를 판별할 수 있습니다.

```
echo %IMNINSTSRV%   //for SBCS installations
echo %IMQINSTSRV%   //for DBCS installations
```

Netscape 시작시 발생하는 오류 메시지

Netscape를 시작하려고 할 때 다음과 같은 오류 메시지가 표시될 경우,

```
<file path> 파일(또는 해당되는 구성요소 중 하나)을 찾을 수 없습니다.
  경로와 파일 이름이 올바르고, 필요한 모든 라이브러리가
  사용 가능한지 확인하십시오.


"D:\Program Files\SQLLIB\CC\..\doc\html\db2help\XXXXX.htm"을 열 수 없습니다.
```

Windows NT, 95 또는 98에서 이 문제점을 정정하려면 다음 단계를 수행하십시오. (Windows 2000에서의 수행 사항에 대해서는 아래를 참조하십시오.)

1. 시작 메뉴에서 프로그램 --> Windows 탐색기를 선택하십시오. Windows 탐색기가 열립니다.
2. Windows 탐색기에서 보기 --> 옵션을 선택하십시오. 옵션 노트북이 열립니다.
3. **파일 형식** 탭을 누르십시오. 파일 형식 페이지가 열립니다.
4. **등록된 파일 형식** 필드에서 Netscape Hypertext Document를 강조표시하고 **편집**을 누르십시오. 파일 형식 편집 창이 열립니다.

5. **작업** 필드에서 "Open"을 강조표시하십시오.
6. **편집** 버튼을 누르십시오. 명령 편집 창이 열립니다.
7. **DDE 사용** 선택란을 선택 취소하십시오.
8. **명령을 수행할 응용프로그램** 필드에서 문자열의 맨 끝에 "%1"이 표시되는지 확인하십시오. (물음표와 첫 번째 물음표 앞의 공백도 확인하십시오.)

Windows 2000에서 해당 오류 메시지가 표시될 경우에는 다음 단계를 수행해야 합니다.

1. **시작** 메뉴에서 **Windows 탐색기**를 선택하십시오. Windows 탐색기가 열립니다.
2. Windows 탐색기에서 **도구 --> 폴더 옵션**을 선택하십시오. 폴더 옵션 노트북이 열립니다.
3. **파일 형식** 탭을 누르십시오.
4. 파일 형식 페이지의 **등록된 파일 형식** 필드에서 HTM Netscape Hypertext Document를 강조표시한 후 **고급**을 누르십시오. 파일 형식 편집 창이 열립니다.
5. **명령** 필드에서 "open"을 강조표시하십시오.
6. **편집** 버튼을 누르십시오. 명령 편집 창이 열립니다.
7. **DDE 사용** 선택란을 선택 취소하십시오.
8. **명령을 수행할 응용프로그램** 필드에서 문자열의 맨 끝에 "%1"이 표시되는지 확인하십시오. (물음표와 첫 번째 물음표 앞의 공백도 확인하십시오.)
9. **확인**을 누르십시오.
10. HTML Netscape Hypertext Document와 SHTML Netscape Hypertext Document 파일 형식에 대해 4 - 8단계를 반복하십시오.

Unix 기반 시스템에서 Adobe Acrobat Reader에 대한 구성 요구사항

Acrobat Reader는 Unix 기반 플랫폼에서 영어로만 제공되며 영어 이외의 언어 로케일로 PDF 파일을 열려고 시도하면 오류가 리턴될 수 있습니다. 이들 오류는 PDF 파일의 글꼴 액세스 또는 추출 문제점을 제시하지만 실제로는 영문 Acrobat Reader가 UNIX 영어 외의 언어 로케일 내에서는 제대로 기능을 수행할 수 없음으로 인해 발생합니다.

이런 경우 PDF 파일을 보려면, 영문 Acrobat Reader를 시작하기 전에 다음의 단계 중 하나를 수행하여 영어 로케일로 전환하십시오.

- 시작 스크립트 파일에서 #!/bin/sh 명령문 이후에 다음의 행을 추가하여 Acrobat Reader의 시작 스크립트를 편집하십시오.

  ```
  LANG=C;export LANG
  ```

  이렇게 하면 Netscape Navigator 등의 다른 응용프로그램이나 응용프로그램 도움말 메뉴에서 Acrobat Reader를 시작할 때 올바르게 작동할 것입니다.
- 명령 프롬프트에 LANG=C를 입력하여 Acrobat Reader의 응용프로그램 환경을 영어로 설정하십시오.

자세한 내용은 Adobe Systems(http://www.Adobe.com)에 문의하십시오.

하나의 PDF 파일에 제공되는 SQL 참조서

각 책의 "DB2 라이브러리 사용" 부록은 SQL 참조서가 두 개의 다른 볼륨으로 된 PDF 형식으로 사용할 수 있다고 설명하고 있는데 이는 잘못된 정보입니다.

인쇄된 책이 두 볼륨으로 나타나고 두 개의 해당 양식 번호가 올바르더라도 하나의 PDF 파일만이 있으며 여기에는 두 볼륨이 모두 들어 있습니다. PDF 파일 이름은 db2s0x70입니다.

제 2 부 설치 및 구성

일반 설치, 이주 및 구성 정보

지원되는 모든 DB2 클라이언트의 설치 패키지 다운로드

버전 7 이전의 클라이언트를 모두 포함하는 지원되는 모든 DB2 클라이언트의 설치 패키지를 다운로드하려면, 다음의 IBM DB2 FixPaks 및 클라이언트 웹 사이트에 연결하십시오.

http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/download.d2w/report.

추가 필수 Solaris 운영 환경 패치 레벨

Solaris 운영 환경 버전 2.6용 DB2 Universal Database 버전 7에는 *UNIX용 DB2 빠른 시작* 매뉴얼에 나열된 패치 이외에 패치 106285-02 이상이 필요합니다.

AIX에 DB2 Enterprise-Extended Edition 설치

UNIX용 DB2 Enterprise-Extended Edition 빠른 시작 매뉴얼의 설치 수행 섹션의 4단계에서는 다음 명령을 입력하여 CD-ROM 파일 시스템을 할당해야 한다고 기술하고 있습니다.

```
crfs -v cdrfs -p ro -d cd0
```

또한 이 명령이 올바르게 완료되려면, --m 명령을 사용하여 마운트 포인트를 지정해야 합니다.

```
crfs -v cdrfs -p ro -d cd0 -m /cdrom
```

또한 설치 수행 섹션에서 누락된 단계가 있습니다. 5단계 'CD-ROM 파일 시스템을 제어 워크스테이션에 마운트'를 수행한 다음에 파티션된 데이터베이스 시스템에 참여할 각 노드는 CD-ROM 파일 시스템을 리모트로 마운트해야 합니다. `/cdrom`이 참여 노드에 없을 경우, 다음 명령을 사용하여 제어 워크스테이션에서 /cdrom 파일 시스템을 익스포트하고 리모트로 마운트하십시오.

```
exportfs -i -o ro /cdrom
dsh mkdir /cdrom
dsh mount cws_hostname: /cdrom /cdrom
```

여기서, *cws_hostname*는 제어 워크스테이션의 호스트 이름입니다.

AIX CICS 사용자를 위한 추가 설치 단계

AIX CICS 시스템에서 DB2 UDB 또는 DB2 UDB FixPak을 설치하는 경우, 설치 후에 다음 추가 단계를 수행해야 합니다. 이러한 단계는 *CICS/6000 관리 안내서*의 『DB2에 대한 구성 단계』 섹션에 자세히 설명되어 있습니다.

1. libdb2.a 라이브러리에서 AIX용 DB2 UDB 공유 오브젝트를 작성하십시오.
2. DB2 switchload 파일을 빌드하여 데이터베이스의 XA 정의에서 지정한 디렉토리에 위치시키십시오.
3. 사용자 환경에서 COBOL을 사용하는 경우에는 cocsmkcobol 도구를 다시 실행하십시오.

Netscape LDAP 디렉토리 지원

DB2는 데이터베이스 및 노드 디렉토리의 중앙 관리 및 통합을 위해 LDAP 디렉토리 사용을 지원합니다. 이전 DB2 릴리스에서는 Microsoft Active Directory 및 IBM SecureWay Directory만 지원되었습니다. 이제 DB2는 LDAP 서버 Netscape Directory Server v4.12 이상 및 iPlanet™ Directory Server 5.0 이상도 지원합니다.

Netscape LDAP 스키마 확장

다음 지시사항은 Netscape Directory Server 4.1에 관련된 것입니다.

Netscape 디렉토리 서버에서는 응용프로그램을 이용해 속성 및 오브젝트 클래스 정의를 slapd.user_oc.conf 및 slapd.user_at.conf 두 파일에 추가하여 스키마를 확장할 수 있습니다. 이들 두 파일은 <Netscape_install path>\slapd-<machine_name>\config 디렉토리에 위치합니다.

DB2 속성을 다음과 같이 slapd.user_at.conf에 추가해야 합니다.

주: 이 컨텍스트에서 bin, cis, ces 및 dn은 각각 2진, 대소문자 구분 문자열, 대소문자 비구분 문자열 및 식별 이름을 나타냅니다.

```
############################################################################
#
# IBM DB2 Universal Database V7.2
# Attribute Definitions
#
############################################################################

attribute binProperty                    1.3.18.0.2.4.305     bin
attribute binPropertyType                1.3.18.0.2.4.306     cis
attribute cesProperty                    1.3.18.0.2.4.307     ces
attribute cesPropertyType                1.3.18.0.2.4.308     cis
attribute cisProperty                    1.3.18.0.2.4.309     cis
attribute cisPropertyType                1.3.18.0.2.4.310     cis
attribute propertyType                   1.3.18.0.2.4.320     cis
attribute systemName                     1.3.18.0.2.4.329     cis
attribute db2nodeName                    1.3.18.0.2.4.419     cis
attribute db2nodeAlias                   1.3.18.0.2.4.420     cis
attribute db2instanceName                1.3.18.0.2.4.428     cis
attribute db2Type                        1.3.18.0.2.4.418     cis
attribute db2databaseName                1.3.18.0.2.4.421     cis
attribute db2databaseAlias               1.3.18.0.2.4.422     cis
attribute db2nodePtr                     1.3.18.0.2.4.423     dn
attribute db2gwPtr                       1.3.18.0.2.4.424     dn
attribute db2additionalParameters        1.3.18.0.2.4.426     cis
attribute db2ARLibrary                   1.3.18.0.2.4.427     cis
attribute db2authenticationLocation      1.3.18.0.2.4.425     cis
attribute db2databaseRelease             1.3.18.0.2.4.429     cis
attribute DCEPrincipalName               1.3.18.0.2.4.443     cis
```

DB2 오브젝트 클래스를 다음과 같이 slapd.user_oc.conf 파일에 추가해야 합니다.

```
############################################################################
#
# IBM DB2 Universal Database V7.2
# Object Class Definitions
#
############################################################################

objectclass eProperty
        oid 1.3.18.0.2.6.90
        requires
                objectClass
        allows
                cn,
                propertyType,
                binProperty,
                binPropertyType,
                cesProperty,
```

```
                cesPropertyType,
                cisProperty,
                cisPropertyType

objectclass eApplicationSystem
        oid 1.3.18.0.2.6.8
        requires
                objectClass,
                systemName


objectclass DB2Node
        oid 1.3.18.0.2.6.116
        requires
                objectClass,
                db2nodeName
        allows
                db2nodeAlias,
                host,
                db2instanceName,
                db2Type,
                description,
                protocolInformation

objectclass DB2Database
        oid 1.3.18.0.2.6.117
        requires
                objectClass,
                db2databaseName,
                db2nodePtr
        allows
                db2databaseAlias,
                description,
                db2gwPtr,
                db2additionalParameters,
                db2authenticationLocation,
                DCEPrincipalName,
                db2databaseRelease,
                db2ARLibrary
```

DB2 스키마 정의를 추가한 다음에는 모든 변경 내용이 활성화되도록 Directory Server를 레지스터에 기록해야 합니다.

Windows ME, Windows XP 및 Windows 2000 Datacenter Edition 플랫폼 지원

DB2는 이제 Microsoft Windows ME, Windows XP 및 Windows 2000 Datacenter Edition 플랫폼을 지원합니다. 다음은 플랫폼별 추가 정보입니다.

Windows XP

다음 제품 및 버전은 FixPak 4 이상과 함께 설치될 때 32비트 Windows XP를 지원합니다.

- IBM DB2 UDB Personal Edition 버전 7.2
- IBM DB2 Personal Developer's Edition 버전 7.2
- IBM DB2 Universal Developer's Edition 버전 7.2
- IBM DB2 Connect Personal Edition 버전 7.2
- IBM DB2 Connect Enterprise Edition 버전 7.2
- IBM DB2 UDB Workgroup Edition 버전 7.2
- IBM DB2 UDB Enterprise Edition 버전 7.2
- IBM DB2 Run-Time Client 버전 7.2
- IBM DB2 Administration Client 버전 7.2
- IBM DB2 Application Development Client 버전 7.2

DB2는 다른 버전과 같이 동일한 자국어를 Windows XP 시스템에서 지원합니다.

제한사항

설치 중 사용자 ID 및 암호를 입력할 때, 설치 패널에 입력한 사용자 어카운트가 유효하더라도 유효하지 않다는 메시지를 받을 수 있습니다. 이것은 머신에 존재하는 사용자 ID에서만 발생합니다. 아직 존재하지 않는 사용자 이름을 입력할 때 이 문제점이 발생하면 안됩니다.

db2admin을 제외한 임의의 사용자 어카운트로 DB2를 설치하도록 선택한 경우, 어카운트 이름이 DB2 이름 지정 규칙과 일치해야 합니다. 가장 중요한 점은 이름에 공백이 포함되면 안됩니다. 예를 들어, `my_name`은 사용할 수 있지만, `my name`은 사용할 수 없습니다.

제품 설치 중에 오류 1052를 수신한 경우, 다음을 수행하십시오.

1. 오류 창을 열어 두십시오.
2. 명령 창을 여십시오.
3. 명령 **db2start.exe**를 실행하십시오.

4. 명령행에 표시된 사용자에 대해 지정한 암호를 사용하여 오류 창에 지정한 명령을 실행하십시오.
5. 오류 창으로 돌아간 다음, **확인**을 누르십시오. 이제 설치가 계속됩니다.

중국어를 사용 중이며 제어 센터의 글꼴이 적절하게 표시하지 않은 경우, `filename.\u5b8b\u4f53=simsun.ttf` 항목을 `filename.\u5b8b\u4f53=simsun.ttc`로 대체하여 `sqllib\java\java12\jdk\jre\lib\font.properties.zh`를 수정하십시오.

Windows ME

다음 제품 및 버전은 FixPak 2 이상과 함께 설치될 때 Windows ME를 지원합니다.

- IBM DB2 UDB Personal Edition 버전 7.1
- IBM DB2 Personal Developer's Edition 버전 7.1
- IBM DB2 Universal Developer's Edition 버전 7.1
- IBM DB2 Connect Personal Edition 버전 7.1
- IBM DB2 Run-Time Client 버전 7.1
- IBM DB2 Administration Client 버전 7.1
- IBM DB2 Application Development Client 버전 7.1

제한사항

HTML 검색 서버 기능이 현재 Window ME에서 지원되지 않습니다.

DB2 설치 제거시, `MFC42U.DLL` 파일을 찾을 수 없다는 오류 메시지를 수신할 수 있습니다. DB2를 완전히 설치 제거하려면, 설치 제거 작업이 완료된 후에 수동으로 sqllib 디렉토리를 삭제하십시오.

Windows 2000 Datacenter Server

다음 DB2 제품은 Windows 2000 Datacenter Server, Windows 2000 Advanced Server 및 Windows 2000 Server에 대해 인증됩니다.

- IBM DB2 Universal Database Enterprise - Extended Edition 버전 7.2
- IBM DB2 Universal Database Enterprise Edition 7.2

- IBM DB2 Database Workgroup Edition 버전 7.2
- IBM DB2 Connect Enterprise Edition 버전 7.2

Windows 95에 DB2 설치

DB2를 비영문 Windows 95 시스템에 설치하는 경우, DB2 UDB를 설치하기 *전에* 수동으로 Winsock 버전을 Winsock 2로 갱신해야 합니다. Winsock 2 업그레이드 유틸리티는 Microsoft로부터 구할 수 있습니다.

Windows 2000에 DB2 설치

Windows 2000에서 이전 버전의 DB2 위에 설치하거나 현재 버전을 재설치하는 경우, 모든 DB2 서비스의 복구 옵션이 ″동작하지 않음″으로 설정되었는지 확인하십시오.

Windows 2000 Terminal Server의 관리 모드에서 DB2 실행

DB2 UDB 버전 7.1, FixPak 3 이상을 사용할 때는 DB2를 Windows 2000 Terminal Server의 관리 모드에서 실행할 수 있습니다. 이전 버전에서는 DB2가 Windows 2000 Terminal Server의 응용프로그램 서버 모드만 지원했습니다.

Microsoft SNA Server 및 SNA 멀티사이트 갱신(2단계 커미트) 지원

호스트와 AS/400 응용프로그램은 SNA 제품으로 Microsoft SNA Server를 사용 중일 때 SNA 2단계 커미트를 사용하여 DB2 UDB 서버에 액세스할 수 없습니다. 이것이 지원된다는 DB2 UDB 서적의 내용은 잘못된 것입니다. Windows NT용 IBM 통신 서버 버전 5.02 이상이 필요합니다.

주: Windows용 DB2 UDB를 사용하여 호스트 및 AS/400 데이터베이스 서버에 액세스하는 응용프로그램은 Microsoft SNA Server 버전 4 서비스 팩 3 이상을 사용하는 SNA 2단계 커미트를 사용*할 수 있습니다.*

Windows NT용 IBM 통신 서버(CS/NT)에서 사용자 ID 및 암호 정의

DB2 서버에 연결하기 위해 리모트 DB2 클라이언트에 대한 통신 프로토콜로 APPC를 사용하며 SNA 제품으로 CS/NT를 사용하는 경우, CS/NT 구성 파일에서 다음의 키워드가 올바르게 설정되었는지 확인하십시오. 이 파일은 `x:\ibmcs\private` 디렉토리에서 주로 찾을 수 있습니다.

노드 정의

TG_SECURITY_BEHAVIOR

이 매개변수는 TP가 보안을 위해 구성되지 않은 경우에 노드가 ATTACH에 제시된 보안 정보를 처리하는 방법을 사용자가 판별할 수 있게 합니다.

IGNORE_IF_NOT_DEFINED

이 매개변수는 사용자가 ATTACH에 보안 매개변수가 제시되었는지의 여부를 판별하고 TP가 보안을 위해 구성되지 않은 경우에 이를 무시할 수 있게 합니다.

IGNORE_IF_NOT_DEFINED를 사용하는 경우, CS/NT에서 사용자 ID와 암호를 정의할 필요가 없습니다.

VERIFY_EVEN_IF_NOT_DEFINED

이 매개변수는 사용자가 ATTACH에 보안 매개변수가 제시되었는지를 판별하고 TP가 보안을 위해 구성되지 않은 경우라도 이를 확인할 수 있게 합니다. 이 값이 기본값입니다.

VERIFY_EVEN_IF_NOT_DEFINED를 사용하는 경우 CS/NT에서 사용자 ID와 암호를 정의해야 합니다.

CS/NT 사용자 ID와 암호를 정의하려면, 다음의 단계를 수행하십시오.

1. **시작 --> 프로그램 --> IBM 통신 서버 --> SNA 노드 구성**. 통신 서버 구성 환영 창이 열립니다.
2. 수정하려는 구성 파일을 선택하십시오. **다음**을 누르십시오. 구성 시나리오 선택 창이 열립니다.
3. CPI-C, APPC 또는 5250 에뮬레이션을 강조표시하십시오. **완료**를 누르십시오. 통신 서버 SNA 노드 창이 열립니다.

4. CPI-C 및 APPC 옆의 [+]를 누르십시오.
5. LU6.2 보안 옆의 [+]를 누르십시오.
6. 사용자 암호를 마우스 오른쪽 버튼으로 누르고 **작성**을 선택하십시오. 사용자 ID 암호 정의 창이 열립니다.
7. 사용자 ID와 암호를 입력하십시오. **확인**을 누르십시오. **완료**를 눌러 변경사항을 적용하십시오.

이동 가능한 드라이브가 접속되지 않은 경우 DB2 설치 정지

접속되지 않은 제거 가능한 드라이브가 있는 컴퓨터를 사용하여 DB2를 설치하는 경우, 설치 유형을 선택한 후 설치가 정지될 수 있습니다. 이 문제점을 해결하려면, -a 옵션을 지정하여 설치 프로그램을 수행하십시오.

```
setup.exe -a
```

Windows 2000에서 CLP 사용시 SQL1035N 오류 발생

일부 사용자(예를 들어, 관리자)만이 쓰기 권한을 가진 디렉토리에 DB2가 설치된 경우, DB2 명령행 처리기를 사용하려고 시도할 때 일반 사용자에게 SQL1035N 오류가 발생할 수 있습니다.

이 문제점을 해결하려면, DB2는 모든 사용자가 쓰기 액세스를 가진 디렉토리에 설치되어야 합니다.

특수 레지스터로 정의된 뷰와 관련된 이주 발행

특수 레지스터 USER 또는 CURRENT SCHEMA가 뷰 컬럼을 정의하는 데 사용된 경우, 데이터베이스 이주 이후에 뷰를 사용할 수 없게 됩니다. 예를 들어, 다음과 같습니다.

```
create view v1 (c1) as values user
```

버전 5에서 USER 및 CURRENT SCHEMA는 데이터 유형이 CHAR(8)이었지만 버전 6 이후로는 VARCHAR(128)로 정의되었습니다. 이 예에서 뷰가 버전 5에서 작성된 경우, 컬럼 c1의 데이터 유형은 CHAR이고 데이터베이스 이주 이후

에 CHAR로 남아 있게 됩니다. 뷰가 이주 이후에 사용되면, 런타임에 컴파일되지만 데이터 유형이 일치하지 않으므로 실패합니다.

이를 해결하려면 뷰를 삭제하고 다시 작성하십시오. 뷰를 삭제하기 전에, SYSCAT.VIEWS 카탈로그 뷰를 쿼리하여 뷰 생성에 사용된 구문을 캡처하십시오. 예를 들어, 다음과 같습니다.

```
select text from syscat.views where viewname='<>'
```

Windows 2000에서 IPX/SPX 프로토콜 지원

이 정보는 빠른 시작 책의 설치 계획 장에 있는 "가능한 클라이언트 서버 연결 시나리오" 절을 참조합니다.

공개된 프로토콜 지원 도표는 완전하게 올바르지는 않습니다. Windows 2000 클라이언트는 IPX/SPX를 사용하는 OS/2 또는 Unix 기반 서버에 연결되지 않습니다. IPX/SPX를 사용하는 Windows 2000 서버에 연결된 OS/2 또는 Unix 기반 클라이언트는 지원되지 않습니다.

DB2의 이전 버전을 업그레이드하기 전에 DB2 프로세스 중지

이 정보는 Windows용 DB2 빠른 시작 책의 이주 정보를 참조합니다.

Windows 머신에서 수행 중인 DB2의 이전 버전을 업그레이드할 경우, 설치 프로그램은 메모리에 DB2 DLL을 보유하는 프로세스 목록을 포함하는 경고를 제공합니다. 이 때, 그 목록에 표시되는 프로세스를 수동으로 중지하거나 설치 프로그램이 이 프로세스를 자동으로 종료할 수 있도록 옵션을 사용할 수 있습니다. 데이터 유실을 피하려면 설치하기 전에 모든 DB2 프로세스를 수동으로 중지하는 것이 좋습니다. DB2 프로세스가 수행되지 않도록 하는 최상의 방법은 Windows 서비스 패널을 통해 시스템의 프로세스를 보는 것입니다. Windows 서비스 패널에서 DB2 서비스, OLAP 서비스 또는 데이터 웨어하우스 서비스가 수행되고 있는지 확인하십시오.

주: Windows 플랫폼에서는 한 번에 하나의 DB2 버전만 수행할 수 있습니다. 예를 들어, 같은 Windows 머신에서 DB2 버전 7과 DB2 버전 6을 동시에 수행할 수 없습니다. DB2 버전 6이 설치된 머신에서 DB2 버전 7을 설치할 경

우, 설치 프로그램은 설치하는 동안 DB2 버전 6을 삭제합니다. DB2 이전 버전으로부터의 이주에 대해서는 해당되는 *빠른 시작* 매뉴얼에서 자세한 내용을 참조하십시오.

다른 DB2 제품이 이미 설치된 경우의 DB2 설치 이후 db2iupdt 수행

다음 정보는 빠른 시작 설치 문서에서 제공되어야 합니다.

UNIX 기반 시스템에서 DB2 UDB 버전 7을 설치할 때 이미 DB2 제품이 설치되어 있으면, **db2iupdt** 명령을 실행하여 이 제품의 새 기능을 사용할 인스턴스를 갱신해야 합니다. 일부 기능은 이 명령을 수행할 때까지 사용할 수 없습니다.

DB2 제어 센터를 수행할 Linux 환경 설정

이 정보는 빠른 시작 책의 "DB2 제어 센터 설치" 장에 포함되어 있어야 합니다.

Linux에서 DB2 설치 프로그램을 나가서 터미널 창으로 복귀하여, 다음 명령을 입력하여 DB2 제어 센터를 수행할 수 있는 올바른 환경을 설정하십시오.

```
su -l <instance name>
export JAVA_HOME=/usr/jdk118
export DISPLAY=<your machine name>:0
```

그리고 나서, 다른 터미널 창을 열고 다음을 입력하십시오.

```
su root
xhost +<your machine name>
```

터미널 창을 닫고 인스턴스 소유자 ID로 로그인한 터미널에 리턴한 후 다음 명령을 입력하십시오.

```
db2cc
```

S/390 상의 Linux용 DB2 Universal Database Enterprise Edition 및 DB2 Connect Enterprise Edition

DB2 Universal Database Enterprise Edition 및 DB2 Connect Enterprise Edition이 이제 S/390의 Linux에서 사용 가능합니다. S/390 머신에 Linux를 설치하기 전에, 소프트웨어 및 하드웨어 요구사항을 알아야 합니다.

하드웨어

S/390 9672 Generation 5 이상, Multiprise 3000.

소프트웨어

- S/390용 SuSE Linux v7.0 또는 zSeries 및 S/390용 Turbolinux Server 6
- S/390용 패치와 함께 커널 레벨 2.2.16(아래 참조)
- glibc 2.1.3
- libstdc++ 6.1

S/390 Linux에는 다음 패치가 필요합니다.

- 지금은 패치가 필요하지 않습니다.

최근 갱신 사항을 보려면, http://www.software.ibm.com/data/db2/linux 웹 사이트로 가십시오.

주:

1. 32비트 Intel 기반 Linux 및 S/390 상의 Linux에서만 지원됩니다.
2. 다음은 DB2 버전 7에서 Linux/390에서 사용할 수 없습니다.
   - DB2 UDB Enterprise - Extended Edition
   - DB2 Extenders
   - DB2 Data Links Manager
   - DB2 Administrative Client
   - 암호 변경 지원
   - LDAP 지원
   - TSM

- 원시 디바이스(raw device) 사용

S/390용 Linux에서의 데이터 손실 가능성

S/390용 Linux에서 2.2 시리즈 커널과 함께 DB2를 사용할 때 Linux 머신에서 사용 가능한 RAM은 1GB 미만으로 제한되어야 합니다. RAM을 1GB로 제한해야 Linux 커널 버그로 인한 DB2에서의 데이터 손실이 발생하지 않습니다.

이러한 제한은 S/390용 Linux의 DB2에만 해당되며 Intel의 Linux에는 해당되지 않습니다.

커널 패치는 IBM developerworks 사이트에서 구할 수 있고, 1GB 이상의 RAM을 사용할 수 있습니다.

Linux에서 DB2용 Gnome 및 KDE 바탕 화면 통합

DB2는 이제 지원되는 Intel 기반 Linux 분산 프로그램용 Gnome 및 KDE 바탕 화면에 가장 일반적으로 사용되는 DB2 도구를 시작하는 DB2 폴더 및 아이콘을 작성하기 위한 유틸리티 세트가 제공됩니다. 이 유틸리티는 DB2 버전 7.2에서 기본 설치되며, 설치한 유틸리티를 사용하여 하나 이상 선택된 사용자의 바탕 화면 아이콘을 작성하거나 제거할 수 있습니다.

하나 이상 선택된 사용자의 바탕 화면 아이콘 세트를 추가하려면 다음 명령을 사용하십시오.

```
db2icons <user1> [<user2> <user3>...]
```

주: Gnome 또는 KDE 바탕 화면 환경을 실행할 때 아이콘이 생성된 경우에는 바탕 화면을 직접 새로 고쳐야 새 아이콘이 표시됩니다.

하나 이상 사용자의 바탕 화면에서 바탕 화면 아이콘 세트를 제거하려면 다음 명령을 사용하십시오.

```
db2rmicons <user1> [<user2> <user3>...]
```

주: 다른 사용자의 아이콘을 생성하거나 제거하려면 충분한 권한이 있어야 합니다. 일반 사용자는 대개 **db2icons** 및 **db2rmicons**를 사용하여 자신의 아이콘을

작성하거나 제거할 수 있고, root 사용자나 지정 사용자의 홈 디렉토리에 대한 쓰기 권한을 가진 사용자만 다른 사용자의 아이콘을 작성하거나 제거할 수 있습니다.

Solaris 커널 구성 매개변수(권장 값)

UNIX용 DB2 빠른 시작 및 *UNIX용 DB2 Enterprise - Extended Edition 빠른 시작*의 Solaris 시스템 장에서 시작하기 전에 섹션은 권장되는 Solaris 커널 구성 매개변수를 제공합니다. 다음 테이블은 512MB보다 많은 실제 메모리를 가진 시스템에 대한 추가적인 커널 구성 매개변수 권장사항을 제공합니다.

표 1. Solaris 커널 구성 매개변수(권장 값)

커널 매개변수	512MB-1GB	1GB-4GB	4GB+
msgsys:msginfo_msgmax	65,535	65,535	65,535
msgsys:msginfo_msgmnb	65,535	65,535	65,535
msgsys:msginfo_msgmap	514	1,026	2,050
msgsys:msginfo_msgmni	512	1,024	2,048
msgsys:msginfo_msgssz	16	32	64
msgsys:msginfo_msgtql	1,024	2,048	4,096
msgsys:msginfo_msgseg	32,767	32,767	32,767
shmsys:shminfo_shmmax	483,183,820 - 966,367,641	966,367,641 - 3,865,470,566	3,865,470,566 - 4,294,967,296
shmsys:shminfo_shmseg	50	100	200
shmsys:shminfo_shmmni	300	1,024	2,048
semsys:seminfo_semmni	1,024	2,048	4,198
semsys:seminfo_semmap	1,026	2,050	4,096
semsys:seminfo_semmns	2,048	4,096	8,192
semsys:seminfo_semmnu	2,048	4,096	8,192
semsys:seminfo_semume	50	50	50

주:

1. shmsys:shminfo_shmmax 매개변수 제한은 32비트 시스템의 경우 4GB입니다.
2. msgsys:msginfo_msgmnb 및 msgsys:msginfo_msgmax 매개변수는 65,535 이상으로 설정되어야 합니다.

3. msgsys:msginfo_msgseg 매개변수는 32,767보다 높지 않게 설정되어야 합니다.
4. shmsys:shminfo_shmmax 매개변수는 위의 테이블에서 제안된 값이나 실제 메모리(바이트 단위)의 90% 수준으로 설정되어야 합니다. 예를 들어, 시스템에 196MB의 실제 메모리가 있는 경우, shmsys:shminfo_shmmax 매개변수를 184,968,806(196*1024*1024*0.9)으로 설정하십시오.

UNIX용 DB2 Universal Database Enterprise - Extended Edition 빠른 시작

제 5 장 Linux에서 DB2 Universal Database 설치 및 구성에서는 Linux EEE 클러스터의 각 물리 노드의 커널, glibc 및 libstdc++ 레벨이 같아야 함을 나타내야 합니다.

Linux용 DB2 EEE의 시험 사용판을 다음 웹 사이트에서 다운로드받을 수 있습니다. http://www6.software.ibm.com/dl/db2udbdl/db2udbdl-p

HP-UX용 shmseg 커널 매개변수

빠른 시작 책에 제공된 HP-UX 커널 구성 매개변수 갱신에 대한 정보가 올바르지 않습니다. HP-UX용 shmseg 커널 매개변수에 대한 권장 값을 무시해야 합니다.

기본 HP-UX 값(120)을 대신 사용해야 합니다.

IBM Visual Warehouse 제어 데이터베이스 이주

*Windows용 DB2 Universal Database 빠른 시작*은 Windows NT와 Windows 2000에서 DB2 Universal Database 버전 7의 일반 설치를 수행하는 동안 활동 중인 웨어하우스 제어 데이터베이스가 이주되는 방법에 대한 정보를 제공합니다. 하나 이상의 웨어하우스 제어 데이터베이스를 이주시켜야 하는 경우, 웨어하우스 제어 데이터베이스 관리 창을 사용하여 추가 데이터베이스를 이주시켜야 합니다. 한 번에 하나의 웨어하우스 제어 데이터베이스만이 활동 상태가 될 수 있습니다. 이주시킨 마지막 데이터베이스가 다음에 Data Warehouse Center에 로그온할 때

사용하려는 것이 아닌 경우, 웨어하우스 제어 데이터베이스 관리 창을 사용하여 사용하려는 데이터베이스를 등록해야 합니다.

db2uiddl 명령을 사용하여 고유 인덱스 이주

DB2 빠른 시작 매뉴얼의 DB2 이전 버전 설치 이주 작업 장의 선택적 이전 버전 이주 작업을 보면, **db2uiddl** 명령을 사용해서 DB2 버전 5.x와 DB2 버전 6에서 고유 인덱스를 이주해야 한다고 적혀 있습니다. 이는 잘못된 설명입니다. **db2uiddl** 명령을 사용하여 고유 인덱스를 이주하는 작업은 DB2 버전 5 이하에서 이주하는 경우에만 필요합니다.

64비트 AIX 버전 설치 오류

db2setup을 사용하여 64비트 AIX DB2 이미지를 기존의 AIX 운영 체제에 설치할 경우, AIX 호환 버전을 사용하고 있는지 확인하십시오. 호환 버전이 아니면 설치가 실패합니다. AIX 버전 5 DB2 이미지는 기존 AIX 버전 4 운영 체제에 설치되지 않습니다. 이에 따라 64비트 AIX 버전 4 DB2 이미지를 기존 AIX 버전 5 운영 체제에 설치하게 되면 설치 오류가 발생합니다.

호환되지 않는 64비트 AIX 버전을 설치하려고 시도하면 db2setup 유틸리티가 요구사항 검사시에 버전 불일치를 찾아내어 다음과 같은 오류 메시지를 출력합니다.

```
DBI1008E 설치 미디어와 AIX 버전이 일치하지 않습니다.
```

올바른 64비트 AIX 버전을 설치하면 이런 오류를 피할 수 있습니다.

SMIT 사용

SMIT를 사용하는 경우 AIX 버전 4 DB2가 AIX 버전 5 DB2에 의해 대체된다는 오류가 표시됩니다. 그러나 그 반대는 발생하지 않습니다. 따라서 64비트 AIX 버전 5 사용자는 반드시 올바른 버전을 사용해야 합니다. 오류 메시지 없이 db2setup을 실행할 수 있다면 올바른 AIX 호환 버전을 사용하고 있음을 나타내는 것입니다.

주: 이 호환성 불일치에 따른 오류는 32비트 AIX 버전에는 적용되지 않습니다.

이주 중 오류

이주에 성공한 경우라도 이주 중에 db2diag.log 파일에 오류 항목(데이터베이스가 이주되지 않음)이 나타나는데, 이는 무시해도 되는 오류입니다.

웨어하우스 제어 데이터베이스 관리를 사용할 경우, 오류는 SQLLIB\LOGGING 디렉토리에 로그됩니다. IWH2RGn.LOG 파일에는 오류 정보가 들어 있습니다. 오류가 발생하면 오류를 정정하고 제어 데이터베이스를 삭제한 다음 재시작해야 합니다. 기존 제어 데이터베이스의 경우, 이 작업을 위해서는 백업 사본을 사용해야 합니다.

IBM® DB2® Connect 사용권 활성화

DB2 Connect Enterprise Edition의 설치 프로그램인 DB2 Connect Unlimited Edition과 DB2 Connect Web Starter Kit는 제품 사용권을 설치하지 않습니다. 따라서 이 제품을 설치하면 사용권 파일이 없는 관계로 90일 동안 시험 후 구매 모드로 작동합니다. 해당 사용권을 설치하지 않을 경우 90일이 지나면 설치 제품의 기능은 중지됩니다.

제품 사용권을 활성화하기 위해 DB2 사용권 센터나 **db2licm** 명령 중 하나를 사용할 수 있습니다.

사용권 센터를 사용한 사용권 키 설치 및 사용권 유형 설정

1. DB2 제어 센터를 시작하고 **도구** 메뉴에서 **사용권 센터**를 선택합니다.
2. 사용권을 설치할 시스템을 선택합니다. **설치된 제품** 필드에 설치한 제품 이름이 표시될 것입니다.
3. **사용권** 메뉴에서 **추가**를 선택합니다.
4. 사용권 추가 창에서 **파일에서 추가** 단일선택 버튼을 선택하여 사용권 파일을 선택합니다.
   - Windows 서버의 경우, x:\db2\license\connect*license_filename*입니다. 여기서, x:는 DB2 Connect 제품 CD가 들어있는 CD-ROM을 나타냅니다.
   - UNIX 서버의 경우, /db2/license/connect/*license_filename*입니다.

여기서, DB2 Connect Enterprise Edition과 DB2 Connect Unlimited Edition에 대한 *license_filename*은 **db2conee.lic**이며, DB2 Connect Web Starter Kit에 대해서는 **db2consk.lic**입니다.

5. **적용**을 눌러 사용권 키를 추가합니다.
6. 사용권 유형을 설정하십시오.
   - DB2 Connect Unlimited Edition과 DB2 Connect Web Starter Kit의 경우:

     사용권 센터에서 **사용권** 메뉴에서 **변경**을 선택하십시오. 사용권 변경 창에서 **측정된 사용** 선택란을 선택합니다. **확인**을 눌러 사용권 변경 창을 닫은 후 사용권 센터로 되돌아갑니다.

     주: DB2 Connect Web Starter Kit의 경우, 만기 날짜가 제품 설치일로부터 270일로 설정되었는지 확인합니다.
   - DB2 Connect Enterprise Edition의 경우:

     사용권 센터에서 **사용권** 메뉴에서 **변경**을 선택하십시오. 사용권 변경 창에서 구매한 사용권의 유형을 선택합니다.

     – 동시 사용자 사용권을 구매한 경우에는 **동시 DB2 Connect 사용자**를 선택한 다음 구매한 사용자 사용권 수를 입력합니다.

       주: DB2 Connect Enterprise Edition은 한 사람의 사용자에게 사용권을 제공합니다. 추가적인 DB2 Connect 사용자의 사용권은 각각 따로 구매해야 합니다.

     – 등록된 사용자 사용권을 구매한 경우에는 **등록된 DB2 Connect 사용자**를 선택한 다음 **확인**을 눌러서 사용권 변경 창을 닫고 사용권 센터로 돌아가십시오. **사용자** 탭을 눌러 사용권을 구매한 모든 사용자 ID를 추가하십시오.

db2licm 명령을 사용하여 사용권 키 설치 및 사용권 유형 설정

사용권 센터가 아닌 **db2licm** 명령을 사용해서 사용권 키를 추가할 수 있습니다. **db2licm** 명령으로 사용자의 사용권 키를 추가하려면 다음과 같이 하십시오.

1. Windows 서버에서는 다음 명령을 입력하십시오.

   ```
   db2licm -a x:\db2\license\connect\license_filename
   ```

여기서, x:는 DB2 Connect 제품 CD가 들어있는 CD-ROM 드라이브를 나타냅니다.

UNIX 서버에서는 다음 명령을 입력하십시오.

```
db2licm -a db2/license/connect/license_filename
```

여기서, DB2 Connect Enterprise Edition과 DB2 Connect Unlimited Edition에 대한 *license_filename*은 db2conee.lic이고, DB2 Connect Web Starter Kit에 대해서는 db2consk.lic입니다.

주: DB2 Connect Web Starter Kit의 경우, 만기 날짜가 제품 설치일로부터 270일로 설정되었는지 확인합니다.

2. 사용권 유형을 설정하십시오.
   - DB2 Connect Unlimited Edition과 DB2 Connect Web Starter Kit의 경우:

     다음 명령을 입력하십시오.

     ```
     db2licm -p db2conee measured
     ```
   - DB2 Connect Enterprise Edition의 경우:

     동시 사용자 사용권을 구매한 경우에는 다음 명령을 입력하십시오.

     ```
     db2licm -p db2conee concurrent
     db2licm -u N
     ```

     여기서, *N*은 구매한 사용권의 동시 사용자 수를 나타냅니다.

     등록된 사용자 사용권을 구매한 경우에는 다음 명령을 입력하십시오.

     ```
     db2licm -p db2conee registered
     ```

분산 설치시의 사용권 고려사항

분산 설치에 필요한 이미지를 작성하는 경우, 제품 설치 후 사용권을 설치할 수 있도록 특별한 조정이 필요합니다. 위에서 설명한 **db2licm** 명령을 분산 설치 스크립트에 추가하십시오.

웨어하우스 제어 데이터베이스에 액세스

Windows NT에서의 DB2 버전 7의 일반 설치에서 DB2 버전 7 웨어하우스 제어 데이터베이스가 웨어하우스 서버와 함께 작성됩니다. Visual Warehouse 웨어하우스 제어 데이터베이스가 있는 경우, 웨어하우스 제어 데이터베이스의 메타데이터를 DB2 버전 7 Data Warehouse Center에서 사용할 수 있도록 이주시키기 전에 웨어하우스 제어 데이터베이스가 포함된 DB2 서버를 DB2 버전 7로 업그레이드해야 합니다. 계속 사용할 웨어하우스 제어 데이터베이스를 버전 7로 이주해야 합니다. 사용 중인 웨어하우스 제어 데이터베이스의 메타데이터는 DB2 버전 7 설치 프로세스 중에 버전 7로 이주됩니다. 추가 웨어하우스 제어 데이터베이스의 메타데이터를 이주시키려면, Windows NT에서 **시작 --> 프로그램 --> IBM DB2 --> 웨어하우스 제어 데이터베이스 관리**를 선택하여 시작하는 웨어하우스 제어 데이터베이스 이주 유틸리티를 사용하십시오. *Windows용 DB2 Universal Database 빠른 시작*에서 웨어하우스 제어 데이터베이스 이주에 대한 내용을 참조하십시오.

Enterprise Edition UNIX CD-ROM에서의 시험 사용판

UNIX 플랫폼용 DB2 Universal Database(UDB) Enterprise Edition(EE) CD-ROM 버전 6 및 버전 7에는 DB2 Connect Enterprise Edition(CEE)의 90일 시험 사용판이 들어 있습니다. DB2 Connect 기능성은 DB2 UDB EE 제품으로 빌드되므로, DB2 Connect 기능성을 사용하기 위해 DB2 UDB EE가 설치된 시스템에 DB2 CEE 제품을 설치할 필요가 없습니다. DB2 CEE의 90일 시험 사용판을 설치하고 사용권 버전으로 업그레이드하려고 결정한 경우, DB2 CEE 제품을 구입하고 DB2 CEE 사용권 키를 설치해야 합니다. 제품을 재설치할 필요가 없습니다. 사용권 키를 설치하는 명령어는 UNIX용 DB2 EE 또는 DB2 CEE 빠른 시작 책에서 제공됩니다.

EE 설치와 함께 시험 CEE 제품을 설치한 경우, 다음 명령어로 CEE 90일 시험 사용판을 제거할 수 있습니다. Connect EE의 시험 사용판을 제거하여도 DB2 CEE와 함께 사용 가능한 DB2 Connect는 여전히 설치되어 있습니다.

DB2 Connect 버전 7을 제거하려면, 각 플랫폼에서 다음 Fileset를 설치 제거하십시오.

- AIX에서 db2_07_01.clic Fileset를 설치 제거하십시오.
- NUMA-Q 및 Solaris 운영 환경에서 db2clic71 패키지를 설치 제거하십시오.
- Linux에서 db2clic71-7.1.0-x RPM을 설치 제거하십시오.
- HP-UX에서 DB2V7CONN.clic Fileset를 설치 제거하십시오.

DB2 Connect 버전 6을 제거하려면, 각 플랫폼에서 다음 Fileset를 설치 제거하십시오.

- AIX에서 db2_06_01.clic Fileset를 설치 제거하십시오.
- NUMA-Q 및 Solaris 운영 환경에서 db2cplic61 패키지를 설치 제거하십시오.
- Linux에서 db2cplic61-6.1.0-x RPM을 설치 제거하십시오.
- HP-UX에서 DB2V6CONN.clic Fileset를 설치 제거하십시오.

DB2 Connect Enterprise Edition UNIX CD-ROM에서의 시험 사용판

UNIX 플랫폼용 DB2 Connect Enterprise Edition(EE) CD-ROM 버전 6 및 버전 7에는 DB2 Universal Database(UDB)의 90일 시험 사용판이 들어 있습니다. DB2 UDB EE 90일 시험 사용판은 평가를 위해 제공되지만, 작업할 DB2 Connect에는 필요하지 않습니다.

DB2 UDB EE의 90일 시험 사용판을 설치하고 사용권 버전으로 업그레이드하려고 결정한 경우, DB2 UDB 제품을 구입하고 DB2 UDB EE 제품을 구입하고 DB2 UDB EE 사용권 키를 설치해야 합니다. 제품을 재설치할 필요가 없습니다. 사용권 키를 설치하는 명령어는 UNIX용 DB2 EE 또는 DB2 CEE 빠른 시작 책에서 제공됩니다. Connect EE 설치와 함께 시험 UDB EE 제품을 설치한 경우, 다음 명령어로 EE 90일 시험 사용판을 제거할 수 있습니다. DB2 UDB EE의 시험 사용판을 제거한 경우, DB2 Connect EE의 기능성에 영향을 미치지 않습니다.

DB2 UDB EE 버전 7을 제거하려면, 각 플랫폼에서 다음 Fileset를 설치 제거하십시오.

- AIX에서 db2_07_01.elic Fileset를 설치 제거하십시오.
- NUMA-Q 및 Solaris 운영 환경에서 db2elic71 패키지를 설치 제거하십시오.

- Linux에서 db2elic71-7.1.0-x RPM을 설치 제거하십시오.
- HP-UX에서 DB2V7ENTP.elic Fileset를 설치 제거하십시오.

DB2 UDB EE 버전 6을 제거하려면, 각 플랫폼에서 다음 Fileset를 설치 제거하십시오.

- AIX에서 db2_06_01.elic Fileset를 설치 제거하십시오.
- NUMA-Q 및 Solaris 운영 환경에서 db2elic61 패키지를 설치 제거하십시오.
- Linux에서 db2elic61-6.1.0-x RPM을 설치 제거하십시오.
- HP-UX에서 DB2V6ENTP.elic Fileset를 설치 제거하십시오.

UNIX의 Merant 드라이버 관리자 및 DB2 UDB 버전 7 ODBC 드라이버

Merant 드라이버 관리자가 UNIX에서 DB2의 ODBC 드라이버를 액세스할 때 유니코드 지원과의 비호환성이 발견되었습니다. 이러한 비호환성으로 인해 응용프로그램의 유니코드 사용 요청 여부에 상관없이 Merant 드라이버 관리자에서 유니코드를 사용합니다. 이로 인해, Merant 드라이버 관리자가 비IBM 데이터 소스를 지원해야 하는 Data Warehouse Center, Information Catalog Manager 및 MQSI와 같은 제품에 문제점이 발생할 수 있습니다. 영구적인 솔루션이 사용 가능할 때까지 유니코드 지원이 사용 가능하지 않은 대체 DB2 ODBC 드라이버 라이브러리를 사용할 수 있습니다. DB2 UDB의 영향을 받는 버전으로는 FixPak 2 이상의 버전 7.1 및 모든 FixPak 레벨의 버전 7.2가 있습니다.

유니코드 지원이 사용 가능하지 않은 대체 DB2 ODBC 드라이버 라이브러리는 AIX, HP-UX 및 Solaris 운영 환경용 DB2 버전 7.1 및 7.2와 함께 제공되었습니다. 이 대체 라이브러리를 사용하려면, 이 라이브러리의 사본을 작성한 후 원래 DB2 ODBC 드라이버 라이브러리의 이름을 제공해야 합니다.

주: 대체(_36) 라이브러리에는 DB2 JDBC 드라이버에서 필요로 하는 유니코드 함수가 들어 있습니다. 이 라이브러리를 사용해도 여전히 WebSphere Application Server를 포함한 JDBC 응용프로그램이 DB2와 성공적으로 작업할 수 있습니다.

AIX, HP-UX 또는 Solaris 운영 환경에서 비유니코드 ODBC 라이브러리로 전환하려면, 다음 지시사항을 참조하십시오. 이것은 수동 프로세스이므로 연속적인 FixPak을 적용한 후 그리고 제품을 갱신할 때마다 이를 수행해야 합니다.

AIX:

AIX에서 필수 라이브러리를 작성하려면 다음을 수행하십시오.

1. 인스턴스 소유자로, **db2stop force** 명령을 사용하여 데이터베이스 인스턴스를 모두 종료하십시오.
2. 관리 인스턴스 ID로, **db2admin stop force** 명령을 사용하여 관리 서버 인스턴스를 종료하십시오.
3. `/usr/lpp/db2_<ver>_<rel>/lib` 아래의 원본 `db2.o`를 백업하십시오.
4. 루트로, **slibclean**을 발행하십시오.
5. db2_36.o를 db2.o로 복사하여 소유권 및 사용권한이 일치하는지 확인하십시오. 다음 명령을 사용하십시오.

   ```
   cp db2_36.o db2.o
   -r--r--r-- bin:bin for db2.o
   ```

원래 오브젝트로 다시 전환하려면, db2_36.o 대신 백업 파일을 사용하여 같은 프로시저를 수행하십시오.

Solaris 운영 환경:

필수 라이브러리를 작성하려면 다음을 수행하십시오.

1. 인스턴스 소유자로, **db2stop force** 명령을 사용하여 데이터베이스 인스턴스를 모두 종료하십시오.
2. 관리 인스턴스 ID로, **db2admin stop force** 명령을 사용하여 관리 서버 인스턴스를 종료하십시오.
3. `/opt/IBMdb2/V<ver>.<rel>/lib` 아래의 원본 `libdb2.so.1`을 백업하십시오.
4. `libdb2_36.so.1`을 `libdb2.so.1`로 복사하여 소유권 및 사용권한이 일치하는지 확인하십시오. 다음 명령을 사용하십시오.

```
cp libdb2_36.so.1 libdb2.so.1
-r-xr-xr-x bin:bin libdb2.so.1
```

5. 각 데이터베이스 인스턴스에 대해 **db2iupdt <instance>**를 발행하고 DAS 인스턴스에 대해 **dasiupt <das_instance>**를 발행하십시오.

원래 오브젝트로 다시 전환하려면, libdb2_36.so.1 대신 백업 파일을 사용하여 같은 프로시저를 수행하십시오.

HP-UX - FixPak 4 이상에서만 사용 가능:

일시적인 해결책을 시도하기 *전에* FixPak 4 이상을 설치해야 합니다.

HP-UX에서 필수 라이브러리를 작성하려면 다음을 수행하십시오.

1. **db2stop force**를 사용하여 데이터베이스 인스턴스를 모두 종료하십시오.
2. **db2admin stop force**를 사용하여 관리 서버 인스턴스를 종료하십시오.
3. /opt/IBMdb2/V<ver>.<rel>/lib 아래의 원본 libdb2.sl을 백업하십시오.
4. libdb2_36.sl을 libdb2.sl로 복사하여 소유권 및 사용권한이 일치하는지 확인하십시오. 다음 명령을 사용하여 일관성이 있는지 확인하십시오.

   ```
   -r-xr-xr-x bin:bin for libdb2.sl
   ```

5. 각 데이터베이스 인스턴스에 대해 **db2iupdt <instance>**를 발행하고 DAS 인스턴스에 대해 **dasiupdt <das_instance>**를 발행하십시오.

원래 오브젝트로 다시 전환하려면, libdb2_36.sl 대신 백업 파일을 사용하여 같은 프로시저를 수행하십시오.

기타 UNIX 운영 체제:

기타 UNIX 운영 체제에서 DB2 및 Merant 드라이버 관리자에 대한 지원이 필요하면, IBM 고객만족센터에 문의하십시오.

웹용 정보 카탈로그 센터를 설치하기 전에 필요한 추가 구성

웹용 정보 카탈로그 센터를 설치하기 전에 설치 중인 언어에 대한 dg_strings.hti 파일을 /sqllib/icuweb/macro 디렉토리에 복사해야 합니다. 해당 언어 디렉토리에는 dg_strings.hti 파일이 있습니다. 빠른 시작 안내서 및 대부분의 다른 DB2 문서의 NLS 부록에는 언어 코드 목록이 있습니다.

코드 페이지 및 언어 지원 정보 - 정정

빠른 시작 매뉴얼의 자국어 지원(NLS) 부록에 수록된 코드 페이지 및 언어 지원 테이블에는 다음과 같은 오류가 있습니다.

- 열 표제, "국가 코드"는 "국가/지역 코드"가 되어야 합니다.
- 열 표제, "언어"는 "언어/스크립트"가 되어야 합니다.
- 슬로베니아 코드는 테이블에 표기된 것처럼 "si"가 아니라 "sl"이 되어야 합니다.

DB2 Data Links Manager 빠른 시작

AIX 5.1 지원

DB2 Data Links File Manager 및 File Filter 구성요소는 현재 AIX 5.1에서 완벽하게 지원됩니다. Data Links와 관련되고 전에 AIX 이전 릴리스에서 지원되던 모든 도구와 지시사항 또한 현재 AIX에서 완벽하게 지원됩니다.

메시지 ″접두부에 대한 afsfid 가져오기 오류 발생″으로 Dlfm 시작이 실패함

DCE-DFS 환경에서 수행 중인 Data Links Manager의 경우, dlfm 시작이 다음 오류로 실패하면 IBM 서비스에 문의하십시오.

```
접두부에 대한 afsfid를 가져오는 중 오류가 발생했습니다
```

″dlfm add_prefix″를 사용하여 Data Links Manager에 등록된 DFS 파일세트가 삭제된 경우 오류가 발생할 수 있습니다.

아카이브 파일에 대한 Tivoli Storage Manager 클래스 설정

아카이브 파일에 사용할 TSM 관리 클래스를 지정하려면, DLFM_TSM_MGMTCLASS DB2 레지스트리 항목을 적합한 관리 클래스 이름으로 설정하십시오.

DFS Client Enabler에 대한 디스크 공간 요구사항

DFS Client Enabler는 DB2 Universal Database 클라이언트 또는 서버 설치 중에 선택할 수 있는 선택 구성요소입니다. DFS Client Enabler가 DB2 UDB 클라이언트 또는 서버를 필요로 하지 않고 자체적으로 수행되더라도 DB2 Universal Database 클라이언트 또는 서버 제품을 설치하지 않고는 DFS Client Enabler를 설치할 수 없습니다. DFS Client Enabler 코드에 필요한 2MB의 디스크 공간 이외에 DB2 Run-Time Client 설치의 일부로 DFS Client Enabler를 설치하는 경

우에는 추가로 40MB를 확보해야 합니다. DB2 Administration Client 또는 DB2 서버 설치의 일부로 DFS Client Enabler를 설치하는 경우, 보다 많은 디스크 공간이 필요합니다. DB2 Universal Database 제품의 디스크 공간 요구사항에 관한 추가 정보는 *UNIX용 DB2 빠른 시작* 설명서를 참조하십시오.

AIX에서 데이터 링크 파일 관리 프로그램 백엔드 프로세스 모니터

dlfm see 명령의 출력이 변경되었습니다. AIX에서 데이터 링크 파일 관리 프로그램 백엔드 프로세스를 모니터하기 위해 이 명령을 발행하면, 리턴되는 출력은 다음과 유사합니다.

```
   PID      PPID      PGID   RUNAME    UNAME        ETIME DAEMON NAME
 17500     60182     40838     dlfm     root        12:18 dlfm_copyd_(dlfm)
 41228     60182     40838     dlfm     root        12:18 dlfm_chownd_(dlfm)
 49006     60182     40838     dlfm     root        12:18 dlfm_upcalld_(dlfm)
 51972     60182     40838     dlfm     root        12:18 dlfm_gcd_(dlfm)
 66850     60182     40838     dlfm     root        12:18 dlfm_retrieved_(dlfm)
 67216     60182     40838     dlfm     dlfm        12:18 dlfm_delgrpd_(dlfm)
 60182         1     40838     dlfm     dlfm        12:18 dlfmd_(dlfm)


DLFM SEE 요청이 성공하였습니다.
```

괄호 안의 이름은 dlfm 인스턴스의 이름이며 이 경우에는 ″dlfm″입니다.

AIX용 DB2 Data Links Manager 설치 및 구성: DCE-DFS 환경에서의 추가 설치 고려사항

″설치 요구사항″ 섹션에 다음 정보가 추가되어야 합니다.

```
DFS 3.1용 e-fix 또는 PTF 세트 1(사용할 수 있는 경우) 중 하나를
설치해야 합니다. e-fix는 다음에서 사용할 수 있습니다.

http://www.transarc.com/Support/dfs/datalinks/efix_dfs31_main_page.html
```

또한,

```
dfs 클라이언트는 Data Links Manager를 설치하기 전에 수행 중이어야 합니다.
db2setup 또는 smitty를 사용하십시오.
```

″키탭 파일″ 섹션에는 다음과 같이 정정해야 하는 오류가 있습니다.

```
핵심부와 암호 정보가 포함되는 이 키탭 파일은
datalink.ktb라고 해야 하며 ....
```

올바른 이름: `datalink.ktb`가 아래 예에서 사용됩니다. 이 파일의 작성은 DLMADMIN 인스턴스가 작성되지 않으면 발생할 수 없기 때문에 "키탭 파일" 절은 "DCE-DFS 사후 설치 태스크" 아래로 이동되어야 합니다.

"Data Links Manager 서버 및 클라이언트" 섹션에 Data Links Manager 서버는 Data Links Manager 클라이언트 이전에 설치되어야 한다는 것이 명시되어야 합니다.

새로운 섹션인 "백업 디렉토리"가 추가되어야 합니다.

```
백업 메소드가 로컬 파일 시스템에 대한 것일 경우,
이 디렉토리는 DFS 파일 시스템의 디렉토리이어야 합니다.
이 DFS 파일 세트가 DFS 관리자에 의해 작성되었는지
확인하십시오. 이것은 DMLFS 파일 세트가 아니어야 합니다.
```

실패한 "dlfm add_prefix" 명령

DCE/DFS 환경에서 수행 중인 Data Links Manager의 경우, **dlfm add_prefix** 명령은 -2061 리턴 코드(백업 실패)로 실패할 수도 있습니다. 이러한 상황이 발생할 경우, 다음 단계를 수행하십시오.

1. **dlfm stop** 명령을 발행하여 Data Links Manager 디먼 프로세스를 중지하십시오.
2. **dlfm stopdbm** 명령을 발행하여 DB2 프로세스를 중지하십시오.
3. **dce_login root** 명령을 발행하여 dce 루트 신임장을 확보하십시오.
4. **dlfm startdbm** 명령을 발행하여 DB2 프로세스를 시작하십시오.
5. **dlfm add_prefix** 명령을 발행하여 Data Links Manager로 파일 세트를 등록하십시오.
6. **dlfm start** 명령을 발행하여 Data Links Manager 디먼 프로세스를 시작하십시오.

복사 디먼이 dlfm stop으로 중지되지 않는 드문 이벤트

사용자가 dlfm stop을 발행하거나 비정상적인 종료가 있는 경우에 dlfm_copyd(복사 디먼)가 중지하지 않는 상황이 가끔 발생하기도 합니다. 이러한 상황이 발생하면, dlfm의 재시작을 시도하기 전에 dlfm shutdown을 발행하십시오.

AIX용 DB2 Data Links Manager 설치 및 구성: db2setup 유틸리티를 사용하여 AIX에 DB2 Data Links Manager 설치

″DB2 데이터베이스 DLFM_DB가 작성됨″ 섹션에서 DLFM_DB는 DCE_DFS 환경에서 작성되지 않습니다. 이는 사후 설치 단계로 수행해야 합니다.

″DMAPP에 대한 DCE-DFS 사전 시작 등록″ 섹션에서 2단계는 다음으로 변경되어야 합니다.

```
2. DFS가 시작될 때 DMAPP가 시작되도록
   명령이 /opt/dcelocal/tcl/user_cmd.tcl에 추가됩니다.
```

AIX용 DB2 Data Links Manager 설치 및 구성: DCE-DFS 사후 설치 태스크

다음의 새로운 섹션인 ″Data Links Manager 설치 완료″가 추가되어야 합니다.

Data Links Manager 서버에서 다음 단계를 수행하여 설치를 완료해야 합니다.

1. ″AIX용 DB2 Data Links Manager 설치 및 구성″ 장의 ″DCE-DFS 환경에서의 추가 설치 고려사항″ 섹션에 있는 ″키탭 파일″에 요약된 대로 키탭 파일을 작성하십시오.
2. root 사용자로 다음 명령을 입력하여 DMAPP를 시작하십시오.

   ```
   stop.dfs all
   start.dfs all
   ```
3. 다음과 같이 dce 루트 신임장을 사용하여 ″dlfm setup″을 수행하십시오.
   a. Data Links Manager 관리자 DLMADMIN으로 로그인하십시오.
   b. root 사용자로 dce_login을 발행하십시오.
   c. dlfm setup 명령을 입력하십시오.

Data Links Manager 클라이언트에서 다음 단계를 수행하여 설치를 완료하십시오.

1. ″AIX용 DB2 Data Links Manager 설치 및 구성″ 장의 ″DCE-DFS 환경에서의 추가 설치 고려사항″ 섹션에 있는 ″키탭 파일″에 요약된 대로 키탭 파일을 작성하십시오.
2. root 사용자로 다음 명령을 입력하여 DMAPP를 시작하십시오.

```
stop.dfs all
start.dfs all
```

AIX용 DB2 Data Links Manager 설치 및 구성: Smit를 사용하여 수동으로 DB2 Data Links Manager 설치

″SMIT 사후 설치 태스크″ 섹션의 7단계에서 명령 ″dce_login root″가 ″dlfm setup″ 이전에 발행되어야 한다고 내용을 수정하십시오. 11단계는 필요하지 않습니다. 이 단계는 6단계(dlfm server_conf)나 8단계(dlfm client_conf)가 수행되면 자동으로 수행됩니다. 또한, 12단계도 제거하십시오(dlfm start). 설치를 완료하려면, 다음 단계를 수행하십시오.

1. ″AIX용 DB2 Data Links Manager 설치 및 구성″ 장의 ″DCE-DFS 환경에서의 추가 설치 고려사항″ 섹션에 있는 ″키탭 파일″에 요약된 대로 키탭 파일을 작성하십시오.
2. root 사용자로 다음 명령을 입력하여 DMAPP를 시작하십시오.

```
stop.dfs all
start.dfs all
```

DB2 Data Links DFS Client Enabler 설치 및 구성

″DFS Client Enabler 구성″ 섹션에서 다음 정보를 2단계에 추가하십시오.

```
"secval" 명령을 수행하면 항상 구성이 완료됩니다.
그러나, 머신을 재부트하려면 이 명령도 필요할 수 있습니다.
READ PERMISSION DB 파일에 액세스하는 중에 문제점이 발견될 경우,
DB2 DFS Client Enabler가 설치된 머신을 재부트하십시오.
```

Solaris 운영 체제용 DB2 Data Links Manager 설치 및 구성

Solaris 운영 체제용 DB2 Data Links Manager를 설치한 후에 다음 조치를 수행해야 합니다.

1. 다음 세 행을 /etc/system 파일에 추가하십시오.

```
set dlfsdrv:glob_mod_pri=0x100800
set dlfsdrv:glob_mesg_pri=0xff
set dlfsdrv:ConfigDlfsUid=UID
```

여기서, *UID*는 id dlfm의 사용자 ID를 나타냅니다.

2. 머신을 재부트하여 변경사항을 활성화하십시오.

Windows NT에서 데이터 링크의 관리자 그룹 특권

Windows NT에서 dlmadmin 사용자는 DataLinks를 사용하여 링크된 파일에 관해서는 UNIX의 root 사용자와 대부분의 함수에 대해 동일한 특권을 가집니다. 다음 표는 둘을 비교합니다.

작업	**Unix(root)**	**Windows NT(dlmadmin)**
이름 바꾸기	예	예
토큰 없이 파일 액세스	예	예
삭제	예	아니오(아래 주 참조)
갱신	예	아니오(아래 주 참조)

주: NTFS는 읽기 전용 파일에 대해 이런 작업을 허용하지 않습니다. dlmadmin 사용자는 파일에 대한 쓰기 사용 권한을 이용하여 이 작업들을 수행할 수 있습니다.

Data Links File System Filter(DLFF) 설치에 대한 로깅 최소화

`dlfs_cfg` 파일을 변경하여 Data Links File System Filter(DLFF) 설치에 대한 로깅을 최소화할 수 있습니다. `dlfs_cfg` 파일은 드라이버 및 구성 매개변수를 로드하기 위해 strload 루틴으로 전달됩니다. 파일은 `/usr/lpp/db2_07_01/cfg/` 디렉토리에 있습니다. 기호 링크를 통해, 파일을 `/etc` 디렉토리에서 찾을 수도 있습니다. `dlfs_cfg` 파일은 다음 형식이 있습니다.

```
d <driver-name> <vfs number> <dlfm id> <global message priority>
  <global module priority> - 0 1
```

여기서,

d d 매개변수는 드라이버가 로드된다는 것을 지정합니다.

driver-name

driver-name은 로드될 드라이버의 전체 경로입니다. 예를 들어, DB2 버전 2의 전체 경로는 /usr/lpp/db2_07_01/bin/dlfsdrv입니다. 드라이버 이름은 dlfsdrv입니다.

vfs number

/etc/vfs의 DLFS에 대한 vfs 항목입니다.

dlfm id

DataLinks Manager 관리자의 사용자 ID입니다.

global message priority

DLFS 드라이버에 있는 구성 가능한 매개변수입니다. 시스템 로그 파일에 로그될 메시지 범주의 목록을 정의합니다.

global module priority

DLFS 드라이버에 있는 구성 가능한 매개변수입니다. 시스템 로그 파일에 로그될 Vnode 조작, VFS 조작 및 드라이버 루틴의 목록을 정의합니다.

0 1 0 1은 이 드라이버에 대한 비복제 노드 작성을 위한 부속 번호입니다. 노드 이름은 복제된 드라이버 노드 이름에 부속 번호를 첨부하여 작성됩니다. 다섯 개의 부속 번호 이상은 제공될 수 없습니다(0-4).

실제 예는 다음과 같습니다.

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,255,-1 - 0 1
```

로그되는 메시지는 전역 메시지 우선순위와 전역 모듈 우선순위에 따라 다릅니다. 로깅을 최소화하기 위해, 전역 메시지 우선순위 값을 변경할 수 있습니다.

네 개의 메시지 우선순위 값을 사용할 수 있습니다.

```
#define LOG_EMERGENCY      0x01
#define LOG_TRACING        0x02
#define LOG_ERROR          0x04
#define LOG_TROUBLESHOOT   0x08
```

DLFF의 메시지 대부분의 메시지 우선순위는 LOG_TROUBLESHOOT입니다. 다음은 몇 가지 대체 구성 예입니다.

비상 메시지와 오류 메시지가 필요할 경우, dlfs_cfg 구성 파일에서 전역 메시지 우선순위를 5(1+4)로 설정하십시오.

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,5,-1 - 0 1
```

오류 메시지만 필요할 경우, 전역 메시지 우선순위를 4로 설정하십시오.

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,4,-1 - 0 1
```

DLFS에 대한 로깅이 필요하지 않으면, 전역 메시지 우선순위를 0으로 설정하십시오.

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,0,-1 - 0 1
```

설치 후 메시지 로깅

설치 후에 비상, 오류 및 문제 해결 메시지를 로그하려면, dlfs_cfg 파일을 수정해야 합니다. dlfs_cfg 파일은 /usr/lpp/db2_07_01/cfg 디렉토리에 있습니다. 전역 메시지 우선순위는 255(최대 우선순위) 또는 13(8+4+1)으로 설정해야 합니다. 우선순위를 13(8+4+1)으로 설정하면 비상, 오류 및 문제 해결 정보를 로그합니다.

전역 메시지 우선순위를 설정한 후, DLFS 필터 파일 시스템을 마운트 해제하고 dlfsdrv 드라이버를 재로드하여 새 우선순위 값이 로드 시간에 설정되도록 하십시오. dlfsdrv 드라이버를 재로드한 후, DLFS 필터 파일 시스템을 다시 마운트해야 합니다.

주: dlfs_cfg의 설정은 dlfs_cfg 파일을 다시 변경하기 전까지 dlfsdrv 드라이버 로드 시마다 유지됩니다.

Sun Solaris 시스템 로그온 최소화

dlfs_cfg 파일은 Sun™ Solaris™ 시스템에 없습니다. 대신 시스템 파일 /etc/syslog.conf에 시스템 메시지를 해당 로그 파일로 전달하기 위해 시스템 로그 디먼(syslogd)이 사용하는 정보가 들어 있습니다. /etc/syslog.conf에서 kern.notice 및 kern.debug에 대한 항목을 주석을 넣어 DLFF 설치에 대한 로그를 최소화할 수 있습니다. 그런 다음 변경사항을 적용하려면 syslogd를 중지하고 시작해야 합니다.

모든 커널 주의사항 및 오류의 로그를 재활성화하려면, /etc/syslog.conf에서 kern.notice 및 kern.debug에 대한 항목의 주석을 삭제한 다음 syslogd를 다시 중지하고 시작하십시오.

DATALINK 리스토어

롤 포워드의 사용에 상관없이 데이터베이스 리스토어 이후에 취한 오프라인 백업의 리스토어에서는 빠른 조정 처리가 수반되지 않습니다. 이러한 경우, 파일 링크 제어 하에 DATALINK 컬럼이 있는 모든 테이블은 데이터 링크 조정 보류(DRP) 상태에 있게 됩니다.

Data Links Manager 삭제

이제 지정된 데이터베이스에 대한 DB2 Data Links Manager를 삭제할 수 있습니다. 백업/리스토어와 같은 유틸리티와 마찬가지로 일부 Data Links 관련 SQL 요청의 처리는 데이터베이스에 구성된 모든 DLM과의 통신을 포함합니다. 이전에, 비록 DB2는 조작 가능하지 않았지만 구성된 DLM을 삭제하는 기능을 갖지 않았습니다. 이것은 SQL 및 유틸리티 처리에 추가적인 오버헤드를 가져옵니다. DLM이 추가되었으면, 일부 SQL 요청(예를 들어, 테이블/테이블 스페이스/데이터베이스 삭제)의 실패를 가져올 수 있는 요청 처리에서 엔진이 DLM과 통신했습니다.

SMIT May Remove Additional Filesets를 사용하여 DLFM 구성요소 설치 제거

Data Links Manager가 설치되어 있는 AIX 머신에서 DB2(버전 5, 6 또는 7)를 설치 제거하기 전에, 다음 단계를 따르십시오.

1. root 사용자로, 다음 명령을 사용하여 /etc/vfs를 복사하십시오.

```
cp -p /etc/vfs /etc/vfs.bak
```

2. DB2를 설치 제거하십시오.
3. root 사용자로, /etc/vfs를 1단계에서 작성된 백업으로 바꾸십시오.

```
cp -p /etc/vfs.bak /etc/vfs
```

호스트 이름을 시작/판별하기 전에

각 DB2 서버와 Data Links 서버의 이름을 판별해야 합니다. 설치를 검증하려면 이러한 호스트 이름을 알아야 합니다. DB2 Data Links File Manager에 연결할 때, DB2 UDB 서버는 내부적으로 다음 정보를 DLFM에 보냅니다.

- 데이터베이스 이름
- 인스턴스 이름
- 호스트 이름

그러면 DLFM은 이 정보를 내부 테이블과 비교하여 연결 허용 여부를 결정합니다. 데이터베이스 이름, 인스턴스 이름 및 호스트 이름의 조합이 dlfm add_db 명령을 사용하여 등록된 경우에만 연결을 허용합니다. dlfm add_db 명령에 사용된 호스트 이름은 DB2 UDB 서버가 내부적으로 보낸 호스트 이름과 정확히 일치해야 합니다.

호스트 이름을 다음과 같이 사용하십시오.

1. DB2 서버에서 hostname 명령을 입력하십시오. 예를 들어, 이 명령이 db2server를 리턴할 것입니다.
2. 플랫폼에 따라, 다음 중 하나를 수행하십시오.
   - AIX에서는 host db2server 명령을 입력하십시오. 여기서, db2server는 이전 단계에서 구해진 이름입니다. 이 명령은 다음과 유사한 출력을 리턴해야 합니다.

     ```
     db2server.services.com is 9.11.302.341,  Aliases:  db2server
     ```
   - Windows NT에서는 nslookup db2server 명령을 입력하십시오. 여기서, db2server는 이전 단계에서 구해진 이름입니다. 이 명령은 다음과 유사한 출력을 리턴해야 합니다.

     ```
     Server: dnsserv.services.com
     Address: 9.21.14.135
     Name: db2server.services.com
     Address: 9.21.51.178
     ```

- Solaris 운영 환경에서는 cat /etc/hosts | grep ‘hostname’을 입력하십시오. /etc/hosts에서 도메인 이름 없이 호스트 이름을 지정할 경우, 명령은 다음과 유사한 출력을 리턴해야 합니다.

  ```
  9.112.98.167 db2server loghost
  ```

  도메인 이름과 함께 호스트 이름을 지정할 경우, 명령은 다음과 유사한 출력을 리턴해야 합니다.

  ```
  9.112.98.167 db2server.services.com loghost
  ```

dlfm add_db 명령을 사용하여 DB2 UDB 데이터베이스를 등록할 때, 호스트 이름에 db2server.services.com을 사용하십시오. dlfm add_db 명령에서 다른 별명을 사용하면 DLFM과의 DB2 서버의 내부 연결이 실패합니다.

Data Links 서버는 DB2 ″add datalinks manager for database database_alias using node hostname port port_number″ 명령을 사용하여 DB2 데이터베이스에 등록됩니다.

hostname은 Data Links 서버의 이름입니다. Data Links 서버의 별명을 이 명령에서 사용할 수 있습니다. 이 Data Links 서버를 참조하는 DATALINK 값은 URL 값에 호스트 이름을 지정해야 합니다. 즉, URL 값을 DATALINK 컬럼에 지정할 때 ″add datalinks manager″ 명령에 사용된 정확한 이름을 사용해야 합니다. 다른 별명을 사용하면 SQL문 실패를 초래합니다.

DB2 Data Links File Manager 작업: DB2 데이터베이스에서 DB2 Data Links Manager를 삭제한 후 정리

DROP DATALINKS MANAGER 명령을 사용하여 DB2 Data Links Manager 삭제시, 명령 자체는 DB2 Data Links Manager에 대한 해당 정보를 정리하지 않습니다. 사용자가 명시적으로 데이터베이스에 링크된 파일의 링크 해제 및 백업 정보의 가비지 콜렉션을 실시할 수 있습니다. dlfm drop_dlm 명령을 사용하여 이를 수행합니다. 이 명령은 특정 데이터베이스에 대한 모든 정보의 비동기 삭제를 시작합니다. 이 명령이 성공하려면 DB2 Data Links Manager가 수행 중이어야 합

니다. 반드시 DB2 Data Links Manager를 삭제한 후에 이 명령을 사용해야 합니다. 그렇지 않으면, DB2 Data Links Manager에 관한 중요 정보가 유실되어 복구할 수 없습니다.

특정 데이터베이스에 대해 링크 해제 처리 및 백업 정보의 가비지 콜렉션을 실시하려면, 다음을 수행하십시오.

1. DB2 Data Links Manager 관리자로 시스템에 로그온하십시오.
2. 다음 명령을 실행하십시오.

   ```
   dlfm drop_dlm database instance hostname
   ```

   여기서,

   ```
   database는 리모트 DB2 UDB 데이터베이스의 이름입니다.
   instance는 데이터베이스가 상주하는 인스턴스입니다.
   hostname은 데이터베이스가 상주하는 DB2 UDB 서버의
   호스트 이름입니다.
   ```
3. 로그오프하십시오.

이 명령에 사용되는 컨텍스트를 보여주는 완전한 사용시나리오는 *Command Reference*를 참조하십시오.

새 오류 코드가 이 명령에 대해 작성되었습니다(57 페이지의 『DLFM1001E(새 오류 메시지)』 참조).

dlfm Client_conf 실패시 사용자 조치

DLFM 클라이언트에서 dlfm client_conf가 실패했다면, DB2 카탈로그에 있는 ″오래된″ 항목 때문일 수도 있습니다. 이를 해결하려면 다음 명령을 발행하십시오.

```
db2 uncatalog db <dbname>
db2 uncatalog node <node alias>
db2 terminate
```

그런 다음, dlfm client_conf를 다시 시도하십시오.

DLFM1001E(새 오류 메시지)

DLFM1001E: drop_dlm 처리에 오류.

원인:

DB2 Data Links Manager가 지정된 데이터베이스에 대해 링크 해제 및 가비지 콜렉션 처리를 시작할 수 없습니다. 다음 중 하나가 이 오류의 원인일 수 있습니다.

- DB2 Data Links Manager가 실행 중이 아닙니다.
- 명령에 유효하지 않은 데이터베이스, 인스턴스 및 호스트 이름이 지정되었습니다.
- DB2 Data Links Manager의 구성요소 서비스 중 하나가 실패했습니다.

조치:

다음 단계를 수행하십시오.

1. DB2 Data Links Manager가 실행 중인지 확인하십시오. DB2 Data Links Manager가 아직 실행 중이 아니면 시작하십시오.
2. 데이터베이스, 인스턴스 및 호스트 이름의 조합이 등록된 데이터베이스를 식별하는지 확인하십시오. Data Links Manager에서 ″dlfm list registered databases″ 명령을 사용하여 이를 확인할 수 있습니다.
3. 오류가 계속 발생하면, db2diag.log 파일의 정보를 참조하여 구성요소 서비스(예를 들어, 연결 관리 서비스, 트랜잭션 관리 서비스, 등등)가 실패했는지 확인하십시오. db2diag.log에서 오류 코드를 보고, 오류 코드 아래에 제안된 적절한 조치를 취하십시오.

DLFM 설정 구성 파일 옵션

dlfm setup dlfm.cfg 옵션이 제거되었습니다. 문서에서 이에 대한 언급은 무시하십시오.

파일 리스토어시 잠재적인 문제점

문제점: 동일한 파일의 다른 버전이 다른 시점에 데이터베이스에 링크될 경우, DLFM(Data Links File Manager) 검색 디먼은 데이터베이스 리스토어시 아카이브로부터 올바른 파일 버전을 검색하지 않습니다.

백그라운드: 백업 이미지로부터 데이터베이스가 리스토어될 경우, 해당 백업 이미지에서 링크된 파일은 아카이브로부터 DLFS(Data Links Manager File System)에 리스토어됩니다. 다음은 DB2 Data Links Manager의 검색 및 리스토어 프로세스의 수행 방식을 보여줍니다.

- 디스크에 있는 파일에 대한 현재 버전의 크기 속성 및 최종 수정 시간이 아카이브에서 리스토어되는 파일의 속성과 다른 경우, 디스크의 현재 파일은 해당 파일의 다른 버전으로 취급됩니다. 디스크의 현재 파일은 `filename.MOD`로 저장되며, DLFM 아카이브에서 원래 파일 버전이 리스토어됩니다. 예를 들어, 현재 파일 이름이 `abc`이면, abc는 `abc.MOD`로 복사됩니다.
- 디스크에서 현재 파일의 최종 수정 시간 및 크기 속성이 아카이브에서 리스토어되는 파일의 해당 속성과 동일한 경우, 데이터 링크 검색 디먼은 파일이 수정되지 않았다고 가정하며 아카이브로부터 파일의 버전을 검색하지 않습니다.

중요: 파일을 수정하는 것은 가능하지만 최종 수정 시간 및 크기 속성은 변경하지 마십시오. 그러한 "숨겨진 수정"은 파일 크기에 영향을 미치지 않는 변경사항을 작성하고 최종 수정 시간 속성을 원래 파일의 해당 속성으로 재설정하여 수행됩니다.

예: DBTEST 데이터베이스가 있으며, DATALINK 컬럼이 있는 테이블을 포함한다고 가정하십시오. 그런 다음 나열된 순서로 다음 태스크를 수행합니다.

1. DLFS 탑재 볼륨에서 `fileA`라는 파일을 작성하십시오. 이것은 파일의 처음 버전입니다.
2. fileA 참조(URL)를 DBTEST 데이터베이스에 삽입하십시오.
3. DBTEST 데이터베이스의 백업을 가져오십시오.
4. DBTEST 데이터베이스에서 fileA 참조를 삭제하십시오.
5. DLFS 탑재 볼륨에서 `fileA`를 삭제하십시오.
6. DLFS 탑재 볼륨에서 `fileA`라는 또 다른 파일을 작성하십시오. 이것은 파일의 두 번째 버전입니다.

7. fileA 참조(URL)를 DBTEST 데이터베이스에 삽입하십시오.
8. 백업 이미지에서 DBTEST 데이터베이스를 리스토어하십시오.

DLFM 검색 디먼은 fileA의 두 번째 버전을 fileA.MOD로 복사한 후, fileA의 첫 번째 버전을 fileA의 작업 버전으로 아카이브에서 DLFS 마운트 볼륨으로 복사합니다.

그러나 fileA의 두 가지 버전이 동일한 최종 수정 시간 및 크기 속성을 가진 경우, 파일은 실제로 동일한 버전을 가지고 있다고 가정하므로 DLFM 검색 디먼은 아무것도 수행하지 않습니다.

그 결과 파일의 두 번째 버전이(처음 버전이 아닌) DLFS 탑재 볼륨에 남아 있습니다. 실제로 백업 시간에 있었던 당시와 동일한 상태로 파일 시스템을 리스토어하지는 않습니다.

솔루션: 응용프로그램이 동일한 속성(최종 수정 시간 및 크기)을 가진 해당 파일의 새로운 버전으로 파일을 대체하지 않게 하십시오.

AIX에서 Data Links/DFS Script dmapp_prestart 수행시 오류

```
/usr/sbin/cfgdmepi  -a "/usr/lib/drivers/dmlfs.ext"
```

이 Data Links/DFS Script dmapp_prestart를 수행할 때 위의 명령이 리턴 코드 1로 실패하면, DFS 3.1 ptfset1을 설치하여 cfgdmepi을 수정하십시오.

Tivoli Space Manager와 Data Links 통합

DB2 Data Links Manager는 Tivoli Space Manager의 장점을 활용할 수 있습니다. Tivoli Space Manager HSM(Hierarchical Storage Manager) 클라이언트 프로그램은 자동으로 적합한 파일을 저장영역으로 이주시켜 로컬 파일 시스템에서 특정 레벨의 여유 공간을 관리합니다. 이 프로그램은 이주된 파일을 액세스할 때 자동으로 다시 호출되어 사용자가 특정 파일을 이주하고 다시 호출하도록 허용합니다.

이 기능의 전제조건은 Tivoli Space Manager 버전 4.2입니다.

파일 시스템 공간을 규칙적으로 관리해야 하는 제 3의 스토리지를 정기적으로 이동해야 하는 대형 파일이 있는 파일 시스템을 가지고 있는 사용자는 이 새로운 기능으로 편리하게 작업할 수 있습니다. Tivoli Space Manager는 현재 많은 사용자를 위해 제 3의 스토리지를 관리할 수 있는 방법을 제공합니다. Tivoli Space Manager의 새로운 DB2 Data Links Manager 지원을 통해 DATALINK 파일 공간을 융통성있게 관리할 수 있습니다. Tivoli Space Manager를 사용하면 DB2 Data Links Manager 파일 시스템에 저장될 모든 파일에 대해 충분한 공간을 미리 할당하지 않아도, 파일 시스템을 사용할 때 실수로 가득 채울 위험없이 일정 기간 이상 Data Links 관리 파일 시스템 할당을 조정할 수 있습니다.

Data Links 및 HSM 지원 모두를 파일 시스템에 추가

HSM(Hierarchical Storage Management)으로 파일 시스템을 등록할 때 *먼저* HSM으로 등록한 다음, DataLinks File Manager로 등록하십시오.

1. ″**dsmmigfs add /fs**″ 명령을 사용하여 HSM으로 등록하십시오.
2. ″**dlfmfsmd /fs**″ 명령을 사용하여 DLM으로 등록하십시오.

Data Links 파일 시스템 지원은 다음 항목들을 통해 HSM 파일 시스템용 `/etc/filesystems`의 절에 영향을 줍니다.

```
vfs = dlfs
mount = false
options = rw,Basefs=fsm
nodename = -
```

Data Links 지원을 기존 HSM 파일 시스템에 추가

″**dlfmfsmd /fs**″ 명령을 사용하여 DLM으로 등록하십시오.

HSM 지원을 기존 Data Links 파일 시스템에 추가

1. ″**dsmmigfs add /fs**″ 명령을 사용하여 HSM으로 등록하십시오.
2. ″**dlfmfsmd /fs**″ 명령을 사용하여 DLM으로 등록하십시오.

Data Links-HSM 파일 시스템에서 Data Links 지원 제거

″**dlfmfsmd -j /fs**″ 명령을 사용하여 Data Links 지원을 제거하십시오.

Data Links-HSM 파일 시스템에서 HSM 지원 제거

1. ″**dsmmigfs remove /fs**″ 명령을 사용하여 HSM 지원을 제거하십시오.

2. ″**dlfmfsmd -j /fs**″ 명령을 사용하여 Data Links 지원을 제거하십시오.

3. ″**dlfmfsmd /fs**″ 명령을 사용하여 DLM으로 등록하십시오.

Data Links-HSM 파일 시스템에서 Data Links 및 HSM 지원 제거

1. ″**dsmmigfs remove /fs**″ 명령을 사용하여 HSM 지원을 제거하십시오.

2. ″**dlfmfsmd -j /fs**″ 명령을 사용하여 Data Links 지원을 제거하십시오.

제한사항 및 한계

이 기능은 현재 AIX에만 지원됩니다.

선택적 이주(dsmmigrate) 및 FC(Read permission DB) 링크 파일의 재호출은 root 사용자만 수행할 수 있습니다.

선택적 이주는 Read Permission DB 파일이 DataLink Manager Administrator(dlfm)인 경우, 파일 소유자만 수행할 수 있습니다. 이들 파일에 액세스하려면 호스트 데이터베이스에 있는 토큰이 필요합니다. 토큰이 필요하지 않은 사용자는 ″root″ 사용자 뿐입니다. ″root″ 사용자가 선택적 이주 및 Read Permission DB 파일의 재호출을 수행하는 것이 더 쉽습니다. dlfm 사용자는 처음에만 유효한 토큰을 사용하여 FC 파일을 이주할 수 있습니다. 재호출 후 두 번째로 이주를 시도하면 오류 메시지 ″ANS1028S 내부 프로그램 오류. 서비스 담당자에게 문의하십시오.″가 나타나면서 조작이 실패합니다. FC 파일에서 root 사용자가 아닌 사용자가 **dsmmigrate**를 수행하면 성공하지 못합니다. 일반적으로 파일 서버의 파일에 관리자가 액세스하므로 이러한 제한은 큰 문제점이 되지 않습니다.

stat 및 statfs 시스템 호출은 dlfs가 fsm에 마운트된 경우에도 Vfs-type을 dlfs가 아니라 fsm으로 표시합니다.

위의 동작은 **dsmrecalld** 디먼의 일반적인 기능으로 파일 시스템에서 statfs를 수행하여 Vfs-type이 fsm인지 아닌지 확인합니다.

최소한의 **inode**를 가진 파일이 **FC(Read permission DB)** 링크인 경우, ″**dsmls**″는 출력을 표시하지 않습니다.

dsmls 명령은 **ls** 명령과 비슷하며 TSM이 관리하는 파일들을 나열합니다. 사용자 조치는 필요하지 않습니다.

제 4 장 AIX용 DB2 Data Links Manager 설치 및 구성

공통적인 설치 고려사항

DB2 File Manager 버전 5.2에서 DB2 Data Links Manager 버전 7로 이주

3단계의 정보가 잘못되었습니다. 3단계는 다음과 같아야 합니다.

″3. DLFM 관리자로서, **/usr/lpp/db2_07_01/adm/db2dlmmg** 명령을 수행하십시오.

제 6 장 AIX에서의 설치 검증

NFS 환경에서의 해결 방법

이 섹션에서는 현재 문서에는 보이지 않는 NFS 환경에서 AIX용 DB2 Data Links Manager를 실행할 때 발생하는 알려진 문제점에 대한 임시 해결책에 대해 살펴봅니다. 이러한 문제점들은 NFS에 한정된 것으로 DB2 Data Links Manager나 DB2 Universal Database와는 관계가 없습니다.

추가 NFS 캐시 발행

AIX용 NTF 클라이언트에는 두 개의 서로 다른 캐시가 유지보수됩니다. NFS 클라이언트는 최근에 액세스된 파일 및 디렉토리의 속성과 함께 캐시를 유지보수합니다. 또한 선택적으로 데이터 캐시를 지원해 클라이언트의 파일 내용을 캐싱합니다.

속성 캐싱 프로세스는 때때로 READ PERMISSION DB 파일을 링크한 후에 NFS 클라이언트에 특별한 상황을 발생시킵니다. 파일이 링크되기 전에 시스템에 연결된 사용자는 종종 액세스 제어 토큰 없이도 READ PERMISSION DB 파일에 액세스할 수 있습니다. 다음 방법 중 하나를 사용하여 권한없이 파일에 함부로 액세스하지 못하도록 하십시오.

- SQL **INSERT**문을 실행해 링크를 설정하기 전에 먼저 파일에서 **touch** 명령을 사용합니다.
- 파일이 들어 있는 디렉토리에서 **touch** 명령을 사용합니다.
- 다섯 개의 속성 캐시 구성 매개변수(actimeo, acregmin, acregmax, acdirmin, acdirmax) 중 하나와 함께 **mount** 명령을 사용해서 파일이나 디렉토리를 수정한 다음에도 캐시된 속성이 그대로 유지되는 시간을 최소화합니다.

단 하나의 파일만이 링크되고 NFS 활동이 거의 없기 때문에 사용자는 Data Links 함수를 테스트하는 동안 권한없이 READ PERMISSION DB 파일에 액세스하는 활동을 관찰할 수 있습니다. 그러나, 제품 환경에서는 NFS 활동이 매우 빈번하게 발생하며 일반적으로 NFS 속성 캐시가 모든 링크된 파일을 보유하지 않기 때문에 이러한 시나리오는 거의 발생하지 않습니다.

설치 및 구성 보충 설명서

제 5 장 UNIX 운영 체제에 DB2 클라이언트 설치

HP-UX 커널 구성 매개변수

HP-UX 커널 매개변수 설정의 권장사항에서 **msgmbn** 및 **msgmax**를 65535 이상으로 설정하라는 것은 잘못된 것입니다. 두 매개변수는 정확하게 65535로 설정되어야 합니다.

제 12 장 사용자 자신의 응용프로그램 수행

Run-time Client를 사용한 데이터베이스 유틸리티 바인딩

Run-time Client는 데이터베이스 유틸리티(임포트, 익스포트, 재구성, 명령행 처리기)와 DB2 CLI 바인드 파일을 이 데이터베이스와 사용하기 전에 각 데이터베이스에 바인드하는 데 사용할 수 없습니다. 그 대신 DB2 Administration Client 또는 DB2 Application Development Client를 사용해야 합니다.

이들 데이터베이스 유틸리티와 DB2 CLI 바인드 파일을 해당 데이터베이스와 함께 사용하려면 각 데이터베이스에 바인드해야 합니다. 네트워크 환경에서 다른 운영 체제에서 수행되거나 DB2의 버전이나 서비스 레벨이 다른 여러 클라이언트를 사용하는 경우, 각 운영 체제와 DB2 버전 조합에 대해 한번씩 유틸리티를 바인드해야 합니다.

ODBC를 사용한 DB2에 대한 UNIX 클라이언트 액세스

제12장("사용자 자신의 응용프로그램 수행")에서는 ODBC 클라이언트 응용프로그램이나 ODBC SDK와 함께 ODBC 드라이버 관리자를 설치할 경우, `odbcinst.ini`를 갱신해야 한다고 설명하고 있습니다. 이는 부분적으로 잘못된 정보입니다. Merant ODBC 드라이버 관리자 제품을 설치할 경우에는 `odbcinst.ini`를 갱신할 필요가 없습니다.

제 24 장 여러 데이터 소스에 액세스하기 위해 페더레이티드 시스템 설정

페더레이티드 시스템

DB2 페더레이티드 시스템은 특별한 유형의 분산 데이터베이스 관리 시스템(DBMS)입니다. 페더레이티드 시스템을 사용하면 DBMS에 있는 데이터를 쿼리하고 검색할 수 있습니다. 단일 SQL문은 다중 DBMS 또는 개별적인 데이터베이스를 참조할 수 있습니다. 예를 들어, DB2 Universal Database 테이블, Oracle 테이블 및 Sybase 뷰에 있는 데이터를 조인할 수 있습니다.

DB2 페더레이티드 시스템은 하나 이상의 데이터 소스 및 DB2 인스턴스(페더레이티드 데이터베이스로 사용할 데이터베이스)가 있는 서버로 구성됩니다. 페더레이티드 데이터베이스에는 데이터 소스 및 그 특성을 식별하는 카탈로그 항목이 있습니다. *데이터 소스*는 DBMS 및 데이터로 구성됩니다. 지원되는 데이터 소스는 다음과 같습니다.

- Oracle
- Sybase
- Microsoft SQL Server
- Informix
- DB2 Universal Database 계열 구성원(예: OS/390용 DB2, AS/400용 DB2 및 Windows용 DB2)

DB2 Universal Database 페더레이티드 서버는 *랩퍼*라는 프로토콜을 사용해서 데이터 소스에서 데이터를 검색하고 데이터 소스와 통신합니다. 사용하는 랩퍼는 DB2 인스턴스가 실행되는 운영 체제에 따라 다릅니다. *별명*은 데이터 소스에 위치한 뷰와 테이블을 식별하는 데 사용됩니다. 응용프로그램은 다른 DB2 데이터베이스와 마찬가지로 페더레이티드 데이터베이스에 연결할 수 있으며, 페더레이티드 데이터베이스에 테이블이나 뷰가 있는 것처럼 별명을 사용하여 데이터 소스를 쿼리할 수 있습니다.

페더레이티드 시스템이 설정되면 마치 데이터가 단일 로컬 데이터베이스에 있는 것처럼 데이터 소스 정보에 액세스할 수 있습니다. 사용자 및 응용프로그램은 페더레이티드 데이터베이스에 쿼리를 보내면, 이 데이터베이스가 데이터 소스에서 데이터를 검색합니다.

DB2 페더레이티드 시스템이 작동되는 데는 제한사항이 있습니다. 분산 요청(DR)은 DB2 버전 7의 읽기 전용 조작으로만 제한됩니다. 또한 별명에 대한 유틸리티 조작(LOAD, REORG, REORGCHK, IMPORT, RUNSTATS 등)을 실행할 수 없습니다. 그러나 DDL 및 DML 명령문을 데이터 소스와 연관된 SQL Dialect를 사용하여 직접 DBMS로 제출하는 통과 기능을 사용할 수 있습니다.

제한사항

버전 7.2의 새 랩퍼(예: AIX, HP 및 Solaris 운영 환경의 Informix; Linux, HP 및 Solaris 운영 환경의 Oracle; AIX 및 Solaris 운영 환경의 Sybase; AIX 및 NT의 Microsoft SQL Server)는 이 FixPak에서 사용 불가능하므로 DB2 Relational Connect 버전 7.2를 구입해야 합니다.

DB2 Relational Connect 설치

이 섹션에는 페더레이티드 시스템 서버로 사용할 서버에 DB2 Relational Connect를 설치하는 과정이 나와 있습니다. Relational Connect는 Oracle, Sybase, Microsoft SQL Server 및 Informix 데이터 소스에 액세스하는 데 필요합니다. DB2 Relational Connect를 사용하면 DB2 Universal Database 계열 구성원에 액세스하지 않아도 됩니다.

DB2 Relational Connect 설치 전에,

- DB2 Universal Database Enterprise Edition 또는 DB2 Universal Database Enterprise -- Extended Edition이 페더레이티드 서버에 설치되어 있는지 확인합니다.

  UNIX 서버용 DB2의 경우

  분산 요청에 DB2 계열 데이터베이스를 포함하려면 DB2 Universal Database를 설치할 때 **DB2용 Distributed Join** 데이터 소스 옵션을 선택해야 합니다. 이 옵션이 구현되었는지 검증하려면, **FEDERATED**

매개변수가 **YES**로 설정되어 있는지 확인합니다. **GET DATABASE MANAGER CONFIGURATION** 명령을 실행하여 이 설정을 확인할 수 있는데, 이 명령은 모든 매개변수 및 현재 설정을 표시합니다.

- 페더레이티드 서버에 데이터 소스의 클라이언트 소프트웨어가 설치되었는지 확인하십시오.

Windows NT 서버에 DB2 Relational Connect 설치

1. DB2 Universal Database를 설치하기 위해 작성한 사용자 어카운트를 이용해 페더레이티드 서버에 로그온하십시오.
2. 설치 프로그램이 필요한 파일을 갱신할 수 있도록 수행 중인 프로그램을 종료하십시오.
3. 설치 프로그램을 호출하십시오. 설치 프로그램은 자동 또는 수동으로 호출할 수 있습니다. 설치 프로그램이 자동으로 시작되지 않거나 다른 언어로 설치하려면 설치 프로그램을 수동으로 호출하십시오.
   - 설치 프로그램을 자동으로 호출하려면 다음을 수행하십시오.
     a. 드라이브에 DB2 Relational Connect CD를 넣으십시오.
     b. 자동 수행 기능이 자동으로 설치 프로그램을 시작합니다. 시스템 언어가 결정되고 이 언어에 해당하는 설치 프로그램이 시작됩니다.
   - 설치 프로그램을 수동으로 호출하려면 다음을 수행하십시오.
     a. **시작**을 누르고 **실행** 옵션을 선택하십시오.
     b. **열기** 필드에 다음 명령을 입력하십시오.

        ```
        x:\setup /i language
        ```

        여기서,

        x CD-ROM 드라이브를 나타냅니다.

        language
        사용자 언어의 국가/지역 코드를 나타냅니다(예: 영어는 EN).
     c. **확인**을 누르십시오.

   설치 런치패드가 열립니다.
4. **설치**를 눌러 설치 프로세스를 시작하십시오.

5. 설치 프로그램의 프롬프트에 따르십시오.

   설치가 끝나면 DB2 Relational Connect가 다른 DB2 제품과 함께 디렉토리에 설치됩니다. 예를 들어, Oracle NET8 클라이언트 소프트웨어(net8.dll)에 대한 랩퍼 라이브러리는 c:\Program Files\SQLLIB\bin 디렉토리에 설치됩니다.

UNIX 서버에 DB2 Relational Connect 설치

UNIX 페더레이티드 서버에 DB2 Relational Connect를 설치하려면, db2setup 유틸리티를 사용하십시오.

주: db2setup 유틸리티를 사용할 때 나타나는 화면은 페더레이티드 서버에 설치된 것에 따라 다릅니다. 여기서는 Relational Connect를 설치하지 않은 상황을 가정합니다.

1. root 권한이 있는 사용자로 로그인하십시오.
2. DB2 제품 CD-ROM을 넣고 마운트하십시오. CD-ROM을 마운트하는 방법에 대한 정보는 *UNIX용 DB2 빠른 시작*을 참조하십시오.
3. **cd** /*cdrom* 명령을 입력하여 CD-ROM이 마운트된 디렉토리로 변경하십시오. 여기서, *cdrom*은 제품 CD-ROM의 마운트 지점입니다.
4. **./db2setup** 명령을 입력하십시오. 잠시 후, DB2 V7 설치 창이 열립니다. 이 창에는 설치한 항목과 설치 가능한 항목이 나열됩니다.
5. 설치할 Distributed Join을 탐색(예: **Informix 데이터 소스용 Distributed Join**)한 다음 스페이스 바를 눌러 선택합니다. 옵션을 선택하면 옆에 별표가 나타납니다.
6. **확인**을 선택하십시오. DB2 Services 작성 창이 열립니다.
7. 페더레이티드 서버에 이미 DB2 인스턴스가 포함되어 있으므로 **DB2 인스턴스를 작성하지 않음** 옵션을 선택한 다음 **확인**를 누르십시오.
8. Administration Server 작성을 선택하지 않으면 경고가 나타납니다. **확인**을 선택하십시오. DB2 설치 유틸리티 창에 설치할 유틸리티에 대한 요약 보고서가 표시됩니다. 이전에 Relational Connect를 설치하지 않았다면 다음 두 항목이 표시됩니다.
   - DB2 Relational Connect를 위한 제품 서명
   - 선택한 데이터 소스의 Distributed Join

9. **계속**을 선택하십시오. Relational Connect 설치를 중지할 수 있는 마지막 기회임을 표시하는 창이 나타납니다. **확인**을 눌러 설치를 계속하십시오. 설치를 완료하려면 몇 분 정도 걸립니다.

10. DB2 설치 유틸리티 창에 설치된 구성요소에 대한 상태 보고서가 표시됩니다. **확인**을 선택하십시오. DB2 설치 유틸리티 창이 열립니다. **닫기**를 선택한 후 **확인**을 눌러 유틸리티를 종료하십시오.

    설치가 끝나면 DB2 Relational Connect가 다른 DB2 제품과 함께 디렉토리에 설치됩니다.

    - AIX 서버용 DB2의 경우, 디렉토리는 `/usr/lpp/db2_07_01`입니다.
    - Solaris 운영 환경 서버용 DB2의 경우, 디렉토리는 `/opt/IBMdb2/V7.1`입니다.
    - HP-UX 서버용 DB2의 경우, 디렉토리는 `/opt/IBMdb2/V7.1`입니다.
    - Linux 서버용 DB2의 경우, 디렉토리는 `/usr/IBMdb2/V7.1`입니다.

제 24 장 여러 데이터 소스에 액세스하기 위해 페더레이티드 시스템 설정

별명과 함께 사용되는 스키마 이해

CREATE NICKNAME 명령문의 nickname 매개변수는 2부분(스키마와 별명)으로 구성된 이름입니다. 별명을 작성할 때 스키마를 생략한 경우, 별명 스키마는 별명을 작성하는 사용자의 AuthID가 됩니다. 별명이 작성되고 나면, 별명에 대한 정보가 카탈로그 뷰 SYSCAT.TABLES, SYSCAT.TABOPTIONS, SYSCAT.COLUMNS, SYSCAT.COLOPTIONS 및 SYSCAT.INDEXES에 저장됩니다.

페더레이티드 데이터베이스를 서로 다른 페더레이티드 서버에 리스토어할 때의 문제점

페더레이티드 데이터베이스 백업을 다른 페더레이티드 서버로 리스토어할 경우, 데이터베이스 이미지에는 DB2 계열 데이터 소스에 액세스하는 데 필요한 새 데이터베이스 및 노드 디렉토리 정보가 들어 있지 않습니다. 리스토어를 수행할 때는 이 정보를 카탈로그해야 합니다.

제 26 장 Oracle 데이터 소스에 액세스

DB2 Universal Database는 AIX 및 Windows NT에서 랩퍼를 지원할 뿐 아니라 이제 Linux, Solaris 운영 환경 및 HP-UX에서 Oracle 랩퍼도 지원합니다. 이러한 지원은 Oracle 버전 8로 제한됩니다. 이들 플랫폼에 대한 랩퍼에 액세스하려면 V7.2 DB2 Relational Connect CD를 넣고 Oracle 데이터 소스용 Distributed Join을 선택합니다.

일단 DB2 Relational Connect를 설치하면 페더레이티드 서버에 Oracle 데이터 소스를 추가할 수 있습니다.

1. DB2 페더레이티드 서버에서 Oracle 클라이언트 소프트웨어를 설치하고 구성하십시오.
2. UNIX에서 수행되는 DB2 페더레이티드 서버인 경우, djxlink 스크립트를 실행하여 Oracle SQL*Net 또는 Net8 라이브러리를 DB2 페더레이티드 서버로 링크 편집하고 Oracle에서 사용할 DB2 연합 랩퍼 라이브러리를 작성하십시오.
3. db2dj.ini 파일을 작성(또는 갱신)하여 Oracle에 대한 환경 변수를 추가하십시오. 이 파일은 ORACLE_HOME 환경 변수에 대한 정의를 포함해야 합니다.
4. DB2_DJ_INI 및 DB2_DJ_COMM 프로파일 레지스트리 변수를 설정하십시오(선택적).
5. DB2 페더레이티드 서버에서 Oracle tnsnames.ora 파일의 내용과 위치를 확인하고, Oracle sqlplus를 사용하여 Oracle 서버에 대한 연결을 테스트하십시오.
6. DB2 인스턴스를 재순환하십시오.
7. 랩퍼를 작성하십시오.
8. 서버 정의를 작성하십시오.
9. 사용자 맵핑을 작성하십시오.
10. Set Passthru를 사용하여 구성을 테스트하십시오.
11. 테이블 및 뷰의 별명을 작성하십시오.

환경 변수 설정을 포함하여 이 단계에 대한 자세한 지침은 *DB2 설치 및 구성 보충 설명서*의 제 26 장 Oracle 데이터 소스에 액세스하기 위한 페더레이티드 시스템 설정에 나와 있습니다. 이 정보는 http://www.ibm.com/software/data/db2/relconnect/에서 온라인으로 사용할 수도 있습니다.

문서 오류

『Oracle 데이터 소스를 페더레이티드 시스템에 추가』 섹션에 다음 오류가 있습니다.

- 이 책의 2단계와 3단계 사이에는 추가 단계가 필요합니다. 처음 세 단계는 다음과 같습니다.

  1. Oracle이 제공하는 문서를 사용하여 DB2 페더레이티드 서버에 Oracle 클라이언트 소프트웨어를 설치 및 구성하십시오.
  2. ORACLE_HOME 환경 변수를 설정하십시오.

     ```
     export ORACLE_HOME=<oracle_home_directory>.
     ```

  3. UNIX 플랫폼에서 수행되는 DB2 페더레이티드 서버의 경우, djxlink 스크립트를 수행하여 Oracle SQL*Net 또는 Net8 라이브러리를 DB2 페더레이티드 서버에 링크 편집하십시오. 플랫폼에 따라 djxlink 스크립트는 다음 위치에 있습니다.

     /usr/lpp/db2_07_01/bin - AIX

     /opt/IBMdb2/V7.1/bin - Solaris 운영 환경

     /opt/IBMdb2/V7.1/bin HP-UX

     /usr/IBMdb2/V7.1/bin - Linux

     Oracle의 클라이언트 소프트웨어를 DB2 페더레이티드 서버에 설치한 후에만 djxlink 스크립트를 수행하십시오.

  4. `db2dj.ini` 파일을 수정하고 **db2set** 명령을 실행하여 데이터 소스 환경 변수를 설정하십시오. **db2set** 명령은 사용자 설정으로 DB2 프로파일 레지스트리를 갱신합니다.

     환경 변수 설정에 대한 자세한 지침은 *DB2 설치 및 구성 보충 설명서*의 제 26 장을 참조하십시오.

  5. 이 책에 쓰여진 대로 3단계부터 다음 단계를 계속 수행하십시오.

- 문서에서 다음과 같이 설정하도록 지시하고 있습니다.

  ```
  DB2_DJ_INI = sqllib/cfg/db2dj.ini
  ```

  이것은 잘못된 정보입니다. 다음과 같이 설정해야 합니다.

  ```
  DB2_DJ_INI = $INSTHOME/sqllib/cfg/db2dj.ini
  ```

리모트 LOB으로 작업할 때의 문제점 예방

리모트 LOB 컬럼에 대한 작업을 할 때 메모리 부족 문제점이 발생할 수 있습니다. 예를 들어, Oracle 컬럼에서 LONG 데이터를 선택하여 DB2 테이블에 CLOB로 삽입하는 쿼리를 실행하는 경우, 데이터베이스 응용프로그램 힙 크기를 늘리지 않으면 "메모리 부족"을 나타내는 SQL 오류가 표시될 수 있습니다. 이 오류를 해결하려면 다음을 수행하십시오.

1. DB2 인스턴스에서 모든 응용프로그램을 연결 해제하십시오.
2. 다음 명령을 사용하여 응용프로그램 힙 크기를 갱신하십시오.

   ```
   db2 udpate db cfg for dbname using APPLHEAPSZ 1024
   ```

   여기서, *dbname*은 페더레이티드 데이터베이스의 이름이고 *1024*는 권장되는 힙 크기입니다.
3. 데이터베이스를 초기화하십시오.

이 오류가 발생하지 않도록 하려면 데이터베이스 응용프로그램 힙 크기를 증가시키십시오. 변경사항이 적용되려면 데이터베이스를 다시 초기화하십시오. 예를 들어, 다음과 같습니다.

1. 다음 명령을 사용하여 응용프로그램 힙 크기를 갱신하십시오.

   ```
   db2 udpate db cfg for dbname using APPLHEAPSZ 1024
   ```

   여기서, *dbname*은 페더레이티드 데이터베이스의 이름이고 *1024*는 권장되는 힙 크기입니다.
2. DB2 인스턴스에서 모든 응용프로그램을 연결 해제하십시오.
3. 데이터베이스를 초기화하십시오.

Sybase 데이터 소스 액세스

페더레이티드 서버에 Sybase 데이터 소스를 추가하기 전에 DB2 페더레이티드 서버에 Sybase Open Client 소프트웨어를 설치 및 구성해야 합니다. Open Client 소프트웨어의 설치에 관한 자세한 사항은 Sybase에 포함된 문서에서 설치 절차를 참조하십시오. 설치의 일부로, Sybase 서버에 설치된 Sybase 카탈로그 저장 프로시저 및 DB2 페더레이티드 서버에 설치된 Sybase Open Client 라이브러리가 포함됐는지 확인하십시오.

클라이언트 소프트웨어에서 Sybase 서버로 연결을 구성한 후에, Sybase 도구 중 하나를 사용하여 연결을 테스트하십시오. UNIX용 isql 도구와 Windows용 SQL Advantage 도구를 사용하십시오.

페더레이티드 서버를 설정하여 Sybase 데이터 소스에 저장된 데이터에 액세스하려면 다음을 수행해야 합니다.

1. DB2 Relational Connect 버전 7.2를 설치하십시오. 67 페이지의 『DB2 Relational Connect 설치』를 참조하십시오.
2. Sybase 데이터 소스를 페더레이티드 서버에 추가하십시오.
3. Sybase 코드 페이지를 지정하십시오.

이 장은 2단계 및 3단계를 설명합니다.

이 장의 지침은 Windows NT, AIX 및 Solaris 운영 환경에 적용됩니다. 플랫폼별 차이는 해당 부분에서 설명합니다.

Sybase 데이터 소스를 페더레이티드 서버에 추가

Sybase 데이터 소스를 페더레이티드 서버에 추가하려면 다음을 수행해야 합니다.

1. 환경 변수를 설정하고 프로파일 레지스트리를 갱신하십시오(AIX 및 Solaris만 해당).
2. DB2를 Sybase 클라이언트 소프트웨어에 링크하십시오(AIX 및 Solaris만 해당).
3. DB2 인스턴스를 재순환하십시오(AIX 및 Solaris만 해당).
4. 인터페이스 파일을 작성하고 설정하십시오.

5. 랩퍼를 작성하십시오.
6. DB2_DJ_COMM 환경 변수를 설정하십시오(선택적).
7. 서버를 작성하십시오.
8. CONNECTSTRING 서버 옵션을 설정하십시오(선택적).
9. 사용자 맵핑을 작성하십시오.
10. 테이블 및 뷰에 대한 별명을 작성하십시오.

각 단계는 다음에 자세히 설명되어 있습니다.

1단계: 환경 변수 설정 및 프로파일 레지스트리 갱신(AIX 및 Solaris만 해당)

db2dj.ini 파일을 수정하고 **db2set** 명령을 실행하여 데이터 소스 환경 변수를 설정하십시오. db2dj.ini 파일에는 페더레이티드 서버에 설치된 Sybase 클라이언트 소프트웨어의 구성 정보가 들어 있습니다. **db2set** 명령은 사용자 설정으로 DB2 프로파일 레지스트리를 갱신합니다.

파티션된 데이터베이스 시스템의 특정 인스턴스에서 모든 노드에 하나의 db2dj.ini 파일을 사용하거나 특정 인스턴스에서 하나 이상의 노드에 고유한 db2dj.ini 파일을 사용할 수 있습니다. 파티션되지 않은 데이터베이스 시스템에는 인스턴스별로 단 하나의 db2dj.ini 파일이 있습니다.

환경 변수를 설정하려면 다음을 수행하십시오.

1. sqllib/cfg에 있는 db2dj.ini 파일을 편집하고 다음 환경 변수를 설정하십시오.

   SYBASE="*<sybase home directory>*"

   여기서, <*sybase home directory*>는 Sybase 클라이언트가 설치된 디렉토리입니다.

2. **db2set** 명령을 실행하여 DB2 프로파일 레지스트리를 변경 내용으로 갱신하십시오. **db2set** 명령의 구문은 데이터베이스 시스템 구조에 따라 다릅니다. 이 단계는 다음 데이터베이스 시스템 구조에서 db2dj.ini 파일을 사용하는 경우에만 필요합니다.

   db2dj.ini 파일을 파티션되지 않은 데이터베이스 시스템에서 사용하거나 db2dj.ini 파일을 현재 노드에만 적용하려면 다음을 실행하십시오.

```
db2set DB2_DJ_INI=$HOME/sqllib/cfg/db2dj.ini
```

db2dj.ini 파일을 파티션된 데이터베이스 시스템에서 사용하면서 db2dj.ini 파일의 값을 이 인스턴스의 모든 노드에 적용하려면 다음을 실행하십시오.

```
db2set -g DB2_DJ_INI=$HOME/sqllib/cfg/db2dj.ini
```

db2dj.ini 파일을 파티션된 데이터베이스 시스템에서 사용하면서, db2dj.ini 파일의 값을 특정 노드에 적용하려면 다음을 실행하십시오.

```
db2set -i INSTANCEX  3 DB2_DJ_INI=$HOME/sqllib/cfg/node3.ini
```

여기서,

INSTANCEX
: 인스턴스 이름입니다.

3
: db2nodes.cfg 파일에 있는 노드 번호입니다.

node3.ini
: db2dj.ini 파일의 이름이 바뀐 수정된 버전입니다.

2단계: DB2를 Sybase 클라이언트 소프트웨어에 링크(AIX 및 Solaris 운영 환경에만 해당)

Sybase 데이터 소스에 액세스하려면 DB2 페더레이티드 서버가 클라이언트 라이브러리에 링크 편집되어야 합니다. 링크 편집 프로세스는 페더레이티드 서버가 통신하는 각 데이터 소스의 랩퍼를 작성합니다. djxlink 스크립트를 실행하면 랩퍼 라이브러리가 작성됩니다. djxlink 스크립트를 실행하려면 다음을 입력하십시오.

```
djxlink
```

3단계: DB2 인스턴스 재순환(AIX 및 Solaris 운영 환경에만 해당)

프로그램에서 환경 변수를 설정하려면 DB2 인스턴스를 재순환하십시오. 인스턴스를 재순환하면 DB2 인스턴스를 새로 고쳐서 변경 내용을 적용하십시오.

다음 명령을 실행하여 DB2 인스턴스를 재순환하십시오.

Windows NT 서버용 DB2의 경우,

```
NET STOP instance_name
NET START instance_name
```

AIX 및 Solaris 서버용 DB2의 경우,

```
db2stop
  db2start
```

4단계: 인터페이스 파일 작성 및 설정

인터페이스 파일을 작성하고 설정하려면 파일을 작성한 후 파일에 액세스할 수 있게 해야 합니다.

1. Sybase 제공 유틸리티를 사용하여 액세스할 모든 Sybase Open Servers용 데이터를 포함하는 인터페이스 파일을 작성하십시오. 유틸리티 사용에 관해 자세한 정보는 Sybase 설치 문서를 참조하십시오.

   Windows NT는 일반적으로 이 파일의 이름을 `sql.ini`로 지정합니다. 지금 작성한 파일의 이름을 `sql.ini`에서 `interfaces`로 바꿔서 모든 플랫폼에서 파일 이름을 동일하게 사용하십시오. `sql.ini`의 이름을 `interfaces`로 바꾸지 않으려면 8단계에서 설명한 CONNECTSTRING 옵션이나 IFILE 매개변수를 사용해야 합니다.

   AIX 및 Solaris 시스템에서 이 파일 이름은 `<instance home>/sqllib/interfaces`입니다.

2. DB2가 인터페이스 파일에 액세스할 수 있게 합니다.

   Windows NT 서버용 DB2의 경우,

   파일을 DB2 인스턴스의 `%DB2PATH%` 디렉토리에 두십시오.

   AIX 및 Solaris 서버용 DB2의 경우,

   파일을 DB2 인스턴스의 `$HOME/sqllib` 디렉토리에 두십시오. **ln** 명령을 사용하여 DB2 인스턴스의 `$HOME/sqllib` 디렉토리에서 파일에 링크하십시오. 예를 들어, 다음과 같습니다.

   ```
   ln -s -f /home/sybase/interfaces  /home/db2djinst1/sqllib
   ```

5단계: 랩퍼 작성

CREATE WRAPPER문을 사용하여 Sybase 데이터 소스에 액세스할 때 사용할 랩퍼를 지정하십시오. 랩퍼는 페더레이티드 서버가 데이터 소스와 통신하고 데이터 소스에서 데이터를 검색할 때 사용하는 메커니즘입니다. DB2의 Sybase용 랩퍼 두 가지는 CTLIB 및 DBLIB입니다. 다음은 CREATE WRAPPER문의 예입니다.

```
CREATE WRAPPER CTLIB
```

여기서, CTLIB는 Sybase Open Client 소프트웨어에서 사용하는 기본 랩퍼 이름입니다. CTLIB 랩퍼는 Windows NT, AIX 및 Solaris 서버에서 사용할 수 있습니다.

기본 랩퍼 이름을 선택한 이름으로 대체할 수 있습니다. 그러나 이 경우 CREATE WRAPPER문에 페더레이티드 서버용 랩퍼 라이브러리 이름 및 LIBRARY 매개변수를 포함해야 합니다. 랩퍼 라이브러리 이름 관련 자세한 정보는 *DB2 SQL 참조서*의 CREATE WRAPPER문을 참조하십시오.

6단계: DB2_DJ_COMM 환경 변수 설정

Sybase 데이터 소스에 액세스할 때 성능을 개선하려면 DB2_DJ_COMM 환경 변수를 설정하십시오. 이 변수는 페더레이티드 서버를 초기화할 때 랩퍼가 로드될 것인지 결정합니다. 앞 단계에서 지정한 랩퍼의 해당 랩퍼 라이브러리를 포함하도록 DB2_DJ_COMM 환경 변수를 설정하십시오. 예를 들어, 다음과 같습니다.

AIX 서버용 DB2의 경우,

```
db2set DB2_DJ_COMM='libctlib.a'
```

Solaris 서버용 DB2의 경우,

```
db2set DB2_DJ_COMM='libctlib.so'
```

등호(=) 앞 뒤에 공백이 없는지 확인하십시오.

랩퍼 라이브러리 이름에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오. DB2_DJ_COMM 환경 변수에 대한 자세한 정보는 *관리 안내서*를 참조하십시오.

7단계: 서버 작성

다음과 같이 CREATE SERVER문을 사용하여 액세스할 데이터 소스가 있는 각 Sybase 서버를 정의하십시오.

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
OPTIONS(NODE 'sybnode', DBNAME'sybdb')
```

여기서,

SYBSERVER

Sybase 서버에 할당할 이름입니다. 이 이름은 고유해야 합니다.

SYBASE
: 액세스를 구성할 데이터 소스 유형입니다. Sybase는 지원되는 유일한 데이터 소스입니다.

12.0
: 액세스하는 Sybase 버전입니다. 지원되는 버전은 10.0, 11.0, 11.1, 11.5, 11.9 및 12.0입니다.

CTLIB
: CREATE WRAPPER문에서 지정한 랩퍼 이름입니다.

'sybnode'
: *SYBSERVER*가 있는 노드 이름입니다. 인터페이스 파일에서 노드 값을 얻으십시오. 이 값은 대소문자를 구별합니다.

  노드 이름이 옵션으로 지정되었지만 Sybase 데이터 소스인 경우에는 필수입니다. 추가 옵션에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

'sybdb'
: 액세스할 Sybase 데이터베이스 이름입니다. Sybase 서버에서 이 이름을 가져오십시오.

CTLIB 및 DBLIB 프로토콜과 함께 IGNORE_UDT 서버 옵션을 사용하여 페더레이티드 서버가 강조 입력이 없는 UDT의 기초가 되는 내장 유형을 판별해야 하는지 여부를 지정할 수 있습니다. 이 서버 옵션은 CTLIB 및 DBLIB 프로토콜을 통해 액세스한 데이터 소스에만 적용됩니다. 유효한 값은 다음과 같습니다.

'Y'
: UDT가 사용자 정의된 것이라는 사실을 무시하고 UDT의 기초가 되는 내장 유형을 결정합니다.

'N'
: UDT의 사용자 정의 스펙을 무시하지 않습니다. 이 값이 디폴트 설정입니다.

DB2가 별칭을 작성할 때, 별칭이 가리키는 오브젝트(테이블, 뷰, 스토어드 프로시저)에 대한 정보를 찾아 카탈로그합니다. 정보를 찾을 때, 일부 오브젝트에 인식되지 않는 데이터 유형(즉, 페더레이티드 서버의 상대 파트로 맵핑되지 않는 데이터 유형)이 있음을 발견할 수도 있습니다. 인식되지 않는 유형은 다음과 같습니다.

- 새 내장 유형
- 강조 입력이 있는 UDT

- 강조 입력이 없는 UDT. 이것은 사용자가 단지 이름을 바꾼 내장 유형입니다. 이러한 유형은 Sybase 및 Microsoft SQL Server와 같은 특정 데이터 소스에서만 지원됩니다.

페더레이티드 서버가 인식하지 못하는 데이터 유형을 찾으면, 오류 메시지 SQL3324N을 리턴합니다. 그러나, 예외가 있을 수 있습니다. CTLIB 또는 DBLIB 프로토콜을 통해 액세스할 수 있는 데이터 소스의 경우, IGNORE_UDT 서버 옵션을 설정하여 페더레이티드 데이터베이스가 인식할 수 없는 강조 입력이 없는 UDT를 발견할 때, 페더레이티드 데이터베이스가 UDT의 기초가 되는 내장 유형을 판별하도록 할 수 있습니다. 그런 다음, 페더레이티드 데이터베이스가 이 내장 유형을 인식하면 페더레이티드 데이터베이스가 내장 유형 정보를 카탈로그에 리턴합니다. 페더레이티드 데이터베이스가 강조 입력이 없는 UDT의 기초 내장 유형을 판별하도록 하려면, IGNORE_UDT를 'Y'로 설정하십시오.

8단계: CONNECTSTRING 서버 옵션 설정(선택적)

시간종료 임계값, 인터페이스 파일의 경로 및 이름, 그리고 인터페이스 파일의 패킷 크기를 지정하십시오. Sybase Open Client는 시간종료 임계값을 사용하여 너무 오랫동안 수행되는 응답 및 쿼리를 인터럽트합니다. CREATE SERVER OPTION DDL문에서 CONNECTSTRING 옵션을 사용하여 DB2에서 이 임계값을 설정할 수 있습니다. CONNECTSTRING 옵션을 사용하여 다음 값을 지정하십시오.

- SQL 쿼리의 시간종료 지속기간
- 로그인 응답의 시간종료 지속기간
- 인터페이스 파일의 경로 및 이름
- 패킷 크기

```
      ┌─;──────────────────────────────┐
      ▼                                │
►►────┬──────────────────────────────┬─┴──────────────────────────────────►◄
      ├─TIMEOUT── = ──seconds────────┤
      ├─LOGIN_TIMEOUT── = ──seconds──┤
      ├─IFILE── = ──"string"─────────┤
      ├─PACKET_SIZE── = ──bytes──────┤
      └─;────────────────────────────┘
```

TIMEOUT

SQL문에 대해 DB2 Universal Database가 Sybase Open Client의 응답을

기다리는 초 수를 지정합니다. *seconds*의 값은 DB2 Universal Database 정수 범위의 양수입니다. 지정할 시간종료 값은 사용하는 랩퍼에 따라 다릅니다. Windows NT, AIX 및 Solaris 서버는 모두 DBLIB 랩퍼를 사용할 수 있습니다. DBLIB 랩퍼의 기본값은 0입니다. Windows NT, AIX 및 Solaris 서버에서 DBLIB의 기본값을 사용하면 DB2 Universal Database가 응답을 무한정 기다리게 됩니다.

LOGIN_TIMEOUT

DB2 Universal Database가 로그인 요청에 대해 Sybase Open Client의 응답을 기다리는 초 수를 지정합니다. 기본값은 TIMEOUT과 동일합니다.

IFILE

Sybase Open Client 인터페이스 파일의 경로 및 이름을 지정합니다. *문자열*로 지정된 경로에는 큰 따옴표를 해야 합니다. Windows NT 서버인 경우 기본값은 %DB2PATH%입니다. AIX 및 Solaris 서버인 경우 기본값은 DB2 Universal Database 인스턴스의 홈 디렉토리에서 `sqllib/interfaces`입니다.

PACKET_SIZE

인터페이스 파일의 패킷 *크기*를 바이트 단위로 지정합니다. 데이터 소스가 지정된 패킷 *크기*를 지원하지 않으면 연결에 실패합니다. 각 레코드가 매우 클 경우 패킷 *크기*를 늘리면(예를 들어, 큰 테이블에 행을 삽입할 때) 성능이 크게 향상됩니다. 바이트 *크기*는 숫자 값입니다. 자세한 정보는 Sybase 참조 매뉴얼을 참조하십시오.

예:

Windows NT 서버인 경우, 시간종료 값을 60초로 설정하고 인터페이스 파일을 `C:\etc\interfaces`로 설정하려면 다음을 사용하십시오.

```
CREATE SERVER OPTION connectstring FOR SERVER sybase1
SETTING 'TIMEOUT=60;LOGIN_TIMEOUT=5;IFILE="C:\etc\interfaces"'
```

AIX 및 Solaris 서버인 경우, 시간종료 값을 60초로 설정하고 인터페이스 파일을 `/etc/interfaces`로 설정하려면 다음을 사용하십시오.

```
CREATE SERVER OPTION connectstring FOR SERVER sybase1
SETTING 'TIMEOUT=60;PACKET_SIZE=4096;IFILE="/etc/interfaces"'
```

9단계: 사용자 맵핑 작성

페더레이티드 서버의 사용자 ID나 암호가 Sybase 데이터 소스의 사용자 ID나 암호와 다를 경우, 다음과 같이 CREATE USER MAPPING문을 사용하여 로컬 사용자 ID를 Sybase 데이터 소스에 정의된 사용자 ID 및 암호로 맵핑하십시오.

```
CREATE USER MAPPING FOR DB2USER SERVER SYBSERVER
OPTIONS ( REMOTE_AUTHID 'sybuser', REMOTE_PASSWORD 'day2night')
```

여기서,

DB2USER
: Sybase 데이터 소스에 정의된 사용자 ID로 맵핑할 로컬 사용자 ID입니다.

SYBSERVER
: CREATE SERVER문에서 정의한 Sybase 데이터 소스 이름입니다.

'sybuser'
: *DB2USER*를 맵핑할 Sybase 데이터 소스의 사용자 ID입니다. 이 값은 대소문자를 구별합니다.

'day2night'
: *'sybuser'*와 연관된 암호입니다. 이 값은 대소문자를 구별합니다.

추가 옵션에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

10단계: 테이블 및 뷰에 대한 별명 작성

Sybase 데이터 소스에 있는 각각의 뷰나 테이블에 대한 별명을 지정하십시오. Sybase 데이터 소스를 쿼리할 때 이 별명을 사용합니다. Sybase 별명은 대소문자를 구별합니다. 스키마 및 테이블 이름은 큰 따옴표로 묶여야 합니다. 다음은 CREATE NICKNAME문의 예입니다.

```
CREATE NICKNAME SYBSALES FOR SYBSERVER."salesdata"."europe"
```

여기서,

SYBSALES
: Sybase 테이블이나 뷰의 고유한 별명입니다.

SYBSERVER."salesdata"."europe"
: 세 부분으로 구성된 식별자인데 다음 형식을 따릅니다.

data_source_name."remote_schema_name"."remote_table_name"

별명을 작성할 각각의 테이블이나 뷰에 대해 이 단계를 반복하십시오. 별명을 작성할 때 DB2는 연결을 통해 데이터 소스 카탈로그를 쿼리합니다. 이 쿼리에서 데이터 소스와의 연결을 테스트합니다. 연결이 작동되지 않으면 오류 메시지가 나타납니다.

CREATE NICKNAME문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오. 일반적인 별명 및 데이터 유형 맵핑을 검증하는 데 대한 자세한 정보는 *DB2 관리 안내서*를 참조하십시오.

Sybase 코드 페이지 지정

이 단계는 DB2 페더레이티드 서버 및 Sybase 서버가 다른 코드 페이지에서 수행되고 있을 때만 필요합니다. DB2와 같은 코드 세트를 사용하는 데이터 소스는 변환할 필요가 없습니다. 다음 테이블은 일반적인 NLS(National Language Support) 코드 페이지에 동등한 Sybase 옵션을 제공합니다. 이들 동등한 값에 대응하여 Sybase 데이터 소스를 구성해야 하거나, 클라이언트 코드가 불일치를 발견하여 오류 플래그를 표시하거나 자체 의미를 사용하여 데이터를 맵핑할 수 있어야 합니다. 소스 코드 페이지에서 목표 코드 페이지로 변환하는 변환 테이블이 없으면 DB2는 오류 메시지를 표시합니다. 자세한 정보는 Sybase 문서를 참조하십시오.

표 2. Sybase 코드 페이지 옵션

코드 페이지	동등한 Sybase 옵션
850	cp850
897	sjis
	iso_1
912	iso_2
1089	iso_6
813	iso_7
916	iso_8
920	iso_9

ODBC를 사용하여 Microsoft SQL Server 데이터 소스에 액세스(새 장)

Microsoft SQL Server 데이터 소스를 DB2 페더레이티드 서버에 추가하기 전에 페더레이티드 서버에서 ODBC 드라이버를 설치하고 구성해야 합니다. ODBC 드라이버 설치 방법에 대한 구체적인 자세한 정보는 ODBC 드라이버에 포함된 문서에 나타난 설치 절차를 참조하십시오.

페더레이티드 서버를 설정하여 Microsoft SQL Server 데이터 소스에 저장된 데이터에 액세스하려면 다음을 수행해야 합니다.

1. 페더레이티드 서버에서 ODBC 드라이버를 설치하고 구성하십시오. ODBC 드라이버 설치에 관해 자세한 정보는 ODBC 드라이버에 포함된 문서에서 설치 절차를 참조하십시오.

   Windows NT 서버용 DB2의 경우,

   ODBC 장치 관리자를 사용하여 시스템 DSN을 구성하십시오. Windows ODBC 데이터 소스 관리자에서 SQL Server 드라이버를 지정하고 대화 상자를 통해 새 시스템 DSN을 추가하십시오. ″사용자가 제공한 로그인 ID 및 암호를 사용하여 SQL 서버 인증″을 지정하십시오.

   AIX 서버용 DB2의 경우,

   MERANT가 제공한 라이브러리의 스레드된 버전을 설치한 다음 MERANT 라이브러리 디렉토리를 LIBPATH의 첫 번째 항목으로 지정하고 `.odbc.ini` 파일을 설정하십시오.
2. DB2 Relational Connect 버전 7.2를 설치하십시오. 67 페이지의 『DB2 Relational Connect 설치』를 참조하십시오.
3. Microsoft SQL Server 데이터 소스를 페더레이티드 서버에 추가하십시오.
4. Microsoft SQL Server 코드 페이지를 지정하십시오(Windows NT에만 해당).

이 장에서는 3단계 및 4단계를 설명합니다.

이 장의 지침은 Windows NT 및 AIX 플랫폼에 적용됩니다. 플랫폼별 차이는 해당 부분에서 설명합니다.

Microsoft SQL Server 데이터 소스를 페더레이티드 서버에 추가

ODBC 드라이버 및 DB2 Relational Connect를 설치한 후, 다음 단계에 따라 Microsoft SQL Server 데이터 소스를 페더레이티드 서버에 추가하십시오.

1. 환경 변수를 설정하십시오(AIX만 해당).
2. 쉘 스크립트를 수행하십시오(AIX만 해당).
3. DB2_DJ_COMM 환경 변수를 설정하십시오(선택적)(AIX만 해당).
4. DB2 인스턴스를 재순환하십시오(AIX만 해당).
5. 랩퍼를 작성하십시오.
6. 서버를 작성하십시오.
7. 사용자 맵핑을 작성하십시오.
8. 테이블 및 뷰의 별명을 작성하십시오.
9. ODBC 추적을 얻으십시오(선택적).

각 단계는 다음에 자세히 설명되어 있습니다.

1단계: 환경 변수 설정(AIX만 해당)

`db2dj.ini` 파일을 수정하고 **db2set** 명령을 실행하여 데이터 소스 환경 변수를 설정하십시오. `db2dj.ini` 파일에는 Microsoft SQL Server 데이터 소스에 연결하는 구성 정보가 들어 있습니다. **db2set** 명령은 사용자 설정으로 DB2 프로파일 레지스트리를 갱신합니다.

파티션된 데이터베이스 시스템의 특정 인스턴스에서 모든 노드에 하나의 `db2dj.ini` 파일을 사용하거나 특정 인스턴스에서 하나 이상의 노드에 고유한 `db2dj.ini` 파일을 사용할 수 있습니다. 파티션되지 않은 데이터베이스 시스템에는 인스턴스별로 단 하나의 `db2dj.ini` 파일이 있습니다.

환경 변수를 설정하려면 다음을 수행하십시오.

1. `$HOME/sqllib/cfg/`에 있는 `db2dj.ini` 파일을 편집하고 다음 환경 변수를 설정하십시오.

   ```
   ODBCINI=$HOME/.odbc.ini
   DJX_ODBC_LIBRARY_PATH=<path to the Merant driver>/lib
   DB2ENVLIST=LIBPATH
   ```

db2set 명령을 실행하여 DB2 프로파일 레지스트리를 변경 내용으로 갱신하십시오. **db2set** 구문은 데이터베이스 시스템 구조에 따라 다릅니다.

- 파티션되지 않은 데이터베이스 시스템에서 db2dj.ini 파일을 사용하거나, 파티션된 데이터베이스 시스템에서 db2dj.ini 파일을 사용하고 있고, db2dj.ini 파일의 값을 현재 노드에만 적용하려면 다음 명령을 실행하십시오.

  ```
  db2set DB2_DJ_INI=<path to ini file>/db2dj.ini
  ```

- 파티션된 데이터베이스 시스템에서 db2dj.ini 파일을 사용하면서, db2dj.ini 파일의 값을 이 인스턴스 내의 모든 노드에 적용하려면 다음 명령을 실행하십시오.

  ```
  db2set -g DB2_DJ_INI=<path to ini file>/db2dj.ini
  ```

- 파티션된 데이터베이스 시스템에서 db2dj.ini 파일을 사용하면서, db2dj.ini 파일의 값을 특정 노드에 적용하려면 다음 명령을 실행하십시오.

  ```
  db2set -i INSTANCEX  3 DB2_DJ_INI=$HOME/sqllib/cfg/node3.ini
  ```

  여기서,

  INSTANCEX
  : 인스턴스 이름입니다.

  3
  : db2nodes.cfg 파일에 있는 노드 번호입니다.

  node3.ini
  : db2dj.ini 파일의 이름이 바뀐 수정된 버전입니다.

클라이언트 라이브러리에 대한 경로를 설정하려면, 다음 명령을 실행하십시오.

```
db2set DB2LIBPATH=<path to the Merant client library>
db2set DB2ENVLIST=LIBPATH
```

2단계: 쉘 스크립트 수행(AIX만 해당)

djxlink.sh 쉘 스크립트는 클라이언트 라이브러리를 랩퍼 라이브러리에 링크합니다. 쉘 스크립트를 수행하려면 다음 명령을 실행하십시오.

```
djxlink
```

3단계: DB2_DJ_COMM 환경 변수 설정(선택적)(AIX만 해당)

Microsoft SQL Server 데이터 소스에 액세스할 때 시간이 너무 많이 걸리면 데이터 소스에 액세스할 때가 아니라 페더레이티드 서버를 초기화할 때 랩퍼를 로드

하기 위해 DB2_DJ_COMM 환경 변수를 설정하여 성능을 개선할 수 있습니다. 다음과 같이 5단계에서 지정한 랩퍼에 해당하는 랩퍼 라이브러리를 포함하는 DB2_DJ_COMM 환경 변수를 설정하십시오.

Windows NT 서버용 DB2의 경우,

```
db2set DB2_DJ_COMM=djxmssql3.dll
```

AIX 서버용 DB2의 경우,

```
db2set DB2_DJ_COMM=libmssql3.a
```

등호(=) 앞 뒤에 공백이 없는지 확인하십시오.

랩퍼 라이브러리 이름에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

4단계: DB2 인스턴스 재순환(AIX만 해당)

프로그램에서 환경 변수를 설정하려면 DB2 인스턴스를 재순환하십시오. 인스턴스를 재순환하면 DB2 인스턴스를 새로 고쳐서 변경 내용을 적용하십시오. 다음 명령을 실행하여 DB2 인스턴스를 재순환하십시오.

```
db2stop
db2start
```

5단계: 랩퍼 작성

DB2 Universal Database에는 Microsoft SQL Server 데이터 소스에 액세스할 때 사용할 수 있는 랩퍼라는 두 개의 서로 다른 프로토콜이 있습니다. 랩퍼는 페더레이티드 서버가 데이터 소스와 통신하고 데이터 소스에서 데이터를 검색할 때 사용하는 메커니즘입니다. 사용하는 랩퍼는 DB2 Universal Database가 수행되는 플랫폼에 따라 달라집니다. 표 3을 참조하여 적절한 랩퍼를 선택하십시오.

표 3. ODBC 드라이버

ODBC 드라이버	플랫폼	랩퍼 이름
ODBC 3.0(또는 이상) 드라이버	Windows NT	DJXMSSQL3
MERANT DataDirect Connect ODBC 3.6 드라이버	AIX	MSSQLODBC3

CREATE WRAPPER문을 사용하여 Microsoft SQL Server 데이터 소스에 액세스할 때 사용할 랩퍼를 지정하십시오. 다음은 CREATE WRAPPER문의 예입니다.

```
CREATE WRAPPER DJXMSSQL3
```

여기서, DJXMSSQL3은 Windows NT용 DB2(ODBC 3.0 드라이버를 사용하여)에서 사용되는 기본 랩퍼 이름입니다. AIX 서버용 DB2가 있는 경우, MSSQLODBC3 랩퍼 이름을 지정합니다.

기본 랩퍼 이름을 선택한 이름으로 대체할 수 있습니다. 그러나 이 경우 CREATE WRAPPER문에 페더레이티드 서버 플랫폼용 랩퍼 라이브러리 이름 및 LIBRARY 매개변수도 포함되어야 합니다. 예를 들어, 다음과 같습니다.

Windows NT 서버용 DB2의 경우,

```
CREATE WRAPPER wrapper_name LIBRARY 'djxmssql3.dll'
```

여기서, *wrapper_name*은 랩퍼에 지정할 이름이며 'djxmssql3.dll'은 라이브러리 이름입니다.

AIX 서버용 DB2의 경우,

```
CREATE WRAPPER wrapper_name LIBRARY 'libmssql3.a'
```

여기서, *wrapper_name*은 랩퍼에 지정할 이름이며 'libdjxmssql.a'는 라이브러리 이름입니다.

랩퍼 라이브러리 이름에 대한 자세한 정보는 *DB2 SQL 참조서*의 CREATE WRAPPER문을 참조하십시오.

6단계: 서버 작성

CREATE SERVER문을 사용하여 연결할 Microsoft SQL Server 데이터 소스를 각각 정의하십시오. 예를 들어, 다음과 같습니다.

```
CREATE SERVER sqlserver TYPE MSSQLSERVER VERSION 7.0 WRAPPER djxmssql3
OPTIONS (NODE 'sqlnode', DBNAME 'database_name')
```

여기서,

sqlserver
: Microsoft SQL Server 서버에 할당할 이름입니다. 이 이름은 고유해야 합니다.

MSSQLSERVER
: 액세스를 구성할 데이터 소스 유형입니다.

7.0
: 액세스할 Microsoft SQL Server 버전입니다. DB2 Universal Database는 Microsoft SQL Server 버전 6.5 및 7.0을 지원합니다.

DJXMSSQL3
: CREATE WRAPPER문에서 정의한 랩퍼 이름입니다.

'sqlnode'
: 액세스할 Microsoft SQL Server의 Microsoft SQL Server 버전을 참조하는 시스템 DSN 이름입니다. 이 값은 대소문자를 구별합니다. DB2 Universal Database는 Microsoft SQL Server 버전 6.5 및 7.0을 지원합니다.

  CREATE SERVER문에는 노드 이름(시스템 DSN 이름)이 옵션으로 지정되었지만 Microsoft SQL Server 데이터 소스의 경우는 필수입니다. Windows에서, Windows ODBC 데이터 소스 관리자 도구의 시스템 DSN 탭에서 DSN을 얻으십시오. AIX에서 DB2 인스턴스 소유자 홈 디렉토리에 있는 .odbc.ini 파일에서 DSN을 얻으십시오.

  CREATE WRAPPER문에서 사용할 수 있는 추가 옵션은 *DB2 SQL 참조서*를 참조하십시오.

'database_name'
: 연결할 데이터베이스 이름입니다.

  CREATE SERVER문에는 데이터베이스 이름이 옵션으로 지정되었지만 Microsoft SQL Server 데이터 소스의 경우는 필수입니다.

7단계: 사용자 맵핑 작성

페더레이티드 서버의 사용자 ID나 암호가 Microsoft SQL Server 데이터 소스의 사용자 ID나 암호와 다르면 다음과 같이 CREATE USER MAPPING문을 사용하여 로컬 사용자 ID를 Microsoft SQL Server 데이터 소스에 정의된 사용자 ID 및 암호로 맵핑하십시오.

```
CREATE USER MAPPING FOR db2user SERVER server_name
OPTIONS (REMOTE_AUTHID 'mssqluser', REMOTE_PASSWORD 'day2night')
```

여기서,

db2user

Microsoft SQL Server 데이터 소스에 정의된 사용자 ID로 맵핑할 로컬 사용자 ID입니다.

server_name

CREATE SERVER문에서 정의한 서버 이름입니다.

'mssqluser'

*db2user*를 맵핑할 Microsoft SQL Server 데이터 소스의 로그인 ID입니다. 이 값은 대소문자를 구별합니다.

'day2night'

*'mssqluser'*와 연관된 암호입니다. 이 값은 대소문자를 구별합니다.

CREATE USER MAPPING문에서 사용할 수 있는 추가 옵션은 *DB2 SQL 참조서*를 참조하십시오.

8단계: 테이블 및 뷰에 대한 별명 작성

액세스할 Microsoft SQL Server 데이터 소스의 뷰나 테이블 각각의 별명을 지정하십시오. Microsoft SQL Server 데이터 소스를 쿼리할 때 이 별명을 사용합니다. CREATE NICKNAME문을 사용하여 별명을 지정하십시오. 별명은 대소문자를 구별합니다. 다음은 CREATE NICKNAME문의 예입니다.

```
CREATE NICKNAME mssqlsales FOR server_name.salesdata.europe
```

여기서,

mssqlsales

Microsoft SQL Server 테이블이나 뷰의 고유한 별명입니다.

server_name.salesdata.europe

세 부분으로 구성된 식별자인데 다음 형식을 따릅니다.

data_source_server_name.remote_schema_name.remote_table_name

별명의 *remote_schema_name* 및 *remote_table_name* 부분은 큰 따옴표를 사용하는 것이 좋습니다.

별명을 만들 때 DB2는 데이터 소스 시스템 카탈로그 테이블(Microsoft SQL Server에서는 시스템 테이블이라고 함)에 액세스하려고 합니다. 여기서 데이터 소스와 연결을 테스트합니다. 연결되지 않으면 오류 메시지가 나타납니다.

별명을 작성할 모든 데이터베이스 테이블 및 뷰에 대해 이 단계를 반복하십시오.

CREATE NICKNAME문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오. 일반적인 별명 및 데이터 유형 맵핑의 검증에 대한 자세한 정보는 *DB2 관리 안내서*를 참조하십시오.

9단계: ODBC 추적 얻기(선택적)

데이터 소스 액세스에 문제점이 있을 경우, ODBC 추적 정보를 얻어서 이 문제점을 분석하고 해결할 수 있습니다. ODBC 추적 작업을 적절하게 수행하려면 ODBC 데이터 소스 관리자가 제공하는 추적 도구를 사용하십시오. 추적을 활성화하면 시스템 성능에 영향을 주므로 문제점을 해결한 후에는 추적을 해제해야 합니다.

Microsoft SQL Server 코드 페이지 검토(Windows NT만 해당)

Microsoft SQL Server는 DB2 UDB와 많은 부분 공통되는 자국어 지원(NLS)의 코드 페이지 옵션을 지원합니다. DB2와 같은 코드 세트를 사용하는 데이터 소스는 변환할 필요가 없습니다. 테이블 3에는 DB2 Universal Database와 Microsoft SQL Server가 모두 지원하는 코드 페이지가 나와 있습니다.

표 4. DB2 UDB 및 Microsoft SQL Server 코드 페이지 옵션

코드 페이지	지원되는 언어
1252	ISO 문자 세트
850	다국어
437	영어
874	태국어
932	일본어
936	중국어
949	한국어
950	대만어
1250	중앙 유럽어
1251	시릴어
1253	그리스어

표 4. DB2 UDB 및 Microsoft SQL Server 코드 페이지 옵션 (계속)

코드 페이지	지원되는 언어
1254	터키어
1255	히브리어
1256	아랍어

DB2 페더레이티드 서버 및 Microsoft SQL Server가 서로 다른 자국어 지원(NLS) 코드 페이지에서 수행되는 경우, 동등한 값에 상응하도록 Microsoft SQL Server 데이터 소스를 구성하거나 클라이언트 코드가 불일치를 발견하여 오류 플래그를 표시하거나 자체 의미를 사용하여 데이터를 맵핑해야 합니다. 소스 코드 페이지에서 목표 코드 페이지로 변환하는 변환 테이블이 없으면 DB2는 오류 메시지를 표시합니다. 자세한 정보는 Microsoft SQL Server 문서를 참조하십시오.

Informix 데이터 소스에 액세스(새 장)

Informix 데이터 소스를 DB2 페더레이티드 서버에 추가하기 전에 페더레이티드 서버에서 Informix Client SDK 드라이버를 설치하고 구성해야 합니다. Client SDK 소프트웨어 설치에 대한 자세한 정보는 Informix 데이터베이스 소프트웨어에 포함된 문서에서 설치 절차를 참조하십시오. 설치의 일부로 Informix Client SDK 라이브러리가 포함되는지 확인합니다.

페더레이티드 서버를 설정하여 Informix 데이터 소스에 저장된 데이터에 액세스하려면 다음을 수행해야 합니다.

1. DB2 Relational Connect를 설치하십시오. 67 페이지의 『DB2 Relational Connect 설치』를 참조하십시오.
2. 가장 최근의 DB2 FixPak를 적용하십시오.
3. Informix 데이터 소스를 페더레이티드 서버에 추가하십시오.

이 장에서는 3단계를 설명합니다.

이 장의 지침은 AIX, Solaris 운영 환경 및 HP-UP 운영 체제에 적용됩니다. 운영 체제별 차이는 해당 부분에서 설명합니다.

Informix 데이터 소스를 페더레이티드 서버에 추가

Informix 데이터 소스를 페더레이티드 서버에 추가하려면 다음을 수행해야 합니다.

1. 환경 변수를 설정하고 프로파일 레지스트리를 갱신하십시오.
2. DB2를 Informix 클라이언트 소프트웨어에 링크하십시오.
3. DB2 인스턴스를 재순환하십시오.
4. Informix `sqlhosts` 파일을 작성하십시오.
5. 랩퍼를 작성하십시오.
6. DB2_DJ_COMM 환경 변수를 설정하십시오(선택적).
7. 서버를 작성하십시오.
8. 사용자 맵핑을 작성하십시오.
9. 테이블, 뷰 및 Informix 동의어에 대한 별명을 작성하십시오.

각 단계는 다음에 자세히 설명되어 있습니다.

1단계: 환경 변수 설정 및 프로파일 레지스트리 갱신

`db2dj.ini` 파일을 수정하고 **db2set** 명령을 실행하여 데이터 소스 환경 변수를 설정하십시오. `db2dj.ini` 파일에는 페더레이티드 서버에 설치된 Informix 클라이언트 소프트웨어의 구성 정보가 들어 있습니다. **db2set** 명령은 사용자 설정으로 DB2 프로파일 레지스트리를 갱신합니다.

파티션된 데이터베이스 시스템의 특정 인스턴스에서 모든 노드에 하나의 `db2dj.ini` 파일을 사용하거나 특정 인스턴스에서 하나 이상의 노드에 고유한 `db2dj.ini` 파일을 사용할 수 있습니다. 파티션되지 않은 데이터베이스 시스템에는 인스턴스별로 단 하나의 `db2dj.ini` 파일이 있습니다.

환경 변수를 설정하려면 다음을 수행하십시오.

1. `sqllib/cfg` 디렉토리에 있는 `db2dj.ini` 파일을 편집하고 다음 환경 변수를 설정하십시오.

   주: 이 파일이 시스템에 없다면 직접 작성하면 됩니다.

   INFORMIXDIR

NFORMIXDIR 환경 변수를 Informix Client SDK 소프트웨어가 설치되는 디렉토리 경로에 설정하십시오. 예를 들어, 다음과 같습니다.

```
INFORMIXDIR=/informix/csdk
```

INFORMIXSERVER

이 변수는 기본 Informix 서버의 이름을 식별합니다.

```
INFORMIXSERVER=inf93
```

주: Informix 랩퍼는 이 변수 값을 사용하지 않으나 Informix 클라이언트는 이 변수를 설정할 것을 요구합니다. 랩퍼는 **노드** 서버 옵션 값을 사용해서 액세스할 Informix 데이터베이스 서버를 지정합니다.

INFORMIXSQLHOSTS

Informix `sqlhosts` 파일($INFORMIXDIR/etc/sqlhosts)의 기본 경로를 사용하고 있다면 이 변수를 설정할 필요가 없습니다. 그러나 Informix `sqlhosts` 파일에 대해 기본 경로가 아닌 다른 경로를 사용하고 있다면, Informix `sqlhosts` 파일의 전체 경로 이름에 이 변수를 설정해 주어야 합니다. 예를 들어, 다음과 같습니다.

```
INFORMIXSQLHOSTS=/informix/csdk/etc/my_sqlhosts
```

2. DB2 인스턴스의 `.profile` 파일을 Informix 환경 변수로 갱신합니다. 이 작업은 각 변수를 설정 및 익스포트할 수 있도록 다음 명령을 발행하여 수행할 수 있습니다.

```
PATH=$INFORMIXDIR/bin:$PATH
export PATH

INFORMIXDIR=<informix_client_path>
export INFORMIXDIR
```

여기서, `informix_client_path`는 페더레이티드 서버에 Informix 클라이언트가 설치된 디렉토리 경로입니다. 만약 경로에 이름이 비어 있다면 경로를 큰따옴표(")로 묶으십시오.

3. 다음을 입력하여 DB2 인스턴스 `.profile`을 실행하십시오.

```
. .profile
```

4. **db2set** 명령을 실행하여 DB2 프로파일 레지스트리를 변경 내용으로 갱신하십시오. **db2set** 명령의 구문은 데이터베이스 시스템 구조에 따라 다릅니다. 이 단계는 다음 데이터베이스 시스템 구조에서 db2dj.ini 파일을 사용하는 경우에만 필요합니다.

   파티션되지 않은 데이터베이스 시스템에서 db2dj.ini 파일을 사용하거나 db2dj.ini 파일을 현재 노드에만 적용하려면 다음을 실행하십시오.

   ```
   db2set DB2_DJ_INI=sqllib/cfg/db2dj.ini
   ```

   주: 이 섹션의 경로 이름은 완전한 형태이어야 합니다. 예를 들어, my_home/my_instance/sqllib/cfg/db2dj.ini와 같습니다.

   파티션된 데이터베이스 시스템에서 db2dj.ini 파일을 사용하면서, db2dj.ini 파일의 값을 이 인스턴스의 모든 노드에 적용하려면 다음을 실행하십시오.

   ```
   db2set -g DB2_DJ_INI = sqllib/cfg/db2dj.ini
   ```

   파티션된 데이터베이스 시스템에서 db2dj.ini 파일을 사용하면서, db2dj.ini 파일의 값을 특정 노드에 적용하려면 다음을 실행하십시오.

   db2set -i *INSTANCEX 3* DB2_DJ_INI=sqllib/cfg/*node3.ini*

   여기서,

   INSTANCEX
   : 인스턴스 이름입니다.

   3
   : db2nodes.cfg 파일에 있는 노드 번호입니다.

   node3.ini
   : db2dj.ini 파일의 이름이 바뀐 수정된 버전입니다.

2단계: DB2를 Informix 클라이언트 소프트웨어에 링크

Informix 데이터 소스에 액세스하려면 DB2 페더레이티드 서버가 클라이언트 라이브러리에 링크 편집되어야 합니다. 링크 편집 프로세스는 페더레이티드 서버가 통신하는 각 데이터 소스의 랩퍼 라이브러리를 작성합니다. djxlinkInformix 스크립트를 실행하면 Informix 랩퍼 라이브러리가 작성됩니다. djxlinkInformix 스크립트를 실행하려면 다음을 입력하십시오.

```
djxlinkInformix
```

주:

`djxlinkInformix` 스크립트는 Informix 랩퍼 라이브러리만 작성합니다. 이와는 달리, djxlink 스크립트는 DB2 Universal Database가 지원하는 모든 데이터 소스(Oracle, Microsoft SQL Server 등)에 대해 랩퍼 라이브러리를 작성하려고 합니다. 일부 데이터 소스에 설치된 클라이언트 소프트웨어가 있는 경우, djxlink 스크립트를 발행할 때 각각의 누락된 데이터 소스에 대해 오류 메시지가 표시됩니다.

djxlinkInformix 및 djxlink 스크립트를 실행하려면 UNIX 시스템 관리자(root) 권한 부여가 필요합니다.

djxlinkInformix 및 djxlink 스크립트는 운영 체제에 따라 자세한 오류 및 경고 메시지를 특정 파일에 작성합니다. 예를 들어, AIX에서 djxlinkInformix 스크립트는 `/usr/lpp/db2_07_01/lib/djxlinkInformix.out`에 작성하며 djxlink 스크립트는 `/usr/lpp/db2_07_01/lib/djxlink.out`에 작성합니다.

djxlinkInformix 및 djxlink 스크립트는 운영 체제에 따라 랩퍼 라이브러리를 특정 디렉토리에서 작성합니다. 예를 들어, AIX에서 libinformix.a 랩퍼 라이브러리는 `/usr/lpp/db2_07_01/lib` 디렉토리에서 작성됩니다.

libinformix.a 랩퍼 라이브러리에서 사용권한을 작성한 후 인스턴스 소유자가 읽고 실행할 수 있는지 확인하십시오. DB2 인스턴스 소유자가 시스템 그룹에 없는 경우 libinformix.a 랩퍼 라이브러리에서의 사용권한은 -rwxr-xr-x root system...libinformix.a가 되어야 합니다.

3단계: DB2 인스턴스 재순환

프로그램에서 환경 변수를 설정하려면 DB2 인스턴스를 재순환하십시오. 인스턴스를 재순환하면 DB2 인스턴스를 새로 고쳐서 변경 내용을 적용하십시오.

다음 명령을 실행하여 DB2 인스턴스를 재순환 하십시오.

AIX, Solaris 운영 환경 및 HP-UX 서버용 DB2

```
db2stop
  db2start
```

4단계: Informix sqlhosts 파일 작성

이 파일은 각 Informix 데이터베이스 서버의 위치와 데이터베이스 서버의 연결 유형(프로토콜)을 지정합니다. 이 파일을 여러가지 방식으로 작성할 수 있습니다. Informix Connect 또는 Informix Client SDK는 Informix 서버에 연결되어 있는 다른 시스템으로부터 이 파일을 복사할 수 있습니다. 또한 DB2 서버에서 Informix Client SDK를 구성하여 sqlhosts 파일을 작성하는 Informix 서버에 연결할 수 있습니다.

sqlhosts 파일을 복사하거나 작성한 후에, DB2 인스턴스 소유자는 Informix dbaccess(DB2 서버에 있는 경우)를 사용하여 Informix 서버에 연결하고 쿼리해야 합니다. 이것은 Informix Client SDK로 작업하기 위해 DB2 Relational Connect를 구성하기 전에 Informix Client SDK를 Informix 서버에 연결할 수 있도록 설정합니다.

파일 설정에 대한 자세한 정보는 Informix 매뉴얼 *Administrators Guide for Informix Dynamic Server*를 참조하십시오.

경고:
`sqlhosts` 파일의 Informix 데이터베이스 서버 이름을 정의하지 않을 경우, Informix 데이터베이스 서버에 연결하려 할 때 오류가 발생합니다.

5단계: 랩퍼 작성

CREATE WRAPPER문을 사용하여 Informix 데이터 소스에 액세스할 때 사용할 랩퍼를 지정하십시오. 랩퍼는 페더레이티드 서버가 데이터 소스와 통신하고 데이터 소스에서 데이터를 검색할 때 사용하는 메커니즘입니다. 다음은 CREATE WRAPPER문의 예입니다.

```
CREATE WRAPPER informix
```

여기서, **informix**는 wrapper_name입니다. 즉, **informix** 는 Informix Client SDK 소프트웨어가 사용하는 기본 랩퍼 이름입니다.

기본 랩퍼 이름을 선택한 이름으로 대체할 수 있습니다. 그러나 이 경우 CREATE WRAPPER문에 페더레이티드 서버용 랩퍼 라이브러리 이름 및 LIBRARY 매개 변수를 포함해야 합니다. 랩퍼 라이브러리 이름에 대한 자세한 정보는 *DB2 SQL 참조서*에서 CREATE WRAPPER문을 참조하십시오.

다음은 Informix에 대한 랩퍼 라이브러리 이름입니다.

- libinformix.a(AIX)
- libinformix.so(Solaris 운영 환경)
- libinformix.sl(HP-UX)

6단계: DB2_DJ_COMM 환경 변수 설정

Informix 데이터 소스에 액세스할 때 성능을 개선하려면, 페더레이티드 서버에 있는 DB2_DJ_COMM 환경 변수를 설정하십시오. 이 변수는 페더레이티드 서버를 초기화할 때 랩퍼가 로드될 것인지 결정합니다. 앞 단계에서 지정한 랩퍼의 해당 랩퍼 라이브러리를 포함하도록 DB2_DJ_COMM 환경 변수를 설정하십시오. Korn 쉘 또는 Bourne 쉘 명령행 인터페이스를 사용하고 있다면, 다음 익스포트 명령을 사용하십시오.

AIX 서버용 DB2의 경우,

```
DB2_DJ_COMM='libinformix.a'
export DB2_DJ_COMM
```

Solaris 운영 환경용 DB2 서버의 경우,

```
DB2_DJ_COMM='libinformix.so'
export DB2_DJ_COMM
```

HP-UX용 DB2 서버의 경우,

```
DB2_DJ_COMM='libinformix.sl'
export DB2_DJ_COMM
```

등호(=) 앞 뒤에 공백이 없는지 확인하십시오.

C 쉘 명령행 인터페이스를 사용하고 있다면, 다음 명령을 사용해서 환경 변수를 설정하십시오.

```
setenv DB2_DJ_COMM 'libinformix.a'(AIX 서버용 DB2)
setenv DB2_DJ_COMM 'libinformix.so'(Solaris 운영 환경 서버용 DB2)
setenv DB2_DJ_COMM 'libinformix.sl'(HP-UX 서버용 DB2)
```

랩퍼 라이브러리 이름 및 DB2_DJ_COMM 환경 변수에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

7단계: 서버 작성

다음과 같이 CREATE SERVER문을 사용하여 액세스할 데이터 소스가 있는 각 Informix 서버를 정의하십시오. 이 명령문에 대한 구문은 다음과 같습니다.

```
CREATE SERVER server_name TYPE server_type VERSION server_version
WRAPPER wrapper_name
OPTIONS (NODE 'node_name', DBNAME 'database_name')
```

여기서,

server_name
: Informix 데이터베이스 서버에 할당할 이름입니다. 이 이름은 고유해야 하며 페더레이티드 데이터베이스에 정의된 다른 server_name과 중복되지 않아야 합니다. server_name은 페더레이티드 데이터베이스에 있는 테이블 스페이스의 이름과 같지 않아야 합니다.

TYPE server_type
: 액세스를 구성할 데이터 소스 유형을 지정합니다.

주: Informix 랩퍼의 경우에 server_type은 informix이어야 합니다.

VERSION server_version
: 액세스할 Informix 데이터베이스 서버의 버전입니다. 지원되는 Informix 버전은 5, 7, 8 및 9입니다.

WRAPPER wrapper_name
: CREATE WRAPPER문에서 지정한 이름입니다.

NODE 'node_name'
: **server_name**이 있는 노드의 이름입니다. **node_name**은 Informix sqlhosts 파일에 정의되어야 합니다(4단계 참조). 비록 **node_name**이 CREATE SERVER SQL 명령문의 옵션으로 지정되더라도 Informix 데이터 소스의 경우에는 필수입니다. 이 값은 대소문자를 구별합니다. 추가 옵션에 대한 정보는 *DB2 SQL 참조서*를 참조하십시오.

DBNAME 'database_name'
: 액세스할 Informix 데이터베이스 이름입니다.

다음은 CREATE SERVER문의 예입니다.

```
CREATE SERVER asia TYPE informix VERSION 9 WRAPPER informix
OPTIONS (NODE 'abc', DBNAME 'sales')
```

FOLD_ID 및 FOLD_PW 서버 옵션은 랩퍼가 Informix로 보내기 전에 사용자 ID 및 암호를 대문자 또는 소문자로 변환하는 데 영향을 줍니다. 다음은 FOLD_ID 및 FOLD_PW 서버 옵션을 사용하는 CREATE SERVER 명령문의 예제입니다.

```
CREATE SERVER asia TYPE informix VERSION 9 WRAPPER informix
OPTIONS (NODE 'abc', DBNAME 'sales', FOLD_ID 'U', FOLD_PW 'U')
```

8단계: 사용자 맵핑 작성

페더레이티드 서버의 사용자 ID 또는 암호가 Informix 데이터 소스의 사용자 ID 또는 암호와 다르면, 다음과 같이 CREATE USER MAPPING문을 사용하여 로컬 사용자 ID를 Informix 데이터 소스에 정의된 사용자 ID 및 암호로 맵핑하십시오.

```
CREATE USER MAPPING FOR local_userid SERVER server_name
OPTIONS (REMOTE_AUTHID 'remode_userid', REMOTE_PASSWORD 'remote_password')
```

여기서,

local_userid
: Informix 데이터 소스에 정의된 사용자 ID로 맵핑할 로컬 사용자 ID입니다.

SERVER server_name
: CREATE SERVER문에서 정의한 Informix 데이터 소스 이름입니다.

REMOTE_AUTHID 'remote_userid'
: **local_userid**를 맵핑할 Informix 데이터베이스 서버의 사용자 ID입니다. 이 값은 CREATE SERVER 명령문에서 FOLD_ID 서버 옵션을 'U' 또는 'L'로 설정하지 않으면 대소문자를 구분합니다.

REMOTE_PASSWORD 'remote_password'
: **remote_userid**와 연관된 암호입니다. 이 값은 CREATE SERVER 명령문에서 FOLD_PW 서버 옵션을 'U' 또는 'L'로 설정하지 않으면 대소문자를 구분합니다.

다음은 CREATE USER MAPPING문의 예입니다.

```
CREATE USER MAPPING FOR robert SERVER asia
OPTIONS (REMOTE_AUTHID 'bob', REMOTE_PASSWORD 'day2night')
```

DB2 특수 레지스터인 **USER**를 사용해서 **REMOTE_AUTHID** 사용자 옵션에 지정된 데이터 소스 권한 부여 ID에 CREATE USER MAPPING 명령문을 발행하여 사용자의 권한 부여 ID를 맵핑할 수 있습니다. 다음은 **USER** 특수 레지스터를 포함하는 CREATE USER MAPPING 명령문의 예제입니다.

```
CREATE USER MAPPING FOR USER SERVER asia
OPTIONS (REMOTE_AUTHID 'bob', REMOTE_PASSWORD 'day2night')
```

추가 옵션에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

9단계: 테이블, 뷰 및 Informix 동의어에 대한 별명 작성

Informix 데이터 소스에 있는 각 테이블, 뷰 또는 Informix 동의어에 별명을 지정하십시오. 별명 길이는 128자까지 입니다. Informix 데이터 소스를 쿼리할 때 이 별명을 사용합니다. DB2는 큰 따옴표(")로 묶지 않으면 서버, 스키마 및 테이블 이름을 대문자로 변환합니다. 다음은 CREATE NICKNAME문의 예입니다.

```
CREATE NICKNAME nickname FOR
server_name."remote_schema_name"."remote_table_name"
```

여기서,

nickname
: Informix 테이블, 뷰 또는 동의어를 식별하는 데 사용되는 고유한 별명입니다.

server_name."remote_schema_name"."remote_table_name"
: 리모트 오브젝트에 대해 세 부분으로 구성된 식별자입니다.

- **server_name**은 CREATE SERVER 명령문에서 Informix 데이터베이스 서버에 지정한 이름입니다.
- **remote_schema_name**은 테이블, 뷰 또는 동의어가 속한 리모트 스키마의 이름입니다.
- **remote_table_name**은 액세스할 리모트 테이블, 뷰 또는 동의어의 이름입니다.

다음은 CREATE NICKNAME문의 예입니다.

```
CREATE NICKNAME salesjapan FOR asia."salesdata"."japan"
```

별명을 작성할 각각의 테이블이나 뷰에 대해 이 단계를 반복하십시오. 별명을 작성할 때 DB2는 연결을 통해 데이터 소스 카탈로그를 쿼리합니다. 이 쿼리에서 데이터 소스와의 연결을 테스트합니다. 연결이 작동되지 않으면 오류 메시지가 나타납니다.

CREATE NICKNAME문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오. 일반적인 별명 및 데이터 유형 맵핑을 검증하는 데 대한 자세한 정보는 *DB2 관리 안내서*를 참조하십시오.

제 3 부 관리

관리 안내서

사용 가능한 갱신사항

관리 안내서는 FixPak 4에서 갱신되었습니다. 최신 PDF는 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다. 또한 갱신된 참조에 이러한 참고 내용을 추가해 놓았습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

관리 안내서: 계획

제 8 장 물리적 데이터베이스 설계

테이블 스페이스 설계 고려사항

데이터가 Place on Raid일 때 테이블 스페이스 성능 최적화

DB2_PARALLEL_IO: DB2_PARALLEL_IO는 정의된 하나 이상의 컨테이너를 통해 테이블 스페이스에 영향을 줍니다. 레지스트리 변수를 따로 설정하지 않으면 입출력 병렬 처리가 테이블 스페이스의 컨테이너 수와 같아집니다. 레지스트리 변수를 설정하면 입출력 병렬 처리가 Extent *크기*로 나눈 프리페치(Prefetch) *크기*와 같아집니다. 테이블 스페이스의 개별 컨테이너가 복수의 실제 디스크에서 스트립(Stripe)될 경우 레지스트리 변수를 설정할 수 있습니다.

예를 들어 테이블 스페이스에 두 개의 컨테이너가 있고 프리페치 *크기*가 Extent *크기*의 네 배인 경우를 가정해 봅시다. 만약 레지스트리 변수를 설정해 주지 않으면 이 테이블 스페이스에 대한 프리페치의 요청은 두 개의 요청으로(두 개의 Extent에 각각 요청) 나뉘어집니다. 프리페처(Prefetcher)가 기능을 수행할 수 있다고 가정할 경우, 두 프리페처는 병렬로 이 두 요청에 기초하여 작업을 수행할 수 있습니다. 레지스트리 변수가 설정되어 있는 경우, 테이블 스페이스에 대한 프리페치 요청은 네 개의 프리페처가 병렬로 요청을 처리할 수 있는 네 개의 요청(요청당 하나의 Extent)으로 나뉘어집니다.

이 예제에서 만약 각각의 컨테이너가 전용의 단일 디스크를 갖고 있는 경우 이 테이블 스페이스에 레지스트리 변수를 설정하게 되면 두 개의 프리페처가 한 번에 두 개의 디스크에 액세스하기 때문에 두 디스크 사이에 경합이 벌어집니다. 그러나 각각의 컨테이너가 다중 디스크에 스트립된 경우 레지스트리 변수를 설정하게 되면 잠재적으로 한 번에 네 개의 다른 디스크에 액세스할 수 있습니다.

파티션 키

『노드그룹 설계』 섹션의 『노드그룹 설계 고려사항』 서브섹션에서, 파티션 키를 정의할 경우의 고려사항을 기술하는 『파티션 키』 서브-서브섹션에서 DB2_UPDATE_PART_KEY=ON인 경우에 한해 다음 텍스트를 삭제해야 합니다.

주: DB2_UPDATE_PART_KEY=OFF에도 여전히 제한사항이 적용됩니다.

주: FixPak 3 이상에서 디폴트값은 OFF입니다.

- 테이블 행에 대해 파티션 키 컬럼 값을 갱신할 수 없습니다.
- 파티션 키 컬럼 값을 삭제하거나 삽입할 수만 있습니다.

부록 D. 릴리스 간의 비호환성

연결 해제 발견시 오류 SQL30081N이 리턴되지 않음

오류 SQL30081N을 점검하여 데이터베이스 서버로의 끊어진 연결을 발견하는 응용프로그램이 DB2 Universal Database 버전 6 이상으로 이주할 때 끊어진 데이터베이스 연결을 더 이상 발견하지 않습니다.

익스포트 유틸리티가 식별 속성을 제대로 처리하려면 FixPak 7 이상이 필요함

익스포트 유틸리티가 모든 식별 속성(예를 들어, minvalue, maxvalue, cycle, order, remarks)을 지원하려면, 최소 레벨인 FixPak 7에서 클라이언트와 서버를 실행해야 합니다. 클라이언트나 서버가 최소한 이 레벨에 있지 않으면, 익스포트 유틸리티가 여전히 기능은 하지만 속성을 해석할 수 없습니다.

부록 E. NLS(자국어 지원)

국가/지역 코드 및 코드 페이지 지원

지원 언어 및 코드 세트 테이블에서는 코드 페이지 5488은 GB 18030으로, 코드 페이지 1394는 ShiftJIS X0213이라고도 합니다.

UTF-8(코드 페이지 1208) 클라이언트의 비유니코드 데이터베이스로의 연결은 지원되지 않습니다.

임포트/익스포트/로드 고려사항 - 코드 페이지 1394와 5488 제한

로드 또는 임포트 유틸리티를 사용해서 코드 페이지 1394(ShiftJISX X213)와 5488(GB 18030)의 데이터를 유니코드 데이터베이스로 이동시킬 수 있습니다. 익스포트 유틸리티를 사용할 경우, 유니코드 데이터베이스에서 코드 페이지 1394 및 5488의 데이터 파일로 데이터를 이동할 수 있습니다.

유니코드 클라이언트와 유니코드 서버 간 연결만이 지원되는 경우, 유니코드 클라이언트를 사용하거나 로드, 임포트 또는 익스포트 유틸리티를 사용하기 전에 먼저 DB2 레지스트리 변수인 DB2CODEPAGE를 1208로 설정해야 합니다.

코드 페이지 1394나 5488을 유니코드로 변환하는 일은 결과적으로 바이트 수를 확장하는 것입니다. 예를 들어, 2바이트 문자를 GRAPHIC 컬럼의 두 16비트 유니코드 문자로 저장할 수 있습니다. 유니코드 데이터베이스의 목표 컬럼이 확장된 유니코드 바이트를 포함할 수 있을 만큼 충분한지 확인할 필요가 있습니다.

날짜 시간 값

날짜 시간 값의 문자열 표현

데이터 유형이 DATE, TIME 또는 TIMESTAMP인 값은 SQL 서버 사용자가 볼 수 있는 내부 형식으로 표시됩니다. 그러나, 날짜, 시간, 시간소인 또한 문자열로 표시할 수 있으며, 이러한 표현은 SQL 사용자에게 직접 전달됩니다. 데이터 유형이 DATE, TIME 또는 TIMESTAMP인 상수 또는 변수가 없기 때문입니다. 따라서, 검색을 위해서는 날짜 시간 값을 문자열 변수에 지정해야 합니다. 프로그램이 프리컴파일되거나 데이터베이스에 바인드될 때 DATETIME 형식 옵션을 지정하여 대체하지 않는 한 문자열 표현은 일반적으로 응용프로그램의 국가/지역 코드와 연관된 날짜 시간 값의 기본 형식입니다.

날짜 시간 값의 올바른 문자열 표현이 내부 날짜 시간 값을 가진 조작에 사용된 경우, 문자열 표현은 조작이 수행되기 전에 내부 날짜, 시간, 시간소인 형태로 변환됩니다. 다음 절에서는 날짜 시간 값의 올바른 문자열 표현에 대해 설명합니다.

주: 날짜 시간 값의 그래픽 문자열 표현은 유니코드 데이터베이스에서만 지원됩니다.

날짜 문자열

날짜의 문자열 표현은 숫자로 시작하며 길이가 적어도 8자 이상인 문자열입니다. 뒤 공백을 넣을 수 있습니다. 날짜의 월 부분과 일 부분에서 앞의 0은 생략됩니다.

"날짜 문자열 표현 형식" 테이블은 변경되지 않습니다.

시간 문자열

시간의 문자열 표현은 숫자로 시작하며 길이가 적어도 4자 이상인 문자열입니다. 뒤 공백을 넣을 수 있습니다. 시간의 시 부분에서 앞의 0은 생략되며 초는 완전 생략할 수 있습니다. 초를 생략하기로 한 경우, 내부 스펙이 0초로 간주됩니다. 따라서 `13:30`은 `13:30:00`과 같습니다.

"시간 문자열 표현 형식" 테이블은 변경되지 않습니다.

시간소인 문자열

시간소인의 문자열 표현은 숫자로 시작하며 길이가 적어도 16자 이상인 문자열입니다. 시간소인의 완전한 문자열 표현 형식은 *yyyy-mm-dd-hh.mm.ss.nnnnnn*입니다. 뒤 공백을 넣을 수 있습니다. 시간소인의 월, 일, 시 부분에서 앞의 0은 생략되며 마이크로초는 자르거나 완전 생략할 수 있습니다. 마이크로초의 숫자 부분을 생략하기로 한 경우, 내부 스펙이 0초로 간주됩니다. 따라서 `1991-3-2-8.30.00`은 `1991-03-02-08.30.00.000000`과 같습니다.

문자 세트 고려사항

날짜 및 시간소인 문자열에는 숫자와 분리문자 기호만 포함되어야 합니다.

날짜 및 시간 형식

날짜 및 시간 형식의 문자열 표현은 응용프로그램의 국가/지역 코드와 연관된 날짜 시간 값의 기본 형식입니다. 프로그램이 프리컴파일되거나 데이터베이스에 바인드될 때 DATETIME 형식 옵션을 지정하여 이 기본 형식을 대체할 수 있습니다.

관리 안내서: 구현

Windows 2000 이상 시스템에 DMS 컨테이너를 지정하기 위한 새로운 방법

현재 DB2는 새로운 방법을 사용하여 Windows 2000 이상 시스템에 DMS 원시 테이블 스페이스 컨테이너를 지정합니다. 각 기본 디스크 파티션이나 동적 볼륨에는 작성시 GUID(globally unique unique identifier)가 지정됩니다. 테이블 스페이스 정의에 컨테이너를 지정할 때 이 GUID를 디바이스 식별자로 사용할 수 있습니다. GUID는 시스템에 고유하기 때문에, 디스크 파티션 정의가 같더라도 멀티 노드 구성은 각 노드마다 고유한 GUID를 가집니다.

`db2listvolumes.exe`라는 도구가 Windows 시스템에 정의된 모든 디스크 볼륨에 대해 GUID를 표시하도록 제공됩니다. 이 도구는 해당 도구가 실행되는 현재 디렉토리에 두 파일을 작성합니다. 그 중 하나인 `volumes.xml` 파일에는 각 디스크 볼륨에 대한 정보가 있습니다. 이 파일은 XML을 지원하는 모든 브라우저에서 쉽게 볼 수 있도록 설계되었습니다. 다른 파일인 `tablespace.ddl`에는 테이블 스페이스 컨테이너를 지정하는 데 필요한 구문이 있습니다. `tablespace.ddl`을 사용하기 전에, 테이블 스페이스 정의에 필요한 나머지 정보를 포함하도록 이 파일을 갱신해야 합니다.

db2listvolumes 도구에는 명령행 인수가 필요하지 않습니다.

제어 센터 확장 예제

제어 센터 확장 부록에 표시된 예는 올바르지 않으며 작동하지 않습니다. 대신 Java 예제로 작성하려면 다음 정보를 참조하십시오.

샘플 프로그램 PluginEx.java는 `samples/java` 서브디렉토리에 있습니다. PluginEx.java는 DB2 Application Development 클라이언트와 함께 설치됩니다. PluginEx.java를 컴파일하려면 *classpath*에 다음을 넣어야 합니다.

- Windows 플랫폼에서는 다음을 사용하십시오.

- DRIVE: \sqllib\java\swingall.jar
  - DRIVE: \sqllib\cc\com.jar
  - DRIVE: \sqllib\cc

  여기서, DRIVE는 DB2가 설치된 드라이브를 나타냅니다.
- UNIX 플랫폼에서는 다음을 사용하십시오.
  - /u/db2inst1/sqllib/java/swingall.jar
  - /u/db2inst1/sqllib/cc

  여기서, /u/db2inst1 DB2가 설치된 디렉토리를 나타냅니다.

PluginEx.java를 컴파일하여 생성된 모든 클래스를 포함하도록 db2plug.zip을 작성하십시오. 파일을 압축해서는 안됩니다. 예를 들어, 다음을 실행하십시오.

```
zip -r0 db2plug.zip PluginEx*.class
```

이 명령은 모든 클래스 파일을 db2plug.zip 파일에 배치하여 상대 경로 정보를 보존합니다.

PluginEx.java 파일의 지시사항에 따라 예제를 컴파일 및 실행하십시오.

CCObject 인터페이스에는 관리 안내서의 제어 센터 확장 부록에 나열된 상수보다 더 정적인 상수가 포함됩니다. 다음은 제어 센터 확장을 위한 Java 인터페이스 (CCExtension, CCObject, CCM enuAction, CCToolBarAction)입니다. 이 인터페이스는 참조용으로 여기에 제시되었습니다.

CCExtension::

```
//  Licensed Materials -- Property of IBM
//
//  (c) Copyright International Business Machines Corporation, 1999.
//      All Rights Reserved.
//
//  US Government Users Restricted Rights -
//  Use, duplication or disclosure restricted by
//  GSA ADP Schedule Contract with IBM Corp.
//

package com.ibm.db2.tools.cc.navigator;

/**
 * The CCExtension interface allows users to extend the Control Center user
```

```
 * interface by adding new toolbar buttons, new menu items and
 * remove some predefined set of existing menu actions.
 *
 * To do so, create a java file which imports the
 * com.ibm.db2.tools.cc.navigator package and implements this interface.
 * The new file provides the implementation of the getObjects() and
 * getToolbarActions() function.
 *
 * The getObjects() function returns an array of CCObjects which defines
 * the existing
 * objects which the user would like to add new menu actions or remove
 * the alter or configure menu actions.
 *
 * The getToolbarActions() function returns an array of CCToolbarActions
 * which is added to the Control Center main toolbar.
 *
 * A single CCExtension subclass file or multiple CCExtension subclass
 * files can be used to define the Control Center extensions.  In order
 * for the Control Center to make use of these extensions, use the
 * following setup procedures:
 * (1) Create a "db2plug.zip" file which contains all the CCExtension
 *     subclass files.  The files should not be compressed. For example,
 *     if the CCExtension files are in the plugin package and they are
 *     located in the plugin directory, issue
 *        zip -r0 db2plug.zip plugin\*.class
 *     This command will put all the plugin package class files into the
 *     db2plug.zip file and preserve their relative path information.
 * (2) To run WEBCC as an applet, put the db2plug.zip file in where the
 *     <codebase> tag points to in the WEBCC html file.
 *     To run the Control Center as an application, put
 *     the db2plug.zip in a directory pointed to by the CLASSPATH
 *     envirnoment variable and where the Control Center is run.
 *
 * For browsers that support multiple archives, just add "db2plug.zip"
 * to the archive list of the WEBCC html page. Otherwise, all the
 * CCExtension, CCObject, CCToolbarAction, CCMenuAction subclass files
 * will have to be in their relative path depending on which package
 * they belong to.
 */

public interface CCExtension
{
   /**
    * Get an array of CCObject subclass objects which define
    * a list of objects to be overrided in the
    * Control Center
    * @return CCObject[] CCObject subclass objects array
    */
   public CCObject[] getObjects();

   /**
    * Get an array of CCToolbarAction subclass objects which represent
    * a list of buttons to be added to the Control Center
    * main toolbar.
```

```
     * @return CCToolbarAction[] CCToolbarAction subclass objects array
     */
    public CCToolbarAction[] getToolbarActions();
}
```

CCObject:

```
CCObject:
//
//  Licensed Materials -- Property of IBM
//
//  (c) Copyright International Business Machines Corporation, 1999.
//      All Rights Reserved.
//
//  US Government Users Restricted Rights -
//  Use, duplication or disclosure restricted by
//  GSA ADP Schedule Contract with IBM Corp.
//

package com.ibm.db2.tools.cc.navigator;

/**
 * The CCObject interface allows users to define a new object to be
 * inserted into the Control Center tree or changing the behavior of the
 * menu actions of an existing object.
 */
public interface CCObject
{
   /**
    * The following static constants defines a list of object type
    * available to be added to the Control Center tree.
    */
   public static final int UDB_SYSTEMS_FOLDER                          = 0;
   public static final int UDB_SYSTEM                                  = 1;
   public static final int UDB_INSTANCES_FOLDER                        = 2;
   public static final int UDB_INSTANCE                                = 3;
   public static final int UDB_DATABASES_FOLDER                        = 4;
   public static final int UDB_DATABASE                                = 5;
   public static final int UDB_TABLES_FOLDER                           = 6;
   public static final int UDB_TABLE                                   = 7;
   public static final int UDB_TABLESPACES_FOLDER                      = 8;
   public static final int UDB_TABLESPACE                              = 9;
   public static final int UDB_VIEWS_FOLDER                            = 10;
   public static final int UDB_VIEW                                    = 11;
   public static final int UDB_ALIASES_FOLDER                          = 12;
   public static final int UDB_ALIAS                                   = 13;
   public static final int UDB_TRIGGERS_FOLDER                         = 14;
   public static final int UDB_TRIGGER                                 = 15;
   public static final int UDB_SCHEMAS_FOLDER                          = 16;
   public static final int UDB_SCHEMA                                  = 17;
   public static final int UDB_INDEXES_FOLDER                          = 18;
   public static final int UDB_INDEX                                   = 19;
   public static final int UDB_CONNECTIONS_FOLDER                      = 20;
```

```
public static final int UDB_CONNECTION                              = 21;
public static final int UDB_REPLICATION_SOURCES_FOLDER              = 22;
public static final int UDB_REPLICATION_SOURCE                      = 23;
public static final int UDB_REPLICATION_SUBSCRIPTIONS_FOLDER        = 24;
public static final int UDB_REPLICATION_SUBSCRIPTION                = 25;
public static final int UDB_BUFFERPOOLS_FOLDER                      = 26;
public static final int UDB_BUFFERPOOL                              = 27;
public static final int UDB_APPLICATION_OBJECTS_FOLDER              = 28;
public static final int UDB_USER_DEFINED_DISTINCT_DATATYPES_FOLDER  = 29;
public static final int UDB_USER_DEFINED_DISTINCT_DATATYPE          = 30;
public static final int UDB_USER_DEFINED_DISTINCT_FUNCTIONS_FOLDER  = 31;
public static final int UDB_USER_DEFINED_DISTINCT_FUNCTION          = 32;
public static final int UDB_PACKAGES_FOLDER                         = 33;
public static final int UDB_PACKAGE                                 = 34;
public static final int UDB_STORE_PROCEDURES_FOLDER                 = 35;
public static final int UDB_STORE_PROCEDURE                         = 36;
public static final int UDB_USER_AND_GROUP_OBJECTS_FOLDER           = 37;
public static final int UDB_DB_USERS_FOLDER                         = 38;
public static final int UDB_DB_USER                                 = 39;
public static final int UDB_DB_GROUPS_FOLDER                        = 40;
public static final int UDB_DB_GROUP                                = 41;
public static final int UDB_DRDA_TABLES_FOLDER                      = 42;
public static final int UDB_DRDA_TABLE                              = 43;
public static final int UDB_NODEGROUPS_FOLDER                       = 44;
public static final int UDB_NODEGROUP                               = 45;

public static final int S390_SUBSYSTEMS_FOLDER                      = 46;
public static final int S390_SUBSYSTEM                              = 47;
public static final int S390_BUFFERPOOLS_FOLDER                     = 48;
public static final int S390_BUFFERPOOL                             = 49;
public static final int S390_VIEWS_FOLDER                           = 50;
public static final int S390_VIEW                                   = 51;
public static final int S390_DATABASES_FOLDER                       = 52;
public static final int S390_DATABASE                               = 53;
public static final int S390_TABLESPACES_FOLDER                     = 54;
public static final int S390_TABLESPACE                             = 55;
public static final int S390_TABLES_FOLDER                          = 56;
public static final int S390_TABLE                                  = 57;
public static final int S390_INDEXS_FOLDER                          = 58;
public static final int S390_INDEX                                  = 59;
public static final int S390_STORAGE_GROUPS_FOLDER                  = 60;
public static final int S390_STORAGE_GROUP                          = 61;
public static final int S390_ALIASES_FOLDER                         = 62;
public static final int S390_ALIAS                                  = 63;
public static final int S390_SYNONYMS_FOLDER                        = 64;
public static final int S390_SYNONYM                                = 65;
public static final int S390_APPLICATION_OBJECTS_FOLDER             = 66;
public static final int S390_COLLECTIONS_FOLDER                     = 67;
public static final int S390_COLLECTION                             = 68;
public static final int S390_PACKAGES_FOLDER                        = 69;
public static final int S390_PACKAGE                                = 70;
public static final int S390_PLANS_FOLDER                           = 71;
public static final int S390_PLAN                                   = 72;
public static final int S390_PROCEDURES_FOLDER                      = 73;
```

```
public static final int S390_PROCEDURE                          = 74;
public static final int S390_DB_USERS_FOLDER                    = 75;
public static final int S390_DB_USER                            = 76;
public static final int S390_LOCATIONS_FOLDER                   = 77;
public static final int S390_LOCATION                           = 78;
public static final int S390_DISTINCT_TYPES_FOLDER              = 79;
public static final int S390_DISTINCT_TYPE                      = 80;
public static final int S390_USER_DEFINED_FUNCTIONS_FOLDER      = 81;
public static final int S390_USER_DEFINED_FUNCTION              = 82;
public static final int S390_TRIGGERS_FOLDER                    = 83;
public static final int S390_TRIGGER                            = 84;
public static final int S390_SCHEMAS_FOLDER                     = 85;
public static final int S390_SCHEMA                             = 86;
public static final int S390_CATALOG_TABLES_FOLDER              = 87;
public static final int S390_CATALOG_TABLE                      = 88;
public static final int DCS_GATEWAY_CONNECTIONS_FOLDER          = 89;
public static final int DCS_GATEWAY_CONNECTION                  = 90;
public static final int S390_UTILITY_OBJECTS_FOLDER             = 91;
public static final int S390_DATASET_TEMPLATES_FOLDER           = 92;
public static final int S390_DATASET_TEMPLATE                   = 93;
public static final int S390_UTILITY_LISTS_FOLDER               = 94;
public static final int S390_UTILITY_LIST                       = 95;
public static final int S390_UTILITY_PROCEDURES_FOLDER          = 96;
public static final int S390_UTILITY_PROCEDURE                  = 97;
/**
 * Total number of object types
 */
public static final int NUM_OBJECT_TYPES                        = 98;

/**
 * Get the name of these object
 *
 * The function returns the name of this object. This name
 * can be of three types:
 * (1) Fully qualified name
 *     Syntax: xxxxx-yyyyy-zzzzz
 *             where xxxxx-yyyyy is the fully quality name of the parent
 *             object and zzzzz is the name of the new object.
 *     Note: Parent and child object name is separated by '-' character.
 *     If a schema name is required to identify object, the fully
 *     qualified name is represented by xxxxx-yyyyy-wwwww.zzzzz
 *     where wwwww is the schema name.
 *     Only the behavior of the object that match this fully
 *     quality name will be affected.
 * (2) Parent fully qualified name
 *     Syntax: xxxxx-yyyyy
 *             where xxxxx-yyyyy is the fully qualified name of the
 *             parent object.
 *     When the object type is folder (ie. DATABASES_FOLDER), the
 *     getName() should only return the fully qualified name of the
 *     folder's parent.
 *     Only the behavior of the object that match this name
 *     and the specific type return by the getType() function will be
 *     affected.
```

```
 * (3) null
 *     Syntax: null
 *     If null is return, the CCMenuActions returns by the
 *     getMenuActions() call will be applied to all objects of type
 *     returns by the getType() call.
 * @return String object name
 */
public String getName();

/**
 * Get the type of this object
 * @return int return one of the static type constants defined in this
 * interface
 */
public int getType();

/**
 * Get the CCMenu Action array which defines the list of menu actions
 *  to be created for object
 * return CCMenuAction[] CCMenuAction array
 */
public CCMenuAction[] getMenuActions();

/**
 * Check if this object is editable.  If not, the Alter related menu
 * items will be removed from the object's popup menu
 * return boolean If false, the Alter menu item will be remove from the
 * object's popup menu.
 * Return true if you do not wish to modify current Alter menu item
 * behaviour.
 */
public boolean isEditable();

/**
 * Check if this object is configurable.  If not, the configuration
 * related menu items will be removed from the object's popup menu
 * return boolean If false, the Configuration related menu item will be
 *  removed from the object's popup menu.
 * Return true if you do not wish to modify current Configuration
 * behaviour.
 */
public boolean isConfigurable();
}
```

CCMenuAction::

```
//
//  Licensed Materials -- Property of IBM
//
//  (c) Copyright International Business Machines Corporation, 1999.
//      All Rights Reserved.
//
```

```
//  US Government Users Restricted Rights -
//  Use, duplication or disclosure restricted by
//  GSA ADP Schedule Contract with IBM Corp.
//

package com.ibm.db2.tools.cc.navigator;
import java.awt.event.*;
import javax.swing.*;

/**
 * The CCMenuAction class allows users to define a new menu item to be added
 * to a Control Center object.  The new menu item will be added at the end of
 * an object's popup menu.
 *
 * Note: If the object has a Control Center Refresh and/or
 * Filter menu item, the new menu item will be inserted before the Refresh
 * and Filter menu. The Control Center Refresh and Filter menu items are
 * always  at the end of the popup menu.
 */
public interface CCMenuAction
{
   /**
    * Get the name of this action
    * @return String Name text on the menu item
    */
   public String getMenuText();

   /**
    * Invoked when an action occurs.
    * @param e Action event
    */
   public void actionPerformed(ActionEvent e);
}
```

CCToolBarAction:

```
//  Licensed Materials -- Property of IBM
//
//  (c) Copyright International Business Machines Corporation, 1999.
//      All Rights Reserved.
//
//  US Government Users Restricted Rights -
//  Use, duplication or disclosure restricted by
//  GSA ADP Schedule Contract with IBM Corp.
//

package com.ibm.db2.tools.cc.navigator;
import java.awt.event.*;
import javax.swing.*;

/**
 * The CCToolbarAction interface class allows users to define a new action
```

```
 *  to be added to the Control Center toolbar.
 */
public interface CCToolbarAction
{
   /**
    * Get the name of this action
    * @return String Name text on the menu item, or toolbar button hover help
    */
   public String getHoverHelpText();

   /**
    * Get the icon for the toolbar button
    * Any toolbar CCAction should override this function and return
    * a valid ImageIcon object. Otherwise, the button will have no icon.
    * @return ImageIcon Icon to be displayed
    */
   public ImageIcon getIcon();

   /**
    * Invoked when an action occurs.
    * @param e Action event
    */
   public void actionPerformed(ActionEvent e);
}
```

관리 안내서: 성능

시스템 임시 테이블 스키마

시스템 임시 테이블의 스키마는 이를 작성하는 권한 부여 ID와 응용프로그램에 의해 결정됩니다. 이 데이터가 사용 가능한 경우, 테이블을 작성한 스키마는 <AUTHID><APPLID>입니다. 일부 상황에서는 스키마를 결정하는 이 ID 중 하나만을 사용하여 테이블을 작성합니다. 그리고 상황에 따라서는 종종 테이블 작성에 어떠한 ID도 사용하지 않습니다. 결과적으로는 AUTHID.TEMPTABLENAME 또는 .TEMPTABLENAME 등의 테이블이 될 수 있습니다. GET SNAPSHOT 명령을 사용하여 테이블의 스키마 정보를 살펴볼 수 있습니다. 이 명령에 대한 자세한 정보는 Command Reference를 참조하십시오.

제 8 장 조작 성능

블록 기반 버퍼 풀

이 기능은 Sun Solaris 운영 환경에서만 지원됩니다.

입출력 오버헤드를 감안할 때 디스크에서 페이지를 프리페치(Prefetch)하는 것은 과잉 조작입니다. DB2의 프리페치 기능은 입출력이 처리를 무시할 수 있을 때 처리량을 눈에 띄게 향상시킬 수 있습니다. 대부분의 플랫폼은 디스크의 연속 페이지에서 메모리의 불연속 페이지로 읽어갈 수 있는 높은 성능을 제공합니다. 이러한 특성을 일반적으로 『분산된 읽기』 또는 『벡터화된 I/O』라고 합니다. 일부 플랫폼의 경우, 이런 성능상의 특성은 블록 크기가 큰 입출력을 실행해서는 달성할 수 없습니다. 기본적으로 버퍼 풀이 페이지의 기본이 됩니다. 즉, 디스크의 연속 페이지는 메모리의 불연속 페이지로 프리페치됩니다. 성능 프리페치는 디스크에서 버퍼 풀의 연속 페이지로 페이지를 읽어들일 수 있는 플랫폼에서 더 확장할 수 있습니다. 레지스트리 변수인 DB2_BLOCK_BASED_BP를 이용하면 연속 페이지 세트를 보유하는 버퍼 풀에서 섹션을 작성할 수 있습니다. 이러한 연속 페이지 세트를 『블록』이라고 합니다. 이 레지스트리 변수를 설정함으로써 순차 프리페치는 개별적으

로 각 페이지를 읽는 대신 디스크에서 직접 블록으로 페이지를 읽어들이게 됩니다. 이 결과 입출력 성능이 향상됩니다. 이 레지스트리 변수에 대한 자세한 정보는 관리 안내서의 '레지스트리 및 환경 변수' 절을 참조하십시오.

Extent의 크기가 서로 다른 다중 테이블 스페이스를 동일 블록 크기의 버퍼 풀에 바운드할 수 있습니다. 비록 둘다 개별 개념을 다루는 것이 사실이나 Extent 크기와 블록 크기 사이에는 매우 밀접한 관계가 있습니다. Extent는 테이블 스페이스가 다중 컨테이너에서 스트립되는 입도입니다. 블록은 순차 프리페치 요청을 처리하는 입출력 서버가 블록 기반의 입출력을 처리하는 것을 고려하는 입도입니다.

개별 순차 프리페치 요청은 Extent 크기 단위의 페이지를 사용합니다. 이와 같은 프리페치 요청이 수신되면 입출력 서버는 분산된 읽기 방법을 사용하여 페이지 기반의 입출력이 아닌 블록 기반의 입출력(버퍼 풀에 블록 기반 영역이 있는 경우)으로 각 요청을 처리할 때의 비용과 이점을 결정합니다. 블록 기반 입출력으로 입출력을 처리하면 연속 디스크에서 연속 메모리로 읽을 수 있다는 성능상의 이점이 있습니다. 비용이란 메소드 사용으로 발생하는 버려진 버퍼 풀 메모리의 용량을 의미합니다.

블록 기반의 입출력을 처리할 때 다음의 두 가지 이유로 버퍼 풀 메모리가 낭비될 수 있습니다.

- 블록 페이지 수보다 적은 프리페치 요청 페이지 수. 즉, Extent 크기가 블록 크기보다 작습니다.
- 프리페치 요청의 일부로 요청된 일부 페이지가 이미 버퍼 풀 페이지 영역에 있습니다.

주: 버퍼 풀 블록 기반 영역의 각 블록은 하위 블록으로 나눌 수 없습니다. 블록 내의 페이지는 모두 연속적이야 합니다. 이에 따라 공간이 낭비될 가능성이 있습니다.

입출력 서버는 블록 기반의 입출력을 처리함으로써 생기는 이점을 얻기 위해 각 블록 내에서 어느 정도 페이지가 낭비되는 것을 허용합니다. 그러나 너무 많은 블록이 낭비되는 경우, 입출력 서버는 페이지 기반 프리페칭 사용을 버퍼 풀의 페이지 영역으로 되돌려 놓습니다. 이에 따라 프리페칭(prefetching)하는 동안 처리된 일부 입출력은 블록에 기반하지 않게 됩니다. 이것은 적절한 조건이 아닙니다.

최상의 성능을 얻기 위해서는 동일한 Extent 크기를 가진 테이블 스페이스가 블록 크기가 같은 버퍼 풀에 바운드되어야 합니다. 일부 테이블 스페이스의 Extent 크기가 바운드되는 버퍼 풀의 블록 크기보다 큰 경우에는 성능이 향상될 가능성이 여전히 있습니다. Extent 크기가 블록 크기보다 작을 때에는 테이블 스페이스를 버퍼 풀에 바인드할 수 없습니다.

주: 버퍼 풀의 블록 영역은 순차 프리페칭에만 사용됩니다. 시스템에 관련된 순차 프리페치가 없거나 거의 없는 경우에는 블록 영역이 버퍼 풀의 버려진 영역이 됩니다.

AWE 및 블록 기반의 지원을 한꺼번에 모두 버퍼 풀용으로 설정할 수는 없습니다. DB2_AWE 및 DB2_BLOCK_BASED_BP 레지스트리 변수가 동일한 버퍼 풀을 참조하는 경우에는 AWE에 우선권이 주어집니다. 블록 기반 지원은 이 경우 사용할 수 없으며 AWE를 사용할 수 없을 때에만 재사용할 수 있습니다.

확장 저장영역을 사용하는 버퍼 풀은 블록 기반의 입출력을 지원하지 않습니다.

블록 기반 버퍼 풀의 예

예제를 다루기 전에 먼저 시스템의 버퍼 풀 ID를 알고 있어야 합니다. 버퍼 풀의 ID는 BUFFERPOOLID 컬럼이나 SYSCAT.BUFFERPOOLS 시스템 카탈로그 뷰에서 확인할 수 있습니다.

시나리오 1

1000 페이지로 구성된, ID가 4인 버퍼 풀을 사용합니다. 이제 블록마다 32 페이지로 구성된 700 페이지짜리 블록 영역을 작성하려고 합니다. 다음을 수행해야 합니다.

```
db2set DB2_BLOCK_BASED_BP=4,700,32
```

데이터베이스가 시작되면 ID가 4인 버퍼 풀이 672 페이지로 구성된 블록 영역 및 328 페이지로 구성된 페이지 영역으로 작성됩니다. 이 예에서 심지어 700을 32로 나눌 수 없습니다. 이것은 지정된 블록 영역 크기가 다음 공식을 사용하여 가장 근사한 블록 크기 경계가 되어야 한다는 것을 의미합니다.

```
      ((블록 영역 크기))
  FLOOR(-----------------) X 블록 크기
      (  (블록 크기)     )
      (      700         )
= FLOOR(-----------------) X 32
      (       32         )
= 21 x 32
= 672
```

시나리오 2

3000 페이지로 구성된, ID가 11인 버퍼 풀을 사용합니다. 이제 2700 페이지로 구성된 블록 영역을 작성하려고 합니다. 다음을 수행해야 합니다.

```
db2set DB2_BLOCK_BASED_BP=11,2700
```

데이터베이스가 시작되면 ID가 11인 버퍼 풀이 2688 페이지로 구성된 블록 영역 및 312 페이지로 구성된 페이지 영역으로 작성됩니다. 블록 *크기*에 명시적으로 주어진 값도 없이 기본값 32가 그대로 사용됩니다. 이 예에서는 심지어 2700을 32로 나눌 수 없습니다. 이것은 지정된 블록 영역 *크기*가 다음 공식을 사용하여 가장 근사한 블록 *크기* 경계가 되어야 한다는 것을 의미합니다.

```
      ((블록 영역 크기))
  FLOOR(-----------------) X 블록 크기
      (  (블록 크기)     )
      (      2700        )
= FLOOR(-----------------) X 32
      (       32         )
= 84 x 32
= 2688
```

제 10 장 프로세서 추가를 통한 구성 스케일링

파티션 데이터베이스에 노드 추가시 문제점

페이지 *크기*가 기본 페이지 *크기*(4KB)와 다른 하나 이상의 시스템 임시 테이블스페이스가 있는 파티션 데이터베이스에 노드를 추가하면, 오류 메시지 『SQL6073N 노드 추가 조작이 실패했습니다』와 SQLCODE가 나타납니다. 이러한 상태는 노드를 작성할 때 페이지 *크기*가 4KB인 IBMDEFAULTBP 버퍼 풀만이 있기 때문에 발생합니다.

예를 들어, **db2start** 명령을 사용하여 현재 파티션 데이터베이스에 노드를 추가할 수 있습니다.

```
DB2START NODENUM 2 ADDNODE HOSTNAME newhost PORT 2
```

파티션 데이터베이스에 기본 페이지 크기의 시스템 임시 테이블 스페이스가 있는 경우, 다음의 메시지가 리턴됩니다.

```
SQL6075W 데이터베이스 관리 프로그램 시작 조작이 정상적으로 노드를 추가하였습니다.
모든 노드가 중지되고 다시 시작될 때까지 노드가 활성화되지 않습니다.
```

그러나, 파티션 데이터베이스에 기본 페이지 크기가 아닌 시스템 임시 테이블 스페이스가 있는 경우, 다음 메시지가 리턴됩니다.

```
SQL6073N 노드 추가 조작이 실패했습니다. SQLCODE = "<-902>"
```

이와 유사한 예로, 새로운 노드 설명으로 db2nodes.cfg 파일을 수동 갱신한 후에 ADD NODE 명령을 사용할 수 있습니다. 파일을 편집하고 기본 페이지 크기의 시스템 임시 테이블 스페이스가 있는 파티션 데이터베이스로 ADD NODE 명령을 수행하면 다음의 메시지가 리턴됩니다.

```
DB20000I ADD NODE 명령이 정상적으로 완료되었습니다.
```

그러나, 파티션 데이터베이스에 기본 페이지 크기가 아닌 시스템 임시 테이블 스페이스가 있는 경우, 다음 메시지가 리턴됩니다.

```
SQL6073N 노드 추가 조작이 실패했습니다. SQLCODE = "<-902>"
```

이와 같은 문제점을 방지하기 위한 한 가지 방법은 다음을 수행하는 것입니다.

```
DB2SET DB2_HIDDENBP=16
```

단, **db2start** 또는 ADD NODE 명령을 발행하기 전에 이를 수행해야 합니다. 이 레지스트리 변수는 DB2가 각각 기본값과 다른 페이지 크기를 사용하여 16 페이지의 숨겨진 버퍼 풀을 할당할 수 있도록 합니다. 따라서, ADD NODE 조작이 성공적으로 완료됩니다.

이들 문제점을 방지하기 위한 다른 방법은 ADD NODE 또는 **db2start** 명령에 WITHOUT TABLESPACES 절을 지정하는 것입니다. 이와 같이 한 후에

CREATE BUFFERPOOL 명령문을 사용하여 버퍼 풀을 작성하고 ALTER TABLESPACE 명령문을 사용하여 시스템 임시 테이블 스페이스를 버퍼 풀에 연관시켜야 합니다.

페이지 크기가 기본 페이지 크기(4KB)와 다른 하나 이상의 테이블 스페이스가 있는 기존의 노드 그룹에 노드를 추가하면,『SQL0647N "" 버퍼 풀이 현재 활성화되지 않았습니다.』라는 오류 메시지가 표시됩니다. 이러한 상태는 새로운 노드에서 작성된 기본값이 아닌 페이지 크기 버퍼 풀이 테이블 스페이스에 대해 활성화되지 않았기 때문에 발생합니다.

예를 들어, ALTER NODEGROUP 명령문을 사용하여 노드 그룹에 노드를 추가할 수 있습니다.

```
DB2START
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2)
```

노드 그룹에 기본 페이지 크기의 테이블 스페이스가 있는 경우, 다음의 메시지가 리턴됩니다.

```
  SQL1759W  추가된 일부 노드를 포함시키거나 삭제된 일부 노드를 제외시켜
노드 그룹 "<ng1>"의 오브젝트 데이터 파티션을 변경하려면
노드 그룹 재분배가 필요합니다.
```

그러나, 노드 그룹에 기본 페이지 크기가 아닌 테이블 스페이스가 있는 경우, 리턴되는 메시지는 다음과 같습니다.

```
SQL0647N "" 버퍼 풀이 현재 활성화되지 않았습니다.
```

이 문제점을 방지하기 위한 한 가지 방법은 ALTER NODEGROUP 명령문을 발행하기 전에 각 페이지 크기에 대해 버퍼 풀을 작성하고 데이터베이스에 다시 연결하는 것입니다.

```
DB2START
CONNECT TO mpp1
CREATE BUFFERPOOL bp1 SIZE 1000 PAGESIZE 8192
CONNECT RESET
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2)
```

이 문제점을 방지하기 위한 두 번째 방법은 다음을 수행하는 것입니다.

```
DB2SET DB2_HIDDENBP=16
```

단, **db2start** 명령과 CONNECT 및 ALTER NODEGROUP 명령문을 발행하기 전에 이를 수행하십시오.

ALTER TABLESPACE 명령문을 사용하여 노드에 테이블 스페이스를 추가하는 경우에는 다른 문제점이 발생할 수 있습니다. 예를 들어, 다음과 같습니다.

```
DB2START
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2) WITHOUT TABLESPACES
ALTER TABLESPACE ts1 ADD ('ts1') ON NODE (2)
```

이 일련의 명령과 명령문은 오류 메시지 SQL0647N(예상하는 메시지 SQL1759W가 아님)을 생성합니다.

이 변경을 올바르게 완료하려면, ALTER NODEGROUP... WITHOUT TABLESPACES 명령문 이후에 데이터베이스에 다시 연결해야 합니다.

```
DB2START
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2) WITHOUT TABLESPACES
CONNECT RESET
CONNECT TO mpp1
ALTER TABLESPACE ts1 ADD ('ts1') ON NODE (2)
```

이 문제점을 방지하기 위한 다른 방법은 다음을 수행하는 것입니다.

```
DB2SET DB2_HIDDENBP=16
```

단, **db2start** 명령과 CONNECT, ALTER NODEGROUP 및 ALTER TABLESPACE 명령문을 발행하기 전에 이를 수행하십시오.

13장. DB2 구성

이제 로그 아카이브 완료가 보다 자주 점검됨

불필요한 로그 아카이브 요청을 예방하여 복구 시간을 개선하기 위해 데이터베이스 서버는 이제 새 로그 파일이 작성될 때와 첫 번째 아카이브 로그가 변경될 때 로그 아카이브 완료를 점검합니다.

DB2 레지스트리 및 환경 변수

성능 변수 정정

표 5. 성능 변수

변수 이름	운영 체제	값
설명		
DB2_BINSORT	All	기본값=NO 값: YES 또는 NO
CPU 시간 및 정렬 경과 시간을 줄이는 새로운 정렬 알고리즘을 사용할 수 있도록 합니다. 이 새 알고리즘은 DB2 UDB의 매우 효과적인 정수 정렬 기술을 BIGINT, CHAR, VARCHAR, FLOAT, DECIMAL과 같은 모든 정렬 데이터 유형 뿐 아니라 이러한 데이터 유형 조합까지 확장합니다. 이러한 새 알고리즘을 사용할 수 있도록 하려면 다음 명령을 사용하십시오. `db2set DB2_BINSORT = yes`		
DB2_BLOCK_BASED_BP	Solaris 운영 환경	기본값=없음 값: 매개변수에 종속됨
버퍼 풀에서 블록 영역을 작성하는데 필요한 값을 지정합니다. 버퍼 풀의 ID가 필요합니다. 버퍼 풀의 ID는 BUFFERPOOLID 컬럼이나 SYSCAT.BUFFERPOOLS 시스템 카탈로그 뷰에서 확인할 수 있습니다. 버퍼 풀에서 블록 기반의 입출력에 할당할 페이지 수가 주어져야 합니다. 블록에 포함할 페이지의 수는 선택적이며, 기본값은 32 입니다. 이 레지스트리 변수의 사용 형식은 다음과 같습니다. `DB2_BLOCK_BASED_BP=BUFFER POOL ID,BLOCK AREA SIZE,[BLOCK SIZE];...` 항목을 분리하는 세미콜론과 함께 동일 변수를 사용하여 다중 버퍼 풀을 블록 기반으로 정의할 수 있습니다. BLOCK SIZE 값은 2 - 256 범위가 될 수 있습니다. BLOCK SIZE가 주어지지 않을 경우, 기본값으로 32가 사용됩니다. 지정된 BLOCK AREA SIZE가 총 버퍼 풀 크기의 98%보다 클 경우, 버퍼 풀은 블록 기반이 되지 않습니다. 버퍼 풀의 페이지 기반 영역에 버퍼 풀의 일정 부분을 보유하는 것이 좋습니다. 비록 대부분의 시스템 입출력이 순차 프리페치라고 하더라도 개별 페이지가 요구될 수 있기 때문입니다. BLOCK AREA SIZE에 지정한 값이 BLOCK SIZE의 배수가 아닌 경우에는 가장 근사치의 블록 크기 경계가 값이 됩니다. 블록 기반 I/O에 대한 자세한 정보는 123 페이지의 『블록 기반 버퍼 풀』을 참조하십시오.		
DB2_NO_FORK_CHECK	UNIX	기본값=OFF 값: ON 또는 OFF

표 5. 성능 변수 (계속)

<table>
<tr><th>변수 이름</th><th>운영 체제</th><th>값</th></tr>
<tr><th colspan="3">설명</th></tr>
<tr><td colspan="3">이 변수가『ON』이면, 클라이언트 프로세스는 실행될 프로세스 사본을 작성하는 응용프로그램에 대해 자체를 보호하지 않습니다(포킹(forking)이라고 함). 포킹이 발생할 경우, 결과를 예측할 수 없습니다. 영향 없음에서부터 일부 잘못된 결과, 일부 오류 코드 리턴 및 응용프로그램에서의 트랩에 이르는 결과 범위를 가질 수 있습니다. 응용프로그램이 포크(fork)하지 않으며 더 나은 성능을 원하는 것이 확실한 경우, 이 변수의 값을『ON』으로 변경해야 합니다.</td></tr>
<tr><td>DB2_MINIMIZE_LIST_PREFETCH</td><td>All</td><td>기본값=NO

값: YES 또는 NO</td></tr>
<tr><td colspan="3">목록 프리페치는 인덱스에서 규정 RID 검색, 페이지 번호 순으로 정렬, 데이터 페이지 프리페치와 관련된 특수한 테이블 액세스 방법입니다.

때로 최적화 알고리즘은 목록 프리페치가 적절한 액세스 방법인지 판별하기 위한 정확한 정보를 가지고 있지 않습니다. 술어 선택성에 최적화 알고리즘이 카탈로그 통계를 사용하여 선택성을 판별하지 못하도록 하는 매개변수 표시기 또는 호스트 변수가 들어 있는 경우 이러한 현상이 발생합니다.

이 레지스트리 변수는 최적화 알고리즘이 그러한 상황에서 목록 프리페치를 고려하지 않도록 합니다.</td></tr>
<tr><td>DB2_INLIST_TO_NLJN</td><td>All</td><td>기본값=NO

값: YES 또는 NO</td></tr>
<tr><td colspan="3">어떤 경우, SQL 컴파일러는 조인할 IN 목록 술어를 재작성할 수 있습니다. 예를 들어, 다음 쿼리와 같습니다.
<pre>
   SELECT *
    FROM EMPLOYEE
    WHERE DEPTNO IN ('D11', 'D21', 'E21')
</pre>
다음과 같이 작성할 수 있습니다.
<pre>
   SELECT *
    FROM EMPLOYEE, (VALUES 'D11', 'D21', 'E21) AS V(DNO)
    WHERE DEPTNO = V.DNO
</pre>
DEPTNO에 대한 인덱스가 있는 경우, 이와 같이 수정해도 성능이 좋아지지 않습니다. 값 목록에 먼저 액세스하여 조인 술어에 적용할 인덱스를 사용하여 EMPLOYEE를 중첩 루프 조인에 조인하십시오.

때로 최적화 알고리즘은 재작성된 쿼리 버전의 최적 조인 방법을 판별하기 위한 정확한 정보를 가지고 있지 않습니다. IN 목록에 최적화 알고리즘이 카탈로그 통계를 사용하여 선택성을 판별하지 못하도록 하는 매개변수 표시기 또는 호스트 변수가 들어 있는 경우 이러한 현상이 발생합니다. 이 레지스트리 변수는 최적화 알고리즘이 IN 목록을 조인의 내부 테이블로 포함시키는 테이블을 사용하여 중첩 루프 조인을 활용하여 값 목록을 조인하도록 합니다.</td></tr>
</table>

레지스트리 변수 DB2BPVARS의 새 매개변수

레지스트리 변수 DB2BPVARS는 두 개의 새 매개변수인 NUMPREFETCHQUEUES와 PREFETCHQUEUESIZE를 지원합니다. 이 매개변수는 모든 플랫폼에 적용되고 버퍼 풀 데이터의 프리페칭(prefetching)을 향상시키는 데 사용할 수 있습니다. 예를 들어, 원하는 PREFETCHSIZE가 PREFETCHSIZE/EXTENTSIZE 프리페치 리퀘스트로 나뉘는 순차 프리페칭을 고려합니다. 이 경우, 리퀘스트는 비동기 입출력을 수행하기 위해 입출력 서버가 디스패치되는 프리페치 큐에 놓입니다. 디폴트로, DB2는 각 데이터베이스 파티션에 대해 한 큐의 크기인 max(100, 2*NUM_IOSERVERS)를 유지보수합니다. 일부 환경에서는 더 많은 큐, 서로 다른 크기의 큐 또는 둘다의 성능이 향상됩니다. 프리페치 큐의 수는 최대한 입출력 서버 수의 절반이어야 합니다. 이러한 매개변수를 설정할 때, 현재 사용자 수와 같은 워크로드 특성뿐 아니라, PREFETCHSIZE, EXTENTSIZE, NUM_IOSERVERS, 버퍼 풀 크기 및 DB2_BLOCK_BASED_BP와 같은 다른 매개변수를 고려하십시오.

디폴트값이 환경에 비해 너무 작다고 생각되면, 먼저 값을 조금만 늘리십시오. 예를 들어, NUMPREFETCHQUEUES=4 및 PREFETCHQUEUESIZE=200을 설정할 수 있습니다. 이 매개변수를 제어된 방식으로 변경하여 변경의 효과를 모니터하고 평가할 수 있습니다.

표 6. 새 매개변수 요약

매개변수 이름	디폴트값	유효한 범위
NUMPREFETCHQUEUES	1	1 - NUM_IOSERVERS 1보다 작은 수로 설정되면, 1로 조정됨 NUM_IOSERVERS보다 큰 수로 설정되면 NUM_IOSERVERS로 조정됨
PREFETCHQUEUESIZE	max(100, 2*NUM_IOSERVERS)	1 - 32767 1보다 작은 수로 설정되면, 디폴트값으로 조정됨 32767보다 큰 수로 설정되면, 32767로 조정됨

기타 레지스트리 변수에 대한 정정 및 추가

DB2_NEWLOGPATH2 레지스트리 변수는 모든 운영 체제에 대해 사용 가능합니다. 새 변수(DB2_ROLLFORWARD_NORETRIEVE)가 추가되었습니다. 두 변수에 대한 올바른 정보는 다음과 같습니다.

표 7. 기타 변수

변수 이름 설명	운영 체제	값
DB2_NEWLOGPATH2	ALL	기본값=NO 값: YES 또는 NO
이 매개변수를 사용하여 이중 로그를 구현하는 데 2차 경로를 사용할지 여부를 지정할 수 있습니다. 사용되는 경로는 *logpath* 데이터베이스 구성 매개변수의 현재 값에 『2』를 추가하여 생성됩니다.		
DB2_ROLLFORWARD_NORETRIEVE	ALL	기본값=(설정되지 않음) 값: YES 또는 NO
데이터베이스 구성 매개변수 USEREXIT가 사용 가능한 경우, 로그 파일은 롤 포워드 조작 중 아카이브에서 자동으로 검색됩니다. DB2_ROLLFORWARD_NORETRIEVE 변수는 롤 포워드 조작이 아카이브에서 로그 파일을 검색하지 않도록 지정하게 합니다. 이 변수를 기본값으로 사용할 수 없습니다. 자동으로 로그 파일을 검색하기 위해 롤 포워드하지 않으려는 경우 이 변수를 YES로 설정하십시오. 예를 들어, 잘못된 응용프로그램이 작성한 로그 레코드가 백업 시스템을 손상시키지 않게 하려고 할 때 긴급 대기(hot-standby) 설정에서 변수를 YES로 설정하십시오.		

일반 레지스트리 변수에 대한 정정 및 추가

새 변수(DB2_REDUCED_OPTIMIZATION)가 추가되었습니다.

표 8. 일반 레지스트리 변수

변수 이름 설명	운영 체제	값
DB2_REDUCED_OPTIMIZATION	ALL	기본값=NO 값: YES, NO 또는 임의의 정수

표 8. 일반 레지스트리 변수 (계속)

변수 이름	운영 체제	값
설명		
이 레지스트리 변수는 특정 최적화 레벨에서 사용되는 일부 최적화 기술을 사용하지 않게 합니다. 사용되는 최적화 기술의 수를 감소시키는 경우, 최적화하는 동안 시간 및 자원을 감소시키십시오. **주:** 최적화 시간 및 자원 사용은 감소시킬 수 있지만, 최적에 미치지 못하는 데이터 액세스 플랜을 생성할 위험이 증가됩니다. • NO로 설정하는 경우 최적화 알고리즘은 최적화 기술을 변경하지 않습니다. • YES로 설정하는 경우 최적화 레벨이 5(기본값)이거나 더 낮은 경우, 중요한 준비 시간 및 자원을 소모할 수 있지만 보통 더 나은 액세스 플랜을 생성하지 않는 일부 최적화 기술을 최적화 알고리즘이 사용하지 않게 합니다. 최적화 레벨이 정확히 5인 경우, 최적화 알고리즘은 일부 추가 기술을 줄이거나 사용하지 않음으로써 최적화 시간 및 자원 사용을 더 줄일 수 있지만, 최적에 미치지 못하는 액세스 플랜의 위험을 증가시킬 수 있습니다. 최적화 레벨이 5 미만인 경우, 일부 이러한 기술은 어떤 경우에는 시행되지 않을 수 있습니다. 그러나 이러한 경우에도 해당 기술이 시행됩니다. • 정수로 설정하는 경우 값이 YES로 설정된 경우와 결과는 같지만, 5 레벨에서 최적화되어 동적으로 준비된 쿼리에 대해 다음과 같은 추가적인 동작이 있습니다. 쿼리 블록에서 총 조인 수가 설정을 초과하면, 최적화 알고리즘은 최적화 레벨 5에 대해 위에서 설명한 대로 추가적인 최적화 기술을 사용하지 않는 대신 그리디(greedy) 조인 열거로 전환하여, 최적화 레벨 2와 유사한 레벨에서 쿼리가 최적화됨을 나타냅니다. 그리디(greedy) 및 동적 조인 열거에 대한 정보는 *관리 안내서: 성능*에서 "최적 조인 선택을 위한 전략 검색"을 참조하십시오. *관리 안내서: 성능*의 "최적화 클래스 조정"에서 설명한 대로 DB2_REDUCED_OPTIMIZATION이 YES로 설정되었을 때 최적화 레벨 5에서 동적 최적화의 감소는 최적화 레벨 5에 대해 설명한 동작보다 우선하며 정수 설정에 대해 설명한 동작보다도 우선합니다.		

위성 관리 안내 및 참조서

버전 7.2 DB2 Personal Edition 및 DB2 Workgroup Edition을 위성으로 설정

다음 섹션에서는 Windows 기반 버전 7.2 DB2 Personal Edition 및 DB2 Workgroup Edition 시스템을 위성 환경에서 모든 기능을 발휘하는 위성으로 사용하도록 설정하는 방법을 설명합니다. 이 설명에서 사용되는 용어 및 개념에 대한 자세한 정보는 *위성 관리 안내 및 참조서*에서 참조하십시오. URL `http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v6pubs.d2w/en_main`에 이 책이 있습니다.

*위성 관리 안내 및 참조서*의 정보를 보충하는 기술 정보는 다음 URL을 참조하십시오.
http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/index.d2w/report

전제조건

DB2 Personal Edition 또는 DB2 Workgroup Edition을 위성으로 설정하려면 다음이 필요합니다.

1. DB2 제어 서버

   DB2 제어 서버는 Windows NT 또는 AIX에서 실행되는 DB2 Enterprise Edition 시스템으로 Control Server 구성요소가 설치되어 있습니다. 사용하는 DB2 Enterprise Edition 시스템은 FixPak 2 이상이 있는 버전 6이거나 FixPak이 있는 버전 7이어야 합니다.

   - DB2 제어 서버로 사용하려는 버전 6 Enterprise Edition 시스템이 있는 경우, 140 페이지의 『버전 6 Enterprise Edition 시스템에 FixPak 2 이상 설치』에서 참조하십시오.
   - 버전 7을 사용하며 제어 서버 구성요소가 설치되어 있지 않은 경우, 이 구성요소를 설치하고 이미 설치되어 있는 FixPak을 다시 설치한 후, DB2 제

어 서버 인스턴스 및 위성 제어 데이터베이스를 작성하십시오. 이들 오브젝트 작성에 대한 지침은 *위성 관리 안내 및 참조서*에서 참조하십시오.

주: 버전 7.2 Enterprise Edition 시스템을 DB2 제어 서버로 사용하기 위해 Windows NT에 설치하고 응답 파일 설치를 수행하려는 경우, 응답 파일에 지정된 키워드에 대한 자세한 정보는 *DB2 제어 서버 응답 파일 키워드*라는 제목의 기술 정보를 참조하십시오.

2. DB2 제어 서버 인스턴스 및 위성 제어 데이터베이스

   DB2 제어 서버 인스턴스는 일반적으로 DB2CTLSV라고 하며 위성 제어 데이터베이스는 SATCTLDB라고 합니다. DB2 제어 서버 인스턴스 및 위성 제어 데이터베이스는 Enterprise Edition 시스템에 있으며 Windows NT에서는 제어 서버 구성요소와 함께 DB2를 설치할 때 자동으로 작성됩니다. AIX에 DB2를 설치할 경우, DB2 제어 서버 및 위성 제어 데이터베이스 작성에 대한 자세한 내용은 *위성 관리 안내 및 참조서*에서 참조하십시오.

3. 위성 관리 센터

   위성 관리 센터는 위성 환경을 설정하고 관리할 때 사용하는 GUI 도구의 세트입니다. 제어 센터에서 이 도구 세트에 액세스합니다. 위성 관리 센터 및 위성 환경에 대한 자세한 정보는 *위성 관리 안내 및 참조서* 및 위성 관리 센터에서 제공되는 온라인 도움말을 참조하십시오. 버전 6 제어 센터를 수행하고 있는 경우, 150 페이지의『버전 6 제어 센터 및 위성 관리 센터 업그레이드』에서 참조하십시오.

   위성 환경을 설정하고 위성 관리 센터의 새 위성을 나타내는 오브젝트를 작성하기 위해 위성 관리 센터를 사용하지 않은 경우, 위성을 설치하기 전에 사용해야 합니다. 자세한 정보는 *위성 관리 안내 및 참조서*의 위성 환경 설정 및 테스트 방법에 대한 설명을 참조하십시오.

4. 위성으로 사용하려는 버전 7.2 Personal Edition 또는 Workgroup Edition 시스템

설치 고려사항

DB2 Personal Edition 또는 DB2 Workgroup Edition을 설치할 경우, 시스템을 동기화 하기 위해 특별한 구성요소를 선택할 필요가 없습니다.

응답 파일 설치를 수행하는 경우 버전 7.2 시스템을 설치할 때 지정해야 하는 키워드에 대해서는 『응답 파일 설치 수행』을 참조하십시오. 버전 7.2 시스템의 대화식 설치를 수행하는 경우 버전 7.2 시스템을 동기화하기 위해 설정해야 하는 값에 대해서는 DB2를 설치한 후 138 페이지의 『동기화하기 위해 버전 7.2 시스템 구성』을 참조하십시오.

응답 파일 설치 수행: 버전 7.2 DB2 Personal Edition 또는 DB2 Workgroup Edition의 응답 파일 설치를 수행하는 경우, 응답 파일에서 다음 키워드를 설정할 수 있습니다.

응답 파일 설치시 이들 키워드 중 하나 이상을 지정하지 않을 경우, 버전 7.2 시스템을 동기화하기 위해 DB2를 설치한 후 수행해야 하는 추가 단계에 대해서는 138 페이지의 『동기화하기 위해 버전 7.2 시스템 구성』을 참조하십시오. 또한 응답 파일 설치시 지정한 값을 변경하려면 이 섹션의 지침을 사용할 수 있습니다.

db2.db2satelliteid

시스템에서 위성 ID를 설정합니다.

주: 키워드를 지정하지 않으면 위성 ID가 DB2를 설치할 때 사용된 사용자 ID로 자동으로 설정됩니다. 사용자 ID를 위성 ID로 사용하려면 키워드 값을 지정할 필요가 없습니다.

db2.db2satelliteappver

시스템의 응용프로그램 버전을 설정합니다.

주: 키워드를 지정하지 않으면 위성의 응용프로그램 버전이 자동으로 V1R0M00으로 설정됩니다. 이 값을 응용프로그램 버전으로 사용하려면 키워드의 값을 지정할 필요가 없습니다.

db2.satctldb_username

위성 제어 데이터베이스에 연결하기 위해 시스템에서 사용할 사용자 이름을 설정합니다.

db2.satctldb_password

사용자 이름이 위성 제어 데이터베이스에 연결될 때 사용자 이름이 DB2 제어 서버로 전달할 암호를 설정합니다.

응답 파일 설치를 완료하면 버전 7.2 시스템은 동기화할 준비가 됩니다. 위성에서 **db2sync -t** 명령을 실행하여 위성에서 지정한 값이 맞는지 그리고 위성이 위성 제어 데이터베이스에 연결될 수 있는지 확인해야 합니다.

응답 파일 설치에 대한 추가 정보는 *위성 관리 안내 및 참조서*에서 참조하십시오.

주:

1. 버전 7에서 사용자 ID 및 암호는 Windows NT 및 Windows 2000에서 모든 서비스를 작성하기 위해 필요합니다. 사용자 ID 및 암호는 응답 파일에서 키워드 쌍으로 지정됩니다. 해당 서비스에 대해 특정 키워드 쌍을 지정하여 서비스를 대체하도록 지정하지 않는 한 응답 파일에서 발견되는 첫 번째 키워드 쌍이 모든 서비스의 기본 사용자 ID 및 암호가 됩니다.

   버전 6에서 **admin.userid** 및 **admin.password** 키워드는 Remote Command Service에서 사용될 사용자 ID 및 암호를 지정하기 위해 DB2 Satellite Edition의 응답 파일 설치시 지정될 수 있습니다. 버전 7.2 Personal Edition 및 Workgroup Edition의 경우, 이들 키워드를 지정하면 버전 7.2 시스템의 DB2DAS00 인스턴스에 대해 사용됩니다. DB2 버전 7.2 시스템의 경우, Remote Command Service는 시스템의 DB2 인스턴스가 사용하는 사용자 ID 및 암호를 사용합니다. **db2.userid** 및 **db2.password**의 값을 지정하지 않으면 위에서 설명한 기본 규칙이 적용됩니다.

2. 버전 6에서는 응답 파일 설치를 사용하여 DB2 Satellite Edition을 설치할 때 데이터베이스를 작성할 수 있었습니다. 위성으로 사용하려는 버전 7.2 Personal Edition 또는 Workgroup Edition 시스템에서는 응답 파일 설치시 데이터베이스를 작성할 수 없습니다. 다음 키워드(*위성 관리 안내 및 참조서*에서 설명)는 지원되지 않습니다.

   - **db2.userdb_name**
   - **db2.userdb_recoverable**
   - **db2.userdb_rep_src**

동기화하기 위해 버전 7.2 시스템 구성

버전 7.2 시스템을 대화식으로 설치할 경우, DB2를 설치한 후 시스템을 동기화하기 전에 DB2 Personal Edition 또는 DB2 Workgroup Edition 시스템에서 몇 가지 값을 설정해야 합니다.

주: 시스템에서 운영 체제 스크립트를 실행하여 위성이 위성 제어 데이터베이스에 연결할 때 사용하는 사용자 ID와 암호를 제외한 모든 값을 위성에서 설정할 수 있습니다(140 페이지의 4단계 참조).

1. **db2set** 명령을 사용하여 위성 ID를 설정하십시오.

   DB2 Personal Edition 또는 DB2 Workgroup Edition을 대화식으로 설치하는 경우, 위성 ID는 자동으로 DB2를 설치할 때 사용된 사용자 ID로 설정됩니다. 사용자 ID를 위성 ID로 사용하려면 이 단계를 수행할 필요가 없습니다. 위성 ID 설정에 대한 자세한 정보는 *위성 관리 안내 및 참조서*에서 참조하십시오.

2. **db2sync -s** 명령을 사용하여 위성에서 응용프로그램 버전을 설정하십시오.

   DB2 Personal Edition 또는 DB2 Workgroup Edition을 대화식으로 설치하는 경우, 위성의 응용프로그램 버전은 자동으로 V1R0M00으로 설정됩니다. 이 값을 응용프로그램 버전으로 사용하려면 이 단계를 수행할 필요가 없습니다.

   위성에서 **db2sync -g** 명령을 수행하여 응용프로그램 버전의 현재 설정을 볼 수 있습니다. 이 값을 변경하려면 **db2sync -s** 명령을 실행하십시오. 응용프로그램 버전에 대해 새 값을 제공하라는 프롬프트가 표시됩니다. 응용프로그램 버전 설정에 대한 자세한 정보는 *위성 관리 안내 및 참조서*에서 참조하십시오.

3. **catalog node** 및 **catalog database** 명령을 위성에서 실행하여 위성에서 DB2 제어 서버 인스턴스 및 위성 제어 데이터베이스 SATCTLDB를 카탈로그할 수 있습니다.

   위성에서 **db2sync -t** 명령을 사용하여 테스트 모드로 DB2 Synchronizer 응용 프로그램을 열 수도 있습니다. 명령을 실행할 때 위성에서 SATCTLDB 데이터베이스의 카탈로그가 만들어지지 않으면 카탈로그 제어 데이터베이스 창이 열립니다. 카탈로그 제어 데이터베이스 창에서 제공되는 DB2 발견 기능을 사용하여 DB2 제어 서버 및 SATCTLDB 데이터베이스의 카탈로그를 만들거나 이 창에서 호스트 이름 및 서버 이름을 입력할 수 있습니다. 140 페이지의 4단계에서 설명한대로 위성이 위성 제어 데이터베이스에 연결할 때 사용할 사용자 ID 및 암호를 지정하라는 프롬프트도 나타납니다.

   주: 버전 7.2 DB2 Personal Edition 또는 DB2 Workgroup Edition을 대화식으로 설치한 후, DB2 Synchronizer는 테스트 모드로 자동으로 시작되지 않습니다(버전 6 DB2 Satellite Edition의 경우와 동일).

4. 위성에서 **db2sync -t** 명령을 실행하여 다음을 수행하십시오.
   - 위성이 위성 제어 데이터베이스에 연결할 때 사용할 사용자 ID 및 암호를 지정하십시오.

     동기화 자격 증명이 아직 위성에 저장되지 않았으면 제어 데이터베이스에 연결 창이 열립니다. 이 창에서 위성이 위성 제어 데이터베이스에 연결할 때 사용할 사용자 ID 및 암호를 지정해야 합니다.
   - 위성에 설정된 값이 올바른지 검증하십시오.
   - 위성을 위성 제어 데이터베이스에 연결할 수 있는지 검증하십시오.

이 구성 태스크를 완료하면 버전 7.2 시스템은 동기화할 준비가 됩니다.

버전 6 Enterprise Edition 시스템에 FixPak 2 이상 설치

버전 6 Enterprise Edition 시스템을 DB2 제어 서버로 사용기 위해서는 시스템에 FixPak 2 이상 버전이 설치되어 있어야 합니다.

다음 섹션에서는 DB2 제어 서버로 사용하기 위해 Windows NT 또는 AIX에서 버전 6 Enterprise Edition 시스템을 업그레이드할 때 수행해야 하는 태스크에 대해 설명합니다. 버전 6 제어 센터를 사용하고 있는 경우에는 150 페이지의 『버전 6 제어 센터 및 위성 관리 센터 업그레이드』의 단계도 수행하여 위성 환경을 관리하는 올바른 레벨의 제어 센터 및 위성 관리 센터가 있음을 검증하십시오.

DB2 제어 서버로 사용하기 위해 버전 6 DB2 Enterprise Edition 업그레이드

버전 6 DB2 Enterprise Edition 시스템을 DB2 제어 서버로 사용하려면 제어 서버 구성요소와 함께 설치되어야 하며 DB2 Enterprise Edition의 레벨이 FixPak 2 서비스 레벨 이상이어야 합니다. DB2 제어 서버 구성요소가 설치되어 있는지 여부에 따라 그리고 DB2 Enterprise Edition의 서비스 레벨에 따라 다음 태스크 중 하나를 수행해야 합니다.

- DB2 제어 서버 구성요소를 기존 DB2 Enterprise Edition V6.1 시스템에 설치하고 FixPak 2 이상을 설치하십시오. 그런 다음 시스템의 위성 제어 데이터베이스(SATCTLDB)를 갱신하십시오.
- 이미 설치된 DB2 제어 서버를 FixPak 2 레벨 이상으로 업그레이드하십시오.

다음 정보를 참조하여 앞의 두 태스크 중 수행해야 할 태스크, 그리고 상황에 적용해야 할 단계를 식별하십시오. 다음은 수행해야 할 단계를 요약한 것입니다.

1. 먼저 현재의 DB2 Enterprise Edition 설치 상태를 살펴보십시오. 제어 서버 구성요소 및 DB2 서비스 레벨이 설치되어 있는지 파악하십시오.
2. 두 번째로 파악한 상태 정보에 따라 수행할 작업을 판단하십시오.
3. 세 번째로 DB2 Enterprise Edition을 업그레이드하기 위해 필요한 단계를 수행하십시오.

DB2 제어 서버는 Windows NT 및 AIX용 DB2 Enterprise Edition에서만 실행될 수 있습니다. 플랫폼에 적합한 지침에 따라 계속하십시오.

- 『Windows NT에서 DB2 Enterprise Edition 업그레이드』
- 146 페이지의『AIX에서 DB2 Enterprise Edition 업그레이드』

Windows NT에서 DB2 Enterprise Edition 업그레이드: 다음 섹션의 정보를 참조하여 버전 6 DB2 Enterprise Edition 시스템의 현재 서비스 레벨 및 시스템을 FixPak 2 서비스 레벨 이상으로 갱신하기 위해 수행해야 할 단계를 파악하십시오. 다음 절 중 하나 이상의 단계를 수행해야 합니다.

- 『Windows NT의 DB2 Enterprise Edition 평가』
- 142 페이지의『수행할 조치 판단』
- 143 페이지의『Windows NT에 제어 서버 구성요소 설치』
- 144 페이지의『Windows NT에 FixPak 2 이상 설치』
- 145 페이지의『Windows NT에서 SATCTLDB 업그레이드』

Windows NT의 DB2 Enterprise Edition 평가: DB2 Enterprise Edition이 Windows NT에 설치되어 있는 경우, 다음 단계를 수행하십시오.

1. 제어 서버 구성요소가 설치되어 있는지 확인하십시오. Registry Editor를 사용하여 설치된 구성요소 목록을 표시하십시오.
   a. 명령 프롬프트에서 `regedit`를 입력하십시오.
   b. HKEY_LOCAL_MACHINE\SOFTWARE\IBM\DB2\Components registry key에서 제어 서버가 목록에 있는지 확인하십시오. 목록에 없으면 제어 서버가 설치되지 않은 것입니다.

2. DB2 Enterprise Edition의 서비스 레벨을 파악하십시오. 명령 프롬프트에서 `db2level` 명령을 실행하십시오. 다음 테이블을 참조하여 출력 내용을 해석하십시오.

db2level 출력의 키 필드 값			**사용자 DB2 시스템**
릴리스	레벨	정보용 토큰	
SQL06010	01010104	db2_v6, n990616	버전 6.1 기본
SQL06010	01020104	DB2 V6.1.0.1, n990824, WR21136	버전 6.1과 FixPak 1
SQL06010	01030104	DB2 V6.1.0.6, s991030, WR21163 또는 DB2 V6.1.0.9, s000101, WR21173	버전 6.1과 FixPak 2

주: 레벨이 01030104 이상인 경우, 사용자 시스템은 FixPak 2 이상의 FixPak 레벨입니다.

3. 찾은 정보를 기록한 후 『수행할 조치 판단』을 계속하십시오.

수행할 조치 판단: 수집한 정보를 참조하여 다음 테이블에서 상황에 맞는 행을 찾아 FixPak 2 레벨 이상의 DB2 제어 서버를 지원하도록 DB2 Enterprise Edition 시스템을 준비하기 위해 필요한 단계를 수행하십시오.

테이블 뒤의 절에서 필요한 단계를 수행하기 위한 지침을 제공합니다. 각 단계를 수행할 때마다 표시하십시오. 상황에 맞는 단계만 수행하십시오.

제어 서버 구성요소가 설치됨	**DB2 Enterprise Edition 시스템의 서비스 레벨**	**DB2 Enterprise Edition 시스템을 준비하기 위해 필요한 단계**
아니오	버전 6.1 기본, 또는 버전 6.1과 FixPak 1, 또는 버전 6.1과 FixPak 2 이상	다음 단계를 수행하십시오. 1. 143 페이지의 『Windows NT에 제어 서버 구성요소 설치』 2. 144 페이지의 『Windows NT에 FixPak 2 이상 설치』 3. 145 페이지의 『Windows NT에서 SATCTLDB 업그레이드』

제어 서버 구성요소가 설치됨	DB2 Enterprise Edition 시스템의 서비스 레벨	DB2 Enterprise Edition 시스템을 준비하기 위해 필요한 단계
예	버전 6.1 기본 또는 버전 6.1과 FixPak 1	다음 단계를 수행하십시오. 1. 144 페이지의 『Windows NT에 FixPak 2 이상 설치』 2. 145 페이지의 『Windows NT에서 SATCTLDB 업그레이드』
예	버전 6.1과 FixPak 2 이상	다음 단계를 수행하십시오. 1. 145 페이지의 『Windows NT에서 SATCTLDB 업그레이드』

Windows NT에 제어 서버 구성요소 설치: Windows NT에 제어 서버 구성요소를 설치하려면 다음을 수행하십시오.

1. 계속하기 전에 시스템의 모든 데이터베이스 활동이 완료되었는지 확인합니다.
2. DB2 Universal Database Enterprise Edition 버전 6.1 CD를 CD 드라이브에 넣으십시오.

   설치 프로그램이 자동으로 시작하지 않으면 CD의 루트에서 `setup` 명령을 수행하여 설치를 시작합니다.
3. 프롬프트가 표시되면 DB2를 사용하는 모든 프로세스를 종료하십시오.
4. 환영 창에서 **다음**을 선택하십시오.
5. 제품 선택 창에서 DB2 Enterprise Edition이 선택되어 있는지 확인하십시오.
6. 설치 유형 선택 패널에서 **사용자 설치**를 누르십시오.
7. 구성요소 선택 패널에서 제어 서버 구성요소가 선택되어 있는지 확인하고 **다음**을 누르십시오.

   주: 시스템에 아직 설치되지 않은 다른 구성요소를 선택하면 이 구성요소도 설치됩니다. DB2가 설치된 드라이브나 디렉토리를 변경할 수 없습니다.
8. DB2 서비스 구성 패널에서 프로토콜 값 및 제어 서버 인스턴스의 시작 옵션을 수정하거나 기본값을 사용할 수 있습니다. 기본값을 수정하고 **다음**을 누르거나 **다음**을 눌러 기본값을 사용할 수 있습니다.
9. 파일 복사 시작 창에서 **다음**을 눌러 설치를 시작하십시오.

10. 파일 복사 프로세스가 완료되면 시스템을 재부트할 수 있는 옵션이 나타납니다. 이제 재부트해야 합니다. 제어 서버에 맞게 시스템을 변경한 내용은 시스템을 재부트할 때까지 적용되지 않습니다.

설치가 완료되고 시스템을 재부트했으면 제어 서버 설치 과정의 일부로 작성된 위성 제어 데이터베이스(SATCTLDB)를 DB2 인스턴스에서 카탈로그해야 시스템에서 로컬 제어 센터 및 위성 관리 센터을 사용할 수 있습니다. SATCTLDB 데이터베이스를 카탈로그하려면 다음을 수행하십시오.

1. **시작 --> 프로그램 --> Windows NT용 DB2 --> 명령 창**을 선택하여 DB2 명령 창을 여십시오.
2. db2 인스턴스를 사용하고 있는지 확인하십시오.

   `set` 명령을 실행하고 db2instance 값을 확인하십시오. 값이 `db2`가 아니면 다음 명령을 실행하십시오.

   ```
   set db2instance=db2
   ```
3. 다음 명령을 입력하여 db2ctlsv 인스턴스를 카탈로그하십시오.

   ```
   db2 catalog local node db2ctlsv instance db2ctlsv
   ```
4. 다음 명령을 입력하여 SATCTLDB 데이터베이스를 카탈로그하십시오.

   ```
   db2 catalog database satctldb at node db2ctlsv
   ```
5. 다음 명령을 입력하여 카탈로그 조치를 커미트하십시오.

   ```
   db2 terminate
   ```
6. DB2 명령 창을 닫으십시오.

Windows NT에 FixPak 2 이상 설치: Windows NT에서 버전 6 DB2 Enterprise Edition 시스템을 FixPak 2 이상으로 업그레이드하려면 다음 중 하나를 수행하십시오.

- Windows NT용 DB2 Enterprise Edition V6.1의 최신 FixPak을 readme와 함께 웹에서 다운로드하십시오. 지침에 따라 다음 URL에서 FixPak을 다운로드할 수 있습니다.

  `http://www.ibm.com/software/data/db2/db2tech/version61.html`

  readme.txt 파일의 지침에 따라 FixPak을 설치하십시오.

- FixPak 2 레벨 이상의 Windows NT용 DB2 Universal Database, 버전 6.1 FixPak CD를 사용하여 CD의 WINNT95 디렉토리의 readme.txt 파일 지침에 따라 설치를 완료하십시오.

Windows NT에서 SATCTLDB 업그레이드: Windows NT에서 SATCTLDB 데이터베이스를 업그레이드하려면 다음을 수행하십시오.

1. SATCTLDB 데이터베이스의 레벨을 파악하십시오.
   a. Windows NT 시스템에서 로컬 관리자 권한을 가진 사용자 ID로 로그온하십시오.
   b. **시작 --> 프로그램 --> Windows NT용 DB2 --> 명령 창**을 선택하여 DB2 명령 창을 여십시오.
   c. 다음 명령을 입력하여 SATCTLDB에 연결하십시오.

      ```
      db2 connect to satctldb
      ```

   d. 다음 쿼리를 실행하여 데이터베이스에 I_BATCHSTEP_TRGSCR 트리거가 있는지 파악하십시오.

      ```
      db2 select name from sysibm.systriggers
                          where name='I_BATCHSTEP_TRGSCR'
      ```

      리턴된 행 수를 기록하십시오.
   e. 다음 명령을 입력하여 데이터베이스와의 연결을 끊으십시오.

      ```
      db2 connect reset
      ```

   1d단계에서 하나의 행을 리턴하면 데이터베이스 레벨이 정확한 것입니다. 이 경우 2단계를 건너뛰고 146 페이지의 3단계로 진행합니다. 0개의 행이 리턴되면 데이터베이스 레벨이 잘못된 것이므로 146 페이지의 3단계를 수행하기 전에 2단계에서 설명한대로 업그레이드해야 합니다.
2. SATCTLDB 데이터베이스를 업그레이드하려면 다음 단계를 수행하십시오. DB2 명령 창에서 모든 명령을 입력하십시오.
   a. <db2path>\misc 디렉토리로 변경하십시오. <db2path>는 설치 드라이브 및 경로입니다(예: `c:\sqllib`).
   b. db2ctlsv 인스턴스를 사용하고 있는지 확인하십시오.

set 명령을 실행하고 db2instance 값을 확인하십시오. 값이 db2ctlsv가 아닌 경우, 다음 명령을 실행하십시오.

```
set db2instance=db2ctlsv
```

c. 다음 명령을 입력하여 SATCTLDB 데이터베이스를 삭제하십시오.

```
db2 drop database satctldb
```

d. 다음 명령을 입력하여 새로운 SATCTLDB 데이터베이스를 작성하십시오.

```
db2 -tf satctldb.ddl  -z satctldb.log
```

e. 다음 명령을 실행하십시오.

```
db2 terminate
```

3. db2satcs.dll 저장 프로시저를 SATCTLDB 데이터베이스에 바인드하십시오. 다음 단계를 수행하십시오.

a. 다음 명령을 입력하여 SATCTLDB 데이터베이스에 연결하십시오.

```
db2 connect to satctldb
```

b. <db2path>\bnd 디렉토리로 변경하십시오. <db2path>는 설치 드라이브 및 경로입니다(예: `c:\sqllib`).

c. 다음과 같이 bind 명령을 실행하십시오.

```
db2 bind db2satcs.bnd
```

4. 다음 명령을 입력하여 데이터베이스와의 연결을 끊으십시오.

```
db2 connect reset
```

5. DB2 명령 창을 닫으십시오.

AIX에서 DB2 Enterprise Edition 업그레이드: 다음 절의 정보를 참조하여 버전 6 DB2 Enterprise Edition 시스템의 현재 서비스 레벨 및 시스템을 FixPak 2 서비스 레벨 이상으로 갱신하기 위해 수행해야 할 단계를 파악하십시오. 다음 절 중 하나 이상의 단계를 수행해야 합니다.

- 147 페이지의『AIX에서 DB2 Enterprise Edition 평가』
- 147 페이지의『수행할 조치 판단』
- 148 페이지의『AIX에 제어 서버 구성요소 설치』
- 149 페이지의『AIX에 FixPak 2 이상 설치』
- 149 페이지의『AIX에서 SATCTLDB 업그레이드』

AIX에서 DB2 Enterprise Edition 평가: 버전 6 DB2 Enterprise Edition이 AIX에 설치되어 있는 경우, 다음 단계를 수행하십시오.

1. 제어 서버 구성요소가 설치되어 있는지 확인하십시오. 다음 명령을 입력하십시오.

   ```
   lslpp -l | grep db2_06_01.ctsr
   ```

   리턴된 데이터가 없으면 제어 서버 구성요소가 설치되지 않은 것입니다.
2. DB2 Enterprise Edition의 서비스 레벨을 파악하십시오. DB2 인스턴스 소유자로 로그온하고 `db2level` 명령을 실행하십시오. 다음 테이블을 참조하여 출력 내용을 해석하십시오.

db2level 출력의 키 필드 값			**사용자 DB2 시스템**
릴리스	레벨	정보용 토큰	
SQL06010	01010104	db2_v6, n990616	버전 6.1 기본
SQL06010	01020104	DB2 V6.1.0.1, n990824, U465423	버전 6.1과 FixPak 1
SQL06010	01030104	DB2 V6.1.0.6, s991030, U468276 또는 DB2 V6.1.0.9, s000101, U469453	버전 6.1과 FixPak 2

   주: 레벨이 01030104 이상인 경우, 사용자 시스템은 FixPak 2 이상의 FixPak 레벨입니다.
3. 찾은 정보를 기록한 후 『수행할 조치 판단』을 계속합니다.

수행할 조치 판단: 수집한 정보를 참조하여 다음 테이블에서 상황에 맞는 행을 찾아 FixPak 2 레벨의 DB2 제어 서버를 지원하도록 DB2 Enterprise Edition 시스템을 준비하기 위해 필요한 단계를 수행하십시오.

테이블 뒤의 절에서 필요한 단계를 수행하기 위한 지침을 제공합니다. 각 단계를 수행할 때마다 표시하십시오. 상황에 맞는 단계만 수행하십시오.

제어 서버 구성요소가 설치됨	**DB2 Enterprise Edition 시스템의 서비스 레벨**	**DB2 Enterprise Edition 시스템을 준비하기 위해 필요한 단계**
아니오	버전 6.1 기본, 또는 버전 6.1과 FixPak 1, 또는 버전 6.1과 FixPak 2 이상	다음 단계를 수행하십시오. 1. 『AIX에 제어 서버 구성요소 설치』 2. 149 페이지의 『AIX에 FixPak 2 이상 설치』 3. 149 페이지의 『AIX에서 SATCTLDB 업그레이드』
예	버전 6.1 기본 또는 버전 6.1과 FixPak 1	다음 단계를 수행하십시오. 1. 149 페이지의 『AIX에 FixPak 2 이상 설치』 2. 149 페이지의 『AIX에서 SATCTLDB 업그레이드』
예	버전 6.1과 FixPak 2 이상	다음 단계를 수행하십시오. 1. 149 페이지의 『AIX에서 SATCTLDB 업그레이드』

AIX에 제어 서버 구성요소 설치: AIX에 제어 서버 구성요소를 설치하려면 다음을 수행하십시오.

1. root 권한이 있는 사용자로 로그온하십시오.
2. DB2 Universal Database Enterprise Edition 버전 6.1 CD를 CD 드라이브에 넣으십시오.
3. CD가 마운트된 디렉토리로 변경하십시오(예: `cd /cdrom`).
4. 다음 명령을 입력하여 DB2 설치 프로그램을 시작하십시오.

   ```
   ./db2setup
   ```
5. DB2 설치 프로그램 창이 열리면 탭 키를 사용하여 **설치** 옵션을 선택하고 Enter를 누르십시오.
6. Enterprise Edition 계열을 찾아 탭 키를 사용하여 옆의 **사용자 정의** 옵션을 선택하십시오. Enter를 누르십시오.
7. DB2 제어 서버 구성요소를 선택하고 탭 키를 사용하여 확인을 선택한 후 Enter를 누르십시오.
8. 남은 창의 지침에 따라 DB2 제어 서버 구성요소의 설치를 완료하십시오.

설치가 완료되면 DB2CTLSV 인스턴스 및 SATCTLDB 데이터베이스를 작성하십시오. 이 태스크를 수행하려면 *위성 관리 안내 및 참조서* 제13장의 "AIX에서 DB2 제어 서버 설정"의 자세한 지침을 따르십시오.

AIX에 FixPak 2 이상 설치: 기존 DB2 Enterprise Edition 시스템 AIX를 FixPak 2 이상으로 업그레이드하려면 다음 중 하나를 수행하십시오.

- AIX용 DB2 Enterprise Edition V6.1의 최신 FixPak을 FixPak readme와 함께 웹에서 다운로드하십시오. 지침에 따라 다음 URL에서 FixPak을 다운로드할 수 있습니다.

  ```
  http://www.ibm.com/software/data/db2/db2tech/version61.html
  ```

  FixPak readme 파일의 지침에 따라 FixPak을 설치하십시오.
- FixPak 2 레벨 이상의 AIX용 DB2 Universal Database, 버전 6.1 FixPak CD를 사용하여 CD의 readme 디렉토리의 지침에 따라 설치를 완료하십시오.

FixPak readme 파일의 지침에 따라 `db2iupdt` 명령을 실행하여 DB2CTLSV 인스턴스를 갱신했는지 확인하십시오.

AIX에서 SATCTLDB 업그레이드: AIX에서 SATCTLDB 데이터베이스를 업그레이드하려면 다음을 수행하십시오.

1. SATCTLDB 데이터베이스의 레벨을 파악하십시오.
   a. `db2ctlsv`로 로그온하십시오.
   b. 데이터베이스 서버가 시작되었는지 확인하십시오. 서버가 시작되지 않았으면 `db2start` 명령을 실행하십시오.
   c. 다음 명령을 입력하여 SATCTLDB 데이터베이스에 연결하십시오.

      ```
      db2 connect to satctldb
      ```

   d. 다음 쿼리를 실행하여 데이터베이스에 I_BATCHSTEP_TRGSCR 트리거가 있는지 파악하십시오.

      ```
      db2 "select name from sysibm.systriggers
                                where name='I_BATCHSTEP_TRGSCR'"
      ```

      리턴된 행 수를 기록하십시오.
   e. 다음 명령을 입력하여 데이터베이스와의 연결을 끊으십시오.

```
db2 connect reset
```

149 페이지의 1d단계에서 하나의 행을 리턴하면 데이터베이스 레벨이 정확한 것입니다. 이 경우 2단계를 건너뛰고 3단계로 진행합니다. 0개의 행이 리턴되면 데이터베이스 레벨이 잘못된 것이므로 3단계를 수행하기 전에 2단계에서 설명한대로 업그레이드해야 합니다.

2. SATCTLDB 데이터베이스를 FixPak 2 레벨로 업그레이드하려면 다음 단계를 수행하십시오. DB2 명령 창에서 모든 명령을 입력하십시오.
   a. $HOME/sqllib/misc 디렉토리로 변경하십시오.
   b. 다음 명령을 입력하여 SATCTLDB 데이터베이스를 삭제하십시오.
      ```
      db2 drop database satctldb
      ```
   c. 다음 명령을 입력하여 새로운 SATCTLDB 데이터베이스를 작성하십시오.
      ```
      db2 -tf satctldb.ddl  -z $HOME/satctldb.log
      ```
   d. 다음 명령을 실행하십시오.
      ```
      db2 terminate
      ```
3. db2satcs.dll 저장 프로시저를 SATCTLDB 데이터베이스에 바인드하십시오. 다음 단계를 수행하십시오.
   a. 다음 명령을 입력하여 SATCTLDB 데이터베이스에 연결하십시오.
      ```
      db2 connect to satctldb
      ```
   b. $HOME/sqllib/bnd 디렉토리로 변경하십시오.
   c. 다음과 같이 bind 명령을 실행하십시오.
      ```
      db2 bind db2satcs.bnd
      ```
4. 다음 명령을 입력하여 데이터베이스와의 연결을 끊으십시오.
   ```
   db2 connect reset
   ```

버전 6 제어 센터 및 위성 관리 센터 업그레이드

제어 센터 및 위성 관리 센터 버전 6을 FixPak 2 이상으로 업그레이드된 DB2 제어 서버 버전 6 및 위성 제어 데이터베이스(SATCTLDB)와 함께 사용하려면 도구도 FixPak 2 이상으로 업그레이드해야 합니다.

제어 센터 및 위성 관리 센터가 DB2 제어 서버와 같은 시스템에서 실행되고 있으면 DB2 Enterprise Edition 시스템이 FixPak 2로 업그레이드될 때 함께 업그레이드된 것입니다. 그러나 이들 도구를 다른 시스템에서 실행하려면 이 시스템을 FixPak 2 레벨 이상으로 업그레이드해야 합니다.

이 시스템을 FixPak 2 이상으로 업그레이드하려면 다음을 수행하십시오.

- V6.1의 사용자 제품용 최신 FixPak을 readme와 함께 웹에서 다운로드하십시오. 지침에 따라 다음 URL에서 FixPak을 다운로드할 수 있습니다.

  ```
  http://www.ibm.com/software/data/db2/db2tech/version61.html
  ```

  readme 파일의 지침에 따라 FixPak을 설치하십시오.
- FixPak 2 레벨 이상의 사용하는 운영 체제용 DB2 Universal Database, 버전 6.1 FixPak CD를 사용하여 readme의 지침에 따라 설치를 완료하십시오.

Command Reference

사용 가능한 갱신사항

Command Reference는 FixPak 4에서 갱신되었습니다. 최신 PDF는 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다. 또한 갱신된 참조사항에 이러한 정보를 추가해 놓았습니다. 모든 갱신된 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

db2updv7 - 버전 7 현재 수정 레벨로 데이터베이스 갱신

이 명령은 데이터베이스의 시스템 카탈로그가 현재 FixPak을 지원하도록 다음 방법으로 갱신합니다.

- 새로운 내장 함수(ABS, DECRYPT_BIN, DECRYPT_CHAR, ENCRYPT, GETHINT, MULTIPLY_ALT 및 ROUND)의 사용을 가능하게 합니다.
- 유니코드 데이터베이스용 새 내장 함수 DATE(vargraphic), TIME(vargraphic), TIMESTAMP(vargraphic), GRAPHIC(datetime-expression), GRAPHIC(date-expression), GRAPHIC(time-expression) 및 VARGRAPHIC(datetime-expression)의 사용을 가능하게 합니다.
- 새로운 내장 프로시저(GET_ROUTINE_SAR 및 PUT_ROUTINE_SAR)의 사용을 가능하게 합니다.
- Windows 및 OS/2 데이터베이스에서 WEEK_ISO 및 DAYOFWEEK_ISO 함수에 정정사항을 추가 또는 적용합니다.
- 버전 2에서 버전 6으로 이주한 테이블에 대한 테이블 팩 설명자에 정정을 적용합니다.
- 뷰 SYSCAT.SEQUENCES를 작성합니다.

권한 부여

sysadm

필요한 연결

데이터베이스. 이 명령은 지정된 데이터베이스에 자동으로 연결을 설정합니다.

명령 구문

```
>>-db2updv7---d--database_name--+----------------------------+--+----+-><
                                '--u--userid---p--password--'  '--h-'
```

명령 매개변수

-d database-name
: 갱신될 데이터베이스의 이름을 지정합니다.

-u userid
: 사용자 ID를 지정합니다.

-p password
: 사용자의 암호를 지정합니다.

-h
: 도움말 정보를 표시합니다. 이 옵션이 지정된 경우, 다른 모든 옵션은 무시되며 도움말 정보만이 표시됩니다.

예 FixPak 설치 후, 다음 명령을 사용하여 샘플 데이터베이스의 시스템 카탈로그를 갱신하십시오.

```
db2updv7 -d sample
```

사용 주의사항

이 도구는 최소한 FixPak 2가 설치된 DB2 버전 7.1 또는 버전 7.2를 수행하는 데이터베이스에서만 사용할 수 있습니다. 명령을 한 번 이상 수행할 경우, 오류가 보고되지 않고 각각의 카탈로그 갱신은 단 한번 적용됩니다.

새로운 내장 함수를 사용하려면, 모든 응용프로그램과 이 데이터베이스와의 연결을 해제하고 데이터베이스가 활성화되어 있을 경우 비활성화해야 합니다.

ARCHIVE LOG의 추가 컨텍스트에 대한 사용 설명

현재 ARCHIVE LOG에 대한 사용법 설명에서는 이 명령을 사용할 경우 데이터베이스가 로그 시퀀스 번호(LSN) 스페이스의 일부를 잃게 되어, 유효한 LSN이 빨리 소모됨을 기술하고 있습니다. 로그 파일 크기가 100M이고 ARCHIVE LOG를 5분마다 실행할 경우, 이 스페이스 사용을 컨텍스트에 넣으려면 유효한 LSN을 모두 소모하는 데 여전히 거의 40년이 걸립니다. 대부분의 운영 조건에서는 영향을 받지 않습니다.

REBIND

REBIND 명령 구문은 다음과 같이 나타나야 합니다.

명령 구문

```
                                                       ┌─ANY──────────┐
►►──REBIND──┬─────────┬──package-name──RESOLVE──┴─CONSERVATIVE─┴────────────►◄
            └─PACKAGE─┘
```

db2inidb - 미러된 데이터베이스 초기화

RELOCATE USING *configfile* 매개변수에 대한 설명은 다음과 같이 표시되어야 합니다.

데이터베이스를 스냅샷, 스탠드바이 또는 미러로 초기화하기 전에 구성 파일에 나열된 정보에 기초하여 데이터베이스 파일의 위치를 바꿀 수 있도록 지정합니다.

주: 구성 파일 형식에 대한 자세한 정보는 *데이터 이동 유틸리티 안내 및 참조서*를 참조하십시오.

사용 정보

RELOCATE USING *configfile* 매개변수가 지정되고 데이터베이스가 다시 배치된 경우, 구성 파일은 데이터베이스 디렉토리로 복사하고 `db2path.cfg`로 이름이 바뀝니다. 차후 응급 복구 또는 롤 포워드 복구 동안, 이 구성 파일을 로그 파일 처리 동안에 컨테이너 경로의 이름을 동적으로 바꾸는 데 사용합니다.

스냅샷 또는 미러 데이터베이스를 초기화한 경우, 구성 파일은 복구가 완료된 후에 자동으로 제거됩니다. 대기 데이터베이스를 초기화한 경우, 구성 파일은 복구가 완료된 후에 제거될 뿐만 아니라, 복구 프로세스를 취소한 경우에도 제거됩니다.

보류 상태를 유지하는 대기 데이터베이스로 작업 중이어서 계속 롤 포워드할 수 있으며, 새 컨테이너를 원래의 데이터베이스로 추가한 경우, 수동으로 `db2path.cfg` 파일을 갱신하여 대기 데이터베이스에 대한 컨테이너가 어디에 저장되어야 하는지를 나타낼 수 있습니다. 새 컨테이너에 대한 위치를 지정하지 않은 경우, DB2는 원래의 동일한 위치에 저장하려고 시도합니다.

db2relocatedb(새 명령)

db2relocatedb - 데이터베이스 위치 재지정

데이터베이스의 이름을 바꾸거나 사용자가 제공한 구성 파일에 지정된 대로 데이터베이스 또는 데이터베이스의 일부분(예: 컨테이너나 로그 디렉토리)의 위치를 바꿉니다. 이 도구를 이용해 DB2 인스턴스와 데이터베이스 지원 파일을 필요에 맞게 변경할 수 있습니다.

권한 부여

없음

필수 연결

없음

명령 구문

```
►►—db2relocatedb—-f—configFilename————————————————————►◄
```

명령 매개변수

-f configFilename

데이터베이스의 위치를 재지정하는 데 필요한 구성 정보가 들어 있는 파일의 이름을 지정합니다. 파일 이름은 상대적이거나 절대적일 수 있습니다. 구성 파일의 형식은 다음과 같습니다.

```
DB_NAME=oldName,newName
DB_PATH=oldPath,newPath
INSTANCE=oldInst,newInst
NODENUM=nodeNumber
LOG_DIR=oldDirPath,newDirPath
CONT_PATH=oldContPath1,newContPath1
CONT_PATH=oldContPath2,newContPath2
...
```

여기서,

DB_NAME

위치 재지정 중인 데이터베이스 이름을 지정합니다. 데이터베이스 이름이 바뀌면 이전 이름과 새 이름을 모두 지정해야 합니다. 이는 필수 필드입니다.

DB_PATH

위치 재지정 중인 데이터베이스 경로를 지정합니다. 데이터베이스

가 원래 작성된 경로입니다. 데이터베이스 경로가 바뀌면 이전 경로 및 새 경로를 모두 지정해 주어야 합니다. 이는 필수 필드입니다.

INSTANCE

데이터베이스가 존재하는 인스턴스를 지정합니다. 데이터베이스가 새 인스턴스로 이동하면 이전 인스턴스와 새 인스턴스 모두 지정해 주어야 합니다. 이는 필수 필드입니다.

NODENUM

변경 중인 데이터베이스 노드에 대한 노드 번호를 지정합니다. 기본값은 0입니다.

LOG_DIR

로그 경로의 위치에서 변경사항을 지정합니다. 로그 경로가 바뀌면 이전 경로 및 새 경로를 모두 지정해 주어야 합니다. 경로가 자동으로 갱신되는 경우, 로그 경로가 데이터베이스 경로 아래에 있으면 이 스펙은 선택적입니다.

CONT_PATH

테이블 스페이스 컨테이너의 위치에서 변경사항을 지정합니다. 이전 컨테이너 경로와 새 컨테이너 경로 모두 지정해 주어야 합니다. 다중 컨테이너 경로를 변경해야 하는 경우에는 다중 CONT_PATH 행을 제공할 수 있습니다. 경로가 자동으로 갱신되는 경우, 컨테이너 경로가 데이터베이스 경로 아래에 있으면 이 스펙은 선택적입니다.

주: 빈 행이나 주석 문자(#)로 시작하는 행은 무시됩니다.

예

예 1

`/home/db2inst1` 경로에 상주하는 DB2INST1 인스턴스에서 데이터베이스 TESTDB의 이름을 PRODDB로 변경하려면, 다음과 같이 구성 파일을 작성하십시오.

```
DB_NAME=TESTDB,PRODDB
DB_PATH=/home/db2inst1
INSTANCE=db2inst1
NODENUM=0
```

구성 파일을 relocate.cfg로 저장하고 나서 다음 명령을 사용해서 데이터베이스 파일을 변경하십시오.

```
db2relocatedb -f relocate.cfg
```

예 2

경로 /dbpath의 인스턴스 JSMITH에서 데이터베이스 DATAB1을 이동하려면, 다음과 같이 하십시오.

1. 디렉토리 /dbpath/jsmith에서 /dbpath/prodinst로 파일을 이동시키십시오.
2. 다음 구성 파일을 **db2relocatedb** 명령과 함께 사용하여 데이터베이스 파일을 변경하십시오.

   ```
   DB_NAME=DATAB1
   DB_PATH=/dbpath
   INSTANCE=jsmith,prodinst
   NODENUM=0
   ```

예 3

경로 /databases/PRODDB의 인스턴스 INST1에 데이터베이스 PRODDB가 존재합니다. 두 테이블 스페이스 컨테이너의 위치를 다음과 같이 변경해야 합니다.

- SMS 컨테이너 /data/SMS1은 /DATA/NewSMS1로 이동시켜야 합니다.
- DMS 컨테이너 /data/DMS1은 /DATA/DMS1로 이동시켜야 합니다.

실제 디렉토리 및 파일을 새 위치로 이동하였으면 다음 구성 파일을 **db2relocatedb** 명령과 함께 사용해서 새 위치를 인식할 수 있도록 데이터베이스 파일을 변경하십시오.

```
DB_NAME=PRODDB
DB_PATH=/databases/PRODDB
INSTANCE=inst1
NODENUM=0
CONT_PATH=/data/SMS1,/DATA/NewSMS1
CONT_PATH=/data/DMS1,/DATA/DMS1
```

예 4

TESTDB 데이터베이스는 DB2INST1 인스턴스에 있으며 /databases/TESTDB 경로에서 작성되었습니다. 그런 다음 테이블 스페이스가 다음 컨테이너와 함께 작성되었습니다.

```
TS1
TS2_Cont0
TS2_Cont1
/databases/TESTDB/TS3_Cont0
/databases/TESTDB/TS4/Cont0
/Data/TS5_Cont0
/dev/rTS5_Cont1
```

TESTDB가 새 시스템으로 이동됩니다. 새 시스템에서 인스턴스는 NEWINST가 되며 데이터베이스의 위치는 /DB2가 됩니다.

데이터베이스를 이동할 때 /databases/TESTDB/db2inst1 디렉토리에 존재하는 모든 파일을 /DB2/newinst 디렉토리로 이동해야 합니다. 이것은 처음 다섯 컨테이너의 위치가 이동의 일부분으로서 재지정된다는 것을 의미합니다. (처음 세 개는 데이터베이스 디렉토리와 관계가 있고 나머지 둘은 데이터베이스 경로와 관계가 있습니다.) 이 다섯 개의 컨테이너는 데이터베이스 디렉토리나 데이터베이스 경로에 위치하기 때문에 구성 파일에 나열할 필요가 없습니다. 그렇지 않고 남아 있는 두 개의 컨테이너를 새 시스템의 다른 위치로 이동하는 경우에는 구성 파일에 나열해야 합니다.

실제 디렉토리 및 파일을 새 위치로 이동하였으면 다음 구성 파일을 **db2relocatedb** 명령과 함께 사용해서 새 위치를 인식할 수 있도록 데이터베이스 파일을 변경하십시오.

```
DB_NAME=TESTDB
DB_PATH=/databases/TESTDB,/DB2
INSTANCE=db2inst1,newinst
NODENUM=0
CONT_PATH=/Data/TS5_Cont0,/DB2/TESTDB/TS5_Cont0
CONT_PATH=/dev/rTS5_Cont1,/dev/rTESTDB_TS5_Cont1
```

예 5

데이터베이스 TESTDB에는 노드 10 및 20에 2개의 파티션이 있습니다. 인스턴스 SERVINST 및 데이터베이스 경로는 두 노드에서 /home/servinst입니다. 두 노드에서 데이터베이스의 이름은 SERVDB로 변경되며 데이터베이스 경로는 /databases로 변경됩니다. 또한, 노드 20의 로그 디렉토리는 /testdb_logdir에서 /servdb_logdir로 변경됩니다.

두 노드가 변경되기 때문에 각 노드에 구성 파일을 따로 작성해야 하고 **db2relocatedb**를 해당 구성 파일과 함께 각 노드에서 실행해야 합니다.

노드 10에서 다음 구성 파일이 사용됩니다.

```
DB_NAME=TESTDB,SERVDB
DB_PATH=/home/servinst,/databases
INSTANCE=servinst
NODE_NUM=10
```

노드 20에서 다음 구성 파일이 사용됩니다.

```
DB_NAME=TESTDB,SERVDB
DB_PATH=/home/servinst,/databases
INSTANCE=servinst
NODE_NUM=20
LOG_DIR=/testdb_logdir,/servdb_logdir
```

사용 참고

데이터베이스가 속한 인스턴스가 변경되는 경우에는 이 명령을 실행해 인스턴스와 데이터베이스 지원을 변경하기 전에 다음 작업을 수행해야 합니다.

- 데이터베이스를 다른 인스턴스로 이동하는 경우, 새 인스턴스를 작성하십시오.
- 복사되는 데이터베이스에 속한 파일/장치를 새 인스턴스가 상주하는 시스템으로 복사하십시오. 경로 이름은 필요에 따라 변경해야 합니다.
- 인스턴스 소유자가 소유할 수 있도록 복사된 파일/장치의 사용권한을 변경하십시오.

인스턴스를 변경하는 경우에는 새 인스턴스 소유자가 이 도구를 실행해야 합니다.

EEE 환경에서 이 도구는 변경을 요구하는 모든 노드에 기초하여 실행되어야 합니다. 따라서 변경되는 노드의 NODENUM 값을 포함하는 개별 구성 파일을 각 노드에 공급해야 합니다. 예를 들어 데이터베이스의 이름을 변경하면 모든 노드가

이에 영향을 받게 되며 각 노드의 개별 구성 파일과 함께 **db2relocatedb** 명령을 실행해야 합니다. 단일 노드에 속한 컨테이너를 이동하는 경우에는 **db2relocatedb** 명령을 해당 노드에서 실행해야 합니다.

참조

자세한 정보는 *Command Reference*의 **db2inidb - 미러된 데이터베이스 초기화** 명령을 참조하십시오.

db2move

db2move 도구는 현재 두 개의 추가 옵션(**--aw** 및 **--sn**)을 갖습니다. 이 도구에 대한 전체 문서는 다음과 같습니다.

데이터베이스 이동 도구

이 도구는 워크스테이션에 있는 DB2 데이터베이스 사이에 많은 수의 테이블 이동을 용이하게 합니다. 도구는 특정 데이터베이스에 대한 시스템 카탈로그 테이블을 쿼리하고 모든 사용자 테이블의 목록을 컴파일합니다. 그런 다음 PC/IXF 형식으로 이 테이블을 익스포트합니다. PC/IXF 파일를 동일한 시스템에 있는 다른 로컬 DB2 데이터베이스로 임포트하거나 로드할 수 있습니다. 또한 다른 워크스테이션 플랫폼으로 전송할 수 있으며, 그 플랫폼에서 DB2 데이터베이스로 임포트하거나 로드할 수 있습니다.

주: 구조화된 유형 컬럼을 가진 테이블은 이 도구를 사용할 때 이동하지 않습니다.

권한 부여

이 도구는 사용자가 요청한 조치에 따라 DB2 익스포트, 임포트 및 API 로드를 호출합니다. 그러므로 요청 사용자 ID는 이 API에서 필요로 하는 올바른 권한 부여를 가져야 하고, 그렇지 않을 경우, 요청은 실패합니다.

명령 구문

```
                               .------------------------.
                               V                        |
>>--db2move--dbname--action--+------------------------+-+---------------><
                             +--tc--table-creators----+
                             +--tn--table-names-------+
                             +--sn--schema names------+
                             +--io--import-option-----+
                             +--lo--load-option-------+
                             +--l--lobpaths-----------+
                             +--u--userid-------------+
                             +--p--password-----------+
                             '--aw--------------------'
```

명령 매개변수

dbname

데이터베이스의 이름.

조치 EXPORT, IMPORT 또는 LOAD 중 하나이어야 합니다.

-tc 테이블 작성자. 기본값은 모든 작성자입니다.

이것은 EXPORT 조치에만 해당합니다. 이 옵션이 표시된 작성자가 작성한 테이블만 익스포트합니다. 지정되지 않은 경우, 기본값으로 모든 작성

자를 사용합니다. 다중 작성자를 지정할 할 경우, 각 작성자는 쉼표로 구분되어야 하며 작성자 ID 사이에는 공백을 허용하지 않습니다. 지정될 수 있는 작성자의 최대 수는 10입니다. 이 옵션은『-tn』및『-sn』옵션과 함께 사용되어 익스포트할 테이블을 선택할 수 있습니다.

별표(*)는 문자열 어디에나 위치될 수 있는 와일드 카드 문자로 사용될 수 있습니다.

-tn 테이블 이름. 기본값은 모든 사용자 테이블입니다.

이것은 EXPORT 조치에만 해당합니다. 지정된 경우, 지정된 문자열에 있는 테이블 이름과 완전히 일치하는 테이블만을 익스포트합니다. 지정되지 않은 경우, 기본값으로 모든 사용자 테이블을 사용합니다. 다중 테이블 이름을 지정할 때, 각 테이블 이름은 쉼표로 구분되어야 하며 테이블 이름 사이에는 공백을 허용하지 않습니다. 지정될 수 있는 테이블 이름의 최대 수는 10입니다. 이 옵션은『-tc』및『-sn』옵션과 함께 사용되어 익스포트할 테이블을 선택할 수 있습니다. **db2move**는 이름이 지정된 테이블 이름과 일치하며 작성자가 지정된 테이블 작성자와 일치하는 테이블만 익스포트합니다.

별표(*)는 문자열 어디에나 위치될 수 있는 와일드 카드 문자로 사용될 수 있습니다.

-sn 스키마 이름. 기본값은 ALL SCHEMAS입니다.

이것은 EXPORT 조치에만 해당합니다. 지정된 경우, 해당 스키마가 지정된 문자열에 있는 스키마와 완전히 일치하는 테이블만 익스포트합니다. 지정되지 않은 경우, 기본값으로 모든 스키마를 사용합니다. 다중 스키마 이름을 지정할 때, 각 스키마 이름은 쉼표로 구분되어야 하며 스키마 이름 사이에는 공백을 허용하지 않습니다. 지정될 수 있는 스키마 이름의 최대 수는 10입니다. 이 옵션은『-tc』및『-tn』옵션과 함께 사용되어 익스포트할 테이블을 선택할 수 있습니다. **db2move**는 이름이 지정된 테이블 이름과 일치하며 작성자가 지정된 테이블 작성자와 일치하는 테이블만을 익스포트합니다.

별표(*)는 문자열 어디에나 위치될 수 있는 와일드 카드 문자로 사용될 수 있습니다.

주: 8문자보다 적은 스키마 이름이 8문자 길이로 채워집니다. 예를 들어, 스키마 『AUSER』 및 『BUSER』를 포함하고 와일드 카드 문자를 사용하려는 경우, `-sn *USER*`를 지정해야 합니다.

-io 임포트 옵션. 기본값은 REPLACE_CREATE입니다.

유효한 옵션은 INSERT, INSERT_UPDATE, REPLACE, CREATE 및 REPLACE_CREATE입니다.

-lo 로드 옵션. 기본값은 INSERT입니다.

유효한 옵션은 INSERT 및 REPLACE입니다.

-l LOB 경로. 기본값은 현재 디렉토리입니다.

이 옵션은 LOB 파일이 작성되거나(EXPORT의 부분으로) 검색되는 (IMPORT 또는 LOAD의 부분으로) 절대 경로 이름을 지정합니다. 다중 LOB 경로를 지정할 때, 각 LOB 경로는 쉼표로 구분되어야 하며 LOB 경로 사이에는 공백이 허용되지 않습니다. 처음 경로의 공간을 모두 사용했거나(EXPORT 중) 파일이 경로에 없는 경우(IMPORT 또는 LOAD 중), 두 번째 경로가 사용되며, 이와 같은 방식으로 다음 경로가 사용됩니다.

조치가 EXPORT이며, LOB 경로가 지정된 경우, LOB 경로 디렉토리의 모든 파일이 검출되며 디렉토리가 제거되고 새 디렉토리가 작성됩니다. LOB 경로가 지정되지 않으면, 현재 디렉토리가 LOB 경로로 사용됩니다.

-u 사용자 ID. 기본값은 로그온된 사용자 ID입니다.

사용자 ID 및 암호는 둘다 선택적입니다. 그러나 하나를 지정한 경우, 다른 하나도 지정해야 합니다. 명령이 리모트 서버에 연결하는 클라이언트에서 수행되는 경우, 사용자 ID 및 암호를 지정해야 합니다.

-p 암호. 기본값은 로그온된 암호입니다.

사용자 ID 및 암호는 둘다 선택적입니다. 그러나 하나를 지정한 경우, 다른 하나도 지정해야 합니다. 명령이 리모트 서버에 연결하는 클라이언트에서 수행되는 경우, 사용자 ID 및 암호를 지정해야 합니다.

-aw 경고 허용.

EXPORT 조치에만 사용됩니다. 이 옵션이 지정되는 경우, 익스포트 중 수신 경고는 db2move.lst 파일에 포함됩니다. 이 옵션이 생략되는 경우, 익

스포트 중 원인 경고는 db2move.lst 파일에 포함되지 않습니다. 테이블의 .ixf 파일 및 .msg 파일은 이 옵션이 사용되는지 여부에 관계없이 생성됩니다.

예

- `db2move sample export`

  이것은 SAMPLE 데이터베이스의 모든 테이블을 익스포트하며, 모든 옵션에 대해 기본값이 사용됩니다.

- `db2move sample export -tc userid1,us*rid2 -tn tbname1,*tbname2`

  이것은 『userid1』 또는 사용자 ID LIKE 『us%rid2』가 작성하고 이름이 『tbname1』이거나 테이블 이름이 LIKE 『%tbname2』인 모든 테이블을 익스포트합니다.

- `db2move sample import -l D:\LOBPATH1,C:\LOBPATH2`

  이 예는 OS/2 또는 Windows 운영 체제에만 적용할 수 있습니다. 이 명령은 SAMPLE 데이터베이스에서 모든 테이블을 임포트하며, LOB 파일에 대한 LOB 경로 『D:\LOBPATH1』 및 『C:\LOBPATH2』가 검색됩니다.

- `db2move sample load -l /home/userid/lobpath,/tmp`

  이 예는 UNIX 기반 시스템에만 적용할 수 있습니다. SAMPLE 데이터베이스에서 모든 테이블을 로드하며, LOB 파일에 대한 `/home/userid/lobpath` 서브디렉토리 및 `tmp` 서브디렉토리 모두를 검색합니다.

- `db2move sample import -io replace -u userid -p password`

  이것은 REPLACE 모드에서 모든 테이블을 임포트하며, 지정된 사용자 ID 및 암호가 사용됩니다.

사용 참고

이 도구는 사용자 작성 테이블을 익스포트, 임포트 또는 로드합니다. 데이터베이스가 하나의 운영 체제에서 또 다른 운영 체제로 중복되는 경우, **db2move**는 테이블의 이동을 용이하게 합니다. 또한 별명, 뷰, 트리거, 사용자 정의 함수 등과 같이 테이블과 연관된 기타 모든 오브젝트를 이동시키는 것이 필요합니다. **db2look**(DB2 Statistics 및 DDL Extraction Tool, *Command Reference* 참조)은 데이터베이스에서 DDL(data definition language)문을 추출하여 이러한 오브젝트 일부의 이동을 용이하게 할 수 있습니다.

API 익스포트, 임포트 또는 로드가 **db2move**에 의해 호출되면, `FileTypeMod` 매개변수는 `lobsinfile`로 설정됩니다. 즉, LOB 데이터는 PC/IXF 파일에서 다른 파일에 보존됩니다. LOB 파일에 사용 가능한 26 000개의 파일 이름이 있습니다.

LOAD 조치는 데이터베이스 및 데이터 파일이 있는 머신에서 로컬로 수행되어야 합니다. 로드 API가 **db2move**에 의해 호출될 경우, `CopyTargetList` 매개변수는 널(NULL)로 설정되어, 복사가 수행되지 않습니다. *logretain*이 설정되는 경우, 로드 조작은 나중에 롤 포워드될 수 없습니다. 로드된 테이블이 상주하는 테이블 스페이스는 백업 보류 상태에 놓이며 액세스할 수 없습니다. 전체 데이터베이스 백업 또는 테이블 스페이스 백업은 백업 보류 상태로부터 테이블 스페이스를 얻는데 필요합니다.

이 도구가 버전 6 데이터베이스에 대해 버전 5.2 클라이언트에서 실행되었을 경우, 이 도구는 18문자 길이보다 긴 테이블이나 컬럼 이름을 지원하지 않습니다.

필수 파일/LOAD 사용시 생성:

- 입력: 없음
- 출력:

EXPORT.out EXPORT 조치의 요약된 결과.

db2move.lst 원래 테이블 이름, 해당 PC/IXF 파일 이름(tabnnn.ixf) 및 메시지 파일 이름(tabnnn.msg)의 목록. 이 목록과 익스포트한 PC/IXF 파일 및 LOB 파일(tabnnnc.yyy)은 **db2move** IMPORT 또는 LOAD 조치에 대한 입력으로 사용됩니다.

tabnnn.ixf 특정 테이블의 익스포트한 PC/IXF 파일.

tabnnn.msg 해당 테이블의 익스포트 메시지 파일.

tabnnnc.yyy 특정 테이블의 익스포트한 LOB 파일.

『nnn』은 테이블 번호이며,『c』는 영문자이고,『yyy』는 001 - 999 범위의 숫자입니다.

이 파일은 익스포트하는 테이블에 LOB 데이터가 들어 있는 경우에만 작성됩니다. 작성되는 경우, 이 LOB 파일은 LOB 경로 디렉토리에 있습니다. LOB 파일에 대해 총 26 000개의 가능한 이름이 있습니다.

system.msg 파일 또는 디렉토리 명령을 작성하거나 삭제하기 위한 시스템 메시지가 있는 메시지 파일. 이것은 조치가 EXPORT이며 LOB 경로가 지정된 경우에만 사용됩니다.

필수 파일/IMPORT 사용시 생성:

- 입력:

db2move.lst EXPORT 조치의 출력 파일.

tabnnn.ixf EXPORT 조치의 출력 파일.

tabnnnc.yyy EXPORT 조치의 출력 파일.

- 출력:

IMPORT.out IMPORT 조치의 요약된 결과.

tabnnn.msg 해당 테이블의 임포트 메시지 파일.

필수 파일/LOAD 사용시 생성:

- 입력:

  db2move.lst EXPORT 조치의 출력 파일.

  tabnnn.ixf EXPORT 조치의 출력 파일.

  tabnnnc.yyy EXPORT 조치의 출력 파일.

- 출력:

  LOAD.out LOAD 조치의 요약된 결과.

  tabnnn.msg 해당 테이블의 LOAD 메시지.

GET ROUTINE 명령에서의 추가 옵션

이 명령은 이제 HIDE BODY 매개변수를 지원하여, 루틴 텍스트가 카탈로그에서 발췌되었을 때 루틴의 본문을 빈 본문으로 대체해야 함을 지정합니다.

이것은 컴파일된 코드에는 영향을 미치지 않으며, 텍스트에만 영향을 미칩니다.

GET ROUTINE

명령 구문

```
>>--GET ROUTINE--INTO--file_name--FROM--+----------+--PROCEDURE--routine_name---------->
                                        '-SPECIFIC-'

>--+-----------+-------------------------------------------------------------------><
   '-HIDE BODY-'
```

CREATE DATABASE

DB2는 이제 유니코드 데이터베이스용 새 조합 순차 키워드인 IDENTITY_16BIT 및 SQL_CS_IDENTITY_16BIT를 지원합니다. IDENTITY_16BIT가 CLP CREATE DATABASE 명령에 지정되거나 SQLEDBDESC.SQLDBCSS가 sqlecrea() -- Create Database API에서 SQL_CS_IDENTITY_16BIT로 설정된 경우, 유니코드 데이터베이스의 모든 데이터는 CESU-8 순서로 배열됩니다. CESU-8은 *Compatibility Encoding Scheme for UTF-16: 8-Bit*의 약자로서, 본 서적이 집필될 무렵 유니코드 기술 콘소시엄 웹 사이트(www.unicode.org)에 있는 Draft Unicode Technical Report #26에 해당 스펙이 포함되어 있었습니다. CESU-8는 유니코드 보조 문자(즉, 16비트 BMP 또는 Plane 0 외부에 정의된 문자)를 제외하고 UTF-8과 동일한 2진수입니다. UTF-8 인코딩에서, 보조 문자는 하나의 4바이트 시퀀스로 표현되지만 CESU-8에서 동일한 문자를 표현하려면 두 개의 3바이트 시퀀스가 필요합니다.

유니코드 데이터베이스에서, CHAR, VARCHAR, LONG VARCHAR, CLOB 데이터는 UTF-8로 저장되고 GRAPHIC, VARGRAPHIC, LONG VARGRAPHIC, DBCLOB 데이터는 UCS-2로 저장됩니다. IDENTITY 또는 SQL_CS_NONE 배열에서 UTF-8 및 UCS-2로 된 기본(non-supplementary) 문자는 동일한 2진 배열을 가지지만 UTF-8로 된 보조 문자는 UCS-2로 된 동일한 문자와 다르게 배열됩니다. IDENTITY_16BIT 또는 SQL_CS_IDENTITY_16BIT는 DB2 유니코드 데이터베이스에 있는 보조 문자와 기본 문자를 포함한 모든 문자가 동일한 2진 배열을 갖도록 해줍니다.

데이터 복구와 고가용성 안내 및 참조서

데이터 복구와 고가용성 안내 및 온라인상에서 이용 가능한 참조서

데이터 복구와 고가용성 안내 및 참조서는 현재 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 HTML과 PDF 두 형식으로 온라인에서 제공됩니다. 이 정보는 이전에 관리 안내서에 포함되어 있었습니다. 또한 갱신된 참조에 이러한 참고 내용을 추가해 놓았습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

새 아카이브 로깅 동작

FixPak 4 이전에는 DB2가 새 로그 파일이 필요할 때만 아카이브가 완료되었는지 점검했습니다. 이제 DB2는 첫 번째 아카이브 로그가 변경될 때마다 아카이브의 완료 여부를 점검합니다. 따라서, 정보가 디스크에 더 빨리 더 자주 기록됩니다.

이 변경에 따른 이점은 시스템이 손상될 경우, 디스크에 저장된 정보(어느 로그 파일이 성공적으로 아카이브되었는지에 대한)가 더 정확하고 DB2가 이미 아카이브된 로그 파일에 대해 아카이브 요청을 다시 발행하지 않아도 됩니다.

특정 로그 파일이 성공적으로 아카이브되었음을 발견한 후 DB2가 수행하는 작업에 대한 변경사항은 없습니다.

이제 DB2가 로그 아카이브의 완료를 더 빨리 발견하고 더 빨리 이름을 바꿉니다. 절단된 사용 중이 아닌 로그 파일은 삭제됩니다. 따라서, 사용 중 로그 경로에 남는 로그 파일의 수는 LOGPRIMARY 데이터베이스 구성 값보다 적을 수 있습니다. 이 경우, 필요하면 DB2가 새 로그 파일을 작성합니다.

이 변경 전에는 데이터베이스를 재시작하면 로그 수가 LOGPRIMARY의 값과 동일하게 줄어들었습니다. 이제, 데이터베이스를 재시작하면 DB2가 먼저 데이터베이스 로그 디렉토리를 조사합니다. 비어 있는 로그 수가 1차 로그 수보다 적으면, DB2가 새 로그를 할당하여 차이를 채웁니다. 데이터베이스 디렉토리에 있는 1차 로그 보다 더 많은 수의 빈 로그가 사용 가능하면, DB2가 데이터베이스 디렉토리에서 사용 가능한 모든 빈 로그로 데이터베이스를 재시작할 수 있도록 합니다.

데이터베이스 종료 후에, 존재하는 모든 2차 로그 파일은 재시작시 사용 중 로그 경로에 남게 됩니다. 사용 중 로그 경로를 지우기 위해 DB2 ARCHIVE LOG 명령을 사용할 수 있습니다.

데이터베이스 복구를 위한 일시중단된 입출력 사용법

db2inidb 유틸리티에 대한 아래 정보는 버전 7.2 새로운 기능 책에 있는 정보를 대신합니다.

db2inidb는 DB2와 함께 제공된 도구로, 응급 복구를 수행하거나 데이터베이스를 롤 포워드 보류 상태가 되도록 할 수 있습니다.

일시중단된 입출력은 데이터베이스를 종료하지 않고 미러를 분할하는 온라인 분할 미러 핸들링을 위한 전체 구현을 제공하여 연속적인 시스템 사용 가능성을 지원합니다. 대형 데이터베이스에서 오프라인 또는 온라인 백업을 수행할 수 없는 경우, 일시중단된 입출력 및 분할 미러 이미지를 사용하여 미러 이미지로부터 백업 또는 시스템 복사를 수행할 수 있습니다.

일시중단된 입출력은 데이터베이스의 분할 미러 이미지를 가져오는 중에 디스크 쓰기를 방지합니다. 온라인 백업 및 리스토어 외에도 모든 데이터베이스 조작은 데이터베이스가 일시중단된 동안 정상적으로 작동해야 합니다. 그러나, 일부 조작은 버퍼 풀 또는 로그 버퍼에서 더티 페이지를 로그로 비워야 하는 경우에 입출력 쓰기의 재개를 대기할 수 있습니다. 이러한 조작은 데이터베이스 입출력이 재개되고 나면 정상적으로 재개되어야 합니다. 원래 일시중단되었던 동일한 연결에서 데이터베이스 입출력이 재개되는 것은 중요합니다. 그렇지 않으면, 더티 페이지를 버퍼 풀에서 디스크로 비워야 하는 경우 후속 연결 시도가 정지될 수 있습니다. 이러한 연결은 데이터베이스 입출력이 재개되면 완료됩니다. 연결 시도가 정지되고, 입출

력을 일시중단할 때 사용한 연결에서 이를 재개할 수 없게 된 경우, RESTART 명령의 WRITE RESUME 옵션을 사용하여 응급 복구를 수행해야 합니다.

파티션된 데이터베이스 환경에서는 모든 파티션에서 동시에 입출력 쓰기를 일시중단할 필요가 없습니다. 하나 이상의 파티션 서브세트를 일시중단하여 분할 미러를 작성함으로써 오프라인 백업을 수행할 수 있습니다. 카탈로그 노드가 서브세트에 포함될 경우에는 이 파티션을 일시중단할 필요가 없습니다.

데이터베이스의 미러링은 주로 데이터베이스 디렉토리와 로컬 데이터베이스 디렉토리의 전체 내용을 복사하는 것입니다. 로컬 데이터베이스 디렉토리인 sqldbdir은 기본 데이터베이스 디렉토리와 같은 레벨의 파일 구조에 위치합니다. 그리고, 로그 디렉토리와 테이블 스페이스 컨테이너가 데이터베이스 디렉토리에 없으면, 이 디렉토리도 복사해야 합니다. 분할 미러 데이터베이스는 이러한 디렉토리 경로에 종속되므로, 이 디렉토리가 복사되는 경로는 기본 시스템의 경로와 동일해야 합니다. 이것은 인스턴스도 동일해야 함을 의미합니다. 이 종속성으로 인해 db2inidb 도구의 새 『relocate』 옵션을 사용하지 않으면 기본 데이터베이스와 같은 시스템에 미러 데이터베이스를 작성할 수 없습니다.

『relocate』 옵션의 목적은 지정된 구성 파일을 사용하여 지정된 시스템에서 시스템을 재배치하는 것입니다. 여기에는 내부 데이터베이스 디렉토리, 컨테이너 디렉토리, 로그 디렉토리, 인스턴스 이름 및 데이터베이스 이름의 변경이 포함됩니다. 데이터베이스 디렉토리, 컨테이너 디렉토리 및 로그 디렉토리가 기본 데이터베이스와 동일한 시스템에 있는 다른 디렉토리 경로로 정상적으로 미러되었음을 가정하여, db2inidb 도구는『relocate』옵션과 함께 사용되어 미러된 데이터베이스의 내부 경로를 갱신할 수 있습니다. 이 옵션이 있는 사용시나리오가 아래에 있습니다.

스토리지 장치가 미러되는 방식에 따라, db2inidb의 사용은 다양합니다. 다음과 같은 사용은 전체 데이터베이스가 스토리지 시스템을 통해 연속적으로 미러됨을 가정합니다.

다중 노드 환경의 경우, 파티션에서 분할 미러를 사용하려면 먼저 모든 파티션에서 db2inidb 도구를 실행해야 합니다. db2inidb 도구는 **db2_all** 명령을 사용하여 모든 파티션에서 동시에 실행될 수 있습니다.

1.

클론 데이터베이스 작성:

여기에서 목표는 기본 데이터베이스의 클론이 다른 시스템에서 사용되도록 하는 것입니다. 다음 프로시저에서는 클론 데이터베이스가 작성되는 방식을 설명합니다.

a. 다음 명령을 입력하여 기본 데이터베이스에서 입출력 쓰기를 일시중단하십시오.

```
db2 set write suspend for database
```

b. 운영 체제 및 디스크 서브시스템 레벨 명령을 사용하여 기본 데이터베이스에서 미러를 분할하십시오. 데이터와 로그를 모두 분할했는지 확인하십시오.

c. 다음 명령을 입력하여 기본 데이터베이스에서 입출력 쓰기를 재개하십시오.

```
db2 set write resume for database
```

명령을 수행한 후에, 기본 데이터베이스는 다시 보통 상태가 되어야 합니다.

d. 다른 시스템에서 기본 데이터베이스의 분할 미러를 마운트하십시오.

e. 다음 명령을 입력하여 다른 시스템에서 데이터베이스 인스턴스를 시작하십시오.

```
db2start
```

f. 다음 명령을 입력하여 DB2 응급 복구를 시작하십시오.

db2inidb *database_name* AS SNAPSHOT

주: 이 명령은 쓰기 일시중단 상태를 제거하고 분할시 발생하는 트랜잭션에 의한 변경사항을 롤백합니다.

이 프로세스를 사용하여 오프라인 백업을 수행할 수도 있지만, 기본 데이터베이스에서 리스토어된 경우에는 로그 체인이 일치하지 않으므로 이 백업을 사용하여 롤 포워드할 수 없습니다.

2.

대기 데이터베이스로 분할 미러 사용:

미러된(대기) 데이터베이스는 로그를 통해 계속 롤 포워드되기 때문에 기본 데이터베이스에서 작성하는 새 로그는 기본 시스템에서 일정하게 페치됩니다. 다음 프로시저는 분할 미러가 대기 데이터베이스로 사용될 수 있는 방식을 설명합니다.

a. 기본 데이터베이스에서 입출력 쓰기를 일시중단하십시오.

```
db2 set write suspend for database
```

b. 운영 체제 및 디스크 서브시스템 레벨 명령을 사용하여 기본 데이터베이스에서 미러를 분할하십시오. 데이터만 분할하고 로그는 분할하지 않았는지 확인하십시오.

c. 기본 데이터베이스에서 입출력 쓰기를 재개하여 다시 정상 처리할 수 있게 하십시오.

```
db2 set write resume for database
```

d. 다른 시스템으로 데이터베이스의 분할 미러를 마운트하십시오.

e. **db2start** 명령을 사용하여 기본 데이터베이스 인스턴스를 시작하십시오.

f. 미러를 롤 포워드 보류 상태로 두고 미러를 롤 포워드하십시오.

db2inidb *database_name* AS STANDBY

주: 이 명령은 쓰기 일시중단 상태를 제거하고 미러된 데이터베이스를 롤 포워드 보류 상태로 둡니다.

g. User Exit 프로그램을 설정하여 로그를 복사하고 기본 시스템에서 로그 파일을 검색하여 최신 로그를 이 미러된 데이터베이스에서 사용 가능하게 하십시오.

h. 데이터베이스를 로그의 끝으로 롤 포워드하십시오.

i. f단계로 이동하여 기본 데이터베이스가 종료될 때까지 이 프로세스를 반복하십시오.

j. AND STOP 옵션을 사용하여 데이터베이스를 로그 끝까지 롤 포워드하여 데이터베이스를 다시 온라인 상태로 만드십시오. 그러면 사용 준비가 된 것입니다.

3.

백업 이미지로 분할 미러 사용:

다음 프로시저는 미러된 데이터베이스를 백업 이미지로 사용하여 기본 데이터베이스에서 리스토어할 수 있게 하는 방법을 설명합니다.

a. **db2stop** 명령을 사용하여 기본 데이터베이스 인스턴스를 중지하십시오.

b. 운영 체제와 디스크 서브시스템 명령을 사용하여 미러된 데이터를 기본 데이터베이스 위에 다시 복사하십시오. 로그 파일은 다시 복사하지 마십시오. 기본 데이터베이스의 로그는 롤 포워드 조작에 사용해야 합니다.

c. **db2start** 명령을 사용하여 기본 데이터베이스 인스턴스를 시작하십시오.

d. 다음 명령을 실행하여 미러된 데이터베이스를 롤 포워드 보류 상태로 두고 쓰기 일시중단 상태를 제거하십시오.

   db2inidb *database_name* AS MIRROR

e. AND STOP 옵션을 사용하여 데이터베이스를 로그의 끝까지 롤 포워드하여 데이터베이스를 다시 온라인 상태로 만드십시오. 그러면 사용 준비가 된 것입니다.

4.

기본 데이터베이스와 동일한 시스템으로 미러 분할:

다음 프로시저에서는 db2inidb 도구의『relocate』옵션을 사용하여 기본 데이터베이스와 동일한 시스템으로 데이터베이스를 미러하는 방법을 설명합니다. 예에서는 데이터베이스가 새 인스턴스에서 사용될 것임을 가정합니다.

a. 현재 시스템에서 새 인스턴스를 작성하십시오.

b. 기본 데이터베이스에서 입출력 쓰기를 일시중단하십시오.

```
db2 set write suspend for database
```

c. 운영 체제 및 디스크 서브시스템 레벨 명령을 사용하여 기본 데이터베이스에서 미러를 분할하십시오.

   주: 데이터베이스 디렉토리, 로컬 데이터베이스 디렉토리, 컨테이너 디렉토리 및 로그 디렉토리를 새 인스턴스로 복사해야 합니다. 컨테이너 디렉토리나 로그 디렉토리가 데이터베이스 디렉토리 아래에 있으면, 데이터베이스 디렉토리와 로컬 데이터베이스 디렉토리만 복사하면 됩니다.

d. 기본 데이터베이스에서 입출력 쓰기를 재개하여 정상 처리로 돌아갈 수 있게 하십시오.

```
db2 set write resume for database
```

e. 다음 정보를 사용하여 구성 파일을 작성하십시오.

DB_NAME=*name*,*optional_new_name*
DB_PATH=*primary_db_dir_path*,*mirrored_db_dir_path*
INSTANCE=*primary_instance*,*mirror_instance*
LOG_DIR=*primary_db_log_dir*,*mirrored_db_log_dir*
CONT_PATH=*primary_db_container_#1_path*,
mirrored_db_container_#1_path ...
CONT_PATH=*primary_db_container_#n_path*,
mirrored_db_container_#n_path
NODENUM=*node_#*

주: LOG_DIR 및 CONT_PATH 필드는 로그 디렉토리 및 컨테이너 디렉토리가 데이터베이스 디렉토리 외부에 있는 경우에만 필요합니다. 지정하지 않는 경우 디폴트값이 0이 되는 NODENUM을 제외하고 다른 모든 필드가 필요합니다.

f. 새로 작성한 인스턴스에서 데이터베이스를 시작하십시오.

```
db2start
```

g. 미러된 데이터베이스를 재배치하고 일시중단 상태를 제거한 후, 미러를 롤 포워드 보류 상태로 두십시오.

```
db2inidb database_name as STANDBY relocate using config_file
```

h. User Exit 프로그램을 설정하여 로그를 복사하고 기본 데이터베이스에서 로그 파일을 검색하여 최신 로그를 이 미러된 데이터베이스에서 사용 가능하게 하십시오.

i. 데이터베이스를 로그의 끝으로 롤 포워드하십시오.

j. h단계로 이동하여 기본 데이터베이스가 종료될 때까지 이 프로세스를 반복하십시오.

k. AND STOP 옵션을 사용하여 데이터베이스를 로그의 끝까지 롤 포워드하여 데이터베이스를 다시 온라인 상태로 만드십시오. 그러면 사용 준비가 된 것입니다.

LOGRETAIN=CAPTURE에서의 새 백업 및 리스토어 동작

데이터베이스의 LOGRETAIN을 CAPTURE로 구성하면 다음 연산을 수행할 수 없습니다.

- 온라인 데이터베이스 백업
- 온라인 또는 오프라인 테이블 스페이스 레벨의 백업
- 온라인 또는 오프라인 테이블 스페이스 레벨의 리스토어

LOGRETAIN이 CAPTURE로 설정되어 있는 동안 생성된 오프라인 백업 이미지를 사용하여 데이터베이스 리스토어 조작을 수행하면 데이터베이스가 롤 포워드 보류 상태에 놓이지 *않게* 됩니다. 또한 LOGRETAIN을 CAPTURE(FixPak 4 이전의 버전 7.2)로 설정하는 동안 생성된 *온라인* 데이터베이스 백업 이미지를 사용하여 데이터베이스 리스토어 연산을 수행할 수 있습니다.

증분 백업 및 복구 - 추가 정보

처리의 두 번째 단계 중에 요청된 리스토어 조작을 수행하는 데 필요한 백업 이미지의 체인을 빌드하기 위해 데이터베이스 실행기록이 쿼리됩니다. 몇몇 이유로 인해 이것이 불가능하여 DB2가 필요한 이미지의 전체 체인을 빌드할 수 없으면, 리스토어 조작이 종료되고 오류 메시지가 리턴됩니다. 이 경우, 자동 증분 리스토어가 불가능하므로 INCREMENTAL ABORT 옵션으로 RESTORE DATABASE 명령을 발행해야 합니다. 그러면 남아 있는 자원을 모두 정리하여 수동 증분 리스토어를 진행할 수 있습니다.

처리의 세 번째 단계 중에 DB2는 생성된 체인에 있는 나머지 백업 이미지를 리스토어합니다. 이 단계 중에 오류가 발생하면, INCREMENTAL ABORT 옵션으로 RESTORE DATABASE 명령을 발행하여 나머지 자원을 정리해야 합니다. 그런 다음, RESTORE 명령을 다시 발행하거나 수동 증분 리스토어를 다시 시도하기 전에 오류를 해결할 수 있는지 판별해야 합니다.

NEWLOGPATH2(현재는 DB2_NEWLOGPATH2라고 함)

NEWLOGPATH2 레지스트리 변수에 대한 참조는 DB2_NEWLOGPATH2로 변경되었습니다.

AIX 또는 Solaris 운영 환경에서 DB2 Data Links Manager용 백업 메소드 선택

Tivoli Storage Manager 시스템 옵션 파일의 *PASSWORDACCESS* 옵션을 설정하기 전에, `/usr/lib`에 `libApiDS.a` 라이브러리 파일에 대한 기호 링크가 포함되어 있는지 확인해야 합니다.

Tivoli Storage Manager -- LAN Free Data Transfer

DB2 Universal Database는 이제 TSM 서버로의 백업 및 리스토어에 대해 Tivoli의 LAN Free Data Transfer 기술을 사용하게 합니다. Tivoli의 ADSM 3.1.x 클라이언트와 함께 DB2 Universal Database의 다음 버전 중 하나를 사용하려는 경우, TSM 서버로 백업하거나 리스토어할 때 문제점이 발생할 수 있습니다.

- AIX용 DB2(32비트)
- Solaris 운영 환경용 DB2(32비트)
- HP-UX용 DB2(32비트)

이러한 문제점이 발생하는 경우, 다음과 같은 단계를 수행하여 정정하십시오.

1. **db2stop** 명령을 실행하십시오.
2. DB2 UDB 서버에서 sqllib/adsm 디렉토리를 찾으십시오.
3. `libtadsm.a`의 백업 사본을 작성하십시오. `libtadsm.a.bak`라는 사본을 작성하면 됩니다.
4. `libadsm.a`를 `libtasdm.a`로 복사하십시오.
5. **db2start** 명령을 실행하십시오.
6. 실패한 백업 또는 리스토어 명령을 다시 실행하십시오.

데이터 이동 유틸리티 안내 및 참조서

이제 익스포트 유틸리티에서 확장 식별 값이 완전히 지원됨

익스포트 유틸리티는 이제 확장 식별 값을 완전히 지원합니다. 이 기능을 이용하려면 클라이언트 및 서버를 FixPak 7 이상과 함께 실행해야 합니다.

익스포트, 임포트 및 로드에 의한 LOB 파일 조절 변경

DB2 UDB는 이제 대형 오브젝트(LOB) 정보를 임포트, 익스포트 및 로드할 때 LOB 위치 지정자(LLS)를 사용합니다. 이것은 다중 LOB를 단일 파일에 저장합니다.

LLS는 LOB 데이터를 파일 내에서 찾을 수 있는 곳을 나타내는 문자열입니다. LLS의 형식은 *filename.ext.nnn.mmm/*이고, 여기서 *filename.ext*는 LOB 파일이 포함될 파일의 이름이고, *nnn*은 파일 내에서 LOB의 오프셋(바이트 단위)이며, `mmm`은 LOB의 길이(바이트 단위)입니다. 예를 들어, `db2exp.001.123.456/`의 LLS는 LOB이 파일 `db2exp.001`에 있고, 오프셋 123바이트에서 시작하며, 길이가 456 바이트임을 나타냅니다. LLS에 표시된 크기가 0이면, LOB이 길이 0을 갖는 것으로 간주됩니다. 길이가 -1이면, LOB이 NULL로 간주되고 파일 이름 및 오프셋은 상관이 없습니다.

lobsinfile 수정자를 사용하여 데이터를 익스포트할 때, LOB이 항상 개별적인 파일에 위치하지 않습니다. 각 LOB 파일에는 여러 개의 LOB이 있을 수 있고 각 LOB 경로당 여러 개의 LOB 파일이 있을 수 있습니다. 이제 데이터 파일은 파일 이름 대신 LLS 레코드를 포함합니다.

임포트 및 로드 기능도 익스포트 기능의 변경사항을 핸들하도록 변경되었습니다. **lobsinfile에 의해 수정** 옵션을 지정하여 데이터를 로드하고 임포트할 때, 해당 LOB 컬럼에 대한 LLS를 예상합니다. LOB 컬럼에 대해 LLS 이외의 어떤 것이 있는 경우, 데이터베이스는 그것을 LOB 파일로 취급하고 전체 파일을 LOB으로 로드합니다.

IXF 고려사항

세 개의 새 IXF 데이터 유형이 있습니다. 이 세 유형은 LLS에 의해 표현될 경우 문자 대형 오브젝트(CLOB), 2진 대형 오브젝트(BLOB) 및 2바이트 문자 대형 오브젝트(DBCLOB)에 해당합니다. 이 데이터 유형의 값은 각각 964, 960 및 968 입니다.

IXF 파일은 이제 각 LOB 컬럼이 자신의 D 레코드를 가지도록 요청합니다. 이것은 익스포트 도구로 자동 작성되지만, IXF 파일을 작성하기 위해 써드 파티 유틸리티를 사용 중인 경우 수동으로 작성해야 합니다. 또한 테이블의 각 LOB에 대해 LLS가 필요하며 널(NULL)이 아닌 LOB에 대해서는 필요하지 않습니다. LOB 컬럼이 널(NULL)인 경우, 널(NULL) LOB를 나타내는 LLS를 작성해야 합니다.

유틸리티 임포트, 익스포트 및 로드를 위한 코드 페이지 지원

이제 임포트, 익스포트 및 로드 유틸리티를 사용해서 데이터를 새 중국어 코드 페이지 GB 18030(코드 페이지 ID 5488)과 새 일본어 코드 페이지 ShiftJIS X213(코드 페이지 ID 1394)에서 DB2 UDB 유니코드 데이터베이스로 전송할 수 있습니다. 또한 익스포트 유틸리티를 사용해서 데이터를 DB2 UDB 유니코드 데이터베이스에서 GB 18030 또는 ShiftJIS X0213 코드 페이지로 전송할 수 있습니다.

예를 들어, 다음 명령은 리모트로 연결된 클라이언트에 상주하는 Shift_JISX0213 데이터 파일인 `u/jp/user/x0213/data.del`을 MYTABLE로 로드합니다.

```
db2 load client from /u/jp/user/x0213/data.del
    of del modified by codepage=1394 insert into mytable
```

여기서, MYTABLE은 DB2 UDB 유니코드 데이터베이스에서의 위치입니다.

제 2 장 임포트

버퍼된 삽입이 있는 임포트 사용

이 섹션 끝의 주는 다음과 같아야 합니다.

주: EEE를 제외한 모든 환경에서 INSERT_UPDATE 매개변수가 지정된 임포트 조작 중에 버퍼된 삽입 기능이 사용 불가능합니다.

제 3 장 로드

로드 조작 이후 보류 상태

이 섹션의 마지막 문단에서 처음 두 문장이 다음과 같이 변경되었습니다.

로드 프로세스와 관련된 네 번째 가능한 상태(점검 보류 상태)는 참조 및 점검 제한조건, DATALINKS 제한조건, AST 제한조건 또는 생성된 컬럼 제한조건과 관련됩니다. 예를 들어, 기존 테이블이 종속 테이블의 외부 키가 참조하는 기본 키가 들어 있는 상위 테이블일 경우, 상위 테이블에서 데이터를 바꾸면 두 테이블 (테이블 스페이스가 아닌) 모두 점검 보류 상태가 됩니다.

로드 제한사항 및 한계

다음의 제한사항이 생성된 컬럼과 로드 유틸리티에 적용됩니다.

- 생성된 컬럼이 인덱스의 "포함 컬럼"이 아니거나 `generatedoverride` 파일 유형 수정자가 사용되지 않은 경우, 생성된 컬럼이 고유 인덱스에 있는 테이블을 로드할 수 없습니다. 이 수정자가 사용된 경우, 컬럼의 모든 값이 입력 데이터 파일에 제공될 것으로 예상됩니다.
- `generatedoverride` 파일 유형 수정자가 사용되지 않은 경우, 파티션 키에 생성된 컬럼이 있는 테이블을 로드할 수 없습니다. 이 수정자가 사용된 경우, 컬럼의 모든 값이 입력 데이터 파일에 제공될 것으로 예상됩니다.

totalfreespace 파일 유형 수정자

`totalfreespace` 파일 유형 수정자(LOAD)가 0에서 2 147 483 647 사이의 값을 승인하도록 수정되었습니다.

제 4 장 자동 로드 프로그램

인증이 YES로 설정된 경우 자동 로드 프로그램 수행에 rexecd 필요

자동 로드 프로그램 옵션 섹션에서 다음 참고가 AUTHENTICATION 및 PASSWORD 매개변수 설명에 추가됩니다.

Linux 환경에서 인증 옵션을 YES로 설정하여 자동 로드 프로그램을 실행할 경우, 모든 머신에서 rexecd를 사용해야 합니다. rexecd가 사용되지 않으면, 다음 오류 메시지가 생성됩니다.

```
openbreeze.torolab.ibm.com: Connection refused
SQL6554N  리모트 프로세스 실행을 시도하는 중 오류가 발생했습니다.
```

다음 오류 메시지는 db2diag.log 파일에 생성됩니다.

```
2000-10-11-13.04.16.832852   Instance:svtdbm   Node:000
PID:19612(db2atld)   Appid:
oper_system_services  sqloRemoteExec   Probe:31
```

4.3.3 이전 버전의 AIX 시스템에서 포크(Fork) 중에 자동 로드 프로그램이 정지할 수도 있음

자동 로드 프로그램은 멀티스레드된 프로그램이고 스레드 중 하나가 다른 프로세스를 포크 해제합니다. 하위 프로세스의 포크는 상위 메모리의 이미지를 하위에서 작성되게 합니다.

AIX 4.3.3 이전의 AIX 시스템에서 libc.a가 동일한 프로세스 내의 힙에서 메모리를 할당하는 여러 스레드를 관리하기 위해 사용하는 잠금이 비포크 스레드에 의해 보류됩니다. 비포크 스레드가 하위 프로세스에는 없으므로 이 잠금은 하위 프로세스에서 해제되지 않고 때때로 상위 프로세스가 정지하게 됩니다.

AIX 4.3.3에서는 포크 중에 자동 로드 프로그램을 정지시킬 가능성이 있는 libc 문제점의 수정사항을 제공합니다.

부록 C 익스포트/임포트/로드 유틸리티 파일 형식

다음 갱신사항이 이 부록에 추가되었습니다.

익스포트, 임포트 및 로드 유틸리티는 비유니코드 데이터베이스에 연결된 유니코드 클라이언트에서 사용될 때 지원되지 않습니다. 유니코드 클라이언트 파일은 유니코드 클라이언트가 유니코드 데이터베이스에 연결된 경우에만 지원됩니다.

복제 안내 및 참조서

복제 및 비IBM 서버

DataJoiner 버전 2 이상을 사용하여 Informix, Microsoft SQL Server, Oracle, Sybase 및 Sybase SQL Anywhere와 같은 비IBM 서버와의 사이에 데이터를 복제해야 합니다. DB2 Relational Connect 버전 7에는 갱신 기능이 없으므로 이 복제 유형으로 관계 연결 기능을 사용할 수 없습니다. 또한, DJRA(DataJoiner Replication Administration)를 사용하여 기존의 모든 DB2 및 DataJoiner 버전에 대해 모든 플랫폼(AS/400, OS/2, OS/390, UNIX 및 Windows)에서 이종 복제를 관리해야 합니다.

Windows 2000에서의 복제

DB2 DataPropagator 버전 7은 Windows 2000 운영 체제와 호환할 수 있습니다.

SQL 파일 저장시 알려진 오류

DB2 Connect Personal Edition에서 제어 센터를 사용하는 경우, SQL 파일을 저장할 수 없습니다. SQL 파일을 저장하려고 하면 DAS(Database Administration Server)가 실행 중이 아님을 나타내는 오류 메시지가 표시됩니다. 실제로 DAS는 DB2 Connect PE를 포함하고 있지 않으므로 사용할 수 없습니다.

DB2 유지보수

복제 환경에서 사용하는 다양한 DB2 제품용 최신 DB2 유지보수를 설치하는 것이 좋습니다.

웹의 Data Difference 유틸리티

ftp://ftp.software.ibm.com/ps/products/datapropagator/fixes/에서 Data Difference 유틸리티(DDU)를 다운로드할 수 있습니다. DDU는 동일한 파일의 두 버전을 비교하여 차이를 나타내는 출력 파일을 만드는 샘플 유틸리티입니다. 자세한 내용은 샘플 유틸리티에 포함된 README 파일을 참조하십시오.

제 3 장 데이터 복제 시나리오

복제 시나리오

새로운 이질형 데이터 복제 시나리오에 대해서는 DataPropagator 웹 사이트 (http://www.ibm.com/software/data/dpropr/)의 라이브러리 페이지를 참조하십시오. 이 시나리오의 단계를 수행하여 AIX의 Oracle 데이터베이스에 있는 복제-소스 테이블에서 Windows NT용 DB2에 있는 데이터베이스의 목표 테이블로 변경사항을 복사하십시오. 이 시나리오에서는 DB2 DataJoiner 복제 관리(DJRA) 도구, Capture 트리거, Apply 프로그램 및 DB2 DataJoiner를 사용합니다.

책의 44 페이지에서 암호 파일 작성을 위한 6단계는 다음과 같습니다.

6단계: 암호 파일 작성

Apply 프로그램은 소스 서버에 연결되어야 하므로, 사용자 인증을 위한 암호 파일을 작성해야 합니다. Apply 프로그램을 수행할 사용자 ID가 암호 파일을 읽을 수 있도록 하십시오.

암호 파일 작성하기

1. Windows NT 명령 프롬프트 창에서 `C:\scripts` 디렉토리를 변경하십시오.
2. `DEPTQUAL.PWD` 디렉토리에서 새 파일을 작성하십시오. 메모장과 같은 텍스트 편집기를 사용하여 이 파일을 작성할 수 있습니다. 암호 파일의 이름 지정 규칙은 *applyqual*.pwd입니다. 여기서, *applyqual*은 복사 작업 내역 세트를 작성할 때 사용했던 Apply 규정자의 대소문자와 값이 *반드시* 일치해야 하는 대소문자 구분 문자열입니다. 이 시나리오의 경우, Apply 규정자는 DEPTQUAL 입니다.

주: 버전 5 DB2 DataPropagator의 파일 이름 지정 규칙도 지원됩니다.

3. 암호 파일 내용에는 다음 두 가지 형식이 있습니다.

   ```
   SERVER=server USER=userid PWD=password
   ```

   여기서,

   server

   복사 작업 내역 세트 테이블에서 표시된 그대로의 소스, 목표 또는 제어 서버의 이름. 이 시나리오의 경우, 이들 이름들은 SAMPLE 및 COPYDB 입니다.

   userid

   특수 데이터베이스를 관리할 때 사용하려는 사용자 ID. 이 값은 Windows NT 및 UNIX 운영 체제에서 대소문자가 구별됩니다.

   password

   해당 사용자 ID와 연관된 암호. 이 값은 Windows NT 및 UNIX 운영 체제에서 대소문자가 구별됩니다.

   이 파일에 공백행이나 주석행을 두지 마십시오. 서버 이름, 사용자 ID 및 암호 정보만을 추가하십시오.

4. 암호 파일 내용은 다음과 비슷해야 합니다.

   ```
   SERVER=SAMPLE USER=subina PWD=subpw
   SERVER=COPYDB USER=subina PWD=subpw
   ```

DB2 인증 및 보안에 대해서는 *IBM DB2 관리 안내서*를 참조하십시오.

제 5 장 복제 계획

테이블 및 컬럼 이름

복제에서는 테이블 및 컬럼 이름에 공백을 지원하지 않습니다.

DATALINK 복제

DATALINK 복제는 버전 7.1 FixPak 1의 일부로 Solaris 운영 환경에서 사용 가능합니다. 여기에는 소스 및 목표 DATALINK 파일 시스템에서 실행되고 제공된

파일의 최종 수정 시간을 표시하는 MDTM(modtime) 명령을 지원하는 FTP 디먼이 필요합니다. 버전 2.6의 Solaris 운영 환경이나 MDTM에 대한 FTP 지원이 포함되지 않는 모든 기타 버전을 사용하는 경우, WU-FTPD와 같은 추가 소프트웨어가 필요합니다.

AS/400의 DB2 데이터베이스와 기타 플랫폼의 DB2 데이터베이스 사이에서 DATALINK 컬럼을 복제할 수 없습니다.

AS/400 플랫폼에서는 DATALINK 값의 속성인 ″주석″의 복제가 지원되지 않습니다.

AIX 4.2를 수행하는 경우, 기본 User Exit 프로그램(ASNDLCOPY)을 수행하기 전에 APAR IY03101에 대한 PTF를 설치해야 합니다(AIX 4210-06 RECOMMENDED MAINTENANCE FOR AIX 4.2.1). 이 PTF에는 FTP 디먼의 ″modtime/MDTM″ 명령에 대한 Y2K 수정이 들어 있습니다. 수정을 확인하려면, ″modtime <file>″ 명령에서 리턴된 최종 수정 시간을 확인하십시오. 여기서, <file>은 2000년 1월 1일 이후에 수정된 파일입니다.

목표 테이블이 외부 CCD 테이블인 경우, DB2 DataPropagator는 ASNDLCOPY 루틴을 호출하여 DATALINK 파일을 복제합니다. 각 프로그램의 소스 코드에 대한 서론 섹션에서 ASNDLCOPY 및 ASNDLCOPYD 프로그램의 사용 방법에 대한 최신 정보를 참조하십시오. 다음의 제한사항이 적용됩니다.

- 내부 CCD 테이블에는 DATALINK 값이 아닌 DATALINK 표시기가 포함될 수 있습니다.
- 압축 외부 CCD 테이블에 DATALINK 값이 포함될 수 있습니다.
- 비압축 CCD 목표 테이블에는 DATALINK 컬럼이 포함될 수 없습니다.
- 소스 및 목표 서버가 동일한 경우, 복사 작업 내역 세트에는 DATALINK 컬럼이 있는 구성원이 포함되면 안됩니다.

LOB 제한사항

압축 내부 CCD 테이블에는 LOB 컬럼이나 LOB 표시기에 대한 참조가 포함될 수 없습니다.

복제 계획

65 페이지에서 ″연결성″에는 다음의 사실이 포함되어야 합니다.

```
Apply 프로그램이 제어 서버에 연결될 수 없는 경우,
Apply 프로그램은 종료됩니다.
```

AS/400에 데이터 블로킹을 사용할 때에는 간격 중에 복제한 총 데이터 양이 ″4 백만 행″을 초과하지 않도록 해야 합니다. 책의 69 페이지에 언급된 ″4MB″는 잘못된 정보입니다.

제 6 장 복제 환경 설정

update-anywhere 전제조건

충돌 발견 및 복사 작업 내역 세트에서 150개 이상의 복사 작업 내역 세트 구성원으로 update-anywhere 복제를 설정하려면, 다음의 DDL을 수행하여 제어 서버에 ASN.IBMSNAP_COMPENSATE 테이블을 작성해야 합니다.

```
CREATE TABLE ASN.IBMSNAP_COMPENSATE (
     APPLY_QUAL char(18) NOT NULL,
     MEMBER SMALLINT,
     INTENTSEQ CHAR(10) FOR BIT DATA,
     OPERATION CHAR(1));
```

복제 환경 설정

95 페이지, ″CD 테이블, 인덱스 및 테이블 스페이스 이름 사용자 정의″에서 DPREPL.DFT 파일이 \sqllib\bin 디렉토리 또는 \sqllib\java 디렉토리에 있다고 서술하고 있습니다. 이는 잘못된 서술로, DPREPL.DFT는 \sqllib\cc 디렉토리에 있습니다.

| 128 페이지의 보유 제한 설명에서 Capture warm starts 또는 Capture prune 명령을 사용할 경우에만 행을 제거하기 위해 보유 제한을 사용해야 함을 명시해야 합니다. auto-pruning 옵션과 함께 Capture를 시작하면 행을 제거할 때 보유 제한을 사용하지 않습니다.

제 8 장 문제점 판별

복제 분석기가 Windows 32비트 시스템과 AIX에서 수행됩니다. AIX에서 분석기를 수행하려면, PATH 환경 변수에서 `sqllib/bin` 디렉토리가 `/usr/local/bin` 앞에 나타나도록 하여 `/usr/local/bin/analyze`와의 충돌을 방지해야 합니다.

복제 분석기에는 CT와 AT의 두 가지 추가 선택 키워드가 있습니다.

CT=*n* 캡처 추적 테이블에서 *n*일 이후의 항목들만을 표시합니다. 이 키워드는 선택적입니다. 이 키워드를 지정하지 않으면, 기본값은 7일입니다.

AT=*n* 적용 추적 테이블에서 *n*일보다 새로운 항목들만을 표시합니다. 이 키워드는 선택적입니다. 이 키워드를 지정하지 않으면, 기본값은 7일입니다.

예:

```
analyze mydb1 mydb2 f=mydirectory ct=4 at=2 deepcheck q=applyqual1
```

복제 분석기의 경우, 다음의 키워드 정보가 갱신됩니다.

deepcheck

분석기가 CD 및 UOW 테이블 제거 정보, OS/390용 DB2 테이블 스페이스 파티션 및 압축 세부사항, 복사 작업 내역 키, 복사 작업 내역 타임라인 및 복사 작업 내역 세트 SQL문 오류와 관련된 목표 인덱스의 분석을 비롯한 보다 완전한 분석을 수행하도록 지정합니다. 이 분석에는 모든 서버가 포함됩니다. 이 키워드는 선택적입니다.

lightcheck

보고서에서 ASN.IBMSNAP_SUBS_COLS 테이블의 모든 컬럼 세부사항, 복사 작업 내역 오류 또는 비정상 또는 생략 및 잘못되었거나 비효율적인 인덱스 정보를 제외하도록 지정합니다. 이러한 정보의 제거는 자원을 절약하고 보다 작은 HTML 출력 파일을 작성합니다. 이 키워드는 선택사항이며 deepcheck 키워드와 함께 사용할 수 없습니다.

분석기 도구는 AS/400 플랫폼에서의 복제를 위해 PTF에서 사용할 수 있습니다. 이들 도구는 복제 환경에 대한 정보를 수집하고 문제점 판별을 돕기 위해 IBM 서비스 담당자에게 보낼 수 있는 HTML 파일을 작성합니다. AS/400 도구를 가져

오려면, 적합한 PTF를 다운로드하십시오. 예를 들어, 제품 5769DP2의 경우, PTF SF61798이나 해당 최신 교체 PTF를 다운로드해야 합니다.

″문제점 해결″ 섹션에 다음의 문제점과 해결책을 추가하십시오.

문제점: Apply 프로그램이 변경사항을 복제하지 않고 루프됩니다. Apply 트레일 테이블에 STATUS=2가 표시됩니다.

복사 작업 내역 세트에는 여러 소스 테이블이 포함됩니다. 세트에 있는 한 소스 테이블의 중요점 처리를 개선하기 위해, 다른 복사 작업 내역 세트의 해당 소스 테이블에 대해 내부 CCD 테이블이 정의됩니다. 소스 테이블 갱신이 이루어지지만 내부 CCD 테이블을 채우는 Apply 프로세스는 비동기로 수행됩니다(예를 들어, Apply 프로그램이 시작되지 않거나 심지어 트리거되지 않습니다). 소스 테이블에서 목표 테이블로 갱신사항을 복제하는 Apply 프로그램은 내부 CCD 테이블이 갱신되기를 기다리기 때문에 루프됩니다.

루핑을 중지하려면, 내부 CCD 테이블에 대해 Apply 프로그램을 시작하십시오(또는 복제를 야기한 이벤트를 트리거하십시오). Apply 프로그램은 내부 CCD 테이블을 채우고 루핑 Apply 프로그램이 모든 소스 테이블 변경사항을 처리하도록 합니다.

여러 Apply 프로그램이 상주하는 내부 CCD 테이블과 함께 소스 테이블이 포함되어 있는 복사 작업 내역 세트의 경우 유사한 상황이 발생할 수 있습니다.

제 9 장 AS/400용 Capture 및 Apply

178 페이지에서 ″작업 관리에 대한 주의사항″은 다음과 같아야 합니다.

```
기본 정의를 변경하거나 사용자 고유 정의를 제공할 수 있습니다.
자신의 서브시스템 설명을 작성하는 경우, 그 이름을 서브시스템
QZSNDPR로 지정하고, 이를 QDPR 이외의 라이브러리에서 작성해야 합니다.
해당 정의 변경에 대한 자세한 정보는
"OS/400 Work Management V4R3", SC41-5306을 참조하십시오.
```

178 페이지의 ″AS/400용 DB2 DataPropagator의 설치 확인 및 사용자 정의″에 다음의 내용을 추가하십시오.

트랜잭션의 수가 많아 잠금 경합에 문제점이 발생할 경우, 기본 시간종료 대기값을 30에서 120으로 높일 수 있습니다. Capture 작업이 시작될 때마다 작업을 변경하거나 다음 프로시저를 사용하여 서브시스템에서 실행되는 모든 작업의 시간종료 대기값을 변경할 수 있습니다.

1. 다음 명령을 발행하여 QGPL/QBATCH를 복제함으로써 새 클래스 오브젝트를 작성하십시오.

```
CRTDUPOBJ OBJ(QBATCH) FROMLIB(QGPL) OBJTYPE(*CLS)
          TOLIB(QDPR) NEWOBJ(QZSNDPR
```

2. 새로 작성된 클래스의 시간종료 대기값을 변경하십시오(예를 들면, 300초로).

```
CHGCLS CLS(QDPR/QZSNDPR) DFTWAIT(300)
```

3. 새로 작성된 클래스를 사용하려면 서브시스템 설명 QDPR/QZSNDPR에 있는 라우팅 항목을 갱신하십시오.

```
CHGRTGE SBSD(QDPR/QZSNDPR) SEQNBR(9999) CLS(QDPR/QZSNDPR
```

194 페이지의 ″delete journal receiver exit 루틴 사용″에는 다음과 같은 문장이 포함되어야 합니다. delete journal receiver exit 루틴에 대한 등록을 제거할 경우 소스 테이블에 사용된 모든 저널에 DLTRCV(*NO)가 있어야 합니다.

195 페이지에서 ADDEXITPGM 명령 매개변수는 다음과 같아야 합니다.

```
ADDEXITPGM EXITPNT(QIBM_QJO_DLT_JRNRCV)
              FORMAT(DRCV0100)
              PGM(QDPR/QZSNDREP)
              PGMNBR(*LOW)
              CRTEXITPNT(*NO)
              PGMDTA(65535 10 QSYS)
```

제 10 장 OS/390용 Capture 및 Apply

제 10 장에서 다음 단락이 갱신되었습니다.

OS/390용 DB2 DataPropagator의 전제조건

OS/390용 DB2 DataPropagator 버전 7(V7)을 수행하려면 OS/390용 DB2 버전 5, OS/390용 DB2 버전 6 또는 OS/390용 DB2 버전 7이 있어야 합니다.

OS/390에서의 유니코드 및 ASCII 코드화 체계

OS/390용 DB2 DataPropagator V7은 유니코드 및 ASCII 코드화 체계를 지원합니다. 새로운 코드화 체계를 사용하려면 OS/390용 DB2 V7이 필요하며 DB2 DataPropagator 소스, 목표 및 제어 테이블을 다음 섹션에서 설명하는 것처럼 수동으로 작성하거나 변환해야 합니다. 그러나 기존 복제 환경에서는 코드화 체계를 수정하지 않아도 OS/390용 DB2 DataPropagator V7로 작업할 수 있습니다.

코드화 체계 선택

소스, CD 및 목표 테이블에서 같은 코드화 체계를 사용하는 경우, 복제 환경에서 데이터 변환의 필요성을 최소화할 수 있습니다. 테이블에 대한 코드화 체계를 선택할 때 단일 CCSID 규칙을 따르십시오. 테이블 스페이스의 문자 데이터를 ASCII, 유니코드 또는 EBCDIC으로 코드화할 수 있습니다. 테이블 스페이스의 모든 테이블은 동일한 코드화 체계를 사용해야 합니다. SQL문의 모든 테이블 코드화 체계는 동일해야 합니다. 또한 뷰 및 조인에서 사용하는 모든 테이블은 동일한 코드화 체계를 사용해야 합니다.

단일 CCSID 규칙을 따르지 않으면 DB2에서 위반을 발견하여 바인드나 실행 시에 SQLCODE -873을 리턴합니다. ASCII 또는 유니코드를 어떤 테이블에 사용해야 하는지는 클라이언트/서버 구성에 따라 달라집니다. 특히 테이블에 대한 코드화 체계를 선택할 때 이 규칙을 따르십시오.

- OS/390용 DB2의 소스 또는 목표 테이블은 EBCDIC, ASCII 또는 유니코드일 수 있습니다. 이들 테이블은 지원되는 DBMS(DB2 계열 또는 DataJoiner가 있는 비DB2 DBMS)에서 동일하거나 다른 코드화 체계를 사용하는 테이블 간에 복사될 수 있습니다.
- OS/390용 DB2 소스 서버에서 한 서버의 모든 CD, UOW, 레지스터 및 제거 제어 테이블은 동일한 코드화 체계를 사용해야 합니다. 일관성 유지를 위해 항상 명시적으로 코드화 체계를 지정하십시오.
- 동일한 제어 서버의 모든 제어 테이블(ASN.IBMSNAP_SUBS_xxxx)은 동일한 코드화 체계를 사용해야 합니다.
- 다른 제어 테이블은 다른 코드화 체계를 사용할 수 있지만 ASN. IBMSNAP_CRITSEC 테이블은 EBCDIC을 사용하는 것이 좋습니다.

코드화 체계 설정

테이블에 대해 적절한 코드화 체계를 지정하려면 테이블을 생성할 때 사용한 SQL을 수정합니다.

- 적합한 코드화 체계를 사용하여 소스 및 목표 테이블을 작성하거나 기존 목표 및 소스 테이블의 코드화 체계를 변경하십시오. 기존 테이블의 코드화 체계를 변경하기 전에 Capture 및 Apply 프로그램을 중지하는 것이 좋으며 Capture 프로그램을 콜드 스타트 후 Apply 프로그램을 다시 시작합니다. 기존 테이블의 코드화 체계를 변경하려면 다음을 수행하십시오.
  1. Reorg 유틸리티를 사용하여 기존 테이블을 복사합니다.
  2. 기존 테이블을 삭제합니다.
  3. 새로운 코드화 체계를 지정하는 테이블을 다시 작성합니다.
  4. 로드 유틸리티를 사용하여 이전 데이터를 새 테이블로 로드합니다.

  Reorg 및 로드 유틸리티에 대한 자세한 정보는 *OS/390용 DB2 Universal Database 유틸리티 안내서 및 참조서*를 참조하십시오.
- 적절한 코드화 체계를 사용하여 새로운 제어 테이블을 작성하거나 기존 테이블의 코드화 체계를 수정하십시오.

  DPCNTL.MVS는 OS/390용 DB2의 `sqllib\samples\repl`을 포함하고 있으며 제어 테이블을 작성하는 몇 개의 CREATE TABLE문이 들어 있습니다. ASCII 또는 유니코드가 필요한 테이블(예: ASN.IBMSNAP_REGISTER 및 ASN.IBMSNAP_PRUNCNTL)의 경우, 다음 예와 같이 CCSID ASCII 또는 CCSID UNICODE 키워드를 추가하십시오.

```
CREATE TABLE ASN.IBMSNAP_PRUNCNTL (
  TARGET_SERVER      CHAR( 18)              NOT NULL,
  TARGET_OWNER       CHAR( 18)              NOT NULL,
  TARGET_TABLE       CHAR( 18)              NOT NULL,
  SYNCHTIME          TIMESTAMP,
  SYNCHPOINT         CHAR( 10)              FOR BIT DATA,
  SOURCE_OWNER       CHAR( 18)              NOT NULL,
  SOURCE_TABLE       CHAR( 18)              NOT NULL,
  SOURCE_VIEW_QUAL   SMALLINT               NOT NULL,
  APPLY_QUAL         CHAR( 18)              NOT NULL,
  SET_NAME           CHAR( 18)              NOT NULL,
  CNTL_SERVER        CHAR( 18)              NOT NULL,
  TARGET_STRUCTURE   SMALLINT               NOT NULL,
```

```
  CNTL_ALIAS         CHAR(  8)
        )  CCSID UNICODE
           DATA CAPTURE CHANGES
           IN TSSNAP02;
```

기존 제어 테이블 및 CD 테이블을 수정하려면 Reorg 및 로드 유틸리티를 사용하십시오.

- 새로운 복제 소스나 복사 작업 내역 세트를 작성할 경우, 관리 도구로 만든 SQL 파일을 수정하여 적절한 코드화 체계를 지정하십시오. SQL에는 복제 소스 및 복사 작업 내역 세트용 CD 및 목표 테이블을 작성할 때 사용되는 몇 개의 CREATE TABLE문이 있습니다. 적절한 위치에 CCSID ASCII 또는 CCSID UNICODE 키워드를 추가하십시오. 예를 들어, 다음과 같습니다.

```
CREATE TABLE   user1.cdtable1 (
  employee_name   varchar,
  employee_age    decimal
    )  CCSID UNICODE;
```

*OS/390용 DB2 UDB SQL 참조서*에는 CCSID에 대한 자세한 내용이 있습니다.

제 11 장 UNIX 플랫폼용 Capture 및 Apply

UNIX 및 Windows용 Capture 및 Apply에 대한 환경 변수 설정

기본 코드 페이지 값 이외의 코드 페이지를 사용하여 소스 데이터베이스를 작성한 경우, DB2CODEPAGE 환경 변수를 해당 코드 페이지로 설정합니다. DB2CODEPAGE를 설정하기 전에 코드 페이지 값을 가져오는 자세한 정보는 *DB2 관리 안내서*를 참조하십시오. Capture는 데이터를 캡처할 데이터베이스와 같은 코드 페이지에서 수행해야 합니다. DB2는 Capture가 수행되고 있는 활동 중인 환경에서 Capture 코드 페이지를 가져옵니다. DB2CODEPAGE가 설정되지 않으면 DB2는 운영 체제에서 코드 페이지 값을 가져옵니다. 데이터베이스를 작성할 때 기본 코드 페이지를 사용한 경우, 운영 체제에서 가져온 값이 Capture에 대해 올바른 값입니다.

제 14 장 테이블 구조

339 페이지에서 값 ″2″에 대한 STATUS 컬럼 설명에 다음의 문장을 추가하십시오.

내부 CCD 테이블을 사용하며 Apply trail 테이블의 상태 컬럼에 반복적으로 ″2″ 값이 표시되는 경우, ″제 8 장: 문제점 판별″로 가서 ″문제점: Apply 프로그램이 변경사항을 복제하지 않고 루프됩니다. Apply trail 테이블에 STATUS=2가 표시됩니다″를 참조하십시오.

제 15 장 Capture 및 Apply 메시지

ASN0017E 메시지는 다음과 같아야 합니다.

ASN0017E

Capture 프로그램에서 심각한 내부 오류가 발생했으며 올바른 오류 메시지를 발행할 수 없습니다. 루틴 이름은 ″routine″입니다. 리턴 코드는 ″return_code″입니다.

ASN1027S 메시지를 추가해야 합니다.

ASN1027S

대형 오브젝트(LOB) 컬럼이 너무 많이 지정되었습니다. 오류 코드는 ″<error_code>″입니다.

설명: 복사 작업 내역 세트 구성원에 대해 대형 오브젝트(BLOB, CLOB 또는 DBCLOB) 컬럼이 너무 많이 지정되었습니다. 허용되는 최대 컬럼 수는 10입니다.

사용자 조치: 복사 작업 내역 세트 구성원에서 초과된 대형 오브젝트 컬럼을 제거하십시오.

ASN1048E 메시지는 다음과 같아야 합니다.

ASN1048E

Apply 순환 실행에 실패했습니다. 자세한 내용은 Apply 추적 테이블을 참조하십시오. ″<text>″

설명: Apply 순환에 실패했습니다. 메시지에서 ″<text>″는 ″<target_server>″, ″<target_owner, target_table, stmt_number>″ 및 ″<cntl_server>″를 식별합니다.

사용자 조치: 감사 추적 테이블의 APPERRM 필드를 확인하여 Apply 순환이 실패한 이유를 살펴보십시오.

부록 A 응용프로그램 내에서 Capture 및 Apply 프로그램 시작

책의 399 페이지에서 Capture 및 Apply 프로그램을 시작하는 샘플 루틴의 주석에 몇 개의 오류가 나타나지만 샘플의 코드는 올바릅니다. 주석은 샘플의 뒤쪽 부분이 Capture 매개변수에 속하는 것으로 나타내지만 샘플의 뒤쪽 부분은 Apply 매개변수에 속합니다.

다음의 디렉토리에서 Apply 및 Capture API의 샘플과 각각의 makefile을 가져올 수 있습니다.

```
NT의 경우 - sqllib\samples\repl
UNIX의 경우  - sqllib/samples/repl
```

시스템 모니터 안내 및 참조서

db2ConvMonStream

사용 주의사항에서 스냅샷 변수 데이터 스트림 유형 SQLM_ELM_SUBSECTION의 구조는 sqlm_subsection이어야 합니다.

문제점 해결 안내서

사용자가 로그온되지 않았을 때 Windows 95, Windows 98 및 Windows ME에서 DB2 시작

Windows 95, Windows 98 또는 Windows Millennium Edition(ME) 환경에서 db2start 명령이 성공하려면, 다음 중 하나를 수행해야 합니다.

- Windows 로그온 창이나 Microsoft 네트워킹 로그온 창을 사용하여 로그온하십시오
- **db2logon** 명령을 발행하십시오(207 페이지의 1에서 **db2logon** 명령에 대한 정보를 참조).

또한, 로그온 중에 지정되었거나 **db2logon** 명령에 대해 지정된 사용자 ID는 DB2의 요구사항을 만족시켜야 합니다(208 페이지의 2 참조).

db2start 명령이 시작되면, 먼저 사용자가 로그온되었는지 확인합니다. 사용자가 로그온된 경우, **db2start** 명령은 이 사용자의 ID를 사용합니다. 사용자가 로그온되지 않은 경우, **db2start** 명령은 **db2logon** 명령이 수행되었는지 확인하고 수행되었으면, **db2start** 명령은 **db2logon** 명령에 대해 지정된 사용자 ID를 사용합니다. **db2start** 명령이 유효한 사용자 ID를 찾을 수 없는 경우, 이 명령은 종료됩니다.

Windows 95, Windows 98 및 Windows ME에서 DB2 Universal Database 버전 7 설치 중에, 기본값으로 설치 소프트웨어는 시스템이 부트될 때 **db2start** 명령을 수행하는 단축 아이콘을 시작 폴더에 추가합니다. (자세한 정보는 207 페이지의 1을 참조하십시오.) 시스템의 사용자가 로그온되지도 않고 **db2logon** 명령을 발행하지도 않은 경우, **db2start** 명령은 종료됩니다.

사용자가 Windows나 네트워크에 정상적으로 로그온하지 않았으면, 다음과 같이 배치 파일에서 명령을 수행하여 **db2start** 명령보다 먼저 **db2logon** 명령을 발행하기 위한 요구사항을 숨길 수 있습니다.

1. **db2logon** 명령 다음에 **db2start.exe** 명령을 발행하는 배치 파일을 작성하십시오. 예를 들어, 다음과 같습니다.

```
@echo off
db2logon  db2local /p:password
db2start
cls
exit
```

2. 배치 파일의 이름을 db2start.bat로 지정하고 DB2를 설치한 드라이브와 경로에 있는 /bin 디렉토리에 저장하십시오. 운영 체제가 배치 파일로의 경로를 찾을 수 있도록 이 위치에 배치 파일을 저장합니다.

   DB2가 설치된 드라이브와 경로는 DB2 레지스트리 변수 DB2PATH에 저장됩니다. DB2를 설치한 드라이브와 경로를 찾으려면, 다음의 명령을 발행하십시오.

   ```
   db2set  -g  db2path
   ```

   db2set 명령이 c:\sqllib 값을 리턴하는 것으로 가정하십시오. 이 경우, 다음과 같이 배치 파일을 저장하게 됩니다.

   ```
   c:\sqllib\bin\db2start.bat
   ```

3. 시스템이 부트될 때 DB2를 시작하려면, 시작 폴더의 단축 아이콘에서 배치 파일을 수행해야 합니다. 두 개의 옵션이 있습니다.

   - **db2start.exe** 대신 배치 파일을 수행하도록 DB2 설치 프로그램이 작성한 단축 아이콘을 수정하십시오. 앞의 예에서 단축 아이콘은 이제 db2start.bat 배치 파일을 수행합니다. DB2 설치 프로그램에 의해 작성된 단축 아이콘을 **DB2 - DB2.lnk**라고 하며 대부분의 시스템에서 c:\WINDOWS\Start Menu\Programs\Start\DB2 - DB2.lnk에 위치합니다.
   - 배치 파일을 수행하는 자신의 단축 아이콘을 추가하고, DB2 설치 프로그램에 의해 추가된 단축 아이콘을 삭제하십시오. 다음의 명령을 사용하여 DB2 단축 아이콘을 삭제하십시오.

     ```
     del  "C:\WINDOWS\Start Menu\Programs\Startup\DB2 - DB2.lnk"
     ```

   사용자 자신의 단축 아이콘을 사용하기로 결정한 경우, 단축 아이콘에 대해 *종료시 닫기* 속성을 설정해야 합니다. 이 속성을 설정하지 않으면, DOS 명

령 프롬프트가 **db2start** 명령이 정상적으로 완료된 이후에도 태스크 표시줄에 남아 있습니다. DOS 창이 **db2start** 프로세스 중에 열려 있는 것을 방지하려면, 최소화하여 수행하도록 설정된 이 단축 아이콘(및 이를 수행하는 DOS 창)을 작성할 수 있습니다.

주: 시스템을 부트하는 동안 DB2를 시작하는 대신, DB2를 사용하는 응용프로그램을 수행하기 전에 DB2를 시작할 수 있습니다. 209 페이지의 5에서 자세한 내용을 참조하십시오.

db2start 명령이 수행되기 전에 배치 파일을 사용하여 **db2logon** 명령을 발행하고 사용자가 때때로 로그온하는 경우, **db2start** 명령은 계속해서 작동되며 DB2가 로그온한 사용자의 사용자 ID를 사용하는 것이 유일한 차이점이 됩니다. 1에서 추가 세부사항을 참조하십시오.

주:

1. **db2logon** 명령은 사용자 로그온을 시뮬레이트합니다. db2logon 명령의 형식은 다음과 같습니다.

   ```
   db2logon userid  /p:password
   ```

   명령에 대해 지정된 사용자 ID는 DB2 이름 지정 요구사항을 만족시켜야 합니다(208 페이지의 2에서 자세한 내용 참조). 명령이 사용자 ID와 암호 없이 발행된 경우, 사용자에게 사용자 ID와 암호를 묻는 프롬프트가 표시된 창이 열립니다. 유일하게 제공되는 매개변수가 사용자 ID인 경우, 사용자에게 암호를 묻는 프롬프트를 표시하지 않으며 아래에 설명된 특정 상황 하에서는 암호가 필요합니다.

   db2logon 명령에 의해 설정된 사용자 ID와 암호는 사용자가 Windows 로그온 창이나 Microsoft 네트워킹 로그온 창을 사용하여 로그온하지 않은 경우에만 사용합니다. 사용자가 로그온했으며 **db2logon** 명령이 발행되었으면, **db2logon** 명령의 사용자 ID는 모든 DB2 조치에 사용되지만 **db2logon** 명령에 지정된 암호는 무시됩니다.

   사용자가 Windows 로그온 창이나 Microsoft Networking 로그온 창을 사용하여 로그온하지 않은 경우, **db2logon** 명령을 통해 제공된 사용자 ID와 암호는 다음과 같이 사용됩니다.

- **db2start** 명령은 시작시에 사용자 ID를 사용하며 암호를 요구하지 않습니다.
- 테이블 작성과 같은 조치에 고급 규정자가 누락된 경우, 사용자 ID를 고급 규정자로 사용합니다. 예를 들어, 다음과 같습니다.
  a. `db2logon db2local`을 발행하십시오.
  b. `create table tab1`을 발행하십시오.

     테이블은 고급 규정자로 `db2local.tab1`을 사용하여 작성됩니다.

  테이블과 다른 오브젝트의 스키마 이름과 동일한 사용자 ID를 사용해야 합니다.
- 시스템이 서버에 대한 클라이언트의 역할을 수행하며 사용자가 사용자 ID와 암호없이 CONNECT 명령문(예: CONNECT TO TEST)을 발행하고 인증이 `server`로 설정된 경우, **db2logon** 명령의 사용자 ID와 암호는 리모트 서버에서 사용자의 유효성을 확인하는 데 사용됩니다. 사용자가 명시적 사용자 ID와 암호를 사용하여 연결하는 경우(예: CONNECT TO TEST USER *userID* USING *password*), CONNECT 명령문에 대해 지정된 값이 사용됩니다.

2. 버전 7에서 로그온하는 데 사용되거나 **db2logon** 명령에 대해 지정된 사용자 ID는 다음 DB2 요구사항을 준수해야 합니다.
   - USERS, ADMINS, GUESTS, PUBLIC, LOCAL 또는 *SQL 참조서*에 나열된 SQL 예약어가 아니어야 합니다.
   - SQL, SYS 또는 IBM으로 시작할 수 없습니다.
   - 문자에 다음이 포함될 수 있습니다.
     – A - Z(Windows 95, Windows 98 및 Windows ME는 대소문자 구별 사용자 ID를 지원함)
     – 0 - 9
     – @, # 또는 $
3. 사용자 정의된 대화식 설치 중에 또는 응답 파일 설치를 수행하고 DB2.AUTOSTART=NO 옵션을 지정한 경우에 시작 폴더에서 **db2start** 단축 아이콘의 작성을 방지할 수 있습니다. 이들 옵션을 사용하는 경우, 시작 폴

더에는 db2start 단축 아이콘이 없으며 `db2start.bat` 파일을 수행하기 위한 사용자 자신의 단축 아이콘을 추가해야 합니다.

4. **Windows 98** 및 **Windows ME**에서 Windows 98 또는 Windows ME가 시작되어 있으면 항상 로그온되는 사용자 ID를 지정하는 데 사용할 수 있는 옵션이 있습니다. 이 경우, Windows 로그온 창은 나타나지 않습니다. 이 옵션을 사용하는 경우, 사용자 ID가 DB2 요구사항을 만족시키면 사용자가 로그온되고 **db2start** 명령이 완료됩니다(208 페이지의 2에서 자세한 내용 참조). 이 옵션을 사용하지 않으면, 사용자에게는 항상 로그온 창이 제시됩니다. 사용자가 로그온하지 않고 이 창을 취소하면, 위에 설명된 대로 **db2logon** 명령이 이전에 발행되지 않았거나 배치 파일에서 호출되지 않은 경우에 **db2start** 명령은 실패합니다.
5. 시스템 부트 중에 DB2를 시작하지 않으면, 응용프로그램이 DB2를 시작할 수 있습니다. DB2를 사용하는 응용프로그램 초기화의 일부로 `db2start.bat` 파일을 수행할 수 있습니다. 이 방법을 사용하면 DB2를 사용할 응용프로그램이 시작되었을 때에만 DB2가 시작됩니다. 사용자가 응용프로그램을 종료할 때 **db2stop** 명령을 발행하여 DB2를 중지할 수 있습니다. DB2가 시스템 부트 중에 시작되지 않은 경우, 비즈니스 응용프로그램은 이러한 방식으로 DB2를 시작할 수 있습니다.

   DB2 동기화 응용프로그램을 사용하거나 응용프로그램에서 동기화 API를 호출하려면, 실행하기 위해 다운로드된 스크립트에 로컬 인스턴스나 로컬 데이터베이스에 대해 작동되는 명령이 포함된 경우, DB2를 시작해야 합니다. 이들 명령은 데이터베이스 스크립트, 인스턴스 스크립트에 있거나 운영 체제(OS) 스크립트에 내재될 수 있습니다. OS 스크립트에 인스턴스나 데이터베이스를 사용하는 명령행 처리기 명령이나 DB2 API가 포함되지 않은 경우, DB2를 시작하지 않고 수행할 수 있습니다. 동기화 프로세스 중에 스크립트에서 수행될 명령을 미리 알려주기 어렵기 때문에 일반적으로 DB2는 동기화가 시작되기 전에 시작해야 합니다.

   응용프로그램에서 **db2sync** 명령이나 동기화 API를 호출하는 경우, 응용프로그램의 초기화 중에 DB2를 시작하게 됩니다. 사용자가 Windows용 DB2 폴더의 DB2 동기화 단축 아이콘을 사용하여 동기화를 시작하는 경우,

db2sync.bat 파일을 수행하도록 DB2 동기화 단축 아이콘을 수정해야 합니다. DB2가 동기화가 시작되기 전에 수행되도록 하려면, 배치 파일에 다음의 명령이 포함되어야 합니다.

```
@echo off
db2start.bat
db2sync.exe
db2stop.exe
cls
exit
```

이 예에서 db2start.bat 파일은 **db2logon** 및 **db2start** 명령을 위에 설명된 대로 호출하는 것으로 가정합니다.

응용프로그램이 시작되면 DB2를 시작하기로 결정한 경우, DB2의 설치가 시작 폴더에 DB2를 시작하기 위한 단축 아이콘을 추가하지 않았는지 확인하십시오. 208 페이지의 3에서 자세한 내용을 참조하십시오.

제 1 장 문제 해결 연습

문제점 분석 및 환경 콜렉션 도구

문제점과 연관된 일부 정보를 식별하고 DB2 고객 지원팀이 사용자의 환경 및 문제점을 이해하는 데 도움이 되는 기타 관련 정보를 수집하는 유틸리티가 있습니다. 지금부터 이 유틸리티를 사용하여 수집하는 것에 대해 살펴봅니다. 유틸리티는 **db2support**입니다.

구문과 명령행 옵션에 대한 세부사항은 *Command Reference*를 참조하십시오.

이 유틸리티의 주요 기능은 DB2를 실행하는 클라이언트 또는 서버 머신에 대해서 환경 데이터를 수집하는 것입니다. 그런 다음 해당 결과를 수집해 탐색 가능한 XML, HTML 또는 압축 파일 아카이브로 패키지하는 것입니다. 또한 이 유틸리티에는 대화식 질문과 응답 프로세스를 통해 사용자의 본질적인 문제점에 관한 일부 데이터를 수집할 수 있는 옵션이 있습니다. 이 프로세스를 통해 문제점을 밝혀내고 최종적으로 DB2 고객 지원팀에 문제점에 대해 문의할 때 필요한 정보를 제공할 수 있습니다.

주: thin 또는 Run-Time Client는 이 유틸리티를 사용할 수 없습니다. 이 유틸리티를 사용하기 위해서는 클라이언트에 DB2 엔진 라이브러리가 설치되어 있어야 합니다.

콜렉션 출력

이 유틸리티는 주요 데이터베이스 시스템 정보를 하나의 압축한 콜렉션(단일 파일 아카이브)으로 생성합니다. 이 아카이브에는 매우 기본적인 정보를 정리한 HTML 보고서가 포함되는데 이를 통해 사용자는 해당 정보를 살펴볼 수 있습니다.

기본적으로 db2support는 고객 데이터의 보안 및 민감성을 보호해야 하기 때문에 테이블 데이터, 스키마(DDL) 또는 로그를 수집하지 못합니다. 그러나 일부 옵션을 통해 사용자는 스키마나 데이터의 일부분을 포함(예: 아카이브된 로그 파일 포함)할 수 있습니다. 데이터베이스 스키마 또는 데이터를 노출하는 옵션은 조심해서 사용해야 합니다. db2support를 호출하면 다룰 데이터의 민감도를 나타내는 메시지가 표시됩니다.

다음은 단일 아카이브로 수집 및 압축되는 파일입니다.

조건에 상관없이 수집

1. db2diag.log
2. 모든 트랩 파일
3. 잠금 목록 파일(-d와 함께 사용)
4. 덤프 파일
5. 사용자 종료(-d와 함께 사용)
6. 버퍼 풀 및 테이블 스페이스(SPCS) 제어 파일(-d와 함께 사용)
7. 여러 시스템 관련 파일
8. 여러 시스템 명령어의 출력
9. db config(-d와 함께 사용)
10. dbm config 파일
11. 로그 파일 헤더 파일(-d와 함께 사용)
12. 복구 실행기록 파일
13. db2cli.ini

선택적 수집

1. 사용 중인 로그 파일
2. db2dump 디렉토리의 목차(예: 위에서 수집하지 않은 파일들)
3. 코어 파일(모든 코어 파일에는 -a, 대부분의 최근 코어 파일에는 -r 사용)
4. 확장 시스템 정보(-s)

다음은 HTML 보고서의 목차를 구성하는 파일입니다.

조건에 상관없이 수집

1. 존재할 경우의 PMR 번호(-n이 지정된 경우)
2. 운영 체제 및 레벨(예: AIX 4.2.1)
3. DB2 릴리스 정보
4. 엔진 라이브러리 헤더 정보
5. 32비트 또는 64비트 발견
6. DB2 설치 경로 정보
7. db2nodes.cfg의 EEE 보고서 목차
8. CPU, 디스크 및 메모리 용량
9. 현재 인스턴스에 대한 데이터베이스 목록
10. & libpath 경로를 포함한 레지스트리 정보 및 환경
11. 현재 파일 시스템에서 사용 가능한 디스크 여유 공간 및 Unix의 inode
12. JDK 레벨
13. dbm config
14. 데이터베이스 복구 실행기록 파일의 목록
15. sqllib 디렉토리의 ‘ls -lR’(또는 windows equivilant)
16. LIST NODE DIRECTORY
17. LIST ADMIN NODE DIRECTORY
18. LIST DCS DIRECTORY
19. LIST DCS APPLICATIONS EXTENDED
20. 설치된 모든 소프트웨어 목록

'-s'로 지정된 경우 수집

1. 디스크 정보 세부사항(예: 파티션 레이아웃, 유형, LVM 정보)
2. 네트워크 정보 세부사항
3. 커널 통계
4. 펌웨어 버전
5. 기타 플랫폼 고유 명령어

DB2가 시작된 경우 수집

1. 클라이언트 연결 상태
2. db/dbm config(db cfg는 -d 옵션이 필수임)
3. CLI config
4. 메모리 풀 정보(크기 및 사용량). -d 옵션이 사용될 경우 데이터 완료
5. LIST ACTIVE DATABASES
6. LIST DATALINKS MANAGERS
7. LIST DCS APPLICATIONS

-c가 지정되고 데이터베이스에 연결할 수 있는 경우 수집

1. 사용자 테이블 수
2. DB 데이터의 추정 크기
3. 데이터베이스 스냅샷
4. 응용프로그램 스냅샷
5. 버퍼 풀 정보
6. LIST APPLICATIONS
7. LIST COMMAND OPTIONS
8. LIST DATABASE DIRECTORY
9. LIST INDOUBT TRANSACTIONS
10. LIST NODEGROUPS
11. LIST NODES
12. LIST ODBC DATA SOURCES

13. LIST PACKAGES/TABLES
14. LIST TABLESPACE CONTAINERS
15. LIST TABLESPACES
16. LIST DRDA IN DOUBT TRANSACTIONS

'-q'가 지정된 경우에는 다음을 수집

대화식 질문 및 응답 모드가 시작됩니다. ″describe your problem″ 옵션 질문을 예외로 처리하고 고객 정보에 대한 요청이 적은 경우, 모든 질문은 선택한 질문에 다중 선택 응답을 갖습니다. 계속되는 모든 질문과 응답은 수집됩니다. 일부 경우, 이 유틸리티는 태스크를 수행해 해당 결과를 추가 디렉토리에 놓으라는 질문을 합니다. 대화식 모드에서 질의할 질문을 결정하는 동안 작은 결정 트리가 사용됩니다. 이 대화식 질문은 질문 범주를 결정하는 데 도움이 되며 이 범주에 기초하여 일부 관련된 기타 질문을 질의할 수 있고 데이터를 추가로 수집할 수 있습니다. 질문이 끝나면 자동 모드로 수집될 데이터가 수집됩니다. 모든 질문에 대한 응답은 자동 모드로 수집된 데이터와 함께 서비스에 보낼 수 있도록 저장됩니다.

detailed_system_info.html 보기

비영문 설치 프로그램에서 db2support를 실행하고 있고, detailed_system_info.html을 보는 데 차이를 느낀다면 DOS 코드화를 지원하는 Internet Explorer 버전 5 이상을 사용해야 합니다. 코드화를 변경하려면, 보기 --> 코드화 --> 중부 유럽어(DOS)를 선택하십시오. 아직 필요한 코드화 지원 기능을 갖지 않고 있다면, Internet Explorer가 Microsoft 갱신 웹 사이트에서 필요한 파일을 다운로드 하라는 프롬프트를 띄우십시오. 이 정보는 2바이트 언어(중국어, 대만어, 일본어 및 한국어)에는 해당되지 않습니다.

한 번에 한 페이지씩 DB2 지원 도구 구문 보기

한 번에 한 페이지씩 DB2 지원 도구 구문을 보려면 다음 명령을 실행하십시오.

```
db2support | more
```

『잠금 및 교착 상태』 섹션의 『응용프로그램이 느리거나 정지된 것처럼 보임』 서브 섹션에서 『잠금 대기 또는 교착 상태가 다음 키 잠금으로 발생하지 않음』의 설명을 다음과 같이 바꾸십시오.

*다음 키 잠금*은 모든 INSERT 및 DELETE문에 대한 다음 키와 SELECT문의 결과 세트보다 더 높은 다음 키 값을 자동으로 잠궈서 반복 읽기(RR)를 보장합니다. 인덱스의 키 부분을 변경하는 UPDATE문의 경우, 원래 인덱스 키가 삭제되고 새 키 값이 삽입됩니다. 다음 키 잠금은 키 삽입과 키 삭제시 모두 수행됩니다. ANSI 및 SQL92 표준 RR을 보장해야 하며 이는 DB2 기본값입니다.

응용프로그램에 대한 스냅샷 정보를 검사하십시오. 다음 키 잠금에 문제점이 있으면 반복 읽기(RR) 동작에 의존하는 응용프로그램이 없고 검색에서 커미트되지 않은 삭제를 건너뛸 수 있는 경우, DB2_RR_TO_RS 옵션을 설정할 수 있습니다.

DB2_RR_TO_RS가 설정되면 다음 키 잠금이 인덱스 키 삽입 및 삭제시 수행되지 않으므로 사용자 테이블에서의 검색에 대한 RR 동작을 보장할 수 없습니다. 카탈로그 테이블은 이 옵션의 영향을 받지 않습니다.

동작의 다른 변경은 DB2_RR_TO_RS를 설정하는 것인데 행이 검색 규정에 맞아도 삭제되고 커미트되지 않은 행을 검색할 경우에는 건너뜁니다.

예를 들어, 트랜잭션 A가 컬럼1=10인 행을 삭제하고 트랜잭션 B가 컬럼1>8이고 컬럼1<12인 행을 검색하는 시나리오를 생각해 봅니다.

DB2_RR_TO_RS를 해제하면 트랜잭션 B는 트랜잭션 A가 커미트하거나 롤백할 때까지 기다립니다. 롤백하는 경우, 컬럼1=10인 행이 트랜잭션 B의 쿼리 결과 세트에 포함됩니다.

DB2_RR_TO_RS가 설정되면 트랜션 B는 트랜잭션 A가 커미트하거나 롤백할 때까지 기다리지 않습니다. 삭제된 행을 포함하지 않은 쿼리 결과를 즉시 받습니다.

ANSI 및 SQL92 표준 RR이 필요한 경우나 검색시 커미트되지 않은 삭제를 건너뛰지 않을 경우에는 이 옵션을 사용하지 마십시오.

제 8 장 DB2 Data Links Manager 문제점 해결

버전 7 FixPak 2에서 FEDERATED 옵션에 값을 지정하지 않고 소스 파일을 사전 처리 컴파일하거나 바인드 파일을 바인딩할 때 서버에 의해 SQL1179W 경고 메시지가 생성됩니다. 소스 파일 또는 바인드 파일에 별명을 참조하는 정적 SQL이 포함되어 있을 때 같은 메시지가 생성됩니다. 두 예외사항이 있습니다.

- 버전 7 FixPak 2 이전의 FixPak인 클라이언트 또는 아래 레벨의 클라이언트의 경우, sqlaprep() API가 메시지 파일에 이 SQL1179W 경고를 보고하지 않습니다. 명령행 처리기 PRECOMPILE 명령 역시 이 경우에 경고를 출력하지 않습니다.
- 버전 7 FixPak 2 이전의 FixPak인 클라이언트 또는 아래 레벨의 클라이언트의 경우, sqlabndx API가 메시지 파일에 이 SQL1179W 경고를 보고합니다. 하지만, 또한 메시지 파일에 아무 패키지도 작성되지 않았음을 의미하는 SQL0092N 메시지가 포함됩니다. 실제로 패키지가 작성되므로 이는 옳지 않습니다. 명령행 처리기 BIND 명령이 동일한 잘못된 경고를 리턴합니다.

제 15 장 로그된 정보

UNIX 기반 시스템의 스택 역추적 정보 수집

*문제점 해결 안내서*에서 다중 노드 시스템의 모든 노드에서 스택 역추적을 활성화하려면, **db2_all** 명령을 사용해야 한다고 잘못 설명하고 있습니다. **db2_call_stack** 명령만 필요합니다. **db2_all**과 **db2_call_stack**을 함께 사용하면 오류가 발생합니다.

64비트 플랫폼에서 DB2 Universal Database 사용

제 5 장 구성

LOCKLIST

다음 정보를 테이블 2에 추가해야 합니다.

매개변수	이전 상한	현재 상한
LOCKLIST	60000	524288

shmsys:shminfo_shmmax

64비트 Solaris 운영 체제의 DB2 사용자들은 대형 데이터베이스 공유 메모리 세트를 할당할 수 있도록 /etc/system에 있는 ″shmsys:shminfo_shmmax″ 값을 필요한 만큼 증가해야 합니다. *UNIX용 DB2 빠른 시작* 책에서는 이 매개변수를 ″바이트로 머신에서 물리적 RAM의 90%″로 설정하도록 권합니다. 이 권장사항은 64비트 구현에서도 유효합니다.

그러나, *UNIX용 DB2 빠른 시작* 책에 있는 다음 권장사항은 잘못된 것입니다. 4GB 이상의 RAM(Solaris 운영 체제에서 사용할 수 있는 전체 양의 최대 64GB)이 있는 32비트 시스템의 경우, 사용자가 shmmax 값을 4GB 이상으로 설정하고 32비트 커널을 사용하면, 커널은 숫자의 하위 32비트만을 참조하므로, shmmax가 아주 작은 값이 되는 경우가 있습니다.

제 6 장 제한사항

현재 64비트 운영 체제에서는 LDAP 지원이 없습니다.

32비트 및 64비트 데이터베이스는 같은 경로에서 작성될 수 없습니다. 예를 들어 32비트 데이터베이스가 <somepath>에 있는 경우,

```
db2 create db <somedb> on <somepath>
```

64비트 인스턴스에서 발행되면, ″SQL10004C 데이터베이스 디렉토리에 액세스하는 동안 입출력 오류가 발생했습니다.″와 함께 실패합니다.

XML Extender 관리 및 프로그래밍

IBM DB2 XML Extender의 릴리스 정보는 DB2 XML 웹 사이트 http://www.ibm.com/software/data/db2/extenders/xmlext/library.html에 있습니다.

MQSeries

이 섹션에서는 DB2 및 MQSeries를 사용하여 메시지 및 데이터베이스 액세스를 결합하는 응용프로그램을 구성하는 방법에 대해 설명합니다. 이 섹션에서는 사용자 정의 함수(UDF)와 비슷한 함수 집합을 중심으로 설명합니다. 이 함수 집합은 DB2 Universal Database, 버전 7.2에서 선택적으로 사용 가능합니다. 이 기본 함수를 사용하여 단순한 이벤트 통지에서부터 데이터 웨어하우스에 이르는 광범위한 응용프로그램을 지원할 수 있습니다.

데이터 웨어하우스 응용프로그램에 대한 자세한 정보는 최근에 새로 고쳐진 Data Warehouse Center Administration 안내서를 참조하십시오. 이는 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 얻을 수 있습니다.

DB2 MQSeries 함수용으로 설치 및 구성

이 섹션에서는 DB2 MQSeries 함수를 사용하기 위해 DB2 환경을 구성하는 방법에 대해 설명합니다. 다음 절차를 완료하면 SQL 내에서 DB2 MQSeries 함수를 사용할 수 있습니다. 이들 함수에 대한 설명은 릴리스 정보의 SQL 참조서 섹션에 있습니다.

DB2 MQSeries 함수를 구성하고 사용하기 위한 기본 절차는 다음과 같습니다.

1. 각 물리적 머신에 MQSeries를 설치합니다.
2. 물리적 머신에 MQSeries AMI를 설치합니다.
3. DB2 MQSeries 함수를 사용하여 구성합니다.

그리고, DB2 MQSeries 함수에서 제공하는 publish/subscribe 기능을 사용하려면 각각의 실제 머신에 MQSeries Integrator 또는 MQSeries Publish/Subscribe 함수도 설치해야 합니다. MQSeries Integrator에 대한 정보는 http://www.ibm.com/

software/ts/mqseries/integrator에 있습니다. MQSeries Publish/Subscribe 기능에 대한 정보는 http://www.ibm.com/software/ts/mqseries/txppacs의 범주 3에 있습니다.

MQSeries 설치

먼저 최신 FixPak과 함께 MQSeries 버전 5.1의 최소 사양이 DB2 서버에 설치되어 있는지 확인하십시오. 이 버전의 MQSeries가 이미 설치되어 있으면 다음 단계인 "MQSeries AMI 설치"로 건너뛰십시오. DB2 버전 7.2에는 DB2와 함께 사용할 MQSeries 서버의 사본이 들어 있습니다. MQSeries를 설치하거나 기존 MQSeries 설치를 업그레이드하기 위한 플랫폼별 지침은 http://www.ibm.com/software/ts/mqseries/library/manuals의 플랫폼별 빠른 시작 책에 있습니다. 설치를 계속할 경우, 기본 대기행렬 관리 프로그램을 설정해야 합니다.

MQSeries AMI 설치

다음 단계는 MQSeries Application Messaging Interface(AMI)를 설치하는 것입니다. 이는 관리 및 프로그래밍 태스크를 완전하게 분리하는 MQSeries 프로그래밍 인터페이스를 확장한 것입니다. DB2 MQSeries 함수를 사용하려면 이 인터페이스를 설치해야 합니다. MQSeries AMI가 DB2 서버에 이미 설치되어 있으면 다음 단계인 "DB2 MQSeries 함수 사용"으로 건너뛰십시오. MQSeries AMI가 설치되어 있지 않으면 DB2 7.2와 함께 제공된 설치 패키지에서 설치하거나 http://www.ibm.com/software/ts/mqseries/txppacs의 MQSeries Support Pacs 웹 사이트에서 복사하십시오. AMI는 "범주 3 - 제품 확장"에 있습니다. 사용자 편의를 위해 MQSeries AMI 사본을 DB2와 함께 제공하고 있습니다. 이 파일은 `sqllib/cfg` 디렉토리에 있습니다. 파일 이름은 운영 체제에 따라 달라집니다.

AIX 버전 4.3 이상	ma0f_ax.tar.Z
HP-UX	ma0f_hp.tar.Z
Solaris 운영 환경	ma0f_sol7.tar.Z 또는 mq0f_sol26.tar.Z
Windows 32비트	ma0f_nt.zip

압축 설치 이미지에 포함된 AMI readme 파일에서 설명하는 일반적인 AMI 설치 절차를 따르십시오.

DB2 MQSeries 함수 사용

이 단계에서 DB2 MQSeries 함수를 구성하고 사용하게 됩니다.

enable_MQFunctions 유틸리티는 먼저 적절한 MQSeries 환경이 설정되었는지 확인한 후, DB2 MQSeries 함수에 대한 기본 구성을 설치 및 작성하고 이들 함수를 사용하여 지정된 데이터베이스를 사용 가능하게 하여 구성이 작동되는지 확인합니다.

1. Windows NT 또는 Windows 2000인 경우 5단계로 진행하십시오.
2. UNIX에서 그룹 설정: UNIX에서 이 함수를 사용 가능하게 하려면 먼저 DB2 인스턴스 소유자(주로 db2inst1) 및 분리(fenced)된 UDF(주로 db2fenc1)를 MQSeries 그룹 mqm에 추가해야 합니다. DB2 함수가 MQSeries를 액세스하기 위해 필요합니다.
3. UNIX에서 DB2 환경 변수 설정: AMT_DATA_PATH 환경 변수를 DB2가 인식한 목록에 추가합니다. `$INSTHOME/sqllib/profile.env` 파일을 편집하고 AMT_DATA_PATH를 DB2ENVLIST에 추가할 수 있습니다. **db2set** 명령을 사용할 수도 있습니다.
4. UNIX에서 데이터베이스 인스턴스 재시작: 변경한 환경 변수를 적용하려면 데이터베이스 인스턴스를 재시작해야 합니다.
5. UNIX일 경우에는 디렉토리를 `$INSTHOME/sqllib/cfg`로, Windows일 경우에는 `%DB2PATH%/cfg`로 변경합니다.
6. **enable_MQFunctions** 명령을 수행하여 DB2 MQSeries 함수용 데이터베이스를 구성하고 사용 가능하게 합니다. DB2 UDB EEE 환경에서, 카탈로그 노드에서 이 단계만 수행하십시오. 이 명령에 대한 자세한 설명은 238 페이지의 『enable_MQFunctions』를 참조하십시오. 일반적인 몇 가지 예가 아래에 있습니다. 성공적으로 완료한 후 지정된 데이터베이스는 사용 가능하며 구성이 테스트됩니다.
7. 명령행 처리기를 사용하여 이 함수를 테스트하려면 사용 가능한 데이터베이스에 연결한 후 다음 명령을 실행합니다.

   ```
   values DB2MQ.MQSEND('a test')
   values DB2MQ.MQRECEIVE()
   ```

   첫 번째 명령문은 ″a test″ 메시지를 DB2MQ_DEFAULT_Q 대기행렬로 보내며 두 번째 명령문은 이 메시지를 다시 받습니다.

주: **enable_MQFunctions**을 수행하면 기본 MQSeries 환경이 설정됩니다. MQSeries 대기행렬 관리 프로그램 DB2MQ_DEFAULT_MQM 및 기본 대기행렬 DB2MQ_DEFAULT_Q가 작성됩니다. amt.xml, amthost.xml 및 amt.dtd 파일은 이 파일들이 AMT_DATA_PATH가 가리키는 디렉토리에 없는 경우, 작성됩니다. amthost.xml 파일이 없고 connectionDB2MQ의 정의가 포함되어 있지 않으면 적절한 정보와 함께 파일에 한 행이 추가됩니다. 원본 파일 사본은 DB2MQSAVE.amthost.xml로 저장됩니다.

MQSeries 메시지 스타일

DB2 MQSeries 함수는 데이터그램, pubslish/subscribe(p/s) 및 요청/응답(r/r)의 세 가지 메시지 모델을 지원합니다.

데이터그램으로 전송된 메시지는 예상되는 응답없이 단일 목적지로 전송됩니다. p/s 모델에서는 하나 이상의 subscriber에게 메시지를 배포하는 publication 서비스에게 하나 이상의 publisher가 메시지를 보냅니다. 요청/응답은 데이터그램과 비슷하지만 보낸 사람이 응답을 받을 것으로 예상합니다.

메시지 구조

MQSeries 자체는 전송하는 메시지의 특정한 구조화를 위임하거나 지원하지 않습니다.

MQSeries Integrator(MQSI) 등의 다른 제품이 C, Cobol 또는 XML 문자열로 형성된 메시지를 지원합니다. MQSI 내의 구조화된 메시지는 메시지 저장소가 정의합니다. XML 메시지는 일반적으로 자체 설명적인 메시지 구조로서 저장소를 통해 관리될 수도 있습니다. 메시지는 구조화되지 않을 수도 있는데 이 경우 메시지 내용을 분석하고 구성하는 사용자 코드가 필요합니다. 이러한 메시지는 주로 반구조화되는데, 즉 메시지 내의 필드를 구분하기 위해 바이트 위치나 분리문자를 사용합니다. 이러한 반구조화된 메시지는 MQSeries 지원 마법사가 지원합니다. XML 메시지는 DB2 XML Extender의 새로운 일부 기능이 지원합니다.

MQSeries 기능 개요

MQSeries 함수 세트는 DB2 UDB 버전 7.2과 함께 제공되며 SQL문에 메시지 조작이 포함될 수 있습니다. 따라서 이런 지원은 데이터베이스 인터페이스를 사용하는 C, Java, SQ와 같은 지원되는 언어로 작성된 응용프로그램에서 사용할 수 있습니다. 아래 예는 SQL로 작성한 것입니다. 이 SQL은 다른 프로그래밍 언어에서 표준 방식으로 사용할 수 있습니다. 위에서 설명한 모든 MQSeries 메시지 스타일이 지원됩니다. MQSeries 함수에 대한 자세한 정보는 릴리스 정보의 SQL 참조서 섹션을 참조하십시오.

기본 구성에서 MQSeries 서버는 DB2와 함께 데이터베이스 서버 머신에 있습니다. MQSeries 함수는 DB2에 설치되며 MQSeries 서버에 대한 액세스를 제공합니다. DB2 클라이언트는 DB2 서버에 액세스할 수 있는 모든 머신에 설치될 수 있습니다. 여러 클라이언트가 데이터베이스를 통해 MQSeries 함수에 동시에 액세스할 수 있습니다. 제공한 함수로 DB2 클라이언트는 SQL문 내에서 메시지 조작을 수행할 수 있습니다. 이 메시지 조작으로 DB2 응용프로그램은 서로 통신하거나 다른 MQSeries 응용프로그램과 통신할 수 있습니다.

enable_MQFunctions 명령을 사용하여 MQSeries 함수에 대해 DB2 데이터베이스를 사용 가능하게 할 수 있습니다.이 명령은 클라이언트 응용프로그램이 추가 관리 조치없이 사용하는 단순한 기본 구성을 설정합니다. 설명은 238 페이지의 『enable_MQFunctions』 및 241 페이지의 『disable_MQFunctions』에서 참조하십시오. 기본 구성을 사용하여 응용프로그램 프로그래머는 개발을 쉽게 시작할 수 있고 개발용으로 간단한 인터페이스를 만들 수 있습니다. 추가 기능은 필요에 따라 계속 구성할 수 있습니다.

예 1: 기본 구성으로 단순한 메시지를 보내는 SQL문은 다음과 같습니다.

```
VALUES DB2MQ.MQSEND('simple message')
```

이 SQL문은 `simple message`라는 메시지를 MQSeries 대기행렬 관리 프로그램 및 기본 구성에 지정된 대기행렬로 보냅니다.

MQSeries의 Application Messaging Interface(AMI)는 메시지 조치와 이 조치를 수행하는 방법에 대해 설명하는 정의를 명확하게 구분합니다. 이 정의는 외부 저장소 파일에 보존하며 AMI 관리 도구로 관리합니다. 따라서 AMI 응용프로그램

을 개발하고 관리하기가 쉬워집니다. DB2와 함께 제공된 MQSeries 함수는 AMI MQSeries 인터페이스를 기준으로 합니다. AMI는 구성 정보를 저장하기 위해 AMI Repository라는 외부 구성 파일을 사용하도록 지원합니다. 기본 구성에는 DB2와 함께 사용하도록 구성된 MQSeries AMI Repository가 포함됩니다.

MQSeries AMI의 두 가지 주요 개념인 서비스 지점 및 규정이 DB2 MQSeries 함수로 전달됩니다. 서비스 지점은 메시지를 보내거나 받는 논리적인 끝점입니다. AMI 저장소에서 각 서비스 지점은 MQSeries 대기행렬 이름 및 대기행렬 관리 프로그램과 함께 정의됩니다. 규정은 주어진 메시지 조작에 사용해야 하는 서비스 옵션의 질을 정의합니다. 서비스의 주요 질에는 메시지 우선순위 및 지속성이 포함됩니다. 기본 서비스 지점 및 규정 정의가 제공되며 개발 프로그램에서 응용프로그램을 더욱 단순하게 만들기 위해 사용될 수 있습니다. 예 1은 기본 서비스 지점 및 규정 이름을 명시적으로 지정하기 위해 다음과 같이 다시 작성할 수 있습니다.

예 2:

```
VALUES DB2MQ.MQSEND('DB2.DEFAULT.SERVICE', 'DB2.DEFAULT.POLICY',
                    'simple message')
```

대기행렬 및 응용프로그램이 있는 서버에서 하나 이상의 응용프로그램이 대기행렬에 서비스를 제공할 수 있습니다. 많은 구성에서 다른 응용프로그램 및 목적을 지원하기 위해 복수의 대기행렬을 정의합니다. 이러한 이유로 MQSeries 요청을 작성할 때 서로 다른 서비스 지점을 정의하는 것이 중요합니다. 다음은 이러한 예를 보여줍니다.

예 3:

```
VALUES DB2MQ.MQSEND('ODS_Input', 'simple message')
```

주: 이 예에서는 규정이 지정되지 않았으므로 기본 규정이 사용됩니다.

제한사항

MQSeries는 메시지 조작과 데이터베이스 조작을 원자 트랜잭션인 단일 작업 단위(UOW)로 결합하는 기능을 제공합니다. 이 기능이 처음부터 Unix 및 Windows의 MQSeries 함수에 의해 지원되는 것은 아닙니다.

송수신 함수를 사용할 때 VARCHAR 유형에 대한 메시지의 최대 길이는 4000 자입니다. CLOB 유형 메시지를 송수신할 때의 최대 길이는 1MB입니다. 이는 MQPublish를 사용하여 메시지를 발행할 때의 최대 메시지 크기이기도 합니다.

CLOB 메시지와 VARCHAR 메시지를 이용해 작업할 때에는 종종 다른 함수가 필요합니다. 일반적으로 MQ 함수의 CLOB 버전은 식별 구문을 사용합니다. 이름 끝에 CLOB이라는 문자가 붙는다는 것 이외에 다른 차이는 없습니다. 예를 들어, MQREAD의 CLOB 이름은 MQREADCLOB입니다. 이들 함수 목록에 대한 세부사항은 407 페이지의 『MQSeries 함수에서의 CLOB 데이터 지원 추가』를 참조하십시오.

오류 코드

MQSeries 함수가 리턴한 리턴 코드가 MQSeries Application Messaging Interface Manual의 부록 B에 나와 있습니다.

사용시나리오

MQSeries 함수를 다양한 시나리오로 사용할 수 있습니다. 이 섹션에서는 기본 메시지, 응용프로그램 연결성 및 데이터 책을 포함한 일반적인 시나리오를 몇 가지 검토합니다.

기본 메시지

모든 데이터베이스 응용프로그램이 같은 DB2 서버에 연결되어 있을 때 MQSeries DB2 함수와의 가장 기본적인 형태의 메시지 교환이 이루어집니다. 클라이언트는 데이터베이스 서버에 있거나 네트워크 환경에 분산되어 있습니다.

간단한 시나리오에서 클라이언트 A는 MQSEND 함수를 호출하여 사용자 정의 문자열을 기본 서비스 위치로 보냅니다. 그러면 MQSeries 함수가 데이터베이스 서버의 DB2에서 실행됩니다. 잠시 후에 클라이언트 B가 MQRECEIVE 함수를 호출하여 기본 서비스에서 정의한 대기행렬 헤드에서 메시지를 제거하여 이를 클라이언트에게 리턴합니다. 이 작업을 수행할 MQSeries 함수가 다시 DB2에 의해 실행됩니다.

데이터베이스 클라이언트는 간단한 메시지를 다양한 방법으로 사용할 수 있습니다. 일반적으로 다음과 같은 경우에 메시징을 사용합니다.

- 데이터 콜렉션 -- 하나 이상의 다양한 정보 소스에서 메시지 형태로 정보를 받습니다. 정보 소스는 SAP나 사내에서 개발된 응용프로그램과 같은 상업용 응용프로그램입니다. 대기행렬에서 이러한 데이터를 받아서 추가로 처리하거나 분석하기 위해 데이터베이스 테이블에 저장합니다.
- 워크로드 분산 -- 작업 요청이 동일한 응용프로그램의 여러 인스턴스가 공유하는 대기행렬에 포스트됩니다. 인스턴스가 일부 작업을 수행할 준비가 되면 수행할 작업 요청이 있는 대기행렬의 최상단에서 메시지를 받습니다. 이러한 기법을 사용하면 요청으로 채워진 단일 대기행렬로 표현되는 워크로드를 여러 인스턴스가 공유할 수 있습니다.
- 응용프로그램 신호 -- 여러 프로세스가 공동 작업하는 경우, 서로 협조하기 위해 메시지를 주로 사용합니다. 이 메시지에는 수행할 작업에 대한 명령이나 요청이 포함됩니다. 일반적으로 이러한 종류의 신호는 단방향입니다. 즉, 메시지를 보내는 개시자가 응답을 기대하지 않습니다. 자세한 정보는 233 페이지의 『요청/응답 통신』을 참조하십시오.
- 응용프로그램 통지 -- 통지는 데이터를 보내는 개시자가 응답을 기대하지 않는다는 점에서 신호와 비슷합니다. 그러나 일반적으로 통지에는 발생한 비즈니스 이벤트에 대한 데이터가 포함됩니다. 234 페이지의 『Publish/Subscribe』는 통지의 개선된 형태입니다.

다음 시나리오는 리모트 메시지를 통합하기 위해 위에서 설명한 단순한 시나리오를 확장한 것입니다. 즉, 머신 A와 머신 B 사이에서 메시지가 전송됩니다. 각 단계는 다음과 같습니다.

1. DB2 클라이언트가 머신 B의 리모트 대기행렬을 나타내기 위해 정의된 대상 서비스를 지정하는 MQSEND 호출을 실행합니다.
2. MQSeries DB2 함수는 메시지를 보내는 실제 MQSeries 작업을 수행합니다. 머신 A의 MQSeries 서버는 메시지를 받아들이고 이 메시지가 머신 A의 현재 MQSeries 구성 및 서비스 지점 정의로 정의한 목적지로 배달되도록 보장합니다. 서버는 머신 B의 대기행렬임을 파악한 후 메시지를 머신 B의 MQSeries 서버로 배달하려고 시도합니다. 이 때 필요하면 투명하게 재시도합니다.

3. 머신 B의 MQSeries서버는 머신 A의 서버에서 보낸 메시지를 받고 이 메시지를 머신 B의 목적지 대기행렬에 갖다 놓습니다.
4. 머신 B의 MQSeries 클라이언트는 대기행렬의 헤드에서 메시지를 요청합니다.

메시지 전송

MQSEND를 사용하여 DB2 사용자나 개발자는 전송할 데이터, 전송할 위치 및 전송 시기를 선택합니다. 업계에서는 이를 "보내고 무시하기" 즉, 보낸 사람은 단지 메시지를 보내기만 하고 메시지가 목적지에 도달했는지 확인하는 것은 MQSeries 배달 프로토콜이라고 합니다. 다음은 이를 보여주는 예입니다.

예 4: 사용자 정의 문자열을 서비스 지점 myPlace로 highPriority 규정과 함께 보냅니다.

```
VALUES DB2MQ.MQSEND('myplace','highPriority','test')
```

여기서, highPriority 규정은 MQSeries 우선순위를 가장 높은 레벨로 설정하고 지속성과 같은 서비스 질도 동일하게 조정하는 AMI Repository에 정의된 규정을 참조합니다.

메시지 내용은 SQL 및 사용자 지정 데이터의 적합한 결합으로 구성될 수 있습니다. 여기에는 중첩된 함수, 연산자 및 캐스트가 포함됩니다. 예를 들어, VARCHAR 컬럼 LASTNAME, FIRSTNAME 및 DEPARTMENT이 있는 EMPLOYEE 테이블에서 DEPARTMENT 5LGA의 각 직원에 대해 이 정보를 포함하는 메시지를 보내려면 다음을 수행해야 합니다.

예 5:

```
SELECT DB2MQ.MQSEND(LASTNAME || ' ' || FIRSTNAME || ' ' || DEPARTMENT)
   FROM EMPLOYEE
   WHERE DEPARTMENT = '5LGA'
```

이 테이블에 정수 AGE 컬럼도 있으면 다음이 포함됩니다.

예 6:

```
SELECT DB2MQ.MQSEND
      (LASTNAME || ' ' || FIRSTNAME || ' ' || DEPARTMENT|| ' ' || char(AGE))
   FROM EMPLOYEE
   WHERE DEPARTMENT = '5LGA'
```

EMPLOYEE 테이블에 ACE 컬럼이 아닌 CLOB 유형의 RESUME 컬럼이 있을 경우, DEPARTMENT 5LGA의 각 고용인에 대한 정보를 포함하는 메시지를 다음과 같이 내보냅니다.

예 7:

```
SELECT DB2MQ.MQSEND
      (clob(LASTNAME) || ' ' || clob(FIRSTNAME) || ' ' ||
       clob(DEPARTMENT) || ' ' || RESUME))
   FROM EMPLOYEE
   WHERE DEPARTMENT = '5LGA'
```

예 8:

마지막으로 다음 예는 유효한 SQL 표현식을 사용하여 메시지 내용을 추출하는 방법을 보여줍니다. 두 번째 테이블 DEPT에는 VARCHAR 컬럼 DEPT_NO 및 DEPT_NAME이 있고 직원의 LASTNAME 및 DEPT_NAME을 포함하는 메시지를 보낼 수 있습니다.

예 8:

```
SELECT DB2MQ.MQSEND(e.LASTNAME || ' ' || d.DEPTNAME) FROM EMPLOYEE e, DEPT d
   WHERE e.DEPARTMENT = d.DEPTNAME
```

메시지 검색

MQSeries DB2 함수를 사용하여 메시지를 받거나 읽을 수 있습니다. 읽는 것과 받는 것의 차이는 읽는 경우 대기행렬의 헤드에서 메시지를 제거하지 않고 메시지를 리턴하지만 받는 경우에는 메시지가 대기행렬에서 제거됩니다. 받는 작업을 사용하여 검색된 메시지는 한 번만 검색할 수 있지만 읽는 작업을 사용하여 검색된 메시지는 여러 번 검색할 수 있습니다. 다음은 이를 보여주는 예입니다.

예 8:

```
VALUES DB2MQ.MQREAD()
```

이 예에서는 기본 서비스 질 규정을 사용하는 기본 서비스가 정의한 대기행렬의 헤드에서 메시지를 포함하는 VARCHAR 문자열을 리턴합니다. 읽을 메시지가 없을 경우, 널(NULL) 값이 리턴된다는 점에 주의하십시오. 이 작업으로 대기행렬은 변경되지 않습니다.

예 9:

```
VALUES DB2MQ.MQRECEIVE('Employee_Changes')
```

위의 예는 기본 규정을 사용하는 Employee_Changes 서비스가 정의한 대기행렬의 헤드에서 메시지를 제거하는 방법을 보여줍니다.

DB2의 강력한 기능 중 하나는 사용자 정의(또는 DB2 제공) 함수에서 테이블을 생성하는 기능입니다. 이 테이블 함수 기능을 사용하여 대기행렬의 내용을 DB2 테이블로 구체화할 수 있습니다. 다음은 이를 보여주는 간단한 예입니다.

예 10:

```
SELECT t.* FROM table ( DB2MQ.MQREADALL()) t
```

이 쿼리는 이 메시지에 대한 메타데이터 및 기본 서비스에서 정의한 대기행렬의 모든 메시지로 구성된 테이블을 리턴합니다. 리턴된 테이블 구조의 전체 정의는 부록에 정의되어 있지만 첫 컬럼은 메시지 내용을 반영하며 나머지 컬럼에는 메타데이터가 포함됩니다. 메시지만 리턴하려면 위의 예를 다음과 같이 바꿀 수 있습니다.

예 11:

```
SELECT t.MSG FROM table (DB2MQ.MQREADALL()) t
```

테이블 함수가 리턴한 테이블은 데이터베이스에서 직접 검색한 테이블과 다르지 않습니다. 따라서 다양한 방법으로 이 테이블을 사용할 수 있습니다. 예를 들어, 테이블 내용을 다른 테이블과 조인하거나 대기행렬의 메시지 수를 셀 수 있습니다.

예 12:

```
SELECT t.MSG, e.LASTNAME
   FROM table (DB2MQ.MQREADALL() ) t, EMPLOYEE e
      WHERE t.MSG = e.LASTNAME
```

예 13:

```
SELECT COUNT(*) FROM table (DB2MQ.MQREADALL()) t
```

테이블 함수에 대한 뷰를 작성하여 테이블 소스가 대기행렬이라는 것을 숨길 수도 있습니다. 예를 들어, 다음 예에서는 NEW_EMPLOYEES 서비스가 참조하는 대기행렬에 대해 NEW_EMP라는 뷰를 작성합니다.

예 14:

```
CREATE VIEW NEW_EMP (msg) AS
   SELECT t.msg FROM table (DB2MQ.MQREADALL()) t
```

이 경우 뷰는 전체 메시지를 포함하는 단일 컬럼과 함께 정의됩니다. 고정된 길이의 두 필드가 포함된 경우와 같이 메시지가 단순하게 구성되면 메시지를 두 개의 컬럼으로 분석하기 위해 DB2 내장 함수를 사용하는 것이 편합니다. 예를 들어, 특정 대기행렬로 전송되는 메시지에는 항상 18문자의 성과 그 뒤에 18문자의 이름이 나온다는 것을 알고 있는 경우, 다음과 같이 별도의 컬럼으로 각 필드를 포함하고 있는 뷰를 정의할 수 있습니다.

예 15:

```
CREATE VIEW NEW_EMP2 AS
   SELECT left(t.msg,18) AS LNAME, right(t.msg,18) AS FNAME
   FROM table(DB2MQ.MQREADALL()) t
```

DB2 Stored Procedure Builder의 새로운 기능인 MQSeries 지원 마법사를 사용하여 분리된 메시지 구조를 컬럼으로 맵핑하는 새로운 DB2 테이블 함수 및 뷰를 작성할 수 있습니다.

이는 하나 이상의 메시지 내용을 데이터베이스로 저장할 경우에 좋습니다. 메시지 내용을 조작하고 저장하는 SQL의 전체 성능을 사용하여 수행할 수 있습니다. 다음은 가장 간단한 예입니다.

예 16:

```
INSERT INTO MESSAGES
   SELECT t.msg FROM table (DB2MQ.MQRECEIVEALL()) t
```

하나의 컬럼 VARCHAR(2000)가 있는 MESSAGES 테이블에서 위의 명령문을 사용하면 기본 서비스 대기행렬의 메시지를 테이블에 삽입합니다. 이 방법을 수정하여 다양한 경우에 사용할 수 있습니다.

응용프로그램 간 연결성

응용프로그램 통합은 많은 솔루션에서 일반적인 요소입니다. 구입한 응용프로그램을 기존 구조에 통합하거나 새로 개발한 응용프로그램만 기존 환경에 통합할 경우, 모두 이기종 서브시스템 콜렉션이 전체적으로 작업할 수 있도록 함께 통합해

야 하는 문제에 직면하게 됩니다. MQSeries는 일반적으로 응용프로그램을 통합하는 필수적인 도구입니다. 대부분의 하드웨어, 소프트웨어 및 언어 환경에서 액세스할 수 있는 MQSeries는 이기종 응용프로그램 콜렉션을 서로 연결하는 방법을 제공합니다.

이 섹션에서는 몇 개의 응용프로그램 통합 시나리오와 DB2와 함께 사용하는 방법에 대해 설명합니다. 주제가 광범위하므로 응용프로그램 통합에 대한 포괄적인 접근은 이 범위에 포함되지 않습니다. 따라서 요청/응답 통신과 MQSeries Integrator 및 Publish/Subscribe의 두 가지 간단한 항목에 초점을 맞춥니다.

요청/응답 통신

요청/응답(R/R) 통신 방법은 하나의 응용프로그램이 다른 응용프로그램의 서비스를 요청하는 일반적인 방법입니다. 이를 수행하는 한 방법은 리퀘스터가 서비스 제공업체에게 수행할 작업을 요청하는 메시지를 보내는 것입니다. 작업이 완료되면 제공업체는 결과(또는 완료 확인만)를 다시 리퀘스터에게 보낼 수 있습니다. 위에서 설명한 기본 메시지 전송 방법 이외에는 보낸 사람의 요청을 서비스 제공업체의 응답과 연결하는 방법이 없습니다. 계속 진행하기 전에 리퀘스터가 응답을 기다리지 않는 경우, 각 응답을 리퀘스터와 연결하는 메커니즘이 사용되어야 합니다. 개발자가 이러한 메커니즘을 만들 필요 없이 MQSeries는 교환시 메시지 상관을 허용하는 상관 ID를 제공합니다.

여러 가지 방법으로 이 메커니즘을 사용할 수 있지만 가장 간단한 방법은 리퀘스터가 다음과 같은 방법 등을 사용하여 알려진 상관 ID로 메시지를 표시하는 것입니다.

예 17:

```
DB2MQ.MQSEND ('myRequester','myPolicy','SendStatus:cust1','Req1')
```

이 명령문은 요청의 상관 ID를 나타내기 위해 위의 MQSEND 명령문에 마지막 매개변수 Req1을 추가합니다.

이 특정 요청에 대한 응답을 받으려면 다음과 같이 해당되는 MQRECEIVE문을 사용하여 이 상관 ID와 일치하는 표시된 서비스가 정의한 첫 메시지를 선택적으로 검색합니다.

예 18:

```
DB2MQ.MQRECEIVE('myReceiver','myPolicy','Req1')
```

요청을 서비스하는 응용프로그램이 사용 중이거나 리퀘스터가 응답을 받기 전에 위의 MQRECEIVE를 실행하면 이 상관 ID와 일치하는 메시지를 찾을 수 없습니다.

서비스 요청 및 상관 ID를 모두 받으려면 다음과 같은 명령문을 사용합니다.

예 19:

```
SELECT msg, correlid FROM
         table (DB2MQ.MQRECEIVEALL('aServiceProvider','myPolicy',1)) t
```

이 명령문은 서비스 aServiceProvider에서 첫 번째 요청의 상관 ID 및 메시지를 리턴합니다.

일단 서비스가 수행되면 aRequester가 설명하는 대기행렬로 응답 메시지를 보냅니다. 그 동안 서비스 리퀘스터는 다른 작업을 할 수 있습니다. 실제로 설정된 시간 내에 초기 서비스 요청에 대한 응답을 받는다는 보장이 없습니다. 이와 같은 응용프로그램 레벨 시간종료는 개발자가 관리해야 합니다. 리퀘스터는 응답을 발견하기 위해 폴링해야 합니다.

이러한 시간과 무관한 비동기 처리의 장점은 리퀘스터 및 서비스 제공업체가 서로 완전히 독립적으로 실행할 수 있다는 점입니다. 또한 응용프로그램이 간헐적으로만 연결되는 환경 및 여러 요청이나 응답이 처리되기 전에 누적되는 배치 위주의 환경에서 사용될 수 있습니다. 이러한 누적은 데이터 웨어하우스나 조작 데이터 저장소를 정기적으로 갱신하는 데이터 웨어하우스 환경에서 주로 사용됩니다.

Publish/Subscribe

단순 데이터 publication: 응용프로그램 통합의 또 다른 일반적인 시나리오는 한 응용프로그램이 다른 응용프로그램에게 이벤트를 통지하는 것입니다. 이는 다른 응용프로그램이 모니터하는 대기행렬로 메시지를 보냄으로써 쉽게 수행할 수 있습니다. 메시지 내용은 사용자 정의 문자열이나 데이터베이스 컬럼으로 구성될 수

있습니다. 단순한 메시지인 경우, 주로 MQSEND 함수를 사용해서 보내야 합니다. 이러한 메시지를 여러 수신자에게 동시에 보내야 할 경우, MQSeries AMI의 분산 목록 기능을 사용할 수 있습니다.

분산 목록은 AMI 관리 도구로 정의됩니다. 분산 목록은 개별 서비스 목록으로 구성됩니다. 분산 목록으로 보낸 메시지는 목록에서 정의된 모든 서비스로 전달됩니다. 이는 일부 서비스가 항상 모든 메시지와 관련이 있다는 것을 알고 있을 경우 특히 유용합니다. 다음 예는 분산 목록 interestedParties에 메시지를 보내는 경우입니다.

예 20:

```
DB2MQ.MQSEND('interestedParties','information of general interest');
```

특정 서비스를 받아야 하는 메시지를 더 잘 제어해야 할 경우, Publish/Subscribe 기능이 필요합니다. Publish/Subscribe 시스템은 일반적으로 여러 subscribers에게 메시지를 받기 위해 여러 subscribers가 등록할 수 있는 확장 가능한 보안 환경을 제공합니다. 이 기능을 지원하기 위해 MQSeries Integrator 또는 MQSeries Publish/Subscribe 기능과 함께 MQPublish 인터페이스를 사용할 수 있습니다.

MQPublish를 사용하여 사용자는 선택적으로 메시지와 연관된 주제를 지정할 수 있습니다. 주제를 사용하여 subscriber는 받을 메시지를 더욱 명확하게 지정할 수 있습니다. 다음과 같은 단계를 수행합니다.

1. MQSeries 관리자가 MQSeries Integrator publish/subscribe 기능을 구성합니다.
2. 관심있는 응용프로그램은 MQSI 구성에서 정의한 subscription 지점에 subscribe 하여 선택적으로 관심있는 주제를 지정합니다. 각 subscriber는 관련 주제를 선택하고 MQSeries Integrator V2의 내용에 기초한 subscription을 이용할 수도 있습니다. 서비스 이름으로 표시된 대기행렬이 subscriber를 정의함에 주의하십시오.
3. DB2 응용프로그램은 서비스 지점 Weather에 메시지를 발행합니다. 메시지는 weather가 Austin 주제의 경우 Sleet임을 나타내므로 관심있는 subscribers에게 Austin의 weather가 Sleet임을 통지합니다.

4. 실제로 메시지를 발행하는 메커니즘은 DB2가 제공하는 MQSeries 함수에 의해 처리됩니다. 메시지는 Weather라는 이름의 서비스를 사용하여 MQSeries Integrator로 보내집니다.
5. MQSI는 Weather 서비스에서 메시지를 받아 MQSI 구성에서 정의한 처리를 수행하며 충족시키는 subscriptions를 판별합니다. 그런 다음 MQSI는 메시지를 기준에 맞는 subscriber 대기행렬에 전달합니다.
6. Weather 서비스에 subscribe하고 Austin에 관심이 있다고 등록한 응용프로그램은 수신하는 서비스에서 `Sleet` 메시지를 받습니다.

모든 기본값 및 널(NULL) 주제를 사용하여 이 데이터를 발행하려면 다음 명령문을 사용해야 합니다.

예 21:

```
SELECT DB2MQ.MQPUBLISH
                  (LASTNAME || ' ' || FIRSTNAME || ' ' ||
                   DEPARTMENT|| ' ' ||char(AGE))
   FROM EMPLOYEE
   WHERE DEPARTMENT = '5LGA'
```

모든 매개변수를 완벽하게 지정하고 LASTNAME만 포함하도록 메시지를 단순화하면 명령문은 다음과 같습니다.

예 22:

```
SELECT DB2MQ.MQPUBLISH('HR_INFO_PUB', 'SPECIAL_POLICY', LASTNAME,
   'ALL_EMP:5LGA', 'MANAGER')
   FROM EMPLOYEE
   WHERE DEPARTMENT = '5LGA'
```

이 명령문은 SPECIAL_POLICY 서비스를 사용하여 메시지를 HR_INFO_PUB publication 서비스에 발행합니다. 메시지는 보낸 사람이 MANAGER 주제임을 나타냅니다. 주제 문자열은 ':'를 사용하여 연결한 여러 주제를 지정할 수 있음을 보여줍니다. 이 예에서 두 개의 주제를 사용함으로써 subscribers는 ALL_EMP 또는 단지 5LGA에만 등록하여 이 메시지를 받을 수 있습니다.

발행된 메시지를 받으려면 먼저 주어진 주제를 포함하는 메시지에 대한 관련 사항을 등록하고 메시지를 받아야 하는 subscriber 서비스 이름을 표시해야 합니다. AMI subscriber 서비스는 중개인 서비스 및 수신자 서비스를 정의합니다. 중개인

서비스는 subscriber가 publish/subscribe 중개인과 통신하는 방법이며 수신자 서비스는 subscription 요청과 일치하는 메시지를 보낼 위치입니다. 다음 명령문은 ALL_EMP 주제에 대한 관련 사항을 등록합니다.

예 23:

```
DB2MQ.MQSUBSCRIBE('aSubscriber', 'ALL_EMP')
```

일단 응용프로그램이 subscribe하면 ALL_EM 주제와 함께 발행된 메시지가 subscriber 서비스에 의해 정의된 수신자 서비스로 전달됩니다. 응용프로그램은 동시에 여러 개의 subscriptions를 가질 수 있습니다. subscription에 맞는 메시지를 받기 위해 표준 메시지 검색 함수를 사용할 수 있습니다. 예를 들어 subscriber 서비스 aSubscriber가 수신자 서비스를 aSubscriberReceiver로 정의하면 다음 명령문은 첫 번째 메시지를 손상시키지 않으면서 읽습니다.

예 24:

```
DB2MQ.MQREAD('aSubscriberReceiver')
```

메시지 및 메시지가 발행될 주제를 파악하기 위해 테이블 함수 중 하나를 사용할 수 있습니다. 다음 명령문은 aSubscriberReceiver에서 처음 다섯 개의 메시지를 받아 메시지 및 주제를 모두 표시합니다.

예 25:

```
SELECT t.msg, t.topic
   FROM table (DB2MQ.MQRECEIVEALL('aSubscriberReceiver',5)) t
```

주제가 ALL_EMP인 모든 메시지를 읽기 위해 SQL의 기능을 사용하여 다음을 실행할 수 있습니다.

예 26:

```
SELECT t.msg FROM table (DB2MQ.MQREADALL('aSubscriberReceiver')) t
   WHERE t.topic = 'ALL_EMP'
```

주: MQRECEIVEALL이 제한조건과 함께 사용되면 ALL_EMP 주제로 발행된 메시지뿐 아니라 전체 대기행렬이 소비됨에 주의해야 합니다. 이는 제한조건이 적용되기 전에 테이블 함수가 수행되기 때문입니다.

특정 주제에 subscribe하는 데 더 이상 관련 사항이 없으면 다음과 같은 명령문을 사용하여 명시적으로 unsubscribe해야 합니다.

예 27:

```
DB2MQ.MQUNSUBSCRIBE('aSubscriber', 'ALL_EMP')
```

일단 이 명령문이 실행되면 publish/subscribe 중개인은 이 subscription과 일치하는 메시지를 더 이상 배달하지 않습니다.

자동 publication: 데이터베이스 메시지의 또 다른 중요한 기법은 자동 publication입니다. DB2의 트리거 기능을 사용하여 트리거 호출의 일부로 메시지를 자동으로 발행할 수 있습니다. 자동 데이터 publication을 위한 다른 방법들도 있지만 트리거에 기반한 접근 방법을 사용하면 관리자나 개발자가 메시지 내용을 더 자유롭게 구성할 수 있으며 트리거 조치를 더 융통성 있게 정의할 수 있습니다. 트리거를 사용할 경우에는 빈도 및 실행 비용에 주의해야 합니다. 다음은 MQSeries DB2 함수와 함께 트리거를 사용하는 방법을 보여주는 예입니다.

아래 예는 새 직원이 고용될 때마다 쉽게 메시지를 발행하는 방법을 보여줍니다. NEW_EMP에 대한 관심을 등록하고 HR_INFO_PUB 서비스에 subscribe한 사용자나 응용프로그램은 각각의 새 직원의 날짜, 이름 및 부서를 포함하는 메시지를 받게 됩니다.

예 28:

```
CREATE TRIGGER new_employee AFTER INSERT ON employee REFERENCING NEW AS n
      FOR EACH ROW MODE DB2SQL
      VALUES DB2MQ.MQPUBLISH('HR_INFO_PUB&',  'NEW_EMP',
      current date || ' ' || LASTNAME || ' ' || DEPARTMENT)
```

enable_MQFunctions

enable_MQFunctions

지정된 데이터베이스에 대한 DB2 MQSeries 함수를 사용 가능하게 하고 DB2 MQSeries 함수가 적절히 실행될 수 있음을 확인합니다. MQSeries 및 MQSeries AMI가 설치되거나 구성되어 있지 않으면 명령은 실패합니다.

권한 부여

다음 중 하나의 권한으로 수행하십시오.

- *sysadm*
- *dbadm*
- 데이터베이스의 IMPLICIT_SCHEMA(함수의 명시적 또는 내재된 스키마 이름이 없는 경우)
- 스키마에서 CREATEIN 특권(스키마 이름 DB2MQ가 있는 경우)

명령 구문

```
►►—enable_MQFunctions—-n—database—-u—userid—-p—password—┬───────┬——————————►
                                                         └─force─┘

►─┬────────────┬——————————————————————————————————————————————————————————►◄
  └─noValidate─┘
```

명령 매개변수

-n database
: 사용 가능한 데이터베이스 이름을 지정합니다.

-u userid
: 데이터베이스에 연결할 사용자 ID를 지정합니다.

-p password
: 사용자 ID의 암호를 지정합니다.

-force 재설치시 발생한 경고를 무시하도록 지정합니다.

-noValidate
: SDB2 MQSeries 함수가 유효화되지 않도록 지정합니다.

예

다음 예에서는 DB2MQ 함수가 작성됩니다. 사용자는 SAMPLE 데이터베이스에 연결합니다. 기본 스키마 DB2MQ가 사용됩니다.

```
enable_MQFunctions -n sample -u user1 -p password1
```

사용 참고

DB2 MQ 함수는 이 명령에서 자동으로 작성되는 DB2MQ 스키마에서 실행됩니다.

이 명령을 실행하기 전에 다음을 수행하십시오.

- MQ 및 AMI가 설치되어 있고 MQSeries 버전이 5.1 이상인지 확인하십시오.
- 환경 변수 $AMT_DATA_PATH가 정의되어 있는지 확인하십시오.
- 디렉토리를 DB2PATH의 서브디렉토리 cfg로 변경하십시오.

UNIX인 경우 다음을 수행하십시오.

- db2set를 사용하여 AMT_DATA_PATH를 DB2ENVLIST에 추가합니다.
- UDF 실행과 연관된 사용자 어카운트가 mqm 그룹의 구성원인지 확인하십시오.
- 이 명령을 호출할 사용자가 mqm 그룹의 구성원인지 확인하십시오.

주: AIX 4.2는 MQSeries 5.2에서 지원되지 않습니다.

disable_MQFunctions

disable_MQFunctions

지정된 데이터베이스에 대해 DB2 MQSeries 함수를 사용하지 않습니다.

권한 부여

다음 중 하나의 권한으로 수행하십시오.

- *sysadm*
- *dbadm*
- 데이터베이스의 IMPLICIT_SCHEMA(함수의 명시적 또는 내재된 스키마 이름이 없는 경우)
- 스키마에서 CREATEIN 특권(스키마 이름 DB2MQ가 있는 경우)

명령 구문

```
>>--disable_MQFunctions---n--database---u--userid---p--password----------------------><
```

명령 매개변수

-n database
: 데이터베이스 이름을 지정합니다.

-u userid
: 데이터베이스에 연결할 사용자 ID를 지정합니다.

-p password
: 사용자 ID의 암호를 지정합니다.

예

다음 예에서는 DB2MQ 함수를 SAMPLE 데이터베이스에 대해 사용하지 않습니다.

```
disable_MQFunctions -n sample -u user1 -p password1
```

제 4 부 관리 도구

도구를 수행하기 전의 추가 설정

Linux에서 부동 소수점 스택 사용 안함

glibc 2.2.x가 있는 Linux 환경에서 제어 센터와 같은 DB2 Java GUI 도구를 수행하기 전에 부동 소수점 스택을 사용하지 않아야 합니다. 부동 소수점 스택을 사용하지 않으려면, 다음과 같이 LD_ASSUME_KERNEL 환경 변수를 2.2.5로 설정하십시오.

```
bash$ export LD_ASSUME_KERNEL=2.2.5
```

일본어 Linux 환경에서 필요한 특정 Java 레벨

일본어 Linux 환경에서의 제어 센터와 같이 DB2 Java GUI 도구를 수행할 때 Linux 사용자에게는 특정 JDK 레벨이 필요합니다. 예를 들어, Red Hat Linux 6.2J/7J/7.1/7.2 사용자는 IBMJava118-SDK-1.1.8-2.0.i386.rpm 레벨을 사용해야 합니다.

제어 센터

경로 재지정 리스토어 선택으로 데이터베이스 리스토어 커미트

GUI 도구를 사용하여 데이터베이스를 리스토어할 때, 경로 재지정된 리스토어 옵션을 선택하면 데이터베이스 리스토어가 커미트됩니다. 이 옵션을 선택하면 리스토어 조작이 백그라운드에서 시작되고 데이터베이스가 리스토어 보류 상태가 됩니다. 그런 다음, 조치를 취소하면, 다른 리스토어를 완료할 때까지 데이터베이스가 사용 불가능하게 됩니다.

VSE 및 VM 서버용 DB2 서버 관리 능력

DB2 Universal Database 버전 7 제어 센터는 VSE 및 VM용 DB2 서버 데이터베이스의 지원을 향상시켰습니다. 모든 VSE 및 VM용 DB2 서버 데이터베이스 오브젝트는 제어 센터에서 볼 수 있습니다. CREATE INDEX, REORGANIZE INDEX 및 UPDATE STATISTICS 명령문과 REBIND 명령에 대한 지원도 있습니다. REORGANIZE INDEX 및 REBIND에서는 VSE 및 VM용 DB2 서버 호스트에서 저장 프로시저가 수행되고 있어야 합니다. 이 저장 프로시저는 VSE 및 VM용 DB2 서버의 VSE 및 VM용 제어 센터에서 제공합니다.

완전히 통합된 제어 센터는 사용자가 DB2 서버가 수행되는 플랫폼에 상관없이 DB2를 관리할 수 있게 합니다. VSE 및 VM용 DB2 서버 오브젝트는 DB2 Universal Database 오브젝트와 함께 제어 센터 기본 창에 표시됩니다. 이들 오브젝트를 관리하기 위한 해당 조치와 유틸리티는 오브젝트를 선택하여 호출합니다. 예를 들어, 사용자는 특정 데이터베이스의 인덱스를 나열하고 인덱스 중 하나를 선택하여 재구성할 수 있습니다. 사용자는 데이터베이스의 테이블을 나열하고 통계를 갱신하거나 테이블을 복제 소스로 정의할 수도 있습니다.

DB2 Connect 사용자 안내서 또는 *설치 및 구성 보충 설명서*에서 VSE 및 VM용 DB2 Server 오브젝트에서 관리 태스크를 수행하기 위한 제어 센터 구성에 대한 자세한 내용을 참조하십시오.

제어 센터에 대한 Java 1.2 지원

제어 센터는 Java 1.2에서 bi-di 지원을 사용하여 아랍어와 히브리어 같은 양방향 언어를 지원합니다. 이 지원은 Windows NT 플랫폼에서만 제공됩니다.

Java 1.2는 제어 센터가 이를 인식하고 사용할 수 있도록 설치해야 합니다.

1. JDK 1.2.2는 DB2 UDB CD의 DB2\bidi\NT 디렉토리에서 사용할 수 있습니다. `ibm-inst-n122p-win32-x86.exe`는 설치 프로그램이고 `ibm-jdk-n122p-win32-x86.exe`는 JDK 분산 프로그램입니다. 두 파일을 모두 하드 드라이브의 임시 디렉토리에 복사한 다음, 거기에서 설치 프로그램을 수행하십시오.
2. `<DB2PATH>\java\Java12`에 설치하십시오. 여기서, <DB2PATH>는 DB2의 설치 경로입니다.
3. JDK/JRE 설치에서 프롬프트를 표시할 때 시스템 VM으로 JDK/JRE를 선택하지 마십시오.

Java 1.2가 성공적으로 설치되면, Java 1.2를 사용하여 보통 방식으로 제어 센터를 시작합니다.

Java 1.2의 사용을 중지하려면, `<DB2PATH>\java\Java12`에서 JDK/JRE를 설치 제거하거나 `<DB2PATH>\java\Java12` 서브디렉토리의 이름을 바꾸십시오.

주: `<DB2PATH>\java\Java12`를 `<DB2PATH>\Java12`와 혼동하지 마십시오. `<DB2PATH>\Java12`는 DB2 설치의 일부이며 Java 1.2에 대한 JDBC 지원이 포함됩니다.

Windows 운영 체제에서 온라인 도움말 사용시 ″유효하지 않은 단축 아이콘″ 오류

제어 센터 온라인 도움말을 사용할 때, ″유효하지 않은 단축 아이콘″과 같은 오류가 발생할 수 있습니다. 최근에 새로운 웹 브라우저나 웹 브라우저의 새로운 버전을 설치한 경우, HTML 및 HTM 문서가 올바른 브라우저와 연관되었는지 확인하십시오. Windows 도움말 항목 ″파일을 열 때 시작되는 프로그램 변경″을 참조하십시오.

단축키가 작동하지 않음

일부 언어에서는 Unix 기반 시스템과 OS/2의 제어 센터에 대해 일부 키보드 단축키(핫키)가 작동되지 않습니다. 마우스를 사용하여 옵션을 선택하십시오.

OS/2의 Java 제어 센터

제어 센터는 HPFS로 포맷된 드라이브에 설치해야 합니다.

Windows 운영 체제의 저널에서 완료된 작업을 보려고 할 때 ″파일 액세스 거부″ 오류

Windows NT용 DB2 Universal Database에서 스크립트 센터에서 작성된 작업의 세부사항을 보기 위해 저널을 열려고 시도하면 ″파일 액세스 거부″ 오류가 발생합니다. 작업 상태는 완료를 나타냅니다. 이 경우는 스크립트 센터에서 작성된 작업에 START 명령이 포함되면 발생합니다. 이 오류를 방지하려면, 두 배치 파일과 작업 자체에서 START 대신 START/WAIT를 사용하십시오.

멀티사이트 갱신 테스트 연결

버전 7 제어 센터의 멀티사이트 갱신 테스트 연결 기능은 목표 인스턴스의 버전에 제한을 받습니다. ″리모트″ 테스트 연결 기능을 수행하려면 목표 인스턴스는 최소한 버전 7이어야 합니다. 버전 6에서 멀티사이트 갱신 테스트 연결 기능을 수행하려면, 목표 인스턴스에서 제어 센터를 로컬에서 시작하고 수행해야 합니다.

OS/390용 DB2 제어 센터

OS/390용 DB2 UDB 제어 센터는 사용권이 부여된 IBM DB2 유틸리티의 사용을 관리할 수 있게 합니다. 별도로 주문할 수 있는 OS/390용 DB2 UDB 기능인 유틸리티 기능은 DB2 제어 센터가 관리하기 전에 사용자의 환경에서 사용권을 부여받고 설치해야 합니다.

OS/390용 DB2 서브시스템을 구성할 때 제어 센터를 통해 정의된 "CC390" 데이터베이스는 제어 센터의 내부 지원용으로 사용됩니다. 이 데이터베이스를 수정하지 마십시오.

OS/390용 DB2 버전 7.1이 제어 센터 목차나 정보 센터 태스크 정보에 특별히 언급되지 않았더라도 이 문서는 OS/390용 DB2 버전 7.1 기능을 지원합니다. 많은 OS/390용 DB2 버전 6 고유 기능도 OS/390용 DB2 버전 7.1과 관련되어 있으며 목차의 OS/390용 DB2 버전 7.1의 일부 고유 기능에는 버전 지정이 없습니다. 제어 센터에서 OS/390용 DB2 버전 7.1 서브시스템을 구성한 경우, 이 버전의 모든 문서에 액세스할 수 있습니다.

OS/390용 DB2 제어 센터에서 DDL 생성 함수에 액세스하여 사용하려면, DDL 생성 함수를 설치해야 합니다.

- 버전 5의 경우, OS/390용 DB2 버전 5와 함께 DB2Admin 2.0을 설치하십시오.
- 버전 6의 경우, OS/390용 DB2 버전 6의 DB2 Admin 기능에 대한 PTF로 사용 가능한 작은 프로그래밍 향상 기능을 설치하십시오.
- 버전 7.1의 경우, DDL 생성 함수는 별도로 가격이 설정된 OS/390용 DB2 버전 7.1의 DB2 관리 기능의 일부입니다.

Stored Procedure Builder는 DB2 UDB 제어 센터를 시작하기 전에 설치되어야 제어 센터에서 액세스할 수 있습니다. 이것은 DB2 Application Development Client의 일부입니다.

워크스테이션에서 직접 OS/390용 DB2 서브시스템을 카탈로그하려면, 클라이언트 구성 지원 프로그램 도구를 사용하도록 선택하십시오.

1. 소스 페이지에서 **데이터베이스에 연결 수동 구성** 단일선택 버튼을 지정하십시오.
2. 프로토콜 페이지에서 적합한 통신 정보를 완성하십시오.
3. 데이터베이스 페이지에서 **데이터베이스 이름** 필드에 서브시스템 이름을 지정하십시오.
4. 노드 옵션 페이지에서 **노드 옵션 구성(선택적)** 선택란을 선택하십시오.
5. **운영 체제** 필드의 목록에서 MVS/ESA, OS/390을 선택하십시오.

6. **완료**를 눌러 구성을 완료하십시오.

게이트웨이 머신을 통해 OS/390용 DB2 서브시스템을 카탈로그하려면, 게이트웨이 머신에서 위의 1-6단계를 따르고 다음과 같이 수행하십시오.

1. 클라이언트 머신에서 제어 센터를 시작하십시오.
2. 시스템 폴더를 마우스 오른쪽 버튼으로 누르고 **추가**를 선택하십시오.
3. 시스템 추가 대화 상자에서 **시스템 이름** 필드에 게이트웨이 머신 이름을 입력하십시오.
4. DB2DAS00을 **리모트 인스턴스** 필드에 입력하십시오.
5. TCP/IP 프로토콜의 경우, 프로토콜 매개변수에서 **호스트 이름** 필드에 게이트웨이 머신의 호스트 이름을 지정하십시오.
6. 523을 **서비스 이름** 필드에 입력하십시오.
7. **확인**을 눌러 시스템을 추가하십시오. 이제 **시스템** 폴더 아래에 추가된 게이트웨이 머신이 표시될 것입니다.
8. 게이트웨이 머신 이름을 확장하십시오.
9. 인스턴스 폴더를 마우스 오른쪽 버튼으로 누르고 **추가**를 선택하십시오.
10. 인스턴스 추가 대화 상자에서 **새로 고침**을 눌러 게이트웨이 머신에서 사용 가능한 인스턴스를 나열하십시오. 게이트웨이 머신이 Windows NT 시스템인 경우, OS/390용 DB2 서브시스템은 인스턴스 DB2에 카탈로그되었을 것입니다.
11. 인스턴스를 선택하십시오. 이 인스턴스에 대한 프로토콜 매개변수는 자동으로 채워집니다.
12. **확인**을 눌러 인스턴스를 추가하십시오.
13. **인스턴스** 폴더를 열어 방금 추가한 인스턴스를 확인하십시오.
14. 인스턴스를 확장하십시오.
15. **데이터베이스** 폴더를 마우스 오른쪽 버튼으로 누르고 **추가**를 선택하십시오.
16. **새로 고침**을 눌러 게이트웨이 머신 상의 로컬 데이터베이스를 표시하십시오. 데이터베이스 추가 대화 상자에서 DB2 서브시스템을 추가하는 경우, 데이터베이스 이름 필드에 서브시스템 이름을 입력하십시오. 서브시스템(또는 데이터베이스)의 로컬 별명 이름을 입력하십시오(선택적).
17. **확인**을 누르십시오.

이제 제어 센터에서 서브시스템을 성공적으로 추가했습니다. 데이터베이스를 열면, OS/390용 DB2 서브시스템이 표시되는 것을 확인할 수 있습니다.

『제어 센터 390』 섹션의 첫 번째 문단이 다음과 같이 기술되어 있습니다.

OS/390용 DB2 UDB 제어 센터는 사용권이 부여된 IBM DB2 유틸리티의 사용을 관리할 수 있게 합니다. 별도로 주문할 수 있는 OS/390용 DB2 UDB 기능인 유틸리티 기능은 DB2 제어 센터가 관리하기 전에 사용자의 환경에서 사용권을 부여받고 설치해야 합니다.

이 섹션은 이제 다음과 같아야 합니다.

OS/390용 DB2 제어 센터를 사용하면 사용권이 부여된 IBM DB2 유틸리티의 사용을 관리할 수 있습니다. 별도로 주문할 수 있는 요소인 유틸리티 기능은 DB2 제어 센터가 관리하기 전에 사용자의 환경에서 사용권을 부여받고 설치해야 합니다.

OS/390에 대한 제어 센터의 필수 수정

APAR PQ36382를 OS/390용 DB2 버전 5 및 OS/390용 DB2 버전 6의 390 Enablement 기능에 적용하여 버전 7용 DB2 UDB 제어 센터를 사용하여 이들 서브시스템을 관리해야 합니다. 이 수정을 사용하지 않으면 버전 7용 DB2 UDB 제어 센터를 사용하여 이들 서브시스템용 유틸리티를 실행할 수 없습니다.

APAR은 다음의 FMID에 적용해야 합니다.

```
    OS/390용 DB2 버전 5 390 Enablement: FMID JDB551D
OS/390용 DB2 버전 6 390 Enablement: FMID JDB661D
```

공간 계층 작성 대화 상자로 변경

″<<″ 및 ″>>″ 버튼이 공간 계층 작성 대화 상자에서 제거되었습니다.

DB2 제어 센터에 대한 문제점 해결 정보

빠른 시작 책의 "제어 센터 설치 및 구성" 장에서 "문제점 해결 정보"는 제어 센터를 애플릿으로 수행할 때 문제점이 있으면 명령 창에서 클라이언트 브라우저의 CLASSPATH 설정을 해제하도록 알려줍니다. 이 섹션에서는 또한 같은 명령 창에서 브라우저를 시작하라고 합니다. 그러나, 브라우저를 시작하는 명령은 제공되지 않습니다. Internet Explorer를 시작하려면, `start iexplore`를 입력하고 **Enter**를 누르십시오. Netscape를 시작하려면, `start netscape`를 입력하고 **Enter**를 누르십시오. 이 명령들은 브라우저가 PATH에 있는 것으로 가정한 것입니다. 그렇지 않으면, 이를 PATH에 추가하거나 브라우저의 설치 디렉토리로 전환한 후 **start** 명령을 다시 발행하십시오.

UNIX 기반 시스템에서 제어 센터 문제점 해결

UNIX 기반 시스템에서 제어 센터를 시작할 수 없는 경우, JAVA_HOME 환경 변수가 Java 분산을 가리키도록 설정하십시오.

- Java가 `/usr/jdk118`에 설치된 경우, JAVA_HOME을 `/usr/jdk118`로 설정하십시오.
- sh, ksh 또는 bash 쉘의 경우

```
export JAVA_HOME=/usr/jdk118.
```

- csh 또는 tcsh 쉘의 경우

```
setenv JAVA_HOME /usr/jdk118
```

OS/2에서 가능한 정보 팝업 상자 문제점

화면 크기가 1024x768이고 256 색상이며 Workplace Shell Palette Awareness가 사용 가능한 상태에서 OS/2에서 제어 센터를 수행하는 경우, 현재 창의 경계를 넘어서 확장되는 정보 팝업 상자는 검정색 배경에 검정색 텍스트로 표시될 수 있습니다. 이 문제점을 수정하려면, 표시 설정을 256 색상 이상으로 변경하거나 Workplace Shell Palette Awareness를 사용 불가능하게 하십시오.

jdk11_path 구성 매개변수에 대한 도움말

제어 센터 도움말에서 Java Development Kit 1.1 설치 경로(jdk11_path) 구성 매개변수의 설명에 하위 표제 **적용 대상** 아래의 한 행이 누락되었습니다. **적용 대상**의 전체 목록은 다음과 같습니다.

- 로컬 및 리모트 클라이언트가 있는 데이터베이스 서버
- 클라이언트
- 로컬 클라이언트가 있는 데이터베이스 서버
- 로컬 및 리모트 클라이언트가 있는 파티션 데이터베이스 서버
- 로컬 클라이언트가 있는 위성 데이터베이스 서버

스크립트 센터 또는 저널 사용시 Solaris 시스템 오류(SQL10012N)

스크립트 센터나 저널에서 Solaris 시스템을 선택할 때, 다음 오류가 발생할 수 있습니다.

```
SQL10012N - 지정된 라이브러리 "/udbprod/db2as/sqllib/function/unfenced/
db2scdar!ScheduleInfoOpenScan"을 로드하는 동안 예상하지 못한 시스템
오류가 수신되었습니다. SQLSTATE=42724.
```

이것은 Solaris 런타임 링커에서의 버그에 의해 야기됩니다. 이 문제점을 정정하려면, 다음 패치를 적용하십시오.

```
105490-06(107733은 105490을 사용하지 않도록 함)
 Solaris 운영 환경 2.6용
```

DPREPL.DFT 파일에 대한 도움말

제어 센터에서 도구 설정 노트북의 복제 페이지에 대한 도움말에서 5d단계가 다음과 같습니다.

```
   시스템이 기본 파일로 사용할 수 있도록 파일을
제어 센터에 대한 작업 디렉토리(예를 들어, SQLLIB\BIN)에
저장하십시오.
```

5d단계는 다음과 같아야 합니다.

```
   시스템이 기본 파일로 사용할 수 있도록 파일을
제어 센터에 대한 작업 디렉토리(SQLLIB\CC)에
저장하십시오.
```

하나 이상의 제어 센터 애플릿 시작

동일한 머신에서 동시에 하나 이상의 제어 센터 애플릿을 시작할 수 없습니다. 이 제한사항은 지원되는 모든 브라우저에서 수행되는 제어 센터 애플릿에 적용됩니다.

애플릿으로 수행되는 제어 센터에 대한 온라인 도움말

제어 센터가 애플릿으로 수행될 경우, F1 키는 정보 팝업 상자가 있는 창과 노트북에서만 작동합니다.

F1 키를 눌러서 다음과 같은 구성요소에서 정보 팝업 상자를 불러올 수 있습니다.

- OS/390용 DB2 Universal Database
- 마법사

제어 센터의 나머지 구성요소에서 F1은 도움말을 불러오지 않습니다. 다른 구성요소에 대한 도움말을 표시하려면, 도움말 누름 버튼이나 도움말 풀다운 메뉴를 사용하십시오.

애플릿 모드에서 제어 센터 수행(Windows 95)

유효하지 않은 사용자 ID와 암호가 지정된 경우, 스크립트 센터를 열려는 시도가 실패할 수 있습니다. 제어 센터를 시작할 때 유효한 사용자 ID와 암호를 입력하였는지 확인하십시오.

대형 쿼리 결과로 작업

사용자는 많은 행 수를 리턴하는 쿼리를 작성하는 것이 쉽습니다. 사용자가 실제로 리턴할 행 수를 예상하는 것은 쉽지 않습니다. 수천개(또는 수백만개)의 행을 리턴할 수 있는 쿼리를 사용할 경우 두 가지 문제점이 있습니다.

1. 결과를 검색하는 데 시간이 많이 걸립니다.
2. 결과를 보유할 대용량의 클라이언트 메모리가 필요합니다.

이 프로세스를 용이하게 하기 위해 DB2는 대형 결과 세트를 청크로 나눕니다. 쿼리 결과를 한 번에 한 청크씩 검색하여 표시합니다.

따라서,

1. 남은 청크가 검색되는 동안 쿼리의 첫 청크를 표시할 수 있으므로 표시 시간이 줄어듭니다.
2. 주어진 시간에 쿼리 결과의 단 한 개의 청크만 클라이언트에 저장되므로 클라이언트의 메모리 요구사항이 줄어듭니다.

메모리의 쿼리 결과 행 수를 제어하려면 다음을 수행하십시오.

1. 도구 설정 메모장의 일반 페이지를 엽니다.
2. **최대 크기** 섹션에서 다음을 선택합니다.
   - **샘플 내용**을 선택하면 샘플 내용 창에 표시된 결과 행 수가 제한됩니다. 항목 필드에 결과 세트의 청크 *크기*(행 수)를 지정합니다.
   - **명령 센터**를 선택하면 명령 센터의 쿼리 결과 페이지에 표시된 결과 행 수가 제한됩니다. 항목 필드에 결과 세트의 청크 *크기*(행 수)를 지정합니다.

샘플 내용 창의 쿼리 결과나 명령 센터의 쿼리 결과 페이지로 작업할 때 **메모리의 행** 필드는 쿼리용 메모리에 있는 행 수를 나타냅니다. 이 수는 설정된 **최대 크기**보다 커서는 안됩니다. **다음**을 눌러 결과 세트의 다음 청크를 검색합니다. **다음**이 비활성화 되어 있으면 결과 세트의 끝에 도달한 것입니다.

명령 센터

명령 센터 대화식 페이지의 명령문 종료자 인식

명령 센터의 대화식 페이지는 이제 도구 설정에 지정된 명령문 종료 문자를 인식합니다. 명령문 종료 문자가 지정되지 않은 경우, 줄 바꾸기 문자가 기본값으로 사용됩니다.

정보 센터

Java 샘플 문서의 정정사항

정보 센터의 Java 샘플 문서는 Java 샘플 소스에 연결되어 있습니다. 이 소스의 PluginEx.Java 섹션은 최신 정보가 아닙니다. 정보 센터 확장에 대한 최신 정보는 Java 샘플 README 파일, PluginEx.java 파일, 113 페이지의『제어 센터 확장 예제』에서 참조하십시오.

Windows 플랫폼에서, README 및 PluginEx.java 파일은 `x:\sqllib\samples\java`에 있습니다. 여기서, *x*는 DB2가 설치된 드라이브입니다.

UNIX에서, README 및 PluginEx.java 파일은 `/u/db2inst1/sqllib/samples/java`에 있습니다. 여기서, */u/db2inst1*은 DB2가 설치된 디렉토리를 나타냅니다.

Windows 운영 체제에서 ″유효하지 않은 단축 아이콘″ 오류

정보 센터를 사용할 때, ″유효하지 않은 단축 아이콘″과 같은 오류가 발생할 수 있습니다. 최근에 새로운 웹 브라우저나 웹 브라우저의 새로운 버전을 설치한 경우, HTML 및 HTM 문서가 올바른 브라우저와 연관되었는지 확인하십시오. Windows 도움말 항목 ″파일을 열 때 시작되는 프로그램 변경″을 참조하십시오.

Netscape가 이미 열려 있을 때 Netscape Navigator에서 외부 웹 링크 열기(Unix 기반 시스템)

Netscape Navigator가 이미 열려 있어서 로컬 DB2 HTML 문서나 외부 웹 사이트를 표시하고 있을 경우, 정보 센터로부터 외부 웹 사이트를 열려고 하면, Netscape 오류가 발생합니다. ″Netscape가 <외부 웹 사이트>라는 파일이나 디렉토리를 찾을 수 없습니다″라는 오류가 표시됩니다.

이 문제점을 해결하려면, 외부 웹 사이트를 열기 *전에* 열려 있는 Netscape 브라우저를 닫으십시오. Netscape를 다시 시작하여 외부 웹 사이트를 엽니다.

Netscape가 이미 열려 있는 상태에서 로컬 DB2 HTML 문서를 열려고 할 때에는 이 오류가 발생하지 않습니다.

정보 센터 시작 문제점

일부 시스템에서 시작 메뉴, 첫 단계 또는 **db2ic** 명령을 사용하여 정보 센터를 호출할 경우 시작하는데 시간이 오래 걸릴 수도 있습니다. 이러한 문제점이 발생하면, 제어 센터를 시작한 후 도움말 --> 정보 센터를 선택하십시오.

Stored Procedure Builder

z/OS 또는 OS/390용 Java 스토어드 프로시저 지원

FixPak 7에서, z/OS 또는 OS/390용 DB2 버전 7의 해석된 Java 스토어드 프로시저를 빌드하기 위해 다음과 같은 개선된 기능이 Stored Procedure Builder에 추가되었습니다.

- 실제 비용 지원
- 향상된 오류 메시지 처리
- LINUX/390 서버 사용 가능
- 스토어드 프로시저 스키마 이름에서 @ 기호 지원

컴파일된 Java 스토어드 프로시저는 z/OS 또는 OS/390의 어느 버전에서도 지원되지 않으므로 Stored Procedure Builder를 사용하여 작성할 수 없습니다. 이것은 모든 DB2 버전에 해당됩니다.

요구사항:

- z/OS 또는 OS/390용 DB2에 관한 전제조건 및 설정 태스크에 대해서는 APAR PQ52329를 참조하십시오.
- 콜렉션 ID는 z/OS 또는 OS/390에서 JDBC 드라이버를 바인드할 때 사용한 ID와 일치해야 합니다.
- DB2SPB.ini 파일을 수정하여 다음 항목을 포함하십시오.

```
SPOPTION_WLM_JAVA_ENVIRONMENT = WLMENVJ
SPOPTION_JAVAPROC_BUILDER = SYSPROC.DSNTJSPP
SPOPTION_BIND_OPTIONS_JAVA = ACT(REP)
SPOPTION_COLLIDJ = DSNJDBC
```

Stored Procedure Builder를 사용하여 z/OS 또는 OS/390용 Java 스토어드 프로시저를 작성하려면 다음을 수행하십시오.

1. Java 저장 프로시저 삽입 마법사를 여십시오.

a. 프로젝트 트리의 z/OS 또는 OS/390 데이터베이스 연결 아래에서 스토어드 프로시저 폴더를 마우스 오른쪽 단추로 누르십시오.

b. **삽입 --> 마법사를 사용하여 Java 스토어드 프로시저**를 누르십시오. 마법사가 열립니다.

2. z/OS 또는 OS/390 옵션을 지정하여 마법사를 완료하십시오.
   - 콜렉션 ID는 z/OS 또는 OS/390에서 JDBC 드라이버가 바인드되었을 때 BIND PACKAGE(collid)에 지정된 ID와 일치해야 합니다.
   - Java 패키지의 디폴트값은 프로시저 이름이지만, 원하는 이름으로 수정할 수 있습니다.
3. **확인**을 누르십시오. 저장 프로시저가 작성되고 프로젝트 트리에 나열됩니다.
4. 스토어드 프로시저를 마우스 오른쪽 단추로 누르고 **빌드**를 누르십시오.

z/OS 또는 OS/390용 SQL 스토어드 프로시저 지원

FixPak 7에서, Stored Procedure Builder는 z/OS 또는 OS/390용 DB2 버전 7(APAR JR16764)의 스토어드 프로시저를 빌드할 때 ALTER 프로시저를 사용하도록 향상되었습니다.

Stored Procedure Builder를 사용하여 z/OS 또는 OS/390용 SQL 스토어드 프로시저를 작성하려면 다음을 수행하십시오.

1. SQL 스토어드 프로시저 삽입 마법사를 여십시오.
   a. 프로젝트 트리의 z/OS 또는 OS/390 데이터베이스 연결 아래에서 스토어드 프로시저 폴더를 마우스 오른쪽 단추로 누르십시오.
   b. **삽입 --> 마법사를 사용하여 SQL 스토어드 프로시저**를 누르십시오. 마법사가 열립니다.
2. z/OS 또는 OS/390 옵션을 지정하여 마법사를 완료하십시오.
   - 콜렉션 ID는 z/OS 또는 OS/390에서 JDBC 드라이버가 바인드되었을 때 BIND PACKAGE(collid)에 지정된 ID와 일치해야 합니다.
   - SQL 패키지의 디폴트값은 프로시저 이름이지만, 원하는 이름으로 수정할 수 있습니다.
3. **확인**을 누르십시오. 저장 프로시저가 작성되고 프로젝트 트리에 나열됩니다.

4. 저장 프로시저를 마우스 오른쪽 버튼으로 누르고 **빌드**를 선택하십시오.

z/OS 또는 OS/390 문서의 Stored Procedure Builder 참조서 갱신

Stored Procedure Builder 온라인 도움말의 "SQL 스토어드 프로시저 개요" 페이지에서 *IBM DB2 Universal Database SQL 프로시저 안내서 및 참조서 버전 6*에 대한 참조사항은 최신 내용이 아닙니다.

z/OS 또는 OS/390 서버에서 SQL 스토어드 프로시저 빌드에 대한 자세한 정보는 다음을 참조하십시오.

- *z/OS 또는 OS/390용 DB2 UDB SQL 참조서*
- *z/OS 또는 OS/390용 DB2 UDB 응용프로그램 프로그래밍 및 SQL 안내서*

결과 세트 등록 정보 설정 지원

FixPak 7에서, Stored Procedure Builder는 결과 세트를 리턴하는 스토어드 프로시저를 실행할 경우에 성능을 향상시켰습니다.

Stored Procedure Builder를 사용하여 테스트용으로 스토어드 프로시저를 실행할 수 있습니다. Stored Procedure Builder를 사용하여 스토어드 프로시저를 실행하면 데이터베이스에 대해 성공적으로 빌드되었는지와 결과 세트가 존재하는지를 테스트할 수 있습니다. 스토어드 프로시저가 대형 결과 세트를 리턴하는 경우, 결과 분할창에 표시되는 행과 컬럼 수를 제한하기를 원할 수 있습니다.

스토어드 프로시저의 결과 세트 등록 정보를 편집하려면 다음을 수행하십시오.

1. **파일 --> 환경 등록 정보**를 누르십시오.
2. 환경 등록 정보 노트북에서 **출력** 탭을 누르십시오.
3. 스토어드 프로시저 결과 세트의 모든 행을 결과 분할창에 표시하려면, **모든 행 표시** 선택란을 선택하십시오. 결과 분할창에 표시되는 행 수를 제한하려면, **모든 행 표시** 선택란을 지우고 **표시할 행 수** 필드에 숫자를 입력하십시오.
4. 스토어드 프로시저 결과 세트의 각 컬럼에 있는 모든 데이터를 결과 분할창에 표시하려면, **각 컬럼의 모든 데이터 표시** 선택란을 선택하십시오. 결과 분할창에 표시되는 컬럼의 너비를 제한하려면, **각 컬럼의 모든 데이터 표시** 선택란을

지우고 **최대 컬럼 너비** 필드에 숫자(문자 수를 나타냄)를 입력하십시오. 데이터가 결과 분할창에 표시될 때 지정된 최대 컬럼 너비에 맞게 절단됩니다.

5. **확인**을 눌러 변경사항을 적용하십시오.

Windows NT의 DB2 데이터베이스에서 프로시저 삭제

이전 버전의 Stored Procedure Builder에서는 Windows NT 시스템에서 실행 중인 DB2 데이터베이스에서 프로시저를 삭제할 때 DROP PROCEDURE 기능이 제대로 작동하지 않았습니다.

FixPak 7에서는 Stored Procedure Builder가 Windows NT 시스템에서 실행되는 DB2 데이터베이스에서 프로시저를 제대로 삭제합니다.

마법사

데이터베이스 작성 마법사에서 Extent 크기 설정

데이터베이스 작성 마법사를 사용하면 새로운 데이터베이스의 사용자 테이블 스페이스에 대한 Extent 크기와 프리페치 크기 매개변수를 설정할 수 있습니다. (그러나 카탈로그 또는 임시 테이블에 대해서는 설정할 수 없습니다.) 이 기능은 마법사의 ″사용자 테이블″ 페이지에서 사용자 테이블 스페이스에 대해 최소한 하나의 컨테이너가 지정된 경우에만 사용 가능합니다.

MQSeries 지원 마법사

DB2 버전 7.2에는 새로운 MQSeries 지원 마법사가 있습니다. 이 마법사는 버전 7.2에서도 새로운 기능인 DB2 MQSeries 함수를 사용하여 MQSeries 대기행렬을 읽는 테이블 함수를 작성합니다. 이 마법사는 스펙에 따라 각 MQSeries 메시지를 구분된 문자열이나 고정된 길이의 컬럼 문자열로 다룰 수 있습니다. 작성된 테이블 함수는 스펙에 따라 문자열을 분석하고 각 MQSeries 메시지를 테이블 함수의 행으로 리턴합니다. 또한 마법사를 사용하여 테이블 함수 최상위에서 뷰를 작성하고 MQSeries 메시지 및 테이블 함수 결과를 미리 열람할 수 있습니다. 마법사는 Stored Procedure Builder 또는 Data Warehouse Center에서 시작할 수 있습니다.

마법사를 실행하기 위해 필요한 요구사항은 다음과 같습니다.

- MQSeries 버전 5.2
- MQSeries Application Messaging Interface(AMI)
- DB2 MQSeries 함수

요구사항에 대한 자세한 내용은 221 페이지의 『MQSeries』를 참조하십시오.

샘플 및 MQSeries 지원 마법사 자습서를 보려면 http://www.ibm.com/software/data/db2/udb/ide의 자습서 섹션을 참조하십시오.

OLE DB 지원 마법사

이 마법사를 사용하여 Microsoft OLE DB 표준을 지원하는 다른 데이터베이스 제공업체에서 데이터를 읽는 테이블 함수를 작성할 수 있습니다. OLE DB 테이블 함수로 읽는 데이터가 포함된 DB2 테이블을 선택적으로 작성할 수 있으며 OLE DB 테이블 함수에 대한 뷰를 작성할 수 있습니다. 마법사는 Stored Procedure Builder 또는 Data Warehouse Center에서 시작할 수 있습니다.

마법사를 실행하기 위해 필요한 요구사항은 다음과 같습니다.

- OLE DB 제공업체(예:Oracle, Microsoft SQL Server)
- OLE DB 지원 함수

샘플 및 OLE DB 지원 마법사 자습서를 보려면 http://www.ibm.com/software/data/db2/udb/ide의 자습서 섹션을 참조하십시오.

제 5 부 비즈니스 인텔리전스

비즈니스 인텔리전스 자습서

개정된 비즈니스 인텔리전스 자습서

FixPak 2에는 버전 7.1에 있는 여러 문제점을 정정한 개정된 비즈니스 인텔리전스 자습서 및 Data Warehouse Center 샘플 데이터베이스가 포함되어 있습니다. 개정된 Data Warehouse Center 샘플 데이터베이스를 적용하려면, 다음을 수행해야 합니다.

샘플 데이터베이스를 아직 설치하지 않았을 경우, 첫 단계 런치 패드를 사용하여 새 샘플 데이터베이스를 작성하십시오. **시작**을 누르고 **프로그램 --> IBM DB2 --> 첫 단계**를 선택하십시오.

이전에 샘플 데이터베이스를 설치했을 경우, 샘플 데이터베이스 DWCTBC, TBC_MD 및 TBC를 삭제하십시오. 보관하고자 하는 데이터를 샘플 데이터베이스에 추가했다면, 샘플 데이터베이스를 삭제하기 전에 백업하십시오. 세 개의 샘플 데이터베이스를 삭제하려면 다음을 수행하십시오.

1. DB2 명령 창을 열려면, **시작**을 누르고 **프로그램 --> IBM DB2 --> 명령 창**을 선택하십시오.
2. DB2 명령 창에서 다음 세 명령을 각각 입력하고 하나를 입력한 뒤 Enter를 누르십시오.

   ```
   db2 drop database dwctbc
   ```

   ```
   db2 drop database tbc_md
   ```

   ```
   db2 drop database tbc
   ```

3. DB2 명령 창을 닫으십시오.
4. 첫 단계 런치 패드를 사용하여 새 샘플 데이터베이스를 작성하십시오. **시작**을 누르고 **프로그램 --> IBM DB2 --> 첫 단계**를 선택하십시오.

DB2 Universal Database 빠른 안내

Quick Tour는 Linux 또는 Linux/390용 DB2에서 사용할 수 없습니다.

Quick Tour는 작은 시스템 글꼴로 수행하도록 최적화되어 있습니다. OS/2에서 Quick Tour를 올바르게 보려면 웹 브라우저의 글꼴 크기를 조정해야 할 수도 있습니다. 글꼴 크기 조정에 대해서는 사용하는 웹 브라우저에 대한 도움말을 참조하십시오. Quick Tour를 올바르게 보려면(SBCS 전용) 8 포인트 Helv 글꼴을 사용하는 것이 좋습니다. 일본어나 한국어 사용자의 경우, 8 포인트 Mincho 글꼴을 사용하는 것이 좋습니다. 기본 글꼴을 설정할 때, 환경설정 창의 글꼴 페이지에서 ″문서에서 지정한 글꼴을 무시하고 기본 글꼴 사용″ 옵션을 선택하십시오.

어떤 경우에는 Quick Tour가 2차 브라우저 창 뒤에서 시작될 수도 있습니다. 이러한 문제점을 정정하려면, Quick Tour를 닫고 12 페이지의 『Netscape 시작시 발생하는 오류 메시지』의 단계를 수행하십시오.

Quick Tour를 시작할 때, 다음과 같은 JavaScript 오류가 표시될 수도 있습니다.

```
file:/C/Program Files/SQLLIB/doc/html/db2qt/index4e.htm, 65행:


창이 정의되어 있지 않습니다.
```

이 JavaScript 오류가 발생하면 Quick Tour가 시작된 후 Quick Tour 시작 페이지인 `index4e.htm`이 자동으로 닫히지 않습니다. `index4e.htm`이 표시되는 브라우저 창을 닫으면 Quick Tour 시작 페이지를 닫을 수 있습니다.

″새로운 기능″ 절의 데이터 관리 주제에서는 ″on-demand 로그 아카이브 지원″이 버전 7.1에서 지원된다고 설명합니다. 이것은 잘못된 내용입니다. 또한 다음 문장도 포함되어 있습니다.

```
로그 파일의 크기가 4GB에서 32GB로 증가되었습니다.
```

이 문장은 다음과 같이 바뀌어야 합니다.

```
사용 중인 전체 로그 공간이 4GB에서 32GB로 증가되었습니다.
```

DB2 Data Links Manager에 대해 설명하는 절에는 다음과 같은 문장이 있습니다.

```
또한, NetBackup을 사용한 백업 및 리스토어를 위한
Veritas XBSA 인터페이스의 사용을 지원합니다.
```

이 문장은 다음과 같이 바뀌어야 합니다.

```
또한, 파일 아카이브 및 리스토어를 위한 XBSA 인터페이스도 지원됩니다.
XBSA 인터페이스를 지원하는 스토리지 관리자에는 Legato NetWorker 및
Veritas NetBackup이 포함됩니다.
```

Data Warehouse Center 관리 안내서

사용 가능한 갱신사항

Data Warehouse Center 관리 안내서는 FixPak 4에서 갱신되었습니다. 최신 PDF는 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다. 또한 갱신된 참조에 이러한 참고 내용을 추가해 놓았습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

웨어하우스 서버 향상

다음은 FixPak 5에서 웨어하우스 서버에 대해 개선된 기능입니다.

구성 매개변수 갱신

서버는 서비스 이름과 같은 중요한 구성 매개변수를 빈 문자열로 더 이상 갱신하지 않습니다.

DWC7906 메시지가 갱신됨

메시지 DWC7906에는 이제 종속 문제점을 보고할 때 선임자의 이름이 포함되어 있습니다.

다음은 FixPak 4에서 웨어하우스 서버에 대해 개선된 기능입니다.

에이전트 종료시 오류(rc = 7170), 2차 rc = 6106.

이 오류는 서버가 종료 요청을 보내기 전에 에이전트를 종료했을 때 발생합니다. 이전에는 불필요하게 이 오류가 보고되었으나 이후로는 보고되지 않을 것입니다.

로그 파일에 기록된 시스템 메시지 및 주석

사용자 정의 프로그램이 실행을 완료하면 시스템 메시지 및 주석이 웨어하우스 로그 파일에 쓰여집니다. 현재 이 메시지는 Work In Progress 표시 창에 표시됩니다.

현재 올바르게 작동하는 증분 커미트

목표 데이터베이스가 상주 단계에 있고 증분 커미트 값이 0보다 클 때 오류가 발생하는 경우 오류 이전에 커미트된 모든 결과 값이 목표 데이터베이스에 표시됩니다. FixPak 4 이전 버전에서는 부분 결과 값이 삭제되었습니다.

정정된 추적 레벨 오류를 변경한 후 웨어하우스 서버를 실행 불가능

웨어하우스 서버는 로그 디렉토리 이름을 시스템 환경 변수인 VWS_LOGGING에서 검색합니다. VWS_LOGGING이 누락되거나 올바르지 않은 디렉토리 이름을 가리키게 되면 TEMP 시스템 환경 변수가 대신 사용됩니다. TEMP가 누락되거나 유효하지 않은 디렉토리 이름을 가리키게 되면 로그 추적 파일이 c:\에 기록됩니다. 이렇게 하면 유효하지 않은 로그 디렉토리 이름을 레지스트리에서 검색할 때 발생한 FixPak 4 이전 버전에서의 오류가 수정됩니다.

저장 프로시저에서의 커미트 명령에 대한 추가 지원

웨어하우스 서버는 사용자 정의 저장 프로시저를 실행한 다음 커미트 명령을 에이전트에 보냅니다.

샘플 목차 확장

웨어하우스 서버가 이제 더 이상 에이전트 종료 메시지를 기다릴 필요가 없기 때문에 샘플 목차가 더 효율적으로 실행됩니다.

현재 추적 로그 파일의 크기 제어 가능

이제 새 시스템 환경 변수인 VWS_SERVER_LOG_MAX를 사용해서 추적 로그 파일의 크기를 제어할 수 있습니다. VWS_SERVER_LOG_MAX 값을 0보다 크게 설정하면, 웨어하우스 서버는 VWS_SERVER_LOG_MAX 값이 나타내는 바이트 크기에 근접하게 도달할 경우 로그 파일의 크기를 더 이상 증가시키지 않습니다. 로그 파일이 최대 크기에 이르면 가장 최신의 추적 로그 항목이 보유되어 가장 오

래된 항목을 겹쳐씁니다. 광범위한 추적을 실행하는데 있어 합리적인 크기는 VWS_SERVER_LOG_MAX=150000000(150M)입니다.

OS/390 에이전트를 사용하여 Trillium Batch System JCL 수행

OS/390 에이전트는 이제 Data Warehouse 메타데이터 임포트 노트북에서 작성된 Trillium Batch System 사용자 정의 프로그램을 지원합니다. 이전에 Trillium Batch System JCL 파일을 실행하려면, 리모트으로 JCL을 수행하는 데 Windows, AIX 또는 Solaris 운영 환경 에이전트를 사용해야 합니다. 이 갱신을 사용하여 OS/390 에이전트에서 JCL을 시작할 수 있습니다.

Trillium Batch System에 대한 메타데이터 임포트 노트북을 사용하여 Trillium Batch System을 작성할 경우, JCL이 에이전트와 동일한 시스템에 있을지라도 OS/390 에이전트에 대한 연결로 리모트 호스트를 선택해야 합니다. 리모트 호스트 연결에 대한 모든 매개변수가 입력되어야 합니다.

Trillium Batch System 사용자 정의 프로그램 단계를 작성한 후에, Trillium Batch System 단계의 등록 정보 노트북을 사용하여, 사용하려는 OS/390 에이전트 사이트로 에이전트 사이트를 변경하십시오.

JCL 또는 출력 오류 파일의 이름에 공백이나 괄호가 들어 있는 경우, **스크립트**, **JCL** 또는 **출력** 오류 파일 필드에 공백이나 괄호를 입력할 때 큰따옴표로 묶어야 합니다.

Data Warehouse Center에서의 두 개의 새로운 샘플 프로그램

두 개의 새로운 샘플 프로그램(EEE_Load 및 File_Wait)이 Data Warehouse Center에 포함됩니다. EEE_Load 프로그램을 사용하여 데이터 웨어하우징 프로세스 내에서 DB2 UDB EEE 자동 로드 프로그램을 실행할 단계를 작성할 수 있습니다. File_Wait 프로그램을 사용하여 파일을 대기할 단계를 작성한 다음, 파일이 사용 가능하게 되면 사용자 프로세스에서 다음 단계를 수행하십시오. 이 프로그램에 대한 자세한 정보는 Data Warehouse Center 서버가 설치된 시스템에서 ..\SQLLIB\TEMPLATES\SAMPLES 디렉토리에 있는 README.UDP 파일을 참조하십시오.

DB2 Warehouse Manager를 사용한 ETI·Extract® 변환 프로그램 관리의 갱신

DB2 Warehouse Manager를 사용한 ETI·Extract® 변환 프로그램 관리 내용이 갱신되었습니다. 이 갱신 내용은 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다.

CWMI(Common Warehouse Metadata Interchange)를 사용하여 메타데이터 임포트 및 익스포트

소개

태그 언어 파일에 대한 기존 지원에 추가하여, Data Warehouse Center는 이제 CWM(Common Warehouse Metamodel) 표준을 준수하는 XML 파일로부터 메타데이터를 임포트 및 익스포트할 수 있습니다. 이들 CWM 준수 XML 파일 임포트 및 익스포트를 *CWMI(Common Warehouse Metadata Interchange)*라고 합니다.

다음 Data Warehouse Center 오브젝트로부터 메타데이터를 임포트 및 익스포트할 수 있습니다.

- 웨어하우스 소스
- 웨어하우스 목표
- 프로세스, 소스, 목표, 단계 및 연쇄 관계를 포함한 주제 영역
- 사용자 정의 프로그램

CWMI 임포트 및 익스포트 유틸리티는 현재 일정한 종류의 메타데이터는 지원하지 않습니다. 여기에는 스케줄, 웨어하우스 스키마, 사용자 및 그룹이 포함됩니다.

Data Warehouse Center는 임포트 및 익스포트 프로세스의 결과가 들어 있는 로그 파일을 작성합니다. 일반적으로, `x:\program files\sqllib\logging` 디렉토리에(여기서, `x:`는 DB2를 설치한 드라이브) 또는 VWS_LOGGING 환경 변수로서 지정한 디렉토리에 로그 파일이 작성됩니다. 로그 파일은 일반 텍스트입니다. 어떤 텍스트 편집기로도 볼 수 있습니다.

메타데이터 임포트

Data Warehouse Center 내에서 또는 명령행에서 메타데이터를 임포트할 수 있습니다.

임포트 프로세스를 통해 작성된 새 오브젝트는 기본 Data Warehouse Center 보안 그룹에 할당됩니다. 추가 정보는 이들 릴리스 정보에서 ″임포트 후 보안 갱신″을 보십시오.

단계에 관한 메타데이터를 임포트할 경우, 여러 파일이 단계와 연관될 수 있습니다. 단계에 관한 메타데이터는 XML 파일에 저장되지만, 때때로 단계에 BLOB으로 저장된 연관 데이터가 있습니다. BLOB 메타데이터의 파일 이름은 XML 파일과 같지만, 번호가 부여된 확장자가 있는 별도 파일로 되어 있습니다. 임포트를 수행할 때 모든 관련된 단계 파일은 같은 디렉토리에 있어야 합니다.

테스트 또는 생산 모드에 있을 때 단계 갱신

단계가 개발 모드에 있어야 Data Warehouse Center가 단계의 메타데이터를 갱신할 수 있습니다. 단계가 테스트 또는 생산 모드에 있으면, 메타데이터를 임포트하기 전에 단계를 개발 모드로 강등하십시오.

1. Data Warehouse Center에 로그온하십시오.
2. 강등하려는 단계를 마우스 오른쪽 버튼으로 누르고 **모드**를 누르십시오.
3. **개발**을 누르십시오.

이제 단계가 개발 모드에 있습니다. 메타데이터를 임포트한 후 단계를 테스트 또는 생산 모드로 되돌리십시오.

Data Warehouse Center에서 데이터 임포트

Data Warehouse Center 내에서 메타데이터를 임포트할 수 있습니다.

1. Data Warehouse Center에 로그온하십시오.
2. 왼쪽 페인에서 **웨어하우스**를 누르십시오.
3. **선택 항목 --> 메타데이터 임포트 --> 파일 교환...**을 누르십시오.
4. 메타데이터 임포트 창에서 임포트하려는 메타데이터가 들어 있는 파일 이름을 지정하십시오. 파일 이름을 입력하거나 파일을 열람할 수 있습니다.

- 위치를 알 경우, 임포트하려는 파일의 완전한 경로와 파일 이름을 입력하십시오. 메타데이터를 XML 형식으로 임포트하도록 지정하려면 `.xml` 파일 확장자를 포함시키십시오. 그렇지 않으면 파일이 올바르게 프로세스되지 않습니다.
- 파일을 찾아보려면 다음을 수행하십시오.
  a. 생략(**...**) 누름 버튼을 누르십시오.
  b. 파일 창에서 **파일 유형**을 XML로 변경하십시오.
  c. 올바른 디렉토리로 가서 임포트하려는 파일을 선택하십시오.

     주: 파일 확장자는 `.xml`이어야 합니다.
  d. **확인**을 누르십시오.
5. **메타데이터 임포트** 창에서 **확인**을 눌러 완료하십시오. Data Warehouse Center 가 파일을 임포트하는 동안 **진행** 창이 표시됩니다.

명령행을 사용하여 메타데이터 임포트

또한 명령행을 사용하여 메타데이터를 임포트할 수 있습니다. 다음은 임포트 명령 구문입니다.

`CWMImport` *XML_file dwcControlDB dwcUserId dwcPW [PREFIX = DWCtbschema]*

XML_file	임포트하려는 XML 파일의 완전한 경로 및 파일 이름(드라이브와 디렉토리 포함). 이 매개변수는 필수입니다.
dwcControlDB	임포트한 메타데이터를 저장할 웨어하우스 제어 데이터베이스의 이름. 이 매개변수는 필수입니다.
dwcUserId	웨어하우스 제어 데이터베이스에 연결하는 데 사용하는 사용자 ID. 이 매개변수는 필수입니다.
dwcPW	웨어하우스 제어 데이터베이스에 연결하는 데 사용하는 사용자 암호. 이 매개변수는 필수입니다.
[PREFIX=DWCtbschema]	Data Warehouse Center 시스템 테이블에 대한 데이터베이스 스키마 이름. `PREFIX=`에 값을 지정하지 않으면, 기본 스키마 이름은 IWH입니다. 이 매개변수는 선택적입니다.

임포트 유틸리티 수행 후 메타데이터 갱신

임포트 후 보안 갱신

보안상의 문제로, Data Warehouse Center는 암호를 임포트하거나 익스포트하지 않습니다. 필요하면 새 오브젝트에서 암호를 갱신해야 합니다. 임포트 고려사항에 관한 세부사항은 *Data Warehouse Center 관리 안내서* 제 12 장 "Data Warehouse Center 메타데이터 익스포트 및 임포트"를 보십시오.

메타데이터를 임포트할 때, 모든 오브젝트는 *기본 보안 그룹*에 할당됩니다. 오브젝트에 액세스 권한을 소유하는 그룹을 변경할 수 있습니다.

1. Data Warehouse Center에 로그온하십시오.
2. 변경하려면 오브젝트가 들어 있는 폴더를 마우스 오른쪽 버튼으로 누르십시오.
3. **등록정보**를 누른 다음, **보안** 탭을 누르십시오.
4. **선택된 웨어하우스 그룹** 목록에서 그룹을 제거하거나 **사용 가능한 웨어하우스 그룹** 목록에 그룹을 추가하십시오.
5. **확인**을 누르십시오.

메타데이터 익스포트

Data Warehouse Center 내에서 또는 명령행에서 메타데이터를 익스포트할 수 있습니다.

일부 단계에는 BLOB으로 저장된 메타데이터가 있습니다. BLOB 메타데이터는 단계의 XML 파일과 이름이 같은 별도 파일로 익스포트하지만, 번호가 부여된 확장자가 붙습니다(`.1`, `.2` 등등).

Data Warehouse Center에서 데이터 익스포트

Data Warehouse Center 내에서 메타데이터를 익스포트할 수 있습니다.

1. Data Warehouse Center에 로그온하십시오.
2. 왼쪽 페인에서 **웨어하우스**를 누르십시오.
3. **선택 항목 --> 메타데이터 익스포트 --> 교환 파일**을 누르십시오.
4. 메타데이터 익스포트 창에서 익스포트하려는 메타데이터가 들어 있는 파일 이름을 지정하십시오. 파일 이름을 입력하거나 파일을 찾아볼 수 있습니다.

- 사용할 파일의 완전한 경로와 파일 이름을 알 경우, **파일 이름** 입력 필드에 입력하십시오. 메타데이터를 XML 형식으로 익스포트하도록 지정하려면 `.xml` 파일 확장자를 포함시키십시오.
- 파일을 찾아보려면 다음을 수행하십시오.
  a. 생략(**...**) 누름 버튼을 누르십시오.
  b. **파일** 창에서 **파일 유형**을 XML로 변경하십시오.
  c. 올바른 디렉토리로 가서 익스포트한 메타베이터를 보관할 파일을 선택하십시오.

     주: 선택한 기존 파일을 익스포트된 메타데이터로 겹쳐씁니다.
  d. **확인**을 누르십시오.
5. 메타데이터 익스포트 창에 올바른 파일 이름이 표시될 때, 메타데이터를 익스포트할 **사용 가능한 오브젝트** 목록에서 오브젝트를 누르십시오.
6. > 기호를 눌러 선택된 오브젝트를 **사용 가능한 오브젝트** 목록에서 **선택된 오브젝트** 목록으로 이동하십시오. 익스포트하려는 모든 오브젝트가 **선택된 오브젝트** 목록에 속할 때까지 반복하십시오.
7. **확인**을 누르십시오.

Data Warehouse Center는 익스포트하기 위해 선택한 Data Warehouse Center 오브젝트에 관한 정보가 들어 있는 *입력* 파일을 작성한 후, 그 오브젝트에 관한 메타데이터를 익스포트합니다. Data Warehouse Center가 메타데이터를 익스포트하는 동안 진행 창이 표시됩니다.

명령행을 사용하여 메타데이터 익스포트

명령행에서 메타데이터를 익스포트하려면, 먼저 *입력* 파일을 작성해야 합니다. 입력 파일은 확장자가 `.INP`인 텍스트 파일이고, 익스포트하려는 오브젝트 유형 별로 모든 오브젝트를 나열합니다. Data Warehouse Center 내에서 익스포트를 수행할 때는 입력파일이 자동으로 작성되지만, 명령행에서 익스포트를 수행할 때는 먼저 입력 파일을 작성해야 합니다. 어떤 텍스트 편집기로도 입력 파일을 작성할 수 있습니다. Data Warehouse Center에 나타나는 대로 모든 오브젝트 이름을 입력하십

시오. 반드시 파일을 읽기/쓰기 디렉토리에 작성하십시오. 익스포트 유틸리티를 수행할 때, Data Warehouse Center는 입력 파일이 있는 곳과 같은 디렉토리에 XML 파일을 씁니다.

다음은 샘플 입력 파일입니다.

```
<PROC>
Tutorial Fact Table Process
<IR>
Tutorial file source
Tutorial target
<UDP>
New Program group
```

<PROC>(processes) 섹션에, 익스포트할 모든 프로세스를 나열하십시오. <IR>(정보 자원) 섹션에, 익스포트할 모든 웨어하우스 소스 및 목표를 나열하십시오. Data Warehouse Center는 이들 소스 및 목표와 연관된 테이블과 컬럼을 자동으로 포함시킵니다. <UDP>(사용자 정의 프로그램) 섹션에서 익스포트할 모든 프로그램 그룹을 나열하십시오.

메타데이터를 익스포트하려면, DOS 명령 프롬프트에서 다음 명령을 입력하십시오.

`CWMExport` *INPcontrol_file dwcControlDB dwcUserID dwcPW* [`PREFIX=`*DWCtbschema*]

INPcontrol_file	익스포트하려는 오브젝트가 들어 있는 `.INP` 파일의 완전한 경로 및 파일 이름(드라이브와 디렉토리 포함). 이 매개변수는 필수입니다.
dwcControlDB	오브젝트를 익스포트할 웨어하우스 제어 데이터베이스의 이름. 이 매개변수는 필수입니다.
dwcUserID	웨어하우스 제어 데이터베이스에 연결하는 데 사용하는 사용자 ID. 이 매개변수는 필수입니다.
dwcPW	웨어하우스 제어 데이터베이스에 연결하는 데 사용하는 암호. 이 매개변수는 필수입니다.
[PREFIX=DWCtbschema]	Data Warehouse Center 시스템 테이블에 대한 데이터베이스 스키마 이름. `PREFIX=`에 값을 지정하지 않으면, 기본값은 IWH입니다. 이 매개변수는 선택적입니다.

태그 언어 메타데이터 임포트/익스포트 유틸리티

키 정의

태그 언어 파일에 정의된 기본 키와 외부 키는 제어 데이터베이스에 정의되어 있는 두 키와 동일하지 않을 경우 무시됩니다. 이미 정의되어 있는 키와 다를 경우 오류가 발생합니다.

스케줄 단계 및 프로세스

임포트 유틸리티는 더 이상 단계 및 프로세스 스케줄을 삭제하지 못합니다. 태그 파일에 정의된 스케줄은 스케줄의 현재 목록에 추가됩니다. 이 때문에 스케줄이 중복되어 나타날 수 있습니다. 사용자는 단계를 제품 모드로 승격하기 전에 중복 스케줄을 삭제해야 합니다.

SAP 단계 정보

논리적으로 불일치하는 테이블 작성 가능성

다음의 조건이 모두 충족되면 결과 대상 테이블이 논리적으로 일치하지 않을 수 있습니다.

1. BO는 GetList 및 GetDetail export 매개변수를 가지며 모든 키 필드를 맵핑했습니다.
2. SAP 단계의 등록 정보 노트북에 있는 출력 매개변수 페이지에서 매개변수를 맵핑하는데 사용한 매개변수와 다른 SAP 매개변수 이름을 가진 GetList export 매개변수를 선택했습니다.

   주: SAP 매개변수 이름은 완전한 이름에서 마침표 앞에 나타나는 매개변수의 부분을 의미합니다. 예를 들어, DocList.DOCNUMBER 매개변수의 경우 ″DocList″가 SAP 매개변수 이름입니다.
3. SAP 단계의 등록 정보 노트북에 있는 출력 매개변수 페이지에서 GetDetail export 매개변수를 선택합니다.

예:

DocumentNumber는 주요 필드입니다. DocList.DOCNUMBER 및 DocNumberSelection.OPTION은 GetList export 매개변수입니다. DocData.USERNAME은 GetDetail export 매개변수입니다.

DocumentNumber를 DocList.DOCNUMBER에 맵핑합니다. (조건 1)

DocNumberSelection.OPTION을 출력 매개변수로 선택합니다. (조건 2. DocNumberSelection 및 DocList는 서로 다른 SAP 매개변수 이름을 가지기 때문.)

DocData.USERNAME을 출력 매개변수로 선택합니다. (조건 3. GetDetail export 매개변수이기 때문.)

이 조건이 충족되면 컬럼 소스가 GetList 및 GetDetail 매개변수인 대상 테이블이 생성됩니다. 그러나 컬럼들 간의 관계가 논리적으로 일관되다는 보장은 없습니다.

SAP Connector 정보

SAP Connector 설치 제한사항

SAP Connector는 SAP R/3 시스템의 영문 설치 버전만 지원합니다.

GetDetail BAPI의 성능

GetDetail에 많은 입력 매개변수가 있으면 GetDetail BAPI의 성능이 떨어집니다.

DB2 OLAP Starter Kit

IBM DB2 OLAP Starter Kit 7.2는 몇몇 운영 체제 플랫폼에서 Oracle, MS-SQL, Sybase, Informix RDBMS(관계형 데이터베이스 관리 시스템)를 추가로 지원합니다. 버전 7.2에는 DB2를 포함하여 지원되는 모든 RDBMS용 스크립트와 도구가 포함됩니다. 몇 가지 제한사항이 있으며, 299 페이지의 『알려진 문제점 및 제한사항』에서 자세한 내용을 참조하십시오.

DB2 Universal Database용 DB2 OLAP Starter Kit 버전 7.2의 서비스 레벨은 Hyperion Essbase 6.1용 패치 2와 HIS(Hyperion Integration Server) 2.0용 패치 2를 더한 것과 같습니다.

OLAP 서버 웹 사이트

DB2 OLAP Starter Kit에 대한 최근 설치 및 사용법 추가 정보는 다음 DB2 OLAP 서버 웹 사이트의 라이브러리 페이지를 보십시오.

http://www.ibm.com/software/data/db2/db2olap/library.html

지원되는 운영 체제 서비스 레벨

버전 7.2용 OLAP Starter Kit의 서버 구성요소는 다음 운영 체제 및 서비스 레벨을 지원합니다.

- Windows NT 4.0 서버 SP 5 및 Windows 2000
- AIX 버전 4.3.3 이상
- Solaris 운영 체제 버전 2.6, 7, 8(Sun OS 5.6, 5.7, 5.8)

클라이언트 구성요소는 Windows 95, Windows 98, Windows NT 4.0 SP5, Windows 2000에서 수행됩니다.

UNIX에서 DB2 OLAP Starter Kit 설정 완료

DB2 OLAP Starter Kit 설치는 UNIX에 대한 DB2 Universal Database 설치의 기본 프로시저를 따릅니다. 제품 파일은 설치 프로그램에 의해 시스템 디렉토리에 놓입니다(AIX: /usr/lpp/db2_07_01, Solaris 운영 환경: /opt/IBMdb2/V7.1).

그러면 인스턴스 작성 단계 중 두 개의 DB2 OLAP 디렉토리(essbase 및 is)가 sqllib 아래의 인스턴스 사용자의 홈 디렉토리에 작성됩니다. 한 번에 OLAP Server의 한 인스턴스만 머신에서 수행될 수 있습니다. 설정을 완료하려면, 사용자는 시스템에서 is/bin 디렉토리에 링크되지 않도록 수동으로 is/bin 디렉토리를 설정해야 합니다. 이것은 인스턴스의 홈 디렉토리 내에서 쓰기 가능한 디렉토리로 링크되어야 합니다.

Solaris 운영 환경에 대한 설치를 완료하려면, 인스턴스 ID를 사용하여 로그온하고 sqllib/is 디렉토리로 변경한 후 다음을 입력하십시오.

```
rm bin
mkdir bin
cd bin
ln -s /opt/IBMdb2/V7.1/is/bin/ismesg.mdb ismesg.mdb
ln -s /opt/IBMdb2/V7.1/is/bin/olapicmd olapicmd
ln -s /opt/IBMdb2/V7.1/is/bin/olapisvr olapisvr
ln -s /opt/IBMdb2/V7.1/is/bin/essbase.mdb essbase.mdb
ln -s /opt/IBMdb2/V7.1/is/bin/libolapams.so libolapams.so
```

Solaris 운영 환경에 대한 추가 구성

Solaris 운영 환경에서는 OLAP Starter Kit가 해당 ODBC 드라이버에 링크되지 않을 경우 오류가 발생할 수 있습니다. 이러한 오류 발생을 방지하려면, 다음 명령을 실행해서 $ARBORPATH/bin에서 링크를 작성하여 sqllib/lib/libdb2.so OLAP 드라이버를 가리키도록 해야 합니다.

```
ln -s  $HOME/sqllib/lib/libdb2.so  libodbcinst.so
```

모든 운영 체제에 대한 추가 구성

DB2 Universal Database 버전 7의 FixPak 3을 시작하면, DB2 OLAP Starter Kit에 Java를 필요로 하는 함수가 포함됩니다. FixPak 3 이상을 설치하면 OLAP 서버 콘솔에 다음 오류 메시지가 표시될 수 있습니다.

```
Can not find [directory] [/export/home/arbor7sk/sqllib/essbase/java/],
required to load JVM.
```

이 오류를 정정하려면 다음 단계대로 실행하십시오.

1. DB2 인스턴스 소유자로 로그온하십시오.
2. DB2 OLAP Starter Kit가 설치된 디렉토리를 찾으십시오. 이 디렉토리의 기본 이름은 `essbase`입니다.
3. essbase 디렉토리에서 `java`라는 서버디렉토리를 작성하십시오.
4. `java` 서브디렉토리에 다음과 같은 빈 파일을 작성하십시오.
   - essbase.jar
   - essdefs.dtd
   - jaxp.jar
   - parser.jar
   - udf.policy

OLAP Starter Kit용 ODBC 구성

IBM DB2 OLAP Starer Kit 7.2에서는 OLAP 통합 서버에서 관계형 데이터 소스와 OLAP 메타데이터 카탈로그로의 ODBC(Open Database Connectivity) 연결 작업을 하려면 ODBC.ini 파일이 있어야 합니다.

- Windows 시스템에서는 이 파일이 레지스트리의 `HKEY_LOCAL_MACHINE/SOFTWARE/ODBC` 아래에 있습니다. 관계형 데이터 소스에 연결하는 방법에 대한 정보를 저장하려면 ODBC 데이터 소스 관리자를 사용하십시오.
- UNIX 시스템에서는 설치 프로그램이 모델 `odbc.ini` 파일을 작성합니다. 관계형 데이터 소스에 연결하는 방법에 대한 정보를 저장하려면 선호하는 편집기를 사용하여 이 파일을 편집하십시오.

ODBC.ini 파일은 ODBC 소프트웨어 패키지에 사용 가능하며, Microsoft Office 소프트웨어에도 포함되어 있습니다. ODBC 드라이버 또는 ODBC 관리자를 설치하는 응용프로그램에 대한 추가 정보는 다음 웹 사이트에서 볼 수 있습니다. http://support.microsoft.com/.

AIX 머신에서 Oracle을 사용하는 사용자가 ODBC를 Oracle용으로 구성하려면 MERANT 3.6 드라이버를 가리키도록 ODBC.ini 파일을 갱신해야 합니다.

버전 7.2에서는 OLAP Starter Kit가 관계형 데이터 소스 및 OLAP 메타데이터 카탈로그로의 ODBC 연결을 관리합니다. 이러한 ODBC 연결을 수용하기 위해 OLAP Starter Kit는 Windows NT 4.0, Windows 2000, AIX 및 Solaris 시스템에서 ODBC 드라이버를 사용합니다.

- DB2 Universal Database 버전 6 데이터베이스 클라이언트: Windows NT 4.0 SP5 또는 Windows 2000, AIX 4.3.3, Solaris 운영 체제 2.6, 7, 8(Sun OS 5.6, 5.7, 5.8)에서 DB2 버전 6 ODBC 드라이버 사용
- DB2 Universal Database 7.1 데이터베이스 클라이언트: Windows NT 4.0 SP5 또는 Windows 2000, AIX 4.3.3, Solaris 운영 체제 2.6, 7, 8(Sun OS 5.6, 5.7, 5.8)에서 DB2 버전 7 ODBC 드라이버 사용
- Oracle 8.04 및 8i SQL*Net 8.0 데이터베이스 클라이언트: Windows NT 4.0 SP5 또는 Windows 2000, AIX 4.3.3, Solaris 운영 체제 2.6, 7, 8(Sun OS 5.6, 5.7, 5.8)에서 MERANT 3.6 ODBC 드라이버 사용
- MS SQL Server 6.5.201(데이터베이스 클라이언트가 필요하지 않음): Windows NT 4.0 SP5 또는 Windows 2000에서 MS SQL Server 6.5 ODBC 드라이버 사용
- MS SQL Server 7.0(데이터베이스 클라이언트가 필요하지 않음): Windows NT 4.0 SP5 또는 Windows 2000에서 MS SQL Server7.0 ODBC 드라이버 사용

UNIX 시스템에서 데이터 소스 구성

AIX 및 Solaris 시스템에서는 ODBC용 환경 변수를 수동으로 설정하고 odbc.ini 파일을 편집하여 관계형 데이터 소스 및 OLAP 메타데이터 카탈로그를 구성해야

합니다. 새 드라이버 또는 데이터 소스를 추가하거나, 드라이버나 데이터 소스를 변경하는 경우에는 반드시 odbc.ini 파일을 편집해야 합니다.

AIX 또는 Solaris 시스템에서 DB2 OLAP Starter Kit를 사용하여 Merant ODBC 소스 및 DB2 데이터베이스에 액세스하는 경우, .odbc.ini 파일의 DB2 소스 섹션에서 ″Driver=″ 속성 값을 다음과 같이 변경하십시오.

AIX에서 드라이버 이름은 /usr/lpp/db2_07_01/lib/db2_36.o입니다.

AIX용 샘플 ODBC 소스 항목은 다음과 같습니다.

```
[SAMPLE] Driver=/usr/lpp/db2_07_01/lib/db2_36.o
Description=DB2 ODBC Database
Database=SAMPLE
```

Solaris 운영 환경: 드라이버 이름은 /opt/IBMdb2/V7.1/lib/libdb2_36.so입니다.

Solaris 운영 환경의 샘플 ODBC 소스 항목:

```
[SAMPLE] Driver=/opt/IBMdb2/V7.1/lib/libdb2_36.so
Description=DB2 ODBC Database
Database=SAMPLE
```

ODBC 환경 변수 구성

UNIX 시스템에서는 ODBC 핵심 구성요소를 액세스할 수 있도록 환경 변수를 설정해야 합니다. 필요한 변수를 설정하는 is.sh 및 is.csh 쉘 스크립트는 Starter Kit의 홈 디렉토리에 있습니다. ODBC를 통해 데이터 소스에 연결하기 전에 이 스크립트 중 하나를 실행해야 합니다. 또한 사용할 사용자 이름으로 OLAP Starter Kit를 수행하려면 로그인 스크립트에 이 스크립트들을 포함해야 합니다.

odbc.ini 파일 편집

odbc.ini 파일에서 데이터 소스를 구성하려면 ODBC 데이터 소스에 대한 설명과 이름을 추가하고, 데이터 소스 이름에 사용하기 위해 작성한 별도의 섹션에 ODBC 드라이버 경로, 파일 이름, 기타 드라이버 설정을 입력해야 합니다. 설치 프로그램은 샘플 odbc.ini 파일을 ISHOME 디렉토리에 설치합니다. 이 파일에는 지원되는 ODBC 드라이버에 대한 일반 ODBC 연결과 구성 정보가 있습니다. 이 파일을 시작점으로 하여 사용할 ODBC 드라이버를 관계형 데이터 소스 및 OLAP 메타데이터 카탈로그에 맵핑하십시오.

odbc.ini가 아닌 다른 파일을 사용할 경우에는 ODBCINI 환경 변수를 사용할 파일 이름으로 설정하십시오.

odbc.ini 파일에 데이터 소스 추가

1. OLAP Starter Kit 서버가 실행되는 시스템에서 **vi** 등의 편집기를 사용하여 odbc.ini 파일을 여십시오.
2. [ODBC Data Sources]로 시작되는 섹션을 찾은 다음 mydata=data source for analysis와 같이 데이터 소스 이름과 설명이 있는 새 줄을 추가하십시오. 혼동하지 않도록 하려면 데이터 소스의 이름을 RDBMS의 데이터베이스 이름과 동일하게 지정하십시오.
3. [mydata]와 같이 꺽쇠 괄호로 묶인 새로운 데이터 소스 이름이 있는 새 줄을 작성하여 파일에 새 섹션을 추가하십시오.
4. 데이터 소스 이름 다음의 줄에는 이 데이터 소스에 필요한 ODBC 드라이버의 전체 경로와 파일 이름 및 기타 필요한 ODBC 드라이버 정보를 추가하십시오. 다음 섹션에 나오는 예를 지침으로 삼아 RDBMS의 데이터 소스에 맵핑하십시오. Driver= 설정에 지정한 위치에 실제로 ODBC 드라이버 파일이 있는지 확인하십시오.
5. odbc.ini 편집을 마치면 파일을 저장하고 텍스트 편집기를 종료하십시오.

DB2용 ODBC 설정 예

다음 예는 AIX에 설치된 DB2 Universal Database 버전 6.1에서 IBM DB2 원시 ODBC 드라이버를 사용하여 db2data 관계형 데이터 소스에 연결하기 위해 odbc.ini 편집 방법을 보여줍니다. **vi** 편집기에서 **$ODBCINI** 명령을 사용하여 odbc.ini 편집 후 다음 명령문을 삽입하십시오.

```
[ODBC Data Sources]
db2data=DB2 Source Data on AIX
...
[db2data]
Driver=/home/db2inst1/sqllib/lib/db2.o
Description=DB2 Data Source - AIX, native
```

Oracle용 ODBC 설정 예

다음 예는 Oracle 버전 8(Solaris 운영 환경)에서 MERANT 버전 3.6 ODBC 드라이버를 사용하여 oradata 관계형 데이터 소스에 연결하기 위해 odbc.ini 파

일을 편집하는 방법을 보여줍니다. 이 예에서 LogonID와 Password는 OLAP Starter Kit의 사용자 이름과 암호에 사용된 실제 값으로 겹쳐씁니다.

```
[ODBC Data Sources]
oradata=Oracle8 Source Data on Solaris
...
[myoracle] Driver=
/export/home/users/dkendric/is200/odbclib/ARor815.so
Description=my oracle source
```

UNIX 시스템에서 OLAP 메타데이터 카탈로그 구성

AIX와 Solaris 시스템에서 OLAP 메타데이터 카탈로그를 구성하는 것은 데이터 소스를 구성하는 것과 비슷합니다. OLAP 메타데이터 카탈로그 데이터베이스의 경우, 289 페이지의『odbc.ini 파일 편집』에 설명된 대로 odbc.ini 파일에 데이터 소스 이름과 섹션을 추가하십시오. 이 외 다른 부분은 변경할 필요가 없습니다.

ODBC 데이터 소스로 구성하기 전에 먼저 지원되는 RDBMS에 OLAP 메타데이터 카탈로그 데이터베이스를 작성해야 합니다.

다음 예는 DB2 버전 6.1(Solaris 운영 환경)에서 원시 ODBC 드라이버를 사용하여 OLAP 메타데이터 카탈로그 TBC_MD에 연결하기 위해 odbc.ini 파일을 편집하는 방법을 보여줍니다.

```
[ODBC Data Sources]
ocd6a5a=db2 v6
...
[ocd6a5a]
Driver=/home/db2instl/sqllib/lib/db2.0
Description=db2
```

Windows 시스템에서 데이터 소스 구성

Windows NT나 Windows 2000 시스템에서 관계형 데이터 소스를 구성하려면 ODBC 관리자를 시작한 다음 OLAP 모델 및 metaoutline 작성에 사용할 데이터 소스에 대한 연결을 작성해야 합니다. Windows 제어판에서 ODBC 관리자 유틸리티를 실행하십시오. 다음 예는 DB2 데이터 소스를 작성합니다. 다른 RDBMS에서는 대화 상자가 다릅니다.

ODBC 관리자를 사용하여 관계형 데이터 소스를 구성하려면 다음 단계를 완료하십시오.

1. Windows 바탕 화면에서 제어판 창을 여십시오.
2. 제어판 창에서 다음 단계 중 하나를 수행하십시오.
   a. Windows NT에서는 **ODBC** 아이콘을 두 번 눌러 ODBC 데이터 소스 관리자 대화 상자를 여십시오.
   b. Windows 2000에서는 **관리 도구** 아이콘을 두 번 누른 다음 **데이터 소스 (ODBC)** 아이콘을 두 번 눌러 ODBC 데이터 소스 관리자 대화 상자를 여십시오.
3. ODBC 데이터 소스 관리자 대화 상자에서 **시스템 DSN** 탭을 누르십시오.
4. **추가**를 눌러 새 데이터 소스 작성 대화 상자를 여십시오.
5. ODBC 관리자의 새 데이터 소스 작성 대화 상자에 있는 드라이버 목록에서 `IBM DB2 ODBC 드라이버`와 같은 적절한 드라이버를 선택하고 **완료**를 눌러 ODBC IBMDB2 드라이버 - 추가 대화 상자를 여십시오.
6. ODBC IBM DB2 드라이버 - 추가 대화 상자의 **데이터베이스 별명** 드롭 다운 목록에서 관계형 데이터 소스(예: 샘플 응용프로그램의 TBC)에 대한 데이터베이스 이름을 선택하십시오.
7. **설명** 텍스트 상자에 드라이버 사용 방법을 나타내는 선택적 설명을 입력하고 **추가**를 누르십시오. 예를 들어, 다음 단어를 입력하여 My Business 데이터베이스를 설명할 수 있습니다.

   ```
   Customers, products, markets
   ```

   다음 단어를 입력하여 샘플 응용프로그램의 데이터베이스를 설명할 수 있습니다.

   ```
   Sample relational data source
   ```

   OLAP Starter Kit 데스크탑에서 연결할 때 이러한 설명을 보고 선택한 항목에 사용 가능한 데이터 소스를 식별할 수 있습니다.
8. 확인을 눌러 ODBC 데이터 소스 관리자 대화 상자로 돌아가십시오. 입력한 데이터 소스 이름과 이 데이터 소스에 맵핑한 드라이버가 **시스템 DSN** 탭의 **시스템 데이터 소스** 목록에 표시됩니다.

데이터 소스의 구성 정보를 편집하려면, 다음을 수행하십시오.

1. 데이터 소스 이름을 선택하고 **구성**을 눌러 ODBC IBM DB2 - 추가 대화 상자를 여십시오.
2. 변경할 정보가 있으면 알맞게 정정하십시오.
3. **확인**을 두 번 눌러 대화 상자를 닫으십시오.

Windows 시스템에서 OLAP 메타데이터 카탈로그 구성

Windows NT나 Windows 2000에서 OLAP 메타데이터 카탈로그를 구성하려면 ODBC 관리자를 시작한 다음 OLAP 메타데이터 카탈로그 데이터베이스가 포함된 데이터 소스에 대한 연결을 작성하십시오.

다음 예는 DB2 데이터 소스를 작성합니다. 다른 RDBMS에서는 대화 상자가 다릅니다. OLAP 메타데이터 카탈로그용 데이터 소스를 작성하려면 다음 단계를 완료하십시오.

1. 바탕 화면에서 제어판 창을 여십시오.
2. 제어판 창에서 다음 단계 중 하나를 수행하십시오.
   a. Windows NT에서는 **ODBC** 아이콘을 두 번 눌러 ODBC 데이터 소스 관리자 대화 상자를 여십시오.
   b. Windows 2000에서는 **관리 도구** 아이콘을 두 번 누른 다음 **데이터 소스 (ODBC)** 아이콘을 두 번 눌러 ODBC 데이터 소스 관리자 대화 상자를 여십시오.
3. ODBC 데이터 소스 관리자 대화 상자에서 **시스템 DSN** 탭을 누르십시오.
4. **추가**를 눌러 새 데이터 소스 작성 대화 상자를 여십시오.
5. ODBC 관리자의 새 데이터 소스 작성 대화 상자에 있는 드라이버 목록에서 `IBM DB2 ODBC 드라이버`와 같은 적절한 드라이버를 선택하고 **완료**를 눌러 ODBC IBMDB2 드라이버 - 추가 대화 상자를 여십시오.
6. ODBC IBM DB2 드라이버 - 추가 대화 상자의 **데이터베이스 별명** 드롭 다운 목록에서 OLAP 메타데이터 카탈로그(예: 샘플 응용프로그램의 TBC_MD)에 대한 데이터베이스 이름을 선택하십시오. 선택한 데이터베이스의 이름이 **데이터 소스 이름** 텍스트 상자에 자동으로 표시됩니다.
7. 데이터 소스의 이름을 변경하려면 **데이터 소스 이름** 텍스트 상자에 표시된 이름을 선택한 다음 드라이버 사용 방법을 나타내는 새 이름을 입력하고 **추가**를

누르십시오. 예를 들어, 다음 이름을 입력하여 첫 번째 OLAP 메타데이터 카탈로그에 연결할 때 이 드라이버를 사용한다는 것을 나타낼 수 있습니다.

```
OLAP Catalog first
```

다음 이름을 입력하여 샘플 응용프로그램의 OLAP 메타데이터 카탈로그 데이터베이스에 연결한다는 것을 나타낼 수 있습니다.

```
 TBC_MD
```

8. **설명** 텍스트 상자에 드라이버 사용 방법을 나타내는 설명을 입력하십시오. 예를 들어, 다음 단어를 입력하여 OLAP 메타데이터 카탈로그를 설명할 수 있습니다.

   ```
   My first models and metaoutlines
   ```

   다음 단어를 입력하여 샘플 응용프로그램의 OLAP 메타데이터 카탈로그 데이터베이스를 설명할 수 있습니다.

   ```
   Sample models and metaoutlines
   ```

   이 설명을 보고 OLAP Starter Kit 데스크탑에서 OLAP 메타데이터 카탈로그에 연결할 때 선택할 카탈로그를 쉽게 식별할 수 있습니다.
9. 확인을 눌러 ODBC 데이터 소스 관리자 대화 상자로 돌아가십시오. 입력한 데이터 소스 이름과 이 데이터 소스에 맵핑한 드라이버가 **시스템 DSN** 탭의 **시스템 데이터 소스** 목록에 표시됩니다.

데이터 소스의 구성 정보를 편집하려면

1. 데이터 소스 이름을 선택하고 **구성**을 눌러 ODBC IBM DB2 - 추가 대화 상자를 여십시오.
2. 변경할 정보가 있으면 알맞게 정정하십시오.
3. **확인**을 두 번 눌러 대화 상자를 닫으십시오.

데이터 소스를 구성한 후의 작업

관계형 데이터 소스와 OLAP 메타데이터 카탈로그를 구성했으면 OLAP Starter Kit에서 관계형 데이터 소스와 OLAP 메타데이터 카탈로그에 연결할 수 있습니다. 연결한 다음 OLAP 모델과 metaoutline을 작성, 수정, 저장할 수 있습니다.

SQL Server 데이터베이스를 호출하는 동안 SQL Server ODBC 드라이버가 시간종료될 수 있습니다. 데이터베이스가 사용 중이지 않으면 다시 시도하십시오. 드라이버 시간종료 시간을 늘려 이 문제점을 방지할 수 있습니다. 자세한 정보는 사용 중인 드라이버의 ODBC 문서를 참조하십시오.

ODBC 연결 문제점 및 솔루션에 대한 자세한 정보는 *OLAP Integration Server 시스템 관리자 안내서*를 참조하십시오.

OLAP Starter Kit 데스크탑에서 로그인

OLAP Starter Kit 데스크탑을 사용하여 OLAP 모델과 metaoutline을 작성하려면, 클라이언트 소프트웨어를 DB2 OLAP Integration Server와 DB2 OLAP Server에 연결해야 합니다. 로그인 대화 상자는 데스크탑이 이들 두 서버에 연결하는 데 필요한 정보를 묻는 프롬프트를 표시합니다. 대화 상자의 왼쪽에 DB2 OLAP Integration Server에 대한 정보를 입력하십시오. 오른쪽에는 DB2 OLAP Server에 대한 정보를 입력하십시오.

DB2 OLAP Integration Server에 연결하려면, 다음과 같이 수행하십시오.

- 서버: 통합 서버의 호스트 이름이나 IP 주소를 입력하십시오. 데스크탑과 동일한 워크스테이션에 통합 서버를 설치한 경우, 일반 값은 ″localhost″ 또는 ″127.0.0.1″입니다.
- OLAP 메타데이터 카탈로그: OLAP Integration Server에 연결할 때에는 메타데이터 카탈로그도 지정해야 합니다. OLAP Integration Server는 작성한 OLAP 모델 및 metaoutline에 대한 정보를 메타데이터 카탈로그로 알려진 관계형 데이터베이스에 저장합니다. 이 관계형 데이터베이스는 ODBC에 대해 등록해야 합니다. 카탈로그 데이터베이스에는 OLAP Integration Server가 인식하는 특정 관계형 테이블 세트가 포함됩니다. 로그인 대화 상자에서 통합 서버를 지정한 다음 OLAP 메타데이터 카탈로그 필드에 대한 풀다운 메뉴를 펼쳐서 OLAP Integration Server에 알려진 ODBC 데이터 소스의 목록을 볼 수 있습니다. 메타데이터 카탈로그 테이블이 포함된 ODBC 데이터베이스를 선택하십시오.
- 사용자 이름 및 암호: OLAP Integration Server는 이 패널에서 지정한 사용자 이름과 암호를 사용하여 메타데이터 카탈로그에 연결합니다. 이것은 서버에 있는 로그인 어카운트입니다. (서버와 클라이언트가 동일한 머신에서 수행되지

않으면 클라이언트에는 이 어카운트가 없습니다.) 사용자 이름은 OLAP 메타 데이터 카탈로그를 작성한 사용자이어야 합니다. 그렇지 않은 경우, OLAP Integration Server는 카탈로그 데이터베이스에서 관계 테이블을 찾지 않습니다. 테이블 스키마 이름들이 다르기 때문입니다.

DB2 OLAP Server 정보는 선택적이므로 로그인 대화 상자의 오른쪽에 있는 입력 필드는 공백으로 둘 수 있습니다. 그러나, 데스크탑 및 Administration Manager의 일부 조작에서는 DB2 OLAP Server에 연결할 것을 요구합니다. 이들 필드를 공백으로 두면, 데스크탑은 요청한 조작을 완료하기 위해 통합 서버가 DB2 OLAP Server에 연결해야 하는 경우에 로그인 대화 상자를 다시 표시합니다. 로그인 대화 상자에서 DB2 OLAP Server 필드를 항상 채우는 것이 좋습니다.

DB2 OLAP Server에 연결하려면, 다음과 같이 수행하십시오.

- 서버: DB2 OLAP Server의 호스트 이름이나 IP 주소를 입력하십시오. OLAP Starter Kit를 수행하고 있는 경우, OLAP Server와 통합 서버는 동일합니다. 통합 서버와 OLAP Server가 다른 호스트에 설치된 경우, OLAP Integration Server에 정의된 호스트 이름이나 IP 주소를 입력하십시오.
- 사용자 이름 및 암호: OLAP Integration Server는 이 패널에서 지정한 사용자 이름과 암호를 사용하여 DB2 OLAP Server에 연결합니다. 이 사용자 이름과 암호는 DB2 OLAP Server에 이미 정의되어 있어야 합니다. OLAP Server는 호스트 운영 체제에서 자체의 사용자 이름과 암호를 별도로 관리합니다.

Starter Kit 로그인 예

다음의 예는 OLAP 샘플을 작성하고 관리자 사용자 ID로 *db2admin*을 선택하고 OLAP Starter Kit 설치 중에 관리자 암호로 *password*를 선택한 것으로 가정합니다.

- OLAP Integration Server의 경우: 서버는 *localhost*, OLAP 메타데이터 카탈로그는 *TBC_MD*, 사용자 이름은 *db2admin*, 암호는 *password*입니다.
- DB2 OLAP Server의 경우: 서버는 *localhost*, 사용자 이름은 *db2admin*입니다.

OLAP Starter Kit에 대한 샘플 데이터베이스 수동 작성 및 구성

샘플 데이터베이스는 OLAP Starter Kit를 설치하면 자동으로 작성됩니다. 다음의 지시사항은 필요한 경우에 카탈로그 및 샘플 데이터베이스를 수동으로 설정하는 방법을 설명합니다.

1. Windows에서는 **시작 --► 프로그램 --► Windows NT용 DB2 --► 명령 창**을 클릭하여 명령 센터를 여십시오.
2. 카탈로그 데이터베이스를 작성하십시오.
   a. `db2 create db OLAP_CAT`를 입력하십시오.
   b. `db2 connect to OLAP_CAT`를 입력하십시오.
3. 데이터베이스에서 테이블을 작성하십시오.
   a. \SQLLIB\IS\ocscript\ocdb2.sql을 찾으십시오.
   b. `db2 -tf ocdb2.sql`을 입력하십시오.
4. 샘플 소스 데이터베이스를 작성하십시오.
   a. `db2 connect reset`를 입력하십시오.
   b. `db2 create db TBC`를 입력하십시오.
   c. `db2 connect to TBC`를 입력하십시오.
5. 데이터베이스에서 테이블을 작성하십시오.
   a. \SQLLIB\IS\samples\를 찾으십시오.
   b. tbcdb2.sql을 \SQLLIB\samples\db2sampl\tbc에 복사하십시오.
   c. lddb2.sql을 \SQLLIB\samples\db2sampl\tbc에 복사하십시오.
   d. \SQLLIB\samples\db2sampl\tbc를 찾으십시오.
   e. `db2 -tf tbcdb2.sql`을 입력하십시오.
   f. `db2 - vf lddb2.sql`을 입력하여 테이블에 샘플 소스 데이터를 로드하십시오.
6. 샘플 카탈로그 데이터베이스를 작성하십시오.
   a. `db2 connect reset`을 입력하십시오.
   b. `db2 create db TBC_MD`를 입력하십시오.
   c. `db2 connect to TBC_MD`를 입력하십시오.

7. 데이터베이스에서 테이블을 작성하십시오.
   a. \SQLLIB\IS\samples\tbc_md를 찾으십시오.
   b. ocdb2.sql을 \SQLLIB\samples\db2sampl\tbcmd에 복사하십시오.
   c. lcdb2.sql을 \SQLLIB\samples\db2sampl\tbcmd에 복사하십시오.
   d. \SQLLIB\samples\db2sampl\tbcmd를 찾으십시오.
   e. `db2 -tf ocdb2.sql`을 입력하십시오.
   f. `db2 -vf lcdb2.sql`을 입력하여 테이블에 샘플 메타데이터를 로드하십시오.
8. TBC_MD, TBC, AND OLAP_CAT용 ODBC를 구성하십시오.
   a. **시작 --▸ 설정 --▸ 제어판**을 눌러 NT 제어판을 여십시오.
   b. 목록에서 ODBC(또는 ODBC 데이터 소스)를 선택하십시오.
   c. 시스템 DSM 탭을 선택하십시오.
   d. **추가**를 누르십시오. **새 데이터 소스 작성** 창이 열립니다.
   e. 목록에서 IBM DB2 ODBC 드라이버를 선택하십시오.
   f. **완료**를 누르십시오. ODBC IBM DB2 드라이버 - 추가 창이 열립니다.
   g. **데이터 소스 이름** 필드에 데이터 소스의 이름(OLAP_CAT)을 입력하십시오.
   h. **데이터베이스 별명** 필드에 별명 이름을 입력하거나 아래 화살표를 누르고 목록에서 OLAP_CAT를 선택하십시오.
   i. 확인을 누르십시오.
   j. TBC_MD 및 TBC 데이터베이스에 대해 이들 단계를 반복하십시오.

| 응용프로그램을 OLAP Starter Kit 버전 7.2로 이주

| 설치 프로그램이 OLAP Starter Kit 샘플 응용프로그램, 데이터베이스, 데이터 파일을 다시 설치하지 않습니다. 기존의 응용프로그램 및 데이터베이스는 전혀 영향을 받지 않습니다. 그러나 설치하기 전에 항상 응용프로그램과 데이터베이스를 백업하는 것이 좋습니다.

| 응용프로그램을 열면 자동으로 버전 7.2로 이주됩니다.

알려진 문제점 및 제한사항

이 섹션에서는 DB2 OLAP Starter Kit에 대해 알려진 한계를 나열합니다.

Windows 플랫폼용 Merant 드라이버와의 Informix RDBMS 호환성

Windows 플랫폼용 Merant 드라이버가 Informix RDBMS와 작동하려면 다음 두 항목을 PATH문에 추가해야 합니다.

- C:\Informix
- C:\Informix\bin

두 항목 모두 PATH 시작 부분에 있어야 합니다.

OLAP 모델과 연관된 metaoutline 간에 발생할 수 있는 차원 불일치

몇몇 경우에는 OLAP 모델에 일치하는 차원이 없는 차원을 metaoutline에 작성할 수 있습니다. 다음과 같은 시나리오에서 가능합니다.

1. 새 OLAP 모델을 작성하고 저장하십시오.
2. 이 모델에 바탕을 두고 metaoutline을 작성한 후 metaoutline을 저장하지 마십시오.
3. OLAP 모델로 돌아가서 metaoutline 차원 중 하나를 작성할 때 기준으로 사용한 차원을 삭제하십시오.
4. metaoutline으로 돌아가서 저장하고 닫은 다음 다시 여십시오. 그러면 이 metaoutline에 OLAP 모델에 일치하는 차원이 없는 차원이 포함됩니다.

OLAP Starter Kit는 이와 같이 작성된 일치하지 않는 차원과 metaoutline에 있는 사용자 정의 차원을 구별하지 못합니다. 따라서 metaoutline에 일치하지 않는 차원이 표시되더라도 OLAP 모델에 일치하는 차원이 없기 때문에 metaoutline은 이를 사용자 정의 차원으로 간주합니다.

Windows 2000 플랫폼에서 구성원 및 데이터를 로드할 수 없게 하는 TMP 환경 변수 설정

Windows 2000과 Windows NT에서는 TMP에 대한 기본 시스템 환경 변수 설정과 사용자 환경 변수 설정이 다르기 때문에, Windows 2000에서 OLAP Starter Kit를 실행하면 구성원 및 데이터를 로드할 수 없습니다. 이 경우 temp 파일을 작성할 수 없다는 오류 메시지가 나타납니다.

다음 단계를 따라서 Windows 2000에서 이러한 제한사항으로 인한 문제점을 일시적으로 해결할 수 있습니다.

1. C:\TEMP라는 디렉토리를 작성하십시오.
2. 시스템과 사용자의 TMP 환경 변수를 모두 TMP=C:\TEMP로 설정하십시오.

ODBC를 설치했는데 기존 Merant 드라이버가 바뀌지 않을 경우

이렇게 설치했을 때는 기존 3.6 Merant ODBC 드라이버가 갱신되지 않습니다. OLAP Starter Kit 버전 7.1, FixPak 2 이전 버전에서 업그레이드할 경우에는 이전에 설치한 ODBC 드라이버를 사용해야 합니다.

UNIX 플랫폼에서 Merant Informix ODBC 드라이버 사용

UNIX 플랫폼에서 Merant Informix ODBC 드라이버를 사용하려면 다음 중 하나를 수행하십시오.

- Starter Kit를 시작하기 전에 LANG 환경 변수를 ″en_US″로 설정하십시오. 예를 들어, Korn 쉘에 다음과 같이 입력하십시오.

  ```
  export LANG='en_US'
  ```

  OLAP Starter Kit를 시작할 때마다 이 변수를 설정하십시오.
- LANG 환경 변수가 다른 값으로 설정되어 있지 않은 경우에는 설치한 후 다음 기호 링크를 만드십시오.

  ```
  ln -s $ISHOME/locale/en_US $ISHOME/locale/$LANG
  ```

OLAP 클라이언트 및 서버의 서비스 레벨 혼용

IBM은 DB2 OLAP Starter Kit의 클라이언트 및 서버 구성요소를 동일한 버전 및 FixPak 레벨로 유지할 것을 권장합니다. 그러나 몇몇 경우에는 클라이언트 및 서버 구성요소를 서로 다른 수준으로 섞어서 사용할 수 있습니다.

같은 버전의 클라이언트와 서버를 다른 서비스 레벨에서 사용

IBM은 이전 버전의 서버에 새 버전의 클라이언트를 사용하는 것을 지원하거나 권장하지 않습니다. 그러나 IBM이 지원하지는 않지만 새 버전의 서버에 이전 버전의 클라이언트를 사용할 수는 있습니다. 이 경우 몇 가지 문제점이 발생할 수 있습니다. 예를 들어, 다음과 같습니다.

- 서버의 메시지가 정확하지 않습니다. 서버의 레벨과 일치하도록 클라이언트의 message.MDB 파일을 업그레이드하여 이 문제점을 일시적으로 해결할 수 있습니다.
- 서버의 새로운 기능이 작동하지 않습니다. 새로운 기능을 사용하려고 하면 클라이언트나 서버 또는 둘다 실패합니다.
- 클라이언트와 서버가 제대로 연결되지 않습니다.

한 버전에서 여러 서버에 단일 클라이언트 사용

클라이언트 하나를 다른 머신이나 운영 체제에 있는 여러 개의 OLAP 서버에 연결해야 할 경우, IBM은 모두 동일한 버전과 서비스 레벨을 사용할 것을 권장합니다. 클라이언트 레벨은 적어도 가장 낮은 서버 레벨과 동일해야 합니다. 문제점이 발생하면 다른 클라이언트 머신을 사용하여 해당 호스트와 일치하도록 하거나, 모든 클라이언트와 서버를 동일한 레벨로 업그레이드해야 합니다.

여러 버전의 클라이언트와 서버 혼용

IBM은 OLAP Starter Kit 버전 7.1의 클라이언트 및 서버를 버전 7.2의 클라이언트 및 서버와 사용하는 것을 지원하지 않습니다. IBM OLAP 제품을 새로운 버전 레벨로 업그레이드하면, 대개 클라이언트와 서버를 동일한 버전 레벨로 설정하기 위해 네트워크가 갱신되고 데이터 형식이 변경됩니다.

IBM 제품(DB2 OLAP Starter Kit)을 Hyperion 제품(Hyperion Essbase 및 Hyperion Integration Server)과 혼용

IBM은 IBM의 OLAP 클라이언트 및 서버를 Hyperion Solutions의 클라이언트 및 서버와 혼용하는 것을 지원하지 않습니다. 이 둘 사이에는 기능 상의 몇 가지 차이가 있기 때문에, 이들 혼용 요소들이 같은 상황에서 작동하더라도 문제점이 발생할 수 있습니다.

OLAP Spreadsheet Add-in EQD 파일 누락

DB2 OLAP Starter Kit에서 Spreadsheet Add-in에는 Query Designer(EQD)라고 하는 구성요소가 있습니다. EQD에 대한 온라인 도움말 중 **자습서**라는 버튼을 누르면 아무런 내용도 표시되지 않습니다. EQD 자습서에 표시되어야 하는 내용은 *Excel용 OLAP Spreadsheet Add-in 사용자 안내서*와 *1-2-3용 OLAP*

*Spreadsheet Add-in 사용자 안내서*의 2장 부속 집합입니다. EQD 자습서의 모든 정보는 정보 센터에서 이 서적들의 HTML 버전 및 PDF 버전으로 사용할 수 있습니다.

Information Catalog Manager 관리 안내서

Information Catalog Manager 초기화 유틸리티

Information Catalog Manager(ICM) 유틸리티로, 이제 다음 명령을 사용하여 SQL 문을 CREATE TABLE문의 끝에 추가할 수 있습니다.

```
CREATEIC \DBTYPE dbtype \DGNAME dgname \USERID userid \PASSWORD password
\KA1 userid \TABOPT "directory:\tabopt.file"
```

DB2가 설치된 디렉토리에서 CREATEIC 유틸리티의 TABOPT 키워드를 지정할 수 있습니다. TABOPT 키워드 뒤의 값은 전체 경로가 있는 tabopt.file 파일 이름입니다. 디렉토리 이름에 공백이 포함되어 있으면, 이름을 인용부호로 묶으십시오. *tabopt.file* 파일의 내용에는 CREATE TABLE문에 추가할 정보가 들어 있어야 합니다. 아래의 SQL문 중 하나를 사용하여 이 *tabopt.file* 파일에 쓸 수 있습니다. ICM 유틸리티는 이 파일을 읽은 후 그것을 CREATE TABLE문에 추가합니다.

표 9. SQL문

IN MYTABLESPACE	MYTABLESPACE의 데이터로 테이블을 작성합니다.
DATA CAPTURE CHANGES	테이블을 작성하고 SQL 변경사항을 확장된 형식으로 로그합니다.
IN ACCOUNTING INDEX IN ACCOUNT_IDX	ACCOUNTING의 데이터와 ACCOUNT_IDX의 인덱스로 테이블을 작성합니다.

내용 파일의 최대 크기는 1000개의 1바이트 문자입니다.

이 새로운 기능은 Windows와 UNIX 시스템에서만 사용 가능합니다.

사용권 문제

다음 메시지가 표시될 경우,

```
FLG0083E: IBM Information Catalog Manager 초기화 유틸리티에
대해 유효한 사용권을 가지고 있지 않습니다.
해당 지역의 소프트웨어 사업부나 IBM
영업 대표에게 문의하십시오.
```


DB2 Warehouse Manager나 IBM DB2 OLAP Server를 구입하고 Information Catalog Manager 구성요소를 설치해야 합니다. 이 구성요소에 Information Catalog Manager 초기화 유틸리티가 있습니다.

설치 문제

DB2 Warehouse Manager나 IBM DB2 OLAP Server를 설치한 후 다른 Information Catalog Manager 관리자 구성요소를 같은 워크스테이션에 설치한 경우(DB2 Universal Database CD-ROM을 사용하여), Information Catalog 초기화 유틸리티 위에 겹쳐썼을 수도 있습니다. 그러한 경우, \sqllib\bin 디렉토리에서 `createic.bak` 및 `flgnmwcr.bak` 파일을 찾고 이름을 각각 `createic.exe` 및 `flgnmwcr.exe`로 변경하십시오.

DB2 Universal Database에서 추가 Information Catalog Manager 구성요소를 설치할 경우, 그 구성요소들은 Data Warehouse Manager를 설치한 곳과 다른 별도의 워크스테이션에 있어야 합니다. 자세한 내용은 DB2 Warehouse Manager 설치 안내서에서 제3장 Information Catalog Manager 구성요소 설치를 참조하십시오.

Information Catalog Manager(ICM) 향상

Information Catalog Manager에는 다음의 성능 향상이 포함됩니다.

현재 ICM은 소스, 목표 데이터베이스, 테이블 또는 컬럼의 ETI 필터 정보에 대해 가져오기 기능을 지원합니다. 웨어하우스로 레지스터를 가져올 때 새 ICM 오브젝트 유형인 ETI 변환 데이터가 필터 정보를 저장하는 데 사용됩니다. 그리고 나서, 이 오브젝트는 정의된 테이블이나 컬럼 또는 소스나 목표 데이터베이스에 링크됩니다.

ICM에는 동일 ICM 범주에 다른 ETI 변환 데이터 오브젝트를 등록하는 결과로 다중 ETI 변환 데이터 오브젝트를 이용해 특정 테이블이나 컬럼 또는 소스나 목표 데이터베이스에 링크를 거는 기능이 있습니다.

특정 목표 컬럼이 현재 동일 ICM 범주에 다른 ETI 변환 데이터 오브젝트를 등록하는 결과로 다중 변환을 포함하고 있기 때문에 동일한 기능이 변환에 적용됩니다. 이를 위해 ICM은 ETI*Extract 맵핑을 가져올 때 변환 키를 변경합니다.

이 기능을 사용하려면 MetaScheduler 4.1.0과 함께 ETI*Extract 4.2.1을 사용해서 데이터 웨어하우스 관리 프로그램을 등록하십시오. 이 기능 사용에 대한 자세한 정보는 http://www.ibm.com/software/data/db2/datawarehouse/support.html의 DB2 Warehouse Manager, Hints and Tips 섹션을 참조하십시오. 키워드 ″ETI″ 또는 ″Application Data″로 검색하십시오.

Windows 환경에서의 Information Catalog Manager와 Sybase 사이의 비호환

Sybase Open Client가 탑재된 Windows NT 또는 Windows 2000에 ICM(Information Catalog Manager) 버전 7을 설치하면 오류가 발생하고 Sybase Utilities는 작동을 중지합니다. 아래와 유사한 오류 메시지가 발생합니다.

```
LIBTCL.DLL을 초기설정할 수 없습니다. SYBASE 환경 변수가
올바르게 설정되었는지 확인하십시오.
```

LC_ALL 환경 매개변수를 Windows 환경 매개변수에서 제거하여 이런 오류가 발생하지 않도록 하십시오. LC_ALL은 로케일 범주 매개변수입니다. 로케일 범주는 지역화 루틴이 프로그램이 로케일 정보의 어떤 부분을 사용할지를 지정하기 위해 사용하는 정확한 상수입니다. *로케일*은 프로그램의 특정 부분에 대해 사용자 정의해야 하는 지역(국가/지역)을 의미합니다. 로케일 종속 영역에는 예를 들어 날짜 형식 또는 화폐 단위 표시형식 등이 포함됩니다. LC_ALL은 모든 로케일 특정 작동(모든 범주)에 영향을 줍니다.

ICM이 Windows NT 플랫폼에서 Sybase와 공존하도록 LC_ALL 환경 매개변수를 제거하면, 다음 기능이 더 이상 작동하지 않습니다.

- Information Catalog User
- Information Catalog Administrator
- Information Catalog Manager

LC_ALL 매개변수의 제거시 ICM 이외에는 영향을 미치지 않습니다.

DB2 버전 7 Information Catalog Manager를 사용하여 DB2 버전 5 정보 카탈로그에 액세스

DB2 버전 7 Information Catalog Manager 하위 구성요소는 DB2 버전 7 설치 프로세스에 의해 구성된대로 DB2 버전 6과 DB2 버전 7 데이터베이스에 저장된 정보 카탈로그에 대한 액세스를 지원합니다. DB2 버전 5 데이터베이스에 저장된 정보 카탈로그에 액세스하기 위해 하위 구성요소의 구성을 수정할 수 있습니다. DB2 버전 7 Information Catalog Manager 하위 구성요소는 DB2 버전 2 또는 다른 이전 버전에서의 데이터에 대한 액세스를 지원하지 않습니다.

DB2 버전 5 데이터베이스에 저장된 정보 카탈로그에 액세스하도록 정보 카탈로그 관리자, 정보 카탈로그 사용자 및 정보 카탈로그 초기화 유틸리티를 설정하려면, 다음과 같이 수행하십시오.

1. DB2 버전 7 Information Catalog Manager가 설치되지 않은 워크스테이션에 DB2 Connect Enterprise Edition 버전 6을 설치하십시오.

   DB2 Connect Enterprise Edition은 DB2 Universal Database Enterprise Edition 및 DB2 Universal Database Enterprise - Extended Edition의 일부로 포함됩니다. 이들 DB2 제품 중 하나의 버전 6이 설치된 경우, DB2 Connect를 별도로 설치할 필요가 없습니다.

   제한사항: 동일한 Windows NT 또는 OS/2 워크스테이션에 여러 DB2 버전을 설치할 수 없습니다. 다른 Windows NT 워크스테이션이나 OS/2 또는 UNIX 워크스테이션에 DB2 Connect를 설치할 수 있습니다.

2. DB2 버전 5 데이터에 액세스할 수 있도록 Information Catalog Manager와 DB2 Connect 버전 6을 구성하십시오. 자세한 정보는 *DB2 Connect 사용자 안내서*를 참조하십시오. 다음의 단계는 필수적인 단계의 개요입니다.

   a. DB2 버전 5 시스템에서 DB2 명령행 처리기를 사용하여 Information Catalog Manager가 액세스할 버전 5 데이터베이스를 카탈로그하십시오.

   b. DB2 Connect 시스템에서 DB2 명령행 처리기를 사용하여 다음을 카탈로그하십시오.

      - DB2 버전 5 시스템에 대한 TCP/IP 노드
      - DB2 버전 5 시스템에 대한 데이터베이스

- DB2 버전 5 시스템에 대한 DCS 항목

c. Information Catalog Manager에서 DB2 명령행 처리기를 사용하여 다음을 카탈로그하십시오.
   - DB2 Connect 시스템에 대한 TCP/IP 노드
   - DB2 Connect 시스템에 대한 데이터베이스

카탈로그 데이터베이스에 대한 정보는 *DB2 Universal Database 설치 및 구성 보충 설명서*를 참조하십시오.

3. Information Catalog Manager가 있는 웨어하우스에서 DB2 Connect를 통해 액세스할 각 데이터베이스에 DB2 CLI 패키지를 바인드하십시오.

다음의 DB2 명령은 가정한 DB2 버전 5 데이터베이스인 v5database로의 바인딩 예를 제공합니다. DB2 명령행 처리기를 사용하여 다음의 명령을 발행하십시오. db2cli.lst와 db2ajgrt는 \sqllib\bnd 디렉토리에 위치합니다.

```
db2 connect to v5database user userid using password
db2 bind db2ajgrt.bnd
db2 bind @db2cli.lst blocking all grant public
```

여기서, *userid*는 v5database의 사용자 ID이며 *password*는 사용자 ID의 암호입니다.

db2cli.list가 DB2 버전 5 데이터베이스에 바인드되면 오류가 발생합니다. 이 오류는 이 구성에서 대형 오브젝트(LOB)가 지원되지 않기 때문에 발생합니다. 이 오류는 DB2 버전 5 데이터베이스에 대한 웨어하우스 에이전트의 액세스에 영향을 주지 않습니다.

2000년 6월에 사용할 수 있는 DB2 Universal Database 버전 5의 FixPak 14는 DB2 Connect를 통해 DB2 버전 5 데이터에 액세스하는 데 필수적입니다. 해당 FixPak에서 APAR 번호 JR14507을 참조하십시오.

정보 카탈로그 설정

제 1 장 "정보 카탈로그 설정"의 첫 번째 섹션에 있는 2단계는 다음과 같습니다.

```
DB2 Warehouse Manager 또는 DB2 OLAP Server를
설치할 경우, 디폴트 정보 카탈로그는
Windows NT용 DB2 Universal Database에 작성됩니다.
```

이것은 잘못된 내용입니다. 새로운 정보 카탈로그를 정의해야 합니다. ″Information Catalog 작성″ 섹션에서 자세한 내용을 참조하십시오.

기타 제품과 메타데이터 교환

제 6 장 ″기타 제품과 메타데이터 교환″의 ″발표할 OLAP 오브젝트 식별″ 섹션의 두 번째 구문입니다.

```
DB2 OLAP Integration Server 메타데이터를 발행할 경우,
정보 카탈로그 "다차원 데이터베이스 내에서의 차원"
오브젝트 유형과 OLAP Integration Server에서 테이블 오브젝트
사이에 링크된 관계가 작성됩니다.
```

이 내용은 다음과 같이 바뀌어야 합니다.

```
DB2 OLAP Integration Server 메타데이터를 발행할 경우,
정보 카탈로그 "다차원 데이터베이스 내에서의 차원"
테이블 오브젝트" 사이에 링크된 관계가 작성됩니다.
```

이 내용은 ″Information Catalog Manager와 OLAP Server 사이의 메타데이터 맵핑″ 섹션에 있는 부록 C ″메타데이터 맵핑″에도 나타납니다.

flgnxoln 명령을 사용하여 메타데이터 교환

제 6 장 ″메타테이터 교환″에서 ″발행을 위한 OLAP 오브젝트 식별″ 섹션이 있습니다. 이 섹션의 끝에서는 flgnxoln 명령을 사용하여 정보 카탈로그로 OLAP 서버 메타데이터를 발행하는 예가 있습니다. 이 예에서 db2olap.ctl 및 db2olap.ff 파일의 디렉토리는 x:\Program Files\sqllib\logging으로 나타나는데, 이는 잘못된 정보입니다. 디렉토리 이름은 87 페이지에 설명된 대로 x:\Program Files\sqllib\exchange이어야 합니다.

MDISDGC 명령을 사용하여 메타데이터 교환

제 6 장 기타 제품과 메타데이터 교환에서 97 페이지의 ″태그 언어 파일로 MDIS-확정 메타데이터 전환″ 섹션이 있습니다. MS-DOS 명령 프롬프트에서는 MDISDGC 명령을 발행할 수 없습니다. DB2 명령 창에서 MDISDGC 명령을 발행해야 합니다. ″태그 언어 파일로 MDIS-확정 메타데이터 전환″ 섹션의 첫 번째

문장은 또한 MS-DOS 명령 프롬프트에서 DGMDISC 명령을 발행해야 한다고 합니다. DB2 명령 창에서 DGMDISC 명령을 발행해야 합니다.

프로그램 호출

정보 카탈로그 관리 안내서에 있는 예제에서는 디렉토리 이름 프로그램 파일이 들어있는 명령을 보여줍니다. 경로 이름의 일부로서 프로그램 파일이 들어있는 프로그램을 호출할 경우, 큰 따옴표로 프로그램 호출을 묶어야 합니다. 예를 들어, 부록 B ″사전 정의된 Information Catalog Manager 오브젝트 유형″에는 제목이 ″사전 정의된 오브젝트 유형으로 정보 카탈로그 초기화″인 섹션에 예가 있습니다. 이 섹션에 있는 예를 사용하는 경우, DOS 프롬프트에서 실행하면 오류를 발생하게 됩니다. 올바른 예는 다음과 같습니다.

```
"X:Program Files\SQLLIB\SAMPLES\SAMPDATA\DGWDEMO"
/T userid password dgname
```

Information Catalog Manager 프로그래밍 안내서 및 참조서

Information Catalog Manager 이유 코드

부록 D: Information Catalog Manager 이유 코드에서 다음의 이유 코드에 대한 가장 오른쪽 컬럼에서 일부 텍스트가 잘릴 수 있습니다. 31014, 32727, 32728, 32729, 32730, 32735, 32736, 32737, 33000, 37507, 37511 및 39206. 텍스트가 잘린 경우, HTML 버전 책을 보고 완전한 컬럼인지 확인하십시오.

Information Catalog Manager 사용자 안내서

제 2 장에는 "서버 노드 및 리모트 정보 카탈로그 등록" 섹션이 있습니다. 이 섹션에서는 Information Catalog Manager를 사용하여 리모트 정보 카탈로그를 등록하기 전에 DB2 제어 센터에서 완료할 수 있는 단계를 나열합니다. 이 절의 마지막 문단에서는 DB2 제어 센터에서 단계 세트를 완료한 뒤(시스템 추가, 인스턴스 추가 및 데이터베이스 추가), Information Catalog Manager를 열기 전에 제어 센터를 종료해야 합니다. 이 내용은 잘못된 것입니다. Information Catalog Manager를 열기 전에 제어 센터를 종료할 필요가 없습니다.

같은 수정 내용이 온라인 도움말 태스크 "서버 노드 및 리모트 정보 카탈로그 등록" 및 레지스터 서버 노드 및 정보 카탈로그 창의 온라인 도움말에도 적용됩니다.

Information Catalog Manager: 온라인 메시지

FLG 메시지 정정

메시지 FLG0260E

메시지 설명의 두 번째 문장은 다음과 같아야 합니다.

```
    오류로 인해 정보 카탈로그가 롤백을 시도했지만, 실패했습니다.
정보 카탈로그가 안정 상태에 있지 않으므로, 변경할 수 없습니다.
```

메시지 FLG0051E

메시지 설명에서 두 번째 점(bullet) 항목은 다음과 같아야 합니다.

```
    정보 카탈로그에 너무 많은 오브젝트 또는 오브젝트 유형이 있습니다.
```

관리자 조치는 다음과 같아야 합니다.

```
    가져오기 함수를 사용하는 현재의 정보 카탈로그에서
일부 오브젝트 및 오브젝트 유형을 삭제하십시오.
```

메시지 FLG0003E

메시지 설명은 다음과 같아야 합니다.

```
    정보 카탈로그는 이를 사용하기 전에 등록되어야 합니다.
정보 카탈로그가 제대로 등록되지 않을 수 있습니다.
```

메시지 FLG0372E

메시지 설명의 첫 번째 문장은 다음과 같아야 합니다.

```
    오브젝트가 Attachment 오브젝트이므로 오브젝트에 대한
ATTACHMENT-IND 값이 무시됩니다.
```

메시지 FLG0615E

메시지의 두 번째 구문은 다음과 같아야 합니다.

Information Catalog Manager가 예상치 못한 데이터베이스 오류를 발견했거나, 현재 디렉토리 또는 경로에서 바인드 파일을 찾을 수 없습니다.

Information Catalog Manager: 온라인 도움말

정보 카탈로그 창: 선택 항목 메뉴 열기 항목에 대한 "선택된 오브젝트를 엽니다" 라는 온라인 도움말은 잘못된 것입니다. 이를 "검색 정의 창을 엽니다"로 정정하십시오.

웹에 대한 Information Catalog Manager

OS/390용 DB2 UDB 시스템에 위치한 정보 카탈로그를 사용할 때 대소문자를 구별하지 않는 검색은 사용할 수 없습니다. 이 점은 간단한 검색과 고급 검색 모두에 해당합니다. 온라인 도움말은 OS/390용 DB2 UDB 정보 카탈로그에 대한 모든 검색이 간단한 검색의 경우 대소문자가 구별됨을 설명하지 않습니다. 게다가 모든 그룹화 범주 오브젝트는 아래에 오브젝트가 없어도 펼칠 수 있습니다.

DB2 Warehouse Manager 설치 안내서

DB2 Warehouse Manager 설치 안내서 갱신

DB2 Warehouse Manager 설치 안내서는 갱신된 최신 .pdf를 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 서비스를 통해 주문할 수 있습니다. 또한 갱신된 참조에 이러한 참고 내용을 추가해 놓았습니다.

웨어하우스 변환기에 대한 소프트웨어 요구사항

웨어하우스 변환기를 사용할 데이터베이스에 JDK(Java Developer's Kit) 버전 1.1.8 이상을 설치해야 합니다.

iSeries 에이전트 설치 후 고려사항

제 4 장 ″AS/400(iSeries) 웨어하우스 에이전트 설치″ 섹션 아래의 ″설치 후 고려사항″ 아래에서 첫 단락을 다음과 같이 바꾸십시오.

> 웨어하우스 에이전트는 모든 단계 기능은 한 작업 단위로 수행합니다. V4R5 이전에 iSeries DB2 Universal Database에서는 한 커미트 범위에 삽입할 수 있는 행 수를 4백만 개로 제한했습니다. V4R5에서는 이 제한이 5억 개 행으로 증가되었습니다. V4R4(이하) 시스템을 사용하며 쿼리가 이 크기를 초과하는 경우, 쿼리를 나누거나 웨어하우스 제공 FTP 프로그램을 사용하여 데이터를 이동하십시오.

iSeries 웨어하우스 에이전트와 함께 변환기를 사용하기 전에

제 4 장에서 ″iSeries 에이전트에 변환기를 사용하기 전에″ 섹션과 모든 서브섹션을 제거하십시오.

Query Patroller 관리 안내서

DB2 Query Patroller 클라이언트는 별도의 구성요소임

DB2 Query Patroller 클라이언트는 DB2 Administration Client의 일부가 아닌 별도의 구성요소입니다. 이것은 Query Patroller 설치 안내서에 설명된 대로 DB2 Administration Client의 설치 중에 함께 설치되지 않는다는 의미입니다. 대신 Query Patroller 클라이언트를 별도로 설치한다는 것을 의미합니다.

Query Patroller 클라이언트와 Query Patroller 서버의 버전이 동일해야 합니다.

노드 상태 변경

다음은 Query Patroller 관리 안내서의 노드 관리 섹션에 대한 갱신사항입니다.

노드 상태를 변경하려면 다음 프로시저를 사용하십시오.

1. 노드 관리 페이지에서 노드를 선택하십시오.
2. **보기/편집**을 누르십시오.

   노드에 대한 세부사항 정보 창이 열립니다.
3. **요청된 상태** 필드에서 새 상태를 선택하십시오.

   주: **요청된 상태**는 변경될 수 있는 노드 창에 대한 세부사항 정보에 있는 유일한 필드이며, 다른 모든 필드는 DB2 Query Patroller에서 제공된 값을 표시합니다.
4. **확인**을 누르십시오.

다음 목록에서는 각 노드 매개변수에 대한 정보를 제공합니다.

노드 ID
: 노드에 대한 ID를 제공합니다.

노드 상태
: 현재 노드 상태를 포함합니다.

- **활성**은 노드가 작업을 실행할 수 있음을 나타냅니다.
- **비활성**은 노드의 DB2 Query Patroller 구성요소가 종료되었음을 나타냅니다. 노드는 DB2 Query Patroller에서 사용 불가능합니다. 노드를 재활성화하려면, `iwm` 관리 사용자 어카운트를 사용하여 `dqpstart` 명령을 실행하십시오.
- **Quiesce 진행 중**은 노드가 quiescent 상태로 변환되고 있음을 나타냅니다. 작업 실행은 완료되지만, 노드에서의 새 작업은 스케줄되지 않습니다.
- **Quiesce 상태**는 노드가 quiescent임을 나타냅니다. 노드가 DB2 Query Patroller에서 사용 가능하지만 해당 노드에 대한 새 작업은 스케줄되고 있지 않습니다.

요청된 상태

노드 상태가 무엇으로 변경되는지를 나타냅니다.

- **활성**은 노드가 활성 상태가 됨을 나타냅니다.
- **비활성**은 노드가 비활성 상태가 됨을 나타냅니다. 작업 실행은 완료되지만, 새 작업은 스케줄되지 않습니다.
- **강제**는 노드가 즉시 비활성 상태가 됨을 나타냅니다. 작업 실행은 즉시 완료되며 새 작업은 스케줄되지 않습니다.
- **Quiesce 상태**는 노드가 quiescent됨을 나타냅니다. 작업 실행이 완료됩니다.

날짜/시간 최종 상태

날짜 및 시간 노드 상태가 최종 변경되었음을 나타냅니다.

스케줄된 작업

이 노드에서 실행하도록 스케줄된 작업 수와 이 노드에서 실행 중인 작업 수를 더하여 제공합니다.

CPU 사용

노드의 CPU 사용을 백분율(0 -100)로 제공합니다. CPU 사용 정보가 수집되고 있지 않은 경우, 값은 -1입니다.

사용 가능한 디스크

결과가 작성된 파일 시스템에서 사용 가능한 바이트 수를 나타냅니다. 디스크 사용이 모니터되고 있지 않은 경우, 값은 -1입니다.

노드 관리 프로그램 PID

노드 관리 프로그램 프로세스의 프로세스 ID를 나타냅니다.

dqpmigrate를 사용하여 DB2 Query Patroller로부터 이주

버전 7 Query Patroller Server가 버전 6 Query Patroller Server 위에 설치되었을 경우 dpqmigrate 명령을 반드시 사용해야 합니다. FixPak 2 이상의 경우, FixPak을 설치하면 이 명령을 대신 수행하므로 직접 dqpmigrate를 수행하지 않아도 됩니다. 이 명령을 사용하지 않으면, v6에 정의된 기본 사용자는 버전 7에서 추가된 여러 새 저장 프로시저에 EXECUTE 특권이 없습니다.

주: `dqpmigrate.bnd`는 `sqllib/bnd` 디렉토리에 그리고 `dqpmigrate.exe`는 `sqllib/bin` 디렉토리에 있습니다.

dqpmigrate를 수동으로 사용하여 EXECUTE 특권을 부여하려면, FixPak을 설치한 후 다음을 수행하십시오.

1. 다음 명령을 입력하여 `/sqllib/bnd/dqpmigrate.bnd` 패키지 파일을 Query Patroller 서버가 설치된 데이터베이스에 바인드하십시오.

   ```
   db2 bind dqpmigrate.bnd
   ```

2. 다음을 입력하여 **dqpmigrate**를 실행하십시오.

   ```
   dqpmigrate dbalias userid passwd
   ```

쿼리 관리 사용

″Getting Started″ 장의 ″Enabling Query Management″에서 문장은 다음과 같아야 합니다.

```
You must be the owner of the data base, or you must have SYSADM,
SYSCTRL, or SYSMAINT authority to set database configuration parameters.
```

제어 테이블용 테이블 스페이스의 위치

제 1 장 "시스템 개요"의 DB2 Query Patroller 제어 테이블에서 다음 텍스트가 이 섹션의 첫 번째 단락 끝에 추가되었습니다.

DB2 Query Patroller 제어 테이블용 테이블 스페이스는 단일 노드로 된 노드 그룹에 있어야 하며, 그렇지 않으면 DB2 Query Patroller가 제대로 작동하지 않습니다.

dqpstart 명령에 사용할 수 있는 새로운 매개변수

제 2 장 시작하기의 DB2 Query Patroller 시작 및 중지에서 다음 텍스트가 마지막 단락 뒤에 추가되었습니다.

dqpstart 명령에 사용할 수 있는 새로운 매개변수는 다음과 같습니다.

RESTART 매개변수:

호스트 이름 및/또는 `dqpnodes.cfg` 파일에 지정된 노드의 노드 유형을 바꿀 수 있도록 합니다. DB2 Query Patroller는 이 노드에서 시작됩니다.

주: RESTART 매개변수를 지정하여 DQPSTART 명령을 수행하기 전에 다음 사항을 확인하십시오.

1. 바뀔 호스트에서 DB2 Query Patroller를 중지했어야 합니다.
2. 새 호스트에서 DB2 Query Patroller가 실행되고 있지 않아야 합니다.

구문은 다음과 같습니다.

dqpstart nodenum *node_num* restart *hostname* server | agent | none

ADDNODE 매개변수:

`dqpnodes.cfg` 파일에 새로운 노드를 추가할 수 있도록 합니다. dqpnodes.cfg 파일에 새로운 노드 항목을 추가하면 DB2 Query Patroller는 추가한 노드에서 시작됩니다. 구문은 다음과 같습니다.

dqpstart nodenum *node_num* addnode *hostname* server | agent | none

DROPNODE 매개변수:

`dqnodes.cfg` 파일에서 노드를 삭제할 수 있도록 합니다. dqpnodes.cfg

파일에서 노드 항목을 삭제하기 전에는 DB2 Query Patroller가 이 노드에서 시작됩니다. 구문은 다음과 같습니다.

```
dqpstop nodenum node_num dropnode
```

iwm_cmd 명령에 사용할 수 있는 새로운 매개변수

새로운 **-v** 매개변수가 **iwm_cmd** 명령에 추가되어 지정한 노드에서 수행 중이었던 작업 상태를 복구할 수 있게 되었습니다. 비활성 노드에 있는 작업만 복구할 수 있습니다. 이 명령은 노드 실패시 이 노드에서 수행 중이었거나 취소 중이었던 작업이 있으면 발행됩니다. ″Running″ 상태에 있었던 작업은 다시 제출되고 ″Queued″ 상태로 되돌려집니다. ″Cancelling″ 상태에 있었던 작업은 ″Cancelled″ 상태로 설정됩니다.

부분 구문은 다음과 같습니다.

```
>>──iwm_cmd──┬─────────────────────────────────┬──-v──node_id_to_recover──────────────><
             └─-u──user_id──┬─────────────────┬┘
                            └─-p──password────┘
```

node_id_to_recover

작업이 복구될 노드를 지정합니다.

새 레지스트리 변수: DQP_RECOVERY_INTERVAL

iwm_scheduler가 복구 파일을 검색하는 시간 간격을 분 단위로 설정하는 데 사용할 수 있는 DQP_RECOVERY_INTERVAL이라는 새로운 레지스트리 변수가 추가되었습니다. 기본값은 60분입니다.

쿼리 관리자 시작

″Using QueryAdministrator to Administer DB2 Query Patroller″ 장에서는 Windows의 시작 메뉴에서 쿼리 관리자를 시작하기 위한 지시사항을 제공합니다. 첫 단계는 다음의 텍스트 내용을 포함하고 있습니다.

```
If you are using Windows, you can select DB2
Query Patroller --> QueryAdministrator
from the IBM DB2 program group.
```

이 문장은 다음과 같이 바뀌어야 합니다.

DB2 Query Patroller --> QueryAdmin.

사용자 관리

″Using QueryAdministrator to Administer DB2 Query Patroller″ 장의 ″User Administration″ 섹션에서 최대 경과 시간 매개변수에 대한 정의는 값이 0 또는 -1로 설정된 경우, 항상 쿼리가 완료될 때까지 수행된다고 나와있습니다. 이 내용은 잘못된 것입니다. 이 매개변수는 음수 값으로 설정할 수 없습니다. 따라서 값이 ″0″으로 설정된 경우, 항상 쿼리 완료될 때까지 수행됩니다.

Max Queries 매개변수는 DB2 Query Patroller가 동시에 수행할 최대 작업 수를 지정합니다. Max Queries는 0 - 32767 범위 내의 정수이어야 합니다.

데이터 소스 관리

제 3 장, Query Administrator를 사용하여 DB2 Query Patroller 관리에서 데이터 소스 매개변수에 대해 새로운 설명 및 변경된 설명이 있습니다.

정적 비용은 계산된 DB2 쿼리 비용(timerons 단위)입니다. 비용은 각 작업의 작업 항목에 저장됩니다. Query Monitor를 사용하여 작업의 작업 세부사항을 살펴볼 때 이 비용은 계산된 비용으로 표시됩니다.

Zero Cost Query는 정적 비용 또는 계산된 비용이 0인 쿼리입니다. 실제로 계산된 비용이 0인 쿼리는 없습니다(아무리 간단한 쿼리도 비용은 약 5입니다). 비용이 0인 경우는 작업이 비용 분석 수행하지 않음 옵션으로 제출된 경우입니다. 사용자 프로파일에서 이를 허용하도록 설정한 경우에만 이 옵션을 선택할 수 있습니다. 대부분의 경우, 사용자 프로파일이 이러한 방식으로 설정되지 않습니다. 다른 관리자, 선택된 특수 사용자 또는 자기 자신과 같은 수퍼유저의 경우 이 옵션을 선택해야 합니다. 그러면 이러한 사용자는 원하는 모든 쿼리를 수행할 수 있습니다. 시스템은 이러한 사용자의 쿼리를 비용 0으로 처리하므로 쿼리는 우선순위가 높은 것으로 처리됩니다.

비용 시간 제로, 비용 시간 기울기, 비용 시간 간격 및 비용 시간 최소값은 더 이상 사용되지 않습니다.

비용 인수는 timeron 단위의 정적 비용을 변환하기 위한 배수이며 회계 테이블의 비용이 아닙니다. 회계 테이블의 비용은 정적 비용을 비용 인수로 곱한 값입니다.

작업 대기행렬 작성

″Using QueryAdministrator to Administer DB2 Query Patroller″ 장의 ″Job Queue Administration″ 섹션에서 ″작업 큐 작성″에 대한 단계의 화면 캡처는 두 번째 단계 이후에 표시되어야 합니다. 새로운 작업 대기행렬에 대한 정보 창은 쿼리 관리자 도구의 작업 대기행렬 관리 페이지에서 **새로 작성**을 누르면 열립니다.

작업 대기행렬 페이지나 작업 대기행렬 탭은 작업 대기행렬 관리 페이지와 작업 대기행렬 관리 탭으로 바뀌어야 합니다.

작업 회계 테이블

제 11 장 DB2 Query Patroller 시스템 모니터링에서, 작업 회계에 대한 섹션에는 작업 회계 테이블의 컬럼에 대한 설명이 있습니다. 테이블 이름은 IWM.IWM003_JOB_ACCT입니다.

명령행 인터페이스 사용

DB2 Query Patroller 시스템에서 사용자 권한이 있는 사용자가 쿼리를 제출하고 결과 테이블을 작성하려면, 이 사용자는 데이터베이스에서 CREATETAB 권한이 필요합니다. 사용자는 DQP_RES_TBLSPC 프로파일 변수가 설정되지 않은 상태로 남아 있거나 DQP_RES_TBLSPC 프로파일 변수가 기본 테이블 스페이스의 이름으로 설정된 경우에는 데이터베이스에서 CREATETAB 권한이 필요하지 않습니다. 사용자가 기본 테이블 스페이스에서 테이블을 작성하는 권한을 가지고 있기 때문에 이 경우, 결과 테이블은 정상적으로 작성됩니다.

Query Enabler 참고

- 키 세트 커서를 사용하는 써드 파티 쿼리 도구를 사용할 경우, 쿼리는 인터셉트되지 않습니다. Query Enabler가 이러한 쿼리를 인터셉트하도록 하려면, 다음을 포함도록 `db2cli.ini` 파일을 수정해야 합니다.

```
[common]
DisableKeySetCursor=1
```

- AIX 클라이언트의 경우, 환경 변수 LIBPATH가 설정되어 있는지 확인하십시오. JDK와 함께 제공되는 라이브러리 libXext.a는 /usr/lib/X11 서브디렉토리의 라이브러리와 호환되지 않습니다. 이것은 Query Enabler GUI에 문제점을 야기합니다.

DB2 Query Patroller 추적 프로그램이 공백 컬럼 페이지를 리턴

FixPak 3에 DB2 Query Patroller 추적 프로그램에 대한 수정사항이 포함되어 있습니다. 추적 프로그램은 이제 아무 컬럼에도 해당되지 않는 쿼리를 올바르게 보고합니다. 그런 쿼리의 예가 ″SELECT COUNT(*) FROM ...″입니다. 이 종류의 쿼리는 테이블에서 아무 컬럼에도 해당되지 않으므로, 추적 프로그램은 컬럼 페이지에 공백 페이지를 나타냅니다. 이 공백 컬럼 페이지는 결함이 아닙니다.

DB2 Query Patroller Tracker GUI 도구에 대한 추가 정보

회계 테이블은 추적 프로그램 도구에서 작업 데이터를 표시하거나 분석하는 데 사용됩니다. 추적 프로그램을 사용하려면, 관리자는 우선 Query Administrator를 사용하여 시스템 관리자 패널의 회계 상태를 테이블에 기록으로 바꾸어야 합니다. 그러면 작업이 완료될 때마다 작업 회계 테이블에 추가 정보가 저장됩니다.

그 다음 관리자는 Query Patroller 서버에 사용자 iwm으로 로그온하여 iwm_tracker(Tracker backend) 도구를 실행해야 합니다. 시스템 로드가 낮거나 추적 프로그램 도구가 사용되기 바로 전에 이 도구를 주기적으로 실행해야 합니다.

마지막으로 이 두 태스크가 완료되면, 추적 프로그램 GUI 도구를 사용하여 작업 데이터를 보거나 분석할 수 있습니다.

비용 인수가 기본값인 1인 경우, 추적 프로그램을 사용하여 각 작업에 대해 표시되는 비용은 Query Monitor를 사용하여 표시되는 비용과 같은 값입니다. 두 경우 모두, 시간은 timerons 단위입니다.

그러나, 값의 단위로 다른 것을 사용하고자 할 수 있습니다. 사용자에게 시스템의 사용에 대해 청구하고자 한다면, 예를 들어 비용이 10 000 timerons의 작업에 1

달러인 경우 비용 인수를 0.0001로 입력하면 됩니다. 그러면 추적 프로그램이 각 작업 비용을 달러로 변환, 저장, 표시하게 됩니다.

*Query Patroller Administration Guide*에서는 메가바이트당 달러로 기술하고 있으나 이 정보는 올바르지 않으며 timeron당 달러로 바꾸어야 합니다.

Query Patroller 및 복제 도구

Query Patroller 버전 7은 복제 도구(asnapply, asnccp, djra and analyze)의 쿼리를 인터셉트하므로 이러한 도구가 제대로 작동하지 않습니다. 이 문제점을 해결하려면 이 도구를 실행할 때 동적 쿼리 관리를 해제하십시오.

Query Patroller 성능 향상

다음 텍스트는 제 6 장 성능 조정 끝부분에서 다룹니다.

BIND 옵션, INSERT BUF를 사용한 DB2 Query Patroller 성능 향상

기본적으로 DB2 Query Patroller는 결과 테이블을 작성해 관리할 쿼리 결과를 저장합니다. 이 결과 테이블의 삽입 성능을 향상시키려면 DB2 Query Patroller 바인드 파일 중 하나를 바인딩할 때 INSERT BUF 옵션을 포함하십시오.

다음과 같이 DB2 Query Patroller 바인드 파일을 데이터베이스에 바인딩하십시오.

Windows의 DB2_RUNTIME\bnd 디렉토리 또는 UNIX의 DB2_RUNTIME/bnd 경로에서 다음 명령을 입력하십시오.

```
db2 connect to database user iwm using password
db2 bind @db2qp.lst blocking all grant public
db2 bind iwmsx001.bnd insert buf
db2 bind @db2qp_sp.lst
db2 commit
```

여기서, `database`는 DB2 Query Patroller에서 관리할 대체 데이터베이스이고, `password`는 관리 사용자 어카운트 iwm의 암호입니다.

버전 6에서 작성된 Query Patroller 사용자의 EXECUTE 특권 상실

Query Patroller 버전 7에 추가된 새로운 저장 프로시저(IWM.DQPGROUP, IWM.DQPVALUR, IWM.DQPCALCT 및 IWM.DQPINJOB)로 인해, Query Patroller 버전 6에서 작성한 기존 사용자들은 이들 패키지에 대해 EXECUTE 권한을 갖지 않습니다. 이 문제점을 자동으로 수정하는 응용프로그램이 FixPak 1에 추가되었습니다.

DQP Query Admin을 사용하여 DQP 사용자 정보를 수정하려 할 때는 사용자 목록에서 기존 사용자를 제거하지 마십시오.

Query Patroller 제한사항

JVM(Java Virtual Machine) 플랫폼 제한사항으로 인해, Query Enabler는 HP-UX 및 NUMA-Q에서 지원되지 않습니다. 뿐만 아니라, Query Patroller 추적 프로그램은 NUMA-Q에서 지원되지 않습니다. 모든 Query Patroller 클라이언트 도구가 필수인 경우, 다른 플랫폼(Windows NT와 같은)을 사용하여 HP-UX 또는 NUMA-Q 서버에서 이들 도구를 실행하십시오.

부록 B DB2 Query Patroller 문제점 해결

부록 B DB2 Query Patroller 클라이언트 문제점 해결에서 Common Query Enabler Problems, 문제점 #2 섹션의 첫 번째 bullet의 텍스트를 다음으로 바꾸어야 합니다.

경로 설정에 `jre`가 포함되어 있는지 확인하십시오.

제 6 부 응용프로그램 개발

Administrative API Reference

db2ArchiveLog(새 API)

db2ArchiveLog

복구 가능한 데이터베이스에 사용 중인 로그 파일을 닫고 뒤를 자릅니다. User Exit를 사용할 경우, 아카이브 요청을 발행하십시오.

권한 부여

다음 중 하나의 권한으로 수행하십시오.

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

필수 연결

이 API는 지정된 데이터베이스에 자동으로 연결을 설정합니다. 지정된 데이터베이스와 연결이 이미 존재할 경우, API는 오류를 리턴합니다.

API 포함 파일

db2ApiDf.h

C API 구문

```
/* File: db2ApiDf.h */
/* API:  Archive Active Log */
SQL_API_RC SQL_API_FN
   db2ArchiveLog (
      db2Uint32 version,
      void *pDB2ArchiveLogStruct,
      struct sqlca * pSqlca);

typedef struct
{
   char                *piDatabaseAlias;
   char                *piUserName;
   char                *piPassword;
   db2Uint16           iAllNodeFlag;
   db2Uint16           iNumNodes;
   SQL_PDB_NODE_TYPE   *piNodeList;
   db2Uint32           iOptions;
} db2ArchiveLogStruct
```

일반 API 구문

```
/* File: db2ApiDf.h */
/* API:  Archive Active Log */
SQL_API_RC SQL_API_FN
   db2gArchiveLog (
      db2Uint32 version,
      void *pDB2ArchiveLogStruct,
      struct sqlca * pSqlca);

typedef struct
{
   db2Uint32           iAliasLen;
   db2Uint32           iUserNameLen;
   db2Uint32           iPasswordLen;
   char                *piDatabaseAlias;
   char                *piUserName;
   char                *piPassword;
   db2Uint16           iAllNodeFlag;
   db2Uint16           iNumNodes;
   SQL_PDB_NODE_TYPE   *piNodeList;
   db2Uint32           iOptions;
} db2ArchiveLogStruct
```

API 매개변수

version

입력. 두 번째 매개변수, *pDB2ArchiveLogStruct*로서 전달된 변수의 버전 및 릴리스 레벨을 지정합니다.

pDB2ArchiveLogStruct

입력. *db2ArchiveLogStruct* 구조를 가리키는 포인터.

pSqlca

출력. *sqlca* 구조를 가리키는 포인터.

iAliasLen

입력. 데이터베이스 별명의 길이를 바이트 단위로 표시하는 4바이트 부호 없는 정수.

iUserNameLen

사용자 이름의 길이를 바이트 단위로 표시하는 4바이트 부호 없는 정수. 사용자 이름을 사용하지 않을 경우 영(0)으로 설정하십시오.

iPasswordLen

입력. 암호의 길이를 바이트 단위로 표시하는 4바이트 부호 없는 정수. 암호를 사용하지 않을 경우 영(0)으로 설정하십시오.

piDatabaseAlias

입력. 사용 중인 로그가 아카이브될 데이터베이스의 별명(시스템 데이터베이스 디렉토리에 카탈로그된 대로)이 들어 있는 문자열.

piUserName

입력. 연결을 시도할 때 사용될 사용자 이름이 들어 있는 문자열.

piPassword

입력. 연결을 시도할 때 사용될 암호가 들어 있는 문자열.

iAllNodeFlag

MPP 전용. 입력. 작업이 `db2nodes.cfg` 파일에 나열된 모든 노드에 적용될지 여부를 표시하는 플래그. 유효한 값은 다음과 같습니다.

DB2ARCHIVELOG_NODE_LIST

*piNodeList*에 전달되는 노드 목록의 노드에 적용합니다.

DB2ARCHIVELOG_ALL_NODES

모든 노드에 적용합니다. *piNodeList*는 널(NULL)이어야 합니다. 이것은 기본값입니다.

DB2ARCHIVELOG_ALL_EXCEPT

*piNodeList*에 전달된 노드 목록에 있는 노드를 제외한 모든 노드에 적용합니다.

iNumNodes

MPP 전용. 입력. *piNodeList* 배열의 노드 수를 지정합니다.

piNodeList

MPP 전용. 입력. 아카이브 로그 작업을 적용할 노드 번호의 배열을 가리키는 포인터.

iOptions

입력. 이후로 사용을 위해 예약되어 있습니다.

db2ConvMonStream

사용 주의사항에서 스냅샷 변수 데이터 스트림 유형 SQLM_ELM_SUBSECTION 의 구조는 sqlm_subsection이어야 합니다.

db2DatabasePing(새 API)

db2DatabasePing - Ping 데이터베이스

클라이언트와 데이터베이스 서버 사이에서 기초적인 연결성의 네트워크 응답 시간을 테스트합니다. 이 API는 응용프로그램이 호스트 데이터베이스 서버에 DB2 Connect를 통해(직접 또는 게이트웨이를 통해) 액세스할 때 사용할 수 있습니다.

권한 부여

없음

필수 연결

데이터베이스

API 포함 파일

db2ApiDf.h

C API 구문

```
/* File: db2ApiDf.h */
/* API: Ping Database */
/* ... */


SQL_API_RC SQL_API_FN
  db2DatabasePing (
   db2Uint32 versionNumber,
   void         *pParmStruct,
   struct sqlca * pSqlca);
/* ... */

typedef SQL_STRUCTURE db2DatabasePingStruct
{
  char          iDbAlias[SQL_ALIAS_SZ + 1];
  db2Uint16     iNumIterations;
  db2Uint32    *poElapsedTime;
}
```

일반 API 구문

```
/* File: db2ApiDf.h */
/* API: Ping Database */
/* ... */
SQL_API_RC SQL_API_FN
  db2gDatabasePing (
   db2Uint32 versionNumber,
```

```
    void          *pParmStruct,
    struct sqlca * pSqlca);
  /* ... */

  typedef SQL_STRUCTURE db2gDatabasePingStruct
  {
    db2Uint16      iDbAliasLength;
    char           iDbAlias[SQL_ALIAS_SZ];
    db2Uint16      iNumIterations;
    db2Uint32     *poElapsedTime;
  }
```

API 매개변수

versionNumber

입력. 응용프로그램이 실행하는 DB2 Universal Database 또는 DB2 Connect 제품의 버전 및 릴리스.

주: db2Version710 상수 이상이 DB2 버전 7.1 이상에서 사용되어야 합니다.

| **pParmStruct**

입력. db2DatabasePingStruct 구조를 가리키는 포인터

iDbAliasLength

입력. 데이터베이스 별명 길이

주: 이 매개변수는 현재 사용되지 않습니다. 이후로 사용되기 위해 예약된 것입니다.

iDbAlias

입력. 데이터베이스 별명.

주: 이 매개변수는 현재 사용되지 않습니다. 이후로 사용되기 위해 예약된 것입니다.

iNumIterations

입력. 테스트 요청 반복 수. 값은 1 - 32767 이내이어야 합니다.

poElapsedTime

출력. 요소 갯수가 iNumIterations과 같은 32비트 정수 배열에 대한 포인터. 배열에 있는 각 요소에는 한 테스트 요청 반복에 소요되는 마이크로 단위의 시간이 들어갑니다.

주: 응용프로그램은 이 API를 호출하기 전에 이 배열에 메모리를 할당해야 합니다.

pSqlca

출력. *sqlca* 구조를 가리키는 포인터. *Administrative API Reference*에서 이 구조에 대한 자세한 내용을 참조하십시오.

사용 참고

이 API를 호출하기 전에 데이터베이스 연결이 있어야 하며, 그렇지 않으면 오류가 발생합니다.

이 기능은 또한 PING 명령을 사용하여 호출될 수 있습니다. 이 명령에 대한 설명은 *Command Reference*를 참조하십시오.

db2HistData

다음 항목을 표 11에 추가해야 합니다. db2HistData 구조의 필드는 다음과 같습니다.

필드 이름	데이터 유형	설명
oOperation	char	표 12 참조.
oOptype	char	표 13 참조.

다음 표가 다음 표 11에 추가됩니다.

표 12. db2HistData 구조에서 oOperation에 대해 유효한 이벤트 값

값	설명	C 정의	COBOL/FORTRAN 정의
A	테이블 스페이스 추가	DB2HISTORY_OP_ADD_TABLESPACE	DB2HIST_OP_ADD_TABLESPACE
B	백업	DB2HISTORY_OP_BACKUP	DB2HIST_OP_BACKUP
C	로드 사본	DB2HISTORY_OP_LOAD_COPY	DB2HIST_OP_LOAD_COPY
D	삭제된 테이블	DB2HISTORY_OP_DROPPED_TABLE	DB2HIST_OP_DROPPED_TABLE
F	롤 포워드	DB2HISTORY_OP_ROLLFWD	DB2HIST_OP_ROLLFWD
G	테이블 재구성	DB2HISTORY_OP_REORG	DB2HIST_OP_REORG
L	로드	DB2HISTORY_OP_LOAD	DB2HIST_OP_LOAD
N	테이블 스페이스 이름 바꾸기	DB2HISTORY_OP_REN_TABLESPACE	DB2HIST_OP_REN_TABLESPACE
O	테이블 스페이스 삭제	DB2HISTORY_OP_DROP_TABLESPACE	DB2HIST_OP_DROP_TABLESPACE
Q	quiesce	DB2HISTORY_OP_QUIESCE	DB2HIST_OP_QUIESCE
R	리스토어	DB2HISTORY_OP_RESTORE	DB2HIST_OP_RESTORE
S	통계 수행	DB2HISTORY_OP_RUNSTATS	DB2HIST_OP_RUNSTATS
T	테이블 스페이스 변경	DB2HISTORY_OP_ALT_TABLESPACE	DB2HIST_OP_ALT_TBS
U	로드 해제	DB2HISTORY_OP_UNLOAD	DB2HIST_OP_UNLOAD

다음 표도 추가됩니다.

표 13. 유효한 oOptype 값 db2HistData 구조

oOperation	oOptype	설명	C/COBOL/FORTRAN 정의

B	F	오프라인	DB2HISTORY_OPTYPE_OFFLINE
	N	온라인	DB2HISTORY_OPTYPE_ONLINE
	I	증분 오프라인	DB2HISTORY_OPTYPE_INCR_OFFLINE
	O	증분 온라인	DB2HISTORY_OPTYPE_INCR_ONLINE
	D	델타 오프라인	DB2HISTORY_OPTYPE_DELTA_OFFLINE
	E	델타 온라인	DB2HISTORY_OPTYPE_DELTA_ONLIN
F	E	로그의 끝	DB2HISTORY_OPTYPE_EOL
	P	특정 시점	DB2HISTORY_OPTYPE_PIT
L	I	삽입	DB2HISTORY_OPTYPE_INSERT
	R	바꾸기	DB2HISTORY_OPTYPE_REPLACE
Q	S	Quiesce 공유	DB2HISTORY_OPTYPE_SHARE
	U	Quiesce 갱신	DB2HISTORY_OPTYPE_UPDATE
	X	Quiesce 독점	DB2HISTORY_OPTYPE_EXCL
	Z	Quiesce 재설정	DB2HISTORY_OPTYPE_RESET
R	F	오프라인	DB2HISTORY_OPTYPE_OFFLINE
	N	온라인	DB2HISTORY_OPTYPE_ONLINE
	I	증분 오프라인	DB2HISTORY_OPTYPE_INCR_OFFLINE
	O	증분 온라인	DB2HISTORY_OPTYPE_INCR_ONLINE
T	C	컨테이너 추가	DB2HISTORY_OPTYPE_ADD_CONT
	R	재조정(Rebalance)	DB2HISTORY_OPTYPE_REB

db2HistoryOpenScan

다음 값이 **iCallerAction** 매개변수에 추가됩니다.

DB2HISTORY_LIST_CRT_TABLESPACE

다른 필터를 전달하는 CREATE TABLESPACE 및 DROP TABLESPACE 레코드만 선택하십시오.

db2XaGetInfo(새 API)

db2XaGetInfo - 자원 관리 프로그램에 대한 정보 얻기

xa_open 호출이 작성되면 특정 자원 관리 프로그램에 대한 정보를 추출합니다.

권한 부여

없음

필수 연결

데이터베이스

API 포함 파일

sqlxa.h

C API 구문

```
/* File: sqlxa.h */
/* API: Get Information for Resource Manager */
/* ... */
SQL_API_RC SQL_API_FN
db2XaGetInfo (
   db2Uint32 versionNumber,
   void * pParmStruct,
   struct sqlca * pSqlca);

typedef SQL_STRUCTURE db2XaGetInfoStruct
{
  db2int32 iRmid;
  struct sqlca oLastSqlca;
} db2XaGetInfoStruct;
```

API 매개변수

versionNumber

입력. 두 번째 매개변수 *pParmStruct*로 전달되는 구조의 버전 및 릴리스 레벨을 지정합니다.

pParmStruct

입력. *db2XaGetInfoStruct* 구조를 가리키는 포인터.

pSqlca

출력. *sqlca* 구조를 가리키는 포인터. *Administrative API Reference*에서 이 구조에 대한 자세한 내용을 참조하십시오.

iRmid 입력. 정보가 필요한 자원 관리 프로그램을 지정합니다.

oLastSqlca

출력. 마지막 XA API 호출에 대한 *sqlca*가 포함됩니다.

주: 마지막으로 실패한 XA API로부터 작성된 *sqlca*만을 검색할 수 있습니다.

db2XaListIndTrans(sqlxphqr을 대신하는 새 API)

db2XaListIndTrans - 2단계 커미트 중 이상 실패 트랜잭션 목록

현재 연결된 데이터베이스에 대한 모든 2단계 커미트 중 이상 실패 트랜잭션의 목록을 제공합니다.

범위

이 API는 발행된 노드에만 영향을 줍니다.

권한 부여

다음 중 하나의 권한으로 수행하십시오.

- *sysadm*
- *dbadm*

필수 연결

데이터베이스

API 포함 파일

db2ApiDf.h

C API 구문

```
/* File: db2ApiDf.h */
/* API: List Indoubt Transactions */
/* ... */
SQL_API_RC SQL_API_FN
db2XaListIndTrans (
   db2Uint32 versionNumber,
   void * pParmStruct,
   struct sqlca * pSqlca);

typedef SQL_STRUCTURE db2XaListIndTransStruct
{
  db2XaRecoverStruct * piIndoubtData;
  db2Uint32            iIndoubtDataLen;
  db2Uint32            oNumIndoubtsReturned;
  db2Uint32            oNumIndoubtsTotal;
  db2Uint32            oReqBufferLen;
} db2XaListIndTransStruct;

typedef SQL_STRUCTURE db2XaRecoverStruct
{
  sqluint32      timestamp;
  SQLXA_XID      xid;
  char           dbalias[SQLXA_DBNAME_SZ];
  char           applid[SQLXA_APPLID_SZ];
  char           sequence_no[SQLXA_SEQ_SZ];
  char           auth_id[SQL_USERID_SZ];
  char           log_full;
  char           connected;
  char           indoubt_status;
  char           originator;
  char           reserved[8];
} db2XaRecoverStruct;
```

API 매개변수

versionNumber

입력. 두 번째 매개변수 *pParmStruct*로 전달되는 구조의 버전 및 릴리스 레벨을 지정합니다.

pParmStruct

입력. *db2XaListIndTransStruct* 구조를 가리키는 포인터.

pSqlca

출력. *sqlca* 구조를 가리키는 포인터. *Administrative API Reference*에서 이 구조에 대한 자세한 내용을 참조하십시오.

piIndoubtData

입력. 2단계 커미트 중 이상 실패 데이터가 리턴될 응용프로그램 제공 버퍼를 가리키는 포인터. 2단계 커미트 중 이상 실패의 형식은 *db2XaRecoverStruct*입니다. 응용프로그램은 이 매개변수가 제공한 주소에서 시작하는 *db2XaRecoverStruct* 구조의 크기를 사용하여 2단계 커미트 중 이상 실패 트랜잭션의 목록을 통과할 수 있습니다.

값이 널(NULL)이면, DB2는 필요한 버퍼의 크기를 계산하여 이 값을 *oReqBufferLen*에 리턴합니다. *oNumIndoubtsTotal*에는 2단계 커미트 중 이상 실패 트랜잭션의 총 수가 포함됩니다. 응용프로그램은 필요한 버퍼 크기를 할당하고 API를 다시 발행할 수 있습니다.

oNumIndoubtsReturned

출력. *pIndoubtData*에 의해 지정된 버퍼에 리턴된 2단계 커미트 중 이상 실패 트랜잭션 레코드의 수.

oNumIndoubtsTotal

출력. API 호출 시에 사용 가능한 총 2단계 커미트 중 이상 실패 트랜잭션 레코드의 수. *piIndoubtData* 버퍼가 모든 레코드를 수용하기에 너무 작은 경우, *oNumIndoubtsTotal*은 *oNumIndoubtsReturned*에 대한 총계보다 커집니다. 응용프로그램은 모든 레코드를 확보하기 위해 API를 다시 발행할 수 있습니다.

주: 이 수는 자동 또는 경험적 2단계 커미트 중 이상 실패 트랜잭션 재동기화의 결과나 2단계 커미트 중 이상 실패 상태에 들어가는 다른 트랜잭션의 결과로 API 호출 간에 변경될 수 있습니다.

oReqBufferLen

출력. API 호출 시에 모든 2단계 커미트 중 이상 실패 트랜잭션 레코드를 보유할 필수 버퍼 길이. 응용프로그램은 *pIndoubtData*를 널(NULL)로 설정하여 API를 호출함으로써 필수 버퍼 크기를 판별하는 데 이 값을 사

용할 수 있습니다. 그런 다음, 이 값은 필수 버퍼를 할당하는 데 사용할 수 있으며 API는 *pIndoubtData*를 할당된 버퍼의 주소로 설정하여 발행할 수 있습니다.

주: 필수 버퍼 크기는 자동 또는 경험적 2단계 커미트 중 이상 실패 트랜잭션 재동기화의 결과나 2단계 커미트 중 이상 실패 상태에 들어가는 다른 트랜잭션의 결과로 API 호출들 간에 변경될 수 있습니다. 응용 프로그램은 이 점을 고려하여 보다 큰 버퍼를 할당할 수 있습니다.

timestamp
출력. 트랜잭션이 2단계 커미트 중 이상 실패 상태에 들어간 시간을 지정합니다.

xid 출력. 전역 트랜잭션을 고유 식별하기 위해 트랜잭션 관리 프로그램이 지정하는 XA 식별자를 지정합니다.

dbalias
출력. 2단계 커미트 중 이상 실패 트랜잭션이 발견된 데이터베이스의 별명을 지정합니다.

applid 출력. 데이터베이스 관리 프로그램이 이 트랜잭션에 지정한 응용프로그램 식별자를 지정합니다.

sequence_no
출력. 데이터베이스 관리 프로그램이 확장자로서 *applid*에 지정한 순차 번호를 지정합니다.

auth_id
출력. 트랜잭션을 수행한 사용자의 권한 부여 ID를 지정합니다.

log_full
출력. 이 트랜잭션으로 인해 로그가 가득 찼는지 여부를 나타냅니다. 유효한 값은 다음과 같습니다.

SQLXA_TRUE
이 2단계 커미트 중 이상 실패 트랜잭션으로 인해 로그가 가득 찼습니다.

SQLXA_FALSE

이 2단계 커미트 중 이상 실패 트랜잭션으로 인해 로그가 가득 차지는 않았습니다.

connected

출력. 응용프로그램의 연결 여부를 나타냅니다. 유효한 값은 다음과 같습니다.

SQLXA_TRUE

트랜잭션이 정상 동기점 처리 중이며, 2단계 커미트 중 두 번째 단계를 기다리는 중입니다.

SQLXA_FALSE

이전 실패에 의해 트랜잭션이 2단계 커미트 중 이상 실패 상태로 남았으며 이제 트랜잭션 관리 프로그램으로부터의 재동기화를 기다리고 있습니다.

indoubt_status

출력. 이 2단계 커미트 중 이상 실패 트랜잭션의 상태를 나타냅니다. 유효한 값은 다음과 같습니다.

SQLXA_TS_PREP

트랜잭션이 준비되었습니다. 연결된 매개변수는 트랜잭션이 정상 커미트 처리의 두 번째 단계를 기다리고 있는지 또는 오류가 발생하여 트랜잭션 관리 프로그램과의 재동기화가 필요한 지를 판별하는 데 사용할 수 있습니다.

SQLXA_TS_HCOM

트랜잭션이 경험적으로 커미트되었습니다.

SQLXA_TS_HROL

트랜잭션이 경험적으로 롤백되었습니다.

SQLXA_TS_MACK

트랜잭션에서 파티션 데이터베이스의 노드로부터 커미트 승인이 누락되었습니다.

SQLXA_TS_END

트랜잭션이 이 데이터베이스에서 종료되었습니다. 이 트랜잭션은 나

중에 다시 활성화되거나 커미트 또는 롤백됩니다. 트랜잭션 관리 프로그램에 오류가 발생하여 트랜잭션이 완료되지 않을 가능성도 있습니다. 이와 같은 경우, 이 트랜잭션에는 경험적 조치가 필요한데, 잠금을 보유하고 있으며 다른 응용프로그램이 데이터에 액세스하는 것을 방지할 수 있기 때문입니다.

사용 참고

일반 응용프로그램은 현재 연결을 데이터베이스 또는 파티션 데이터베이스 조정자 노드로 설정한 후에 다음의 단계를 수행합니다.

1. *piIndoubtData*를 널(NULL)로 설정하여 **db2XaListIndTrans**를 호출하십시오. 그러면, *oReqBufferLen* 및 *oNumIndoubtsTotal*에 값이 리턴됩니다.
2. *oReqBufferLen*에 리턴된 값을 사용하여 버퍼를 할당하십시오. 이 API의 초기 호출은 *oReqBufferLen*을 확보하기 위한 것이므로 추가 2단계 커미트 중 이상 실패 트랜잭션이 있으면 이 버퍼는 크기가 충분하지 않을 수도 있습니다. 응용프로그램은 *oReqBufferLen*보다 큰 버퍼를 제공합니다.
3. 모든 2단계 커미트 중 이상 실패 트랜잭션 레코드가 확보되었는지 판별하십시오. 이 작업은 *oNumIndoubtsReturned*를 *oNumIndoubtTotal*과 비교하여 수행할 수 있습니다. *oNumIndoubtsTotal*이 *oNumIndoubtsReturned*보다 크면, 응용프로그램은 위의 단계를 반복할 수 있습니다.

참조

*Administrative API Reference*에서 ″sqlxhfrg - Forget Transaction Status″, ″sqlxphcm - Commit an Indoubt Transaction″ 및 ″sqlxphrl - Roll Back an Indoubt Transaction″을 참조하십시오.

db2GetSnapshot - 스냅샷 가져오기

db2GetSnapshot API의 구문은 다음과 같아야 합니다.

```
int db2GetSnapshot( unsigned char version;
db2GetSnapshotData *data,
struct sqlca *sqlca);


데이터에 설명된 매개변수는 다음과 같습니다.
typedef struct db2GetSnapshotData{
   sqlma  *piSqlmaData;
   sqlm_collected  *poCollectedData
   void  *poBuffer;
   db2uint32 iVersion;
   db2int32 iBufferSize;
   db2uint8 iStoreResult;
db2uint16 iNodeNumber;
db2uint32 *poOutputFormat;


}db2GetSnapshotData;
```

Forget 로그 레코드

다음 정보는 부록 F의 MPP Subordinator Prepare 섹션 다음에 추가됩니다.

이 로그 레코드는 2단계 커미트중 이상 트랜잭션이 롤백되거나 2단계 커미트가 커미트된 후 기록되며, 트랜잭션의 종료와 보유된 모든 로그 자원이 해제되었음을 표시합니다. 트랜잭션을 종료하려면 트랜잭션이 경험적으로 완료된 상태에 있어야 합니다.

표 10. Forget 로그 레코드 구조

설명	유형	오프셋(바이트)
로그 헤더	LogManagerLogRecordHeader	0(20)
time	sqluint64	20(8)
총 길이: 28바이트		

sqlaintp - 오류 메시지 가져오기

다음의 사용 주의사항이 이 API의 설명에 추가됩니다.

```
멀티스레드 응용프로그램인 sqlaintp를 올바른 컨텍스트에
추가해야 합니다. 아니면, SQLCODE -1445용 메시지
텍스트를 볼 수 없습니다.
```

sqlbctcq - 테이블 스페이스 컨테이너 쿼리 닫기

이 API에 대해 로드는 유효한 권한 부여 레벨이 아닙니다.

sqleseti - 클라이언트 정보 설정

SQL 특수 레지스터는 API가 제공하는 데이터 값에 액세스할 수 있습니다. 이 레지스터에 들어 있는 값은 데이터베이스 코드 페이지에 저장됩니다. API가 제공하는 데이터 값은 데이터베이스 코드 페이지로 변환된 다음 특수 레지스터에 저장됩니다. 만약 데이터 값이 데이터베이스 코드 페이지로 변환한 뒤 지원되는 최대 크기를 초과하게 되면 *크기*가 절단되어 서버에 저장되게 됩니다. 이 절단된 값은 특수 레지스터에 의해 리턴됩니다. 원래 데이터 값은 서버에 저장되며 데이터베이스 코드 페이지로 변환되지 않습니다. 변환되지 않은 값은 **sqleqryi** API를 호출하여 리턴할 수 있습니다.

sqlubkp - 데이터베이스 백업

BackupType 매개변수에 대해 SQLUB_FULL 값이 SQLUB_DB로 바뀌었습니다. 데이터베이스의 모든 테이블 스페이스를 백업합니다.

새 증분 백업 기능을 지원하기 위해, SQLUB_INCREMENTAL 및 SQLUB_DELTA 매개변수도 추가되었습니다. 증분 백업 이미지는 가장 최근의 성공적인 전체 백업 이후에 변경된 모든 데이터베이스 데이터의 사본입니다. 델타 백업 이미지는 어떤 유형이든 가장 최근의 성공적인 백업 이후에 변경된 모든 데이터베이스 데이터의 사본입니다.

sqlureot - 테이블 재구성

다음 문장이 사용 참고에 추가됩니다.

REORGANIZE TABLE은 인덱스 확장자에 기반한 인덱스를 사용할 수 없습니다.

sqlurestore - 데이터베이스 리스토어

RestoreType 매개변수에 대해 SQLUD_FULL 값이 SQLUD_DB로 바뀌었습니다. 데이터베이스의 모든 테이블 스페이스를 리스토어합니다. 이는 오프라인으로 수행됩니다.

새 증분 리스토어 기능을 지원하기 위해, SQLUD_INCREMENTAL 매개변수도 추가되었습니다.

증분 백업 이미지는 가장 최근의 성공적인 전체 백업 이후에 변경된 모든 데이터베이스 데이터의 사본입니다.

AIX Extended Shared Memory Support(EXTSHM)에 관한 문서 오류

"부록 E 동시적인 액세스와 스레드된 응용프로그램", 참고 2가 다음과 같아야 합니다.

2. 기본값으로 AIX에서는 32비트 응용프로그램에 프로세스당 11개 이상의 공유 메모리 세그먼트를 연결할 수 없으며, 이 중 최대 10개의 공유 메모리 세그먼트를 로컬 DB2 연결에 사용할 수 있습니다.

DB2가 있는 EXTSHM을 사용하려면 다음을 수행하십시오.

클라이언트 세션에서

```
export EXTSHM=ON
```

DB2 서버를 시작할 때

```
export EXTSHM=ON
db2set DB2ENVLIST=EXTSHM
db2start
```

EEE에서 sqllib/db2profile에 다음 행을 추가하십시오.

```
EXTSHM=ON
export EXTSHM
```

SQLFUPD

locklist

토큰의 이름이 SQLF_DBTN_LOCKLIST에서 SQLF_DBTN_LOCK_LIST로 변경되었습니다. locklist 매개변수는 SMALLINT에서 64비트 부호없는 INTEGER로 변경되었습니다. 갱신 가능한 데이터베이스 구성 매개변수의 테이블에 다음을 추가해야 합니다.

매개변수 이름	토큰	토큰 값	데이터 유형
locklist	SQLF_DBTN_ LOCK_LIST	704	Uint64

이 매개변수에 대한 새 최대값은 524 288입니다.

또한『제 3 장 데이터 구조』, 표 53. Updatable Database Configuration Parameters에 dbheap에 대한 토큰 값이 701로 잘못 나와 있습니다. 올바른 값은 58입니다.

SQLEDBDESC

SQLDBCSS(sqlenv에 정의)에 유효한 값의 목록에 두 개의 값이 추가됩니다. 다음과 같습니다.

SQL_CS_SYSTEM_NLSCHAR
: 문자 유형에 대해 NLS 버전의 비교 루틴을 사용하여 시스템으로부터 순차를 조합.

SQL_CS_USER_NLSCHAR
: 문자 유형에 대해 NLS 버전의 비교 루틴을 사용하여 사용자로부터 순차를 조합.

응용프로그램 빌드 안내서

사용 가능한 갱신사항

응용프로그램 빌드 안내서는 FixPak 4에서 갱신되었습니다. 최신 PDF는 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

S/390 및 zSeries에서 Linux 지원

S/390 및 zSeries에서 Linux용 DB2는 다음과 같은 운영 체제 환경을 지원합니다.

- SuSE v7.0
- SuSE SLES 7
- TurboLinux v6.1

주: SuSE SLES 7에서 DB2 버전 7을 실행하려면, SuSE Linux Enterprise Server Developer's Edition CD 세트의 CD1에서 경로 ″CD1/suse/a1/compat.rpm″에 있는 libstdc++ v6.1 compat RPM을 설치해야 합니다. 이 작업은 루트로 수행해야 합니다. RPM을 설치하려면, CD 이미지를 디렉토리로 마운트하십시오. 예를 들어, /mnt 디렉토리로 마운트하려면, 다음 명령으로 RPM을 설치하십시오.

```
rpm -Uh /mnt/CD1/suse/a1/compat.rpm
```

그리고 나서 **ldconfig** 명령을 실행하십시오.

Linux Rexx 지원

Intel x86(32비트)의 Linux용 DB2는 Linux 버전 2.1에서 Object REXX Interpreter를 지원합니다.

S/390에서 Linux용 DB2는 Linux/390용 Object REXX 2.2.0을 지원합니다.

컴파일된 SQL 프로시저 분배에 대한 추가 참고사항

UNIX 시스템에서, 인스턴스 소유자(즉, DB2 엔진의 실행 대상 사용) 및 `$DB2PATH/adm/.fenced` 파일의 소유자가 동일한 1차 그룹에 속하는지 확인하십시오. 아니면, 이 두 사용자는 각각 상대방의 1차 그룹에 속해야 합니다.

GET ROUTINE 또는 PUT ROUTINE 조작(또는 해당 프로시저)이 올바르게 실행되지 않을 경우 항상 실패 원인에 대한 정보를 알려주는 진단 테스트와 함께 오류(SQLSTATE 38000)가 리턴됩니다. 예를 들어, GET ROUTINE을 위해 제공된 프로시저 이름이 SQL 프로시저를 나타내지 않을 경우 진단 ″100, 02000″ 텍스트가 리턴됩니다. 여기서, ″100″ 및 ″02000″은 각각 문제점의 원인을 나타내는 SQLCODE 및 SQLSTATE입니다. 이 예에서 SQLCODE 및 SQLSTATE는 해당 프로시저 이름에 대해 지정된 행이 카탈로그 테이블에 없음을 나타냅니다.

응용프로그램 개발 안내서

사용 가능한 갱신사항

응용프로그램 개발 안내서는 FixPak 4에서 갱신되었습니다. 최신 PDF는 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다. 또한 갱신된 참조에 이러한 참고 내용을 추가해 놓았습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

″Java로 작성된 스토어드 프로시저 디버깅″에 대한 수정사항

제 21장의 섹션 ″디버그 준비″에서 **db2dbugd** 명령을 사용하도록 지시되어 있습니다. 이는 잘못된 정보입니다. 대신, 다음 명령을 사용하십시오.

```
idebug -qdaemon -quiport=portno
```

디폴트 포트 번호는 8000입니다. idebug는 IBM Distributed Debugger의 클라이언트 측 디먼이고 VisualAge for Java 및 WebSphere Studio Application Developer와 함께 제공됩니다.

executeQuery 및 executeUpdate에 대한 새로운 요구사항

J2EE 1.3 표준을 준수하기 위해 FixPak 5 시점의 DB2 JDBC 드라이버는 executeQuery에 비쿼리문을 더 이상 사용할 수 없도록 하고 executeUpdate에 쿼리문을 사용할 수 없도록 하였습니다. 해당 구문을 사용하려 하면 다음 예외 중 하나가 발생하게 됩니다.

- CLI0637E QUERY가 없습니다.
- CLI0637E UPDATE가 없습니다.

명령문 유형을 알 수 없는 경우, execute()를 사용하십시오.

추가 메소드에 대한 JDBC 드라이버 지원

CallableStatement.getBlob() 및 CallableStatement.getClob():

이제 JDBC 드라이버는 CallableStatement.getBlob() 및 CallableStatement.getClob() 메소드를 지원합니다. DB2는 스토어드 프로시저 매개변수에서 LOB 로케이터를 지원하지 않으므로, CREATE PROCEDURE에 지정된 값인 LOB 데이터의 가능한 최대 크기를 수용할 수 있을 만큼 충분한 시스템 메모리가 사용 가능해야 합니다. 메모리가 부족할 경우 메모리 부족 예외가 발생합니다.

카탈로그되지 않은 저장 프로시저의 경우 이 지원을 사용할 수 없습니다.

Statement.setFetchSize(int rows) 및 ResultSet.setFetchSize(int rows):

이제 JDBC 드라이버는 Statement.setFetchSize(int rows) 및 ResultSet.setFetchSize(int rows)를 지원합니다. 이제 이 메소드를 사용하여 유형 3(또는 ″net″) 드라이버의 ResultSet 성능을 개선할 수 있습니다.

DB2 UDB용 IBM OLE DB Provider

DB2용 IBM OLE DB Provider의 사용에 대한 자세한 정보는 http://www.ibm.com/software/data/db2/udb/ad/v71/oledb.html을 참조하십시오.

CLI 안내서 및 참조서

Run-time Client를 사용한 데이터베이스 유틸리티 바인딩

Run-time Client는 데이터베이스 유틸리티(임포트, 익스포트, 재구성, 명령행 처리기)와 DB2 CLI 바인드 파일을 이 데이터베이스와 사용하기 전에 각 데이터베이스에 바인드하는 데 사용할 수 없습니다. 그 대신 DB2 Administration Client나 DB2 Application Development Client를 사용해야 합니다.

이들 데이터베이스 유틸리티와 DB2 CLI 바인드 파일을 해당 데이터베이스와 함께 사용하려면 각 데이터베이스에 바인드해야 합니다. 네트워크 환경에서 다른 운영 체제에서 수행되거나 DB2의 버전이나 서비스 레벨이 다른 여러 클라이언트를 사용하는 경우, 각 운영 체제와 DB2 버전 조합에 대해 한번씩 유틸리티를 바인드해야 합니다.

CLI 응용프로그램에서 정적 SQL 사용

CLI 응용프로그램에서의 정적 SQL 사용에 대한 자세한 내용은 http://www.ibm.com/software/data/db2/udb/staticcli/ 웹 사이트를 참조하십시오.

JDBC/ODBC/CLI 정적 프로파일의 한계

JDBC/ODBC/CLI 정적 프로파일은 현재 간단한 응용프로그램을 목표로 합니다. 여러 기능적 구성요소가 있는 복잡한 응용프로그램과 실행 중의 복잡한 프로그램 로직을 위한 것이 아닙니다.

SQL문을 프로파일 세션에서 캡처하려면 성공적으로 실행해야 합니다. 명령문 일치 세션에서 일치하지 않는 동적 명령문은 동적 JDBC/ODBC/CLI 호출로 계속 실행됩니다.

SQL문은 일치하는 명령문의 유효한 후보로 캡처 및 바인드된 것과 문자마다 동일해야 합니다. 공백 또한 의미를 가집니다. 예를 들어, "COL = 1"은 "COL=1"과 다르게 간주됩니다. 일치 확률을 높이려면 리터럴 대신에 매개변수 표시문자를 사용하십시오.

사전 바인드된 정적 SQL문과 함께 응용프로그램을 실행할 때, 동적 명령문 활동을 제어하는 동적 레지스터는 정적으로 변환되는 명령문에 대해 아무 영향을 주지 않습니다.

응용프로그램이 후속 DML 명령문에 참조된 오브젝트에 대해 DDL 명령문을 발행하면, 캡처 파일에서 이들 모든 명령문을 찾을 수 있습니다. JDBC/ODBC/CLI 정적 프로파일 바인드 도구는 이들을 바인드하려고 시도합니다. 바인드 시도는 VALIDATE(RUN) 바인드 옵션을 지원하는 DBMS를 사용하면 성공하지만 이를 지원하는 않는 것을 사용하면 실패합니다. 이 경우, 응용프로그램은 정적 프로파일을 사용해야 합니다.

데이터베이스 관리자는 응용프로그램 특정 요구사항에 따라 SQL문을 추가, 변경 또는 제거하여 캡처 파일을 편집할 수 있습니다.

ADT 변환

다음은 책의 기존 정보를 대체합니다.

- 새로운 설명자 유형(smallint) SQL_DESC_USER_DEFINED_TYPE_CODE가 다음 값과 함께 있습니다.

```
SQL_TYPE_BASE 0 (이것은 USER_DEFINED_TYPE이 아님)
SQL_TYPE_DISTINCT 1
SQL_TYPE_STRUCTURED 2
이 값은 SQLColAttribute 또는
  SQLGetDescField(IRD에만 해당)를 사용하여 쿼리할 수 있습니다.

다음 속성은 실제 유형 이름을 확보하기 위해 추가됩니다.
SQL_DESC_REFERENCE_TYPE
SQL_DESC_STRUCTURED_TYPE
SQL_DESC_USER_TYPE
위의 값은 SQLColAttribute 또는
  SQLGetDescField(IRD에만 해당)를 사용하여 쿼리할 수 있습니다.
```

- 응용프로그램에서 필요로 할 경우, SQL_DESC_BASE_TYPE을 추가하십시오. 예를 들어, 응용프로그램이 구조화 유형을 인식하지 못하지만 이를 페치하거나 삽입하기 위해 다른 코드가 세부사항을 처리하도록 할 수도 있습니다.
- SQL_ATTR_TRANSFORM_GROUP이라고 하는 새 연결 속성을 추가하여 응용프로그램이 변환 그룹을 설정할 수 있도록 하십시오(SQL ″SET CURRENT DEFAULT TRANSFORM GROUP″문을 사용하기 보다는).
- CLI가 값 SQL_DESC_USER_DEFINED_TYPE_CODE를 유효한 SQL 유형으로 리턴하도록 하는 SQLSetConnectAttr을 사용하여 설정하거나 쿼리할 수 있는 SQL_ATTR_RETURN_USER_DEFINED_TYPES라고 하는 새 명령문/연결 속성을 추가하십시오. 이 속성은 변환을 사용하기 전에 필요합니다.
  - 기본적으로, 그 속성은 해제되어 있으며 기본 유형 정보가 SQL 유형으로 리턴되도록 합니다.
  - 이 속성을 설정하면 SQL_DESC_USER_DEFINED_TYPE_CODE가 SQL_TYPE으로 리턴됩니다. 응용프로그램은 SQL_DESC_USER_DEFINED_TYPE_CODE에 대해 확인한 후 적절한 유형 이름을 검색할 것으로 예상됩니다. 이는 SQLColAttribute, SQLDescribeCol 및 SQLGetDescField에 대해 사용 가능하게 됩니다.
- SQLBindParameter는 SQL_C_DEFAULT를 바인드할 때 오류를 제공하지 않습니다. SQLBindParameter가 유형 SQL_USER_DEFINED_TYPE을 지정할 수 있도록 허용하는 코드가 없기 때문입니다. 서버에 유입되는 기본 SQL 유형을 근거로 표준 기본 C 유형이 사용됩니다. 예를 들어, 다음과 같습니다.

```
sqlrc = SQLBindParameter (hstmt, 2, SQL_PARAM_INPUT, SQL_C_CHAR,
                          SQL_VARCHAR, 30, 0, &c2, 30, NULL);
```

제 1 장 CLI 소개

DB2 CLI와 Embedded SQL 간의 차이점

″DB2 CLI 사용상의 이점″ 섹션의 목록 끝에서 세 번째 항목은 무시하십시오. 올바른 정보는 다음과 같습니다.

DB2 CLI는 DB2 Universal Database 서버, MVS/ESA 서버용 DB2(버전 5 이상) 또는 OS/400 서버(버전 5 이상)에 상주하는 저장 프로시저에서 생성된 다중

행과 결과 세트를 검색할 수 있는 기능을 제공합니다. OS/400에서 다중 결과 세트를 검색하기 위해서는 PTF(Program Temporary Fix) SI01761를 서버에 적용해야 합니다. 이 PTF가 적용되었는지 확인하려면 OS/400 시스템 관리자에게 문의하십시오.

제 3 장 고급 기능 사용

멀티스레드 응용프로그램 작성

"멀티스레드 혼합 응용프로그램" 섹션에 다음의 내용을 추가해야 합니다.

주: 기본 스택 크기를 사용하지 않고, 대신 스택 크기를 최소한 256 000으로 늘리는 것이 좋습니다. DB2에서는 DB2 함수를 호출할 때 최소 스택 크기는 256 000이어야 합니다. 따라서 스택의 총 크기를 할당할 때, 응용프로그램과 DB2 함수 호출을 위한 최소 요구사항을 충족시킬 수 있을 정도의 크기를 할당해야 합니다.

DB2 CLI 유니코드 응용프로그램 작성

다음은 이 장의 새로운 섹션입니다.

DB2 CLI 유니코드 응용프로그램 지원의 두 주요 영역이 있습니다.

1. ANSI 문자열 인수 대신에 유니코드 문자열 인수를 승인할 수 있는 함수 세트 추가.
2. 유티코드 데이터를 설명하는 새로운 C 및 SQL 데이터 유형 추가.

다음 섹션에는 이들 두 영역에 대한 추가 정보가 있습니다. 유니코드 응용프로그램으로 취급되려면, 연결하기 전에 응용프로그램이 SQL_ATTR_ANSI_APP 연결 속성을 SQL_AA_FALSE로 설정해야 합니다. 이와 같이 설정하면 CLI는 CLI와 데이터베이스 간의 기본 통신 방법으로서 유니코드를 사용합니다.

유니코드 함수

ODBC API 함수에는 해당 문자열 인수 형식을 나타내는 접미부가 붙어 있습니다. 유니코드를 승인하는 함수는 W로 끝나고 ANSI를 승인하는 함수에는 접미부가 없습니다.

주: ODBC는 A로 끝나는 이름을 가진 동등한 함수를 추가하지만 이는 DB2 CLI에는 사용되지 않습니다.

다음은 ANSI 및 유니코드 버전을 모두 가지는 DB2 CLI에서 사용할 수 있는 함수 목록입니다.

SQLBrowseConnect	SQLForeignKeys	SQLPrimaryKeys
SQLColAttribute	SQLGetConnectAttr	SQLProcedureColumns
SQLColAttributes	SQLGetConnectOption	SQLProcedures
SQLColumnPrivileges	SQLGetCursorName	SQLSetConnectAttr
SQLColumns	SQLGetDescField	SQLSetConnectOption
SQLConnect	SQLGetDescRec	SQLSetCursorName
SQLDataSources	SQLGetDiagField	SQLSetDescField
SQLDescribeCol	SQLGetDiagRec	SQLSetStmtAttr
SQLDriverConnect	SQLGetInfo	SQLSpecialColumns
SQLGetStmtAttr	SQLStatistics	SQLError
SQLNativeSQL	SQLTablePrivileges	SQLExecDirect
SQLPrepare	SQLTables	

항상 문자열 길이가 인수가 되는 유니코드 함수는 해당 인수를 문자 수로 해석합니다. 서버 데이터에 대한 길이 정보를 리턴하는 함수의 경우에도 표시 크기와 정밀도를 문자 수로 설명합니다. 길이(데이터의 전송 크기)가 문자열 또는 비문자열 데이터를 나타낼 수 있는 경우, 길이는 바이트 수로 해석됩니다. 예를 들어, SQLGetInfoW는 길이를 바이트 수로 취하지만, SQLExecDirectW는 문자수를 사용합니다. CLI는 응용프로그램의 바인딩에 따라 결과 세트를 유니코드 또는 ANSI로 리턴합니다. 응용프로그램이 SQL_C_CHAR에 바인드할 경우, 드라이버는 SQL_WCHAR 데이터를 SQL_CHAR로 변환합니다. ODBC 드라이버 관리자는 ANSI 드라이버용으로 SQL_C_WCHAR을 SQL_C_CHAR에 맵핑하지만 유니코드 드라이버용으로는 맵핑하지 않습니다.

새 데이터 유형과 유효한 변환

추가 ODBC 및 CLI 정의 데이터 유형이 유니코드 데이터베이스를 수용하기 위해 추가되었습니다. 이러한 유형은 기존에 존재하는 C 및 SQL 유형 세트를 대체합니다. 새로운 C 유형인 SQL_C_WCHAR은 C 버퍼에 UCS-2 데이터가 원시 endian 형식으로 있음을 의미합니다. 새로운 SQL 유형인 SQL_WCHAR, SQL_WVARCHAR 및 SQL_WLONGVARCHAR은 특정 열이나 매개변수 표시기에 유니코드 데이터가 들어 있음을 나타냅니다. DB2 유니코드 데이터베이스의 경우, 그래픽 컬럼은 새로운 유형을 사용하여 설명됩니다.

표 11. 지원되는 데이터 변환

SQL 데이터 유형	SQL_C_CHAR	SQL_C_WCHAR	SQL_C_LONG	SQL_C_SHORT	SQL_C_TINYINT	SQL_C_FLOAT	SQL_C_DOUBLE	SQL_C_TYPE_DATE	SQL_C_TYPE_TIME	SQL_C_TYPE_TIMESTAMP	SQL_C_BINARY	SQL_C_BIT	SQL_C_DBCHAR	SQL_C_CLOB_LOCATOR	SQL_C_BLOB_LOCATOR	SQL_C_DBCLOB_LOCATOR	SQL_C_BIGINT	SQL_C_NUMERIC
BLOB	X	X									D				X			
CHAR	D	X	X	X	X	X	X	X	X	X	X	X					X	X
CLOB	D	X									X			X				
DATE	X	X						D		X								
DBCLOB		X									X		D			X		
DECIMAL	D	X	X	X	X	X	X				X	X					X	X
DOUBLE	X	X	X	X	X	X	D					X					X	X
FLOAT	X	X	X	X	X	X	D					X					X	X
GRAPHIC (비유니코드)	X	X											D					
GRAPHIC (유니코드)	X	X	X	X	X	X	X	X	X	X	X	X	D				X	
INTEGER	X	X	D	X	X	X	X					X					X	X
LONG VARCHAR	D	X									X							
LONG VARGRAPHIC (비유니코드)	X	X									X		D					

표 11. 지원되는 데이터 변환 (계속)

SQL 데이터 유형	SQL_C_CHAR	SQL_C_WCHAR	SQL_C_LONG	SQL_C_SHORT	SQL_C_TINYINT	SQL_C_FLOAT	SQL_C_DOUBLE	SQL_C_TYPE_DATE	SQL_C_TYPE_TIME	SQL_C_TYPE_TIMESTAMP	SQL_C_BINARY	SQL_C_BIT	SQL_C_DBCHAR	SQL_C_CLOB_LOCATOR	SQL_C_BLOB_LOCATOR	SQL_C_DBCLOB_LOCATOR	SQL_C_BIGINT	SQL_C_NUMERIC
LONG VARGRAPHIC (유니코드)	X	X									X		D					
NUMERIC	D	X	X	X	X	X	X					X						X
REAL	X	X	X	X	X	D	X					X						X
SMALLINT	X	X	X	D	X	X	X					X					X	X
BIGINT	X	X	X	X	X	X	X				X	X					D	X
TIME	X	X							D	X								
TIMESTAMP	X	X						X	X	D								
VARCHAR	D	X	X	X	X	X	X	X	X	X	X	X					X	X
VARGRAPHIC (비유니코드)	X	X											D					
VARGRAPHIC (유니코드)	X	X	X	X	X	X	X	X	X	X	X	X	D				X	

주:

D 변환이 지원됩니다. SQL 데이터 유형에 대한 기본 변환입니다.

X 모든 IBM DBMS가 변환을 지원합니다.

공백 IBM DBMS가 변환을 지원하지 않습니다.

- 데이터가 LOB 위치 지정자로 변환되지 않고, 위치 지정자는 데이터 값을 나타냅니다. 추가 정보는 대형 오브젝트(LOB) 사용을 참조하십시오.
- SQL_C_NUMERIC은 32비트 Windows 운영 체제에서만 사용 가능합니다.

폐기된 키워드/패치 값

유니코드 응용프로그램이 지원되기 전에 1바이트 문자 데이터로 작업하도록 작성된 응용프로그램은은 GRAPHIC=1,2 또는 3, Patch2=7 등과 같이 일련의 `cli ini` 파일 키워드 시리즈에 의해 2바이트 그래픽 데이터로 작업하도록 만들 수 있습니다. 이와 같은 임시 조치에 따라 그래픽 데이터가 문자 데이터로 표시되며 또한 보고되는 데이터의 길이가 달라집니다.

이들 키워드는 유니코드 응용프로그램에 더 이상 필요하지 않으며, 실제로 심각한 부작용이 있으므로 사용해서는 안됩니다. 이를 알고 있지 않으면, 특정 응용프로그램이 유니코드 응용프로그램일 경우, 그래픽 데이터 처리에 영향을 주는 키워드 없이 시도하도록 권장합니다.

유니코드 데이터베이스의 리터럴

비유니코드 데이터베이스에서 LONG VARGRAPHIC 및 LONG VARCHAR 컬럼의 데이터는 비교될 수 없습니다. 내재된 코드 페이지 변환이 지원되지 않으므로 GRAPHIC/VARGRAPHIC 및 CHAR/VARCHAR 컬럼의 데이터는 명시적 유형변환(`cast`) 함수를 사용하여 서로 비교하거나 할당될 수만 있었습니다. 여기에는 GRAPHIC/VARGRAPHIC 리터럴이 CHAR/VARCHAR 리터럴과 G 접두부에 의해 차별화되는 GRAPHIC/VARGRAPHIC 및 CHAR/VARCHAR 리터럴이 포함됩니다.

유니코드 데이터베이스의 경우, GRAPHIC/VARGRAPHIC과 CHAR/VARCHAR 리터럴 사이의 유형변환(CAST)은 필요하지 않습니다. 또한, G 접두부가 GRAPHIC/VARGRAPHIC 리터럴 앞에 필요하지 않습니다. 최소한 한 인수가 리터럴이면, 내재된 변환이 발생합니다. 이는 G 접두부가 있거나 없는 리터럴을 SQLPrepareW() 또는 SQLExecDirect()를 사용하는 명령문 내에서 사용할 수 있도록 해줍니다. LONG VARGRAPHIC의 리터럴은 G 접두부를 가져야합니다.

자세한 정보는 *SQL 참조서*의 "제 3 장 언어 요소"에서 "데이터 유형간 유형변환"을 참조하십시오.

새로운 CLI 구성 키워드

다음 세 개의 키워드를 추가하여 유니코드 응용프로그램이 데이테이스에 연결될 때 중복 오버헤드를 피할 수 있도록 하였습니다.

1. **DisableUnicode**

   키워드 설명

   유니코드에 대한 기본 지원 사용안함

   db2cli.ini 키워드 구문

   DisableUnicode = **0** | **1**

   기본 설정

   0(false)

   DB2 CLI/ODBC 설정 탭

   이 키워드는 CLI/ODBC 설정 노트북을 사용하여 설정할 수 없습니다. 이 키워드를 사용하려면 `db2cli.ini` 파일을 직접 수정해야 합니다.

   사용 주의사항:

   유니코드 지원이 사용 가능하고, 유니코드 응용프로그램이 호출한 경우, 코드 페이지 변환에 의한 불필요한 자료 손실이 없도록 보장하기 위하여 CLI는 최선의 클라이언트 코드 페이지를 사용하여 데이터베이스에 연결하려고 시도합니다. 이는 코드 페이지가 교환되므로 연결 시간을 증가시키거나 또는 이러한 지원이 추가되기 이전에는 발생하지 않았던 클라이언트에서의 코드 페이지 변환을 초래할 수 있습니다.

   이 키워드를 True(1)로 설정하면, 데이터가 서버에 송신되기 이전에 먼저 모든 유니코드 데이터가 응용프로그램의 로컬 코드 페이지로 변환됩니다. 이는 로컬 코드 페이지에서 대표할 수 없는 데이터의 데이터 손실을 초래할 수 있습니다.

2. **ConnectCodepage**

   키워드 설명

   중복 연결 오버헤드를 피하기 위해 데이터 소스에 연결하는 경우, 사용할 구체적 코드 페이지를 지정합니다.

db2cli.ini 키워드 구문

ConnectCodepage = **0** | **1** | <any valid db2 code page>

기본 설정

0

DB2 CLI/ODBC 설정 탭

이 키워드는 CLI/ODBC 설정 노트북을 사용하여 설정할 수 없습니다. 이 키워드를 사용하려면 db2cli.ini 파일을 직접 수정해야 합니다.

사용 주의사항:

비유니코드 응용프로그램은 언제나 응용프로그램의 로컬 코드페이지 또는 DB2Codepage 환경설정을 사용하여 데이터베이스에 연결합니다. 기본적으로 CLI는 유니코드 응용프로그램이 UTF-8 및 UCS-2 코드 페이지를 사용하여 유니코드 데이터베이스에 연결되도록 합니다. 데이터베이스 서버가 Windows용 DB2, Unix용 DB2 또는 OS/2용 DB2를 실행 중일 경우, 비유니코드 데이터베이스에 연결하기 위한 디폴트값은 데이터베이스의 코드 페이지를 사용하는 것입니다. 그러면, 코드 페이지 변환으로 인해 불필요한 데이터가 손실되지 않습니다.

연결시에 중복 오버헤드를 피하기 위하여 비유니코드 데이터베이스에 연결하는 경우, 사용자는 이 키워드를 사용하여 데이터베이스의 코드 페이지를 지정할 수 있습니다.

1 값을 지정하면 SQLDriverConnect()가 출력 연결 문자열에 올바른 값을 리턴하게 되므로, 이후 SQLDriverConnect() 호출에 값을 사용할 수 있습니다.

3. **유니코드 서버**

키워드 설명

데이터 소스가 유니코드 서버임을 표시. 설정 ConnectCodepage=1208과 동일.

db2cli.ini 키워드 구문

UnicodeServer = **0** | **1**

기본 설정

0

DB2 CLI/ODBC 설정 탭

이 키워드는 CLI/ODBC 설정 노트북을 사용하여 설정할 수 없습니다. 이 키워드를 사용하려면 `db2cli.ini` 파일을 직접 수정해야 합니다.

사용 주의사항:

이 키워드는 ConnectCodepage=1208과 동일하며 편리성을 위해 추가하였습니다. OS/390 버전 7 이상의 DB2에 연결하는 경우, 이 키워드를 사용하여 중복 연결 오버헤드를 피할 수 있습니다. 중복 처리가 필요하지 않으므로, 이 키워드를 Windows의 DB2, Unix의 DB2 또는 OS/2 데이터베이스의 DB2에 사용할 필요는 없습니다.

트랜잭션 모니터로 MTS(Microsoft Transaction Server) 사용

다음은 ″설치 및 구성″ 서브섹션의 DISABLEMULTITHREAD 구성 키워드 기본값을 정정합니다.

- DISABLEMULTITHREAD 키워드(기본값 0)

화면 이동 커서

다음 정보는 ″화면 이동 커서″ 섹션에 추가되어야 합니다.

OS/390용 서버측 화면 이동 커서 지원

Unix, Windows 및 OS/2 플랫폼용 UDB 클라이언트는 OS/390 버전 7 데이터베이스에 대해 수행할 때 갱신 가능한 서버측 화면 이동 커서를 지원합니다. 3계층(three-tier) 환경에서 OS/390 화면 이동 커서를 액세스하려면, 클라이언트 및 게이트웨이가 DB2 UDB 버전 7.1, FixPak 3 이상을 실행 중이어야 합니다.

화면 이동 커서를 액세스할 수 있는 2개의 응용프로그램 사용 인터페이스(ODBC 및 JDBC)가 있습니다. JDBC 인터페이스는 정적 화면 이동 커서만을 액세스할 수 있지만, ODBC 인터페이스는 정적 및 keyset 위주(keyset-driven) 서버측 화면 이동 커서를 액세스할 수 있습니다.

커서 속성: 아래 테이블은 ODBC에서 OS/390 버전 7 커서에 대한 기본 속성을 나열합니다.

표 12. ODBC에서 OS/390 커서에 대한 기본 속성

커서 유형	커서 민감성	커서 갱신가능	커서 동시성	커서 화면 이동
포워드 전용[a]	미지정	갱신가능하지 않음	읽기 전용 동시성	화면 이동 가능하지 않음
정적	민감하지 않음	갱신가능하지 않음	읽기 전용 동시성	화면 이동
keyset 위주 (keyset-driven)	민감함	갱신가능	값 동시성	화면 이동

a 포워드 전용은 FOR UPDATE 절이 없는 화면 이동 커서의 기본 활동입니다. 포워드 전용 커서에서 FOR UPDATE를 지정하면 갱신 가능한, 잠금 동시성, 화면 이동 가능하지 않은 커서를 작성합니다.

지원되는 페치 방향: 모든 ODBC 페치 방향은 SQLFetchScroll 또는 SQLExtendedFetch 인터페이스를 통해 지원됩니다.

keyset 위주(keyset-driven) 커서 갱신: keyset 위주(keyset-driven) 커서는 갱신 가능한 커서입니다. CLI 드라이버는 쿼리가 SELECT ... FOR READ ONLY 쿼리로서 발행되거나 FOR UPDATE 절이 이미 존재하는 경우를 제외하고 FOR UPDATE 절을 쿼리에 추가합니다. OS/390용 DB2에서 구현되는 keyset 위주(keyset-driven) 커서는 값 동시성 커서입니다. 값 동시성 커서는 최적 잠금의 결과를 가져오며, 여기서, 값은 갱신이나 삭제가 시도될 때까지 보유되지 않습니다. 갱신 또는 삭제가 시도되면, 데이터베이스 서버는 응용프로그램이 검색한 이전 값과 주요 테이블의 현재 값을 비교합니다. 값이 일치하는 경우, 갱신 또는 삭제가 성공합니다. 값이 일치하지 않는 경우, 조작이 실패합니다. 실패한 경우, 응용프로그램은 다시 값을 쿼리하고 여전히 적용가능하다면 갱신 또는 삭제를 다시 발행해야 합니다.

응용프로그램은 두 가지 방식으로 keyset 위주(keyset-driven) 커서를 갱신할 수 있습니다.

- SQLExecute() 또는 SQLExecDirect()가 있는 SQLPrepare()를 사용하여 UPDATE WHERE CURRENT OF "<cursor name>" 또는 DELETE WHERE CURRENT OF "<cursor name>"을 발행하십시오.
- SQLSetPos() 또는 SQLBulkOperations()를 사용하여 결과세트에 행을 추가, 갱신 또는 삭제하십시오.

주: `SQLSetPos()` 또는 `SQLBulkOperations()`를 통해 결과 세트에 추가된 행은 서버에서 테이블로 삽입되지만, 서버의 결과 세트에 추가되지는 않습니다. 그러므로, 이러한 행은 갱신가능하지 않으며 다른 트랜잭션이 작성한 변경사항에 민감하지도 않습니다. 그러나, 삽입된 행은 클라이언트에 캐시되므로 결과 세트의 일부로 나타납니다. 삽입된 행에 적용되는 모든 트리거는 적용되지 않았던 것처럼 응용프로그램에 나타납니다. 삽입된 행을 갱신가능하고, 민감하게 하고, 적용 가능한 트리거의 결과를 보려면, 응용프로그램이 쿼리를 다시 발행하여 결과 세트를 재생성해야 합니다.

화면 이동 커서 지원 이전에 작성된 응용프로그램의 문제점 해결: 화면 이동 커서 지원은 새로운 기능이므로, OS/390용 UDB 또는 Unix, Windows 및 OS/2용 UDB는 행동 또는 성능 변경이 발생할 수 있습니다. 화면 이동 커서가 지원되기 전에는, 화면 이동 커서를 요청한 응용프로그램이 포워드 전용 커서를 받기 때문에 이런 경우가 발생합니다. 화면 이동 커서 지원하기 전의 응용프로그램의 이전 활동을 리스토어하려면, 다음과 같은 구성 키워드를 `db2cli.ini` 파일에 설정하십시오.

표 13. 화면 이동 커서가 지원하기 전에 응용프로그램 활동을 리스토어하는 구성 키워드 값

구성 키워드 설정	설명
`PATCH2=6`	화면 이동 커서(keyset 위주(keyset-driven) 및 정적 모두)가 지원되지 않음을 나타내는 메시지를 리턴합니다. CLI는 자동으로 화면 이동 커서에 대한 모든 요청을 포워드 전용 커서로 낮춥니다.
`DisableKeysetCursor=1`	서버측 및 클라이언트측 keyset 위주(keyset-driven) 화면 이동 커서 모두를 사용하지 않습니다. 이것은 keyset 위주(keyset-driven) 커서가 요청될 때 CLI 드라이버가 응용프로그램에 정적 커서를 부여하도록 강제하는 데 사용될 수 있습니다.
`UseServerKeysetCursor=0`	클라이언트측 keyset 위주(keyset-driven) 커서 라이브러리를 사용 중인 응용프로그램의 서버측 keyset 위주(keyset-driven) 커서를 사용하지 않음으로써 keyset 위주(keyset-driven) 커서를 시뮬레이트합니다. 클라이언트측 커서가 많은 양의 오버헤드를 발생시키고 일반적으로 서버특 커서보다 성능이 나쁘므로 서버측 keyset 위주(keyset-driven) 커서에서 문제점이 발생할 때만 이 옵션을 사용하십시오.

복합 SQL 사용

다음의 주의사항이 책에서 누락되었습니다.

```
Any SQL statement that can be prepared dynamically, other than a query,
can be executed as a statement inside a compound statement.
```

```
Note: Inside Atomic Compound SQL, savepoint, release savepoint, and
rollback to savepoint SQL statements are also disallowed. Conversely,
Atomic Compound SQL is disallowed in savepoint.
```

저장 프로시저 사용

SQL 및 Java 저장 프로시저 빌드 및 디버그를 위한 특권

SQL 저장 프로시저를 빌드, 디버그, 실행하려는 사용자에게 다음 특권을 부여해야 합니다.

- `db2 grant CONNECT on database to userid`
- `db2 grant IMPLICIT_SCHEMA on database to userid`
- `db2 grant BINDADD on database to userid`
- `db2 grant SELECT on SYSIBM.SYSDUMMY1 to userid`
- `db2 grant SELECT on SYSCAT.PROCEDURES to userid`
- `db2 grant UPDATE on DB2DBG.ROUTINE_DEBUG to userid`

Java 저장 프로시저를 빌드, 디버그, 실행하려는 사용자에게 다음 특권을 부여해야 합니다.

- `db2 grant CONNECT on database to userid`
- `db2 grant IMPLICIT_SCHEMA on database to userid`
- `db2 grant BINDADD on database to userid`(SQLJ를 사용하여 정적 SQL로 Java 저장 프로시저를 빌드하는 경우에만 필요함)
- `db2 grant SELECT on SYSIBM.SYSDUMMY1 to userid`
- `db2 grant SELECT on SYSCAT.PROCEDURES to userid`
- `db2 grant UPDATE on DB2DBG.ROUTINE_DEBUG to userid`

DB2DBG.ROUTINE_DEBUG 테이블을 작성하려면 다음 명령을 실행하십시오.

```
db2 -tf sqllib/misc/db2debug.ddl
```

| Java 저장 프로시저 디버그에 대한 자세한 정보는 *응용프로그램 개발 안내서*를 참
| 조하십시오.

CLI로 저장 프로시저 작성

다음은 CLI 저장 프로시저에 대해 문서화되지 않은 제한사항입니다.

```
If you are making calls to multiple CLI stored procedures,
the application must close the open cursors from one stored procedure
before calling the next stored procedure. More specifically, the first
set of open cursors must be closed before the next stored procedure
tries to open a cursor.
```

CLI 저장 프로시저 및 자동 바인드

다음은 이 책의 정보를 보충하는 내용입니다.

CLI/ODBC 드라이버는 사용자가 적합한 특권 또는 권한을 갖고 있는 경우, 보통 CLI/ODBC 응용프로그램이 데이터베이스에 대해 SQL을 처음 실행할 때 CLI 패키지를 자동으로 바인드합니다. CLI 패키지의 자동 바인드는 저장 프로시저내에서 수행될 수 없으므로, 응용프로그램이 수행하는 제일 처음 동작이 CLI 저장 프로시저를 호출하는 것인 경우 발생하지 않습니다. 그러므로, 새 데이터베이스에 대해 CLI 저장 프로시저를 호출하는 CLI 응용프로그램을 실행하기 전에, 이 명령으로 CLI 패키지를 한번 바인드해야 합니다.

UNIX

```
db2 bind <BNDPATH>/@db2cli.lst blocking all
```

Windows 및 OS/2

```
db2bind "%DB2PATH%\bnd\@db2cli.lst" blocking
```

권장하는 접근 방법은 런타임시 자동 바인드를 예방하기 위해 데이터베이스 작성시 이들 패키지를 항상 바인드하는 것입니다. 사용자에게 특권이 없거나 다른 응용프로그램이 동시에 자동 바인드를 시도하는 경우 자동 바인드가 실패할 수 있습니다.

제 4 장 CLI/ODBC 구성 및 샘플 응용프로그램 수행

구성 키워드

CURRENTFUNCTIONPATH

CURRENTFUNCTIONPATH 키워드에서 마지막 단락을 무시하십시오. 올바른 정보는 다음과 같습니다.

이 키워드는 현재 사용자의 스키마 이외의 스키마 이름으로 정의될 수 있었던 저장 프로시저 및 비규정 함수를 분석하는 프로세스의 일부로서 사용됩니다. 스키마 이름의 순서는 함수 및 프로시저 이름이 분석될 순서를 결정합니다. 함수 및 프로시저 분석에 대한 자세한 정보는 SQL 참조서를 참조하십시오.

SKIPTRACE

이 새로운 구성 키워드에 대한 설명은 다음과 같습니다.

키워드 설명

CLI 응용프로그램을 추적 기능에서 제외시킵니다.

db2cli.ini 키워드 구문

SKIPTRACE = **0** | **1**

기본 설정

추적 기능을 건너뛰지 마십시오.

DB2 CLI/ODBC 설정 탭

이 키워드는 CLI/ODBC 설정 노트북을 사용하여 설정할 수 없습니다. 이 키워드를 사용하려면 db2cli.ini 파일을 직접 수정해야 합니다.

사용 주의사항:

이 키워드는 추적 기능에서 CLI 응용프로그램을 제외하여 성능을 개선시킬 수 있습니다. 그러므로 DB2 추적 기능 db2trc이 설정되고 이 키워드가 1로 설정되면, 해당 추적은 CLI 응용프로그램의 실행에서 정보를 포함하지 않습니다.

추적 정보가 필요하지 않은 UNIX 플랫폼의 제품 환경에 대해서는 SKIPTRACE 설정이 권장됩니다. 추적 출력을 가져서 테스트 환경은 개선되지만, 실행 정보가 필요할 때 이 키워드를 해제(또는 기본 설정으로 남음)할 수 있습니다.

SKIPTRACE는 db2cli.ini 구성 파일의 [COMMON] 섹션에서 설정되어야 합니다.

제 5 장 DB2 CLI 함수

SQLBindFileToParam - LOB 파일 참조를 LOB 매개변수에 바인드

SQLBindFileToParam() CLI 함수의 마지막 매개변수인 IndicatorValue는 현재 ″output(deferred)″으로 설명됩니다. 이것은 ″input(deferred)″이어야 합니다.

SQLColAttribute - 컬럼 속성 리턴

다음 갱신사항을 SQL_DESC_AUTO_UNIQUE_VALUE 및 SQL_DESC_UPDATABLE 인수의 ″설명″ 컬럼에 추가합니다.

SQL_DESC_AUTO_UNIQUE_VALUE

SQL_FALSE는 모든 DB2 SQL 데이터 유형에 대해 *NumericAttributePtr* 에 리턴됩니다. 컬럼이 식별 컬럼인 경우에는 현재 DB2 CLI를 사용할 수는 없으나, SQL_FALSE는 항상 리턴됩니다. 이 제한사항은 ODBC 권장 스펙과 일치하지 않습니다. 차기 Unix 및 Windows 서버용 DB2 CLI 버전은 auto-unique 값을 지원하게 될 것입니다.

SQL_DESC_UPDATABLE

컬럼 데이터 유형이 갱신 가능한 데이터 유형이면 다음을 표시합니다.

- SQL_ATTR_READWRITE_UNKNOWN은 모든 DB2 SQL 데이터 유형에 대해 *NumericAttributePtr*에 리턴됩니다. 이는 컬럼이 갱신 가능할 경우에는 DB2 CLI를 현재 판별할 수 없으므로 리턴됩니다. Unix 및 Windows 서버용의 차기 DB2 CLI에서는 컬럼을 갱신할 수 있는지 여부를 확인할 수 있는 기능이 제공될 것입니다.

SQLGetData - 컬럼에서 데이터 가져오기

다음 텍스트는 SQLGetData에 대한 SQLSTATE 테이블의 SQLSTATE 22007 의 설명 컬럼 아래 있는 현재 문장을 대체합니다.

문자열에서 날짜 시간 형식으로의 변환이 지정되었지만, 올바르지 않은 문자열 표현 또는 값이 지정되거나 값이 올바르지 않은 날짜입니다.

SQLGetInfo - 일반 정보 가져오기

다음은 ″Information Returned by SQLGetInfo″의 ″Usage″ 섹션 정보를 정정합니다.

- *InfoType* SQL_CURSOR_CLOSE_BEHAVIOR는 SQL_CLOSE_BEHAVIOR가 되어야 합니다.
- SQL_DATABASE_NAME(문자열)에 대한 참고사항은 다음과 같습니다.

  주: 이 문자열은 호스트가 아닌 시스템에서 SELECT CURRENT SERVER 명령문이 리턴한 문자열과 동일합니다. OS/390용 DB2나 OS/400용 DB2 등의 호스트 데이터베이스의 경우 리턴된 문자열이 DB2 Connect 게이트웨이에서 CATALOG DCS DATABASE DIRECTORY 명령을 발행할 때 제공되는 DCS 데이터베이스 이름입니다.

SQLGetLength - 문자열 값의 길이 검색

다음은 ″테이블 113. SQLGetLength 인수″의 주석을 정정합니다.

참고: **a** 이는 DBCLOB 데이터용 문자입니다.

SQLNextResult - 다음 결과 세트를 다른 명령문 핸들과 연결

다음 텍스트가 제 5 장 『DB2 CLI 함수』에 추가되어야 합니다.

목적

스펙: DB2 CLI 7.x

구문

```
SQLRETURN   SQLNextResult (SQLHSTMT StatementHandle1
                           SQLHSTMT StatementHandle2);
```

함수 인수

표 14. SQLNextResult 인수

데이터 유형	인수	사용	설명
SQLHSTMT	*StatementHandle*	입력	명령문 핸들.
SQLHSTMT	*StatementHandle*	입력	명령문 핸들.

사용법

저장 프로시저는 나간 후에 하나 또는 그 이상의 커서를 열어 두어 여러 개의 결과 세트를 리턴합니다. 첫 번째 결과 세트는 저장 프로시저를 호출했던 명령문 핸들을 사용하여 항상 액세스됩니다. 여러 개의 결과 세트가 리턴될 경우, SQLMoreResults() 또는 SQLNextResult()를 사용하여 결과 세트를 설명하고 가져올 수 있습니다.

SQLMoreResults()를 사용하여 첫 번째 결과 세트를 닫고 다음 결과 세트가 처리되도록 하는 반면, SQLNextResult()는 *StatementHandle1*의 커서를 닫지 않고 다음 결과 세트를 *StatementHandle2*로 이동합니다. 가져올 결과 세트가 없으면 두 기능은 SQL_NO_DATA_FOUND를 리턴합니다.

SQLNextResult()를 사용하면, 결과 세트들이 한번 다른 명령문 핸들로 전송되면 어떤 순서로든 처리되도록 합니다. SQLMoreResults() 및 SQLNextResult()에 대한 혼합된 호출은 *StatementHandle1*에 더 이상의 커서(열린 결과 세트)가 없을 때까지 허용됩니다.

SQLNextResult()가 SQL_SUCCESS를 리턴하면, 다음 결과 세트는 *StatementHandle1*과 더 이상 연결되지 않습니다. 대신, SQLExecDirect() 호출이 *StatementHandle2*에 대한 쿼리에서 방금 성공적으로 실행된 것 처럼 다음 결과 세트가 *StatementHandle2*와 연결됩니다. 따라서, 커서는 SQLNumResultSets(), SQLDescribeCol() 또는 SQLColAttribute()를 사용하여 설명될 수 있습니다.

SQLNextResult()가 호출된 후, 지금 *StatementHandle2*과 연결된 결과 세트는 남은 결과 세트 체인에서 제거되고 SQLNextResult() 또는 SQLMoreResults()에서 다시 사용될 수 없습니다. 이는 'n' 결과 세트, SQLNextResult()가 최대 'n-1' 횟수만큼 성공적으로 호출될 수 있음을 의미합니다.

SQLFreeStmt()가 SQL_CLOSE 옵션과 함께 호출되거나, SQLFreeHandle()이 *HandleType*가 SQL_HANDLE_STMT로 설정되어 호출될 경우, 이 명령문 핸들에 대한 모든 보류 중인 결과 세트가 버려집니다.

SQLNextResult()는 *StatementHandle2*에 열린 커서가 있거나 *StatementHandle1* 및 *StatementHandle2*가 동일한 연결 상에 있지 않을 경우 SQL_ERROR를 리턴합니다. 오류나 경고가 리턴되면, SQLError()가 *StatementHandle1*에서 항상 호출되어야 합니다.

주: SQLMoreResults()는 또한 입력 매개변수 값이 SQLParamOptions() 및 SQLBindParameter()로 지정된 배열이 있는 매개변수화된 쿼리와 함께 작업합니다. 하지만, SQLNextResult()는 이를 지원하지 않습니다.

리턴 코드

- SQL_SUCCESS
- SQL_SUCCESS_WITH_INFO
- SQL_STILL_EXECUTING
- SQL_ERROR
- SQL_INVALID_HANDLE
- SQL_NO_DATA_FOUND

진단

표 15. SQLNextResult SQLSTATE

SQLSTATE	설명	자세한 설명
40003 **08**S01	통신 링크 실패.	응용프로그램과 데이터 소스 사이의 통신 링크가 함수가 완료되기 전에 실패했습니다.
58004	예상하지 못한 시스템 실패.	복구할 수 없는 시스템 오류입니다.
HY001	메모리 할당 실패.	DB2 CLI가 함수 실행 또는 완료 지원에 필요한 메모리를 할당할 수 없습니다.

표 15. *SQLNextResult SQLSTATE (계속)*

SQLSTATE	설명	자세한 설명
HY010	함수 순차 오류.	data-at-execute (SQLParamData(), SQLPutData()) 조작 중에 함수가 호출되었습니다. *StatementHandle2*에 그와 연결된 열린 커서가 있습니다. BEGIN COMPOUND 및 END COMPOUND SQL 작업 내에 있는 동안 함수가 호출되었습니다.
HY013	예상하지 못한 메모리 처리 오류.	DB2 CLI가 함수 실행 또는 완료 지원에 필요한 메모리에 액세스할 수 없습니다.
HYT00	시간종료 만기.	데이터 소스가 결과 세트를 리턴하기 전에 시간종료 기간이 만기되었습니다. 시간종료는 Windows 3.1 및 Macintosh System 7과 같은 비멀티태스크 시스템에서만 지원됩니다. 시간종료 기간은 SQLSetConnectAttr()에 대한 SQL_ATTR_QUERY_TIMEOUT 속성을 사용하여 설정할 수 있습니다.

제한사항

SQLMoreResults()만 매개변수화된 쿼리에 대해 사용할 수 있습니다.

참조

- 535 페이지의 "SQLMoreResults - 추가 결과 세트가 있는지 판단"
- 120 페이지의 "저장 프로시저로부터 결과 세트 리턴"

SQLSetEnvAttr - 환경 속성 설정

다음은 "사용" 아래에 있는 "환경 속성" 섹션에 속하는 추가적인 환경 속성입니다.

SQL_ATTR_KEEPCTX

환경 핸들이 해제될 때 컨텍스트가 유지되어야 할 지 여부를 지정하는 32 비트 정수 값. 이 속성은 환경 레벨에서 설정되어야 합니다. 각 스레드의 연결, 데이터베이스 자원 및 데이터 전송과 연관된 컨텍스트를 관리하기 위해 멀티스레드 응용프로그램에서 사용될 수 있습니다. 가능한 값은 다음과 같습니다.

- **SQL_FALSE:** 스레드의 환경 핸들이 해제되면 응용프로그램은 컨텍스트를 릴리스합니다. 이것은 기본값입니다.

- SQL_TRUE: 스레드의 환경 핸들이 해제되면 컨텍스트가 유효하게 남으며, 동일한 연결에서 기존의 다른 스레드에 대한 컨텍스트를 사용 가능하게 합니다. SQL_ATTR_KEEPCTX를 SQL_TRUE로 설정하면 멀티스레드 응용프로그램에서 컨텍스트 충돌과 연관된 일부 문제점을 해결할 수 있습니다.

주: 이것은 IBM 확장 기능입니다.

SQLSetStmtAttr - 명령문 관련 옵션 설정

다음은 SQL_ATTR_QUERY_TIMEOUT 명령문 속성에 맞게 기존 정보를 바꿉니다.

SQL_ATTR_QUERY_TIMEOUT(DB2 CLI v2)

응용프로그램에 리턴하는 동안 실행한 SQL문을 기다리는 초 단위의 32 비트 정수 값입니다. 이 옵션을 설정해서 장기간 실행되는 쿼리를 종료할 수 있습니다. 0 값은 시간종료를 하지 않는다는 의미입니다. DB2 CLI는 멀티스레드를 지원하는 모든 플랫폼에 0이 아닌 값을 지원합니다.

부록 C DB2 CLI 및 ODBC

다음은 이 부록에 추가된 새 섹션입니다.

ODBC 유니코드 응용프로그램

유니코드 ODBC 응용프로그램은 주로 UCS-2로 문자 데이터를 전송하고 검색합니다. ODBC 함수(‘W’ 접미부)의 유니코드 버전을 호출하여 그리고 유니코드 데이터 유형을 지시하여 이를 수행합니다. 응용프로그램은 명시적으로 로컬 코드 페이지를 지정하지 않습니다. 응용프로그램은 계속 ANSI 함수를 호출하고 로컬 코드 페이지 문자열을 전달합니다.

예를 들어, 응용프로그램은 SQLConnectW()를 호출하고 DSN 사용자 ID와 암호를 유니코드 인수로 전달합니다. 그런 뒤 SQLExecDirectW()를 호출하고 유니코드 SQL문에 전달한 뒤, ANSI 로컬 코드 페이지 버퍼(SQL_C_CHAR)와 유니코드 버퍼(SQL_C_WCHAR)의 조합을 바인드합니다. 데이터베이스 데이터 유형은 유니코드일 수도 있고 유니코드가 아닐 수도 있습니다.

CLI 응용프로그램이 SQL_ATTR_ANSI_APP를 SQL_AA_FALSE로 설정하여 SQLConnectW를 호출하거나 SQL_ATTR_ANSI_APP 값을 설정하지 않고 SQLSetConnectAttr을 호출할 경우, 응용프로그램은 유니코드 응용프로그램으로 간주됩니다. 즉, 모든 CHAR 데이터가 UTF-8 형식으로 데이터베이스로 송수신됩니다. 그런 뒤 응용프로그램은 CHAR 데이터를 로컬 코드 페이지의 SQL_C_CHAR 버퍼로(데이터 유실 가능), 데이터 유실없이 UCS-2의 SQL_C_WCHAR 버퍼로 가져옵니다.

응용프로그램이 위 두 호출 중 어느 것도 수행하지 않으면, CHAR 데이터는 서버에서응용프로그램 로컬 코드 페이지로 변환됩니다. 이는 SQL_C_WCHAR로 가져온 CHAR 데이터에 데이터 유실이 있을 수 있음을 의미합니다.

DB2CODEPAGE 인스턴스 변수가 코드 페이지 1208(UTF-8)로 설정되어 있으면 (db2set를 사용하여), 응용프로그램은 UTF-8이 이제 로컬 코드 페이지이므로 모든 CHAR 데이터를 UTF-8로 수신합니다. 응용프로그램은 또한 모든 CHAR 입력 데이터 역시 UTF-8인지 확인해야 합니다. 또한 ODBC는 모든 SQL_C_WCHAR 데이터가 원시 endian 형식이라고 가정합니다. CLI는 SQL_C_WCHAR에 필요한 바이트 리버설을 수행합니다.

ODBC 유니코드 대 비유니코드 응용프로그램

이 DB2 Universal Database 릴리스에는 SQLConnectW() API가 포함되어 있습니다. 유니코드 드라이버는 드라이버 관리자가 유니코드 드라이버로 인식하는 순서대로 SQLConnectW를 내보내야 합니다. 많은 응용프로그램(Microsoft Access 및 Visual Basic과 같이)이 SQLConnectW()를 호출함을 유의해야 합니다. 이전 릴리스의 DB2 Universal Database에서는 DB2 CLI가 이 API를 지원하지 않았고, 따라서 ODBC 드라이버에 의해 유니코드 드라이버로 인식될 수 없었습니다. 이는 ODBC 드라이버 관리자가 모든 유니코드 데이터를 응용프로그램의 로컬 코드 페이지로 변환하도록 만들었습니다. SQLConnectW() 함수의 지원이 추가되어, 이들 응용프로그램이 이제 유니코드 응용프로그램으로 연결되고 DB2 CLI가 모든 필요한 데이터 변환을 관장합니다.

DB2 CLI는 이제 유니코드 API(접미부 ″W″) 및 정규 ANSI API를 승인합니다. ODBC는 접미부 ″A″로 함수 세트를 정의하지만, 드라이버 관리자는 접미부가

″A″인 ANSI 함수를 드라이버로 전달하지 않습니다. 대신, 이 함수들을 접미부가 없는 ANSI 함수 호출로 변환한 뒤, 그것을 드라이버로 전달합니다.

SQLConnectW() API를 호출하는 ODBC 응용프로그램을 유니코드 응용프로그램이라고 합니다. ODBC 드라이버 관리자는 호출된 응용프로그램 버전에 상관없이 항상 SQLConnectW() API를 호출하므로, ODBC는 응용프로그램이 ANSI 또는 UNICODE 응용프로그램으로 취급되어야 할 경우 드라이버에게 알리기 위해 SQL_ATTR_ANSI_APP 연결 속성을 도입했습니다. SQL_ATTR_ANSI_APP가 SQL_AA_TRUE로 설정되어 있지 않으면, DB2 CLI는 모든 유니코드 데이터를 서버로 보내기 전에 로컬 코드 페이지로 변환합니다.

부록 D Extended Scalar Functions

날짜 및 시간 함수

다음 함수는 부록 D ″Extended Scalar Functions″의 날짜 및 시간 함수 섹션에서 누락된 것입니다.

DAYOFWEEK_ISO(*date_exp* **)**

*date_exp*에 있는 요일을 1-7 범위내의 정수값으로 리턴합니다. 여기서, 1은 월요일을 나타냅니다. 이 함수와 DAYOFWEEK() 함수의 차이점을 기억하십시오. DAYOFWEEK() 함수에서는 1이 일요일을 나타냅니다.

WEEK_ISO(*date_exp* **)**

*date_exp*에 있는 일년중 주수를 1-53 범위내의 정수값으로 리턴하십시오. 1주는 목요일이 들어가는 연도의 첫 번째 주로서 정의됩니다. 그러므로, 1주는 1월 4일이 들어가는 첫 번째 주와 같으므로, 월요일이 주의 첫 번째 날로서 간주됩니다.

WEEK_ISO()는 WEEK()의 현재 정의와 다름을 기억하십시오. 이 WEEK()는 최대 54까지의 값을 리턴합니다. WEEK() 함수의 경우, 주 1은 첫 번째 토요일이 들어있는 주입니다. 이것은 주에 1일만 있는 경우라도 1월 1일이 들어있는 주와 같습니다.

DAYOFWEEK_ISO() 및 WEEK_ISO()는 버전 7에서 작성된 데이터베이스에서는 자동으로 사용 가능합니다. 버전 7 이전에서 작성된 데이터베이스에서는 이러

한 함수를 사용할 수 없습니다. 그러한 데이터베이스에서 DAYOFWEEK_ISO() 및 WEEK_ISO() 함수를 사용 가능하게 하려면, **db2updb** 시스템 명령어를 사용하십시오. **db2updb**에 대한 내용은 이들 릴리스 정보에 있는 ″Command Reference″ 섹션을 참조하십시오.

부록 K DB2 CLI/ODBC/JDBC 추적 기능 사용

이 부록의 섹션이 갱신되었습니다. 이 추적 기능에 관한 최근 정보는 *문제점 해결 안내서*의 『추적』 장을 참조십시오.

메시지 참조서

사용 가능한 갱신사항

메시지 참조서는 FixPak 4에서 갱신되었습니다. 최신 PDF는 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

메시지 갱신사항

다음 테이블은 최종 메시지 참조서 서적이 발간된 후 변경된 메시지를 보여줍니다. 이 참고 항목에는 메시지 텍스트를 온라인으로 보는 데 대한 지시사항이 포함되어 있습니다.

표 16. 새 메시지와 변경된 메시지

메시지 코드	변경 특성
SQL2554N	새 이유 코드
SQL0490N	새 메시지
SQL20214N	새 메시지
SQL20211N	새 메시지

다음 테이블은 최종 메시지 참조서 서적이 발간된 후 변경된 SQL을 보여줍니다. 이 참고 항목에는 메시지 텍스트를 온라인으로 보는 데 대한 지시사항이 포함되어 있습니다.

표 17. 새 SQLSTATE 메시지

SQLSTATE 값	의미
428B7	SQL문에 지정된 숫자는 유효한 범위를 벗어났습니다.

표 17. 새 SQLSTATE 메시지 (계속)

SQLSTATE 값	의미
428FI	ORDER OF가 지정되었지만, 테이블 지정자가 ORDER BY 절을 포함하지 않습니다.
428FJ	ORDER BY는 뷰 또는 요약 테이블의 외부 fullselect에서 허용되지 않습니다.

메시지 텍스트를 온라인으로 읽기

사용자는 DB2가 설치된 운영 체제 기능에 대해 잘 알고 있다고 가정됩니다.

운영 체제 명령행에서 다음 DB2 메시지에 액세스할 수 있습니다.

접두부 설명

ASN DB2 복제를 통해 생성된 메시지

CCA CCA(Client Configuration Assistant)를 통해 생성된 메시지

CLI 콜 레벨 인터페이스(CLI)를 통해 생성된 메시지

DBA 제어 센터 및 데이터베이스 관리 유틸리티를 통해 생성된 메시지

DBI 설치 및 구성을 통해 생성된 메시지

DB2 명령행 처리기에 의해 생성된 메시지

DWC Data Warehouse Center에 의해 생성된 메시지

FLG Information Catalog Manager를 통해 생성된 메시지 및 이유 코드

GSE DB2 Spatial Extender를 통해 생성된 메시지

SAT DB2 Satellite를 통해 생성된 메시지

SPM 동기점 관리 프로그램을 통해 생성된 메시지

SQJ Embedded SQL in Java(SQLJ)를 통해 생성된 메시지

SQL 경고 및 오류 조건이 발견되었을 때 데이터베이스 관리 프로그램을 통해 생성된 메시지

또한 SQLSTATE 값과 연관된 메시지 텍스트 또한 온라인으로 볼 수 있습니다.

메시지 ID는 3자 메시지 접두부(위 목록 참조) 및 4-5자 메시지 번호로 구성됩니다. 오류 메시지의 심각도를 나타내는 끝의 1자는 선택사항입니다.

도움말에서 이러한 오류 메시지를 보려면 운영 체제 명령 프롬프트에 다음을 입력하십시오.

```
db2 "? XXXnnnnn"
```

여기서, *XXX* 메시지 접두부를 나타내고
*nnnnn*은 메시지 번호를 나타냅니다.

주: db2 명령 매개변수로 채택된 메시지 ID는 대소문자 구별되지 않으며 끝의 문자는 필요하지 않습니다.

따라서, 다음 명령은 동일한 결과를 산출합니다.

- db2 “? SQL0000N”
- db2 “? sql0000”
- db2 “? SQL0000n”

메시지 텍스트가 너무 길어서 화면 길이를 넘는 경우, 다음 명령을 사용하십시오 (unix 기반 시스템 및 기타 ‘more’를 지원하는 시스템용).

```
db2 "? XXXnnnnn" | more
```

또한 대화식 입력 모드로 도움말을 호출할 수도 있습니다. 대화식 입력 모드로 들어가려면 운영 체제 명령 프롬프트에 다음을 입력하십시오.

```
db2
```

대화식 입력 모드로 들어가면, 다음 명령 프롬프트에 명령을 입력할 수 있습니다.

```
db2 =>
```

이 모드에서 DB2 메시지 도움말을 가져오려면 명령 프롬프트에 다음을 입력하십시오.

```
?
```

주: 메시지 텍스트가 화면 길이를 초과할 경우, 그래픽이 아닌 워크스테이션을 가진 사용자는 출력에 파이프 기호를 사용하여 'more' 프로그램을 표시하거나 (unix 기반 시스템의 경우) -based systems) 출력을 파일로 경로 재지정한 다음 찾아볼 수 있습니다.

다음 명령을 실행하면 해당 SQLSTATE 값과 연관된 메시지 텍스트를 검색할 수 있습니다.

```
db2 "? nnnnn"
```

또는

```
db2 "? nn"
```

여기서, *nnnnn*은 5자 SQLSTATE(영숫자)이고 *nn*은 2자 SQLSTATE 클래스 코드입니다(SQLSTATE 값의 처음 2자).

SQL 참조서

SQL 참조서 사용 가능한 갱신사항

SQL 참조서는 갱신된 최신 .pdf를 http://www.ibm.com/software/data/db2/udb/winos2unix/support에서 온라인으로 다운로드할 수 있습니다. 또한 갱신된 참조에 이러한 참고 내용을 추가해 놓았습니다. 갱신된 모든 문서는 CD에서도 사용할 수 있습니다. 이 CD는 PTF 번호 U478862를 사용하여 DB2 서비스를 통해 주문할 수 있습니다. DB2 서비스에 대한 문의는 http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/help.d2w/report를 참조하십시오.

새로운 함수 및 프로시저 사용

버전 7 Fixpaks는 새로운 SQL 내장 스칼라 함수를 제공합니다. 이들 새 함수의 설명은 *SQL 참조서* 갱신내용을 참조하십시오. 새 함수는 데이터베이스 서버 코드가 새 서비스 레벨로 업그레이드될 때 각 데이터베이스에서 자동으로 사용 가능하지 않습니다. 이들 새 함수를 사용하려면, 시스템 관리자가 서버에서 각 데이터베이스를 지정하여 **db2updv7** 명령을 발행해야 합니다. 이 명령은 이 명령 실행 이전에 작성된 데이터베이스 오브젝트가 새로운 함수 서명과 일치하는 기존 함수 서명을 사용하도록 보장하는 항목을 데이터베이스에 작성합니다.

MQSeries 함수(DB2MQ 스키마에 정의되어 있음) 사용에 대한 자세한 정보는 221 페이지의 『MQSeries』를 참조하십시오.

SET SERVER OPTION - 문서 오류

SET SERVER OPTION 명령문의 참고 섹션에 잘못된 정보가 있습니다. 현재 참고는 다음과 같습니다.

- SET SERVER OPTION은 현재 password, fold_id 및 fold_pw 옵션만을 지원합니다.

이 문장은 올바르지 않습니다. SET SERVER OPTION는 IBM이 제공하지 않는 랩퍼용 서버 옵션을 포함한 모든 서버 옵션을 지원합니다. 이 참고는 무시해야 합니다.

CREATE TABLESPACE 컨테이너 절의 정정사항 및 컨테이너 문자열 정보

현재 리모트 자원(예를 들어, LAN 방향 재지정 드라이브 또는 NFS 마운트 파일 시스템)은 Network Appliance Filers, IBM iSCSI 또는 IBM Network Attached Storage를 사용하는 경우에만 지원됩니다. 이것은 리모트 자원이 지원되지 않는다고 표시한 현재 문서의 정정사항입니다.

GRANT(테이블, 뷰, 별명 특권) - 문서 오류

GRANT(테이블, 뷰 또는 별명 특권) 명령문의 참고 섹션의 글머리표가 올바르지 않습니다. 현재 참고는 다음과 같습니다.

- 별명에 대한 조작은 별명을 참조하는 명령문이 처리될 때 데이터 소스에 사용되는 인증 ID 특권에 따라 다르기 때문에, DELETE, INSERT, SELECT 및 UPDATE 특권은 별명에 대해 정의되지 않습니다.

이 문장은 무시해야 합니다. 이 문장을 빼면 섹션의 나머지 문장은 올바릅니다.

MQSeries 정보

스칼라 함수

MQPUBLISH

```
>>──MQPUBLISH──(──┬──────────────────────────────────────────────────────┬──msg-data──────────►
                  └─publisher-service──,──┬─────────────────────────┬───┘
                                          └─service-policy──,───────┘

►──┬──────────────────────────────────┬──)─────────────────────────────────────────────────►◄
   └─,──topic──┬───────────────────┬──┘
               │              (1)  │
               └─,──correl-id──────┘
```

주:

1 *service*와 *policy*를 이전에 정의하지 않은 경우, *correl-id*를 지정할 수 없습니다.

스키마는 DB2MQ입니다.

MQPUBLISH 함수는 데이터를 MQSeries에 발행합니다. 이 함수를 사용하려면 MQSeries Publish/Subscribe나 MQSeries Integrator 중 하나를 설치해야 합니다. 자세한 정보는 www.ibm.com/software/MQSeries를 참조하십시오.

MQPUBLISH 함수는 *service-policy*가 정의한 서비스 규정 품질을 사용하여 *msg-data*에 포함된 데이터를 *publisher-service*에 지정된 MQSeries 발행자에 발행합니다. 선택적으로 메시지에 주제를 지정할 수 있고, 사용자 정의 메시지 상관 ID를 지정할 수도 있습니다. 이 함수는 성공할 경우 ″1″값을 리턴하고, 성공하지 못할 경우 ″0″값을 리턴합니다.

publisher-service
: 메시지가 송신될 논리적 MQSeries 목적지가 포함된 문자열. 이 옵션을 지정하면 *publisher-service*가 AMT.XML 저장소 파일에 정의된 publisher Service Point를 참조해야 합니다. 서비스 지점은 메시지가 송신되거나 수신되는 논리적 끝점입니다. 서비스 지점 정의에는 MQSeries 대기행렬 관리자 이름 및 대기행렬 이름이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *publisher-service*가 지정되어 있지 않으면 DB2.DEFAULT.PUBLISHER가 사용됩니다. *publisher-service*의 최대 크기는 48바이트입니다.

service-policy
: 이 메시지를 처리할 때 사용할 MQSeries AMI Service Policy가 포함된 문자열. 이 옵션을 지정하면 *service-policy*가 AMT.XML 저장소 파일에 정의된 규정을 참조해야 합니다. 서비스 규정은 이 메시징 작업에 적용되어야 할 서비스 옵션의 품질 세트를 정의합니다. 이 옵션에는 메시지 우선순위와 메시지 지속성이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *service-policy*가 지정되어 있지 않으면 기본 DB2.DEFAULT.POLICY가 사용됩니다. *service-policy*의 최대 크기는 48바이트입니다.

msg-data

MQSeries를 통해 전송할 데이터가 포함된 문자열 표현식. VARCHAR 유형 문자열의 최대 크기는 4000바이트입니다. CLOB 유형 문자열의 최대 크기는 1MB입니다.

topic

메시지 발행에 대한 주제가 포함된 문자열 표현식. 주제가 지정되어 있지 않으면 이 메시지에 주제가 연관되지 않습니다. *topic*의 최대 크기는 40바이트입니다. 문자열 하나에 주제를 여러 개(40문자까지) 지정할 수 있습니다. 각 주제는 콜론으로 구분해야 합니다. 예를 들어, ″t1:t2:the third topic″은 이 메시지가 t1, t2, ″the third topic″이라는 세 주제와 연관되어 있음을 나타냅니다.

correl-id

이 메시지와 연관시킬 상관 ID가 포함된 선택적 문자열 표현식. 대개 *correl-id*를 요청과 응답 시나리오에 지정하여 요청과 응답을 연관시킵니다. 이 옵션을 지정하지 않으면 메시지에 상관 ID가 추가되지 않습니다. *correl-id*의 최대 크기는 24바이트입니다.

예

예 1: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 문자열 ″Testing 123″을 기본 publisher 서비스(DB2.DEFAULT.PUBLISHER)에 발행합니다. 메시지에 상관 ID나 주제는 지정되지 않습니다.

```
VALUES MQPUBLISH('Testing 123')
```

예 2: 이 예에서는 ″TESTS″라는 주제 하에 문자열 ″Testing 345″를 ″MYPUBLISHER″ publisher 서비스에 발행합니다. 기본 규정이 사용되고 상관 ID는 지정되지 않습니다.

```
VALUES MQPUBLISH('MYPUBLISHER','Testing 345', 'TESTS')
```

예 3: 이 예에서는 상관 ID를 ″TEST1″로 지정하고 ″MYPOLOCY″ 규정을 사용하여 문자열 ″Testing 678″을 ″MYPUBLISHER″ publisher 서비스에 발행합니다. 이 메시지는 ″TESTS″라는 주제로 발행됩니다.

```
VALUES MQPUBLISH('MYPUBLISHER','MYPOLICY','Testing 678','TESTS','TEST1')
```

예 4: 이 예에서는 상관 ID는 지정하지 않고 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 ″TESTS″라는 주제 하에 문자열 ″Testing 901″을 ″MYPUBLISHER″ publisher 서비스에 발행합니다.

```
VALUES MQPUBLISH('Testing 901','TESTS')
```

성공할 경우, 모든 예는 ‘1’값을 리턴합니다.

MQREADCLOB

```
►►──MQREADCLOB──(──┬──────────────────────────────────────────┬──)──────────────►◄
                   └─receive-service──┬────────────────────┬──┘
                                      └─,──service-policy──┘
```

스키마는 DB2MQ입니다.

MQREADCLOB 함수는 *service-policy*가 정의한 서비스 규정 품질을 사용하여 *receive-service*가 지정한 MQSeries 위치에서 메시지를 리턴합니다. 이 작업을 실행하면 대기행렬에서 *receive-service*와 연관된 메시지가 제거되지 않고, 대기행렬 헤드에 있는 메시지가 리턴됩니다. 리턴값은 메시지를 포함하는, 최대 길이가 IMB인 CLOB입니다. 리턴할 메시지가 없을 때는 널(NULL)이 리턴됩니다.

receive-service
: 메시지가 수신될 논리적 MQSeries 목적지가 포함된 문자열. 이 옵션을 지정하면 *receive-service*가 AMT.XML 저장소 파일에 정의된 Service Point를 참조해야 합니다. 서비스 지점은 메시지가 송신되거나 수신되는 논리적 끝점입니다. 서비스 지점 정의에는 MQSeries 대기행렬 관리자 이름 및 대기행렬 이름이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *receive-service*가 지정되어 있지 않으면 DB2.DEFAULT.SERVICE가 사용됩니다. *receive-service*의 최대 크기는 48바이트입니다.

service-policy
: 이 메시지를 처리할 때 사용할 MQSeries AMI Service Policy가 포함된 문자열. 이 옵션을 지정하면 *service-policy*가 AMT.XML 저장소 파일에 정의된 규정을 참조해야 합니다. 서비스 규정은 이 메시징 작업에 적용되어야 할 서비스 옵션의 품질 세트를 정의합니다. 이 옵션에는 메시지 우선순위와 메시지 지속성이 포함됩니다. 자세한 정보는 MQSeries Application Messaging

Interface를 참조하십시오. *service-policy*가 지정되어 있지 않으면 기본 DB2.DEFAULT.POLICY가 사용됩니다. *service-policy*의 최대 크기는 48바이트입니다.

예:

예 1: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 기본 서비스(DB2.DEFAULT.SERVICE)가 지정한 대기행렬 헤드에 있는 메시지를 읽습니다.

```
VALUES MQREADCLOB()
```

예 2: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 ″MYSERVICE″ 서비스가 지정한 대기행렬 헤드에 있는 메시지를 읽습니다.

```
VALUES MQREADCLOB('MYSERVICE')
```

예 3: 이 예에서는 ″MYPOLOCY″ 규정을 사용하여 ″MYSERVICE″ 서비스가 지정한 대기행렬 헤드에 있는 메시지를 읽습니다.

```
VALUES MQREADCLOB('MYSERVICE','MYPOLICY')
```

성공할 경우, 모든 예는 메시지 내용을 최대 크기가 1MB인 CLOB으로 리턴합니다. 리턴할 메시지가 없을 때는 널(NULL)이 리턴됩니다.

MQRECEIVECLOB

```
>>--MQRECEIVECLOB---------------------------------------------------->

>--(--+---------------------------------------------------------+--)------><
      '-receive-service--+-------------------------------------+-'
                         '-,--service-policy--+---------------+-'
                                              '-,--correl-id--'
```

스키마는 DB2MQ입니다.

MQRECEIVECLOB 함수는 *service-policy*가 정의한 서비스 규정 품질을 사용하여 *receive-service*가 지정한 MQSeries 위치에서 메시지를 리턴합니다. 이 작업을 수행하면 *receive-service*와 연관된 대기행렬에서 메시지가 제거됩니다. *correl-id*가 지정되어 있으면 일치하는 상관 ID가 있는 첫 번째 메시지가 리턴됩니다.

*correl-id*가 지정되어 있지 않으면 대기행렬 헤드에 있는 메시지가 리턴됩니다. 리턴값은 메시지를 포함하는, 최대 길이가 IMB인 CLOB입니다. 리턴할 메시지가 없을 때는 널(NULL)이 리턴됩니다.

receive-service

메시지가 수신될 논리적 MQSeries 목적지가 포함된 문자열. 이 옵션을 지정하면 *receive-service*가 AMT.XML 저장소 파일에 정의된 Service Point를 참조해야 합니다. 서비스 지점은 메시지가 송신되거나 수신되는 논리적 끝점입니다. 서비스 지점 정의에는 MQSeries 대기행렬 관리자 이름 및 대기행렬 이름이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *receive-service*가 지정되어 있지 않으면 DB2.DEFAULT.SERVICE가 사용됩니다. *receive-service*의 최대 크기는 48바이트입니다.

service-policy

이 메시지를 처리할 때 사용할 MQSeries AMI Service Policy가 포함된 문자열. 이 옵션을 지정하면, *service-policy*가 AMT.XML 저장소 파일에 정의된 규정을 참조해야 합니다[1]. *service-policy*가 지정되어 있지 않으면 기본 DB2.DEFAULT.POLICY가 사용됩니다. *service-policy*의 최대 크기는 48바이트입니다.

correl-id

이 메시지와 연관시킬 선택적 상관 ID가 포함된 문자열. 대개 *correl-id*를 요청과 응답 시나리오에 지정하여 요청과 응답을 연관시킵니다. 이 옵션을 지정하지 않으면 상관 ID가 사용되지 않습니다. *correl-id*의 최대 크기는 24바이트입니다.

예:

예 1: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 기본 서비스(DB2.DEFAULT.SERVICE)가 지정한 대기행렬 헤드에 있는 메시지를 수신합니다.

```
VALUES MQRECEIVECLOB()
```

1. 서비스 규정은 이 메시징 작업이 적용되어야 할 서비스 옵션 품질 세트를 정의합니다. 이 옵션에는 메시지 우선순위와 메시지 지속성이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오.

예 2: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 ″MYSERVICE″ 서비스가 지정한 대기행렬 헤드에 있는 메시지를 수신합니다.

```
VALUES MQRECEIVECLOB('MYSERVICE')
```

예 3: 이 예에서는 ″MYPOLOCY″ 규정을 사용하여 ″MYSERVICE″ 서비스가 지정한 대기행렬 헤드에 있는 메시지를 수신합니다.

```
VALUES MQRECEIVECLOB('MYSERVICE','MYPOLICY')
```

예 4: 이 예에서는 ″MYPOLOCY″ 규정을 사용하여 ″MYSERVICE″ 서비스가 지정한 대기행렬 헤드에서 ‘1234’와 일치하는 상관 ID가 있는 첫 번째 메시지를 수신합니다.

```
VALUES MQRECEIVECLOB('MYSERVICE',MYPOLICY','1234')
```

성공할 경우, 모든 예는 메시지 내용을 최대 크기가 1MB인 CLOB으로 리턴합니다. 리턴할 메시지가 없을 때는 널(NULL)이 리턴됩니다.

MQSEND

```
►►──MQSEND──(──┬────────────────────────────────────────────┬──msg-data──────────►
               └─send-service──,──┬──────────────────────┬──┘
                                  └─service-policy──,────┘

►──┬──────────────────────┬──)──────────────────────────────────────────────────►◄
   │                 (1)  │
   └─,──correl-id─────────┘
```

주:

1 *service*와 *policy*를 이전에 정의하지 않은 경우, *correl-id*를 지정할 수 없습니다.

스키마는 DB2MQ입니다.

MQSEND 함수는 *service-policy*가 정의한 서비스 규정 품질을 사용하여 *msg-data*에 포함된 데이터를 *send-service*가 지정한 MQSeries로 송신합니다. *correl-id*로 선택적 사용자 정의 메시지 상관 ID를 지정할 수도 있습니다. 이 함수는 성공할 경우 ″1″값을 리턴하고, 성공하지 못할 경우 ″0″값을 리턴합니다.

msg-data

MQSeries를 통해 전송할 데이터가 포함된 문자열 표현식. VARCHAR 유형에 속하는 데이터의 최대 크기는 4000바이트이고 CLOB 유형에 속하는 데이터의 최대 크기는 1MB입니다.

send-service

메시지가 송신될 논리적 MQSeries 목적지가 포함된 문자열. 이 옵션을 지정하면 *send-service*가 AMT.XML 저장소 파일에 정의된 service point를 참조해야 합니다. 서비스 지점은 메시지를 보내거나 받는 논리적인 끝점입니다. 서비스 지점 정의에는 MQSeries 대기행렬 관리자 이름 및 대기행렬 이름이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *send-service*가 지정되어 있지 않으면 DB2.DEFAULT.SERVICE가 사용됩니다. *send-service*의 최대 크기는 48바이트입니다.

service-policy

이 메시지를 처리할 때 사용할 MQSeries AMI Service Policy가 포함된 문자열. 이 옵션을 지정하면 *service-policy*가 AMT.XML 저장소 파일에 정의된 서비스 규정을 참조해야 합니다. 서비스 규정은 이 메시징 작업에 적용되어야 할 서비스 옵션의 품질 세트를 정의합니다. 이 옵션에는 메시지 우선순위와 메시지 지속성이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *service-policy*가 지정되어 있지 않으면 기본 DB2.DEFAULT.POLICY가 사용됩니다. *service-policy*의 최대 크기는 48바이트입니다.

correl-id

이 메시지와 연관시킬 상관 ID가 포함된 선택적 문자열. 대개 *correl-id*를 요청과 응답 시나리오에 지정하여 요청과 응답을 연관시킵니다. 이 옵션을 지정하지 않으면 상관 ID가 송신되지 않습니다. *correl-id*의 최대 크기는 24바이트입니다.

예:

예 1: 이 예에서는 상관 ID를 지정하지 않고 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 문자열 ″Testing 123″을 기본 서비스(DB2.DEFAULT.SERVICE)에 송신합니다.

```
VALUES MQSEND('Testing 123')
```

예 2: 이 예에서는 상관 ID를 지정하지 않고 ″MYPOLOCY″ 규정을 사용하여 문자열 ″Testing ″345″를 ″MYSERVICE″ 서비스에 송신합니다.

```
VALUES MQSEND('MYSERVICE','MYPOLICY','Testing 345')
```

예 3: 이 예에서는 상관 ID를 ″TEST3″으로 지정하고 ″MYPOLOCY″ 규정을 사용하여 문자열 ″Testing 678″을 ″MYSERVICE″ 서비스에 송신합니다.

```
VALUES MQSEND('MYSERVICE','MYPOLICY','Testing 678','TEST3')
```

예 4: 이 예에서는 상관 ID를 지정하지 않고 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 문자열 ″TESTING 901″을 ″MYSERVICE″ 서비스에 송신합니다.

```
VALUES MQSEND('MYSERVICE','Testing 901')
```

성공할 경우, 모든 예는 '1' 스칼라 값을 리턴합니다.

테이블 함수

MQREADALLCLOB

```
>>─MQREADALLCLOB─(─┬─────────────────────────────────────────────┬─┬──────────┬─)─────><
                   └─receive-service─┬────────────────────────┬──┘ └─num-rows─┘
                                     └─,─service-policy───────┘
```

스키마는 DB2MQ입니다.

MQREADALLCLOB 함수는 *service-policy* 서비스 규정 품질을 사용하여 *receive-service*가 지정한 MQSeries 위치에서 메시지와 메시지 메타데이터가 포함된 테이블을 리턴합니다. 이 작업을 수행하면 대기행렬에서 *receive-service*와 연관된 메시지가 제거되지 않습니다.

*num-rows*가 지정되어 있으면 *num-rows* 메시지의 최대 크기가 리턴됩니다. *num-rows*가 지정되어 있지 않으면 사용 가능한 모든 메시지가 리턴됩니다. 리턴된 테이블에는 다음 컬럼이 포함됩니다.

- MSG - MQSeries 메시지의 내용이 포함된 CLOB 컬럼
- CORRELID - 메시지를 연결할 때 사용되는 상관 ID가 포함된 VARCHAR(24) 컬럼

- TOPIC - 사용 가능한 경우, 메시지와 함께 발행된 제목이 포함된 VARCHAR(40) 컬럼
- QNAME - 메시지가 수신된 대기행렬의 이름이 포함된 VARCHAR(48) 컬럼
- MSGID - 메시지에 할당된 MQSeries 고유 ID가 포함된 CHAR(24) 컬럼
- MSGFORMAT - MQSeries가 정의한 메시지 형식이 포함된 VARCHAR(8) 컬럼. 일반적인 문자열의 형식은 MQSTR입니다.

receive-service

메시지를 읽을 논리적 MQSeries 목적지가 포함된 문자열. 이 옵션을 지정하면 *receive-service*가 AMT.XML 저장소 파일에 정의된 서비스 지점을 참조해야 합니다. 서비스 지점은 메시지가 송신되거나 수신되는 논리적 끝점입니다. 서비스 지점 정의에는 MQSeries 대기행렬 관리자 이름 및 대기행렬 이름이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *receive-service*가 지정되어 있지 않으면 DB2.DEFAULT.SERVICE가 사용됩니다. *receive-service*의 최대 크기는 48바이트입니다.

service-policy

이 메시지를 처리할 때 사용할 MQSeries AMI Service Policy가 포함된 문자열. 이 옵션을 지정하면 *service-policy*가 AMT.XML 저장소 파일에 정의된 규정을 참조해야 합니다. 서비스 규정은 이 메시징 작업에 적용되어야 할 서비스 옵션의 품질 세트를 정의합니다. 이 옵션에는 메시지 우선순위와 메시지 지속성이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *service-policy*가 지정되어 있지 않으면 기본 DB2.DEFAULT.POLICY가 사용됩니다. *service-policy*의 최대 크기는 48바이트입니다.

num-rows

이 함수가 리턴할 수 있는 메시지의 최대 개수가 포함된 양의 정수

예:

예 1: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 기본 서비스(DB2.DEFAULT.SERVICE)가 지정한 대기행렬에서 모든 메시지를 수신합니다. 메시지와 모든 메타데이터가 테이블로 리턴됩니다.

```
SELECT *
   FROM table (MQREADALLCLOB()) T
```

예 2: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 MYSERVICE 서비스가 지정한 대기행렬의 헤드에서 모든 메시지를 수신합니다. MSG와 CORRELID 컬럼만 리턴됩니다.

```
SELECT T.MSG, T.CORRELID
   FROM table (MQREADALLCLOB('MYSERVICE')) T
```

예 3: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 기본 서비스 (DB2.DEFAULT.SERVICE)가 지정한 대기행렬의 헤드를 읽습니다. CORRELID가 '1234'인 메시지와 모든 컬럼이 리턴됩니다.

```
SELECT *
   FROM table (MQREADALLCLOB()) T
   WHERE T.CORRELID = '1234'
```

예 4: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 기본 서비스 (DB2.DEFAULT.SERVICE)가 지정한 대기행렬 헤드에 있는 처음 10개의 메시지를 수신합니다. 모든 컬럼이 리턴됩니다.

```
SELECT *
   FROM table (MQREADALLCLOB(10)) T
```

MQRECEIVEALLCLOB

```
>>—MQRECEIVEALLCLOB—(——————————————————————————————————————————————>

>——┬————————————————————————————————————————————————————————┬——┬——————————————————┬——>
   └receive-service——┬——————————————————————————————————————┤  └┬———┬—num-rows——┘
                     └,—service-policy——┬———————————————————┤   └,——┘
                                        └,—correl-id————————┘

>—)————————————————————————————————————————————————————————————————><
```

스키마는 DB2MQ입니다.

MQRECEIVEALLCLOB 함수는 *service-policy* 서비스 규정 품질을 사용하여 *receive-service*가 지정한 MQSeries 위치에서 메시지와 메시지 메타데이터가 포함된 테이블을 리턴합니다. 이 작업을 수행하면 *receive-service*와 연관된 대기행렬에서 메시지가 제거됩니다.

*correl-id*를 지정하면, 일치하는 상관 ID가 있는 메시지만 리턴됩니다. *correl-id*가 지정되어 있지 않으면 대기행렬 헤드에 있는 메시지가 리턴됩니다.

*num-rows*가 지정되어 있으면 *num-rows* 메시지의 최대 크기가 리턴됩니다. *num-rows*가 지정되어 있지 않으면 사용 가능한 모든 메시지가 리턴됩니다. 리턴된 테이블에는 다음 컬럼이 포함됩니다.

- MSG - MQSeries 메시지의 내용이 포함된 CLOB 컬럼
- CORRELID - 메시지를 연결할 때 사용되는 상관 ID가 포함된 VARCHAR(24) 컬럼
- TOPIC - 사용 가능한 경우, 메시지와 함께 발행된 제목이 포함된 VARCHAR(40) 컬럼
- QNAME - 메시지가 수신된 대기행렬의 이름이 포함된 VARCHAR(48) 컬럼
- MSGID - 메시지에 할당된 MQSeries 고유 ID가 포함된 CHAR(24) 컬럼
- MSGFORMAT - MQSeries가 정의한 메시지 형식이 포함된 VARCHAR(8) 컬럼. 일반적인 문자열의 형식은 MQSTR입니다.

receive-service

메시지가 수신될 논리적 MQSeries 목적지가 포함된 문자열. 이 옵션을 지정하면 *receive-service*가 AMT.XML 저장소 파일에 정의된 서비스 지점을 참조해야 합니다. 서비스 지점은 메시지가 송신되거나 수신되는 논리적 끝점입니다. 서비스 지점 정의에는 MQSeries 대기행렬 관리자 이름 및 대기행렬 이름이 포함됩니다. 자세한 정보는 MQSeries Application Messaging Interface를 참조하십시오. *receive-service*가 지정되어 있지 않으면 DB2.DEFAULT. SERVICE가 사용됩니다. *receive-service*의 최대 크기는 48바이트입니다.

service-policy

이 메시지를 처리할 때 사용할 MQSeries AMI Service Policy가 포함된 문자열. 이 옵션을 지정하면 *service-policy*가 AMT.XML 저장소 파일에 정의된 규정을 참조해야 합니다. 서비스 규정은 이 메시징 작업에 적용되어야 할 서비스 옵션의 품질 세트를 정의합니다. 이 옵션에는 메시지 우선순위와 메시지 지속성이 포함됩니다. 자세한 정보는 MQSeries Application Messaging

Interface를 참조하십시오. *service-policy*가 지정되어 있지 않으면 기본 DB2.DEFAULT.POLICY가 사용됩니다. *service-policy*의 최대 크기는 48바이트입니다.

correl-id

이 메시지와 연관시킬 상관 ID가 포함된 선택적 문자열. 대개 *correl-id*를 요청과 응답 시나리오에 지정하여 요청과 응답을 연관시킵니다. 이 옵션을 지정하지 않으면 상관 ID가 사용되지 않습니다. correl-id의 최대 크기는 24바이트입니다.

num-rows

이 함수가 리턴할 수 있는 메시지의 최대 개수가 포함된 양의 정수

예:

예 1: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 기본 서비스(DB2.DEFAULT.SERVICE)가 지정한 대기행렬에서 모든 메시지를 수신합니다. 메시지와 모든 메타데이터가 테이블로 리턴됩니다.

```
SELECT *
   FROM table (MQRECEIVEALLCLOB()) T
```

예 2: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 MYSERVICE 서비스가 지정한 대기행렬의 헤드에서 모든 메시지를 수신합니다. MSG와 CORRELID 컬럼만 리턴됩니다.

```
SELECT T.MSG, T.CORRELID
   FROM table (MQRECEIVEALLCLOB('MYSERVICE')) T
```

예 3: 이 예에서는 ″MYPOLOCY″ 규정을 사용하여 ″MYSERVICE″ 서비스가 지정한 대기행렬 헤드에서 모든 메시지를 수신합니다. CORRELID가 ‘1234’인 메시지와 MSG와 CORRELID 컬럼만 리턴됩니다.

```
SELECT T.MSG, T.CORRELID
   FROM table (MQRECEIVEALLCLOB('MYSERVICE','MYPOLICY','1234')) T
```

예 4: 이 예에서는 기본 규정(DB2.DEFAULT.POLICY)을 사용하여 기본 서비스(DB2.DEFAULT.SERVICE)가 지정한 대기행렬 헤드에 있는 처음 10개의 메시지를 수신합니다. 모든 컬럼이 리턴됩니다.

```
SELECT *
   FROM table (MQRECEIVEALLCLOB(10)) T
```

MQSeries 함수에서의 CLOB 데이터 지원 추가

MQSeries 함수(DB2MQ 스키마에 정의되어 있음)에는 현재 VARCHAR 뿐만 아니라 CLOB 데이터와 함께 사용할 수 있는 함수가 포함되어 있습니다. 일부 상황에서는 새 함수가 CLOB 데이터 유형을 다루고 또 다른 일부 상황에서는 기존 함수가 CLOB 및 VARCHAR 데이터 유형을 다룹니다. 어떤 경우든 간에, CLOB 함수 구문은 VARCHAR 함수 구문과 일치합니다. 다음 테이블에 CLOB 데이터 사용을 지원하는 함수와 이에 상응하는 VARCHAR 함수를 나열해 놓았습니다.

표 18. CLOB 데이터 유형을 지원하는 MQSeries 함수

VARCHAR 데이터에 사용할 수 있는 함수	**CLOB 데이터에 사용할 수 있는 함수**
MQPUBLISH	MQPUBLISH
MQREAD	MQREADCLOB
MQRECEIVE	MQRECEIVECLOB
MQSEND	MQSEND
MQREADALL	MQREADALLCLOB
MQRECEIVEALL	MQRECEIVEALLCLOB

MQSeries 함수(DB2MQ 스키마에 정의되어 있음) 사용에 대한 자세한 정보는 221 페이지의 『MQSeries』를 참조하십시오.

데이터 유형 정보

데이터 유형의 승격

이 절에서 테이블 5는 각 데이터 유형에 대한 프로시저 목록을 보여줍니다. 다음에 주의하십시오.

1. 유니코드 데이터베이스의 경우, 다음은 데이터 유형과 같은 것으로 간주됩니다.
   - CHAR 및 GRAPHIC
   - VARCHAR 및 VARGRAPHIC
   - LONG VARCHAR 및 LONG VARGRAPHIC
   - CLOB 및 DBCLOB

2. 유니코드 데이터베이스에서 예를 들어, foo(CHAR(8))과 foo(GRAPHIC(8))과 같이 함수 서명에 유일한 차이점이 있는 CHAR와 GRAPHIC 데이터 유형 사이에 있는 함수를 작성할 수 있습니다. 이후 릴리스로 이주할 때 이주를 진행하려면 그 중 하나를 삭제해야 하므로 이와 같은 중복 함수를 정의하지 말도록 강력히 권장합니다.

   그런 중복 함수가 존재할 경우, 어느 것을 호출할 지는 두 전달 알고리즘에 의해 결정됩니다. 첫 번째 전달은 비유니코드 데이터베이스에서 함수 결정에 사용되는 것과 같은 알고리즘을 사용하여 일치하는 것을 찾으려 합니다. 일치하는 것을 찾을 수 없으면, CHAR 및 GRAPHIC 문자열에 대한 다음 승격 선행을 고려한 두 번째 전달이 수행됩니다.

   ```
   GRAPHIC->CHAR->VARGRAPHIC->VARCHAR->LONG VARGRAPHIC->LONG VARCHAR->
        ->DBCLOB->CLOB
   ```

데이터 유형 사이의 유형 변환(CAST)

″구별 유형과 관련된 다음 유형 변환(CAST)이 지원됩니다″라고 소개된 목록에 다음 항목이 추가되었습니다.

- 유니코드 데이터베이스의 경우, VARCHAR 또는 VARGRAPHIC으로부터 소스 데이터 유형이 CHAR 또는 GRAPHIC인 구별 유형 DT로 유형 변환(CAST) 하십시오.

다음은 ″표 6. 내장 데이터 유형 사이에서 지원되는 유형 변환(CAST)″의 갱신입니다. 영향을 받는 표 행만 포함되었습니다.

표 19. 내장 데이터 유형 사이에서 지원되는 유형 변환(CAST)

목표 데이터 유형 → / 소스 데이터 유형↓	**CHAR**	**VARCHAR**	**LONGVARCHAR**	**CLOB**	**GRAPHIC**	**VARGRAPHIC**	**LONGVARGRAPHIC**	**DBCLOB**
CHAR	Y	Y	Y	Y	Y[1]	Y[1]	-	-
VARCHAR	Y	Y	Y	Y	Y[1]	Y[1]	-	-
LONGVARCHAR	Y	Y	Y	Y	-	-	Y[1]	Y[1]
CLOB	Y	Y	Y	Y	-	-	-	Y[1]
GRAPHIC	Y[1]	Y[1]	-	-	Y	Y	Y	Y
VARGRAPHIC	Y[1]	Y[1]	-	-	Y	Y	Y	Y
LONGVARGRAPHIC	-	-	Y[1]	Y[1]	Y	Y	Y	Y
DBCLOB	-	-	Y[2]	Y[1]	Y	Y	Y	Y

1 유형 변환(CAST)은 유니코드 데이터베이스에만 지원됩니다.

2 유형 변환(CAST)은 유니코드 데이터베이스에만 지원됩니다. 명시적 캐스팅만 지원됩니다.

할당 및 비교

문자 및 그래픽 데이터 모두와 관련된 할당 및 비교는 문자열 중 하나가 리터럴일 때만 지원됩니다. 함수 결정의 경우, 그래픽 리터럴과 문자 리터럴은 모두 문자 및 그래픽 함수 매개변수에 일치합니다.

다음은 ″표 7. 할당 및 비교를 위한 데이터 유형 호환성″의 갱신입니다. 영향을 받는 표의 행 그리고 새로운 주석 6만 포함되었습니다.

피연산자	2진 정수	소수	부동 소수점	문자열	그래픽 문자열	날짜	시간	시간소인	2진 문자열	UDT
문자열	아니오	아니오	아니오	예	예 [6]	[1]	[1]	[1]	아니오 [3]	[2]
그래픽 문자열	아니오	아니오	아니오	예 [6]	예	아니오	아니오	아니오	아니오	[2]

6 유니코드 데이터베이스에만 지원됩니다.

문자열 할당

저장영역 할당

이 서브섹션의 마지막 단락이 다음과 같이 수정되었습니다.

문자열이 고정 길이 컬럼에 할당되고 문자열의 길이가 목표의 길이 속성보다 작을 때, 문자열은 필요한 수의 1바이트, 2바이트 또는 UCS-2[2] 공백으로 오른쪽으로 패드(pad)됩니다. 컬럼이 FOR BIT DATA 속성으로 정의되었더라도 패드 문자는 항상 공백입니다.

검색 할당

이 서브섹션의 세 번째 단락이 다음과 같이 수정되었습니다.

문자열이 고정 길이 변수에 할당되고 문자열의 길이가 목표의 길이 속성보다 작을 때, 문자열은 필요한 수의 1바이트, 2바이트 또는 UCS-2[2] 공백으로 오른쪽으로 패드(pad)됩니다. 문자열 FOR BIT DATA 속성으로 정의되었더라도 패드 문자는 항상 공백입니다.

2 UCS-2는 다른 등록정보로 여러 SPACE 문자를 정의합니다. 유니코드 데이터베이스의 경우, 데이터베이스 관리 프로그램은 항상 ASCII SPACE를 위치 x‘0020’에서 UCS-2 공백으로 사용합니다. EUC 데이터베이스의 경우, x‘3000’ 위치의 IDEOGRAPHIC SPACE가 패딩 GRAPHIC 문자열로 사용됩니다.

문자열 할당 변환 규칙

다음 단락이 이 서브섹션 끝에 추가되었습니다.

유니코드 데이터베이스의 경우, 문자열은 그래픽 컬럼에 할당될 수 있고 그래픽 문자열은 문자 컬럼에 할당될 수 있습니다.

그래픽 문자열 할당을 위한 DBCS 고려사항

이 서브섹션의 첫 번째 단락이 다음과 같이 수정되었습니다.

그래픽 문자열 할당은 문자열에 대한 것과 비슷한 방식으로 처리됩니다. 비유니코드 데이터베이스의 경우, 그래픽 문자열 데이터 유형은 다른 그래픽 문자열 데이터 유형만 호환되고, 숫자, 문자열 또는 날짜시간 데이터 유형과는 호환되지 않습니다. 유니코드 데이터베이스의 경우, 그래픽 문자열 데이터 유형은 문자열 데이터 유형과 호환됩니다.

문자열 비교

비교를 위한 변환 규칙

이 서브섹션은 다음과 같이 수정되었습니다.

두 문자열이 비교될 때, 필요하면 문자열 중 하나가 코드화 체계 및 다른 문자열의 코드 페이지로 먼저 변환됩니다. 세부사항은, *SQL 참조서*의 ″제 3 장 언어 요소″의 "문자열 변환 규칙"을 참조하십시오.

결과 데이터 유형 규칙

유니코드 데이터베이스의 문자 및 그래픽 문자열

이것은 ″그래픽 문자열″ 서브섹션 뒤에 삽입되는 새로운 서브섹션입니다.

유니코드 데이터베이스에서 문자열 및 그래픽 문자열은 호환됩니다.

피연산자가 ...이고	기타 피연산자가 ...일 경우	결과의 데이터 유형은 ...입니다
GRAPHIC(x)	CHAR(y) 또는 GRAPHIC(y)	GRAPHIC(z) 여기서, z = max(x,y)
VARGRAPHIC(x)	CHAR(y) 또는 VARCHAR(y)	VARGRAPHIC(z) 여기서, z = max(x,y)
VARCHAR(x)	GRAPHIC(y) 또는 VARGRAPHIC	VARGRAPHIC(z) 여기서, z = max(x,y)

피연산자가 ...이고	기타 피연산자가 ...일 경우	결과의 데이터 유형은 ...입니다
LONG VARGRAPHIC	CHAR(y) 또는 VARCHAR(y) 또는 LONG VARCHAR	LONG VARGRAPHIC
LONG VARCHAR	GRAPHIC(y) 또는 VARGRAPHIC(y)	LONG VARGRAPHIC
DBCLOB(x)	CHAR(y) 또는 VARCHAR(y) 또는 CLOB (y)	DBCLOB(z) 여기서, z = max(x,y)
DBCLOB(x)	LONG VARCHAR	DBCLOB(z) 여기서, z = max(x,16350)
CLOB(x)	GRAPHIC(y) 또는 VARGRAPHIC(y)	DBCLOB(z) 여기서, z = max(x,y)
CLOB(x)	LONG VARGRAPHIC	DBCLOB(z) 여기서, z = max(x,16350)

문자열 변환 규칙

세 번째 항목이 이 섹션의 다음 목록에 추가되었습니다.

각 코드 페이지 쌍에 대해, 결과는 다음 규칙을 순차적으로 적용하여 결정됩니다.

- 코드 페이지가 같으면, 결과는 그 코드 페이지입니다.
- 한 코드 페이지가 BIT DATA(코드 페이지 0)이면, 결과 코드 페이지는 BIT DATA입니다.
- 유니코드 데이터베이스에서 한 코드 페이지가 다른 코드 페이지와 다른 코드화 체계로 데이터를 표시하면, 결과는 UTF-8 위의 UCS-2입니다. (즉, 문자 데이터 유형 위의 그래픽 데이터 유형입니다.)[1]
- 그렇지 않으면, 결과 코드 페이지는 *SQL 참조서*의 ″제 3 장 언어 요소″의 "문자열 변환 규칙" 테이블 8에 의해 결정됩니다. 테이블의 ″첫 번째″ 항목은 첫 번째 피연산자의 코드 페이지가 선택되었음을 의미하고 ″두 번째″는 두 번째 피연산자의 코드 페이지가 선택되었음을 의미합니다.

1 비유니코드 데이터베이스에서 다른 코드화 체계 사이의 변환은 지원되지 않습니다.

표현식

다음이 추가되었습니다.

유니코드 데이터베이스에서 문자 또는 그래픽 문자열을 승인하는 표현식은 변환이 지원되는 모든 문자열 유형을 승인합니다.

병합 연산자

다음이 이 서브섹션 끝에 추가되었습니다.

유니코드 데이터베이스에서 문자열 피연산자와 그래픽 문자열 피연산자 모두와 관련된 병합은 먼저 문자 피연산자를 그래픽 피연산자로 변환합니다. 비유니코드 데이터베이스에서 병합에 문자 및 그래픽 피연산자 모두 포함될 수 없음을 주의하십시오.

술어

″다음 규칙이 모든 술어 유형에 적용됩니다″로 소개된 목록에 다음 항목이 추가되었습니다.

- 유니코드 데이터베이스에서 문자 또는 그래픽 문자열을 승인하는 모든 술어는 변환이 지원되는 모든 문자열 유형을 승인합니다.

유니코드 정보

스칼라 함수와 유니코드

유니코드 데이터베이스에서 character GRAPHIC 또는 graphic 문자열을 승인하는 모든 스칼라 함수는 변환이 지원되는 이들 문자열을 승인합니다.

GRAPHIC 유형 및 DATE/TIME/TIMESTAMP 호환성

다음 절에서, ″character string″을 가진 날짜 시간 값에 대한 언급은 ″string″ 표현으로 바뀌었습니다. DB2는 현재 유니코드 데이터베이스의 경우 날짜 시간 값의 ″graphic string″ 표현만을 지원합니다.

날짜 시간 값의 문자열 표현

데이터 유형이 DATE, TIME 또는 TIMESTAMP인 값은 사용자가 볼 수 있는 내부 형식으로 표시됩니다. 그러나, 날짜, 시간, 시간소인 값은 문자열로도 표현할 수 있습니다. 데이터 유형이 DATE, TIME 또는 TIMESTAMP인 상수나 변수가 없기 때문에 이 기능은 유용합니다. 검색을 위해서는 날짜 시간 값을 문자열 변수에 지정해야 합니다. CHAR 함수 또는 GRAPHIC 함수(유니코드 데이터베이스의 경우에 한해)를 사용하여 날짜 시간 값을 문자열 표현으로 바꿀 수 있습니다. 프로그램이 프리컴파일되거나 데이터베이스에 바인드될 때 DATETIME 옵션을 지정하여 대체하지 않는 한 날짜 및 시간 형식의 문자열 표현은 응용프로그램의 국가/지역 코드와 연관된 날짜 시간 값의 기본 형식입니다.

길이에 상관 없이, 대형 오브젝트(LOB) 문자열, LONG VARCHAR 값 또는 LONG VARGRAPHIC 값을 사용하여 날짜 시간 값을 나타낼 수 없습니다 (SQLSTATE 42884).

날짜 시간 값의 올바른 문자열 표현이 내부 날짜 시간 값을 가진 조작에 사용된 경우, 문자열 표현은 조작이 수행되기 전에 내부 날짜, 시간, 시간소인 형태로 변환됩니다.

날짜 및 시간소인 문자열은 문자 및 숫자로만 구성되어야 합니다.

날짜 문자열, 시간 문자열, 날짜 시간 문자열
이러한 용어 정의는 약간 변경되었습니다. ″character string″ 표현에 대한 언급은 ″string″ 표현으로 바뀌었습니다.

데이터 유형 변환

이제 DATE, TIME, TIMESTAMP를 GRAPHIC 및 VARGRAPHIC으로 유형을 변환할 수 있습니다. GRAPHIC 및 VARGRAPHIC도 DATE, TIME, TIMESTAMP으로 유형을 변환할 수 있습니다. 그래픽 문자열은 유니코드 데이터베이스에만 지원됩니다.

할당 및 비교

그래픽 문자열, DATE, TIME, TIMESTAMP 값 사이의 할당 및 비교와 관련하여 새로운 데이터 유형 호환성이 있습니다. 그래픽 문자열은 유니코드 데이터베이스에만 지원됩니다.

날짜 시간 할당

날짜 시간 할당에 대한 기본 규칙은 DATE, TIME 또는 TIMESTAMP 값을 일치하는 데이터 유형(DATE, TIME 또는 TIMESTAMP)을 가진 컬럼 또는 고정 또는 가변 길이 문자열 변수 또는 문자열 컬럼에만 할당할 수 있다는 점입니다. 할당은 LONG VARCHAR, CLOB, LONG VARGRAPHIC, DBCLOB 또는 BLOB 변수나 컬럼에 대한 것이 아니어야 합니다.

날짜 시간 값이 문자열 변수나 문자열 컬럼에 지정되면, 문자열 표현으로 자동으로 변환이 이루어집니다. 날짜, 시간 또는 시간소인의 부분에서 앞의 0은 생략되지 않습니다. 문자열 표현 형식에 따라 필요한 대상 길이가 달라집니다. 대상의 길이가 필요한 길이보다 길다면, 대상은 고정 길이 문자열로서 오른쪽에 공백이 삽입됩니다. 대상의 길이가 필요한 길이보다 적다면, 관련 날짜 시간 값 유형 및 대상 유형에 따라 결과는 달라집니다.

대상이 호스트 변수인 경우, 다음 규칙이 적용됩니다.

- **DATE:** 변수 길이가 10자 미만인 경우, 오류가 발생합니다.
- **TIME:** USA 형식이 사용된 경우, 변수 길이는 8자 이상이어야 합니다. 다른 형식의 경우 길이는 5자 이상이어야 합니다.

  ISO 또는 JIS 형식이 사용되었으며 호스트 변수 길이가 8자 미만인 경우, 결과에서 시간의 두 번째 부분이 생략되어 표시기 변수(제공된 경우)에 지정됩니다. SQLCA의 SQLWARN1 필드는 생략을 나타내도록 설정됩니다.
- **TIMESTAMP** 호스트 변수가 19자 미만인 경우, 오류가 발생합니다. 길이가 26자 미만이지만 19자보다 긴 경우, 값을 마이크로초 부분의 뒷 자리수가 생략됩니다. SQLCA의 SQLWARN1 필드는 생략을 나타내도록 설정됩니다.

DATE

```
▶▶—DATE—(—표현식—)————————————————————————▶◀
```

스키마는 SYSIBM입니다.

DATE 함수는 값에서 날짜를 리턴합니다.

인수는 날짜, 시간소인 3 652 059 미만의 양수, 날짜 또는 시간소인의 올바른 문자열 표현, LONG VARCHAR, CLOB, LONG VARGRAPHIC, DBCLOB 또는 BLOB이 아닌 길이 7인 문자열이어야 합니다.

유니코드 데이터베이스만이 날짜 또는 시간소인의 그래픽 문자열 표현으로 된 인수를 지원합니다.

인수가 길이 7인 문자열인 경우, *yyyynnn* 형태로 올바른 날짜를 나타내야 합니다. 여기서, *yyyy*은 연도를 나타내는 숫자이고 *nnn*이고 001에서 366 사이의 숫자로서 해당 연도의 날짜를 나타냅니다.

함수의 결과는 날짜입니다. 인수가 널(NULL)일 수 있는 경우, 결과는 널(NULL)일 수 있습니다. 인수가 널(NULL)인 경우, 결과는 널(NULL) 값입니다.

다른 규칙은 인수 데이터 유형에 따라 달라집니다.

- 인수가 날짜, 시간소인 또는 날짜나 시간소인의 올바른 문자열 표현인 경우,
  – 결과는 값의 날짜 부분입니다.
- 인수가 숫자인 경우,
  – 결과는 0001년 1월 1일 이후 *n*-1일인 날짜로서, *n*은 숫자의 정수 부분입니다.
- 인수가 길이 7인 문자열인 경우,
  – 결과는 문자열로 표현된 날짜입니다.

예:

RECEIVED(timestamp) 컬럼의 내부 값이 ‘1988-12-25-17.12.30.000000’인 경우,

- 이 예의 결과는 ‘1988-12-25’의 내부 표현이 됩니다.

```
DATE(RECEIVED)
```

- 이 예의 결과는 '1988-12-25'의 내부 표현이 됩니다.

  DATE('1988-12-25')

- 이 예의 결과는 '1988-12-25'의 내부 표현이 됩니다.

  DATE('25.12.1988')

- 이 예의 결과는 '0001-02-04'의 내부 표현이 됩니다.

  DATE(35)

GRAPHIC

```
>>--GRAPHIC--(--graphic-expression--+--------------+--)----------------><
                                    '-,--integer--'
```

스키마는 SYSIBM입니다.

GRAPHIC 함수는 그래픽 문자열 유형의 GRAPHIC 표현 또는 날짜 시간 유형의 GRAPHIC 표현을 나타냅니다.

graphic-expression

  그래픽 문자열인 값을 리턴하는 *표현식*

integer

  결과로 표시되는 GRAPHIC 데이터 유형의 길이 속성을 지정하는 정수 값. 값은 1 - 127이어야 합니다. *integer*가 지정되지 않은 경우, 결과의 길이는 첫 번째 인수의 길이와 같습니다.

함수의 결과는 GRAPHIC입니다. 인수가 널(NULL)일 수 있는 경우, 결과는 널(NULL)일 수 있습니다. 인수가 널(NULL)인 경우, 결과는 널(NULL) 값입니다.

날짜 시간을 그래픽으로:

```
>>--GRAPHIC--(--datetime-expression--+---------------+--)--------------><
                                     '-,--+-ISO---+-'
                                          +-USA---+
                                          +-EUR---+
                                          +-JIS---+
                                          '-LOCAL-'
```

날짜 시간을 그래픽으로:

datetime-expression

다음 세 가지 데이터 유형 중 하나인 표현식

date 결과는 두 번째 인수가 지정하는 형식의 날짜 그래픽 문자열 표현입니다. 결과 길이는 10입니다. 두 번째 인수가 지정되었으며 올바른 값이 아닌 경우 오류가 발생합니다(SQLSTATE 42703).

time 결과는 두 번째 인수가 지정하는 형식의 시간 그래픽 문자열 표현입니다. 결과 길이는 8입니다. 두 번째 인수가 지정되었으며 올바른 값이 아닌 경우 오류가 발생합니다(SQLSTATE 42703).

timestamp

두 번째 인수를 적용할 수 없으며 지정할 수 없습니다(SQLSTATE 42815). 결과는 시간소인 그래픽 문자열 표현입니다. 결과 길이는 26입니다.

문자열 코드 페이지는 응용프로그램 서버(AS)에서의 데이터베이스 코드 페이지입니다.

TIME

```
►►—TIME—(—표현식—)——————————————————————————————————————►◄
```

스키마는 SYSIBM입니다.

TIME 함수는 값에서 시간을 리턴합니다.

인수는 날짜, 시간소인, LONG VARCHAR, CLOB, LONG VARGRAPHIC, DBCLOB 또는 BLOB이 아닌 시간이나 시간소인의 올바른 문자열 표현이어야 합니다.

유니코드 데이터베이스만이 날짜 또는 시간소인의 그래픽 문자열 표현으로 된 인수를 지원합니다.

함수의 결과는 시간입니다. 인수가 널(NULL)일 수 있는 경우, 결과는 널(NULL)일 수 있습니다. 인수가 널(NULL)인 경우, 결과는 널(NULL) 값입니다.

다른 규칙은 인수 데이터 유형에 따라 달라집니다.

- 인수가 시간인 경우,
  – 결과는 해당 시간입니다.
- 인수가 시간소인인 경우,
  – 결과는 시간소인의 시간 부분입니다.
- 인수가 문자열인 경우,
  – 결과는 문자열로 표현된 시간입니다.

예:

- 해당 날짜에 현재 시간보다 적어도 한 시간 후에 수신된 IN_TRAY 샘플 테이블에서 모든 참고를 선택하십시오.

```
SELECT * FROM IN_TRAY
  WHERE TIME(RECEIVED) >= CURRENT TIME + 1 HOUR
```

TIMESTAMP

```
►►—TIMESTAMP—(—표현식—┬─────────────┬—)——————————————————►◄
                      └─,expression─┘
```

스키마는 SYSIBM입니다.

TIMESTAMP 함수는 값이나 값 쌍으로부터 시간소인을 리턴합니다.

유니코드 데이터베이스만이 날짜, 시간 또는 시간소인의 그래픽 문자열 표현으로 된 인수를 지원합니다.

인수에 대한 규칙은 두 번째 인수가 지정되었는지 여부에 따라 다릅니다.

- 한 인수만 지정된 경우,
  – 인수는 시간소인, 시간소인의 올바른 문자열 표현 또는 LONG VARCHAR, CLOB, LONG VARGRAPHIC, DBCLOB 또는 BLOB이 아닌 길이가 14인 문자열이어야 합니다.

    길이가 14인 문자열은 올바른 날짜 및 시간을 *yyyyxxddhhmmss* 형태로 나타내는 숫자 문자열이어야 합니다. 여기서, *yyyy*는 연도이고, *xx*는 월이고, *dd*는 일이고, *hh*는 시간이고, *mm*은 분이고, *ss*는 초입니다.

- 두 인수 모두 지정된 경우,
  - 첫 번째 인수는 날짜 또는 날짜의 올바른 문자열 표현이어야 하며, 두 번째 인수는 시간 또는 시간의 올바른 문자열 표현이어야 합니다.

함수의 결과는 시간소인입니다. 인수가 널(NULL)일 수 있는 경우, 결과는 널(NULL)일 수 있습니다. 인수가 널(NULL)인 경우, 결과는 널(NULL) 값입니다.

다른 규칙은 두 번째 인수가 지정되었는지 여부에 따라 다릅니다.

- 두 인수 모두 지정된 경우,
  - 결과는 첫 번째 인수로 지정된 날짜와 두 번째 인수로 지정된 시간으로 구성된 시간소인입니다. 시간소인의 마이크로초 부분은 0입니다.
- 한 인수만 지정되었으며 해당 인수가 시간소인인 경우,
  - 결과는 해당 시간소인입니다.
- 한 인수만 지정되었으며 해당 인수가 문자열인 경우,
  - 결과는 해당 문자열로 표현된 시간소인입니다. 인수가 길이 14인 문자열인 경우, 시간소인의 마이크로초 부분은 0입니다.

예:

- START_DATE(date) 컬럼의 값은 1988-12-25와 같고, START_TIME(time) 컬럼의 값은 17.12.30인 경우,

  ```
  TIMESTAMP(START_DATE, START_TIME)
  ```

  '1988-12-25-17.12.30.000000' 값을 리턴합니다.

VARGRAPHIC

문자열을 Vargraphic으로:

```
►►—VARGRAPHIC—(—character-string-expression—)——————————————►◄
```

날짜 시간을 Vargraphic으로:

```
►►—VARGRAPHIC—(—datetime-expression—)——————————————►◄
```

그래픽을 Vargraphic으로:

```
►►—VARGRAPHIC—(—graphic-string-expression——┬──────────┬─)——————————►◄
                                           └─,—integer─┘
```

스키마는 SYSIBM입니다.

VARGRAPHIC 함수는 다음에 대한 그래픽 문자열 표현을 리턴합니다.

- 1바이트 문자를 2바이트 문자로 변환하는 문자열 값
- 날짜 시간 값(유니코드 데이터베이스에서만 지원됨)
- 첫 번째 인수가 그래픽 문자열 유형인 경우 그래픽 문자열 값

함수의 결과는 가변 길이 그래픽 문자열(VARGRAPHIC 데이터 유형)입니다. 인수가 널(NULL)일 수 있는 경우, 결과는 널(NULL)일 수 있습니다. 인수가 널(NULL)인 경우, 결과는 널(NULL) 값입니다.

문자를 Vargraphic으로:

character-string-expression
: 값이 LONG VARCHAR 또는 CLOB이 아닌 문자열 데이터 유형이어야 하며 최대 길이가 16 336바이트보다 길어야 하는 표현식.

결과의 길이 속성은 인수의 길이 속성과 같습니다.

S가 *character-string-expression* 값을 나타낼 경우, S의 각 1바이트 문자는 결과의 해당 2바이트 표현 또는 2바이트 대체 문자로 변환됩니다. S의 각 2바이트 문자는 '현상태대로' 맵핑됩니다. 2바이트 문자의 첫 번째 바이트가 S의 마지막 바이트로 표시되는 경우, 2바이트 대체 문자로 변환됩니다. S의 문자 순서는 보존됩니다.

다음은 변환에 대한 추가 고려사항입니다.

- 유니코드 데이터베이스의 경우, 이 함수는 문자열을 연산자 코드 페이지에서 UCS-2로 변환합니다. DBCS 문자를 포함한 모든 연산자 문자가 변환됩니다. 두 번째 인수가 제공되면, 결과 UCS-2 문자열의 원하는 길이(UCS-2 문자 수)를 지정합니다.
- VARGRAPHIC 함수에 의해 2바이트 코드 포인트로 변환은 연산자 코드 페이지에 기초합니다.

- 연산자의 2바이트 문자는 변환되지 않습니다. 기타 모든 문자는 해당 2바이트 표현으로 변환됩니다. 해당 2바이트 표현이 없는 경우, 코드 페이지의 2바이트 대체 문자가 사용됩니다.
- 하나 이상의 2바이트 대체 문자가 결과에 리턴되지 않는 경우 경고 또는 오류 코드가 생성됩니다.

날짜 시간을 Vargraphic으로:

datetime-expression
: 값이 DATE, TIME 또는 TIMESTAMP 데이터 유형이어야 하는 표현식.

그래픽을 Vargraphic으로:

graphic-string-expression
: 그래픽 문자열인 값을 리턴하는 표현식.

integer
: 결과 가변 길이 그래픽 문자열의 길이 속성. 값은 0 - 16 336이어야 합니다. 이 인수가 지정되지 않은 경우, 결과의 길이는 인수의 길이와 같습니다.

*graphic-string-expression*의 길이가 결과의 길이 속성보다 길 경우, 잘린 문자가 모두 공백이고 *graphic-string-expression*이 긴 문자열(LONG VARGRAPHIC 또는 DBCLOB)이 아닌 경우를 제외하고 데이터가 잘리면서 경고가 리턴됩니다 (SQLSTATE 01004).

유니코드 데이터베이스의 더 큰 인덱스 키

ALTER TABLE

1차 및 고유 키를 포함하는 인덱스의 일부인 변수 길이 컬럼의 길이는 레지스트리 변수 DB2_INDEX_2BYTEVARLEN이 설정되었을 때 정의되며, 255바이트보다 긴 길이로 변경될 수 있습니다. 가변 길이 컬럼이 외부 키와 관련되어 있다는 사실은 레지스트리 변수 설정과 관계없이 더 이상 해당 컬럼의 길이가 255바이트보다 더 크게 변경되지 않도록 하지는 못합니다. 그러나, 255보다 큰 길이는 해당

기본 키에 있는 컬럼이 255바이트보다 큰 길이를 가지고 있는 경우를 제외하고는 테이블로 삽입될 수 없습니다. 이것은 기본 키가 레지스트리 변수 ON으로 작성된 경우에만 가능합니다.

CREATE INDEX

레지스트리 변수 DB2_INDEX_2BYTEVARLEN이 ON인 경우 길이가 255바이트보다 큰 변수 길이 컬럼에 인덱스가 정의될 수 있습니다.

CREATE TABLE

변수 keypart가 있는 1차 및 고유 키는 레지스트리 변수 DB2_INDEX_2BYTEVARLEN이 ON인 경우 255보다 큰 크기를 가질 수 있습니다. 외부 키는 길이가 255바이트보다 큰 변수 길이 컬럼에 정의될 수 있습니다.

ALLOCATE CURSOR 명령문 참고 섹션이 올바르지 않음

ALLOCATE CURSOR 명령문의 참고 섹션에서 두 가지 글머리표 항목이 잘못 인쇄되었습니다. 이 항목에 포함된 정보는 무시하십시오.

GET DIAGNOSTICS문에서의 추가 옵션

GET DIAGNOSTICS문

GET DIAGNOSTICS문은 이전에 실행된 SQL문에 대한 정보를 얻는 데 사용됩니다. 이 명령문의 구문은 다음과 같이 갱신되었습니다.

명령 구문

```
>>─GET DIAGNOSTICS──┬─SQL-variable-name─=─┬─ROW_COUNT─────┬─┬──────────────────────────><
                    │                     └─RETURN_STATUS─┘ │
                    └─┤ condition-information ├─────────────┘
```

condition-information:

```
                     ┌─,──────────────────────────────────────────────┐
                     ▼                                                │
├──EXCEPTION──1──────SQL-variable-name──=──┬─MESSAGE_TEXT──────┬──────┴──────────────┤
                                           └─DB2_TOKEN_STRING──┘
```

명령 매개변수

SQL-variable-name
: 지정 목표인 변수를 식별합니다. ROW_COUNT 또는 RETURN_STATUS가 지정된 경우, 변수는 정수 값이어야 합니다. 그렇지 않으면, 변수는 CHAR 또는 VARCHAR이어야 합니다. SQL 변수는 복합 텍스트 명령문에서 정의될 수 있습니다.

ROW_COUNT
: 이전 SQL문과 연관된 행 수를 지정합니다. 이전 SQL문이 DELETE, INSERT 또는 UPDATE문이면, ROW_COUNT는 트리거나 참조 무결정 제한조건에 의해 영향을 받는 행을 제외하고 해당 명령문으로 삭제, 삽입 또는 갱신된 행 수를 식별합니다. 이전 명령문이 PREPARE문인 경우, ROW_COUNT는 준비된 명령문에서 결과 행의 *estimated* 수를 식별합니다.

RETURN_STATUS
: 명령문이 상태를 리턴하는 프로시저를 호출하는 CALL문인 경우, 이전에 실행된 SQL문과 연관된 저장 프로시저에서 리턴된 상태 값을 식별합니다. 이전 명령문이 그러한 명령문이 아닌 경우, 리턴된 값은 의미가 없으며 임의 정수일 수 있습니다.

조건 정보
: 이전에 실행된 SQL문이 리턴되었다는 오류나 경고 정보를 지정합니다. 오류

에 대한 정보가 필요한 경우, GET DIAGNOSTICS문이 오류를 핸들할 핸들러에 지정된 처음 명령문이어야 합니다. 경고에 대한 정보가 필요하며 핸들러가 경고 조건의 제어를 가질 경우, GET DIAGNOSTICS문이 핸들러에 지정된 처음 명령문이어야 합니다. 핸들러가 경고 조건을 제어하지 않는 경우, GET DIAGNOSTICS문은 실행되는 다음 명령문이어야 합니다.

MESSAGE_TEXT

이전에 실행된 SQL문에서 리턴된 오류나 경고 메시지 텍스트를 식별합니다. 메시지 텍스트는 명령문이 처리된 데이터베이스 서버의 언어로 리턴됩니다. 명령문의 SQLCODE가 0으로 완료되는 경우, 빈 문자열이나 공백이 리턴됩니다.

DB2_TOKEN_STRING

이전에 실행된 SQL문에서 리턴된 오류나 경고 메시지 토큰을 식별합니다. 명령문의 SQLCODE가 0으로 완료되거나 SQLCODE에 토큰이 없는 경우, 빈 문자열이나 공백이 리턴됩니다.

Subselect에서의 ORDER BY

DB2는 이제 ORDER BY를 subselects 및 fullselect에서 지원합니다.

fullselect

다음은 order-by 절의 위치를 표시하는 수정된 fullselect의 부분 구문 다이어그램입니다.

```
                                   ┌──────────────────────────────────────────────────────────────┐
                                   ▼                                                              │
►►──┬─subselect─────────┬──────────┬───────────────────────────────────────────────────────┬─────┴───────►
    ├─(fullselect)──────┤          └──┬─UNION─────────────┬──┬─subselect────────────┬──┘
    └─┤ values-clause ├─┘             ├─UNION ALL─────────┤  ├─(fullselect)─────────┤
                                      ├─EXCEPT────────────┤  └─┤ values-clause ├────┘
                                      ├─EXCEPT ALL────────┤
                                      ├─INTERSECT─────────┤
                                      └─INTERSECT ALL─────┘

►──┬──────────────────┬─────────────────────────────────────────────────────────────────────────────►◄
   └─order-by-clause──┘
```

ORDER BY 절이 포함된 fullselect는 다음에서 지정할 수 없습니다.

- 요약 테이블
- 뷰의 가장 외부 fullselect(SQLSTATE 428FJ SQLCODE -20211)

fullselect에서 ORDER BY 절은 쿼리로 리턴되는 행의 순서에 영향을 미치지 않습니다. ORDER BY 절이 가장 외부 fullselect에서 지정되는 경우 리턴되는 행의 순서에만 영향을 미칩니다.

subselect

다음은 order-by 절의 위치를 표시하는 수정된 subselect의 전체 구문 다이어그램입니다.

```
►►──select-clause──from-clause──┬────────────────┬──┬───────────────────┬──────────────────►
                                └─where-clause───┘  └─group-by-clause───┘

►──┬─────────────────┬──┬──────────────────┬──────────────────────────────────────────────►◄
   └─having-clause───┘  └─order-by-clause──┘
```

subselect 절은 다음 순서로 처리됩니다.

1. FROM 절
2. WHERE 절
3. GROUP BY 절
4. HAVING 절
5. SELECT 절
6. ORDER BY 절

ORDER BY 절이 포함된 subselect는 다음에서 지정할 수 없습니다.

- 뷰의 가장 외부 fullselect
- 요약 테이블
- subselect를 괄호로 묶은 경우 외

예를 들어, 다음은 유효하지 않습니다(SQLSTATE 428FJ SQLCODE -20211).

```
SELECT * FROM T1
   ORDER BY C1
UNION
SELECT * FROM T2
   ORDER BY C1
```

다음 예는 유효합니다.

```
(SELECT * FROM T1
   ORDER BY C1)
UNION
(SELECT * FROM T2
   ORDER BY C1)
```

subselect에서 ORDER BY 절은 쿼리로 리턴되는 행의 순서에 영향을 미치지 않습니다. ORDER BY 절이 가장 외부 fullselect에서 지정되는 경우 리턴되는 행의 순서에만 영향을 미칩니다.

order-by 절

다음은 수정된 order-by 절의 전체 구문 다이어그램입니다.

```
               ┌─,───────────────────────────────┐
               ▼          ┌─ASC──┐                │
►►──ORDER BY──┬─sort-key─┤      ├───────────────┴┬──────────────────────────────►◄
              │          └─DESC─┘                 │
              └─ORDER OF──table-designator───────┘
```

sort-key:

```
├──┬─simple-column-name──┬────────────────────────────────────────────┤
   ├─simple-integer──────┤
   └─sort-key-expression─┘
```

ORDER OF *table-designator*

*table-designator*에서 사용되는 동일한 순서가 subselect의 결과 테이블에 적용되어야 함을 지정합니다. 이 절을 지정하는 subselect의 FROM 절에는 *table-designator*과 일치하는 테이블 참조가 있어야 합니다(SQLSTATE 42703). 지정된 *table-designator*에 해당하는 subselect(또는 fullselect)는 데이터에 종속하는 ORDER BY 절을 포함해야 합니다(SQLSTATE 428FI SQLCODE -20210). 적용되는 순서는 중첩된 subselect(또는 fullselect)에서 ORDER BY 절의 컬럼이 외부 subselect(또는 fullselect) 절에 포함된 경우 동일하며, 이 컬럼이 ORDER OF 절 대신 지정된 경우 동일합니다. 테이블 지정자에 대한 자세한 정보는 *SQL 참조서*에서 "모호함을 피하기 위한 컬럼 이름 규정자"를 참조하십시오.

이 양식은 fullselect(fullselect의 퇴화 양식과 다른)에서 허용되지 않음을 주의하십시오. 예를 들어, 다음은 유효하지 않습니다.

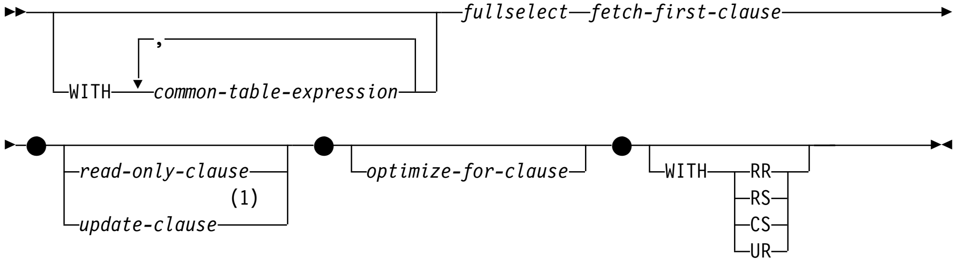
```
(SELECT C1 FROM T1
   ORDER BY C1)
UNION
SELECT C1 FROM T2
   ORDER BY ORDER OF T1
```

다음 예는 유효합니다.

```
SELECT C1 FROM
   (SELECT C1 FROM T1
      UNION
    SELECT C1 FROM T2
    ORDER BY C1 ) AS UTABLE
ORDER BY ORDER OF UTABLE
```

select문

다음은 수정된 select문의 전체 구문 다이어그램입니다.



주:

1 update절은 fullselect가 order-by절을 포함하는 경우 지정될 수 없습니다.

SELECT INTO문

구문

```
                               ┌─,─────────────┐
                               ▼               │
►►──select-clause──INTO──────host-variable─┴──from-clause──┬──────────────┬──────────►
                                                          └─where-clause─┘

►──┬──────────────────┬──┬───────────────┬──┬───────────────────┬──┬────────────────┬───►◄
   └─group-by-clause──┘  └─having-clause─┘  └─order-by-clause───┘  └─WITH──┬─RR─┬──┘
                                                                            ├─RS─┤
                                                                            ├─CS─┤
                                                                            └─UR─┘
```

OLAP 함수(window-order 절)

다음은 수정된 window-order 절을 표시하는 OLAP 함수에 대한 부분 구문 다이어그램입니다.

window-order 절:

```
           ┌─,──────────────────────────────────────────┐
           ▼                      ┌┤ asc option ├─┐     │
├──ORDER BY───┬─sort-key-expression─┼──────────────────┼───┴─────────────────────┤
              │                    └┤ desc option ├─┘  │
              └─ORDER OF──table-designator──────────────┘
```

asc 옵션:

```
        ┌─NULLS LAST──┐
├──ASC──┤             ├────────────────────────────────────┤
        └─NULLS FIRST─┘
```

desc 옵션:

```
         ┌─NULLS FIRST─┐
├──DESC──┤             ├───────────────────────────────────┤
         └─NULLS LAST──┘
```

ORDER BY (*sort-key-expression*,...)

OLAP 함수의 값을 결정하는 파티션 내에서 행의 순서를 정의하거나 window-aggregation-group 절에서 ROW 값의 의미를 정의합니다. (쿼리 결과 세트의 순서는 정의하지 않습니다.)

sort-key-expression

창 파티션 내에서 행의 순서를 정의하는 데 사용되는 표현식입니다. sort-key-expression에서 참조되는 각 컬럼 이름은 OLAP 함수를 포함하여 subselect의 결과 세트 컬럼을 명확하게 참조해야 합니다(SQLSTATE 42702 또는 42703). 각 sort-key-expression의 길이는 255바이트보다 작아야 합니다 (SQLSTATE 42907). sort-key-expression은 스칼라 fullselect를 포함할 수 없으며(SQLSTATE 42822) 결정적이지 않거나 외부 조치가 있는 모든 함수를 포함할 수 없습니다(SQLSTATE 42845). 이 절은 RANK 및 DENSE_RANK 함수에 필요합니다(SQLSTATE 42601).

ASC

오름차순으로 sort-key-expression의 값을 사용합니다.

DESC

내림차순으로 sort-key-expression의 값을 사용합니다.

NULLS FIRST

창 순서는 정렬 순서에서 모두 널(NULL)이 아닌 값 *이전*에 널(NULL) 값을 두는 것으로 고려합니다.

NULLS LAST

창 순서는 정렬 순서에서 모두 널(NULL)이 아닌 값 *이후*에 널(NULL) 값을 두는 것으로 고려합니다.

ORDER OF *table-designator*

*table-designator*에서 사용되는 동일한 순서가 subselect의 결과 테이블에 적용되어야 함을 지정합니다. 이 절을 지정하는 subselect의 FROM 절에는 *table-designator*과 일치하는 테이블 참조가 있어야 합니다(SQLSTATE 42703). 지정된 *table-designator*에 해당하는 subselect(또는 fullselect)는 데이터에 종속하는 ORDER BY 절을 포함해야 합니다(SQLSTATE 428FI SQLCODE -20210). 적용되는 순서는 중첩된 subselect(또는 fullselect)에서 ORDER BY 절의 컬럼이 외부 subselect(또는 fullselect) 절에 포함된 경우 동일하며, 이 컬럼이 ORDER OF 절 대신 지정된 경우 동일합니다. 테이블 지정자에 대한 자세한 정보는 *SQL 참조서*에서 "모호함을 피하기 위한 컬럼 이름 규정자"를 참조하십시오.

GET_ROUTINE_SAR 프로시저의 새 입력 인수

이 프로시저는 이제 루틴 텍스트가 카탈로그에서 발췌될 때 숨길지 여부에 대해서 지정하는(다음 값 중 하나 사용) 유형 INTEGER의 입력 인수인 *hide_body_flag*를 지원합니다.

0 루틴 텍스트를 그대로 두십시오. 이것은 기본값입니다.

1 루틴 텍스트가 카탈로그에서 발췌될 때 빈 본문으로 루틴 본문을 대체하십시오.

```
>>--GET_ROUTINE_SAR------------------------------------------------------->

>--(--sarblob--,--type--,--routine_name_string--+---------------------+--)----><
                                                '-,--hide_body_flag-'
```

SET INTEGRITY문에 대한 필수 권한 부여

이 명령문이 무결성 검사를 *해제*하기 위해 사용될 경우, 명령문의 권한 부여 ID 특권은 다음 중 최소한 하나를 포함해야 합니다.

- CONTROL 특권 설정
  - 지정된 테이블
  - 명령문에 의해 무결성 검사를 해제할 하위 외부 키
  - 명령문에 의해 무결성 검사를 해제할 바로 하위 요약 테이블
- SYSADM 또는 DBADM 권한
- LOAD 권한

부록 N. 예외 테이블

발견된 첫 번째 및 다음 제한조건 위반 유형을 나타내는 문자를 기술하는 "예외 테이블 메시지 컬럼 구조" 테이블의 2행과 6행에서 다음에 대한 언급이 누락되었습니다.

```
'D' - Delete Cascade violation
```

유니코드 갱신

소개

유니코드 표준은 문자 및 텍스트에 대한 공통 문자 코드화 체계입니다. 코드화 수도 적을 뿐만 아니라 문자 세트를 매우 정밀하게 정의합니다. 국제적으로 텍스트 데이터 교환을 가능케 하고 전역 소프트웨어의 기초를 만드는 복수 언어 텍스트 코드화의 일관된 방법입니다.

유니코드가 제공하는 두 코드화 체계는 UTF-16과 UTF-8입니다.

기본 코드화 체계는 UTF-16으로, 16비트 코드화 체계입니다. UCS-2는 한 문자를 표현하기 위해 2바이트를 사용하는 UTF-16의 부속 집합입니다. UCS-2는 일반적으로 모든 기존 하나 또는 2바이트 코드 페이지로부터 모든 필요한 문자를 표시할 수 있는 보편적인 코드 페이지로 인정됩니다. UCS-2는 IBM에 코드 페이지 1200으로 등록되어 있습니다.

다른 유니코드 코드화 체계는 UTF-8로, 바이트 지향으로 기본 ASCII 기반 시스템에서 사용하기 쉽게 고안되었습니다. UTF-8은 다양한 수의 바이트(보통 1-3, 때로는 4)를 사용하여 각 문자를 저장합니다. 상이하지 않은 ASCII 문자는 한 바이트로 저장됩니다. 모든 기타 문자는 여러 바이트를 사용하여 저장됩니다. 일반적으로, 복수 바이트 코드 페이지를 위해 설계되지 않은 코드는 UTF-8 데이터를 확장 ASCII 데이터로 취급합니다. UTF-8는 IBM에 코드 페이지 1208로 등록되어 있습니다.

데이터는 로컬 코드 페이지, UCS-2와 UTF-8 사이에서 변환되므로 응용프로그램은 데이터의 요구사항을 고려해야 합니다. 예를 들어, 원래의 코드 페이지 및 사용된 문자에 따라 20자는 UCS-2에서 정확히 40바이트 그리고 UTF-8에서는 20에서 60바이트 사이의 바이트가 필요합니다.

DB2 유니코드 데이터베이스 및 응용프로그램

UTF-8 코드 세트를 지정하여 작성한 Unix, Windows 또는 OS/2용 DB2 Universal Database를 사용하여 데이터를 UCS-2 및 UTF-8 형식 모두로 저장할 수 있습니다. 그런 데이터베이스를 유니코드 데이터베이스라고 합니다. SQL 문자 데이터는 UTF-8을 사용하여, 그리고 SQL 그래픽 데이터는 UCS-2를 사용하여 코드화됩니다. 이것은 1바이트 또는 2바이트 문자를 포함한 MBCS 문자가 문자 컬럼에 저장되고 DBCS 문자가 그래픽 컬럼에 저장된다는 것을 나타냅니다.

응용프로그램의 코드 페이지는 DB2가 데이터 저장에 사용하는 코드 페이지와 일치하지 않을 수 있습니다. 비유니코드 데이터베이스에서 코드 페이지가 같지 않으면, 데이터베이스 관리 프로그램은 클라이언트와 서버 사이에 전송되는 문자 및 그래픽(순수 DBCS) 데이터를 변환합니다. 유니코드 데이터베이스에서는 클라이언트 코드 페이지와 UTF-8 사이의 문자 데이터 변환이 데이터베이스 관리 프로그램에 의해 자동으로 수행되지만, 모든 그래픽(UCS-2) 데이터는 클라이언트와 서버 사이의 변환없이 전달됩니다.

데이터베이스 관리 프로그램이 수행하는 코드 페이지 변환

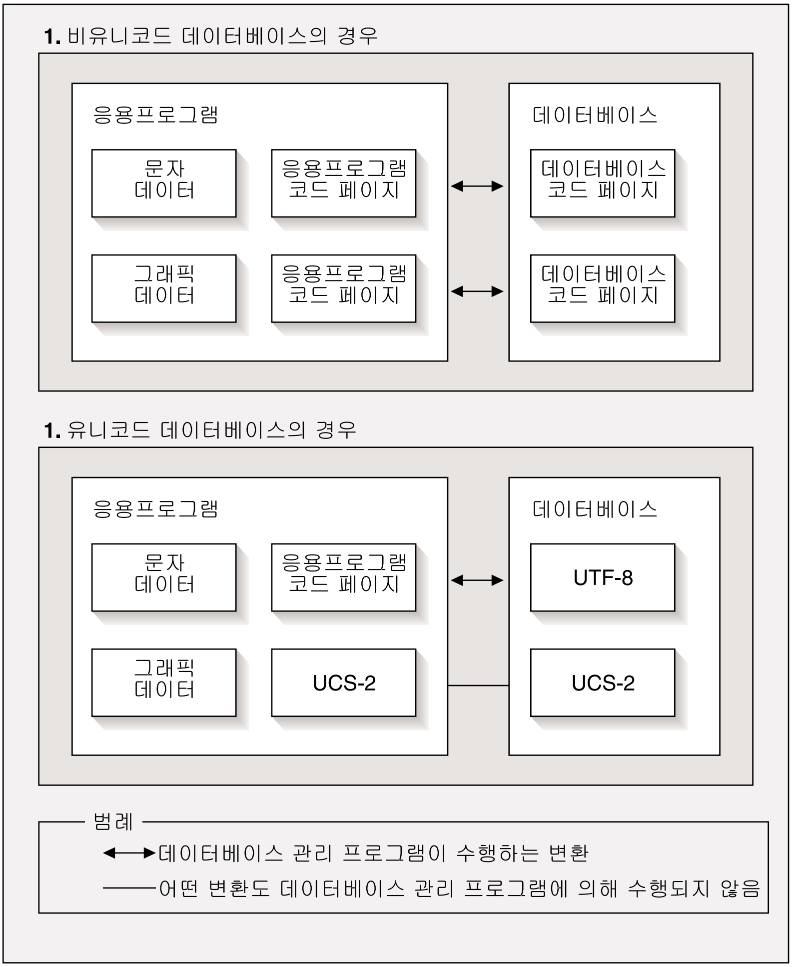


그림 1. 데이터베이스 관리 프로그램이 수행하는 코드 페이지 변환

주:

1. 유니코드 데이터베이스에 연결할 때, 응용프로그램이 DB2CODEPAGE=1208을 설정하고, 로컬 코드 페이지가 UTF-8이면 아무 코드 페이지 변환도 필요하지 않습니다.
2. 유니코드 데이터베이스에 연결될 때, CLI 응용프로그램은 문자 데이터를 그래픽 데이터로 그리고 그래픽 데이터를 문자 데이터로 수신할 수도 있습니다.

응용프로그램이 UTF-8 코드 페이지를 지정하면, 모든 그래픽 데이터를 UCS-2로 그리고 문자 데이터를 UTF-8로 송수신함을 의미할 수 있습니다. 이 응용프로그램 코드 페이지는 유니코드 데이터베이스에만 지원됩니다.

유니코드 사용시 기타 고려사항:

1. 데이터베이스 코드 페이지는 데이터베이스가 작성될 때 결정되고, 기본으로 그 값은 운영 체제 로케일(또는 코드 페이지)로부터 결정됩니다. CODESET 및 TERRITORY 키워드를 사용하여 유니코드 DB2 데이터베이스를 명시적으로 작성할 수 있습니다. 예를 들어, 다음과 같습니다.

   ```
   CREATE DATABASE unidb USING CODESET UTF-8 TERRITORY US
   ```

2. 또한 응용프로그램 코드 페이지는 로컬 코드 페이지의 기본값을 설정하지만, 이는 두 방법 중 하나로 UTF-8이 겹쳐씁니다.
   - 다음 명령으로 응용프로그램 코드 페이지를 UTF-8(1208)로 설정하십시오.

     ```
     db2set DB2CODEPAGE=1208
     ```

   - CLI/ODBC 응용프로그램의 경우, SQLSetConnectAttr()를 호출하고 SQL_ATTR_ANSI_APP를 SQL_AA_FALSE로 설정하십시오. 기본 설정 값은 SQL_AA_TRUE입니다.
3. GRAPHIC 컬럼의 데이터는 각 유니코드 문자에 대해 정확히 2바이트를 차지하는 반면, CHAR 컬럼의 데이터는 각 유니코드 문자에 대해 1에서 3바이트를 차지합니다. GRAPHIC 컬럼과 관련된 문자에서의 SQL 한계는 일반적으로 CHAR 컬럼에 대한 것의 반이지만, 바이트 면에서는 동등합니다. 예를 들어 CHAR 컬럼의 최대 문자 길이는 254이고 그래픽 컬럼의 최대 문자 길이는 127입니다. 자세한 정보는 *SQL 참조서*의 "기능" 장에서 MAX를 참조하십시오.
4. 그래픽 리터럴은 G 접두부로 인해 문자 리터럴과 차별화됩니다. 예를 들어, 다음과 같습니다.

   ```
   SELECT * FROM mytable WHERE mychar = 'utf-8 data'
                                    AND mygraphic = G'ucs-2 data'
   ```

   주: G 접두부는 유니코드 데이터베이스에 대해 선택적입니다.

   추가 정보 및 갱신된 지원은 370 페이지의『유니코드 데이터베이스의 리터럴』에서 참조하십시오.

5. CLI/ODBC 및 JDBC 응용프로그램에 대한 지원은 삽입된 응용프로그램에 대한 지원과 다릅니다. CLI/ODBC 지원에 특정한 정보는 363 페이지의 『CLI 안내서 및 참조서』를 보십시오.
6. UCS-2 데이터의 바이트 순서는 플랫폼들 간에 다를 수 있습니다. 내부적으로, DB2는 big-endian 형식을 사용합니다.

문서 갱신

이 릴리스 참고 문서에는 DB2 버전 7.1에서의 유니코드 사용에 관한 다음 갱신 정보가 들어 있습니다.

- *SQL 참조서*:

  제 3 장 언어 요소

  제 4 장 함수

  제 6 장 SQL문

- *CLI Guide and Reference*:

  제 3 장 고급 기능 사용

  부록 C DB2 CLI 및 ODBC

- *데이터 이동 유틸리티 안내 및 참조서*, 부록 C 익스포트/임포트/로드 유틸리티 파일 형식

DB2에서 유니코드 사용에 대한 자세한 정보는 *관리 안내서*, 자국어 지원(NLS) 부록: "DB2 UDB에서 유니코드 지원"을 참조하십시오.

제 7 부 호스트 시스템에 연결

DB2 Connect 사용자 안내서

DB2 Connect 데이터 전송률 향상

쿼리 결과 세트의 행 블로킹에 새로운 것이 아무 것도 없는 반면 버전 6.1 릴리스 이후의 z/OS용 DB2(이전에는 OS/390용 DB2라고 함)에는 DB2 Connect 같이 리모트 클라이언트에 대한 OPEN 또는 FETCH 요청에 응답하여 여러 쿼리 블록을 리턴하는 기능이 있습니다. 현재 클라이언트는 한 번에 하나의 행 데이터 블록을 요청하는 z/OS 서버용 DB2에 요청을 반복해 보내는 대신, 선택적으로 서버가 추가로 쿼리 블록을 되돌려보내도록 요청할 수 있습니다. 이와 같은 추가 쿼리 블록을 잉여 쿼리 블록이라 합니다.

이 새로운 기능을 이용해서 클라이언트는 네트워크 성능에 커다란 영향을 미치는 네트워크 회선 변경 수를 최소화합니다. 클라이언트가 서버에 쿼리 블록을 요청하는 수를 줄이면 성능이란 관점에서 송수신간 전환이 매우 번거로운 연산이기 때문에 상당한 수준의 성능 향상이 이루어집니다. DB2 Connect는 현재 기본적으로 z/OS 서버용 DB2에서 잉여 쿼리 블록을 요청하기 때문에 이러한 성능 향상을 저해할 수 있습니다.

기본 네트워크 프로토콜인 TCP/IP에 잉여 쿼리 블록(각 블록의 최대 크기는 32KB임)을 최대한 리턴시키기 위해서는 창 스케일링 확장자 또한 DB2 Connect의 RFC-1323에 구조화된 것처럼 사용되어야 합니다. 이 기능을 통해 TCP/IP는 동적 및 효과적으로 송수신 창 크기를 조정하여 잉여 쿼리 블록이 리턴할지도 모르는 엄청난 양을 처리할 수 있습니다.

잉여 쿼리 블록

버전 6.1 이상의 z/OS 서버용 DB2에서의 잉여 쿼리 블록 지원은 DB2 DDF 설치시 EXTRA BLOCKS SRV 매개변수를 통해 구성됩니다. 이 매개변수는 요청에 대해 DB2가 클라이언트에 되돌려 보낼 수 있고 0과 100 사이의 값으로 설정할 수 있는 최대 잉여 쿼리 블록 수를 제어합니다. 매개변수 값을 0으로 설정하면

잉여 쿼리 블록이 리턴되지 않습니다. 기본값 100을 사용해야 이 설정값을 렌더링 하는 네트워크의 어떤 특성도 차단되어 가장 효과적으로 이 기능을 활용할 수 있습니다.

응용프로그램이 직접 상호 위치한 DB2 Connect 설치를 통하거나 개별 DB2 Connect 서버 설치를 통해 z/OS용 DB2에 액세스하는 클라이언트측에는 다음 사용을 통해 커서당 또는 명령문에 기초하여 해당 DB2 Connect 지원을 활성화할 수 있는 여러 방법이 있습니다.

- 커서의 쿼리 행 세트 크기
- 커서와 연결된 SELECT문의 'N 행에 대해 최적화' 절
- 커서와 연결된 SELECT문의 'N 행만을 먼저 사전 추출' 절

옵션 1에 대해서는 이미 DB2 Connect 버전 7.1 FixPak 2에서 z/OS 스크롤 가능 지원용 DB2의 부분으로 구현했기 때문에 이 절에서는 다루지 않습니다. 여기서는 다음과 같이 다른 SQL API를 이용해서 잉여 쿼리 블록 지원을 사용하는 옵션 2와 3의 사용에 대해 집중적으로 다룰 것입니다.

1. Embedded SQL
   - SELECT문 자체의 'N 행에 대해 최적화' 절이나 'N 행만을 먼저 사전 추출' 절을 지정하여 쿼리를 지원하는 잉여 쿼리 블록을 호출합니다.
   - z/OS용 DB2는 'N 행에 대해 최적화' 절을 이용해서 DB2 Connect에 리턴시킬 원하는 행 수를 블록화하고 EXTRA BLOCKS SRV DDF 설치 매개변수로 설정할 수 있습니다. 응용프로그램은 z/OS용 DB2가 N에 설정된 쿼리 결과에 궁극적으로 리턴시키는 총 행의 수를 제한하지 않을 때 N 행을 초과하여 사전 추출할 수 있도록 선택할 수 있습니다.
   - 'N 행만을 먼저 사전 추출' 절 또한 이와 유사하게 기능합니다. 쿼리 결과 세트가 z/OS용 DB2에 의해 N 행에 제한된다는 점이 다릅니다. N 행을 초과하여 사전 추출하게 되면 결과적으로 SQL 코드 +100(데이터 끝)이 됩니다.
2. CLI/ODBC
   - SQL_MAX_ROWS 명령문 속성을 통해 쿼리에 대한 잉여 쿼리 블록 지원을 호출합니다.

- DB2 Connect는 z/OS 6.x 서버용 DB2에 대해 'N 행에 대해 최적화' 절에 태그됩니다. 궁극적으로 쿼리 결과 세트에 리턴시킬 수 있는 행 수가 z/OS용 DB2에 의해 N에 제한되지 않는다 하더라도 CLI/ODBC는 N 행을 초과하여 사전 추출하려 하는 시도가 있는 경우 SQL_NO_DATA_FOUND를 응용프로그램에 리턴합니다.
- 'N 행만을 먼저 사전 추출' 절은 z/OS 7.1 이상 서버용 DB2에 사용됩니다. Embedded SQL의 경우와 마찬가지로 쿼리 결과 세트는 z/OS용 DB2에 의해 N 행에 제한됩니다. N 행을 초과하여 사전 추출하게 되면 결과적으로 SQL_NO_DATA_FOUND가 됩니다.

3. JDBC

- setMaxRows 메소드를 통해 쿼리에 대한 잉여 쿼리 블록 지원을 호출합니다. CLI/ODBC를 사용하는 것과 마찬가지로 DB2 Connect는 z/OS 6.x 서버용 DB2에 대해 'N 행에 대해 최적화' 절에 태그되고 'N 행만을 먼저 사전 추출' 절은 z/OS 서버 버전 7.1 이상용의 DB2에 대해 태그됩니다.

RFC-1323 창 스케일링

창 스케일링은 TCP/IP에 RFC-1323 확장자를 지원하는 모든 Windows 및 Unix 플랫폼에서 FixPak 4부터 지원됩니다. 이 기능은 DB2 레지스트리 변수인 DB2SORCVBUF를 통해 Windows 및 Unix용의 DB2에서 사용할 수 있습니다. 창 스케일링을 사용하려면 DB2 레지스트리 변수인 DB2SORCVBUF를 65K 이상의 값으로 설정하십시오. (예를 들어, Windows 또는 UNIX용 DB2에서 db2set DB2SORCVBUF =65537을 발행할 수 있습니다.) 최대 송수신 버퍼 크기는 특정 운영 체제에 따라 다릅니다. 구성된 버퍼 크기가 승인되었는지 확인하기 위해 사용자는 데이터베이스 관리 프로그램 구성 매개변수인 DIAGLEVEL을 4(정보용)로 설정하고 db2diag.log 파일을 검사하여 메시지를 살펴볼 수 있습니다.

창 스케일링이 효과를 나타내게 하려면 연결 양 끝점에서 사용되어야 합니다. 예를 들어, DB2 Connect 워크스테이션과 호스트 사이에서 창 스케일링을 사용하려 한다면 이 기능은 직접 운영 체제 TCP/IP 스택을 통하거나 간접적으로 DB2 제품을 통해 워크스테이션과 호스트 양쪽 모두에서 활성화되어야 합니다. 예를 들어 z/OS용 DB2에서는 현재 TCPRCVBUFRSIZE를 65K보다 높은 값으로 설정하여 운영 체제를 통해 창 스케일링을 활성화할 수 있습니다.

DB2 Connect 서버 워크스테이션을 통해 호스트 DB2에 액세스하는데 리모트 DB2 클라이언트를 사용하는 경우에는 클라이언트에서도 창 스케일링을 사용할 수 있습니다. 어떤 호스트 DB2도 포함되지 않은 경우, 리모트 DB2 클라이언트와 워크스테이션 DB2 서버 사이에서 동일 토큰으로 창 스케일링을 사용할 수도 있습니다.

창 스케일링이 네트워크 성능을 향상시키도록 설계되기는 했지만 언제나 기대했던 만큼 네트워크 성능이 향상되지는 않습니다. Ethernet에 사용되는 프레임 크기나 토큰 링 LAN 어댑터, IP MTU 크기 및 기타 통신 링크 전반에 걸쳐 구성되어 있는 라우터 설정값 등의 인수간 상호 작용은 궁극적으로 창 스케일링이 사용될 때 성능을 떨어뜨릴 수 있습니다. 기본적으로 송수신 버퍼 모두 64K로 설정된 상태에서는 창 스케일링을 사용할 수 없습니다. 사용자는 창 스케일링을 선택할 경우에 대비하고 있어야 하며 네트워크에 맞게 필요한 조정을 할 수 있어야 합니다. 네트워크를 조정하여 네트워크의 성능을 향상시키는 방법에 대해서는 http://www.networking.ibm.com/per/per10.html의 백서를 참조하십시오.

Loosely Coupled Transaction에 대한 DB2 Connect 지원

Loosely Coupled Transactions에 대한 DB2 Connect 내의 지원은 OS/390용 DB2 버전 6 이상에 액세스하는 XA 분산 응용프로그램을 구현하는 사용자가 사용하도록 설계되었습니다. 이러한 지원을 통해 동일한 전역 트랜잭션의 서로 다른 분기에서 OS/390용 DB2에 대한 잠금 스페이스를 공유할 수 있습니다. 이 기능은 동일한 전역 트랜잭션 내의 다른 분기 때문에 분산 트랜잭션의 한 분기에 잠금 시간 종료 또는 교착 상태가 발생한 창을 줄입니다. OS/390용 DB2 버전 6은 DB2 Connect가 동일한 전역 트랜잭션의 서로 다른 분기를 처리하는 각 연결에 대해 XID를 전송하는 경우 이 상황에서 잠금 스페이스를 공유합니다.

Kerberos 지원

현재 DB2 Universal Database는 비 DRDA 환경에서 사용자를 인증하기 위한 수단으로 Kerberos 보안 프로토콜을 지원합니다. DB2/390 V7.1은 Kerberos 보안을 지원하기 시작했으므로, DB2 Connect는 Kerberos 인증을 DB2/390에 연결할 수 있도록 하기 위해 DRDA AR 기능을 추가합니다.

티켓팅 시스템을 처리하는 Kerberos 인증 레이어는 Win2K Active Directory 메커니즘으로 통합됩니다. 응용프로그램의 클라이언트 및 서버측은 각각 Kerberos SSP(Security Support Provider) 클라이언트 및 서버 모듈과 통신합니다. SSPI(Security Support Provider Interface)는 Kerberos SSP 및 기타 보안 프로토콜에 상위 레벨 인터페이스를 제공합니다.

통신 프로토콜 지원:

SNA 연결의 경우 APPC 노드를 카탈로그할 때 SECURITY=NONE을 사용해야 합니다.

일반 설치:

DB2가 Kerberos 인증을 사용하도록 구성하는 프로시저를 수행하려면 다음을 설정해야 합니다.

- 네트워크에서 공유되는 Active Directory에서 DB2의 인증 정책(서비스로서)
- KDC(Kerberos Key Distribution Centers) 간의 신뢰 관계

가장 단순한 시나리오에서는 구성해야 할 적어도 하나의 KDC 신뢰 관계가 있습니다. 즉, 클라이언트 워크스테이션을 제어하는 KDC와 OS/390 시스템 간에 관계가 형성됩니다. OS/390 R10은 호스트가 UNIX KDC로 작동하도록 하는 RACF 기능을 통해 Kerberos 티켓 처리를 제공합니다.

DB2 Connect는 평상시와 같이 라우터 기능을 3계층 설정으로 제공합니다. Kerberos 보안이 사용될 때 인증에 역할이 있다고 가정하지 않습니다. 때신 클라이언트 보안 토큰을 DB2/390에 전달합니다. 따라서 DB2 Connect 게이트웨이가 클라이언트나 호스트의 Kerberos 범위의 구성원이 될 필요가 없습니다.

Kerberos를 사용하려면 DB2 Connect 게이트웨이 모두 인증 유형 KERBEROS 와의 해당 연결을 카탈로그해야 합니다. 클라이언트는 인증 NOT_SPEC 또는 Kerberos를 통해 카탈로그할 수 있습니다. 클라이언트 및 게이트웨이에서 다른 인증 유형 조합의 결과는 sqlcode -1401(인증 유형 불일치)가 됩니다.

다운 레벨 호환성:

Kerberos 지원을 위한 DB2 요구사항:

DB2 UDB 클라이언트:
버전 7.1(OS: Win2K)

DB2 Connect:
버전 7.1 + Fix Pack 1 (OS: Any)

DB2/390:
버전 7.1

DB2/390에는 OS/390 버전 2 릴리스 10 이상에서 실행되어야 한다는 요구사항이 있습니다. 다운 레벨 DB2/390 시스템에는 DB2 Connect 버전 7.1 클라이언트로부터 연결할 때 추가적인 요구사항이 있습니다. 이러한 DB2/390 시스템은 Kerberos를 지원하지 않지만 지원되지 않는 DRDA SECMEC에 적절히 응답하지 않습니다. 이 문제점을 정정하려면 적절한 PTF를 적용하십시오.

- UQ41941(DB2/390 V5.1용)
- UQ41942(DB2/390 V6.1용)

연결성 보충 설명서

VM 환경에서 응용프로그램 서버 설정

″네트워크 정보 제공″ 섹션, ″응용프로그램 서버 정의″ 하위 섹션의 하나의 문장 다음에 다음 문장을 추가하십시오.

RDB_NAME은 SQLSTART EXEC에서 DBNAME 매개변수로 제공됩니다.

CLI/ODBC/JDBC 구성 PATCH1 및 PATCH2 설정

CLI/ODBC/JDBC 드라이버는 클라이언트 구성 지원 프로그램이나 ODBC 드라이버 관리자(시스템에 설치된 경우)를 통해 구성하거나 db2cli.ini 파일을 수동으로 편집하여 구성할 수 있습니다. *설치 및 구성 보충 설명서* 또는 *CLI Guide and Reference*에서 자세한 내용을 참조하십시오.

DB2 CLI/ODBC 드라이버 기본 활동은 db2cli.ini 파일을 통해서나 SQLDriverConnect() 또는 SQLBrowseConnect() CLI API를 통해 PATCH1 및 PATCH2 키워드 모두에 값을 지정하여 수정할 수 있습니다.

PATCH1 키워드는 사용자가 설정하려는 모든 키워드를 지정합니다. 예를 들어, 패치 1, 2 및 8이 지정된 경우, PATCH1의 값은 11입니다. 다음은 각 키워드 값의 설명과 드라이버에 미치는 영향입니다.

1 이 옵션은 드라이버에서 ″count(exp)″를 검색하여 ″count(distinct exp)″로 바꿉니다. 일부 DB2 버전은 ″count(exp)″ 구문을 지원하고, 그 구문이 일부 ODBC 응용프로그램에서 생성되기 때문에 이 옵션이 필요합니다. 서버가 ″count(exp)″ 구문을 지원하지 않을 경우 Microsoft 응용프로그램에서 필요로 합니다.

2 LITERAL_PREFIX 또는 LITERAL_SUFFIX 컬럼에 대해 SQLGetTypeInfo() 함수에 SQL_NULL_DATA가 리턴되는 경우 일부 ODBC 응용프로그램은 트랩됩니다. 그러면 드라이버가 빈 문자열을 대신 리턴하게 됩니다. Impromptu 2.0에서 필요합니다.

4 이 옵션은 시간소인에서 시간과 분수 부분이 0일 경우, 드라이버가 입력 시간소인 데이터를 날짜 데이터로 처리하도록 합니다. Microsoft Access 에서 필요합니다.

8 이 옵션은 시간소인에서 날짜 부분이 1899-12-30일 경우, 드라버가 입력 시간소인 데이터를 시간 데이터로 처리하도록 합니다. Microsoft Access 에서 필요합니다.

16 사용되지 않습니다.

32 이 옵션은 드라이버가 SQL_LONGVARCHAR, SQL_LONGVARBINARY 및 SQL_LONGVARGRAPHIC 컬럼에 대한 정보를 리턴하지 않도록 합니다. 응용프로그램에는 긴 필드가 지원되지 않는 것처럼 나타납니다. Lotus 123에서 필요합니다.

64 이 옵션은 드라이버가 그래픽 출력 문자열을 널(NULL)로 종료하도록 합니다. 2바이트 환경에서 Microsoft Access에 필요합니다.

128 이 옵션은 ″SELECT Config, nValue FROM MSysConf″ 쿼리가 서버로 전달되도록 합니다. 현재 드라이버는 관련 SQLSTATE 값 S0002(테이블 없음)를 가진 오류를 리턴합니다. 사용자가 이 구성 테이블을 데이터베이스에 작성하고 응용프로그램이 이 테이블에 액세스하도록 하려는 경우 필요합니다.

256 이 옵션은 드라이버가 SQLStatistics() 호출에서 기본 키 컬럼을 먼저 리턴하도록 합니다. 현재, 드라이버는 표준 ODBC 동작인 인덱스 이름별로 정렬된 인덱스를 리턴합니다.

512 이 옵션은 드라이버가 SQL_API_SQLTABLEPRIVILEGES 및 SQL_API_SQLCOLUMNPRIVILEGES에 대해 SQLGetFunctions()에 FALSE를 리턴하도록 합니다.

1024 이 옵션을 사용하면 실행된 UPDATE 또는 DELETE문이 어떤 행에도 영향을 미치지 못할 경우 드라이버가 SQLExecute() 또는 SQLExecDirect()에서 SQL_NO_DATA_FOUND가 아닌 SQL_SUCCESS를 리턴하게 됩니다. Visual Basic 응용프로그램에서 필요합니다.

2048 사용되지 않습니다.

4096 이 옵션은 드라이버가 자동 커미트 모드일 때 커서를 닫은 후 COMMIT 문을 실행하지 않도록 합니다.

8192 이 옵션은 드라이버가 저장 프로시저를 호출한 후 추가 결과 세트를 리턴하도록 합니다. 이 결과 세트는 저장 프로시저의 출력 값으로 구성된 한 행의 결과 세트입니다. Powerbuild 응용프로그램에서 액세스할 수 있습니다.

32768 이 옵션은 드라이버가 Microsoft Query 응용프로그램이 DB2 MVS 동의어와 함께 작동하도록 설정할 수 있도록 합니다.

65536 이 옵션 드라이버가 사실상 그래픽 리터럴인 문자 리터럴 앞에 수동으로 ″G″를 삽입하도록 합니다. 2바이트 환경에서 작업할 때 항상 이 패치를 제공해야 합니다.

131072

이 옵션은 시간소인 컬럼이 고유 인덱스의 일부일 경우 드라이버가 시간소인 컬럼을 CHAR(26)으로 기술하도록 합니다. Microsoft 응용프로그램에서 필요합니다.

262144

이 옵션은 드라이버가 SYSCAT.PROCEDURES 및 SYSCAT.PROCPARMS 테이블이 아닌 db2cli.procedures 의사 카탈로그 테이블을 사용하도록 합니다.

524288

이 옵션은 드라이버가 DB2/400 V3.x 시스템에 대한 시스템 테이블을 쿼리할 때 TABLE_SCHEMA가 아닌 SYSTEM_TABLE_SCHEMA를 사용하도록 합니다. 그 결과 성능이 향상됩니다.

1048576

이 옵션은 드라이버가 길이가 0인 문자열을 SQLPutData()를 통해 SQL_NULL_DATA로 취급하도록 합니다.

PATCH2 키워드는 PATCH1 키워드와 다릅니다. 이 경우, 쉼표 구분 문자를 사용하여 여러 패치를 지정합니다. 예를 들어, 패치 1, 4 및 5가 지정된 경우, PATCH2의 값은 ″1,4,5″입니다. 다음은 각 키워드 값의 설명과 드라이버에 미치는 영향입니다.

1 - 드라이브가 CALL문에서 저장 프로시저의 이름을 대문자로 바꾸도록 강제 수행합니다.

2 - 사용되지 않습니다.

3 - 드라이버가 스키마 호출에 대한 모든 인수를 대문자로 변환하도록 강제 수행합니다.

4 - 드라이버가 스키마 호출에 대해 버전 5 유사 결과 세트 대신 버전 2.1.2 유사 결과 세트(즉, SQLColumns(), SQLProcedureColumns() 등)를 리턴하도록 강제 수행합니다.

5 - 드라이버가 입력 VARCHAR 컬럼의 처리를 최적화하지 않도록 강제 수행합니다. 이 때, 데이터에 대한 포인터와 길이에 대한 포인터는 메모리에서 연속됩니다.

6 - 드라이버가 화면 이동이 가능한 커서가 지원되지 않는 메시지를 리턴하도록 강제 수행합니다. DB2 클라이언트가 버전 5이고 서버가 DB2 UDB 버전 5일 경우 Visual Basic 프로그램에서 필요합니다.

7 - 드라이버가 모든 GRAPHIC 컬럼 데이터 유형을 CHAR 컬럼 데이터 유형에 맵핑하도록 강제 수행합니다. 이는 2바이트 환경에서 필요합니다.

8 - 드라이버가 스키마 호출에서 카탈로그 검색 인수를 무시하도록 강제 수행합니다.

9 - 커서의 초기 닫기에 대해 커미트하지 않습니다.

10 - 사용되지 않습니다.

11 - 카탈로그 이름이 지원됨을 보고합니다(VB 저장 프로시저).

12 - 스키마 호출 인수에서 큰 따옴표를 제거합니다(Visual Interdev).

13 - db2cli.ini에서 출력 연결 문자열에 키워드를 추가하지 않습니다.

14 - SQLProcedures() 및 SQLProcedureColumns()에서 스키마 이름을 무시합니다.

15 - 문자 출력에서 소수점에 대해 항상 마침표를 사용합니다.

16 - 열 때마다 설명 정보를 리턴하도록 강제 수행합니다.

17 - 설명에서 컬럼 이름을 리턴하지 않습니다.

18 - 리터럴을 매개변수 표시문자로 대체하려고 합니다.

19 - 현재, DB2 MVS V4.1은 외부 조인 절의 ON 절에서 괄호가 허용되는 ODBC 구문을 지원하지 않습니다. 이 PATCH2를 작동시키면 IBM DB2 ODBC 드라이버는 외부 조인 절이 ODBC escape sequence에 있을 때 괄호를 제거합니다. 이 PATCH2는 DB2 MVS 4.1대해서 일때만 사용해야 합니다.

20 - 현재, MVS의 DB2는 매개변수 표시문자가 모두 피연산자인 BETWEEN 술어(expression ? BETWEEN ?)를 지원하지 않습니다. 이 패치를 작동시키면 IBM ODBC 드라이버가 술어를 (expression >= ? and expression <= ?)로 재작성하도록 만듭니다.

21 - 스토어드 프로시저의 모든 OUTPUT 전용 매개변수를 SQL_NULL_DATA로 설정합니다.

22 - 이 PATCH2는 IBM ODBC 드라이버가 외부 조인을 지원하지 않는 것으로 보고하도록 합니다. 이 옵션은 col1 길이가 254자 보다 큰 외부 조인 명령문을 사용하는 SELECT DISTICNT col1 또는 ORDER BY col1을 생성하여 DB2 UDB가 오류를 리턴하는(DB2 UDB는 254바이트 이상의 컬럼을 이 사용법에서 지원하지 않으므로) 응용프로그램에 해당됩니다.

23 - cbColDef=0과 바인드되는 매개변수에 대한 입력을 최적화하지 않습니다.

24 - 시간 값을 문자로 맵핑하기 위한 일시적인 해결책에 액세스합니다.

25 - 10진수 컬럼에 대한 일시적인 액세스 해결책 - 문자 표현에서 끝의 0을 모두 제거합니다.

26 - sqlcode 464를 응용프로그램에 리턴하지 않습니다. 이는 결과 세트가 리턴됨을 나타냅니다.
27 - 응용프로그램이 유효한 값을 지정할지라도, SQLTable이 TABLETYPE 키워드 값을 사용하도록 강제 수행합니다.
28 - 실수 컬럼을 DOUBLE 컬럼으로 설명합니다.
29 - 10진수 컬럼에 대한 ADO 일시적인 해결책 - 값 x의 선행 0을 제거합니다. 여기서 1 > x > -1(일부 MDAC 버전에만 필요)
30 - 저장 프로시저 캐싱 최적화를 사용할 수 없게 합니다.
31 - SQLStatistics 호출에서 별명에 대한 통계를 보고합니다.
32 - sqlcode -727 이유 코드 4 처리를 대체합니다.
33 - char로 바뀌었을 때 시간소인의 ISO 버전을 리턴합니다. (ODBC 버전의 반대로)
34 - CHAR FOR BIT DATA 컬럼을 CHAR로 보고합니다.
35 - SQL_DESC_BASE_TABLE_NAME이 요청되었을 때 유효하지 않은 TABLENAME을 보고합니다 - ADO 읽기전용 최적화
36 - 예약됨
37 - 예약됨

제 8 부 추가 정보

추가 정보

DB2 Everywhere는 현재 DB2 Everyplace를 의미함

DB2 Everywhere의 이름이 DB2 Everyplace로 변경되었습니다.

DB2 UDB 버전 7의 액세스 가능성 기능

DB2 UDB 제품군은 장애자들이 보다 쉽게 제품에 액세스할 수 있도록 해주는 여러 기능을 제공합니다. 다음과 같은 기능이 제공됩니다.

- 키보드 입력 및 탐색을 용이하게 하는 기능
- 표시 등록정보를 향상시키는 기능
- 오디오 및 비주얼 경보 신호에 대한 옵션
- 지원 기술과의 호환성
- 운영 체제의 액세스 가능성 기능과의 호환성
- 액세스 가능한 문서 형식

키보드 입력 및 탐색

키보드 입력

DB2 제어 센터는 키보드만을 사용하여 작동시킬 수 있습니다. 메뉴 항목과 제어는 사용자들이 제어를 활성화하거나 키보드에서 직접 메뉴 항목을 선택할 수 있게 하는 액세스 키를 제공합니다. 이들 키는 액세스 키가 나타나는 제어나 메뉴에서 해당 액세스 키에 밑줄이 표시되어 자체적으로 설명됩니다.

키보드 촛점

UNIX 기반 시스템에서 키보드 촛점의 위치가 강조표시되어 활동 중인 창의 영역과 사용자의 키 누름이 영향을 주는 위치를 나타냅니다.

액세스 가능한 표시에 대한 기능

DB2 제어 센터에는 사용자 인터페이스를 향상시키고 시력이 좋지 않은 사용자를 위해 액세스 가능성을 높이는 여러 가지 기능들이 있습니다. 이러한 액세스 가능성을 높인 기능으로는 고대비 설정 및 사용자 정의가 가능한 글꼴 등록정보에 대한 지원이 있습니다.

고대비 모드

제어 센터 인터페이스는 운영 체제에서 제공하는 고대비 모드 옵션을 지원합니다. 이 기능은 배경 색상과 전경 색상 간의 높은 대비 차가 필요한 사용자들에게 도움이 됩니다.

글꼴 설정

제어 센터 인터페이스로 사용자는 메뉴와 대화 상자 창에서 텍스트의 색상, 크기 및 글꼴을 선택할 수 있습니다.

색상에 관계 없음

이 제품의 어떤 기능을 사용하기 위해 색상을 구별할 필요는 없습니다.

대체 경보 신호

사용자는 오디오 또는 비주얼 신호를 통해 경보를 수신하도록 선택할 수 있습니다.

지원 기술과의 호환성

DB2 제어 센터 인터페이스는 Via Voice와 같은 화면 판독기 응용프로그램과 호환됩니다. 응용프로그램 모드에 있으면, 제어 센터 인터페이스는 이러한 액세스 가능성을 높인 응용프로그램에 필요한 등록정보를 제공하여 화면상의 정보를 시각 장애 사용자가 사용할 수 있게 해줍니다.

액세스 가능한 문서

DB2 제품군의 문서는 HTML 형식으로 사용할 수 있습니다. 따라서, 사용자는 브라우저에 설정된 표시 기본 설정에 따라 문서를 볼 수 있습니다. 또한, 화면 판독기 및 다른 지원 기술도 사용 가능합니다.

마우스 필요

Windows를 제외한 모든 플랫폼의 경우, 도구를 사용하려면 마우스가 필요합니다.

DB2 Run-Time Client에서 바인드 시도시 ″바인드 파일이 없습니다″ 라는 오류를 리턴

DB2 Run-Time Client에는 완전한 바인드 파일 세트가 없기 때문에 GUI 도구의 바인딩은 DB2 Run-Time Client에서 수행할 수 없으며 DB2 Administration Client에서만 수행할 수 있습니다.

검색 발견

검색 발견은 브로드캐스트 미디어에서만 지원됩니다. 예를 들어, 검색 발견은 ATM 어댑터를 통해서는 기능을 수행하지 않습니다. 그러나, 이 제한사항은 알려진 발견에는 적용되지 않습니다.

HP-UX 11에 대한 메모리 창

메모리 창은 대형 HP 64비트 머신에 있으며 32비트 응용프로그램에 대한 공유 메모리의 1.75GB보다 많은 장점을 활용하고자 하는 사용자를 위한 것입니다. DB2의 64비트 버전을 수행 중인 경우 메모리 창이 필요하지 않습니다. 메모리 창은 프로세스당 또는 프로세스 그룹당 별도로 1GB씩을 사용할 수 있게 합니다. 그러면, 인스턴스는 자체적으로 1GB의 공유 메모리와 0.75GB의 전역 메모리를 가질 수 있습니다. 사용자가 이러한 장점을 활용하려는 경우, 여러 인스턴스를 각각의 창에서 수행할 수 있습니다. 다음은 메모리 창의 사용을 위한 전제조건과 조건입니다.

- DB2 EE 환경
  - 패치: 확장 소프트웨어 12/98 및 PHKL_17795.
  - 인스턴스에 대해 $DB2INSTANCE 변수를 설정해야 합니다.
  - `/etc/services.window` 파일에 메모리 창에서 수행하고자 하는 각 DB2 인스턴스에 대한 항목이 있어야 합니다. 예를 들어, 다음과 같습니다.

```
db2instance1 50
db2instance2 60
```

주: 이름과 ID 사이에는 단 하나의 공백만 둘 수 있습니다.

– 서버에서 수행하려고 하는, 둘 이상의 단일 명령문이 필요한 DB2 명령은 TCPIP 루프백 메소드를 사용하여 수행해야 합니다. 이것은 메모리 창이 첫 번째 명령문의 처리를 완료하면 쉘이 종료되기 때문입니다. DB2 서비스는 이를 수행하는 방법을 알고 있습니다.

– 메모리 창에서 수행 중인 인스턴스에 대해 수행하려는 DB2 명령은 앞에 db2win(sqllib/bin에 위치)이 붙어야 합니다. 예를 들어, 다음과 같습니다.

```
db2win db2start
db2win db2stop
```

– 메모리 창 외부에서 수행되는 DB2 명령(그러나 메모리 창이 수행 중인 경우)은 1042를 리턴합니다. 예를 들어, 다음과 같습니다.

```
db2win db2start <== OK
db2 connect to db  <==SQL1042
db2stop <==SQL1042
db2win db2stop   <== OK
```

• DB2 EEE 환경

– 패치: 확장 소프트웨어 12/98 및 PHKL_17795.

– 인스턴스에 대해 $DB2INSTANCE 변수를 설정해야 합니다.

– DB2_ENABLE_MEM_WINDOWS 레지스트리 변수는 TRUE로 설정해야 합니다.

– /etc/services.window 파일에 메모리 창에서 수행하고자 하는 각 인스턴스의 각 논리 노드에 대한 항목이 있어야 합니다. 각 항목의 첫 번째 필드는 포트 번호로 연결된 인스턴스 이름이어야 합니다. 예를 들어, 다음과 같습니다.

```
=== db2instance1에 대한 $HOME/sqllib/db2nodes.cfg ===
5 host1 0
7 host1 1
9 host2 0

=== db2instance2에 대한 $HOME/sqllib/db2nodes.cfg ===
1 host1 0
2 host2 0
3 host2 1
```

```
=== host1에서의 /etc/services.window  ===
db2instance10 50
db2instance11 55
db2instance20 60

=== host2에서의 /etc/services.window ===
db2instance10 30
db2instance20 32
db2instance21 34
```

– DB2 명령의 앞에 EE 환경에서만 사용되는 db2win을 붙여서는 안됩니다.

DB2 DFS Client Enabler 설치 제거

DB2 DFS Client Enabler가 설치 제거되기 전에 루트는 DFS 파일을 사용하고 있지 않으며 사용자가 DFS 파일 공간에 쉘을 열어 두지 않았는지 확인해야 합니다. root 사용자로서 다음의 명령을 발행하십시오.

```
stop.dfs dfs_cl
```

/...가 더 이상 마운트되어 있지 않은지 확인하십시오.

```
mount | grep -i dfs
```

이 작업이 수행되지 않았으며 DB2 DFS Client Enabler가 설치 제거된 경우, 머신을 재부트해야 합니다.

Windows NT에서의 클라이언트 인증

Windows NT 환경에서 기존의 클라이언트 인증 메커니즘을 구현하기 위해 새 DB2 레지스트리 변수 DB2DOMAINLIST가 도입되었습니다. 이 변수는 Windows NT용 DB2 서버에서 하나 이상의 Windows NT 도메인을 정의하기 위해 사용됩니다. 이 목록에 정의된 도메인에 속한 사용자의 연결 또는 접속 요청만이 수용됩니다.

이 레지스트리 변수는 버전 7(또는 그 이상)에서 DB2 서버와 클라이언트가 수행 중인 순수한 Windows NT 도메인 환경에서만 사용해야 합니다.

*관리 안내서: 성능*의 "DB2 레지스트리 및 환경 변수" 섹션에서 이 레지스트리 변수의 설정에 대한 자세한 내용을 참조하십시오.

페더레이티드 시스템 제한사항

다음은 페더레이티드 시스템에 적용되는 제한사항입니다.

- Oracle 데이터 유형 NCHAR, NVARCHAR2, NCLOB 및 BFILE은 별명을 포함하는 쿼리에서 지원되지 않습니다.
- Create Server Option, Alter Server Option 및 Drop Server Option 명령은 제어 센터에서 지원되지 않습니다. 이러한 명령을 발행하려면, 명령행 처리기(CLP)를 사용해야 합니다.
- 별명을 포함하는 쿼리의 경우, DB2 UDB는 항상 DFT_SQLMATHWARN 데이터베이스 구성 옵션을 따르지는 않습니다. 그 대신, DB2 UDB는 DFT_SQLMATHWARN 설정에 관계없이 리모트 데이터 소스에서 직접 산술 오류나 경고를 리턴합니다.
- CREATE SERVER문은 조합 순서에서 대소문자가 구별되는 데이터 소스에 대해 COLSEQ 서버 옵션이 ‘I’로 설정되지 않도록 합니다.
- ALTER NICKNAME문은 유효하지 않은 옵션이 지정될 때 SQL0901N을 리턴합니다.
- Oracle, Microsoft SQL Server 및 Sybase 데이터 소스의 경우, 숫자 데이터 유형은 DB2의 BIGINT 데이터 유형으로 맵될 수 없습니다. 기본적으로, Oracle의 숫자(p,s) 데이터 유형(10 <= p <= 18, s = 0)은 DB2의 DECIMAL 데이터 유형에 맵핑됩니다.

MPP 파티션 테이블의 연합 한계

하나의 SQL문을 사용해서 데이터 소스에서 선택한 데이터를 DB2 페더레이티드 서버의 MPP 파티션 테이블에 직접 삽입 및 갱신 또는 삭제하려 하면 오류 SQL0901N이 표시됩니다. 연합 기능의 특성상 데이터 별명을 선택해서 MPP 파티션 테이블에 삽입할 수 없기 때문입니다.

FixPak 4 이상을 사용하고 있다면 다음 단계를 통해 데이터를 선택하여 이를 MPP 파티션 테이블에 삽입할 수 있습니다.

1. 고객 응용프로그램 환경에서 응용프로그램을 항상 연결시켜 놓을 노드를 지정할 수 있도록 DB2NODE 환경 변수를 익스포트합니다.

```
EXPORT DB2NODE=x
```

여기서, *x*는 노드 번호입니다.

2. 지정된 노드만을 포함하는 노드 그룹을 작성합니다.

```
CREATE NODEGROUP nodegroup_name ON NODE(x)
```

여기서, *x*는 노드 번호입니다.

3. 노드 그룹에 테이블 스페이스를 작성하십시오.

```
CREATE TABLESPACE tablespace_name IN NODEGROUP nodegroup_name
```

4. 테이블 스페이스에 임시 테이블을 작성하십시오.

```
CREATE TABLE temp_table_name IN tablespace_name
```

5. 응용프로그램의 INSERT 연산을 다음 두 단계로 나눕니다.
   - INSERT INTO temp_table_name SELECT * FROM nickname
   - INSERT INTO MPP_partitioned_table SELECT * from temp_table_name

INSERT 명령문을 두 개의 명령문으로 나누면 명령문 레벨 커미트와 롤백 기능이 변경됩니다. 예를 들어, 하나의 명령문이 아니라 두 개의 명령문을 롤백해야 합니다. 또한, DB2NODE 환경 변수와 연결된 노드 번호를 변경하는 경우에는 응용프로그램 패키지를 무효화한 다음 다시 바인딩해야 합니다.

다음 단계를 통해 데이터 소스에서 선택한 데이터를 MPP 파티션 테이블에 삽입할 수 있습니다. 하나의 명령문을 사용해서 데이터 소스에서 데이터를 선택한 다음 MPP 파티션 테이블의 데이터를 갱신 또는 삽입할 때 여전히 SQL0901N 오류가 표시될 것입니다. DB2 Universal Database 버전 8에서는 이 제한사항이 제거될 것입니다.

DataJoiner 제한사항

연합 환경 내에서 발행된 DR(Distributed requests)은 읽기 전용 조작으로 제한됩니다.

Windows NT용 히브리어 Information Catalog Manager

Information Catalog Manager는 히브리어로 사용 가능하며 Windows NT용 DB2 Warehouse Manager CD에 제공됩니다.

히브리어 변환은 zip 파일인 `IL_ICM.ZIP`으로 제공되며 Windows NT용 DB2 Warehouse Manager CD의 `DB2\IL` 디렉토리에 있습니다.

Information Catalog Manager의 히브리어 버전을 설치하려면, 먼저 Windows NT용 DB2 Warehouse Manager 영어 버전과 모든 전제조건을 Windows NT의 Hebrew Enabled 버전에 설치하십시오.

Windows NT용 DB2 Warehouse Manager가 설치되고 나면, `DB2\IL` 디렉토리에서 Windows NT용 DB2 Warehouse Manager가 설치된 디렉토리에 `IL_ICM.ZIP` 파일의 압축을 해제하십시오. zip 파일에 포함된 디렉토리 구조를 작성하려면 unzip 프로그램에 올바른 옵션을 제공하십시오.

파일 압축이 해제되면, 전역 환경 변수 LC_ALL을 En_US에서 Iw_IL로 변경하십시오. 설정을 변경하려면 다음을 수행하십시오.

1. Windows NT 제어판을 열고 **시스템** 아이콘을 두 번 누르십시오.
2. **시스템 등록정보** 창에서 **환경** 탭을 누른 다음, **시스템 변수** 섹션에서 **LC_ALL** 변수를 찾으십시오.
3. 변수를 눌러 **값** 편집 상자에 값을 표시하십시오. 값을 **En_US**에서 **Iw_IL**로 값을 변경하십시오.
4. **설정** 버튼을 누르십시오.
5. **시스템 등록정보** 창과 **제어판**을 닫으십시오.

이제 Information Catalog Manager의 히브리어 버전을 설치하십시오.

DB2의 SNA SPM이 Windows 부팅 후에 실패함

Microsoft SNA Server 버전 4 SP3 이상을 사용할 경우, DB2의 SNA SPM이 재부트 후에 적절하게 시작되었는지 확인하십시오. 다음과 유사한 항목에 대해 `\sqllib\<instance name>\db2diag.log` 파일을 확인하십시오.

```
2000-04-20-13.18.19.958000   Instance:DB2   Node:000
PID:291(db2syscs.exe)   TID:316   Appid:none
common_communication  sqlccspmconnmgr_APPC_init   Probe:19
SPM0453C  Microsoft SNA Server가 시작하지 않았으므로 동기점 관리 프로그램이
시작하지 않았습니다.


2000-04-20-13.18.23.033000   Instance:DB2   Node:000
PID:291(db2syscs.exe)   TID:302   Appid:none
common_communication  sqlccsna_start_listen   Probe:14
DIA3001E "SNA SPM" 프로토콜이 정상적으로 시작되지 않았습니다.


2000-04-20-13.18.23.603000   Instance:DB2   Node:000
PID:291(db2syscs.exe)   TID:316   Appid:none
common_communication  sqlccspmconnmgr_listener   Probe:6
DIA3103E APPC 프로토콜 지원에 오류가 발생했습니다. APPC 명령어 "APPC(DISPLAY 1
BYTE)". 1차 rc는 "F004"이며 2차 rc는 "00000000"입니다.
```

db2diag.log에 이러한 항목이 있고, 시간소인이 최근 재부트 시간과 일치할 경우, 다음을 수행해야 합니다.

1. db2stop을 실행하십시오.
2. SnaServer 서비스를 시작하십시오(아직 시작되지 않은 경우).
3. db2start를 실행하십시오.

db2diag.log 파일을 다시 확인하여 항목이 더 이상 추가되지 않는지 확인하십시오.

Windows NT 및 Windows 2000에서 DB2에 대한 서비스 어카운트 요구사항

Windows NT 또는 Windows 2000용 DB2를 설치하는 동안, 설치 프로그램은 여러 Windows 서비스를 작성하고, 각 서비스에 서비스 어카운트를 할당합니다. DB2가 제대로 실행되도록, 설치 프로그램이 다음 사용자 권한을 DB2 서비스와 연관된 서비스 어카운트에 부여합니다.

- 운영 체제의 일부로서 동작
- 토큰 오브젝트 작성
- 할당량 증가
- 서비스로 로그온

- 프로세스 레벨 토큰 바꾸기

DB2 서비스에 대해 다른 서비스 어카운트를 사용하려는 경우, 이들 사용자 권한을 서비스 어카운트에 부여해야 합니다.

이들 사용자 권한 뿐만 아니라, 서비스 어카운트도 DB2 제품이 설치된 디렉토리로의 쓰기 액세스 권한을 가져야 합니다.

DB2 Administration Server 서비스(DB2DAS00 서비스)에 대한 서비스 어카운트도 다른 DB2 서비스를 시작 및 중지할 권한을 가지고 있어야 하며(즉, 서비스 어카운트는 POWER Users 그룹에 속해야 함), 이것이 관리하는 DB2 인스턴스에 대해 DB2 SYSADM 권한을 갖고 있어야 합니다.

DWC(Data Warehouse Center)에서 사용될 모든 사용자 정의 프로그램에 커미트 필요

DWC(Data Warehouse Center)에서 사용자 정의 프로그램으로서 DB2 저장 프로시저가 작성한 저장 프로시저를 사용하려는 경우, 저장 프로시저에서 `con.close();` 명령문 앞에 다음 명령문을 삽입해야 합니다.

```
con.commit();
```

이 명령문이 삽입되지 않으면, DWC에서 저장 프로시저를 실행할 때 저장 프로시저가 수행한 변경사항이 롤백됩니다.

DWC에 있는 모든 사용자 정의 프로그램에 대해서는 변경사항이 데이터베이스에 적용되도록 포함된 모든 DB2 함수들을 *명시적으로* 커미트해야 합니다. 즉, COMMIT 명령문을 사용자 정의 프로그램에 추가해야 합니다.

Windows NT에서 클라이언트측 캐싱

사용자가 공유 드라이브를 사용하여 DB2 Datalinks를 설치한 Windows NT 서버 머신에 있는 READ PERM DB 파일을 유효한 토큰을 사용하여 액세스하려할 경우, 예상대로 파일이 열립니다. 하지만, 그 후에, 동일한 토큰을 사용하는 이어지는 열기 요청은 서버에 실제로 도달하지 않고, 클라이언트의 캐시로부터 서비스를 받습니다. 심지어 토큰이 만료된 후에도, 항목이 캐시에 아직 있으므로 사용

자가 파일의 내용을 계속 볼 수 있습니다. 이문제는 파일이 Windows NT 워크스테이션에 있을 경우에는 발생하지 않습니다.

솔루션은 Windows NT 서버에서 레지스트리 항목 \\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Lanmanserver\Parameters\EnableOpLocks를 영(0)으로 설정하는 것입니다. 이 레지스트리를 설정하면, 클라이언트가 공유 드라이브를 통해 서버에 상주하는 파일을 액세스할 때마다, 클라이언트 캐시에서 서비스되지 않고 요청이 항상 서버에 도달합니다. 따라서, 토큰은 모든 요청에 대해 재검증됩니다.

이 솔루션은 공유 드라이브를 통한 서버로부터의 모든 파일 액세스의 전반적인 성능에 영향을 준다는 단점이 있습니다. 이렇게 설정하더라도, 다른 클라이언트 머신으로부터의 액세스와는 달리 서버 자체에 있는 공유 드라이브 맵핑을 통해 파일을 액세스할 경우, 요청은 여전히 캐시로부터 서비스 받는 것처럼 보입니다. 따라서, 토큰 만기에 영향을 받지 않습니다.

주: 어떤 경우라도 파일 액세스가 로컬 액세스이고 공유 드라이브를 통하지 않으면, 토큰 검증 및 이어지는 토큰 만기는 예상대로 발생합니다.

Life Sciences Data Connect

새 랩퍼

FixPak 4에서 두 개의 새 랩퍼가 Life Sciences Data Connect에 추가되었습니다. 하나는 AIX의 Documentum에, 다른 하나는 Windows NT의 Excel에 추가되었습니다. 그리고, 테이블 구조 파일 랩퍼가 AIX에서 Windows NT, Solaris, Linux 및 HP-UX 시스템으로 포트되었습니다.

FixPak 5의 경우, AIX에서의 BLAST 랩퍼가 DB2 Life Sciences Data Connect에 추가되었습니다. Documentum 랩퍼가 AIX에서 Windows NT, Windows 2000 및 Solaris 운영 환경으로 포트되었습니다.

FixPak 6의 경우, BLAST 랩퍼가 AIX에서 Windows NT, Windows 2000, HP-UX 및 Solaris 운영 환경으로 포트되었습니다.

주의사항

Life Sciences Data Connect는 Apache Software 및 ICU의 코드를 포함합니다. 특정 목적과 비침해에의 적합성에 대한 묵시적 보증을 포함하여(단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증없이 이 코드를 ″현상태대로″를 제공합니다. 그리고 배상에 대한 IBM 책임은 적용되지 않습니다.

Apache Software License, 버전 1.1

ICU 1.8.1 이상

SQL 지원에 대해 개선된 기능

SQL 지원 도구는 이제 사용자가 테이블 조인에 대해 ″=″ 이외의 조인 연산자를 지정하게 합니다. SQL 지원 도구의 조인 페이지에서 조인 유형 버튼을 눌러 시작된 조인 유형 대화 상자는 조인 연산자의 드롭 다운 목록을 포함하도록 확장되었습니다.

사용 가능한 연산자는 ″=″, ″<>″, ″<″, ″>″, ″<=″ 및 ″>=″입니다. SQL 지원은 단순 SQL문 작성시 사용자를 돕는 도구입니다. Command Center(대화식 탭)에서 사용 가능하며, Control Center(뷰 작성 및 트리거 작성 대화 상자, Stored Procedure Builder(″SQL Stored Procedure 삽입″ 마법사) 및 Data Warehouse Center(SQL 프로세스 단계)에서 사용 가능합니다.

백업 및 리스토어 명령 도움말

`db2 ? backup`을 입력하면 잘못된 정보가 나타납니다. 올바른 정보는 다음과 같습니다.

```
BACKUP DATABASE database-alias [USER username [USING password]]
[TABLESPACE (tblspace-name [ {,tblspace-name} ... ])] [ONLINE]
[INCREMENTAL [DELTA]] [USE TSM [OPEN num-sess SESSIONS]] |
TO dir/dev [ {,dir/dev} ... ] | LOAD lib-name [OPEN num-sess SESSIONS]]
[WITH num-buff BUFFERS] [BUFFER buffer-size] [PARALLELISM n]
[WITHOUT PROMPTING]
```

db2 ? restore를 입력하면 잘못된 정보가 나타납니다. 올바른 정보는 다음과 같습니다.

```
RESTORE DATABASE source-database-alias { restore-options | CONTINUE | ABORT }";

restore-options:";
  [USER username [USING password]] [{TABLESPACE [ONLINE] |";
  TABLESPACE (tblspace-name [ {,tblspace-name} ... ]) [ONLINE] |";
  HISTORY FILE [ONLINE]}] [INCREMENTAL [ABORT]]";
  [{USE TSM [OPEN num-sess SESSIONS] |";
  FROM dir/dev [ {,dir/dev} ... ] | LOAD shared-lib";
  [OPEN num-sess SESSIONS]}] [TAKEN AT date-time] [TO target-directory]";
  [INTO target-database-alias] [NEWLOGPATH directory]";
  [WITH num-buff BUFFERS] [BUFFER buffer-size]";
  [DLREPORT file-name] [REPLACE EXISTING] [REDIRECT] [PARALLELISM n]";
  [WITHOUT ROLLING FORWARD] [WITHOUT DATALINK] [WITHOUT PROMPTING]";
```

″Warehouse Manager″는 ″DB2 Warehouse Manager″이어야 함

제품 화면과 제품 문서에서 나타나는 모든 문구 ″Warehouse Manager″는 ″DB2 Warehouse Manager″를 의미합니다.

부록. 주의사항

IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산권을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 사용권까지 부여하는 것은 아닙니다. 사용권에 대한 의문사항은 다음으로 문의하십시오.

135-270
서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩
한국 아이.비.엠 주식회사
고객만족센터
전화번호: 080-023-8080

2바이트(DBCS) 정보에 관한 사용권 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다. IBM은 타인의 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여(단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증없이 이 책을『현상태대로』제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 이 변경사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통고 없이 언제든지 개선 및/또는 변경할 수 있습니다.

이 정보에서 비IBM의 웹 사이트는 단지 편의상 제공된 것으로, 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트의 사용으로 인한 위험은 사용자 본인이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

(i) 독립적으로 작성된 프로그램 및 기타 프로그램(이 프로그램 포함)간의 정보 교환 (ii) 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램 사용권자는 다음 주소로 문의하십시오.

| 135-270
| 서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩
| 한국 아이. 비. 엠 주식회사
| 고객만족센터

이러한 정보는 해당 조항 및 조건(예를 들면, 사용권 지불 포함)에 따라 사용할 수 있습니다.

이 정보에 기술된 사용권 프로그램 및 사용가능한 모든 사용권 자료는 IBM이 IBM 기본 계약, IBM 프로그램 사용권 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

여기에 있는 모든 성능 데이터는 제한된 환경에서 산출된 것입니다. 따라서, 다른 운영 환경에서 얻어진 결과는 상당히 다를 수 있습니다. 일부 측정치는 개발 레벨 시스템에서 작성되었을 수 있으며, 이러한 측정치가 일반적으로 사용가능한 시스템에서도 동일하다고는 보장하지 않습니다. 더욱이, 일부 측정치는 추정을 통해 추측되었을 수도 있으므로 실제 결과는 다를 수 있습니다. 이 책의 사용자는 본인의 특정 환경에 적용할 수 있는 데이터를 검증해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 제품들을 테스트하지 않았으므로, 비IBM 제품과 관련된 성능의 정확성, 호환성 또는 배상 청구에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM이 제시하는 방향 또는 의도에 관한 모든 언급은 특별한 통지없이 변경될 수 있습니다.

이 정보에는 일상의 업무에서 사용되는 자료와 보고의 예제가 포함되어 있을 수 있습니다. 가능한 완벽하게 설명하기 위해 개인, 회사, 상표 및 제품의 이름이 예제에 들어 있습니다. 이들 이름은 모두 가공의 것이며, 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

저작권:

이 정보에는 여러 가지 운영 플랫폼에서의 프로그래밍 기법을 보여주는 소스 언어로 된 샘플 응용프로그램이 포함되어 있을 수 있습니다. 샘플 응용프로그램의 작성 기준이 된 운영 플랫폼의 응용프로그램 프로그래밍 인터페이스에 부합하는 응용프로그램을 개발, 사용, 마케팅 또는 배포를 목적으로 이들 샘플 프로그램을 복사, 수정 및 배포할 수 있으며 IBM에 대한 지불 의무는 없습니다. 이들 예제 프로그램은 모든 조건에서 철저하게 검사된 것은 아닙니다. 따라서, IBM은 이들 프로그램의 신뢰성, 서비스 가능성 또는 기능에 대해 어떠한 보증도 하지 않습니다.

각 사본이나 이들 샘플 프로그램의 일부 또는 파생본에는 다음과 같은 저작권 주의사항을 포함시켜야 합니다.

© (귀하의 회사명) (연도). 이 코드 부분은 IBM 샘플 프로그램에 나와 있습니다.
© Copyright IBM Corp. _연도 입력_. All rights reserved.

상표

별표(*)로 표시된 다음의 용어는 전세계에서 IBM의 상표입니다.

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2OS/390
BookManager CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000S/370
DataRefresher	SP
DB2	SQL/DS
DB2 ConnectDB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational	SystemView
Database Architecture	VisualAge
DRDA	VM/ESA
eNetwork	VSE/ESA
Extended Services	VTAM
FFST	WebExplorer
First Failure Support Technology	WIN-OS/2

다음 용어는 기타 회사의 상표 또는 등록상표입니다.

Microsoft, Windows 및 Windows NT는 Microsoft Corporation의 상표 또는 등록상표입니다.

Java 또는 모든 Java 관련 상표와 로고 및 Solaris는 미국 또는 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표입니다.

Tivoli 및 NetView는 미국 또는 기타 국가에서 사용되는 Tivoli Systems Inc.의 상표입니다.

UNIX는 미국 또는 기타 국가에서 X/Open Company Limited가 독점적인 사용권을 가진 등록상표입니다.

두 개의 별표(**)가 붙은 기타 회사 이름, 제품 이름 또는 서비스 이름은 해당 회사의 상표이거나 서비스표입니다.

색인

[바]

[사]

[아]

[자]

[카]

E

G

H

I

J

L

M

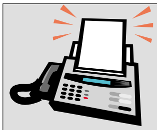
T

V

W

X

IBM 한글 지원에 관한 설문



FAX : (02) 3787-0123

보내 주시는 의견은 더 나은 고객 지원 체제를 위한 귀중한 자료가 됩니다.
독자 여러분의 좋은 의견을 기다립니다.

책 제목: IBM® DB2® Universal Database
릴리스 정보
버전 7 -- FixPak 7

성 명		직위/담당업무	
회 사 명		부 서 명	
주 소			
전화번호		팩스번호	
전자우편 주소			
사용중인 시스템	o 중대형 서버	o UNIX 서버	o PC 및 PC 서버

1. IBM에서 제공하는 한글 책자와 영문 책자 중 어느 것을 더 좋아하십니까? 그 이유는 무엇입니까?
   - ☐ 한글 책자 ☐ 영문 책자
   - (이유:)

2. 본 책자와 해당 소프트웨어에서 사용된 한글 용어에 대한 귀하의 평가 점수는?
   - ☐ 수 ☐ 우 ☐ 미 ☐ 양 ☐ 가

3. 본 책자와 해당 소프트웨어에서 번역 품질에 대한 귀하의 평가 점수는?
   - ☐ 수 ☐ 우 ☐ 미 ☐ 양 ☐ 가

4. 본 책자의 인쇄 상태에 대한 귀하의 평가 점수는?
   - ☐ 수 ☐ 우 ☐ 미 ☐ 양 ☐ 가

5. 한글 소프트웨어 및 책자가 지원되는 분야에 대해 귀하는 어떻게 생각하십니까?
   - ☐ 한글 책자를 늘려야 함 ☐ 현재 수준으로 만족
   - ☐ 그다지 필요성을 느끼지 않음

6. IBM은 인쇄물 형식(hardcopy)과 화면 형식(softcopy)의 두 종류로 책자를 제공합니다. 어느 형식을 더 좋아하십니까?
   - ☐ 인쇄물 형식(hardcopy) ☐ 화면 형식(softcopy) ☐ 둘 다

IBM 한글 지원 서비스에 대해 기타 제안사항이 있으시면 적어주십시오.

☺ 설문에 답해 주셔서 감사합니다.
귀하의 의견은 저희에게 매우 소중한 것이며, 고객 여러분들께 보다 좋은 제품을 제공해 드리기 위해 최선을 다하겠습니다.