IBM® DB2® Connect

사용자 안내서

버전 7

SA30-0993-00

IBM® DB2® Connect

사용자 안내서

버전 7

SA30-0993-00

이 책의 정보와 지원하는 제품을 사용하기 전에 반드시 253 페이지의 『부록H. 주의사항』을 읽으십시오.

이 책에는 IBM의 특허 정보가 나와 있습니다. 이 정보는 사용권 계약하에서 제공되며, 저작권법으로 보호받습니다. 이 책에 있는 정보는 어떠한 제품도 보증하지 않으며, 이 책에 제공된 어떤 내용도 이와 같이 해석되어서는 안됩니다.

책에 대한 주문은 IBM 영업 대표나 IBM 해당 지역 사무소를 통하거나 미국의 경우 1-800-879-2755번, 캐나다의 경우 1-800-IBM-4YOU 번으로 전화하십시오.

IBM으로 정보를 보내면, IBM은 적절한 방식으로 이를 사용하거나 배포할 수 있으며, 제공한 독자는 이에 대해 책임을 지지 않습니다.

© Copyright International Business Machines Corporation 1993, 2000. All rights reserved.

목차

이 책에 대하여 vii
이 책의 사용자. viii

제1부 DB2 Connect 소개 1

제1장 DB2 Connect 개요 3
데이터베이스 개념 5
DB2 Connect 설정 6
DB2 Connect와 SQL 7
관리 유틸리티 7

제2장 DRDA 개념 9
DRDA와 DB2 Connect 10
원격 작업 단위(RUOW) 11
분산 요청(DR) 13
다중 사이트 갱신 사용(2단계 확약) 14
SPM을 필요로 하는 호스트 및 AS/400 다중 사이트 갱신 시나리오. 16
DRDA와 데이터 액세스 19
제어 센터를 사용하여 다중 사이트 갱신을 사용 19
관련 DRDA 온라인 문서 21

제3장 DB2 Connect 사용 시나리오 23
직접 데이터베이스 액세스 24
연결성 서버로서의 DB2 Connect Enterprise Edition 27
DB2 Connect 및 웹 응용프로그램 29
전형적인 CGI 프로그래밍의 잇점과 한계 29
웹 서버에서의 DB2 Connect 30
jAVA 응용프로그램 서버로서의 DB2 Connect 32
Net.Data 33
IBM WebSphere 34
응용프로그램 서버와 함께 DB2 Connect 사용 36
응용프로그램 서버 솔루션 37
응용프로그램 서버 및 DB2 Connect. . . 38
DB2 Connect 및 응용프로그램 서버 구성 39
트랜잭션 처리 모니터와 함께 DB2 Connect 사용 40
TP 모니터의 예. 42
Tuxedo 및 DB2 Connect 43
X/Open 분산 트랜잭션 프로세싱(DTP) 모델 43
XA 추종 트랜잭션 관리 프로그램과 함께 DB2 Connect를 사용하는 방법 43

제4장 DB2 Connect 환경에서의 프로그래밍 47
분산 환경에서의 프로그래밍. 47
DDL 사용 49
DML 사용 49
데이터 제어 언어(DCL) 사용 51
연결과 연결해제. 51
사전 처리 컴파일 52
정렬 순서 정의 55
참조 무결성 관리 55
잠금 56
SQLCODE와 SQLSTATE의 차이 . . . 56
시스템 카탈로그 사용. 56
검색 지정의 숫자 변환 오버플로우 . . . 57
분리 레벨. 57
저장 프로시듀어. 58
NOT ATOMIC 복합 SQL. 61
DB2 Connect로 다중 사이트 갱신 . . . 62
DB2 Connect에서 지원하는 호스트나 AS/400 서버 SQL문 64
DB2 Connect에서 거부하는 호스트나 AS/400 서버 SQL문 64

© Copyright IBM Corp. 1993, 2000

OS/390용 DB2 Universal Database에서의 사용 비용 청구 계정 구현 65
사용 통계 정보를 OS/390용 DB2 서버에 전송 67
사용 통계 문자열 설정 68
유용한 책 69

제5장 사용자 응용프로그램 수행 71
데이터베이스 유틸리티 바인딩 72
CLI/ODBC 프로그램 수행 72
CLI/ODBC 액세스에 대한 플랫폼 특정 세부사항 74
자세한 구성 정보 78
Java 프로그램 수행 79
환경 구성 80
Java 응용프로그램 83
Java 애플릿 83

제2부 참조 및 문제점 해결 85

제6장 데이터베이스 디렉토리 갱신 87
정보 수집 88
노드 디렉토리 88
DCS 디렉토리 89
시스템 데이터베이스 디렉토리 99
같은 데이터베이스용 다중 항목 정의 . . 100
디렉토리 갱신 100

제7장 응용프로그램 및 유틸리티 바인딩 . . 103
BIND 명령 109
리바인딩 109

제8장 데이터베이스 시스템 모니터 111
원격 클라이언트의 연결 모니터링 111
DB2 Connect의 모니터 스위치 작동 . . 112
모니터 스위치의 상태 나열 112
GET SNAPSHOT 명령 사용 112
DCS 응용프로그램 상태 목록 115
LIST DCS APPLICATIONS 115
LIST DCS APPLICATIONS SHOW DETAIL 117
LIST DCS APPLICATIONS EXTENDED 119
DB2 제어 센터를 사용하여 확장 DCS 응용프로그램 정보 나열 121
Windows 성능 모니터 사용 122

제9장 관리 유틸리티 125
명령행 처리기 125
가져오기 및 내보내기 유틸리티 사용 . . . 126
워크스테이션에서 S/390 또는 AS/400 데이터베이스 서버로 데이터 이동 127
DRDA 서버에서 워크스테이션으로 데이터 이동 127
1바이트 및 2바이트 혼합 데이터 128
SQLQMF 유틸리티 대체 128

제10장 보안 129
인증 129
보안 유형 132
APPC 연결의 보안 유형 132
TCP/IP 연결의 보안 유형 133
보안 유형에 대한 논의 134
MVS 암호 변경 135
암호 만기 관리용으로 DB2 Connect 워크스테이션 구성 136
암호 만기 관리를 위한 호스트 구성 . . . 137
보안에 대한 추가 힌트와 추가 정보 137
확장된 보안 코드 137
TCP/IP 보안이 확인됨 138
데스크탑 ODBC 및 Java 응용프로그램 보안 138
암호 변경 지원 139

제11장 SQLCODE 맵핑 141
SQLCODE 맵핑 중단 141
SQLCODE 맵핑 조정 141

제12장 성능 147
성능 개념과 도구 147
데이터 흐름 148
병목 현상 149
벤치마킹 149
성능 도구 150
ODBC 액세스 최적화 151
응용프로그램 설계 152
복합 SQL과 저장 프로시듀어 152
요청 그룹화 154
술어 논리 154
데이터 블로킹 154
정적 SQL과 동적 SQL 156
기타 SQL 고려사항 156
DB2 Connect 조정 157
RQRIOBLK 157
DIR_CACHE 158
기타 DB2 Connect 매개변수 158
연결 풀링 159
연결 풀링의 작업 방법 160
DB2 Connect 연결 집중기 161
데이터베이스 조정 166
네트워크 조정 170
시스템 자원에 대한 경합 174
성능 문제점 해결 175
추가 SNA 성능 조정 힌트와 추가 정보 . . 175
DB2 Connect에 대한 일반적인 성능 정보 176
네트워크 접속 장치 선택과 조정 176
기타 DB2 Connect 성능 정보 소스 . . 177
ESCON을 통한 SNA의 다중 경로 채널 지원 178
NCP를 통한 DB2 Connect 연결 조정 방법 178
OSA-2 향상에 관한 정보 182
기타 정보 소스 183
기타 책 184
World Wide Web 사용 184
SNA 사용자를 위한 추가 힌트와 추가 정보 184

제13장 문제점 판별 185
기타 정보 소스 185
문제 해결 안내서 사용 185
World Wide Web 사용 185
APPC, CPI-C 및 SNA 감지 코드 문서화 186
관련 정보 수집 186
초기 연결에 성공하지 못한 경우 187
초기 연결 후 문제점이 발생했을 경우 . . . 188
진단 도구 189
추적 유틸리티(ddcstrc) 190
추적 구문 192
추적 매개변수 192
추적 출력 193
추적 출력 파일 분석 195
가장 일반적인 DB2 Connect 문제점 . . . 202
SQL0965 또는 SQL0969 203
CONNECT시 SQL1338 203
CONNECT시 SQL1403N 204
SQL5043N 205
SQL30020 206
SQL30060 206
SQL30061 207
CONNECT시 리턴 코드 119C를 수반한 SQL30073 208
리턴 코드 1을 수반한 SQL30081N . . . 208
리턴 코드 2를 수반한 SQL30081N . . . 210
리턴 코드 9를 수반한 SQL30081N . . . 210
리턴 코드 10을 수반한 SQL30081N . . 211
리턴 코드 20을 수반한 SQL30081N . . 211
리턴 코드 27을 수반한 SQL30081N . . 212
리턴 코드 79를 수반한 SQL30081N . . 212
프로토콜 특정 오류 코드 10032를 수반한 SQL30081N 213

제3부 부록 및 끝머리 215

부록A. 이전 릴리스에서 제공되는 기능 . . . 217
DB2 Connect 버전 6 릴리스 1 217
DB2 Connect 버전 5 릴리스 2 218
DB2 Connect 버전 5.0 218
DDCS 버전 2 릴리스 4 221
DDCS 버전 2 릴리스 3 221

부록B. 디렉토리 사용자 정의 워크쉬트 . . . 223

부록C. 자국어 지원에 대한 고려사항 . . . 225
문자 데이터의 변환 225

부록D. DCE 디렉토리 서비스 사용 229
데이터베이스 오브젝트 작성 230
데이터베이스 위치 지정자 오브젝트 작성 . . 233
라우팅 정보 오브젝트 작성. 235
구성 매개변수 설정 236
데이터베이스 카탈로그화 237
DCE 디렉토리 서비스에서의 보안 237

부록E. 백 레벨 클라이언트용 유틸리티 바인딩 241

부록F. CLISCHEMA 키워드를 사용한 CLI/ODBC 응용프로그램 성능 조정 . . . 243
목표 환경 243
CLI/ODBC. 244
DB2 CLISCHEMA 초기화 키워드. 244
사용시의 주의사항 245
db2cli 및 bldschem 유틸리티 246
바람직한 접근 방법 247
추가 힌트와 추가 정보 248
db2ocat 카탈로그 최적화 알고리즘 도구 . . 249
추가 정보 소스 249

부록G. 추가 및 관련 정보 소스 251
기타 관련 책 251

부록H. 주의사항 253
등록상표. 256

색인 259

IBM에 문의 269
제품 정보 269

이 책에 대하여

이 책에는 다음과 같은 IBM 제품에 관한 일반적인 사용 정보가 들어 있습니다.

- OS/2 및 Windows 32비트 운영 체제용 DB2 Connect Personal Edition
- AIX, HP-UX, Linux, PTX, Solaris, OS/2 및 Windows 32비트 운영 체제용 DB2 Connect Enterprise Edition(EE)
- OS/390용 DB2 Connect Unlimited Edition

DB2 Connect 사용자 안내서는 세 부분으로 나눠져 있습니다.

- 제1부 DB2 Connect 소개 - DB2 Connect, Distributed Relational Database Architecture(DRDA) 및 가능한 사용 시나리오에 대한 개념적인 개요를 제공합니다.
- 제2부 참조 및 문제점 해결 - 데이터베이스 디렉토리 갱신, 응용프로그램 바인딩, 관리 유틸리티, DB2 시스템 모니터, 보안, 문제점 판별 및 성능에 대한 정보를 제공합니다.
- 제3부 부록 - 기타 정보, 힌트 및 추가 정보를 제공합니다.

이 책은 DB2 Connect 제품 모두에 적용되는 개념을 설명합니다. 특정 플랫폼에 관한 내용은 다음 중 하나를 참조하십시오.

- *DB2 Connect Personal Edition 빠른 시작* - OS/2 및 Windows 32비트 운영 체제에서의 단일 사용자 DB2 Connect 설정의 경우
- *DB2 Connect Personal Edition for Linux Quick Beginnings* - Linux 에서의 단일 사용자 DB2 Connect 설정의 경우
- *OS/2 및 Windows용 DB2 Connect Enterprise Edition 빠른 시작* - OS/2 또는 Windows 32비트 운영 체제에서의 다중 사용자 DB2 Connect 게이트웨이 설정의 경우
- *UNIX용 DB2 Connect Enterprise Edition 빠른 시작* - AIX, HP-UX, Linux, PTX 또는 Solaris에서의 다중 사용자 DB2 Connect 게이트웨이 설정의 경우

© Copyright IBM Corp. 1993, 2000

이 책의 사용자

이 책은 DB2 클라이언트와 다음의 다른 Distributed Relational Database Architecture(DRDA) 응용프로그램 서버 데이터베이스 관리 시스템 사이의 DB2 Connect 연결을 설정하고 유지보수하는 프로그래머와 관리자를 대상으로 합니다.

- OS/390용 DB2 Universal Database 버전 5 이상
- MVS용 DB2 버전 3 이상
- VSE & VM용 DB2
- AS/400용 DB2 Universal Database
- DRDA 응용프로그램 서버 기능을 구현하는 기타 다른 관계형 데이터베이스 관리 시스템

주:

1. DB2 Universal Database(DB2 UDB)에서는 DB2 Connect없이도 호스트 또는 AS/400 응용프로그램으로 DB2 UDB 데이터를 액세스할 수 있습니다.
2. OS/390용 DB2 Universal Database 버전 5.1 이상은 TCP/IP 데이터베이스 연결, 다중 행 응답 세트를 갖는 저장 프로시듀어를 포함하는 DRDA 레벨 3 기능을 사용하는 데 필요합니다.
3. OS/390용 DB2 Universal Database 버전 6.1 이상은 DB2 Connect에서 제공되는 지원에 대해 DRDA 레벨 4 기능을 사용하는 데 필요합니다. 이 기능에는 큰 정수, 대형 오브젝트(LOB), 행 ID 및 사용자 정의 구별 데이터 유형에 대한 지원이 포함됩니다.

제1부 DB2 Connect 소개

© Copyright IBM Corp. 1993, 2000

제1장 DB2 Connect 개요

DB2 Connect에서는 여러 UNIX 및 비UNIX 운영 체제에서 수행 중인 e-business 및 기타 응용프로그램을 위해 IBM 메인프레임 데이터베이스에 대해 매우 신속하고 강력한 연결성을 제공합니다.

DB2 Connect에는 여러 가지 연결 솔루션이 있습니다. DB2 Connect Personal Edition은 호스트 또는 AS/400 데이터베이스에 직접적인 연결성을 제공하는 반면, DB2 Connect Enterprise Edition은 클라이언트가 DB2 Connect 서버를 통해 호스트 또는 AS/400 데이터베이스에 액세스할 수 있도록 간접적인 연결성을 제공합니다. DB2 Connect Unlimited Edition에서는 보다 쉽게 제품을 선택하고 사용권 인가를 받을 수 있는 고유의 패킹 솔루션을 제공합니다.

DB2 Connect Enterprise Edition

DB2 Connect Enterprise Edition은 다중 데스크탑 클라이언트 및 웹 응용프로그램으로부터 호스트 또는 AS/400 시스템상에 수행중인 DB2 데이터베이스 서버로의 연결을 모아서 관리하는 연결성 서버입니다. IBM의 AS/400용 DB2, OS/390용 DB2, VSE 및 VM 데이터베이스용 DB2는 여전히 거대 조직의 중요 데이터를 관리하는 데 사용되는 시스템들입니다. 이러한 호스트 및 AS/400 데이터베이스가 데이터를 관리할 경우, Windows, UNIX 및 OS/2 워크스테이션에서 수행되는 응용프로그램과 이 데이터들을 통합해야 할 필요가 있습니다.

DB2 Connect Enterprise Edition에서는 지역 및 원격 클라이언트 응용프로그램이 SQL(구조화 조회 언어), DB2 API(응용프로그램 프로그래밍 인터페이스), ODBC(Open Database Connectivity), JDBC(Java Database Connectivity), SQLJ(Java용 Embedded SQLJ) 또는 DB2 CLI(콜 레벨 인터페이스)를 사용하여 DB2 데이터베이스 및 호스트 시스템을 작성, 갱신, 제어 및 관리할 수 있습니다. 또한, DB2 Connect는 ActiveX 데이터 오브젝트(ADO), 원격 데이터 오브젝트(RDO) 및 OLE DB와 같은 Microsoft Windows 데이터 인터페이스를 지원합니다.

© Copyright IBM Corp. 1993, 2000

DB2 Connect Enterprise Edition은 AIX, HP-UX, Linux, OS/2, PTX, Solaris 및 Windows 32비트 운영 체제용으로 현재 사용 가능합니다. 이 서버에서는 OS/2, UNIX(AIX, HP-UX, Linux, PTX, Solaris, Silicon Graphics IRIX) 및 Windows 32비트 워크스테이션에서 수행 중인 응용 프로그램에 대한 지원을 제공합니다.

DB2 Connect Personal Edition

DB2 Connect Personal Edition에서는 단일 워크스테이션으로부터 OS/2, UNIX 및 Windows 32비트 운영 체제에 상주하는 DB2 Universal Database 서버 뿐만 아니라 MVS/ESA, OS/390, OS/400, VM 및 VSE 와 같은 서버상에 상주하는 DB2 데이터베이스로의 액세스를 제공합니다. DB2 Connect Personal Edition에서는 DB2 Connect Enterprise Edition 와 동일한 많은 세트의 API를 제공하고 모든 Windows 플랫폼에서의 통합 SNA 지원 기능도 제공합니다.

이 제품은 현재 OS/2, Linux 및 Windows 32비트 운영 체제용으로 사용 가능합니다.

DB2 Connect Unlimited Edition

DB2 Connect Unlimited Edition은 DB2 Connect 배치에 있어서 전반적인 유연성을 허용하고 제품 선택 및 사용권 인가를 간소화할 수 있는 고유 패키지 제공품입니다. 이 제품에는 DB2 Connect 제품을 제한없이 배치할 수 있는 사용권 기간과 조건을 가진 DB2 Connect Personal Edition 및 DB2 Connect Enterprise Edition 둘다 들어 있습니다. 사용권 수수료는 DB2 Connect 사용자가 작업하는 Systme/390의 크기에 따라 다릅니다.

이 새로운 패키지 제공품은 OS/390 시스템에서만 사용가능하며 사용권은 OS/390용 DB2 데이터 소스에 대해서만 유효합니다.

데이터베이스 개념

*데이터베이스*란 용어는 이 책 전반에 걸쳐서 관계형 데이터베이스 관리 시스템(RDBMS)을 설명하는 데 사용됩니다. DB2 Connect와 통신하는 다른 시스템에서는 데이터베이스라는 용어가 약간 다른 개념을 설명하는 데 사용됩니다. DB2 Connect 용어 데이터베이스에서는 또한 다음을 참조할 수 있습니다.

MVS (버전 4 및 그 이전)

해당 LOCATION NAME으로 식별되는 MVS/ESA용 DB2 서브시스템

LOCATION NAME은 TSO에 로그인해서, 사용 가능한 조회 도구 중 하나를 사용하여 다음의 SQL 조회를 발행함으로써 결정될 수 있습니다.

```
select current server from sysibm.sysdummy1
```

이 이름은 부트 스트랩 데이터 세트(BSDS)에 정의되며 분산 데이터 기능(DDF)이 시작될 때 작성되는 DSNL004I 메시지(LOCATION=위치)에도 정의됩니다.

OS/390 (버전 5 이상)

해당 LOCATION NAME으로 식별되는 OS/390용 DB2 Universal Database 서브시스템

LOCATION NAME은 TSO에 로그인해서, 사용 가능한 조회 도구 중 하나를 사용하여 다음의 SQL 조회를 발행함으로써 결정될 수 있습니다.

```
select current server from sysibm.sysdummy1
```

LOCATION NAME은 부트 스트랩 데이터 세트(BSDS)에 정의되며 분산 데이터 기능(DDF)이 시작될 때 작성되는 DSNL004I 메시지(LOCATION=위치)에도 정의됩니다.

VSE 해당 DBNAME으로 정의되는 파티션에서 수행하는 VSE용 DB2

VM 해당 DBNAME으로 식별되는 CMS 가상 머신에서 수행하는 VM용 DB2

OS/400

AS/400용 DB2 Universal Database, OS/400 운영 체제에 통합되어 있습니다. 하나의 데이터베이스만이 한 AS/400 머신에 존재할 수 있습니다. 데이터베이스가 AS/400 시스템 외부에서 응용프로그램에 의해 사용될

경우, 데이터베이스 이름이 관계형 데이터베이스 디렉토리에 있어야 합니다. 이 이름을 RDB 이름(관계형 데이터베이스 이름)이라고 합니다.

AS/400 시스템의 RDB 이름을 표시하려면 사용자의 AS/400에서 **WRKRDBDIRE** 명령을 실행하십시오. 지역 시스템의 RDB 이름이 원격 위치 컬럼에 *LOCAL로 지정됩니다. RDB 이름을 변경하려면 CHGRDBDIRE 명령을 사용하십시오.

DB2 Connect 설정

DB2 Connect를 사용하려면, 먼저 다음 단계를 수행하십시오.

단계 1. DB2 Connect를 설치하고 해당 *DB2 Connect 빠른 시작* 또는 *설치 및 구성 보충 설명서*에서 설명한 방식으로 호스트 또는 AS/400 서버와 워크스테이션 통신 둘다를 구성하십시오.

단계 2. 87 페이지의 『제6장 데이터베이스 디렉토리 갱신』에서 설명한 방식으로 데이터베이스 디렉토리를 갱신합니다.

주: OS/2 및 Windows 32비트 운영 체제에서는 클라이언트 구성 지원 프로그램(CCA)을 사용하는 것이 바람직합니다.

다른 모든 플랫폼에서 데이터베이스 디렉토리는 DB2 명령행 처리기(CLP)를 사용하여 갱신해야 합니다. *설치 및 구성 보충 설명서*에서 두 접근 방식에 대한 내용을 참조하십시오.

단계 3. 103 페이지의 『제7장 응용프로그램 및 유틸리티 바인딩』에서 설명한 것처럼 DB2 Connect 유틸리티를 각 호스트나 AS/400 데이터베이스 관리 시스템에 바인드하십시오.

이 타스크는 또한 CCA 또는 데이터 소스 설정 대화를 사용하여 실행할 수 있습니다.

DB2 Connect와 SQL

DB2 Connect는 응용프로그램에 의해 제출된 SQL문을 호스트나 AS/400 데이터베이스 서버로 보냅니다. DB2 Connect는 거의 모든 유효한 SQL문을 보낼 수 있습니다. 예외 사항은 64 페이지의 『DB2 Connect에서 거부하는 호스트나 AS/400 서버 SQL문』에 설명되어 있습니다.

Embedded SQL 처리에는 정적 SQL과 동적 SQL의 두 유형이 있습니다. 정적 SQL은 SQL문을 먼저 처리하여 실행시 필요한 시간을 최소화합니다. 동적 SQL은 SQL문이 호스트나 AS/400 데이터베이스 서버로 제출될 때 처리됩니다. 동적 SQL이 보다 편리하기는 하나, 잠재적으로 더 느려질 수 있습니다. 정적 SQL이나 동적 SQL의 사용 여부는 응용프로그래머가 결정합니다. 모두 DB2 Connect에 의해 지원됩니다.

서로 다른 호스트나 AS/400 데이터베이스 서버는 SQL을 다르게 구현합니다. *SQL 참조서*에서 모든 IBM 시스템이 지원하는 공통 SQL문에 대한 내용을 참조하십시오.

DB2 Connect는 OS/390용 DB2 Universal Database, MVS/ESA용 DB2, VSE & VM용 DB2(예전의 SQL/DS)와 AS/400용 DB2 Universal Database SQL뿐만 아니라 공통적인 IBM SQL을 완벽하게 지원합니다. IBM SQL은 데이터베이스 독립성 유지보수용으로 많이 권장됩니다. 47 페이지의 『제4장 DB2 Connect 환경에서의 프로그래밍』에서 자세한 내용을 참조하십시오.

관리 유틸리티

다음 유틸리티들은 DB2 Connect 관리자에게 도움이 될 것입니다.

- 명령행 처리기를 사용하여 호스트나 AS/400 데이터베이스 서버에 대해 SQL문을 발행할 수 있습니다 이 처리기는 SQL문을 지정한 데이터베이스로 보냅니다.
- DB2 명령 센터는 명령행 처리기에 대해 그래픽 인터페이스를 제공합니다.
- 가져오기 및 내보내기 유틸리티를 사용하여 워크스테이션과 호스트 또는 AS/400 데이터베이스 서버 데이터베이스상의 파일에서 데이터를 상호간에 로드, 가져오기 및 내보내기할 수 있습니다. 그리고 이 파일을 사용하여 워크스테이션에서 수행중인데이터베이스, 스프레드시트, 기타 응용프로그램으로 데이터를 가져오기할

수 있습니다. *데이터 이동 유틸리티 안내 및 참조서*에서 가져오기 및 내보내기 유틸리티에 대한 자세한 내용을 참조하십시오.

- Windows NT 및 Windows 2000에서 수행 중인 DB2 Connect Enterprise Edition 사용자들은 이벤트 표시기 및 성능 모니터를 사용할 수 있습니다. 이벤트 표시기를 사용하여 DB2 Connect가 로그한 예외 이벤트를 볼 수 있습니다. 성능 모니터를 사용하여 지역적으로 또는 원격으로 DB2 Connect 서버의 성능을 모니터 및 관리할 수 있습니다.
- DB2 제어 센터를 사용하여 DB2 Connect 서버의 모든 면을 관리하고 모니터할 수 있습니다. 또한 이를 통해 관리자가 테이블, 뷰, 버퍼 풀 및 스레드와 같은 OS/390용 DB2 데이터베이스 오브젝트에 대해 작업할 수 있습니다. *응용프로그램 개발 안내서*에서 DB2 제어 센터로부터 OS/390용 DB2 시스템을 관리하는 방법에 대한 자세한 내용을 참조하십시오.

이 유틸리티에 대해서는 125 페이지의 『제9장 관리 유틸리티』에서 자세한 내용을 참조하십시오.

또한 시스템 관리자는 데이터베이스 시스템 모니터 유틸리티를 사용하여 시스템 연결을 모니터할 수 있습니다. 또한 이 유틸리티는 시스템 관리자가 오류의 원인을 알아내는 데 도움을 줍니다. 시스템 관리자는 클라이언트 응용프로그램과 호스트 또는 AS/400 데이터베이스 서버에서 수행되는 해당 작업을 연관시킬 수 있습니다. 111 페이지의 『제8장 데이터베이스 시스템 모니터』에서 자세한 내용을 참조하십시오.

제2장 DRDA 개념

Distributed Relational Database Architecture(DRDA)는 응용프로그램 뿐만 아니라, IBM이나 IBM이 아닌 다중 데이터베이스 시스템의 공동 작업을 허용하는 프로토콜의 세트입니다. DRDA를 사용하는 관계형 데이터베이스 관리 제품의 모든 조합은 분산 관계형 데이터베이스 관리 시스템을 형성하도록 연결할 수 있습니다. DRDA는 교환될 정보와 그 교환 방식을 정의하여 시스템간의 통신을 조정합니다.

DB2 Connect에 대해 거론할 때 종종 작업 단위(UOW)라는 용어를 사용합니다. *작업 단위(UOW)*란 하나의 논리적인 트랜잭션입니다. 이 단위는 모든 조작이 성공적으로 실행되거나 아니면 순서 전체로서 실패한 것으로 생각되는 SQL문의 순서로 구성됩니다.

또다른 키 개념은 다중 사이트 갱신이라고도 하는 분산 작업 단위(DUOW)입니다. *분산 작업 단위(DUOW)*는 작업 단위(UOW) 내에 둘 이상의 데이터베이스 서버를 포함합니다. *다중 사이트 갱신*을 보다 특별하게 다음과 같은 특성을 가진 트랜잭션으로 정의합니다.

- 작업 단위(UOW) 당 둘 이상의 데이터베이스 관리 서버가 갱신됩니다.
- 응용프로그램이 작업의 분산을 지시하고 확약을 시작합니다.
- 작업 단위(UOW) 당 여러 개의 요청이 있을 수 있습니다.
- 요청 당 하나의 데이터베이스 관리 서버가 있습니다.
- 여러 데이터베이스 서버를 통해 확약을 조정합니다.

14 페이지의 『다중 사이트 갱신 사용(2단계 확약)』에서 다중 사이트 갱신에 대한 자세한 내용을 참조하십시오.

© Copyright IBM Corp. 1993, 2000

DRDA와 DB2 Connect

DB2 Connect는 DRDA 아키텍처를 구현하여 AS/400용 DB2 Universal Database, OS/390용 DB2 Universal Database, MVS/ESA용 DB2, VSE & VM용 DB2와 다른 DRDA 종속 데이터베이스 서버에 저장된 데이터를 액세스할 때 드는 비용과 복잡성을 줄여줍니다. DRDA 아키텍처를 완벽히 활용하여, DB2 Connect는 고객이 요구하는 시스템 관리 특성을 갖추어 잘 수행되고 비용이 덜드는 솔루션을 제공합니다

DRDA 용어에서, *응용프로그램 리퀘스터*(AR)는 분산 연결의 응용프로그램 쪽을 처리하는 코드이고, 데이터를 요구하는 응용프로그램입니다. *응용프로그램 서버(AS)*는 연결에 있어서 데이터베이스 쪽을 처리하는 코드입니다. DB2 Connect 환경에서 DB2 Connect 워크스테이션은 응용프로그램을 대신하여 응용프로그램 리퀘스터로서의 기능만 수행할 수 있습니다.

그림1에서 지역 클라이언트만 있는 경우 DB2 Connect 워크스테이션과 DRDA 서버 사이의 데이터 흐름을 나타냅니다 그리고, 전용 프로토콜은 DB2 Connect 워크스테이션과 다른 원격 클라이언트 사이에 존재합니다.

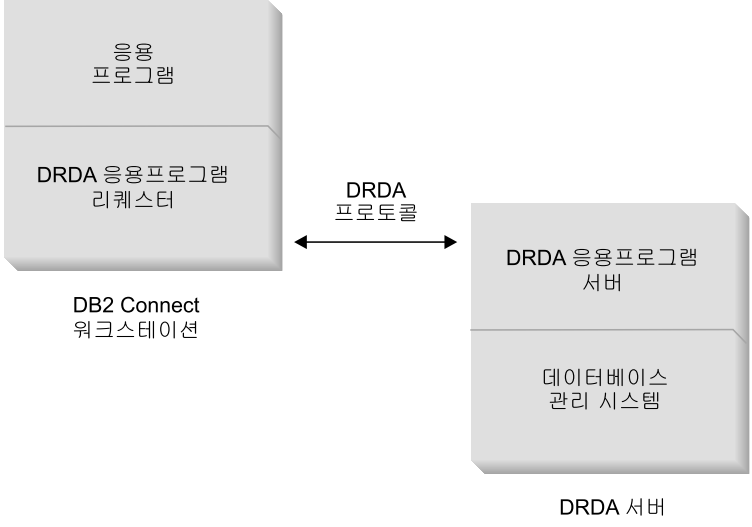


그림 1. DB2 Connect 워크스테이션과 DRDA 서버간의 데이터 흐름

DRDA 서버 데이터베이스 관리 시스템과 데이터베이스 클라이언트간의 연결을 실행하기 위해, DRDA는 다음 아키텍처를 사용합니다.

- CDRA(Character Data Representation Architecture)
- 분산 데이터 관리 아키텍처(DDM)
- 형식화된 데이터 오브젝트 내용 아키텍처(FD:OCA)
- SNA(Systems Network Architecture)
- SNA 관리 서비스 아키텍처(MSA)
- 전송 제어 프로토콜/인터넷 프로토콜(TCP/IP)

이 아키텍처들이 빌딩 블록처럼 사용했습니다. 네트워크를 흐르는 데이터 스트림은 DRDA라는 분산 관계형 데이터베이스 액세스를 지원하는 데이터 스트림 프로토콜을 기술한 아키텍처에 의해 지정됩니다.

다양한 유형의 통신 정보와 액세스될 DRDA 서버 데이터베이스 이름이 있는 디렉토리를 사용하여 요청이 올바른 목적지로 라우팅됩니다.

원격 작업 단위(RUOW)

사용자나 응용프로그램에서는 *원격 작업 단위(RUOW)*를 사용하여 각 작업 단위(UOW)마다 한 위치에서 데이터를 읽거나 갱신할 수 있습니다. 이 단위는 작업 단위 안에서 하나의 데이터베이스로의 액세스를 지원합니다. 응용프로그램이 여러 원격 데이터베이스를 갱신하는 동안에는 작업 단위 안에서 하나의 데이터베이스만 액세스할 수 있습니다.

원격 작업 단위에는 다음과 같은 특징이 있습니다.

- 작업 단위마다 다중 요청(SQL문)이 지원됩니다.
- 작업 단위마다 다중 커서가 지원됩니다.
- 각 작업 단위는 하나의 데이터베이스만 갱신할 수 있습니다.
- 응용프로그램은 작업 단위를 확약하거나 구간 복원합니다. 어떤 오류 상황에서는 데이터베이스 서버나 DB2 Connect가 구간 복원합니다.

예를 들어, 그림2에서는 은행 수수료 스케쥴 뿐만 아니라 당좌 및 저축 계좌 테이블이 포함된 데이터베이스에 액세스하는 자금 전송 응용프로그램을 수행 중인 데이터베이스 클라이언트를 보여 줍니다. 응용프로그램에서는 다음을 수행해야 합니다.

- 사용자 인터페이스로부터 전송할 양을 접수합니다.
- 저축 계좌에서 그 양을 빼서 새로운 차액 잔고를 결정합니다.
- 수수료 스케줄을 읽어서 주어진 차액 잔고를 가진 저축 계좌에 대한 트랜잭션을 결정합니다.
- 저축 계좌에서 트랜잭션 수수료를 뺍니다.
- 당좌 계좌에 전송양을 더합니다.
- 트랜잭션(작업 단위)을 확약합니다.

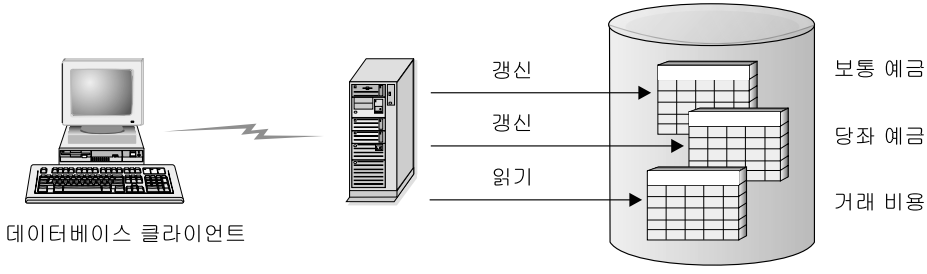


그림 2. 트랜잭션에서 단일 데이터베이스 사용

그와 같은 응용프로그램을 설정하려면 다음을 수행하십시오.

1. *관리 안내서*에 서술된대로 동일한 데이터베이스에 저축 계좌, 당좌 계좌 및 은행 수수료에 대한 테이블을 작성하십시오.
2. 물리적으로 원거리인 경우에 *빠른 시작*에 서술된대로 데이터베이스 서버를 설정하여 해당 통신 프로토콜을 사용하십시오.
3. 물리적으로 원거리인 경우에 노드와 데이터베이스를 카탈로그화하여 *빠른 시작*에 서술된대로 데이터베이스 서버상에서 데이터베이스를 식별하십시오.
4. 응용프로그램을 사전 처리 컴파일하여 유형 1 연결을 지정하십시오. 즉, *응용프로그램 개발 안내서*에 서술된대로 PREP 명령에 CONNECT(1)을 지정하십시오.

분산 요청(DR)

*분산 요청(DR)*이란 응용프로그램 및 사용자가 단일 명령문으로 둘 이상의 DBMS 또는 데이터베이스를 참조하는 SQL문을 제출할 수 있도록 하는 분산 데이터베이스 기능입니다. 예를 들어, 두 개의 서로 다른 OS/390용 DB2 서브시스템에 있는 테이블간의 결합이 해당됩니다.

DB2 Connect 버전 7에서는 데이터베이스 및 DBMS를 통한 분산 요청(DR)을 지원합니다. 예를 들어, DB2 테이블과 Oracle 뷰간의 UNION 조작을 수행할 수 있습니다. 지원되는 DBMS에는 DB2 계열의 구성원(예: Windows, UNIX 및 OS/2용 DB2 UDB, OS/390용 DB2, AS/400용 DB2) 및 Oracle이 포함됩니다.

분산 요청(DR)에서는 데이터베이스 오브젝트에 대한 *위치 투명성*을 제공합니다. 정보(테이블 및 뷰에 있는)가 이동되면 정보를 요청하는 응용프로그램을 변경시키지 않고도 해당 정보에 대한 언급(*별명*이라고 함)을 갱신할 수 있습니다. 분산 요청에서는 또한 모든 DB2 SQL 통용어 또는 특정 최적화 기능을 지원하지 않는 DBMS에 대해 *보상*을 제공합니다. 그러한 DBMS(예: 순환 SQL)하에서 수행될 수 없는 조작들은 DB2 Connect하에서 수행됩니다.

분산 요청은 *반 자치적인* 방식으로 기능합니다. 예를 들어, Oracle 오브젝트에 대한 참조가 들어 있는 DB2 조회들은 Oracle 응용프로그램들이 동일한 서버에 액세스되는 동안에 제출될 수 있습니다. 분산 요청은 Oracle 또는 다른 DBMS 오브젝트에 대한 액세스(무결성 및 잠금 제한조건을 넘어서)를 독점하거나 제한하지 않습니다.

분산 요청 기능의 구현은 DB2 Connect 버전 7 인스턴스, 연합 데이터베이스처럼 취급되는 데이터베이스 및 하나 이상의 원격 데이터 소스로 구성됩니다. *연합 데이터베이스*에는 데이터 소스 및 그의 특성을 식별하는 카탈로그 항목들이 들어 있습니다. *데이터 소스*는 DBMS 및 데이터로 구성됩니다. 응용프로그램은 다른 DB2 데이터베이스처럼 연합 데이터베이스에 연결됩니다. DB2 Connect 연합 데이터베이스는 사용자 데이터를 관리하는 용도로 인가되지 않았습니다. 그 데이터베이스의 유일한 목적은 데이터 소스 관련 정보를 포함하는 것입니다.

연합 시스템이 설정되고 나면 데이터 소스 내의 정보를 마치 하나의 큰 데이터베이스에 있는 것처럼 액세스할 수 있습니다. 사용자 및 응용프로그램에서는 하나의

연합 데이터베이스로 조회들을 전송한 후, DB2 계열 및 Oracle 시스템으로부터 필요할 때마다 데이터를 검색합니다. 사용자 및 응용프로그램은 조회에 별명을 지정합니다. 이 별명들은 데이터 소스에 위치지정된 테이블 및 뷰에 대한 참조를 제공합니다. 일반 사용자 관점에서 본다면 별명(nickname)은 별칭(aliase)과 유사합니다.

여러 인수들이 분산 요청의 성능에 영향을 미칠 수 있습니다. 가장 중요한 인수는 데이터 소스와 그의 오브젝트에 관해 정확한 최신 정보가 연합 데이터베이스 전역 카탈로그에 저장되는 것입니다. 이 정보는 DB2 최적화 알고리즘에 의해 사용되며 데이터 소스에서의 평가를 위해 조작을 밀어내는 결정에 영향을 미칠 수 있습니다. *관리 안내서: 성능*에서 연합 시스템 성능에 대한 자세한 내용을 참조하십시오.

다중 사이트 갱신 사용(2단계 확약)

분산 작업 단위(DUOW) 및 2단계 확약이라고도 하는 다중 사이트 갱신은 응용프로그램이 무결성을 잃지 않으면서 여러 개의 원격 데이터베이스 서버에 있는 데이터를 갱신할 수 있도록 하는 기능입니다. 다중 사이트 갱신의 좋은 예로, 한 계정의 돈을 다른 데이터베이스 서버에 있는 또다른 계정으로 이동하는 은행 트랜잭션이 있습니다.

이러한 트랜잭션에서는, 한 계정에 대한 대변을 처리하는 데 필요한 갱신이 확약되지 않는 한, 다른 계정에서 차변 조작을 수행하는 갱신도 확약되지 않도록 하는 것이 중요합니다. 이러한 계정을 나타내는 데이터가 두 개의 서로 다른 데이터베이스 서버에 의해 관리될 때 다중 사이트 갱신 고려사항이 적용됩니다.

DB2 제품은 다중 사이트 갱신을 위한 여러 가지 지원을 제공합니다. 이러한 지원은 X/Open XA 인터페이스 스펙을 구현하는 트랜잭션 모니터(TP 모니터) 제품을 활용하는 응용프로그램뿐 아니라 정규 SQL을 사용하여 개발된 응용프로그램에도 사용할 수 있습니다. 이러한 트랜잭션 모니터 제품으로는 IBM TxSeries(CICS 및 Encina), 메시지 및 대기행렬 시리즈, 구성요소 브로커 시리즈, Microsoft Transaction Server(MTS)뿐 아니라 San Francisco 프로젝트, BEA Tuxedo 및 기타 여러 가지가 있습니다. 원시(native) SQL 다중 사이트 갱신 또는 TP 모니터 다중 사이트 갱신 중 어떤 것을 사용하는지 따라 설정 요건이 달라집니다.

원시 SQL 및 TP 모니터 다중 사이트 갱신 프로그램은 CONNECT 2 SYNCPOINT TWOPHASE 옵션을 사용하여 사전에 컴파일되어야 합니다. 이 둘 모두 SQL Connect 명령문을 사용하여 후속 SQL문에 사용할 데이터베이스를 나타낼 수 있습니다. 트랜잭션을 조정할 것임을 DB2에 알려주는 TP 모니터가 없는 경우(데이터베이스 연결을 설정하기 위해 TP 모니터로부터 xa_open 호출을 수신하여 DB2가 표시하는 대로), DB2 소프트웨어가 사용되어 트랜잭션을 조정합니다.

TP 모니터 다중 사이트 갱신을 사용하는 경우, 응용프로그램은 TP 모니터의 API, 즉 CICS SYNCPOINT, Encina Abort(), MTS SetAbort()를 사용하여 확약 또는 구간 복원을 요청해야 합니다. 원시 SQL 다중 갱신을 사용하는 경우, 일반적인 SQL COMMIT 및 ROLLBACK을 사용해야 합니다.

TP 모니터 다중 사이트 갱신은 Oracle, Informix, SQLServer 등과 같은 DB2 및 비 DB2 자원 관리 프로그램에 모두 액세스하는 트랜잭션을 조정할 수 있습니다. 원시 SQL 다중 사이트 갱신은 DB2 서버에만 사용됩니다.

다중 사이트 갱신 트랜잭션이 작동하게 하려면, 분산 트랜잭션에 참여하는 각 데이터베이스가 분산 작업 단위(DUOW)를 지원할 수 있어야 합니다. 이 때 분산 트랜잭션에 참여할 수 있도록 하는 DUOW 지원이 다음과 같은 DB2 서버에 제공되어 있습니다.

- DB2 Universal Database V5, V6 및 V7
- MVS/ESA용 DB2 V3.1 및 4.1
- OS/390용 DB2 V5.1
- OS/390용 DB2 Universal Database V6.1
- DB2/400 V3.1 이상 (이때에만 SNA 사용)
- DB2 Server for VM 및 VSE DB2 서버 V5.1 및 6.1
- 데이터베이스 서버 4

분산 트랜잭션은 지원되는 데이터베이스 서버의 어떤 조합도 갱신할 수 있습니다. 예를 들어, 응용프로그램은 단일 트랜잭션 내에서 Windows NT의 DB2 Universal Database, OS/390용 DB2 데이터베이스 및 DB2/400 데이터베이스 모두에 있는 여러 테이블들을 갱신할 수 있습니다.

SPM을 필요로 하는 호스트 및 AS/400 다중 사이트 갱신 시나리오

호스트 및 AS/400 데이터베이스 서버에서는 DB2 Connect가 PC, UNIX 및 웹 응용프로그램에서 발생된 분산 트랜잭션에 참여해야 합니다. 또한 호스트와 AS/400 데이터베이스 서버가 관련된 여러 다중 사이트 갱신 시나리오에서는 동기 지점 관리 프로그램(SPM) 구성요소가 구성되어야 합니다.

SPM의 필요 여부는 선택된 프로토콜(SNA 또는 TCP/IP)과 사용되는 TP 모니터에 의해 결정됩니다. 다음 테이블에서 SPM을 사용해야 하는 시나리오에 대한 요약 내용을 참조하십시오. 테이블에서는 Intel 또는 UNIX 머신에서 호스트나 AS/400으로의 액세스에 DB2 Connect가 필요하다는 점을 보여 줍니다. 또한, 다중 사이트 갱신에서 SNA를 통해 액세스되거나 TP 모니터가 액세스에 사용되는 경우 DB2 Connect의 SPM 구성요소가 필요합니다.

표 1. SPM을 필요로 하는 호스트와 AS/400 다중 사이트 갱신 시나리오

TP 모니터가 사용됩니까?	**프로토콜**	**SPM이 필요합니까?**	**필요한 제품(한 개 선택)**	**지원되는 호스트와 AS/400 데이터베이스**
예	TCP/IP	예	• DB2 Connect Enterprise Edition • DB2 Universal Database Enterprise Edition • DB2 Universal Database Enterprise-Extended Edition	• OS/390용 DB2 V5.1 • OS/390용 DB2 Universal Database V6.1

표 1. SPM을 필요로 하는 호스트와 AS/400 다중 사이트 갱신 시나리오 (계속)

TP 모니터가 사용됩니까?	프로토콜	**SPM이 필요합니까?**	필요한 제품(한 개 선택)	지원되는 호스트와 **AS/400** 데이터베이스
예	SNA	예	• DB2 Connect Enterprise Edition* • DB2 Universal Database Enterprise Edition* • DB2 Universal Database Enterprise-Extended Edition* **주:** *AIX, OS/2, Windows NT 및 Windows 2000 플랫폼만.	• MVS/ESA용 DB2 V3.1 및 4.1 • OS/390용 DB2 V5.1 • OS/390용 DB2 Universal Database V6.1 • DB2/400 V3.1 이상 • VM 또는 VSE용 DB2 서버 V5.1 이상
아니오	TCP/IP	아니오	• DB2 Connect Personal Edition • DB2 Connect Enterprise Edition • DB2 Universal Database Enterprise Edition • DB2 Universal Database Enterprise-Extended Edition	• OS/390용 DB2 V5.1 • OS/390용 DB2 Universal Database V6.1

표 1. SPM을 필요로 하는 호스트와 AS/400 다중 사이트 갱신 시나리오 (계속)

TP 모니터가 사용됩니까?	**프로토콜**	**SPM이 필요합니까?**	**필요한 제품(한 개 선택)**	**지원되는 호스트와 AS/400 데이터베이스**
아니오	SNA	예	• DB2 Connect Enterprise Edition* • DB2 Universal Database Enterprise Edition* • DB2 Universal Database Enterprise-Extended Edition* **주:** *AIX, OS/2, Windows NT 및 Windows 2000 플랫폼만.	• MVS/ESA용 DB2 V3.1 및 4.1 • OS/390용 DB2 V5.1 • OS/390용 DB2 Universal Database V6.1 • DB2/400 V3.1 이상 • VM 및 VSE용 DB2 서버 V5.1 이상

주: 분산 트랜잭션은 지원되는 데이터베이스 서버의 어떤 조합도 갱신할 수 있습니다. 예를 들어, 응용프로그램은 단일 트랜잭션 내에서 Windows NT의 DB2 UDB, OS/390용 DB2 및 DB2/400 데이터베이스에 있는 여러 테이블을 갱신할 수 있습니다.

다음 책에서 여러 TP 모니터의 설정에 대한 명령어 뿐만 아니라 2단계 확약에 대한 자세한 내용을 참조하십시오.

- *관리 안내서*
- *CICS for AIX Customization and Operation Guide*

다음의 월드 와이드 웹에 있는 DB2 제품 및 서비스 기술 라이브러리에 액세스할 수도 있습니다.

1. 웹 페이지 http://www.ibm.com/software/data/db2/library/로 가십시오.

2. **DB2 Universal Database** 링크를 선택하십시오.
3. 검색 키워드 ″DDCS″, ″SPM″, ″MTS″, ″CICS″ 및 ″ENCINA″를 사용하여 ″Technotes″를 검색하십시오.

DRDA와 데이터 액세스

DRDA는 데이터베이스 통신 프로토콜을 정의하지만, 프로그래밍 인터페이스나 API를 정의하지 않으며 반드시 응용프로그래머가 사용해야 합니다. 일반적으로, DRDA는 응용프로그램이 사용할 수 있으며, 목표 DRDA 서버가 실행 가능한 요청을 모두 전달할 수 있습니다. 요즘 사용 가능한 DRDA 서버에서는 모두 DB2 Connect를 통해 응용프로그램에 의해 전송된 SQL 요청을 실행할 수 있습니다.

IBM은 응용프로그래머에게 Windows, OS/2 및 여러 UNIX 플랫폼에 대한 SQL 요청을 생성하는 도구를 제공합니다. 이런 도구는 DB2 응용프로그램 개발 클라이언트의 일부입니다. DB2 응용프로그램 개발 클라이언트는 여러 가지의 API 유형인 Embedded SQL, JDBC, SQLJ 및 DB2 콜 레벨 인터페이스(DB2 CLI)를 지원합니다. 이 API들은 프로그래머가 다양한 프로그래밍 언어로 응용프로그램를 빌드하는 데 사용할 수 있습니다. *응용프로그램 빌드 안내서*에서 이러한 API에 대한 자세한 내용을 참조하십시오.

응용프로그램 개발자는 또한 다른 회사에서 제공하는 API를 사용할 수 있습니다. 예를 들어, Microsoft ODBC 및 ADO는 Windows 응용프로그래머들이 데이터베이스 응용프로그램을 개발하는 데 사용할 수 있습니다. DB2 Connect는 ODBC 및 ADO API를 사용하여 개발된 응용프로그램을 지원하는 ODBC 드라이버 및 OLE DB Provider를 제공합니다. IBM은 ODBC 응용프로그램 개발을 위한 도구를 제공하지 않으며, 이런 도구는 Microsoft사에서 제공합니다.

제어 센터를 사용하여 다중 사이트 갱신을 사용

버전 6 이후에는 제어 센터를 사용하여 다중 사이트 갱신을 제공할 수 있습니다. 프로시듀어는 간단하며 다음과 같은 윤곽을 가집니다. 온라인 *연결성 보충 설명서*에서 시스템을 수동으로 구성하는 방법을 포함하여 다중 사이트 갱신 구성 프로세스에 대한 자세한 내용을 참조하십시오.

다중 사이트 갱신 마법사 시작

제어 센터에서, [+] 부호를 클릭하여 트리 뷰를 확장하십시오. 오른쪽 마우스 버튼으로 구성하려는 인스턴스를 선택하십시오. 팝업 메뉴가 열립니다. **다중 사이트 갱신 —> 구성** 메뉴 항목을 선택하십시오.

마법사 단계

마법사에서는 공책과 같은 형태의 인터페이스를 제공합니다. 각 마법사 페이지에서 사용자 구성에 대한 특정 정보를 프롬프트합니다. 다음의 페이지들은 사용자에게 보여지는 순서대로 표시됩니다.

단계 1. 트랜잭션 프로세서(TP) 모니터를 지정하십시오.

이 필드는 사용 가능한 TP 모니터에 대한 기본값들을 보여 줍니다. TP 모니터를 사용하지 않으려면 **TP 모니터 사용 안함**을 선택하십시오.

단계 2. 사용할 통신 프로토콜을 지정하십시오.

단계 3. 트랜잭션 관리 프로그램 데이터베이스를 지정하십시오.

이 패널의 기본값은 사용자가 연결하는 첫번째 데이터베이스(1ST_CONN)입니다. 이 기본값을 그대로 두거나 카타로그화된 다른 데이터베이스를 선택할 수 있습니다.

단계 4. 갱신에 포함된 데이터베이스 서버의 유형 및 TCP/IP를 독점적으로 사용할 것인지 여부를 지정하십시오.

단계 5. 동기 지점 관리 프로그램 설정값을 지정하십시오.

이 페이지는 이전 페이지의 설정값이 다중 사이트 갱신 시나리오에서 DB2의 동기 지점 관리 프로그램을 사용해야 한다고 지정하는 경우에만 나타납니다.

다중 사이트 갱신 기능 테스트

단계 1. 오른쪽 마우스 버튼으로 인스턴스를 선택한 후 팝업 메뉴에서 **다중 사이트 갱신 —> 테스트** 메뉴 옵션을 선택하십시오. 다중 사이트 갱신 테스트 창이 열립니다.

단계 2. **사용 가능** 목록 박스의 사용 가능한 데이터베이스에서 테스트할 데이터베이스를 선택하십시오. 중간에 있는 화살표 버튼을 사용하여 **선택 항목** 목록 박스 사이에서 선택사항을 이동시키십시오. **선택 항목** 목록 박스에서 선택된 사용자 ID와 암호를 직접 편집하여 이들을 변경할 수도 있습니다.

단계 3. 선택을 모두 마쳤으면 창의 맨아래에 있는 **테스트** 버튼을 클릭하십시오. 다중 사이트 갱신 테스트 결과 창이 열립니다.

단계 4. 다중 사이트 갱신 테스트 결과 창에서는 사용자가 선택한 데이터베이스 중 어느 것이 갱신 테스트에 성공했는지 또는 실패했는지를 보여줍니다. 창에서는 실패한 데이터베이스에 대해 SQL 코드와 오류 메시지도 표시합니다.

관련 DRDA 온라인 문서

다음 온라인 문서에는 DRDA에 관련된 유용한 정보가 들어 있습니다.

AS/400 경우:

http://www.as400.ibm.com/db2/v4r4book.htm

OS/390 경우:

http://www.ibm.com/software/data/db2/os390/library.html

DataJoiner 경우:

http://www.ibm.com/software/data/datajoiner/library.html

데이터베이스/데이터 관리 온라인 문서 경우:

http://www.ibm.com/software/data/pubs/

제3장 DB2 Connect 사용 시나리오

DB2 Connect는 호스트 또는 AS/400 데이터베이스 액세스 요구사항에 대한 다양한 솔루션을 제공할 수 있습니다. 이 절에서는 사용자의 특정 요구 또는 환경에 적용할 수 있는 여러 가지 시나리오를 대략적으로 작성해 보겠습니다.

*DB2 Connect Personal Edition*은 단일 Windows 32–비트 시스템, Linux 또는 OS/2 워크스테이션을 S/390 또는 AS/400 데이터베이스에 연결하는 데 사용됩니다. DB2 Connect Personal Edition은 데이터베이스 서버가 원시(native) TCP/IP 지원을 제공하고 전개될 응용프로그램이 전형적인 2단 클라이언트 서버 응용프로그램인 환경에 가장 적합합니다.

예를 들어, DB2 Connect Personal Edition을 선택하면 전형적인 2단 VisualBasic 및 Microsoft 액세스 응용프로그램을 사용하기가 좋습니다. 미들 타이어(mid-tier) 응용프로그램 서버가 필요한 응용프로그램들은 DB2 Connect Enterprise Edition을 사용해야 합니다. 24 페이지의 『직접 데이터베이스 액세스』에서 DB2 Connect Personal Edition을 사용한 배치 시나리오에 대해 자세한 내용을 참조하십시오.

*DB2 Connect Enterprise Edition*은 DB2 클라이언트를 호스트 또는 AS/400 데이터베이스에 연결하기 위해 종종 중간 서버에 설치됩니다. 다중 지역 사용자들이 호스트나 AS/400 서버에 직접 액세스하려는 머신에서도 이를 사용할 수 있습니다.

예를 들어, DB2 Connect Enterprise Edition은 지역 사용자들이 많이 있는 대형 머신에 설치할 수도 있습니다. 이것은 또한 웹 서버, 트랜잭션 프로세서(TP) 모니터 또는 다중 지역 SQL 응용프로그램 프로세스/스레드가 있는 기타 3 계층 응용프로그램 서버에도 설치할 수 있습니다. 이 경우, 간단하게는 동일한 머신에, 또는 CPU 사이클의 부하를 덜어주기 위해 별도의 머신에, DB2 Connect Enterprise Edition을 설치할 수 있습니다.

DB2 Connect Enterprise Edition은 다음과 같은 환경에 가장 적합합니다.

© Copyright IBM Corp. 1993, 2000

- 호스트 및 AS/400 데이터베이스 서버는 원시 TCP/IP 연결을 지원하지 않으며 SNA를 통한 데스크탑 워크스테이션으로부터의 직접 연결은 바람직하지 않습니다. 27 페이지의 『연결성 서버로서의 DB2 Connect Enterprise Edition』에서 자세한 내용을 참조하십시오
- 웹 서버는 웹 기반 응용프로그램을 수행합니다. 29 페이지의 『DB2 Connect 및 웹 응용프로그램』에서 자세한 내용을 참조하십시오
- 웹 서버는 data-aware Java 애플릿을 사용하는 웹 기반 응용프로그램을 수행합니다.
- 미들 타이어(middle-tier) 응용프로그램 서버가 사용됩니다. 36 페이지의 『응용프로그램 서버와 함께 DB2 Connect 사용』에서 자세한 내용을 참조하십시오
- CICS, Encina, Microsoft Transaction Server(MTS), Tuxedo, Component Broker 및 MQSeries와 같은 TP 모니터가 사용됩니다. 40 페이지의 『트랜잭션 처리 모니터와 함께 DB2 Connect 사용』에서 자세한 내용을 참조하십시오

*DB2 Connect Unlimited Edition*은 DB2 Connect 배치에 있어서 전반적인 유연성을 허용하고 제품 선택 및 사용권 인가를 간소화할 수 있는 고유 패키지 제공품입니다. 이 제품에는 DB2 Connect 제품을 제한없이 배치할 수 있는 사용권 기간과 조건을 가진 DB2 Connect Personal Edition 및 DB2 Connect Enterprise Edition 둘다 들어 있습니다. 사용권 수수료는 DB2 Connect 사용자가 작업하는 Systme/390의 크기에 따라 다릅니다. 이 새로운 패키지 제공품은 OS/390 시스템에서만 사용가능하며 사용권은 OS/390용 DB2 데이터 소스에 대해서만 유효합니다.

직접 데이터베이스 액세스

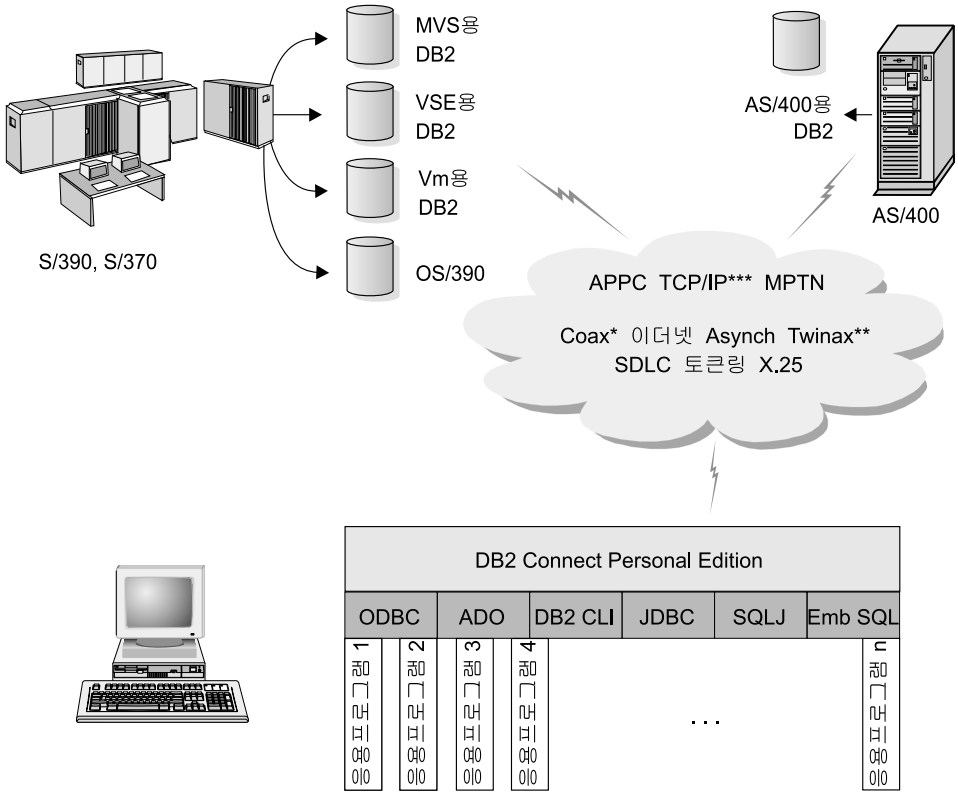
DB2 Connect의 기본 기능에서는 Windows 32비트 시스템, Linux 또는 OS/2 워크스테이션에서 수행 중인 데스크탑 기본으로부터 호스트 데이터베이스로의 직접 연결을 제공합니다. DB2 Connect Personal Edition이 이 솔루션을 구현하는 가장 단순한 방법입니다.

DB2 Connect Personal Edition이 설치된 각 워크스테이션에서는 OS/390용 DB2, DB2/400 그리고 Windows NT, Windows 2000, UNIX 및 OS/2 서버용 DB2 UDB에 대한 직접적인 TCP/IP 연결을 설정할 수 있습니다. 또한, 응용프로그램에

서는 2단계 확약 프로토콜에 의해 제공되는 완전 데이터 무결성 상태의 동일한 트랜잭션으로 여러 개의 DB2 계열 데이터베이스들을 연결 및 갱신할 수 있습니다.

DB2 Connect Personal Edition에는 또한 통합 APPC 지원이 있어 MVS용 DB2 그리고 APPC를 요청하는 다른 호스트 및 AS/400 데이터베이스와 통신합니다. 그러나, 원시 TCP/IP 지원이 사용 가능한 경우에는 SNA 대신 TCP/IP를 사용할 것을 강력히 권장합니다.

26 페이지의 그림3에서는 호스트 또는 AS/400 데이터베이스 서버에 직접 연결된 워크스테이션을 보여줍니다. 각 워크스테이션에는 DB2 Connect Personal Edition이 설치되어 있습니다.



모든 프로토콜이 모든 플랫폼을 지원하지는 않습니다.

* 호스트 연결용
** AS/400용
*** TCP/IP 연결은 OS/390 V5R1용 DB2, AS/400 V4R2용 DB2 또는 VM V6.1용 DB2를 필요로 합니다

그림 3. DB2 Connect와 호스트 또는 AS/400 데이터베이스 서버간의 직접 연결

주:

1. 어느 DRDA AR과 그의 각 호스트 및 AS/400 DRDA에서 어떤 프로토콜이 지원되는가에 대한 자세한 내용을 해당 *DB2 Connect 빠른 시작* 책에서 참조하십시오.
2. DB2 Connect 워크스테이션에 DB2 Universal Database를 설치하지 않아도 됩니다. DB2 Connect 워크스테이션에서 완전한 관계형 데이터베이스 관리 시스템을 원하면, DB2 Universal Database를 주문하십시오.
3. DB2 응용프로그램 개발 클라이언트는 이제 DB2 Connect 패키지의 일부분이며 고객이 응용프로그램 개발에 이를 사용하려는 경우에는 설치할 수 있습니다. 또한, DB2

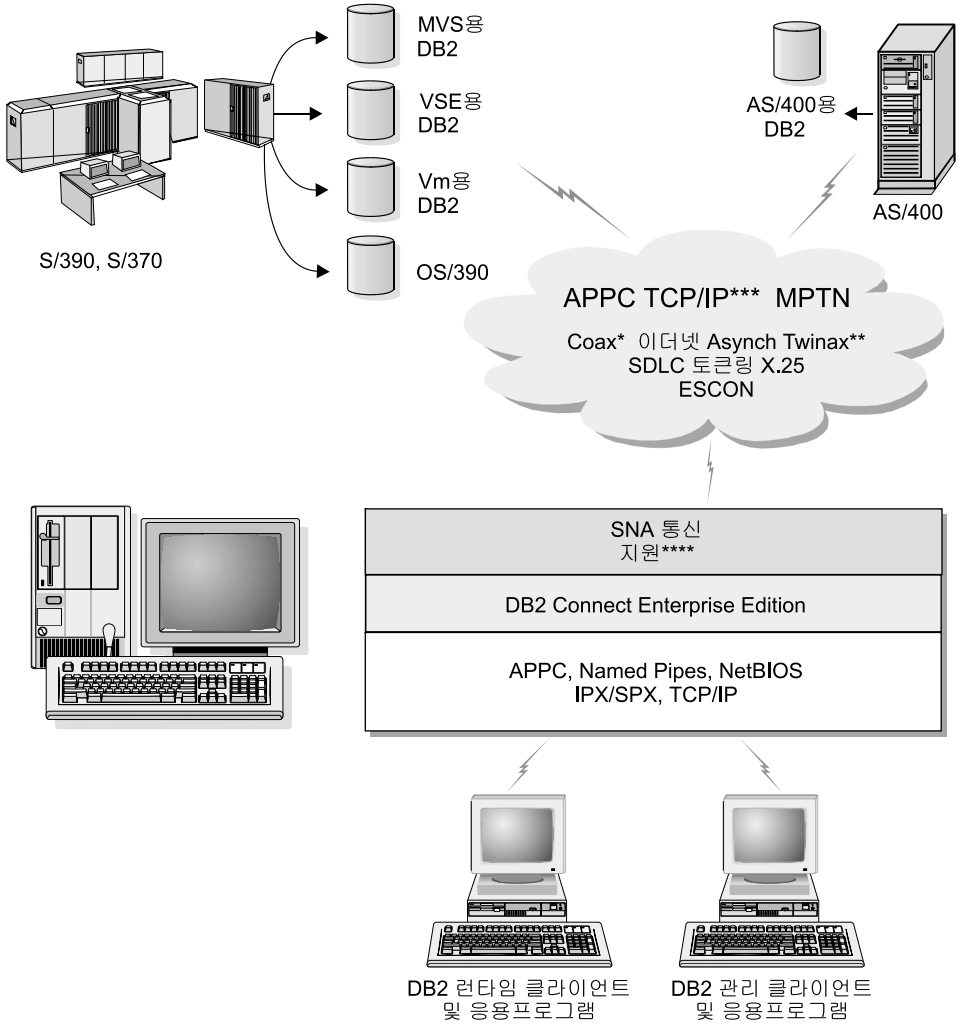
Connect에는 OS/390용 DB2에 대해 저장 프로시듀어를 빌드, 테스트 및 배치하는 데 사용할 수 있는 Stored Procedure Builder가 포함됩니다.

4. Microsoft ODBC, OLE DB 또는 ActiveX 데이터 오브젝트(ADO)를 사용하는 Windows 응용프로그램을 개발 중인 C 프로그래머들은 *Microsoft Open Database Connectivity Software Development Kit*을 사용해야 합니다. Java 프로그래밍 언어를 사용하여 응용프로그램을 개발하려는 프로그래머들은 IBM Java용 VisualAge와 같은 Java 개발 환경을 사용할 수 있습니다.

연결성 서버로서의 DB2 Connect Enterprise Edition

DB2 Connect 서버를 통해 여러 클라이언트가 호스트 또는 AS/400 데이터에 연결할 수 있으며 기업 데이터에 대한 액세스를 구축하고 유지보수하는 데 드는 노력을 현저히 줄일 수 있습니다. 28 페이지의 그림4에서는 DB2 Connect Enterprise Edition을 통해 DB2 클라이언트가 호스트 또는 AS/400 데이터베이스 서버에 간접적으로 연결될 수 있는 환경에 대한 IBM의 솔루션을 보여 줍니다.

다음 예에서 DB2 Connect 서버를 DB2 Connect 서버 지원 구성요소가 설치된 DB2 UDB Enterprise Edition 또는 Enterprise - Extended Edition 서버로 대체할 수 있습니다.



모든 프로토콜이 모든 플랫폼을 지원하지는 않습니다.

* 호스트 연결용
** AS/400용
*** TCP/IP 연결성은 OS/390 V5R1용 DB2, AS/400 V4R2용 DB2 또는 VM V6.1용 DB2를 필요로 합니다.
**** SNA 통신 지원은 각 운영 체제 별로 고유하며 원시 TCP/IP연결이 작동하지 않을 경우에만 필요합니다.

그림 4. DB2 Connect Enterprise Edition

DB2 Connect 및 웹 응용프로그램

웹 브라우저는 온라인 카탈로그에서 인트라넷 응용프로그램에 이르기까지 급속하게 모든 것에 대한 표준 인터페이스가 되고 있습니다. 단순한 웹 응용프로그램에서는 웹 서버 단독으로도 충분할 수 있습니다. 데이터베이스 액세스와 트랜잭션 처리를 필요로 하는 대용량 응용프로그램의 경우에는, DB2 Connect를 사용하여 웹을 통한 많은 수의 동시 트랜잭션을 관리하는 솔루션을 IBM에서 제공합니다.

이 절에서는 DB2 Connect를 사용하여 잇점을 얻을 수 있는 웹 기본 비지니스 솔루션에 대해 설명합니다.

전형적인 CGI 프로그래밍의 잇점과 한계

월드 와이드 웹에서의 e-business 응용프로그램들은 일반적으로 공통 게이트웨이 인터페이스(CGI)를 사용하여 사용자가 백엔드 데이터베이스를 조회할 수 있도록 합니다. 수많은 회사에서는 또한 웹 응용프로그램을 내부적으로 사용하며 이들은 데이터베이스를 백그라운드로 가지고 있습니다.

사용자들이 웹 페이지의 양식을 채우고 이 양식들은 CGI를 통해 웹 서버상의 응용프로그램 또는 스크립트에 전달됩니다. 스크립트는 제공된 데이터베이스 API를 차례로 사용하여 호스트 데이터베이스에 SQL 조회를 제출합니다. 그런 다음, 동일한 스크립트에서 조회 결과가 담긴 웹(HTML) 페이지를 빌드하여 사용자의 웹 브라우저에서 이를 표시하도록 되돌려 전송합니다. 예를 들어, 온라인 카탈로그가 해당되는데 여기에서 사용자는 특정 상품 또는 서비스의 사용 가능성 및 현재 가격을 조회할 수 있습니다.

CGI 응용프로그램은 디자인하기가 간단하고 유지보수하기도 쉽습니다. CGI 표준은 운영 체제 및 언어 양쪽 모두에 독립적이므로, 거의 모든 컴퓨팅 플랫폼에서 사용 가능합니다. CGI 프로그램은 C++ 또는 스크립트 언어(예: Perl)로 작성할 수 있습니다.

CGI가 웹 기본 응용프로그램에 대해 이상적인 솔루션으로 보일 수도 있지만, 여기에는 중요한 결점이 있습니다. CGI에 대한 프로그래밍 환경은 다른 API들처럼 복잡하지 않습니다. 또한 대규모 e-commerce 조작에 영향을 미치는 확장성 문제가 있습니다. CGI 응용프로그램이 호출될 때마다 웹 서버에서 새 프로세스가 작

성됩니다. 각 인스턴스에서는 데이터베이스에 대해 자신 소유의 연결을 확립하고 각 인스턴스에서는 자신의 조회를 제출합니다. 대용량의 트랜잭션 환경에서는 이러한 제한사항로 인해 중요한 성능 문제가 발생될 수 있습니다.

웹 서버와 함께 DB2 Connect를 사용하여 강력한 대용량의 e-commerce 응용프로그램을 작성할 수 있습니다. DB2 Connect에서는 웹 기본 응용프로그램 성능을 개선하는 여러 가지 솔루션을 제공합니다. 저장 프로시듀어(『웹 서버에서의 DB2 Connect』 참조)를 사용하여 DB2 Connect 사용자가 데이터베이스로 전송될 조회 수를 줄일 수 있습니다.

연결 풀링(31 페이지의 『연결 풀링』 참조)은 데이터베이스에 대한 연결 및 연결 해제의 빈도수를 줄입니다. CGI에 대한 제한사항이 매우 큰 영향을 미치는 대규모 작업의 경우에 대형 엔터프라이즈 응용프로그램에 대해 CGI가 아닌 연결을 제공하려면 IBM Net.Data(33 페이지의 『Net.Data』 참조) 및 WebSphere(34 페이지의 『IBM WebSphere』 참조)를 참조하십시오.

웹 서버에서의 DB2 Connect

IBM에서는 OS/2, UNIX, Windows NT 및 Windows 2000에 대한 모든 DB2 Connect 제품이 들어 있는 HTTP(웹) 서버를 제공합니다. DB2 Connect Enterprise Edition에서는 Apache 또는 Lotus Domino Go 웹 서버에 대해 별도의 지원을 제공하며 Microsoft Internet Information Server 또는 Netscape Enterprise Server와 같은 다른 웹 서버에 대해서도 작업할 수 있습니다.

OS/390, AS/400, VM 및 VSE 시스템에서 수행 중인 데이터베이스의 DB2 계열에 대해 작업하는 경우에는, 웹 서버에 DB2 Connect Enterprise Edition이 필요합니다. DB2 Connect Enterprise Edition에서는 웹 서버가 이 호스트 및 AS/400 플랫폼에 액세스할 수 있도록 라이브러리 및 통신 인터페이스들을 제공합니다. TCP/IP 또는 SNA는 웹 서버와 OS/390, AS/400, VM 또는 VSE에서 수행 중인 데이터베이스 사이의 통신에 사용될 수 있습니다.

주: IBM 웹 솔루션에서는 동일한 CGI 스크립트 또는 CGI 스크립트 내의 동일한 트랜잭션에 있는 여러 데이터베이스에 대해 작업할 수 있는 기능을 제공합니다.

다음의 두 절에서는 DB2 데이터베이스에 액세스하는 CGI 응용프로그램에서 사용할 수 있는 성능 향상에 대해 설명합니다. 뒤에 있는 절에서는 Java와 같은 표준 CGI에 대한 대안을 생각합니다.

저장 프로시듀어

클라이언트/서버 월드에서처럼 웹 응용프로그램의 경우에 중요한 고려사항은 HTTP 서버와 백 엔드 데이터베이스 사이에 발생하는 전송량을 최소화하는 것입니다. 이러한 고려사항은 대부분의 e-business 응용프로그램에 있어서 가장 중심이 되는 대용량 트랜잭션 처리에 특히 중요합니다.

바람직한 접근 방법은 저장 프로시듀어에 캡슐화된 프로그래밍 및 비지니스 로직으로 CGI 응용프로그램 프로그래밍을 결합하는 것입니다. OS/2, UNIX 및 Windows상의 DB2 Universal Database와 OS/390, AS/400 및 VSE상의 DB2는 모두 저장 프로시듀어를 호출하는 데 대해 동일한 매개변수 규약을 공유합니다.

일반 CGI의 경우에는 웹 브라우저가 웹 서버로 양식을 제출하며 그 곳에서 CGI 스크립트가 수행됩니다. 그러나 DB2 데이터베이스에 전송될 각각의 개별 SQL문 대신에 저장 프로시듀어를 실행하려는 요청이 전송됩니다. 이 저장 프로시듀어는 달리 지정하지 않으면 개별적으로 수행되는 여러 SQL문들을 캡슐화합니다. 저장 프로시듀어는 CGI 스크립트와 백 엔드 데이터베이스 사이에 앞뒤로 흐르는 메시지 블록 수를 줄입니다.

저장 프로시듀어의 중요한 잇점은 HTTP 서버와 DB2 데이터베이스 백 엔드 사이의 축소된 네트워크 교통량입니다. *응용프로그램 개발 안내서* 또는 DB2 저장 프로시듀어 빌더 온라인 도움말에서 저장 프로시듀어에 대한 자세한 내용을 참조하십시오.

연결 풀링

DB2 Connect 서버로부터 호스트로의 연결을 설정하기 위해서는 컴퓨팅 자원 및 시간이 필요합니다. 수 천개의 클라이언트가 DB2 Connect 서버를 통해 호스트로부터 연결 및 연결해제를 자주 반복하는 환경에서는 처리 시간의 상당 부분이 연결을 설정하고 연결을 제거하는 데 사용됩니다.

DB2 Connect의 연결 풀링은 그러한 환경에서의 중요한 성능 개선점을 제공합니다. DB2 Connect는 사용 가능 풀에 있는 데이터베이스에 대해 열린 연결을 유지보수합니다. 클라이언트가 연결을 요청하면 준비된 연결들이 들어 있는 이 풀에서 연결을 제공할 수 있습니다. 연결 풀링은 이 연결을 열고 닫는 데 일반적으로 소요되는 오버헤드를 상당 부분 줄일 수 있습니다.

159 페이지의 『연결 풀링』에서 연결 풀링의 작업 방법에 대한 자세한 내용을 참조하십시오.

jAVA 응용프로그램 서버로서의 DB2 Connect

CGI의 여러 결점들은 이 곳에서 멀리 이동시켜서 Java를 대신 사용하면 극복할 수 있습니다. IBM은 웹 트랜잭션의 모든 단계에서 CGI 대신에 Java를 쓸 수 있도록 애플릿 및 응용프로그램 모두를 제공합니다. IBM이 제공하는 솔루션에서는 기법들을 혼용할 수 있는데, 이는 Net.Data 및 Microsoft Active Server Pages와 같은 스크립트 솔루션들을 DB2와 함께 사용하거나 IBM WebSphere와 같은 Java 응용프로그램 서버가 제공하는 보다 강력한 구현 방향으로 옮겨갈 수 있음을 의미합니다.

Java 프로그래머를 위한 두 가지 API(Application Programming Interfaces)가 있습니다. JDBC는 먼저 Java servlets, Java 서버 페이지(JSP) 및 Enterprise Java Beans(EJB) 뿐만 아니라 데이터 인식 Java 애플릿, Java 응용프로그램을 개발하는 데 Java를 사용하도록 지원됩니다. JDBC는 호출 레벨 또는 메소드 호출 API입니다. 다른 Java API는 SQLJ입니다. SQLJ는 Java 프로그램 내에 SQL 인라인을 지정할 수 있는 기능을 제공합니다. DB2는 웹 트랜잭션의 클라이언트 또는 서버 측에서 양쪽 API 모두를 사용할 수 있습니다.

클라이언트측에서는 애플릿, 데이터 인식 애플릿 및 응용프로그램들이 지원됩니다. 데이터베이스측에서는 Java 지원이 사용자 정의 함수 및 저장 프로시듀어와 같은 데이터베이스 오브젝트로 구성됩니다.

OS/390용 DB2, VSE 및 VM용 DB2, OS/400용 DB2 경우에는 Java 응용프로그램을 배치하는 두 가지 서로 다른 방법이 있습니다. TCP/IP 또는 SNA와 함께 DB2 Connect Personal Edition에 의해 제공된 직접 연결성을 사용하거나 메인

프레임 또는 AS/400 백 엔드에 연결성을 제공하는 DB2 Connect Enterprise Edition 서버를 통과하도록 선택할 수 있습니다.

양쪽 경우에 웹 사용자에게는 데이터베이스에 액세스하기 위한 별도의 소프트웨어는 필요하지 않고 표준 웹 브라우저만 필요합니다. DB2 Connect 서버와 임의의 산업 표준 웹 서버만 설치하면 됩니다. 웹 서버 및 DB2 Connect가 동일한 실제 머신상에 없으면 웹 서버에 DB2 클라이언트를 설치해야 합니다.

OS/390용 DB2의 경우에는 주요 구성요소가 미들 타이어(mid-tier) 서버에서 수행 중인 DB2 Connect Enterprise Edition입니다. 이는 OS/390, VSE 및 VM용 DB2 또는 AS/400 서버에 대한 연결 뿐만 아니라 JDBC 서버 지원을 제공하는 구성요소입니다. 다시 말해, 클라이언트의 웹 브라우저 경우에는 어떠한 특수 소프트웨어도 필요하지 않습니다.

IBM에서는 Java 응용프로그램 및 애플릿을 개발하기 위해 광범위한 도구 세트를 제공합니다. 데이터베이스 연결성의 경우에 DB2 개발 프로그램판에서는 테스트용으로 DB2 Universal Database 및 DB2 Connect 뿐만 아니라 VisualAge for Java Professional Edition, WebSphere Application Server, Net.Data가 들어 있는 전체 킷을 제공합니다. IBM VisualAge for Java Enterprise Edition에는 대규모 엔터프라이즈 응용프로그램에 대한 개발 도구도 들어 있습니다. Borland JBuilder 또는 Symantec Visual Cafe와 같은 써드 파티 도구들도 IBM의 데이터베이스 솔루션에 대해 작업합니다.

Net.Data

DB2 Universal Database 및 DB2 Connect 계열의 일부분인 Net.Data는 웹 기본 트랜잭션 응용프로그램을 작성하고 유지보수하는 데 도움을 주기 위해 설계된 응용프로그램 개발 도구의 세트입니다. Net.Data를 사용하여 OS/2, Windows NT, Windows 2000, UNIX, OS/390, VM, VSE 및 OS/400용 DB2 UDB에 저장된 데이터를 액세스 및 변경할 수 있습니다. 사용자가 Net.Data를 사용하여 작성하는 응용프로그램들은 웹 서버에 저장되며 웹 브라우저를 통해 활성화될 수 있습니다.

Net.Data는 HTML 및 SQL에 대해 기본적인 내용만 알고 있는 사용자들이 매우 정교한 웹 응용프로그램을 빌드할 수 있도록 매크로 또는 템플리트를 사용합니다.

매크로는 Java, Java 스크립트, HTML 태그 및 내장 함수로 구성되는 텍스트 파일입니다. 이 매크로들은 사전정의된 배치, 변수 및 함수를 사용하여 동적 웹 페이지를 작성하는 데 사용될 수 있습니다.

기본 Net.Data 매크로에는 7개의 개별 섹션이 있습니다.

- 공통 섹션 : 기본적으로 프로그래머에 대한 문서 보조 프로그램으로 사용됩니다.
- 정의 섹션 : 변수 정의를 지정하기 위한 장소를 제공합니다.
- 기능 섹션 : 주 프로그래밍 논리가 들어 있습니다.
- 보고서 섹션 : Net.Data 매크로 출력에 대한 포맷팅 논리를 지정합니다.
- HTML 섹션 : 웹 페이지에서 사용되는 대부분의 HTML이 들어 있습니다.
- 포함 섹션 : 다른 매크로가 재사용할 수 있는 매크로의 공통 부분들을 포함하고 있는 편리한 방법입니다.
- 메시지 섹션 : 오류 조절이 제공됩니다.

특별히 DB2에 대한 Net.Data의 주요 기능은 클라이언트 배치가 필요하지 않다는 것입니다. 이러한 구현에서는 클라이언트가 단순히 웹 브라우저입니다.

Net.Data 프로세서는 웹 서버 뿐만 아니라 Windows NT, Windows 2000, OS/2 또는 UNIX 워크스테이션에 DB2 Universal Database와 함께 설치됩니다. OS/390용 DB2, VSE와 VM용 OS/390 및 400에 연결할 때 모든 Net.Data 하부 구조가 웹 서버 뿐만 아니라 DB2 Connect 서버에 배치됩니다.

IBM WebSphere

IBM WebSphere는 전형적인 CGI 프로그래밍에서보다 완벽한 e-business 솔루션을 제공합니다. WebSphere Application Server는 CGI의 스크립트 가능성을 수행할 뿐만 아니라 servlets, Active Server Pages 및 엔터프라이즈 JavaBeans를 사용하여 웹을 통한 복합적이고 최상의 서비스를 제공할 수 있도록 합니다. WebSphere로 다음을 수행할 수 있습니다.

- 개발 속도를 높이고 내부 조작 가능성을 최대화하기 위해 산업 표준을 이용합니다.
- 써드 파티 도구 기술 및 응용프로그램 프레임워크를 플러그인합니다.
- 웹 사이트 내용 성능 및 사용을 분석합니다.

- 보다 많은 사용자를 수용하고 처리량을 용이하게 유지보수하기 위해 사이트를 평가합니다.
- 대다수의 전 운용 환경(IBM AIX, HP-UX, Linux, Novell NetWare, IBM OS/2, IBM OS/390, IBM OS/400, Sun Solaris, Microsoft Windows NT 및 Windows 2000)에 배치합니다.
- Apache, IBM, Netscape 및 Microsoft의 웹 서버를 포함하여 기존의 웹 서버를 사용합니다.

WebSphere는 제품 하나가 아니라 세 개의 서로 다른 목표 시장을 겨냥한 세 가지의 제품 계열입니다. WebSphere 솔루션의 가장 중심은 Websphere Application Server입니다.

WebSphere Application Server는 세 가지 유형의 오브젝트에 대한 환경을 제공합니다. 하나는 Java 서버 페이지로서 이는 활동 서버 페이지와 유사합니다. 두 번째 구성요소는 Java servlet들로 구성되며 세 번째는 엔터프라이즈 JavaBeans입니다. 엔터프라이즈 JavaBeans는 초대형, 초강력 엔터프라이즈 클래스 응용프로그램을 배치하기 위한 새로운 표준입니다.

또한, Data Access JavaBeans는 특별히 DB2에 적용된 매우 정교한 데이터베이스 함수들을 제공합니다. DB2는 또한 JDBC 및 SQLJ를 통해 직접적으로 액세스할 수 있습니다. CORBA 뿐만 아니라 COM+도 모두 지원됩니다.

WebSphere 응용프로그램은 웹 서버 및 DB2 Universal Database와 동일한 플랫폼상에 배치될 수 있습니다. OS/390, VM, VSE 및 AS/400용 DB2의 경우에 WebSphere는 DB2 Connect Enterprise Edition와 동일한 플랫폼상에 배치됩니다.

Web Studio 및 WebSphere Performance Packs 뿐만 아니라 여러 개의 WebSphere 솔루션들이 있습니다. 세 개의 WebSphere 버전은 다음과 같습니다.

표준판 Web 사이트 작성자의 경우에 이 서버를 통해 Java servlets 및 JSP 기술을 사용하여 정적 페이지로부터 개인화된 동적 웹 내용의 주요 소스로 웹 사이트 및 포탈을 빠르고 쉽게 전송할 수 있습니다. 여기에는 또한 그룹 간에 또는 엔터프라이즈 간에 정보와 데이터를 쉽게 공유하기 위한 산

업 선두 XML 지원과, 웹 사이트에 대한 기업 차원의 재투자를 최대화하는 데 도움이 될 성능 및 사용 정보를 제공하는 내장 사이트 분석 기술도 포함됩니다.

고급판 응용프로그램 프로그래머의 경우에 높은 성능의 이 EJB 서버를 통해 DJB 구성요소를 사용하여 비지니스 로직을 배치할 수 있습니다. 이는 확장성 보안 연결성 및 Java 지원을 제공하며 표준판의 모든 기능도 포함합니다.

Enterprise Edition

엔터프라이즈 설계자의 경우에는 강력한 e-business 응용프로그램을 빌드하고 자원 재사용을 최대화하기 위해 이 서버가 사용자의 구조를 통해 다른 비지니스 시스템들을 통합합니다. Enterprise Edition은 award-wining IBM TXSeries 기술과 Component Broker 기술의 기능을 통합합니다. 여기에는 고급 및 표준판의 모든 기능도 포함됩니다.

응용프로그램 서버와 함께 DB2 Connect 사용

향상된 클라이언트 서버 응용프로그램을 사용하여 응용프로그램 설계자가 Windows 및 OS/2와 같은 플랫폼에서 그래픽 사용자 인터페이스로 응용프로그램을 제공함으로써 사용법을 개선하고 학습 비용을 줄일 수 있습니다. 동시에 다양한 운영 체제 및 하드웨어 플랫폼상의 강력한 데이터베이스 서버에 데이터베이스 관리 기능을 위임하는 융통성도 허용합니다.

응용프로그램 로직이 클라이언트 워크스테이션으로 분산되는 클라이언트 서버 모델을 일반적으로 *2 계층 클라이언트 서버*라고 합니다. 2 계층 모델에서는 응용프로그램이 클라이언트 계층상에 배치되고 데이터베이스 서버는 서버 또는 백엔드 계층을 구현합니다. 24 페이지의 『직접 데이터베이스 액세스』에 표시된대로 DB2 Connect는 2 계층 클라이언트 서버 응용프로그램에 대해 전체적인 지원을 제공하며, 여기서 데이터베이스 서버는 OS/390용 DB2, MVS/ESA용 DB2, DB2/400 또는 VM 및 VSE용 DB2입니다.

클라이언트 서버 응용프로그램의 크기를 증가시키면 2 계층 클라이언트 서버 모델에 중대한 한계점이 나타납니다. 수백 또는 수천개의 클라이언트 워크스테이션에 대용량의 비지니스 로직을 분산시키면 변경을 관리하기가 복잡하고 비용도 많이 듭니다. 비지니스 규칙을 변경하면 응용프로그램의 클라이언트 부분을 대체해야 합

니다. 종종 비지니스 규칙이 일관성있게 적용되는지 확인하려면 이 응용프로그램 구간 방출(rollout)이 동시에 엔터프라이즈 내의 모든 클라이언트 워크스테이션에 있어야 합니다.

스케일에 따라 명백해지는 2 계층 클라이언트 서버 모델의 또다른 결점은 그러한 응용프로그램들이 소비하는 자원의 양입니다. 2 계층 클라이언트라고도 하는 *fat 클라이언트*를 수백 또는 수천개 배치하면, 각 클라이언트 워크스테이션의 즉시 처리 능력 및 용량이 증가합니다. 더구나, 데이터베이스 서버에 대한 요구는 각 클라이언트가 전용 데이터베이스 연결과 그러한 연결을 유지보수하는 데 연관된 자원을 요청할 때도 많이 증가합니다. 분산 비지니스 로직의 2 계층 클라이언트 서버 종속성은 저장 프로시듀어를 광범위하게 사용함으로써 어느 정도 줄일 수 있지만, 다른 결점들은 모델을 변경하지 않고는 쉽게 해결되지 않습니다.

응용프로그램 서버 솔루션

증가된 2 계층 클라이언트 서버 응용프로그램의 단가 및 복합도로 인해 대부분의 초대형 응용프로그램들은 다중 계층 클라이언트 서버에 대한 경로상에 놓이게 됩니다. 다중 계층 모델에서는 데이터베이스 계층의 역활이 변경되지 않고 남아 있습니다. 그러나, 클라이언트 계층에는 하나 이상의 미들 타이어 즉, 일반적으로 하나의 계층이 추가됩니다. 그러므로, 이름은 *3 계층*이 됩니다.

3 계층 모델에서는 사용자 상호 작용에 대한 조절이 클라이언트에 위임되며 클라이언트에는 어떠한 비지니스 로직도 포함되지 않습니다. 미들 타이어는 하나 이상의 응용프로그램 서버로 구성됩니다. 응용프로그램 서버의 목표는 비지니스 프로세스 및 비지니스 규칙을 따르면서 강력하고도 효율적인 비용으로 구현하는 것입니다. 2 계층 모델에서처럼 비지니스 규칙 구현에서도 성능을 개선하기 위해 저장 프로시듀어를 종종 사용합니다.

클라이언트 워크스테이션은 더 이상 많은 양의 응용프로그램 로직을 구현하지 않고 사용자 상호 작용만을 핸들하기 때문에, 클라이언트 계층에 대한 자원 요청도 상당히 감소합니다. 사실상, 3 계층 모델에서의 클라이언트 계층을 종종 *thin 클라이언트*라고 합니다. 또한, 중앙 집중 응용프로그램 서버는 모든 클라이언트의 요청을 조절하기 때문에 모든 클라이언트 사이의 데이터베이스 연결과 같이 자원을 공유하는 기능을 가집니다. 결과적으로 데이터베이스 서버는 각 응용프로그램 사용자에 대한 전용 연결을 더 이상 유지보수하지 않습니다.

오늘날 산업 부분에는 3 계층 응용프로그램 서버에 대한 예들이 많이 있습니다. 거의 모든 엔터프라이즈 자원 플랜(ERP) 벤더들은 SAP R/3 및 PeopleSoft V7 응용프로그램과 같은 3 계층 모델을 사용하여 그 응용프로그램을 구현합니다. 기타 다른 예에는 Siebel 및 Vantive와 같은 엔터프라이즈 관계 관리 벤더가 포함됩니다.

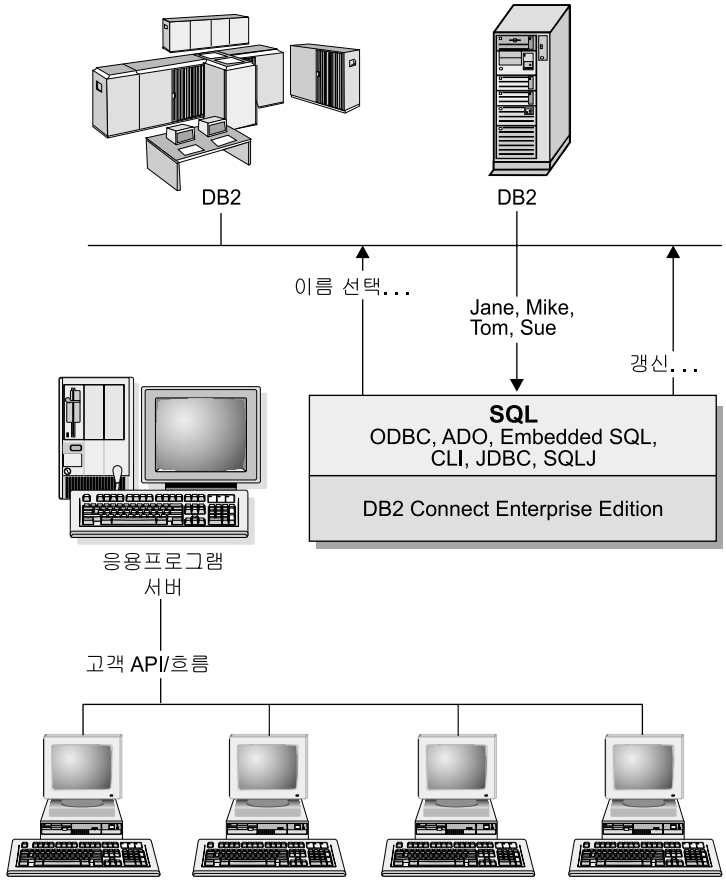
응용프로그램 서버 및 DB2 Connect

DB2 Connect Enterprise Edition 서버는 다중 계층 응용프로그램을 배치하기 위해 광범위한 지원을 제공합니다. DB2 Connect가 제공하는 지원에는 DB2 계열 데이터베이스 서버와 상호 작용하는 전체적인 통신 하부 구조 뿐만 아니라 응용프로그램 로직(ODBC, ADO, DB2 CLI, Embedded SQL, JDBC 및 SQLJ)을 개발하는 데 사용할 수 있는 여러 API도 포함됩니다.

DB2 Connect에서는 또한 데이터베이스 계층이 여러 개의 DB2 계열 데이터베이스 서버로 구성된 구현도 지원합니다. 이를 통해 응용프로그램 서버는 단일 트랜잭션으로 여러 데이터베이스 서버에 상주하는 데이터를 갱신하는 트랜잭션을 구현할 수 있습니다.

그와 같은 분산 트랜잭션의 무결성은 DB2 Connect가 제공하는 2 단계 확약 프로토콜 지원에 의해 확실해졌습니다. 예를 들어, 응용프로그램은 동일한 트랜잭션으로 OS/390용 DB2 데이터베이스 및 Windows NT상의 DB2 UDB에 있는 데이터를 갱신할 수 있습니다. 분산 요청(DR) 지원이 설치되어 사용 가능한 경우에는 응용프로그램이 동일한 트랜잭션에서 Oracle 데이터베이스를 읽은 후 DB2 계열 데이터베이스를 갱신할 수 있습니다.

다음 그림에서는 응용프로그램 서버와 백엔드 데이터베이스 서버 사이의 연결성 메카니즘 뿐만 아니라 API들이 DB2 Connect Enterprise Edition에 의해 제공됩니다.



연결 풀링(159 페이지의 『연결 풀링』 참조) 및 연결 집중기(161 페이지의 『DB2 Connect 연결 집중기』 참조)와 같은 DB2 Connect의 고급 기능들은 응용프로그램 자원 요구량을 상당히 줄이고 응용프로그램 서버 구현을 간소화합니다.

DB2 Connect 및 응용프로그램 서버 구성

DB2 Connect Enterprise Edition 제품(그 자체에서 사용 가능하거나 DB2 Connect Unlimited Edition 제품 패키지의 일부분으로서)은 응용프로그램 서버와 함께 사용하는 데 필요합니다. DB2 Connect Personal Edition은 응용프로그램 서버와 함께 사용하는 용도로는 지원되지도 않고 인가되지도 않았습니다. 또한 응용프로그램 서버를 구현하는 고객들은 획득해야 할 사용자 사용권 수를 알기 위해 DB2 Connect의 사본과 함께 제공된 기간 및 조건들을 읽어 두어야 합니다. 응용프로

그램 서버 환경에서 DB2 Connect를 배치하는 방식에는 두 가지가 있습니다. DB2 Connect Enterprise Edition은 다음에 설치됩니다.

- 응용프로그램 서버 머신 및
- 기타 통신 서버 머신.

대부분의 상황에서 응용프로그램 서버 자체와 동일한 서버상에 DB2 Connect 사본을 설치하는 것이 우선적인 솔루션입니다. 응용프로그램 서버에 DB2 Connect를 설치하면 응용프로그램 서버가 구현 중일 수도 있는 fail-over 및 로드 밸런스 스킴에 참여할 수 있습니다. 이렇게 설정하면 DB2 Connect가 개별 서버에 설치될 때 필요한 추가의 네트워크 홉을 제거하기 때문에 잠재적으로 보다 나은 성능을 제공할 수 있습니다. 더구나, 추가 서버를 설치 및 유지보수할 필요가 없기 때문에 관리를 간소화할 수 있습니다.

개별 서버에 DB2 Connect를 설치하는 것은 응용프로그램 서버가 수행 중인 운영 체제 또는 하드웨어 플랫폼에서 DB2 Connect Enterprise Edition이 사용 불가능한 경우에 적합한 옵션입니다. 예를 들어, 응용프로그램 서버가 Silicone Graphics (SGI) 또는 SCO UnixWare 서버에 배치된 경우에, DB2 Connect Enterprise Edition이 이 플랫폼에서 사용할 수 없는 유일한 옵션이 개별 서버에 DB2 Connect를 배치하는 것입니다.

트랜잭션 처리 모니터와 함께 DB2 Connect 사용

이전 절에서는 응용프로그램 서버와 함께 DB2 Connect를 사용하는 것에 대해 배웠습니다. 응용프로그램 서버는 많은 수의 사용자가 최소의 시스템 자원을 사용하여 응용프로그램을 실행할 수 있도록 합니다.

응용프로그램 서버는 응용프로그램 서버가 실행하는 응용프로그램으로부터 조정된 트랜잭션을 호출할 수 있도록 확장될 수 있습니다. 이 트랜잭션 조정을 일반적으로 트랜잭션 처리(TP) 모니터라고 합니다. TP 모니터는 응용프로그램 서버와 결합하여 작업합니다.

*트랜잭션*은 조직의 일일 조작을 수행하는 면에서 일상적인 이벤트(일반적으로는 서비스에 대한 요청)와 같은 개념일 수 있습니다. 트랜잭션의 규칙적인 처리는 TP 모니터가 설계된 작업의 유형입니다.

모든 조직에는 어떻게 작동될 것인지에 대해 서술하는 규칙 및 프로시듀어가 있습니다. 이러한 규칙을 구현하는 사용자 응용프로그램을 *비지니스 로직*이라고 합니다. 이 비지니스 응용프로그램이 실행하는 트랜잭션들을 종종 트랜잭션 처리 또는 온라인 트랜잭션 처리(OLTP)라고 합니다.

상업상 OLTP의 주요 특성은 다음과 같습니다.

많은 사용자
: 조직의 대다수 사람들이 사용하게 될 트랜잭션 처리에 많기 때문에 많은 사람들이 비지니스의 현재 상태에 영향을 미칩니다.

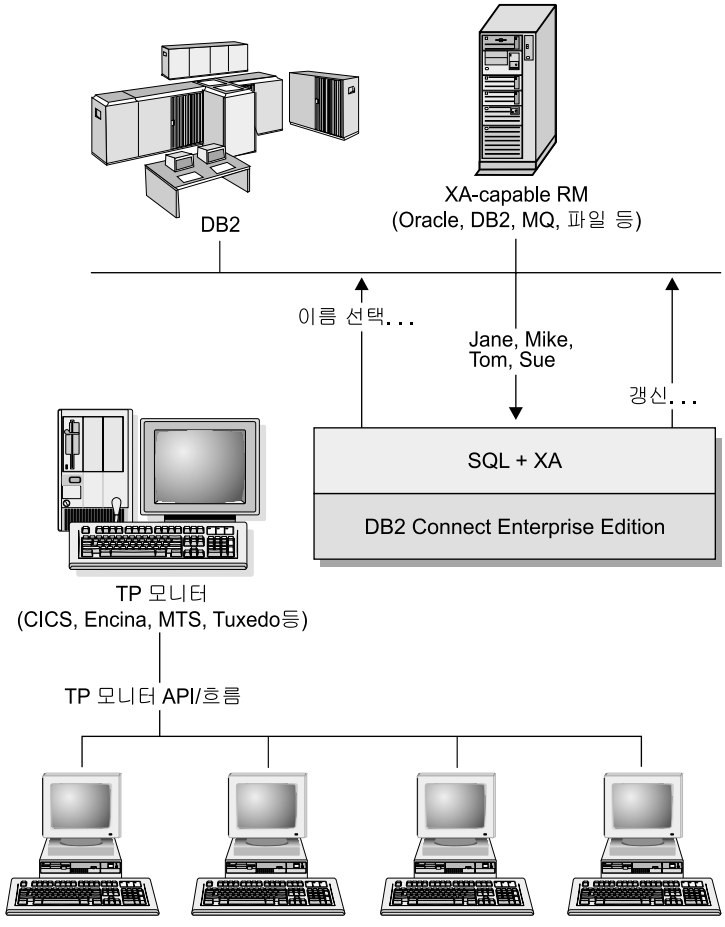
반복적 컴퓨터와의 상호 작용 대부분이 반복해서 실행되는 동일한 프로세스일 가능성이 높습니다. 예를 들어, 주문 입력이나 지불 처리는 매일 몇번씩 사용됩니다.

단기 상호 작용
: 조직의 사람들이 트랜잭션 처리 시스템과 나누게 될 상호 작용들의 지속 기간이 대부분 짧습니다.

데이터 공유
: 데이터는 조직의 상태를 나타내기 때문에 데이터의 사본이 하나만 있을 수 있습니다.

데이터 무결성
: 데이터는 조직의 현재 상태를 나타내야 하며 내부적으로 일치해야 합니다. 예를 들어, 모든 주문은 고객 레코드와 연관되어야 합니다.

저비용/트랜잭션
: 트랜잭션 처리가 비지니스 수행의 직접적인 단가를 나타내기 때문에, 시스템의 단가는 최소값이어야 합니다. DB2 Connect는 UNIX, Windows NT, Windows 2000 또는 OS/2에서 수행 중인 응용프로그램 서버의 제어 하에 있는 응용프로그램들이 원격 LAN, 호스트 및 AS/400 데이터베이스 서버에 대한 트랜잭션을 실행하고 이 트랜잭션들을 TP 모니터가 조정할 수 있도록 합니다.



이 그림에서 응용프로그램 서버와 백엔드 데이터베이스 서버 사이의 연결성 메카니즘 뿐만 아니라 API들은 DB2 Connect Enterprise Edition에서 제공됩니다.

TP 모니터의 예

오늘날 가장 많이 사용되는 TP 모니터는 다음과 같습니다.

- IBM TxSeries CICS
- IBM TxSeries Encina Monitor
- BEA Tuxedo.

Microsoft Transaction Server Remote S/390, AS/400 및 LAN 데이터베이스 서버는 이 TP 모니터에 의해 조정된 트랜잭션 내에서 사용될 수 있습니다.

Tuxedo 및 DB2 Connect

DB2 Connect 버전 6 및 이전 버전에서는 Tuxedo 기본 응용프로그램들을 호스트 및 AS/400 데이터베이스 서버에 대해 읽기 전용 액세스로 제한했습니다. DB2 Connect 버전 7에서는 이러한 제한사항들이 없어졌습니다. 이제 Tuxedo 기본 응용프로그램들은 Tuxedo 통합 트랜잭션 내에서 호스트 및 AS/400 데이터베이스 서버를 갱신할 수 있습니다. 특수 구성 요구사항 및 제한사항들이 적용됩니다. 161 페이지의 『DB2 Connect 연결 집중기』에서 자세한 내용을 참조하십시오.

X/Open 분산 트랜잭션 프로세싱(DTP) 모델

응용프로그램 실행 비지니스 로직은 단일 트랜잭션으로 여러 자원을 갱신하는 데 필요할 수도 있습니다. 예를 들어, 한 계정에서 다른 계정으로의 자금 이체를 구현하는 은행 응용프로그램에서는 한 데이터베이스(″from″ 계정)에 차변을 기입하고 다른 데이터베이스(″to″ 계정)에 예금할 수 있습니다.

서로 다른 벤더들이 이 두 데이터베이스를 제공할 수도 있습니다. 예를 들어, 한 데이터베이스는 OS/390용 DB2 Universal Database이고 다른 데이터베이스는 Oracle 데이터베이스일 수 있습니다. 모든 TP 모니터가 각 데이터베이스 벤더의 전매 트랜잭션 인터페이스를 구현하기 보다는, 응용프로그램이 액세스하는 자원과 TP 모니터 사이에 공통 트랜잭션 인터페이스가 정의됩니다. 이 인터페이스를 *XA 인터페이스*라고 합니다. XA 인터페이스를 사용하는 TP 모니터를 *XA 추종 트랜잭션 관리 프로그램(TM)*이라고 합니다. XA 인터페이스를 구현하는 갱신 가능한 자원을 *XA 추종 자원 관리 프로그램(RM)*이라고 합니다.

위에 나열된 TP 모니터들은 모두 XA 추종 TM들입니다. 원격 호스트, AS/400 및 DB2 UDB LAN 기본 데이터베이스 서버들은 DB2 Connect를 통해 액세스될 때 XA 추종 RM들입니다. 그러므로, XA 추종 TM이 있는 TP 모니터들은 비지니스 응용프로그램 실행 트랜잭션 내에서 호스트, AS/400 및 LAN 기본 DB2 UDB 데이터베이스 서버들을 사용할 수 있습니다.

XA 추종 트랜잭션 관리 프로그램과 함께 DB2 Connect를 사용하는 방법

이 절에서는 TP 모니터 내에서 S/390 및 AS/400 데이터베이스 서버를 사용하는 데 필요한 구성 단계에 대해 설명합니다. 여기에서는 호스트 또는 AS/400 데이터

베이스 서버에 대한 연결을 구성 및 테스트할 뿐만 아니라 사용자에게 조작 가능한 TP 모니터가 있고 DB2 Connect를 설치했다고 가정합니다. *DB2 Connect 빠른 시작* 책에서 자세한 내용을 참조하십시오.

가장 대중적인 TP 모니터를 구성하는 데 필요한 단계는 *관리 안내서*에서 제공됩니다. LAN 기본 DB2 UDB 데이터베이스 서버 대 호스트 또는 AS/400 데이터베이스 서버에 대한 액세스의 구성 사이에는 아무런 차이가 없습니다. 다음 지시사항들은 *관리 안내서*에 나열되지 않은 TP 모니터에 대한 일반적인 구성 단계를 요약한 것입니다.

TP 모니터 내에서 S/390 및 AS/400 데이터베이스 서버를 사용하기 위해 DB2 Connect를 구성하려면 다음 단계를 수행하십시오.

1. DB2 XA 스위치를 액세스할 수 있도록 TP 모니터를 구성하십시오. DB2 XA 스위치는 DB2 Connect의 XA API의 주소를 가진 TP 모니터를 제공합니다. TP 모니터 마다 이를 수행하기 위한 서로 다른 방법을 가지고 있습니다. *관리 안내서*에서 TP 모니터에 DB2 XA 스위치 제공에 대한 자세한 내용을 참조하십시오.
2. DB2의 XA_OPEN 문자열이 있는 TP 모니터를 구성하십시오. 각 TP 모니터는 자신만의 방법으로 이를 수행합니다. *관리 안내서*에서 DB2 Connect의 XA OPEN 문자열에 대한 자세한 내용을 참조하십시오. TP 모니터가 사용할 DB2의 XA OPEN 문자열을 구성하는 방법에 대한 내용을 해당 TP 모니터 문서에서 참조하십시오.
3. 필요하다면 DB2 Connect 동기 지점 관리 프로그램(SPM) 기본값 구성 매개변수를 수정하십시오. 호스트 및 AS/400 데이터베이스 서버에서는 아직 XA 인터페이스를 지원하지 않습니다.

   SPM은 XA 2단계 확약 프로토콜을 호스트 및 AS/400 데이터베이스 서버가 사용하는 2단계 확약 프로토콜로 맵핑하는 DB2 Connect의 구성요소입니다. 기본값으로 DB2 인스턴스는 SPM 구성 매개변수에 대해 사전정의 값을 가집니다. 가장 중요한 매개변수는 데이터베이스 관리 프로그램 구성 매개변수 SPM_NAME입니다. 기본값은 TCP/IP 호스트 이름의 첫 7자를 변형한 값입니다.

TCP/IP를 사용하여 OS/390용 DB2에 연결할 경우에는, 어떠한 기본 설정값도 변경하지 말아야 합니다. 이러한 경우에는 이미 조작 가능하므로 SPM 구성이 필요하지 않습니다. SNA를 사용하여 호스트 또는 AS/400 데이터베이스 서버에 액세스하는 경우에는 SPM_NAME값이 네트워크에서 유효한 SNA LU를 나타내는지 확인해야 합니다. 기본 SPM_NAME 값이 허용되지 않으면, 다중 사이트 갱신 마법사를 사용하여 이 값을 수정해야 합니다.

제4장 DB2 Connect 환경에서의 프로그래밍

이 절은 DB2 Connect를 사용하는 응용프로그램 작성에 대한 정보를 제공합니다. *CLI Guide and Reference*, *Command Reference* 및 *응용프로그램 개발 안내서*에서 자세한 내용을 참조하십시오.

분산 환경에서의 프로그래밍

DB2 Connect에서는 응용프로그램이 System/390 및 AS/400 서버에 있는 DB2 데이터베이스의 데이터에 액세스할 수 있습니다. 예를 들어, Windows에서 수행중인 응용프로그램은 OS/390용 DB2 Universal Database 데이터베이스의 데이터를 액세스할 수 있습니다. 새로운 응용프로그램을 작성하거나 호스트나 AS/400 환경에서 수행되도록 기존의 응용프로그램을 수정할 수 있습니다. 그리고 어떤 환경의 응용프로그램을 개발하여 다른 환경으로 이식할 수 있습니다.

DB2 Connect에서는 OS/390용 DB2 Universal Database와 같은 호스트 데이터베이스 제품에 다음 API를 사용할 수 있는데, 해당 항목은 호스트 데이터베이스 제품에 의해 지원되어야 합니다.

- 정적 및 동적 Embedded SQL
- DB2 콜 레벨 인터페이스
- Microsoft ODBC API
- JDBC

어떤 SQL문은 관계형 데이터베이스 제품마다 서로 다릅니다. 다음과 같은 SQL문도 있습니다.

- 표준에 관계없이 사용하는 모든 데이터베이스 제품에 대해 동일한 명령문
- *SQL 참조서*에 설명되어 있어 모든 IBM 관계형 데이터베이스 제품에서 사용가능한 명령문
- 액세스하는 임의의 데이터베이스 시스템에 대해 유일한 명령문

© Copyright IBM Corp. 1993, 2000

첫 번째 두 가지 범주의 SQL문은 이식이 매우 용이하지만, 세 번째 범주의 SQL 문은 먼저 변경을 해야 가능합니다. 일반적으로, DDL의 SQL문은 DML의 SQL 문처럼 이식이 가능하지 않습니다.

DB2 Connect는 DB2 Universal Database에 의해 지원되지 않은 SQL문을 승인할 경우가 있습니다. DB2 Connect는 이런 명령문을 호스트나 AS/400 서버에 전달합니다. 최대 컬럼 길이와 같은 각 플랫폼에서의 제한사항에 대해 알려면, *SQL 참조서*에서 자세한 내용을 참조하십시오.

CICS 응용프로그램을 OS/390나 VSE로부터 이동하여 다른 CICS 제품(예를 들어, AIX용 CICS)에서 수행하려면, DB2 Connect를 사용하는 OS/390 또는 VSE 데이터베이스를 또한 액세스할 수 있습니다. 자세한 내용은 *CICS/6000 Application Programming Guide*와 *CICS Customization and Operation* 매뉴얼을 참조하십시오.

호스트나 AS/400 환경에서 프로그램을 작성할 경우, 다음과 같은 사항을 고려하십시오.

- DDL 사용
- DML 사용
- 데이터 제어 언어(DCL) 사용
- 연결과 연결해제
- 사전 처리 컴파일
- 정렬 순서 정의
- 참조 무결성 관리
- 잠금
- SQLCODE와 SQLSTATE에서의 차이
- 시스템 카탈로그 사용
- 분리 레벨
- 저장 프로시듀어
- NON AATOMIC 복합 SQL
- 분산 작업 단위(DUOW)

- DB2 Connect에 의해 지원되거나 거부되는 SQL문

DDL 사용

DDL문은 다른 시스템에서 저장영역이 다르게 처리되므로 IBM 데이터베이스 제품이 서로 다릅니다. 호스트나 AS/400 서버 시스템에서, 데이터베이스의 설계와 CREATE TABLE문의 실행 간에는 여러 단계가 있습니다. 예를 들어, 일련의 명령문은 논리 오브젝트 설계를 저장영역의 물리적으로 표현되는 해당 오브젝트로 번역할 것입니다.

사전 처리 컴파일러는 호스트나 AS/400 서버 데이터베이스로 사전 처리 컴파일할 때, 이러한 DDL 명령문을 호스트나 AS/400 서버로 전달합니다. 동일한 명령문은 응용프로그램이 수행중인 시스템의 데이터베이스에 대해 사전 처리 컴파일하지 않습니다. 예를 들어, OS/2 응용프로그램에서 CREATE STORGROUP문은 OS/390용 DB2 Universal Database 데이터베이스로 성공적으로 사전 처리 컴파일되지만, OS/2 데이터베이스용 DB2로는 사전 처리 컴파일되지 않습니다.

DML 사용

일반적으로, DML 명령문은 이식이 매우 용이합니다. SELECT, INSERT, UPDATE, DELETE문은 IBM 관계형 데이터베이스 제품에서 유사합니다. 대부분의 응용프로그램은 DB2 Connect 프로그램에 의해 지원되는 DML SQL문을 주로 사용합니다.

숫자 데이터 유형

숫자 데이터가 DB2 Universal Database로 전송되면, 데이터 유형은 변환됩니다. 숫자와 Zoned 10진 SQLTYPE(AS/400용 DB2 Universal Database에 의해 지원된)은 고정된(압축된) 십진 SQLTYPE으로 변환됩니다.

혼합 바이트 데이터

혼합 바이트 데이터는 같은 컬럼의 확장 UNIX 코드(EUC) 문자 집합, 2바이트 문자 세트(DBCS), 1바이트 문자 세트(SBCS)의 문자들로 구성됩니다. EBCDIC(OS/390, OS/400, VSE, VM)으로 데이터를 저장하는 시스템에서, 시프

트 인(SI) 및 시프트 아웃(SO) 문자는 2바이트 데이터의 시작과 끝을 표시합니다. ASCII(OS/2와 UNIX 같은)에 데이터를 저장하는 시스템에서, 시프트 인(SI) 및 시프트 아웃(SO)문자는 필요없습니다.

응용프로그램이 ASCII 시스템으로부터 혼합 바이트 데이터를 전송하려면, 시프트 문자의 공간이 충분한지 확인하십시오. SBCS 데이터에서 DBCS 데이터까지의 스위치마다, 데이터 길이에 2바이트를 추가하십시오. 더 나은 이식성을 위해서, 혼합 바이트 데이터를 사용하는 응용프로그램에 가변 길이 문자열을 사용하십시오.

긴 필드

긴 필드(254자보다 긴 문자열)는 시스템마다 서로 다르게 처리됩니다. 호스트나 AS/400 서버는 긴 필드용 스칼라 함수의 부속 집합만 지원하는데, 예를 들어 OS/390용 DB2 Universal Database에서는 긴 필드에 대해 **LENGTH** 함수와 **SUBSTR** 함수만 허용됩니다. 또한 호스트나 AS/400 서버는 SQL문에 대해 서로 다른 처리를 필요로 합니다. 예를 들어, VSE & VM용 DB2의 경우에는 INSERT 명령문에 호스트 변수인, SQLDA나 NULL 값만을 사용해야 합니다.

대형 오브젝트(LOB) 데이터 유형

LOB 데이터 유형은 DB2 Connect에서 지원합니다.

사용자 정의 유형(UDT)

DB2 Connect에서는 사용자 정의 구별 유형만 지원됩니다. 추상적 데이터 유형은 지원되지 않습니다.

ROWID 데이터 유형

ROWID 데이터 유형은 DB2 Connect에서 VARCHAR 2진 데이터용으로 처리됩니다.

64비트 정수(BIGINT) 데이터 유형

DB2 Connect에서는 8 바이트(64비트) 정수를 지원합니다. BIGINT 내부 데이터 유형은 데이터 정밀도를 유지하면서 매우 큰 데이터베이스의 기본 행수(cardinality)를 지원하는 데 사용됩니다.

데이터 제어 언어(DCL) 사용

각 IBM 관계형 데이터베이스 관리 시스템은 GRANT 및 REVOKE SQL문에 대해 다른 레벨의 세분화도(granularity)를 제공합니다. 제품에 맞는 책을 검색하여 각 데이터베이스 관리 시스템용으로 사용하는 데 적합한 SQL문인지 확인하십시오.

연결과 연결해제

DB2 Connect는 매개변수가 없는 CONNECT뿐만 아니라 CONNECT 명령문의 CONNECT TO 및 CONNECT RESET 버전을 지원합니다. 응용프로그램이 명시적 CONNECT TO 명령문을 먼저 실행하지 않고 SQL문을 호출하면, *내재된* 연결은 기본 응용프로그램 서버(정의된 경우)로 실행됩니다.

데이터베이스에 연결할 경우, 관계형 데이터베이스 관리 시스템을 식별하는 정보는 SQLCA의 SQLERRP 필드에서 리턴됩니다. 응용프로그램 서버(AS)가 IBM 관계형 데이터베이스인 경우, 첫 번째 3바이트 SQLERRP에는 다음 내용이 들어 있습니다.

DSN OS/390용 DB2 Universal Database

ARI VSE & VM용 DB2

QSQ AS/400용 DB2 Universal Database

SQL DB2 Universal Database.

DB2 Connect 사용 중에 CONNECT TO 또는 널(NULL) CONNECT 명령문을 실행하려는 경우, SQLCA의 SQLERRMC 필드에서 국가 코드 또는 지역 토큰은 공백으로 리턴됩니다. 응용프로그램 서버(AS)의 CCSID는 코드 페이지 또는 코드 세트 토큰으로 리턴됩니다.

연결해제는 CONNECT RESET 명령문(1 연결 유형용), RELEASE 및 COMMIT 명령문(2 연결 유형용) 또는 DISCONNECT 명령문(연결 유형이지만 TP 모니터 환경에 없는)을 사용하여 할 수 있습니다.

연결이 확실하게 해제되지 않은 상태로 응용프로그램이 정상적으로 종료된 경우, DB2 Connect는 결과 데이터를 내재적으로 확약합니다.

주: 응용프로그램은 오류를 나타내는 SQLCODE를 수신하고 여전히 정상적으로 종료할 수 있습니다; DB2 Connect은 이런 경우의 데이터를 확약합니다. 데이터의 확약을 원하지 않으면, ROLLBACK 명령을 실행해야 합니다.

FORCE 명령을 사용하면 선택된 사용자나 모든 사용자를 데이터베이스로부터 연결해제합니다. 이 명령은 호스트나 AS/400 서버 데이터베이스에 대해 지원됩니다. 사용자가 DB2 Connect 워크스테이션에서 강제 종료될 수 있습니다.

사전 처리 컴파일

다른 IBM 관계형 데이터베이스 시스템용 사전 처리 컴파일에는 몇 가지 차이점이 있습니다. DB2 Universal Database용 사전 처리 컴파일러는 다음과 같은 점에서 호스트나 AS/400 서버 사전 처리 컴파일러와 다릅니다.

- 하나의 응용프로그램을 통하여 한번의 전달만 있습니다.
- DB2 Universal Database 데이터베이스에 대해 바인딩하는 경우, 오브젝트가 성공적인 바인드를 위해 존재해야 합니다. VALIDATE RUN은 지원되지 않습니다.

블로킹

DB2 Connect 프로그램은 DB2 데이터베이스 관리 프로그램 블로킹 바인드 옵션을 지원합니다.

UNAMBIG
명확한 커서만 블록화됩니다(기본 값).

ALL 불명확한 커서가 블록화됩니다.

NO 커서는 블록화되지 않습니다.

DB2 Connect 프로그램은 RQRIOBLK 필드에 대해 DB2 데이터베이스 관리 프로그램의 구성 파일에 정의된 블록 *크기*를 사용합니다. DB2 Connect의 현재 버전은 최대 32 767의 블록 *크기*를 지원합니다. DB2 데이터베이스 관리 프로그램 구성 파일에 더 큰 값이 지정된 경우, DB2 Connect는 32 767 값을 사용하되 DB2 데이터베이스 관리 프로그램의 구성 파일을 재설정하지는 않습니다. 블로킹은 동적 SQL과 정적 SQL용으로 같은 블록 *크기*를 사용하여 같은 식으로 처리됩니다.

주: 대부분의 호스트나 AS/400 서버 시스템은 동적 커서를 모호한 것으로 간주하지만, DB2 Universal Database 시스템은 일부 동적 커서를 모호하지 않은 것

으로 간주합니다. 혼란을 피하기 위해, DB2 Connect와 함께 BLOCKING ALL을 지정할 수 있습니다.

*Administrative API Reference*와 *Command Reference*에 설명된대로 CLP, 제어 센터 또는 API를 사용하여 DB2 데이터베이스 관리 프로그램의 구성 파일에 블록 크기를 지정하십시오.

패키지 속성

패키지에는 다음 속성이 있습니다.

콜렉션 ID
: 패키지 ID. PREP 명령에서 지정할 수 있습니다.

소유자
: 패키지 소유자의 권한 부여 ID. PREP 명령이나 BIND 명령에서 지정할 수 있습니다.

작성자
: 패키지를 바인드하는 사용자 이름입니다.

규정자
: 패키지의 오브젝트에 대해 내재된 규정자입니다. PREP 명령이나 BIND 명령에서 지정할 수 있습니다.

각 호스트나 AS/400 서버 시스템마다 이러한 속성 사용에 한계가 있습니다.

OS/390용 DB2 Universal Database
: 네 가지 속성이 모두 다를 수 있습니다. 다른 규정자 사용에는 특별한 관리 특권이 필요합니다. 이러한 속성 사용과 관련한 조건에 대한 자세한 내용은 OS/390용 DB2 Universal Database의 *Command Reference*를 참조하십시오.

VSE & VM용 DB2
: 모든 속성은 식별되어야 합니다. USER1이 바인드 파일(PREP 사용)을 작성하고, USER2는 실제 바인드를 실행하는 경우, USER2는 USER1용으로 바인드하려면 DBA 권한이 필요합니다. 그 속성에 대해 USER1 사용자 이름만 사용됩니다.

AS/400용 DB2 Universal Database
: 규정자는 콜렉션 이름을 나타냅니다. 규정자와 소유자간의 관계는 오브젝트에 대한 GRANT 및 REVOKE 특권에 영향을 줍니다. 로그온된 사용

자 이름은 콜렉션 ID로 규정되지 않은 경우, 작성자이며 소유자입니다. 이런 경우 콜렉션 ID는 소유자입니다. 콜렉션 ID는 규정자로 사용되기 전에 이미 존재해야 합니다.

DB2 Universal Database
네 가지 속성이 모두 다를 수 있습니다. 서로 다른 소유자를 사용하려면 관리자 권한이 필요하며 바인더가 스키마(존재할 경우)에 대해 CREATEIN 특권을 가지고 있어야 합니다.

주: DB2 Connect는 OS/390용 DB2 Universal Database 및 DB2 Universal Database의 *SET CURRENT PACKAGESET* 명령에 대한 지원을 제공합니다.

C 널(NULL) 종료 문자열

CNULREQD 바인드 옵션은 LANGLEVEL 옵션을 사용하여 지정된 널(NULL) 종료 문자열의 처리를 대체합니다.

MIA 또는 SAA1으로 설정된 LANGLEVEL 옵션이 준비되었을 때 널(NULL) 종료 문자열의 처리 방식에 대한 내용을 *응용프로그램 개발 안내서*에서 참조하십시오.

기본 값으로, CNULREQD는 YES로 설정됩니다. 이렇게 설정하면 널(NULL) 종료 문자열이 MIA 표준에 따라 해석됩니다. OS/390용 DB2 Universal Database 서버에 연결할 경우에는 CNULREQD를 YES로 설정하도록 권장합니다. CNULREQD 옵션이 NO로 설정되어 SAA1 표준으로 코딩된(널(NULL) 종료 문자열에 관해서) 응용프로그램을 바인드해야 합니다. 그렇지 않으면, LANGLEVEL을 사용하여 SAA1로 설정될 준비가 되어 있더라도 널(NULL) 종료 문자열은 MIA에 따라 해석될 것입니다.

독립형 SQLCODE 및 SQLSTATE

독립형 SQLCODE 및 SQLSTATE 변수는, ISO/ANS SQL92에서 정의된 것처럼 LANGLEVEL SQL92E 사전 처리 컴파일 옵션을 통하여 지원됩니다. 사전 처리 컴파일시, SQL0020W 경고가 LANGLEVEL은 지원되지 않는다고 지시합니다. 이 경고는 *Command Reference*의 LANGLEVEL MIA 아래에 설명된 기능에만 적용되며, LANGLEVEL SQL92E의 부속 집합입니다.

정렬 순서 정의

EBCDIC과 ASCII간의 차이점으로 다양한 데이터베이스 제품의 정렬 순서가 달라지고, 또한 ORDER BY 및 GROUP에 영향을 줍니다. 이런 차이를 최소화하는 한 방법은 EBCDIC 정렬 순서와 같게 한 사용자 정의 조합 순서를 작성하는 것입니다. 새 데이터베이스를 작성할 경우에만 조합 순서를 지정할 수 있습니다. *응용프로그램 개발 안내서*, *Administrative API Reference* 및 *Command Reference* 에서 자세한 내용을 참조하십시오.

주: 데이터베이스 테이블은 ASCII 형식으로 OS/390용 DB2 Universal Database에 지금 저장될 수 있습니다. 이렇게 하면 DB2 Connect와 OS/390용 DB2 Universal Database 사이에 더 빨리 데이터를 교환하며, 그렇지 않을 경우 데이터를 변환하고 재배열하는 데 사용되어야 하는 필드 프로시듀어를 제공할 필요는 없습니다.

참조 무결성 관리

다른 시스템은 다음과 같이 서로 다르게 참조 제한조건을 처리합니다.

OS/390용 DB2 Universal Database

기본 키를 사용하여 외부 키를 작성하려면, 먼저 기본 키에 대해 색인을 작성해야 합니다. 테이블을 자체 참조할 수 있습니다.

VSE & VM용 DB2

색인은 외부 키에 대해 자동으로 작성됩니다. 테이블은 자체 참조할 수 없습니다.

AS/400용 DB2 Universal Database

색인은 외부 키에 대해 자동으로 작성됩니다. 테이블을 자체 참조할 수 있습니다.

DB2 Universal Database

DB2 Universal Database 데이터베이스에서, 색인은 기본 키와 같은 고유한 제한 조건에 대해 자동으로 작성됩니다. 테이블을 자체 참조할 수 있습니다.

기타 규칙은 연쇄 레벨에 대하여 다릅니다.

잠금

데이터베이스 서버가 잠금을 수행하는 방식은 여러 응용프로그램에 영향을 줄 수 있습니다. 예를 들어, 행 레벨 잠금과 커서 안정성(CS)의 분리 레벨로 설계된 응용프로그램은 페이지 레벨 잠금을 수행하는 시스템으로 직접적으로 이식할 수 없습니다. 이런 보이지 않는 차이가 있어서, 응용프로그램을 조정해야 합니다.

OS/390용 DB2 Universal Database 및 DB2 Universal Database 제품은 잠금을 시간종료하고 대기 중인 응용프로그램으로 오류 리턴 코드를 송신할 수 있습니다.

SQLCODE와 SQLSTATE의 차이

다른 IBM 관계형 데이터베이스 제품은 같은 오류에 대해 같은 SQLCODE를 제공하지 않습니다. 다음 두 가지 방법으로 이 문제를 처리할 수 있습니다.

- 특정한 오류에 SQLCODE 대신에 SQLSTATE를 사용하십시오.

  SQLSTATE는 데이터베이스 제품 전체에서 거의 동일한 의미를 가지며, 그 제품은 SQLCODE에 해당하는 SQLSTATE를 생성합니다.

- SQLCODE를 시스템간에 맵핑하십시오.

  기본적으로 DB2 Connect는 SQLCODE와 토큰을 각 IBM 호스트나 AS/400 서버 시스템에서 DB2 Universal Database 시스템으로 맵핑합니다. 기본 맵핑을 겹쳐쓰거나 SQLCODE 맵핑이 없는 데이터베이스 서버(IBM이 아닌 데이터베이스 서버)를 사용하려면, 사용자의 SQLCODE 맵핑 파일을 지정할 수 있습니다. SQLCODE 맵핑을 작동 중지할 수도 있습니다.

  141 페이지의 『제11장 SQLCODE 맵핑』에서 자세한 내용을 참조하십시오.

시스템 카탈로그 사용

시스템 카탈로그는 IBM 데이터베이스 제품간에 서로 다릅니다. 뷰의 사용으로 많은 차이점이 잘 드러나지 않을 수 있습니다. 자세한 내용은 사용 중인 데이터베이스 서버의 사용 설명서를 참조하십시오.

CLI의 카탈로그 기능은 DB2 계열 전반에 걸쳐 카탈로그 조회에 대해 동일한 API와 결과 집합의 지원을 제공함으로써, 이 문제를 해결합니다.

검색 지정의 숫자 변환 오버플로우

검색 지정의 숫자 변환 오버플로우는 다른 IBM 관계형 데이터베이스 제품에서 서로 다르게 처리됩니다. 예를 들어, 부동 소수 컬럼을 OS/390용 DB2 Universal Database 및 DB2 Universal Database의 정수 호스트 변수로 패치한다고 간주하십시오. 부동 소수 값을 정수 값으로 변환할 때, 변환 오버플로우가 일어납니다. 기본 값으로, OS/390용 DB2 Universal Database는 경고 SQLCODE와 널(NULL) 값을 응용프로그램에 리턴합니다. 반대로, DB2 Universal Database는 변환 오버플로우 오류를 리턴합니다. 응용프로그램이 적절한 크기의 호스트 변수로 페치해서 검색 지정의 숫자변환 오버플로우를 피하도록 권장됩니다.

분리 레벨

DB2 Connect는 응용프로그램을 prep하거나 바인드하는 경우에 다음 분리 레벨을 채택합니다.

RR 반복 읽기(RR)

RS 읽기 안정성

CS 커서 안정성(CS)

UR 미확약 읽기

NC 확약 없음

분리 레벨은 최대의 보호 상태에서 최소 보호 상태까지 순서대로 나열됩니다. 호스트나 AS/400 서버가 지정된 분리 레벨을 지원하지 않는 경우, 다음으로 높은 지원 레벨이 사용됩니다.

표2는 각 호스트나 AS/400 응용프로그램 서버(AS)마다 각 분리 레벨의 결과를 보여줍니다.

표 2. 분리 레벨

DB2 Connect	OS/390용 DB2 Universal Database	VSE & VM용 DB2	AS/400용 DB2 Universal Database	DB2 Universal Database
RR	RR	RR	주 1	RR
RS	주 2	RR	COMMIT(*ALL)	RS
CS	CS	CS	COMMIT(*CS)	CS
UR	주 3	CS	COMMIT(*CHG)	UR

표 2. 분리 레벨 (계속)

DB2 Connect	OS/390용 DB2 Universal Database	VSE & VM용 DB2	AS/400용 DB2 Universal Database	DB2 Universal Database
NC	주 4	주 5	COMMIT(*NONE)	UR

주:

1. AS/400용 DB2 Universal Database는 RR과 일치하는 COMMIT 옵션은 없습니다. AS/400용 DB2 Universal Database는 테이블 전체를 잠궈서 RR을 지원합니다.
2. 버전 3.1의 경우 RR이 되고, APAR PN75407가 있는 버전 4.1 또는 버전 5.1의 경우 RS가 됩니다.
3. 버전 3.1의 경우 CS가 되고, 버전 4.1 또는 버전 5.1의 경우 UR이 됩니다.
4. 버전 3.1의 경우 CS가 되고, APAR PN60988이 있는 버전 4.1 또는 버전 5.1의 경우 RS가 됩니다.
5. 분리 레벨 NC는 VSE & VM용 DB2에서는 지원되지 않습니다.

AS/400용 DB2 Universal Database와 같이, 응용프로그램이 UR의 분리 레벨로 바인드되고 블로킹이 ALL로 설정되거나 분리 레벨이 NC로 설정되면, 저널링되지 않은 테이블을 액세스할 수 있습니다.

저장 프로시듀어

- 호출

  클라이언트 프로그램은 SQL CALL문을 실행하여 서버 프로그램을 호출할 수 있습니다. 각 서버는 이런 경우에 기타 서버에 대해 약간 다르게 작업합니다.

  OS/390

  스키마 이름은 8 바이트를 초과하면 안되며, 프로시듀어 이름은 18바이트 이상 될 수 없습니다. 저장 프로시듀어는 서버의 SYSIBM.SYSPROCEDURES 카탈로그에 정의되어야 합니다.

  VSE 또는 VM

  프로시듀어 이름은 18바이트 이상 될 수 없고, 서버의 SYSIBM.SYSPROCEDURES 카탈로그에 정의되어야 합니다.

  OS/400

  프로시듀어 이름은 SQL 식별자여야 합니다. DECLARE PROCEDURE 또는 CREATE PROCEDURE문을 사용하여 저장 프로시듀어의 위치를 지정할 실제 경로 이름을 (스키마 이름이나 콜렉션 이름) 지정할 수도 있습니다.

REXX/SQL로부터 AS/400용 DB2에 대한 모든 CALL문은 CALL USING DESCRIPTOR에 대한 REXX/SQL 맵에서 실행되는 CALL 문처럼 응용프로그램에 의해 동적으로 준비되고 실행되어야 합니다.

SQL CALL문의 구문에 대해서는 *SQL 참조서*에서 자세한 내용을 참조하십시오. 응용프로그램을 작성할 경우 저장 프로시듀어의 사용 방법에 대해서는 *응용프로그램 개발 안내서*에서 자세한 내용을 참조하십시오.

서버 프로그램이 OS/390용 DB2 Universal Database, AS/400용 DB2 Universal Database 또는 VSE & VM용 DB2에서 사용하는 것과 동일한 매개변수 규약을 사용하여 DB2 Universal Database에서 서버 프로그램을 호출할 수 있습니다. *응용프로그램 개발 안내서*에서 DB2 Universal Database 저장 프로시듀어 호출에 대한 자세한 내용을 참조하십시오. 다른 플랫폼에서의 매개변수 규약에 대한 내용은 해당 플랫폼의 DB2 제품 문서를 참조하십시오.

저장 프로시듀어의 모든 SQL문은 클라이언트 SQL 프로그램에 의해 시작된 SQL 작업 단위(UOW)의 일부로 실행됩니다.

- 특별한 의미를 가진 표시기 값 또는 저장 프로시듀어로부터 표시기 값을 전달하지 마십시오.

  DB2 Universal Database 사이에서, 시스템은 표시기 변수로 지정하는 모든 값을 전달합니다. 그러나, DB2 Connect를 사용하는 경우, 표시기 변수에서 0, -1 및 -128만 전달할 수 있습니다.

- 서버 응용프로그램에 의해 발생한 경고나 오류를 리턴하는 매개변수를 정의해야 합니다.

  DB2 Universal Database의 서버 프로그램은 SQLCA를 갱신하여 오류나 경고를 리턴할 수 있지만 OS/390용 DB2 Universal Database, AS/400용 DB2 Universal Database 또는 MVS/ESA용 DB2의 저장 프로시듀어에는 이러한 지원이 없습니다. 저장 프로시듀어로부터 오류 코드를 리턴하려면, 해당 오류 코드를 매개변수로 전달해야 합니다. SQLCODE와 SQLCA는 시스템 검출 오류용 서버에 의해서만 설정됩니다.

- VSE & VM용 DB2 버전 7 이상과 OS/390용 DB2 Universal Database 버전 5.1 이상은 이 때 저장 프로시듀어의 결과 집합을 리턴할 수 있는 유일한 호스트 또는 AS/400 응용프로그램 서버입니다.

Stored Procedure Builder

DB2 Stored Procedure Builder는 저장 프로시듀어를 작성, 설치 및 테스트하기 위해 사용이 간단한 개발 환경을 제공합니다. 이를 사용함으로써 사용자는 DB2 서버에 저장 프로시듀어를 등록, 빌딩 및 설치하는 데 따른 세부사항 보다는 저장 프로시듀어 논리를 작성하는 데 중점을 둘 수 있습니다. 또한 Stored Procedure Builder를 사용함으로써 한 운영 체제에서 저장 프로시듀어를 개발하고 다른 서버 운영 체제에서 이들을 빌드할 수 있습니다.

Stored Procedure Builder는 신속한 개발을 지원하는 그래픽 응용프로그램입니다. Stored Procedure Builder를 사용하여 다음의 타스크를 수행할 수 있습니다.

- 신규 저장 프로시듀어를 작성.
- 지역 및 원격 DB2 서버에 저장 프로시듀어를 빌드
- 기본의 저장 프로시듀어를 수정 및 리빌드.
- 설치된 저장 프로시듀어의 실행을 테스트 및 디버그

DB2 Universal Database 프로그램 그룹으로부터 개별적인 응용프로그램으로서 Stored Procedure Builder를 시작하거나 다음과 같은 개발 응용프로그램에서 Stored Procedure Builder를 시작할 수 있습니다.

- Microsoft Visual Studio
- Microsoft Visual Basic
- IBM VisualAge for Java

OS/390용 DB2에 대한 제어 센터에서 Stored Procedure Builder를 시작할 수도 있습니다. 제어 센터 도구 메뉴, 도구 모음 또는 저장 프로시듀어 폴더로부터 개별적인 프로세스로서 Stored Procedure Builder를 시작할 수 있습니다. 또한, Stored Procedure Builder 프로젝트 창에서 OS/390용 DB2 서버에 빌드된 하나 이상의 선택된 SQL 저장 프로시듀어를 명령행 처리기(CLP)로 수행할 수 있는 지정된 파일로 내보내기할 수 있습니다.

Stored Procedure Builder는 프로젝트를 사용하여 사용자의 작업을 관리합니다. 각 Stored Procedure Builder 프로젝트는 OS/390용 DB2 서버와 같은 특정 데이터베이스에 대한 연결을 저장합니다. 또한, 각 데이터베이스에 있는 저장 프로시듀어의 부속 집합을 표시하기 위해 필터를 작성할 수 있습니다. 신규 또는 기존의

Stored Procedure Builder를 열 때는 이름, 스키마, 콜렉션 ID 또는 콜렉션 ID(OS/390 경우에만)에 따라 저장 프로시듀어를 볼 수 있도록 저장 프로시듀어를 필터링할 수 있습니다.

연결 정보는 Stored Procedure Builder 프로젝트에 저장됩니다. 그러므로, 기존의 프로젝트를 열 때는 데이터베이스에 대해 사용자 ID와 암호를 입력하도록 자동으로 프롬프트됩니다. SQL 저장 프로시듀어 삽입 마법사를 사용하여 OS/390용 DB2 서버에 SQL 저장 프로시듀어를 빌드할 수 있습니다. OS/390용 DB2 서버에 빌드된 SQL 저장 프로시듀어의 경우에는 특정 컴파일, 사전링크, 링크, 바인드, 런타임, WLM 환경 및 외부 보안 옵션을 설정할 수 있습니다.

또한, SQL 저장 프로시듀어가 수행중인 스레드에 대한 CPU 시간 관련 정보 및 다른 DB2 단가 정보를 포함하여, SQL 저장 프로시듀어에 관한 SQL 단가 정보를 얻을 수 있습니다. 특히, 시작/잠금 경합 대기 시간, getpage 수, 읽기 I/O 수 및 쓰기 I/O 수에 관한 단가 정보를 얻을 수 있습니다.

단가 정보를 얻기 위해 Stored Procedure Builder는 OS/390용 DB2 서버에 연결하고, SQL문을 실행하고, 저장 프로시듀어(DSNWSPM)를 호출하여 SQL 저장 프로시듀어가 사용한 CPU 시간을 알아 냅니다.

NOT ATOMIC 복합 SQL

복합 SQL을 사용하면 많은 SQL문을 하나의 실행 가능한 블록으로 그룹화할 수 있습니다. 그러면, 네트워크 오버헤드를 줄이고 응답 시간을 개선합니다.

DB2 Connect는 NOT ATOMIC 복합 SQL을 지원합니다. 이것은 복합 SQL 처리시 오류가 발생해도 계속된다는 것을 의미합니다.(DB2 Connect에 의해 지원되지 않은 NOT ATOMIC 복합 SQL을 사용하면, 오류가 발생했을 때 전체 복합 SQL 그룹을 구간 복원합니다.)

명령문은 응용프로그램 서버(AS)에 의해 종료될 때까지 계속 실행됩니다. 일반적으로, 복합 SQL문 실행은 심각한 오류가 발생할 경우에만 중단됩니다.

NOT ATOMIC 복합 SQL은 지원되는 모든 호스트 또는 AS/400 응용프로그램 서버(AS)에서 사용할 수 있습니다.

많은 SQL 오류가 발생하면, 첫 번째 일곱 개의 SQLSTATE 실패 명령문은 많은 오류가 발생했다는 메시지와 함께 SQLCA의 SQLERRMC 필드에서 리턴됩니다. *SQL 참조서*에서 자세한 내용을 참조하십시오.

DB2 Connect로 다중 사이트 갱신

DB2 Connect에서는 하나의 분산 작업 단위(DUOW) 내에서 여러 개의 데이터베이스를 갱신할 수 있습니다. 이 기능의 사용 가능 여부는 여러 요인에 달려 있습니다.

- 응용프로그램은 CONNECT 2와 SYNCPOINT TWOPHASE 옵션으로 사전 처리되어야 합니다.
- SNA 네트워크 연결이 있는 경우, AIX, OS/2 및 Windows NT에서 DB2 Connect Enterprise Edition 버전 7의 DB2 동기점 관리 프로그램(SPM) 기능이 제공하는 2단계 확약 지원을 사용할 수 있습니다. 그러면 아래의 데이터베이스 서버는 분산 작업 단위에 참여할 수 있습니다.
  - AS/400용 DB2 버전 3.1 또는 후속 버전
  - MVS/ESA용 DB2 버전 3.1 또는 후속 버전
  - OS/390용 DB2 버전 5.1 또는 후속 버전
  - VM &VSE용 DB2 버전 V5.1 또는 후속 버전

  위의 사실은 원시(native) DB2 UDB 응용프로그램과 IBM TXSeries, Open Systems용 CICS, BEA Tuxedo, Encina Monitor 및 Microsoft Transaction Server와 같은 외부 트랜잭션 처리(TP) 모니터로 조정하는 응용프로그램에 적용됩니다.

  주: 161 페이지의 『DB2 Connect 연결 집중기』에서 XA 집중기에 대한 자세한 내용을 참조하십시오.
- TCP/IP 네트워크 연결이 있으면 OS/390용 DB2 V5.1 이상의 서버가 분산 작업 단위에 참여할 수 있습니다. IBM TXSeries, Open Systems용 CICS, Encina Monitor 또는 Microsoft Transaction Server와 같은 트랜잭션 처리 모니터로 응용프로그램을 제어한다면 DB2 동기점 관리 프로그램(SPM)을 사용해야 합니다.

DB2 UDB 버전 7(DB2 UDB EE, DB2 UDB EEE 및 DB2 Connect EE 등)에서, DB2 SPM은 TCP/IP 연결을 지원하도록 확장되었습니다. 응용프로그램이 원시 DB2 UDB 응용프로그램인 경우에는, DB2 SPM에 2단계 확약이 필요없습니다.

원시 DB2 응용프로그램과 TP 모니터 응용프로그램에서 모두 공통 DB2 Connect Enterprise Edition 서버를 사용하여 TCP/IP 연결을 통해 호스트 데이터를 액세스한다면 DB2 SPM을 사용해야 합니다.

SNA와 TCP/IP 네트워크 프로토콜 둘다를 사용하여 호스트 데이터를 액세스하기 위해 단일 DB2 Connect Enterprise Edition 서버를 사용할 예정이고 2단계 확약이 필요한 경우에는, DB2 SPM을 사용해야 합니다. 이는 DB2 응용프로그램과 TP 모니터 응용프로그램에 모두 적용됩니다.

주:

1. DB2 일반 서버 버전 2.1 데이터베이스는 한 작업 단위(UOW)에서 2단계 확약으로 갱신될 수 있으며 OS/390용 DB2 Universal Database 버전 5.1이 트랜잭션 관리 프로그램 데이터베이스가 *아닐* 경우에만 적용됩니다.
2. 트랜잭션용 TM_DATABASE가 OS/390용 DB2 Universal Database 버전 5.1이면, 해당 트랜잭션에 참여하는 DB2 CS V2.1 데이터베이스는 클라이언트 응용프로그램용 *읽기 전용*입니다.
3. DB2 Universal Database 버전 7 클라이언트 응용프로그램은 OS/390용 DB2 Universal Database 버전 5.1이 트랜잭션용 *TM_DATABASE*인 경우에만 서로 다른 레벨의 데이터베이스 서버에서 분산 작업 단위(DUOW)에 참여할 수 있습니다. 해당 *DB2 Connect 빠른 시작* 책에서 자세한 내용을 참조하십시오.
4. TCP/IP에서 2단계 확약 트랜잭션용으로 지원되는 데이터베이스 레벨은 DB2 클라이언트 레벨, TM_DATABASE 레벨, 구성원(participant) 데이터베이스 레벨에 따라 결정됩니다.

   DB2 Universal Database 버전 7 및 OS/390용 DB2 Universal Database 버전 5 데이터베이스에 액세스하는 모든 클라이언트는 DB2 Universal Database 버전 7에 있어야 바람직합니다. DB2 버전 2.1 클라이언트는 OS/390용 DB2 Universal Database 버전 5 데이터베이스가 트랜잭션을 구성하는 경우, 2단계 확약 트랜잭션을 시작할 수 *없습니다*.

DB2 Connect에서 지원하는 호스트나 AS/400 서버 SQL문

다음 명령문은 호스트나 AS/400 서버 처리에 대해서는 성공적으로 컴파일하지만 DB2 Universal Database 시스템의 처리에 대해서는 컴파일되지 않습니다.

- ACQUIRE
- DECLARE(수정자.(규정자.)테이블_이름 TABLE ...
- LABEL ON

이 명령문들은 또한 명령행 처리기에 의해 지원됩니다.

다음 명령문은 호스트나 AS/400 서버 처리에는 지원되지만, 바인드 파일이나 패키지에 추가되지 않으며 명령행 처리기에서도 지원하지 않습니다.

- DESCRIBE 명령문_이름 INTO 설명자_이름 USING NAMES
- PREPARE 명령문_이름 INTO 설명자_이름 USING NAMES FROM ...

사전 처리 컴파일러는 다음을 가정합니다.

- 호스트 변수는 입력 변수입니다.
- 명령문은 유일한 섹션 번호로 지정됩니다.

DB2 Connect에서 거부하는 호스트나 AS/400 서버 SQL문

다음 SQL문은 DB2 Connect와 명령행 처리기에 의해 지원되지 않습니다.

- COMMIT WORK RELEASE
- DECLARE 상태_이름, 명령문_이름 STATEMENT
- DESCRIBE 명령문_이름 INTO 설명자_이름 USING xxxx (xxxx는 ANY, BOTH 또는 LABELS임)
- PREPARE 명령문_이름 INTO 설명자_이름 USING xxxx FROM :호스트_변수(xxxx는 ANY, BOTH 또는 LABELS임)
- PUT ...
- ROLLBACK WORK RELEASE
- SET :호스트_변수 = CURRENT ...

VSE & VM용 DB2 확장된 동적 SQL문은 -104와 구문 오류 SQLCODE로 거부됩니다.

OS/390용 DB2 Universal Database에서의 사용 비용 청구 계정 구현

여러 OS/390용 DB2 Universal Database 설치에서는 시스템 관리자가 개별 사용자 액세스와 자원 사용을 연관시키는 과정을 모니터하는 자원을 구현합니다. 이것은 소비 자원에 대해 개별 사용자 또는 부서 비용을 청구하는 데 사용할 수 있습니다. 이 과정은 흔히 *사용 비용 청구 계정*으로 참조됩니다

DB2 Connect 제품에서 시스템 사용자는 DB2 Connect를 통하여 데이터베이스를 액세스하는 사용자가 소비한 대형 컴퓨터 자원을 모니터할 수 있습니다. 계정 문자열을 사용하여 DB2 Connect로부터 데이터베이스 서버용 DB2로 계정 데이터를 송신할 수 있습니다. 사용 통계 문자열은 시스템에 생성된 데이터와 사용자에게 제공된 데이터를 결합합니다. 이 데이터를 사용하여 시스템 관리자는 자원 사용과 각 사용자 액세스를 결합하여 사용자에게 비용을 청구할 수 있습니다.

사용 통계 문자열은 DRDA 매개변수 PRDDTA를 사용하여 송신됩니다. 이 매개변수의 내용이 DRDA에 구성되지 않았으므로, 응용프로그램 서버(AS)가 사용 통계 데이터로서 데이터를 인식할 것이라고 확실히 말할 수 없습니다. 현재는 PRDDTA가 MVS 및 OS/390 OS/390에서만 지원됩니다. 문자열은 사용 통계 레코드로서 저장됩니다.

사용 통계 문자열은 DB2 Connect에 의해 생성된 56 바이트(접두부)와 사용자에 의해 지정된 199바이트(접미부)까지 최대 길이 255로 이루어져 있습니다.

표3은 시스템 생성 필드를 나타냅니다. 각 필드의 오른쪽은 공백으로 채워집니다.

표 3. DB2 Connect에 의해 생성되는 사용 통계 문자열 필드

필드 이름	길이	설명.
acct_str_len	1	계정 문자열 -1의 길이를 나타내는 16진수 값. 예를 들어 X'3C' 입니다.

표 3. DB2 Connect에 의해 생성되는 사용 통계 문자열 필드 (계속)

필드 이름	길이	설명.
client_prdid	8	클라이언트 소프트웨어의 제품 ID. 예를 들어, DB2 Universal Database 버전 7의 제품 ID는 SQL07010입니다.
client_platform	18	클라이언트가 있는 플랫폼. 예를 들어 AIX, OS/2, DOS 또는 Windows입니다.
client_appl_name	20	사용자 응용프로그램 이름의 첫 번째 20 문자. 예를 들어, payroll입니다.
client_authid	8	사용자 응용프로그램의 인증 이름. 예를 들어, SMITH입니다.
suffix_len	1	사용자 제공 접미사 길이를 나타내는 16진수 값. X'00'은 사용자 제공 접미사가 없다는 뜻입니다.

사용자 정의 접미사는 다음 중 하나입니다.

- sqlesact() API도 응용프로그램에 의해 지정된 값
- DB2ACCOUNT 환경 변수 값
- DFT_ACCOUNT_STR (기본 계정 문자열) 구성 매개변수의 값
- 널(NULL) 문자열

접미사가 199 문자보다 긴 경우에는 절단됩니다. 사용 통계 문자열이 호스트나 AS/400 데이터베이스 서버로 전송될 때 올바르게 변환되게 하려면, 문자 A-Z, 0-9 및 밑줄(_)만 사용해야 합니다.

사용 통계 문자열 설정의 API 방식이 권장됩니다. 응용프로그램은 데이터베이스에 연결하기 전에 API를 호출해야 합니다. 응용프로그램 안에서 사용 통계 문자열을 변경하려면(예를 들어, 다른 데이터베이스에 연결할 경우 다른 문자열을 송신하려면), API를 다시 호출하십시오. 그렇지 않으면, 그 값은 응용프로그램 종료시까지 영향을 줍니다.

sqlesact() API가 첫 번째 데이터베이스 연결 요청 전에 호출되지 않으면, DB2ACCOUNT 환경 변수를 읽습니다. 이 값은 응용프로그램 또는 백그라운드 명령행 처리기 프로세스 종료시까지 영향을 줍니다. 첫 번째 데이터베이스 연결 후에 새로운 사용 통계 문자열 접미사를 지정하려면, sqlesact() API를 사용하거나 응용프로그램 또는 백그라운드 CLP 프로세스를 종료하고 새로운 값으로 설정된 DB2ACCOUNT 세트를 가지고 재시작하십시오.

DB2ACCOUNT 값이 없으면, DFT_ACCOUNT_STR 시스템 구성 매개변수 값이 사용됩니다. 이 기본 값은 사용 통계 문자열을 DB2 Connect로 전송할 수 없는 데이터베이스 클라이언트용으로 사용할 수 있습니다. 이 값이 없으면, 널(NULL) 문자열이 사용됩니다.

다음은 사용 통계 문자열의 예입니다.

```
x'3C'SQL07010OS/2          cheque          SMITH   x'05'DEPT1
x'37'SQL07010OS/2          cheque          SMITH   x'00'
```

첫 번째 예에서, 사용자 정의 접미사는 DEPT1입니다. 두 번째 예에서는 널(NULL) 문자열입니다.

사용 통계 정보를 OS/390용 DB2 서버에 전송

수많은 메인프레임 고객들은 서로 다른 응용프로그램에서 사용하는 자원의 상세한 계정을 작동 프로시듀어의 중요한 부분으로 간주합니다. OS/390용 DB2는 정보 시스템 부서에서, 사용되는 메인프레임 자원에 대해 개별 사용자 부서에 요금을 부과할 수 있도록 하는 사용 통계 보고서를 생성하는 많은 기능을 제공합니다. 이러한 프로세스를 *종종 사용 비용 청구 계정*이라고 합니다. DB2 Connect 제품은 기존의 사용 통계 보고서와 프로시듀어를 사용하여 PC와 UNIX 응용프로그램에 사용되는 호스트 자원에 대한 정확한 사용 통계를 제공합니다.

DB2 Connect는 다음을 제공하여 융통성 있게 이 특징을 구현합니다.

- 특정 DB2 Connect Enterprise Edition 서버가 생성한 모든 사용에 대한 기본 사용 통계 문자열.

- 요금이 부과되는 계정을 지정하기 위한 개별 사용자나 응용프로그램용의 두 메카니즘.

사용 통계 문자열 설정

기본 사용 통계 문자열은 DB2 Connect 워크스테이션의 dft_account_str 구성 매개변수에 의해 설정됩니다. 이 기본 메카니즘은 계정 문자열을 DB2 Connect로 전송할 수 없는 데이터베이스 클라이언트에 유용합니다. 예를 들어, 버전 2 제품 이전에 개발된 응용프로그램들이 있습니다.

클라이언트 응용프로그램이 DB2 Connect 서버에서 설정된 기본 사용 통계 문자열을 겹쳐쓰는 방법에는 두 가지가 있습니다.

- 통계 문자열 설정 API 사용:sqlesact()

  sqlesact API는 응용프로그램이 데이터베이스에 연결하기 전에 호출됩니다. 다음과 같은 이유로 이 방법을 사용해야 합니다.

  – API를 호출하여도 레지스트리 값을 읽는 비용이 들지 않습니다.

  – 향후의 연결 요청에 대해 새로운 사용 통계 문자열을 사용하지 않을 경우에는 이 API를 다시 호출할 필요가 없습니다. 레지스트리 값을 사용하고 있는 경우, 응용프로그램 프로세스를 종료하고 DB2ACCOUNT를 다시 정의한 다음 프로세스를 다시 시작해야 합니다.

  *Administrative API Reference*에서 이 API의 사용에 대한 자세한 내용을 참조하십시오.

- 클라이언트 워크스테이션에서 DB2ACCOUNT 레지스트리 값 사용

  sqlesact() API가 첫 번째 데이터베이스 연결 요청 이전에 호출되지 않으면, DB2ACCOUNT 레지스트리 값이 읽혀집니다. 사용 통계 문자열은 모든 후속 데이터베이스 연결 요청에 사용됩니다.

주: 사용 통계 문자열을 정의할 때 다음 규칙을 따라야 합니다.

1. 문자 A-Z, 0-9 또는 '_'(밑줄)을 사용합니다.
2. 사용 통계 문자열을 199 바이트로 제한합니다. 이보다 긴 문자열은 생략됩니다.

유용한 책

다음 책들은 분산 환경에서 수행하는 응용프로그램을 개발하는 데 도움이 됩니다.

- 특정 데이터베이스 제품용 응용프로그래밍 책에는 제품별로 서로 다른 정보가 들어 있습니다.
- 특정 데이터베이스 제품용 SQL 참조서를 사용하여, 응용프로그램에 올바른 구문으로 지원된 SQL문만 들어있는 것을 확인할 수 있습니다.
- The *원격 DRDA 리퀘스터 및 서버에 대한 OS/390용 DB2 Universal Database 참조서*에서는 OS/390용 DB2 Universal Database 사용자에 대한 사용 비용 청구 계정 관련 최신 정보를 제공합니다.
- *SQL 참조서*에서는 IBM 관계형 데이터베이스 제품간의 차이점에 대한 높은 수준의 논의를 제공하며 특정 차이점을 처리하는 방법도 논의합니다.
- DRDA 책은 DRDA 환경에서 계획, 연결성, 프로그래밍, 문제점 판별에 대한 정보를 제공합니다. 책 제목 및 구문 번호 목록은 21 페이지의 『관련 DRDA 온라인 문서』에서 자세한 내용을 참조하십시오.

제5장 사용자 응용프로그램 수행

다음과 같이 여러 가지 유형의 응용프로그램이 DB2 데이터베이스에 액세스할 수 있습니다.

- Embedded SQL, API, 저장 프로시듀어, DB2 CLI에 대한 사용자 정의 함수 또는 호출이 포함된 DB2 응용프로그램 개발 클라이언트를 사용하여 개발된 응용프로그램.
- ODBC 응용프로그램(예: Lotus Approach).
- JDBC 응용프로그램 및 애플릿.
- HTML과 SQL이 들어 있는 Net.Data 매크로

DB2 클라이언트상의 응용프로그램은 그의 실제 위치를 몰라도 원격 데이터베이스에 액세스할 수 있습니다. DB2 클라이언트는 데이터베이스의 위치를 판별하고, 데이터베이스 서버에 대한 요구의 전송을 관리하며, 그 결과를 리턴합니다.

일반적으로 데이터베이스 클라이언트 응용프로그램을 수행하려면 다음 단계를 사용하십시오.

단계 1. 서버가 구성되어 수행 중인지 확인하십시오.

데이터베이스 관리 프로그램이 응용프로그램이 연결 중인 데이터베이스 서버에서 시작되는지 확인하십시오. 그렇지 않은 경우에는 응용프로그램을 시작하기 전에 서버에서 **db2start** 명령을 발행해야 합니다.

단계 2. 응용프로그램이 사용하는 데이터베이스에 연결할 수 있는지 확인하십시오.

단계 3. 유틸리티와 응용프로그램을 데이터베이스로 바인드하십시오. 72 페이지의 『데이터베이스 유틸리티 바인딩』에서 유틸리티 바인딩에 대한 자세한 내용을 참조하십시오.

단계 4. 응용프로그램을 수행하십시오.

© Copyright IBM Corp. 1993, 2000

데이터베이스 유틸리티 바인딩

사용자는 데이터베이스 유틸리티들(가져오기, 내보내기,Reorg, 명령행 처리기) 및 DB2 CLI 바인드 파일들을 해당 데이터베이스에서 사용하기 전에 각 데이터베이스로 바인드해야 합니다. 네트워크 환경의 경우에 서로 다른 운영 체제에서 수행되거나 서로 다른 DB2 버전이나 서비스 레벨에 있는 여러 클라이언트들을 사용하면, 각 운영 체제 및 DB2 버전 조합에 대해 유틸리티들을 한번씩 바인드해야 합니다.

유틸리티를 바인딩하면 *패키지*가 작성되는데, 이는 단일 소스 파일로부터 특정 SQL 문을 처리하는 데 필요한 모든 정보가 들어 있는 오브젝트입니다.

바인드 파일은 설치 디렉토리하의 bnd 디렉토리에 있는 서로 다른 .lst 파일들로 함께 그룹화됩니다. 각 파일은 서버에 특정적입니다.

CLI/ODBC 프로그램 수행

DB2 콜 레벨 인터페이스 (CLI) 런타임 환경 및 DB2 CLI/ODBC 드라이버는 설치 동안 선택 구성요소로서 DB2 클라이언트에 포함됩니다.

이러한 지원을 통해 응용프로그램에서는 ODBC 및 DB2 CLI API를 사용하여 DB2 서버에 대해 작업하도록 개발했습니다. DB2 CLI 응용프로그램 개발 지원은 DB2 서버와 함께 제공된 DB2 응용프로그램 개발 클라이언트(DB2 응용프로그램 개발 클라이언트)에 의해 제공됩니다.

DB2 CLI 또는 ODBC 응용프로그램들은 DB2를 액세스하기 전에, DB2 CLI 패키지를 서버에서 바운드해야 합니다. 사용자에게 패키지를 바인드할 필수 권한이 있는 경우에는 첫 번째 연결에서 이러한 상황이 자동으로 발생한다 할지라도, 관리자는 서버에 액세스하는 각 플랫폼에서 클라이언트의 각 버전에 대해 이 작업을 맨먼저 수행하는 것이 좋습니다.

다음은 DB2 데이터베이스에 대한 DB2 CLI 및 ODBC 응용프로그램 액세스권을 부여하기 위해 클라이언트 시스템에서 필요한 일반적인 단계들입니다. 이 지시사항에서는 사용자가 유효한 사용자 ID 및 암호를 사용하여 DB2에 정상적으로

연결되었다고 가정합니다. 플랫폼에 따라 이 단계의 많은 부분이 자동입니다. 특별히 해당 플랫폼에 대해 다루는 절에서 자세한 내용을 참조하십시오.

단계 1. 인스턴스와 데이터베이스가 제어 센터에 알려질 수 있도록 클라이언트 구성 지원 프로그램(CCA)을사용하여 데이터베이스를 추가한 후 (개별적인 클라이언트 및 서버 머신이 있는 경우), 해당 시스템에 대한 인스턴스와 데이터베이스를 추가하십시오. (지역 시스템은 **지역** 아이콘으로 표시됩니다.) 이 프로그램에 대한 액세스권이 없으면, 명령행 처리기에서 **catalog** 명령을 사용할 수 있습니다.

단계 2. OS/2 이외의 다른 플랫폼에서는 DB2 CLI/ODBC 드라이버가 DB2 클라이언트 설치 과정에서 선택 가능한 구성요소입니다. 그때 드라이버를 선택했는지 확인하십시오. OS/2에서 DB2 CLI/ODBC 드라이버와 ODBC 드라이버 관리 프로그램 모두를 둘다 설치하려면 **ODBC 드라이버 설치** 아이콘을 사용해야 합니다.

단계 3. ODBC에서 DB2 데이터베이스에 액세스하려면 다음을 수행하십시오.

a. ODBC 드라이버 관리자(Microsoft 또는 다른 벤더로부터)가 이미 설치되어 있어야 합니다(이는 32비트 Windows 시스템상에서만 DB2 설치 동안에 기본값으로 수행됩니다).

b. DB2 데이터베이스가 ODBC 데이터 소스로서 등록되어야 합니다. ODBC 드라이버 관리 프로그램은 DB2 카탈로그 정보를 읽지 않고 대신, 데이터 소스에 대한 자신 고유의 목록을 참조합니다.

c. DB2 테이블에 고유 색인이 없으면 수많은 ODBC 응용프로그램들이 테이블을 읽기 전용으로 열 것입니다. ODBC 응용프로그램이 갱신할 각 DB2 테이블에 대해 고유 색인이 작성되어야 합니다. *SQL 참조서*에서 **CREATE INDEX**문에 대한 자세한 내용을 참조하십시오. 제어 센터를 사용하여 테이블의 설정값을 변경한 후 **기본 키** 탭을 선택하고 사용 가능한 컬럼 목록에서 기본 키 컬럼 목록으로 하나 이상의 컬럼을 이동합니다. 사용자가 기본 키의 일부분으로 선택한 컬럼은 NOT NULL로 정의되어야 합니다.

단계 4. 필요하다면, 여러 CLI/ODBC 구성 키워드를 설정하여 DB2 CLI/ODBC 및 그를 사용하는 응용프로그램의 동작을 수정할 수 있습니다.

위의 단계에 따라 ODBC 지원을 설치하고 DB2 데이터베이스를 ODBC 데이터 소스로서 추가했으면, 이제 ODBC 응용프로그램에서 이 데이터베이스들에 액세스할 수 있습니다.

CLI/ODBC 액세스에 대한 플랫폼 특정 세부사항

DB2에 대한 DB2 CLI 및 ODBC 응용프로그램 액세스를 제공하는 방법에 대한 플랫폼 특정 세부사항은 다음의 범주로 나눠집니다.

- 『CLI/ODBC를 사용하여 DB2에 대한 Windows 32비트 운영 체제 클라이언트 액세스』
- 76 페이지의 『CLI/ODBC를 사용하여 DB2에 대한 OS/2 클라이언트 액세스』

CLI/ODBC를 사용하여 DB2에 대한 Windows 32비트 운영 체제 클라이언트 액세스

DB2 CLI 및 ODBC 응용프로그램이 Windows 클라이언트로부터 DB2 데이터베이스에 정상적으로 액세스하기 전에, 클라이언트 시스템에서 다음의 단계를 수행하십시오.

단계 1. DB2 데이터베이스(및 데이터베이스가 원격인 경우에는 노드)가 카탈로그화되어야 합니다. 그러기 위해서는 CCA (또는 명령행 처리기)를 사용하십시오.

CCA의 온라인 도움말(또는 *Command Reference*의 **CATALOG DATABASE** 및 **CATALOG NODE** 명령)에서 자세한 내용을 참조하십시오.

단계 2. Microsoft ODBC 드라이버 관리자 및 DB2 CLI/ODBC 드라이버가 설치되었는지 확인하십시오. Windows 32비트 운영 체제에서 설치 동안에 ODBC 구성요소를 수동으로 선택해제하지 않는 한 이들 모두는 DB2와 함께 설치되어야 합니다.

머신에서 이들이 있는지 확인하려면,

a. 제어판에서 Microsoft ODBC 관리자를 수행하거나 명령행에서 적당한 명령을 발행하십시오. **odbcad32.exe.**

b. **ODBC 드라이버** 탭을 클릭하십시오.

c. ″IBM DB2 ODBC 드라이버″가 목록에 표시되는지 검증하십시오.

Microsoft ODBC 드라이버 관리자 또는 IBM DB2 CLI/ODBC 드라이버가 설치되지 않으면, Windows 32비트 운영 체제에 DB2 설치를 재수행하고 ODBC 구성요소를 선택하십시오.

단계 3. ODBC 드라이버 관리 프로그램과 함께 DB2 데이터베이스를 *데이터* 소스로서 등록하십시오. Windows 32비트 운영 체제에서 데이터 소스를 시스템의 모든 사용자(시스템 데이터 소스) 또는 현재 사용자만(사용자 데이터 소스) 사용 가능하게 만들 수 있습니다. 다음의 방법 중 하나를 사용하여 데이터 소스를 추가하십시오.

- CCA를 사용하는 방법:
  a. 데이터 소스로서 추가하려는 DB2 데이터베이스 별명을 선택하십시오.
  b. **등록 정보** 누름 버튼을 클릭하십시오. 데이터베이스 등록 정보 창이 열립니다.
  c. **이 데이터베이스를 ODBC용으로 등록** 선택란을 선택하십시오.
  d. Windows 32비트 운영 체제에서 단일선택 버튼을 사용하여 데이터 소스를 사용자 또는 시스템 데이터 소스로서 추가할 수 있습니다.
- Microsoft **32비트 ODBC 관리 도구**를 사용하는 방법: 이는 제어 패널의 아이콘으로부터 또는 명령행에서 **odbcad32.exe**를 수행하여 액세스할 수 있습니다.
  a. Windows 32비트 운영 체제에서는 사용자 데이터 소스의 목록이 기본값으로 표시됩니다. 시스템 데이터 소스를 추가하려면 **시스템 DSN** 버튼 또는 **시스템 DSN** 탭(플랫폼에 따라)을 클릭하십시오.
  b. **추가** 누름 버튼을 클릭하십시오.
  c. 목록에서 IBM DB2 ODBC 드라이버를 두 번 클릭하십시오.
  d. 추가할 DB2 데이터베이스를 선택한 후 **확인**을 클릭하십시오.
- Windows 32비트 운영 체제에서는 ODBC 드라이버 관리 프로그램과 함께 DB2 데이터베이스를 데이터 소스로서 등록하기 위해 명령행 처리기에서 발행할 수 있는 명령들이 있습니다. 관리자는 필수 데이터베이스를 등록하기 위해 명령행 처리기 스크립트를 작성할 수 있습니다.

그러면, ODBC를 통해 DB2 데이터베이스에 대한 액세스를 요청하는 모든 머신에서 이 스크립트를 수행할 수 있습니다.

*Command Reference*에는 CATALOG 명령에 대한 추가 정보가 들어 있습니다.

```
CATALOG [ user | system ] ODBC DATA SOURCE
```

단계 4. CCA를 사용하여 DB2 CLI/ODBC 드라이버를 구성하십시오(선택적).

a. 구성하려는 DB2 데이터베이스 별명을 선택하십시오.

b. **등록 정보** 누름 버튼을 클릭하십시오. 데이터베이스 등록 정보 창이 열립니다.

c. **설정** 누름 버튼을 클릭하십시오. CLI/ODBC 설정 창이 열립니다.

d. **고급** 누름 버튼을 클릭하십시오. 열린 창에 구성 키워드를 설정할 수 있습니다. 이 키워드들은 데이터베이스 *별명 이름*과 연관되며, 데이터베이스에 액세스하는 모든 DB2 CLI/ODBC 응용프로그램에 영향을 미칩니다. 온라인 도움말에서는 *설치 및 구성 보충 설명서* 온라인 매뉴얼의 부록에서처럼 모든 키워드에 대해 설명합니다.

단계 5. ODBC 액세스를 설치했으면(위에서 설명한대로), 이제 ODBC 응용프로그램을 사용하여 DB2 데이터에 액세스할 수 있습니다. ODBC 응용프로그램을 시작하여 열기 창으로 가십시오. **ODBC 데이터베이스** 파일 유형을 선택하십시오. 사용자가 ODBC 데이터 소스로서 추가한 DB2 데이터베이스를 목록에서 선택할 수 있습니다. 고유 색인이 존재하지 않는 한, 많은 ODBC 응용프로그램들은 읽기 전용으로 테이블을 열게 됩니다.

CLI/ODBC를 사용하여 DB2에 대한 OS/2 클라이언트 액세스

DB2 CLI 및 ODBC 응용프로그램이 OS/2 클라이언트로부터 DB2 데이터베이스에 정상적으로 액세스하기 전에, 클라이언트 시스템에서 다음의 단계를 수행하십시오.

1. DB2 데이터베이스(및 데이터베이스가 원격인 경우에는 노드)가 카탈로그화되어야 합니다. 그러기 위해서는 CCA(또는 명령행 처리기)를 사용하십시오.

   CCA에서 온라인 도움말에 대한 자세한 내용을 참조하십시오 (또는 *Command Reference*에서 **CATALOGDATABASE** 및 **CATALOG NODE** 명령).

2. ODBC 응용프로그램을 사용하여 DB2 데이터에 액세스할 경우에는 다음 단계를 수행하십시오. (CLI 응용프로그램만을 사용할 경우에는 이 단계를 생략하고 다음 단계로 가십시오.)

   a. ODBC 드라이버 관리자가 설치되어 있는지 확인하십시오. ODBC 드라이버 관리자는 DB2와 함께 설치되지 않습니다. ODBC 응용프로그램과 함께 제공된 드라이버 관리 프로그램을 사용하는 것이 바람직합니다. DB2 CLI/ODBC 드라이버가 설치되어 있는지도 확인하십시오.

      1) 문서에 설명된대로 ODBC 관리 도구를 수행하십시오. 이는 보통 다음의 두 가지 방법 중 하나로 수행됩니다.

         - OS/2에서 **ODBC** 폴더를 더블 클릭하고 **ODBC 관리자** 아이콘을 더블 클릭하십시오.
         - 명령행에서 **odbcadm.exe**를 수행하십시오.

         데이터 소스 창이 열립니다.

      2) **드라이버** 버튼을 클릭하십시오. 드라이버 창이 열립니다.

      3) "IBM DB2 ODBC 드라이버"가 목록에 표시되는지 검증하십시오.

      ODBC 드라이버 관리자가 설치되지 않았으면, ODBC 응용프로그램과 함께 제공된 설치 명령어를 따르십시오. IBM DB2 CLI/ODBC 드라이버가 설치되지 않은 경우에, DB2 CLI/ODBC 드라이버를 설치하려면 DB2 폴더의 **ODBC 드라이버 설치** 아이콘을 두 번 클릭하십시오.

   b. 다음의 방법 중 하나를 사용하여 ODBC 드라이버 관리자와 함께 DB2 데이터베이스를 *데이터 소스*로서 등록하십시오.

      - CCA를 사용하는 방법:

        1) 데이터 소스로서 추가하려는 DB2 데이터베이스 별명을 선택하십시오.

        2) **등록 정보** 누름 버튼을 클릭하십시오.

        3) **이 데이터베이스를 ODBC용으로 등록** 선택란을 선택하십시오.

      - ODBC 드라이버 관리자를 사용하는 방법:

        1) 문서에 설명된대로 ODBC 드라이버 관리자를 수행하십시오. 이는 보통 다음의 두 가지 방법 중 하나로 수행됩니다.

- OS/2의 **ODBC** 폴더를 두 번 클릭하고 **ODBC Administrator** 아이콘을 두 번 클릭하십시오.
- 명령행에서 **odbcadm.exe**를 수행하십시오.

2) 데이터 소스 창에서 **추가** 누름 버튼을 클릭하십시오. 데이터 소스 추가 창이 열립니다.
3) 목록에서 IBM DB2 ODBC 드라이버를 두 번 클릭하십시오.
4) 추가할 DB2 데이터베이스를 선택한 후 **확인**을 클릭하십시오.

3. CCA를 사용하여 DB2 CLI/ODBC 드라이버를 구성하십시오(선택적).
   a. 구성하려는 DB2 데이터베이스 별명을 선택하십시오.
   b. **등록 정보** 누름 버튼을 클릭하십시오. 데이터베이스 등록 정보 창이 열립니다.
   c. **설정** 누름 버튼을 클릭하십시오. CLI/ODBC 설정 창이 열립니다.
   d. **고급** 누름 버튼을 클릭하십시오. 나타나는 창에 구성 키워드를 설정할 수 있습니다. 이 키워드들은 데이터베이스 *별명 이름*과 연관되며, 데이터베이스에 액세스하는 모든 DB2 CLI/ODBC 응용프로그램에 영향을 미칩니다. 온라인 도움말에서는 *설치 및 구성 보충 설명서* 매뉴얼의 부록에서처럼 모든 키워드에 대해 설명합니다.
4. ODBC 액세스를 설치했으면(위에서 설명한대로), 이제 ODBC 응용프로그램을 사용하여 DB2 데이터에 액세스할 수 있습니다. ODBC 응용프로그램을 시작하여 열기 창으로 가십시오. **ODBC 데이터베이스** 파일 유형을 선택하십시오. 사용자가 ODBC 데이터 소스로서 추가한 DB2 데이터베이스를 목록에서 선택할 수 있습니다. 고유 색인이 존재하지 않는 한, 많은 ODBC 응용프로그램들은 읽기 전용으로 테이블을 열게 됩니다.

자세한 구성 정보

74 페이지의 『CLI/ODBC 액세스에 대한 플랫폼 특정 세부사항』 절에서는 사용자에게 필요한 모든 정보를 제공해야 합니다. *설치 및 구성 보충 설명서*에는 DB2 CLI 및 ODBC 응용프로그램의 설정과 사용에 대한 추가 정보가 들어 있습니다. (온라인 *설치 및 구성 보충 설명서* 매뉴얼은 디렉토리 *x*:\doc\en\html에 있으며, 여기서 *x:*는 사용자 CD-ROM의 위치이고 en은 사용자 언어를 나타내는 두문자의

국가 코드입니다. 예를 들면 영문에 대해서는 en입니다.) 이 정보는 DB2 도구 지원을 사용할 수 없고 더 많은 정보를 요구하는 관리자에게 유용한 정보입니다.

다음의 주제는 *설치 및 구성 보충 설명서* 온라인 매뉴얼에서 제공됩니다.

- DB2 CLI/ODBC 드라이버를 데이터베이스로 바인드하는 방법
- CLI/ODBC 구성 키워드를 설정하는 방법
- db2cli.ini 구성

Java 프로그램 수행

사용자는 AIX, HP-UX, Linux, OS/2, PTX, Silicon Graphics IRIX, Solaris 또는 Windows 32비트 운영 체제에서 해당 JDK(Java Development Kit)로 DB2 데이터베이스에 액세스하는 Java 프로그램을 개발할 수 있습니다. JDK에는 Java 용 동적 SQL API인 Java 데이터베이스 연결성(JDBC)이 포함됩니다.

DB2 JDBC 지원의 경우에는 DB2 클라이언트를 설치할 때 DB2 Java 지원 구성요소를 포함시켜야 합니다. DB2 JDBC 지원을 사용하여 JDBC 응용프로그램 및 애플릿을 빌드 및 수행할 수 있습니다. 이 지원에는 동적 SQL만 들어 있으며 Java 호출 인터페이스를 사용하여 SQL문을 DB2에 전달합니다.

DB2 응용프로그램 개발 클라이언트는 SQLJ(Java embedded SQL)에 대한 지원을 제공합니다. DB2 SQLJ 지원 및 DB2 JDBC 지원을 사용하여 SQLJ 응용프로그램 및 애플릿을 빌드 및 수행할 수 있습니다. 이 지원에는 정적 SQL이 들어 있으며 DB2 데이터베이스로 바인드되는 Embedded SQL문을 사용합니다.

Java는 JDBC 및 SQLJ 저장 프로시듀어와 사용자 정의 함수(UDF)를 작성하기 위해 서버에서 사용할 수 있습니다.

서로 다른 유형의 Java 프로그램을 빌드하여 수행하려면 서로 다른 DB2 구성요소로부터의 지원이 있어야 합니다.

- JDBC 응용프로그램을 빌드하려면 DB2 Java 지원 구성요소가 있는 DB2 클라이언트를 설치해야 합니다. JDBC 응용프로그램을 수행하려면 DB2 Java 지원 구성요소가 있는 DB2 클라이언트가 DB2 서버에 연결되어야 합니다.

- SQLJ 응용프로그램을 빌드하려면 DB2 Java 지원 구성요소가 있는 DB2 SDK 및 DB2 관리 클라이언트를 설치해야 합니다. SQLJ 응용프로그램을 수행하려면 DB2 Java 지원 구성요소가 있는 DB2 클라이언트가 DB2 서버에 연결되어야 합니다.
- JDBC 애플릿을 빌드하려면 DB2 Java 지원 구성요소가 있는 DB2 클라이언트를 설치해야 합니다. DB2 애플릿을 수행하는 데는 클라이언트 머신에 어떠한 DB2 구성요소도 필요하지 않습니다.
- SQLJ 애플릿을 빌드하려면 DB2 Java 지원 구성요소가 있는 DB2 SDK 및 DB2 관리 클라이언트를 설치해야 합니다. SQLJ 애플릿을 수행하는 데는 클라이언트 머신에 어떠한 DB2 구성요소도 필요하지 않습니다.

*응용프로그램 빌드 안내서*에서 JDBC 및 SQLJ 프로그램의 빌드 및 수행에 대한 자세한 내용을 참조하십시오. *응용프로그램 개발 안내서*에서 Java로 DB2 프로그래밍하는 데에 대한 자세한 내용을 참조하십시오. 여기에서는 JDBC 및 SQLJ 응용프로그램, 애플릿, 저장 프로시듀어 및 UDF의 작성 및 수행에 대해 다룹니다.

최신의 DB2 Java 정보는 다음 웹 페이지를 방문하십시오.

```
http://www.ibm.com/software/data/db2/java
```

환경 구성

DB2 Java 프로그램을 빌드하고 수행하려면 사용하는 개발 머신에 JDK(Java Development Kit)의 해당 버전을 설치 및 구성해야 합니다.

AIX: IBM에서 Java Development Kit (JDK) 버전 1.1.2.

HP-UX:

HP에서 Java용 HP-UX Developer's Kit 릴리스 1.1.3 이상.

Linux:

Blackdown Organization에서 Linux용 Java Development Kit (JDK) 1.1.7 버전 1a 이상.

OS/2: IBM에서 OS/2용 Java Development Kit (JDK) 버전 1.1.4.

PTX: IBM에서 Java Development Kit (JDK), ptx/JSE, v3.0.0 (Sun JDK v1.2.1).

Silicon Graphics IRIX:

Silicon Graphics, Inc.에서 Java Development Environment 3.1 (Sun JDK 1.1.5) 및 Java Execution Environment 3.1 (Sun JRE 1.1.5).

Solaris:

Sun Microsystems에서 Solaris용 Java Development Kit (JDK) 버전 1.1.4 및 Solaris Native Thread pack.

Windows 32비트 운영 체제

Sun Microsystems에서 Win32용 Java Development Kit (JDK) 버전 1.1.

앞의 JDK를 설치하고 구성하는 정보에 대해서는 다음을 참조하십시오.

```
http://www.ibm.com/software/data/db2/java
```

지원되는 모든 플랫폼에 대해서도 DB2 Java 지원 구성요소가 있는 DB2 클라이언트를 설치 및 구성해야 합니다. SQLJ 프로그램을 데이터베이스에 바인드하려면 DB2 Java 지원 구성요소가 있는 DB2 관리 클라이언트를 설치 및 구성해야 합니다.

DB2 Java 저장 프로시듀어 또는 UDF를 수행하려면, 사용하는 개발 머신에 JDK 버전 1.1이 설치된 경로를 포함시키기 위해 DB2 데이터베이스 관리 프로그램 구성도 갱신해야 합니다. 이는 명령행에 다음을 입력하여 수행할 수 있습니다.

```
db2 update dbm cfg using JDK11_PATH /home/smith/jdk11
```

여기서, `/home/smith/jdk11`은 JDK 버전 1.1이 설치된 경로를 나타냅니다.

`JDK11_PATH` 필드에 대한 값이 올바른지 검증하기 위해 다음 명령을 입력하여 DB2 데이터베이스 관리 프로그램 구성을 점검할 수 있습니다.

```
db2 get dbm cfg
```

보다 편리하게 보기 위해 출력을 파일로 파이프시킬 수도 있습니다. `JDK11_PATH` 필드는 출력의 시작 부분에 나타납니다. *Command Reference*에서 이 명령에 대한 자세한 내용을 참조하십시오.

Solaris의 경우에는 일부 Java 가상 머신이 "setuid" 환경으로 수행하는 프로그램에서 잘 작동하지 않습니다. Java 번역기인 libjava.so가 들어 있는 공유 라이브러리가 로드하는 데 실패했을 수도 있습니다. 일시적인 해결책으로서, 필요한 모든 JVM 공유 라이브러리에 대한 심볼릭 링크를 다음과 유사한 명령(Java가 머신에 설치된 위치에 따라 다름)으로 /usr/lib에 작성할 수 있습니다.

```
ln -s /opt/jdk1.1.3/lib/sparc/native_threads/*.so /usr/lib
```

보다 자세한 정보는 다음을 참조하십시오.

```
http://www.ibm.com/software/data/db2/java/v5/faq.html
```

Java 프로그램을 수행하기 위해서 OS/2 및 Windows 운영 체제상의 DB2 설치 동안에 그리고 UNIX 플랫폼상의 인스턴스 작성 동안에 다음의 환경 변수들이 자동으로 갱신됩니다.

UNIX 플랫폼에서:

- CLASSPATH는 "."과 파일 sqllib/java/db2java.zip을 포함합니다.
- AIX, Linux, Silicon Graphics IRIX 및 Solaris 경우에는 LD_LIBRARY_PATH에 디렉토리 sqllib/lib가 포함되며, HP-UX 경우에는 SHLIB_PATH에 디렉토리 sqllib/lib가 포함됩니다.
- AIX, Linux, PTX, Silicon Graphics IRIX 및 Solaris 경우에는 LD_LIBRARY_PATH에 디렉토리 sqllib/lib가 포함되며, HP-UX 경우에는 SHLIB_PATH에 디렉토리 sqllib/lib가 포함됩니다.
- Solaris 경우에만 THREADS_FLAG가 "native"로 설정됩니다.

Windows 및 OS/2 플랫폼에서:

- CLASSPATH는 "."과 파일 %DB2PATH%\java\db2java.zip을 포함합니다.

SQLJ 프로그램을 구축하고 수행하려면, CLASSPATH는 또한 이들 파일을 포함하기 위해 자동적으로 갱신됩니다.

UNIX 플랫폼에서:

- sqllib/java/sqlj.zip (SQLJ 프로그램을 구축하기 위해)
- sqllib/java/runtime.zip (SQLJ 프로그램을 수행하기 위해)

Windows 및 OS/2 플랫폼에서:

- %DB2PATH%\java\sqlj.zip (SQLJ 프로그램을 구축하기 위해)
- %DB2PATH%\java\runtime.zip (SQLJ 프로그램을 수행하기 위해)

Java 응용프로그램

다음의 명령으로 실행 프로그램에서 Java 번역기를 수행하여 데스크탑 또는 명령행으로부터 응용프로그램을 시작하십시오.

```
java prog_name
```

여기서, prog_name은 프로그램의 이름입니다.

DB2 JDBC 드라이버는 응용프로그램으로부터 JDBC API 호출을 핸들하고 DB2 CAE를 사용하여 서버에 대한 요청을 통신하고 결과를 수신합니다. SQLJ 응용프로그램은 수행되기 전에 데이터베이스에 바인드되어야 합니다.

Java 애플릿

Java 애플릿은 웹을 통해 전달되기 때문에 DB2 머신(서버 또는 클라이언트)에 웹 서버가 설치되어 있어야 합니다.

애플릿을 수행하려면 .html 파일이 제대로 구성되어 있는지 확인하십시오. .html 파일에 지정된 TCP/IP 포트에서 JDBC 애플릿 서버를 시작하십시오. 예를 들어, 다음과 같이 지정한 경우에는,

```
param name=port value='6789'
```

아래와 같이 입력하십시오.

```
db2jstrt 6789
```

웹 브라우저에서 작업 디렉토리를 액세스할 수 있는지 확인해야 합니다. 액세스하지 못하는 경우에는 애플릿의 .class 및 .html 파일을 액세스 가능한 디렉토리로 복사하십시오. SQLJ 애플릿의 경우에는 프로파일 .class 및 .ser 파일도 복사해야 합니다.

sqllib/java/db2java.zip 파일을 다른 파일들과 동일한 디렉토리로 복사하십시오. SQLJ 애플릿의 경우에는 sqllib/java/runtime.zip 파일도 이 디렉토리

로 복사하십시오. 그런 다음, 클라이언트 머신에서 웹 브라우저(JDK 1.1을 지원하는)를 시작하여 `.html` 파일을 로드하십시오.

DB2에 연결하기 위해 애플릿이 JDBC API를 호출할 때는, JDBC 드라이버가 DB2 서버에 상주하는 JDBC 애플릿 서버를 통해 DB2 데이터베이스와 개별적인 통신을 설정합니다. SQLJ 애플릿은 수행되기 전에 데이터베이스에 바인드되어야 합니다.

제2부 참조 및 문제점 해결

© Copyright IBM Corp. 1993, 2000

제6장 데이터베이스 디렉토리 갱신

DB2 Connect는 다음 디렉토리를 사용하여 연결되는 데이터베이스에 대한 정보를 관리합니다.

- 노드 *디렉토리*에는 DB2 Connect가 액세스하는 모든 호스트나 AS/400 데이터베이스 서버에 대한 네트워크 주소와 통신 프로토콜 정보가 들어 있습니다.
- *데이터베이스 연결 서비스*(DCS) 디렉토리에는 호스트나 AS/400 데이터베이스 서버 데이터베이스에 대한 정보가 들어 있습니다.
- *시스템 데이터베이스 디렉토리*에는 DB2 Connect가 액세스하는 모든 데이터베이스의 이름, 노드, 인증 정보가 들어 있습니다.

주:

1. 이 디렉토리를 갱신하기 전에, 호스트나 AS/400 데이터베이스 서버와 워크스테이션상에 통신을 구성해야 합니다. *설치 및 구성 보충 설명서*에서 자세한 내용을 참조하십시오.
2. OS/2, Windows 32비트 운영 체제에서 데이터베이스 디렉토리는 DB2 Universal Database 클라이언트 구성 지원 프로그램(CCA)을 사용하여 갱신할 수 있습니다.

   다른 모든 플랫폼에서, 데이터베이스 디렉토리는 DB2 명령행 처리기(CLP)를 사용하여 갱신해야 합니다.
3. 100 페이지의 『디렉토리 갱신』에서는 예제 명령 구문을 제공합니다. *Command Reference*에서 자세한 내용을 참조하십시오.
4. 연결하는 각 호스트 또는 AS/400 데이터베이스 서버 데이터베이스에 대해 DCE를 사용할 경우, 이 디렉토리들을 갱신하거나 전역 DCE 디렉토리에 해당 정보를 저장해야 합니다. DCE에 대해서는 229 페이지의 『부록D. DCE 디렉토리 서비스 사용』과 *관리 안내서*에서 자세한 내용을 참조하십시오. 이 절에서는 사용자가 DCE 디렉토리 서비스를 사용하지 않는다고 간주합니다.

© Copyright IBM Corp. 1993, 2000

정보 수집

223 페이지의 『부록B. 디렉토리 사용자 정의 워크쉬트』는 수집해야 하는 정보를 보여 줍니다. 해당 워크쉬트의 사본을 만들어 사용하는 시스템의 값을 기록해 두면 편리합니다.

노드 디렉토리

노드 디렉토리에 다음 정보를 지정할 수 있습니다.

노드 이름

원격 데이터베이스가 상주하는 호스트나 AS/400 데이터베이스 서버 시스템의 별명. 이 이름은 사용자가 정의합니다. 노드 디렉토리 매개변수 테이블과 시스템 데이터베이스 디렉토리 매개변수 테이블 모두에 동일한 노드 이름을 쓰십시오.

형식: 1–8자의 1바이트 영숫자(숫자 기호(#), at 기호(@), 달러 기호($) 및 밑줄(_) 등). 이 형식은 밑줄 문자나 숫자로 시작할 수 없습니다.

프로토콜

APPC 또는 TCP/IP

기호 목적지 이름

APPC 노드를 정의할 때, CPI 통신 사이드 정보 테이블(예를 들어, Microsoft SNA 서버를 사용할 때 CPI-C 기호 목적지 특성 이름)에서 지정되었던 기호 목적지 이름을 사용하십시오. SNA를 설치했거나 구성했던 사람으로부터 이 값을 확보해야 합니다. 기호 목적지 이름에서는 대문자와 소문자가 구별됩니다(이름에서 대문자와 소문자가 일치하지 않으면 SQL1338 리턴 코드가 수신됩니다).

보안 유형

보안 유형 확인이 수행됩니다. APPC 노드에서 유효한 값은 SAME, PROGRAM 및 NONE입니다. TCP/IP 노드에서, SECURITY SOCKS는 노드가 SOCKS-작동 가능하기 위해 지정하는 옵션이며, 이런 경우 SOCKS_NS 및 SOCKS_SERVER 환경 변수는 필수적이고 SOCKS를 작동 가능하기 위해 설정되어야 합니다. 129 페이지의 『제10장 보안』과 *Command Reference*에서 자세한 내용을 참조하십시오.

TCP/IP 원격 호스트 이름 또는 IP 주소

TCP/IP 노드를 정의할 때 원격 TCP/IP 호스트 이름이나 원격 TCP/IP 주소. 호스트 이름이 지정되면, 도메인 이름 서버(DNS) 찾아보기나 지역 TCP/IP 호스트 항목을 통해 DB2 Connect 워크스테이션에서 이름이 해결되어야 합니다.

OS/390용 DB2 원격 호스트의 호스트 이름은 분산 데이터 기능(DDF)이 시작될 때 DSNL004I 메시지(DOMAIN=호스트 이름)에 나타납니다.

TCP/IP 서비스 이름 또는 포트 번호

TCP/IP 노드를 정의할 때 원격 TCP/IP 서비스 이름이나 포트 번호. 이것은 원격 호스트에서 TCP/IP로 정의되어야 합니다. 포트 번호 446은 DRDA용 기본 포트 번호로 등록되어 있습니다.

OS/390용 DB2 원격 호스트의 포트 번호는 부트스트랩 데이터 세트(BSDS)에서 PORT로 정의되며 분산 데이터 기능(DDF)이 시작될 때 DSNL004I 메시지(TCPPORT=포트 번호)에도 제공됩니다.

주: TCP/IP 연결을 통해 2단계 확약 재동기화 조작에 사용되는 두 번째 포트는 서버가 지정합니다. 예를 들어, OS/390용 DB2 Universal Database 부트스트랩 데이터 세트는 OS/390용 DB2 Universal Database와의 인바운드 연결을 위한 재동기화에만 사용되는 포트 번호(RESPORT)를 지정합니다. 여기서는 서비스 이름을 정의할 필요가 없습니다.

DCS 디렉토리

DCS 디렉토리에 다음 정보를 지정할 수 있습니다.

데이터베이스 이름

호스트나 AS/400 데이터베이스 서버에 대한 사용자 정의 별명. DCS 디렉토리 매개변수 테이블과 시스템 데이터베이스 디렉토리 매개변수 테이블 모두에 동일한 데이터베이스 이름을 사용하십시오.

형식: 1–8자의 1바이트 영숫자(숫자 기호(#), at 기호(@), 달러 기호($) 및 밑줄(_) 등). 이 형식은 밑줄 문자나 숫자로 시작할 수 없습니다.

목표 데이터베이스 이름

호스트나 AS/400 데이터베이스 서버 시스템의 데이터베이스.

MVS/ESA

해당 LOCATION NAME으로 식별되는 OS/390용 DB2 Universal Database 서브시스템

LOCATION NAME은 TSO에 로그인해서, 사용 가능한 조회 도구 중 하나를 사용하여 다음의 SQL 조회를 발행함으로써 결정될 수 있습니다.

```
select current server from sysibm.sysdummy1
```

LOCATION NAME은 MVS/ESA 부트 스트랩 데이터 세트(BSDS)에 정의되며 분산 데이터 기능(DDF)이 시작될 때 작성되는 DSNL004I 메시지(LOCATION=위치)에도 정의됩니다.

OS/390

해당 LOCATION NAME으로 식별되는 OS/390용 DB2 Universal Database 서브시스템

LOCATION NAME은 TSO에 로그인해서, 사용 가능한 조회 도구 중 하나를 사용하여 다음의 SQL 조회를 발행함으로써 결정될 수 있습니다.

```
select current server from sysibm.sysdummy1
```

LOCATION NAME은 부트 스트랩 데이터 세트(BSDS)에 정의되며 분산 데이터 기능(DDF)이 시작될 때 작성되는 DSNL004I 메시지(LOCATION=위치)에도 정의됩니다.

VSE 또는 VM

데이터베이스 이름(DBNAME)

OS/400

관계형 데이터베이스 이름(RDBNAME)

기타 OS/2, Windows NT, Windows 2000 및 UNIX 기반 시스템의 경우에 데이터베이스 디렉토리에 있는 데이터베이스 별명.

응용프로그램 리퀘스터 이름

SQL 요청을 DRDA 응용프로그램 서버(AS)로 전달하는 응용프로그램 리퀘스터 이름. 응용프로그램 리퀘스터는 응용프로그램을대신하여 요청을 처리합니다.

형식: AR <application_requester_name>

기본 값은 DB2 Connect 응용프로그램 리퀘스터입니다

매개변수 문자열

기본 값을 변경하려면, 다음 순서로 아래 매개변수의 일부나 전부를 지정하십시오. 매개변수 문자열은 클라이언트 구성 지원 프로그램을 사용하여 설정할 수 없으며, CLP를 사용할 경우 매개변수 문자열을 작은 따옴표(예: OS/2나 Windows NT에서) 또는 큰 따옴표(예: AIX에서)로 묶어야 합니다.

맵 파일

기본 SQLCODE 맵핑을 겹쳐쓰는 SQLCODE 맵핑 파일 이름. SQLCODE 맵핑을 작동 중지하려면, **NOMAP**을 지정하십시오. 141 페이지의 『제11장 SQLCODE 맵핑』에서 더 자세한 정보를 참조하십시오.

,D 이것은 두 번째 위치 매개변수입니다. 이것을 지정할 경우, 다음 SQLCODES 중 하나가 리턴될 때 응용프로그램이 DRDA 서버 데이터베이스와 연결을 해제합니다.

```
SQL30000N
SQL30040N
SQL30050N
SQL30051N
SQL30053N
SQL30060N
SQL30070N
SQL30071N
SQL30072N
SQL30073N
SQL30074N
SQL30090N
```

연결해제 매개변수, **D**가 지정되지 않으면, 다음 SQLCODES가 리턴되는 경우에만 연결이 해제됩니다.

```
SQL30020N
SQL30021N
SQL30041N
SQL30061N
SQL30081N
```

*메시지 참조서*에서 이들 코드에 대한 자세한 내용을 참조하십시오.

주: DB2 Connect가 오류로 인해 연결이 해제되면, 자동적으로 구간 복원이 됩니다.

„INTERRUPT_ENABLED

이것은 세 번째 위치 매개변수입니다. INTERRUPT_ENABLED가 DB2 Connect 워크스테이션의 DCS 디렉토리에서 구성되고 클라이언트 응용프로그램이 호스트나 AS/400 데이터베이스 서버에 연결될 때 인터럽트를 발행하면, DB2 Connect는 연결을 삭제하고 작업 단위(UOW)를 구간 복원하여 인터럽트를 실행합니다. 이 인터럽트 작동은 AIX, OS/2, Windows NT 및 Windows 2000에서 지원됩니다.

응용프로그램은 서버와의 연결이 종료됨을 나타내는 sqlcode(-30081)를 수신합니다. 그런 다음 응용프로그램은 추가 데이터베이스 요청을 처리하기 위해, 호스트나 AS/400 데이터베이스 서버에서 새로운 연결을 수행합니다. AIX V4.1 이상, SNA 서버 V3.1 이상, OS/2, Windows NT 및 Windows 2000 이외의 플랫폼에서, DB2 Connect는 이를 사용하는 응용프로그램이 인터럽트 요청을 수신할 때 자동으로 연결을 끊는 옵션을 지원하지 않습니다.

주: 이런 지원은 모든 플랫폼의 TCP/IP 연결에 대해 작동합니다. 클라이언트는 소켓을 삭제할 수도 있지만, 서버 구현에 따라 확실한 수신이 될 수도 있고 아닐 수도 있습니다. OS/390용 DB2 Universal Database

는 비동기 소켓 호출을 사용하여 진행 중인 다른 긴 SQL문의 연결 및 구간 복원 손실을 검출할 수 있습니다.

,,,,,SYSPLEX

6번째 위치 매개변수인 이 매개변수는 특정 데이터베이스를 위해 DB2 Connect SYSPLEX 지원을 명시적으로 작동가능하게 할 때 사용할 수 있습니다.

또한 새 프로파일(환경 또는 레지스트리) 변수 DB2SYSPLEX_SERVER가 도입되어 워크스테이션 레벨에서 SYSPLEX 지원을 작동불가능하게 할 때 사용할 수 있음을 유의하십시오.

,,,,,LOCALDATE="<value>"

7번째 위치 매개변수인 이 매개변수는 DB2 Connect 날짜 형식화 지원을 작동가능하게 할 때 사용됩니다. 이는 다음과 같이 <*value*>의 날짜 마스크를 사용하여 구현됩니다.

다음 CLP(명령행 처리기) 명령문을 실행한다고 가정해 보십시오.

```
catalog appc node nynode remote nycpic security program
catalog dcs database nydb1 as new_york
catalog database nydb1 as newyork1 at node nynode
            authentication dcs
```

데이터베이스 별명 *newyork1*은 날짜 마스크가 지정되지 않았으므로 날짜 변형 없이 호스트 데이터베이스를 액세스하기 위해 사용됩니다.

그러나 이제는 새로운 날짜 형식화를 지원하므로 다음 CLP 명령을 사용할 수 있습니다. 이 경우, CLP가 사용되고 매개변수 문자열 자체는 큰 따옴표를 사용하여 지정하므로, 두쌍의 큰 따옴표 안에 LOCALDATE 값을 지정해야 합니다. LOCALDATE 스펙에서 큰 따옴표가 제

거되지 않도록 운영 체제의 Escape 문자 "\"(역슬래시) 사용에 유의하십시오. 98 페이지의 『매개변수 문자열 지정』도 참조하십시오.

```
catalog dcs database nydb2 as new_york
     parms \",,,,,,LOCALDATE=\"\"YYYYMMDD\"\"\"
catalog database nydb2 as newyork2 at node nynode
             authentication dcs
```

데이터베이스 별명 "newyork2"를 사용하면 같은 호스트 데이터베이스를 액세스할 수 있지만, 그 외에도 이 별명에는 날짜 형식 마스크가 지정되었습니다. 이 예는 날짜 형식 마스크가 LOCALDATE 키워드를 사용하여 지정되며 DCS 디렉토리 항목의 PARMS 필드에서 7번째 위치 매개변수임을 보여 줍니다.

날짜 마스크가 유효하려면 아래의 조건이 모두 충족되어야 합니다.

1. Y, M, D 중 기껏해야 한 개만이 있을 수 있습니다. 여기서 Y는 연도 자리, M은 월 그리고 D는 날짜를 나타내는 자리입니다.
2. Y의 최대수는 4입니다.
3. M의 최대수는 2입니다.
4. D의 최대수는 2입니다.

예를 들어, 다음은 모두 유효한 날짜 마스크입니다.

```
"YYyyMmDd"    - Y, M, D 자리는 대소문자를
                구별하지 않습니다.
"MM+DD+YYYY" - 마스크가 10바이트보다 길고
                마스크에 Y, M, D 이외의 문자를
                사용해도 좋습니다.
"abcYY+MM"    - D 순서는 사용하지 않아도 좋습니다.
```

다음은 모두 무효 날짜 마스크입니다.

```
"YYYYyMMDD"   - Y가 5개이므로 잘못됨
"YYYYMDDM"    - M 순서가 두 번 사용되었으므로
                잘못됨
```

날짜 형식 마스크가 잘못되었더라도 오류는 표시되지 않습니다. 무시될 뿐입니다. 날짜 마스크가 유효하다고 해서 사용되리라는 것을 의미하지는 않습니다. 유효한 날짜 마스크에 기초를 둔 날짜 형식 변형은 다음 조건이 모두 충족되는 경우에만 수행됩니다.

1. SQL 오류가 없습니다.
2. 출력은 ISO 같은(ISO와 JIS) 형식으로 된 날짜 값입니다.
3. 출력 데이터 영역의 길이는 최소한 10바이트입니다. 이것은 날짜 형식 변형이 수행되지 않더라도 날짜 값이 출력 데이터 영역에 저장되도록 하기 위한 최소 출력 데이터 영역 크기입니다. 이 요구 사항은 날짜 형식 마스크가 10바이트보다 짧은 경우에도 적용됩니다.
4. DCS 디렉토리 항목에 유효한 날짜 형식 마스크가 지정되며 이 마스크는 출력 데이터 영역에 들어갑니다.

,,,,,,,CHGPWD_SDN=<name>

8번째 위치 매개변수인 이 매개변수는 암호 만기 관리(PEM)에 사용되는 기호 목적지 이름을 지정하는 데 사용됩니다. <name>에 지정되는 값은 대소문자가 구별됩니다.

135 페이지의 『MVS 암호 변경』에서는 다음과 같이 CHGPWD_SDN을 사용하여 dcs 데이터베이스 디렉토리를 카탈로그화하는 예를 보여 줍니다.

```
catalog dcs database db1 as dsn_db_1 parms ",,,,,,,CHGPWD_SDN=pempgm"
```

,,,,,,,BIDI=<ccsid>

8번째 위치 매개변수인 이 매개변수는 기본 서버 데이터베이스 BiDi CCSID를 대체하는 양방향(Bidi) CCSID를 지정하는 데 사용됩니다. 예를 들면, 다음과 같습니다.

```
",,,,,,,,BIDI=xyz"
```

여기서 *xyz*는 CCSID 대체입니다(97 페이지의 1 참조).

*관리 안내서*에서 문자열 유형과 더불어 지원되는 BiDi CCSID 목록에 대한 자세한 내용을 참조하십시오.

서로 다른 플랫폼에서 BiDi 데이터를 정확히 처리하려면 다음과 같은 BiDi 속성이 필요합니다.

- 숫자 형태(ARABIC 대 HINDI)
- 방향(RIGHT-TO-LEFT 대 LEFT-TO-RIGHT)
- 형태(SHAPED 대 UNSHAPED)
- 대칭 스와핑(YES 또는 NO)
- 텍스트 유형(LOGICAL 대 VISUAL)

각 플랫폼의 기본 값이 같지 않으므로, 한 플랫폼에서 다른 플랫폼으로 DB2 데이터가 전송될 때 문제가 발생합니다. 예를 들어, Windows 플랫폼은 LOGICAL UNSHAPED 데이터를 사용하는 반면, MVS 및 OS/390의 데이터는 보통 SHAPED VISUAL 형식을 취합니다. 따라서, BiDi 속성 지원 없이 MVS 또는 OS/390용 DB2에서 Windows의 DB2 Connect로 전송되는 데이터는 틀리게 표시됩니다.

DB2 Connect와 서버의 데이터베이스 사이에 데이터가 교환되는 경우, 이것이 주로 수신 데이터에 대한 변환을 수행하는 수신자입니다. 동일한 규칙이 일반 코드 페이지 변환에 추가하여 BiDi 레이아웃 변환에도 적용됩니다. 그러나, 현재로서는 호스트 DB2 제품이 BiDi 특정 CCSID 또는 BiDi 레이아웃 변환을 지원하지 않습니다. 따라서, 서버 데이터베이스에서 수신되는 데이터 외에 서버 데이터베이스로 전송되는 데이터에 대해 BiDi 레이아웃 변환을 수행할 수 있도록 추가 기능이 DB2 Connect에 포함되어 있습니다.

DB2 Connect가 서버 데이터베이스로 수신되는 데이터에 대해 BiDi 레이아웃 변환을 수행하려면, 서버 데이터베이스의 BiDi CCSID를 대체해야 합니다(98 페이지의 2 참조). 이것은 서버 데이터베이스에 대한 DCS 데이터베이스 디렉토리 항목의 PARMS 필드에서 BIDI 매개변수를 사용하여 수행됩니다.

이 기능의 사용은 예제에 가장 잘 표시되어 있습니다.

CCSID 62213(BiDi 문자열 유형 5)을 실행하는 Hebrew DB2 클라이언트를 가지고 있고 CCSID 424(BiDi 문자열 유형 4)를 실행하는 DB2 호스트 데이터베이스에 액세스하려 한다고 가정합시다. 그러나 DB2 호스트 데이터베이스에 들어 있는 데이터는 CCSID 8616(BiDi 문자열 유형 6)에 기반하고 있다는 것은 알고 있습니다.

이 경우에는 두 가지 문제가 발생합니다. 첫 번째 문제는 DB2 호스트 데이터베이스가 CCSID 424 및 8616를 갖는 BiDi 문자열 유형 사이의 차이를 모른다는 점입니다. 두 번째 문제는 DB2 호스트 데이터베이스가 62213의 DB2 클라이언트 CCSID를 인식하지 못한다는 것입니다. 이는 CCSID 62213과 동일한 코드 페이지에 기반하고 있는 CCSID 862만을 지원합니다.

DB2 호스트 데이터베이스에 전송된 데이터가 BiDi 문자열 유형 6 형식으로 시작되도록 하고 또한 DB2 Connect가 DB2 호스트 데이터베이스로부터 수신하는 데이터에 대해 BiDi 레이아웃 변환을 수행해야 함을 알게 해야 합니다. DB2 호스트 데이터베이스에 대해 다음과 같이 카탈로그화할 수 있습니다.

```
catalog dcs database nydb1 as TELAVIV parms ",,,,,,,,BIDI=8616"
```

이는 DB2 호스트 데이터베이스 CCSID 424를 8616으로 대체하도록 DB2 Connect에 지시합니다. 이 대체 작업에는 다음과 같은 처리가 포함됩니다.

1. DB2 Connect가 CCSID 862를 사용하여 DB2 호스트 데이터베이스에 연결합니다.
2. DB2 Connect가 CCSID 62213(BiDi 문자열 유형 5)에서 CCSID 62221(BiDi 문자열 유형 6)으로 전송되는 데이터에 대해 BiDi 레이아웃 변환을 수행합니다.
3. DB2 Connect는 CCSID 8616(BiDi 문자열 유형 6)에서 CCSID 62213(BiDi 문자열 유형 5)으로 DB2 호스트 데이터베이스로부터 수신하는 데이터에 대해 BiDi 레이아웃 변환을 수행합니다.

주:

1. BIDI 매개변수가 효력을 갖게 하려면 환경 변수나 레지스트리 값 DB2BIDI가 YES로 설정되어야 합니다.

2. DB2 Connect를 사용하여 CCSID를 대체할 필요가 없는데도 DB2 호스트 데이터베이스로 전송할 데이터에 대해 레이아웃 변환을 수행하려는 경우에는, DCS 데이터베이스 디렉토리 PARMS 필드에 BIDI 매개변수를 추가해야 합니다. 이 경우, 제공해야 하는 CCSID는 기본 DB2 호스트 데이터베이스 CCSID입니다.
3. 어떤 경우에는 양방향 CCSID를 사용하면 SQL 조회 자체가 변경되어 DB2 서버가 이를 인식하지 못하는 경우도 있습니다. 특히, 서로 다른 문자열 유형을 사용할 수 있는 경우에는 IMPLICIT CONTEXTUAL과 IMPLICIT RIGHT-TO-LEFT CCSID를 사용하지 말아야 합니다. CONTEXTUAL CCSID는 SQL 조회에 따옴표로 묶인 문자열이 들어 있는 경우 예상하지 못한 결과를 초래할 수 있습니다. SQL문에는 따옴표가 있는 문자열을 사용하지 말고 가능할 경우 호스트 변수를 대신 사용하십시오.

   특정 양방향 CCSID가 이러한 조건을 이행하고도 치유할 수 없는 문제를 초래하는 경우, 환경 변수나 레지스트리 값 DB2BIDI를 NO로 설정해야 합니다.

매개변수 문자열 지정

다음은 사용자가 지정할 수 있는 몇 개의 매개변수 문자열을 예한 것입니다.

예를 들어 다음과 같이 지정할 수 있는데, 여기서 "\"(역슬래시)는 운영 체제 Escape 문자입니다.

AIX에서:

```
NOMAP
/u/username/sqllib/map/dcs1new.map,D
,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE=\"\"YYMMDD\"\",,
```

OS/2, Windows NT 또는 Windows 2000에서:

```
NOMAP
d:\sqllib\map\dcs1new.map,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE=\"\"YYMMDD\"\",,
```

또는 매개변수 문자열을 지정하지 않고 기본 값을 사용할 수 있습니다.

주: 매개변수 문자열에서 LOCALDATE 마스크를 지정할 때 두 쌍의 큰 따옴표를 지정해야 하는 필요성 때문에 운영 체제 Escape 문자 "\"(역슬래시)를 사용해야 합니다. 예를 들면, 다음과 같습니다.

```
db2 catalog dcs db x as y parms \",,,,,,LOCALDATE=\"\"YYMMDD\"\"\"
```

이 결과 다음과 같은 DCS 디렉토리 항목이 생깁니다.

```
DCS 1 entry:

 Local database name                = X
 Target database name               = Y
 Application requestor name         =
 DCS parameters                     = ,,,,,,LOCALDATE="YYMMDD"
 Comment                            =
 DCS directory release level        = 0x0100
```

시스템 데이터베이스 디렉토리

시스템 데이터베이스 디렉토리에 다음 정보를 지정할 수 있습니다.

데이터베이스 이름

DCS 디렉토리 매개변수 테이블에 썼던 값과 같습니다.

데이터베이스 별명

호스트나 AS/400 데이터베이스 서버에 대한 별명. 이 이름은 데이터베이스를 액세스하는 다른 응용프로그램에서 사용될 수 있습니다. 기본 값은 데이터베이스 이름으로 지정한 값이 사용됩니다.

형식: 1–8자의 1바이트 영숫자(숫자 기호(#), at 기호(@), 달러 기호($) 및 밑줄(_) 등). 이 형식은 밑줄 문자나 숫자로 시작할 수 없습니다.

노드 이름

노드 디렉토리 매개변수 테이블에 썼던 값과 같습니다.

인증 사용자 이름과 암호의 유효성이 확인되는 위치를 지정합니다. 유효한 옵션은 SERVER, SERVER_ENCRYPT, CLIENT, DCE, DCS 및 DCS_ENCRYPT입니다. 129 페이지의 『제10장 보안』에서 더 자세한 정보를 참조하십시오.

같은 데이터베이스용 다중 항목 정의

각 데이터베이스용으로, 세 개의 디렉토리(노드 디렉토리, DCS 디렉토리, 시스템 데이터베이스 디렉토리) 적어도 한 항목은 정의해야 합니다. 어떤 경우에는, 데이터베이스용으로 둘 이상의 항목을 정의하려는 경우도 있습니다.

예를 들어, 호스트나 AS/400 데이터베이스 서버로부터 이식된 응용프로그램에 대해 SQLCODE 맵핑을 작동 중지시키려고 하지만, 클라이언트/서버 환경용으로 개발된 응용프로그램에 대한 기본 값 맵핑은 그대로 사용할 수 있습니다. 이를 다음과 같이 실행합니다.

- 노드 디렉토리에서 한 항목을 정의합니다.
- DCS 디렉토리에서 다른 데이터베이스 이름으로 두 항목을 정의합니다. 한 항목에는 매개변수 문자열에 NOMAP을 지정하십시오.
- 시스템 데이터베이스 디렉토리에서 다른 데이터베이스 별명으로 두 항목을 정의하고, DCS 디렉토리에서 지정한 두 데이터베이스 이름으로 정의합니다.

두 별명은 모두 같은 데이터베이스를 액세스하며, 하나는 SQLCODE 맵핑이 있고 다른 하나는 SQLCODE 맵핑이 없습니다.

디렉토리 갱신

DB2 Connect 시스템에서 CATALOG 명령 또는 OS/2 및 Windows 32비트 운영 체제에서 CCA의 데이터베이스 추가 마법사를 사용할 수 있습니다. DB2 응용프로그램 개발 클라이언트가 있으면 항목을 카탈로그화하는 응용프로그램을 작성할 수도 있습니다. *Administrative API Reference* 및 *Command Reference*에서 API에 대한 자세한 내용을 참조하십시오.

주: 데이터베이스를 카탈로그화하려면 *sysadm* 또는 *sysctrl* 권한이 반드시 있어야 합니다.

명령행 처리기를 사용하여 디렉토리를 갱신하려면, 다음 단계를 수행하십시오.

1. 다음 명령 중 하나를 사용하여 노드 디렉토리를 갱신하십시오.
   - APPC 연결이 있는 노드의 경우:

     ```
     db2 CATALOG APPC NODE 노드 이름
       REMOTE 기호 목적지 이름 SECURITY 보안 유형
     ```

예를 들면, 다음과 같습니다.

```
db2 CATALOG APPC NODE DB2NODE REMOTE DB2CPIC SECURITY PROGRAM
```

- TCI/IP 연결이 있는 OS/390용 DB2 Universal Database 버전 5.1 또는 AS/400용 DB2 Universal Database 버전 4.2 데이터베이스의 경우:

```
db2 CATALOG TCPIP NODE 노드 이름
REMOTE 호스트 이름 또는 IP 주소
SERVER 서비스 이름 또는 포트번호
SECURITY 보안유형
```

예를 들면, 다음과 같습니다.

```
db2 CATALOG TCPIP NODE MVSIPNOD REMOTE MVSHOST SERVER DB2INSTC
```

TCI/IP 연결용 기본 DRDA 포트 번호는 446입니다.

2. 다음 명령을 사용하여 DCS 디렉토리를 갱신하십시오.

```
db2 CATALOG DCS DATABASE 데이터베이스 이름 AS 목표 데이터베이스 이름
[AR 응용프로그램 리퀘스터]
[PARMS "매개변수 문자열"]
```

예를 들면, 다음과 같습니다.

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3
```

OS/2, Windows NT 또는 Windows 2000의 경우:

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3 PARMS "NOMAP,D"
```

또는 AIX의 경우:

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3 PARMS '"NOMAP,D"'
```

주: 매개변수 문자열에 LOCALDATE 마스크를 지정할 때 운영 체제 Escape 문자를 사용하는 것에 관해서는 98 페이지의 『매개변수 문자열 지정』에서 내용을 참조하십시오.

3. 다음 명령을 사용하여 시스템 데이터베이스 디렉토리를 갱신하십시오.

```
db2 CATALOG DATABASE 데이터베이스 이름
   AS 별명 AT NODE 노드 이름
AUTHENTICATION 인증유형
```

예를 들면, 다음과 같습니다.

```
db2 CATALOG DATABASE DB2DB AS NYC3 AT NODE DB2NODE AUTHENTICATION DCS
```

원격 클라이언트가 있으면, 각 원격 클라이언트마다 디렉토리를 또한 갱신해야 합니다. 해당 *DB2 Connect 빠른 시작* 책에서 자세한 내용을 참조하십시오.

제7장 응용프로그램 및 유틸리티 바인딩

내재적 SQL을 사용하여 개발된 응용프로그램은 해당 프로그램이 작동하는 데이터베이스마다 바인드되어야 합니다. 이런 기능이 사용가능한 플랫폼에서, 명령 센터와 클라이언트 구성 지원 프로그램을 사용하여 이를 실행할 수 있습니다.

바인딩은 각 데이터베이스에서 응용프로그램마다 일단 실행되어야 합니다. 바인드 프로세스 중에, 데이터베이스 액세스 플랜은 각 SQL문에 대해 저장됩니다. 이 액세스 플랜은 응용프로그램 개발자에 의해 제공되며, *바인드 파일*에 들어 있고 바인드 파일은 사전 처리 컴파일중에 작성됩니다. 바인딩은 호스트나 AS/400 데이터베이스 서버에서 이들 바인드 파일을 처리하는 프로세스입니다. 바인딩에 대해서는 *응용프로그램 개발 안내서*에서 자세한 내용을 참조하십시오

DB2 Connect에 제공되는 여러 유틸리티는 내재적 SQL을 사용하여 개발되었기 때문에, 이 유틸리티를 해당 시스템에서 사용하려면, 먼저 호스트나 AS/400 데이터베이스 서버로 바인드해야 합니다 107 페이지의 표4에 나열된 DB2 Connect 유틸리티와 인터페이스를 사용하지 않는 경우에는, 각 호스트나 AS/400 데이터베이스 서버에 이를 바인드할 필요가 없습니다. 이 유틸리티에 필요한 바인드 파일의 목록에는 다음 파일이 있습니다.

ddcsmvs.lst
: MVS 또는 OS/390용

ddcsvse.lst
: VSE용

ddcsvm.lst
: VM용

ddcs400.lst
: OS/400용

데이터베이스에 이 파일 목록 중 하나를 바인딩하면 해당 데이터베이스에 개별적인 각 유틸리티를 바인드합니다.

© Copyright IBM Corp. 1993, 2000

DB2 Connect Enterprise Edition이 설치된 경우, DB2 Connect 유틸리티는 각 호스트나 AS/400 데이터베이스 서버에 바인드되어야 하는데, 해당 시스템에서 사용할 수 있으려면 각 클라이언트 플랫폼 유형에 한번씩 바인드해야 합니다

예를 들어, Window NT용 DB2 Connect Enterprise Edition 서버를 통하여 OS/390용 DB2 Universal Database와 연결한 10개의 OS/2 클라이언트, 10개의 Windows 클라이언트, 10개의 AIX 클라이언트가 있는 경우, 다음 단계를 수행하십시오.

1. Windows 클라이언트로부터 ddcsmvs.lst를 바인드하십시오.
2. OS/2 클라이언트로부터 ddcsmvs.lst를 바인드하십시오.
3. AIX 클라이언트로부터 ddcsmvs.lst를 바인드하십시오.
4. DB2 Connect 서버로부터 ddcsmvs.lst를 바인드하십시오.

주: 이 내용에서는 모든 클라이언트가 동일한 서비스 레벨에 있다고 가정합니다. 그렇지 않으면, 추가로 특정한 서비스 레벨의 각 클라이언트로부터 바인드해야 할 수도 있습니다. DB2 버전 2.1 이전의 클라이언트가 있는 경우에는 241 페이지의 『부록E. 백 레벨 클라이언트용 유틸리티 바인딩』에서 자세한 내용을 참조하십시오.

DB2 Connect 유틸리티 이외에, Eebedded SQL을 사용하는 다른 응용프로그램도 또한 같이 작업하려는 데이터베이스마다 바인드되어야 합니다 바인드되지 않은 응용프로그램을 실행하면 자주 SQL0805N 오류 메시지가 나타납니다. 바인드해야 할 모든 응용프로그램에 추가 바인드 목록 파일을 작성할 수 있습니다.

바인드하려는 각 호스트나 AS/400 데이터베이스 서버에 대해 다음을 수행하십시오.

1. 호스트나 AS/400 데이터베이스 서버 관리 시스템에 대해 충분한 권한이 있어야 합니다.

   MVS 또는 OS/390

   요구되는 권한 부여는 다음과 같습니다.

   - SYSADM 또는
   - SYSCTRL 또는

- BINDADD *와* CREATE IN COLLECTION NULLID

주: BINDADD 및 CREATE IN COLLECTION NULLID 특권은 패키지가 아직 없는 경우에**만** 충분한 권한을 제공합니다. 예를 들어, 처음으로 패키지를 작성하고 있는 경우가 해당됩니다.

패키지가 이미 있고 다시 패키지를 바인드하는 중이면, 타스크 완료에 필요한 권한은 원래 바인드를 했던 사람에게 있습니다.

A 사용자가 원래 바인드를 했고 다시 바인드를 하는 중이라면, 위에 나열된 권한으로 바인드를 완료할 수 있습니다.

B 최초의 바인드가 다른 사람에 의해 실행되었고 지금 두 번째 바인드를 수행하려는 경우, 바인드를 완료하려면 SYSADM 또는 SYSCTRL 권한이 필요합니다. BINDADD와 CREATE IN COLLECTION NULLID 권한만으로는 바인드를 완료할 수 없습니다. SYSADM이나 SYSCTRL 특권이 없는 경우라도 패키지를 작성할 수 있습니다. 이런 상황에서는 바꾸려는 기존의 패키지마다 적절한 BIND 특권이 필요할 것입니다.

VSE 또는 VM

필요한 권한 부여는 DBA 권한입니다. 바인드 명령에서 GRANT 옵션을 사용하려면(개별적으로 DB2 Connect 패키지마다 액세스 권한을 부여하지 않도록), NULLID 사용자 ID에 다음 테이블에 대해 다른 사용자에게 권한을 부여하는 권한이 있어야 합니다.

- system.syscatalog
- system.syscolumns
- system.sysindexes
- system.systabauth
- system.syskeycols
- system.syssynonyms
- system.syskeys
- system.syscolauth

VSE 또는 VM 시스템에서 다음을 실행할 수 있습니다.

```
grant select on table to nullid with grant option
```

OS/400

NULLID 콜렉션 증 CHANGE 권한 또는 그 이상

2. 다음과 같은 명령을 실행하십시오.

```
db2 connect to DBALIAS user USERID using PASSWORD
db2 bind path@ddcsmvs.lst blocking all
      sqlerror continue messages ddcsmvs.msg grant public
db2 connect reset
```

여기서 *DBALIAS*, *USERID* 및 *PASSWORD*는 호스트나 AS/400 데이터베이스 서버에 적용되고 ddcsmvs.lst는 MVS에 대한 바인드 목록 파일의 이름이며 *path*는 바인드 목록 파일의 위치입니다.

예를 들어, *drive*:\sqllib\bnd\는 모든 Intel 운영 체제에 적용되고, *INSTHOME*/sqllib/bnd/는 모든 UNIX 운영 체제에 적용됩니다. 여기서 *drive*는 DB2 Connect가 설치된 논리 드라이브이고 *INSTHOME*은 DB2 Connect 인스턴스의 홈 디렉토리입니다.

bind 명령의 권한 부여 옵션을 사용하면 EXECUTE 특권을 PUBLIC으로 또는 지정된 사용자 이름이나 그룹 ID에 권한 부여할 수 있습니다. **bind** 명령의 권한 부여 옵션을 사용하지 않으면, 개별적으로 GRANT EXECUTE(RUN) 해야 합니다.

바인드 파일용 패키지 이름을 찾으려면, 다음 명령을 입력하십시오.

```
ddcspkgn @bindfile.lst
```

예를 들면, 다음과 같습니다.

```
ddcspkgn @ddcsmvs.lst
```

위 명령은 다음으로 출력됩니다.

```
바인드 파일                    패키지 이름
------------------------------ ------------------------------
f:\sqllib\bnd\db2ajgrt.bnd     SQLAB6D3
```

참고로, 표4는 DB2 Connect의 여러 구성요소에 의해 사용된 바인드 파일과 패키지 이름을 나타냅니다. 어떤 경우에는 다른 바인드 파일과 패키지가 다른 운영 체제에서 사용됩니다.

표 4. 바인드 파일과 패키지

구성요소	바인드 파일	패키지	MVS 또는 OS/390	VSE	VM	OS/400
바인더(GRANT 바인드 옵션에서 사용)	db2ajgrt.bnd	sqlab*xxx*	예	예	예	예
DB2 콜 레벨 인터페이스						
분리 레벨 CS	db2clics.bnd	sqll1*xxx*	예	예	예	예
분리 레벨 RR	db2clirr.bnd	sqll2*xxx*	예	예	예	예
분리 레벨 UR	db2cliur.bnd	sqll3*xxx*	예	예	예	예
분리 레벨 RS	db2clirs.bnd	sqll4*xxx*	예	예	예	예
분리 레벨 NC	db2clinc.bnd	sqll5*xxx*	아니오	아니오	아니오	예
MVS 테이블 이름 사용	db2clims.bnd	sqll7*xxx*	예	아니오	아니오	아니오
OS/400 테이블 이름 사용(OS/400 3.1 또는 그 이후)	db2clias.bnd	sqlla*xxx*	아니오	아니오	아니오	예
VSE/VM 테이블 이름 사용	db2clivm.bnd	sqll8*xxx*	아니오	예	예	아니오
명령행 처리기						
분리 레벨 CS	db2clpcs.bnd	sqlc2*xxx*	예	예	예	예
분리 레벨 RR	db2clprr.bnd	sqlc3*xxx*	예	예	예	예
분리 레벨 UR	db2clpur.bnd	sqlc4*xxx*	예	예	예	예
분리 레벨 RS	db2clprs.bnd	sqlc5*xxx*	예	예	예	예
분리 레벨 NC	db2clpnc.bnd	sqlc6*xxx*	아니오	아니오	아니오	예
REXX						
분리 레벨 CS	db2arxcs.bnd	sqla1*xxx*	예	예	예	예
분리 레벨 RR	db2arxrr.bnd	sqla2*xxx*	예	예	예	예
분리 레벨 UR	db2arxur.bnd	sqla3*xxx*	예	예	예	예
분리 레벨 RS	db2arxrs.bnd	sqla4*xxx*	예	예	예	예
분리 레벨 NC	db2arxnc.bnd	sqla5*xxx*	아니오	아니오	아니오	예
유틸리티						
내보내기	db2uexpm.bnd	sqlub*xxx*	예	예	예	예
가져오기	db2uimpm.bnd	sqluf*xxx*	예	예	예	예

DB2 Connect의 경우 이 값을 알아내려면, *ddcspkgn* 유틸리티를 실행하십시오. 예를 들면, 다음과 같습니다.

```
ddcspkgn @ddcsmvs.lst
```

선택적으로, 이 유틸리티를 사용하면 개별적인 바인드 파일의 패키지 이름을 알 수 있으며, 그 예는 다음과 같습니다.

```
ddcspkgn bindfile.bnd
```

MVS/ESA용 DB2 시스템에 APAR PN60988용 수정사항이 설치된 경우(또는 버전 3 릴리스 1 이후의 릴리스인 경우), 분리 레벨 NC용 바인드 파일을 ddcsmvs.lst 파일에 추가할 수도 있습니다.

*Command Reference*에서 바인드 옵션에 대한 자세한 내용을 참조하십시오.

주:

a. 바인드 옵션 `sqlerror continue`를 반드시 사용해야 하는데, 이 옵션은 사용자가 DB2 도구 또는 명령행 처리기를 사용하여 응용프로그램을 바인드할 때 자동으로 지정됩니다. 이 옵션을 지정하면 바인드 오류가 경고로 바뀌어 오류가 있는 파일의 바인딩이 여전히 패키지를 생성할 수 있습니다. 따라서, 이 옵션의 지정으로 하나의 바인드 파일이 여러 서버에 대해 사용될 수 있으며, 비록 특정한 서버 구현이 SQL 구문이 유효하지 않다고 플래그할 경우에도 마찬가지입니다. 이런 이유로, 다른 특정 호스트나 AS/400 데이터베이스 서버에 대해 목록 파일 `ddcsxxx.lst`를 바인딩하면 경고가 발생합니다. 예를 들어, VM용 DB2에 대한 바인딩시, VM용 DB2는 커서가 `"WITH HOLD"`로 나타나지 않으면 많은 경고 메시지가 발생합니다.

b. DB2 Connect를 통해 DB2 Universal Database 데이터베이스에 연결하는 경우, 바인드 목록 `db2ubind.lst`를 사용하고 DRDA 서버와 연결할 때만 유효한 `sqlerror continue`를 지정하지 마십시오. 또한 DB2 Universal Database 데이터베이스에 연결하려면, DB2에 제공되고 DB2 Connect에는 제공되지 않은 DB2 클라이언트를 사용하도록 하십시오.

3. 유사한 명령문을 사용하여 각 응용프로그램이나 응용프로그램 목록을 바인드하십시오.

4. 이전 버전의 DB2 원격 클라이언트가 있으면, 해당 클라이언트의 유틸리티를 DB2 Connect로 바인드해야 할 경우가 있습니다. 241 페이지의 『부록E. 백레벨 클라이언트용 유틸리티 바인딩』에서 자세한 내용을 참조하십시오.

BIND 명령

DB2 **바인드** 명령은 응용프로그램을 지정한 데이터베이스로 바인드합니다. 독립된 조작으로 사전 처리 컴파일하고 바인드하는 경우, 바인드에서 지정한 옵션은 사전에 컴파일 단계에서 지정한 옵션을 겹쳐씁니다.

*Command Reference*에 DB2 Connect를 통하여 응용프로그램을 호스트나 AS/400 데이터베이스 서버에 바인딩하는 경우 사용해야 하는 BIND 명령 구문에 대한 설명이 있습니다. DRDA 지정 설명을 참조하고 있는지 확인하십시오.

주: BIND 명령의 일부 매개변수는 호스트나 AS/400 데이터베이스 서버에서 지원하지 않습니다. 호스트나 AS/400 데이터베이스 서버 RDBMS에 제공된 문서에서 자세한 내용을 참조하십시오.

리바인딩

응용프로그램을 바인드하고 나면(그리고, 호스트나 AS/400 데이터베이스 서버에서 패키지를 작성하고 나면), 그 패키지를 재작성해야 합니다. 원본 바인드 파일이 없어도 명령행 처리기 **REBIND PACKAGE** 명령이나 해당하는 API를 사용하여 이를 작성할 수 있습니다.

이 명령을 사용하면 다음과 같은 이점이 있습니다.

- 원본 바인드 파일 없이 새로운 패키지 섹션을 다시 최적화하고 구축하여 시스템 변경사항을 이용할 수 있습니다.
- 작동 불능의 유효하지 않은 패키지를 재작성할 수 있습니다.
- 이주를 통하여 무효한 패키지를 재작성할 수 있습니다.
- 내재된 리바인드나 바인드를 사용하기 보다는 명시된 리바인드를 사용하여 성능을 향상시킬 수 있습니다.

- 사용자가 특성을 변경할 수 있습니다. 예를 들어, OS/390용 DB2 Universal Database에서 규정되지 않은 테이블의 규정자를 테스트 목적이나 이주 목적용으로 변경할 수 있습니다.

프로그램, 바인드 옵션 또는 소유자 정보를 변경하려면, **BIND** 명령을 사용해야 합니다. 그리고 패키지가 데이터베이스에 없는 경우나 모든 바인드 오류(첫 번째 검출된 오류 포함)를 보려는 경우에는, **BIND** 명령을 사용해야 합니다.

이 명령을 수행하려면, 호스트나 AS/400 데이터베이스 서버에서 요구하는 권한 레벨이 있어야 합니다. 사용자가 데이터베이스에 연결되지 않은 경우, 명령은 내재된 연결로써 기본 데이터베이스로(연결 특권이 있으면) 연결됩니다.

명령행 처리기 명령 구문은 *Command Reference*에 설명되어 있습니다.

바인드 파일용 패키지 이름을 찾으려면, 다음 명령을 입력하십시오. **ddcspkgn** *bindfile***.bnd**

제8장 데이터베이스 시스템 모니터

이 장에서는 DB2 Connect 사용자를 위해 DB2 시스템 모니터의 기능을 요약하여 설명합니다. 아래의 시스템 모니터 향상이 DB2 Connect 버전 7에서 제공됩니다.

- 스냅샷 모니터링. 시스템의 스냅샷은 특정 시점에 대한 정보를 제공합니다. 스냅샷은 특정 오브젝트 또는 오브젝트 그룹에 대한 데이터베이스 관리 프로그램의 현재 활동 상태를 그림으로 표현한 것입니다. 모니터를 통해 사용할 수 있는 5가지의 DCS 데이터베이스 정보 기본 스냅샷이 있습니다.
- LIST DCS APPLICATIONS CLP 명령과 동등한 그래픽 사용자 인터페이스. LIST 명령은 스냅샷보다 시스템 상태에 대해 보다 정교한 읽을 거리를 제공합니다. 그래픽 LIST 기능은 DB2 명령 센터를 통해 제공됩니다. 119 페이지의 『LIST DCS APPLICATIONS EXTENDED』에서 자세한 내용을 참조하십시오.

*시스템 모니터 안내 및 참조서*에서 데이터베이스 시스템 모니터에 대한 자세한 내용을 참조하십시오.

원격 클라이언트의 연결 모니터링

DB2 Connect Enterprise Edition에 데이터베이스 시스템 모니터를 사용하여 원격 클라이언트 연결을 모니터할 수 있습니다. DB2 Connect 서버에 지역적인 즉, 서버상에서 수행 중인 클라이언트들을 모니터하려면 다음의 환경 변수를 설정해야 합니다.

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

예를 들어, 호스트나 AS/400 시스템에서 오류가 발생하면 시스템 관리자는 DB2 Connect 워크스테이션에서 발생한 문제인지 알아낼 수 있습니다. 데이터베이스 시스템 모니터는 다음과 관련이 있습니다.

- 보호되지 않은 대화용 DRDA 상관 토큰(CRRTKN)

© Copyright IBM Corp. 1993, 2000

- SNA 동기점 관리 프로그램(SPM)에 의해 보호되는 2단계 대화용 논리적 작업 단위(UOW) 식별자(LUWID)
- DRDA-3 동기점 관리 프로그램(TCP/IP 연결에 사용된)에 의해 보호되는 2단계 연결용 작업 단위(UOW) ID(UOWID)
- DB2 Connect 연결 식별자(응용프로그램 ID)

이 정보는 문제를 일으킨 DB2 Connect 연결을 보여 주므로, 시스템 관리자가 DB2 Connect 연결을 사용한 다른 클라이언트에 영향을 주지 않으면서 시스템으로부터 개별적인 클라이언트 응용프로그램을 강제 중단할 수 있게 합니다.

DB2 Connect의 모니터 스위치 작동

시스템 모니터는 항상 작동합니다. 그러나 GET SNAPSHOT 출력으로 더 많은 세부사항을 가져오려면 해당 모니터 스위치를 작동시켜야 합니다. DB2 Connect에 관련된 모니터 스위치는 STATEMENT(명령문 레벨 정보의 경우)와 UOW(트랜잭션 레벨 정보의 경우)입니다.

모니터 스위치를 변경하려면, **db2 update monitor switches** 명령을 사용하십시오. 이 명령 구문에 대해서는 *Command Reference*에서 참조하십시오. 다음은 작업 단위(UOW)에 대해 DB2 시스템 모니터 통계를 작성하는 예입니다.

```
db2 update monitor switches using uow on
```

모니터 스위치의 상태 나열

모니터 스위치 상태를 나열하려면, **db2 get monitor switches** 명령을 사용하십시오.

GET SNAPSHOT 명령 사용

DB2 모니터에서는 가치있는 시스템 정보의 수행 중 부합물을 유지보수합니다. GET SNAPSHOT 명령을 발행하여 아무 때나 시스템 상태의 요약을 가져올 수 있습니다. 모니터하려는 데이터베이스 관리 프로그램 인스턴스에 대해 SYSMAINT, SYSCTRL 또는 SYSADM 권한을 가지고 있는 경우에는 모니터 스냅샷을 취할 수 있습니다.

DCS 정보를 모니터하는 데 다음의 5개 스냅샷 명령을 사용할 수 있습니다.

- GET SNAPSHOT FOR ALL DCS DATABASES
- GET SNAPSHOT FOR ALL DCS APPLICATIONS
- GET SNAPSHOT FOR DCS APPLICATION ...
- GET SNAPSHOT FOR DCS DATABASE ON db_alias
- GET SNAPSHOT FOR DCS APPLICATIONS ON db_alias

각 스냅샷 명령은 사용자가 요청한 영역에 관해 상세한 보고서를 작성합니다.

인스턴스의 경우에 GET SNAPSHOT FOR DCS DATABASE ON DCSDB를 발행하면 다음과 같은 보고서가 작성됩니다.

```
                DCS Database Snapshot

  DCS database name                          = DCSDB
  Host database name                         = GILROY
First database connect timestamp           = 12-15-1999 10:28:24.596495
Most recent elapsed time to connect        = 0.950561
Most recent elapsed connection duration    = 0.000000
Host response time (sec.ms)                = 0.000000
Last reset timestamp                       =
Number of SQL statements attempted         = 2
Commit statements attempted                = 1
Rollback statements attempted              = 0
Failed statement operations                = 0
Total number of gateway connections        = 1
Current number of gateway connections      = 1
Gateway conn. waiting for host reply       = 0
Gateway conn. waiting for client request   = 1
Gateway communication errors to host       = 0
Timestamp of last communication error      = None
High water mark for gateway connections    = 1
Rows selected                              = 0
Outbound bytes sent                        = 140
Outbound bytes received                    = 103
```

이 보고서에서는 데이터베이스 연결, 성능, SQL 요청의 오류 및 처리량에 관한 정보를 제공합니다. 실제로는 DB2 모니터 스냅샷이 이 보다 훨씬 상세합니다. 인스턴스의 경우에 GET SNAPSHOT FOR ALL DCS APPLICATIONS 명령을 발행하면 다음과 유사한 보고서를 볼 수 있습니다.

```
               DCS Application Snapshot
```

```
Client application ID                        = 09150F74.B6A4.991215152824
  Sequence number                            = 0001
  Authorization ID                           = SMITH
  Application name                           = db2bp
  Application handle                         = 1
   Application status                      = waiting for request
  Status change time                         = 12-15-1999 10:29:06.707086
  Client node                                = sys143
  Client release level                       = SQL06010
   Client platform                         = AIX
   Client protocol                         = TCP/IP
  Client codepage                            = 850
  Process ID of client application           = 49074
  Client login ID                            = smith
  Host application ID                        = G9150F74.B6A5.991215152825
  Sequence number                            = 0000
  Database alias at the gateway              = MVSDB
  DCS database name                          = DCSDB
  Host database name                         = GILROY
  Host release level                         = DSN05012
   Host CCSID                              = 500

Outbound communication address               = 9.21.21.92 5021
Outbound communication protocol              = TCP/IP
Inbound communication address                = 9.21.15.116 46756
First database connect timestamp             = 12-15-1999 10:28:24.596495
Host response time (sec.ms)                  = 0.000000
Time spent on gateway processing             = 0.000000
Last reset timestamp                         =
Rows selected                                = 0
Number of SQL statements attempted           = 2
Failed statement operations                  = 0
Commit statements                            = 1
Rollback statements                          = 0
Inbound bytes received                       = 404
Outbound bytes sent                          = 140
Outbound bytes received                      = 103
Inbound bytes sent                           = 287
Number of open cursors                       = 0
Application idle time                        = 1 minute and 32 seconds

UOW completion status                        =
Previous UOW completion timestamp            = 12-15-1999 10:28:25.592631
UOW start timestamp                          = 12-15-1999 10:29:06.142790
UOW stop timestamp                           =
Elapsed time of last completed uow (sec.ms)= 0.034396

Most recent operation                        = Execute Immediate
```

```
Most recent operation start timestamp      = 12-15-1999 10:29:06.142790
Most recent operation stop timestamp       = 12-15-1999 10:29:06.707053

Statement                                  = Execute Immediate
Section number                             = 203
Application creator                        = NULLID
Package name                               = SQLC2C07
SQL compiler cost estimate in timerons     = 0
SQL compiler cardinality estimate          = 0
Statement start timestamp                  = 12-15-1999 10:29:06.142790
Statement stop timestamp                   = 12-15-1999 10:29:06.707053
Host response time (sec.ms)                = 1.101612
Elapsed time of last completed stmt(sec.ms)= 0.564263
Rows fetched                               = 0
Time spent on gateway processing           = 0.013367
Inbound bytes received for statement       = 220
Outbound bytes sent for statement          = 130
Outbound bytes received for statement      = 49
Inbound bytes sent for statement           = 27
SQL statement text:
create table t12 (col1 int, col2 char)
```

*시스템 모니터 안내 및 참조서*에서 GET SNAPSHOT 명령 및 다른 유용한 DB2 모니터 명령에 대한 자세한 내용을 참조하십시오.

DCS 응용프로그램 상태 목록

DB2 Connect V5.2부터는 시스템 모니터에서 다음과 같이 세 가지 양식의 LIST DCS APPLICATIONS 명령을 제공합니다.

- LIST DCS APPLICATIONS
- LIST DCS APPLICATIONS SHOW DETAIL
- LIST DCS APPLICATIONS EXTENDED.

LIST DCS APPLICATIONS

응용프로그램 레벨에서 모니터가 제공하는 정보를 보려면, DB2 LIST DCS APPLICATIONS 명령을 실행하십시오. 이 명령은 APPC 연결(DB2 Connect Enterprise Edition 버전 7에서 OS/390용 DB2 Universal Database로의 연결)에 대해 다음 정보를 리턴합니다.

```
Auth Id  Application Name     Appl.      Host Application ID
                              Handle
-------- -------------------- ---------- --------------------------------
USERID   db2bp_41             0          CAIBMOML.OMXT4H0A.A79EAA3C6E29
```

이는 TCP/IP 연결(DB2 Connect Enterprise Edition 버전 7에서 OS/390용 DB2 Universal Database로의 연결)에 대해 다음 정보를 리턴합니다.

```
Auth Id  Application Name     Appl.      Host Application ID
                              Handle
-------- -------------------- ---------- --------------------------------
USERID   db2bp_41             2          0915155C.9704.1517172201BE
```

Auth.Id

호스트 또는 AS/400 데이터베이스 서버에 로그온하는 데 사용된 권한 부여 ID. 이는 응용프로그램을 수행 중인 사용자를 식별합니다.

응용프로그램 이름

DB2 Connect에 알려진대로 클라이언트에서 수행 중인 응용프로그램의 이름. 최종 경로 분리자 다음의 첫째 20 바이트만이 사용 가능합니다. 응용프로그램 이름은 OS/2용 DB2 버전 1에서 수행되는 응용프로그램에는 사용할 수 없습니다.

Appl.Handle

DB2 Connect 워크스테이션에서 실행 중인 에이전트. 이 요소를 사용하여 데이터베이스 시스템 모니터 정보를 다른 진단 정보와 연결할 수 있습니다.(예는, 190 페이지의 『추적 유틸리티(ddcstrc)』에서 자세한 내용을 참조하십시오.) 에이전트 ID는 또한 FORCE USERS 명령이나 API를 사용할 경우에 필요합니다.

Host Application ID

다음 중 하나입니다.

- 보호되지 않은 대화용 DRDA 상관 토큰(CRRTKN)
- SNA 동기점 관리 프로그램(SPM)에 의해 보호되는 2단계 대화용 논리적 작업 단위(UOW) 식별자(LUWID)
- DRDA-3 동기점 관리 프로그램(TCP/IP 연결에 사용된)에 의해 보호되는 2단계 연결용 작업 단위(UOW) ID(UOWID)

이 고유 식별자는 응용프로그램이 호스트나 AS/400 데이터베이스 서버에 연결할 때 생성됩니다. 응용프로그램 ID와 함께 이 요소를 사용하여 응용프로그램 정보의 클라이언트와 서버 부분을 연관시킬 수 있습니다.

LIST DCS APPLICATIONS SHOW DETAIL

DB2 LIST DCS APPLICATIONS SHOW DETAIL 명령 형식이 지정된 경우, 다음과 같은 추가 정보가 표시됩니다.

```
Auth Id  Application Name     Appl.      Client Application Id         Seq# Client
                              Handle                                       DB Alias
-------- -------------------- ---------- ------------------------------ ---- --------
NEWTON   db2bp                0          09151251.07D3.980925183850    0001 MVSDB

 Client   Client   Client     Host Application Id              Seq# Host DB Name
 Node     Release  Codepage
 -------- -------- ---------- -------------------------------- ---- ----------------
 antman   SQL05020 819        G9151251.G7D4.980925183851       0000 GILROY

Host
Release
--------
DSN05011
```

이 보고서는 형식화되지 않은 것이므로, 사용자에게는 119 페이지의 『LIST DCS APPLICATIONS EXTENDED』 보고서가 더 유용할 것입니다.

클라이언트 응용프로그램 ID

DB2 Connect 워크스테이션으로 연결된 응용프로그램을 고유하게 식별합니다. 응용프로그램 ID의 형식은 클라이언트와 DB2 Connect 워크스테이션간의 통신 프로토콜에 따라 다릅니다. *관리 안내서*에서 형식에 대한 자세한 내용을 참조하십시오.

이 값은 클라이언트로부터 DB2 Connect 워크스테이션, DB2 Connect 워크스테이션에서 호스트나 AS/400 데이터베이스 서버로의 연결을 서로 연관시켜줍니다.

클라이언트 순서 번호(순번)

클라이언트 순서 번호는 트랜잭션 순서 번호입니다. 이 번호를 사용하여 다른 시스템상의 트랜잭션 스프레드를 연관하도록 도움을 줍니다.

클라이언트 DB 별명

데이터베이스에 연결하기 위해 응용프로그램에 의해 제공된 데이터베이스 별명입니다. 이 요소를 사용하면 데이터베이스가 액세스하는 실제 데이터

베이스를 식별할 수 있습니다. 이 이름과 데이터베이스 이름간의 맵핑은 클라이언트 노드 및 데이터베이스 관리 프로그램 서버 노드의 데이터베이스 디렉토리를 사용하여 이루어질 수 있었습니다.

클라이언트 NNAME(노드)

클라이언트 응용프로그램이 수행중인 노드를 식별합니다. 정보는 사용 중인 클라이언트 프로토콜에 따라 달라집니다. 예를 들어, NetBIOS를 통해 연결된 클라이언트에서는, 이 값은 NNAME 데이터베이스 관리 프로그램 구성 매개변수 값입니다. TCP/IP를 통해 연결된 클라이언트에서는, 호스트 이름입니다.

클라이언트 제품 ID(클라이언트)

클라이언트에서 수행중인 제품 및 버전입니다. 클라이언트 제품 ID는 다음과 같습니다.

- OS/2용 DB2의 버전 1 경우 SQL01010.
- UNIX 기반 DB2 제품 및 CAE의 버전 1 경우 SQL01011.
- DB2 제품 및 CAE의 버전 2 경우 SQL0210.
- DB2 제품 및 CAE의 버전 2.1.2 경우 SQL02020.
- DB2 Universal Database와 DB2 Connect 제품 및 해당 클라이언트의 버전 5.0 경우 SQL05000.
- DB2 Universal Database와 DB2 Connect 제품 및 해당 클라이언트의 버전 5.2 경우 SQL05020.
- DB2 Universal Database와 DB2 Connect 제품 및 해당 클라이언트의 버전 6.1 경우 SQL06010.
- DB2 Universal Database와 DB2 Connect 제품 및 해당 클라이언트의 버전 7 경우 SQL07010.

코드 페이지 ID

모니터된 응용프로그램이 시작한 노드의 코드 페이지 식별자입니다.

이 정보를 사용하여 데이터 변환이 응용프로그램 코드 페이지와 데이터베이스 코드 페이지(또는 호스트나 AS/400 데이터베이스 서버 데이터베이스, 호스트나 AS/400 데이터베이스 서버 CCSID) 사이에서 지원되도록 할 수 있습니다.

응용프로그램 코드 페이지가 데이터베이스 시스템 모니터가 수행중인 코드 페이지와 다른 경우, 이 코드 페이지 요소는 응용프로그램으로부터 전달되고 데이터베이스 시스템 모니터로 표시장치된 데이터를 직접 변환하는 데 도움을 줄 수 있습니다. 예를 들어, 이 요소를 사용하여 응용프로그램 이름을 해석할 수 있습니다.

아웃바운드 순서 번호

이 번호는 아웃바운드 순서 번호를 나타냅니다. 그리고 다른 시스템에서 트랜잭션의 상호 관련을 위해 사용됩니다.

호스트 데이터베이스 이름

응용프로그램이 연결되는 실제 데이터베이스 이름입니다. DCS 디렉토리에서, 이 이름은 *목표 데이터베이스 이름*입니다.

호스트 제품 ID

서버에서 수행중인 제품 및 버전입니다. 이것은 다음과 같이 *PPPVVRRM* 형식으로 되어 있습니다.

PPP 호스트나 AS/400 데이터베이스 서버 제품(예: OS/390용 DB2 Universal Database의 경우 DSN, VSE & VM용 DB2의 경우 ARI, AS/400용 DB2 Universal Database의 경우 QSQ)을 식별합니다.

VV 01과 같은 두자리 버전 번호입니다.

RR 두자리 릴리스 번호입니다.

M 한자리 수정 레벨입니다.

LIST DCS APPLICATIONS EXTENDED

EXTENDED 옵션에서 LIST DCS APPLICATIONS 명령을 사용하여 확장 보고서를 생성할 수 있습니다. 확장 보고서에는 명령에 SHOW DETAIL 옵션이 지정된 경우 나열되는 모든 필드와 아래와 같은 9개의 새 필드가 나열됩니다.

- DCS 응용프로그램 상태
- 상태 변경 시간
- 클라이언트 플랫폼
- 클라이언트 프로토콜

- 호스트의 코드화된 문자 세트 식별자(CCSID)
- 클라이언트 로그인 ID
- 클라이언트 응용프로그램의 프로세스 ID
- 게이트웨이에서의 데이터베이스 별명
- DCS 데이터베이스 이름

기존 명령 옵션은 응용프로그램당 한 줄씩 필드를 수평으로 나열하는 반면, 새 옵션은 한 줄에 한 필드씩 수직으로 나열합니다.

다음은 새로운 명령 구문입니다.

```
LIST DCS APPLICATIONS [SHOW DETAIL | EXTENDED ]
```

그리고 다음은 새 옵션 EXTENDED를 사용할 때 이 명령의 샘플 출력입니다.

```
                List of DCS Applications - Extended Report

Client application ID                        = 09151251.0AA7.981015204853
  Sequence number                            = 0001
  Authorization ID                           = NEWTON
  Application name                           = db2bp
  Application handle                         = 1
   Application status                        = waiting for request
  Status change time                         = 10-15-1998 16:50:29.489160
   Client node                               = antman
   Client release level                      = SQL05020
   Client platform                           = AIX
   Client protocol                           = TCP/IP
  Client codepage                            = 819
  Process ID of client application           = 39324
  Client login ID                            = smith
  Host application ID                        = G9151251.GAA8.981015204854
  Sequence number                            = 0000
  Database alias at the gateway              = MVSDB
  DCS database name                          = DCSDB
  Host database name                         = GILROY
  Host release level                         = DSN05011
   Host CCSID                                = 500
```

응용프로그램 상태 필드에는 다음 세 값 중 하나가 들어 있습니다.

1. `connect pending - outbound.` 이것은 호스트 데이터베이스로의 연결 요청이 실행되었으며 DB2 Connect가 연결이 이루어지기를 기다리고 있음을 의미합니다.

2. `waiting for request.` 이것은 호스트 데이터베이스와의 연결이 이루어졌으며 DB2 Connect가 클라이언트 응용프로그램에서 SQL문을 보내기를 기다리고 있음을 의미합니다.
3. `waiting for reply.` 이것은 SQL문이 호스트 데이터베이스로 보내졌음을 의미합니다.

또한 상태 변경 시간은 시스템 모니터 UOW 스위치가 처리하는 동안 작동된 경우에만 보고서에 나타납니다. 그렇지 않으면, "Not Collected"가 나타납니다.

DB2 제어 센터를 사용하여 확장 DCS 응용프로그램 정보 나열

DB2 버전 7 제어 센터를 사용하여 DB2 Connect 게이트웨이 모니터링을 수행할 수 있습니다. 이 절에서는 제어 센터를 사용하여 **list dcs applications extended** 명령에서 제공하는 것과 동일한 보고서를 표시하는 방법에 대해 설명합니다.

응용프로그램에 대한 확장 보고서를 보려면 다음과 같이 하십시오.

1. 제어 센터의 **시스템** 아이콘 아래에서 트리를 확장하여 **시스템 —> 인스턴스 —> 게이트웨이 연결**을 표시하십시오. **게이트웨이 연결** 폴더에 있는 임의의 인스턴스에서 오른쪽 마우스 버튼으로 누르면, 팝업 메뉴가 나타납니다. 이 메뉴에서 응용프로그램... 항목을 선택하십시오. 응용프로그램 창이 나타납니다. 이 창은 탭이 붙은 공책 형태인데, 이중 **응용프로그램**이라는 제목의 탭이 있습니다. 사용자 인스턴스에 게이트웨이 응용프로그램이 있는 경우, 두 번째 탭에는 **게이트웨이 응용프로그램**이라는 제목이 표시됩니다.
2. 각 페이지의 기본 창에는 `LIST DCS APPLICATIONS EXTENDED` 보고서의 필드에 해당하는 정보 열이 들어 있습니다. 창에 표시되는 처음 6 컬럼에는 다음과 같은 데이터가 제공됩니다.

   ```
   클라이언트 노드
   응용프로그램 이름
   클라이언트 응용프로그램 ID
   호스트 응용프로그램 ID
   게이트웨이에서의 데이터베이스 별명
   상태
   ```

   보고서의 나머지 필드는 창 하단의 수평 화면 이동 막대를 이동하면 볼 수 있습니다.

LIST DCS APPLICATIONS EXTENDED 명령이 표시하는 모든 필드는 이 뷰에 표시됩니다.

Windows 성능 모니터 사용

Windows NT 및 Windows 2000에서는 DB2 응용프로그램의 성능을 모니터하는 데 유익한 도구를 제공합니다. Windows 관리 도구 중 하나인 성능 모니터는 시스템 성능을 그래픽으로 표시합니다. 모니터할 여러 가지 시스템, 데이터베이스 및 통신 관련 항목을 선택하여 이들을 그래픽 화면으로 함께 맵핑할 수 있습니다.

예를 들어, **GET SNAPSHOT FOR ALL DCS DATABASES** 또는 **GET SNAPSHOT FOR ALL DCS APPLICATIONS** 명령을 통해 사용 가능한 보고서들을 실시간으로 모니터를 사용하여 그래프 처리하고 CPU 사용과 같은 값과 직접적으로 비교할 수 있습니다. 데이터베이스 또는 통신 성능에 대한 여러 설정값의 효과를 직접적으로 비교할 수 있습니다. PMC 파일에 설정값의 특수화된 구성을 저장할 수 있으며 이 파일은 나중에 검색할 수 있습니다.

예를 들어, 아래의 그림에는 CPU 사용에 대한 여러 개의 DB2 측정량이 그래프로 표시됩니다. 도형화된 값의 콜렉션은 파일 db2chart.pmc에 저장되어 있습니다. 각각 시스템 성능의 서로 다른 단면을 반영하는 PMC 파일들을 원하는 수만큼 저장할 수도 있습니다.

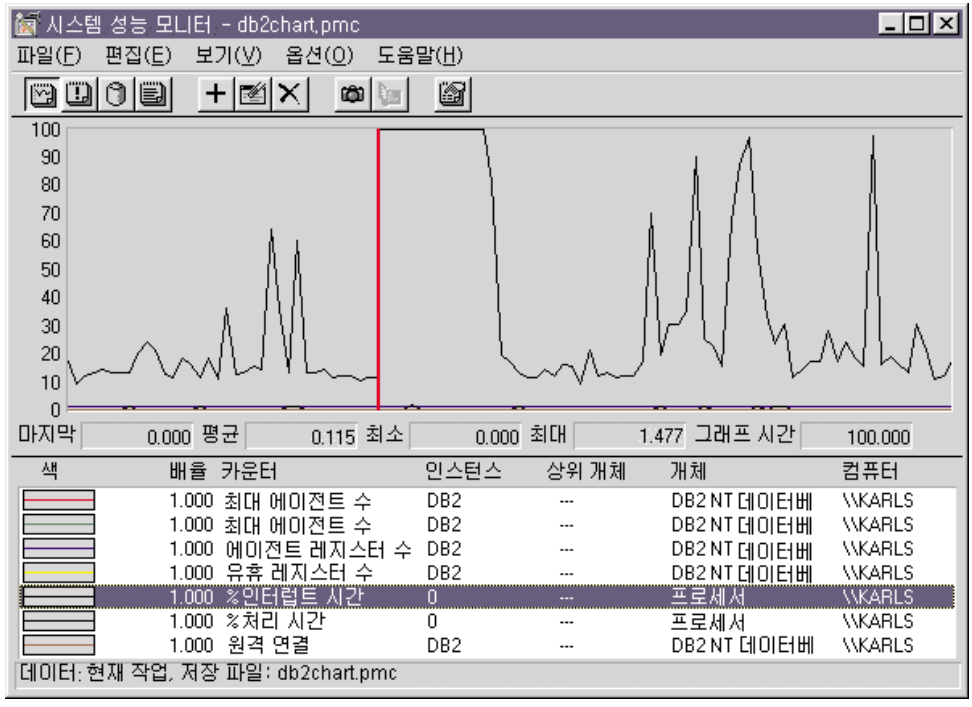


그림 5. 성능 모니터

지역 응용프로그램의 모니터링을 사용하려면 DB2CONNECT_IN_APP_PROCESS 환경 변수를 OFF시켜야 합니다.

*관리 안내서*에서, DCS APPLICATIONS 명령에 대한 내용이나 DB2 UDB 및 DB2 Connect에서의 Windows 성능 모니터 사용에 대해 자세한 내용을 참조하십시오.

제9장 관리 유틸리티

이 장에서는 관리 타스크 수행을 돕는 유틸리티를 설명합니다. 다음 절이 들어 있습니다.

- 『명령행 처리기』
- 126 페이지의 『가져오기 및 내보내기 유틸리티 사용』

명령행 처리기

명령행 처리기는 호스트나 AS/400 데이터베이스 서버에 대해 db2가 선행하는 SQL 문을 발행할 수 있도록 해줍니다 호스트와 AS/400 SQL 및 DB2 Connect SQL 사이의 차이에 대해서는 64 페이지의 『DB2 Connect에서 지원하는 호스트나 AS/400 서버 SQL문』과 64 페이지의 『DB2 Connect에서 거부하는 호스트나 AS/400 서버 SQL문』에서 자세한 내용을 참조하십시오.

대화식 입력 모드에서 명령행 처리기를 호출하려면 다음을 수행하십시오.

OS/2 **OS/2 Warp**를 클릭한 후 **IBM DB2 —> 명령행 처리기**를 선택하거나 **db2** 명령을 입력합니다.

Windows 32비트 운영 체제
시작을 클릭한 후 **프로그램 —> IBM DB2 —> 명령행 처리기**를 선택합니다.

주: MS-DOS 프롬프트에서 **db2cmd** 명령 다음에 **db2** 명령을 입력하여 명령행 처리기를 대화식 입력 모드로 호출할 수도 있습니다.

UNIX 명령행 처리기에서 **db2** 명령을 입력합니다.

대화식 입력 모드에서 프롬프트는 다음과 같습니다.

```
db2 =>
```

© Copyright IBM Corp. 1993, 2000

대화식 입력 모드에서는 DB2 명령에 db2 접두부를 첨부해서 입력할 필요가 없고, 대신 DB2 명령만을 입력하십시오. 예를 들면, 다음과 같습니다.

```
db2 => list database directory
```

운영 체제 명령을 대화식 모드로 입력하려면, 운영 체제 명령 앞에 느낌표(!)를 붙이십시오. 예를 들면, 다음과 같습니다.

```
db2 => !dir
```

한 행을 초과하는 긴 명령을 입력해야 할 경우에는, 공백 뒤에 행 계속 문자 \를 입력하십시오. 행 끝에 도달했을 때 계속해서 다음 행에 명령을 입력하려면 **Enter** 키를 누르십시오. 예를 들면, 다음과 같습니다.

```
db2 => select empno, function, firstname, lastname, birthdate, from \
db2 (cont.) => employee where function='service' and \
db2 (cont.) => firstname='Lily' order by empno desc
```

대화식 입력 모드를 종료하려면 **quit** 명령을 입력하십시오.

명령행 처리기를 사용하기 전에, 103 페이지의 『제7장 응용프로그램 및 유틸리티 바인딩』에서 설명된 대로 바인드되어야 합니다.

주: UNIX 기반 시스템의 명령행 처리기를 대화식 입력 모드가 아닌 모드에서 사용하는 경우, SQL문을 나타낼 때 특수 문자(예를 들어 *, ?)에 큰 따옴표로 묶어야 합니다.

*Command Reference*에서 SQL 명령에 대한 자세한 내용을 참조하십시오.

가져오기 및 내보내기 유틸리티 사용

가져오기 및 내보내기 유틸리티를 사용하면 데이터를 S/390 호스트나 AS/400 데이터베이스로부터 DB2 Connect 워크스테이션상의 파일로 이동할 수 있으며 그 반대도 가능합니다. 그후에 이 가져오기/내보내기 형식을 지원하는 다른 응용프로그램이나 RDBMS을 가지고 이 데이터를 사용할 수 있습니다. 예를 들어, 데이터를 OS/390용 DB2로부터 컬럼 식별자가 있는 ASCII 파일로 내보내기 하였다가 나중에 이를 Windows 2000용 DB2 UDB 데이터베이스로 가져오기할 수 있습니다.

내보내기 및 가져오기 기능을 데이터베이스 클라이언트로부터 또는 DB2 Connect 워크스테이션으로부터 실행할 수 있습니다.

주:

1. 가져오기되거나 내보내기된 데이터는 두 데이터베이스에서의 *크기*와 데이터 유형 제한사항이 서로 맞아야 합니다.
2. 가져오기 성능을 향상시키려면, 복합 SQL을 사용할 수 있습니다. COMPOUND=*number*를 가져오기 API 또는 CLP *파일 유형 모드* 문자열 매개변수에서 지정하여 지정된 SQL문을 블록으로 그룹화하십시오. 이렇게 지정하면 네트워크 오버헤드를 줄이고 응답 시간을 개선할 것입니다.
3. *Command Reference*에서 명령행 처리기로부터 가져오기 및 내보내기 유틸리티의 구문에 대한 내용을 참조하십시오.

워크스테이션에서 S/390 또는 AS/400 데이터베이스 서버로 데이터 이동

S/390 또는 AS/400 데이터베이스 서버로 내보내기하려면 다음을 수행하십시오.

1. Unix, Windows NT, Windows 2000, OS/2용 DB2 UDB 테이블을 PC/IXF 파일로 내보내기하십시오.
2. DRDA 서버 데이터베이스에 가져오기될 정보와 호환되는 속성을 가진 테이블이 들어 있지 않으면, 호환되는 테이블을 작성하십시오.
3. INSERT 옵션을 사용하여 PC/IXF 파일을 DRDA 서버 데이터베이스의 테이블로 가져오기 하십시오.

DRDA 서버에서 워크스테이션으로 데이터 이동

DRDA 서버 데이터베이스에서 데이터를 가져오기하려면 다음을 수행하십시오.

1. DRDA 서버 데이터베이스 테이블의 정보 행을 PC/IXF 파일로 내보내기하십시오.
2. PC/IXF 파일을 DB2 테이블로 가져오기하십시오.

제한사항

DB2 Connect 프로그램에서, 가져오기 또는 내보내기 조작은 다음 조건을 충족해야 합니다.

- 파일 유형은 PC/IXF여야 합니다.

- 색인 정의는 내보내기에 저장되거나 가져오기에 사용되지 않습니다.
- 가져오기하려면, 먼저 해당 데이터의 속성과 호환되는 속성을 가진 테이블이 있어야 합니다. DB2 Connect 프로그램을 통한 가져오기는 INSERT만 지원된 옵션이기 때문에 테이블을 작성할 수 없습니다.
- 확약 카운트 간격은 가져오기로 지정해서는 안됩니다.

이런 조건에 위반되면, 조작은 실패하고 오류 메시지가 생성됩니다.

1바이트 및 2바이트 혼합 데이터

혼합된 데이터(1바이트 및 2바이트 데이터 모두가 들어 있는 컬럼)를 가져오기하고 내보내기하려면, 다음을 생각하십시오.

- EBCDIC(MVS, OS/390, OS/400, VM, VSE)에 데이터를 저장하는 시스템에서, 시프트 인 및 시프트 아웃 문자는 2바이트의 시작과 끝을 표시합니다. 데이터베이스 테이블에 대한 컬럼 길이를 정의하려면, 이들 문자용 공간이 충분한지 확인하십시오.
- 변수 길이 문자 컬럼은 컬럼의 데이터가 일관성있는 양식이 아닌 경우에 권장됩니다. 일관성 있는 양식인 경우에는, 고정 길이가 선택됩니다.

SQLQMF 유틸리티 대체

OS/2용 DDCS의 SQLQMF 유틸리티 기능이 DB2 Connect 가져오기/내보내기 기능으로 바뀌었습니다. 이점은 다음과 같습니다.

- 호스트에서 QMF가 필요없습니다.
- 호스트로 로그온할 필요가 없습니다.(TSO ID는 여전히 MVS/ESA용 DB2 또는 OS/390용 DB2 Universal Database에서 요구됩니다.)
- MVS용 DB2, OS/390용 DB2, OS/400용 DB2, VM 및 VSE용 DB2를 지원합니다.
- 복합 SQL을 사용하여 성능이 향상됩니다.
- ASCII 이외에 여러 파일 형식을 지원합니다.
- SNA 연결없이 클라이언트 기기로부터 수행할 수 있습니다.

*Command Reference*에서 이 명령의 사용에 대한 자세한 내용을 참조하십시오.

제10장 보안

이 장에서는 인증 유형을 포함한 DB2 Connect 보안 고려사항을 설명합니다. 또한 OS/390용 DB2 Universal Database 사용자용 보안에 대한 추가 힌트와 추가 정보를 제공합니다.

DCE에서 보안을 설정하는 데 대한 자세한 내용은 *관리 안내서* 및 호스트 또는 AS/400 데이터베이스 서버용 데이터베이스 및 DCE 매뉴얼을 참조하십시오.

주: DCE 보안에 DB2 Connect를 사용하는 경우, DB2 클라이언트 워크스테이션과 호스트나 AS/400 데이터베이스 서버에 DCE 소프트웨어를 설치해야 하지만 DB2 Connect 워크스테이션에는 설치할 필요가 없습니다. *DB2 Connect 빠른 시작*에서 DCE에 대한 소프트웨어 전제조건에 대해 자세한 내용을 참조하십시오.

인증

DB2 Connect 관리자 자격으로, 호스트나 AS/400 데이터베이스 관리자와 함께 사용자 이름과 암호의 유효성이 점검되는 위치를 결정할 수 있습니다. 다섯 가지 경우가 가능합니다.

- 클라이언트에서의 유효성 점검
- DB2 Connect 워크스테이션에서의 유효성 점검
- DB2 Connect 워크스테이션과 호스트 또는 AS/400 서버 모두에서의 유효성 점검
- 호스트나 AS/400 서버에서의 유효성 검검
- DCE 보안 서버에서의 유효성 점검

시스템 데이터베이스 디렉토리에서 인증 유형 매개변수를 설정하고 APPC 또는 APPN 노드에 대한 노드 디렉토리에서 보안 유형 매개변수를 설정하여 유효성 점검이 실행되는 위치를 결정합니다. 이 디렉토리의 갱신에 대해서는 87 페이지의 『제6장 데이터베이스 디렉토리 갱신』에서 자세한 내용을 참조하십시오.

© Copyright IBM Corp. 1993, 2000

주:

1. DB2 Connect는 스스로 사용자의 유효성 점검없이 수행됩니다. DB2 Connect 워크스테이션이 유효성 점검을 수행하게 하려면, 지역 보안 서브시스템을 사용하여 CONNECT 요청에 제공되는 사용자 이름과 암호를 확인해야 합니다. 따라서, DB2 Connect Enterprise Edition 서버를 설정할 때 AUTHENTICATION=SERVER를 사용하면, 서버 시스템에서 필요한 사용자 ID와 암호 모두를 설정해야 합니다.
2. DCE 디렉토리 서비스를 사용하면, 인증은 다르게 작동합니다. 237 페이지의 『DCE 디렉토리 서비스에서의 보안』에서 자세한 내용을 참조하십시오.

다음 인증 유형은 DB2 Connect에서 허용됩니다.

CLIENT
사용자 이름과 암호의 클라이언트에서 유효합니다.

SERVER
사용자 이름과 암호의 유효성이 DB2 Connect 워크스테이션에서 점검됩니다. 인증이 지정되지 않았으면, **SERVER**로 간주됩니다.

SERVER_ENCRYPT
SERVER 인증의 경우, 사용자 이름과 암호는 DB2 Connect 워크스테이션에서 유효성이 점검되지만 전송된 암호는 클라이언트에서 암호화되어 DB2 Connect 워크스테이션에서 해독됩니다.

DCS 사용자 이름과 암호의 유효성이 호스트나 AS/400 데이터베이스 서버에서 점검됩니다.

DCS_ENCRYPT
DCS 인증의 경우, 사용자 이름과 암호는 호스트나 AS/400 데이터베이스 서버에서 유효성이 점검되지만 전송된 암호는 클라이언트에서 암호화되어 DB2 Connect 워크스테이션에 지정된 인증 유형에 따라 DB2 Connect 워크스테이션이나 호스트 또는 AS/400 데이터베이스 서버에서 해독됩니다.

DCE 사용자 이름과 암호의 유효성이 DCE 보안 서버에서 점검됩니다.

SERVER_ENCRYPT와 DCS_ENCRYPT 인증은 인증 위치면에서 SERVER 및 DCS 인증과 동일한 의미를 갖습니다. 차이점은 전송된 암호가 소스(클라이언트 또

는 DB2 Connect 서버)에서 암호화되어 카탈로그화된 인증 유형이 지정하는 목표(DB2 Connect 서버나, 호스트 또는 AS/400 데이터베이스 서버)에서 해독된다는 점입니다.

해당 인증 위치의 암호화된 값 및 암호화되지 않은 값은 클라이언트와 DB2 Connect 서버 사이 또는 DB2 Connect 서버와 호스트 또는 AS/400 데이터베이스 서버 사이에서 서로 다른 암호 조합을 선택하는 데 사용할 수 있으며, 인증이 발생하는 위치에는 아무런 영향을 미치지 않습니다. 다음은 게이트웨이 시나리오에서 사용되는 방법을 보여주는 예로, 여기서 "게이트웨이"는 DB2 Connect 서버를 가리키는 데 사용됩니다.

클라이언트에서의 인증	게이트웨이에서의 인증	인증 위치	클라이언트-게이트웨이 암호화?	게이트웨이-서버 암호화?
SERVER_ENCRYPT	SERVER	게이트웨이	예	아니오
DCS_ENCRYPT	DCS	서버	예	아니오
DCS	DCS_ENCRYPT	서버	아니오	예
DCS_ENCRYPT	DCS_ENCRYPT	서버	예	예

SERVER_ENCRYPT 또는 DCS_ENCRYPT가 사용되는 경우 지원되는 유일한 APPC 보안 매개변수는 SECURITY=NONE입니다.

주:

1. DB2 Connect가 연결하기 위해 사용하는 시스템 데이터베이스 디렉토리용으로, 인증 매개변수가 지정되어 있지 않은 경우에는, DB2 Connect는 인증 **SERVER**를 사용합니다.
2. DB2 Universal Database 클라이언트 서버 통신에서처럼 DB2 Connect Enterprise Edition 게이트웨이에 접속된 원격 클라이언트에는 인증 유형이 필요하지 않습니다. 성능을 최적화하는 데 도움이 되도록 인증 유형을 지정하면, 게이트웨이에서 이를 가져올 필요가 없으므로 트랜잭션에 대한 경과 시간을 줄일 수 있습니다.
3. 클라이언트의 값과 게이트웨이의 값 사이에 차이가 있는 경우, DB2 Connect 게이트웨이에 지정된 값을 우선으로 합니다.

보안 유형

이 절에서는 APPC 및 TCP/IP 연결을 통해 DB2 Connect에 지원되는 인증 및 보안 설정값의 여러 조합을 나열합니다.

다음 논의는 두 연결 유형 모두에 적용됩니다.

APPC 연결의 보안 유형

다음의 보안 유형은 어떤 보안 정보가 통신 계층으로 흘러가는지를 지정하기 위해서 APPC 연결용으로 허용됩니다.

SAME

사용자 이름만이 호스트나 AS/400 데이터베이스 서버로 전달됩니다.

PROGRAM

사용자 이름과 암호가 호스트나 AS/400 데이터베이스 서버로 전달됩니다.

NONE

어떤 보안 정보도 흘러가지 않습니다.

표5에서는 이 값들의 가능한 조합과 DB2 Connect 워크스테이션에서 지정된 인증 유형, 각 조합마다 유효성 점검이 수행되는 위치를 나타냅니다. 이 테이블에서 나타난 조합만이 APPC 연결에서 DB2 Connect에 의해 지원됩니다.

표 5. APPC 연결용으로 유효한 보안 시나리오

경우	**DB2 Connect 워크스테이션의 데이터베이스 디렉토리 항목에 있는 인증 문자열**	보안	유효성 점검
1	CLIENT	SAME	클라이언트
2	SERVER	SAME	DB2 Connect 서버
3	SERVER	PROGRAM	DB2 Connect 서버 및 호스트 또는 AS/400 데이터베이스 서버
4	SERVER_ENCRYPT 또는 DCS_ENCRYPT	NONE	호스트 또는 AS/400 데이터베이스 서버
5	DCS	PROGRAM	호스트 또는 AS/400 데이터베이스 서버

표 5. APPC 연결용으로 유효한 보안 시나리오 (계속)

경우	**DB2 Connect 워크스테이션의 데이터베이스 디렉토리 항목에 있는 인증 문자열**	보안	유효성 점검
6	DCE	NONE	DCE 보안 서버

원격 클라이언트가 DB2 Connect Enterprise Edition 서버에 연결된 경우, 다음과 같은 인증 및 보안 유형을 지정하십시오.

- APPC를 통해 원격 클라이언트가 DB2 Connect 서버에 연결된 경우, 원격 클라이언트에서 NONE 보안 유형을 지정하십시오.
- DB2 Connect 서버에서 데이터베이스 관리 프로그램 구성의 인증 유형이 CLIENT인 경우, 각 클라이언트에서 CLIENT를 지정하십시오.
- DB2 Connect 서버에서의 인증 유형이 SERVER, SERVER_ENCRYPT, DCS 또는 DCS_ENCRYPT인 경우, 각 원격 클라이언트에서 다음 중 하나를 지정하십시오. (원격 클라이언트에서 지정하는 이들 4가지 유형은 차이가 없습니다.)

주:

1. AIX 시스템의 경우, APPC 보안 유형 SAME을 사용하는 모든 로그인 사용자는 AIX 시스템 그룹에 속해 있어야 합니다.
2. 원격 클라이언트가 있는 AIX 시스템 경우, DB2 Connect 워크스테이션에서 수행중인 DB2 Connect 제품의 인스턴스는 AIX 시스템 그룹에 속해 있어야 합니다.
3. 호스트나 AS/400 데이터베이스 서버에 대한 액세스는 자체 보안 메커니즘이나 서브시스템에 의해 제어됩니다. 예를 들어, 가상 원격 통신 액세스 방식(VTAM)과 자원 액세스 제어 기능(RACF)이 있습니다. 보호된 데이터베이스 오브젝트로의 액세스는 SQL **GRANT** 및 **REVOKE** 명령문으로 제어됩니다.

TCP/IP 연결의 보안 유형

TCP/IP 통신 프로토콜은 네트워크 프로토콜 계층에서 보안 옵션을 지원하지 않습니다. 따라서, 인증 유형만이 인증이 수행되는 위치를 제어합니다. 이 테이블에서 나타난 조합만이 TCP/IP 연결에서 DB2 Connect에 의해 지원됩니다.

표 6. TCP/IP 연결용으로 유효한 보안 시나리오

경우	**DB2 Connect 워크스테이션의 데이터베이스 디렉토리 항목에 있는 인증 문자열**	유효성 점검
1	CLIENT	클라이언트
2	SERVER 또는 SERVER_ENCRYPT	DB2 Connect 워크스테이션
3	적용되지 않음	없음
4	DCS 또는 DCS_ENCRYPT	호스트 또는 AS/400 데이터베이스 서버
5	DCE	DCE 보안 서버

보안 유형에 대한 논의

다음 논의는 APPC 및 TCP/IP 연결에 적용되며, 위의 132 페이지의 표5와 표6에 나타나 있습니다. 각 경우는 다음과 같이 자세히 설명되어 있습니다.

- 1번의 경우에는, 사용자 이름과 암호가 원격 클라이언트에서만 유효합니다. (지역 클라이언트의 경우, 사용자 이름과 암호가 DB2 Connect 서버에서만 유효합니다.)

  사용자는 처음으로 사인온 한 위치에서 인증됩니다. 사용자 ID는 네트워크를 통하여 송신되지만 암호는 그렇지 않습니다. 모든 클라이언트 워크스테이션이 믿을 수 있는 적합한 보안 기능을 가지고 있을 경우에만, 이 보안 유형을 사용하십시오.

- 2번의 경우에는, 사용자 이름과 암호가 DB2 Connect 서버에서만 유효합니다. 암호는 네트워크를 통해 원격 클라이언트에서 DB2 Connect 서버로 송신되며 호스트나 AS/400 데이터베이스 서버로는 송신되지 않습니다.

- 3번의 경우에는, 사용자 이름과 암호가 DB2 Connect 서버와 호스트 또는 AS/400 데이터베이스 서버 모두에서 유효합니다. 암호는 네트워크를 통해 원격 클라이언트에서 DB2 Connect 워크스테이션으로, DB2 Connect 워크스테이션에서 호스트나 AS/400 데이터베이스 서버로 송신됩니다.

  유효성 점검이 두 군데에서 실행되므로, 동일한 사용자 이름과 암호 세트는 DB2 Connect 워크스테이션과 호스트 또는 AS/400 데이터베이스 서버에서 유지보수되어야 합니다.

- 4번의 경우에는, 사용자 이름과 암호가 호스트나 AS/400 데이터베이스 서버에서만 유효합니다. 사용자 ID와 암호는 네트워크를 통해 원격 클라이언트로부터 DB2 Connect 서버로, DB2 Connect 서버에서 호스트나 AS/400 데이터베이스 서버로 송신됩니다.
- 5번의 경우에는, DCE 암호화된 티켓을 클라이언트에 의해 DCE 보안 서버로부터 얻습니다. 티켓은 DB2 Connect를 통하여 서버로 변경되지 않은 상태로 전달되고, DCE 보안 서비스에 의해 유효성이 점검됩니다.

MVS 암호 변경

이제 DB2 Connect는 사용자 암호를 변경할 수 있는 기능을 제공합니다. 이 기능은 자원 액세스 제어 기능(RACF)와 같은 호스트 보안 서비스가 사용자 인증에 사용되는 경우에 특히 유용합니다. 이전에는 호스트 암호를 변경하려면 사용자가 TSO 세션에 로그인하여 자신의 암호를 변경해야 했습니다. DB2 Connect 제품이 제공하는 새로운 암호 유지보수 지원을 사용하면, 사용자는 DB2 명령행 처리기(CLP)에서 SQL CONNECT 명령을 발행하거나, DB2 클라이언트 구성 지원 프로그램(CCA)의 PASSWORD 버튼을 사용하거나, ODBC 로그인 대화 상자의 CHANGE 버튼을 눌러 호스트 암호를 변경할 수 있습니다.

TCP/IP를 통해 OS/390용 DB2 V5.1에 연결된 DB2 Connect 워크스테이션에서 MVS 암호를 변경하려면 DB2 OS/390 확장 보안 필드가 "Yes"로 설정되어 있어야 합니다. 이 필드는 DB2 OS/390 DSNTIPR 패널에 표시됩니다.

SNA를 통해 연결된 호스트 시스템에서 MVS 암호를 변경하려면 특별한 암호 만기 관리 프로그램이 호스트에 설정되어 있고 DB2 Connect 워크스테이션이 이 호스트 프로그램과 통신하도록 구성되어 있어야 합니다.

호스트 암호 만기 관리 프로그램은 다음과 같은 MVS 프로그램 제품의 일부로 제공됩니다.

- MVS/ESA SP 버전 4.2 이상(암호 만기 관리가 APPC/MVS 구성요소의 일부임)
- CICS/ESA 버전 3.3 이상

그리고 IBM 자원 액세스 제어 기능(RACF) 1.9.2가 설치되어 있어야 합니다.

다음과 같이 수행해야 합니다.

1. 암호 만기 유지보수 요청을 수신하도록 호스트의 트랜잭션 프로그램을 구성하십시오.
2. 호스트 트랜잭션 프로그램과 통신하도록 DB2 Connect 워크스테이션을 구성하십시오.

암호 만기 관리용으로 DB2 Connect 워크스테이션 구성

일단 호스트 암호 만기 관리 트랜잭션 프로그램이 구성되면, 호스트 프로그램과 통신하도록 DB2 Connect 워크스테이션을 구성해야 합니다. 이 구성에는 두 가지 단계가 필요합니다.

1. DB2 Connect 워크스테이션의 SNA 서브시스템에 호스트 암호 유지보수 프로그램을 위한 기호 목적지 이름을 정의합니다.
2. 이 기호 목적지 이름을 이 호스트 시스템에 상주하는 데이터베이스에 대한 DCS 디렉토리에 기록합니다.

단계 1. 기호 목적지 이름 정의

호스트 암호 만기 관리 프로그램에 대한 기호 목적지 이름을 정의하는 방법은 사용 중인 SNA 서브시스템에 따라 달라집니다.

- DB2 클라이언트 구성 지원 프로그램(CCA)으로 구성할 수 있는 SNA 서브시스템을 사용하고 있는 경우, CCA를 사용하여 이 기호 목적지 이름을 구성해야 합니다. MVS 관리자로부터 암호 만기 관리 프로그램에 대한 LU 이름을 입수해야 합니다.
- CCA를 사용하여 SNA 서브시스템을 구성할 수 없는 경우, SNA 서브시스템에 제공된 문서를 따라 기호 목적지 이름을 구성해야 합니다. MVS 관리자로부터 다음 정보를 입수해야 합니다.
  - 연결할 호스트의 네트워크 이름
  - 호스트 암호 만기 관리 프로그램의 LU 이름

  기호 목적지 이름을 구성하는 경우, 트랜잭션 프로그램(TP) 이름에 x'06F3F0F1'(16진 숫자)를 지정하고 보안을 NONE으로 설정해야 합니다. #INTER 또는 MVS에서 제안하는 다른 모드를 지정할 수 있습니다.

단계 2. 기호 목적지 이름을 DCS 디렉토리에 기록

CCA를 제공하는 플랫폼에서 DB2 Connect를 수행하는 경우, CCA를 사용하여 호스트 암호 만기 관리 프로그램에 대한 기호 목적지 이름으로 DCS 디렉토리를 갱신해야 합니다. 이는 DB2 Connect 워크스테이션의 SNA 서브시스템에 관계없이 수행할 수 있어야 합니다.

또한 **catalog dcs database** 명령(DB2 CLP의)을 사용하여 기호 목적지 이름을 DCS 디렉토리에 기록할 수 있습니다 예를 들면, 다음과 같습니다.

```
catalog dcs database db1 as dsn_db_1 parms ",,,,,,,CHGPWD_SDN=pempgm"
```

이는 사용자가 데이터베이스 *db1*에 대한 암호 변경을 요청할 때 사용할 기호 목적지 이름으로 *pempgm*을 기록합니다.

암호 만기 관리를 위한 호스트 구성

다음의 온라인 문서 중 하나에서 MVS 암호에 대한 자세한 내용을 참조하십시오.

APPC/MVS:

http://www.s390.ibm.com/products/appc/library

SecureWay 통신:

http://www.ibm.com/software/network/commserver/library

TxSeries:

http://www.ibm.com/software/ts/txseries/library

보안에 대한 추가 힌트와 추가 정보

이 절에서는 DB2 Connect의 사용자를 위해 보안에 대한 추가 힌트와 추가 정보를 제공합니다.

확장된 보안 코드

OS/390용 DB2 Universal Database 버전 5.1까지, 사용자 ID 또는 암호를 제공한 연결 요청이 SQL30082 이유 코드 0으로 실패할 수 있었지만 잘못된 내용에 대한 어떠한 지시 사항도 없었습니다.

OS/390용 DB2 Universal Database 버전 5.1은 확장된 보안 코드에 대한 지원을 제공하는 방향으로 향상되었습니다. 확장된 보안 코드를 지정하면 이유 코드 이외에 (PASSWORD EXPIRED)같은 추가 진단을 제공합니다.

이 기능을 사용하려면, 확장된 보안용 OS/390용 DB2 Universal Database ZPARM 설치 매개변수를 YES 값으로 설정해야 합니다. OS/390용 DB2 Universal Database 설치 패널 DSN6SYSP를 사용하여 EXTSEC=YES로 설정하십시오. 또한 DDF 패널 1(DSNTIPR)을 사용하여 설정할 수도 있습니다. 기본 값은 EXTSEC=NO입니다. 만기된 암호의 경우, DB2 Connect를 사용하는 PC, UNIX, Apple Macintosh 및 Web 응용프로그램은 오류 메시지 SQL01404를 수신합니다.

TCP/IP 보안이 확인됨

DB2 Universal Database 보안 옵션 AUTHENTICATION=CLIENT에 대한 지원을 제공하려면, OS/390용 DB2 Universal Database 설치 패널 DSNTIP4(DDF 패널 2)를 사용하여 TCP/IP 보안이 확인됨을 YES로 설정하십시오.

데스크탑 ODBC 및 Java 응용프로그램 보안

워크스테이션 ODBC와 자바 응용프로그램은 동적 SQL을 사용합니다. 이 응용프로그램은 여러 설치와 관계있는 보안을 작성합니다. OS/390용 DB2 Universal Database는 소유자나 바인더의 인증 하에서 동적 SQL을 실행할 수 있는 새로운 바인드 옵션 DYNAMICRULES(BIND)를 도입합니다. *Command Reference*에서 DB2 Connect를 통해 DYNAMICRULES를 지정하는 방법에 대한 내용을 참조하십시오.

DB2 Universal Database 및 DB2 Connect에서는 DB2CLI.INI 구성 파일에 새로운 CLI/ODBC 구성 매개변수 CURRENTPACKAGESET를 제공합니다. 이 매개변수는 적합한 특권이 있는 스키마 이름으로 설정해야 합니다. SQL SET CURRENT PACKAGESET 스키마 명령문이 응용프로그램용 연결마다 자동으로 실행됩니다.

ODBC 관리 프로그램을 사용하여 DB2CLI.INI를 갱신하십시오. *설치 및 구성 보충 설명서*에서 자세한 내용을 참조하십시오.

암호 변경 지원

SQL CONNECT문이 사용자 ID의 암호가 만기되었음을 나타내는 메시지를 리턴하면, DB2 Connect 버전 5.2 이상에서는 TSO에 알리지 않고도 암호를 변경할 수 있습니다. OS/390용 DB2 Universal Database에서는 DRDA를 통해 사용자의 암호를 변경할 수 있습니다.

사용자는 새 암호와 함께 이전 암호와 확인 암호를 제공해야 합니다. DB2 Connect Enterprise Edition 서버에 지정된 보안이 DCS이면 암호 변경 요청이 OS/390용 DB2 Universal Database 데이터베이스 서버로 전송됩니다. 지정된 보안이 SERVER이면 DB2 Connect 서버의 암호가 변경됩니다.

또한, 개별적인 LU 정의가 필요하지 않다는 것도 하나의 잇점입니다. DB2 Connect Enterprise Edition *빠른 시작* 매뉴얼에서 자세한 내용을 참조하십시오.

제11장 SQLCODE 맵핑

다른 IBM 관계형 데이터베이스 제품은 유사한 오류에 대해 동일한 SQLCODE를 제공하지 않습니다. SQLCODE가 같더라도, 지정된 토큰이 다를 수 있습니다. 토큰 목록은 SQLCA의 SQLERRMC 필드에서 전달됩니다. 기본적으로 DB2 Connect는 SQLCODE와 토큰을 각 IBM 호스트나 AS/400 데이터베이스 서버에서 적절한 DB2 Universal Database SQLCODE로 맵핑합니다.

SQLCODE 맵핑 중단

SQLCODE 맵핑을 중단하려면, DCS 디렉토리 또는 DCE 라우팅 정보 오브젝트의 매개변수 문자열에서 NOMAP을 지정하십시오. DCS 디렉토리 갱신에 대해서는 87 페이지의 『제6장 데이터베이스 디렉토리 갱신』에서 내용을 참조하십시오. DCE 사용에 대해서는 229 페이지의 『부록D. DCE 디렉토리 서비스 사용』에서 내용을 참조하십시오.

호스트나 AS/400 데이터베이스 서버(예: OS/390용 DB2 UDB)로부터 직접 응용프로그램을 이식하는 경우에는 SQLCODE 맵핑을 중단할 수 있습니다. 이런 경우, 참조하는 SQLCODE를 변경하지 않고 응용프로그램을 사용할 수 있습니다.

SQLCODE 맵핑 조정

기본적으로 DB2 Connect는 SQLCODE와 토큰을 각 IBM 호스트나 AS/400 데이터베이스 서버에서 적절한 DB2 UDB SQLCODE로 맵핑합니다. 다음 파일은 기본 SQLCODE 맵핑의 복사 파일입니다.

dcs1dsn.map

MVS/ESA용 DB2 및 OS/390용 DB2 UDB SQLCODE를 맵핑합니다.

dcs1ari.map

VSE & VM용 DB2 SQLCODE를 맵핑합니다.

© Copyright IBM Corp. 1993, 2000

dcs1qsq.map

AS/400용 DB2 UDB SQLCODE를 맵핑합니다.

OS/2 및 UNIX 기반 DB2 시스템에서 맵핑은 필요없습니다.

기본 SQLCODE 맵핑을 겹쳐쓰려 하거나 SQLCODE 맵핑이 없는(IBM이 아닌 데이터베이스 서버) 호스트나 AS/400 데이터베이스 서버를 사용하려면, 이들 파일 중 하나를 복사하여 새로운 SQLCODE 맵핑 파일에 대한 기본 파일로 사용할 수 있습니다. 직접 파일을 편집하기보다는 복사해 두면 필요한 경우 원래 SQLCODE 맵핑을 항상 참조할 수 있다는 점에 유의하십시오.

DOS 디렉토리 또는 DCE 라우팅 정보 오브젝트의 매개변수 문자열에서 새 SQLCODE 맵핑 파일의 파일 이름을 지정하십시오. DCS 디렉토리 갱신에 대해서는 87 페이지의 『제6장 데이터베이스 디렉토리 갱신』에서 내용을 참조하십시오. DCE 사용에 대해서는 229 페이지의 『부록D. DCE 디렉토리 서비스 사용』에서 내용을 참조하십시오.

각 맵핑 파일은 ASCII 파일이며, ASCII 편집기를 사용하여 작성하고 편집합니다. 처음 설치하면, 파일은 설치 경로의 map 디렉토리에 저장됩니다.

파일에는 다음과 같은 특정 유형의 라인이 포함될 수 있습니다.

&& 파일의 논리적 시작입니다. &&가 처음으로 나타나기 전 모든 라인은 특별한 형식이 없는 것으로 가정되며 무시됩니다. 파일에서 && 다음에 아무것도 없으면, SQLCODE 맵핑은 수행되지 않습니다. 이전에 설명한 대로 NOMAP 매개변수로 SQLCODE 맵핑을 중단할 수도 있습니다.

***** 라인에서 첫 번째 문자로서 주석을 나타냅니다.

W 라인에서 오직 하나의 문자로서, 경고 플래그가 다시 맵핑되는 것을 나타냅니다. 기본적으로, 원래 경고 플래그가 전달됩니다. W는 대문자여야 합니다.

&& 이후의 다른 라인은 모두 공백이거나 다음 형식의 맵핑 명령문이어야 합니다.

입력 코드 **[,** 출력 코드 **[,** 토큰 목록**]]**

*input_code*은 다음 중 하나입니다.

sqlcode

호스트나 AS/400 데이터베이스 서버의 SQLCODE.

U 정의되지 않은 음수의 SQLCODE(이 파일에는 없는 코드)는 모두 지정된 *출력_코드*로 맵됩니다. 이 행에 *출력_코드*가 지정되지 않으면, 원래 SQLCODE가 사용됩니다. 이 문자는 대문자여야 합니다.

P 정의되지 않은 양수의 SQLCODE(이 파일에는 없는 코드)는 모두 지정된 *출력_코드*로 맵됩니다. 이 행에 *출력_코드*가 지정되지 않으면, 원래 SQLCODE가 사용됩니다. 이 문자는 대문자여야 합니다.

cc*nn* 호스트나 AS/400 데이터베이스 서버의 SQLSTATE 클래스 코드.*nn*은 다음 중 하나입니다.

00 규정되지 않은 성공적 완료

01 경고

02 데이터 없음

21 기본 행수(cardinality) 위반

22 데이터 예외

23 제한 조건 위반

24 유효하지 않은 커서 안정성(CS)

26 유효하지 않은 SQL문 식별자

40 트랜잭션 구간 복원

42 액세스 위반

51 유효하지 않은 응용프로그램 상태

55 오브젝트가 전제조건 상태에 없음

56 기타 SQL 또는 제품 오류

57 자원 사용 불능 또는 조작원 개입

58 시스템 오류

지정된 *출력_코드*는 맵핑 파일에서 명시적으로 지정되지 않은 이러한 클래스 코드가 있는 모든 SQLCODE에 사용됩니다. 이 행에 *출력_코드*가 지정되지 않으면, 원래 SQLCODE가 토큰 복사없이 자체로 맵핑됩니다.

문자 **cc**는 소문자여야 합니다.

같은 *입력_코드*가 맵핑 파일에서 두 번 이상 나타나면, 처음 나타난 코드가 사용됩니다.

*output_code*는 출력 SQLCODE입니다. 지정된 값이 없으면, 원래 SQLCODE가 사용됩니다.

출력 코드를 지정하면, 다음 중 하나를 지정할 수도 있습니다.

(s) 입력 SQLCODE+제품 ID(ARI, DSN 또는 QSQ)는 SQLCA 메시지 토큰 필드로 들어갈 것입니다.

원래 SQLCODE가 유일한 토큰으로 리턴됩니다. 이 옵션은 +965와 -969를 제외하고 정의되지 않은 SQLCODE를 처리하도록 설계되었습니다. +965나 -969가 *출력_코드*인 경우, SQLCA의 SQLERRMC 필드에서 리턴된 토큰 목록에는 원래 SQLCODE가 있고, 제품 식별자, 원래 토큰 목록이 다음에 옵니다.

문자 **s**는 소문자여야 합니다.

(토큰-목록)

토큰 목록은 쉼표로 나눕니다. 쉼표만 지정하여 특정한 토큰을 생략하십시오. 예를 들어, 형식(,*t2*,,*t4*)은 첫 번째 및 세 번째 출력 토큰이 널(NULL)이라는 것을 의미합니다.

토큰마다 숫자의 형식(*n*)이 있으며, 선택적으로 **c**가 앞에 오거나, 선택적으로 **c** 또는 **i**가 올 수 있습니다. 다음과 같이 해석됩니다.

c 여기서, 토큰의 데이터 유형은 CHAR(기본 값)입니다. **c**가 *n*전에 나오는 경우 입력 토큰을 참조하고, *n* 다음에 나오면 출력 토큰을 참조하십시오. 문자 **c**는 소문자여야 합니다.

i 여기서, 토큰의 데이터 유형은 INTEGER입니다. **i**가 *n* 다음에 나오면, 출력 토큰을 참조하십시오. **i**는 *n*전에 나올 수 없는데, IBM

호스트나 AS/400 데이터베이스 서버 제품은 CHAR 토큰만 지원하기 때문입니다. 문자 **i**는 소문자여야 합니다.

n 사용되는 호스트나 AS/400 데이터베이스 서버를 나타내는 번호 수. 번호는 출력 SQLCA에서 예정된 위치 순서로 배열됩니다. 번호는 호스트나 AS/400 데이터베이스 서버 토큰을 나타내고 배열은 SQLCA에서 토큰이 배치되는 순서를 나타냅니다.

예를 들어, 호스트나 AS/400 데이터베이스 서버는 1과 2의 두 토큰을 리턴합니다. 출력 SQLCA에서 토큰 2를 토큰 1 전에 나타나게 하려면, (2,1)를 지정하십시오.

많은 토큰 번호는 마침표로 연결하여 CHAR 출력 토큰 형식으로 결합할 수 있습니다.

쉼표는 출력 토큰을 분리하는 데 사용됩니다. 쉼표 앞에 지정된 토큰이 없으면, 거기서 SQLCA에 포함되는 출력 토큰은 없습니다. 마지막으로 지정된 토큰 다음에 나타나는 출력 SQLCA의 토큰은 모두 널(NULL) 토큰으로 맵핑됩니다.

그림6에서 샘플 SQLCODE 맵핑 파일을 나타냅니다.

```
&&
  -007    ,   -007   ,   (1)
  -010
  -060    ,   -171   ,   (2)
...
  -204    ,   -204   ,   (c1.2c)
...
  -633    ,   -206   ,   (,c1i)

  -30021  ,   -30021 ,   (c1c,c2c)

  cc00    ,   +000
...
  U       ,   -969   ,   (s)
  P       ,   +965   ,   (s)
```

그림 6. SQLCODE 맵핑 파일

파일의 각 맵핑 명령문은 아래에 설명되어 있습니다.

1. SQLCODE는 -007에서 -007까지 맵핑됩니다. 호스트나 AS/400 데이터베이스 서버로부터 수신된 첫 번째 입력 토큰이 첫 번째 입력 토큰으로 사용되는데, 기본 값은 CHAR입니다. 다른 토큰은 전송되지 않습니다.
2. SQLCODE는 -010에서 -010까지 맵핑됩니다(지정된 출력 SQLCODE가 없습니다). 출력 SQLCA로 들어간 토큰은 없습니다.
3. SQLCODE는 -060에서 -171까지 맵핑됩니다. 호스트나 AS/400 데이터베이스 서버로부터 수신된 첫 번째 입력 토큰은 버려집니다. 두 번째 토큰은 출력 SQLCA에서 첫 번째 토큰으로 사용되었고, CHAR입니다. 출력 SQLCA에 두 번째 토큰은 없습니다.
4. SQLCODE는 -204에서 -204까지 맵핑됩니다. 호스트나 AS/400 데이터베이스 서버로부터 수신된 첫 번째와 두 번째 토큰은 CHAR입니다. 두 입력 토큰은 CHAR 출력 토큰을 형성하기 위해 결합되며, SQLCA의 첫 번째 출력 토큰이 됩니다.
5. SQLCODE는 -633에서 -206까지 맵핑됩니다. 호스트나 AS/400 데이터베이스 서버로부터 수신된 첫 번째 입력 토큰은 CHAR입니다. 토큰은 INTEGER로 변환되며 출력 SQLCA에서 두 번째 토큰으로 사용됩니다. 출력 SQLCA의 첫 번째 토큰은 널(NULL)이고 쉼표로 나타납니다.
6. SQLCODE는 -30021에서 -30021까지 맵핑됩니다. 호스트나 AS/400 데이터베이스 서버로부터 수신된 첫 번째와 두 번째 토큰은 CHAR이고, 이들은 출력 SQLCA에서 첫 번째와 두 번째 토큰으로 사용됩니다.
7. 00 클래스에서 SQLSTATE과 함께 SQLCA의 SQLCODE는 모두 SQLCODE +000로 맵핑됩니다.
8. 정의되지 않은 모든 SQLCODE는 -969로 맵핑됩니다. 이 옵션은 모든 맵핑이 가능한, 식별할 수 있으며 맵핑이 필요없는 코드를 포함한 코드가 나열된 경우에만 사용할 수 있습니다. **(s)** 옵션은 SQLCA의 SQLERRMC 필드에서 리턴된 토큰 목록에 원래 SQLCODE가 있으며, 오류가 발생한 제품, 원래 토큰 목록이 다음에 온다는 것을 나타냅니다. **U** 항목이 없으면, 나열되지 않은 코드 모두는 맵핑 없이 패스됩니다.
9. 정의되지 않은 긍정 SQLCODE는 모두 +965로 맵핑됩니다. 이 옵션은 모든 맵핑이 가능한, 식별할 수 있으며 맵핑이 필요없는 코드를 포함한 코드가 나열된 경우에만 사용할 수 있습니다. **(s)** 옵션은 SQLCA의 SQLERRMC 필드에서 리턴된 토큰 목록에는 원래 SQLCODE가 있으며, 경고가 발생한 제품, 원래 토큰 목록이 다음에 온다는 것을 나타냅니다. **P** 항목이 없으면, 나열되지 않은 긍정 코드 모두는 맵핑없이 전달됩니다.

제12장 성능

DB2 Connect 제품은 DRDA 응용프로그램 서버(AS), 클라이언트 제품, 통신 제품을 포함한 많은 다른 제품과 상호 작용합니다. 성능은 같이 작업하는 제품에 따라 결정됩니다.

성능 개념과 도구

*성능*은 컴퓨터 시스템이 주어진 특정 워크로드를 수행하는 방법입니다. 이는 사용 가능한 자원에 의해 영향을 받고 사용 및 공유된 방식에 영향을 받습니다. 성능을 향상시키려면, 성능에 대해 원하는 것을 먼저 결정해야 합니다. 다음을 포함한 다른 많은 *성능 메트릭*을 선택할 수 있습니다.

응답 시간
응용프로그램이 데이터베이스 요청을 송신하는 시간과 응용프로그램이 응답을 수신하는 시간 사이의 간격.

트랜잭션 처리량
시간 단위마다 완료되는 작업 단위(UOW)의 수. 작업 단위(UOW)는 행 페치와 갱신 같이 단순할 수도 있으며, 수백개의 SQL문을 포함하여 복잡할 수도 있습니다.

데이터 전송율
시간 단위마다 DB2 Connect 응용프로그램과 호스트 또는 AS/400 데이터베이스 사이에 전송되는 데이터의 바이트 수.

사용가능한 하드웨어와 소프트웨어 자원은 성능에 제약할 것입니다. CPU, 메모리 및 네트워크 어댑터가 하드웨어 자원의 예입니다. 통신 서브시스템, 페이징 서브시스템, AIX용 `mbuf`, SNA용 링크는 소프트웨어 자원의 예입니다.

© Copyright IBM Corp. 1993, 2000

데이터 흐름

그림7은 DB2 Connect를 통한 호스트 또는 AS/400 데이터베이스 서버와 워크스테이션간의 데이터 흐름 경로를 나타냅니다.

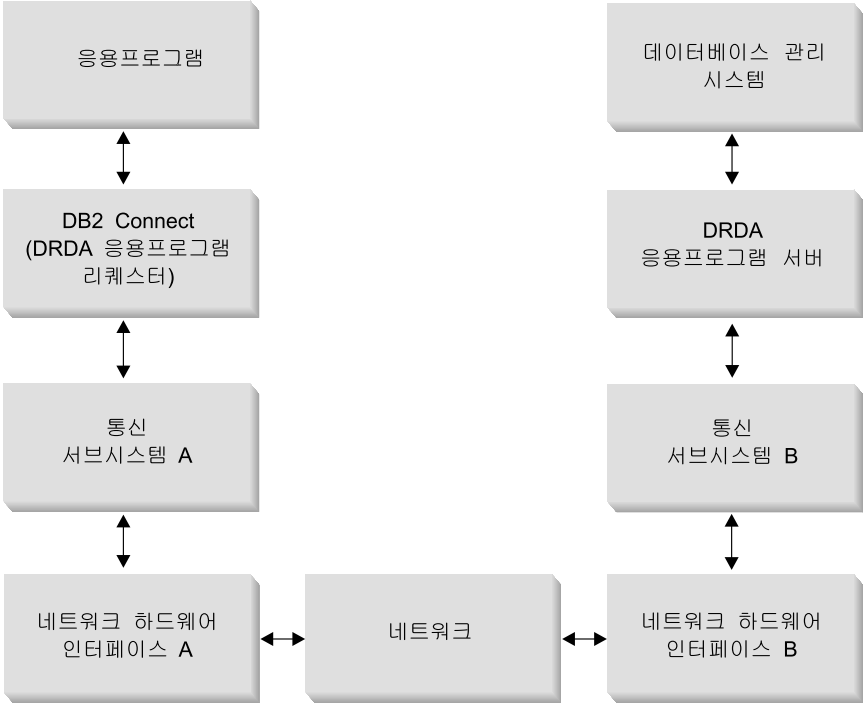


그림 7. DB2 Connect의 데이터 흐름

- 호스트나 AS/400 데이터베이스 서버와 통신 서브시스템 B의 일부는 보통 같은 시스템에서 수행됩니다. 이 시스템은 하나 이상의 CPU, 기본 저장영역, I/O 서브시스템, DASD, 운영 체제로 이루어져 있습니다. 다른 프로그램이 이 구성요소를 공유하기 때문에, 자원 경합은 성능상의 문제를 일으킬 수 있습니다.
- 네트워크는 케이블, 허브, 통신 회선, 스위치, 기타 통신 제어기의 결합으로 이루어져 있습니다. 예를 들어, 네트워크 하드웨어 인터페이스 B는 3745 또는 3172 같은 통신 제어기이거나 AS/400용 토큰 링 어댑터가 될 수 있습니다. 네트워크 하드웨어 A와 B 사이에는 하나 이상의 전송 미디어가 있습니다.

- 네트워크 하드웨어 인터페이스 A는 토큰 링, Ethernet**, 기타 LAN 어댑터 또는 SDLC나 X.25 프로토콜을 지원하는 어댑터가 될 수 있습니다. 통신 서브시스템 A는 OS/2용 IBM 통신 서버, Microsoft SNA 서버, AIX용 IBM SNA 서버 또는 HU-UX용 SNAplus2와 같은 제품이 될 수 있습니다.
- DB2 Connect 제품과 통신 서브시스템 A는 흔히 같은 시스템에 있습니다. 이 장에서는 응용프로그램도 같은 시스템에 있다고 가정합니다.

병목 현상

트랜잭션 처리량은 시스템의 가장 속도가 느린 구성요소에 따라 결정됩니다. 성능 병목 현상을 식별하려면, 구성 매개변수를 변경하거나 문제되는 구성요소에 더 많은 자원을 할당하거나 구성요소를 업그레이드하거나 새로운 구성요소를 추가하여 일부 작업을 로드 해제함으로써 종종 문제를 완화시킬 수 있습니다.

다양한 도구를 사용하여 구성요소마다 걸리는 조회 시간을 결정할 수 있습니다. 성능을 향상하기 위해 조정되고 업그레이드된 구성요소를 알 수 있습니다. 예를 들어, 조회에 걸리는 시간이 DB2 Connect 머신 시간의 60%라고 결정하면, DB2 Connect를 조정하거나(원격 클라이언트가 있으면) 또다른 DB2 Connect 기계를 네트워크에 추가할 수 있습니다.

성능 도구에 대해서는 150 페이지의 『성능 도구』에서 자세한 내용을 참조하십시오.

벤치마킹

*벤치마킹*은 임의의 환경 성능과 다른 환경 성능을 비교하는 방법입니다.

벤치마킹은 정상적인 환경에서 테스트 응용프로그램을 수행하여 시작할 수 있습니다. 성능상의 문제가 좁혀짐에 따라, 특정한 테스트 케이스가 기능을 테스트하고 관찰할 범위를 제한하도록 개발될 수 있습니다.

벤치마킹은 복잡할 필요가 없습니다. 지정된 테스트 케이스는 원하는 정보를 얻기 위해 응용프로그램 전체를 에뮬레이트할 필요가 없습니다. 간단한 수치로 시작하여 보증이 되는 경우에만 복잡성을 계속 증가시키십시오.

바람직한 벤치마킹의 특성은 다음과 같습니다.

- 각 테스트는 반복 가능합니다.

- 각 테스트 반복은 동일한 시스템 상태에서 시작됩니다.
- 벤치마킹에 사용된 하드웨어와 소프트웨어는 사용자의 제품 환경과 일치합니다.
- 측정 중인 시스템 이외의 다른 시스템에 활동 중인 다른 기능이나 응용프로그램이 없습니다. (시스템에서 진행 중인 기타 다른 활동량이 시나리오에 포함되지 않은 한).

  주: 시작된 응용프로그램은 최소화되거나 유휴 상태가 되더라도 메모리를 사용합니다. 이것은 페이징을 일으키고 벤치마크 결과와 다르게 할 수 있습니다.

성능 도구

다음 표에서는 시스템 성능 측정을 도와주는 여러 도구를 나열합니다. 이런 도구은 자체적으로 시스템 자원을 사용하므로, 도구를 항상 사용 중인 상태로 할 수 없습니다.

표 7. 성능 도구

시스템	도구	설명.
CPU와 메모리 사용		
AIX	vmstat, time, ps, tprof	DB2 Connect 워크스테이션과 원격 클라이언트에서 CPU 또는 메모리 경합 문제점에 대한 정보를 제공합니다.
HP-UX	사용가능하면 vmstat, time, ps, monitor, glance	
OS/2	SPM/2, THESEUS/2, pstat	
Win NT 및 Windows 2000	MS 성능 모니터	
데이터베이스 활동		
모두	데이터베이스 모니터	문제가 데이터베이스로부터 발생했는지 판별합니다.
MVS 또는 OS/390	DB2PM (IBM), OMEGAMON/DB2 (Candle), TMON (Landmark), INSIGHT (Goal Systems), DB2AM (BMC)	
Win NT 및 Windows 2000	MS 성능 모니터	
네트워크 활동		

표 7. 성능 도구 (계속)

시스템	도구	설명.
AIX	netpmon	초단위의 패킷 수나 프레임 수와 같은 TCP/IP 및 SNA 통계를 포함한 낮은 레벨 네트워크 통계를 보고합니다.
DOS 또는 OS/2	토큰링 네트워크 16/4 추적 및 성능 프로그램	대부분의 네트워크 모니터는 플랫폼 종속입니다. 이 도구은 토큰링 전용으로 작업합니다.
3745 같은 네트워크 제어기	NetView 성능 모니터	통신 제어 이용과 VTAM 이용을 보고합니다.
OS/2	DatagLANce	성능 관련 데이터를 사용자에게 그림으로 나타내는 추적 도구.
UNIX 기반	netstat	TCP/IP 통신량을 핸들합니다.

ODBC 액세스 최적화

DB2 Universal Database는 ODBC를 통한 통신 성능을 개선하기 위해 설계된 특수 최적화를 제공합니다. 이 향상된 기능들은 Microsoft Access, Lotus Approach 또는 Visual Basic에 적용될 수 있습니다. DB2의 클라이언트 구성 지원 프로그램(CCA)을 사용하여 보다 빨라진 ODBC 처리율의 잇점을 얻을 수 있습니다.

최적화된 ODBC를 활성화하려면 다음을 수행하십시오.

신규 연결을 정의할 경우:

1. DB2 CCA를 시작하십시오.
2. 최적화하려는 데이터베이스 별명을 선택하십시오.
3. **등록 정보** 누름 버튼을 클릭하십시오.
4. **이 데이터베이스를 ODBC용으로 등록** 선택란을 선택하십시오.
5. 해당 데이터베이스를 등록하는 방법에 대해 설명하는 단일선택 버튼을 선택하십시오.
6. **설정** 누름 버튼을 클릭하십시오.
7. **최적화** 누름 버튼을 클릭한 후 ODBC 설정값을 최적화할 응용프로그램을 선택하십시오.

8. **확인**을 클릭하여 CCA에서 나가십시오.

기존 연결을 갱신하려면 다음을 수행하십시오.

1. DB2 CCA를 시작하십시오.
2. 최적화하려는 데이터베이스 별명을 선택하십시오.
3. **등록 정보** 누름 버튼을 클릭하십시오.
4. **설정** 누름 버튼을 클릭하십시오.
5. CLI/ODBC 설정 창에서 **최적화** 누름 버튼을 클릭한 후 최적화할 응용프로그램을 선택하십시오.
6. **확인**을 클릭하여 CCA에서 나가십시오.

DB2 Connect 빠른 시작 책에서 CCA에 대한 자세한 내용을 참조하십시오. 브라우저를 통해 다음의 주소를 찾아보면 OS/390용 DB2에 대한 ODBC 최적화 관련 최신 정보를 볼 수 있습니다. http://www.ibm.com/software/data/db2/os390/odbcatlg.html

응용프로그램 설계

응용프로그램을 작성하는 경우, 다음의 여러 방법으로 성능을 향상시킬 수 있습니다.

- 복합 SQL과 저장 프로시듀어를 사용하십시오.
- 요청을 그룹화하십시오.
- 술어 논리를 사용하여 필요한 데이터만 요청하십시오.
- 데이터 블로킹을 사용하십시오.
- 가능한 항상 정적 SQL을 사용하십시오.

복합 SQL과 저장 프로시듀어

많은 명령과 응답을 송신하고 수신하는 응용프로그램에서, 네트워크 오버헤드는 심각한 문제입니다. 복합 SQL과 저장 프로시듀어는 이 오버헤드를 줄이는 두 가지 방법입니다.

응용프로그램이 프로그래밍 논리의 간섭 없이 여러 SQL문을 송신하는 경우, 복합 SQL을 사용할 수 있습니다. SQL문 그룹에 프로그래밍 논리가 필요한 경우에는 저장 프로시듀어를 사용할 수 있습니다.

다음 명령을 제외한 모든 실행 가능한 명령문은 복합 SQL문 안에 들어 있습니다.

```
       CALL
       FETCH
       CLOSE
       OPEN
       Compound SQL
       Connect
       Prepare
Release
       Describe
       Rollback
       Disconnect
       Set connection
       execute immediate
```

*SQL 참조서*에서 자세한 내용을 참조하십시오.

응용프로그램에서 복합 SQL 사용에 대해서는 61 페이지의 『NOT ATOMIC 복합 SQL』에서 자세한 정보를 참조하십시오. 126 페이지의 『가져오기 및 내보내기 유틸리티 사용』에서 가져오기 유틸리티와 함께 복합 SQL에 대한 자세한 내용을 참조하십시오.

저장 프로시듀어는 프로그램 로직을 서버에 배치함으로써 네트워크 통신량을 줄여 줍니다. 버전 5.0 이전의 DB2에서는 저장 프로시듀어가 출력 매개변수만을 리턴할 수 있었으며 응용프로그램을 통해 별도의 확약 명령을 발행해야 했습니다. 이는 두 번의 네트워크 이동을 발생시킵니다. DB2 버전 5.0 이상에서는 프로시듀어에서 나갈 때 자동으로 확약할 수 있습니다. 또한 결과 집합을 리턴할 수도 있어서 클라이언트에서의 응용프로그램 로직이 최소화됩니다.

저장 프로시듀어의 사용에 대해서는 58 페이지의 『저장 프로시듀어』에서 정보를 참조하십시오.

요청 그룹화

관련 데이터베이스 요청(SQL문)을 하나의 데이터베이스 요청으로 그룹화하면 네트워크를 통하여 전송되는 요청 수와 응답 수를 줄일 수 있습니다. 예를 들어, 다음 명령문을

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=2
```

다음처럼

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1 OR ROW_ID=2
```

그룹화하면 네트워크를 통하여 더 적은 요청을 송신합니다.

IN과 BETWEEN과 같은 키워드를 사용하여 리턴되는 행 수를 줄일 수도 있습니다. 추가로, UPDATE와 DELETE문에서 WHERE, IN, BETWEEN 키워드를 사용할 수 있습니다.

술어 논리

술어 논리를 사용하여 필요한 행과 컬럼만을 요청할 수 있습니다. 그러면 데이터 전송에서 네트워크 통신량과 CPU 오버헤드를 최소화합니다.

예를 들어, 다음과 같은 조회를 사용하지 마십시오.

```
SELECT * FROM TABLEA
```

(TABLEA with ROW_ID=1의 첫 번째 행이 정말로 필요하다거나 컬럼 1과 컬럼 2만 필요한 경우입니다.)

데이터 블로킹

서버로부터 많은 양의 데이터를 원하는 경우 데이터 블로킹을 사용해야 합니다. 블로킹은 네트워크 대역폭의 활용을 개선하고 호스트나 AS/400 데이터베이스 서버 및 DB2 Connect 워크스테이션 양쪽 모두의 CPU 오버헤드를 줄여줍니다.

크기에 관계없이 메시지마다 고정된 CPU 양과 네트워크 오버헤드량이 있습니다. 데이터 블로킹은 같은 양의 데이터 전송에 필요한 메시지 수를 줄입니다.

블로킹을 사용하면, 조회로부터의 첫 번째 데이터 행은 첫 번째 블록이 수신될 때까지 응용프로그램에 전달되지 않습니다. 블로킹으로 첫 번째 행에 대한 검색 시간은 증가하지만, 다음 행에 대한 검색 시간은 개선됩니다.

사용된 메모리 양을 고려해 볼 수 있습니다. 메모리 작업 세트는 블로킹이 작동되면 보통 증가합니다. *DRDA Connectivity Guide*에서 SNA 연결 사용시 블로킹에 대한 자세한 내용을 참조하십시오.

DB2 Connect에서, 각 블록 내에서 전송되는 데이터의 양을 157 페이지의 『RQRIOBLK』에 설명된 것처럼 제어할 수 있습니다

블로킹을 호출하려면, prep 또는 바인드 명령의 BLOCKING 옵션을 사용하십시오.(109 페이지의 『BIND 명령』에서 자세한 내용을 참조.) 블로킹은 다음 경우에 작동됩니다.

- 커서가 읽기 전용이거나
- 커서가 미결정(ambiguous) 상태이고 블로킹이 prep 또는 바인드중에 지정된 경우.

읽기 전용, 갱신 가능, 불명확한 커서의 정의에 대해서는 *응용프로그램 개발 안내서*에서 내용을 참조하십시오.

주: 동적 SQL을 사용할 경우, 커서는 항상 미결정(ambiguous) 상태입니다.

BLOCKING을 가진 SQL문

갱신 가능한 SELECT 명령문(UPDATE/DELETE WHERE CURRENT OF 명령문을 사용하는)은 블로킹이 아닌 조회이므로, 확실히 필요한 경우에만 사용해야 합니다.

갱신 가능한 SELECT로 행은 SELECT가 완료된 시간과 UPDATE/DELETE가 나타나는 시간 사이에는 변경되지 않는다는 것을 보장합니다. 이동시 처리 레벨이 사용자 응용프로그램에서 중요하지 않으면, 갱신이 가능하지 않은 SELECT로부터 리턴된 값에 기초한 검색 기준으로 DELETE 또는 UPDATE를 사용할 수 있습니다.

읽기 전용 SELECT에 대해, FOR FETCH ONLY를 지정하십시오(VM과 VSE에서는 지원되지 않으므로 제외).

정적 SQL과 동적 SQL

가능하면 정적 SQL을 사용하십시오. 이것은 런타임 SQL 섹션 준비와 미결정(ambiguous) 커서를 피합니다. 동적 SQL을 반드시 사용해야 하는 경우, 다음 단계를 수행하여 네트워크 통신량을 최소화하고 성능을 향상시킬 수 있습니다.

- 명령문이 SELECT이고 준비되었으면, PREPARE ... INTO SQLDA를 수행하십시오. SQLDA는 설정에 필요한 전체 크기로 할당되어야 합니다. 최대 컬럼 수가 *x*이고 변경되지 않는 경우 *x* SQLVAR로 SQLDA를 할당하십시오. 가능한 컬럼 수를 알 수 없으면(메모리는 문제가 되지 않음), 최대 SQLVAR 수(256)를 사용하십시오.

  SQLDA 할당이 충분하지 않아서 리턴하는 SQLDA를 저장할 수 없으면, 프로그램은 그 결과를 다시 저장할 수 있는 큰 SQLDA로 다른 DESCRIBE를 실행해야 합니다. 이로 인해 네트워크 통신량을 증가시킵니다.

  PREPARE 순서와 DESCRIBE 순서를 사용하지 마십시오. PREPARE.....INTO 문을 사용하면 더 나은 성능을 얻을 수 있습니다.
- 동적 COMMIT 또는 ROLLBACK문 대신에 정적으로 바인드된 SQL COMMIT 또는 ROLLBACK문을 수행하십시오.
- SELECT, COMMIT 또는 ROLLBACK문이 아니면, PREPARE 순서와 EXECUTE 순서 대신에 명령문을 실행하기 위해 EXECUTE IMMEDIATE를 실행하십시오.
- ODBC 응용프로그램은 동적 SQL을 사용합니다. CLI/ODBC 정적 프로파일링을 사용하여 성능을 개선할 수도 있습니다. 이 기능을 통해 ODBC 호출을 캡쳐하여 데이터베이스 패키지에 저장된 정적 명령문으로 변환시킬 수 있습니다. 실제적으로 얻게 될 성능은 응용프로그램의 복합도에 따라 다릅니다. *CLI Guide and Reference*에서 자세한 내용을 참조하십시오.

기타 SQL 고려사항

명령행 처리기는 데이터베이스 엔진에서 SQL을 실행하기 전 입력을 구문 분석해야 하므로 명령행 처리기를 사용하면 일반적으로 프로그램에서 동적 SQL보다 속도가 더 느립니다 명령행 처리기는 또한 수신된 데이터를 형식화하는데, 사용자의 응용프로그램에서는 이 과정이 필요없습니다

인터프리터된 언어(REXX와 같은)의 SQL문은 대체적으로 컴파일러 언어(C와 같은)의 같은 SQL문보다 속도가 더 느립니다.

CONNECT문에는 유형 1과 유형 2라는 두 가지 유형이 있습니다. 유형 2로, 데이터베이스에 연결하는 것은 이전의 연결을 잠복 상태로만 두고 삭제하지는 않습니다. 나중에 잠복 연결로 스위치하면, 로딩 라이브러리와 내부 데이터 구조 설정의 오버헤드를 피합니다. 이런 이유로, 유형 2 연결을 사용하여 하나 이상의 데이터베이스를 액세스하는 응용프로그램 성능을 향상할 수 있습니다. *관리 안내서* 및 *SQL 참조서*에서 유형 2 연결에 대한 자세한 내용을 참조하십시오.

DB2 Connect 조정

데이터베이스 관리 프로그램 구성 파일의 다양한 매개변수를 사용하여 DB2 Connect를 조정할 수 있습니다. *관리 안내서*에서 이 매개변수의 변경에 대한 자세한 내용을 참조하십시오.

RQRIOBLK

RQRIOBLK 매개변수는 네트워크 입출력 블록의 최대 *크기*를 설정합니다. 블록 *크기*를 더 크게 설정하면 많은 요청의 성능을 향상시킵니다. 블록 *크기*는 한 행의 데이터 요청 같은 적은 요청에 대한 응답 시간에 보통 영향을 주지 않습니다.

블록 *크기*가 더 크면 보통 DB2 Connect 워크스테이션에 더 많은 메모리가 필요합니다. 작업 세트의 *크기*를 증가시키고 작은 워크스테이션에서의 페이징 양이 많아집니다.

응용프로그램을 실행하는데 너무 많은 페이징이 생기지 않도록 기본 DRDA 블록 *크기*(32767)를 사용하십시오. 그렇지 않으면, 페이징이 없어질 때까지 입출력 블록 *크기*를 줄이십시오. 페이징이 시작되면, 성능이 급격히 저하됩니다. 성능 모니터 도구(UNIX 기반 시스템의 vmstat 도구 또는 OS/2용 SPM/2 같은)을 사용하여 시스템에 페이징이 생기는지 판별하십시오. 기타 도구에 대해서는 150 페이지의 『성능 도구』에서 자세한 내용을 참조하십시오.

DIR_CACHE

DIR_CACHE 매개변수는 디렉토리 정보가 캐쉬되었는지 판별합니다. 캐쉬(DIR_CACHE=YES)이 있으면, 디렉토리 파일은 메모리에서 읽히고 캐쉬되어, 연결이 이루어질 때마다 내부 디렉토리 구조를 작성하는 오버헤드와 디렉토리 파일을 읽는 오버헤드를 최소화합니다.

캐쉬(DIR_CACHE=NO)이 없으면, 데이터베이스에 연결할 때마다 적합한 디렉토리를 디스크로부터 읽고 검색이 수행됩니다. 요청된 항목을 찾으면, 모든 디렉토리 관련 메모리 검색이 해제됩니다.

캐쉬이 있으면, 공유 디렉토리 캐쉬는 **db2start** 처리 중에 구축되고 DB2가 중단하면 해제됩니다 이 캐쉬는 모든 DB2 서버 프로세스(db2agent)에서 사용됩니다. 또한, 개인용 응용프로그램 디렉토리 캐쉬는 응용프로그램이 데이터베이스로 첫 번째 연결할 때 구축되고 응용프로그램이 종료하면 해제됩니다.

캐쉬마다 시스템 데이터베이스 디렉토리, 데이터베이스 연결 서비스 디렉토리, 노드 디렉토리의 이미지를 제공합니다. 캐쉬를 사용하면, 디렉토리 파일 입출력을 제거하고 디렉토리 검색을 최소화하므로, 연결 비용이 줄어듭니다.

캐쉬된 디렉토리가 갱신되는 경우, 변경사항이 즉시 캐쉬로 전달되지 않습니다. 디렉토리 항목이 캐쉬에 없는 경우, 원래 디렉토리에서 검색됩니다.

캐쉬을 사용하면 응용프로그램 라이프에 필요한 개인용 메모리를 증가시킵니다. 캐쉬이 없으면, 이 메모리는 디렉토리 찾아보기가 처리중일 때만 필요합니다. DB2의 공유 메모리의 전체 사용은, 데이터베이스 에이전트간에 공유된 디렉토리 정보가 공유 메모리로 이동하였으므로 다소 증가합니다. 캐쉬에 필요한 메모리 크기는 디렉토리마다 정의된 항목 수에 따라 결정됩니다.

기타 DB2 Connect 매개변수

MAXDARI와 NUMDB는 DB2 Connect 워크스테이션에 지역 데이터베이스가 없는 경우, 최소값으로 설정됩니다. 이렇게 설정하면 자원 소비를 최소화합니다.

AGENTPRI는 원격 클라이언트에만 적합합니다. AGENTPRI는 DB2 Connect 인스턴스의 에이전트에 대한 운영 체제 스케줄러에 의해 주어진 우선순위를 제어합니다. DB2 Connect 인스턴스에서는 더 높은 우선순위(더 낮은 수)가 있으면, 더

많은 CPU 주기 권한을 부여합니다. DB2 Connect 워크스테이션에서 수행중인 다른 프로세스용으로 남은 CPU 주기 수를 줄입니다. 예를 들어, AGENTPRI 값으로 같은 워크스테이션에서 수행중인 높은 우선순위 DB2 Connect 인스턴스와 낮은 우선순위 DB2 Connect 인스턴스를 가질 수 있습니다.

DB2 Connect를 통해 클라이언트 기계에서 DRDA 데이터베이스로 연결하려면 DB2 Connect 워크스테이션에서 에이전트가 수행 중이어야 합니다. DB2 Connect 워크스테이션을 통해 호스트나 AS/400 데이터베이스 서버에 액세스하는 원격 클라이언트 연결의 최대 수보다 크거나 같은 값으로 MAXAGENTS를 설정하십시오.

사용 통계 문자열 사용을 결정하는 경우, sqlesact() API를 사용하면 DB2ACCOUNT 환경 변수 스레드에 대한 성능상의 이점이 있습니다. 65 페이지의 『OS/390용 DB2 Universal Database에서의 사용 비용 청구 계정 구현』에서 자세한 내용을 참조하십시오.

조정된 SQLCODE 맵핑 파일이 필요없는 경우, 기본 SQLCODE 맵핑을 사용하거나 SQLCODE 맵핑을 작동 중지하여 성능을 향상시킬 수 있습니다. (기본 맵핑 파일은 DB2 Connect 라이브러리에 있으며 조정된 맵핑 파일은 디스크에서 읽혀져야 하므로 성능에 영향을 줍니다.) SQLCODE 맵핑에 대해서는 141 페이지의 『제11장 SQLCODE 맵핑』에서 자세한 내용을 참조하십시오.

연결 풀링

DB2 Connect Enterprise Edition 서버들은 종종 수천개의 동시 클라이언트 요청에 대해 데이터베이스 연결을 제공합니다. 데이터베이스 서버에 대한 연결의 확립 및 단절은 데이터베이스 서버와 DB2 Connect 서버 성능 모두에 역으로 영향을 미치는 매우 자원 집약적인 프로세스일 수 있습니다. 이는 웹 페이지를 각 방문할 때마다 데이터베이스 서버에 대해 새로 연결을 빌딩해서 조회를 수행하고 연결을 종료해야 하는 웹 환경에서 특히 분명합니다. 이러한 오버헤드를 줄이기 위해 DB2 Connect Enterprise Edition에서는 손쉽게 액세스할 수 있는 풀 내의 데이터베이스에 대해 열려 있는 연결들을 유지보수하기 위해 *연결 풀링*을 사용합니다.

연결 풀링의 작업 방법

연결 풀링은 DB2 Connect를 통해 호스트에 연결하는 응용프로그램에 투명합니다. 응용프로그램이 호스트로부터의 연결 해제를 요청하면, DB2 Connect는 응용프로그램과의 인바운드 연결은 제거하지만 호스트에 대한 아웃바운드 연결을 풀에 그대로 유지시킵니다. 새로운 응용프로그램이 연결을 요청하면, DB2 Connect는 기존의 풀에서 하나를 사용합니다. 이미 존재하는 연결을 사용하면 호스트에서 고가의 CPU 연결 단가 뿐만 아니라 전체적인 연결 시간도 줄일 수 있습니다.

연결 풀링을 사용하려면 다음의 APAR이 OS/390용 DB2 버전 6.1에 적용되어야 합니다.

APAR PQ33473

DB2 Connect 에이전트는 유휴 또는 활동중 상태 가운데 하나일 수 있습니다. 에이전트가 응용프로그램에 대한 작업을 실행하고 있을 때는 활동중입니다. 이 작업이 완료되면 에이전트는 동일하거나 다른 응용프로그램으로부터 또다른 작업을 대기하는 유휴 상태로 들어 갑니다. 모든 유휴 에이전트들은 유휴 에이전트 풀이라는 곳에 함께 보존됩니다. 사용자는 NUM_POOLAGENTS 구성 매개변수를 사용하여 이 풀의 크기를 구성할 수 있습니다. 이 매개변수는 시스템을 유지보수하려는 유휴 에이전트의 최대수와 같습니다. 이 매개변수를 0으로 설정하는 것은 연결 풀링 기능을 끄는 것과 같습니다.

DB2 Connect는 첫번째 클라이언트 요청을 수신하기 전에 데이터베이스에 대한 연결을 설정하지 않습니다. 그러나, 사용자가 원할 경우에는 클라이언트가 요청하기 전에 유휴 에이전트의 풀을 꽉 채울 수도 있습니다. 풀은 NUM_INITAGENTS 구성 매개변수를 사용하여 시동시 채울 수 있습니다. 이 매개변수는 작성되어야 할 유휴 에이전트의 수를 시동시 결정합니다. 이 유휴 에이전트들은 호스트 데이터베이스 서버에 초기에 연결하지 않습니다.

클라이언트가 호스트에 대한 연결을 요청할 때, DB2 Connect는 호스트 데이터베이스 서버에 대한 연결이 존재하는 풀 내의 에이전트들 중에서 하나의 에이전트를 가려오려고 시도합니다. 그 에이전트에 문제가 있으면 유휴 풀에 있는 사용 가능한 다른 에이전트를 찾으려고 시도합니다. 풀이 비어 있으면 DB2 Connect는 새로운 에이전트를 작성합니다.

MAX_COORDAGENTS 구성 매개변수를 사용하여 동시에 활동할 수 있는 에이전트의 최대 수를 제어할 수 있습니다. 이 수를 일단 초과하면 오류 sqlcode SQL1226으로 인해 연결을 새로 작성할 수 없게 됩니다. (이 코드는 동시 아웃바운드 연결의 최대수가 초과되었음을 나타냅니다.)

DB2 Connect EE와 동일한 머신에서 수행 중인 응용프로그램은, DB2 레지스트리 변수 DB2CONNECT_IN_APP_PROCESS를 사용하여, DB2 Connect가 응용프로그램 프로세스 내에서 수행되게 하거나, DB2 Connect EE 서버에 응용프로그램이 연결된 후 에이전트 내에서 호스트 연결을 수행하게 할 수 있습니다. 응용프로그램에서 연결 풀링을 사용하려면, DB2 Connect EE 서버 에이전트 내에서 호스트에 대한 연결을 확립한 후 DB2CONNECT_IN_APP_PROCESS를 NO로 설정해야 합니다.

DB2 Connect 연결 집중기

DB2 Connect의 *연결 집중기* 기법을 사용하면 DB2 Connect Enterprise Edition 서버들이 수천명의 사용자가 동시에 실행하는 비지니스 트랜잭션들에 대한 지원을 제공하면서도 S/390 호스트 또는 AS/400 데이터베이스 서버에 필요한 자원을 과감하게 줄일 수 있습니다. 보다 적은 수의 S/390 호스트 또는 AS/400 데이터베이스 서버 연결에 모든 응용프로그램으로부터의 워크로드를 집중시킴으로써 이러한 목표를 이룰 수 있습니다. 이는 위에 설명된 연결 풀링 기능과 유사한 것처럼 보이지만, 실제로 많은 볼륨의 온라인 트랜잭션 처리(OLTP) 응용프로그램에 대한 자원 소비를 줄일 수 있는 보다 정교한 접근 방법입니다.

응용프로그램을 종료함으로써 더 이상 연결이 필요하지 않게 될 때 연결 풀링으로 인해 연결 설정 단가를 절약할 수 있습니다. 즉, 한 응용프로그램에서 풀 처리된 연결을 재사용하기 전에 다른 응용프로그램은 연결해제되어야 합니다.

한편, DB2 Connect는 다른 응용프로그램이 트랜잭션을 마치자마자 연결 집중기를 통해 응용프로그램에 대한 연결을 사용가능하게 하므로 다른 응용프로그램을 연결해제하지 않아도 됩니다. 본질적으로 데이터베이스 서버 연결과 그의 연관된 호스트 및 DB2 Connect 자원들은 활동중인 트랜잭션을 가지고 있는 동안에만 응용프로그램에 의해 사용됩니다. 트랜잭션이 완료되자마자, 실행되는 트랜잭션을 필요로 하는 다른 응용프로그램에서 연결 및 연관된 자원들을 사용할 수 있습니다.

연결 집중기의 구현 방법

이전 버전의 DB2 Connect에서는 모든 활동 응용프로그램에 응용프로그램 요청 뿐만 아니라 데이터베이스 연결을 관리하는 EDU(Engine Dispatchable Unit)가 있었습니다. 이 EDU를 일반적으로 *조정자 에이전트*라고 합니다. 각 조정자 에이전트는 응용프로그램 및 EDU의 상태나 문맥을 추적합니다. 각 EDU는 연결 수가 증가하고 에이전트들간에 문맥을 전환한 결과로 오버헤드가 추가로 발생될 때는 많은 양의 메모리를 사용합니다.

위의 아키텍처에서는 연결과 EDU간에 일대일 대응 관계가 성립됩니다. 그러나 연결 집중기는 연결과 EDU간에 다대일 대응 관계를 허용합니다. 즉, 이제는 연결 (X)와 EDU (Y)가 X >= Y 관계에 있습니다.

연결 집중기는 에이전트를 두 개의 엔터티 즉, *논리 에이전트*와 *작업자 에이전트*로 분할합니다. 논리 에이전트는 특정 EDU에 대한 참조없이 응용프로그램만을 나타냅니다. 논리 에이전트에는 응용프로그램에 필요한 모든 정보와 제어 블록이 들어 있습니다. 서버에 *n* 개의 응용프로그램이 연결되어 있다면 서버에는 *n* 개의 논리 에이전트가 있습니다. 작업자 에이전트는 응용프로그램 요청을 실행하는 물리적인 EDU이지만 주어진 응용프로그램에 대해 영구적으로 접속하지 않습니다. 작업자 에이전트는 논리 에이전트와 연관하여 트랜잭션을 수행하고 트랜잭션 경계에서 연관을 종료한 후 사용 가능한 풀로 리턴합니다.

*논리 에이전트 스케줄러*로 알려진 엔터티는 작업자 에이전트를 논리 에이전트에 할당합니다. 특정 컴퓨팅 플랫폼에서 열린 파일 핸들 수를 제한하면 논리 에이전트의 수가 파일 핸들 한계를 초과할 때 둘 이상의 스케줄러 인스턴스로 귀착될 수도 있습니다.

집중기 활성화

연결 집중기를 사용하려면 다음의 APAR이 OS/390용 DB2 버전 6.1에 적용되어야 합니다.

```
APAR PQ33473
```

데이터베이스 관리 프로그램 구성 매개변수 MAX_LOGICAGENTS는 논리 에이전트의 최대수를 설정합니다. MAX_LOGICAGENTS 값을 기본값보다 큰 수로 설정하여 집중기 기능을 활성화할 수 있습니다. MAX_LOGICAGENTS의 기본값은 MAX_COORDAGENTS 값과 같습니다. 각 응용프로그램에는 하나의 논리 에

이전트가 있기 때문에, MAX_LOGICAGENTS가 실제로 데이터베이스 인스턴스에 연결될 수 있는 응용프로그램의 수를 제어하는 반면에, MAX_COORDAGENTS는 아무 때나 활동할 수 있는 인바운드 연결의 수를 제어합니다. MAX_LOGICAGENTS의 범위는 MAX_COORDAGENTS에서 최대 64,000 사이의 숫자입니다. 논리 에이전트의 최대수는 MAX_COORDAGENTS와 같습니다.

기존의 여러 구성 매개변수들은 에이전트를 구성하는 데 사용됩니다. 이 매개변수들은 다음과 같습니다.

MAXAGENTS
: 작업자 에이전트의 최대수.

MAX_COORDAGENTS
: 활동중인 조정자 에이전트의 최대수.

NUM_POOLAGENTS
: 에이전트 풀 크기. 에이전트 풀에는 비활동 에이전트 및 유휴 에이전트가 포함됩니다.

NUM_INITAGENTS
: 풀에 있는 작업자 에이전트의 초기수. 이것이 유휴 에이전트가 됩니다.

XA 트랜잭션 지원

연결 집중기의 아키텍처를 통해 DB2 Connect는 OS/390용 DB2 및 AS/400용 DB2에 긴밀 결합 XA 트랜잭션 지원을 제공할 수 있습니다. 집중기는 다른 트랜잭션에 대해서처럼 특정 XA 트랜잭션(단일 XID)과 작업자 에이전트를 연관시킵니다. 그러나, XA 트랜잭션이 `xa_end()`(분기 경계)로 끝나면, 작업자 에이전트가 그 자신을 일반 풀로 해제시키지 않습니다. 대신, 작업자는 그 특정 XA 트랜잭션과 연관된 상태로 남아 있습니다. 다른 응용프로그램이 동일한 XA 트랜잭션을 조인할 때, 작업자 에이전트가 해당 응용프로그램에 첨부됩니다.

트랜잭션 경계 호출은 모두 에이전트를 풀로 리턴시킵니다. 인스턴스의 경우에 읽기 전용인 `xa_prepare()`, `xa_rollback()`, `xa_recover()`, `xa_forget()`, `xa_commit()` 또는 구간 복원을 일으키는 XA 오류는 에이전트를 보통 풀로 리턴시킵니다. `Xa_end()`는 트랜잭션 분기만을 종료시키며 XID와 연관된 것을 종료시키기에는 충분치 않습니다.

예

1. 4,000개 이상의 동시 연결이 필요한 환경을 고려해 봅시다. CGI 응용프로그램을 사용하는 웹 서버 또는 수많은 데스크탑 사용자가 있는 오피스 시스템에서는 둘다 이러한 요구사항을 초과할 수 있습니다. 이러한 경우에는 효율성을 위해 일반적으로 DB2 Connect가 독립형 게이트웨이로서 작동해야 합니다. 즉, 데이터베이스와 DB2 Connect 시스템이 서로 분리된 머신상에 있어야 합니다.

   DB2 Connect 서버 시스템은 데이터베이스 머신에 대해 4,000개의 동시 개방 연결을 유지보수하지 못할 수도 있습니다. 대부분의 경우에, 주어진 순간에 발생하는 트랜잭션수는 동시 연결 수보다 상당히 적을 것입니다. 그러면, 시스템 관리자는 데이터베이스 구성 매개변수를 다음과 같이 설정하여 시스템의 효능을 최대화할 수 있습니다.

   ```
   MAX_LOGICAGENTS =  4,000
   MAX_AGENTS      =  1,000
   MAX_COORDAGENTS =  1,000
   NUM_POOLAGENTS  =  1,000
   ```

   집중기는 게이트웨이가 한번에 1,000개의 트랜잭션만 관리한다 할지라도 최대 4,000개의 동시 세션을 열어 놓습니다.

2. 위의 예에서 작업자 에이전트는 논리 에이전트에 대한 관계를 끊임없이 연결하고 해제합니다. 유휴 상태가 아닌 그러한 에이전트들은 특정 트랜잭션에 참여하지 않는 데이터베이스에 대한 연결을 유지보수하지 않을 수도 있기 때문에, 이 에이전트들은 연결을 요청하는 논리 에이전트(응용프로그램)에 대해 사용 가능합니다.

   XA 트랜잭션의 경우에는 약간 다릅니다. 이 예에서는 TP 모니터가 DB2 Connect 게이트웨이 및 OS/390 또는 AS/400 데이터베이스와 함께 사용된다고 가정할 수 있습니다. 응용프로그램에서 연결을 요청하면, 집중기는 비활동 에이전트를 인수하여 해당 요청을 처리하거나 새로운 작업자 에이전트를 작성합니다. 응용프로그램에서 XA 트랜잭션을 요청한다고 가정합시다. 이 트랜잭션에 대해 XID가 작성되며 작업자 에이전트가 이에 연관됩니다.

   응용프로그램의 요청이 서비스될 때 이는 `xa_end()`를 발행하여 작업자 에이전트로부터 접속 해제합니다. 작업자 에이전트는 트랜잭션의 XID와 연관된 상태로 남아 있습니다. 이제 이는 연관된 XID를 가진 트랜잭션에 대한 요청만 서비스할 수 있습니다.

이때, 또다른 응용프로그램에서 비XA 트랜잭션을 요청할 수도 있습니다. 사용 가능한 다른 작업자 에이전트가 없을지라도 XID와 연관된 에이전트를 두 번째 응용프로그램에 대해 사용할 수 없습니다. 이 에이전트는 활동중인 것으로 간주됩니다. 두 번째 응용프로그램은 그에 맞게 작성된 새로운 작업자 에이전트를 가집니다. 그 두 번째 응용프로그램이 해당 트랜잭션을 완료하고 나면, 작업자 에이전트가 사용 가능한 풀로 해제됩니다.

한편, 첫번째 에이전트의 XID와 연관된 트랜잭션을 요청하는 다른 응용프로그램들은 그 에이전트에 접속 및 접속 해제될 수도 있으며, 그 에이전트는 그 응용프로그램들에 대해 전용 XA 트랜잭션을 실행합니다. 특정 트랜잭션을 요청하는 응용프로그램은 에이전트가 자유 상태인 경우에 이 작업자 에이전트로 전송됩니다.

작업자 에이전트는 응용프로그램에서 트랜잭션 경계 호출(`xa_end()`가 아님)을 발행할 때까지 일반 풀로 다시 되돌려 해제되지 않습니다. 인스턴스 경우에는 응용프로그램이 트랜잭션을 `xa_commit()`로 끝내게 되며 이 지점에서 작업자 에이전트는 XID와의 관계를 제거하고 사용 가능 풀로 리턴합니다. 이 때, 응용프로그램을 요청하면 또다른 XA 또는 비XA 트랜잭션용으로 이를 사용할 수 있습니다.

제한사항

게이트웨이 집중기를 사용하는 데는 여러 가지 중요한 제한사항이 있습니다. 시스템에서 연결 집중기를 사용하려고 시도하기 전에 다음 내용을 모두 읽어 보십시오.

- 연결 집중기는 DB2 버전 7 이상 클라이언트에서만 사용할 수 있습니다.
- 집중기는 OS/390용 DB2 또는 AS/400용 DB2 호스트만 지원합니다.
- 집중기는 지역 및 원격 클라이언트로부터 인바운드 연결을 설정하는 데 TCP/IP 프로토콜을 따릅니다. TCP/IP 또는 지역(IPC)을 사용하는 인바운드 연결만이 풀처리된 아웃바운드 연결을 이용할 수 있습니다. 집중기는 SNA와 같은 다른 통신 프로토콜을 통한 연결도 허용하지만, 그러한 연결에는 XA 집중 기능을 사용할 수 없습니다.
- 게이트웨이에서 집중기가 사용 가능한 경우에는 클라이언트 응용프로그램에서 정적 `SET`문을 사용하지 말아야 합니다. 정적 `SET`을 사용하는 경우에는 DB2가 오

류를 리턴하지만, 사용자의 응용프로그램 및 동일한 아웃바운드 연결을 공유하는 다른 응용프로그램들에게 역으로 영향이 미칠 수 있습니다.

- SET문의 경우에는 즉시 실행만 지원됩니다.
- 전역 임시 테이블을 선언하는 경우에는 트랜잭션 또는 분기 경계에서 이들을 명확하게 닫아야 합니다. 테이블을 닫는 데 실패하면 나중의 트랜잭션 동안에 오류가 발생할 수도 있습니다.
- XA 긴밀 결합 트랜잭션 지원의 경우에는 동일한 XA 트랜잭션에 참여하는 모든 응용프로그램들이 동일한 게이트웨이를 사용하여 호스트에 연결해야 합니다.
- 확약 전에 선언된 모든 보류(withhold) 커서를 닫는 응용프로그램만이 집중기 이점을 얻을 수 있습니다. 보류 커서를 닫지 않은 트랜잭션들은 계속 진행되기는 하지만, 전용 작업자 에이전트를 할당받게 되고 이로 인해 집중기의 전체 기능 세트를 사용할 수 없게 됩니다.
- 동일한 XA 트랜잭션에 참여하는 모든 응용프로그램들은 동일한 CCSID를 갖고 동일한 사용자 ID를 사용하여 연결을 해야 합니다.
- 아웃바운드 연결이 2단계 연결을 지원할 목적으로 설정된 경우에는 그 연결의 에이전트는 2단계 연결을 지원하는 데만 사용될 수 있습니다. 마찬가지로 1단계 연결을 지원할 목적으로 설정된 에이전트는 1단계 연결만 지원할 수 있습니다.
- 인증 유형 DCS_ENCRYPT는 버전 7에서 집중기에 대해 작업하지 않습니다.
- 집중기는 명령행 인터페이스에서 동적 SQL만 지원합니다. Embedded 동적 SQL 응용프로그램으로부터의 동적 PREPARE 요청은 거부됩니다. 정적 SQL을 사용하거나 동적 SQL문에 대해 CLI를 사용하도록 응용프로그램을 변경해야 합니다.

데이터베이스 조정

시스템 성능은 호스트나 AS/400 데이터베이스 서버 데이터베이스의 성능에 따라 영향을 받습니다.

다른 데이터베이스 관리 시스템에는 서로 다른 성능 기능이 있습니다. 다른 시스템의 SQL 최적화 알고리즘은, 예를 들어, 같은 응용프로그램을 가지고 다르게 작동할 수 있습니다. 자세한 내용은 호스트나 AS/400 데이터베이스 서버 시스템 성능 관련 문서를 참조하십시오.

AS/400용 DB2 Universal Database에서, 저널링되는 것을 피하는 미확약 읽기(UR) 또는 미확약(NC) 바인드 옵션을 사용하여 성능을 향상시킬 수도 있습니다.

주: UR을 사용하면, 블로킹이 ALL로 설정되어 있는 경우에만 저널링되지 않은 데이터를 읽을 수 있으며 이 때에도 갱신은 가능하지 않습니다.

응용프로그램 서버와 이 응용프로그램 서버가 제공하는 잠금 세분성에 따라, 조회 또는 응용프로그램에 사용되는 분리 레벨이 성능에 심각한 영향을 줄 수도 있습니다.

데이터베이스에는 적합한 표준화의 레벨, 효율적인 색인 사용, 데이터베이스 공간의 적당한 할당이 있어야 합니다. 성능은 다음 섹션에 설명된 사용하는 데이터 유형에 의해 영향을 받을 수 있습니다.

OS/390용 DB2 조정

TCP/IP 지원을 사용하려면 최소 OS/390 V1R3 이상이어야 합니다. OS/390 V2R5 이상을 사용하는 것이 가장 바람직합니다.

분산 데이터 기능(DDF)이 분산 응용프로그램을 OS/390용 DB2에 연결하는 것을 담당합니다. DDF가 응용프로그램 서버(AS)로 설정되어야 합니다. 이를 수행하려면, 원격 시스템의 LU 이름을 SYSIBM.LUNAMES 테이블에 삽입하거나 LUNAME, SYSMODENAME, USERSECURITY, ENCRYPTPSWDS, MODESELECT 및 USERNAMES 값을 SYSIBM.SYSLUNAME 테이블에 삽입하십시오. 그런 다음, 부트 스트랩 데이터 세트(BSDS)에 대해 DDF 갱신을 수행하십시오. 예를 들면, 다음과 같습니다.

```
DDF LOCATION=LOC1,LUNAME=LU1,PORT=8000,RESPORT=8001
```

최상의 성능을 발휘하게 하려면, 권장 DDF 주소 공간 prioritisation(COMPAT 모드에 있을 경우, DBM1보다 약간 낮거나 같은 공간)을 사용해야 합니다. VLF에서 RACF 권한 부여 캐싱을 사용할 수 있으면, 가능한 V5 패키지 권한 부여 캐싱을 사용하십시오. 대부분의 조작에는 CACHEPAC=32768 값이면 충분합니다.

DDF는 VTAM에 연결을 시도하므로, DDF가 시작될 때 VTAM이 사용 중이어야 합니다. 다음은 샘플 VTAM APPL 정의입니다.

```
SYD51TC* APPL  AUTH=(ACQ),                                  X
               PARSESS=YES,                                    X
               HAVAIL=YES,                                     X
               EAS=1600,                                       X
               APPC=YES,                                       X
               DSESLIM=1024,                                   X
               DMINWNL=512,                                    X
               DMINWNR=512,                                    X
               AUTOSES=1,                                      X
               SECACPT=ALREADYV,                               X
               SRBEXIT=YES,                                    X
               SYNCLVL=SYNCPT,                                 X
               MODETAB=DB2MODET,                               X
               VPACING=63                                      X
```

OS/390에서 처리되는 비활성 스레드를 최적화할 수 있습니다. V3에서는 최대 10,000개의 클라이언트를 동시에 연결할 수 있으며 V4와 V5에서는 최대 25,000개를 동시에 연결할 수 있습니다. 그러나 모든 경우에 동시에 연결할 수 있는 최대 수는 1999입니다. 각 워크스테이션 클라이언트는 비활성일 때에도 연결된 상태로 남아 있을 수 있으며 이의 스레드는 각 확약의 비활성 체인에 배치됩니다.

DSNZPARM 매개변수 CMTSTAT, CONDBAT 및 MAXDBAT는 스레드 처리에 영향을 미칩니다. 최적의 성능을 발휘하려면, CMTSTAT를 INACTIVE로 설정하고 CONDBAT를 양호한 성능을 발휘하는 연결 DBAT의 최대수로 설정하고 MAXDBAT를 활동중인 DBAT의 수용 가능한 최대수로 설정하십시오.

VTAM 구성 등 DRDA 네트워크에서 OS/390용 DB2를 연결하는 데 대한 자세한 내용은 *연결성 보충 설명서*에서 참조하십시오.

데이터 변환

데이터가 환경들 간에 서로 전송되는 경우, 변환될 필요가 있습니다. 이 변환은 성능에 영향을 줍니다.

다음 플랫폼을 고려하십시오.

- Intel(OS/2, Windows NT 또는 Windows 2000)
- IEEE(UNIX 방식의 시스템)
- System/370과 System/390(MVS, OS/390, VM, VSE)
- OS/400.

그리고 숫자 데이터 유형을 고려하십시오.

- Packed 십진수
- Zoned 십진수
- 정수
- 부동 소수점 수

표8은 변환이 일어나는 시기를 보여줍니다.

표 8. 데이터 변환

	Intel	**IEEE**	**S/370 & S/390**	**OS/400**
Packed 십진수 데이터				
Intel	아니오	아니오	아니오	아니오
IEEE	아니오	아니오	아니오	아니오
S/370/390	아니오	아니오	아니오	아니오
OS/400	아니오	아니오	아니오	아니오
Zoned 십진수 데이터				
Intel	아니오	아니오	예	예
IEEE	아니오	아니오	예	예
S/370/390	예	예	아니오	아니오
OS/400	예	예	아니오	아니오
정수 데이터				
Intel	아니오	예	예	예
IEEE	예	아니오	아니오	아니오
S/370/390	예	아니오	아니오	아니오
OS/400	예	아니오	아니오	아니오
부동 소수점 수 데이터				
Intel	아니오	예	예	예
IEEE	예	아니오	예	아니오
S/370/390	예	예	아니오	예
OS/400	예	아니오	예	아니오

1바이트 문자 데이터 변환에 드는 CPU 비용은 대체적으로 숫자 데이터 변환(데이터 변환이 필요한)에 드는 비용보다 적습니다.

DATE/TIME/TIMESTAMP의 데이터 변환 비용은 1바이트 CHAR 변환 비용과 거의 같습니다. 부동 소수점 수 데이터 변환의 비용이 가장 많이 듭니다. 응용프로그램 설계자는 DB2 Connect 방식의 응용프로그램을 설계할 때 이런 요인을 이용할 수 있습니다.

데이터베이스 테이블에 '2진 데이터용'으로 정의된 컬럼이 있으면, 응용프로그램과 데이터베이스 간에 전송 중인 문자 데이터는 다른 데이터 변환이 필요치 않습니다. 이는 호스트나 AS/400 데이터베이스 서버에 데이터를 아카이브할 경우에 사용할 수 있습니다.

문자 데이터용 데이터 유형

문자 데이터는 CHAR 또는 VARCHAR 데이터 유형을 가질 수 있습니다. 더 효율적인 데이터 유형은 필드에 일반적인 데이터의 길이에 따라 다릅니다.

- 실제 데이터 크기가 크게 변하는 경우, VARCHAR가 더 효율적인데 그 이유는 CHAR는 필드를 채우기 위해 여분의 공백 문자를 추가하기 때문입니다. 이 공백 문자는 다른 문자처럼 네트워크를 통해 전송되어야 합니다.
- 실제 데이터 크기가 별로 변하지 않는 경우, CHAR가 더 효율적인데 그 이유는 VARCHAR 필드마다 전송되어야 하는 일정 바이트 길이의 정보가 조금씩 더 있기 때문입니다.

네트워크 조정

분산 데이터베이스 환경에서 전반적인 성능을 향상하는 가장 좋은 방법은 네트워크에서 지연을 제거하는 것입니다. 네트워크 관리자들은 보통 전송시 가능한 많은 데이터가 수집될 경우 네트워크가 좀 더 효율적이 되는 것으로 간주합니다. 그러나 이러한 접근 방식은 네트워크에 지연을 축적하는 분산 데이터베이스와 같은 응용프로그램에는 적합하지 않습니다. 일반 사용자는 지연 때문에 네트워크의 효율성을 깨닫지 못합니다.

대부분의 네트워크 장치는 지연 매개변수를 가지고 있으며 이들 대부분은 분산 데이터베이스에는 좋지 않은 값으로 기본 설정되어 있습니다. 성능을 개선하려면, 이들 매개변수를 찾아 가능하면 이들을 0으로 설정해야 합니다. 또한, 장치의 버퍼 크기가 데이터 유실로 인한 재전송을 방지할 만큼 충분한 공간이 되도록 해야 합니다. 예를 들어, UNIX 시스템은 보통 전송 또는 수신 대기행렬 깊이에 대해 32

의 기본 값을 갖습니다. 결과를 더 좋게 하려면, 대기행렬 깊이를 150으로 설정하십시오. DLC 설정에서 이에 상응하는 매개변수는 수신 깊이로, 이것 또한 150이 되어야 합니다.

대부분의 사이트에서 IOBUF 매개변수가 너무 낮게 설정되어 있습니다. 이는 보통 500으로 설정되어 있지만, 사용해 본 바에 의하면 특히 ESCON이나 3172와 같은 채널 연결에서 다량의 데이터를 이동할 경우 3992 값이 최상의 성능을 나타냅니다.

SNA 연결의 경우, 워크스테이션의 모드 프로파일을 63으로 설정해야 합니다. 일반적으로 네트워크 전반의 수신 페이싱 값을 가장 높은 값으로 설정해야 하므로 DB2 APPL 명령문의 VPACING과 PACING 매개변수, 스위치 기본 모드에서의 워크스테이션에 대한 PU/LU도 63으로 설정해야 합니다. 이와 같이 하면 송신자가 응답을 기다리기까지 메시지 흐름의 양이 점차 증가하게 됩니다.

LAN 시스템에서, DLC 또는 LLC 전송 및 수신 창의 *크기*가 성능에 커다란 영향을 미칠 수 있습니다. 전송 값을 7 이상으로 설정해야 하고 대다수 구성에서는 수신 값이 4 이하이면 최적으로 작동합니다.

이더넷을 수행하는 경우, TCP 세그먼트 *크기*를 1500 바이트로 설정해야 합니다. 토큰 링이나 FDDI 네트워크에서, 이 값은 4400 바이트가 되어야 하며 TCP/IP에 ESCON 어댑터를 사용하는 경우 세그먼트 *크기*는 반드시 4096이 되어야 합니다.

마지막으로 TCP/IP 네트워크의 경우, TCP 전송 및 수신 버퍼 *크기*는 32768 이상으로 설정해야 합니다. 보통 65536 값이 가장 좋습니다.

주: 게이트웨이에서 서버로의 연결(아웃바운드 연결)을 설정하는 것은 클라이언트에서 게이트웨이로의 연결(인바운드 연결)을 설정하는 것보다 훨씬 더 비용이 많이 듭니다. 수 천개의 클라이언트가 게이트웨이를 통해 서버와 연결 및 연결 해제를 반복하는 환경에서는 처리 시간의 상당 부분이 아웃바운드 연결을 설정하는 데 사용됩니다. DB2 Connect는 TCP/IP에서 연결 풀링을 제공합니다. 클라이언트가 서버로부터의 연결 해제를 요청하면, 게이트웨이는 클라이언트와의 인바운드 연결은 해제하지만 서버에 대한 아웃바운드 연결을 풀에 그대로 유지시킵니다. 새로운 클라이언트가 게이트웨이에서 연결을 요청하면, 게

이트웨이는 풀에서 기존의 연결을 제공하므로 전반적인 연결 시간이 줄어 들고 서버에서의 CPU 연결 비용이 절약됩니다.

*관리 안내서*에서 DB2에서의 연결 풀링에 대한 자세한 내용을 참조하십시오.

네트워크 성능 조정 방법이 다음 표에 요약되어 있습니다.

확인 대상	예	설정	주의사항
고의 지연	네트워크 장치의 지연 매개변수	0으로 설정	보통 기본 값이 가장 높습니다.
버퍼	IOBUF 매개변수	최대 3992로 설정	ESCON이나 기타 채널 어댑터에 특히 유용
	RUSIZE	최적의 *크기*는 4096입니다.	RUSIZE와 RQRIOBLK를 같은 *크기*로 설정하면 최상의 성능을 얻을 수 있습니다.
	페이싱	VPACING, PACING 및 모드 프로파일을 63으로 설정해야 합니다.	가능할 경우 적응 페이싱을 사용하십시오.
어댑터 설정	전송/수신 대기행렬 깊이	권장 값은 150입니다.	기본 값은 보통 32입니다.
	SNA의 DLC 창	전송 창 *크기*를 높게 (>7) 설정하십시오. 수신 창 *크기*를 낮게(예: 1) 설정한 후, 이 값을 점차 증가시키면서 최적의 값을 찾으십시오.	모든 논리 장치가 지연을 가중시킵니다. 가능한한 네트워크 토폴로지를 단순하게 하십시오.
TCP 설정	세그먼트 *크기*	이더넷의 경우 1500, 토큰 링 및 FDDI의 경우 4400	TCP/IP에 사용되는 ESCON 어댑터는 반드시 4096으로 설정되어야 합니다.
	송신/수신 공간 *크기*	두 경우 모두 64K여야 합니다.	기본 값은 Windows의 경우 8192입니다. Windows 레지스트리에서 설정할 수 있습니다.

네트워크 하드웨어

다음은 하드웨어와 관련이 있는 내용입니다.

- 네트워크 또는 전송 미디어 속도

  전송 미디어 속도가 빠를수록 성능이 향상됩니다. 예를 들어, 다음은 일반적인 원시 데이터 전송률입니다.

  채널에서 채널로(광섬유)
  4.0 MB/s

  16Mbps LAN
  2.0 MB/s

  채널에서 채널로(정규)
  1.0 MB/s

  4Mbps LAN
  0.5 MB/s

  고속 T1 반송자(1.544Mbps)
  0.193 MB/s

  고속 원격 56Kbps 전화선
  0.007 MB/s

  19.6Kbps 모뎀
  0.002 MB/s

  9600bps 모뎀
  0.001 MB/s

  데이터 전송률은 호스트나 AS/400 데이터베이스 서버에 대한 경로에서 가장 속도가 느린 전송 미디어에 의해 제한됩니다.

- 네트워크 어댑터 또는 통신 제어기

  네트워크 어댑터와 통신 제어기의 메모리 사용을 주의깊게 계획해야 합니다. 추가로, 네트워크 전문가와 함께 제어기가 DB2 Connect가 생성한 여분의 통신량을 조절할 수 있는지 확인해야 합니다.

- 네트워크 토폴로지

데이터가 LAN에서 LAN으로, SNA 네트워크에서 SNA 네트워크로 이동할 경우, 이동 시간을 고려하십시오. 브릿지, 라우터, 게이트웨이는 경과 시간에 추가될 것입니다. 예를 들어, 이동한 브릿지 수를 줄이면 요청마다 필요한 홉 수가 줄어들게 됩니다.

노드 사이의 물리적 거리도 또한 생각해야 합니다. 메시지가 위성으로 전송되더라도, 전송 시간은 빛의 속도(3 * 10**8 m/s)와 송신자와 수신자간의 왕복 거리에 의해 제한됩니다.

- 네트워크 통신량

  네트워크 대역폭 전체를 사용하는 경우, 응용프로그램에 대한 응답 시간과 데이터 전송률은 낮아집니다.

  네트워크의 특정 부분, 예를 들어, 아주 작은 버퍼 크기를 가진 오래된 NCP 같은 곳에 데이터를 축적할 때 과다 전송이 발생할 수 있습니다.

- 네트워크 신뢰성

  네트워크의 오류율이 높으면, 네트워크의 처리량은 감소하여 데이터 재전송으로 인해 성능이 저하됩니다.

시스템 자원에 대한 경합

시스템의 많은 타스크가 시스템 자원에 대해 경합하면, 성능이 저하될 수 있습니다. 다음 질문을 생각하십시오.

- CPU는 포화 상태입니까? 처리 오버헤드를 줄이기 위해 시스템을 업그레이드하고, 시스템 워크로드를 줄이며 시스템을 조정하는 것을 생각하십시오.
- 메모리는 과잉확약 되었습니까? 메모리 작업 세트를 줄이기 위해 메모리를 업그레이드하고, 시스템 워크로드를 줄이며, 시스템 조정을 생각하십시오.
- 통신 어댑터/통신 제어기를 너무 많이 사용하고 있습니까? 네트워크를 업그레이드하고 토큰링 카드를 쌍으로 만드는 것을 생각하십시오.
- 서브시스템 중 한 시스템을 너무 많이 사용하는데, 이 서브시스템이 데이터 경로에 있습니까?
- 시스템에서 수행중인 불필요한 프로세스나 타스크가 있습니까? 시스템 자원을 낭비하기 때문에 규칙적으로 사용하지 않을 경우 서비스를 구성하거나 시작하지 말아야 하는 것이 일반적입니다.

- 프로세스나 타스크가 대부분의 자원을 사용합니까? 중단될 수 있습니까? 우선 순위를 낮출 수 있습니까? 세분되면 많은 자원을 사용하지 않습니까?

성능 문제점 해결

DB2 Connect 사용자가 호스트나 AS/400 서버의 대형 조회를 수행하는 동안 응답 시간이 길어지면, 다음 부분이 성능 문제의 원인이 될 수 있는지 검토해야 합니다.

1. 호스트나 AS/400 데이터베이스 서버로부터 대형 블록이 리턴되는 조회의 경우 (32K나 그 이상의 데이터), 데이터베이스 관리 프로그램 구성 매개변수 PQRIOBLK가 32767로 설정되었는지 확인하십시오. 명령행 처리기(CLP)를 사용하면 다음처럼 실행될 수 있습니다.

   ```
   db2 update database manager configuration using RQRIOBLK 32767
   ```

2. VTAM이 호스트나 AS/400 서버와의 연결에 사용되면, "스위치된 주요 노드" 구성에서 PACING 매개변수 값을 살펴보십시오. DB2 Connect 워크스테이션에서, IBMRDB 모드 정의용 "LU 6.2 모드 프로파일"의 통신 설정을 검색하십시오. 이 정의에서, "페이싱 창 수신" 매개변수 값이 VTAM에서 정의된 PACING 값과 같은지 아니면 작은지를 확인하십시오. DB2 Connect 워크스테이션의 "페이싱 창 수신"용 값과 VTAM의 "PACING"의 공통된 값은 8입니다.
3. IBMROB 모드 정의에서 정의된 RU의 최대 *크기*가 적합한 값으로 설정되었는지 확인하십시오. 토큰링 하드웨어를 사용하는 연결용으로 적어도 4K를 권장합니다. 하드웨어를 사용하는 연결용의 경우, 이더넷 프레임의 최대 *크기*는 1536바이트임을 유의하십시오. 이 *크기*는 제한요인이 됩니다.
4. 사용자 환경의 VTAM 관리자에게 문의하여 DB2 Connect 워크스테이션의 LU-LU 세션에서 VTAM이 "적응형 페이싱"을 사용하는지 확인하십시오.

추가 SNA 성능 조정 힌트와 추가 정보

여기에는 DB2 Connect 사용에 대한 추가 SNA 성능 조정 힌트와 추가 정보가 들어 있습니다.

DB2 Connect에 대한 일반적인 성능 정보

DB2 Connect의 성능 특성은 처리기를 두드러지게 사용하는 것과 매우 소량의 입출력을 수행한다는 것입니다. 일반적으로 처리기의 속도가 더 빠를 수록 DB2 Connect도 더 빨리 수행됩니다. DB2 Connect는 SMP 처리기 구성을 최대한 활용합니다.

고속 DB2 Connect Enterprise Edition 서버는 호스트나 AS/400 데이터베이스 서버에서 클라이언트 시간, 네트워크 시간 및 처리 시간을 계산하지 않고 SQL 요청/응답을 5 밀리초 내에 처리할 수 있습니다. 소수의 데이터 행이 있는 간단한 SQL문이나 조회는 0.1초 내에 끝에서 끝까지(클라이언트에서 호스트나 AS/400 데이터베이스 서버 또는 그 반대로) 완료될 수 있습니다.

조회에 4개 또는 5개 이상의 SQL문이 있을 때, 저장 프로시듀어를 사용하면 높은 OLTP 성능을 보장하고 SQL문들 사이의 네트워크 지연으로 인한 잠금 경합 증가를 피하는 데 도움이 됩니다.

성능 문제는 대개 사용 중인 호스트 접속 유형, 네트워크 라우트와 조정 특성 및 응용프로그램 설계에 의해 유발됩니다. 일부 일반적인 DB2 Connect 성능 정보는 177 페이지의 『기타 DB2 Connect 성능 정보 소스』에 있습니다.

네트워크 접속 장치 선택과 조정

DB2 Connect를 사용할 때 최고의 성능을 발휘하도록 하는 다음과 같은 여러 가지 네트워크 접속 장치가 있습니다.

1. 채널 접속 카드
2. IBM 3172 모델 3 또는 새로운 모델이나 이에 상당하는 모델
3. IBM 2216
4. 오픈 시스템 어댑터 카드(OSA-1이 아닌 OSA-2)
5. NCP(Network Control Program)가 있는 IBM 3745
6. IBM 3174 터미널 제어기 또는 이에 상당하는 것

마지막 옵션은 사용하지 않는 것이 좋습니다. 아래를 참조하십시오.

호스트에 연결하기 위한 바람직한 최선의 방법은 AIX, Windows NT용 또는 Windows 2000용 ESCON 채널 접속 카드를 사용하는 것입니다. IBM 3172 모델 3과 2216도 성능이 우수하지만, ESCON에 비하면 처리량이 뒤쳐집니다.

ESCON 카드가 있는 AIX를 사용할 때, MPC(Multi Path Channel)와 관련된 PTF를 사용하십시오. 이 PTF가 없으면 AIX SNA ESCON 드라이버가 뒤떨어진 성능을 제공할 것입니다. 178 페이지의 『ESCON을 통한 SNA의 다중 경로 채널 지원』에서 자세한 내용을 참조하십시오. 웹 사이트 http://www.networking.ibm.com.cms/cmsnew01.html에서도 자세한 내용이 있습니다.

DB2 Connect 성능을 최적화하기 위해 조정해야 할 통신 서버, NCP 및 VTAM 매개변수의 점검목록을 보려면 178 페이지의 『NCP를 통한 DB2 Connect 연결 조정 방법』에서 자세한 내용을 참조하십시오. NCP 이외의 모든 특정 권장 사항은 모든 유형의 DB2 Connect와 클라이언트/서버 접속에 적용될 수 있습니다.

System/390의 OSA-2 카드는 더 낮은 초당 프레임 수 기능으로 인해 대량의 소규모 트랜잭션이 있을 때는 3272 Model 3만큼 높은 처리량을 제공하지는 못합니다. 최근에 개선된 사항에 대해서는 182 페이지의 『OSA-2 향상에 관한 정보』에서 자세한 내용을 참조하십시오.

NCP가 있는 3145는 대개 특히 기존 네트워크 통신량에 대해 조정됩니다. 따라서 데이터베이스 클라이언트/서버 응용프로그램에 대해서도 마찬가지로 잘 수행되지는 않습니다. 대부분의 DB2 Connect 성능 문제는 NCP와 VTAM 간 또는 NCP 간의 시간 지연에 기인합니다. 조정 점검목록을 제공하는 178 페이지의 『NCP를 통한 DB2 Connect 연결 조정 방법』에서 자세한 내용을 참조하십시오.

일반적으로 256 바이트의 3174 터미널 제어기 패킷 크기(RU크기)는 너무 작으므로 이 제어기를 사용하지 않는 것이 좋습니다. APPC 데이터베이스 연결에 독립 LU 지원을 제공하기 위해서는 3174 마이크로코드 레벨 C가 필요합니다. 이와 동등한 일부 OEM 3174 제품에도 유사한 종속성이 있을 수 있습니다.

기타 DB2 Connect 성능 정보 소스

- DB2 Technical Library 웹 사이트 http://www.ibm.com/software/data/db2/library를 검색하십시오. World Wide Web에서 DB2 Connect 고려 사항에 대한 절

이 있는 최신 정보를 찾으려면 키워드 "DB2CONNECT"와 "Performance"를 사용하여 "Technotes"에 대한 DB2 Universal Database 라이브러리를 검색하십시오.

- 또한 http://www.ibm.com/software/data/performance에는 DB2 Connect와 관련된 많은 DB2 클라이언트/서버 성능 보고서가 있습니다.

ESCON을 통한 SNA의 다중 경로 채널 지원

ESCON을 통한 SNA의 MPC(Multi Path Channel) 지원은 IBM eNetwork 통신 서버를 수행하는 시스템에서 ESCON 어댑터를 사용하여 호스트에 대한 MPC 링크 스테이션을 작성할 수 있게 합니다. MPC는 다음 이유 때문에 보통 CDLC보다 더 빠릅니다.

1. MPC는 읽기와 쓰기를 위해 별개의 서브채널을 사용합니다.
2. MPC는 IOBUF 크기에 제한을 받지 않습니다. 프레임은 4K이며 함께 제한될 수 있습니다.

테스트 결과 IOBUF 크기가 1K 미만인 ESCON CDLC(Channel Data Link Control) 링크에 비해 MPC 링크에서 세 배의 성능 향상이 나타났습니다. AIX SNA MPC에는 ESCON과 MVS VTAM V4R4 이상과 AIX용 통신 서버(5765-652)의 기능 코드 4024가 필요합니다. Windows NT 시스템에서는 Windows NT용 IBM eNetwork 통신 서버 버전 6을 사용해야 합니다.

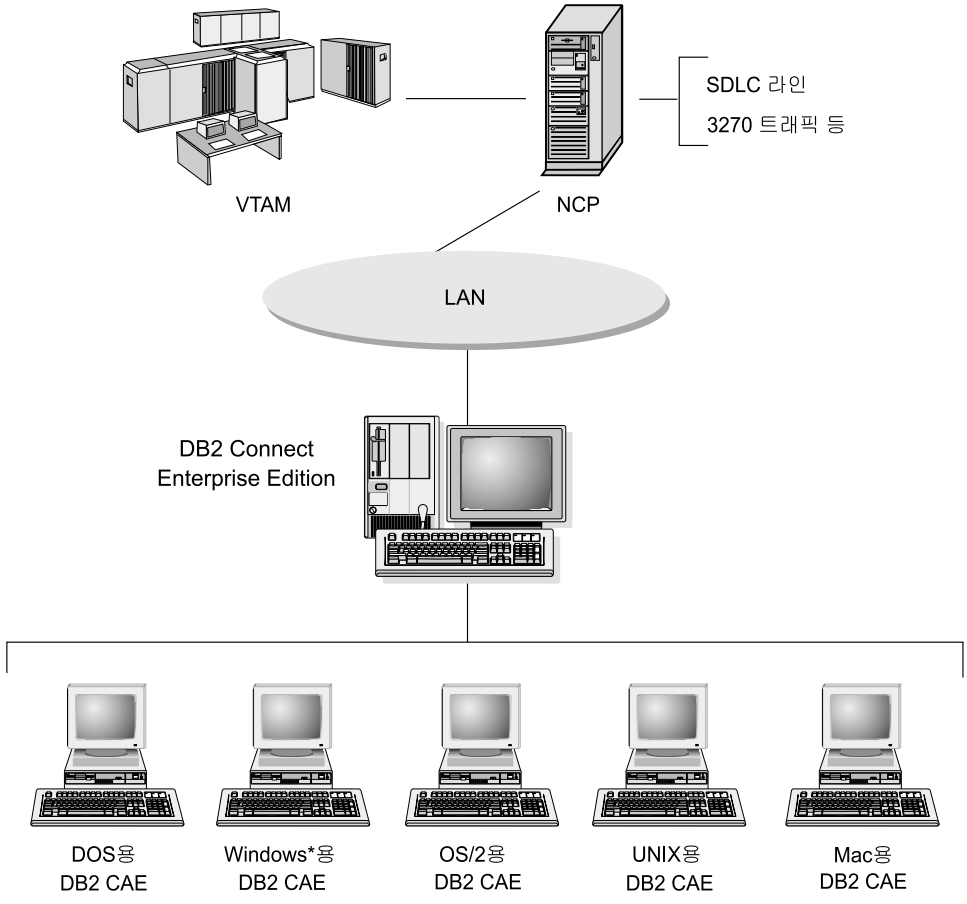
다음은 MPC에 필요한 AIX용 통신 서버 PTF입니다.

```
APAR #      PTF #     LPP name

IX67032   U449693   sna.books.chdoc
IX67032   U449693   sna.books.escdoc
IX67032   U449300   sna.rte
IX67032   U450027   sna.msg.en_US.rte
IX65820   U447759   sna.dlcchannel
IX67618   U449691   mpc.rte
IX65813   U447758   devices.mca.8fc3.rte
```

NCP를 통한 DB2 Connect 연결 조정 방법

통상적인 네트워크 구성은 다음과 같습니다.



*16 및 32 비트 Windows 운영 체제.

그림 8. DB2 Connect Enterprise Edition 게이트웨이 SNA 네트워크 시나리오

이 시나리오는 DB2 Connect Enterprise Edition 게이트웨이에 대한 호스트나 AS/400 데이터베이스 서버들 사이의 처리량 및 응답 시간, 이에 영향을 줄 수 있는 다양한 매개변수에 초점을 두고 있습니다.

조정 기준

이렇게 변경하는 제안 순서는 다음과 같습니다.

```
1 - DELAY on PCCU macro*
2 - DLC/LLC Tuning*
3 - PIU size*
4 - Pacing window changes*
5 - DELAY on LINE macro*
```

```
6 - MAXBFRU changes
7 - LAN Frame sizes

* Major improvement in throughput is possible
```

PIU 크기(RU + 29바이트): 호스트와 DB2 Connect 서버에서 RU 크기는 최대화되어야 합니다. 이는 VTAM 프로그램 스택이 거쳐야 하는 시간을 최소화하기 위해 RU 크기가 API 교차(가능한 곳에서 트랜잭션에 대한 SEND와 RECEIVE 데이터 모두)를 포함할 만큼 충분히 커야 함을 암시합니다. 또한 네트워크 프레임 크기는 RU 단편화가 바람직하지 않은 경우 최대 RU 크기를 제한할 수 있습니다.

DB2 Connect 블록 크기(RQRIOBLK), RU 및 페이싱 값을 RU * pacing >= RQRIOBLK와 같이 설정하는 것은 좋은 생각입니다. 예를 들어, 기본 RQRIOBLK 크기인 32K는 대부분의 상황에 좋은 값이며, 이 값을 이용하려면 RU = 4K로 설정하고 수신 창 페이싱을 8로 설정해야 합니다.

- RU 크기와 페이싱은 DB2 Connect 워크스테이션과 VTAM에 정의된 모드 테이블에 의해 설정됩니다. 모드 테이블 정의는 두 곳에서 같아야 합니다.
- RQRIOBLK는 DB2 UPDATE DBM CFG 명령을 사용하여 설정됩니다.
- 네트워크 프레임 크기 I 프레임은 DB2 Connect 워크스테이션과 NCP의 DLC 구성에서 설정됩니다.

페이싱 창 크기: 세션과 VR 페이싱 창은 최대화되어야 합니다. 즉, 네트워크 과다 전송이나 VR 보유 상황 등을 일으키지 않는 가장 큰 값이 사용되어야 합니다. 테스트 환경의 경우, 페이싱을 0(페이싱하지 않음)으로 설정하거나 최대값 X'3F'로 설정하십시오.

편승 값(DELAY): 편승((Coat-tailing))은 DELAY 매개변수로 제어됩니다. PCCU 매크로에서 DELAY 매개변수는 아웃바운드 편승(호스트를 참조하는 아웃바운드)을 제어합니다. NCP에 대한 LINE 정의 명령문에서 DELAY 값은 인바운드 편승(호스트를 참조하는 인바운드)을 제어합니다.

DELAY 값은 PIU가 전송되기 전에 대기행렬(NCP 또는 VTAM)에 들어있는 기간을 결정합니다. 이 대기의 목적은 다른 PIU가 그 사이에 도착할 가능성을 높이는 것이며 이 PIU는 모두 하나의 채널 프로그램에서 전송될 수 있습니다. 최저의 대기를 위해 DELAY 값은 0으로 설정해야 합니다. 아웃바운드 편승 지연 값을 0

으로 변경하더라도 아웃바운드 통신의 성능 개선을 제외하고 호스트에 두드러진 영향을 주어서는 안됩니다. 약간의 인바운드 통신 성능 개선도 이루어질 것입니다.

NCP에서 DELAY를 0으로 변경하는 것은 좀 더 신중하게 해야 합니다. NCP에 과부하가 걸리지 않고 인바운드 통신이 상당한 비율의 소규모 프레임들로 구성되지 않는 경우, DELAY 값을 0으로 설정할 수 있습니다. DELAY 값을 0으로 설정하면 응답 시간이 특히 가벼운 로드나 테스트/벤치마크 환경에서 상당히 개선될 수 있습니다.

```
VTAMB7   PCCU  CUADDR=CAF,
                                    AUTODMP=NO,
                                    AUTOIPL=NO,
                                    AUTOSYN=YES,
                                    BACKUP=YES,
                                    DELAY=0,
                                    VFYLM=YES,
                                    CHANCON=UNCOND,
                                    MAXDATA=32768,
                                    DUMPDS=NCPDUMP,
                                    OWNER=HOSTB7,
                                    SUBAREA=17


LNCTLS        GROUP   LNCTL=CA,CA=TYPE6,DELAY=0.0,TIMEOUT=500.0
CA0                 LINE         ADDRESS=00
PUCHAN0   PU             PUTYPE=5,TGN=1
CA1                 LINE         ADDRESS=01
PUCHAN1   PU             PUTYPE=5,TGN=1
```

DELAY 고려사항은 *VTAM Network Implementation Guide*에 있습니다.

MAXBFRU: MAXBFRU 값은 가장 큰 PIU 크기보다 2-3배 큰 값으로 설정해야 합니다.

DLC/LLC 계층 조정: NCP와 DB2 Connect Enterprise Edition 게이트웨이 사이의 LLC2 창 크기(DLC 송신과 수신 창 내용)가 같은지 확인하십시오. 이것은 서버가 AIX용 DB2 Connect일 때 특히 상당한 영향을 미칩니다. 송신 창 계수를 수신 창 계수보다 크게 설정하는 것이 좋습니다.

일반적으로 토큰링을 거치는 SNA 연결의 경우 LLC2 타이머/창이 최적화되어야 합니다. 어떤 경우, 이러한 변경은 처리량과 응답 시간에 있어서 6배의 성능 향상으로 이어집니다.

LAN 프레임 크기: 토큰링의 최대 프레임 크기는 가능하면 커야 합니다.

OSA-2 향상에 관한 정보

다음 정보는 IBM WSC Flash 문서 번호 9718에서 발췌한 것입니다.

```
TITLE: WSC  FLASH 9718:  OSA-2 ENHANCEMENTS AVAILABLE
DOCUMENT ID G023691 UNCLASSIFIED

 Open Systems Adapter 2(OSA-2) SNA(Systems Network Architecture)
향상은 이전에 발표된 것보다 일찍 사용 가능합니다.
그 향상은 다음과 같습니다.
o   OS/390, MVS/ESA, VM/ESA 및 VSE/ESA용 SNA/APPN 향상

    -   향상된 가용성: 로드 균형, 중복성 및 오버플로우
    -   향상된 연결성: 증가된 물리 장치(PU) 지원
        (포트당 255개의 PU에서 포트당 2047개의 PU로)

o   VSE/ESA용 ACF/VTAM 네트워크 지원

주의 사항: 이러한 향상은 OSA-1에 관계된 것이 아닙니다.

로드 균형, 중복성 및 오버플로우
_____________________________________

로드 균형: 여러개의 물리적 포트를 경유한 연결이더라도, 이제는
접속된 OSA-2 SNA/APPN PU(Physical Unit)에 대해 단일 MAC(Medium
Access Control) 주소를 정의할 수 있습니다. 이 지원은
브릿지된 소스 라우트 환경에만 제공됩니다(토큰링과 FDDI).
포트를 통해 설정된 세션 수는 모니터되며, 사용자 세션 로드는
동등하게 구성된 포트들 사이에 공평하게 분배됩니다.
중복성: 이제는 LAN 워크스테이션과 호스트 시스템 간의 2차 경로를
구성할 수 있습니다. 1차 경로를 사용할 수 없게 되면, 2차 경로에서
LAN 통신을 수신합니다. 이렇게 되면 시스템 가용성이 높아지고
네트워크 관리가 간편해집니다.
오버플로우: 사용자 세션은 세션 용량에 도달할 때까지 1차 OSA-2
포트를 통해 흐릅니다. 추가 사용자 세션은 자동으로 그 다음 OSA-2
포트로 흐릅니다. 모든 사용자 워크스테이션은 동일하게 구성되므로
네트워크 관리가 간편해지고 네트워크는 더 확장성이 있게 됩니다.
새 사용자를 아무 혼란없이 추가할 수 있습니다.
로드 균형, 중복성 및 오버플로우 지원은 다음과 같이 OSA/SF용
PTF에서 제공됩니다.
o   OS/390과 MVS    - OW20205/UW34618         03/31/97
o   VM/ESA          - OW23952/UW37028         03/31/97
o   VSE/ESA         - VSE/ESA V2.2.1에서 제공됨      04/29/97
증가된 물리 장치(PU) 지원(OSA/SF 경유):
_____________________________________________

OSA-2 이더넷, 토큰링 및 FDDI 기능에 대해 현재 포트당 255개의 PU
```

대신 물리적 포트당 최대 2047개의 PU를 정의할 수 있도록 아키텍처가 변경되었습니다. 이 향상은 새로운 설치뿐만 아니라 현재 설치되어 있는 기능에도 적용됩니다. 실제 연결성은 사용자의 워크로드에 따라 다를 수 있습니다.
증가된 물리 장치(PU) 지원은 다음과 같이 OSA/SF용 PTF에서 제공됩니다.

- OS/390과 MVS - OW23429/UW37210 03/31/97
- VM/ESA - OW24952/UW37028 03/31/97
- VSE/ESA - PQ03091/UQ04224 04/29/97

증가된 물리 장치(PU) 지원은 다음과 같이 ACT/VTAM용 PTF에서 제공됩니다.

- OS/390과 MVS용 ACF/VTAM
  - VTAM 4.1 OW14043/UW24904
  - VTAM 4.2 OW14043/UW24905
  - VTAM 4.3 OW14043/UW24906
- ACF/VTAM VM/ESA
  - VM60877/UV59834
- ACF/VTAM VSE/ESA
  - DY44347/UD50254

VSE/ESA - SNA SUPPORT

OSA-2와 OSA/SF 지원은 VSE/ESA 버전 2 릴리스 2.1을 통해 제공됩니다. 이러한 VSE/ESA 지원에 대한 발표는 1996년 9월 10일자 하드웨어 발표문 196-194 및 196-193에 들어 있는 일반 지시문에 부합합니다.

OSA-2 기능은 VSE/ESA용 ACF/VTAM 호스트 응용프로그램에 이더넷, 토큰링 및 FDDI LAN과 비동기 전송 모드(ATM) 포럼 준수 LAN 에뮬레이션 네트워크에 대한 직접 액세스를 제공합니다.

OSA/SF는 다음과 같이 사용 가능합니다.

- OS/390 릴리스 1 또는 그 이상의 비독점 요소(5645-001)
- 별개의 프로그램 제품인 MVS/ESA 4.3 또는 그 이상용 S/390 Open Systems Adapter Support Facility 버전 1 릴리스 2(5655-104)
- VM/ESA 버전 2 릴리스 2.0의 기능(5654-030)
- VSE/ESA 버전 2 릴리스 2.1에서 VSE Central Functions 6.1.1의 구성요소(5690-VSE).

자세한 정보

발표 297-043, 297-040

기타 정보 소스

이 절에서 추가 정보 소스를 나열합니다.

기타 책

다음에서 성능에 대한 자세한 내용을 참조하십시오.

- *DB2 Connect for OS/2 to DB2 Performance Benchmark*
- *SNA Server for AIX and SNA Server Gateway for AIX Performance Guide*

World Wide Web 사용

World Wide Web에서는 사례 연구 및 예제뿐 아니라 DB2 Connect 성능 조정에 대한 광범위한 정보를 찾아볼 수 있습니다. 웹 브라우저를 다음 URL로 설정하십시오.

```
http://www.ibm.com/software/data/db2/performance/
```

SNA 사용자를 위한 추가 힌트와 추가 정보

175 페이지의 『추가 SNA 성능 조정 힌트와 추가 정보』에서 자세한 내용을 참조하십시오

제13장 문제점 판별

DB2 Connect 환경에는 많은 소프트웨어, 하드웨어, 통신 제품이 포함되어 있습니다. 문제점 판별은 결론(오류 위치)에 이르기 위해 사용 가능한 데이터를 삭제하고 세분하는 프로세스에 의해 가장 잘 수행될 수 있습니다.

다음 주제는 문제점 판별 프로세스에서의 지원을 위해 제공됩니다.

- 『기타 정보 소스』
- 186 페이지의 『관련 정보 수집』
- 187 페이지의 『초기 연결에 성공하지 못한 경우』
- 188 페이지의 『초기 연결 후 문제점이 발생했을 경우』
- 189 페이지의 『진단 도구』
- 190 페이지의 『추적 유틸리티(ddcstrc)』
- 202 페이지의 『가장 일반적인 DB2 Connect 문제점』

관련 정보를 수집하고 적용 가능한 주제를 선택한 후 참조 절로 진행하십시오.

기타 정보 소스

이 절에서 추가 정보 소스를 나열합니다.

문제 해결 안내서 사용

*문제점 해결 안내서*에서 DB2 Connect 및 DB2 Universal Database 문제점 판별 주제에 대한 자세한 내용을 참조하십시오.

World Wide Web 사용

World Wide Web의 DB2 제품 및 서비스 기술 라이브러리에서 DB2 Connect 문제점 판별 힌트와 추가 정보에 대한 최신 정보를 찾을 수 있습니다.

1. 웹 페이지: http://www.ibm.com/software/data/db2/library/ 로 가십시오.
2. DB2 Universal Database 링크를 선택하십시오.

© Copyright IBM Corp. 1993, 2000

3. “DDCS” 또는 “Connect”의 키워드를 사용하여 “Technotes”를 검색하십시오.

APPC, CPI-C 및 SNA 감지 코드 문서화

제공된 APPC, CPI-C 및 SNA 감지 코드에 대한 문서는 다시 패키지에 포함되어 포스트스크립트 파일과 HTML 책(영문판만 있음)으로 제공됩니다.

이 책의 PDF 버전은 DB2 서적 CD-ROM에 있습니다. 이 책의 HTML판은 DB2 Connect가 설치될 때 항상 설치되며, 이 책은 DB2 정보 센터에 통합되어 있습니다.

관련 정보 수집

문제점 판별은 문제의 범위를 좁히고 가능한 원인을 조사하는 과정을 포함합니다. 관련 정보를 수집하고, 아는 정보, 수집되지 않은 데이터와 삭제 가능한 경로를 판별하는 것이 올바른 시작점입니다. 최소한 다음 문제에 대해 대답해야 합니다.

- 초기 연결은 성공적이었습니까?
- 하드웨어가 기능을 잘 수행하고 있습니까?
- 통신 경로가 작동하고 있습니까?
- 이전의 디렉토리 항목이 유효하지 않게 된 통신 네트워크 변경사항이 있습니까?
- 데이터베이스가 시작되었습니까?
- 클라이언트와 DB2 Connect 워크스테이션, DB2 Connect 워크스테이션과 호스트나 AS/400 데이터베이스 서버, 모든 클라이언트 또는 한 클라이언트에서 통신 중단이 있습니까?
- 메시지에서 리턴된 메시지와 토큰 내용으로 무엇을 알 수 있습니까?
- 진단 도구 사용이 현재 지원을 제공합니까?
- 다른 머신에서 같은 타스크가 올바르게 작동하고 있습니까?
- 원격 타스크인 경우, 지역적으로 실행해도 성공적입니까?

초기 연결에 성공하지 못한 경우

다음 문제를 검토하고 설치 단계를 따랐는지 확인하십시오.

1. *설치 처리가 성공적으로 완료되었습니까?*
   - 전제조건 소프트웨어 제품이 모두 사용 가능했습니까?
   - 메모리와 디스크 공간이 적당했습니까?
   - 원격 클라이언트 지원이 설치되었습니까?
   - 다른 오류 조건 없이 통신 소프트웨어 설치가 완료되었습니까?
2. *UNIX 기반 시스템용으로 작성된 제품의 인스턴스입니까?*
   - 사용자와 그룹을, 인스턴스 소유자와 SYSADM 그룹으로 작성했습니까?
3. *적용 가능하다면, 사용권 정보가 성공적으로 처리되었습니까?*
   - UNIX 기반 시스템에서, 노드잠금 파일을 편집하고 IBM이 지원한 암호를 입력했습니까?
4. *호스트나 AS/400 데이터베이스 서버와 워크스테이션 통신이 제대로 구성되어 있습니까?*
   - 반드시 고려해야 하는 세 가지 구성이 있습니다.
     a. 호스트나 AS/400 데이터베이스 서버 구성은 서버에 대한 응용프로그램 리퀘스터를 식별합니다. 호스트나 AS/400 서버 데이터베이스 관리 시스템에는 위치, 네트워크 프로토콜 및 보안면에서 리퀘스터를 정의하는 시스템 카탈로그 항목이 있습니다.
     b. DB2 Connect 워크스테이션 구성은 서버에 대해 클라이언트를 정의하고, 클라이언트에 대해 호스트나 AS/400 서버를 정의합니다.
     c. 클라이언트 워크스테이션 구성에는 워크스테이션 이름과 통신 프로토콜 이름이 반드시 정의되어 있어야 합니다.
   - 초기 연결 실패 문제에 대한 분석은 모든 LU(논리 장치) 이름과 PU(물리 장치) 이름이 SNA 연결에 대해 정확하고 완전한지 확인하거나, 올바른 포트 번호와 호스트 이름이 TCP/IP 연결용에 대해 지정되었는지를 확인하는 과정을 포함합니다.
   - 호스트나 AS/400 서버 데이터베이스 관리자와 네트워크 관리자는 문제를 진단하는 데 사용 가능한 유틸리티를 가지고 있습니다.

5. *호스트나 AS/400 서버 데이터베이스를 사용하기 위해 호스트나 AS/400 서버 데이터베이스 관리 시스템이 필요로 하는 권한 레벨을 가지고 있습니까?*
   - 사용자 액세스 권한, 테이블 규정자 규칙, 예상되는 결과를 고려하십시오.
6. *명령행 처리기를 사용하여 AS/400 데이터베이스 서버에 대해 SQL문을 실행하는 데 성공하지 못했습니까?*
   - 명령행 처리기를 호스트나 AS/400 서버 데이터베이스에 바인드하는 프로시듀어를 따랐습니까?

초기 연결 후 문제점이 발생했을 경우

다음 질문은 문제의 범위를 좁히는 것을 돕는 시작 지점으로 제공됩니다.

1. *특별하거나 이상한 작동 상황이 있습니까?*
   - 이것이 새로운 응용프로그램입니까?
   - 새 프로시듀어가 사용 중입니까?
   - 최근에 시스템에 영향을 줄 만한 변경사항이 있었습니까? 예를 들어, 응용프로그램이나 시나리오가 성공적으로 최종 수행된 후에 소프트웨어 제품 또는 응용프로그램이 변경되었습니까?
   - 응용프로그램의 경우, 어떤 API를 사용하여 프로그램을 작성하였습니까?
   - 소프트웨어나 통신 API를 사용한 다른 응용프로그램이 사용자의 시스템에서 수행되었습니까?
   - PTF가 최근에 설치되었습니까? 설치 후에 사용자가 운영 체제에서 사용되지 않은(로드되지 않은) 기능을 사용했을 때 문제가 발생한 경우, IBM의 최신 PIF 레벨을 알아내고 그 기능을 설치한 후 로드하십시오.
2. *이 오류가 이전에도 발생했습니까?*
   - 이전 오류 상태에 대해 문서로 남아 있는 솔루션이 있습니까?
   - 관련된 사람은 누구이며, 그 사람이 가능한 조치 과정에 대한 힌트를 제공할 수 있습니까?
3. *네트워크에 대하여 정보를 리턴하는 통신 소프트웨어 명령을 사용하여 조사했습니까?*
   - SNA 소프트웨어용으로 사용 가능한 검증 도구가 있습니까?

- TCP/IP를 사용 중이면, TCP/IP 명령을 사용하여 검색되는 중요한 정보가 있을 것입니다.

4. *SQLCA(SQL 통신 영역)에서 리턴된 유용한 정보가 있습니까?*
   - 문제점 처리 프로시듀어에는 SQLCODE 및 SQLSTATE 필드 내용을 검사하는 단계가 있어야 합니다.
   - SQLSTATE를 사용하여 응용프로그래머가 DB2 데이터베이스 제품군에 공통적인 오류의 클래스를 테스트할 수 있습니다. 분산 관계형 데이터베이스 네트워크에서, 이 필드는 공통적인 기본 값을 제공합니다. *메시지 참조서*에서 자세한 내용을 참조하십시오.
5. *DB2START가 서버에서 실행되었습니까?* 추가로, DB2COMM 환경 변수가 서버를 원격으로 액세스하는 클라이언트용으로 올바르게 설정되었는지 확인하십시오.
6. *같은 타스크를 실행하는 다른 머신에서 서버에 성공적으로 연결할 수 있습니까?* 서버에 연결하려는 최대 클라이언트 수에 이르렀을 수 있습니다. 다른 클라이언트가 서버로부터 연결을 해제하면, 클라이언트는 그 이전처럼 연결할 수 없습니까, 아니면 지금은 연결할 수 있습니까?
7. *해당 머신은 올바른 주소를 가지고 있습니까?* 머신이 네트워크에서 유일한지 확인하십시오.
8. *원격 연결인 경우, 클라이언트에 올바른 권한이 부여되었습니까?* 인스턴스에 대한 연결이 성공적이더라도, 데이터베이스 또는 테이블 레벨에서 권한은 부여되지 않았을 수 있습니다.
9. *원격 데이터베이스로 연결하는 첫 번째 머신입니까?* 네트워크간의 분산 환경 라우터또는 브릿지에서는 클라이언트와 서버 사이의 통신을 방해할 수 있습니다. 예를 들어, APPC를 사용하면 세션이 설정될 수 있는지 확인하십시오. TCP/IP를 사용하면, 원격 호스트를 PING 할 수 있는지 확인하십시오.

진단 도구

문제가 발생하는 경우, 다음을 사용할 수 있습니다.

- 첫번째 오류 서비스 로그(FFSL)에서는 진단 정보가 읽을 수 있는 형식으로 통합되고 저장되어 있습니다. *문제점 해결 안내서*에서 자세한 내용을 참조하십시오. *메시지 참조서*에서 로그에 있는 메시지에 대한 자세한 내용을 참조하십시오.
- db2diag.log

  이 파일은 UNIX 시스템에서 /u/db2/sqllib/db2dump/db2diag.log에 있으며, 여기서 db2는 인스턴스 이름입니다.

  이 파일은 Intel 시스템에서 x:\sqllib\db2\db2diag.log에 있으며, 여기서 x:는 논리 드라이브이고 db2는 인스턴스 이름입니다.
- db2alert.log(db2diag.log와 같은 파일 위치).
- 추적 유틸리티는 『추적 유틸리티(ddcstrc)』에 설명되어 있습니다.
- UNIX 기반 시스템에서, **ps** 명령은 사용 중인 프로세스에 대한 프로세스 상태 정보를 표준 출력으로 리턴합니다.
- UNIX 기반 시스템에서, 심각한 오류가 발생할 경우 코어 파일이 현재 디렉토리에서 작성됩니다. 이 파일에는 종료된 프로세스의 메모리 이미지가 들어 있으며, 오류를 발생시킨 기능을 판별하는 데 사용될 수 있습니다.
- Windows NT 및 Windows 2000 시스템에서는, 이벤트 표시기를 사용하십시오.

DB2 제품 및 서비스 기술 라이브러리의 "Technotes" 및 *문제점 해결 안내서*에서 TCP/IP 연결 문제 해결(또는 기타 주제)에 대한 자세한 내용을 참조하십시오(185 페이지의 『World Wide Web 사용』 참조).

추적 유틸리티(ddcstrc)

ddcstrc 유틸리티는 DB2 Connect 워크스테이션(데이터베이스 클라이언트를 대신)과, 호스트 또는 AS/400 서버 데이터베이스 관리 시스템 사이에서 교환된 데이터 레코드를 제공합니다.

데이터베이스 관리자로서(또는 응용프로그램 개발자로서) 이런 데이터의 흐름 작업 방식을 이해하는 것이 필요합니다. 왜냐하면 특정한 문제 발생의 원인을 판별할 수 있기 때문입니다. 예를 들어, 호스트나 AS/400 데이터베이스 서버에 대해

CONNECT TO 데이터베이스 명령문을 실행하려 하지만 명령은 실패하고 실패한 리턴 코드를 수신했다고 합시다. 호스트나 AS/400 데이터베이스 서버 관리 시스템에 전달된 정보를 정확히 알고 있다면, 리턴 코드 정보가 명확하지 않더라도 실패 원인을 판별할 수 있습니다. 실패는 단순한 사용자 오류로 많이 발생합니다.

ddcstrc의 출력은 DB2 Connect 워크스테이션과 호스트 또는 AS/400 데이터베이스 서버 관리 시스템간에 교환된 데이터 스트림을 나타냅니다. 호스트나 AS/400 데이터베이스 서버로 전송된 데이터에는 SEND BUFFER라는 레이블이 붙고, 호스트나 AS/400 데이터베이스 서버로부터 수신된 데이터에는 RECEIVE BUFFER라는 레이블이 붙습니다.

수신 버퍼에 SQLCA 정보가 들어 있으면, 해당 데이터의 포맷된 해석과 레이블된 SQLCA가 다음에 나타납니다. SQLCA의 SQLCODE 필드는 호스트나 AS/400 데이터베이스 서버에서 리턴하는 *맵핑되지* 않은 값입니다. (141 페이지의 『제11장 SQLCODE 맵핑』에서 맵핑 내용을 참조하십시오.) 송신 및 수신 버퍼는 파일안에서 가장 오래된 것부터 가장 최근 것 순으로 배치됩니다. 각 버퍼에는 다음 정보가 있습니다.

- 프로세스 ID
- SEND BUFFER, RECEIVE BUFFER 또는 SQLCA 레이블. 버퍼의 첫 번째 DDM 명령이나 오브젝트는 DSS TYPE로 레이블됩니다.

송신 및 수신 버퍼에 남아 있는 데이터는 다음 내용이 들어 있는 5개의 컬럼으로 나누어집니다.

- 바이트 수.
- 2 컬럼과 3 컬럼은 두 시스템간에 ASCII나 EBCDIC의 교환된 DRDA 데이터 스트림을 나타냅니다.
- 2 컬럼과 3 컬럼의 ASCII 표현.
- 2 컬럼과 3 컬럼의 EBCDIC 표현.

다음 책에서 DDM에 대한 자세한 내용을 참조하십시오.

- *DB2 for OS/390 Reference for Remote DRDA Requesters and Servers*
- *Distributed Relational Database Reference*
- *Distributed Data Management Architecture Level 3: Reference*

추적 구문

이 명령은 운영 체제 명령 프롬프트로부터 다음과 같은 구문으로 호출됩니다.

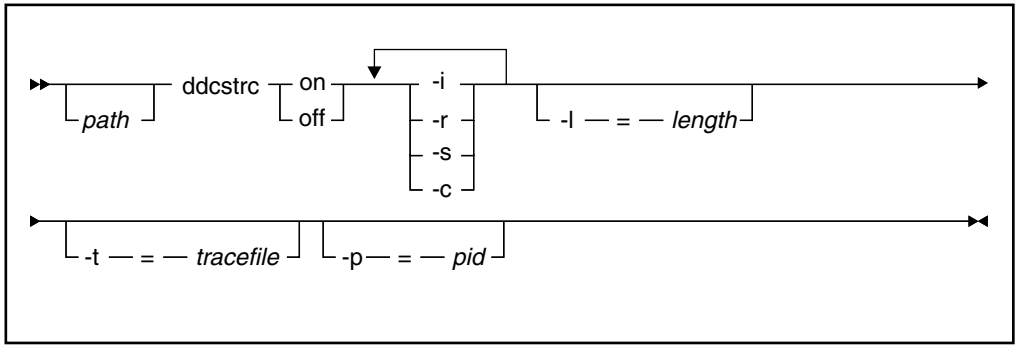


그림 9. ddcstrc 명령 구문

주: 이 명령 구문은 사용 중인 운영 체제에 따라 약간은 다를 수 있습니다. 예를 들어, **/**는 OS/2 운영 체제에서의 **-**을 대신하여 사용됩니다.

추적 매개변수

on 호스트나 AS/400 데이터베이스 서버에서 DRDA 흐름의 DB2 Connect 추적을 작동시킵니다.

off 호스트나 AS/400 데이터베이스 서버에서 DRDA 흐름의 DB2 Connect 추적을 해제시킵니다.

-i 시간소인이 추적 정보에 포함됩니다.

-r 호스트나 AS/400 서버 시스템에서 수신된 DRDA 데이터 스트림을 추적합니다.

-s 호스트나 AS/400 데이터베이스 서버에서 송신한 DRDA 데이터 스트림을 추적합니다.

-c 호스트나 AS/400 데이터베이스 서버에서 수신된 SQLCA를 추적합니다.

기본 값은 -r, -s, -c입니다.

-l=*길이*

추적 정보를 저장하는데 사용되는 버퍼 *크기*를 지정합니다. 기본 값은 1M이며, 최소값은 64K입니다.

-t=*추적파일*

추적용 목적지를 지정하여, *추적파일*은 파일 이름이거나 표준 장치 이름입니다. 파일 이름이 완전한 경로없이 지정되면, 현재 경로가 빠진 부분에 대해 사용됩니다. 기본 파일 이름은 `ddcstrc.dmp`입니다.

-p=*pid* 이 프로세스용 이벤트만 추적합니다. -p가 지정되지 않으면, 사용자 인스턴스의 프로세스는 모두 출력 파일에 기록됩니다.

주: 원격 클라이언트에서, *pid*는 데이터베이스 시스템 모니터에 의해 리턴된 에이전트 ID 필드에서 찾을 수 있습니다.

111 페이지의 『제8장 데이터베이스 시스템 모니터』에서 자세한 내용을 참조하십시오.

추적 출력

ddcstrc는 다음 정보를 *추적파일*에 기록합니다.

- -r
  - DRDA 응답/오브젝트 유형
  - 수신 버퍼
- -s
  - DRDA 요청 유형
  - 송신 버퍼
- -c
  - SQLCA
- CPI-C 오류 정보
  - 수신 함수 리턴 코드
  - 심각도
  - 사용된 프로토콜
  - 사용된 API

  - 함수
  - CPI-C 리턴 코드
  - 오류 번호
  - 내부 리턴 코드
- SNA 오류 정보
  - 수신 함수 리턴 코드
  - 심각도
  - 사용된 프로토콜
  - 함수
  - 상대 LU 이름
  - 오류 번호
- TCP/IP 오류 정보
  - 수신 함수 리턴 코드
  - 심각도
  - 사용된 프로토콜
  - 사용된 API
  - 함수
  - 오류 번호

주:

1. Exit 코드 0 값은 성공적으로 완료된 명령을 나타내고, 0이 아닌 값은 반대의 경우를 나타냅니다.
2. 리턴되는 필드는 사용되는 API에 따라 다릅니다. SNA API는 2PC SPM 연결용으로만 사용됩니다.
3. 리턴되는 필드는 같은 API용이더라도 DB2 Connect가 수행중인 플랫에 따라 다릅니다.
4. ddcstrc가 이미 존재하는 파일로 출력을 송신하는 경우, 파일상에서 삭제를 허용하면 오래된 파일은 삭제됩니다.

추적 출력 파일 분석

다음 페이지에 DB2 Connect 워크스테이션과 호스트 또는 AS/400 데이터베이스 서버 사이에 교환된 여러 DRDA 데이터 스트림을 설명하는 출력의 예가 나와 있습니다. 사용자의 관점에서, CONNECT TO 데이터베이스 명령은 명령행 처리기를 사용하여 실행됩니다.

그림10은 APPC 연결에서 DB2 Connect Enterprise Edition 버전 7과 OS/390용 DB2 Universal Database 버전 5.1을 사용합니다.

197 페이지의 그림11은 TCI/IP 연결에서 DB2 Connect Enterprise Edition 버전 7과 OS/390용 DB2 Universal Database 버전 5.1을 사용합니다.

```
1       DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
        pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 177

        SEND BUFFER:  EXCSAT RQSDSS         (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  006AD04100010064 10410020115E8482   .j.A...d.A. .^..  .|}.........;db
  0010  F282974040404040 4040404040404040   ...@@@@@@@@@@@@@  2bp
  0020  4040F0F0F0F1F7F3 C5C3000C116DA685   @@...........m..    000173EC..._we
  0030  81A2859340400013 115AC4C2F240C396   ....@@...Z...@..  asel  ...]DB2 Co
  0040  95958583A340F54B F200141404140300   .....@.K........  nnect 5.2.......
  0050  0414440003240700 05240F0003000D11   ..D..$...$......  ................
  0060  47D8C4C2F261F6F0 F0F00085D0010002   G....a..........  .QDB2/6000.e}...
  0070  007F200100162110 E2C1D56DC6D9C1D5   .. ...!....m....  ."......SAN_FRAN
  0080  C3C9E2C3D6404040 40400006210F2407   .....@@@@@..!.$.  CISCO     ......
  0090  000D002FD8E3C4E2 D8D3C1E2C3000C11   .../............  ....QTDSQLASC...
  00A0  2EE2D8D3F0F5F0F2 F0003C210437E2D8   ..........
```

그림 10. 추적 출력 예(APPC 연결) (1/2)

```
3       DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
        pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 177

        SEND BUFFER:  RDBCMM  RQSDSS        (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  000AD00100010004 200E               ........ .        ..}.......

4       DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
        pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 178

        RECEIVE BUFFER:  ENDUOWRM RPYDSS    (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  002BD05200010025 220C000611490004   .+.R...%"....I..  ..}.............
  0010  00162110E2C1D56D C6D9C1D5C3C9E2C3   ..!....m........  ....SAN_FRANCISC
  0020  D6404040404000005 211501000BD00300   .@@@@@..!.......  O     .......}..
  0030  0100052408FF                        ...$..            ......

5       DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
        pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 179
SQLCA

        SQLCAID:  SQLCA
        SQLCABC:  136
        SQLCODE:  0
        SQLERRML: 0
        SQLERRMC:
        SQLERRP:  DSN
        SQLERRD[0->5]: 00000000, 00000000, 00000000, 00000000, 00000000, 00000000
        SQLWARN(0->A):  ,  ,  ,  ,  ,  ,  ,  ,  ,  ,
        SQLSTATE: 00000
```

그림 10. 추적 출력 예(APPC 연결) (2/2)

```
1       DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
        pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 177

        SEND BUFFER:  EXCSAT RQSDSS         (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  006ED04100010068 10410020115E8482   .n.A...h.A. .^..  .>}.........;db
  0010  F282974040404040 4040404040404040   ...@@@@@@@@@@@@@  2bp
  0020  4040F0F0F0F1F3F9 F9C5000C116DA685   @@..........m..    0001399E..._we
  0030  81A2859340400013 115AC4C2F240C396   ....@@...Z...@..  asel  ...]DB2 Co
  0040  95958583A340F54B F200181404140300   .....@.K........  nnect 5.2.......
  0050  0514740005240700 05240F0003144000   ..t..$...$....@.  ..............
  0060  05000D1147D8C4C2 F261F6F0F0F00010   ....G....a......  .....QDB2/6000..
  0070  D0410002000A106D 000611A20003003C   .A.....m.......<  }......_...s....
  0080  D04100030036106E 000611A200030016   .A...6.n........  }......>...s....
  0090  2110E2C1D56DC6D9 C1D5C3C9E2C3D640   !....m.........@  ..SAN_FRANCISCO
  00A0  40404040000C11A1 9781A2A2A6969984   @@@@............      ....password
  00B0  000A11A0A4A28599 8984009CD0010004   ................  ....userid..}...
  00C0  0096200100162110 E2C1D56DC6D9C1D5   .. ...!....m....  .o......SAN_FRAN
  00D0  C3C9E2C3D6404040 40400006210F2407   .....@@@@@..!.$.  CISCO     ......
  00E0  000D002FD8E3C4E2 D8D3C1E2C3000C11   .../............  ....QTDSQLASC...
  00F0  2EE2D8D3F0F5F0F2 F0003C210437E2D8   ..........
```

그림 11. 추적 출력의 예(TCP/IP 연결) (1/2)

```
3       DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
        pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 177

        SEND BUFFER:  RDBCMM  RQSDSS        (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  000AD00100010004 200E               ........ .        ..}.......

4       DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
        pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 178

        RECEIVE BUFFER:  ENDUOWRM RPYDSS    (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  002BD05200010025 220C000611490004   .+.R...%"....I..  ..}.............
  0010  00162110E2C1D56D C6D9C1D5C3C9E2C3   ..!....m........  ....SAN_FRANCISC
  0020  D6404040404000005 211501000BD00300   .@@@@@..!.......  O     .......}..
  0030  0100052408FF                        ...$..            ......

5       DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
        pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 179
SQLCA

        SQLCAID:  SQLCA
        SQLCABC:  136
        SQLCODE:  0
        SQLERRML: 0
        SQLERRMC:
        SQLERRP:  DSN
        SQLERRD[0->5]: 00000000, 00000000, 00000000, 00000000, 00000000, 00000000
        SQLWARN(0->A):  ,  ,  ,  ,  ,  ,  ,  ,  ,  ,
        SQLSTATE: 00000
```

그림 11. 추적 출력의 예(TCP/IP 연결) (2/2)

다음 정보는 추적에서 캡처됩니다.

- 클라이언트 응용프로그램의 프로세스 ID(PID)
- 데이터베이스 연결 서비스(DCS) 디렉토리에 카탈로그화된 RDB_NAME
- DB2 Connect CCSID
- 호스트나 AS/400 데이터베이스 서버 CCSID
- DB2 Connect 시스템이 통신하고 있는 호스트나 AS/400 데이터베이스 서버 관리 시스템

첫 번째 버퍼에는 교환 서버 속성(EXCSAT)과 호스트 또는 AS/400 데이터베이스 서버 관리 시스템으로 송신된 액세스 RDB(ACCRDB) 명령이 들어 있습니다. CONNECT TO 데이터베이스 명령의 결과로서 이 속성과 명령을 송신합니다.

다음 버퍼에는 DB2 Connect가 호스트나 AS/400 데이터베이스 서버 관리 시스템으로부터 수신한 응답이 들어 있습니다. 이 응답에는 교환 서버 속성 응답 데이터(EXCSATRD)와 액세스 RDB 응답 메시지(ACCRDBRM)가 들어 있습니다.

EXCSAT과 ACCRDB 분석

EXCSAT 명령에는 서버 이름(SRVNAM) 오브젝트에 의해 지정된 클라이언트의 워크스테이션 이름이 들어 있으며, DDM 스펙에 의한 코드 포인트 X'116D'입니다. EXCSAT 명령은 첫 번째 버퍼에 있습니다. EXCSAT 명령내에서, X'116DA68581A28593'(CCSID 500에 코드화된) 값은 X'116D'가 삭제되면 *weasel*로 번역됩니다.

또한 EXCSAT 명령에는 EXTNAM(외부 이름) 오브젝트가 들어 있으며, 이 오브젝트는 호스트나 AS/400 데이터베이스 관리 시스템상의 진단 정보에 있는 경우가 많습니다. 이것은 8바이트 프로세스 ID(또는 4바이트 프로세스 ID와 4바이트 스레드 ID)와 20바이트 응용프로그램 ID로 이루어져 있습니다. 코드 포인트 X'115E'로 나타나고, 이번 예에서는 해당 값이 0000BE5c가 뒤에 오는 공백들로 채워진 db2bp_32입니다. UNIX 기반 데이터베이스 클라이언트에서, 이 값은 **ps** 명령과 상관될 수 있으며, 사용 중인 프로세스에 대한 프로세스 상태 정보를 표준 출력으로 리턴합니다.

ACCRDB 명령에는 RDBNAM 오브젝트의 RDB_NAME이 들어 있으며 코드 포인트 X'2110'입니다. ACCRDB 명령은 첫 번째 버퍼의 EXCSAT 명령 다음에 옵니다. ACCRDB 명령 안에서, X'2110E2C1D56DC6D9C1D5C3C9E2C3D6' 값은 X'2110'가 삭제되면 SAN_FRANCISCO로 번역됩니다. DCS 디렉토리의 목표 데이터베이스 이름 필드에 해당합니다.

계정 문자열에는 코드 포인트 X'2104'가 있습니다(65 페이지의 『OS/390용 DB2 Universal Database에서의 사용 비용 청구 계정 구현』 참조).

DB2 Connect용으로 구성된 코드 세트는 ACCRDB 명령의 코드 포인트 X'119C'를 가진 CCSID 오브젝트 CCSIDSBC(1바이트 문자용 CCSID)를 위치지정하여 나타납니다. 이번 예에서, CCSIDSBC는 X'0352'이고 850입니다.

각기 코드 포인트 X'119D' 및 X'119E'를 가진 추가 오브젝트 CCSIDDBC(2바이트 문자용 CCSID)와 CCSIDMBC(혼합 바이트 문자용 CCSID)가 나타나면, DB2 Connect 워크스테이션은 DBCS 코드 페이지 지원용으로 구성됩니다. 예를 들면 출력 파일에는 두 개의 추가 코드 포인트가 없으므로, 워크스테이션이 DBCS용으로 구성되지 않습니다.

주: TCP/IP 흐름에는 두 가지 새로운 명령이 들어 있습니다. ACCSEC는 보안 관리 프로그램을 액세스하고 보안 메카니즘이 지원되는 교환을 하는데 사용되며, SECCHK에는 일반 사용자 연결을 인증하는 데 사용된 인증 토큰이 들어 있습니다. ACCSEC과 SECCHK은 TCP/IP 연결에서만 나타나고 EXCSAT과 ACCRDB 사이에도 나타납니다.

EXCSATRD와 ACCRDBRM 분석

CCSID 값은 또한 두 번째 버퍼 안에서 액세스 RDB 응답 메시지(ACCRDBRM)의 호스트나 AS/400 데이터베이스 서버로부터 리턴됩니다. 이 버퍼에는 ACCRDBRM 다음에 나오는 EXCSATRD가 들어 있습니다. 예로 든 출력 파일에는 500(X'01F4', SBCS CCSID)의 호스트나 AS/400 데이터베이스 서버 시스템용 CCSID 값이 들어 있습니다.

DB2 Connect가 호스트나 AS/400 데이터베이스 서버로부터 리턴된 코드 페이지를 인식하지 못하는 경우, SQLCODE -332는 소스 및 목표 코드 페이지와 함께 사용자에게 리턴됩니다. 호스트나 AS/400 데이터베이스 서버가 DB2 Connect로부터 송신된 코드 세트를 인식하지 못할 경우, VALNSPRM(지원되지 않는 매개변수 값으로 DDM 코드 포인트 X'1252'를 가짐)이 리턴되는데 이 값은 사용자를 위해 SQLCODE -30073로 변환됩니다.

ACCRDBRM은 또한 매개변수 PRDID(코드 포인트 X'112E'를 가진 특정 제품 식별자)가 들어 있습니다. 그 값은 X'C4E2D5F0F5F0F1F0'입니다. 이 16진수 문자열은 EBCDIC의 DSN05010에 해당합니다. 표준에 의하면, DSN는 MVS/ESA용

DB2 또는 OS/390용 DB2 Universal Database입니다. 버전 5.1도 해당됩니다. ARI는 VSE & VM용 DB2이고, SQL은 DB2 일반 서버이며, QSQ는 AS/400용 DB2 Universal Database입니다.

다음의 버퍼 분석

추가 정보를 얻기 위해 다음 송신 및 수신 버퍼를 분석할 수 있습니다. 세 번째 버퍼에는 연결이 들어 있습니다. **commit** 명령은 호스트나 AS/400 데이터베이스 서버 관리 시스템에 현재의 작업 단위(UOW)를 확약하도록 지시합니다 네 번째 버퍼는 확약이나 구간 복원의 결과로 호스트나 AS/400 데이터베이스 서버 관리 시스템으로부터 수신됩니다. 이 버퍼에는 종료 작업 단위(UOW) 응답메시지(ENDUOWRM)가 들어 있고, 현재 작업 단위(UOW)가 종료된 것을 나타냅니다. 이 예에서, 버퍼에 X'FF' 다음에 나오는 DDM 코드 포인트 X'2408'로 나타나는 널(NULL) SQLCA가 들어 있습니다. 널(NULL) SQLCA(X'2408FF')는 성공을 나타냅니다(SQLCODE 0). 수신 버퍼에 SQLCA(아마 널(NULL) SQLCA)가 들어 있으면, ddcstrc는 이 SQLCA 정보의 형식화된 번역이 있는 이 수신 버퍼 다음에 올 것입니다.

202 페이지의 그림12에서 오류 SQLCA와 SQLCA의 형식화된 화면이 들어 있는 수신 버퍼의 예를 보여줍니다. 이 SQLCA는 존재하지 않은 테이블로부터 행을 삭제하려고 시도한 결과입니다.

```
1       DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
        pid 48732; tid 1; node 0; cpid 0; sec 0; nsec 0; tpoint 178

        RECEIVE BUFFER:  SQLCARD OBJDSS     (ASCII)          (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  0065D0030001005F 240800FFFFFF34F4   .e....._$.....4.  ..}....^.......4
  0010  F2F7F0F4C4E2D5E7 D6E3D34000E2C1D5   ...........@....  2704DSNXOTL .SAN
  0020  6DC6D9C1D5C3C9E2 C3D64040404040FF   m.........@@@@@.  _FRANCISCO     .
  0030  FFFE0C0000000000 000000FFFFFFFF00   ................  ................
  0040  0000000000000040 4040404040404040   .......@@@@@@@@@  .......
  0050  40400000000FC4C4 C3E2E4E2F14BD4E8   @@...........K..    ....DDCSUS1.MY
  0060  E3C1C2D3C5                          .....             TABLE

2       DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
        pid 48732; tid 1; node 0; cpid 0; sec 0; nsec 0; tpoint 179
SQLCA

        SQLCAID:  SQLCA
        SQLCABC:  136
        SQLCODE:  -204
        SQLERRML: 15
        SQLERRMC: DDCSUS1.MYTABLE
        SQLERRP:  DSNXOTL
        SQLERRD[0->5]: FFFFFE0C, 00000000, 00000000, FFFFFFFF, 00000000, 00000000
        SQLWARN(0->A):  ,  ,  ,  ,  ,  ,  ,  ,  ,  ,
        SQLSTATE: 42704
```

그림 12. 수신 버퍼의 예

가장 일반적인 DB2 Connect 문제점

이 절에서는 DB2 Connect를 사용할 때 발생하는 연결 문제의 가장 일반적인 징후를 설명합니다. 각 경우마다 다음 항목이 제공됩니다.

- 해당 메시지와 관련된 메시지 번호와 리턴 코드(또는 프로토콜 특정의 리턴 코드)의 조합. 각 메시지와 리턴 코드 조합은 별도의 표제를 가지며 표제는 메시지 번호, 그 다음에 리턴 코드순으로 정렬됩니다.
- 징후는 대개 샘플 메시지 목록 형식으로 제공됩니다.
- 오류의 가능한 원인을 나타내는 솔루션을 제안합니다. 어떤 경우에는 둘 이상의 솔루션을 제안할 수도 있습니다.

주:

1. 권장되는 소프트웨어 수정 레벨에 관한 가장 최신 정보를 얻으려면 해당 제품의 빠른 시작 매뉴얼과 최신 릴리스 정보를 참조하십시오.
2. APPC 통신의 특정 메시지와 리턴 코드 조합의 경우, SNA 감지 코드도 표시될 수 있습니다. 현재는 특정 메시지와 관련된 SNA 감지 코드 정보를 SNA 서브시스템에서 얻어야 합니다.

   때때로 시스템 로그를 살펴봄으로써 SNA 감지 코드를 볼 수 있습니다. 이 경우에 해당하는지의 여부는 사용되는 SNA 서브시스템에 달려 있으며, 어떤 상황에서는 감지 코드 정보를 얻기 위해 SNA 추적을 작동시켜 문제를 다시 만들어야 할 수도 있습니다.
3. 게이트웨이라는 용어는 DB2 Connect Enterprise Edition과 관련됩니다.

SQL0965 또는 SQL0969

징후

메시지 SQL0965와 SQL0969는 AS/400용 DB2 Universal Database, OS/390용 DB2 Universal Database, MVS/ESA용 DB2 및 VM과 VSE용 DB2에서 서로 다른 리턴 코드 번호로 발행될 수 있습니다.

둘 중에서 하나의 메시지가 표시될 때, 그 메시지를 제시한 데이터베이스 서버 제품을 알아내려면 문서에서 원래 SQL 코드를 찾아보아야 합니다.

솔루션

호스트 데이터베이스로부터 수신된 SQL 코드는 변환할 수 없습니다. 오류 코드를 기초로 문제를 정정한 다음 실패 명령을 다시 제출하십시오.

CONNECT시 SQL1338

징후 / 원인

기호 목적지 이름이 정의되지 않았거나 제대로 정의되지 않았습니다.

예를 들어, APPC 노드가 사용되고 DB2 노드 디렉토리에서 지정된 기호 목적지 이름이 지역 APPC 통신 서브시스템 구성에 있는 CPI-C 항목과 일치하지 않을 때 이 오류가 발생할 수 있습니다.

또다른 원인으로는 머신에 둘 이상의 SNA 스택이 설치된 것일 수 있습니다. 사용하고자 하는 스택이 먼저 참조되는지 확인하려면 PATH와 LIBPATH를 확인해야 합니다.

솔루션

1. DB2 노드 디렉토리 항목에서 지정된 CPIC쪽 정보 프로파일 이름이 SNA 구성과 일치하는지 확인하십시오(이 프로파일 이름은 대소문자를 구별합니다).
2. 사용하려는 SNA 스택을 먼저 참조하는지 확인하려면 PATH와 LIBPATH를 확인해야 합니다.

CONNECT시 SQL1403N

징후

SQL1403N 제공된 사용자 이름 또는 암호가 틀립니다.

솔루션

1. 사용자가 DB2 Connect 워크스테이션에서 인증을 받을 수 없습니다. 사용자가 DB2 Connect 워크스테이션에서 인증을 받기로 되어 있는지를 알아보십시오.

   인증을 받기로 되어 있으면, 필요한 경우 CONNECT 명령문에 올바른 암호를 제공하였는지 확인하십시오.

   인증을 받기로 되어 있지 않으면, 시스템 데이터베이스 디렉토리 항목이 AUTHENTICATION SERVER를 사용하여 틀리게 카탈로그화되어 있습니다(AUTHENTICATION이 명시적으로 지정되지 않으면 이것이 기본 값입니다). 이 경우에는 AUTHENTICATION DCS나 CLIENT를 사용하여 항목을 다시 카탈로그화하십시오.
2. 암호를 목표 서버 데이터베이스로 보낼 수 없습니다. 시스템 데이터베이스 디렉토리 항목이 AUTHENTICATION DCS를 사용하여 카탈로그화된 경우, DB2 Client에서 목표 서버 데이터베이스로 암호를 보내야 합니다. 예를 들어, AIX와 같은 특정 플랫폼에서는 CONNECT 명령문에 암호를 제공하는 경우에만 암호를 얻을 수 있습니다.

SQL5043N

징후

하나 이상의 통신 프로토콜 지원이 성공적으로 시작되지 않았습니다. 그러나 핵심 데이터베이스 관리 프로그램 기능은 성공적으로 시작되었습니다.

아마 DB2 Connect 게이트웨이에서 TCP/IP 프로토콜이 시작되지 않았을 것입니다. 이전에는 클라이언트 연결에 성공하였을 것입니다.

`diaglevel = 4`이면 `db2diag.log`에 예를 들면 다음과 유사한 항목이 들어 있을 것입니다.

```
1997-05-30-14.09.55.321092   Instance:svtdbm5   Node:000
PID:10296(db2tcpcm)   Appid:none
common_communication  sqlcctcpconnmgr_child   Probe:46
DIA3205E Socket address "30090" configured in the TCP/IP
services file and
required by the TCP/IP server support is being used by another
process.
```

솔루션

이 경고는 원격 클라이언트의 게이트웨이 역할을 하는 DB2 Connect에서 하나 이상의 클라이언트 통신 프로토콜을 처리하는 데 문제가 있음을 알리는 징후입니다. 이 프로토콜은 TCP/IP, APPC 및 기타의 것일 수 있으며, 메시지는 보통 DB2 Connect에 정의된 통신 프로토콜 중 하나가 제대로 구성되지 않았음을 나타냅니다.

DB2COMM 프로파일 변수가 정의되지 않았거나 제대로 정의되지 않은 것이 흔한 원인일 수 있습니다. 일반적으로 DB2COMM 변수와 데이터베이스 관리 프로그램 구성에 정의된 이름(예를 들면 svcename, nname 또는 tpname)간의 불일치가 문제입니다.

하나의 가능한 시나리오는 이전에는 연결이 성공적이었다가 구성이 아무 것도 변경되지 않은 상태에서 SQL5043 오류 메시지가 나타난 것입니다. 이것은 TCP/IP 프로토콜 사용시 원격 시스템이 어떤 이유로 연결을 비정상적으로 종료할 때, 발생할 수 있습니다. 이 오류가 발생할 때, 클라이언트에는 여전히 연결이 존재하는 것으로 나타날 수 있으며, 아래의 명령을 실행함으로써 더 이상의 개입없이 연결을 복원하는 것이 가능해질 수 있습니다.

아마 게이트웨이에 연결한 클라이언트 중 하나는 여전히 TCP/IP 포트에 핸들을 가지고 있을 것입니다. 게이트웨이에 연결된 각 클라이언트 머신에는 다음과 같은 핸들이 있습니다.

1. db2 terminate
2. db2stop

SQL30020

징후

SQL30020N 후속 명령과 SQL문의 성공적인 실행에 영향을 줄 분산 프로토콜 오류 때문에 실행하지 못하였습니다.

솔루션

이 오류가 발생하면 서비스를 받아야 합니다.

db2dump 디렉토리에서 ffdc 덤프(pid.000)가 있는지 확인합니다. 그런 다음, db2fdump를 사용하여 이 덤프 파일을 형식화하고 결과 파일에서 ″ERROR″를 살펴봅니다. MVS ABEND가 여기에 나열되어 있을 수 있습니다. 이 경우에는 MVS 콘솔을 확인하여 자세한 정보를 알아보고 MVS 메시지와 코드 매뉴얼의 DB2 부분에서 이상 종료 코드를 살펴봅니다.

SQL30060

징후

SQL30060N <권한-ID>는 조작 <조작>″을 수행할 수 있는 특권이 없습니다.

솔루션

MVS용 DB2나 OS/390용 DB2에 연결할 때, 통신 데이터베이스(CDB) 테이블이 제대로 갱신되지 않았습니다. 다음 설명서를 참조하십시오.

- DB2 Connect 빠른 시작 또는
- DB2 연결성 보충 설명서

SQL30061

징후

잘못된 호스트나 AS/400 데이터베이스 서버 위치에 연결하고 있습니다 - 목표 데이터베이스를 찾을 수 없습니다.

솔루션

잘못된 서버 데이터베이스 이름이 DCS 디렉토리 항목에 지정되었을 수 있습니다. 이 오류가 발생하면 SQLCODE -30061이 응용프로그램에 리턴됩니다.

DB2 노드, 데이터베이스 및 DCS 디렉토리 항목을 확인해 보십시오. DCS 디렉토리 항목의 목표 데이터베이스 이름 필드가 플랫폼에 있는 데이터베이스 이름과 일치해야 합니다. 예를 들어 OS/390용 DB2 Universal Database 데이터베이스의 경우, 사용될 이름이 부트 스트랩 데이터 세트(BSDS) ″LOCATION=locname″ 필드에 사용된 이름과 같아야 하는데, 이 필드에 사용된 이름은 분산 데이터 기능(DDF)이 시작될 때 DSNL004I 메시지(LOCATION=위치)에도 제공됩니다. 5 페이지의 『데이터베이스 개념』 및 87 페이지의 『제6장 데이터베이스 디렉토리 갱신』에서도 참조하십시오.

DB2 Connect 빠른 시작 매뉴얼에 DB2 카탈로그 갱신 방법을 보여주는 예도 들어 있습니다. SNA 구성을 설명하는 각 장에서 ″DB2 디렉토리 갱신″ 절을 참조하거나 ″DB2 Connect의 호스트와 AS/400 데이터베이스 구성″ 장의 ″TCP/IP 연결 구성″ 절을 참조하십시오.

APPC 또는 APPN 노드에 대한 올바른 명령은 다음과 같습니다.

```
db2 catalog appc node <node_name> remote <sym_dest_name> security program
db2 catalog dcs database <local_name> as <real_db_name>
db2 catalog database <local_name> as <alias> at node <node_name>
            authentication dcs
```

TCP/IP 노드에 대한 올바른 명령은 다음과 같습니다.

```
db2 catalog tcpip node <node_name> remote <host_name_or_address>
             server <port_no_or_service_name>
db2 catalog dcs database <local_name> as <real_db_name>
db2 catalog database <local_name> as <alias> at node <node_name>
             authentication dcs
```

그런 다음 데이터베이스에 연결하려면 다음 명령을 실행하십시오.

```
db2 connect to <alias> user <user_name> using <password>
```

CONNECT시 리턴 코드 119C를 수반한 SQL30073

징후

메시지 SQL30073은 리턴 코드 119C와 함께 제시됩니다. 이 오류는 목표 서버 데이터베이스가 DB2 클라이언트에 사용된 코드 페이지를 지원하지 않을 때 발생합니다(DB2 Connect를 통해). 코드 페이지는 DB2 클라이언트가 수행중인 운영 환경의 구성에서 가져온 것입니다.

*관리 안내서*에서 자세한 내용을 참조하십시오.

솔루션

이 문제는 종종 목표 서버 데이터베이스 시스템에 수정 패치를 설치하여 해결할 수 있습니다. 해당 서비스 부서에 연락하여 이 징후에 대해 권고하는 수정 패치를 얻어 설치하십시오.

일시적인 해결책으로, 사용자는 DB2CODEPAGE 환경 변수를 설정하여 기본 코드 페이지를 겹쳐쓸 수 있습니다. 로케일을 확인하거나 DB2CODEPAGE=850으로 설정하십시오.

UNIX 플랫폼의 사용자는 LANG 환경 변수를 다른 값으로 설정하여 다른 코드 페이지로 전환할 수 있습니다.

리턴 코드 1을 수반한 SQL30081N

징후

징후는 다음 메시지와 SNA 감지 코드입니다.

```
  db2 connect to <database name> user <userid>
  Enter password for <userid>:
SQL30081N  A communication error has been detected.
   Communication protocol
   being used: "APPC".  Communication API being used: "CPI-C".
   Location where
   the error was detected: "".  Communication function detecting
   the error:
  "cmallc".  Protocol specific error code(s): "1", "*",
  "0x10030021".
   SQLSTATE=08001
```

솔루션

이 샘플에서 감지 코드는 10030021입니다.

이 오류 메시지에 관련된 가장 일반적인 감지 코드와 각 경우에 제안되는 루션은 다음과 같습니다.

1.

   리턴 코드 1을 수반한 SQL30081N과 sna 감지 코드 0877002C

   잘못된 네트워크 이름이 지정되었습니다.

2.

   리턴 코드 1을 수반한 SQL30081N과 SNA 감지 코드 ffff0003

   잘못된 MAC 주소가 지정되었거나 SNA 링크가 사용 중이 아닙니다.

3.

   리턴 코드 1을 수반한 SQL30081N과 SNA 감지 코드 10030021

   LU 유형이 일치하지 않습니다.

4.

   리턴 코드 1을 수반한 SQL30081N과 SNA 감지 코드 084B6031

   DSNZPARM(MVS용 DB2나 OS/390 호스트용 DB2에서)의 MAXDBAT가 0으로 설정되었습니다.

다음은 다른 제안입니다.

1. 지역 LU 프로파일을 작성할 때 LU를 기본 LU로 정의하십시오. 예를 들어, CM/2의 SNA 기능 목록 패널에서 다음 중 하나를 수행하십시오.
   - '이 지역 LU를 기본 지역 LU 별명으로 사용' 선택란에 체크 표시를 합니다.
   - DB2 Connect Enterprise Edition 게이트웨이 시스템의 프로파일이나 환경 변수 APPCLLU를 지역 LU 이름으로 설정합니다. 이는 CONFIG.SYS를 편집하여 OS/2 시스템에서 수행하거나 제어판을 통해 Windows NT 시스템에서 수행할 수 있습니다.
2. DB2 Connect 게이트웨이에서 SNA가 시작되는지 확인하십시오.

3. MVS용 DB2나 OS/390용 DB2를 사용 중인 경우, 분산 데이터 기능(DDF) 주소 공간이 시작되는지와 해당 DB2가 수행중인지 확인하십시오.

리턴 코드 2를 수반한 SQL30081N

징후

메시지 SQL30081N이 리턴 코드 2 및 SNA 감지 코드 08120022와 함께 표시됩니다.

솔루션

NCP(링크의 호스트 쪽)에서 NUMILU 매개변수가 기본 값(0)으로 설정되었을 수 있습니다. 이를 확인하십시오. 변경 사항을 적용한 후 재시도하기 전에 필요한 경우 NCP 정의를 수정하십시오.

리턴 코드 9를 수반한 SQL30081N

징후

징후는 다음 메시지입니다(이 경우에는 SNA 감지 코드가 필요없습니다).

```
   db2 connect to <database> user <userid>
SQL30081N  A communication error has been detected.
   Communication protocol
   being used: "APPC".  Communication API being used: "CPI-C".
   Location where
   the error was detected: "".  Communication function detecting
   the error:
   "cmsend".  Protocol specific error code(s): "9", "*",
   "0x10086021".
   SQLSTATE=08001
```

솔루션

트랜잭션 프로그램 이름(TPNAME)이 DB2 Connect 시스템에서 올바르게 정의되지 않은 것이 문제입니다. 예를 들어, SNA 구성을 갱신했지만 DB2 Connect 게이트웨이에서 아직 확인하지는 않았을 수 있습니다. 자세한 내용은 *OS/2 및 Windows용 DB2 Connect Enterprise Edition 빠른 시작*이나 *DB2 Connect Personal Edition 빠른 시작* 매뉴얼을 참조하십시오.

리턴 코드 10을 수반한 SQL30081N

징후

징후는 다음 메시지입니다(SNA 감지 코드가 필요없습니다).

```
SQL30081N  A communication error has been detected.
   Communication protocol
   being used: "APPC".  Communication API being used: "CPI-C".
   Location where
   the error was detected: "".  Communication function detecting
   the error:
   "cmrcv".  Protocol specific error code(s): "10", "*", "*".
   SQLSTATE=08001
```

솔루션

DB2가 올바르게 설치되었는지 확인하십시오.

OS/2용 DB2 Connect 게이트웨이를 사용할 경우에, TP명이 제대로 정의되지 않으면 다음 사항이 표시될 수도 있습니다.

```
Protocol specific error code(s): "10", "*", "0x084C0000".
SQLSTATE=08001
```

예를 들어, CM/2에서는 이 경우 TP 이름이 다음과 같이 정의되어야 합니다.

```
트랜잭션 프로그램 이름          = 'tpname'    (사용자 정의)
OS/2 프로그램 경로와 파일 이름 = notused
```

그리고 그 다음 CM/2 구성 화면은 아래와 같아야 합니다.

```
Presentation type - background
Operation type - Queued, operator preloaded
```

리턴 코드 20을 수반한 SQL30081N

징후

```
SQL30081N  A communication error has been detected.
Communication protocol
being used: "APPC".  Communication API being used: "CPI-C".
Location where
the error was detected: "".  Communication function detecting
the error:
"xcstp".  Protocol specific error code(s): "20", "*", "*".
SQLSTATE=08001
```

솔루션

DB2 Connect 시스템에서 SNA 서브시스템이 시작되는지 확인하십시오.

리턴 코드 27을 수반한 SQL30081N

징후

메시지 SQL30081N이 리턴 코드 27 및 SNA 감지 코드 800Axxxx와 함께 표시됩니다.

솔루션

VTAM PIU(Path Information Unit)가 너무 큽니다.

리턴 코드 79를 수반한 SQL30081N

징후

```
SQL30081N  A communication error has been detected.
Communication protocol
being used: "TCP/IP".  Communication API being used: "SOCKETS".
Location
where the error was detected: "".  Communication function
detecting the error:
"connect".  Protocol specific error code(s): "79", "*", "*".
SQLSTATE=08001
```

솔루션

이 오류는 원격 클라이언트가 DB2 Connect 게이트웨이에 연결할 수 없는 경우에 발생할 수 있습니다. DB2 Connect 게이트웨이에서 호스트로 연결할 때도 발생할 수 있습니다.

1. DB2COMM 프로파일 변수가 DB2 Connect 게이트웨이에서 틀리게 설정되었을 수 있습니다. 이를 확인하십시오. 예를 들어, db2set db2comm=tcpip 명령은 AIX에서 DB2 Extended Enterprise Edition 수행시 sqllib/db2profile로 표시되어야 합니다.
2. DB2 클라이언트와 DB2 Connect 게이트웨이에서 TCP/IP 서비스 이름과 포트 번호 스펙간 또는 포트 번호 스펙들간에 불일치가 있을 수 있습니다. 두 머신의 TCP/IP 서비스 파일에 있는 항목을 확인하십시오.

3. DB2 Connect 게이트웨이에서 DB2가 시작되는지 확인하십시오. 다음 명령을 사용하여 데이터베이스 관리 프로그램 구성 diaglevel을 4로 설정하십시오.

   ```
   db2 update dbm cfg using diaglevel 4
   ```

   DB2를 정지하였다가 재시작한 후, DB2 TCP/IP 통신이 시작되었는지를 db2diag.log 파일에서 확인하십시오. 다음과 유사한 출력이 표시되어야 합니다.

   ```
   1998-02-03-12.41.04.861119   Instance:svtdbm2   Node:00
   PID:86496(db2sysc)   Appid:none
   common_communication  sqlcctcp_start_listen   Probe:80
   DIA3000I "TCPIP" protocol support was successfully started.
   ```

프로토콜 특정 오류 코드 10032를 수반한 SQL30081N

징후

```
SQL30081N  A communication error has been detected.
Communication protocol
being used: "TCP/IP".  Communication API being used: "SOCKETS".
Location
where the error was detected: "9.21.85.159".  Communication
function detecting
the error: "send".  Protocol specific error code(s): "10032",
"*", "*".
SQLSTATE=08001
```

솔루션

이 오류 메시지는 TCP/IP 통신이 이미 실패한 머신으로부터 연결을 끊으려 할 때 나타날 수 있습니다. TCP/IP 서브시스템의 문제를 정정하십시오.

대부분의 머신에서 TCP/IP 프로토콜을 재시작하기만 하면 문제가 정정됩니다. 전체 머신을 다시 순환시켜야 할 수도 있습니다.

제3부 부록 및 끝머리

© Copyright IBM Corp. 1993, 2000

부록A. 이전 릴리스에서 제공되는 기능

최신 버전과 릴리스를 시작할 때 각 버전과 릴리스에 도입된 향상 기능의 요약이 표시됩니다.

DB2 Connect 버전 6 릴리스 1

DB2 Connect 버전 6.1은 다음과 같이 향상된 사항이 있습니다.

- TCP/IP 통신 프로토콜 사용
- 2단계 확약
- 여러 행 저장 프로시듀어
- DCE 보안
- DCE 셀 디렉토리 지원 및 호스트 시스템
- 향상된 보안 실패 통지
- 향상된 System/390 SYSPLEX 이용
- ODBC와 JDBC 응용프로그램의 카탈로그 액세스 최적화
- 새로운 BIND 옵션
- Microsoft 트랜잭션 서버 지원
- 간편해진 암호 관리
- 클라이언트 정보 향상
- 양방향 언어 지원
- DB2 Connect 응용프로그램 모니터링
- 2단계 확약 향상
- 간편해진 DB2 동기점 관리 프로그램 구성
- 추가적인 데이터베이스 및 지원된 유형
- 퍼스널 통신을 위한 DB2 Connect

© Copyright IBM Corp. 1993, 2000

DB2 Connect 버전 5 릴리스 2

DB2 Connect 버전 5.2에서는 다음과 같이 호스트와 AS/400 DRDA 기능이 개선되었습니다.

- DCE 셀 디렉토리 지원
- 향상된 보안 실패 통지
- 향상된 System/390 SYSPLEX 이용
- ODBC와 JDBC 응용프로그램의 카탈로그 액세스 최적화
- Microsoft 트랜잭션 서버 지원
- 새 BIND 옵션(DYNAMICRULES)
- 클라이언트 정보 설정 API 향상
- DB2 Connect의 SQLDescribeParam 지원
- 양방향 언어 지원
- 시스템 모니터 향상
- 2단계 확약 지원 향상
- 간편해진 DB2 동기점 관리 프로그램 구성
- SCO** 운영 체제 지원
- 큰 정수, 대형 오브젝트(LOB), 행 ID 및 사용자 정의 구별 데이터 유형 지원

DB2 Connect 버전 5.0

- 패키지 구입이 더 쉬워졌습니다.
  - 하나의 DB2 Connect Personal Edition 패키지에는 OS/2, Windows 3.1, Windows 95, Windows NT 버전 제품이 들어 있습니다. 이 패키지에는 Lotus Approach 증정본을 포함하여 시작에 필요한 모든 정보가 들어 있습니다.
  - 하나의 DB2 Connect Enterprise Edition 패키지에는 OS/2, Windows NT 및 모든 UNIX 버전이 들어 있습니다.
- 성능
  - 향상된 새로운 레벨 3 ODBC 드라이버

  - 더 나은 자바 지원을 위해 갱신된 JDBC 드라이버
  - 다중행 결과 집합과 다중 결과 집합을 리턴하는 저장 프로시듀어에 대한 지원(OS/390용 DB2 Universal Database버전 5.1 이상이 필요함)
  - 내장 복제 지원
  - 일반적인 바인드 옵션: 호스트 데이터베이스에 의해 지원되는 모든 바인드 옵션을 지정할 수 있습니다.
  - SYSPLEX 활용(DB2 Connect Personal Edition만 해당. OS/390용 DB2 Universal Database 버전 5.1 이상이 필요함)
- 가용성:
  - 새로운 설치 방법
  - TCP/IP 데이터베이스 연결 구성이 훨씬 쉽습니다(OS/390용 DB2 Universal Database 버전 5.1 이상이나 AS/400용 DB2 Universal Database 버전 4.2가 필요함).
  - 지정 후 클릭 구성의 통합 SNA 지원(DB2 Connect Personal Edition만 해당)
  - 호스트 연결을 구성하는 새로운 지정 후 클릭 구성 유틸리티.
  - 데스크탑 클라이언트 시스템을 DB2 Connect Enterprise Edition 서버에 연결하는 훨씬 쉬워진 프로세스. 클라이언트는 DB2 Connect 서버와 각 서버에서 정의된 모든 데이터베이스를 발견할 수 있습니다.
  - 성능 분석에 대한 세부 정보를 제공하는 향상된 ODBC 추적
  - 여러 DBA 타스크를 단순화한 제어 센터와 기타 GUI 도구
- 보안
  - DCE 보안(OS/390용 DB2 Universal Database 버전 5.1 이상이 필요함)
  - 기본 테이블에 대해 각 사용자에게 권한을 부여하지 않고 ODBC 응용프로그램을 수행할 수 있습니다. 사용자는 ODBC 드라이버를 바인드했던 개인의 권한 하에서 응용프로그램을 수행하는 방식으로 이제 ODBC 드라이버를 바인드할 수 있습니다.
- 성능:
  - ODBC 응용프로그램용 DB2 카탈로그로에 더 빠른 액세스

  - 감소된 네트워크 통신량
    - 커서 빨리 닫기
    - 지연된 준비
    - 복합 SQL에 대한 감소된 바이트 카운트
    - 여러 기타 네트워크 흐름 향상
    - 호스트에서 ASCII 저장영역 지원(OS/390용 DB2 Universal Database 버전 5.1 이상이 필요함)
- 연결성:
  - TCP/IP에 대한 지원의 도입과 함께 다른 IBM DRDA 응용프로그램 서버(AS)에 대한 TCP/IP 연결을 통한 DRDA의 지원.
  - 통합 MPTN 지원에 의해 TCP/IP를 경유하는 SNA(호스트에 AnyNet이 필요함)
  - 추가 SNA 연결 옵션에 대한 지원:
    - Windows NT용 IBM 통신 서버
    - IBM 퍼스널 통신
- 기타 사항:
  - TCP/IP를 통해 2단계 확약 트랜잭션을 시작할 수 있는 능력(OS/390용 DB2 Universal Database 버전 5.1 이상이 필요함)
  - 게이트웨이 없이도 데스크탑 응용프로그램이 2단계 확약 트랜잭션에 참여할 수 있는 능력(TCP/IP에만 해당. OS/390용 DB2 Universal Database 버전 5.1 이상이 필요함)
  - 트랜잭션 조정의 신뢰성이 추가되어 OS/390용 DB2 Universal Database를 사용할 수 있는 능력(OS/390용 DB2 Universal Database 버전 5.1 이상과 TCP/IP가 필요함)
  - 시스템 성능, 신뢰성, 가용성의 모든 면에 영향을 주는 기타 추가 처리 및 수정

DDCS 버전 2 릴리스 4

Windows용 분산 데이터베이스 연결 서비스(DDCS) 단일 사용자 버전 2.4에서 다음 기능이 도입되었습니다.

- 데이터 소스 설치 프로그램을 사용하면 호스트와 AS/400 서버에 대한 연결을 빠르고 쉽게 정의할 수 있습니다.
- Wall Data Rumba는 이런 연결 과정에 필요한 통신 지원을 제공합니다.
- DB2 암호 만기 유지보수 유틸리티(DB2PEM). TSO에 로그온하지 않고도 MVS/ESA용 DB2 암호를 변경할 수 있습니다.
- DB2 Connect의 성능 및 융통성을 향상시키는 개선 사항
  - 지연된 준비. PREPARE 요청을 개별 전송하는 대신 후속 요청에 첨부시킴으로써 ODBC와 다른 동적 SQL 응용프로그램의 성능을 향상시킵니다.
  - 비동기 ODBC. ODBC 응용프로그램의 가용성을 증진시킵니다. 이전에는 일부 네트워크 상황에서 길이가 긴 조회를 처리할 때 ODBC 응용프로그램이 지연되는 것처럼 보였을 수 있습니다.
  - AIX와 OS/2에서, 다중 스레드 응용프로그램에 대한 지원. ODBC가 아닌 응용프로그램이 해당 문맥과의 다중 데이터베이스 연결을 유지보수할 수 있게 합니다.

DDCS 버전 2 릴리스 3

DDCS 버전 2 릴리스 3.1에는 다음의 새로운 기능이 포함되어 있습니다.

- OS/2와 AIX에서 LU6.2 동기점 관리 프로그램(SPM)을 사용한 DRDA 연결을 위한 2단계 확약.

DDCS 버전 2 릴리스 3.0에는 다음의 새로운 기능이 포함되어 있습니다.

- 클라이언트 응용프로그램 성능은 MVS/ESA용 DB2 버전 4.1 및 AS/400용 DB2 Universal Database 버전 3.1 서버에서 저장 프로시듀어를 수행하여 향상될 수 있습니다. 58 페이지의 『저장 프로시듀어』에서 자세한 내용을 참조하십시오.
- 한 트랜잭션에서 많은 데이터베이스에 대해 작업할 수 있습니다.

- SQL문을 연결하여 성능을 향상시킬 수 있습니다. 61 페이지의 『NOT ATOMIC 복합 SQL』와 126 페이지의 『가져오기 및 내보내기 유틸리티 사용』에서 자세한 내용을 참조하십시오.
- 사용 통계 문자열로 사용 비용 청구를 구현할 수 있습니다. 65 페이지의 『OS/390용 DB2 Universal Database에서의 사용 비용 청구 계정 구현』에서 자세한 내용을 참조하십시오.
- 응용프로그램을 DRDA 응용프로그램 서버(AS)에 바인딩하면, 새로운 바인드 옵션을 많이 사용할 수 있습니다. 109 페이지의 『BIND 명령』에서 자세한 내용을 참조하십시오.
- DCE 디렉토리를 사용할 경우, 중앙 저장소에서 모든 사용자 클라이언트에 필요한 디렉토리 정보를 통합할 수 있습니다. 229 페이지의 『부록D. DCE 디렉토리 서비스 사용』에서 자세한 내용을 참조하십시오.
- SQLCODE 처리가 더욱 다양해졌습니다. 141 페이지의 『제11장 SQLCODE 맵핑』에서 자세한 내용을 참조하십시오.
- 진단 정보(첫번째 오류 서비스 로그)가 읽을 수 있는 형식으로 저장되고 한 위치에서 통합됩니다. *문제점 해결 안내서*에서 자세한 내용을 참조하십시오.
- DDCSSETP 환경 변수가 SQLERROR CONTINUE 같은 조작을 단순화하는 BIND 및 PREPARE 옵션으로 바뀌었습니다.
- 기타 다양한 성능 개선 사항들이 구현되었습니다.

부록B. 디렉토리 사용자 정의 워크쉬트

이 워크쉬트를 사용하여 디렉토리를 사용자 정의하십시오. *Command Reference* 또는 100 페이지의『디렉토리 갱신』에서 명령 구문을 참조하십시오.

표 9. 노드 디렉토리 매개변수

매개변수	예	사용자 값
노드 이름	DB2NODE 또는 MVSIPNOD	
기호 목적지 이름(APPC 노드)	DB2CPIC	
원격 호스트 이름(TCP/IP 노드)	MVSHOST	
서버(TCP/IP 서비스 이름 또는 포트 번호)	db2inst1c (또는 446)	
보안 유형	APPC 노드에 대해 PROGRAM, TCP/IP 노드에 대해 NONE.	

주:

1. DRDA용 기본 TCP/IP 포트 번호는 446입니다.
2. 호스트나 AS/400 데이터베이스 서버가 SECURITY SOCKS를 지원하는지 모르면, TCP/IP 노드에 대해 SECURITY를 지정하지 마십시오.

표 10. DCS 디렉토리 매개변수

매개변수	예	사용자 값
데이터베이스 이름	DB2DB	
목표 데이터베이스 이름	NEW_YORK3	
응용프로그램 리퀘스터		
매개변수 문자열	",,,,,,LOCALDATE=\"\"YYMMDD\"\"\"	

표 11. 시스템 데이터베이스 디렉토리 매개변수

매개변수	예	사용자 값
데이터베이스 이름	DB2DB	
데이터베이스 별명	NYC3	
노드 이름	DB2NODE	
인증	DCS	

© Copyright IBM Corp. 1993, 2000

부록C. 자국어 지원에 대한 고려사항

DB2 Connect에는 다음의 자국어 지원(NLS) 고려사항이 있습니다.

- DB2 Connect 메시지는 임의의 언어로 번역되어 있습니다. 번역된 메시지를 액세스하는 데 대한 자세한 내용은 해당 플랫폼용 *빠른 시작*에서 참조하십시오.
- DB2 Connect는 많은 언어와 코드 페이지를 지원합니다. *관리 안내서*에서 이러한 코드 페이지 목록에 대한 내용을 참조하십시오.
- 데이터가 DB2 Connect와 호스트 또는 AS/400 데이터베이스 서버 사이에서 전송되면, 워크스테이션 코드 페이지로부터 호스트 CCSID로 변환되며 반대 경우도 마찬가지입니다. 『문자 데이터의 변환』에서 자세한 내용을 참조하십시오

다음을 포함하여 DB2 Connect 사용에 대한 자세한 내용이 *DB2 Connect 빠른 시작*에 있습니다.

- 날짜 및 시간 형식
- DB2 Connect Enterprise Edition과 DB2 Connect Personal Edition에서 지원하는 언어
- 특정한 자국어 환경용으로 DB2 Connect 워크스테이션을 사용자 정의하는 방법
- 호스트의 코드화 문자 세트 식별자(CCSID) 설정을 사용자 정의하는 방법

문자 데이터의 변환

문자 데이터가 머신들 사이에서 전송될 경우, 수신 머신이 사용할 수 있는 형식으로 문자 데이터를 변환해야 합니다.

예를 들어, 데이터가 DB2 Connect 워크스테이션과 호스트나 AS/400 데이터베이스 서버 사이에서 전송되면, 워크스테이션 코드 페이지로부터 호스트 CCSID로 변환되며 반대 경우도 마찬가지입니다. 두 머신이 서로 다른 코드 페이지 또는 CCSID를 사용할 경우, 한 코드 페이지 또는 CCSID에서 다른 것으로 코드 포인트가 대응됩니다. 이 변환은 항상 수신측에서 수행됩니다.

© Copyright IBM Corp. 1993, 2000

데이터베이스로 전송된 문자 데이터는 SQL문과 입력 데이터로 이루어집니다 데이터베이스*로부터* 전송된 문자 데이터는 출력 데이터로 이루어집니다. 비트 데이터로 해석된 출력 데이터(2진 데이터용 절로 나타난 컬럼으로부터의 데이터)는 변환되지 않습니다. 그렇지 않으면, 두 머신의 코드페이지 또는 CCSID가 서로 다를 경우, 입력 및 출력 문자 데이터가 모두 변환됩니다.

예를 들어, DB2 Connect가 OS/390용 DB2 Universal Database 또는 DB2/MVS 데이터를 액세스하는 데 사용되면, 다음 경우가 발생합니다.

1. DB2 Connect는 SQL문과 입력 데이터를 OS/390 또는 MVS로 송신합니다.
2. OS/390용 DB2 Universal Database는 데이터를 EBCDIC CCSID로 변환하여 처리합니다.
3. OS/390용 DB2 Universal Database는 그 결과를 다시 DB2 Connect 워크스테이션으로 송신합니다.
4. DB2 Connect는 결과를 ASCII 또는 ISO 코드 페이지로 변환시키고 사용자에게 리턴합니다.

다음 표는 코드 페이지(워크스테이션)와 CCSID(호스트) 간의 지원되는 변환을 보여줍니다.

지원되는 코드 페이지 변환에 대해서는 *관리 안내서*에서 자세한 내용을 참조하십시오.

표 12. 워크스테이션 코드 페이지에서 호스트 CCSID로의 변환

호스트 **CCSID**	코드 페이지	국가
037, 273, 277, 278, 280, 284, 285, 297, 500, 871, 1140-1149	437, 819, 850, 858, 860, 863, 1004, 1051, 1252, 1275	알바니아, 오스트레일리아,오스트리아, 벨기에, 브라질, 캐나다, 덴마크, 핀란드, 프랑스, 독일, 아이슬랜드,아일랜드, 이탈리아, 라틴 아메리카, 네델란드,뉴질랜드, 노르웨이, 포르투갈, 남아프리카,스페인, 스웨덴, 스위스, 영국, 미국
423, 875	737, 813, 869, 1253, 1280	그리스

표 12. 워크스테이션 코드 페이지에서 호스트 CCSID로의 변환 (계속)

호스트 CCSID	코드 페이지	국가
870	852, 912, 1250, 1282	크로아티아, 체코공화국, 헝가리, 폴란드, 루마니아, 세르비아/몬테니그로(라틴), 슬로바키아, 슬로베니아
1025	855, 866, 915, 1251, 1283	불가리아, FYR 마케도니아, 러시아, 세르비아/몬테니그로(시릴)
1026	857, 920, 1254, 1281	터키
424	862, 916, 1255	이스라엘 - 아래의 주 3 참조
420	864, 1046, 1089, 1256	아랍 국가 - 아래의 주 3 참조
838	874	태국
930, 939, 5026, 5035	932, 942, 943, 954, 5039	일본
937	938, 948, 950, 964	대만
933, 1364	949, 970, 1363	한국
935, 1388	1381, 1383, 1386	중국
1112, 1122	921, 922	에스토니아, 라트비아, 리투아니아
1025	915, 1131, 1251, 1283	벨라루스
1123	1124, 1125, 1251	우크라이나

주:

1. 코드 페이지 1004는 코드 페이지 1252로 지원됩니다.
2. 일반적으로, 코드 페이지로부터 CCSID로 데이터가 변환되고, 변경없이 같은 코드 페이지로 다시 변환될 수 있습니다. 다음은 이 규칙에 대한 유일한 예외 조항입니다.
   - 2바이트 문자 세트(DBCS) 코드 페이지에서, 사용자 정의 문자가 들어 있는 일부 데이터가 손실될 수 있습니다.
   - 혼합 바이트 코드 페이지 내에 정의된 1바이트 코드 페이지의 경우와 일부 새로운 1바이트 코드 페이지의 경우, 소스 코드 페이지와 목표 코드 페이지에 없는 문자는 대체 문자에 맵핑된 다음 데이터가 다시 원래 코드 페이지로 변환될 때 손실될 수 있습니다.
3. 양방향 언어의 경우, IBM은 다수의 특수 ″BiDi CCSIDS″를 정의하고 있으며 이는 DB2 Connect 버전 7에서 지원합니다.

데이터베이스 서버의 양방향 속성이 클라이언트의 속성과 다른 경우, 이 특수 CCSIDS를 사용하여 차이점을 관리할 수 있습니다.

이러한 CCSID에 대해서는 *관리 안내서*에서 자세한 내용을 참조하십시오. DRDA 호스트 연결을 위해 이 특수 CCSID를 설정하는 방법에 관한 자세한 내용은 DB2 Connect 버전 7 릴리스 정보를 참조하십시오.

부록D. DCE 디렉토리 서비스 사용

DCE 셀 디렉토리 서비스(CDS)를 사용하여, 각 클라이언트에 서버 정보를 저장하지 않고 CDS에 서버 정보를 저장할 수 있습니다. CDS는 모든 플랫폼에서 모든 DB2 Universal Database 클라이언트와 DB2 Connect Enterprise Edition에 대해 지원됩니다.

주: DB2 Connect의 DCE 셀 디렉토리 서비스 지원을 사용하여 SNA 연결을 통해 MVS/ESA용 DB2로 연결하려면, 원격 트랜잭션 프로그램 이름(RTPN)으로 DB2DRDA 사용을 지원하는 MVS/ESA용 DB2 PTF UN73393을 적용해야 합니다.

DCE 디렉토리를 사용하려면, 다음을 작성해야 합니다.

- *데이터베이스 오브젝트* - 데이터베이스 정보가 들어 있음.
- *데이터베이스 위치 지정자 오브젝트* - 원격 클라이언트와 DB2 Connect 워크스테이션 간의 연결 정보가 들어 있음.
- *라우팅 정보 오브젝트* - 데이터베이스 오브젝트를 데이터베이스 위치 지정자 오브젝트와 일치시킴.

이 오브젝트를 작성하기 전에 액세스할 호스트나 AS/400 데이터베이스 서버에서 다음을 수행해야 합니다.

- 다음의 DCE 속성이 오브젝트를 작성한 워크스테이션의 cds 속성 파일에 추가되었는지 확인하십시오.

  AIX 시스템의 경우

  파일 이름은 */etc/dce/cds_attributes*입니다.

  OS/2 시스템의 경우

  파일 이름은 *x:\opt\dcelocal\etc\cds_attr*이며 *x:*는 드라이브 이름입니다.

  Windows 32-비트 시스템의 경우

  파일 이름은 *x:\root\dcelocal\etc\cds_attributes*이며 *x:*는 드라이브 이름이고 *root*는 DCE가 설치된 디렉토리입니다.

© Copyright IBM Corp. 1993, 2000

```
1.3.18.0.2.4.30 DB_Comment                 char
1.3.18.0.2.4.31 DB_Communication_Protocol  char
1.3.18.0.2.4.32 DB_Database_Protocol       char
1.3.18.0.2.4.33 DB_Database_Locator_Name   char
1.3.18.0.2.4.34 DB_Native_Database_Name    char
1.3.18.0.2.4.35 DB_Object_Type             char
1.3.18.0.2.4.36 DB_Product_Name            char
1.3.18.0.2.4.37 DB_Product_Release         char
1.3.18.0.2.4.38 DB_Target_Database_Info    char
1.3.18.0.2.4.39 DB_Authentication          char
1.3.18.0.2.4.63 DB_Principal               char
```

- 오브젝트를 작성하는데 충분한 권한을 가지고 DCE로 로그인했는지 확인하십시오. 다음 DCE 명령을 사용하여 UNIX나 Windows 2000 시스템에 로그인할 수 있습니다.

  ```
  dce_login principal-id password
  ```

  다음 DCE 명령을 사용하여 OS/2 시스템에 로그인할 수 있습니다.

  ```
  dcelogin principal-id password
  ```

주: 이 오브젝트를 사용하여 데이터베이스에 연결하려면, 호스트나 AS/400 데이터베이스 서버 및 워크스테이션에서 통신을 구성해야 합니다. 해당 *빠른 시작* 책에서 자세한 내용을 참조하십시오.

데이터베이스 오브젝트 작성

*데이터베이스 오브젝트*는 호스트나 DB2 Connect의 AS/400 데이터베이스 서버를 정의하는데, 이 오브젝트는 항상 정의되어야 합니다. 액세스할 각 호스트나 AS/400 데이터베이스 서버에서, DCE 명령 **cdscp create object**를 사용하여 데이터베이스 오브젝트를 작성하십시오 예를 들면, 다음과 같습니다.

```
cdscp create object database_global_name
```

다음 속성을 해당 오브젝트에 추가하십시오.

DB_Object_Type
데이터베이스의 경우 D

DB_Product_Name
관계형 데이터베이스 제품(예: DB2_for_MVS 또는 DB2_for_OS390)

DB_Native_Database_Name

다음과 같은 호스트나 AS/400 데이터베이스 서버 상의 데이터베이스 이름

MVS 또는 OS/390

LOCATION 값

VSE 또는 VM

데이터베이스 이름

OS/400

관계형 데이터베이스 이름

DB_Database_Protocol

DRDA

DB_Authentication

SERVER, CLIENT 또는 DCE. 237 페이지의 『DCE 디렉토리 서비스에서의 보안』에 설명되어 있습니다.

DB_Principal

인증 방식이 DCE인 경우, 이 속성에 DCE 주요 서버 이름을 입력하십시오.

DB_Communication_Protocol

다음은 DB2 Connect 서버와 호스트나 AS/400 데이터베이스 서버 간의 통신 프로토콜에 대한 정보입니다.

- 통신 프로토콜 APPC의 경우:
  1. 통신 프로토콜(APPC)
  2. 호스트나 AS/400 데이터베이스 서버의 네트워크 ID
  3. 호스트나 AS/400 데이터베이스 서버에 대한 LU 이름
  4. 호스트나 AS/400 데이터베이스 서버에 연결하기 위한 트랜잭션 프로그램 이름. MVS/ESA용 DB2에서, DB2DRDA를 지정하십시오. 다른 운영 체제에서는, 16진수 형식이 아닌 유효한 값을 지정하십시오.
  5. 모드 이름

6. 보안 유형. 237 페이지의 『DCE 디렉토리 서비스에서의 보안』에 설명되어 있습니다. 예를 들면, 다음과 같습니다.

   ```
   APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;PROGRAM
   ```

- 통신 프로토콜 TCPIP의 경우:
  1. 통신 프로토콜(TCPIP)
  2. 목적지 TCP/IP 호스트 이름(호스트나 AS/400 데이터베이스 서버의 경우)
  3. TCP/IP 포트 번호
  4. 연결 유형(SOCKS 또는 NONE 사용). 선택적입니다. 지정되지 않으면, NONE이 사용됩니다. 예를 들어, 다음은 통신 프로토콜 TCP/IP의 속성 값입니다.

     ```
     tcpip;jaguar;19713;NONE
     ```

시스템 보안 기능을 갖는 데이터베이스 오브젝트를 작성하려면, 다음 명령어를 파일에 입력하십시오.

```
create object /.../cdscell1/subsys/database/DBMVS01
add    object /.../cdscell1/subsys/database/DBMVS01 DB_Object_Type=D
add    object /.../cdscell1/subsys/database/DBMVS01 DB_Product_Name=DB2_for_MVS
add    object /.../cdscell1/subsys/database/DBMVS01 DB_Database_Protocol=DRDA
add    object /.../cdscell1/subsys/database/DBMVS01 DB_Native_Database_Name=\
                  NEW_YORK
add    object /.../cdscell1/subsys/database/DBMVS01 DB_Authentication=SERVER
add    object /.../cdscell1/subsys/database/DBMVS01 DB_Communication_Protocol=\
                  APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;PROGRAM
```

그리고 다음 명령을 입력하십시오.

```
cdscp < 파일 이름
```

주: 파일에서, 명령문을 다음 라인에서 계속하려면 백슬래시(\)를 지정하십시오.

DCE 보안으로 데이터베이스 오브젝트를 작성하기 위해, 다음 명령어를 파일에 입력하십시오.

```
create object /.../cdscell1/subsys/database/DBMVS02
add    object /.../cdscell1/subsys/database/DBMVS02 DB_Object_Type=D
add    object /.../cdscell1/subsys/database/DBMVS02 DB_Product_Name=DB2_for_MVS
add    object /.../cdscell1/subsys/database/DBMVS02 DB_Database_Protocol=DRDA
add    object /.../cdscell1/subsys/database/DBMVS02 DB_Native_Database_Name=\
                  NEW_YORK
add    object /.../cdscell1/subsys/database/DBMVS02 DB_Authentication=DCE
add    object /.../cdscell1/subsys/database/DBMVS02 DB_Principal=\
```

```
                /.../cdscell1/principal_name
add    object /.../cdscell1/subsys/database/DBMVS02 DB_Communication_Protocol=\
                  APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;NONE
```

그리고 다음 명령을 입력하십시오.

```
cdscp < 파일 이름
```

데이터베이스 위치 지정자 오브젝트 작성

*데이터베이스 위치 지정자 오브젝트*를 사용하여 해당 클라이언트에 DB2 Connect Enterprise Edition 서버를 정의합니다. DB2 Connect 워크스테이션에서, DCE 명령 **cdscp create object**를 사용하여 데이터베이스 위치 지정자 오브젝트를 작성하십시오 예를 들면, 다음과 같습니다.

```
cdscp create object 전역 오브젝트 이름
```

다음 속성을 해당 오브젝트에 추가하십시오.

DB_Object_Type

위치 지정자 오브젝트의 경우

DB_Communication_Protocol

다음은 DB2 Connect 워크스테이션과 원격 클라이언트 간의 각 통신 프로토콜 설정에 필요한 정보입니다.

다음은 플랫폼별로 프로토콜 지원을 요약한 것입니다.

- OS/2: APPC, IPX, NETBIOS 및 TCP/IP
- Windows 32비트 운영 체제: APPC, IPX, NETBIOS, NPIPE 및 TCP/IP
- UNIX: APPC와 TCP/IP

APPC의 경우:

1. 통신 프로토콜(APPC)
2. DB2 Connect 워크스테이션의 네트워크 ID
3. DB2 Connect 워크스테이션용 LU 이름
4. 원격 클라이언트로부터의 연결용 트랜잭션 프로그램 이름
5. 모드 이름

6. 보안 유형. 237 페이지의 『DCE 디렉토리 서비스에서의 보안』에 설명되어 있습니다.

TCP/IP의 경우:

1. 통신 프로토콜(TCPIP)
2. DB2 Connect 워크스테이션의 호스트 이름
3. 원격 클라이언트로부터 연결을 승인하기 위해 DB2 Connect 워크스테이션이 사용하는 연결 포트
4. 연결 유형(SOCKS 또는 NONE 사용). 선택적입니다. 지정되지 않으면, NONE이 사용됩니다.

IPX/SPX의 경우:

1. 통신 프로토콜(IPXSPX)
2. 파일 서버 이름. 직접 주소 지정에는 *를 사용하십시오.
3. 오브젝트 이름. 직접 주소 지정에는 인터네트워크 주소를 사용하십시오.

Named Pipes의 경우:

1. 통신 프로토콜(NPIPE)
2. DB2 Connect 워크스테이션의 컴퓨터 이름
3. 인스턴스 이름

NETBIOS의 경우:

1. 통신 프로토콜(NETBIOS)
2. 서버나 또는DB2 Connect Enterprise Edition 게이트웨이의 NNAME.

예를 들어, 다음 라인을 파일에 입력할 수 있습니다.

```
create object /.../cdscell1/subsys/database/DBAIX01
add    object /.../cdscell1/subsys/database/DBAIX01 DB_Object_Type= L
add    object /.../cdscell1/subsys/database/DBAIX01 DB_Communication_Protocol=\
          TCPIP;AIX001;3700
add    object /.../cdscell1/subsys/database/DBAIX01 DB_Communication_Protocol=\
          APPC;SPIFNET;NYX1GW01;NYSERVER;IBMRDB;NONE
```

그리고 다음 명령을 입력하십시오.

```
cdscp < 파일 이름
```

Windows 32비트 운영 체제에서 유사한 방식으로 Named Pipe를 지정할 수 있습니다. 예를 들면, 다음과 같습니다.

```
add    object /.../cdscell1/subsys/database/DBAIX01 DB_Communication_Protocol=\
           NPIPE;computer_name;instance_name
```

OS/2에서 DB_Communication_Protocol 속성으로 프로토콜을 지정할 수 있습니다. 예를 들면, 다음과 같습니다.

```
- IPXSPX;fileserver;object name
- NETBIOS;nname
```

라우팅 정보 오브젝트 작성

*라우팅 정보 오브젝트*는 DCE에서 정의해야 하고, DB2 클라이언트가 검색합니다. DCE 명령 **cdscp create object**를 사용하여 라우팅 정보 오브젝트를 작성하십시오 예를 들면, 다음과 같습니다.

```
cdscp create object 전역 오브젝트 이름
```

R의 **DB_Object_Type** 속성을 추가하십시오.

데이터베이스 오브젝트마다 하나의 **DB_Target_Database_Info** 속성을 추가하십시오. 각 **DB_Target_Database_Info** 속성은 다음 매개변수로 구성되어 있습니다.

데이터베이스

전체 경로를 포함하는 데이터베이스 오브젝트 이름. *OTHERDBS를 지정하여 명시적으로 나타나지 않은 다른 데이터베이스 모두를 나타내십시오.

아웃바운드 프로토콜

호스트나 AS/400 데이터베이스 서버 연결을 위한 데이터베이스 프로토콜(DRDA)

인바운드 프로토콜

원격 클라이언트 연결용 데이터베이스 프로토콜(DB2RA)

게이트웨이 인증

237 페이지의 『DCE 디렉토리 서비스에서의 보안』에서 설명된 것처럼 0(아니오) 또는 1(예).

게이트웨이용 매개변수 문자열

게이트웨이에서 사용된 매개변수가 들어 있는 문자열. 그 내용은 특정한 게이트웨이입니다. DB2 Connect 특유의 게이트웨이 문자열에 대해서는 89 페이지의 『DCS 디렉토리』에서 자세한 내용을 참조하십시오.

데이터베이스 위치 지정자

DB2 Connect 워크스테이션을 나타내는 데이터베이스 위치 지정자 오브젝트 이름

예를 들어, 다음 라인을 파일에 입력할 수 있습니다.

```
create object /.../cdscell1/subsys/database/ROUTE1
add    object /.../cdscell1/subsys/database/ROUTE1 DB_Object_Type=R
add    object /.../cdscell1/subsys/database/ROUTE1 DB_Target_Database_Info=\
/.../cdscell1/subsys/database/DBMVS01;DRDA;DB2RA;0;;\
/.../cdscell1/subsys/database/DBAIX01
add    object /.../cdscell1/subsys/database/ROUTE1 DB_Target_Database_Info=\
*OTHERDBS;DRDA;DB2RA;0;;\
/.../cdscell1/subsys/database/DBAIX02
```

그리고 다음 명령을 입력하십시오.

```
cdcsp < 파일 이름
```

구성 매개변수 설정

다음처럼 클라이언트의 데이터베이스 관리 프로그램 구성을 갱신하십시오.

```
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING
[DIR_PATH_NAME 경로]
DIR_OBJ_NAME loc_obj
DIR_TYPE DCE
[ROUTE_OBJ_NAME route_obj]
[DFT_CLIENT_COMM 프로토콜]
[DFT_CLIENT_ADPT 0-15]
```

여기서,

- *경로*는 목표 데이터베이스의 완전한 이름을 형성하는 데 사용된 기본 경로 이름입니다(기본값 /.:/subsys/database/).
- *loc_obj*를 사용하여 DCE 이름 공간의 클라이언트를 식별합니다.
- DIR_TYPE DCE는 DCE 디렉토리가 클라이언트 응용프로그램에 의해 사용되는 것을 지정합니다.

- *route_obj*는 라우팅 정보 오브젝트 이름입니다(예: ROUTE1).
- 프로토콜은 클라이언트와 DB2 Connect 워크스테이션(APPC나 UNIX용 TCPIP, APPC, IPXSPX, NETBIOS 또는 OS/2용 TCP/IP; Windows 32비트 운영 체제의 경우 APPC, TCPIP, IPXSPX, NETBIOS 및 NPIPE) 간의 통신 프로토콜입니다.
- NetBIOS에 대해 0에서 15까지의 기본 클라이언트 어댑터. 프로토콜이 NETBIOS이고 클라이언트 어댑터 번호가 기본 값 0이 아닌 경우에는, 클라이언트 어댑터 번호를 지정하십시오.

주: 다음의 환경 변수는 위에 나열된 내용을 각기 겹쳐쓸 수 있습니다.

- DB2DIRPATHNAME은 DIR_PATH_NAME을 겹쳐쓸 수 있습니다
- DB2ROUTE은 ROUTE_OBJ_NAME을 겹쳐쓸 수 있습니다
- DB2CLIENTCOMM은 DFT_CLIENT_COMM을 겹쳐쓸 수 있습니다
- DB2CLIENTADPT는 DFT_CLIENT_ADPT를 겹쳐쓸 수 있습니다

데이터베이스 카탈로그화

데이터베이스가 기본 값과 다른 경로에 있는 경우 또는 데이터베이스 이름과 다른 별명을 사용하려는 경우, 전역 데이터베이스를 카탈로그화할 수 있습니다. 명령행 처리기 CATALOG GLOBAL DATABASE 명령을 다음처럼 사용할수 있습니다.

```
db2 CATALOG GLOBAL DATABASE 전역 데이터베이스 이름
    AS alias USING DIRECTORY DCE
```

이 별명은 데이터베이스를 액세스하는 다른 응용프로그램에서 사용될 수 있습니다.

예:

```
db2 CATALOG GLOBAL DATABASE /.../cdscell2/subsys/database/dbmvs12 AS NYC3
    USING DIRECTORY DCE
```

DCE 디렉토리 서비스에서의 보안

DB2 Connect 관리자로서, 사용자 이름과 암호가 유효한 위치를 결정할 수 있습니다. DCE 디렉토리에서, 다음을 설정하여 실행할 수 있습니다.

- DB2 Connect 워크스테이션을 나타내는 데이터베이스 위치 지정자 오브젝트에서 통신 프로토콜의 보안 유형. 보안 유형 NONE을 사용하십시오.
- 데이터베이스 오브젝트의 인증 유형
- 데이터베이스 오브젝트에서 통신 프로토콜의 보안 유형
- 라우팅 정보 오브젝트의 게이트웨이 인증 매개변수

표13과 239 페이지의 표14에서는 이 값들의 가능한 결합과 각 결합에 대한 유효성이 실행되는 위치를 보여줍니다. 이 표에 나타난 결합만을 DCE 디렉토리 서비스에서 DB2 Connect가 지원합니다.

표 13. APPC 연결을 사용한 DCE에서 유효한 보안 시나리오

	서버의 데이터베이스 오브젝트		라우팅 오브젝트	유효성 점검
경우	인증	보안	**DB2 Connect 게이트웨이에서 인증 (1=맞음, 0=틀림)**	
1	CLIENT	SAME	0	원격 클라이언트(또는 DB2 Connect 워크스테이션)
2	CLIENT	SAME	1	DB2 Connect 워크스테이션
3	SERVER	PROGRAM	0	호스트 또는 AS/400 데이터베이스 서버
4	SERVER	PROGRAM	1	DB2 Connect 워크스테이션 및 호스트나 AS/400 데이터베이스 서버
5	DCE	NONE	N/A	DCE 보안 서버에서

주: 원격 클라이언트를 APPC를 통하여 DB2 Connect Enterprise Edition 게이트웨이 워크스테이션에 연결하려면, 게이트웨이의 DCE 위치 지정자 오브젝트에서 NONE 보안 유형을 지정하십시오.

표 14. TCP/IP 연결을 사용한 DCE에서 유효한 보안 시나리오

경우	서버의 데이터베이스 오브젝트	라우팅 오브젝트	유효성 점검
	인증	**DB2 Connect Enterprise Edition 게이트웨이에서 인증(1=맞음, 0=틀림)**	
1	CLIENT	0	원격 클라이언트(또는 DB2 Connect 워크스테이션)
2	CLIENT	1	DB2 Connect 워크스테이션
3	SERVER	0	호스트 또는 AS/400 데이터베이스 서버
4	N/A	N/A	없음
5	DCE	N/A	DCE 보안 서버에서

각 결합은 다음과 같이 자세히 설명되어 있습니다.

- 첫 번째 경우에, 사용자 이름과 암호가 원격 클라이언트에서만 유효합니다. (지역 클라이언트의 경우는, 사용자 이름과 암호가 DB2 Connect 워크스테이션에서만 유효합니다.)

  사용자는 처음으로 시작(sign on)하는 위치에서 인증될 것입니다. 사용자 ID는 네트워크를 통하여 송신되지만 암호는 그렇지 않습니다. 모든 클라이언트 워크스테이션이 적합한 보안 기능을 가지고 있을 경우에만, 이 보안 유형을 사용하십시오.
- 두 번째 경우에, 사용자 이름과 암호는 DB2 Connect 워크스테이션에서만 유효합니다. 암호는 네트워크를 통해 원격 클라이언트에서 DB2 Connect 서버로 송신되며 호스트나 AS/400 데이터베이스 서버로는 송신되지 않습니다.
- 세 번째 경우에, 사용자 이름과 암호는 호스트나 AS/400 데이터베이스 서버에서만 유효합니다. 암호는 네트워크를 통해 원격 클라이언트에서 DB2 Connect 서버로, DB2 Connect 워크스테이션에서 호스트나 AS/400 데이터베이스 서버로 송신됩니다.
- 네 번째 경우에, 사용자 이름과 암호는 DB2 Connect 워크스테이션과 호스트나 AS/400 데이터베이스 서버 모두에서 유효합니다. 암호는 네트워크를 통해

원격 클라이언트에서 DB2 Connect 서버로, DB2 Connect 서버에서 호스트나 AS/400 데이터베이스 서버로 송신됩니다.

유효성 점검이 두 군데에서 실행되므로, 동일한 사용자 이름과 암호 세트는 DB2 Connect 워크스테이션과 호스트 또는 AS/400 데이터베이스 서버에서 유지보수되어야 합니다.

- 다섯 번째 경우에, DCE 티켓은 DCE 보안 서버로부터 확보됩니다.

주:

1. AIX 시스템에서, 보안 유형 SAME을 사용하는 사용자 모두는 AIX **system** 그룹에 있어야 합니다.
2. 원격 클라이언트가 있는 AIX 시스템의 경우, DB2 Connect 서버에서 수행중인 DB2 Connect 제품의 인스턴스는 AIX **system** 그룹에 속해야 합니다.
3. 호스트나 AS/400 데이터베이스 서버에 대한 액세스는 자체 보안 메커니즘이나 서브시스템에 의해 제어됩니다. 예를 들어, 가상 원격 통신 액세스 방식(VTAM)과 자원 액세스 제어 기능(RACF)이 있습니다. 보호된 데이터베이스 오브젝트로의 액세스는 SQL **GRANT** 및 **REVOKE** 명령문으로 제어됩니다.

부록E. 백 레벨 클라이언트용 유틸리티 바인딩

이전 릴리스의 원격 클라이언트가 있으면, 해당 클라이언트의 유틸리티를 호스트나 AS/400 데이터베이스 서버로 바인드해야 합니다.

- 이전의 클라이언트가 동일한 호스트나 AS/400 데이터베이스 서버에 대해 DB2 Connect의 이전 릴리스와 함께 사용될 경우, 추가 단계를 수행할 필요는 없습니다.
- 이전의 클라이언트가 DB2 Connect와 함께 사용되지 않았으면(예를 들어, 여러 개의 OS/2 머신이 호스트나 AS/400 데이터베이스 서버에 연결되지 않은 상태로 연결되었으면), 다음 단계를 수행하십시오.

  1. OS/2용 DB2 버전 1.0 또는 1.2 클라이언트가 있으면, 다음 라인을 포함하여 바인드 목록 파일을 작성하십시오.

     sqlabind.bnd+
     sqlueiwi.bnd+
     sqluigsi.bnd+
     sqluiici.bnd+
     sqluiict.bnd+
     sqluexpm.bnd+
     sqluimpm.bnd+
     sqlurexp.bnd+
     sqlarxcs.bnd+
     sqlarxrr.bnd+
     sqlarxur.bnd

     그리고 사용자의 클라이언트 중 하나에서 DB2 Connect 워크스테이션으로 이 바인드 파일을 복사하십시오.

  2. CAE 버전 1.0 또는 1.2가 있는 경우, 다음 행을 사용하여 바인드 목록 파일을 작성하십시오.

© Copyright IBM Corp. 1993, 2000

db2ajgrt.bnd+
db2clics.bnd+
db2clpcs.bnd+
db2clprr.bnd+
db2clpur.bnd+
db2ueiwi.bnd+
db2uigsi.bnd+
db2uiici.bnd+
db2uiict.bnd+
db2uexpm.bnd+
db2uimpm.bnd+
db2urexp.bnd

그리고 사용자의 클라이언트 중 하나에서 DB2 Connect 워크스테이션으로 이 바인드 파일을 복사하십시오.

3. DB2 Connect 서버에서, 각 바인드 목록 파일을 각 호스트나 AS/400 데이터베이스 서버 데이터베이스에 바인드하십시오. 다음과 같은 명령을 실행하십시오.

```
db2 connect to DBALIAS user USERID using PASSWORD
db2 bind path@bindfile.lst blocking all
      sqlerror continue messages bindfile.msg grant public
db2 connect reset
```

여기서 *DBALIAS*, *USERID* 및 *PASSWORD*는 호스트나 AS/400 데이터베이스 서버 데이터베이스에 적용되고 *bindfile*은 바인드 목록 파일의 이름이며 *path*는 바인드 목록 파일의 위치입니다.

바인드 명령의 **권한 부여** 옵션을 사용하면 EXECUTE 특권을 PUBLIC 또는 지정된 사용자 이름, 그룹 ID에 권한 부여할 수 있습니다. 바인드 명령의 **권한 부여** 옵션을 사용하지 않으면, 각 패키지에 대해 개별적으로 GRANT EXECUTE(RUN)해야 합니다.

바인드 파일용 패키지 이름을 찾으려면, 다음 명령을 입력하십시오.

```
ddcspkgn @bindfile.lst
```

부록F. CLISCHEMA 키워드를 사용한 CLI/ODBC 응용프로그램 성능 조정

이 절에는 CLISCHEMA 초기화 키워드를 사용하여 ODBC/CLI 응용프로그램의 성능 조정을 도와주는 새 정보가 들어 있습니다. 네트워크나 데이터베이스의 성능 조정에 관한 일반적인 정보는 들어 있지 않습니다(147 페이지의 『제12장 성능』 참조). 다음은 앞으로 이 절에서 설명할 내용을 요약한 것입니다.

- 『목표 환경』
- 244 페이지의 『CLI/ODBC』
- 244 페이지의 『DB2 CLISCHEMA 초기화 키워드』
- 247 페이지의 『바람직한 접근 방법』
- 248 페이지의 『추가 힌트와 추가 정보』
- 249 페이지의 『db2ocat 카탈로그 최적화 알고리즘 도구』
- 249 페이지의 『추가 정보 소스』

목표 환경

여기에서 설명하는 내용은 주로 OS/390용 DB2 Universal Database 사용자를 위한 것이며 목표 환경은 아래와 같이 구성되어 있습니다.

- DB2 Universal Database 클라이언트에서 수행되는 CLI/ODBC 응용프로그램
- DB2 Connect 버전 5 이상(Personal Edition 또는 Enterprise Edition)
- OS/390용 DB2 Universal Database 버전 5.1 이상(또는 달리 명시되는 곳은 제외하고 MVS/ESA용 DB2)

© Copyright IBM Corp. 1993, 2000

CLI/ODBC

CLI/ODBC는 사용자의 데이터베이스 응용프로그램에서 호출할 수 있는 SQL API입니다. 이것은 데이터베이스 함수에서 호출할 때 동적 SQL문을 전달합니다. Embedded SQL과는 달리 CLI/ODBC에는 호스트 변수나 사전 처리 컴파일러가 필요없습니다.

응용프로그램에서 CLI/ODBC를 호출할 때, 처음으로 목표 데이터베이스에 있는 일부 시스템 카탈로그 테이블을 대상으로 SQL을 호출합니다. CLI/ODBC 응용프로그램은 항상 이러한 방식으로 시스템 카탈로그 테이블을 액세스합니다. 연결할 데이터베이스에 관한 정보를 수집하기 위해 호출할 수 있는 API는 10개입니다. 다음은 이러한 API 호출입니다.

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns.

이 API 호출과 이들이 참조하는 테이블에 관한 자세한 내용은 *CLI Guide and Reference*에 있습니다.

데이터베이스에 연결할 때 CLI/ODBC 응용프로그램은 그 데이터베이스에 있는 모든 데이터베이스 테이블에 관한 정보를 얻기 위해 기본 값으로 시스템 카탈로그 테이블을 조회합니다. 특히 대형 시스템에서는 이것이 대량의 네트워크 통신을 일으켜 응용프로그램을 시작할 때 상당한 지연을 초래할 수 있습니다.

DB2 CLISCHEMA 초기화 키워드

DB2 Universal Database 제품은 데이터베이스에 처음 연결된 후 "정보 수집" 단계에서 최초의 API 호출에 의해 리턴될 데이터의 양을 제한하는 데 사용할 수 있는 몇 개의 CLI/ODBC 초기화 키워드를 제공합니다. 이 키워드들은 아래의 방법으로 설정할 수 있습니다.

1. db2cli.ini 파일을 직접 편집
2. 클라이언트 구성 지원 프로그램을 사용하는 데이터베이스의 ODBC/CLI 설정 값 변경(클라이언트 구성 지원 프로그램을 지원하는 플랫폼에서)
3. DBA 명령행 인터페이스를 사용하는 데이터베이스 CLI 구성 갱신

다음은 키워드입니다.

- DBNAME
- TABLETYPE
- SCHEMALIST
- SYSSCHEMA
- CLISCHEMA

CLISCHEMA에 대한 정보는 제외하고 이 키워드들은 CLI/ODBC 도움말과 *CLI Guide and Reference*에 설명되어 있습니다 이 절의 나머지 부분에서는 CLISCHEMA의 사용에 관해서만 설명합니다.

다음은 정당한 순서를 밟아서 *CLI Guide and Reference*에 추가될 CLISCHEMA에 대한 문서 내용입니다.

```
db2cli.ini Keyword Syntax:  CLISCHEMA = clischema

Default Setting:            No alternatives specified.

DB2 CLI/ODBC Settings Tab:  Not present.
```

사용시의 주의사항

CLISCHEMA 옵션은 카탈로그 정보를 얻기 위해 DB2 CLI와 ODBC 카탈로그 함수 호출이 수행될 때 SYSIBM(또는 SYSTEM, QSYS2) 스키마 대신 대체 스키마, 테이블 및 색인을 검색하도록 지시합니다.

예를 들어, CLISCHEMA='SERGE'라고 지정하면 보통 시스템 테이블을 참조하는 내부 CLI/ODBC API 호출에서 대신에 아래의 사용자 테이블을 참조합니다.

- SERGE.TABLES
- SERGE.COLUMNS
- SERGE.SPECIALCOLUMNS
- SERGE.TSTATISTICS
- SERGE.PRIMARYKEYS
- SERGE.FOREIGNKEYS

```
- SERGE.TABLEPRIVILEGES
- SERGE.COLUMNTABLES
- SERGE.PROCEDURES
- SERGE.PROCEDURESCOLUMNS.
```

CLISCHEMA를 사용할 수 있으려면 데이터베이스 관리자가 먼저 이 사용자 테이블(user table)을 만들어야 합니다.

주: DataPropagator는 CLISCHEMA를 지원하므로 데이터베이스 관리자는 아래의 세 가지 방법으로 이 타스크를 수행할 수 있습니다.

1. 클라이언트에서 db2cli.exe 사용
2. DataPropagator를 사용하는 서버에서는 자동으로 수행
3. 서버에서 직접 수행

다음은 클라이언트에서 이 타스크를 수행하는 방법에 관해 설명합니다.

db2cli 및 bldschem 유틸리티

CLISCHEMA에 필요한 사용자 테이블을 설정하는 유틸리티는 이전에는 문서화되지 않은 CLI 명령행 인터페이스의 bldschem 지원 명령 형식으로 제공되는데, 이것은 /samples/cli/db2cli.exe에 있습니다. bldschem 지원 명령을 포함하지 않은 db2cli.exe에 대한 문서는 /samples/cli/INTCLI.DOC에 있습니다.

예를 들어, 샘플 데이터베이스에서 스키마 소유자(작성자) USERID가 소유한 테이블 이름 STAFF에 대해 CLISCHEMA='SERGE'를 가지고 작업하는 데 필요한 사용자 테이블 세트를 만들려면, db2start를 실행하고 데이터베이스를 ODBC/CLI에 등록한 후 아래 명령을 수행해야 합니다.

```
db2cli < addstaff.txt
```

"addstaff.txt"에는 아래 스크립트가 들어 있습니다.

```
opt callerror on
opt echo on
quickc 1 1 sample userid password
#
# Repeat next line for each table to add.
#
bldschem 1 SERGE USERID STAFF
```

```
#
# Exit
#
killenv 1
```

이 결과 USERID.STAFF 테이블에 대한 시스템 카탈로그 테이블 데이터를 사용하여 만들어진, 색인을 가진 SERGE.* 테이블 세트가 위에서 나열된대로 만들어집니다. 예를 들어, SERGE.TABLES는 대응되는 각 항목의 새 행이 모여 만들어집니다. 추가 bldschem 호출은 기존 SERGE.* 테이블에 추가되어 기존 행을 대체합니다.

요약하면 bldschem 지원 명령의 구문은 아래와 같습니다.

```
bldschem <handle_number> <value_of_CLISCHEMA> <schema_owner> <table_name>
```

여기서,

- <handle_number>는 1이어야 합니다.
- <value_of_CLISCHEMA>는 CLISCHEMA 키워드로 지정한 스키마 이름과 같아야 합니다.
- <schema_owner>는 테이블 작성자입니다.
- <table_name>은 user table, 뷰, 별명, 동의어 또는 시스템 테이블 이름입니다(와일드카드 문자를 사용할 수 있습니다).

이어서 db2cli.exe를 통해 다음의 예를 수행하면, 앞의 예에서 만들어진 SERGE.* 사용자 테이블에 스키마 소유자가 FRED와 BERT인 각 테이블의 시스템 카탈로그 테이블에 있는 데이터를 반영하는 행이 추가됩니다.

```
bldschem 1 SERGE FRED %
bldschem 1 SERGE BERT %
```

이어서 CLISCHEMA CLI/ODBC 키워드가 SERGE에 설정되면, ODBC/CLI 응용프로그램으로 샘플 데이터베이스를 처리할 때 시스템 카탈로그 테이블 대신 SERGE.* 테이블 세트를 참조합니다.

바람직한 접근 방법

대부분의 제품 환경에서 기본 값으로 시스템 카탈로그 테이블을 검색하면 대량의 데이터가 리턴될 수 있으므로 CLI/ODBC 응용프로그램이 데이터베이스를 열 때마다 상당한 지연을 초래할 수 있습니다. 통상적인 테스트 데이터베이스에서도 약 25초 정도의 지연은 초래할 수 있을 것입니다.

위의 CLI 키워드가 아무 것도 설정되지 않은 상태에서 초기에 이러한 지연을 측정하는 한편, 연결 시간과 특히 DB2 클라이언트에서 새 데이터베이스에 처음 연결할 때 발생할 수 있는 긴 지연을 줄여야 합니다.

다음에 계속하는 방법은 데이터와 조직의 구조에 따라 달라집니다. 어떤 경우에는 특정 응용프로그램이나 응용프로그램 그룹에서 사용하는 검색을 제한하기 위해 DBNAME, SCHEMALIST 및 TABLETYPE을 조합하여 사용할 수 있습니다. 예를 들어, 제품 DBA 클라이언트에서 대개 주어진 DBNAME과 스키마에 따라 테이블을 액세스한다면 이것은 지정하기가 쉽습니다.

CLISCHEMA는 대부분의 사용자에게 최고의 성능상 장점을 제공합니다. 이러한 이유 때문에 일반적으로 제품 환경에서 CLISCHEMA를 사용하는 것이 좋습니다. CLI 명령행 인터페이스(db2cli.exe)와 bldschem 지원 명령을 통해 CLISCHEMA의 사용자 테이블을 설정하고 수정하는 것이 훨씬 더 쉽기 때문입니다. 249 페이지의 『db2ocat 카탈로그 최적화 알고리즘 도구』에서도 자세한 내용을 참조하십시오.

추가 힌트와 추가 정보

CLISCHEMA 키워드는 db2cli.ini 파일의 DSN 이름 섹션이나 공통 섹션 안에 추가되어야 합니다. 섹션은 대괄호 안에 묶인 텍스트입니다. COMMON 섹션은 대괄호 안에 묶인 텍스트 "COMMON"으로 표시됩니다. 키워드와 섹션 이름은 대소문자를 구별하지 않음을 유의하십시오.

연결시에 먼저 DSN 이름 아래에서 가능한 각 키워드를 확인하고, 없으면 COMMON 섹션 아래에서 찾습니다. 이것은 DSN 특정 키워드와 글로벌(클라이언트) 키워드 둘 다를 고려한 것입니다.

또한 DBALIAS 키워드는 같은 데이터베이스에 맵되는 서로 다른 DSN(ODBC 데이터 소스)을 만드는 데 사용할 수 있습니다. (DSN 이름의 길이로 최대 255개의 문자까지 사용할 수 있으며, 이 이름은 8개의 char로 된 dbname에 맵핑됩니다.)

아래의 예에서, 사용자가 TESTDB나 파일에 없는 임의의 DSN에 연결할 때는 언제나 clischema=ODBCCAT를 사용합니다. TestDBcat2에 연결하는 경우에는 clischema=odbccat2를 사용하지만 여전히 testdb 데이터베이스에 연결됩니다.

db2cli.ini 파일의 예:

```
[TESTDB]

[COMMON]
clischema=odbccat

[TestDBcat1]
DBALIAS=testdb
clischema=odbccat1

[TestDBcat2]
DBALIAS=testdb
clischema=odbccat2
```

db2ocat 카탈로그 최적화 알고리즘 도구

새 도구 db2ocat는 ODBC와 JDBC 응용프로그램의 시스템 카탈로그 검색을 최적화하도록 돕기 위해 Windows 32비트 운영 체제에서 제공됩니다.

아래의 주소에서 db2ocat.zip을 다운로드함으로써 db2ocat 지정 후 클릭(point-and-click) 카탈로그 최적화 알고리즘 유틸리티를 얻을 수 있습니다.

```
ftp://ftp.software.ibm.com/ps/products/db2/tools.
```

추가 정보 소스

다음은 유익한 추가 정보 소스입니다.

- DataPropagator를 사용하는 카탈로그 자동 생성에 대한 문서

  ```
  http://www.ibm.com/software/data/db2/os390/odbcatlg.html
  ```

- DataPropagator를 사용하지 않는 수동 접근 방법에 대한 문서

  ```
  http://www.ibm.com/software/data/db2/os390/odbcmanu.html
  ```

부록G. 추가 및 관련 정보 소스

기타 관련 책

문서 번호	책 제목
SG24-2006	*Migrating to DB2 Universal Database Version 5*
SG24-2213	*DB2 for OS/390 Version 5 Performance Topics*
SG24-4893	*DB2 Meets NT*
SG24-4894	*The Universal Connectivity Guide to DB2*
SG24-4693	*Getting Started with DB2 Stored Procedures*
SG24-2212	*DRDA Support for TCP/IP in DB2 Universal Database for OS/390 V5.1 and DB2 Universal Database V5.0*
SC33-0814	*CICS for AIX Application Programming Guide*
SC33-0931	*CICS for AIX Customization and Operation Guide*
GA30-0973	*UNIX용 DB2 Connect Enterprise Edition 빠른 시작*
GA30-0974	*OS/2 및 Windows용 DB2 Connect Enterprise Edition 빠른 시작*
GA30-0981	*DB2 Connect Personal Edition 빠른 시작*
GG24-4155	*DRDA: Using DDCS for AIX DRDA support with DB2 for MVS/ESA and DB2 Universal Database for AS/400*
GG24-4311	*DRDA Cross Platform Connectivity and Application*
SC23-2443	*Encina for AIX Product Family Overview*

© Copyright IBM Corp. 1993, 2000

부록H. 주의사항

IBM은 이 책에서 논의된 제품, 서비스 또는 기능을 다른 나라에서는 제공하지 않을 수 있습니다. 현재 사용자가 사용할 수 있는 제품 및 서비스에 대한 정보는 해당 지역의 IBM 영업대표에게 문의하십시오. IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 반드시 IBM 제품, 프로그램 또는 서비스만을 사용해야 함을 의미하지는 않습니다. IBM의 지적 소유권을 침해하지 않는 기능상으로 동등한 타사의 제품, 프로그램 또는 서비스를 대신 사용할 수 있습니다. 그러나, 타사 제품, 프로그램 또는 서비스의 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대한 특허를 보유하고 있거나 출원중일 수 있습니다. 이 책을 구입한다고 해서 그 특허에 대한 사용권을 부여받는 것은 아닙니다. 특허 사용권에 대한 문의는 다음 주소로 하십시오.

135-270
서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩
한국 아이.비.엠 주식회사
지적 재산권부

2바이트(DBCS) 정보에 관한 사용권 문의는 사용자 국가의 IBM 지적 재산권부나 다음 주소로 서면 문의하십시오.

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

다음 사항은 영국이나 이 조항이 현지법과 상충되는 나라에는 적용되지 않습니다.
IBM에서는 이 책을 명시적 또는 암시적인 어떠한 종류의 보증없이『있는 그대로』제공하므로, 판매 가능성을 보장하거나 특정 목적에 적합한지 여부에 대해서는 책임질 수 없습니다. 일부 국가에서는 특정 거래의 명시적 또는 암시적인 보증을 부인하는 문장을 허용하지 않으므로, 이 사항이 사용자에게 적용되지 않을 수도 있습니다.

© Copyright IBM Corp. 1993, 2000

이 책에는 기술상 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 책의 내용은 정기적으로 변경되며, 이들 변경사항은 개정판에 통합됩니다. IBM은 사전 통지없이 언제든지 이 책에 설명된 제품과 프로그램을 개선 및 변경할 수 있습니다.

이 책에서 타사의 웹 사이트를 언급한 것은 단지 편의를 위해서일 뿐이며 이런 웹 사이트를 추천하려는 의도는 아닙니다. 이런 웹 사이트의 데이터가 이 IBM 제품에 대한 데이터의 일부는 아니므로 이런 웹 사이트 사용에 대한 책임은 사용자가 져야 합니다.

IBM은 독자가 제공한 정보를 적절한 방식으로 사용하거나 배포할 수 있으며, 제공한 독자는 이에 대해 책임을 지지 않습니다.

이 프로그램의 사용권을 보유한 고객이(i) 별도로 작성된 프로그램과 기타 프로그램(이 프로그램 포함) 사이의 정보 교환 및(ii) 교환된 정보의 독점적 사용을 가능케 할 목적으로 이에 대한 정보를 원하는 경우, 한국 IBM 소프트웨어 사업부(02-3781-7777)로 문의하십시오.

135-270
서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩
한국 아이.비.엠 주식회사
소프트웨어 사업부

그런 정보는 적절한 조건과 조항에 따라 사용할 수 있으며, 경우에 따라 사용료를 지불해야 합니다.

이 책에 기술된 사용권 프로그램 및 이 프로그램에 사용가능한 모든 사용권 데이터는 IBM 고객 협약, IBM 국제 프로그래밍 사용권 협약 또는 이와 동등한 모든 협약 조건하에 IBM에서 제공됩니다.

여기에 제시된 어떠한 성능 데이터는 주위 환경에 따라 결정될 수 있습니다. 따라서, 다른 운영 체제에서 제시된 결과 값과 다를 수 있습니다. 몇몇 측정값은 개발 단계에서 얻은 값일 수 있습니다. 따라서 일반적인 사용자 시스템에서 얻은 값과 다를 수 있습니다. 또한 몇몇 측정값은 보외법을 통해 측정된 값입니다. 실제 값과는 다를 수 있습니다. 이 책의 사용자는 사용자의 특정 환경에 맞게 적용가능한 데이터를 변경해야 합니다.

타사 제품과 관련된 정보는 해당 제품의 공급자, 공개 발표 또는 기타 공개적으로 사용가능한 소스에서 확보한 것입니다. IBM은 이들 제품을 검사하지 않았고 성능 상의 정확성, 호환성 또는 타사 제품과 관련된 기타 주장을 확인할 수 없습니다. 타사 제품의 성능에 관한 문제는 해당 제품의 공급자에게 제기되어야 합니다.

IBM이 제시하는 방향 또는 의도에 관한 어떠한 언급도 특별한 통지없이 변경될 수 있습니다.

이 정보는 일상적인 비지니스 처리에 사용되는 데이터와 보고서의 예가 들어 있을 수 있습니다. 이러한 예를 가능하면 완벽하게 보여주기 위해 예에는 개인, 회사, 상표 및 제품의 이름이 포함되어 있습니다. 이 이름은 모두 가공의 것이며, 실제 기업에서 사용되는 이름 및 주소와 유사하더라도 이는 전적으로 우연의 일치입니다.

사용권:

이 정보에는 여러 운영 체제에서 프로그래밍 소스 언어로 예제 응용프로그램이 들어 있을 수 있습니다. 사용자는 이들 예제 프로그램을 IBM에게 비용을 지급하지 않고 복사, 수정 및 분배할 수 있습니다. 이들 예제 프로그램은 모든 조건에서 철저하게 검사되지 않았습니다. 따라서, IBM은 이들 프로그램에 대해 어떠한 보증도 할 수 없습니다.

이들 예제 프로그램의 각각의 복사본이나 특정 부분은 다음과 같은 사용권 주의사항을 포함해야 합니다.

© (사용자 회사 이름) (년도). 이 코드의 일부는 IBM Corp. 예제 프로그램에서 발췌된 것입니다. © Copyright IBM Corp. (년도 입력). All rights reserved.

등록상표

별표(*)로 표시된 다음의 용어는 전세계에서 IBM의 상표입니다.

ACF/VTAM
AISPO
AIX
AIX/6000
AIXwindows
AnyNet
APPN
AS/400
BookManager
CICS
C Set++
C/370
DATABASE 2
DataHub
DataJoiner
DataPropagator
DataRefresher
DB2
DB2 Connect
DB2 Extenders
DB2 OLAP Server
DB2 Universal Database
Distributed Relational Database Architecture
DRDA
eNetwork
Extended Services
FFST
First Failure Support Technology
IBM
IMS
IMS/ESA
LAN DistanceMVS
MVS/ESA
MVS/XA
Net.Data
OS/2
OS/390
OS/400
PowerPC
QBIC
QMF
RACF
RISC System/6000
RS/6000
S/370
SP
SQL/DS
SQL/400
System/370
System/390
SystemView
VisualAge
VM/ESA
VSE/ESA
VTAM
WebExplorer
WIN-OS/2

다음 용어는 해당 회사의 상표 또는 등록 상표입니다.

Microsoft, Windows 및 Windows NT는 Microsoft Corporation의 상표 또는 등록 상표입니다.

Java 또는 모든 Java 관련 상표 및 로고 그리고 Solaris는 전세계에서 Sun Microsystems, Inc.의 상표입니다.

Tivoli 및 NetView는 전세계에서 Tivoli Systems Inc.의 상표입니다.

UNIX는 전세계에서 X/Open Company Limited가 독점권을 갖는 등록 상표입니다.

두 개의 별표(**)가 붙은 기타 회사 이름, 제품 이름 또는 서비스 이름은 해당 회사의 상표이거나 서비스 표시입니다.

색인

[가]

가변 길이 문자 컬럼 128
가상 원격 통신 액세스 방식 (VTAM) 133, 240
가져오기
  제한사항 127
가져오기 유틸리티
  일반 설명 126
계정 문자열 65
계정 문자열 필드 65
공통적인 SQL 7
관리 유틸리티 7
교환 서버 속성 명령 199
구문
  bldschem 247
구성
  DB2 Connect 6
  ODBC 드라이버 76, 78
구성 고려 사항
  암호 변경 139
구에 의한 그룹
  정렬 순서 55
구에 의한 순서
  정렬 순서 55
국가 코드
  SQLCA의 SQLERRMC 필드에서 51
국가 코드 페이지 지원 225
권한 부여 ID(모니터) 116
규정자 속성
  다른 플랫폼 53
  패키지 53
기본 키 55
기호 목적지 이름 223
  대문자와 소문자 구별 88
긴 필드 50

[나]

날짜와 시간대 지원 93
내보내기
  제한사항 127
내보내기 유틸리티
  일반 설명 126
내재된 연결 51
네트워크
  신뢰성 174
  어댑터 또는 통신 제어기 173
  조정 170
  토폴로지 174
  통신량 174
네트워크 성능 도구 150
노드 디렉토리 87, 88
노드 이름 88, 99, 223

[다]

다른 DB2 제품간의 차이 48
다중 갱신
  지원 62
다중 사이트 갱신 14
  제어 센터 19
다중 사이트 갱신 기능
  테스트 20
다중 사이트 갱신 마법사 20
대화형 입력 모드(CLP) 126
데이터 변환 168
  예외 227
  코드 페이지 225
  CCSID 225
데이터 블로킹 154
데이터 소스 13
데이터 유형
  변환 168
  부동 소수점 수 169
  숫자 49
  압축 십진수 169
  정수 169
  CHAR 170
  VARCHAR 170
  zoned 십진수 169
데이터 전송
  호스트와 워크스테이션간 126
데이터 전송률 147
  성능 173
데이터 제어 언어(DCL) 51
데이터 흐름 10, 148
데이터베이스
  별명 223
  성능 도구 150
  오브젝트 229
  요청 그룹화 154
  위치 지정자 오브젝트 229
  이름 223
  조정 166
데이터베이스 디렉토리
  갱신 87
  노드 87
  데이터베이스 연결 서비스(DCS) 87
  시스템 데이터베이스 87
데이터베이스 디렉토리 갱신 87
데이터베이스 별명 99
데이터베이스 시스템 모니터 111
데이터베이스 시스템 모니터 유틸리티 8
데이터베이스 연결 서비스(DCS) 디렉토리 87
데이터베이스 이름 89, 99

© Copyright IBM Corp. 1993, 2000

도구
메모리 사용 150
성능 150
CPU 사용 150
동적 커서 52
동적 SQL 7, 156
CURRENTPACKAGESET 138
DB2 Connect 지원 47
등록
ODBC 드라이버 관리 프로그램 75

[라]

라우팅 요청 11
라우팅 정보 오브젝트 229

[마]

마법사
다중 갱신 20
매개변수
AGENTPRI 159
AUTHENTICATION 130
BIDI 95
DB_Authentication 231
DB_Communication_Protocol 231, 233
DB_Database_Protocol 231
DB_Native_Database_Name 231
DB_Object_Type 230, 233
DB_Principal 231
DB_Product_Name 230
DB_Target_Database_Info 235
DFT_ACCOUNT_STR 66
DIR_CACHE 158
INTERRUPT_ENABLED(연결해제) 92
LOCALDATE 93
MAXAGENTS 159, 163
MAXDARI 158
MAX_COORDAGENTS 163
매개변수 *(계속)*
NOMAP 91
NUMDB 158
NUM_INITAGENTS 163
NUM_POOLAGENTS 163
PRDDTA 65
PRDID 200
RQRIOBLK 157
SYSPLEX 93
매개변수 문자열 223
매개변수 문자열의 쉼표 91
매개변수 문자열의 쉼표 쉼표 91
매개변수 문자열의 ,(쉼표) 91
매개변수 문자열의 ,,(쉼표 쉼표) 91
메모리 사용 도구 150
명령
확약 201
ACCRDB 199, 200
ACCRDBRM 199, 200
BIND 109
EXCSAT 199
EXCSATRD 199, 200
EXCSQLSTT 64
FORCE 52
quit 126
REBIND PACKAGE 109
terminate 126
명령문
연결 51
호출 58
ACQUIRE 64
COMMIT 156
COMMIT WORK RELEASE 64
CREATE STORGROUP 49
CREATE TABLESPACE 49
DECLARE 64
DELETE 49
DESCRIBE 64, 156
EXECUTE IMMEDIATE 156
FOR FETCH ONLY 155
명령문 *(계속)*
GRANT 51
INSERT 49, 50
LABEL ON 64
PREPARE 64, 156
ROLLBACK 52, 156
SELECT 49, 155, 156
SET CURRENT 64
UPDATE 49
명령행 처리기 (CLP) 7, 125
명령행 처리기(CLP)
성능 156
REBIND PACKAGE 명령 109
명확한 커서 52
모니터링
DB2 Connect 게이트웨이에서 연결 111
목표 데이터베이스 이름 90, 223
문자 대체 227
문자 번역 50
문제점 판별 185
문제점 해결 185
숫자 변환 오버플로우 57

[바]

바인드 목록 103, 242
바인딩
유틸리티 72
유틸리티와 응용프로그램 103
패키지 107
필요한 권한 104
바인딩에 필요한 권한 104
번역
문자 50
벤치마킹
성능 149
변수 길이 문자열 50
변환 225
변환 예외 227

병목 현상
  트랜잭션 149
보안 237
  고려사항 129
  유형 88, 132, 223, 237
  APPC 132
  DCE 129
  GRANT 241
  GRANT 명령문 134
  NONE 132, 133
  PROGRAM 132
  REVOKE 241
  REVOKE 명령문 134
  SAME 132
복합 SQL
  NOT ATOMIC 61, 152
부동 소수점 수 데이터 유형 169
분리 레벨 57
분산 관계형 데이터베이스 아키텍처 (DRDA)
  개념 9
  데이터 흐름 10
  사용된 아키텍처 11
  응용프로그램 리퀘스터 10
  응용프로그램 서버(AS) 10
  책 21
분산 데이터 관리 191
분산 데이터 관리(DDM) 11
분산 요청(DR) 13
분산 환경 47
불명확한 커서 52
뷰
  시스템 카탈로그 56
블로킹 52
  데이터 154
블록 크기 157

[사]

사용 비용 청구 계정
  정의 65
사용 비용 청구 계정 *(계속)*
  OS/390용 DB2 Universal Database 65
사용자 이름 237
사용자 정의 유형
  DB2 Connect에서 지원 50
사용자 정의된 조합 순서 55
사전 처리 컴파일러
  지원 49
  DB2 Connect 지원 52
설정
  DB2 Connect 6
설치
  DB2 Connect 6
성능
  네트워크 도구 150
  네트워크 하드웨어 173
  도구 150
  명령행 처리기 156
  문제점 해결 175
  벤치마킹 149
  병목 현상 149
  일반 147
  조정 167
  CLI 응용프로그램 243
  CLISCHEMA 키워드 243
  ODBC와 JDBC 응용프로그램 성능 조정 243
  OS/390용 DB2 167
  PIU 크기 180
  RU 크기 180
  SNA 조정 기준 179
  SNA 조정 추가 정보 175
섹션 번호 64
소유자 속성
  패키지 53
송신 버퍼(추적) 191
수신 버퍼(추적) 191
술어 논리 154
숫자 데이터 유형 49
숫자 변환 오버플로우 57
시간대 지원 93
시스템 데이터베이스 디렉토리 87, 99
시스템 자원
  경합 174
시스템 자원에 대한 경합 174
시스템 카탈로그
  사용 56
시프트 인 및 시프트 아웃 문자 49, 128

[아]

아웃바운드 순서 번호(모니터) 119
암호
  DCE 디렉토리 서비스 237
암호 만기 관리(PEM) 95
액세스 RDB 액세스 199
양방향 언어 지원 228
양방향 CCSID 지원 95
연결
  내재된 연결 51
  널(NULL) CONNECT 51
  CONNECT RESET 명령문 51
  CONNECT TO 명령문 51
연결 집중기
  개요 161
  구성 매개변수 163
  연결 오버헤드 162
  예 164
  제한사항 165
  XA 트랜잭션 지원 163
연결 풀링
  개요 31
연결성 서버
  DB2 Connect Enterprise Edition 27
연쇄 55
예
  연결 집중기 164
  XA 집중기 164

오류 185
외부 키 55
요청 그룹화
  데이터베이스 154
원격 작업 단위(RUOW) 11
웹 응용프로그램
  개요 29
유틸리티
  가져오기 126
  관리 7, 125
  내보내기 126
  데이터베이스 시스템 모니터 8
  바인딩 72, 103
  추적 190
  프로세스 상태 199
  bldschem 246
  db2cli 246
  db2ocat 249
  ddcspkgn 108, 110
  ddcstrc 190
  ps 199
유형
  보안 132
  인증 130
  ROWID 50
응답 시간 147
응용 프로그램 핸들 116
응용프로그램
  바인딩 103
응용프로그램 개발 47, 152
  DB2 응용프로그램 개발 클라이언트 사용 27
  ODBC 사용 27, 72
응용프로그램 리퀘스터
  매개변수 223
  DRDA 정의 10
응용프로그램 리퀘스터(AR) 이름 91
응용프로그램 서버
  개요 36
  구성 39
응용프로그램 서버 *(계속)*
  배치 39
  DB2 Connect 지원 38
  DRDA 정의 10
응용프로그램 설계 152
응용프로그램 성능
  CLISCHEMA 키워드 243
응용프로그램 수행
  데이터베이스 클라이언트 71
응용프로그램 이름(모니터) 116
응용프로그램 포팅 47
이벤트
  추적 192
이식용 책 69
인증 99, 223
  유효성 확인 129
인증 유형
  기본값 130
  CLIENT 130
  DCE 130
  DCE 디렉토리 서비스 237
  DCS 130
  DCS_ENCRYPT 130
  SERVER 130
  SERVER_ENCRYPT 130

[자]

자국어 지원(NLS)
  고려사항 225
  혼합 바이트 데이터 128
자국어 (NLS)
  문자 데이터 변환 225
  혼합 바이트 데이터 49
자원 액세스 제어 기능(RACF) 133, 240
자체 참조 테이블 55
작성자 속성
  패키지 53
작업 단위
  분산 14
작업 단위(UOW)
  원격 11
잠금
  시간종료 56
  페이지 레벨 56
  행 레벨 56
잠금 시간종료 56
저장 프로시듀어
  개요 31
  일반 58
전제조건
  DCE 129
정렬 순서
  정의 55
  조합 순서 55
정수 데이터 유형 169
정적 SQL 7, 156
  DB2 Connect 지원 47
제한사항
  가져오기 및 내보내기 127
  연결 집중기 165
조정
  네트워크 성능 170
  데이터베이스 166
  응용프로그램 성능 243
조합 순서
  EBCDIC과 ASCII 55
종료 작업 단위(UOW) 응답 메시지 (ENDUOWRM) 201
지역
  SQLCA의 SQLERRMC 필드에서 51
직접 데이터베이스 액세스 24
직접 연결
  호스트 데이터베이스로 연결 24
진단 도구 189

[차]

참조 무결성 55
처리량
  트랜잭션 147
첫번째 오류 서비스 로그 189
최소단위 복합 SQL
  지원되지 않음 61
  DB2 Connect에서 지원되지 않음 152
추적 유틸리티 190
  구문 192
  출력 193
  출력 파일 191

[카]

캐쉬된 디렉토리 정보 158
커서
  동적 52
  명확함 52
  불명확함 52
커서 안정성(CS) 56
코드 세트
  SQLCA의 SQLERRMC 필드에서 51
코드 페이지 225
  SQLCA의 SQLERRMC 필드에서 51
코드 페이지 변환 예외 227
코드 페이지 ID(모니터) 118
코드화된 문자 세트 식별자 (CCSID) 225
코어 파일 190
콜렉션 53
콜렉션 ID 속성
  패키지 53
  AS/400용 DB2 Universal Database 53
클라이언트 순서 번호(모니터) 117
클라이언트 응용프로그램 ID(모니터) 117
클라이언트 제품 ID(모니터) 118
클라이언트 DB 별명(모니터) 117
클라이언트 NNAME(모니터) 118
키
  기본 55
  외부 55
키워드
  CLISCHEMA 244, 246, 247, 248
  DBALIAS 248

[타]

테스트
  다중 사이트 갱신 기능 20
토큰과 SQLCODE 141
트랜잭션
  처리량 147
트랜잭션 프로세싱
  특성 41

[파]

패키지
  속성 53
  호스트나 AS/400 데이터베이스 서버에 작성된 패키지 107
패키지 속성
  규정자 53
  소유자 53
  작성자 53
페이지 레벨 잠금 56
페이징
  블록 크기 157
프로그래밍 가정 47
  호스트나 AS/400 환경에서 47
프로그래밍 정보 47
프로세스 상태 유틸리티 190, 199

[하]

하드웨어
  네트워크 성능 173
행 레벨 잠금 56
향상된 사항
  DB2 Connect 버전 5.0 218
  DB2 Connect 버전 5.2 218
  DB2 Connect 버전 6.1 217
  DDCS 버전 2.3 221
  DDCS 버전 2.4 221
형식화된 데이터 오브젝트 내용 아키텍처 (FD:OCA) 11
호스트 데이터베이스 이름(모니터) 119
호스트 응용프로그램 ID(모니터) 116
호스트 제품 ID(모니터) 119
호스트나 AS/400 서버에서 데이터 확장 50
호스트나 AS/400 서버와 워크스테이션 사이의 차이 64
혼합 바이트 데이터 49
  가져오기 128
  내보내기 128
확약 명령 201
확장된 동적 SQL문
  지원되지 않음 65
환경 변수
  DB2ACCOUNT 66

[숫자]

2단계 확약 14
  TCP/IP 연결에 의해 사용된 재동기화 포트 89
2바이트 문자 227
64비트 정수(BIGINT) 데이터 유형
  DB2 Connect 버전 7에서 지원 50

A

ACCRDB 명령 199, 200
ACCRDBRM 명령 199, 200
ACCSEC 200
ACQUIRE 명령문 64
AGENTPRI 매개변수 159
AIX용 CLP의 *(별표) 126
AIX용 CLP의 "(큰 따옴표) 126
API
  데이터베이스 디렉토리 갱신 100
APPC
  기호 목적지 이름 223
AR 이름 91
ARI(VSE & VM용 DB2) 51
AS 목표 데이터베이스 이름 90
ASCII
  정렬 순서 55
  혼합 바이트 데이터 49
AS/400
  DRDA 10
AUTHENTICATION 매개변수 130
AUTHENTICATION=CLIENT 138

B

BIDI 매개변수 95
BIND 명령
  구문 109
BINDADD 특권 104
bldschem 246
  구문 247

C

CALL USING DESCRIPTOR 명령문 (OS/400) 58
CALL 명령문
  다른 플랫폼 58
CCSID 227
CDRA(Character Data Representation Architecture) 11
CGI 프로그래밍
  잇점 29
  제한사항 29
CHAR 데이터 유형 170
CHGPWD_SDN 매개변수 95
CICS 48
CLI
  유틸리티 246
CLIENT 인증 유형 130
CLISCHEMA 키워드 244, 246, 247, 248
CLI/ODBC 응용프로그램
  CURRENTPACKAGESET 138
CLI/ODBC 응용프로그램 성능
  응용프로그램 성능 243
COMMIT WORK RELEASE 명령문
  지원되지 않음 64
COMMIT 명령문
  정적으로 바인드된 156
CPU 사용 도구 150
CREATE IN COLLECTION NULLID 104
CREATE STORGROUP 명령문
  지원 49
CREATE TABLESPACE 명령문
  지원 49
CURRENTPACKAGESET 138

D

DataPropagator 246
DB2 Connect Enterprise Edition
  연결성 서버로서 27
DB2 Connect RQRIOBLK 크기 180
DB2 Connect 버전 5.0
  향상된 사항 218
DB2 Connect 버전 5.2
  향상된 사항 218
DB2 Connect 버전 6.1
  향상된 사항 217
DB2 Connect 사용
  시나리오 23
  TP 모니터와 함께 사용 40
  Tuxedo 43
  XA 추종 트랜잭션 관리 프로그램과 함께 사용 43
DB2ACCOUNT 환경 변수 66
db2cli.exe 유틸리티 246
db2cli.ini 파일
  ODBC와 JDBC 응용프로그램 성능 조정 245
DB2CONNECT_IN_APP_PROCESS 환경 변수 111
db2ocat
  카탈로그 최적화 알고리즘 유틸리티 249
DBALIAS 키워드 248
DB_Authentication 매개변수 231
DB_Communication_Protocol 매개변수 231, 233
DB_Database_Protocol 매개변수 231
DB_Native_Database_Name 매개변수 231
DB_Object_Type 매개변수 230, 233
DB_Principal 매개변수 231
DB_Product_Name 매개변수 230
DB_Target_Database_Info 매개변수 235
DCE
  디렉토리 서비스 229
  보안 129
  설정 정보 129
  소프트웨어 전제조건 129
  인증 유형 130
DCL(데이터 제어 언어) 51
DCS
  인증 유형 130
DCS 디렉토리 89

DCS 디렉토리 89 *(계속)*
데이터베이스 이름 89
매개변수 문자열 지정 98
목표 데이터베이스 이름 90
응용프로그램 리퀘스터(AR) 이름 91
AR 이름 91
AS 목표 데이터베이스 이름 90
BIDI 매개변수 95
CHGPWD_SDN 매개변수 95
LOCALDATE 매개변수 93
SYSPLEX 매개변수 93
dcs1ari.map 파일 141
dcs1dsn.map 파일 141
dcs1qsq.map 파일 142
DCS_ENCRYPT 인증 유형 130
DDCS 버전 2.3
향상된 사항 221
DDCS 버전 2.4
향상된 사항 221
ddcs400.lst file 103
ddcsmvs.lst file 103
ddcstrc 유틸리티 190
구문 192
출력 193
출력 파일 191
ddcstrc.dmp 파일 193
ddcsvm.lst file 103
ddcsvse.lst file 103
DDL 49
DECLARE PROCEDURE 명령문 (OS/400) 58
DECLARE 명령문 64
지원 64
DELETE
지원 49
DESCRIBE 명령문 64, 156
지원 64
DFT_ACCOUNT_STR 구성 매개변수 66
DIR_CACHE 매개변수 158
DML 49
DRDA 호스트에 연결
DRDA 호스트로 직접 연결 25
DSN(OS/390용 DB2 Universal Database) 51
DSS 유형(추적) 191
D(연결해제) 매개변수 91

E

EBCDIC
정렬 순서 55
혼합 바이트 데이터 49
Embedded SQL 19
ENDUOWRM 메시지 201
Escape 문자 98
EXCSAT 명령 199
EXCSATRD 명령 199, 200
EXCSQLSTT 명령 64
EXECUTE IMMEDIATE 명령문 156
EXTNAM 오브젝트 199

F

FORCE 명령 52, 116

G

GRANT 명령문
보안 134, 241

I

IBM SQL 7
IBM WebSphere
개요 34
INSERT 명령문
지원 49, 50
INTERRUPT_ENABLED(연결해제) 매개변수 92
ISO/ANS SQL92 54

J

Java
응용프로그램 서버(AS)
DB2 Connect 사용 32
프로그램 수행 79
JDBC
응용프로그램 성능 243
카탈로그 최적화 알고리즘 유틸리티 249
프로그램 수행 79

L

LABEL ON 명령문 64
LANGLEVEL SQL92E 사전 처리 컴파일 옵션 54
LIST DCS APPLICATIONS 명령 116
LOB 데이터 유형
DB2 Connect 버전 7에서 지원 50
LOCALDATE 매개변수 93

M

MAXAGENTS 매개변수 159, 163
MAXDARI 매개변수 158
MAX_COORDAGENTS 매개변수 163
Microsoft ODBC 드라이버 관리자 74
Microsoft Windows 응용프로그램 27
MVS
DRDA 10
MVS/ESA용 DB2 또는 OS/390용 DB2 Universal Database
부트스트랩 데이터 세트 89
DOMAIN 89
RESPORT 89
TCPPORT 89
MVS/ESA용 DB2 또는 OS/390용 DB2 Universal Database의 부트스트랩 데이터 세트
BSDS 매개변수 89

N

Net.Data
  개요 33
  기능 34
NOMAP 매개변수 91, 141
NON AATOMIC 복합 SQL 61, 152
NONE 보안 유형 132, 133
NUMDB 매개변수 158
NUM_INITAGENTS 매개변수 163
NUM_POOLAGENTS 매개변수 163

O

ODBC
  드라이버 관리 프로그램 등록 75
  응용프로그램 성능 243
  인터페이스 27
  프로그램 수행 72
ODBC 응용프로그램
  CURRENTPACKAGESET 138
odbcad32.exe 74
ODBC/CLI
  카탈로그 최적화 알고리즘 유틸리티 249
OS/2의 \(백 슬래시) 192
OS/390
  DRDA 10
OS/390용 DB2 Universal Database
  DYNAMICRULES(BIND) 138
  TCP/IP 이미 확인 138
OS/400
  DRDA 10
OS/400용 NULLID 104

P

Packed 십진수 데이터 유형 169
PC/IXF 파일 형식 127
PIU 180
PRDDTA 매개변수(DRDA) 65
PRDID 매개변수 200
PREPARE 명령문 156
  지원 64
PROGRAM 보안 유형 132
ps(프로세스 상태) 유틸리티 190, 199
PUT 명령문
  지원되지 않음 64

Q

QSQ(AS/400용 DB2 Universal Database) 51
quit 명령(CLP) 126

R

RACF 240
RDBNAM 오브젝트 199
REBIND PACKAGE 명령(CLP) 109
REVOKE 명령문
  명령문 51
  보안 134, 241
ROLLBACK WORK RELEASE
  지원되지 않음 64
ROLLBACK 명령
  정적으로 바인드된 156
ROLLBACK 명령문 52
ROWID 데이터 유형
  DB2 Connect 버전 7에서 지원 50
RQRIOBLK 매개변수 157
RQRIOBLK 크기 180
RQRIOBLK 필드 52
RU 크기 180

S

SAME 보안 유형 132
SECCHK 200
SELECT 명령문 155, 156
  지원 49
SELECT 명령문의 FOR FETCH ONLY 155
SERVER 인증 유형 130
SERVER_ENCRYPT 인증 유형 130
SET CURRENT PACKAGESET 138
SET CURRENT 명령문
  지원 64
SHOW DETAIL 모니터 옵션 117
SNA 관리 서비스 아키텍처(MSA) 11
SNA 성능
  조정 추가 정보 175
SOCKS
  설정 환경 변수 88
SQL
  동적 156
  정적 156
SQL1338 리턴 코드 88
SQL92 54
SQLCA
  데이터 버퍼 191
  SQLCODE 필드 191
  SQLERRMC 필드 51, 61
  SQLERRP 필드 51
SQLCA의 SQLERRMC 필드 51, 61
SQLCA의 SQLERRP 필드 51
SQLCODE
  독립형 54
  맵핑 141
  플랫폼 차이 56
  SQLCA의 필드 191
SQLCODE 맵핑 141
SQLCODE 맵핑 파일
  구문 142
  별표 142
  cc 143
  i 144
  P 143
  s 144
  U 143
  W 142

SQLCODE 맵핑 파일 *(계속)*
* (별표) 142
SQLCODE 맵핑 파일의 더블(&&) 142
SQLCODE 맵핑 파일의 && 142
SQLCODE와 SQLSTATE에서의 차이 56
SQLDA, 할당에 제일 적합한 크기 156
sqlesact API 66
SQLSTATE
독립형 54
차이 56
클래스 코드 143
SQLCA의 SQLERRMC 필드에서 61
SQL문
범주 47
지원 64
Embedded 19
SQL의 특징 7
SQL/DS
DRDA 10
SRVNAM 오브젝트 199
Stored Procedure Builder
개요 60
기능 60
SYSIBM.SYSPROCEDURES 카탈로그 (OS/390) 58
SYSPLEX 매개변수 93

T

TCP/IP
서비스 이름 89
서비스 이름 또는 포트 번호 223
원격 호스트 이름 89, 223
이미 확인된 보안 138
재동기화 포트 89
호스트 연결 구성 25
호스트 이름 223
ACCSEC 200
DOMAIN 89
TCP/IP *(계속)*
RESPORT 89
SECCHK 200
TCPPORT 89
terminate 명령(CLP) 126
TP 모니터
예 42
Tuxedo
DB2 Connect 사용 43

U

UPDATE 명령문
지원 49

V

VALIDATE RUN
DB2 Connect 지원 52
VALNSPRM 값 200
VARCHAR 데이터 유형 170
VM
DRDA 10
VSE
DRDA 10
VTAM 240

W

WebSphere
개요 34
고급판 36
기능 34
표준판 35
enterprise edition 36
Windows 응용프로그램 27

X

XA 인터페이스
정의 43
XA 집중기
예 164
XA 추종 트랜잭션 관리 프로그램
정의 43
XA 추종 자원 관리 프로그램 43
XA 트랜잭션 지원
연결 집중기 163
X/Open 분산 트랜잭션 프로세싱(DTP) 모델
개요 43

Z

Zoned 십진수 데이터 유형 169

IBM에 문의

기술적인 문제가 발생한 경우에는 DB2 고객 지원 센터에 문의하기 전에 *문제점 해결 안내서*에서 제안한 조치를 검토하고 실행해 보십시오. 이것은 DB2 고객 지원 부서로 하여금 사용자를 보다 더 잘 지원할 수 있도록 사용자가 모을 수 있는 정보를 제공합니다.

DB2 Universal Database 제품에 대한 정보나 주문은 그 지역의 IBM 영업 대표나 공인 IBM 소프트웨어 재판매업자에게 문의하십시오.

미국에 사시는 분은 다음 번호 중 하나를 선택하여 전화하십시오.

- 고객 지원을 받으려면, 1-800-237-5511.
- 사용가능한 서비스 옵션을 알려면, 1-888-426-4343.

제품 정보

미국에 사시는 분은 다음 번호 중 하나를 선택하여 전화하십시오.

- 제품 주문이나 일반 정보를 얻으려면, 1-800-IBM-CALL (1-800-426-2255)이나 1-800-3IBM-OS2 (1-800-342-6672).
- 책에 대한 주문은 1-800-879-2755.

http://www.ibm.com/software/data/

DB2 월드 와이드 웹 페이지에는 새로운 소식, 제품 설명, 교육 일정 등에 관한 현재의 DB2 정보를 제공합니다.

http://www.ibm.com/software/data/db2/library/

DB2 제품 및 서비스 기술 라이브러리는 빈도 높은 질문(FAQ), 수정사항(fixes), 책 및 최신 DB2 기술 정보에 대한 액세스를 제공합니다.

주: 이러한 정보는 영어로만 제공됩니다.

http://www.elink.ibmlink.ibm.com/pbl/pbl/

여기에서는 책을 웹 사이트에서 주문할 수 있는 방법을 제공합니다.

© Copyright IBM Corp. 1993, 2000

http://www.ibm.com/education/certify/

IBM 웹 사이트에서 기술 전문 인증 프로그램은 DB2를 포함하여 다른 IBM 제품의 기술 전문 인증 테스트 정보를 제공합니다.

ftp.software.ibm.com

anonymous로 로그인하십시오. /ps/products/db2 디렉토리에서, DB2와 많은 관련 제품에 관한 데이터, 수정사항, 도구 등을 찾을 수 있습니다.

comp.databases.ibm-db2, bit.listserv.db2-l

이러한 인터넷 뉴스 그룹으로 사용자는 DB2 제품에 대한 자신의 사용 경험을 토론할 수 있습니다.

Compuserve에서, GO IBMDB2

이 명령을 입력하여 IBM DB2 계열 포럼을 액세스하십시오. 모든 DB2 제품이 이러한 포럼을 통해 지원됩니다.

미국 외 지역에서 IBM에 연락하는 방법에 관한 정보는 *IBM Software Support Handbook*의 Appendix A를 참조하십시오. 이 문서에 액세스하려면, 웹 사이트 http://www.ibm.com/support/로 가서 페이지 맨 밑에 있는 IBM Software Support Handbook 링크를 클릭하십시오.

주: 일부 국가에서는 IBM 공인 딜러는 IBM 지원 센터 대신 해당 딜러 지원 부서에 연락해야 합니다.

IBM 한글 지원에 관한 설문

FAX : (02) 781-7778

보내 주시는 의견은 더 나은 고객 지원 체제를 위한 귀중한 자료가 됩니다.
독자 여러분의 좋은 의견을 기다립니다.

책 제목: IBM® DB2® Connect
사용자 안내서
버전 7

책 번호:SA30-0993-00

성 명		직위/담당업무	
회 사 명		부 서 명	
주 소			
전화번호		팩스번호	
전자우편 주소			
사용중인 시스템	☐ 중대형 서버 ☐ UNIX 서버 ☐ 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.

Printed in Australia

SA30-0993-00