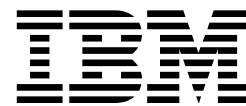


IBM DB2 Connect



# Ръководство за потребителя

*Версия 7*



IBM DB2 Connect



# Ръководство за потребителя

*Версия 7*

Преди да използвате тази информация и продукта, за който тя се отнася, задължително прочетете общата информация със заглавие Приложение Н, “Забележки” на страница 193.

Този документ съдържа информация, която е собственост на IBM. Той се предоставя съгласно лицензно споразумение и е защитен от закона за авторските права. Информацията в тази публикация не включва никакви гаранции за продукта и нито едно направено в ръководството твърдение не трябва да се тълкува като някаква гаранция.

Може да поръчате тези ръководства чрез представителството или офиса на IBM във вашата страна или като позвъните на телефон 1-800-879-2755 в САЩ или на телефон 1-800-IBM-4YOU в Канада.

Когато изпращате информация до IBM, вие предоставяте на IBM правото да ползва или разпространява тази информация по всякакъв начин, който фирмата счита за подходящ, без това да води до никакви задължения към вас.

© Авторско право International Business Machines Corporation 1993, 2000. Всички права запазени.

# Съдържание

За тази книга	vii
Кой трябва да прочете тази книга	vii

## Част 1. Въведение в DB2 Connect 1

### Глава 1. Запознаване с DB2 Connect 3

Концепцията база данни	4
Настройка на DB2 Connect	5
DB2 Connect и SQL	5
Средства за администриране	6

### Глава 2. Архитектура на разпределена реляционна база данни 7

DRDA и DB2 Connect	7
Отдалечена единица работа	8
Разпределени заявки	9
Активиране на Многосайтовото обновяване (Двуфазов протокол за записване на промените)	10
Сценарии за многосайтово обновяване на хост и AS/400 машини, за които е необходим SPM	11
DRDA и достъп до данни	13
Използване на Центъра за управление за разрешаване на многосайтово обновяване	14
Електронни издания, свързани с DRDA	15

### Глава 3. Сценарии за използване на DB2 Connect 17

Директен достъп до база данни	18
DB2 Connect Enterprise Edition като сървър за свързване	20
DB2 Connect и Web приложения	21
Предимства и ограничения на традиционното CGI програмиране	21
DB2 Connect на Web сървър	22
DB2 Connect като Java сървър на приложения	23
Net.Data	24
IBM WebSphere	24
Използване на DB2 Connect със сървъри на приложения	25
Решение Сървър на приложения	26
Сървъри на приложения и DB2 Connect	27
DB2 Connect и конфигуриране на сървър на приложения	28
Използване на DB2 Connect с монитори на обработка на транзакции	29
Примери на TP монитори	30
Tuxedo и DB2 Connect	31
X/Open Distributed Transaction Processing (DTP) модел	31
Как да се използва DB2 Connect с XA съвместим Мениджър на транзакции	31

### Глава 4. Програмиране в DB2 Connect обкръжение 33

Програмиране в разпределена среда	33
Използване на Data Definition Language (DDL)	34
Използване на Data Manipulation Language (DML)	34
Използване на Data Control Language (DCL)	35
Свързване и прекъсване на връзката	35
Предкомпилиране	36
Дефиниране на ред за сортиране	38
Управление на референциален интегритет	38
Заклучване	39
Разлики в SQLCODE и SQLSTATE	39
Използване на системни каталози	39
Преплъвания при числено конвертиране вследствие присвояване при извличане	40
Нива на изолация	40
Запомнени процедури	41
НЕ АТОМАРЕН SQL блок	43
Многосайтово обновяване с DB2 Connect	44
SQL оператори за хост или AS/400 сървър, поддържани от DB2 Connect	45
SQL оператори за хост или AS/400 сървър, отхвърлени от DB2 Connect	45
Реализиране на обратно таксуване на DB2 Universal Database за OS/390	46
Изпращане на информация за отчитане към DB2 за OS/390 сървър	47
Определяне на низът на отчитане	48
Полезни издания	48

### Глава 5. Изпълнение на ваши собствени приложения 51

Свързване на помощните програми на база данни	51
Изпълнение на CLI/ODBC програми	52
Специфична за платформата подробна информация за CLI/ODBC достъп	53
Подробна конфигурационна информация	56
Изпълнение на Java програми	56
Конфигуриране на обкръжението	57
Java приложения	59
Java аплети	59

## Част 2. Справочник и отстраняване на проблеми 61

### Глава 6. Обновяване на директории на бази данни 63

Събиране на информация	63
Директория на възлите	63
DCS Директория	65
Системна директория на база данни	71
Задаване на множество записи	72
Обновяване на директорииите	72

<b>Глава 7. Свързване на приложения и помощни програми</b>	<b>75</b>
Командата BIND	79
Повторно свързване	79

<b>Глава 8. Системен монитор на базата данни</b>	<b>81</b>
Следене на свързванията от отдалечени клиенти	81
Включване на параметрите за наблюдение за DB2 Connect	81
Представяне на състоянието на параметрите за наблюдение	82
Използване на командите GET SNAPSHOT	82
Представяне състоянието на DCS приложение	85
LIST DCS APPLICATIONS	85
LIST DCS APPLICATIONS SHOW DETAIL	86
LIST DCS APPLICATIONS EXTENDED	88
Използване на DB2 Центъра за управление за представяне на разширена информация за DCS приложения	89
Използване на Монитор на производителността на Windows	89

<b>Глава 9. Средства за администриране</b>	<b>91</b>
Процесор за обработка на команди	91
Използване на средствата за импорт и експорт	92
Преместване на данни от работна станция на S/390 или AS/400 сървър на база данни	92
Преместване на данни от DRDA сървър в работна станция	92
Смесени еднобайтови и двубайтови данни	93
Заменяне на Помощната програма SQLQMF	93

<b>Глава 10. Защита</b>	<b>95</b>
Разпознаване	95
Типове защита	97
Типове защита при APPC свързване	97
Типове защита при TCP/IP свързване	98
Дискусия за типовете защита	98
Променяне на MVS парола	99
Конфигуриране на DB2 Connect работна станция за Управление валидността на паролите	99
Конфигуриране на хоста за Управление валидността на паролите	100
Допълнителни препоръки и съвети за защитата	101
Кодове за разширена защита	101
TCP/IP предварителна проверка за защита	101
Защита на ODBC и Java приложения	101
Поддръжка на смяна на пароли	102

<b>Глава 11. Преобразуване на SQLCODE</b>	<b>103</b>
Изключване на SQLCODE преобразуването	103
Обвързване на SQLCODE преобразуването	103

<b>Глава 12. Производителност</b>	<b>109</b>
Концепции и инструменти за производителността	109

Потоци данни	109
Тесни места	110
Сравнителни характеристики	111
Средства за проверка на производителността	111
Оптимизиране на ODBC достъпа	112
Проектиране на приложения	113
SQL блокове и запомнени процедури	113
Групиране на заявки	114
Предикатна логика	114
Създаване на блокове с данни	115
Статичен и динамичен SQL	115
Други SQL съображения	116
Настройка на DB2 Connect	116
RQRIOBLK	117
DIR_CACHE	117
Други DB2 Connect параметри	118
Пул за връзки	118
Как работи пултът за връзките	118
DB2 Connect концентратор за връзки	119
Настройка на базата данни	123
Мрежови настройки	126
Конкурентно използване на системните ресурси	129
Отстраняване на проблеми, свързани с производителността	129
Допълнителни препоръки и съвети за настройка на SNA	130
Обща информация за производителността на DB2 Connect	130
Избор и настройка на Мрежовото приложение	130
Други източници на информация за производителността на DB2 Connect	131
Поддръжка на Multi Path Channel за SNA през ESCON	131
Как да се настроят DB2 Connect свързвания през NCP	132
Информация за OSA-2 усъвършенствания	135
Други източници на информация	138
Други издания	138
Използване на World Wide Web	138
Допълнителни препоръки и съвети за SNA потребители	138

<b>Глава 13. Откриване на проблеми</b>	<b>139</b>
Други източници на информация	139
Използване на Ръководството за отстраняване на проблеми	139
Използване на World Wide Web	139
Документация за APPC, CFI-C и SNA кодове на състояние	139
Събиране на подходящата информация	140
Първоначалното свързване не е успешно	140
Проблеми, възникнали след първоначално свързване	141
Средства за диагностика	142
Помощна програма за трасиране (ddcstrc)	143
Синтаксис на трасиране	144
Параметри на трасиране	144
Резултат от трасиране	145
Анализиране на изходния файл от трасирането	146
Най-разпространени проблеми с DB2 Connect	150
SQL0965 или SQL0969	151
SQL1338 по време на CONNECT	151

SQL1403N по време на CONNECT	152
SQL5043N	152
SQL30020	153
SQL30060	153
SQL30061	154
SQL30073 с код на връщане 119C при CONNECT	154
SQL30081N с код на връщане 1	155
SQL30081N с код на връщане 2	156
SQL30081N с код на връщане 9	156
SQL30081N с код на връщане 10	157
SQL30081N с код на връщане 20	157
SQL30081N с код на връщане 27	157
SQL30081N с код на връщане 79	158
SQL30081N със специфичен за протокола код за грешка 10032	158

### Част 3. Приложения . . . . . 161

#### Приложение А. Функции, характерни и за предишните версии . . . . . 163

DB2 Connect версия 6 подверсия 1	163
DB2 Connect версия 5 подверсия 2	163
DB2 Connect версия 5.0	164
DDCS версия 2 подверсия 4	165
DDCS версия 2 подверсия 3	166

#### Приложение В. Работен лист за персонализиране на директории . . . . . 167

#### Приложение С. Съображения за поддръжка на национални езици . . . . . 169

Преобразуване на символни данни	169
---------------------------------	-----

#### Приложение D. Използване на DCE директорните услуги . . . . . 173

Създаване на обект в базата данни	174
-----------------------------------	-----

Създаване на обект локатор в базата данни	175
Създаване на обект с маршрутизираща информация	177
Настройка на конфигурационните параметри	178
Каталогизиране на базата данни	178
Защита с DCE директорни услуги	179

#### Приложение Е. Средства за свързване на клиенти от предишни версии . . . . . 183

#### Приложение F. Настройка работата на CLI/ODBC приложение с помощта на ключовата дума CLISHEMA . . . . . 185

Обкръжение приемник	185
CLI/ODBC	185
DB2 инициализиращата ключова дума CLISHEMA	186
Използване	186
Помощните програми db2cli и bldscem	187
Предлаган подход	188
Допълнителни препоръки и съвети	188
db2ocat оптимизатор за каталози	189
Допълнителни източници на информация	189

#### Приложение G. Източници с допълнителна информация . . . . . 191

Други свързани издания	191
------------------------	-----

#### Приложение H. Забележки . . . . . 193

Търговски марки	195
-----------------	-----

#### Индекс . . . . . 197

#### Свързване с IBM . . . . . 199

Информация за продукти	199
------------------------	-----





---

## За тази книга

Тази книга съдържа информация за общото използване на следните продукти на IBM DB2 Connect:

- DB2 Connect Personal Edition за OS/2 и Windows 32-битова операционна система.
- DB2 Connect Enterprise Edition (EE) за AIX, HP-UX, Linux, PTX, Solaris, OS/2 и Windows 32-битова операционна система.
- DB2 Connect Unlimited Edition за OS/390.

DB2 Connect: Ръководство на потребителя съдържа три части:

- Част 1, Въведение в DB2 Connect осигурява концептуален преглед на DB2 Connect, Архитектура на разпределена релационна база данни (DRDA) и възможните сценарии за използване.
- Част 2, Справочник и отстраняване на проблеми осигурява информация за обновяване на директории на базата данни, свързани приложения, средства за администриране, DB2 Системен монитор, сигурност, откриване на проблеми и производителност.
- Част 3. Приложения, които осигуряват различна информация, препоръки и съвети.

Освен това тази книга обяснява концепциите, които са свързани с всички DB2 Connect продукти. Информация за специфична платформа можете да потърсите в:

- *DB2 Connect Personal Edition: Бърз старт*, за настройка на еднопотребителска DB2 Connect на OS/2 и Windows 32-битова операционна система.
- *DB2 Connect Personal Edition за Linux Бърз старт*, за настройка на еднопотребителска DB2 Connect за Linux.
- *DB2 Connect Enterprise Edition за OS/2 и Windows – Бърз старт*, за настройка на многопотребителски DB2 Connect шлюз за OS/2 или Windows 32-битова операционна система.
- *DB2 Connect Enterprise Edition за UNIX – Бърз старт*, настройка на многопотребителски DB2 Connect шлюз за AIX, HP-UX, Linux, PTX или Solaris.

---

## Кой трябва да прочете тази книга

Тази книга е предназначена за програмисти и администратори, които отговарят за настройването и поддръжката на DB2 Connect свързвания. Тези свързвания може да са между DB2 клиенти и някой от следните Архитектура на разпределена релационна база данни (DRDA) сървъри на приложения в система за управление на база данни:

- DB2 Universal Database за OS/390 Версия 5 или следваща
- DB2 за MVS Версия 3 или следваща
- DB2 за VSE и VM
- DB2 Universal Database за AS/400
- Всяка друга система за управление на релационна база данни, която реализира функцията DRDA сървър на приложения.

**Забележки:**

1. DB2 Universal Database (DB2 UDB) не изисква DB2 Connect, за да позволи хост или AS/400 приложения да имат достъп до DB2 UDB данни.
2. DB2 Universal Database за OS/390 Версия 5.1 или следваща е необходима, за да се използват функциите DRDA ниво 3, включително TCP/IP свързвания към бази данни и запомнени процедури с групи отговори от по няколко реда.
3. DB2 Universal Database за OS/390 Версия 6.1 или следваща е необходима, за да се използват функции DRDA ниво 4, за които се осигурява поддръжка в DB2 Connect. Тези функции включват поддръжка на формат голям integer, големи обекти (LOB), идентификатор на ред и отделни типове данни, дефинирани от потребителя.

---

## Част 1. Въведение в DB2 Connect



---

## Глава 1. Запознаване с DB2 Connect

DB2 Connect осигурява изключително бързо и устойчиво свързване към IBM мейнфрейм бази данни за електронен бизнес и други приложения, които работят под различни операционни системи, базирани на UNIX или не.

DB2 Connect има различни решения за свързване. DB2 Connect Personal Edition осигурява директно свързване към хост или AS/400 бази данни, докато DB2 Connect Enterprise Edition осигурява косвено свързване, което позволява на клиентите достъп до хост или AS/400 бази данни чрез DB2 Connect сървър. DB2 Connect Unlimited Edition осигурява уникални решения за пакетирание, които улесняват избирането на продукт и лицензирането.

### **DB2 Connect Enterprise Edition**

DB2 Connect Enterprise Edition е сървър за свързване, който концентрира и управлява свързванията от множество настолни клиенти и web приложения към DB2 сървъри за бази данни, които работят върху хост или AS/400 системи. Базите данни на IBM DB2 за AS/400, DB2 за OS/390 и DB2 за VSE & VM продължават да са предпочитани системи за управление на критични данни за най-големите организации в света. Тези хост и AS/400 бази данни управляват данни, но има голяма необходимост да се интегрират данните с приложения, работещи на работни станции с Windows, UNIX и OS/2.

DB2 Connect Enterprise Edition позволява на локални и отдалечени клиентски приложения да създават, обновяват, контролират и управляват DB2 бази данни и хост системи с помощта на програмните интерфейси Structured Query Language (SQL), DB2 API (Application Programming Interfaces), ODBC (Open Database Connectivity), JDBC (Java Database Connectivity), SQLJ (Embedded SQLJ for Java) или DB2 CLI (Call Level Interface). Освен това DB2 Connect поддържа интерфейсите за данни на Microsoft Windows като ActiveX Data Objects (ADO), Remote Data Objects (RDO) и OLE DB.

DB2 Connect Enterprise Edition в момента се доставя за AIX, HP-UX, Linux, OS/2, PTX, Solaris, и Windows 32-битова операционна система. Тези сървъри осигуряват поддръжка за приложения, които работят на работни станции под OS/2, UNIX (AIX, HP-UX, Linux, PTX, Solaris, Silicon Graphics IRIX) и 32-битов Windows.

### **DB2 Connect Personal Edition**

DB2 Connect Personal Edition осигурява достъп от дадена работна станция до DB2 бази данни, разположени на сървъри като MVS/ESA, OS/390, OS/400, VM и VSE, както и достъп до DB2 Universal Database сървъри на OS/2, UNIX и Windows 32-битова операционна система. DB2 Connect Personal Edition осигурява същото богато множество от API като DB2 Connect Enterprise Edition, като освен това се отличава с интегрирана поддръжка на SNA на всички Windows платформи.

Този продукт в момента се доставя за OS/2, Linux и Windows 32-битова операционна система.

### **DB2 Connect Unlimited Edition**

DB2 Connect Unlimited Edition е уникален пакет, който предлага комплексна

гъвкавост на DB2 Connect развой и опростява избирането на продукта и лицензирането. Този продукт съдържа както DB2 Connect Personal Edition, така и DB2 Connect Enterprise Edition с лицензионни условия, които позволяват неограничен развой на DB2 Connect продукти. Лицензионните такси се определят от размера на System/390, с който потребителите на DB2 Connect ще работят.

Този нов пакет се предлага само за OS/390 системи и лицензирането е валидно само за DB2 за OS/390 източници на данни.

---

## Концепцията база данни

Терминът *база данни* се използва в книгата в смисъл на система за управление на реляционна база данни (RDBMS – relational database management system). Другите системи, с които DB2 Connect комуникира, може да използват термина база данни с малко по-различен смисъл. Освен това DB2 Connect терминът база данни може да се отнася и за:

### **MVS (Версия 4 и предишна)**

DB2 за MVS/ESA подсистема идентифицирана от своето ИМЕ НА МЯСТО.

ИМЕТО НА МЯСТОТО може да се определи след свързване към TSO и генериране на следната SQL заявка с помощта на някое от достъпните средства:

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

ИМЕТО НА МЯСТОТО освен това се дефинира в Boot Strap Data Set (BSDS) и се съдържа в съобщението DSNL004I (LOCATION=място), което се записва при стартирането на Distributed Data Facility (DDF).

### **OS/390 (версия 5 и следваща)**

DB2 Universal Database за OS/390 подсистема, идентифицирана от своето ИМЕ НА МЯСТО.

ИМЕТО НА МЯСТОТО може да се определи след свързване към TSO и генериране на следната SQL заявка с помощта на някое от достъпните средства:

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

ИМЕТО НА МЯСТОТО освен това се дефинира в Boot Strap Data Set (BSDS) и се съдържа в съобщението DSNL004I (LOCATION=място), което се записва при стартирането на Distributed Data Facility (DDF).

**VSE** DB2 за VSE работеща в дял и идентифицирана от своето DBNAME

**VM** DB2 за VM, работеща на CMS виртуална машина и идентифицирана от своето DBNAME

### **OS/400**

DB2 Universal Database за AS/400, неделима част на операционната система OS/400. На една AS/400 машина може да съществува само една база данни. Ако базата данни ще се използва от приложения извън AS/400 системата, базата данни трябва да има име в директорията на реляционната база данни. Това име е известно като Име на реляционна база данни (RDB име – Relation Database Name).

За да видите RDB името на вашата AS/400 система, изпълнете командата **WRKRDBDIRE**. Срещу RDB името на локалната система в колоната

Отдалечено място се записва \*LOCAL. За да промените RDB име, използвайте командата CHGRDBDIRE.

---

## Настройка на DB2 Connect

Преди да използвате DB2 Connect, трябва да изпълните следните стъпки:

Стъпка 1. Инсталирайте DB2 Connect и конфигурирайте както хост или AS/400 сървър, така и комуникациите на работната станция, както е описано в съответната книга *DB2 Connect Бърз старт* или в *Приложение за инсталиране и конфигуриране*.

Стъпка 2. Обновете директориите на базата данни, както е описано в Глава 6, “Обновяване на директории на бази данни” на страница 63.

**Забележка:** На OS/2 и Windows 32-битова операционна система препоръчваме да използвате Асистент за конфигуриране на клиенти (CCA).

На всички други платформи директориите на базата данни трябва да се обновят с помощта на DB2 Процесорът на команди (CLP). И двата подхода са описани в *Приложение за инсталиране и конфигуриране*.

Стъпка 3. Свържете DB2 Connect помагалата към всяка хост или AS/400 система за управление на база данни, както е описано в Глава 7, “Свързване на приложения и помощни програми” на страница 75.

Тази задача може да се изпълни и с помощта на CCA или диалога за настройка на източниците за данни, където е осигурен.

---

## DB2 Connect и SQL

DB2 Connect препраща SQL оператори, изпратени от приложни програми към хост или AS/400 сървъри на бази данни. DB2 Connect може да препрати почти всеки валиден SQL оператор. Изключенията са документирани в “SQL оператори за хост или AS/400 сървър, отхвърлени от DB2 Connect” на страница 45.

Съществуват два типа обработка на вграден SQL: статичен SQL и динамичен SQL. Статичният SQL минимизира времето, необходимо за изпълнение на SQL оператор с помощта на предварителна обработка. Динамичният SQL се обработва, когато SQL операторът се предаде на хост или AS/400 сървър на база данни. Динамичният SQL е по-гъвкав, но потенциално е по-бавен. Приложният програмист взема решението дали да използва статичен или динамичен SQL. И двата се поддържат от DB2 Connect.

Различните хост или AS/400 сървъри на бази данни интерпретират по различен начин SQL. Повече информация за общите SQL оператори, които се поддържат от всички системи на IBM, ще намерите в *SQL Справочник*.

DB2 Connect изцяло поддържа общия SQL на IBM, както и вариантите на SQL за DB2 Universal Database за OS/390, DB2 за MVS/ESA, DB2 за VSE и VM (по-рано SQL/DS) и DB2 Universal Database за AS/400. За да се осигури независимост на базата данни, се препоръчва SQL на IBM. За повече информация вижте Глава 4, “Програмиране в DB2 Connect обкръжение” на страница 33.

---

## Средства за администриране

Следните помагала са в помощ на администраторите на DB2 Connect:

- Процесор за обработка на команди ви позволява да съставите SQL изрази спрямо хост или AS/400 сървър на база данни. Насочва SQL изразите към определената от вас база данни.
- DB2 Command Center осигурява графичен интерфейс към Процесор за обработка на команди.
- Помагалата за импортиране и експортиране ви позволяват да зареждате, импортирате и експортирате данни от и до файл на работна станция и база данни на хост или AS/400 сървър на база данни. Тези файлове могат след това да се използват за импортиране на данни в бази данни, електронни таблици и други приложения, работещи на работната станция. Повече информация за средствата за импортиране и експортиране ще намерите в *Ръководство и справочник за помощните средства за преместване на данни*.
- Потребителите на DB2 Connect Enterprise Edition под Windows NT и Windows 2000 могат да използват Event Viewer и Performance Monitor. С помощта на Event Viewer можете да видите отделни събития, записани от DB2 Connect. С помощта на Performance Monitor можете да следите и управлявате производителността на DB2 Connect сървъри както локално, така и отдалечено.
- DB2 Центърът за управление ви позволява да администрирате и наблюдавате всички страни на DB2 Connect сървъри. Освен това позволява на администраторите да работят с обекти на DB2 за OS/390 бази данни, като таблици, производни таблици, буфери и канали. Допълнителна информация за управление на системи DB2 за OS/390 от DB2 Център за управление потърсете в *Ръководство за разработка на приложения*.

Допълнителна информация за тези помощни програми вижте в Глава 9, “Средства за администриране” на страница 91.

Допълнително помощната програма системен монитор за базата данни дава възможност на системния администратор да следи връзките в системата. Освен това му помага да определи източника на грешка. Системният администратор може да свърже клиентските приложения със съответните задания, работещи на хост или AS/400 сървъра на база данни. За повече информация вижте Глава 8, “Системен монитор на базата данни” на страница 81.



---

## Глава 2. Архитектура на разпределена реляционна база данни

Архитектура на разпределена реляционна база данни (DRDA) е множество от протоколи, които дават възможност за съвместна работа на няколко системи на бази данни, на IBM или не, както и на приложни програми. Всяка комбинация от продукти за управление на реляционна база данни, които използват DRDA могат да се свържат към форма на разпределена система за управление на реляционна база данни. DRDA координира комуникацията между системите, като определя какво и как трябва да се обменя.

Когато разглеждаме DB2 Connect, често ще срещаме термина единица работа. *Единица работа (UOW)* представлява отделна логическа операция. Състои се от последователност от SQL оператори, в които всички операции са изпълнени успешно или последователността като цяло се разглежда като неуспешна.

Друга важна концепция е разпределена единица работа, също така известна като многосайтово обновяване. *Разпределената единица работа (DUOW)* включва повече от един сървър на база данни в рамките на единица работа. По-специално дефинираме *многосайтовото обновяване* като операция със следните характеристики:

- На единица работа се обновява повече от един сървър за управление на база данни.
- Приложението нарежда разпределението на работа и активира записване на данни.
- Може да има няколко заявки на единица работа.
- За всяка заявка има по един сървър за управление на база данни.
- Записването се координира в рамките на няколко сървъра на база данни.

Допълнителна информация за многосайтово обновяване вижте в “Активиране на Многосайтовото обновяване (Двуфазов протокол за записване на промените)” на страница 10.

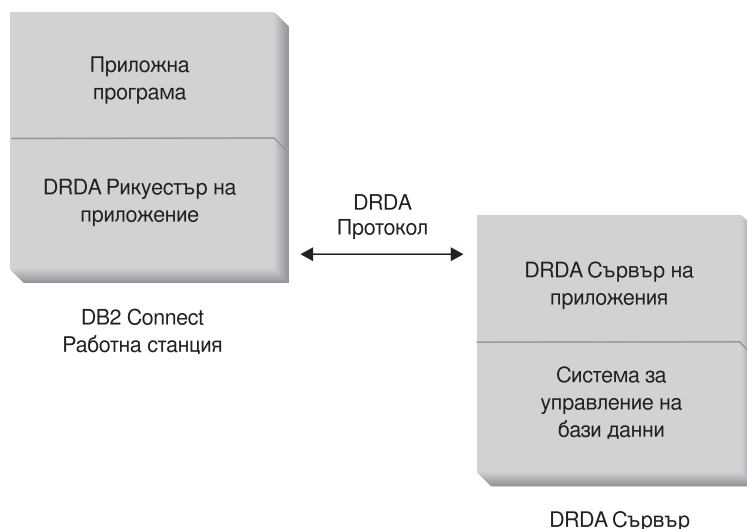
---

### DRDA и DB2 Connect

DB2 Connect използва DRDA архитектура, за да намали разходите и сложността при достъпа на данни, записани в DB2 Universal Database за AS/400, DB2 Universal Database за OS/390, DB2 за MVS/ESA, DB2 за VSE и VM и други съвместими с DRDA сървъри на бази данни. Като се възползва изцяло от DRDA архитектурата, DB2 Connect предлага добре работещо, евтино решение с необходимите за клиента възможности за управление на системата.

В DRDA терминологията *средство за обработка на запитвания (AR)* е кодът, който се грижи за отдалеченото свързване в разпределената система от страната на приложението; на практика приложението заявява данните. *Сървър на приложения (AS)* е кодът, който се грижи от страна на базата данни за отдалеченото свързване. В DB2 Connect среда само DB2 Connect работна станция може да функционира като средство за обработка на запитвания от името на приложните програми.

Фигура 1 на страница 8 показва движението на данните между DB2 Connect работна станция и DRDA сървър, когато има само локални клиенти. Освен това съществува отделен протокол между DB2 Connect работна станция и отдалечени клиенти.



Фигура 1. Поток данни между DB2 Connect работна станция и DRDA сървър

За да реализира свързването между системи за управление на база данни с DRDA сървър и клиент на базата данни, DRDA използва следните архитектури:

- Архитектура за представяне на символни данни (CDRA)
- Архитектура за разпределено управление на данни (DDM)
- Архитектура на обекти форматиранни данни (FD:OCA)
- Архитектура на мрежови системи (SNA)
- Архитектура за управление на SNA услуги (MSA)
- Transmission Control Protocol/Internet Protocol (TCP/IP).

Тези архитектури се използват като градивни блокове. Потоците данни, които преминават през мрежата се определят от DRDA архитектурата, която документира протокола на потока данни, поддържащ достъпа до разпределената реляционна база данни.

Заявката се препраща на правилното място с помощта на директори, които съдържат различни типове комуникационна информация и името на DRDA сървъра на базата данни, до която се осъществява достъп.

## Отдалечена единица работа

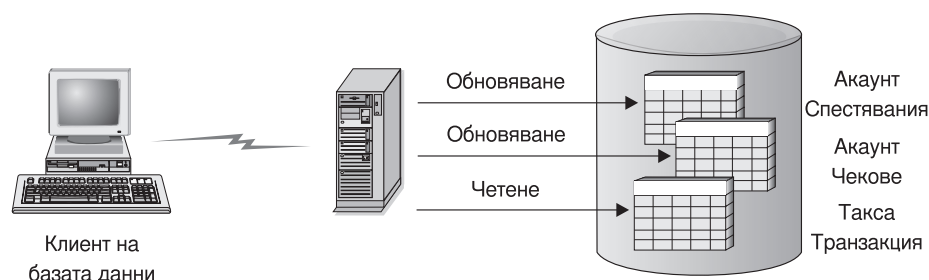
*Отдалечената единица работа* позволява на потребителя или приложната програма да прочете или обнови данни от едно място на всяка единица работа. Поддържа достъп до една база данни в рамките на единица работа. Въпреки че приложна програма може да обнови няколко отдалечени бази данни, тя може да осъществи достъп само до една база данни в рамките на единица работа.

Отдалечената единица работа има следните характеристики:

- Поддържат се по няколко заявки (SQL изрази) на единица работа.
- Поддържат се по няколко указатели на единица работа.
- Всяка единица работа може да обнови само една база данни.
- Приложната програма извършва комит или ролбек за единицата работа. При определени състояния на грешки сървъра на база данни или DB2 Connect може да извърши ролбек на единица работа.

Например, Фигура 2 на страница 9 показва клиент на базата данни, който изпълнява приложение за финансови операции и осъществява достъп до база данни с таблици на разплащателни и спестовни сметки, както и банкови такси. Приложението трябва:

- Да приеме размера на трансфера от потребителския интерфейс.
- Да извади сумата от спестовната сметка и да определи новия баланс.
- Да прочете предвидените такси и да определи таксата за операцията за спестовната сметка спрямо дадения баланс.
- Да извади таксата за операцията от спестовната сметка.
- Да прибави сумата на трансфера към разплащателната сметка.
- За запише операцията (единицата работа).



Фигура 2. Използване на една база данни в транзакция

За да настроите такова приложение, трябва:

1. Да създадете таблици за спестовната сметка, разплащателна сметка и за банкови такси в една и съща база данни, както е описано в *Ръководство за администриране*.
2. Ако физически е отдалечена, настройте сървъра на база данни да използва съответните комуникационни протоколи, както е описано в книгите *Бърз старт*
3. Ако физически е отдалечена, каталогизирайте възела и базата данни да идентифицират базата данни на сървъра за база данни, както е описано в книгите *Бърз старт*
4. Да компилирате приложната програма, за да определите свързване от тип 1; което означава да определите CONNECT(1) в команда PREP, както е описано в *Ръководство за разработка на приложения*.

## Разпределени заявки

*Разпределената заявка* е функция на разпределена база данни, която позволява на приложения и потребители да предават SQL изрази, които се обръщат към две или повече DBMS или бази данни в един израз. Например команда за сливане (join) между таблици в две различни подсистеми DB2 за OS/390.

DB2 Connect версия 7 осигурява поддръжка за разпределени запитвания спрямо бази данни и DBMS. Например можете да изпълните операция UNION между DB2 таблица и производна таблица на Oracle. Сред поддържаните DBMS са членове на фамилията DB2 (като DB2 UDB за Windows, UNIX и OS/2, DB2 за OS/390 и DB2 за AS/400) и Oracle.

Разпределеното запитване осигурява *прозрачност за мястото* по отношение на обектите в базата данни. Ако се премести информация (в таблици) обръщанията към тази информация (наречени *пъкори*) могат да се обновят, без да се променят приложенията, за които е необходима тази информация. Освен това разпределеното

запитване осигурява *компенсация* за онези DBMS, които не поддържат всички SQL диалекти на DB2 или определени възможности за оптимизиране. Операциите, които не могат да се изпълнят под такава DBMS (като рекурсивен SQL) се изпълняват под DB2 Connect.

Разпределеното запитване действа по *почти автономен* начин. Например DB2 запитвания, които съдържат обръщения към обекти в Oracle могат да се предават, докато Oracle приложения се обръщат към същия сървър. Разпределеното запитване не монополизира и не ограничава достъпа (освен ограниченията за интегритет и заключване) до обекти на Oracle или на други DBMS.

Реализирането на функцията за разпределено запитване се състои от потребителски модел на DB2 Connect версия 7, база данни, която ще служи като обединена и един или повече отдалечени източници на данни. *Обединената база данни* съдържа каталог с елементи, които идентифицират източниците на данни и техните характеристики. *Източникът на данни* се състои от DBMS и данни. Приложенията се свързват към обединена база данни точно като всяка друга DB2 база данни. DB2 Connect обединена база данни не се лицензира за управление на потребителски данни. Единствената цел е да съдържа информация за източниците на данни.

След установяването на обединената система, информацията в източниците на данни може да се достигне като че ли е в една голяма база данни. Потребителите и приложенията изпращат запитвания към обединената база данни, която след това извлича необходимите данни от системи от DB2 фамилията или Oracle. Потребителят и приложенията определят прякори в запитванията; тези прякори осигуряват обръщения към таблици – основни и производни, разположени в източниците на данни. От гледна точка на крайния потребител прякорите са подобни на псевдоними.

Много фактори могат да повлияят върху изпълнението на разпределените запитвания. Най-критичният фактор е да се осигури, че в глобалния каталог на обединената база данни е записана акуратна и актуална информация за източниците на данни и техните обекти. Тази информация се използва от DB2 оптимизатора и може да повлияе върху решения, да подложи операции на оценка на източниците на данни. Повече информация за производителността на обединена система можете да намерите в *Ръководство за администриране: Производителност*.

---

## Активиране на Многосайтовото обновяване (Двуфазов протокол за записване на промените)

Функцията Многосайтово обновяване, позната също и като Разпределена единица работа (distributed unit of work –DUOW) и Двуфазов протокол за записване на промените, е функция, която позволява на вашите приложения да актуализират данни в множество отдалечени сървъри на бази данни при гарантиран интегритет. Например банкова транзакция, която включва трансфер на пари от една сметка в друга, намираща се в различен сървър на базата данни.

При подобна операция е съществено актуализациите за дебитната операция да не се запишат, докато не се запишат също и актуализациите, необходими да се кредитира другата сметка. Операцията многосайтово обновяване се прилага, когато данни, представляващи тези сметки, се управляват от два различни сървъра на базата данни.

DB2 продуктите осигуряват пълна поддръжка за многосайтово обновяване. Тази поддръжка е налична за приложения, разработени на стандартен SQL, както и за приложения, използващи продукти за следене на транзакции, които реализират спецификацията на интерфейса X/Open XA. Примери за такива продукти за следене

на транзакции са IBM TxSeries (CICS и Encina), Message and Queuing Series, Component Broker Series, San Francisco Project, както и Microsoft Transaction Server (MTS), BEA Tuxedo и други. Изискванията за настройка са различни в зависимост от това дали се използва многосайтово обновяване на собствен SQL или на TP Монитор.

И двете програми за многосайтово обновяване собствен SQL и TP Монитор трябва да се предкомпилират с опциите CONNECT 2 SYNCPOINT TWOPHASE. И двете могат да използват оператора SQL Connect, за да посочат коя база данни да се използва за SQL операторите, които следват след това. Ако няма TP Монитор, който да укаже на DB2 да координира транзакцията (както се указва от DB2, получаваща обръщението ха\_ореп от TP Монитор, за да се установи свързване към базата данни), тогава DB2 софтуерът ще се използва за координиране на транзакцията.

Когато се използва многосайтово обновяване с TP Монитор, приложението трябва да заяви комит или ролбек, като използва приложния интерфейс на TP монитор, например CICS SYNCPOINT, Encina Abort(), MTS SetAbort(). Когато се използва многосайтово обновяване на собствен SQL, трябва да се зададат нормалните SQL COMMIT и ROLLBACK.

Многосайтово обновяване чрез TP Монитор може да координира транзакция, която осъществява достъп до DB2 и до не-DB2 мениджъри на ресурси, като Oracle, Informix, SQLServer и т.н. Многосайтово обновяване на собствен SQL се използва само с DB2 сървъри. Многосайтово обновяване на собствен SQL се използва само с DB2 сървъри.

За да работи транзакция за многосайтово обновяване, всяка от базите данни, участващи в разпределената транзакция, трябва да може да поддържа Разпределена единица работа. В момента следните DB2 сървъри осигуряват поддръжка на DUOW, което им позволява да участват в разпределени транзакции:

- DB2 Universal Database V5, V6 и V7
- DB2 за MVS/ESA V3.1 и 4.1
- DB2 за OS/390 V5.1
- DB2 Universal Database за OS/390 V6.1
- DB2/400 V3.1 или по-нова (като за момента използва само SNA архитектура)
- DB2 Server за VM и VSE V5.1 и 6.1
- Сървър на база данни 4

Разпределената транзакция може да актуализира всяка комбинация от поддържани сървъри на бази данни. Например вашето приложение може да актуализира в рамките на една транзакция няколко таблици в DB2 Universal Database on Windows NT, DB2 за OS/390 база данни и DB2/400 база данни.

## **Сценарии за многосайтово обновяване на хост и AS/400 машини, за които е необходим SPM**

Хост и AS/400 сървъри на база данни изискват DB2 Connect, за да участват в разпределена транзакция, заявена от приложения за PC, UNIX и Web. Освен това много от сценариите за многосайтово обновяване, които включват хост и AS/400 сървъри на база данни, изискват да бъде конфигуриран компонента Мениджър за синхронизация (Syncpoint Manager – SPM).

Необходимостта от SPM се определя от избора на протокол (SNA или TCP/IP) и от използването на TP монитор. В следващата таблица е дадено резюме на сценариите, които изискват използването на SPM. Таблицата показва, че DB2 Connect е необходима при всяко осъществяване на достъп от Intel или UNIX

машини до хост или AS/400 машини. Освен това компонентът SPM на DB2 Connect е необходим при многосайтово обновяване, ако достъпът се осъществява през SNA архитектура или се използва TP монитор.

Таблица 1 (Страница 1 от 2). Сценарии за многосайтово обновяване на хост и AS/400 машини, за които е необходим SPM.

Използва ли се TP Монитор?	Протокол	Необходим ли е SPM?	Необходим продукт (изберете един)	Поддържани хост и AS/400 бази данни
Да	TCP/IP	Да	<ul style="list-style-type: none"> <li>DB2 Connect Enterprise Edition</li> <li>DB2 Universal Database Enterprise Edition</li> <li>DB2 Universal Database Enterprise-Extended Edition</li> </ul>	<ul style="list-style-type: none"> <li>DB2 за OS/390 V5.1</li> <li>DB2 Universal Database за OS/390 V6.1</li> </ul>
Да	SNA	Да	<ul style="list-style-type: none"> <li>DB2 Connect Enterprise Edition*</li> <li>DB2 Universal Database Enterprise Edition*</li> <li>DB2 Universal Database Enterprise-Extended Edition*</li> </ul> <p><b>Забележка:</b> *само платформите AIX, OS/2, Windows NT и Windows 2000.</p>	<ul style="list-style-type: none"> <li>DB2 за MVS/ESA V3.1 и 4.1</li> <li>DB2 за OS/390 V5.1</li> <li>DB2 Universal Database за OS/390 V6.1</li> <li>DB2/400 V3.1 или по-нова</li> <li>DB2 Server за VM или VSE V5.1 или по-нова</li> </ul>
Не	TCP/IP	Не	<ul style="list-style-type: none"> <li>DB2 Connect Personal Edition</li> <li>DB2 Connect Enterprise Edition</li> <li>DB2 Universal Database Enterprise Edition</li> <li>DB2 Universal Database Enterprise-Extended Edition</li> </ul>	<ul style="list-style-type: none"> <li>DB2 за OS/390 V5.1</li> <li>DB2 Universal Database за OS/390 V6.1</li> </ul>

Таблица 1 (Страница 2 от 2). Сценарии за многосайтово обновяване на хост и AS/400 машини, за които е необходим SPM.

Използва ли се TP Монитор?	Протокол	Необходим ли е SPM?	Необходим продукт (изберете един)	Поддържани хост и AS/400 бази данни
Не	SNA	Да	<ul style="list-style-type: none"> <li>DB2 Connect Enterprise Edition*</li> <li>DB2 Universal Database Enterprise Edition*</li> <li>DB2 Universal Database Enterprise-Extended Edition*</li> </ul> <p><b>Забележка:</b> *само платформите AIX, OS/2, Windows NT и Windows 2000.</p>	<ul style="list-style-type: none"> <li>DB2 за MVS/ESA V3.1 и 4.1</li> <li>DB2 за OS/390 V5.1</li> <li>DB2 Universal Database за OS/390 V6.1</li> <li>DB2/400 V3.1 или по-нова</li> <li>DB2 Server за VM и VSE V5.1 или по-нова</li> </ul>

**Забележка:** Разпределената транзакция може да актуализира всяка комбинация от поддържани сървъри на бази данни. Например вашето приложение може да актуализира в рамките на една транзакция няколко таблици в DB2 UDB Database на Windows NT, DB2 за OS/390 база данни и DB2/400 база данни.

Повече информация за двуфазово записване на промените, както и инструкции за настройка на няколко популярни TP монитори потърсете в:

- *Ръководство за администриране*
- *Ръководство за настройка и работа със CICS за AIX*

Освен това можете да осъществите достъп до DB2 Техническа библиотека за продукти и услуги в World Wide Web:

1. Отидете на следната страница в Web:  
<http://www.ibm.com/software/data/db2/library/>
2. Изберете връзката **DB2 Universal Database**.
3. Търсете "Технически бележки" ("Technotes"), като използвате ключови думи за търсене "DDCS", "SPM", "MTS", "CICS" и "ENCINA".

## DRDA и достъп до данни

Въпреки че DRDA дефинира комуникационните протоколи в базата данни, не дефинира програмните интерфейси, или API, които трябва да се използват от приложните програмисти. Като цяло DRDA може да се използва от приложна програма за предаване на заявка, която съответния DRDA сървър може да изпълни. Всички използвани сега DRDA сървъри могат да изпълняват SQL заявки, насочени от приложна програма чрез DB2 Connect.

IBM предоставя на приложните програмисти средства, с които да генерират SQL заявки за Windows, OS/2 и редица UNIX платформи. Тези средства са част от Клиент за разработка на DB2 приложения. Клиент за разработка на DB2 приложения поддържа редица типове API: вграден SQL, JDBC, SQLJ и DB2 интерфейс за вградени SQL оператори (DB2 CLI). Тези API могат да се използват от програмистите при изграждане на приложения с помощта на много езици за програмиране. Допълнителна информация за тези API потърсете в *Ръководство за създаване на приложения*.

Приложните програмисти могат да използват и API, осигурени от други компании. Например Microsoft ODBC и ADO се използват от приложните програмисти за Windows при разработка на приложения за бази данни. DB2 Connect осигурява ODBC драйвер и OLE DB Provider, който поддържа приложенията, разработени с помощта на ODBC и ADO API. IBM не предоставя средства за разработка на ODBC приложения; тези инструменти се осигуряват от Microsoft Corporation.

## Използване на Центъра за управление за разрешаване на многосайтово обновяване

От версия 6 можете да използвате Центъра за управление, за да предоставите възможност за многосайтово обновяване. Процедурата е много проста и е очертана по-долу. За допълнителна информация относно процеса на конфигуриране на многосайтово обновяване, включително как ръчно да конфигурирате вашата система, се обърнете към електронната *Приложение за свързваемост*.

### Стартиране на Помощника за многосайтово обновяване

От Центъра за управление щракнете върху знака [+], за да разширите изгледа на дървото. С десния бутон на мишката изберете потребителския модел, който искате да конфигурирате. Отваря се изскачащо меню. Изберете елемента **Многосайтово обновяване** —> **Конфигуриране**.

### Стъпки с помощника

Помощникът предоставя интерфейс от тип тетрадки. Всяка страница от тетрадката ви подсказва определена информация за вашата конфигурация. По-долу страниците са показани в реда, в който ще ги срещнете.

Стъпка 1. Задайте Монитор за Процесора на транзакции (Transaction Processor – TP).

Това поле ще ви покаже подразбиращите се стойности за TP Монитора, който сте активирали. Ако не искате да използвате TP монитор, изберете **Не използвай TP Монитор**.

Стъпка 2. Задайте комуникационните протоколи, които ще използвате.

Стъпка 3. Задайте базата данни на Мениджъра на транзакции.

Този панел ви показва първата база данни, към която се свързвате по подразбиране (1ST\_CONN). Можете да оставите подразбиращата се база данни или да изберете друга каталогизирана база данни.

Стъпка 4. Задайте типовете сървъри на бази данни, включени в актуализацията, а също и дали ще се използва или не само TCP/IP.

Стъпка 5. Задайте настройките на Мениджъра за синхронизация.

Тази страница ще се появи само ако настройките на предишната страница показват, че в сценария за многосайтово обновяване ви е необходимо да използвате Мениджъра за синхронизация на DB2.



## Тестване на модула Многосайтово обновяване

- Стъпка 1. С десния бутон на мишката изберете потребителския модел и от контекстното меню изберете опцията **Многосайтово обновяване** → **Тест**. Отваря се прозореца Тест за многосайтово обновяване.
- Стъпка 2. От списъка **Достъпни** изберете базите данни, които искате да тествате. Може да използвате бутоните стрелки, за да преместите изборите си в и от подпрозореца **Избрани**. Освен това можете да промените изборения потребителски идентификатор и парола, като директно ги редактирате в списъка **Избрани**.
- Стъпка 3. Когато приключите с избирането, щракнете бутона **Тест** в долната част на прозореца. Отваря се прозорецът Резултати при тестване на многосайтово обновяване.
- Стъпка 4. Този прозорец показва коя от избраните от вас бази данни е преминала успешно или не теста за обновяване. В прозореца са показани SQL кодовете и съобщенията за грешки за базите данни, които не са преминали успешно теста.

---

## Електронни издания, свързани с DRDA

Следните електронни издания съдържат полезна информация, свързана с DRDA.

**За AS/400:**

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

**За OS/390:**

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

**За DataJoiner:**

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

**Електронни издания за бази данни/управление на данни:**

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



---

## Глава 3. Сценарии за използване на DB2 Connect

DB2 Connect може да осигури множество решения при необходимостта от достъп до хост или AS/400 база данни. В този раздел са очертани редица сценарии, които може да се отнасят до вашите конкретни нужди или вашето обкръжение.

*DB2 Connect Personal Edition* се използва при свързване на отделна 32-битова Windows система, Linux или OS/2 работна станция към S/390 или AS/400 база данни. DB2 Connect Personal Edition е най-подходящ за обкръжения, в които се осигурява собствена TCP/IP поддръжка от сървърите на базата данни и използваните приложения са традиционни двустранни клиент-сървър приложения.

Например DB2 Connect Personal Edition е подходящ избор за активиране на традиционни двустранни приложения на VisualBasic и Microsoft Access. Приложенията, за които е необходим междинен сървър на приложения, трябва да използват DB2 Connect Enterprise Edition. За информация за развой с помощта на DB2 Connect Personal Edition вижте “Директен достъп до база данни” на страница 18.

*DB2 Connect Enterprise Edition* често се инсталира на междинен сървър, който да свързва DB2 клиенти към хост или AS/400 база данни. Освен това може да се използва на компютри, където няколко локални потребители искат да имат директен достъп до хост или AS/400 сървъри.

Например DB2 Connect Enterprise Edition може да се инсталира на голяма машина с много локални потребители. Също така може да се инсталира на web сървър, монитор за обработка на транзакции (TP) или други 3-странни сървъри на приложения с множество локални SQL приложни процеси и нишки. В тези случаи можете за по-просто да инсталирате DB2 Connect Enterprise Edition на същата машина или на отделна машина, за да облекчите натоварването на процесора.

DB2 Connect Enterprise Edition е най-подходящ за обкръжения, където:

- Хост и AS/400 сървърите на база данни нямат собствена поддръжка на TCP/IP връзки, а не е желателно директно свързване от настолните работни станции чрез SNA. Вижте “DB2 Connect Enterprise Edition като сървър за свързване” на страница 20.
- На Web сървъри работят web-базирани приложения. Вижте “DB2 Connect и Web приложения” на страница 21.
- Web сървърите изпълняват web-базирано приложение с помощта на Java аплети за данни.
- Използва се междинен сървър на приложения. Вижте “Използване на DB2 Connect със сървъри на приложения” на страница 25.
- Използват се TP монитори като CICS, Encina, Microsoft Transaction Server (MTS), Tuxedo, Component Broker и MQSeries. Вижте “Използване на DB2 Connect с монитори на обработка на транзакции” на страница 29.

*DB2 Connect Unlimited Edition* е уникален пакет, който предлага комплексна гъвкавост на DB2 Connect развой и опростява избирането на продукта и лицензирането. Този продукт съдържа както DB2 Connect Personal Edition, така и DB2 Connect Enterprise Edition с лицензионни условия, които позволяват неограничен развой на DB2 Connect продукти. Лицензионните такси се определят от размера на System/390, с който потребителите на DB2 Connect ще работят. Този нов пакет се предлага само за OS/390 системи и лицензирането е валидно само за DB2 за OS/390 източници на данни.

---

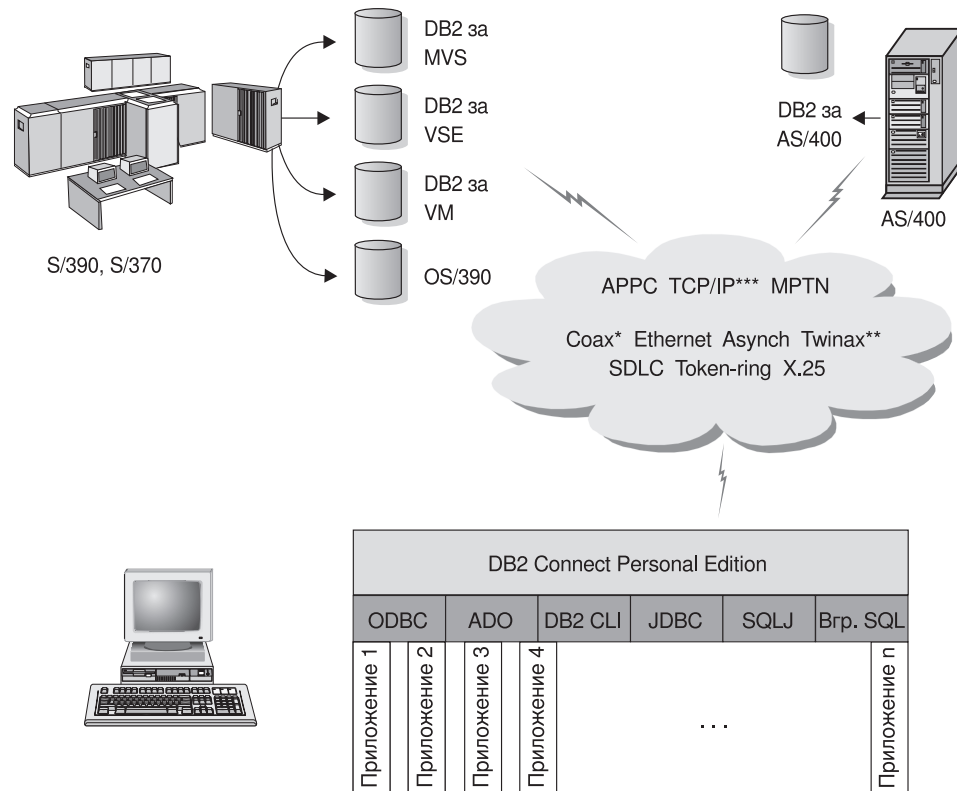
## Директен достъп до база данни

Основната функция на DB2 Connect е да осигури директно свързване към хост база данни от настолни приложения, които работят на 32-битови Windows системи, Linux или OS/2 работни станции. DB2 Connect Personal Edition е най-лесният начин за реализиране на това решение.

Всяка работна станция, на която е инсталирано DB2 Connect Personal Edition, може да установи директна TCP/IP връзка към DB2 за OS/390, DB2/400 и DB2 UDB за Windows NT, Windows 2000, UNIX и OS/2 сървъри. Освен това приложенията могат да се свързват и да обновяват няколко бази данни от DB2 фамилията в една и съща транзакция, като се осигурява цялостен интегритет на данните с помощта на двуфазовият протокол за записване на промените.

Освен това DB2 Connect Personal Edition има интегрирана поддръжка на APPC за комуникации с DB2 за MVS и други хост и AS/400 бази данни, които изискват APPC. Независимо от това се препоръчва използването на TCP/IP вместо SNA, когато има собствена TCP/IP поддръжка.

Фигура 3 на страница 19 показва работните станции, директно свързани към хост или AS/400 сървър на базата данни. На всяка работна станция е инсталиран DB2 Connect Personal Edition.



Не всички протоколи се поддържат за всички платформи

\* Само за хост свързвания

\*\* За AS/400

\*\*\* TCP/IP свързваемостта изисква DB2 за OS/390 V5R1, DB2 за AS/400 V4R2 или DB2 за VM V6.1

Фигура 3. Директно свързване между DB2 Connect и хост или AS/400 сървър на базата данни

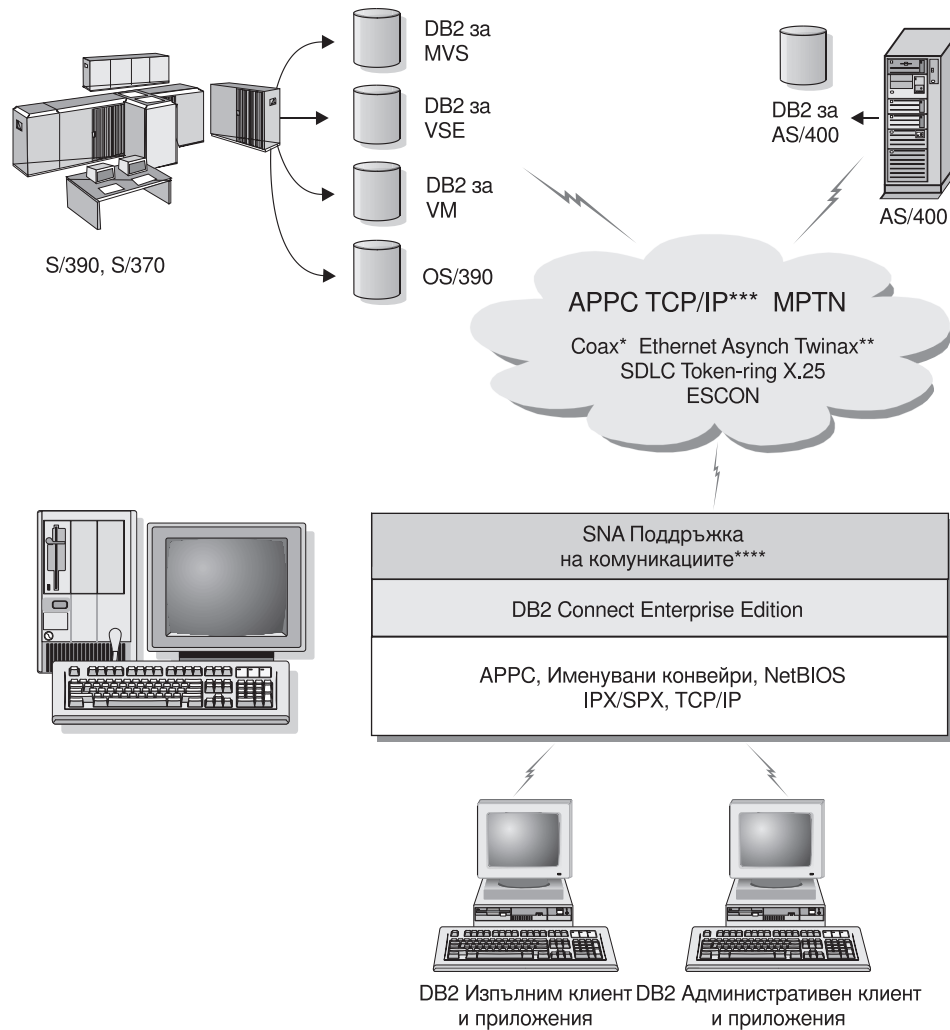
#### Забележки:

1. Информация за това кои протоколи се поддържат от кои DRDA AR и техните съответни хост и AS/400 DRDA, потърсете в съответните книги *DB2 Connect: Бърз старт*.
2. Не е необходимо DB2 Universal Database да е инсталиран на DB2 Connect работната станция. Ако искате да имате пълна система за управление на релационна база данни на DB2 Connect работна станция, поръчайте DB2 Universal Database.
3. Клиент за разработка на DB2 приложения сега е част от пакета DB2 Connect и може да се инсталира, ако потребителят иска да го използва за разработка на приложения. Допълнително DB2 Connect включва Stored Procedure Builder, който може да се използва за изграждане, тестване и развой на запомнени процедури за DB2 за OS/390.
4. Програмистите на C, които разработват Windows приложения и използват Microsoft ODBC, OLE DB или ActiveX Data Objects (ADO), трябва да използват *Microsoft Open Database Connectivity Software Development Kit*. Програмистите, които искат да разработват приложения с помощта на езика за програмиране Java, могат да използват всяко от обкръженията за разработка на Java, като VisualAge за Java на IBM.

## DB2 Connect Enterprise Edition като сървър за свързване

DB2 Connect сървър позволява на няколко клиенти да се свържат към хост или AS/400 данни и може значително да намали усилията, които са необходими за установяване и поддръжка на достъпа до данните. Фигура 4 илюстрира решение на IBM за обкръжения, в които искате DB2 клиент да осъществи непряко свързване към хост или AS/400 сървър на базата данни чрез DB2 Connect Enterprise Edition.

В следващия пример можете да замените DB2 Connect сървъра с DB2 UDB Enterprise Edition или Enterprise – Extended Edition сървър, на който е инсталиран компонента за поддръжка на сървър DB2 Connect.



Не всички протоколи се поддържат за всички платформи.

- \* Само за хост свързвания
- \*\* За AS/400
- \*\*\* TCP/IP свързваемостта изисква DB2 за OS/390 V5R1, DB2 за AS/400 V4R2 или DB2 за VM V6.1
- \*\*\*\* Поддръжката на SNA комуникациите е различна за всяка операционна система и се изисква само в случаите, когато няма вградена TCP/IP свързваемост.

Фигура 4. DB2 Connect Enterprise Edition

---

## DB2 Connect и Web приложения

Web браузърите бързо стават стандартен интерфейс за всичко – от електронните каталози до интранет приложенията. За прости web приложения може да е достатъчно наличието само на web сървър. За приложения с голям обем, за които може да е необходим достъп до база данни и обработка на транзакции, IBM предлага решения, които използват DB2 Connect за управление на много голям брой едновременни транзакции през web.

В този раздел са описани web-базираните бизнес решения, които може да спечелят от използването на DB2 Connect.

### Предимства и ограничения на традиционното CGI програмиране

Приложенията за e-бизнес в World Wide Web обикновено използват интерфейса Common Gateway Interface (CGI), за да могат потребителите да отправят запитвания към бази данни. Освен това много компании използват вътрешни web приложения и те най-често са изградени върху база данни, която работи във фонов режим.

Потребителите попълват форми в web страница и ги предават чрез CGI към приложения или скриптове на web сървъра. От своя страна скриптът използва осигурения API за база данни, за да предаде SQL запитвания към хост база данни. Същият скрипт може след това да конструира web (HTML) страница с резултатите от запитването и да я изпрати обратно за представяне от web браузъра на потребителя. Например електронен каталог, в който потребителят може да отправи запитване за наличностите и текущата цена на определени стоки или услуги.

CGI приложенията може да са прости за проектиране и лесни за поддръжка. Тъй като CGI стандартът не зависи от операционната система и от езика, се предлага почти на всички компютърни платформи. CGI програми могат да се напишат на C++ или в език със скриптове като Perl.

Въпреки че CGI може да изглежда като идеалното решение за web-базирани приложения, има сериозни недостатъци. Средата за програмиране за CGI не е толкова комплексна, колкото другите API. Освен това има проблем със скалируемостта, който ще се отрази върху работата на електронна търговия с голям обем. При всяко обръщение към CGI приложение на web сървъра се създава нов процес. Всеки потребителски модел трябва да направи свое собствено свързване към базата данни и да изпрати свое собствено запитване. В обкръжения с голям обем на транзакциите това ограничение може да създаде значителни проблеми с производителността.

Можете да използвате DB2 Connect с web сървър, за да създадете стабилни приложения за електронна търговия с голям обем. DB2 Connect осигурява редица решения, които подобряват производителността на web-базираните приложения. Запомнените процедури (вижте “DB2 Connect на Web сървър” на страница 22) позволяват на потребителите на DB2 Connect да намалят броя на запитванията, които се изпращат към базата данни.

Пулът на свързването (вижте “Създаване на пул за връзките” на страница 22) намалява броя на свързванията и прекъсванията на връзката към и от база данни. При голямо натоварване, когато ограниченията на CGI стават съществени, вижте IBM Net.Data (вижте “Net.Data” на страница 24) и WebSphere (вижте “IBM WebSphere” на страница 24), които осигуряват свързване без CGI за приложения в големи предприятия.

## DB2 Connect на Web сървър

IBM осигурява HTTP (Web) сървъри с всички DB2 Connect продукти за OS/2, UNIX, Windows NT и Windows 2000. DB2 Connect Enterprise Edition осигурява незабавна поддръжка, директно след изваждане от кутията, на web сървъри Apache или Lotus Domino и освен това може да работи с други web сървъри като Microsoft Internet Information Server или Netscape Enterprise Server.

Ако работите с база данни от DB2 фамилията върху OS/390, AS/400, VM и VSE системи, на Web сървъра трябва да имате DB2 Connect Enterprise Edition. DB2 Connect Enterprise Edition ще осигури библиотеките и комуникационните интерфейси, които да позволят на Web сървърите да имат достъп до тези хост и AS/400 платформи. Може да се използва TCP/IP или SNA за комуникация между Web сървъра и база данни, която работи върху OS/390, AS/400, VM или VSE.

**Забележка:** Web решенията на IBM осигуряват възможността да се работи с няколко бази данни в рамките на един и същи CGI скрипт или в рамките на една и съща транзакция в CGI скрипт.

В следващите два раздела са разгледани усъвършенстванията на производителността, достъпни за CGI програмите, които имат достъп до DB2 бази данни. По-нататък са разгледани алтернативите на стандартните CGI като Java.

### Запомнени процедури

Важно съображение за web приложенията, както при работа клиент/сървър, е да се намали до минимум трафика между HTTP сървъра и крайната база данни. Това съображение е особено важно при обработка на голям обем транзакции, което е основата за повечето приложения, свързани с електронния бизнес.

Препоръчваният подход е да се комбинира CGI приложно програмиране с програмирането и бизнес логиката, която се съдържа в запомнените процедури. DB2 Universal Database на OS/2, UNIX и Windows, както и DB2 на OS/390, AS/400 и VSE имат едни и същи правила за дефиниране на параметрите при извикване на запомнени процедури.

Както е обикновено при CGI, web браузърът предава форма към web сървъра, където се изпълнява CGI скриптът. Но вместо всеки отделен SQL израз да се изпраща към DB2 базата данни, се изпраща заявка за изпълнение на запомнена процедура. Тази запомнена процедура съдържа няколко SQL изрази, които в противен случай щяха да се изпълняват отделно. Запомнените процедури намаляват броя на съобщенията, които се разменят между CGI скрипт и крайната база данни.

Основното им предимство е, че намаляват мрежовия трафик между HTTP сървъра и DB2 базата данни. За допълнителна информация за запомнените процедури се обърнете към *Ръководство за разработка на приложения* или електронната помощ за DB2 Stored Procedure Builder.

### Създаване на пул за връзките

Установяването на връзка от DB2 Connect сървър към хоста изисква компютърни ресурси и време. В среда, в която хиляди клиенти често се свързват към хоста чрез DB2 Connect сървъра, се отделя значителна част от времето за установяване и прекъсване на свързването.

Създаването на пул за връзките на DB2 Connect осигурява значително увеличение на производителността при такива среди. DB2 Connect поддържа отворени връзките към базата данни в достъпния пул. Когато клиент заяви свързване, то може да се



осигури от този пул с готови връзки. Създаването на пул за връзки значително намалява натоварването, което обикновено е необходимо при генериране и приключване на такива свързвания.

За допълнителна информация как се създава пул за връзки, вижте “Пул за връзки” на страница 118.

## DB2 Connect като Java сървър на приложения

Много от недостатъците на CGI може да се преодолеят, ако вместо тях се използва Java. IBM осигурява както аплети, така и приложения, които ви позволяват да използвате Java вместо CGI на всеки етап от web транзакциите. Решенията, осигурени от IBM, позволяват смесени техники, което означава, че можете да използвате процедурни решения като Net.Data и Microsoft Active Server Pages с DB2 или да се насочите към по-стабилните реализации, осигурени от Java сървър на приложения като IBM WebSphere.

Има два интерфейса за приложно програмиране (API– Application Programming Interfaces) за Java програмисти. Първият е JDBC и се поддържа за използване на Java при разработка на Java аплети за данни, Java приложения, както и Java servlet, Java страници на сървъри (JSP) и Enterprise Java Beans (EJB). JDBC е интерфейс за приложно програмиране на ниво обръщение или извикване на метод. Другият Java API е SQLJ. SQLJ дава възможност да се определи SQL в рамките на Java програма. При web транзакциите DB2 може да използва и двата вида API, както от страната на клиента, така и на сървъра.

За клиентските програми се поддържат аплети, аплети за данни и приложения. За страната на базата данни са предназначени обекти в базата данни, като дефинирани от потребителя функции и запомнени процедури.

При DB2 за OS/390, DB2 за VSE и VM и DB2 за OS/400 има два различни начина за развой на Java приложение. Можете да използвате директните възможности за свързване, осигурени от DB2 Connect Personal Edition с TCP/IP или SNA, или да изберете да преминете през DB2 Connect Enterprise Edition сървър, който осигурява възможност за свързване към мейнфрейм или AS/400 база данни.

И в двата случая за потребителя в Web не е необходим никакъв специален софтуер при достъп до базата данни, а само стандартен web браузър. Единствено трябва да е инсталиран DB2 Connect сървър и никакъв стандартен Web сървър. Ако web сървърът и DB2 Connect не са физически на един и същи компютър, на web сървъра трябва да е инсталиран DB2 клиент.

При DB2 за OS/390 основният компонент е DB2 Connect Enterprise Edition, който работи на междинния сървър. Това е компонентът, който ще добави функции на JDBC сървър към възможностите за свързване към DB2 за OS/390, VSE и VM или AS/400 сървър. Отново не е необходим никакъв специален софтуер за web брауъра на клиента.

IBM осигурява разширено множество от средства за разработка на Java приложения и аплети. За свързваемост към база данни пакетът за разработчици за DB2 осигурява пълен набор от средства, сред които VisualAge for Java Professional Edition, WebSphere Application Server, Net.Data, както и DB2 Universal Database и DB2 Connect за тестване. Освен това IBM VisualAge for Java Enterprise Edition съдържа средства за разработка на приложения за големи предприятия. Инструменти на други фирми като Borland JBuilder или Symantec Visual Cafe също работят с решенията за база данни на IBM.

## Net.Data

Частта Net.Data на фамилията DB2 Universal Database и DB2 Connect е множество от средства за приложно програмиране, проектирани да ви помогнат при създаване и поддържане на приложения за web-базирани транзакции. Можете да използвате Net.Data при осъществяване на достъп и промяна на данни, съхранени на DB2 UDB за OS/2, Windows NT, Windows 2000, UNIX, OS/390, VM, VSE и OS/400. Приложенията, създадени с помощта на Net.Data, се съхраняват на web сървър и може да се активират чрез web браузър.

Net.Data използва макроси или образци, с които дори потребители само с повърхностно разбиране на HTML и SQL могат да създават сложни web приложения. Макросът е текстов файл, който може да е изграден от Java, Java Script, HTML и вградени функции. Тези макроси могат да се използват за генериране на динамични web страници с предварително дефиниран формат, променливи и функции.

Прост Net.Data макрос има седем отделни раздела:

- Общи раздели, които основно служат като средства за документиране за програмиста.
- Раздел за дефиниране, който осигурява място за дефиниране на променливите.
- Раздел за функция, който съдържа основния програмен код.
- Раздел за отчет, който определя форматирането за резултата от Net.Data макроса.
- HTML раздел, който съдържа по-голямата част от HTML кода, използван в web страницата.
- Раздел Include, който е просто удобен начин да се включат общи части от макроса, които може да се използват повторно от други макроси.
- Раздел за съобщения, където се осигурява обработката на грешките.

Съществена характеристика на Net.Data, особено за DB2, е, че не е необходим специален софтуер за клиента. В тази реализация клиентът просто е web браузър.

Net.Data процесорът се инсталира заедно с DB2 Universal Database на Windows NT, Windows 2000, OS/2 или UNIX работна станция, заедно с Web сървър. При свързване към DB2 OS/390, DB2 за VSE и VM и 400 цялата инфраструктура на Net.Data се разполага на DB2 Connect сървър, заедно с Web сървър.

## IBM WebSphere

IBM WebSphere осигурява по-завършено решение за електронен бизнес, отколкото е възможно с традиционното CGI програмиране. WebSphere сървърите на приложения не само изпълняват процедурните възможности на CGI, но освен това ви позволяват да осигурите сложни и мощни услуги чрез web с помощта на аплети за сървъри, активни сървърни страници и JavaBeans. С WebSphere можете:

- Да се възползвате от стандартите, за да ускорите разработката и да увеличите максимално междуплатформената поддръжка.
- Да включите технологии от инструменти на други фирми и структури за приложения;
- Да анализирате производителността и използването на съдържанието на Web сайт;
- Да разширите лесно сайта си, за да събира повече потребители и да запази производителността;

- Да разработвате за редица основни операционни системи (IBM AIX, HP-UX, Linux, Novell NetWare, IBM OS/2, IBM OS/390, IBM OS/400, Sun Solaris, Microsoft Windows NT и Windows 2000);
- Да използвате съществуващия web сървър, включително от Apache, IBM, Netscape, и Microsoft.

WebSphere не е един продукт, а фамилия от три продукта, предназначени за три различни пазара. Сърцевината на WebSphere решението е WebSphere сървърът на приложения.

WebSphere сървърът на приложения осигурява обкръжение за три типа обекти. Един е Java server pages, който е аналог на Active Server Pages. Вторият компонент се състои от Java servlet, а третият е промишлен JavaBeans. Промислен JavaBeans е развиващ се стандарт за развой на много големи, стабилни приложения от промишлен клас.

Освен това Data Access JavaBeans осигурява комплексни функции за база данни, специално свързани за DB2. До DB2 може директно да се осъществи достъп чрез JDBC и SQLJ. Освен това се поддържат както COM+, така и CORBA.

WebSphere приложенията могат да се разполагат на същата платформа, както web сървъра и DB2 Universal Database. В случая с DB2 за OS/390, VM, VSE и AS/400, WebSphere се разполага на същата платформа като DB2 Connect Enterprise Edition.

Има редица WebSphere решения, като Web Studio и WebSphere Performance Packs. Трите версии на WebSphere са:

#### **Standard Edition**

За производителите на Web сайтове този сървър позволява използването на Java servlet и JSP технология, за да се трансформират бързо и лесно Web сайтове и входове от статични страници в жизнени източници на персонализирано динамично Web съдържание. Освен това включва водещата в индустрията XML поддръжка за улесняване на обмена на информация и данни в рамките на групи или между предприятия и вградена технология за анализ на сайтове, която осигурява информация за производителност и използване, за да ви помогне да се увеличи максимално възвръщаемостта от инвестициите на компанията в Web сайта.

#### **Advanced Edition**

За приложните програмисти този мощен EJB сървър позволява развой на бизнес логика с помощта на EJB компоненти. Осигурява скалируемост на защитата при свързването и поддръжка на Java, като включва пълната функционалност на Standard Edition.

#### **Enterprise Edition**

За промишлените архитекти този сървър интегрира коренно различни бизнес системи в рамките на организацията, за да изгради стабилни приложения за електронен бизнес и да увеличи максимално общото използване на ресурсите. Enterprise Edition включва възможностите на получилите награда продукти IBM TXSeries и Component Broker. Освен това включва пълната функционалност на пакетите Advanced Edition и Standard Edition.

---

## **Използване на DB2 Connect със сървъри на приложения**

Наличието на клиент-сървър приложения позволява на приложните проектанти да повишат възможностите за използване и да намалят разходите за обучение, като осигурят приложения с графичен потребителски интерфейс на платформи като Windows и OS/2. В същото време позволява гъвкавостта на делегиране на функция

за управление на база данни на устойчиви сървъри на база данни върху множество операционни системи и хардуерни платформи.

Моделът клиент–сървър, при който логиката на приложението се разпределя между клиентските работни станции, често се разглежда като *2–страниен клиент–сървър*. В двустранния модел приложението се разполага на клиента, а сървърът на базата данни реализира сървъра или крайното стъпало. Както се вижда в “Директен достъп до база данни” на страница 18, DB2 Connect осигурява пълна поддръжка за двустранни клиент–сървър приложения, при които сървърите на база данни са DB2 за OS/390, DB2 за MVS/ESA, DB2/400 или DB2 за VM и VSE.

С увеличаването на размера на клиент–сървър приложенията става ясно, че двустранният модел клиент–сървър има съществени ограничения. Разпределянето на големи количества бизнес логика сред стотици или дори хиляди клиентски работни станции прави промяната на управлението сложно и скъпо начинание. Всяка промяна в бизнес правилата изисква замяна на клиентската част на приложението. Често тази промяна на приложенията трябва да се изпълни едновременно на всички клиентски работни станции в предприятието, за да се осигури, че бизнес правилата ще се прилагат последователно.

Друг недостатък на двустранния модел клиент–сървър, който става очевиден при увеличаването на размера е количеството ресурси, които се консумират от такива приложения. Развой на стотици или хиляди *"дебели"* *клиенти*, както двустранните клиенти често се наричат, увеличени изисквания за възможностите и капацитета за обработка на всяка клиентска работна станция. Още повече, че изискванията за сървъра на базата данни също се увеличават значително, тъй като за всеки клиент е необходима отделна връзка към базата данни и ресурси, които да обслужват такова свързване. Докато двустранната клиент–сървър зависимост от разпределената бизнес логика може някак си да се намали от разширено използване на запомнени процедури, другите недостатъци не могат да се решат лесно без промени в модела.

## Решение Сървър на приложения

С увеличаването на разходите и сложността на двустранните клиент–сървър приложения, повечето от най–големите приложения се насочват към многостранния модел клиент–сървър. При многостранния модел ролята на базата данни остава непроменена. Обаче клиентската страна се допълва от една или повече междинни; обикновено една и от там името *3–страниен*.

В 3–страниния модел клиентът се свежда до обработка на взаимодействието с потребителя и не съдържа никаква бизнес логика. Междинната страна се състои от един или повече сървъри на приложения. Целта на сървъра на приложения е да се осигури стабилна, ефективна по отношение на разходите реализация на логиката, която стои в основата на бизнес процесите и бизнес правилата. Както при двустранния модел реализирането на бизнес правила често се допълва от използването на запомнени процедури, за да се увеличи производителността.

Тъй като клиентските работни станции повече не поемат товара да реализират логиката на приложението и служат само за взаимодействие с потребителя, изискванията по отношение на ресурсите за клиентската страна са силно намалени. На практика клиентската страна в 3–страниния модел често се нарича *тънък клиент*. Освен това тъй като централизиран сървър на приложения обработва заявките от всички клиенти, има възможност да използва съвместно ресурси, като свързвания към база данни, общи за всички клиенти. В резултат сървърът на базата данни повече не е необходимо да поддържа отделни свързвания за всеки потребител.

Много примери на 3–страни сървъри на приложения могат да се посочат в практиката днес. Почти всички доставчици на Enterprise Resource Planning (ERP) реализират своите приложения с помощта на 3–страниния модел, като приложенията SAP R/3 и PeopleSoft V7. Други примери включват водещите доставчици на Enterprise Relationship Management, като Siebel и Vantive.

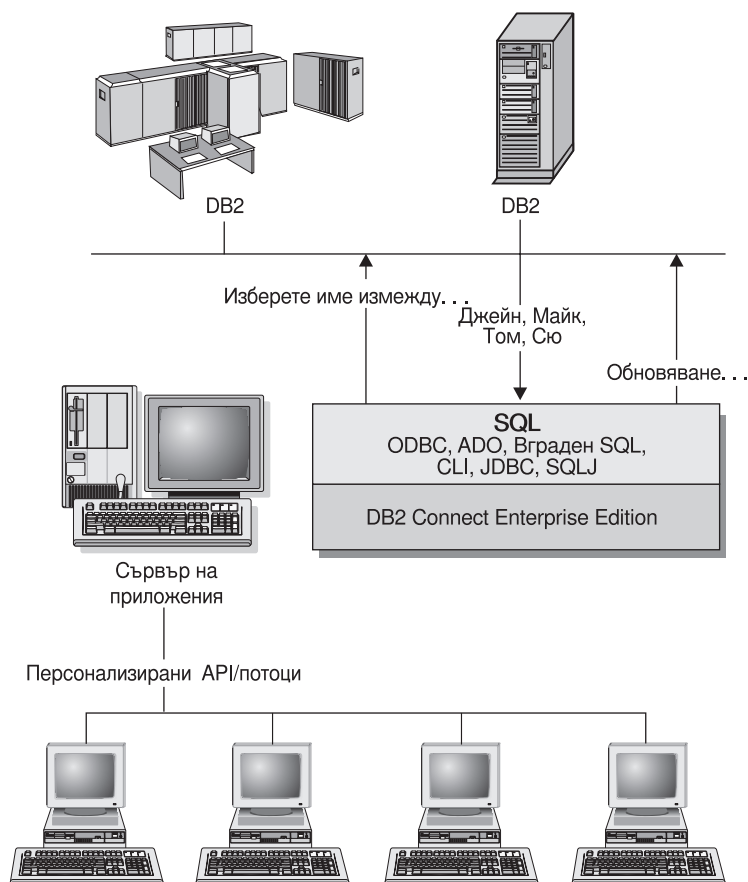
## Сървъри на приложения и DB2 Connect

Сървърите DB2 Connect Enterprise Edition осигуряват комплексна поддръжка за развой на многострани приложения. Поддръжката, осигурена от DB2 Connect, включва множество API, които могат да се използват за разработка на логиката на приложението (ODBC, ADO, DB2 CLI, Вграден SQL, JDBC и SQLJ), както и пълна комуникационна инфраструктура за взаимодействие с фамилията DB2 сървъри на база данни.

DB2 Connect поддържа и реализации, в които базата данни се състои от няколко сървъри на база данни от фамилията DB2. Това позволява на сървърите на приложения да реализират транзакции, които в една транзакция обновяват данните, разположение на няколко сървъра на база данни.

Целостта на такива разпределени транзакции се осигурява от поддръжката на протокола за 2–фазово записване на промените, осигурено от DB2 Connect. Например приложение може да обнови данни в DB2 за OS/390 база данни и DB2 UDB на Windows NT в една и съща транзакция. Ако е инсталирана и разрешена поддръжка за разпределена заявка, приложението може в една и съща транзакция да чете Oracle база данни и да обнови база данни от DB2 фамилията.

В следващата диаграма са представени API както и механизъмът за свързване между сървъра на приложения и крайните сървъри на база данни, осигурен от DB2 Connect Enterprise Edition.



Усъвършенствани функции на DB2 Connect, като създаване на пул за връзки (вижте “Пул за връзки” на страница 118) и концентратор на връзки (вижте “DB2 Connect концентратор за връзки” на страница 119), намалява силно изискванията за ресурси на приложението и опростява сървърната реализация на приложението.

## DB2 Connect и конфигуриране на сървър на приложения

DB2 Connect Enterprise Edition (доставя се самостоятелно или като част от пакета DB2 Connect Unlimited Edition) е необходим за използване със сървъри на приложения. DB2 Connect Personal Edition не се поддържа и не е лицензиран за използване със сървъри на приложения. Освен това при реализиране на сървъри на приложения трябва да разгледате условията и обстоятелствата, осигурени с копието на DB2 Connect, за да се разбере колко потребителски лиценза ще трябва да се придобият.

Има два метода за развой за DB2 Connect в обкръжение на сървър на приложения. DB2 Connect Enterprise Edition инсталиран на:

- Сървър на приложения; или
- Отделен комуникационен сървър.

В повечето случаи инсталирането на копие на DB2 Connect на същия сървър като самия сървър на приложения е предпочитаното решение. Като се инсталира DB2 Connect на сървъра на приложения, той може да участва във всяка схема за поемане при отказ и балансиране на натоварване, която сървърът на приложения може да реализира. Тази настройка потенциално може да осигури по-добра производителност, тъй като отстранява допълнителната мрежова отсечка, необходима в случаите, когато DB2 Connect е инсталиран върху отделен сървър.

Още повече, че администрирането може да са опрости, тъй като не е необходимо да се инсталира и поддържа допълнителен сървър.

Да се инсталира DB2 Connect на отделен сървър е добра възможност в ситуации, при които не се предлага версия на DB2 Connect Enterprise Edition за операционната система или хардуерната платформа, на която работи сървъра на приложения. Например, ако сървърът на приложения е разположен върху Silicone Graphics (SGI) или SCO UnixWare сървър, разполагането на DB2 Connect на отделен сървър е единствената възможност, тъй като DB2 Connect Enterprise Edition не е достъпен за тези платформи.

---

## Използване на DB2 Connect с монитори на обработка на транзакции

В предишния раздел научихте за използването на DB2 Connect със сървър на приложения. Сървърът на приложения позволява голям брой потребители да изпълняват приложения, като използват минимално количество системни ресурси.

Сървърът на приложения може да се разшири и да позволи координирани транзакции да се извикват от приложения, изпълнявани от сървъра на приложения. Това координиране на транзакциите обикновено се разглежда като монитор за обработка на транзакции (TP – Transaction Processing). TP мониторът работи заедно със сървъра на приложения.

*Транзакцията* може да се разглежда като обичайно събитие, обикновено заявка за услуга, в изпълнението на ежедневните операции на организация. Обработката подред на транзакциите е типът работа, за която са проектирани TP мониторите.

Всяка организация има правила и процедури, които описват как се предполага, че работи. Потребителските приложения, които реализират тези правила, може да се нарекат *бизнес логика*. Транзакциите, които изпълняват тези бизнес приложения, често се разглеждат като Обработка на транзакции или електронна обработка на транзакции (OLTP – Online Transaction Processing).

Съществените характеристики на предлаганите OLTP са:

### **Много потребители**

Нормално е обработката на транзакции да се използва от мнозинството от хората в организацията, тъй като толкова много хора влияят върху текущото състояние на бизнеса.

### **Повторяемост**

Има тенденция в повечето случаи при работа с компютър един и същи процес да се изпълнява отново и отново. Например въвеждането на поръчка или обработката на плащане се използват много пъти всеки ден.

### **Кратки взаимодействия**

Къси по продължителност са повечето взаимодействия, които хората в организацията имат със системата за обработка на транзакции.

### **Общи данни**

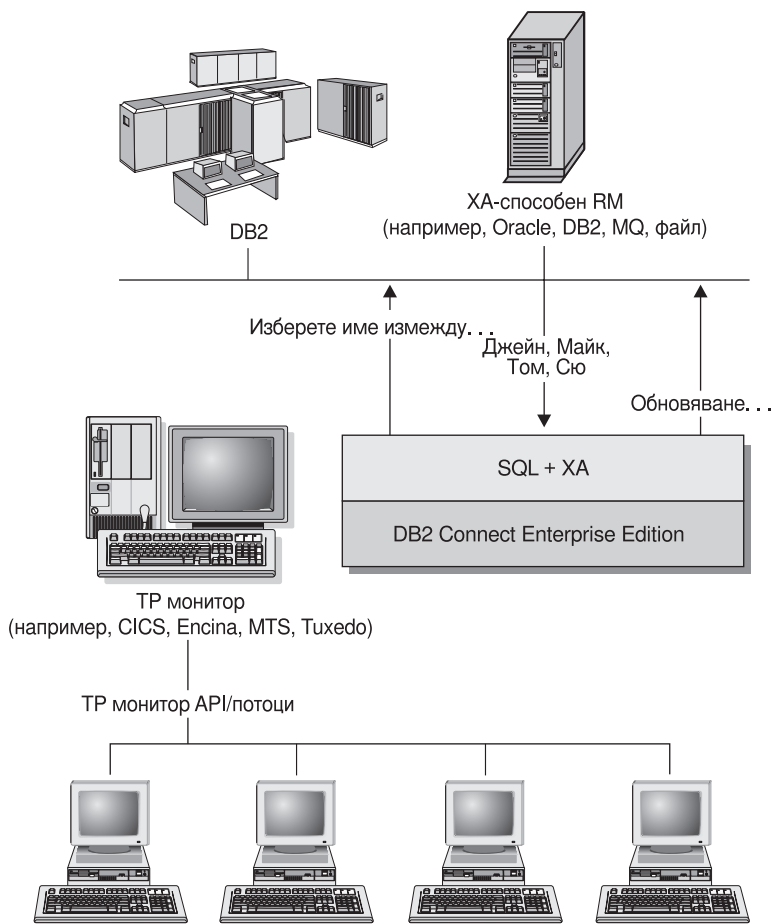
Тъй като данните отразяват състоянието на организацията, може да има само едно копие на данните.

### **Интегритет на данни**

Данните трябва да отразяват текущото състояние на организацията и трябва да са вътрешно последователни. Например всяка поръчка трябва да се асоциира със запис за клиент.

## Ниски разходи/Транзакция

Тъй като обработката на транзакциите представлява директен бизнес разход, разходите за системата трябва да са минимални. DB2 Connect позволява на приложенията под контрола на сървър на приложения, който работи на UNIX, Windows NT, Windows 2000 или OS/2, да изпълнява транзакции спрямо отдалечени LAN, хост и AS/400 сървъри на база данни, като тези транзакции се координират от TP монитор.



В следващата фигура са представени API както и механизмът за свързване между сървъра на приложения и крайните сървъри на база данни, осигурени от DB2 Connect Enterprise Edition.

## Примери на TP монитори

Най-разпространените TP монитори на пазара днес са:

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

Microsoft Transaction Server Remote S/390, AS/400 и LAN сървърите на база данни могат да се използват в рамките на транзакции, координирани от тези TP монитори.



## Tuxedo и DB2 Connect

При DB2 Connect версия 6 и предишните версии Tuxedo базираните приложения се ограничаваха до достъп само за четене до хост и AS/400 сървъри на база данни. Това ограничение е премахнато с DB2 Connect версия 7. Tuxedo–базирани приложения сега могат да обновяват хост и AS/400 сървъри на база данни в рамките на координирана от Tuxedo транзакция. Прилагат се специални изисквания и ограничения при конфигуриране. За повече информация вижте “DB2 Connect концентратор за връзки” на страница 119.

## X/Open Distributed Transaction Processing (DTP) модел

Може да е необходимо приложение с бизнес логика, за да се обновят няколко ресурса в рамките на една транзакция. Например, банково приложение, което реализира трансфер на пари от една сметка до друга, може да изисква дебитиране на една база данни (сметката "от") и кредитиране на друга база данни (сметката "на").

Освен това е възможно различни доставчици да осигуряват двете бази данни. Например едната база данни е DB2 Universal Database за OS/390, а другата е Oracle. Вместо всеки TP монитор да реализира собствения интерфейс за транзакции на доставчика на всяка база данни, се дефинира общ интерфейс за транзакции между TP монитор и всеки ресурс, до който се осъществява достъп от приложение. Този интерфейс е известен като *XA интерфейс*. TP монитор, който използва XA интерфейс се разглежда като *XA съвместим мениджър на транзакции (TM – Transaction Manager)*. Ресурс, който може да се обнови и се реализира с XA интерфейса се разглежда като *XA съвместим мениджър за ресурси (RM – Resource Manager)*.

Посочените горе TP монитори са XA съвместими. При достъп до отдалечени хост, AS/400 и DB2 UDB LAN–базирани сървъри на база данни чрез DB2 Connect се използват XA съвместими RM. Следователно всеки TP монитор, който има XA съвместим TM може да използва хост, AS/400 и LAN базирани DB2 UDB сървъри на база данни в рамките на бизнес приложения, изпълняващи транзакции.

## Как да се използва DB2 Connect с XA съвместим Мениджър на транзакции

В този раздел са описани стъпките за конфигуриране при използване на S/390 и AS/400 сървъри на база данни в рамките на вашия TP монитор. Прието е, че имате работещ TP монитор и инсталиран DB2 Connect, както и конфигурирана и тествана връзка към хоста или AS/400 сървъра на база данни. За по–подробна информация се обърнете към книгата *DB2 Connect: Бърз старт*.

Необходимите стъпки при конфигуриране на най–популярните TP монитори са представени в *Ръководство за администриране*. Не се различава конфигурирането за достъп до LAN–базиран DB2 UDB сървър на база данни и хост или AS/400 сървър на база данни. Следващите инструкции очертават общите стъпки при конфигуриране на TP монитори, които не са посочени в *Ръководство за администриране*.

За да конфигурирате DB2 Connect така, че да използва S/390 и AS/400 сървъри на база данни в рамките на вашия TP монитор, изпълнете следните стъпки:

1. Конфигурирайте така TP монитор, че да има достъп до DB2 XA Switch. DB2 XA Switch предоставя на TP монитора адресите на XA API на DB2 Connect. Всеки TP монитор изпълнява това по различен начин. За информация как да се

осигури DB2 XA Switch за TP монитор, се обърнете към *Ръководство за администриране*.

2. Конфигурирайте TP монитора с низа XA\_OPEN на DB2. Всеки TP монитор има свой собствен начин да направи това. За информация за низа XA\_OPEN на DB2 Connect се обърнете към *Ръководство за администриране*. За информация как да конфигурирате низа XA\_OPEN на DB2, за да се използва от TP монитора, се обърнете към документацията на TP монитора.
3. Ако е необходимо, променете конфигурационните параметри по подразбиране на DB2 Connect Sync Point Manager (SPM). Хост и AS/400 сървърите на база данни все още не поддържат XA интерфейса.

SPM е компонент на DB2 Connect, който трансформира XA двуфазовия протокол за записване на промените в двуфазовия протокол, използван от хост и AS/400 сървъри на база данни. По подразбиране DB2 потребителския модел има предварително дефинирани стойности за конфигурационните параметри на SPM. Най-важният параметър е конфигурационният параметър на мениджъра на базата данни SPM\_NAME. По подразбиране е разновидност на първите седем символа на TCP/IP името на хоста.

Ако използвате TCP/IP за свързване към DB2 за OS/390, тогава не е необходимо да промените параметрите по подразбиране. В този случай не е необходима SPM конфигурация, тъй като вече работи. Ако използвате SNA за достъп до хост или AS/400 сървъри на база данни, тогава трябва да се уверите, че стойността SPM\_NAME представлява валидна SNA LU във вашата мрежа. Ако стойността SPM\_NAME по подразбиране не е приемлива, тогава трябва да използвате помощника за многосайтово обновяване, за да я промените.

---

## Глава 4. Програмиране в DB2 Connect обкръжение

Този раздел осигурява информация за създаването на приложения, които използват DB2 Connect. Повече информация потърсете в *CLI Ръководство и справочник*, *Справочник на командите* и *Ръководство за разработка на приложения*.

---

### Програмиране в разпределена среда

DB2 Connect позволява на приложни програми да имат достъп до данни в DB2 бази данни на сървъри System/390 и AS/400. Например приложение, което работи под Windows може да има достъп до данни в DB2 Universal Database за OS/390 база данни. Можете да създадете нови приложения или да промените съществуващите, за да работят в хост или AS/400 среда. Освен това можете да разработите приложения в една среда и да ги прехвърлите в друга.

DB2 Connect ви позволява да използвате следните API с продукти на хост база данни като DB2 Universal Database за OS/390, в случай, че елементът се поддържа от продукта на хост базата данни:

- Вграден SQL, както статичен, така и динамичен
- DB2 интерфейс за вградени SQL оператори
- Microsoft ODBC API
- JDBC.

Някои SQL изрази се различават между различните реляционни бази данни. Може да срещнете SQL изрази, които са:

- Еднакви за всички видове бази данни, които използвате, независимо от стандартите
- Документирани в *SQL Справочник* и следователно достъпни във всички реляционни бази данни на IBM
- Уникални за системата на база данни, до която осъществявате достъп.

SQL изразите в първите две категории имат големи възможности за прехвърляне, докато тези в третата категория ще трябва най-напред да се променят. Като цяло SQL изразите в Data Definition Language (DDL) не могат толкова лесно да се прехвърлят, колкото тези в Data Manipulation Language (DML).

DB2 Connect приема някои SQL изрази, които не се поддържат от DB2 Universal Database. DB2 Connect предава тези изрази към хоста или AS/400 сървъра. Информация за ограниченията на различните платформи, като максималната ширина на колона вижте в *SQL Справочник*.

Ако преместите CICS приложение от OS/390 или VSE, за да работи под друг CICS продукт (например CICS за AIX), то освен това може да осъществи достъп до OS/390 или VSE база данни с помощта на DB2 Connect. Повече информация потърсете в *Ръководство за програмиране на CICS/6000 приложения* и *Настройване и работа на CICS*.

Когато програмирате в хост или AS/400 среда, трябва да разгледате следните специфични фактори:

- Използване на Data Definition Language (DDL)
- Използване на Data Manipulation Language (DML)
- Използване на Data Control Language (DCL)

- Свързване и прекъсване на връзката
- Предкомпилиране
- Дефиниране на ред за сортиране
- Управление на референциален интегритет
- Заключване
- Разлики в SQLCODE и SQLSTATE
- Използване на системни каталози
- Нива на изолация
- Запомнени процедури
- НЕ АТОМАРЕН SQL блок
- Разпределена единица работа
- SQL изрази, поддържани или не от DB2 Connect.

## Използване на Data Definition Language (DDL)

DDL изразите се различават в рамките на различните бази данни на IBM, защото съхранението се обработва по различен начин от различните системи. При системите на хост или AS/400 сървър може да има няколко стъпки между проектирането на база данни и израза CREATE TABLE. Например серия от изрази може да конвертира един дизайн от логически обекти във физическото представяне на тези обекти на мястото за съхранение.

Предкомпиляторът предава много такива DDL оператори на хоста или AS/400 сървъра, когато предкомпилирате до хост или AS/400 сървър на база данни. Същите оператори няма да се предкомпилират спрямо база данни на системата, където се изпълнява приложението. Например в приложение за OS/2 операторът CREATE STORGROUP ще се предкомпилира успешно за DB2 Universal Database за OS/390 база данни, но не и при база данни DB2 за OS/2.

## Използване на Data Manipulation Language (DML)

Като цяло DML операторите са лесни за прехвърляне. Операторите SELECT, INSERT, UPDATE и DELETE са подобни в рамките на различните реляционни бази данни на IBM. Повечето приложения основно използват DML SQL оператори, които се поддържат от DB2 Connect програмата.

### Числени типове данни

Когато се прехвърлят числени данни в DB2 Universal Database, типът на данните може да се промени. Числените и зонирани десетични типове SQLTYPE (поддържани от DB2 Universal Database за AS/400) се конвертират до фиксирани (пакетни) десетични типове SQLTYPE.

### Смесени едно– и двубайтови данни

Смесените данни съдържат символи от разширената UNIX кодова таблица (EUC), двубайтовата кодова таблица (DBCS) и еднобайтовата кодова таблица (SBCS) в една и съща колона. На системи, които съхраняват данни в EBCDIC (OS/390, OS/400, VSE, and VM), символите shift-out и shift-in маркират началото и края на двубайтовите данни. На системите, които съхраняват данните в ASCII (като OS/2 и UNIX), не са необходими символите shift-in и shift-out.

Ако вашето приложение прехвърля едно– и двубайтови данни от ASCII система в EBCDIC система, внимавайте да оставите достатъчно място за символите shift.

При всяко преминаване от SBCS до DBCS данни добавете 2 байта към дължината на данните. За по-голяма съвместимост използвайте низове с променлива дължина в приложенията, които използват смесени данни.

## Дълги полета

Дългите полета (низове по-дълги от 254 символа) се обработват по различен начин от различните системи. Хост или AS/400 сървър може да поддържа само подмножество от скаларните функции за дълги полета, например DB2 Universal Database за OS/390 поддържа само функциите **LENGTH** и **SUBSTR** върху дълги полета. Освен това хост или AS/400 сървър може да изисква различна обработка за определени SQL оператори; например DB2 за VSE и VM изисква с оператора INSERT да се използва само хост променлива, SQLDA или NULL стойност.

## Тип данни големи обекти (LOB)

Типът данни LOB се поддържа от DB2 Connect.

## Дефинирани от потребителя типове (UDT)

Само отделни дефинирани от потребителя типове се поддържат от DB2 Connect. Абстрактните типове данни не се поддържат.

## Тип данни ROWID

Типът данни ROWID се обработва от DB2 Connect като VARCHAR за двоични данни.

## Тип данни 64-битов Integer (BIGINT)

Осем байтов (64-битов) integer се поддържа от DB2 Connect. Вътрешният тип данни BIGINT се използва за поддръжка на броя редове в таблици в много големи бази данни, като се запазва точността на данните.

## Използване на Data Control Language (DCL)

Всяка система за управление на релационна база данни на IBM осигурява различни степени на обособеност за SQL операторите GRANT и REVOKE. Проверете специфичните за продукта публикации, за да проверите подходящите SQL изрази, които да се използват за всяка система за управление на база данни.

## Свързване и прекъсване на връзката

DB2 Connect поддържа версиите на CONNECT TO и CONNECT RESET на оператора CONNECT, както и CONNECT без параметри. Ако приложение подаде SQL оператор, без най-напред да изпълни явния оператор за свързване CONNECT TO, се изпълнява *неявно* свързване към сървъра за приложения по подразбиране (ако е дефиниран).

Когато се свържете към база данни, в полето SQLERRP на SQLCA се връща информация за системата за управление на релационна база данни. Ако сървърът на приложенията е релационна база данни на IBM, първите три байта на SQLERRP съдържат едно от следните:

**DSN** DB2 Universal Database за OS/390

**ARI** DB2 за VSE и VM

**QSQ** DB2 Universal Database за AS/400

**SQL** DB2 Universal Database.

Ако използвате оператор **CONNECT TO** или **null CONNECT**, докато използвате DB2 Connect, токенът с кода на страната или територията в полето **SQLERRMC** на **SQLCA** се връща като празен; **CCSID** на сървъра на приложенията се връща в токена на кодовата страница или кодовата таблица.

Можете явно да прекъснете връзката, като използвате оператора **CONNECT RESET** (за свързване тип 1), операторите **RELEASE** и **COMMIT** [за свързване тип 2) или оператор **DISCONNECT** (и за двата типа свързване, но не в среда на TP следене).

Ако връзката не е прекъсната явно и приложението приключи нормално, DB2 Connect записва безусловно получените в резултат данни.

**Забележка:** Приложението може да получи съобщения **SQLCODE**, които да посочват грешки, но независимо от това да приключи нормално; DB2 Connect записва данните в този случай. Ако не искате данните да се запишат, трябва да използвате командата **ROLLBACK**.

Командата **FORCE** ви позволява да прекъснете връзката на избрани потребители или всички потребители към базата данни. Тази функция се поддържа за хост или AS/400 сървър на база данни; потребителят може да бъде принуден да се прекъсне връзката от DB2 Connect работната станция.

## Предкомпилиране

Има някои разлики в предкомпиляторите за различните системи на релационни бази данни на IBM. Предкомпиляторът за DB2 Universal Database се различава от предкомпиляторите за хост или AS/400 сървър в следното:

- Извършва само едно преминаване през приложението.
- При свързване с DB2 Universal Database бази данни обектите трябва да съществуват, за да се изпълни успешно свързване. **VALIDATE RUN** не се поддържа.

### Създаване на блокове

Програмата DB2 Connect поддържа възможностите за свързване на блокове на мениджъра на DB2 база данни:

#### **UNAMBIG**

Само дефинирани указатели са в блок (по подразбиране).

**ALL** Неопределените указатели не са в блок.

**NO** Указателите не са в блок.

Програмата DB2 Connect използва размера на блока, дефиниран в конфигурационния файл на мениджъра на DB2 базата данни за полето **RQRIOBLK**. Текущите версии на DB2 Connect поддържат размери на блокове до 32 767. Ако в конфигурационния файл на мениджъра на DB2 базата данни се определят по-големи стойности, DB2 Connect използва стойността 32 767, но не инициализира конфигурационния файл на мениджъра на DB2 базата данни. Създаването на блокове се извършва по един и същи начин с помощта на един и същи размер на блок за динамичен и статичен SQL.

**Забележка:** Повечето хост или AS/400 сървъри разглеждат динамичните указатели като неопределени, но DB2 Universal Database системите разглеждат

някои динамични указатели като дефинирани. За да се избегне объркване, можете да определите BLOCKING ALL с DB2 Connect.

В конфигурационния файл на мениджъра на DB2 базата данни определете размера на блока, като използвате CLP, Център за управление или API, както е посочено в *Административен API Справочник* и *Справочник на командите*.

## Атрибути на пакети

Пакет има следните атрибути:

### идентификатор на колекция

Идентификаторът на пакета. Може да се определи с командата PREP.

**Owner** Идентификатор за разпознаване на собственика на пакета. Може да се определи с команда PREP или BIND.

### Създател

Името на потребителя, който свързва пакета.

### Квалификатор

Неявен квалификатор за обекти в пакета. Може да се определи с команда PREP или BIND.

Всеки хост или AS/400 сървър има ограничения при използването на тези атрибути:

### DB2 Universal Database за OS/390

Всичките четири атрибути може да са различни. Използването на различен квалификатор изисква специални права на администратор. За повече информация за условията при използването на тези атрибути се обърнете към *Справочник на командите* за DB2 Universal Database за OS/390.

### DB2 за VSE и VM

Всички атрибути трябва да са идентични. Ако USER1 създаде свързан файл (с PREP), а USER2 изпълни действителното свързване, USER2 трябва да има DBA права, за да свърже вместо USER1. Само името на потребителя USER1 се използва за атрибутите.

### DB2 Universal Database за AS/400

Квалификаторът посочва името на колекцията. От взаимовръзката между квалификаторите и правата на собственика зависи предоставянето или отнемането на правата върху обекта. Името на потребителя, който е влязъл в системата е създател и собственик, освен ако не е квалифициран от идентификатор на колекция, като в този случай собственикът е идентификаторът на колекцията. Идентификаторът на колекцията трябва да съществува, преди да се използва от квалификатора.

### DB2 Universal Database

Всичките четири атрибути може да са различни. Използването на различен собственик изисква права на администратор и наличието на право на достъп CREATEIN до схемата (ако вече съществува).

**Забележка:** DB2 Connect осигурява поддръжка на командата *SET CURRENT PACKAGESET* за DB2 Universal Database за OS/390 и DB2 Universal Database.

## С низове без символ за край

Опцията за свързване CNULREQD замества работата с низове без символ за край, които са определени с помощта на опцията LANGLEVEL.

Информация за това как се обслужват низове без символ за край, когато са подготвени с опцията LANGLEVEL, установена на MIA или SAA1, ще намерите в *Ръководство за разработка на приложения*.

По подразбиране CNULREQD се установява на YES. Така низовете без символ за край ще се интерпретират според MIA стандартите. При свързване към а DB2 Universal Database за OS/390 сървър се препоръчва CNULREQD да се установи на YES. Трябва да свържете приложенията, кодирани според SAA1 стандартите (по отношение на низовете без символ за край), като установите опцията CNULREQD на NO. В противен случай низовете без символ за край ще се интерпретират според MIA стандартите, дори ако са подготвени с опция LANGLEVEL, установена на SAA1.

## Самостоятелен SQLCODE и SQLSTATE

Самостоятелните променливи SQLCODE и SQLSTATE, както са дефинирани в ISO/ANS SQL92, се поддържат от опцията за предкомпилиране LANGLEVEL SQL92E. По време на предкомпилиране ще се генерира предупреждение SQL0020W, което показва, че LANGLEVEL не се поддържа. Това предупреждение се отнася само за компонентите, изброени под LANGLEVEL MIA в *Справочник на командите*, което е подмножество на LANGLEVEL SQL92E.

## Дефиниране на ред за сортиране

Разликите между EBCDIC и ASCII предизвикват разлики в реда на сортиране в различните бази данни и освен това влияят върху клаузите ORDER BY и GROUP BY. Един начин да се намалят тези разлики е да се създаде дефинирана от потребителя последователност, която да замества реда за сортиране EBCDIC. Можете да определите последователност, само когато създавате нова база данни. Повече информация потърсете в *Ръководство за разработка на приложения*, *Административен API Справочник* и *Справочник на командите*.

**Забележка:** Таблиците на базата данни сега могат да се съхраняват в DB2 Universal Database за OS/390 в ASCII формат. Това позволява по-бърз обмен на данни между DB2 Connect и DB2 Universal Database за OS/390, като премахва необходимостта да се осигуряват процедури за полета, които в противен случай трябва да се използват за конвертиране на данните и преподреждането им.

## Управление на референциален интегритет

Различните системи по различен начин обработват референциалните ограничения:

### DB2 Universal Database за OS/390

Трябва да е създаден индекс върху първичен ключ, преди да може да се създаде външен ключ с помощта на първичния ключ. Таблиците могат да се обръщат към себе си.

### DB2 за VSE и VM

За външния ключ автоматично се създава индекс. Таблиците не могат да се обръщат към себе си.

### DB2 Universal Database за AS/400

За външния ключ автоматично се създава индекс. Таблиците могат да се обръщат към себе си.



## DB2 Universal Database

При DB2 Universal Database базите данни автоматично се създава индекс за уникално ограничение, включително първичен ключ. Таблиците могат да се обръщат към себе си.

Други правила се различават по отношение степента на каскадност.

## Заклучване

Начинът, по който сървърът на базата данни изпълнява заключването може да повлияе върху някои приложения. Например приложенията, които са изградени на базата на заключване на ниво ред и ниво на изолация на защита на ниво ред не могат директно да се прехвърлят към системи, които изпълняват заключване на ниво страница. Поради тези скрити разлики може да се наложи приложенията да се пренастройват.

DB2 Universal Database за OS/390 и DB2 Universal Database има възможност за таймаут на заключване и да изпратят код за грешка на чакащите приложения.

## Разлики в SQLCODE и SQLSTATE

Различните реляционни бази данни на IBM не винаги генерират едни и същи кодове SQLCODE за подобни грешки. Можете да решите този проблем по един от следните начини:

- Използвайте SQLSTATE вместо SQLCODE за определена грешка.

SQLSTATE имат приблизително едно и също значение в различните видове бази данни и те генерират код SQLSTATE, който отговаря на SQLCODE.

- Трансформирайте кодовете SQLCODE от една система на друга система.

По подразбиране DB2 Connect трансформира кодовете SQLCODE и токените от всеки IBM хост или AS/400 сървър на вашата система DB2 Universal Database. Можете да определите ваш собствен файл за трансформиране на SQLCODE, ако искате да замените трансформирането по подразбиране или използвате сървър на база данни, който няма SQLCODE трансформиране (сървър на база данни, който не е на IBM). Освен това можете да изключите SQLCODE трансформирането.

За повече информация вижте Глава 11, “Преобразуване на SQLCODE” на страница 103.

## Използване на системни каталози

Системните каталози се различават между различните бази данни на IBM. Много от разликите могат да се скрият с използването на производни таблици. Информация потърсете в документацията на сървъра на база данни, който използвате.

Функциите на каталог в CLI заобикалят този проблем, като предоставят поддръжка на същия API и резултатни набори за заявките за каталог в рамките на фамилията DB2.

## Препълвания при числено конвертиране вследствие присвояване при извличане

Препълванията при числено конвертиране вследствие присвояване при извличане може да се обработва по различен начин от различните релационни бази данни на IBM. Например да приемем, че се извлича колона с данни от тип плаваща запетая в хост променлива с тип integer от DB2 Universal Database за OS/390 и от DB2 Universal Database. При конвертирането на стойност с тип плаваща запетая в стойност от тип integer, може да възникне препълване. По подразбиране DB2 Universal Database за OS/390 ще върне предупреждение SQLCODE и стойност NULL на приложението. За разлика DB2 Universal Database ще върне грешка за препълване при конвертиране. Препоръчва се, приложенията да избягват препълванията при числени конвертирания, възникнали при присвояване след извличане, като се поставят в хост променливи с подходящи размери.

## Нива на изолация

DB2 Connect приема следните нива на изолация, когато подготвите или свържете приложение:

- RR**     Защита при повторно четене
- RS**     Защита при четене
- CS**     Защита на ниво ред
- UR**     Защита при четене
- NC**     Без комит

Нивата на изолация са изброени от тези с най–силна защита до най–слаба защита. Ако хост или AS/400 сървър не поддържа нивото на изолация, което определите, се използва следващото по–високо поддържано ниво.

Таблица 2 показва резултата от всяко ниво на изолация на всеки хост или AS/400 сървър на приложения.

Таблица 2 (Страница 1 от 2). Нива на изолация

<b>DB2 Connect</b>	<b>DB2 Universal Database за OS/390</b>	<b>DB2 за VSE и VM</b>	<b>DB2 Universal Database за AS/400</b>	<b>DB2 Universal Database</b>
RR	RR	RR	note 1	RR
RS	note 2	RR	COMMIT(*ALL)	RS
CS	CS	CS	COMMIT(*CS)	CS
UR	note 3	CS	COMMIT(*CHG)	UR
NC	note 4	note 5	COMMIT(*NONE)	UR

Таблица 2 (Страница 2 от 2). Нива на изолация

DB2 Connect	DB2 Universal Database за OS/390	DB2 за VSE и VM	DB2 Universal Database за AS/400	DB2 Universal Database
-------------	----------------------------------	-----------------	----------------------------------	------------------------

**Забележки:**

1. Няма еквивалентна опция COMMIT на DB2 Universal Database за AS/400, която да съответства на RR. DB2 Universal Database за AS/400 поддържа RR като заключва цялата таблица.
2. В резултат се получава RR за версия 3.1 и RS за версия 4.1 с APAR PN75407 или версия 5.1.
3. В резултат се получава CS за версия 3.1 и UR за версия 4.1 или версия 5.1.
4. В резултат се получава CS за версия 3.1 и UR за версия 4.1 с APAR PN60988 или версия 5.1.
5. Ниво на изолация NC не се поддържа на DB2 за VSE и VM.

При DB2 Universal Database за AS/400 имате достъп до нежурнална таблица, ако приложението е свързано с ниво на изолация UR и създаване на блокове е установено на ALL, или ако нивото на изолация е NC.

## Запомнени процедури

- Обръщение

Клиентска програма може да се обърне към сървър чрез оператора SQL CALL. В този случай различните сървъри работят по малко по-различен начин.

### OS/390

Името на схемата не трябва да е с дължина повече от 8 байта, името на процедурата не трябва да е по-дълго от 18 байта, а запомнената процедура трябва да е дефинирана в каталога SYSIBM.SYSPROCEDURES на сървъра.

### VSE или VM

Името на процедурата не трябва да е по-дълго от 18 байта и трябва да е дефинирано в каталога SYSTEM.SYSROUTINES на сървъра.

### OS/400

Името на процедурата трябва да е SQL идентификатор. Освен това можете да използвате операторите DECLARE PROCEDURE или CREATE PROCEDURE, за да определите действителното име на пътека (името на схемата или на колекцията), за да се намери запомнената процедура.

Всички оператори CALL към DB2 за AS/400 от REXX/SQL трябва динамично да се подготвят и изпълнят от приложението, тъй като варианта на оператора CALL в REXX/SQL се трансформира до CALL USING DESCRIPTOR.

Синтаксисът на оператора SQL CALL можете да намерите в *SQL Справочник*. За информация как да използвате запомнени процедури, когато създавате приложни програми, се обърнете към *Ръководство за разработка на приложения*.

Можете да се обърнете към сървърна програма DB2 Universal Database при същите правила за параметрите, които сървърните програми използват на DB2 Universal Database за OS/390, DB2 Universal Database за AS/400, или DB2 за VSE и VM. За информация за обръщение към DB2 Universal Database запомнени

процедури се обърнете към *Ръководство за разработка на приложения*. Повече информация за правилата при използване на параметри на други платформи ще намерите в документацията на DB2 за съответната платформа.

Всички SQL оператори в запомнени процедури се изпълняват като част от SQL единица работа, стартирана от клиентска SQL програма.

- Не прехвърляйте стойности на индикатори със специално значение към или от запомнени процедури.

Между DB2 Universal Database, системите прехвърлят каквото поставите в променливите индикатори. Обаче когато използвате DB2 Connect, можете да прехвърлите само 0, -1 и -128 в променлива индикатор.

- Трябва да дефинирате параметър, за да върнете съобщения за грешка или предупреждения, срещнати от сървъра на приложения.

Сървърна програма на DB2 Universal Database може да обнови SQLCA, за да върне грешка или предупреждение, но запомнена процедура на DB2 Universal Database за OS/390 или DB2 Universal Database за AS/400 не поддържа тази възможност. Ако искате да върнете код за грешка от вашата запомнена процедура, трябва да го предадете като параметър. SQLCODE и SQLCA се определя само от сървъра за откритите от системата грешки.

- DB2 за VSE и VM Версия 7 или следваща и DB2 Universal Database за OS/390 версия 5.1 или следваща засега са единствените хост или AS/400 сървъри на приложения, които могат да върнат резултат от запомнена процедура.

## Stored Procedure Builder

DB2 Stored Procedure Builder осигурява лесна за използване среда за създаване, инсталиране и тестване на запомнени процедури. Позволява ви да се съсредоточите върху създаването на логиката на запомнената процедура, вместо върху подробностите, свързани с регистрирането, изграждането и инсталирането на запомнената процедура на DB2 сървър. Освен това при Stored Procedure Builder можете да разработите запомнени процедури на една операционна система и да ги изградите на други сървърни операционни системи.

Stored Procedure Builder е графично приложение, което поддържа бърза разработка. С помощта на Stored Procedure Builder можете да изпълните следните задачи:

- Да създавате нови запомнени процедури.
- Да изградите запомнени процедури на локални или отдалечени DB2 сървъри.
- Да модифицирате и изградите повторно съществуващи запомнени процедури.
- Да тествате и да откривате грешки при изпълнението на инсталирани запомнени процедури.

Можете да стартирате Stored Procedure Builder като отделно приложение от DB2 Universal Database програмната група или от някое от следните приложения за разработка:

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

Освен това можете да стартирате Stored Procedure Builder от Центъра за управление на DB2 за OS/390. Можете да стартирате Stored Procedure Builder като отделен процес от менюто Инструменти на центъра за управление, от линията с инструменти или от папка Stored procedures. Освен това от прозореца Stored Procedure Builder Project можете да експортирате една или повече избрани SQL

запомнени процедури, вградени в DB2 за OS/390 сървър до определен файл, който може да работи в рамките на процесора за обработка на команди (CLP).

Stored Procedure Builder управлява работата ви, като използва проекти. Във всеки проект на Stored Procedure Builder се записват вашите свързвания към определени бази данни, като DB2 за OS/390 сървъри. Допълнително можете да създадете филтри, които да представят подмножества на запомнените процедури на всяка база данни. Когато отворите нов или съществуващ проект на Stored Procedure Builder, можете да филтрирате запомнените процедури, така че да виждате само някои на база на тяхното име, схема, език или идентификатор (само за OS/390).

Информацията за свързване се записва в проекта на Stored Procedure Builder; следователно, когато отворите съществуващ проект, автоматично се появява промпт, за да въведете вашия потребителски идентификатор и парола за базата данни. Като използвате помощника Inserting SQL Stored Procedure, можете да вградите SQL запомнени процедури в DB2 за OS/390 сървър. За SQL запомнена процедура, вградена в DB2 за OS/390 сървър, можете да определите специфични параметри за компилиране, предварителни връзки, връзки, обвързване, време за изпълнение, WLM обкръжение и външни защиты.

Допълнително можете да получите информация за SQL разходите за SQL запомнената процедура, включително информация за процесорното време, други DB2 разходи за нишката, в която работи SQL запомнената процедура. По-специално можете да получите информация за разходите, свързани с времето за изчакване при повторно заключване, броя на извлечените страници, броя на прочетените Входно/Изходни елементи, броя на записаните Входно/Изходни елементи.

За да получите информация за разходите, Stored Procedure Builder се свързва към DB2 за OS/390 сървър, изпълнява SQL израз и се обръща към запомнена процедура (DSNWSPM), за да разбере колко процесорно време е използвала SQL запомнената процедура.

## НЕ АТОМАРЕН SQL блок

SQL блокът позволява няколко SQL оператора да се групират в отделен изпълним блок. Така може да се намали натоварването на мрежата и времето за отговор.

DB2 Connect поддържа НЕ АТОМАРЕН SQL блок. Това означава, че обработката на SQL блока продължава след грешка. (При АТОМАРЕН SQL блок, който не се поддържа от DB2 Connect, при възникване на грешка ще върне обратно цялата група на SQL блока.)

Операторите ще продължат изпълнението, докато се прекъсне от сървъра на приложения. Като цяло изпълнението на SQL блок от оператори ще се спре само в случай на сериозни грешки.

НЕ АТОМАРЕН SQL блок може да се използва с всички от поддържаните хост или AS/400 сървъри на приложения.

Ако възникнат няколко SQL грешки, кодовете SQLSTATE на първите седем оператора, които не са изпълнени, ще се върнат в полето SQLERRMC на SQLCA със съобщение, че са възникнали няколко грешки. За допълнителна информация се обърнете към *SQL Справочник*.

## Многосайтово обновяване с DB2 Connect

DB2 Connect ви позволява да изпълните многосайтово обновяване, също така известно като обновяване на няколко бази данни в рамките на една разпределена единица работа (DUOW). Дали можете да използвате тази възможност зависи от редица фактори:

- Приложната ви програма трябва да е предкомпилирана с опциите CONNECT 2 и SYNCPOINT TWOPHASE.
- Ако имате SNA мрежови връзки, можете да използвате двуфазов протокол на записване, който се поддържа от функцията DB2 Syncpoint Manager (SPM) на DB2 Connect Enterprise Edition Версия 7 за AIX, OS/2 и Windows NT. По този начин следните сървъри на бази данни могат да участват в разпределената единица работа:
  - DB2 за AS/400 версия 3.1 или следваща
  - DB2 за MVS/ESA версия 3.1 или следваща
  - DB2 за OS/390 версия 5.1 или следваща
  - DB2 за VM & VSE версия V5.1 или следваща

Горното е вярно за собствени DB2 UDB приложения и приложения, координирани от външен Монитор на Обработка на транзакции (TP) като IBM TXSeries, CICS за Open Systems, BEA Tuxedo, Encina Monitor и Microsoft Transaction Server.

**Забележка:** За допълнителна информация за XA концентратор вижте “DB2 Connect концентратор за връзки” на страница 119.

- Ако имате TCP/IP мрежа, сървър DB2 за OS/390 5.1 или следваща версия може да участва в разпределена единица работа. Ако приложението се контролира от Монитор за обработка на транзакции като IBM TXSeries, CICS за Open Systems, Encina Monitor или Microsoft Transaction Server, тогава трябва да използвате DB2 Мениджър за синхронизация (SPM).

в DB2 UDB Версия 7 (включително DB2 UDB EE, DB2 UDB EEE и DB2 Connect EE), DB2 SPM е усъвършенстван, като поддържа свързване по TCP/IP. Ако приложението е собствено приложение на DB2 UDB, тогава не е необходимо използването на DB2 SPM при двуфазово записване.

Ако се използва общ DB2 Connect Enterprise Edition сървър от собствени DB2 приложения и TP приложения за следене, за да се осъществи достъп до хост данни през TCP/IP, тогава трябва да се използва DB2 SPM.

Ако се използва отделен DB2 Connect Enterprise Edition сървър за достъп до хост данни с помощта едновременно на SNA и TCP/IP мрежови протоколи и е необходим двуфазово записване, тогава трябва да се използва DB2 SPM. Това е вярно както за DB2 приложенията, така и за TP приложенията за следене.

### Забележки:

1. Базите данни на DB2 Common сървър версия 2.1 могат да се обновят с двуфазов комит в единица работа *само* когато DB2 Universal Database за OS/390 версия 5.1 *не* е база данни за управление на транзакции.
2. Ако TM\_DATABASE за транзакциите е DB2 Universal Database за OS/390 версия 5.1, тогава DB2 CS V2.1 базите данни, участващи в тази транзакция са разрешени *САМО ЧЕТЕНЕ* за клиентското приложение.
3. Клиентските приложения за DB2 Universal Database Версия 7 могат да участват в разпределени единици работа с различни нива на сървъра на база данни, само

ако DB2 Universal Database за OS/390 версия 5.1 е *TM\_DATABASE* за транзакцията. За допълнителна информация се обърнете към съответната книга *DB2 Connect: Бърз старт*.

4. Нивата на бази данни, поддържани за операциите с двуфазово записване през TCP/IP зависят от нивото на DB2 клиента, нивото на *TM\_DATABASE* и нивата на участващите бази данни.

Силно препоръчваме *всички* клиенти, които имат достъп до базите данни DB2 Universal Database Версия 7 и DB2 Universal Database за OS/390 версия 5 да са DB2 Universal Database Версия 7. DB2 версия 2.1 клиенти *не могат* да инициират транзакции с двуфазово записване, ако в транзакцията участват база данни DB2 Universal Database за OS/390 версия 5.

## SQL оператори за хост или AS/400 сървър, поддържани от DB2 Connect

Следните оператори се компилират успешно за хост или AS/400 сървър, но не и за обработка върху DB2 Universal Database системи:

- ACQUIRE
- DECLARE (модификатор.(квалификатор.)таблица\_име TABLE ...
- LABEL ON

Тези оператори се поддържат и от процесор за обработка на команди.

Следните оператори се поддържат за обработка от хост или AS/400 сървър, но не са добавени към файла за свързване или пакета и не се поддържат от процесор за обработка на команди:

- DESCRIBE име\_на\_израз INTO име\_на\_състояние USING NAMES
- PREPARE име\_на\_израз INTO име\_на\_дескриптор USING NAMES FROM...

Предкомпиляторът извършва следните приемания:

- Хост променливите са входни променливи
- На оператора се присвоява уникален номер на раздел.

## SQL оператори за хост или AS/400 сървър, отхвърлени от DB2 Connect

Следните SQL оператори не се поддържат от DB2 Connect и не се поддържат от процесор за обработка на команди:

- COMMIT WORK RELEASE
- DECLARE име\_на\_състояние, име\_на\_израз STATEMENT
- DESCRIBE име\_на\_състояние INTO име\_на\_дескриптор USING xxxx (където xxxx е ANY, BOTH или LABELS)
- PREPARE име\_на\_израз INTO име\_на\_дескриптор USING xxxx FROM :променлива\_на\_хост (където xxxx е ANY, BOTH, или LABELS)
- PUT ...
- ROLLBACK WORK RELEASE
- SET :променлива\_на\_хост = CURRENT ...

DB2 за VSE и VM допълнителните оператори от динамичния SQL се отхвърлят със SQLCODE -104 и синтактична грешка.

## Реализиране на обратно таксуване на DB2 Universal Database за OS/390

Много инсталации на DB2 Universal Database за OS/390 реализират варианти за следене на ресурсите, които позволяват на системните администратори да асоциират използването на ресурси с достъпа на отделните потребители. Това може да се използва за таксуване на отделните потребители или техните отдели за използваните от тях ресурси. Тази практика обикновено се нарича *обратно таксуване*.

DB2 Connect продуктите позволяват на системните администратори да следят ресурсите на системата, консумирани от потребителите, които се свързват към базите данни чрез DB2 Connect. Можете да използвате низовете за отчитане, за да изпращате данни за отчитане от DB2 Connect към DB2 за сървъра на базата данни. Низът за отчитане комбинира данни, генерирани от системата и данни, въведени от потребителя. Тези данни позволяват на системните администратори да свържат използването на ресурси с всеки достъп на потребител и съответно да ги таксуват.

Низът за отчитане се изпраща с помощта на DRDA параметъра PRDDTA. Тъй като съдържанието на този параметър не е в архитектурата на DRDA, няма гаранция, че вашият сървър на приложения ще разпознае данните като данни за отчитане. Засега PRDDTA се поддържа само на MVS и OS/390 системи. Низът се съхранява като запис за отчитане.

Низът за отчитане се състои от 56 байта, генерирани от DB2 Connect (префиксът), следвани от до 199 байта, определени от потребителя (суфиксът), като има максимална дължина от 255.

Таблица 3 показва генерираните от системата полета. Всяко от тези полета се запълва влясно с интервали.

<i>Таблица 3 (Страница 1 от 2). Полета на низа на отчитане, генерирани от DB2 Connect</i>		
<b>Име на поле</b>	<b>Дължина</b>	<b>Описание</b>
acct_str_len	1	Шестнайсетично–десетична стойност, представляваща дължината на низа на отчитане минус 1. Например X'3C'.
client_prdid	8	Идентификационният номер на клиентския софтуер. Например идентификационния номер за DB2 Universal Database Версия 7 е SQL07010.
client_platform	18	Платформата на клиента например е AIX, OS/2, DOS, or Windows.
client_appl_name	20	Първите 20 символа от името на приложението на потребителя, например payroll.



Таблица 3 (Страница 2 от 2). Полета на низа на отчитане, генерирани от DB2 Connect		
Име на поле	Дължина	Описание
client_authid	8	Оторизиращия идентификатор на приложението на потребителя, например SMITH.
suffix_len	1	Шестнайсетично–десетична стойност, представляваща дължината на въведения от потребителя суфикс. X'00' означава, че няма въведен от потребителя суфикс.

Въведеният от потребителя суфикс е един от следните:

- Стойността, определена от приложението със sqlesact() API
- Стойността на променливата на обкръжението DB2ACCOUNT
- Стойността на конфигурационния параметър DFT\_ACCOUNT\_STR (низът за отчитане по подразбиране)
- Низ null.

Ако суфиксът е по-дълъг от 199 символа, той се съкращава. За да сте сигурни, че низът за отчитане е конвертиран правилно, когато е прехвърлен на хоста или AS/400 сървъра на базата данни, трябва да използвате символите от A до Z, 0 до 9 и долната черта (\_).

Препоръчва се низът на отчитане да се определя по API метода. Приложението трябва да се обърне към API, преди да се свърже към база данни. Ако искате да промените низа на отчитане в рамките на приложение (например да изпратите различен низ при свързване към различна база данни), отново се обърнете към API. В противен случай стойността остава в сила до приключване на приложението.

Ако преди първата заявка за свързване към база данни не използвате sqlesact() API, ще се прочете променливата на обкръжението DB2ACCOUNT. Тази стойност остава в сила до приключване на приложението или на фонов процесор за обработка на команди процес. За да определите нов суфикс в низа на отчитане след първото свързване към базата данни, трябва да използвате sqlesact() API или да приключите приложението или фоновия процес CLP и да рестартирате, като установите DB2ACCOUNT с новата стойност.

Ако не съществува стойност за DB2ACCOUNT, се използва системния конфигурационен параметър DFT\_ACCOUNT\_STR. Тази стойност по подразбиране може да е полезна за клиенти на базата данни, които нямат възможност да препращат низ на отчитане към DB2 Connect. Ако не съществува, се използва низ null.

Следват примери за низове на отчитане:

```
x'3C'SQL070100S/2      cheque      SMITH  x'05'DEPT1
x'37'SQL070100S/2      cheque      SMITH  x'00'
```

В първия пример дефинираният от потребителя суфикс е DEPT1. Във втория пример това е низът null.

## Изпращане на информация за отчитане към DB2 за OS/390 сървър

Много потребители на мейнфрейм разглеждат подробното отчитане на ресурсите, използвани от различните приложения, като важна част от техните процедури на работа. DB2 за OS/390 осигурява разширени възможности за изготвяне на доклади за отчитане, които позволяват на информационните отдели да таксуват отделните потребители и отдели за използваните от тях ресурси. Този процес често се нарича *обратно таксуване*. DB2 Connect продуктите позволяват точно отчитане на хост ресурсите, използвани от приложенията за PC и UNIX на базата на съществуващите отчети и процедури за отчитане.

DB2 Connect реализира тази функция гъвкаво, като осигурява:

- Низ на отчитане по подразбиране за всеки вид сесия, генериран от определен сървър DB2 Connect Enterprise Edition.
- Два механизма за отделните потребители и приложенията, чрез които да определят акаунта, на който да се записва таксуването.

### Определяне на низът на отчитане

Низът на отчитане по подразбиране се определя от конфигурационния параметър `dft_account_str` на DB2 Connect работната станция. Тази стойност по подразбиране може да е полезна за клиенти на базата данни, които нямат възможност да препращат низ на отчитане към DB2 Connect. Например приложенията, разработени преди продуктите от версия 2.

Има два начина за клиентските приложения да заменят низът на отчитане по подразбиране, установен от DB2 Connect сървъра:

- С помощта на API командата за установяване на низа на отчитане: `sqlsact()`

API командата `sqlsact()` се извиква преди приложението да се свърже към базата данни. Трябва да използвате този метод, защото:

- Обръщението към API не предизвиква разходите за четене на стойност на регистър.
- Не е необходимо да се обръщате към тази API команда отново, освен ако не искате да използвате нов низ на отчитане при следващи заявки за свързване. Ако използвате стойност от регистъра, трябва да приключите процеса на приложението, да дефинирате отново `DB2ACCOUNT` и след това да стартирате отново процеса.

За допълнителна информация относно използването на тази API команда се обърнете към *Административен API Справочник*.

- Използване на стойността от регистъра `DB2ACCOUNT` на клиентската работна станция.

Ако не използвате `sqlsact()` API, преди първата заявка при свързване към база данни, ще се прочете променливата на обкръжението `DB2ACCOUNT`. Низът на отчитане се използва при всички последващи заявки към базата данни.

**Забележка:** Когато определяте низа на отчитане, трябва да спазвате следните правила:

1. Използвайте символите A–Z, 0–9 или '\_' (долно тире).
2. Ограничете низа на отчитане до 199 байта— по-дългите низове се съкращават.

---

## Полезни издания

Следните издания може да ви помогнат да разработите приложения, които да работят в разпределено обкръжение:

- Книгите за приложно програмиране за отделните бази данни може да съдържат информация, която да е различна за всеки продукт.
- SQL справочниците за всяка база данни ще ви помогнат да се уверите, че приложението съдържа само поддържаните SQL оператори и че са с правилен синтаксис.
- *DB2 Universal Database за OS/390 Reference for Remote DRDA Requesters and Servers* осигурява най-новата информация за таксуването на DB2 Universal Database за OS/390 потребители.
- *SQL Справочник* осигурява дискусия от високо ниво за разликите в релационните бази данни на IBM и освен това дискутира за това как да се работи с някои специфични разлики.
- DRDA изданията осигуряват информация за планиране, свързване, програмиране и определяне на проблемите в DRDA обкръжение. Списък със заглавията и номерата за поръчка вижте в “Електронни издания, свързани с DRDA” на страница 15.



---

## Глава 5. Изпълнение на ваши собствени приложения

Различни типове приложения могат да осъществяват достъп до DB2 бази данни:

- Приложения, разработени чрез Клиент за разработка на DB2 приложения, които включват вграден SQL, API, запомнени процедури, дефинирани от потребителя функции или обръщения към DB2 CLI.
- ODBC приложения като Lotus Approach.
- JDBC приложения и аплети.
- Net.Data макроси, съдържащи HTML и SQL.

Приложение на DB2 клиент може да осъществява достъп до отдалечена база данни, без да знае физическото ѝ местоположение. DB2 клиентът определя местоположението на базата данни, управлява предаването на заявките към сървъра на базата данни и връща резултатите.

Най-общо, за да изпълните приложение на клиент на базата данни, използвайте следните стъпки:

Стъпка 1. Уверете се, че сървърът е конфигуриран и работи.

Уверете се, че мениджър на базата данни е стартиран на сървъра на базата данни, към който се свързва приложната програма. Ако не е, трябва да подадете командата **db2start** на сървъра, преди да стартирате приложението.

Стъпка 2. Уверете се, че можете да се свързвате към базата данни, която използва приложението.

Стъпка 3. Свържете помощните програми и приложенията към базата данни. Вижте “Свързване на помощните програми на база данни” за информация относно свързването на помощните програми.

Стъпка 4. Изпълнете приложната програма.

---

### Свързване на помощните програми на база данни

Трябва да свържете помощните програми на базата данни (`import`, `export`, `georg`, процесор за обработка на команди) и файловете на DB2 CLI за свързване към всяка база данни, преди те да могат да се използват с тази база данни. В мрежово обкръжение, ако използвате множество клиенти, които се изпълняват на различни операционни системи или са с различни версии или сервизни нива на DB2, трябва да свържете помощните програми по веднъж за всяка комбинация от операционна система и DB2 версия.

Свързването на помощна програма създава *пакет*, който е обект, включващ цялата необходима информация за обработката на конкретни SQL оператори от даден файл с код.

Файловете за свързване са групирани в различни `.lst` файлове в директорията `bnd` под инсталационната директория. Всеки файл е специфичен за сървъра.

---

### Изпълнение на CLI/ODBC програми

CLI (DB2 интерфейс за вградени SQL оператори) средата за изпълнение и ODBC драйверът са включени в DB2 клиентите като опционални компоненти при инсталиране.

Тази поддръжка разрешава на приложения, разработени чрез API на ODBC и DB2 CLI да работят с всяка DB2 сървър. Поддръжка на разработка на приложения за DB2 CLI се осигурява от Клиент за разработка на DB2 приложения (Клиент за разработка на DB2 приложения), който е включен във вашия DB2 сървър.

За да могат DB2 CLI или ODBC приложенията да осъществят достъп до DB2, за DB2 CLI пакетите трябва да е изпълнено свързване на сървъра. Въпреки че това ще стане автоматично при първата връзка, ако потребителят има необходимото право за свързване на пакетите, препоръчва се администраторът да направи това с всяка версия на клиента на всяка платформа, която ще осъществява достъп до сървъра.

Следващите основни стъпки трябва да се предприемат на клиентската система, за да се даде възможност на DB2 CLI и ODBC приложения да имат достъп до DB2 бази данни. В тези инструкции се предполага, че вие успешно сте се свързали с DB2 посредством валиден потребителски ID и парола. В зависимост от платформата, много от тези стъпки се изпълняват автоматично. За пълна подробна информация, вижте раздела, който се отнася специално за вашата платформа.

- Стъпка 1. Използвайте Асистента за конфигуриране на клиенти, за да добавите базата данни (ако имате отделни клиент и сървър машини), за да може нейните потребителски модели и бази данни да са известни за Центъра за управление, а след това добавете моделите и базите данни за тази система. (Вашата локална система е представена като иконата **Локална**.) Ако нямате достъп до тази програма, може да използвате команда **catalog** в процесора за обработка на команди.
- Стъпка 2. На различни от OS/2 платформи, CLI/ODBC драйверът е опционален компонент при инсталиране на DB2 клиент. Уверете се, че той е избран при инсталирането. На OS/2 трябва да използвате иконата **Инсталиране на ODBC драйвер**, за да инсталирате едновременно DB2 CLI/ODBC драйвера и мениджъра за ODBC драйвера.
- Стъпка 3. За да осъществите достъп до DB2 базата данни от ODBC:
  - a. Трябва да бъде инсталиран мениджърът на ODBC драйвери (от Microsoft или друг доставчик). Това се прави по подразбиране при инсталиране на DB2 на 32-битови Windows системи).
  - b. DB2 базите данни трябва да бъдат регистрирани като ODBC източници на данни. Мениджърът за ODBC драйвери не прочита информацията от DB2 каталога; вместо това, той се обръща към свой собствен списък на източниците на данни.
  - c. Ако DB2 таблица няма уникален индекс, много ODBC приложения ще я отворят само за четене. Трябва да бъде създаден уникален индекс за всяка DB2 таблица, която ще се обновява от ODBC приложение. Вижте оператора **CREATE INDEX** в *SQL Справочник*. Посредством Центъра за управление, променете настройките на таблицата, изберете етикета **Първичен ключ** и преместете една или повече колони от списъка на наличните колони в списъка на колоните за първичен ключ. Всяка колона, която избирате като част от първичния ключ, трябва да бъде дефинирана като NOT NULL.
- Стъпка 4. Ако е необходимо, може да настроите различни конфигурационни ключови думи за CLI/ODBC, за да промените поведението на CLI/ODBC и приложенията, които го използват.

Ако сте следвали горните стъпки за инсталиране на ODBC поддръжка и сте добавили DB2 бази данни като ODBC източници на данни, вашите ODBC приложения сега ще могат да осъществяват достъп до тях.

## Специфична за платформата подробна информация за CLI/ODBC достъп



Специфичната за платформата подробна информация как да дадем на DB2 CLI и ODBC приложения възможност за достъп до DB2 е разделена на следните категории:

- “Достъп на клиенти от Windows 32–битова операционна система до DB2 посредством CLI/ODBC”
- “OS/2 Достъп на клиенти от DB2 посредством CLI/ODBC” на страница 55

### Достъп на клиенти от Windows 32–битова операционна система до DB2 посредством CLI/ODBC

Преди DB2 CLI и ODBC приложения да могат успешно да осъществяват достъп до DB2 база данни от Windows клиент, изпълнете на клиентската система следните стъпки:

1. DB2 базата данни (и възелът, ако базата данни е отдалечена) трябва да бъде каталогизирана. За да направите това, ползвайте CCA (или процесор за обработка на команди).

За повече информация се обърнете към електронната помощ в CCA (или към командите **CATALOG DATABASE** и **CATALOG NODE** в *Справочник на командите*).

2. Проверете дали са инсталирани Microsoft мениджърът за ODBC драйвери и DB2 CLI/ODBC драйверът. На Windows 32–битова операционна система се инсталират и двете с DB2, освен ако при инсталирането не е премахната ръчно маркировката на ODBC компонента.

За да се убедите, че и двете са инсталирани на машината:

- a. Стартирайте Microsoft ODBC Administrator от иконата в Control Panel, или задайте подходящата команда от командния ред: **odbcad32.exe**.
- b. Щракнете върху етикета **ODBC драйвери**.
- c. Проверете дали в списъка се появява "IBM DB2 ODBC DRIVER".

Ако Microsoft ODBC Мениджърът за драйверите или IBM DB2 CLI/ODBC драйверът не е инсталиран, стартирайте отново DB2 инсталацията и изберете компонента ODBC на Windows 32–битова операционна система.

3. Регистрирайте DB2 базата данни с мениджъра за ODBC драйвери като *източник на данни*. На Windows 32–битова операционна система вие може да направите източника на данни достъпен за всички потребители на системата (системен източник на данни) или само за текущия потребител (потребителски източник на данни). Използвайте един от тези методи за добавяне източника на данни:

- Ползвайки CCA:
  - a. Изберете псевдонима на DB2 базата данни, която желаете да добавите като източник на данни.

- b. Щракнете върху бутона **Характеристики**. Отваря се прозорец Характеристики на базата данни.
- c. Изберете кутия за избор **Регистрирай тази база данни за ODBC**.
- d. На Windows 32–битова операционна система може да използвате радио–бутоните, за да добавите източника на данни като потребителски или като системен източник на данни.
- С помощта на Microsoft **32–битовото ODBC Средство за администриране**, до което може да достигнете от иконата в Контролния панел или чрез стартиране на **odbcad32.exe** от командния ред:
  - a. В Windows 32–битова операционна система списъка на потребителските източници на данни се появява по подразбиране. Ако желаете да добавите системен източник на данни, щракнете върху бутон **Системен DSN** или върху маркер **Системен DSN** (в зависимост от платформата).
  - b. Щракнете върху бутона **Добави**.
  - c. Щракнете двукратно върху IBM DB2 ODBC драйвера в списъка.
  - d. Изберете DB2 базата данни, която да добавите и щракнете върху **ОК**.
- На Windows 32–битова операционна система има команда, която може да бъде подадена в процесор за обработка на команди, за регистриране на DB2 базата данни с мениджъра за ODBC драйвери като източник на данни. Администраторът може да създаде скрипт на процесор за обработка на команди за регистриране на необходимите бази данни. Тогава този скрипт ще може да бъде пуснат на всички машини, които трябва да осъществяват достъп до DB2 базите данни през ODBC.

*Справочник на командите* съдържа повече информация за командата CATALOG:

CATALOG [ потребител | система ] ODBC DATA SOURCE

- Стъпка 4. Конфигурирайте CLI/ODBC драйвера посредством CCA: (опционално)
- a. Изберете псевдонима на DB2 базата данни, която желаете да конфигурирате.
  - b. Щракнете върху бутона **Характеристики**. Отваря се прозорец Характеристики на базата данни.
  - c. Щракнете върху бутон **Настройки**. Отваря се прозорец CLI/ODBC настройки.
  - d. Щракнете върху бутона **Допълнителни**. Може да настроите конфигурационните ключови думи от прозореца, който се отваря. Тези ключови думи са асоциирани с *псевдонима* на базата данни и имат ефект върху всички CLI/ODBC приложения, които осъществяват достъп до базата данни. Електронната помощ обяснява всички ключови думи, както и приложение в електронното ръководство *Приложение за инсталиране и конфигуриране*.
- Стъпка 5. Ако сте инсталирали ODBC достъп (както е описано по–горе), сега може да осъществявате достъп до DB2 данни чрез ODBC приложения. Стартирайте ODBC приложението и отидете в прозорец Отвори. Изберете тип на файл **ODBC бази данни**. DB2 базите данни, които сте добавили като ODBC източници на данни, ще могат да се изберат от



списъка. Много ODBC приложения ще отворят таблицата в режим само за четене, освен ако не съществува уникален индекс.

## OS/2 Достъп на клиенти от DB2 посредством CLI/ODBC

За да могат DB2 CLI и ODBC приложения успешно да осъществяват достъп до DB2 база данни от OS/2 клиент, на клиентската система изпълнете следните стъпки:

1. DB2 базата данни (и възелът, ако базата данни е отдалечена) трябва да бъде каталогизирана. За да направите това, ползвайте CCA (или процесор за обработка на команди).

За повече информация вижте електронната помощ в CCA. (or the **CATALOG DATABASE** and **CATALOG NODE** commands in the *Справочник на командите*).

2. Ако използвате ODBC приложения за достъп до DB2 данни, изпълнете следните стъпки. (Ако използвате само CLI приложения, прескочете тази стъпка и отидете на следващата стъпка.)

- a. Проверете дали е инсталиран Мениджър за ODBC драйвери. Мениджърът за ODBC драйвери не се инсталира с DB2. Препоръчваме ви да използвате мениджъра за драйвери, предоставен с вашето ODBC приложение. Уверете се, че е инсталиран и CLI/ODBC драйвера:

- 1) Стартирайте средството за ODBC администриране, както е описано в документацията му. Това обикновено се прави по един от двата начина:

- Щракнете двукратно върху **ODBC** в OS/2, и щракнете два пъти върху иконата **ODBC Administrator**.
- Стартирайте **odbcadm.exe** от командния ред.

Ще се отвори прозорецът Data Sources.

- 2) Щракнете върху бутона **Drivers**. Ще се отвори прозорецът Drivers.
- 3) Проверете дали "IBM DB2 ODBC DRIVER" е указан в списъка.

Ако Мениджърът за ODBC драйвери не е инсталиран, следвайте инсталационните инструкции, които се предоставят с вашето ODBC приложение. Ако не е инсталиран IBM DB2 CLI/ODBC драйвера, щракнете двукратно върху иконата **Инсталиране на ODBC драйвер** в папка DB2, за да инсталирате DB2 CLI/ODBC драйвер.

- b. Регистрирайте DB2 базата данни с мениджъра за ODBC драйвери като *източник на данни*, чрез един от следните методи:

- С помощта на CCA:

- 1) Изберете псевдонима на DB2 базата данни, която желаете да добавите като източник на данни.
- 2) Щракнете върху бутона **Характеристики**.
- 3) Изберете **Регистрирай тази база данни за ODBC**.

- Използване на Мениджъра за ODBC драйвери:

- 1) Стартирайте Мениджъра за ODBC драйвери, както е описано в документацията му. Това обикновено се прави по един от двата начина:
  - Щракнете двукратно върху папка **ODBC** в OS/2 и след това върху икона **ODBC администратор**.
  - Стартирайте **odbcadm.exe** от командния ред.
- 2) Щракнете върху бутон **Добави** от прозорец Източници на данни. Отваря се прозорец Добавяне на източници на данни.
- 3) Щракнете двукратно върху IBM DB2 ODBC DRIVER в списъка.

- 4) Изберете DB2 базата данни, която да добавите и щракнете върху **ОК**.
3. Конфигурирайте CLI/ODBC драйвера посредством CCA: (опционално)
  - a. Изберете псевдонима на DB2 базата данни, която желаете да конфигурирате.
  - b. Щракнете върху бутона **Характеристики**. Отваря се прозорец Характеристики на базата данни.
  - c. Щракнете върху бутон **Настройки**. Отваря се прозорец CLI/ODBC настройки.
  - d. Щракнете върху бутона **Допълнителни**. Може да настроите конфигурационните ключови думи и прозореца, който се появява. Тези ключови думи са асоциирани с *псевдонима* на базата данни и имат ефект върху всички CLI/ODBC приложения, които осъществяват достъп до базата данни. Електронната помощ обяснява всички ключови думи, както и приложението в ръководството *Приложение за инсталиране и конфигуриране*.
4. Ако сте инсталирали ODBC достъп (както е описано по-горе), сега може да осъществявате достъп до DB2 данни чрез ODBC приложения. Стартирайте ODBC приложението и отидете в прозорец Отвори. Изберете тип на файл **ODBC бази данни**. DB2 базите данни, които сте добавили като ODBC източници на данни, ще могат да се изберат от списъка. Много ODBC приложения ще отворят таблицата в режим само за четене, освен ако не съществува уникален индекс.

## Подробна конфигурационна информация

Разделът “Специфична за платформата подробна информация за CLI/ODBC достъп” на страница 53 трябва да ви предостави необходимата информация. *Приложение за инсталиране и конфигуриране* съдържа допълнителна информация за настройването и ползването на DB2 CLI и ODBC приложения. (Подръчното *Приложение за инсталиране и конфигуриране* ръководство се намира в директорията `x:\doc\en\html`, където *x*: е буквата, приписана на CD-ROM устройството, а *en* е двусимволния код на страната, с който се представя езикът, например *en* за английски.) Тази информация е полезна, когато DB2 средството за поддръжка не е достъпно, и за администратори, които търсят по-подробна информация.

Следващите теми са покрити в *Приложение за инсталиране и конфигуриране* подръчното ръководство:

- Как да свържем DB2 CLI/ODBC драйвера с базата данни
- Как да настроим ключовите думи на CLI/ODBC конфигурацията
- Конфигуриране на `db2cli.ini`

---

## Изпълнение на Java програми

Може да създавате Java програми, които да използват DB2 бази данни с помощта на съответния Набор средства за създаване на Java приложения (JDK) за AIX, HP-UX, Linux, OS/2, PTX, Silicon Graphics IRIX, Solaris или Windows 32-битова операционна система. JDK съдържа Java Database Connectivity (JDBC), динамичен SQL API за Java.

За DB2 поддръжка на JDBC трябва да включите компонента DB2 Java Разрешител при инсталиране на DB2 клиента. С DB2 поддръжката на JDBC може да създавате

и изпълняват JDBC приложения и аплети. Те съдържат само динамичен SQL и използват Java интерфейса за подаване на SQL операторите към DB2.

Клиент за разработка на DB2 приложения осигурява поддръжка на Java вградения SQL (SQLJ). С DB2 поддръжката на SQLJ и на JDBC може да създавате и изпълняват SQLJ приложения и аплети. Те съдържат статичен SQL и използват вградените SQL оператори, които са свързани с DB2 базата данни.

Java може да се използва и на сървър за създаване на JDBC и SQLJ запомнени процедури и потребителски дефинирани функции (UDF).

Създаването и изпълнението на различни типове Java програми изисква поддръжката от различни компоненти на DB2:

- За да изградите JDBC приложения, трябва да инсталирате DB2 клиент с компонента DB2 Java Разрешител. За да изпълняват JDBC приложения, вашият DB2 клиент с компонента DB2 Java Разрешител трябва да се свърже с DB2 сървър.
- За да изградите SQLJ приложения, трябва да инсталирате DB2 SDK и DB2 Административен клиент с компонента DB2 Java Разрешител. За да изпълняват SQLJ приложения, вашият DB2 клиент с компонента DB2 Java Разрешител трябва да се свърже с DB2 сървър.
- За да изградите JDBC аплети, трябва да инсталирате DB2 клиент с компонента DB2 Java Разрешител. За да изпълняват JDBC аплети, не е необходимо на клиентската машина да бъдат инсталирани никакви DB2 компоненти.
- За да изградите SQLJ аплети, трябва да инсталирате DB2 SDK и DB2 Административен клиент с компонента DB2 Java Разрешител. За да изпълняват SQLJ аплети, не е необходимо на клиентската машина да бъдат инсталирани никакви DB2 компоненти.

За подробна информация относно създаването и изпълняването на JDBC и SQLJ програми вижте *Ръководство за създаване на приложения*. За допълнителна информация за DB2 програмирането на Java вижте *Ръководство за разработка на приложения*. Там се обяснява създаването и изпълняването на JDBC и SQLJ приложения, аплети, запомнени процедури и UDF.

За последната, обновена информация за DB2 и Java, посетете Web страницата на:

<http://www.ibm.com/software/data/db2/java>

## Конфигуриране на обкръжението

За да създавате и изпълняват DB2 Java програми, трябва да инсталирате и конфигурирате съответната версия на Набора средства за създаване на Java приложения (JDK) на вашата машина за разработка:

<b>AIX:</b>	Набор средства за създаване на Java приложения (JDK) Версия 1.1.2 за AIX от IBM.
<b>HP-UX:</b>	Набор средства за създаване на Java приложения за HP-UX Версия 1.1.3 или по-нова от Hewlett-Packard.
<b>Linux:</b>	Набор средства за създаване на Java приложения (JDK) 1.1.7 Версия 1a или по-нова за Linux от Blackdown Organization.
<b>OS/2:</b>	Набор средства за създаване на Java приложения (JDK) Версия 1.1.4 за OS/2 от IBM.

<b>PTX:</b>	Набор средства за създаване на Java приложения (JDK), ptx/JSE, версия 3.0.0 (Sun JDK v1.2.1) от IBM.
<b>Silicon Graphics IRIX:</b>	Среда за разработка на Java приложения 3.1 (Sun JDK 1.1.5) и Java Изпълнима среда 3.1 (Sun JRE 1.1.5) от Silicon Graphics, Inc.
<b>Solaris:</b>	Набор средства за създаване на Java приложения (JDK) Версия 1.1.4 за Solaris и пакета Solaris Native Thread, от Sun Microsystems.
<b>Windows 32-битова операционна система</b>	Набор средства за създаване на Java приложения (JDK) Версия 1.1 за Win32 от Sun Microsystems.

За информация относно инсталирането и конфигурирането на всеки JDK вижте:

<http://www.ibm.com/software/data/db2/java>

За всички поддържани платформи трябва да инсталирате и конфигурирате DB2 клиент с компонента DB2 Java Разрешител. За да изградите SQLJ програми, трябва да инсталирате и конфигурирате DB2 Административен клиент с компонента DB2 Java Разрешител.

За да изпълнявате DB2 Java запомнени процедури или UDF, трябва да промените и конфигурацията на DB2 мениджъра на базата данни така, че да съдържа пътеката, където е инсталиран JDK версия 1.1 на машината за разработка. Може да направите това, като въведете следното от командния ред:

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

където /home/smith/jdk11 е пътеката, където е инсталиран JDK версия 1.1.

Може да видите конфигурацията на DB2 мениджъра на базата данни, за да проверите правилната стойност на полето JDK11\_PATH, като въведете следната команда:

```
db2 get dbm cfg
```

Може да пренасочите изхода към файл за по-лесно преглеждане. Полето JDK11\_PATH е близо до началото. За допълнителна информация за тези нови команди се обърнете към *Справочник на командите*.



На Solaris, някои реализации на Java Virtual Machine не работят добре в програми, които се изпълняват в "setuid" обкръжение. Поделената библиотека, която съдържа Java интерпретатора, libjava.so, също може да не се зареди успешно. Като решение, може да създадете символни връзки за всички необходими JVM поделени библиотеки в /usr/lib с помощта на команда, подобна на следната (в зависимост от това, къде Java е инсталирана на вашата система):

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

За повече информация за подобни решения вижте:

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

За да изпълнявате Java програми, променливите на обкръжението, изброени по-долу, се обновяват автоматично при инсталирането на DB2 на OS/2 и Windows операционни системи и при създаването на модела на UNIX платформи.

**На UNIX платформи:**

- CLASSPATH съдържа "." и файла sqllib/java/db2java.zip

- На AIX, Linux, Silicon Graphics IRIX и Solaris: LD\_LIBRARY\_PATH съдържа директорията sqllib/lib; на HP-UX: SHLIB\_PATH съдържа директорията sqllib/lib
- На AIX, Linux, PTX, Silicon Graphics IRIX и Solaris: LD\_LIBRARY\_PATH съдържа директорията sqllib/lib; на HP-UX: SHLIB\_PATH съдържа директорията sqllib/lib
- Само на Solaris: THREADS\_FLAG е настроен на "native"

#### На Windows и OS/2 платформи:

- CLASSPATH съдържа "." и файла %DB2PATH%\java\db2java.zip

За да се създават и изпълняват SQLJ програми, CLASSPATH също трябва да е автоматично обновена със следните файлове:

#### На UNIX платформи:

- sqllib/java/sqlj.zip (необходим за създаването на SQLJ програми)
- sqllib/java/runtime.zip (необходим за изпълняването на SQLJ програми)

#### На Windows и OS/2 платформи:

- %DB2PATH%\java\sqlj.zip (необходим за създаването на SQLJ програми)
- %DB2PATH%\java\runtime.zip (необходим за изпълняването на SQLJ програми)

## Java приложения

Стартирайте вашето приложение от Работната площ или от командния ред, изпълнявайки Java интерпретатора с изпълнимата програма със следната команда:

```
java име_на_програма
```

където име\_на\_програма е името на програмата.

DB2 JDBC драйверът обслужва JDBC API обръщанията от вашето приложение и използва DB2 CAE за предаване на заявките на сървъра и получаване на резултатите. SQLJ приложението трябва да бъде свързано с базата данни, преди да бъде изпълнено.

## Java аплети

Тъй като Java аpletите се предават по Web, на вашата DB2 машина (сървър или клиент) трябва да бъде инсталиран web сървър.

За да изпълните аплета, уверете се, че вашият .html файл е правилно конфигуриран. Стартирайте JDBC аplet сървъра на TCP/IP порта, указан в .html файла. Например, ако сте указали:

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

тогава трябва да въведете:

```
db2jstrt 6789
```

Трябва да се уверите, че работната директория е достъпна за вашия web браузър. Ако не е, копирайте .class и .html файловете на вашия аplet в директория, която е достъпна. За SQLJ аплети, трябва да копирате също така и .class и .ser файловете на профила.

Копирайте файла sqllib/java/db2java.zip в същата директория, като тази на другите файлове. За SQLJ аплети, копирайте и файла sqllib/java/runtime.zip в

тази директория. На вашата клиентска машина стартирайте web браузъра (който поддържа JDK 1.1) и заредете .html файла.

Когато вашият аplet направи обръщение към JDBC API за свързване с DB2, JDBC драйверът реализира отделни комуникации с DB2 базата данни през JDBC аplet сървъра, намиращ се на DB2 сървъра. SQLJ аpletът трябва да бъде свързан с базата данни преди да бъде изпълнен.

---

## Част 2. Справочник и отстраняване на проблеми





---

## Глава 6. Обновяване на директории на бази данни

DB2 Connect използва следните директории за управление на информацията за базите данни, към които се свързва:

- *Директория на възли*, която съдържа информация за мрежовия адрес и комуникационния протокол за всеки хост или AS/400 сървър на база данни, до която има достъп DB2 Connect.
- *Директорията на услугите за връзка на базата данни (DCS)*, която съдържа специфична информация за хост или AS/400 сървър на бази данни.
- *Системна директория на база данни*, която се състои от информация за име, възел и автентичност за всяка база данни, до която се осъществява достъп от DB2 Connect.

### Забележки:

1. Преди да обновите тези директории, трябва да конфигурирате комуникациите между хост или AS/400 сървъра на базата данни и работните станции. За допълнителна информация се обърнете към *Приложение за инсталиране и конфигуриране*.
2. Под OS/2 и Windows 32-битова операционна система директориите на базата данни може да се обновят с помощта на Асистента за конфигуриране на клиент (CCA) на DB2 Universal Database.

На всички други платформи директориите на базата данни трябва да се обновят с помощта на процесора за обработка на команди (CLP).

3. “Обновяване на директориите” на страница 72 предоставя примерен синтаксис на командите. За допълнителна информация се обърнете към *Справочник на командите*.
4. Ако използвате DCE за всеки хост или AS/400 сървър на база данни за базата данни, към която се свързвате, трябва да обновите тези директории или да съхраните еквивалентна информация в глобална директорията DCE. Информация за DCE потърсете в Приложение D, “Използване на DCE директорните услуги” на страница 173 и *Ръководство за администриране*. Този раздел приема, че *не* използвате DCE директорни услуги.

---

## Събиране на информация

Приложение В, “Работен лист за персонализиране на директории” на страница 167 показва информацията, която трябва да съберете. Може да се окаже удобно за вас, ако направите копие на таблицата и въведете стойностите за вашата система.

### Директория на възлите

Можете да определите следната информация в директорията на възлите:

#### Име на възел

Прякор за хоста или AS/400 сървъра на базата данни, на който се намира отдалечената база данни. Това име се определя от потребителя. Напишете едно и също име на възел и в двете таблици – с параметрите на директорията на възлите и с параметрите на системната директория на базата данни.

Формат: 1–8 еднобитови буквено–цифрови символи, включително числен знак (#), знака at (@), знака за долар (\$), и долно тире (\_). Не може да започва с долно тире или число.

#### **Протокол**

Може да е APPC или TCP/IP.

#### **Име на символно предназначение**

При определяне на APPC възел използвайте името на символно предназначение, което е било определено в таблицата с информацията за CPI комуникациите (например името на CPI–C Symbolic Destination Properties, когато се използва Microsoft SNA сървър). Трябва да получите тази стойност от човека, който е инсталирал и/или конфигурирал SNA. В името на символното предназначение е от значение използването на малки и главни букви (може да получите код на връщане SQL1338, ако има несъответствие между главни и малки букви в имената).

#### **Тип на защита**

Типът на защитните проверки, които ще се изпълнят. За APPC възлите валидните опции са SAME, PROGRAM и NONE. За TCP/IP възли SECURITY SOCKS е опция, която определя, че възелът ще е активен за SOCKS, като в този случай променливите от обкръжението SOCKS\_NS и SOCKS\_SERVER са задължителни и трябва да са настроени да разрешават SOCKS. За допълнителна информация вижте Глава 10, “Защита” на страница 95 и се обърнете към *Справочник на командите*.

#### **TCP/IP име на отдалечен хост или IP адрес**

При дефиниране на TCP/IP възел или името на отдалечения TCP/IP хост, или отдалечен TCP/IP адрес. Ако е определено име на хост, тогава трябва да е получено на DB2 Connect работната станция чрез сървъра на имена на области (DNS – Domain Name Server), или чрез запис в локалния файл на TCP/IP хост.

При отдалечен хост на DB2 за OS/390 името се появява в съобщението DSNL004I (DOMAIN=име на хост), когато се стартира помощното средство за разпределени данни (DDF – Distributed Data Facility).

#### **Име на TCP/IP услуга или номер на порт**

При определяне на TCP/IP възел – име на отдалечена TCP/IP услуга или номер на порт. Трябва да се определи за TCP/IP на отдалечения хост. Номерът на порт 446 е регистриран като номер на порт по подразбиране за DRDA.

При отдалечен хост на DB2 за OS/390 номерът на порта се определя в Boot Strap Data Set (BSDS) като PORT и освен това се намира в съобщението DSNL004I (TCPSPORT=номер на порт) при стартиране на помощното средство за разпределени данни (DDF).

**Забележка:** Сървърът присвоява втория порт, използван при операции с двуфазов протокол за записване на промените и синхронизиране през TCP/IP свързвания. Например DB2 Universal Database за OS/390 bootstrap dataset присвоява номер на порт (RESPORT), който да се използва за повторно синхронизиране само на входящите свързвания към DB2 Universal Database за OS/390. В този случай не е необходимо да се дефинира име на услуга.

## DCS Директория

Можете да определите следната информация в DCS директорията:

### Име на базата данни

Дефиниран от потребителя прякор за хоста или AS/400 сървър на базата данни. Използвайте едно и също име на база данни в двете таблици – с параметрите на DCS директорията и с параметрите на системната директория на базата данни.

Формат: 1–8 еднобитови буквено–цифрови символи, включително числен знак (#), знака at (@), знака за долар (\$), и долно тире (\_). Не може да започва с долно тире или число.

### Име на базата данни приемник

Базата данни на хоста или AS/400 сървър на база данни, както следва:

#### MVS/ESA

DB2 Universal Database за OS/390 подсистема, идентифицирана от своето ИМЕ НА МЯСТО.

ИМЕТО НА МЯСТОТО може да се определи след влизане в TSO и генериране на следното SQL запитване с помощта на някое от достъпните средства:

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

ИМЕТО НА МЯСТОТО освен това се дефинира в MVS/ESA Boot Strap Data Set (BSDS) и се съдържа в съобщението DSNL004I (LOCATION=място), което се записва при стартирането на Distributed Data Facility (DDF).

#### OS/390

DB2 Universal Database за OS/390 подсистема, идентифицирана от своето ИМЕ НА МЯСТО.

ИМЕТО НА МЯСТОТО може да се определи след влизане в TSO и генериране на следното SQL запитване с помощта на някое от достъпните средства:

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

ИМЕТО НА МЯСТОТО освен това се дефинира в Boot Strap Data Set (BSDS) и се съдържа в съобщението DSNL004I (LOCATION=място), което се записва при стартирането на Distributed Data Facility (DDF).

#### VSE или VM

Името на базата данни (DBNAME)

#### OS/400

Името на релационна база данни (RDBNAME)

**Други** За OS/2, Windows NT, Windows 2000 и UNIX–базирани системи псевдонимът на базата данни, който се намира в директорията на базата данни.

### Име на средство за обработка на запитвания

Името на средството за обработка на запитвания, което препраща SQL заявките към DRDA сървър на приложения. Средството за обработка на запитвания обработва заявките от името на приложната програма.

Формат: AR <име\_на\_рикуестър\_на\_приложения>

По подразбиране е DB2 Connect средството за обработка на запитвания.

### Параметричен низ

Ако искате да промените настройките по подразбиране, определете следните параметри в посочения ред. Параметричният низ не може да се определи с помощта на Асистента за конфигуриране на клиенти, а като се използва CLP, параметричният низ трябва да е обграден от единични кавички (например на OS/2 или Windows NT) или двойни кавички (например на AIX):

#### *файл–карта*

Името на файла със SQLCODE съответствията, който заменя SQLCODE съответствията по подразбиране. За да изключите SQLCODE съответствията, определете **NOMAP**. За повече информация вижте Глава 11, “Преобразуване на SQLCODE” на страница 103.

**,D** Това е втория позиционен параметър. Ако е определен, приложението ще прекъсне връзката към базата данни на хоста или на AS/400 сървъра на база данни, когато се върне един от следните SQLCODE кодове:

SQL3000N  
SQL30040N  
SQL30050N  
SQL30051N  
SQL30053N  
SQL30060N  
SQL30070N  
SQL30071N  
SQL30072N  
SQL30073N  
SQL30074N  
SQL30090N

Когато не е определен параметър за прекъсване на връзката **,D** свързването ще се прекъсне само когато се върнат следните SQLCODE кодове:

SQL30020N  
SQL30021N  
SQL30041N  
SQL30061N  
SQL30081N

Обяснения на тези кодове потърсете в *Справочник на съобщенията*.

**Забележка:** Ако поради грешка DB2 Connect прекъсне връзката, автоматично се изпълнява ролбек.

### „INTERRUPT\_ENABLED

Това е третият позиционен параметър. Ако INTERRUPT\_ENABLED е конфигуриран в DCS директория на DB2 Connect работна станция и приложение на клиент генерира прекъсване, докато е свързано към хоста или AS/400 сървъра на база данни, DB2 Connect ще изпълни прекъсване, като премахне свързването и изпълни ролбек за единицата работа. Такова прекъсване се поддържа на AIX, OS/2, Windows NT и Windows 2000.

Приложението ще получи sqlcode (-30081), който показва, че е прекъсната връзката към сървъра. Приложението трябва след това да установи нова връзка към хоста или AS/400 сървъра на база данни, за да обработи останалите заявки. На платформи, различни от AIX V4.1 и следващи, SNA Server V3.1 и следващи, OS/2, Windows NT и Windows 2000, DB2 Connect не поддържа възможността за автоматично прекъсване на връзка, когато приложение, което я използва, получи заявка за прекъсване.

**Забележка:** Тази поддръжка работи на TCP/IP свързвания на всички платформи. Клиентът може да прекъсне връзката, но – в зависимост от реализацията на сървъра – може да има или неполучени неща. DB2 Universal Database за OS/390 използва асинхронни сокет повиквания и следователно може да открие загубата на връзката и да върне обратно всички дълготрайни SQL изрази, които са активни в момента.

#### ,,,,,SYSPLEX

Този параметър, шестият позиционен параметър, може да се използва, за да се активира явно SYSPLEX поддръжката на DB2 Connect за определена база данни.

Въведена е нова променлива на профила (обкръжението или регистъра), наречена DB2SYSPLEX\_SERVER, която може да се използва за деактивиране поддръжката на SYSPLEX на ниво работна станция.

#### ,,,,,LOCALDATE=<стойност>"

Този параметър, седмият позиционен параметър, се използва за активиране поддръжка на формат за датата от DB2 Connect. Това се реализира с помощта на маска за дата за <стойност> както следва:

Да предположим, че подадете следните оператори на процесора за обработка на команди (CLP):

```
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
```

Псевдонимът на базата данни *newyork1* се използва за достъп до хост база данни без трансформиране на датата, тъй като не е определена маска за датата.

Въпреки това с новия тип поддръжка за форматиране на датата можете да използвате следните CLP команди. Тъй като в този случай се използва CLP и самият параметричен низ се определя с двойни кавички, стойността LOCALDATE трябва да се определи вътре в две двойки двойни кавички. Отбележете използването на специалния символ на операционната система "\" (наклонена черта), за да сте сигурни, че двойните кавички няма да се пропуснат от спецификацията на LOCALDATE. Вижте също "Определяне на параметричния низ" на страница 70.

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

Псевдонимът на базата данни "newyork2" ви дава достъп до същата хост база данни, но освен това има определена маска за формат на датата. Този пример показва, че маската за формата на датата се определя с помощта на ключовата дума LOCALDATE и е седмият позиционен параметър в полето PARMS на запис в DCS директорията.

За да бъде валидна маската на датите, ВСИЧКИ следващи изисквания трябва да бъдат изпълнени:

1. Може да има най-много една последователност от Y, M и D, където Y е цифра на годината, M е цифра на месеца, а D – цифра на деня.
2. Максималният брой на Y подред е 4.
3. Максималният брой на M подред е 2.
4. Максималният брой на D подред е 2.

Например, следните са валидни маски на дати:

```
"YyуMmDd" - Y, M и D не са
чувствителни към големи/малки букви.
"MM+DD+YYYY" - може да има маска,
по-дълга от 10 байта и да има символи,
различни от Y, M и D в маската
"abcYY+MM" - може да няма последователност от D
```

Следните са невалидни маски на дати:

```
"YYYYуMMDD" - невалидна,
защото има 5 Y.
"YYYYMDDM" - невалидна,
защото има две M.
```

Ако форматът на маската на датите е невалидна, няма да бъде отчетена грешка. Тя ще бъде игнорирана. Само защото маска на датите е валидна, това не значи, че ще бъде използвана. Ще бъде извършена трансформация на данните, базирана на валидна маска на данните, само ако са изпълнени ВСИЧКИ следващи условия:

1. Няма SQL грешка.
2. Стойността е дата във формат, подобен на ISO (ISO и LIS).
3. Размерът на изходните данни е поне 10 байта. Това е минималният размер на изходните данни, за да се запише стойността, даже и да не бъде извършена трансформация на формата на датата. Това изискване се отнася даже, ако маската на формата на датите е по-къса от 10 байта.
4. Има валидна маска за формата на датите, указана в запис на DCS директорията, и тази маска се вметва в областта на изходните данни.

,,,,,,CHGPPWD\_SDN=<име>

Този параметър е осмият позиционен параметър и се използва за определяне на име на символно предназначение, което да се използва при управлението на

изтичане на срока на паролите (PEM – Password Expiration Management). От значение е използването на главни и малки букви при определяне на <име>.

“Променяне на MVS парола” на страница 99 показва пример за каталогизиране на dcs директория на база данни с помощта на CHGPGW\_SDN, както следва:

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

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

Този параметър е деветият позиционен параметър и се използва за определяне на двупосочен (BiDi –Bidirectional) CCSID, който да се използва за заменяне на стандартния за сървъра на базата данни BiDi CCSID. Например:

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

където xyz представлява заменянето на CCSID (вижте 1 на страница 70).

Списък с поддържаните BiDi CCSID заедно с техните типове низове ще намерите в *Ръководство за администриране*.

Изискват се следните BiDi атрибути за правилното управление на двупосочни данни на различни платформи:

- Форма на числата (ARABIC c/y HINDI)
- Ориентация (RIGHT-TO-LEFT c/y LEFT-TO-RIGHT)
- Форма (SHAPED c/y UNSHAPED)
- Симетрична размяна (YES или NO)
- Тип текст (LOGICAL c/y VISUAL)

Тъй като подразбиращите се стойности не са едни и същи на различните платформи, възникват проблеми, когато DB2 данни се изпращат от една платформа на друга. Например, Windows платформите използват данни LOGICAL UNSHAPED, докато данните на MVS и OS/390 обикновено са във формат SHAPED VISUAL. Следователно без никаква поддръжка за BiDi атрибути ще се представят неправилно данните, изпратени от DB2 за MVS или OS/390 на DB2 Connect за Windows.

Когато се обменят данни между DB2 Connect и база данни на сървър, обикновено получателят изпълнява конвертирането на входящите данни. Същото правило обикновено се прилага и при трансформация на BiDi форматиране, което е допълнително спрямо обикновеното конвертиране на кодова страница. Въпреки това засега няма DB2 продукти за хост, който да поддържа трансформации на специфични двупосочни CCSID или двупосочен формат. Следователно DB2 Connect е усъвършенстван с допълнителна възможност за изпълнение на трансформации на двупосочен формат върху данни, които ще се изпратят на сървър на база данни в допълнение към данните, получени от сървъра на базата данни.

За да може DB2 Connect да изпълни трансформация на двупосочен формат върху изходящи данни към сървър на база данни, BiDi CCSID на сървъра на базата данни ще трябва да се замени (вижте 2 на страница 70). Това се осъществява чрез използването на параметъра BIDI в полето PARMS на запис в DCS директорията за сървъра на базата данни.

Използването на тази възможност може да се илюстрира най-добре с пример.

Да разгледаме DB2 клиент с език иврит, който използва CCSID 62213 (BiDi низ тип 5) и би искал да осъществи достъп до DB2 хост база данни, която използва CCSID 424 (BiDi низ тип 4). Обаче знаете, че данните в DB2 хост базата данни са базирани на CCSID 8616 (BiDi низ тип 6).

В тази ситуация има два проблема. Първият е, че DB2 хост базата данни не знае разликата между типовете BiDi низове със CCSIDs 424 и 8616. Вторият проблем е, че DB2 хост базата данни не разпознава CCSID с номер 62213 на DB2 клиента. Поддържа само CCSID 862, който е базиран на същата кодова таблица, като CCSID 62213.

Най-напред ще трябва да се уверите, че данните, изпратени към DB2 хост базата данни, са във формат с BiDi низ тип 6 и освен това да уведомите DB2 Connect, че ще трябва да изпълни трансформация на BiDi формат върху данните, които получи от DB2 хост базата данни. Ще използвате следното каталогизиране за DB2 хост базата данни:

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

Така съобщавате на DB2 Connect да промени CCSID на DB2 хост базата данни от 424 на 8616. Тази замяна включва следните обработки:

1. DB2 Connect ще се свърже към DB2 хост базата данни с помощта на CCSID 862.
2. DB2 Connect ще изпълни трансформация на двупосочно форматиране върху данните, които ще изпрати към DB2 хост базата данни от CCSID 62213 (BiDi низ тип 5) на CCSID 62221 (BiDi низ тип 6).
3. DB2 Connect ще изпълни трансформация на двупосочно форматиране на данните, които получи от DB2 хост базата данни от CCSID 8616 (BiDi низ тип 6) на CCSID 62213 (BiDi низ тип 5).

#### **Забележки:**

1. Променливата на обкръжението или регистърната стойност DB2BIDI трябва да се установи на YES, за да влезе в сила параметърът BiDi.
2. Ако искате DB2 Connect да изпълни трансформация на форматирането на данните, които ще изпрати към DB2 хост база данни, дори ако не е необходимо да се заменя нейния CCSID, отново ще трябва да добавите параметър BIDI към полето PARMs на DCS директорията. В този случай CCSID, който трябва да осигурите, ще е еднакъв със CCSID по подразбиране на DB2 хост базата данни.
3. В някои случаи в резултат от използването на двупосочен CCSID самото SQL запитване може да се промени така, че да не се разпознае от DB2 сървър. Особено трябва да избягвате използването на CCSID от тип IMPLICIT CONTEXTUAL и IMPLICIT RIGHT-TO-LEFT, когато може да се използва различен тип низ. Използването на CCSID от тип CONTEXTUAL може да доведе до непредсказуеми резултати, ако SQL запитването съдържа низове в кавички. Избягвайте използването на низове в кавички в SQL изрази и когато е възможно вместо тях използвайте хост променливи.

Ако определен двупосочен CCSID причинява проблеми, които не могат да се отстранят, като изпълните следващите препоръки, тогава трябва да установите променливата от обкръжението или регистърната стойност на NO.

## **Определяне на параметричния низ**

Ето примери за някои параметрични низове, които можете да определите.

Например можете да определите някой от следните, като символът "\" (наклонена черта) е специален символ escape на операционната система:



На AIX:

```
NOMAP
/u/username/sql1lib/map/dcs1new.map,D
,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE="\YYMMDD"\",,
```

За OS/2, Windows NT или Windows 2000.

```
NOMAP
d:\sql1lib\map\dcs1new.map,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE="\YYMMDD"\",,
```

В противен случай можете да приемете стойностите по подразбиране, като не определите параметричния низ.

**Забележка:** Поради необходимостта да въведете две двойки двойни кавички, когато определяте маската LOCALDATE в параметричния низ, трябва да използвате символа escape на операционната система, който е "\" (наклонена черта), например:

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

В резултат се получава следния запис в DCS директорията:

DCS 1 запис:

Име на локална база данни	= X
Име на базата данни приемник	= Y
Име на риквестъра за приложения	=
DCS параметри	= ,,,,,,LOCALDATE="YYMMDD"
Коментар	=
Версия на DCS директорията	= 0x0100

## Системна директория на база данни

Можете да определите следната информация в системната директория на базата данни:

### Име на базата данни

Същата стойност, която сте записали в таблицата с параметрите на DCS директорията.

### Псевдоним на базата данни

Псевдоним за хоста или AS/400 сървъра на базата данни. Това име ще се използва от приложните програми при достъп до базата данни. По подразбиране се използва стойността, която определите за име на базата данни.

Формат: 1–8 еднобайтови буквено–цифрови символи, включително числен знак (#), знака at (@), знака за долар (\$), и долно тире (\_). Не може да започва с долно тире или число.

### Име на възел

Същата стойност, която сте записали в таблицата с параметрите на директорията на възлите.

### Разпознаване

Определя къде ще се провери валидността на името и паролата на потребителя. Валидни опции са: SERVER, SERVER\_ENCRYPT, CLIENT, DCE, DCS и DCS\_ENCRYPT. За допълнителна информация вижте Глава 10, “Защита” на страница 95.

## Задаване на множество записи

За всяка база данни трябва да определите поне един запис във всяка от трите директории (директория на възлите, DCS директория, системна директория). Понякога може да е необходимо да дефинирате повече от един запис за базата данни.

Например може да предпочитате да изключите преобразуването на SQLCODE за приложенията, които се прехвърлят от хоста или AS/400 сървъра на базата данни, но да приемете преобразуването по подразбиране, което е разработено за обкръжението клиент/сървър. Можете да направите това, както следва:

- Дефинирайте един запис в директорията на възлите.
- Дефинирайте два записа в DCS директорията с различни имена на база данни. За единия запис въведете NOMAP в параметричния низ.
- Дефинирайте два записа в системната директория с различни псевдоними на база данни и двете имена на база данни, които сте определили в DCS директорията.

И двата псевдонима осъществяват достъп до една и съща база данни, като при единия има преобразуване на SQLCODE, а при другия няма.

---

## Обновяване на директориите

Можете да използвате командата CATALOG на всяка DB2 Connect система, а на OS/2 и Windows 32-битова операционна система можете да се възползвате от Помощника за добавяне на база данни на CCA. Ако имате Клиент за разработка на DB2 приложения, можете също така да създадете приложна програма, която да въвежда записите в каталога. За информация за API се обърнете към *Административен API Справочник* и *Справочник на командите*.

**Забележка:** За да каталогизирате база данни, трябва да имате права на *sysadm* или *sysctrl*.

За да обновите директориите с помощта на процесор за обработка на команди, направете следното:

1. С една от следните команди обновете директорията на възлите:

- За възел с APPC свързване:  
db2 CATALOG APPC NODE *име на възел*  
REMOTE *име на символно разположение* security *тип на защита*

Например:

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

- При база данни DB2 Universal Database за OS/390 версия 5.1 или DB2 Universal Database за AS/400 версия 4.2 с TCP/IP свързване:

```
db2 CATALOG TCP/IP NODE име на възел  
REMOTE име на хост или IP адрес  
SERVER име на услуга или номер на порт  
SECURITY тип на защита
```

Например:

```
db2 CATALOG TCP/IP NODE MVSIPNOD REMOTE MVSHOST SERVER DB2INSTC
```

Номерът на DRDA порт по подразбиране при TCP/IP свързване е 446.

2. Използвайте следната команда, за да обновите DCS директорията:

```
db2 CATALOG DCS DATABASE име_на_база_данни AS име_на_база_данни_приемник  
[AR рикуестър_на_приложение]  
[PARMS "параметричен низ"]
```

Например:

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

Или за OS/2, Windows NT или Windows 2000:

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

Или за AIX:

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

**Забележка:** Вижте “Определяне на параметричния низ” на страница 70 за информация как да използвате символа escape на операционната система, когато определяте маската LOCALDATE в параметричния низ.

3. Използвайте следната команда, за да обновите системната директория:

```
db2 CATALOG DATABASE име_на_база_данни  
AS псевдоним AT NODE име_на_възел  
AUTHENTICATION тип_на_автентичност
```

Например:

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

Ако имате отдалечени клиенти, трябва също така да обновите директориите на всеки отдалечен клиент. За информация се обърнете към съответната книга *DB2 Connect: Бърз старт book*.



---

## Глава 7. Свързване на приложения и помощни програми

Приложните програми, разработени с вграден SQL, трябва да се свържат с всяка база данни, с която ще работят. На платформите, където тези функции са достъпни, можете да направите това с Командния център и с Помощника за конфигуриране на клиенти.

Свързването трябва да се изпълни еднократно за всяко приложение и за всяка база данни. По време на свързването плановете за достъп до базата данни се съхраняват за всеки SQL оператор, който ще се изпълни. Те се доставят от разработчиците на приложението и се съдържат във *файлове за свързване*, които се създават по време на предкомпилирането. Свързването просто е процес на обработка на тези файлове за свързване от хоста или AS/400 сървър на базата данни. За допълнителна информация за свързване на приложения се обърнете към *Ръководство за разработка на приложения*.

Тъй като редица помощни програми, доставени с DB2 Connect, са разработени с помощта на вграден SQL, те трябва да се свържат към хоста или AS/400 сървър, преди да ги използвате с тази система. Ако не използвате DB2 Connect помощните програми и интерфейси, изброени в Таблица 4 на страница 77, не е необходимо да ги свързвате с всеки хост или AS/400 сървър на база данни. Списъците с файловете за свързване, необходими за тези помощни програми, се съдържат в следните файлове:

**ddcsmvs.lst**

За MVS или OS/390

**ddcsvse.lst**

За VSE

**ddcsvm.lst**

За VM

**ddcs400.lst**

За OS/400

При свързването на един от тези списъци с файлове с база данни ще се свърже всяка отделна помощна програма с тази база данни.

Ако е инсталиран DB2 Connect Enterprise Edition, DB2 Connect помощните програми трябва да се свържат с всеки хост или AS/400 сървър на база данни; еднократно от всеки тип платформа на клиент, преди да може да се използват с тази система.

Например, ако имате 10 OS/2 клиенти, 10 Windows клиенти и 10 AIX клиенти, които се свързват към DB2 Universal Database за OS/390 чрез DB2 Connect Enterprise Edition за Window NT сървър, направете следното:

1. Свържете ddcsmvs.lst от един от Windows клиентите.
2. Свържете ddcsmvs.lst от един от OS/2 клиентите.
3. Свържете ddcsmvs.lst от един от AIX клиентите.
4. Свържете ddcsmvs.lst от DB2 Connect сървъра.

**Забележка:** Тук се приема, че всички клиенти имат едно и също ниво на услуги. Ако това не е така, допълнително може да се наложи да свържете от всеки клиент с определено ниво на услуги. Обърнете се към Приложение Е, “Средства за свързване на клиенти от предишни версии” на страница 183, ако имате клиенти преди DB2 версия 2.1.

Освен DB2 Connect помощните програми, всички други приложения, които използват вграден SQL, трябва също да се свържат с всяка база данни, с която искате да работят. При изпълнението на приложение, което не е свързано, обикновено се генерира съобщение за грешка SQL0805N. Вероятно ще искате да създадете допълнителен файл със списък за свързване за всички приложения, които трябва да се свържат.

За всеки хост или AS/400 сървър на база данни, с който се свързвате, направете следното:

1. Проверете дали имате достатъчно права за системата за управление на вашия хост или AS/400 сървър на база данни:

#### **MVS или OS/390**

Необходимата оторизация е:

- SYSADM или
- SYSCTRL или
- BINDADD и CREATE IN COLLECTION NULLID

**Забележка:** Правата на достъп BINDADD и CREATE IN COLLECTION NULLID са достатъчни **само** когато пакетите все още не съществуват. Например, ако ги създавате за първи път.

Ако пакетите вече съществуват и ги свързвате отново, тогава необходимите права на достъп за изпълнението на задачата(ите) зависи от това кой е изпълнил първоначалното свързване.

**A** Ако вие сте извършили първоначалното свързване и искате да го изпълните отново, тогава с един от горните типове права ще можете да направите свързването.

**B** Ако първоначалното свързване е изпълнено от някой друг и искате да извършите второ свързване, тогава ще са ви необходими права за управление SYSADM или SYSCTRL. Ако имате само BINDADD и CREATE IN COLLECTION NULLID, няма да можете да изпълните свързване. Все пак е възможно да създадете пакет, ако нямате правата за управление SYSADM или SYSCTRL. В този случай ще ви е необходимо право на достъп BIND за всеки от съществуващите пакети, които смятате да замените.

#### **VSE или VM**

Оторизацията изисква DBA права. Ако искате да използвате опцията GRANT на командата bind (за да избегнете предоставянето на права за достъп отделно за всеки DB2 Connect пакет), потребителският идентификатор NULLID трябва да има право за предоставяне на права на други потребители в следните таблици:

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

- system.syscolauth

На VSE или VM система можете да изпълните:

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

#### OS/400

\*CHANGE право или по-високо на NULLID колекцията.

2. Генерирайте команди, подобни на следните:

```
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
```

Където *DBALIAS*, *USERID* и *PASSWORD* се прилагат за хоста или AS/400 сървъра на базата данни, *ddcsmvs.lst* е файла със списъка за свързване за MVS, а *path* представлява пътеката до файла със списъка за свързване.

Например *drive:\sqllib\bnd\* се отнася за всички Intel операционни системи, а *INSTHOME/sqllib/bnd/* за всички UNIX операционни системи, където *drive* представлява логическото устройство, където е инсталирана DB2 Connect, а *INSTHOME* е собствената директория на DB2 Connect потребителския модел.

Можете да използвате опцията за предоставяне на права на командата **bind**, за да предоставите права EXECUTE на PUBLIC или на определен идентификатор на потребител или на група. Ако не използвате опцията за предоставяне на права на командата **bind**, трябва да изпълните отделно за всеки пакет GRANT EXECUTE (RUN).

За да разберете имената на пакетите за файловете за свързване, въведете следната команда:

```
ddcspkgn @bindfile.lst
```

Например:

```
ddcspkgn @ddcsmvs.lst
```

може да генерира следния резултат:

```
Файл за свързване                Име на пакет
-----
f:\sqllib\bnd\db2ajgrt.bnd      SQLAB6D3
```

За информация Таблица 4 показва файловете за свързване и имената на пакетите, които се използват от различните компоненти на DB2 Connect. В някои случаи на различните операционни системи се използват различни файлове за свързване и пакети.

Таблица 4 (Страница 1 от 2). Файлове за свързване и пакети						
Компонента	Файл за свързване	Пакет	MVS или OS/390	VSE	VM	OS/400
Свързващ (използван при опцията за свързване GRANT)	db2ajgrt.bnd	sqlabxxx	да	да	да	да
<b>DB2 интерфейс за вградени SQL оператори</b>						
Ниво на изолация CS	db2clics.bnd	sql11xxx	да	да	да	да
Ниво на изолация RR	db2clirr.bnd	sql12xxx	да	да	да	да
Ниво на изолация UR	db2cliur.bnd	sql13xxx	да	да	да	да

Таблица 4 (Страница 2 от 2). Файлове за свързване и пакети

Компонента	Файл за свързване	Пакет	MVS или OS/390	VSE	VM	OS/400
Ниво на изолация RS	db2clirs.bnd	sql14xxx	да	да	да	да
Ниво на изолация NC	db2clinc.bnd	sql15xxx	не	не	не	да
Използване на MVS имена на таблици	db2clims.bnd	sql17xxx	да	не	не	не
Използване на OS/400 имена на таблици (OS/400 3.1 или следваща)	db2clias.bnd	sql1axxx	не	не	не	да
Използване на VSE/VM имена на таблици	db2clivm.bnd	sql18xxx	не	да	да	не
<b>Процесор за обработка на команди</b>						
Ниво на изолация CS	db2clpcs.bnd	sqlc2xxx	да	да	да	да
Ниво на изолация RR	db2clpr.bnd	sqlc3xxx	да	да	да	да
Ниво на изолация UR	db2clpur.bnd	sqlc4xxx	да	да	да	да
Ниво на изолация RS	db2clprs.bnd	sqlc5xxx	да	да	да	да
Ниво на изолация NC	db2clpnc.bnd	sqlc6xxx	не	не	не	да
<b>REXX</b>						
Ниво на изолация CS	db2arxcs.bnd	sqla1xxx	да	да	да	да
Ниво на изолация RR	db2arxrr.bnd	sqla2xxx	да	да	да	да
Ниво на изолация UR	db2arxur.bnd	sqla3xxx	да	да	да	да
Ниво на изолация RS	db2arxrs.bnd	sqla4xxx	да	да	да	да
Ниво на изолация NC	db2arxnc.bnd	sqla5xxx	не	не	не	да
<b>Помощни програми</b>						
Експорт	db2uexpm.bnd	sqlubxxx	да	да	да	да
Импорт	db2uimpb.bnd	sqlufxxx	да	да	да	да

За да определите тези стойности за DB2 Connect, изпълнете помощната програма *ddcspkgn*, например:

```
ddcspkgn @ddcsmvs.lst
```

Допълнително тази помощна програма може да се използва, за да се определи името на пакета за отделните файлове за свързване, например:

```
ddcspkgn bindfile.bnd
```

Ако във вашата DB2 за MVS/ESA система е инсталирана корекцията за APAR PN60988 (или ако е след версия 3 подверсия 1), можете също така да добавите файловете за свързване към изолационното ниво NC към файла *ddcsmvs.lst*.

За допълнителна информация за възможностите за свързване се обърнете към *Справочник на командите*.

#### Забележки:

- Необходимо е използването на опцията за свързване `sqlerror continue`; тази опция автоматично се определя вместо вас, когато свързвате приложения с помощта на DB2 средствата или процесора за обработка на команди. Определянето на тази опция превръща грешките при свързването в предупреждения, така че може да се създаде пакет дори и при свързване на файл, в който има грешки. Така един файл за свързване може да се използва спрямо няколко сървъра, дори когато реализацията на



определения сървър може да посочи като невалиден SQL синтаксисът на друг сървър. Поради тази причина при свързването на файловете със списъци `ddcsxxx.lst` спрямо определен хост или AS/400 сървър на база данни би трябвало да се очаква, че ще се получат някои предупреждения. Например, когато свързвате спрямо DB2 за VM, може да се получат редица предупредителни съобщения, защото DB2 за VM не позволява курсорите да се декларират като "WITH HOLD".

- b. Ако се свързвате към DB2 Universal Database база данни чрез DB2 Connect, използвайте списъка за свързване `db2ubind.lst` и не определяйте `sqlerror continue`, защото е валидно само когато се свързвате към хост или AS/400 сървър на база данни. Освен това за свързване към DB2 Universal Database база данни ви препоръчваме да използвате DB2 клиентите, осигурени с DB2 , а не с DB2 Connect.
3. Използвайте подобни изрази, за да свържете всяко приложение или списък с приложения.
4. Ако имате отдалечени клиенти с предишна версия на DB2, може да се наложи да свържете помощните програми на тези клиенти към DB2 Connect. За допълнителна информация вижте Приложение Е, "Средства за свързване на клиенти от предишни версии" на страница 183.

---

## Командата BIND

DB2 командата **bind** свързва приложение към определена база данни. Ако в отделни операции изпълнявате предкомпилирането и свързването, тогава опциите, които определите при свързването, ще заменят опциите от стъпката на предкомпилирането.

В *Справочник на командите* е описан синтаксисът на командата BIND, който трябва да използвате, когато свързвате приложение към хост или AS/400 сървър на база данни чрез DB2 Connect. Трябва да се обърнете към специфичното за DRDA описание.

**Забележка:** Някои параметри на командата BIND може да не се поддържат от вашия хост или AS/400 сървър на база данни. За допълнителна информация се обърнете към документацията, доставена с RDBMS на вашия хост или AS/400 сървър на база данни.

---

## Повторно свързване

След като свържете приложението (и създадете пакета на хоста или AS/400 сървъра на база данни), може да откриете, че трябва да създадете отново пакета. Можете да направите това и без първоначалния файл за свързване, като използвате Процесор за обработка на команди командата **REBIND PACKAGE** или съответния API.

Предимствата от използването на тази команда са:

- Можете да се възползвате от промените в системата чрез повторно оптимизиране и като изградите новите раздели на пакета, без да използвате първоначалния файл за свързване.
- Можете да създадете повторно пакетите, които повече не работят или са невалидни.
- Можете да създадете повторно пакетите, които са станали невалидни вследствие миграцията.

- Можете да подобрите производителността, като използвате явното повторно свързване вместо неявното повторно свързване или само свързване.
- Можете да промените характеристиките. Например при DB2 Universal Database за OS/390 можете за нуждите на тестването или миграцията да промените квалификатор на неквалифицирана таблица.

Ако искате да промените програма, опциите за свързване или някаква друга информация за собственика, трябва да използвате командата **BIND**. Освен това, ако пакетът не съществува в базата данни, или ако искате да видите всички грешки при свързване (не само първата открита), трябва да използвате командата **BIND**.

За да изпълните тази команда, трябва да имате правата, които се изискват от вашия хост или AS/400 сървър на база данни. Ако не сте свързани към база данни, командата ще доведе до косвено свързване към базата данни по подразбиране (ако имате права за достъп).

Синтаксисът на процесор за обработка на команди командата е описан в *Справочник на командите*.

Въведете следната команда, за да разберете името на пакета за файла за свързване:  
**ddcspkgn bindfile.bnd**

---

## Глава 8. Системен монитор на базата данни

В тази глава са обобщени възможностите на DB2 Системния монитор за потребителите на DB2 Connect. Следните усъвършенствания на Системния монитор са осигурени в DB2 Connect Версия 7:

- Следене на моментно изображение. Моментното изображение на системата ви дава информация за специфичен момент от времето. Моментното изображение отразява текущото състояние на активност в мениджъра на базата данни за определен обект или група обекти. Има пет основни вида моментно изображение на DCS информацията, които са достъпни чрез монитора.
- Еквивалентен графичен потребителски интерфейс на CLP командата LIST DCS APPLICATIONS. Командата LIST осигурява по-сбито отчитане на състоянието на системата, отколкото моментното изображение. Графичните възможности на LIST се осигуряват чрез DB2 Командния център. Вижте “LIST DCS APPLICATIONS EXTENDED” на страница 88

За информация за системен монитор за базата данни се обърнете към *Ръководство и справочник за системния монитор*.

---

### Следене на свързванията от отдалечени клиенти

Можете да използвате Системния монитор на базата данни с DB2 Connect Enterprise Edition, за да следите свързванията на отдалечените клиенти. За наблюдение на клиенти, които са локални на DB2 Connect сървъра или работят на самия сървър, ще трябва да определите следната променлива на обкръжението:

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

Например, когато възникне грешка на хоста или AS/400 системата, системният администратор може да определи дали проблемът е бил на DB2 Connect работната станция. Системният монитор на базата данни е свързан с:

- DRDA корелационен токен (CRRTKN), за незащитена сесия.
- Идентификатор за логическа единица работа (LUWID), за двуфазова сесия, защитена от SNA Syncpoint Manager (SPM).
- Идентификатор на единица работа (UOWID), за двуфазови свързвания, защитени от DRDA-3 Syncpoint Manager (използван при TCP/IP свързване).
- Идентификатор на DB2 Connect свързване (Идентификаторът на приложението).

Тази информация показва коя DB2 Connect връзка е причинила проблема, като позволява на системния администратор да изключи отделното клиентско приложение от системата, без да повлияе на другите клиенти, които използват DB2 Connect свързване.

### Включване на параметрите за наблюдение за DB2 Connect

Системният монитор винаги е активен. Независимо от това, ако искате да получите повече подробности в резултата от GET SNAPSHOT, трябва да включите съответните параметри за наблюдение. Параметрите за наблюдение, които се отнасят за DB2 Connect са STATEMENT (за информацията от ниво оператор) и UOW (за информацията от ниво транзакция).

За да промените параметрите за наблюдение, използвайте командата **db2 update monitor switches**. Обърнете се към *Справочник на командите* за синтаксиса на тази команда. Следва пример, който създава статистика за единиците работа (UOW) с помощта на DB2 Системен монитор:

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

## Представяне на състоянието на параметрите за наблюдение

За да представите състоянието на параметрите за наблюдение, използвайте командата **db2 get monitor switches**.

---

## Използване на командите GET SNAPSHOT

DB2 Мониторът поддържа активен списък с ценна системна информация. Можете да получите обобщена информация за състоянието на системата, като изпълните командата GET SNAPSHOT. Можете да получите моментно изображение, ако имате права на SYSMAINT, SYSCTRL или SYSADM за мениджър на базата данни потребителския модел, който искате да наблюдавате.

Пет от командите за получаване на моментно изображение са удобни за извличане на DCS информация. Те са:

- 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

В резултат от всяка команда за моментно изображение се получава подробен отчет за съответната област.

Например при изпълнение на командата GET SNAPSHOT FOR DCS DATABASE ON DCSDB ще се получи следния отчет:

Моментно изображение на DCS база данни

Име на DCS база данни	= DCSDB
Име на хост базата данни	= GILROY
Времеви индикатор на първо свързване с БД	= 12-15-1999 10:28:24.596495
Последно изминало време за свързване	= 0.950561
Последна продължителност на свързване	= 0.000000
Време за отговор на хоста (sec.ms)	= 0.000000
Час на последното рестартиране	=
Брой на опитаните SQL оператори	= 2
Брой на опитаните оператори за комит	= 1
Брой на опитаните оператори за ролбек	= 0
Операции с неуспешно изпълнени оператори	= 0
Общ брой на шлюзовите връзки	= 1
Текущ брой на шлюзовите връзки	= 1
Шлюзови свързвания, чакащи отговор от хост	= 0
Шлюзови свързвания, чакащи заявка от клиент	= 1
Шлюзови комуникационни грешки към хост	= 0
Час на последната комуникационна грешка	= Няма
Най-големият брой страници за шлюз връзки	= 1
Избрани редове	= 0
Изпратени външни байтове	= 140
Получени външни байтове	= 103

Този отчет съдържа информация за свързванията към базата данни, производителността, грешките и изпълнението на SQL запитванията. На практика моментните изображения от DB2 Монитор може да са много по-подробни. Например, ако изпълните командата `GET SNAPSHOT FOR ALL DCS APPLICATIONS`, ще получите отчет, подобен на показания:

Моментно изображение на DCS приложение

ID на клиентско приложение	= 09150F74.B6A4.991215152824
Номер на последователност	= 0001
ID на оторизация	= SMITH
Име на приложение	= db2bp
Указател на приложение	= 1
Състояние на приложение	= изчаква заявка
Час на смяна на състоянието	= 12-15-1999 10:29:06.707086
Възел на клиент	= sys143
Версия на клиента	= SQL06010
Платформа на клиента	= AIX
Протокол на клиента	= TCP/IP
Кодова страница на клиента	= 850
ID на процеса на клиентското приложение	= 49074
ID за включване на клиента	= smith
ID на хост приложението	= G9150F74.B6A5.991215152825
Номер на последователност	= 0000
Псевдоним на база данни за шлюза	= MVSDB
Име на DCS база данни	= DCSDB
Име на хост база данни	= GILROY
Версия на хоста	= DSN05012
Хост CCSID	= 500
Изходен комуникационен адрес	= 9.21.21.92 5021
Изходен комуникационен протокол	= TCP/IP
Входен комуникационен адрес	= 9.21.15.116 46756
Час за първо свързване с базата данни	= 12-15-1999 10:28:24.596495
Време за отговор на хоста (sec.ms)	= 0.000000
Време за обработка от шлюза	= 0.000000
Час на последното рестартиране	=
Избрани редове	= 0
Брой на опитаните SQL оператори	= 2
Операции с неуспешно изпълнени оператори	= 0
Оператори за комит	= 1
Оператори за ролбек	= 0
Получени вътрешни байтове	= 404
Изпратени външни байтове	= 140
Получени външни байтове	= 103
Изпратени вътрешни байтове	= 287
Брой на отворените указатели	= 0
Продължителност на престой на приложение	= 1 минута и 32 секунди
Състояние за приключване на UOW	=
Час на предишна приключена UOW	= 12-15-1999 10:28:25.592631
Начален час на UOW	= 12-15-1999 10:29:06.142790
Час на спиране на UOW	=
Изминало време от последната uow (sec.ms)	= 0.034396
Последна изпълнена операция	= Execute Immediate
Начален час на последната операция	= 12-15-1999 10:29:06.142790
Час на спиране на последната операция	= 12-15-1999 10:29:06.707053
Оператор	= Execute Immediate
Номер на раздел	= 203
Създател на приложение	= NULLID
Име на пакет	= SQLC2C07
SQL compiler cost estimate in timerons	= 0
SQL compiler cardinality estimate	= 0
Начален час на оператор	= 12-15-1999 10:29:06.142790
Час на спиране на оператор	= 12-15-1999 10:29:06.707053
Време за отговор на хост (sec.ms)	= 1.101612
Време от последен изпълнен оператор(sec.ms)	= 0.564263
Извлечени редове	= 0

```

Време за обработка от шлюза           = 0.013367
Вътрешни байтове, получени за оператора = 220
Външни байтове, изпратени за оператора  = 130
Външни байтове, получени за оператора  = 49
Вътрешни байтове, изпратени за оператора = 27
Текст на SQL оператор:
create table t12 (col1 int, col2 char)

```

Допълнителна информация за командата GET SNAPSHOT и други полезни команди на DB2 МОнитор потърсете в *Ръководство и справочник за системния монитор*.

---

## Представяне състоянието на DCS приложение

От DB2 Connect версия 5.2 Системният монитор осигурява три разновидности на командата LIST DCS APPLICATIONS, както следва:

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

### LIST DCS APPLICATIONS

За да прегледате информацията, осигурена от Монитора на ниво приложение, изпълнете командата DB2 LIST DCS APPLICATIONS. Тя връща следната информация за APPC свързване (DB2 Connect Enterprise Edition Версия 7 и DB2 Universal Database за OS/390):

ID за оториз.	Име на приложение	Указател на прилож.	ID на хост приложение
USERID	db2bp_41	0	CAIBMOML.0MXT4H0A.A79EAA3C6E29

Връща следната информация за TCP/IP свързване (DB2 Connect Enterprise Edition Версия 7 и DB2 Universal Database за OS/390):

ID за оториз.	Име на приложение	Указател на прилож.	ID на хост приложение
USERID	db2bp_41	2	0915155C.9704.1517172201BE

#### Id за оторизация

Идентификаторът за оторизация, който е използван при влизане в хоста или AS/400 сървъра на базата данни. Определя кой изпълнява приложението.

#### Име на приложение

Името на приложението, което се изпълнява на клиента, както е известно на DB2 Connect. Достъпни са само първите 20 байта след последния разделител в пътеката. Името на приложението не е достъпно за приложения, които работят на DB2 за OS/2 версия 1.

#### Указател на приложение

Агентът, който се изпълнява на DB2 Connect работната станция. Можете да използвате този елемент, за да свържете системен монитор за базата данни информацията с друга диагностична информация. (Например вижте “Помощна програма за трасиране (ddcstrc)” на страница 143.) Идентификаторът на агента е необходим и когато използвате командата FORCE USERS или API.

### Идентификатор на хост приложение

Един от следните:

- DRDA корелационният токен (CRRTKN), за незащитена сесия.
- Идентификатор за логическа единица работа (LUWID), за двуфазова сесия, защитена от SNA Syncpoint Manager (SPM).
- Идентификатор на единица работа (UOWID), за двуфазови свързвания, защитени от DRDA-3 Syncpoint Manager (използван при TCP/IP свързване).

Този уникален идентификатор се генерира, когато приложението се свърже към хоста или AS/400 сървъра на базата данни. Можете да използвате този елемент заедно с идентификатора на приложението, за да свържете частите за клиента и сървъра от информацията за приложението.

## LIST DCS APPLICATIONS SHOW DETAIL

Ако е определен формат за командата DB2 LIST DCS APPLICATIONS SHOW DETAIL, се представя допълнителна информация, включително:

Идент.	Име на приложение	Указ. на прилож.	Идентификатор на клиентско приложение	No	Псевдоним клиент
NEWTON	db2bp	0	09151251.07D3.980925183850	0001	MVSDB
клиент възел	клиент версия	клиент код. стр.	Идент. на хост приложение	No	име на хост БД
antman	SQL05020	819	G9151251.G7D4.980925183851	0000	GILROY

Хост  
Версия  
-----  
DSN05011

Този отчет не е форматиран и поради тази причина за вас може да е по-полезен отчета "LIST DCS APPLICATIONS EXTENDED" на страница 88.

### Идентификатор на клиентско приложение

Идентифицира уникално приложението, свързано към DB2 Connect работната станция. Има различни формати за идентификатора на приложението, като те се определят от комуникационния протокол между клиента и DB2 Connect работната станция. За допълнителна информация за форматите се обърнете към *Ръководство за администриране*.

Тази стойност ви позволява да съпоставите свързванията от клиенти към DB2 Connect работната станция и от DB2 Connect работната станция към хоста или AS/400 сървъра на база данни.

### Номер последователност за клиент (Seq#)

Номерът на последователност за клиент е поредния номер на транзакцията. Използва се за съпоставяне на транзакция, разпространена в рамките на различни системи.

### Псевдоним на клиент в БД

Псевдонимът на базата данни, осигурен от приложението при свързването към нея. Този елемент може да се използва за определяне коя е действителната база данни, към която приложението е осъществило достъп. Съответствието между това име и името на базата данни може да се направи с помощта на директориите на базата данни на възела на клиента и на възела на мениджър на базата данни сървъра.



### **NNAME на клиент (възел)**

Идентифицира възела, на който се изпълнява клиентското приложение. Информацията зависи от използвания от клиента протокол. Например за клиент, свързан чрез NetBIOS, това е стойността на мениджър на базата данни конфигурационния параметър NNAME. За клиент, свързан чрез TCP/IP, това е името на хоста.

### **ID на клиентско приложение (клиент)**

Името и версията на продукта, който работи на клиента. Идентификаторът на продукта на клиента може да е:

- SQL01010 за версия 1 на DB2 за OS/2
- SQL01011 за версия 1 на UNIX–базирани DB2 продукти и Разрешители на приложения на клиенти.
- SQL02010 за версия 2 на DB2 продукти и Разрешители на приложения на клиенти.
- SQL02020 за версия 2.1.2 на DB2 продукти и Разрешители на приложения на клиенти.
- SQL05000 за версия 5.0 на DB2 Universal Database и DB2 Connect и техните клиенти.
- SQL05020 за версия 5.2 на DB2 Universal Database и DB2 Connect и техните клиенти.
- SQL06010 за версия 6.1 на DB2 Universal Database и DB2 Connect и техните клиенти.
- SQL07010 за Версия 7 на DB2 Universal Database и DB2 Connect и техните клиенти.

### **Идентификатор за кодова страница**

Идентификаторът за кодовата страница на възела, където наблюдаваното приложение е стартирано.

Можете да използвате тази информация, за да се уверите, че се поддържа конвертирането между кодовата страница на приложението и кодовата страница на базата данни (или за база данни на хост или AS/400 сървър на база данни, CCSID на хост или AS/400 сървър на база данни).

Ако кодовата страница на приложението е различна от кодовата страница, под която работи системен монитор за базата данни, този елемент ще ви помогне ръчно да конвертирате данните, които се предават от приложението и се представят от системен монитор за базата данни. Например може да ви помогне при превеждане името на приложението.

### **Э на изходяща последователност**

Представява номера на изходящата последователност. Използва се за съпоставяне на транзакциите на различни системи.

### **Име на хост базата данни**

Истинското име на базата данни, към която е свързано приложението. В DCS директорията това е *името на базата данни приемник*.

### **ID на хост програма**

Името и версията на продукта, който работи на сървъра. То е във вида *PPPVVRRM*, където:

- PPP** Идентифицира продукта на хоста или AS/400 сървъра на базата данни (например DSN за DB2 Universal Database за OS/390, ARI за DB2 за VSE и VM или QSQ за DB2 Universal Database за AS/400)
- VV** Представява двуцифрен номер на версия, като 01.
- RR** Представява двуцифрен номер на подверсия.

M           Представява едноцифрено ниво на модификация.

## LIST DCS APPLICATIONS EXTENDED

За да генерирате Разширен отчет, можете да използвате командата LIST DCS APPLICATIONS с опцията EXTENDED. Разширеният отчет съдържа всички полета, които се представят с опцията SHOW DETAIL на същата команда, както и девет нови полета:

- Статус на DCS приложение
- Час на промяна на статуса
- Платформа на клиент
- Протокол на клиент
- Идентификатор на кодиран набор от символи (CCSID).
- Идентификатор на клиент при влизане в системата
- Идентификатор на процеса на клиентско приложение
- Псевдоним на база данни на шлюза
- Име на DCS база данни

Докато съществуващите опции на команди представят полетата хоризонтално, по един ред на приложение, новата опция ги представя вертикално, по едно поле на ред.

Следва новият синтаксис на командата:

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

А тук е представен примерен резултат от тази команда, когато е използвана новата опция EXTENDED:

Списък на DCS приложения - Разширен отчет

ID на клиентско приложение	= 09151251.0AA7.981015204853
Номер на последователност	= 0001
ID на оторизация	= NEWTON
Име на приложение	= db2bp
Указател на приложение	= 1
Състояние на приложение	= изчаква заявка
Час за промяна на състоянието	= 10-15-1998 16:50:29.489160
Възел на клиент	= antman
Версия на клиента	= SQL05020
Платформа на клиента	= AIX
Протокол на клиента	= TCP/IP
Кодова страница на клиента	= 819
ID на процеса на клиентското приложение	= 39324
ID за включване на клиента	= smith
ID на хост приложението	= G9151251.GAA8.981015204854
Номер на последователност	= 0000
Псевдоним на база данни за шлюза	= MVSDB
Име на DCS база данни	= DCSDB
Име на хост база данни	= GILROY
Версия на хоста	= DSN05011
Хост CCSID	= 500

Полето за състоянието на приложението съдържа една от следните стойности:

1. изчакване на свързване - външно. Това означава, че е генерирана заявка за свързване към хост база данни и DB2 Connect изчаква да се установи връзката.
2. чакане на заявка. Това означава, че е установено свързване с хост база данни и DB2 Connect чака SQL оператор от приложението на клиента

3. чакане на отговор. Това означава, че на хост базата данни е изпратен SQL оператор.

Освен това датата и часа, когато е променено състоянието, се показва в отчета, ако по време на обработката е включен параметъра за наблюдение UOW на Системния монитор. В противен случай се представя съобщение, че не се събира тази информация.

## Използване на DB2 Центъра за управление за представяне на разширена информация за DCS приложения

Можете да използвате DB2 Версия 7 Центъра за управление, за да наблюдавате DB2 Connect шлюза. Този раздел показва как можете да използвате Центъра за управление, за да получите същия отчет, както при командата **list dcs applications extended**.

За да прегледате разширения отчет за някоя програма:

1. Представете разширено дървото под иконата **systems** на Центъра за управление, за да представите **Свързванията към шлюза на системните** → **потребителски модели** →. Ако щракнете с десен бутон на мишката върху потребителски модел в папката **Gateway Connections**, ще се появи контекстно меню. Изберете елемента **Приложения...** от това меню. Появява се прозорецът **Приложения**. Този прозорец има вид на тетрадка с етикети, като един от етикетите е **Приложения**. Ако в потребителския ви модел има приложения за шлюз, ще има втори етикет с име **Приложения за шлюз**.
2. Основният прозорец на всяка страница съдържа колони с информация, съответстващи на полетата в отчета LIST DCS APPLICATIONS EXTENDED. Първите шест колони, които се виждат в прозореца, осигуряват следните данни:

- Възел на клиент
- Име на приложението
- ID на клиентско приложение
- ID на хост приложение
- Псевдоним на базата данни на шлюза
- Състояние

Останалите полета от отчета могат да се видят, като преместите хоризонталния плъзгач за преглед в долната част на прозореца.

В този отчет ще намерите всички полета, които се получават от командата LIST DCS APPLICATIONS EXTENDED.

---

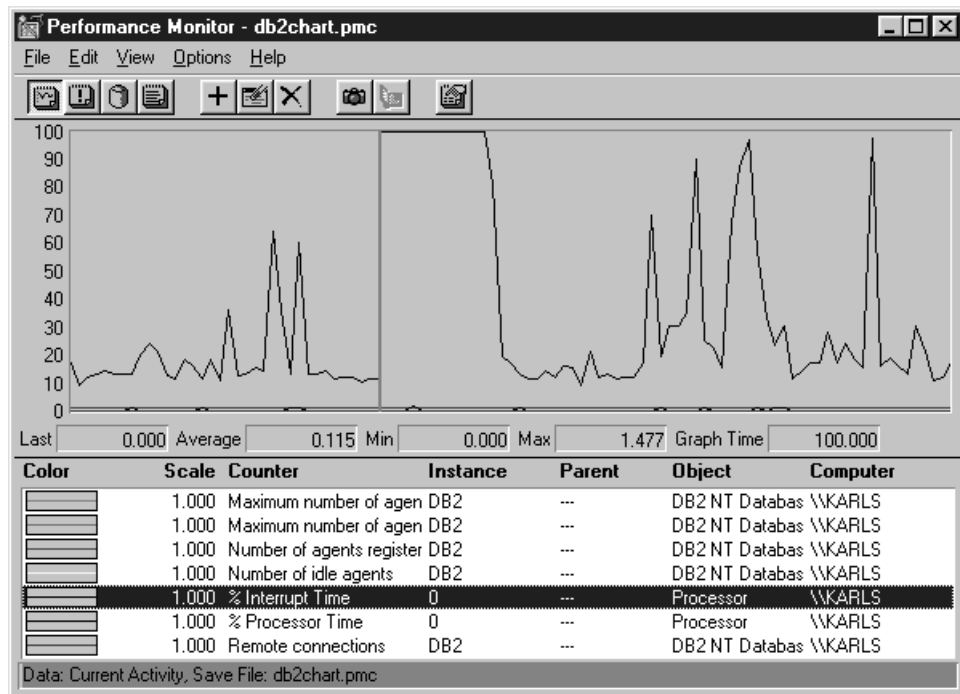
## Използване на Монитор на производителността на Windows

Windows NT и Windows 2000 осигуряват удобен инструмент за наблюдение на производителността на вашите DB2 приложения. Мониторът на производителността (Performance Monitor), който е един от административните инструменти на Windows, представя графично производителността на системата. Можете да изберете сред множество елементи на системата, базата данни или свързани с комуникациите, които да наблюдавате и да ги съпоставите в графичното представяне.

Например отчетите, получени чрез командите **GET SNAPSHOT FOR ALL DCS DATABASES** или **GET SNAPSHOT FOR ALL DCS APPLICATIONS** могат да се представят графично в реално време с помощта на Монитора и да се сравнят директно със стойности като използването на процесора. Можете директно да

сравните влиянието на различните параметри върху производителността на базата данни или комуникациите. Можете да запишете специализирани конфигурации от параметри във файловете PMC, откъдето по-късно ще можете да ги вземете.

Например в следващата фигура са представени графично редица DB2 показатели, като са съпоставени с използването на процесора. Наборът от представени стойности са записани във файла db2chart.pmc. Можете да запишете толкова PMC файлове, колкото желаете, като всеки от тях може да отразява различна страна от производителността на системата.



Фигура 5. Монитор на производителността

За да активирате наблюдението на локалните приложения, трябва да изключите променливата на обкръжението DB2CONNECT\_IN\_APP\_PROCESS.

Допълнителни DCS APPLICATIONS команди, както и повече информация за използването на Монитора на производителността в Windows с DB2 UDB и DB2 Connect потърсете в *Ръководство за администриране*.

---

## Глава 9. Средства за администриране

В тази глава са описани средствата, които могат да ви помогнат при администрирането. Съдържа следните раздели:

- “Процесор за обработка на команди”
- “Използване на средствата за импорт и експорт” на страница 92

---

### Процесор за обработка на команди

процесор за обработка на команди ви помага да съставите SQL изрази спрямо хост или AS/400 сървър на база данни, предшествани от db2. За информация за разликите между хост и AS/400 SQL и DB2 Connect SQL, вижте “SQL оператори за хост или AS/400 сървър, поддържани от DB2 Connect” на страница 45 и “SQL оператори за хост или AS/400 сървър, отхвърлени от DB2 Connect” на страница 45.

За да се обърнете към процесор за обработка на команди в интерактивен режим на вход, направете следното:

**OS/2** Щракнете върху **OS/2 Warp**, и изберете **IBM DB2** → **Процесор за обработка на команди** или въведете командата **db2**.

#### **Windows 32-битова операционна система**

Щракнете върху **Старт** и изберете **Програми** → **IBM DB2** → **Процесор за обработка на команди**.

**Забележка:** Може също да извикате процесор за обработка на команди в интерактивен режим на вход, като въведете от MS-DOS промпт командата **db2cmd**, последвана от командата **db2**.

**UNIX** Въведете командата **db2** от процесора за обработка на команди

В интерактивен режим на вход промптът изглежда по следния начин:

```
db2 =>
```

В интерактивен режим на вход не трябва да въвеждате DB2 командите с префикс db2. Вместо това просто въведете DB2 командата. Например:

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

За да изпълните команди на операционната система в интерактивен режим, поставете удивителен знак (!) преди командата. Например:

```
db2 => !dir
```

Ако е необходимо да въведете дълга команда, която не се събира на един ред, използвайте знака за интервал, следван от знака за продължение на ред, \. Когато достигнете края на реда, натиснете клавиша **Enter**, за да продължите да въвеждате командата на следващия ред. Например:

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

За да прекратите интерактивния режим на вход, въведете командата **quit**.

За да можете да използвате процесор за обработка на команди, трябва да приложите свързване, както е описано в Глава 7, “Свързване на приложения и помощни програми” на страница 75.

**Забележка:** Когато използвате процесор за обработка на команди на UNIX–базирани системи, които не се намират в режим на интерактивен вход, трябва да поставите двойни кавички около специалните символи (като \* и ?), когато съставяте SQL изрази.

За допълнителна информация за SQL командите се обърнете към *Справочник на командите*.

---

## Използване на средствата за импорт и експорт

Средствата за импорт и експорт ви позволяват да преместите данни от S/390 хост или AS/400 база данни във файл на DB2 Connect работна станция или обратно. След това можете да използвате тези данни с друго приложение или RDBMS, което поддържа импортирания/експортирания формат. Например, можете да експортирате данни от DB2 за OS/390 във ASCII файл с разделители и по–късно да го импортирате в база данни DB2 UDB за Windows 2000.

Можете да изпълните експортирането или импортирането от клиент на базата данни или от DB2 Connect работна станция.

### Забележки:

1. Данните, които ще се импортират или експортират трябва да са в съответствие с ограниченията за размера и типа на данните и на двете бази данни.
2. За да повишите производителността при импортирането, можете да използвате командата SQL. Определете COMPOUND=*число* в импортирания API или параметъра за низ CLP *файлов тип*, за да групирате определения брой SQL изрази в блок. Така може да се намали натоварването на мрежата и времето за отговор.
3. Информация за синтаксиса на средствата за импорт и експорт от Процесор за обработка на команди можете да получите от *Справочник на командите*.

## Преместване на данни от работна станция на S/390 или AS/400 сървър на база данни

За да експортирате до S/390 или AS/400 сървър на база данни:

1. Експортирайте редовете с информация от таблиците на DB2 UDB за Unix, Windows NT, Windows 2000, OS/2 в PC/IXF файл.
2. Ако DRDA сървърът на база данни не съдържа таблица с атрибутите, съвместими с информацията, която ще се импортира в нея, създайте съвместима таблица.
3. Като използвате опцията INSERT, импортирайте PC/IXF файла в таблица в DRDA сървъра на база данни.

## Преместване на данни от DRDA сървър в работна станция

За да импортирате данни от DRDA сървър на база данни:

1. Експортирайте редовете с информация от таблицата на DRDA сървъра на база данни в PC/IXF файл.

2. Използвайте PC/IXF файла, за да импортирате в DB2 таблица.

## Ограничения

При програмата DB2 Connect операциите за импорт или експорт трябва да отговарят на следните условия:

- Файловият тип трябва да е PC/IXF.
- Дефинициите за индекси не се съхраняват при експорт и не се използват при импорт.
- Трябва да съществува таблица с атрибути, които са съвместими с тези на данните, преди да можете да импортирате в нея. При импорт с програмата DB2 Connect не може да се създаде таблица, защото единствената поддържана опция е INSERT.
- Интервалът за записване на променливи не трябва да е определен при импорт.

Ако тези условия не се спазят, операцията няма да се изпълни и ще се генерира съобщение за грешка.

## Смесени еднобайтови и двубайтови данни

Ако импортирате или експортирате смесени данни (колони, в които има както еднобайтови, така и двубайтови данни), разгледайте следното:

- На системи, които съхраняват данни в EBCDIC (MVS, OS/390, OS/400, VM и VSE), shift-out и shift-in символите маркират началото и края на двубайтовите данни. Когато дефинирате дължините на колоните за таблиците на базата данни, трябва да внимавате да оставите достатъчно място за тези символи.
- Препоръчват се колоните с променлива дължина на символите, освен ако данните в колоната не са с постоянен размер. Ако това е така, може да се приеме и фиксирана дължина.

## Заменяне на Помощната програма SQLQMF

Функцията на помощната програма SQLQMF в DDCS за OS/2 се изпълнява от функциите Импорт/Експорт на DB2 Connect. Предимствата са:

- Няма необходимост от QMF на хоста
- Няма необходимост от влизане в системата на хоста (все още е необходим TSO идентификатор на DB2 за MVS/ESA или DB2 Universal Database за OS/390)
- Поддържа на DB2 за MVS, DB2 за OS/390, DB2 за OS/400 и DB2 за VM и VSE
- Постига се добра производителност с помощта на съставен SQL
- Поддържа редица файлови формати освен ASCII
- Може да се изпълни от компютър клиент без SNA възможности за свързване.

За допълнителна информация за използването на тези команди се обърнете към *Справочник на командите*.





---

## Глава 10. Защита

В тази глава са представени съображенията за защитата в DB2 Connect, включително типовете оторизация. Освен това осигурява някои допълнителни препоръки и съвети за защита на потребителите на DB2 Universal Database за OS/390.

Допълнителна информация за настройка на защитата с DCE потърсете в *Ръководство за администриране*, както и в ръководствата на базата данни и DCE за вашия хост или AS/400 сървър на база данни.

**Забележка:** Когато използвате DB2 Connect с DCE защита, е необходимо DCE софтуерът да се инсталира на работната станция на DB2 клиент, както и на хоста или AS/400 сървъра на база данни, но не е необходимо да го инсталирате на работна станция с DB2 Connect. Допълнителна информация за необходимия софтуер за DCE потърсете в книгата *DB2 Connect: Бърз старт*.

---

### Разпознаване

Като администратор на DB2 Connect заедно с администратора на хоста или AS/400 базата данни можете да определите къде да се проверява валидността на имената и паролите на потребителите. Има пет възможности:

- Валидността се проверява върху клиента
- Валидността се проверява върху DB2 Connect работна станция
- Валидността се проверява едновременно на DB2 Connect работната станция и на хоста или AS/400 сървъра
- Валидността се проверява на хоста или AS/400 сървъра
- Валидността се проверява на сървъра за DCE защита

Къде да се проверява валидността, се определя чрез параметъра за типа на разпознаването в системната директория на базата данни и параметъра за типа защита в директорията на възела при APPC или APPN възли. Допълнителна информация за обновяване на тези директории можете да намерите в Глава 6, “Обновяване на директории на бази данни” на страница 63.

#### **Забележки:**

1. DB2 Connect не проверява валидността на потребителя. Ако искате валидността да се проверява от DB2 Connect работна станция, ще се използва локалната подсистема за защита, която ще проверява въведения потребителски идентификатор и парола при всяка заявка за свързване CONNECT. Следователно, когато настройвате DB2 Connect Enterprise Edition сървър и искате да използвате опцията AUTHENTICATION=SERVER, трябва да настроите всички необходими потребителски идентификатори и пароли на сървъра.
2. Ако използвате DCE директорни услуги, разпознаването се изпълнява по различен начин. За повече информация вижте “Защита с DCE директорни услуги” на страница 179.

Следните типове разпознаване са разрешени при DB2 Connect:

#### **CLIENT**

Валидността на името и паролата на потребителя се проверява на клиента.

## SERVER

Валидността на името и паролата на потребителя се проверява на DB2 Connect работната станция. Ако не се посочи тип разпознаване, се приема **SERVER**.

## SERVER\_ENCRYPT

Както и при типа разпознаване SERVER, валидността на името и паролата на потребителя се проверяват на DB2 Connect работната станция, но предаваните пароли се кодират на клиента и се декодират на DB2 Connect работната станция.

**DCS** Валидността на името и паролата на потребителя се проверяват на хоста или AS/400 сървъра на база данни.

## DCS\_ENCRYPT

Както и при DCS разпознаването, валидността на името и паролата на потребителя се проверяват на хоста или AS/400 сървъра на база данни, но изпращаните пароли се кодират на клиента и в зависимост от типа разпознаване, определено на DB2 Connect работната станция се декодират на DB2 Connect работната станция или хоста или AS/400 сървъра на база данни.

**DCE** Валидността на името и паролата на потребителя се проверява на DCE сървър за защита.

SERVER\_ENCRYPT и DCS\_ENCRYPT разпознаването имат еднаква семантика като SERVER и DCS разпознаването по отношение на мястото, където се изпълнява то. Те се различават по това, че обменните пароли се кодират в източника (клиента или DB2 Connect сървъра) и се декодират в приемника (DB2 Connect сървъра или хоста или AS/400 сървъра на база данни), както е определено в типа разпознаване, записано в източника.

Кодирани и некодирани стойности с еднакви места за разпознаване могат да се използват, за да се изберат различни комбинации за кодиране между клиент и DB2 Connect сървър или DB2 Connect сървър и хост или AS/400 сървър на база данни, като не се променя мястото, където се извършва разпознаването. Ето някои примери за това как това може да се използва в шлюз, като с "шлюз" се обозначава DB2 Connect сървър:

Разпознаване от клиента	Разпознаване от шлюз	Място на разпознаване	Кодиране клиент-шлюз?	Кодиране шлюз-сървър?
SERVER_ENCRYPT	SERVER	шлюз	да	не
DCS_ENCRYPT	DCS	сървър	да	не
DCS	DCS_ENCRYPT	сървър	не	да
DCS_ENCRYPT	DCS_ENCRYPT	сървър	да	да

Единственият APPC параметър за защита, който се поддържа при използване на SERVER\_ENCRYPT или DCS\_ENCRYPT, е SECURITY=NONE.

### Забележки:

1. Ако не е определен параметър за разпознаване в някой запис на системната база данни, който се използва от DB2 Connect при установяване на връзка, тогава DB2 Connect ще използва разпознаване от тип **SERVER**.
2. Както при комуникациите клиент-сървър в DB2 Universal Database, типът разпознаване не е задължителен на отдалечен клиент, свързан към DB2 Connect Enterprise Edition шлюз. Може да е определен, за да се оптимизира производителността, тъй като в този случай няма да е необходимо да се получава от шлюза и така ще се намали времето, необходимо за транзакция.

3. В случай на несъответствие между стойността на клиента и стойността на шлюза, предимство има стойността, определена на DB2 Connect шлюза.

## Типове защита

В този раздел са представени различни комбинации от параметри за разпознаване и защита, които се поддържат от DB2 Connect при APPC и TCP/IP свързване.

Следващата дискусия се отнася и за двата типа свързване.

## Типове защита при APPC свързване

Следващите типове защита са разрешени при APPC свързване, за да определят каква информация, свързана със защитата, ще се предава на комуникационното ниво:

**SAME** Само името на потребителя се предава на хоста или AS/400 сървър на базата данни.

### PROGRAM

Името и паролата на потребителя се предават на хоста или AS/400 сървър на база данни.

**NONE** Не се предава информация, свързана със защитата.

Таблица 5 показва възможните комбинации от тези стойности и типа разпознаване, определено на DB2 Connect работната станция, като посочва къде се проверява валидността при всяка комбинация. Само посочените в тази таблица комбинации се поддържат от DB2 Connect при APPC свързване.

Случай	Тип разпознаване в записа на директорията на DB2 Connect работната станция	Защита	Проверка за валидност
1	CLIENT	SAME	Клиент
2	SERVER	SAME	DB2 Connect сървър
3	SERVER	PROGRAM	DB2 Connect сървър и хост или AS/400 сървър на база данни
4	SERVER_ENCRYPT или DCS_ENCRYPT	NONE	хост или AS/400 сървър на база данни
5	DCS	PROGRAM	хост или AS/400 сървър на база данни
6	DCE	NONE	DCE сървър за защита

Ако към DB2 Connect Enterprise Edition сървър са свързани отдалечени клиенти, определете следните типове разпознаване и защита:

- Ако отдалечен клиент е свързан към DB2 Connect сървър чрез APPC, определете тип защита NONE на отдалечения клиент.
- Ако типът разпознаване в конфигурацията на мениджъра на базата данни на DB2 Connect сървъра е CLIENT, на всеки отдалечен клиент определете CLIENT.
- Ако типът разпознаване на DB2 Connect сървъра е SERVER, SERVER\_ENCRYPT, DCS или DCS\_ENCRYPT, на всеки отдалечен клиент определете един от тези типове. (Няма значение кой от тези 4 типа ще определите на отдалечения клиент.)

### Забележки:

1. При AIX системи всички потребители, които влизат в системата чрез APPC и тип защита SAME, трябва да принадлежат на AIX системната група.
2. При AIX системи с отдалечени клиенти, потребителският модел на DB2 Connect, който работи на DB2 Connect работната станция, трябва да принадлежи на AIX системната група.
3. Достъпът до хоста или AS/400 сървъра на базата данни се контролира от неговия собствен механизъм за защита или подсистема; например Virtual Telecommunications Access Method (VTAM) и Resource Access Control Facility (RACF). Достъпът до защитени обекти на база данни се управлява от SQL операторите **GRANT** и **REVOKE**.

## Типове защита при TCP/IP свързване

TCP/IP комуникационният протокол не поддържа възможност за защита на ниво мрежов протокол. Ето защо къде ще се изпълни проверката се определя само от типа разпознаване. Само посочените в тази таблица комбинации се поддържат от DB2 Connect при TCP/IP свързване.

Случай	Тип разпознаване в записа на директорията на DB2 Connect работната станция	Проверка за валидност
1	CLIENT	Клиент
2	SERVER или SERVER_ENCRYPT	DB2 Connect работна станция
3	Неприложимо	Няма
4	DCS или DCS_ENCRYPT	хост или AS/400 сървър на база данни
5	DCE	DCE сървър за защита

## Дискусия за типовете защита

Следващата дискусия се отнася както за APPC, така и за TCP/IP свързвания, както са описани по-горе и представени в Таблица 5 на страница 97 и Таблица 6. Всеки случай е описан по-подробно, както следва:

- В случай 1 валидността на името и паролата на потребителя се проверява само на отдалечения клиент. (За локален клиент валидността на името и паролата на потребителя се проверява само на DB2 Connect сървъра.)

Очаква се, че потребителят ще бъде разпознат там, където за първи път е въвел регистрацията си. През мрежата се изпраща идентификаторът на потребителя, но не и неговата парола. Използвайте този тип защита, само ако всички работни станции клиенти имат адекватни средства за защита, на които можете да се доверите.

- В случай 2 валидността на името и паролата на потребителя се проверява само на DB2 Connect сървъра. Паролата се изпраща през мрежата от отдалечения клиент до DB2 Connect сървъра, но не и към хоста или AS/400 сървъра на база данни.
- В случай 3 валидността на името и паролата на потребителя се проверява както на DB2 Connect сървъра, така и на хоста или AS/400 сървъра на база данни. Паролата се изпраща през мрежата от отдалечения клиент до DB2

Connect работната станция и от DB2 Connect работната станция до хоста или AS/400 сървъра на база данни.

Тъй като валидността се проверява на две места, трябва да се поддържа едно и също множество от имена и пароли на потребители както на DB2 Connect сървъра, така и на хоста или AS/400 сървъра на база данни.

- В случай 4 валидността на името и паролата на потребителя се проверява само на хоста или AS/400 сървъра на база данни. Идентификаторът и паролата на потребителя се изпращат през мрежата от отдалечения клиент до DB2 Connect сървъра и от DB2 Connect сървъра до хоста или AS/400 сървъра на база данни.
- В случай 5 клиентът получава DCE закодиран билет от DCE сървъра за защита. Билетът се предава непроменен през DB2 Connect към сървъра, където сървърът проверява валидността с помощта на DCE услугите за защита.

---

## Променяне на MVS парола

Сега DB2 Connect осигурява възможност да се променят паролите на потребителите. Това е особено полезно, когато за разпознаване на потребителите се използват хост услуги за защита, като помощното средство за управление на достъпа (RACF – Resource Access Control Facility). По-рано, за да се промени паролата на хоста, трябваше потребителите да влязат в TSO сесия, в която да променят своите пароли. С новия механизъм за обслужване на паролите, който се осигурява от DB2 Connect продуктите, за да променят своята парола, потребителите могат да изпълнят SQL оператор CONNECT от DB2 процесора за команди (CLP), да използват бутона PASSWORD в DB2 Асистента за конфигуриране на клиент (CCA) или да натиснат бутона CHANGE в диалога за влизане в ODBC.

За да се променят MVS пароли от DB2 Connect работни станции, свързани към DB2 за OS/390 версия 5.1 през TCP/IP, полето за разширена защита трябва да е установено на "Да". Това поле се появява в панела DB2 OS/390 DSNTIPR.

За да се променят MVS пароли на хост системи, свързани чрез SNA, на хоста трябва да е настроена специална програма за управление валидността на паролите, а DB2 Connect работната станция трябва да е конфигурирана да комуникира с тази хост програма.

Програмата за управление валидността на паролите на хоста се осигурява като част от следните MVS програмни продукти:

- MVS/ESA SP версия 4.2 или следваща (управлението на валидността на паролите е част от компонента APPC/MVS)
- CICS/ESA версия 3.3 или следваща

и с инсталиран IBM Resource Access Control Facility (RACF) 1.9.2.

Вие трябва:

1. Да конфигурирате транзакционната програмата на хоста да получава вашите заявки за обслужване валидността на паролата.
2. Да конфигурирате вашата DB2 Connect работна станция да комуникира с транзакционната програма на хоста.

## Конфигуриране на DB2 Connect работна станция за Управление валидността на паролите

След като е конфигурирана транзакционната програма за управление валидността на паролите, ще трябва да конфигурирате вашата DB2 Connect работна станция да комуникира с нея. Тази конфигурация включва две стъпки:

1. Дефиниране на име на символно предназначение за хост програмата за обслужване на паролите в SNA подсистемата на вашата DB2 Connect работна станция.
2. Записване на това име в DCS директорията за базата данни, която е разположена на тази хост система.

## Стъпка 1. Дефиниране на име на символно предназначение

Как ще дефинирате името на символното предназначение за програмата за управление валидността на паролите зависи от това каква SNA подсистема използвате:

- Ако използвате SNA подсистема, която може да се конфигурира от DB2 Асистент за конфигуриране на клиент (CCA), тогава трябва да го използвате и за конфигуриране името на символното предназначение. Ще трябва да получите LU името за програмата за управление валидността на паролите от вашия MVS администратор.
- Ако вашата SNA подсистема не може да се конфигурира с Асистента за конфигуриране на клиент, трябва да следвате документацията, получена заедно с вашата SNA подсистема. Ще трябва да получите следната информация от вашия MVS администратор.
  - Мрежово име за хоста, към който се свързвате.
  - LU име за програмата за управление валидността на паролите на хоста.

Когато конфигурирате името на символното предназначение, трябва да определите също така 'x'06F3F0F1' (шестнайсетично–десетично число) за името на Транзакционната програма (TP) и да настроите защитата да е NONE. Можете да определите режим като #INTER или някой друг режим, предложен от MVS.

## Стъпка 2. Записване на името на символно предназначение в DCS директорията

Ако използвате DB2 Connect на платформа, която поддържа Асистент за конфигуриране на клиенти, трябва да го използвате, за да обновите DCS директорията с името на символно предназначение за програмата за управление валидността на паролите на хоста. Би трябвало да можете да направите това, независимо от SNA подсистемата на вашата DB2 Connect работна станция.

Освен това можете да използвате командата **catalog dcs database** (от DB2 CLP), за да запишете името на символно предназначение в DCS директорията. Например:

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

записва *pempgm* като име на символно предназначение, което ще се използва, когато потребителите подадат заявка за промяна на паролата за базата данни *db1*.

## Конфигуриране на хоста за Управление валидността на паролите

За по–подробна информация за MVS паролите се обърнете към едно от следните електронни издания:

**APPC/MVS:**

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

**SecureWay Communications:**

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

**TxSeries:**

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

---

## Допълнителни препоръки и съвети за защитата

В този раздел са осигурени някои допълнителни препоръки и съвети за защита на потребителите на DB2 Connect.

### Кодове за разширена защита

До DB2 Universal Database за OS/390 версия 5.1, заявките за свързване, които съдържаха идентификаторите на потребителите или паролите биха могли да приключат с грешка със SQL30082 код за причина 0, но без никаква информация за това какво може да е грешно.

DB2 Universal Database за OS/390 версия 5.1 въведе усъвършенстване, което осигурява поддръжка на разширени кодове за защита. Използването на разширена защита осигурява допълнителна диагностична информация освен кода за причината, като например за изтичане валидността на парола (PASSWORD EXPIRED).

За да се възползвате от тази възможност, инсталационният параметър на DB2 Universal Database за OS/390 за разширена защита ZPARM трябва да е установен със стойност ДА. Използвайте DB2 Universal Database за OS/390 инсталационния панел DSN6SYSP, за да определите EXTSEC=YES. Освен това можете да използвате DDF панел 1 (DSNTIPR), за да направите това. Стойността по подразбиране е EXTSEC=NO. В случай, че валидността на паролата е изтекла, приложенията за PC, UNIX, Apple Macintosh и Web, които използват DB2 Connect, ще получат съобщението за грешка SQL01404.

### TCP/IP предварителна проверка за защита

Ако искате да осигурите поддръжка за опцията за защита на DB2 Universal Database AUTHENTICATION=CLIENT, използвайте DB2 Universal Database за OS/390 инсталационния панел DSNTIP4 (DDF панел 2), за да установите TCP/IP проверката за защита на ДА.

### Защита на ODBC и Java приложения

ODBC и Java приложенията на работната станция използват динамичен SQL. При някои инсталации това може да създаде проблеми по отношение на защитата. DB2 Universal Database за OS/390 въвежда нова опция за свързване DYNAMICRULES(BIND), която позволява изпълнението на динамичен SQL под оторизацията на собственика или този, който е изпълнил свързването. Обърнете се към *Справочник на командите*, за да видите как може да се определи DYNAMICRULES в DB2 Connect.

DB2 Universal Database и DB2 Connect осигуряват нов CLI/ODBC конфигурационен параметър CURRENTPACKAGESET в конфигурационния файл DB2CLI.INI. Той трябва да съдържа името на схемата, която има съответните права на достъп. SQL операторът SET CURRENT PACKAGESET schema автоматично ще се генерира след всяко свързване към приложението.

Използвайте ODBC Мениджъра, за да обновите DB2CLI.INI. За допълнителна информация вижте *Приложение за инсталиране и конфигуриране*.

## Поддръжка на смяна на пароли

Ако операторът SQL CONNECT върне съобщение, посочващо, че е изтекла валидността на паролата на потребителя, при DB2 Connect версия 5.2 и следваща е възможно да промените паролата, без да се свързвате към TSO. През DRDA, DB2 Universal Database за OS/390 може да промени паролата вместо вас.

Старата парола, заедно с новата парола и паролата за потвърждение трябва да се въведат от потребителя. Ако защитата, определена на DB2 Connect Enterprise Edition сървъра е DCS, тогава заявка за промяна на паролата се изпраща към DB2 Universal Database за OS/390 сървъра на базата данни. Ако типът на защитата е SERVER, тогава се променя паролата на DB2 Connect сървъра.

Допълнително удобство е, че не е необходима отделна LU дефиниция. За допълнителна информация се обърнете се към DB2 Connect Enterprise Edition ръководството *Бърз старт*.



---

## Глава 11. Преобразуване на SQLCODE

Различните реляционни бази данни на IBM не винаги генерират едни и същи кодове SQLCODE за подобни грешки. Дори когато се използва един и същи SQLCODE, той може да се съпътства от токени, които се определят по различен начин. Списъкът с токени се предава в полето SQLERRMC на SQLCA. По подразбиране DB2 Connect преобразува кодовете SQLCODE и токените от всеки IBM хост или AS/400 сървър на база данни до подходящия вид DB2 Universal Database SQLCODE.

---

### Изключване на SQLCODE преобразуването

Ако искате да изключите преобразуването на SQLCODE, определете NOMAP в параметричния низ на DCS директорията или на информационния обект за DCE маршрутизиране. Информация за обновяването на DCS директория вижте в Глава 6, “Обновяване на директории на бази данни” на страница 63. Информация за използването на DCE потърсете в Приложение D, “Използване на DCE директорните услуги” на страница 173.

Ако прехвърлите приложение директно от хост или AS/400 сървър на база данни (като DB2 UDB за OS/390), може да предпочетете да изключите преобразуването на SQLCODE. Така ще можете да използвате приложението, без да променяте използвания код SQLCODE.

---

### Обвързване на SQLCODE преобразуването

По подразбиране DB2 Connect преобразува кодовете SQLCODE и токените от всеки IBM хост или AS/400 сървър на база данни към подходящите кодове DB2 UDB SQLCODE. Следните файлове са копия на SQLCODE преобразуването по подразбиране:

**dcs1dsn.map**

Преобразува DB2 за MVS/ESA и DB2 UDB за OS/390 SQLCODE

**dcs1ari.map**

Преобразува DB2 за VSE и VM SQLCODE

**dcs1qsq.map**

Преобразува DB2 UDB за AS/400 SQLCODE

Не е необходимо преобразуване на DB2 системите за OS/2 и базираните на UNIX.

Ако искате да замените стандартното SQLCODE преобразуване или ако използвате хост или AS/400 сървър на база данни, който не поддържа преобразуване на SQLCODE (сървър на база данни, който не е на IBM), можете да копирате един от тези файлове и да го използвате като основа за новия файл за преобразуване на SQLCODE. Като копирате файла, вместо да го редактирате директно, ще можете при необходимост винаги да се обърнете към оригиналното преобразуване на SQLCODE.

Определете името на файла с новото преобразуване на SQLCODE в параметричния низ на DCS директорията или информационния обект за DCE маршрутизиране. Информация за обновяване на директорията за DCS вижте в Глава 6, “Обновяване на директории на бази данни” на страница 63. Информация за използване на DCE потърсете в Приложение D, “Използване на DCE директорните услуги” на страница 173.

Всеки файл за преобразуване е ASCII файл, който се създава и редактира с помощта на ASCII редактор. При първоначалното инсталиране файлът се съхранява в директорията `par` в инсталационната пътека.

Файлът може да съдържа следните специални типове редове:

- &&** Логическото начало на файла. Всички редове преди първата поява на символите `&&` се разглеждат като коментар и се игнорират. Ако файлът не съдържа нищо след `&&`, не се изпълнява преобразуване на `SQLCODE`. Освен това можете да изключите преобразуването на `SQLCODE` с помощта на параметъра `NOMAP`, както беше обяснено преди това.
- \*** Ако е в началото на реда означава коментар.
- W** Ако е единственият символ на реда, означава, че предупредителните флагове трябва да се преобразуват. По подразбиране се предават оригиналните предупредителни флагове. `W` трябва да е главна буква.

Всички други редове след `&&` трябва или да са празни, или да съдържат изрази за преобразуване в следния вид:

```
input_code [, output_code [, token_list]]
```

*input\_code* представлява едно от следните:

*sqlcode* `SQLCODE` от хоста или `AS/400` сървър на базата данни.

- U** Всички недефинирани отрицателни кодове `SQLCODE` (тези, които не са изброени в този файл) се преобразуват до определения *изходящ\_код*. Ако на този ред не е определен *изходящ\_код*, се използва първоначалния `SQLCODE`. Този символ трябва да е главна буква.
- P** Всички недефинирани положителни кодове `SQLCODE` (тези, които не са изброени в този файл) се преобразуват до определения *изходящ\_код*. Ако на този ред не е определен *изходящ\_код*, се използва първоначалния `SQLCODE`. Този символ трябва да е главна буква.
- csnp** Кода за клас `SQLSTATE` от хоста или `AS/400` сървър на базата данни. *np* е един от следните:

- 00** Неопределено успешно приключване
- 01** Предупреждение
- 02** Няма данни
- 21** Нарушаване на броя на редовете в таблицата
- 22** Изключение в данните
- 23** Нарушаване на ограничение
- 24** Невалидно състояние на указател
- 26** Невалиден SQL оператор
- 40** Ролбек на транзакция
- 42** Грешка в достъпа
- 51** Невалидно състояние на приложение
- 55** Обект не в необходимото състояние
- 56** Други разнородни грешки в продукта или SQL оператор
- 57** Недостъпен ресурс или вмешателство на оператор
- 58** Системна грешка

Определеният код *изходящ\_код* се използва за всички кодове `SQLCODE` с този код за клас, който не е определен изрично във файла за преобразуване. Ако на този ред не е определен *изходящ\_код*,

първоначалният SQLCODE се преобразува в себе си, като не се копират никакви токени.

Символите **с** трябва да са малки букви.

Ако във файла за преобразуване се среща повече от веднъж един и същи код *входящ\_код*, се използва първия.

*изходящ\_код* е получения в резултат SQLCODE. Ако не е определена никаква стойност, се използва първоначалния SQLCODE.

Ако определите изходящ код, можете също така да определите и едно от следните:

- (s) Началният SQLCODE плюс номера на продукта (ARI, DSN или QSQ) ще се постави в полето за токен на SQLCA.

Първоначалният SQLCODE се връща като единствен токен. Тази възможност е подходяща за обработка на недефинирани кодове SQLCODE, с изключение на +965 и -969. Ако +965 или -969 е *изходящ\_код*, списъкът с токени, върнат в полето SQLERRMC на SQLCA включва първоначалния SQLCODE, следван от идентификаторът на продукта, следван от първоначалния списък с токени.

Символът **s** трябва да е малка буква.

(списък-токени)

Списък с токени, отделени със запетаи. С въвеждането само на запетая се пропуска определен токен. Например (*t2,,t4*) означава, че първия и третия изходящ токен са null.

Всеки токен е във вид на число (*n*), като може да се предшества от **с** или след него да има **с** или **i**. Интерпретират се както следва:

- с** Типът данни на токена на тази позиция е CHAR (по подразбиране). Ако **с** е преди *n*, се отнася за входящия токен; ако е след *n*, се отнася за изходящия. Символът **с** трябва да е малка буква.
- i** Типът данни на токена на тази позиция е INTEGER (по подразбиране). Ако **i** е след *n*, се отнася за изходния токен. **i** не трябва да е преди *n*, защото продуктите на IBM за хост или AS/400 сървър на база данни поддържат само токени от тип CHAR. Символът **i** трябва да е малка буква.
- n* Число или числа, което посочва кои токени за хост или AS/400 сървър на база данни да се използват. Подреждат се в желанния ред за мястото в изходящия SQLCA. Числото посочва токена на хоста или AS/400 сървъра на базата данни; подреждането посочва реда, по който токени се поставят в SQLCA.

Например хоста или AS/400 сървъра на база данни може да върне два токена – 1 и 2. Ако искате токен 2 да се появи преди токен 1 в изходящия SQLCA, определете (2,1).

Като се свържат с точки, няколко числа токени могат да се комбинират, за да образуват един изходящ токен от тип CHAR.

Запетаите се използват, за да се разделят изходящите токени. Ако не е определен токен преди запетаята, на тази позиция в SQLCA няма да се включи изходящ токен. Всички токени, които се намират в изходящия SQLCA след последния определен токен, се преобразуват в токен от тип null.

Фигура 6 на страница 106 показва прост файл за преобразуване на SQLCODE.

```

&&
-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)

```

Фигура 6. Файл за преобразуване на SQLCODE

Всеки израз за преобразуване във файла се описва както следва:

1. SQLCODE се преобразува от -007 до -007. Първия входящ токен, получен от хоста или AS/400 сървъра на базата данни, се използва като първи изходящ токен и по подразбиране е CHAR. Не се прехвърлят други токени.
2. SQLCODE се преобразува от -010 до -010 (не е определен изходящ SQLCODE). В изходящия SQLCA не се поставят никакви токени.
3. SQLCODE се преобразува от -060 до -171. Първия входящ токен, получен от хоста или AS/400 сървъра на базата данни, се пренебрегва. Вторият се използва като първи токен в изходящия SQLCA и е от тип CHAR. Няма втори токен в изходящия SQLCA.
4. SQLCODE се преобразува от -204 до -204. Първият и вторият токен, получени от хоста или AS/400 сървъра на базата данни, са CHAR. Тези два входящи токена се комбинират, за да образуват един изходящ токен от тип CHAR, който ще е първия изходящ токен в SQLCA.
5. SQLCODE се преобразува от -633 до -206. Първият входящ токен, получен от хоста или AS/400 сървъра на базата данни, е CHAR. Конвертира се до INTEGER и се използва като втори токен в изходящия SQLCA. Първият токен в изходящия SQLCA е null, както е посочено със запетаята.
6. SQLCODE се преобразува от -30021 до -30021. Първият и вторият токен, получени от хоста или AS/400 сървъра на базата данни, са CHAR и те се използват като първи и втори токен в изходящия SQLCA.
7. Всички SQLCODE в SQLCA със SQLSTATE в класа 00 ще се преобразуват до SQLCODE +000.
8. Всеки недефиниран SQLCODE се преобразува до -969. Тази опция трябва да се използва само ако са изброени всички кодове, които могат да се преобразуват, включително онези, които са идентични и не е необходимо да се преобразуват. Опцията (s) посочва, че списъкът с токени, който ще се върне в полето SQLERRMC на SQLCA, включва първоначалния SQLCODE, следван от продукта, в който е възникнала грешката и след това първоначалния списък с токени. Ако не е включен записът U, всички неизброени кодове се прехвърлят, без да се преобразуват.
9. Всеки недефиниран положителен SQLCODE се преобразува до +965. Тази опция трябва да се използва само ако са изброени всички кодове, които могат да се преобразуват, включително онези, които са идентични и не е необходимо да се преобразуват. Опцията (s) посочва, че списъкът с токени, който ще се върне в полето SQLERRMC на SQLCA включва първоначалния SQLCODE, следван от продукта, в който е възникнала грешката и след това първоначалния списък с

токени. Ако не е включен записът **P**, всички непосочени положителни кодове се прехвърлят, без да се преобразуват.



---

## Глава 12. Производителност

DB2 Connect си взаимодейства с много различни продукти, включително DRDA сървъри на приложения, клиентски програми и комуникационни продукти. Производителността зависи от ефективната съвместна работа на всички тези елементи.

---

### Концепции и инструменти за производителността

*Производителността* е начинът, по който компютърната система се държи при определено натоварване. Зависи от достъпните ресурси и от това как те се използват и поделят. Ако искате да повишите производителността, трябва най-напред да решите какво разбирате под производителност. Можете да изберете много различни *метрики на производителността*, включително:

#### **Време за отговор**

Интервалът време между момента, когато приложение изпрати заявка към база данни до момента, когато получи отговор.

#### **Производителност на транзакция**

Броя единици работа, които могат да се завършат за единица време. Единицата работа може да е проста, като извличане или обновяване на ред, или сложна, например на включва стотици SQL оператори.

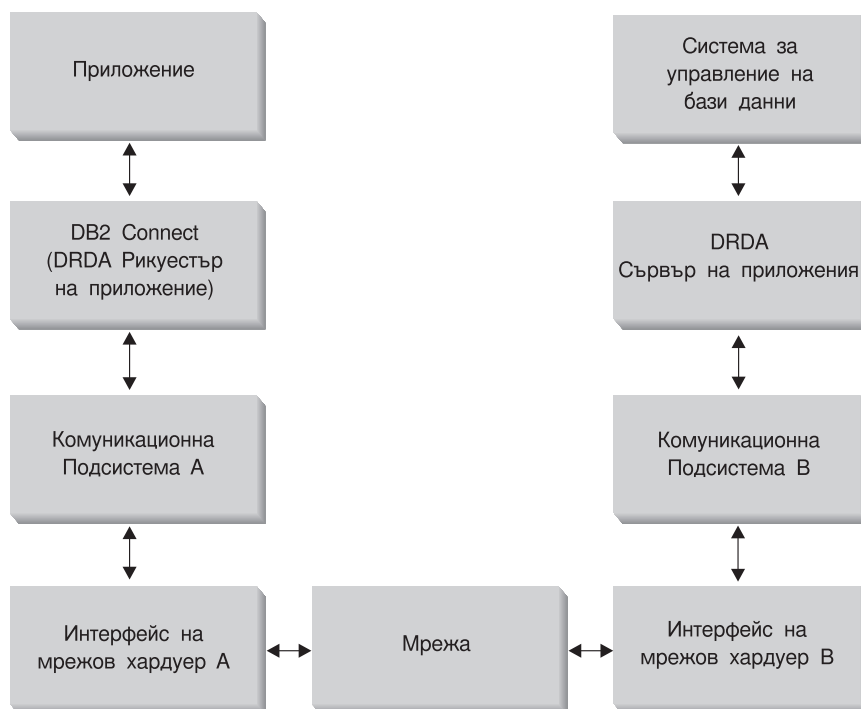
#### **Скорост на обмен на данни**

Количеството данни в байтове, които се обменят между DB2 Connect приложение и хост или AS/400 база данни за единица време.

Производителността ще се ограничи от достъпните хардуерни и софтуерни ресурси. Процесор, памет и мрежови адаптери са примери за хардуерни ресурси. Комуникационните подсистеми, пейджинг подсистемите, mbuf за AIX и link за SNA са примери за софтуерни ресурси.

### Потоци данни

Фигура 7 на страница 110 показва пътя на потока данни между хоста или AS/400 сървъра на база данни и работната станция през DB2 Connect.



Фигура 7. Потоци данни в DB2 Connect

- Хостът или AS/400 базата данни и част от комуникационната подсистема В обикновено работят на една и съща система. Тази система е изградена от един или повече CPU, място за съхранение, Входно/изходна подсистема, DASD и операционна система. Тъй като и други програми може също да използват тези ресурси, това би могло да доведе до проблеми с производителността.
- Мрежата се състои от комбинация от кабели, концентратори, комуникационни линии, превключватели и други комуникационни контролери. Например мрежовият хардуерен интерфейс В може да е с комуникационни контролери като 3745 или 3172 или адаптер token ring за AS/400. Може да има повече от една среда за пренос, която да се използва между мрежовите хардуерни интерфейси А и В.
- Мрежовият хардуерен интерфейс А може да е token ring, Ethernet\*\*, друг LAN адаптер или адаптер, който поддържа SDLC или X.25 протоколи. Комуникационната подсистема А може да е продукт като IBM Комуникационен сървър за OS/2, Microsoft SNA Сървър, IBM SNA сървър за AIX или SNAplus2 за HP-UX.
- DB2 Connect и комуникационната подсистема А обикновено се разполагат на една и съща система. В тази глава приемаме, че приложението също се намира на същата система.

## Тесни места

Производителността на транзакцията зависи от най-бавния компонент в системата. Ако определите кое е тясното място за производителността, често можете да решите проблема, като промените конфигурационните параметри, заделите повече ресурси на проблемния компонент, обновите компонента или добавите нов, за да разтоварите част от работата.

Можете да използвате различни средства, за да определите колко време е необходимо за обработката на запитването от всеки компонент. Така ще получите



представа за това кои компоненти трябва да се настройат или обновят, за да се увеличи производителността. Например, ако установите, че 60% от времето за обработка на запитването се пада на DB2 Connect машината, може да решите, че е хубаво да настроите DB2 Connect или (ако имате отдалечени клиенти) да добавите друга DB2 Connect машина към мрежата.

Допълнителна информация за средствата, свързани с производителността, потърсете в “Средства за проверка на производителността.”

## Сравнителни характеристики

*Генерирането на сравнителни характеристики* е начин да се сравни производителността в едно обкръжение с производителността на друго.

Може да се започне, като се изпълни тестово приложение в нормално обкръжение. Когато причините за проблема с производителността се стеснят, може да се разработят специализирани тестове, за да се ограничи обема на функцията, която се тества и наблюдава.

Приложенията за генериране на сравнителни характеристики не е необходимо да са сложни. За да се получи ценна информация, не е необходимо специализираните тестове да емулират цялото приложение. Започнете с прости измервания и увеличете сложността, само когато имате основание за това.

Характеристики на добрите тестове, генериращи сравнителни показатели:

- Всеки тест може да се повтори.
- Всяка итерация на теста започва от едно и също състояние на системата.
- Използвания хардуер и софтуер при генерирането на сравнителните характеристики съответства на вашето работно обкръжение.
- В системата няма други активни функции или приложения освен тези, които се измерват. Освен ако сценарият не включва някакъв обем друга активност, която да се извършва в системата.

**Забележка:** Стартираните приложения заемат памет, дори когато не се използват или са минимизирани. Това може да причини пейджинг и да изопачи резултатите от измерването.

## Средства за проверка на производителността

В следващата таблица се представени някои от средствата, които могат да ви помогнат да оцените производителността на системата. Тъй като тези средства също използват системни ресурси, може да предпочетете да ги оставите активни през цялото време.

<i>Таблица 7 (Страница 1 от 2). Средства за проверка на производителността</i>		
Система	Средство	Описание
<b>CPU и използване на паметта</b>		
AIX	vmstat, time, ps, tprof	Осигурява информация за проблеми при конкурентно използване на процесора и паметта на DB2 Connect работната станция и отдалечени клиенти.

<i>Таблица 7 (Страница 2 от 2). Средства за проверка на производителността</i>		
<b>Система</b>	<b>Средство</b>	<b>Описание</b>
HP-UX	vmstat, time, ps, монитор и наблюдение, ако са достъпни	
OS/2	SPM/2, THESEUS/2, pstat	
Win NT и Windows 2000	MS Performance Monitor	
<b>Активност на базата данни</b>		
Всички	Монитор на базата данни	Определя дали проблемът е причинен от базата данни.
MVS или OS/390	DB2PM (IBM), OMEGAMON/DB2 (Candle), TMON (Landmark), INSIGHT (Goal Systems) и DB2AM (BMC)	
Win NT и Windows 2000	MS Performance Monitor	
<b>Мрежова активност</b>		
AIX	netrmon	Отчита мрежова статистика на ниско ниво, включително статистика за TCP/IP и SNA, като брой пакети или фреймове, получени за секунда.
DOS или OS/2	Програма за трасиране и производителност на Token-Ring мрежа 16/4	Повечето програми за следене на мрежата зависят от платформата; тази програма работи само за мрежи token-ring.
Мрежов контролер като 3745	Монитор за производителност на NetView	Отчита използването на комуникационния контрол и VTAM.
OS/2	DatagLANce	Средство за трасиране, което представя графично данните, свързани с производителността.
UNIX-базирани	netstat	Предназначено за TCP/IP трафик.

## Оптимизиране на ODBC достъпа

DB2 Universal Database осигурява специална оптимизация, проектирана, за да подобри комуникационната производителност през ODBC. Тези усъвършенствания са достъпни за Microsoft Access, Lotus Approach или Visual Basic. Можете да се възползвате от по-добра ODBC производителност с помощта на Асистента за конфигуриране на клиенти (CCA) на DB2.

За да активирате оптимизиран ODBC, изпълнете следното:

Ако дефинирате ново свързване:

1. Стартирайте Асистента за конфигуриране на клиенти на DB2

2. Изберете псевдонима на базата данни, която искате да оптимизирате.
3. Щракнете върху бутона **Характеристики**.
4. Уверете се, че е избрано полето **Регистрирай тази база данни за ODBC**.
5. Изберете радио-бутона, който описва как искате да регистрирате тази база данни.
6. Щракнете върху бутона **Настройки**.
7. Щракнете върху бутона **Оптимизирай** и изберете приложението, чиито ODBC параметри искате да оптимизирате.
8. Щракнете **ДА** и излезте от Асистента за конфигуриране на клиенти.

Ако обновявате съществуващо свързване:

1. Стартирайте Асистента за конфигуриране на клиенти на DB2.
2. Изберете псевдонима на базата данни, която искате да оптимизирате.
3. Щракнете върху бутона **Характеристики**.
4. Щракнете върху бутона **Настройки**.
5. Щракнете върху бутона **Оптимизирай** от прозореца с параметри на CLI/ODBC и изберете приложението, чиито параметри искате да оптимизирате.
6. Щракнете **ДА** и излезте от Асистента за конфигуриране на клиенти.

За допълнителна информация за Асистента за конфигуриране на клиенти се обърнете към вашата книга *DB2 Connect: Бърз старт*. Най-новата информация за оптимизиране на ODBC достъп до DB2 за OS/390 можете да намерите, като посетите с вашия браузър: <http://www.ibm.com/software/data/db2/os390/odbcattlg.html>

---

## Проектиране на приложения

Когато създавате приложение, можете да подобрите производителността по редица начини, включително:

- Използване на SQL блокове и запомнени процедури.
- Групиране на заявките.
- Използване на предикатна логика, за да се изискат само данните, от които се нуждаете.
- Създаване на блокове данни.
- Използване на статичен SQL, когато е възможно.

### SQL блокове и запомнени процедури

Натоварването на мрежата може да е значително при приложения, които изпращат и получават много команди и отговори. SQL блоковете и запомнените процедури са два начина за намаляване на това натоварване.

Ако приложение изпраща редица SQL оператори, без да изисква програмна логика, може да използвате SQL блок. Ако е необходима логика на програма в рамките на група от SQL оператори, можете да използвате запомнени процедури.

Всички изпълними оператори освен следните могат да се поставят в рамките на SQL блок с оператори:

CALL  
FETCH  
CLOSE  
OPEN  
SQL блок  
Connect  
Prepare  
Release  
Describe  
Rollback  
Disconnect  
Set connection  
execute immediate

За допълнителни подробности се обърнете към *SQL Справочник*.

Информация за използването на SQL блокове в приложение потърсете в “НЕ АТОМАРЕН SQL блок” на страница 43. Информация за използването на SQL блокове с помагалото за импортиране потърсете в “Използване на средствата за импорт и експорт” на страница 92.

Запомнените процедури помагат да се намали мрежовия трафик, като поставят програмна логика на сървъра. В DB2 преди версия 5.0 запомнените процедури можеха да връщат само параметри и приложението трябваше да генерира отделна команда за записване на промените. В резултат се получават два мрежови хода. В DB2 версии 5.0 и следващи можете автоматично да запишете промените, когато излизате от процедурата. Освен това можете да върнете резултатен набор, който намалява логиката на приложението от страната на клиента.

Информация за използване на запомнени процедури потърсете в “Запомнени процедури” на страница 41.

## Групиране на заявки

Групирането на свързаните заявки към база данни (SQL операторите) в една заявка към базата данни може да намали броя на заявките и отговорите, прехвърляни през мрежата. Например ако се групират следните оператори:

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1  
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=2
```

в

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1 OR ROW_ID=2
```

ще се изпратят по-малко заявки през мрежата.

Освен това можете да използвате ключови думи като IN и BETWEEN, за да намалите броя на редовете, които се връщат. Допълнително можете да използвате ключовите думи WHERE, IN и BETWEEN в операторите UPDATE и DELETE.

## Предикатна логика

Можете да използвате предикатна логика, за да заявите само редовете и колоните, които са необходими. По този начин се намалява мрежовия трафик и натоварването на процесора при обмена на данни.

Например, не използвайте запитването:

```
SELECT * FROM TABLEA
```

ако на практика се е необходим само първия ред от TABLEA с ROW\_ID=1 или ако са необходими само колона 1 или колона 2.

## Създаване на блокове с данни

Трябва да създадете блокове данни, ако очаквате голямо количество данни от сървъра. Чрез създаването на блокове се подобрява използването на мрежовата ширина на лентата и се намалява натоварването на процесора както на хоста или AS/400 сървъра на базата данни, така и на DB2 Connect работната станция.

При изпращането и получаването на всяко съобщение има фиксирано натоварване на процесора и мрежата, независимо от размера на съобщението. Блоковете с данни намаляват броя на съобщенията, необходими за трансфер на едно и също количество данни.

Чрез създаването на блокове първият ред от данните от запитването няма да се достави на приложението, докато не се получи първия блок. Създаването на блокове увеличава времето за извличане на първия ред, но подобрява времето за извличане на следващите редове.

Друго съображение е количеството памет, което се използва. Използваната памет обикновено се увеличава, когато е активирана функцията за създаване на блокове. За пълна дискусия относно създаването на блокове, когато се използва SNA свързване, се обърнете към *DRDA Ръководство за свързваемост*.

В DB2 Connect можете да контролирате количеството данни, които се обменят в рамките на всеки блок, както е описано в “RQRIOBLK” на страница 117.

За да активирате създаването на блокове, използвайте опцията BLOCKING на командата rgr или bind. (За повече информация вижте “Командата BIND” на страница 79.) Създаването на блокове е включено, ако:

- Указателят е само за четене или
- Указателят е неопределен и създаването на блокове е определено при rgr или bind.

Дефиниции на неопределен указател, само за четене и указател, който може да се обновява, потърсете в *Ръководство за разработка на приложения*.

**Забележка:** Когато се използва динамичен SQL, указателят винаги е неопределен.

### SQL оператори със създаване на блокове

Обновяващите SELECT изрази (с оператори UPDATE/DELETE WHERE CURRENT OF) не могат да работят с блокове, така че ги използвайте, само когато е абсолютно необходимо.

Обновяващ SELECT осигурява, че редът не е променен между времето, когато се изпълни SELECT и се генерира UPDATE/DELETE. Ако това ниво на конкурентност не е важно за вашето приложение, алтернативата е да се използва DELETE или UPDATE с критерии за търсене, базирани на стойностите, върнати от не-обновяващ SELECT.

При SELECT само за четене определете FOR FETCH ONLY (освен под VM и VSE, където не се поддържа).

## Статичен и динамичен SQL

Използвайте статичен SQL, колкото се може повече. Избягва подготовка на изпълними SQL раздели и неопределени курсори. Ако не може да се избегне използването на динамичен SQL, можете да направите следното, за да минимизирате мрежовия трафик и да увеличите производителността:

- Ако операторът е SELECT и трябва да се подготви, изпълнете PREPARE ... INTO SQLDA. За SQLDA трябва да е заделен пълния размер, необходим за вашите параметри. Ако максималният брой колони е  $x$  и се очаква да се запази, заделете за SQLDA  $x$  SQLVAR. Ако броят на потенциалните колони не е сигурен (и паметта не е проблем), използвайте максималния брой SQLVAR (256).

Ако SQLDA приложението не е достатъчно голямо, за да съхрани върнатия SQLDA, програмата трябва да генерира друга команда DESCRIBE с достатъчно голям SQLDA, за да съхрани отново резултата. Това би увеличило мрежовия трафик.

Не използвайте последователност PREPARE и DESCRIBE. Използването на оператори PREPARE...INTO осигурява по-добра производителност.

- Изпълнете статично свързани оператори SQL COMMIT или ROLLBACK вместо динамични оператори COMMIT или ROLLBACK.
- Ако не е оператор SELECT, COMMIT или ROLLBACK, използвайте EXECUTE IMMEDIATE, за да изпълните оператора, вместо последователността PREPARE и EXECUTE.
- ODBC приложения използват динамичен SQL. Можете да използвате функцията за статични профили на CLI/ODBC, за да подобрите производителността. Тази функция ви позволява да прихванете и конвертирате ODBC обръщания в статични оператори, записани в пакет на база данни. Действителната производителност, която ще получите, зависи от сложността на вашето приложение. За повече информация се обърнете към *CLI Ръководство и справочник*.

## Други SQL съображения

Като цяло използването на Процесор за обработка на команди е по-бавно отколкото използването на динамичен SQL в програмата, защото Процесор за обработка на команди трябва да анализира входа, преди да предаде SQL заявката към базата данни. Освен това Процесор за обработка на команди форматира данните при получаването им, което може да не е необходимо за вашето приложение.

SQL операторите в интерпретивен език (като REXX) са значително по-бавни, отколкото същите оператори в компилиран език (като C).

Има два типа оператори CONNECT, наречени тип 1 и тип 2. При тип 2 свързването към базата данни поставя предишната връзка в пасивно състояние и не я прекъсва. Ако по-късно превключите към пасивната връзка, ще избегнете необходимостта от зареждане на библиотеки и настройката на вътрешните структури данни. Поради тази причина като се използва тип 2 свързване може да се увеличи производителността за приложения, които осъществяват достъп до повече от една база данни. Допълнителна информация за свързване тип 2 потърсете в *Ръководство за администриране* и *SQL Справочник*.

---

## Настройка на DB2 Connect

За настройка на DB2 Connect могат да се използват различни параметри от конфигурационния файл на мениджъра на базата данни. За информация как да промените тези параметри, се обърнете към *Ръководство за администриране*.

## RQRIOBLK

Параметърът RQRIOBLK определя максималния размер на входно/изходните мрежови блокове. По-големият размер на блока може да увеличи производителността при големи заявки. Размерът на блока обикновено не се отразява върху времето за отговор на малките заявки, като заявка за един ред данни.

По-големият размер на блока обикновено изисква повече памет на DB2 Connect работната станция. Това увеличава размера на работното множество и може да доведе до голям обем пейджинг при марки работни станции.

Използвайте размера на DRDA блока по подразбиране (32767), ако не причинява прекалено много пейджинг при изпълнението на вашето приложение. В противен случай намалете размера на входно/изходния блок, докато не се получава пейджинг. Ако се получи пейджинг, производителността забележително ще се влоши. Използвайте средствата за следене на производителността (като помощната програма vmstat за UNIX-базирани системи или SPM/2 за OS/2), за да определите дали в системата ви няма пейджинг. За други средства се обърнете към “Средства за проверка на производителността” на страница 111.

## DIR\_CACHE

Параметърът DIR\_CACHE определя дали се кешира информацията за директорията. При кеширането (DIR\_CACHE=YES), директорните файлове се прочитат и кешират в паметта, за да се намали натоварването от създаването на вътрешна директорна структура и прочитането на директорните файлове всеки път, когато се установява връзка.

Без кеширане (DIR\_CACHE=NO) при всяко свързване към база данни съответната директория се прочита от диска и след това се изпълнява търсенето. След като се намерят заявените записи, се освобождава цялата памет, свързана с търсенията в директорията.

При кеширането се изгражда общ директорен кеш по време на обработката на командата **db2start** и се освобождава при спирането на DB2. Този кеш се използва от всички DB2 сървърни процеси (db2agent). Освен това отделен директорен кеш за приложения се изгражда при първото свързване на приложението към базата данни и се освобождава при неговото приключване.

Всеки кеш осигурява образ на системната директория на базата данни, директорията на услугите за връзка с базата данни и директорията на възлите. Кешът намалява натоварването при свързването, като елиминира входно/изходния директорен файл и минимизира търсенето в директориите.

При обновяване на записана в кеша директория промените не се прехвърлят незабавно. Ако запис от директорията не бъде намерен в този кеш, се претърсва оригиналната директория.

Кеширането увеличава собствената памет, необходима за приложение. Без кеширането тази памет ще е необходима, само когато се обработва търсене в директория. Като цяло използването на поделената памет от DB2 се увеличава малко, тъй като директорната информация, която е обща за агентите в базата данни се премества в поделената памет. Размерът на необходимата за кеш памет зависи от броя на записите, дефинирани във всяка директория.

## Други DB2 Connect параметри

MAXDARI и NUMDB трябва да се установят на техните минимални стойности, ако няма локална база данни на DB2 Connect работната станция. Тези параметри ще минимизират консумирането на ресурси.

AGENTPRI се прилага само с отдалечени клиенти. AGENTPRI контролира приоритетите, които Планировчикът на операционната система дава на агентите в DB2 Connect потребителски модел. На DB2 Connect потребителския модел се предоставят повече процесорни цикли, ако има по-голям приоритет (по-малко число). Така се намалява броя на процесорните цикли, които остават за другите процеси, изпълнявани върху DB2 Connect работната станция. Например може да имате DB2 Connect потребителски модел с висок приоритет и DB2 Connect модел с нисък приоритет на една и съща работна станция с различни стойности на AGENTPRI.

За всяко свързване от клиентска машина към хост или AS/400 сървър на база данни чрез DB2 Connect е необходим агент, който да работи на DB2 Connect работната станция. Задайте на MAXAGENTS стойност, която е по-голяма или равна на най-големия брой свързвания от отдалечени клиенти, които осъществяват достъп до хост или AS/400 сървър на база данни чрез DB2 Connect работна станция.

Ако решите да използвате низове за отчитане, използването на `sqlsact()` API има предимства по отношение на производителността пред метода с променливата на обкръжението DB2ACCOUNT. За повече информация вижте “Реализиране на обратно таксуване на DB2 Universal Database за OS/390” на страница 46.

Ако нямате свързан файл за SQLCODE преобразуване, можете да увеличите производителността, като използвате SQLCODE преобразуването по подразбиране или да го изключите. (Файлът за преобразуване по подразбиране е вграден в DB2 Connect библиотеката; свързан файл за преобразуване трябва да се прочете от диска, което влияе върху производителността.) Допълнителна информация за SQLCODE преобразуването потърсете в Глава 11, “Преобразуване на SQLCODE” на страница 103.

---

## Пул за връзки

DB2 Connect Enterprise Edition сървърите често осигуряват възможност за свързване към базата данни на хиляди едновременни заявки от клиенти. Установяването и прекъсването на свързванията към сървъра на базата данни може да е процес, за който са необходими много ресурси и неблагоприятно влияе върху производителността на сървъра на базата данни и DB2 Connect сървъра. Това е особено очевидно в web обкръжение, където всяко посещение на web страница може да изисква изграждането на ново свързване към сървъра на базата данни, изпълняване на запитване и прекратяване на свързването. За да се намали това натоварване, DB2 Connect Enterprise Edition използва *пул за връзки*, с който поддържа отворените връзки към базата данни в лесно достъпен пул.

## Как работи пулт за връзките

Пулът за връзките е прозрачен за приложенията, които се свързват към хоста чрез DB2 Connect. Когато приложение заяви прекъсване на връзката към хоста, DB2 Connect прекъсва входящата връзка с приложението, но запазва в пул изходящата връзка към хоста. Когато ново приложение заяви свързване, DB2 Connect използва една от връзките в съществуващия пул. Използването на вече установена връзка



намалява общото време за свързване, както и високото натоварване на процесора при свързване към хоста.

За да се използва пул за връзки, трябва да се приложи следния APAR върху DB2 за OS/390 версия 6.1:

APAR PQ33473

DB2 Connect агентите може да са в едно от следните две състояния: свободно или активно. Агент е активен, когато изпълнява работа за приложение. След като приключи тази работа, агентът преминава в състояние свободно, като чака по-нататъшна работа от същото или друго приложение. Всички свободни агенти се пазят заедно в така наречения пул на свободни агенти. Можете да конфигурирате размера на този пул с помощта на конфигурационния параметър NUM\_POOLAGENTS. Този параметър е равен на максималния брой свободни агенти, които искате да се поддържат от системата. Ако определите този параметър да е нула, това е равносилно на изключване на функцията за пул за връзки.

DB2 Connect не установява свързване към базата данни, преди да получи първата заявка от отдалечен клиент. Независимо от това ако искате, можете да попълните пула от свободните агенти, преди някой клиент да отправи заявка. Пултът може да се запълни при стартирането с помощта на конфигурационния параметър NUM\_INITAGENTS. Този параметър определя колко свободни агенти трябва да се създадат при стартирането. Тези свободни агенти първоначално няма да имат свързвания към хост сървър на базата данни.

Когато клиент заяви свързване към хоста, DB2 Connect ще опита да вземе агент сред тези в пула, които имат връзка към хост сървър на базата данни. Ако не успее, ще се опита да намери достъпен агент в пула със свободни агенти. Ако пултът е празен, DB2 Connect ще създаде нов агент.

Можете да контролирате максималния брой агенти, които може да са едновременно активни, с помощта на конфигурационния параметър MAX\_COORDAGENTS. Ако се надвиши този брой, новите свързвания няма да се изпълнят, а ще се върне грешка с код sqlcode SQL1226. (Този код означава, че е надвишен максималният брой на едновременните изходящи свързвания.)

DB2 регистърната променлива DB2CONNECT\_IN\_APP\_PROCESS позволява на приложенията, които работят на същата машина, като DB2 Connect EE да накарат DB2 Connect да работи в рамките на процеса на приложенията, поведение по подразбиране, или приложението да трябва да се свързва към DB2 Connect EE сървър и след това свързването към хост да се изпълни в рамките на агент. За да може приложение да използва пул за връзки, свързването към хоста трябва да се направи вътре от агенти на DB2 Connect EE сървър и така DB2CONNECT\_IN\_APP\_PROCESS трябва да се установи на NO.

## DB2 Connect концентратор за връзки

Технологията на DB2 Connect *концентратор на връзки* позволява на DB2 Connect Enterprise Edition сървърите да осигурят поддръжка на хиляди потребители, които едновременно изпълняват бизнес транзакции, като драстично намалява ресурсите, необходими за S/390 хоста или AS/400 сървър на базата данни. Изпълнява тази задача, като концентрира натоварването от всички приложения в много по-малък брой свързвания към S/390 хост или AS/400 сървър на базата данни. Въпреки че този начин може да изглежда подобен на функцията на пул за връзки, описан горе, на практика е много по-усъвършенствен подход за намаляване консумирането на ресурси за много голям обем OLTP (On-line Transaction Processing) приложения.

Пулът на връзките спестява разходите по установяване на свързване, когато връзката повече не е необходима за приложение, което е приключило. С други думи едно приложение трябва да се откачи, за да може друго да използва запазеното в пул свързване.

Концентраторът на връзките от друга страна, позволява на DB2 Connect да направи свързването достъпно за приложение, веднага след като друго приложение е приключило с транзакция и не е необходимо другото приложение да се откачи. По същество свързването към сървър на базата данни и асоциираните хост и DB2 Connect ресурси се използват от приложение, само докато има активна транзакция. Веднага, след като транзакцията приключи, свързването и асоциираните ресурси са достъпни за използване от всяко друго приложение, което трябва да изпълни транзакция.

## Как се реализира концентратор на връзки

В предишните версии на DB2 Connect всяко активно приложение имаше Engine Dispatchable Unit (EDU), който управляваше свързването към базата данни, както и всички заявки от приложения. Тази EDU обикновено се разглежда като *агент координатор*. Всеки агент координатор проследяваше състоянието или контекста на приложението и EDU. Всяка EDU заема значително количество памет, когато броя на свързванията се увеличи, а заради превключването на контекста между агентите се получава допълнително натоварване.

В горната архитектура има директна взаимовръзка един–към–един между свързванията и всяка EDU. Обаче концентраторът на връзките позволява взаимовръзка много–към–един между свързванията и EDU. Това означава, че отношението между свързванията (X) спрямо EDU (Y) сега е  $X \geq Y$ .

Концентраторът на свързванията разделя агента на две части – *логически агент* и *работещ агент*. Логическите агенти представляват приложение, но без препратка към определена EDU. Логическият агент съдържа цялата информация и контролира блоковете, необходими за приложението. Ако има  $n$  приложения, свързани към сървъра, ще има  $n$  логически агенти на сървъра. Работещите агенти са физически EDU, които изпълняват заявки на приложения, но не са прикрепени постоянно към дадено приложение. Работещите агенти се асоциират към логически агенти, за да изпълнят транзакции и при приключване на транзакцията прекъсват асоциацията и се връщат към достъпния пул.

Модул, известен като *планировчик на логически агенти* присвоява работещи агенти на логически агенти. Ограничението по отношение на броя на указателите за отворени файлове на определени компютърни платформи може да наложи наличието на повече от един екземпляр на планировчика, когато броя на логическите агенти го надвиши.

## Активиране на концентратора

За да се използва концентратор на връзки, трябва да се приложи следния APAR върху DB2 за OS/390 версия 6.1:

APAR PQ33473

Конфигурационния параметър MAX\_LOGICAGENTS на мениджъра на базата данни определя максималния брой логически агенти. Можете да активирате функцията на концентратор, като зададете за стойността на MAX\_LOGICAGENTS число, по-голямо от стойността по подразбиране. Стойността по подразбиране за MAX\_LOGICAGENTS е равна на стойността на MAX\_COORDAGENTS. Тъй като всяко приложение ще има един логически агент, MAX\_LOGICAGENTS на практика контролира броя на приложенията, които могат да се свържат към потребителския

модел на базата данни, а MAX\_COORDAGENTS контролира броя на входящите свързвания, които може да са активни в даден момент. MAX\_LOGICAGENTS може да е число в обхвата от MAX\_COORDAGENTS до 64 000. Броят по подразбиране на логическите агенти е равен на MAX\_COORDAGENTS.

Редица съществуващи конфигурационни параметри се използват за конфигуриране на агенти. Тази параметри са както следва:

#### **MAXAGENTS**

Максималният брой работещи агенти.

#### **MAX\_COORDAGENTS**

Максималният брой на активни агенти координатори.

#### **NUM\_POOLAGENTS**

Размер на пула за агенти. Пултът за агенти включва неактивните агенти и свободните агенти.

#### **NUM\_INITAGENTS**

Първоначалният брой на работещите агенти в пула. Това ще са свободните агенти.

### **Поддръжка на XA транзакции**

Архитектурата на концентратора за връзки позволява на DB2 Connect да осигури тясно свързана поддръжка на XA транзакции за DB2 за OS/390 и DB2 за AS/400. Концентраторът ще асоциира работещ агент с определена XA транзакция (единичен XID), както би направил за всяка друга транзакция. Обаче ако XA транзакцията приключи с xa\_end() (край на разклонение), работещият агент няма да се освободи в общия пул. Вместо това работещият агент остава асоцииран с тази определена XA транзакция. Когато друго приложение се присъедини към същата XA транзакция, работещият агент ще се прикрепя към това приложение.

Всяко обръщение за край на транзакция ще върне агента в пула. Например, командите xa\_prepare() само за четене, xa\_rollback(), xa\_recover(), xa\_forget(), xa\_commit() или някоя XA грешка, която причинява отхвърляне на последните промени, ще върне агента в обикновения пул. Самият xa\_end() приключва само разклонението на транзакцията и това не е достатъчно за прекратяване на асоциирането с XID.

### **Примери**

1. Да разгледаме обкръжение, в което са необходими 4000 или повече едновременни свързвания. Web сървър, който използва CGI приложения или офис система с много настолни потребители може да надвишат това изискване. В тези случаи ефективността обикновено налага DB2 Connect да работи като отделен шлюз; това означава, че базата данни и DB2 Connect системата са на отделни машини.

DB2 Connect сървърът може да няма възможност да обслужва 4000 едновременно отворени свързвания към машината на базата данни. В повечето случаи броят на транзакциите, които се изпълняват в даден момент ще бъде значително по-малък от броя на едновременните свързвания. Тогава системният администратор може да увеличи максимално ефективността на системата, като настрои конфигурационните параметри на базата данни, както следва:

```
MAX_LOGICAGENTS = 4,000
MAX_AGENTS      = 1,000
MAX_COORDAGENTS = 1,000
NUM_POOLAGENTS  = 1,000
```

Концентраторът ще поддържа отворени максимално 4000 едновременни сесии, независимо, че в даден момент шлюзът може да управлява само 1000 транзакции.

2. В горния пример работещите агенти непрекъснато ще образуват и прекъсват асоциации към логически агенти. Онези агенти, които не са свободни, може да поддържат свързване към базата данни, но не участват в някоя определена транзакция, и следователно са достъпни за всеки логически агент (приложение), който заяви свързване.

Случаят с ХА транзакциите е някак си различен. За този пример можете да приемете, че се използва TP монитор с DB2 Connect шлюз и OS/390 или AS/400 база данни. Когато приложение заяви свързване, концентраторът или ще превключи неактивен агент, за да обслужи тази заявка, или ще създаде нов работещ агент. Нека да приемем, че приложението е изпратило заявка за ХА транзакция. Създава се XID за тази транзакция и се асоциира работещият агент.

Когато заявката на приложението се обслужи, се генерира `xa_end()` и се откача от работещия агент. Работещият агент остава асоцииран с идентификатора XID на транзакцията. Сега може да обслужва заявки само за транзакции с неговия асоцииран XID.

В този момент друго приложение може да направи заявка за транзакция, която не е ХА. Дори ако няма други достъпни работещи агенти, агентът, асоцииран с XID, няма да е достъпен за второто приложение. Той се разглежда като активен. За второто приложение ще се създаде нов работещ агент. Когато това второ приложение приключи своята транзакция, неговия работещ агент се освобождава в достъпния пул.

Междувременно други приложения, които са заявили транзакцията, асоциирана с XID на първия агент, може да са се закачали и откачали от този агент, който е изпълнявал заделената ХА транзакция за тях. Всяко приложение, което изпрати заявка за тази определена транзакция, ще се изпрати на този работещ агент, ако е свободен.

Работещият агент няма да се освободи в основния пул, докато приложение не генерира обръщение за приключване на транзакция (не `xa_end()`). Например, приложение може да приключи транзакция с `xa_commit()`, като в този момент работещият агент прекъсва асоциацията си с XID и се връща в достъпния пул. На този етап всяко приложение, което е изпратило заявка, може да го използва за транзакция, която е или не ХА.

## Ограничения

Има редица важни ограничения по отношение използването на концентратор на шлюз. Разгледайте следващата информация в нейната цялост, преди да се опитате да използвате концентратор за връзки във вашата система.

- Концентраторът за връзки може да се използва само от DB2 клиенти версия 7 или следваща.
- От концентратора се поддържат само DB2 за OS/390 или DB2 за AS/400 хост системи.
- Концентраторът разчита на TCP/IP протокол, за да установи входящи свързвания от локалните към отдалечените клиенти. Само входящите свързвания, които използват TCP/IP или Local (IPC) ще могат да се възползват от пула с изходящи връзки. Концентраторът ще приеме свързвания чрез друг комуникационен протокол, като SNA, но няма да можете да използвате неговите функции за ХА концентриране с този вид свързване.

- Не трябва да използвате статичния оператор SET в приложенията на клиента, ако концентраторът е активиран на шлюза. DB2 няма да върне грешка, ако използвате статичен оператор SET, но ще има неблагоприятно влияние върху вашето приложение и всички други приложения, които използват съвместно същата изходяща връзка.
- За операторите SET се поддържа само незабавно изпълнение.
- Ако декларирате глобални временни таблици, те трябва изрично да се затварят при приключване на транзакция или разклонение. Ако не успеят да се затворят таблиците, може да се получи грешка при следващи транзакции.
- За поддръжка на XA тясно свързани транзакции, при свързване към хоста всички приложения, които участват в една и съща XA транзакция, трябва да използват един и същи шлюз.
- Само приложенията, които приключват всички използвани указатели за граница на транзакция, могат да се възползват от концентратора. Транзакциите, които не затварят използваните указатели, ще преминават, но за тях ще се присвои отделен работещ агент и следователно няма да могат да използват пълното множество от функции на концентратора.
- При генериране на свързането всички приложения, участващи в една и съща XA транзакция, трябва да имат един и същи CCSID и да използват един и същи потребителски идентификатор.
- Ако изходящо свързване е установено да поддържа дву-фазово свързване, тогава неговият агент може да се използва само за поддръжка на дву-фазови свързвания. Аналогично, агенти, установени да поддържат едно-фазово свързване, могат да поддържат само едно-фазови свързвания.
- Типът разпознаване DCS\_ENCRYPT няма да работи с концентратора във версия 7.
- Концентраторът поддържа само динамичен SQL от интерфейс за команден ред. Ще се отхвърлят заявки от приложения с вграден динамичен SQL. Приложенията ви трябва така да се променят, че или да използват статичен SQL, или да използват CLI за операторите от динамичен SQL.

## Настройка на базата данни

Производителността на системата ще зависи от производителността на базата данни на хоста или AS/400 сървъра на базата данни.

Различните системи за управление на базата данни имат различни характеристики по отношение на производителността. Например, SQL оптимизаторите на различните системи може да се държат различно с едно и също приложение. Повече информация за производителността потърсете в документацията за вашия хост или AS/400 сървър на база данни.

При DB2 Universal Database за AS/400 може да имате възможност да повишите производителността, като използвате опциите при свързване защита при четене на незаписани промени (UR – uncommitted read) или без записване на промените (NC – no commit), за да избегнете записването в журналите.

**Забележка:** Когато използвате UR, данните извън журнала могат само да се четат, но не и да се обновяват и то само ако параметърът за създаване на блокове е установен на ALL.

В зависимост от сървъра на приложения и степента на заключване, която осигурява, използваното ниво на изолация за запитването или приложението може да има значителен ефект върху производителността.

Базата данни трябва да има съответното ниво за нормализация, ефективно използване на индекси и подходящо заделяне на пространството на базата данни. Освен това производителността може да се повлияе от типовете данни, които използвате, както е описано в следващите раздели.

## Настройка на DB2 за OS/390

OS/390 V1R3 е минималното изискване за TCP/IP поддръжка. OS/390 V2R5 или следващи се препоръчват.

Помощното средство за разпределени данни (DDF – Distributed Data Facility) отговаря за свързването на разпределените приложения към DB2 за OS/390. DDF трябва да се настрои като сървър на приложения. За да направите това, можете да вмъкнете LU името на отдалечената система в таблицата SYSIBM.LUNAMES или да вмъкнете стойностите LUNAME, SYSMODENAME, USERSECURITY, ENCRYPTPSWDS, MODESELECT и USERNAMES в таблицата SYSIBM.SYSLUNAME. След това обновете DDF до Boot Strap Data Set (BSDS). Например:

```
DDF LOCATION=LOC1,LUNAME=LU1,PORT=8000,RESPORT=8001
```

За да постигнете най-добра производителност, трябва да използвате препоръчаните приоритети в DDF адресното пространство (малко по-малко или равно на DBM1, ако сте в режим COMPAT). Използвайте RACF кеширане на оторизациите във VLF и ако можете използвайте V5 кеширане на оторизации за пакети. За повечето случаи е достатъчна стойността CACHEPAC=32768.

Тъй като DDF ще се опита да се свърже към VTAM, VTAM трябва да е активен при стартирането на DDF. Отдолу е представена примерна дефиниция на VTAM APPL:

```
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
          AUTOSSES=1, X
          SECACPT=ALREADYV, X
          SRBEXIT=YES, X
          SYNCLVL=SYNCPT, X
          MODETAB=DB2MODET, X
          VPACING=63 X
```

Можете да оптимизирате обработката на неактивна нишка в OS/390. Във V3 е разрешено да има до 10 000 едновременно свързани клиенти, а във V4 и V5 до 25 000. Във всички случаи обаче, максималният брой на едновременно активните клиенти е 1999. Всеки клиент работна станция може да остане свързан, когато не е активен; неговата нишка се поставя в неактивна верига при всяко записване на промените.

DSNZPARM параметрите CMTSTAT, CONDBAT и MAXDBAT влияят върху обработката на нишките. За да постигнете най-добра производителност, установете CMTSTAT на INACTIVE, настройте CONDBAT на максималния брой свързани DBAT, при който се осигурява добра производителност, а MAXDBAT на максималната приемлива стойност от активни DBAT.

За пълно представяне на свързването на DB2 за OS/390 в DRDA мрежа, включително конфигурирането на VTAM, се обърнете към *Приложение за свързваемост*.

## Преобразуване на данни

Когато данните се прехвърлят от едно обкръжение в друго, може да се наложи тяхното преобразуване. Това преобразуване може да засегне производителността.

Разгледайте следните платформи:

- Intel (OS/2, Windows NT или Windows 2000)
- IEEE (UNIX-базирани системи)
- System/370 и System/390 (MVS, OS/390, VM и VSE)
- OS/400.

и следните типове числени данни:

- Packed decimal
- Zoned decimal
- Integer
- Floating point.

Таблица 8 показва кога се изпълнява преобразуване.

Таблица 8. Преобразуване на данни

	Intel	IEEE	S/370 & S/390	OS/400
Packed decimal data				
Intel	Не	Не	Не	Не
IEEE	Не	Не	Не	Не
S/370/390	Не	Не	Не	Не
OS/400	Не	Не	Не	Не
Zoned decimal data				
Intel	Не	Не	Да	Да
IEEE	Не	Не	Да	Да
S/370/390	Да	Да	Не	Не
OS/400	Да	Да	Не	Не
Integer data				
Intel	Не	Да	Да	Да
IEEE	Да	Не	Не	Не
S/370/390	Да	Не	Не	Не
OS/400	Да	Не	Не	Не
Данни плаваща запетая				
Intel	Не	Да	Да	Да
IEEE	Да	Не	Да	Не
S/370/390	Да	Да	Не	Да
OS/400	Да	Не	Да	Не

Натоварването на процесора при преобразуването на еднобайтови символни данни е като цяло по-малко, отколкото при преобразуването на числени данни (където е необходимо преобразуване на данните).

Натоварването при преобразуване на данни от тип DATE/TIME/TIMESTAMP е почти същото, както при еднобайтови CHAR. Най-голямо е натоварването при преобразуване на данни от тип FLOATING (плаваща запетая). При проектирането на приложението разработчикът може да пожелае да се възползва от тези факти, когато проектира приложение, базирано на DB2 Connect.

Ако таблица в база данни има колона, дефинирана като 'FOR BIT DATA', няма да изискват никакво преобразуване данните символи, които се прехвърлят между

приложението и базата данни. Това може да се използва, когато архивирате данни на хоста или AS/400 сървър на базата данни.

## Типове символни данни

Символните данни могат да са с тип CHAR или VARCHAR. Кой тип данни е по-ефективен, зависи от типичната дължина на данните в полето:

- Ако размерът на действителните данни се различава значително, VARCHAR е по-ефективен, защото CHAR добавя допълнителни празни символи, за да допълни полето. Тези празни символи трябва да се прехвърлят през мрежата подобно на всички други.
- Ако размерът на действителните данни не се различава много, CHAR е по-ефективен, защото всяко поле VARCHAR има няколко байта с информация за дължината, които трябва да се прехвърлят.

## Мрежови настройки

Най-добрият начин да се повиши общата производителност в обкръжение на разпределена база данни, е да се отстранят забавянията от мрежата. Обичайно е за мрежовите администратори, да обмислят как мрежата да стане по-ефективна, като събират колкото е възможно повече данни между прехвърлянията. Този подход не работи за приложения, като разпределени бази данни, защото генерира забавяния в мрежата. Крайният потребител не вижда ефективността на мрежата, а само забавянията.

Повечето мрежови устройства имат параметри за забавяне и повечето от тях имат стойности по подразбиране, които са много неподходящи за разпределени бази данни. За да се повиши производителността, трябва да намерите тези параметри и ако е възможно, да ги установите на нула. Освен това трябва да се уверите, че размерът на буфера на устройството е достатъчно голям, за да предотврати повторни прехвърляния поради загуба на данни. Например, UNIX системите обикновено приемат по подразбиране стойност от 32 за размер на опашката за прехвърляне и получаване. За да постигнете по-добри резултати, определете размерът на опашката да е 150. Съответния параметър при DLC настройките е Receive Depth, който също трябва да е 150.

Параметърът IOBUF има прекалено ниска стойност на повечето сайтове. Обикновено е 500, но опитът е показал, че стойността 3992 работи най-добре, ако премествате големи количества данни, особено за канални свързвания като ESCON или 3172.

При SNA свързвания трябва да определите параметъра Mode Profile на софтуера на всяка работна станция да е 63. Като цяло стойностите за стъпката при получаване в рамките на мрежата трябва да са установени на техните най-големи стойности, така че параметрите VPACING и PACING на оператора DB2 APPL и PU/LU за работната станция при превключен основен режим трябва също да се установят на 63. Така ще може прогресивно да се увеличава количеството на потоците съобщения, преди изпращача да трябва да чака за отговор.

При LAN система размерите на прозорците DLC или LLC предаване и получаване може да имат значителен ефект върху производителността. Стойността за изпращане трябва да е установена на седем или повече, а при повечето конфигурации стойност за получаване от четири или по-малко работи най-добре.

Ако използвате Ethernet, трябва да определите размера на TCP сегмента на 1500 байта. При token ring или FDDI мрежа тази стойност трябва да е 4400 байта, а ако



използвате ESCON адаптер с TCP/IP, размерът на сегмента трябва винаги да е 4096.

Накрая за TCP/IP мрежи размерите на буфера за изпращане и получаване трябва да са определени на повече от 32768. Като цяло стойността от 65536 е най-добра.

**Забележка:** Установяването на свързване от шлюза към сървъра (изходящо свързване) е много по-скъпо от установяването на свързване от клиент към шлюз (входящо свързване). В среда, в която хиляди клиенти често се свързват и прекъсват връзката от сървъра чрез шлюза, се отделя значителна част от времето за обработка при установяване на изходящи свързвания. DB2 Connect осигурява пул за връзки през TCP/IP. Когато клиент заяви прекъсване на свързването към сървъра, шлюзът прекъсва входящата връзка с клиента, но запазва в пул изходящата връзка към сървъра. Когато нов клиент пристигне в шлюза, за да заяви свързване, шлюзът осигурява съществуваща връзка от пула, като така намалява общото време за свързване и спестява голямото натоварване на процесора върху сървъра.

Допълнителна информация за създаването на пул за връзки под DB2 потърсете в *Ръководство за администриране*.

Обобщение на методите за настройка на мрежовата производителност е представено в следващата таблица.

Какво да се търси	Пример	Настройка	Забележки
Съзнателни забавяния	Параметри за забавяне на мрежови устройства	Установени на 0.	Стойностите по подразбиране обикновено са по-високи.
Буфери	IOBUF параметър	Установен на 3992.	Особено полезно при ESCON или друг канален адаптер.
	RUSIZE	Оптимален размер – 4096.	Може да се получи най-добра производителност, ако се определи един и същи размер за RUSIZE и RQRIOLBK.
	Стъпка	VPACING, PACING и Mode Profiles трябва да се установят на 63.	Използвайте адаптивна стъпка, където е приложимо
Настройки на адаптер	Размер на опашка Предаване/Получаване	Препоръчаната стойност е 150	По подразбиране обикновено е 32.
	DLC прозорци в SNA	Определете висок размер за предаване (>7). Определете нисък размер на прозорец за получаване (например 1), тествайте и увеличете последователно, за да намерите идеалната стойност.	Всяко логическо устройство добавя забавяне. Опростете колкото се може повече мрежовата топология.

Какво да се търси	Пример	Настройка	Забележки
TCP настройки	Размери на сегменти	1500 в Ethernet, 4400 в token ring и FDDI.	ESCON адаптерите, използвани за TCP/IP трябва винаги да се установяват на 4096.
	Размери на пространства за Изпращане/Получаване	Трябва да е 64К и за двете.	По подразбиране е само 8192 при Windows. Може да се установи в Windows регистъра.

## Мрежов хардуер

Следните съображения се отнасят за хардуера:

- Скорост на мрежата или носителя за предаването

Производителността се увеличава при по-бърза среда за пренос. Например някои типични скорости на трансфер на данни са следните:

### Канал-до-канал (оптични влакна)

4.0 MB/s

### 16 Mbps LAN

2.0 MB/s

### Канал-до-канал (обикновен)

1.0 MB/s

### 4 Mbps LAN

0.5 MB/s

### Високо скоростен T1 носител (1.544 Mbps)

0.193 MB/s

### Бърза отдалечена телефонна линия 56 Kbps

0.007 MB/s

### 19.6 Kbps модем

0.002 MB/s

### 9600 bps модем

0.001 MB/s

Скоростта на трансфер на данните се ограничава от най-бавната среда за пренос по пътя до хоста или AS/400 сървъра на базата данни.

- Мрежов адаптер или комуникационен контролер

Трябва внимателно да планирате използването на паметта от мрежовия адаптер и комуникационния контролер. Освен това трябва да работите със специалист по мрежите, за да се уверите, че контролерът има възможност да поеме допълнителния трафик, генериран от DB2 Connect.

- Топология на мрежата

Ако данните преминават от LAN в LAN и от една SNA мрежа към друга SNA мрежа, разгледайте времето за пътуване. Мостовете, маршрутизаторите и шлюзовете ще увеличат необходимото време. Например намаляването на броя на мостовете, които се пресичат, намалява броя на отсечките, необходими за всяка заявка.

Физическото разстояние между възлите също трябва да се разгледа. Дори ако съобщението се прехвърля чрез сателит, времето за прехвърляне се ограничава от скоростта на светлината ( $3 \times 10^8$  m/s) и разстоянието в двете посоки между мястото на изпращане и получаване.

- Мрежов трафик

Ако ширината на лентата на мрежата се използва изцяло, ще се намали както времето за отговор, така и скоростта за трансфер на данните за отделното приложение.

Може да се получи задръстване в мрежата, когато се съберат данни в определена част от мрежата; например в стар NCP с много малък размер на буфер.

- Надеждност на мрежата

Ако нивото на грешките в мрежата е високо, скоростта на мрежата ще намалее и това ще доведе до слаба производителност поради необходимостта от повторно прехвърляне на данни.

## Конкурентно използване на системните ресурси

Производителността може да се влоши, ако много задачи в системата се борят за системните ресурси. Разгледайте следните въпроси:

- Процесорът наситен ли е? Обмислете възможността за обновяване на системата, намаляване на натоварването на системата и настройка на системата, така че да се намали натоварването от обработките.
- Паметта препълнена ли е? Обмислете разширяване на паметта, намаляване на натоварването на системата и настройка на системата, така че да намалите използваната памет.
- Комуникационният адаптер/комуникационният контролер прекалено зает ли е? Обмислете разширяване на мрежата или групиране по двойки на карти token-ring.
- Една от подсистемите прекалено заета ли е и тази подсистема на пътя на данните ли се намира?
- Работят ли на системата някои ненужни процеси или задачи? Общото правило е да не се конфигурират или стартират услуги, освен ако не се използват редовно, тъй като те ще изразходват системни ресурси.
- Няколко процеса или задачи ли използват по-голяма част от ресурсите? Могат ли да се спрат? Могат ли да се намалят приоритетите им? Могат ли така да се подобрят, че да не използват толкова много ресурси?

## Отстраняване на проблеми, свързани с производителността

Ако потребители на DB2 Connect се сблъскват с дълги времена за отговор при големи запитвания от хост или AS/400 сървъри, следните области трябва да се проверят за възможна причина на проблема с производителността:

1. При запитвания, които в резултат връщат големи блокове с данни от хоста или AS/400 сървъра (обикновено 32K данни или повече), се уверете, че конфигурационният параметър RQRIOBLK на мениджъра на базата данни е установен на 32767. Това може да се направи с помощта на процесора за обработка на команди (CLP), както е посочено:
 

```
db2 update database manager configuration using RQRIOBLK 32767
```
2. Ако се използва VTAM при свързването към хоста или AS/400 сървъра, погледнете под конфигурацията "превключен основен режим" за стойността на параметъра PACING. На DB2 Connect работна станция проверете комуникационната настройка на "LU 6.2 Mode Profile" за дефиницията на

режим IBMRDB. В тази дефиниция се уверете, че стойността за параметъра "Receive pacing window" е по-малка или равна на стойността PACING, дефинирана във VTAM. Разпространена стойност за "Receive pacing window" при DB2 Connect работна станция и "PACING" на VTAM е 8.

3. Уверете се, че максималният размер на RU, определен в дефиницията на IBMRDB режим, е установен на подходяща стойност. Препоръчваме не по-малко от 4K при свързване с помощта на Token-ring хардуер. При свързвания с помощта на Ethernet хардуер отбележете максималния размер на Ethernet фрейм от 1536 байта, който може да е ограничаващ фактор.
4. Консултирайте се с VTAM администратора във вашето обкръжение, за да се уверите, че VTAM използва "адаптивна стъпка" в LU-LU сесии с вашата DB2 Connect работна станция.

---

## Допълнителни препоръки и съвети за настройка на SNA

Този раздел съдържа допълнителни препоръки и съвети за настройка работата на SNA, които могат да се използват с DB2 Connect.

### Обща информация за производителността на DB2 Connect

Характерна черта на работата на DB2 Connect е, че основно използва процесора и изпълнява много малко входно/изходни задачи. Като цяло, колкото по-бърз е процесорът, толкова по-бързо ще работи DB2 Connect. DB2 Connect напълно се възползва от SMP процесорните конфигурации.

Бърз DB2 Connect Enterprise Edition сървър може да обработи двойката SQL заявка/отговор за по-малко от пет милисекунди, без да се отчита времето за клиента, за мрежата и за обработка от страна на хоста или AS/400 сървъра. Прост SQL израз или запитване с няколко реда данни би могъл да се завърши от начало до край за по-малко от 0.1 секунди (от клиента до хоста или AS/400 сървъра и обратно).

Когато има повече от четири или пет SQL израза в запитване, тогава използването на запомнени процедури ще помогне, за да се осигури висока OLTP производителност и да се избегне увеличаването на опитите за повторно заключване поради мрежови забавяния между SQL изразите.

Проблемите с производителността обикновено се причиняват от типа на използваното хост приложение, мрежовите характеристики, свързани с маршрутизирането и настройката и дизайна на приложенията. Обща информация, свързана с производителността на DB2 Connect, може да се намери в "Други източници на информация за производителността на DB2 Connect" на страница 131.

### Избор и настройка на Мрежовото приложение

За да се постигне възможно най-добра производителност при използването на DB2 Connect, различните типове мрежови приложения включват:

1. Карта за прикрепване на канал
2. IBM 3172 модел 3 или по-нови модели, или еквивалентни
3. IBM 2216
4. Карта Open System Adaptor (OSA-2, не OSA-1)
5. IBM 3745 с Network Control Program (NCP)
6. IBM 3174 Терминални контролери или еквивалентни

Последната опция не се препоръчва – вижте отдолу.

Препоръваният най-добър начин за свързване към хоста е да се използва карта за прикрепване на канал ESCON за AIX, Windows NT или Windows 2000. IBM 3172 модел 3 и 2216 също работят добре, но има наблюдения, че имат по-ниска производителност сравнение с ESCON.

Когато използвате AIX с ESCON карти, прилагайте PTF, свързани с MPC (Multi Path Channel). Без тези PTF, AIX SNA ESCON драйверът може да достави по-слаба производителност. Повече подробности вижте в “Поддръжка на Multi Path Channel за SNA през ESCON.” Освен това допълнителна информация може да се намери на адрес: <http://www.networking.ibm.com.cms/cmsnew01.html>

В “Как да се настройат DB2 Connect свързвания през NCP” на страница 132 ще намерите списък, в който е посочено кои параметри на комуникационния сървър, NCP и VTAM да настроите, за да оптимизирате работата на DB2 Connect. Всички специфични препоръки, които не са за NCP, са приложими за всички типове DB2 Connect и клиент/сървър свързвания.

Когато има голям брой малки транзакции, с картата OSA-2 на System/390 може да не се постигне толкова висока производителност, както с 3272 модел 3 поради нейния по-нисък показател за брой фреймове-за-секунда. Подробности за някои последни усъвършенствания вижте в “Информация за OSA-2 усъвършенствания” на страница 135.

3145 с NCP обикновено се настройва специално за съществуващия мрежов трафик. Следователно може да не се представи толкова добре при приложения клиент/сървър в база данни. Повечето проблеми с производителността на DB2 Connect се дължат на закъснение между NCP и VTAM и/или между NCP. В “Как да се настройат DB2 Connect свързвания през NCP” на страница 132 можете да намерите списък за настройка.

Като цяло препоръчваме да се избягва използването на 3174 Терминални контролери, защото техният размер на пакет (RU размер) от 256 байта е прекалено малък. Изисква се 3174 микрокод ниво C, за да се осигури независима LU поддръжка за APPC свързвания в базата данни. Някои еквиваленти OEM 3174 може да имат подобни зависимости.

## **Други източници на информация за производителността на DB2 Connect**

- Потърсете в web сайта на DB2 техническата библиотека на адрес <http://www.ibm.com/software/data/db2/library>. В World Wide Web в библиотеката DB2 Universal Database потърсете "Technotes" с ключови думи "DB2CONNECT" и "Performance", за да намерите най-новата информация с раздел за съображения за DB2 Connect.
- Освен това има редица доклади за производителността на DB2 клиенти/сървъри, които включват DB2 Connect на адрес <http://www.ibm.com/software/data/performance>

## **Поддръжка на Multi Path Channel за SNA през ESCON**

Поддръжката на Multi Path Channel (MPC) за SNA през ESCON позволява на система, върху която работи IBM eNetwork Communications Server да използва ESCON адаптер, за да създаде MPC станция за връзки към хоста. MPC обикновено е по-бърза от CDLC, защото:

1. MPC използва отделни подканални при четене и запис
2. MPC не се ограничава от размера на IOBUF. Фреймовете са 4К и може да се групират заедно в блокове.

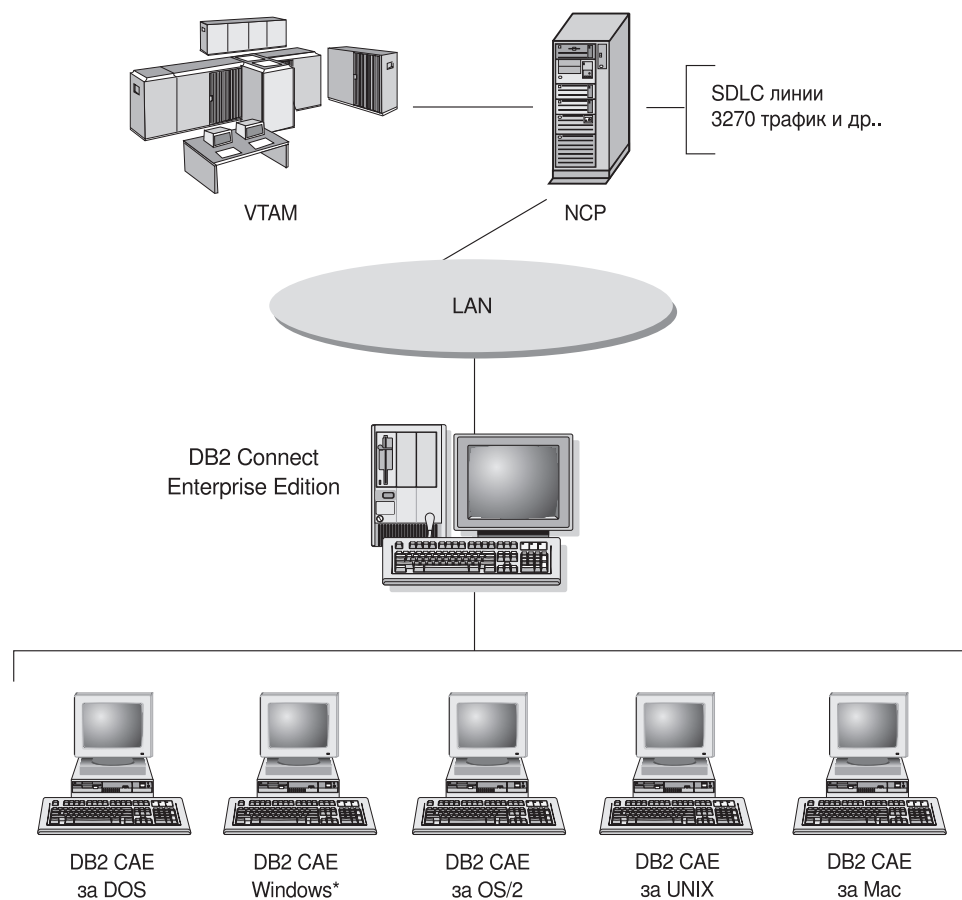
Тестовите показват, че повишението на производителността при MPC връзка достига три пъти в сравнение с връзка ESCON Channel Data Link Control (CDLC) с размер на IOBUF по-малък от 1К. AIX SNA MPC изисква ESCON и MVS VTAM V4R4 или следваща версия и компонентен код 4024 на Комуникационен сървър за AIX (5765–652). Windows NT системите трябва да използват IBM eNetwork Communications Server за Windows NT версия 6.

Следват PTF на Комуникационен сървър за AIX, необходими за MPC:

APAR #	PTF #	LPP име
IX67032	U449693	sna.books.chdoc
IX67032	U449693	sna.books.esdoc
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

## Как да се настроят DB2 Connect свързвания през NCP

Типична мрежова конфигурация може да е:



\*16 и 32-битови Windows операционни системи.

Фигура 8. Сценарий на DB2 Connect Enterprise Edition шлюз в SNA мрежа

Този сценарий се фокусира върху производителността и времето за отговор между хоста или AS/400 сървъра на базата данни към DB2 Connect Enterprise Edition шлюза и различните параметри, които може да окажат влияние.

## Критерии при настройка

Препоръчва се да направите промените в следния ред:

- 1 - DELAY на PCCU макрос\*
- 2 - DLC/LLC настройка\*
- 3 - Размер на PIU\*
- 4 - Промени в прозореца Pacing\*
- 5 - DELAY на LINE макрос\*
- 6 - MAXFRU промени
- 7 - Размери на LAN фреймове

\* Възможно е значително увеличение на производителността

**Размер на PIU (RU + 29 байта):** Размерът на RU на хоста и DB2 Connect сървъра трябва да се увеличи максимално. Това означава, че размерът на RU трябва да е достатъчно голям, за да съдържа преминаващите API (където е възможно, както данните SEND, така и RECEIVE за транзакцията), за да се намали доколкото е възможно колко пъти трябва да се прехвърля стека на програмата VTAM. Освен това размерът на мрежовия фрейм може да ограничи максималния размер на RU, ако не е желателно той да се разделя на сегменти.

Хубаво е да се настроят стойностите за размер на DB2 Connect блок (RQRIOBLK), RU и стъпка, като например  $RU * \text{стъпка} \geq RQRIOBLK$ . Например размерът по подразбиране на RQRIOBLK от 32K е подходяща стойност за повечето ситуации и за да се възползвате от това, можете да определите  $RU = 4K$  и да получите стъпка за прозорец от 8.

- Размерът на RU и стъпката се определят от таблицата за режима, която се дефинира както на DB2 Connect работна станция, така и във VTAM. Дефинициите на таблицата за режима трябва да са еднакви и на двете места.
- RQRIOBLK се определя с помощта на командата DB2 UPDATE DBM CFG.
- Размерът на мрежовия фрейм I-фрейм се определя в DLC конфигурацията на DB2 Connect работната станция и в NCP.

**Размери на стъпки за прозорец:** Прозорците за сесията и VR стъпката трябва да са максимално големи: трябва да се използва най-голямата стойност, която не причинява задръстване в мрежата или VR условия, и т.н. За тестова среда определете стъпката на 0 (без стъпка) или я определете на максималната стойност X'3F'.

**Coat-tailing стойности (DELAY):** Coat-tailing се контролира от параметъра DELAY. Параметърът DELAY в макроса PCCU контролира изходящия coat-tailing (изходящ с отнасяне към хоста). Стойността DELAY в определящия израз LINE за NCP контролира входящия coat-tailing (входящи с отнасяне към хоста).

Стойността DELAY определя колко дълго се задържа PIU в опашката (NCP или VTAM), преди да се прехвърли. Целта на това забавяне е да се увеличи вероятността през това време да пристигнат и други PIU, така че всички те да могат да се прехвърлят с една програма канал. За най-малко забавяне стойността DELAY трябва да се установи на 0. Промяната на стойността на забавянето на изходящия coat-tailing на 0 не би трябвало да има забележим ефект на хоста, с изключение на подобрената производителност спрямо изходящия трафик. Ще се постигнат и някои подобрения в производителността на входящия трафик.

Установяването на DELAY в NCP на 0 трябва да се прави с малко по-голямо внимание. Стойността може да се установи на 0, ако NCP не се претоварва и входящият трафик не се състои от значителен процент малки фреймове. Ако стойността на DELAY е 0, може значително да се подобри времето за отговор, особено при слабо натоварване или тестови обкръжения.

```
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
```

Съображенията за DELAY са документирани в *Ръководство за VTAM мрежови реализации*.



**MAXBFRU:** Стойността MAXBFRU трябва да е два или три пъти по-голяма от най-големия размер на PIU.

**Настройка на DLC/LLC слоя:** Уверете се, че са еднакви размерите на LLC2 прозореца (брой DLC прозорци за изпращане и получаване) между NCP и DB2 Connect Enterprise Edition шлюза. Това има значителен ефект, особено когато сървърът е DB2 Connect за AIX. Препоръчва се броячът за прозореца за изпращане да е по-голям от този за получаване.

Като цяло при всяко SNA свързване през Token-ring LLC2 таймерите/прозорците трябва да се оптимизират. В някои случаи тази промяна води до шест-кратно повишение на производителността и времето за отговор.

**Размери на LAN фреймове:** Максималния размер на фрейм на token ring трябва да е колкото се може по-голям.

## Информация за OSA-2 усъвършенствания

Следващата информация е получена от документа IBM WSC Flash номер 9718.

ЗАГЛАВИЕ: WSC FLASH 9718: OSA-2 УСЪВЪРШЕНСТВАНИЯ

ДОКУМЕНТ НОМЕР G023691 НЕПОВЕРИТЕЛЕН

Open Systems Adapter 2 (OSA-2) Systems Network Architecture (SNA)  
подобренията станаха достъпни по-рано от обявеното.

Усъвършенстванията са:

- o SNA/APPN усъвършенствания за OS/390, MVS/ESA, VM/ESA и VSE/ESA
  - Подобен достъп: баланс на товара, резервен достъп, препълвания
  - Подобрено свързване: поддръжка на увеличен брой физ. единици (PU) (от 255 PU за порт до 2047 PU за порт).
- o Поддръжка на ACF/VTAM за VSE/ESA мрежи

ЗАБЕЛЕЖКА: Тези подобрения не се отнасят за OSA-1.

## БАЛАНСИРАНЕ НА НАТОВАРВАНЕТО, РЕЗЕРВЕН ДОСТЪП, ПРЕПЪЛВАНЕ

---

**БАЛАНС НА ТОВАРА:** Един Medium Access Control (MAC) адрес може да бъде дефиниран за прикрепени OSA-2 SNA/APPN физически единици (PU) и когато свързванията са чрез няколко физически порта. Тази поддръжка се предлага само за source-route bridged среди (Token-Ring и FDDI). Следи се броят на сесиите, установени през порт, като натоварванията от сесиите се разпределят равномерно в рамките на еднакво конфигурираните портове.

**РЕЗЕРВЕН ДОСТЪП:** Може да се конфигурира вторична пътека между LAN работната станция и хост системата. Ако първичната пътека стане недостъпна, вторичната ще получи LAN трафика. Така се увеличава готовността на системата и се опростява управлението на мрежата.

**ПРЕПЪЛВАНЕ:** Потребителските сесии преминават през първичния OSA-2 порт, докато се достигне капацитета на сесията. Следващите потребителски сесии автоматично ще се насочат към следващия OSA-2 порт. Тъй като всички потребителски работни станции са конфигурирани идентично, администрирането на мрежата се опростява и се опростява разширението ѝ. Може да се добавят нови потребители без прекъсване.

Поддръжка на баланс на товара, резервен достъп и препълване се осигурява от PTF за OSA/SF както следва:

- |                |                             |          |
|----------------|-----------------------------|----------|
| o OS/390 и MVS | - OW20205/UW34618           | 03/31/97 |
| o VM/ESA       | - OW23952/UW37028           | 03/31/97 |
| o VSE/ESA      | - Осигурен с VSE/ESA V2.2.1 | 04/29/97 |

## УВЕЛИЧЕНА ПОДДРЪЖКА НА ФИЗИЧЕСКА ЕДИНИЦА (PU) (ЧРЕЗ OSA/SF):

---

Архитектурата е променена, за да позволи дефиниране максимум на 2047 PU на физически порт за OSA-2 Ethernet, Token-Ring и FDDI компоненти вместо досегашните 255 PU за порт. Това увеличение е достъпно за инсталираните в момента компоненти, както и при нови инсталации. Действителните възможности при свързване може да се различават в зависимост от натоварването.

Поддръжката на увеличен брой физически единици (PU) се осигурява от PTFs за OSA/SF както следва:

- o OS/390 и MVS - OW23429/UW37210 03/31/97
- o VM/ESA - OW24952/UW37028 03/31/97
- o VSE/ESA - PQ03091/UQ04224 04/29/97

Поддръжката на увеличен брой физически единици (PU) се осигурява от PTFs за ACF/VTAM както следва:

- o ACF/VTAM за OS/390 и MVS
  - VTAM 4.1 OW14043/UW24904
  - VTAM 4.2 OW14043/UW24905
  - VTAM 4.3 OW14043/UW24906
- o ACF/VTAM VM/ESA
  - VM60877/UV59834
- o ACF/VTAM VSE/ESA
  - DY44347/UD50254

### VSE/ESA - SNA ПОДДРЪЖКА

---

Поддръжката на OSA-2 и OSA/SF се осигурява чрез VSE/ESA версия 2 подверсия 2.1. Това изложение за поддръжка на VSE/ESA отговаря на Бюлетина с общи насоки, който се съдържа в Хардуерното изложение 196-194 и Хардуерното изложение 196-193 от 10 септември 1996.

OSA-2 компонентът осигурява ACF/VTAM за VSE/ESA хост приложения с директен достъп до Ethernet, Token-Ring и FDDI LAN и асинхронен режим на трансфер (ATM) Forum-съвместими LAN емуляционни мрежи.

OSA/SF се предлага като:

- o Неразделен елемент от OS/390 подверсия 1 или следваща (5645-001)
- o Отделен програмен продукт, помощно средство за S/390 OSA поддръжка версия 1 подверсия 2 за MVS/ESA 4.3 или следващи (5655-104)
- o Помощно средство за VM/ESA версия 2 подверсия 2.0 (5654-030)
- o Компонент на VSE основни функции 6.1.1 в VSE/ESA версия 2 подверсия 2.1 (5690-VSE).

### ПОВЕЧЕ ИНФОРМАЦИЯ

---

Изложения 297-043, 297-040

---

## Други източници на информация

В този раздел са изброени допълнителни източници на информация.

### Други издания

За допълнителна информация за производителността се обърнете към:

- *Сравнителни характеристики за производителността на DB2 Connect за OS/2 до DB2*
- *Ръководство за производителност на SNA сървър за AIX и SNA сървър шлюз за AIX*

### Използване на World Wide Web

В World Wide Web можете да намерите обширна информация за настройките, с които да повишите производителността на DB2 Connect, както и описание на отделни случаи и примери. Задайте на вашия Web браузър следния URL:

<http://www.ibm.com/software/data/db2/performance/>

### Допълнителни препоръки и съвети за SNA потребители

Вижте “Допълнителни препоръки и съвети за настройка на SNA” на страница 130.

---

## Глава 13. Откриване на проблеми

DB2 Connect обкръжението включва няколко софтуерни, хардуерни и комуникационни продукта. Най-добрият подход за идентифициране на проблема е чрез елиминиране и анализиране на наличните данни, за да се достигне до заключение (мястото на грешката).

Следващите теми могат да ви помогнат в процеса на определяне какъв е проблемът:

- “Други източници на информация”
- “Събиране на подходящата информация” на страница 140
- “Първоначалното свързване не е успешно” на страница 140
- “Проблеми, възникнали след първоначално свързване” на страница 141
- “Средства за диагностика” на страница 142
- “Помощна програма за трасиране (ddcstrc)” на страница 143
- “Най-разпространени проблеми с DB2 Connect” на страница 150

След като съберете съответната информация и изберете подходящата тема, преминете към посочения раздел.

---

### Други източници на информация

В този раздел са изброени допълнителни източници на информация.

#### Използване на Ръководството за отстраняване на проблеми

За допълнителна информация относно определянето на проблеми в DB2 Connect и DB2 Universal Database се обърнете към *Ръководство за отстраняване на проблеми*.

#### Използване на World Wide Web

Най-новата информация, както и съвети и препоръки за откриване на проблеми в DB2 Connect, можете да намерите в DB2 Техническа библиотека за продукти и услуги в World Wide Web:

1. Отидете на следната Web страница: <http://www.ibm.com/software/data/db2/library/>
2. Изберете връзката DB2 Universal Database.
3. Потърсете “Technotes” с ключовата дума “DDCS” или “Connect”.

#### Документация за APPC, CPI-C и SNA кодове на състояние

Документацията за APPC, CPI-C и SNA кодовете на състояние е в нов вид и се осигурява като PostScript файл и като HTML книга (само на английски).

PDF версията на тази книга може да се намери на компактдиска с DB2 изданията. HTML копието на тази книга се инсталира винаги при инсталирането на DB2 Connect, като тази книга е интегрирана в DB2 Центъра за информация.

---

## Събиране на подходящата информация

Процеса на определяне на проблема включва стесняване на възможностите и изследване какви могат да са причините. Подходяща отправна точка е да съберете подходящата информация и да определите какво знаете, какви данни не са събрани и кои пътеки можете да елиминирате. Най-малко отговорете на следните въпроси.

- Успешно ли е било първоначалното свързване?
- Хардуерът функционира ли правилно?
- Действат ли комуникационните връзки?
- Има ли някакви промени в комуникационната мрежа, които може да са направили невалидни предишни записи в директории?
- Стартирана ли е базата данни?
- Къде се намира комуникационното прекъсване – между клиент и DB2 Connect работна станция, между DB2 Connect работна станция и хост или AS/400 сървър на база данни, при всички клиенти или при един клиент?
- Какво можете да определите от съдържанието на съобщението и токените, върнати в съобщението?
- На този етап използването на средствата за диагностика ще осигурят ли някаква помощ?
- Другите машини, които изпълняват подобни задачи, работят ли правилно?
- При отдалечена задача – успешно ли щеше да се изпълни, ако беше локална?

---

## Първоначалното свързване не е успешно

Прегледайте следните въпроси и проверете дали са следвани инсталационните стъпки.

1. *Успешно ли приключи инсталирането?*
  - Налични ли са всички необходими софтуерни продукти?
  - Достатъчно ли дисковото пространство и паметта?
  - Инсталирана ли е поддръжка за отдалечен клиент?
  - Приключи ли без условия за грешка инсталирането на комуникационния софтуер?
2. *При UNIX-базираните системи – създаден ли е потребителски модел на продукта?*
  - Като root създадохте ли потребител и група, които да станат собственик на потребителския модел и sysadm група?
3. *Ако е приложимо, успешно ли беше обработена лицензионната информация?*
  - При UNIX-базираните системи – редактирахте ли файла nodelock и въведохте ли паролата, която сте получили от IBM?
4. *Конфигурирани ли са правилно комуникациите между хоста или AS/400 сървъра на базата данни и работната станция?*
  - Има три конфигурации, които трябва да се разгледат:
    - a. Конфигурацията на хоста или AS/400 сървъра на базата данни определя на сървъра средството за обработка на заявки. Системата за управление на хоста или AS/400 сървъра на базата данни има записите на системния каталог, които определят средството за обработка на заявки по отношение на място, мрежов протокол и защита.
    - b. Конфигурацията на DB2 Connect работната станция дефинира клиента за сървъра и хоста или AS/400 сървъра за клиента.

- c. Конфигурацията на клиентската работна станция трябва да съдържа името на работната станция, а комуникационният протокол трябва да е дефиниран.
  - При анализирането на проблема, когато не е изпълнено успешно първоначалното свързване, в случай на SNA свързване трябва да се провери, дали са довършени и дали са правилни всички имена на LU (логически единици) и PU (физически единици), а при TCP/IP свързване дали е определен правилен номер на порт и име на хост.
  - Както администраторите на хост или AS/400 сървър на база данни, така и мрежовите администратори имат помощни програми, с които да диагностицират тези проблеми.
5. *Имате ли необходимите права, които се изискват от системата за управление на хоста или AS/400 сървъра, за да използвате хоста или AS/400 сървъра на базата данни?*
- Разгледайте правата за достъп на потребителя, правилата за квалификаторите на таблиците и очакваните резултати.
6. *Ако се опитате да използвате процесора за обработка на команди и изпълните SQL оператори спрямо хост или AS/400 сървър на база данни, ще успеете ли?*
- Следвахте ли процедурата при свързване на процесора за обработка на команди към хоста или AS/400 сървъра на базата данни?

---

## Проблеми, възникнали след първоначално свързване

Следните въпроси се предлагат като отправна точка, за да ви помогнат да стесните възможните причини за проблема.

1. *Има ли специални или необикновени обстоятелства в момента?*
  - Това ново приложение ли е?
  - Използват ли се нови процедури?
  - Има ли някакви промени, които може да са повлияли на системата? Например, променени ли са някои софтуерни продукти или приложения, след като приложението или сценарият е изпълнен успешно за последен път?
  - При приложните програми, какъв интерфейс за приложно програмиране (API) е използван за създаването на програмата?
  - На системата на потребителя изпълнявани ли са други приложения, които използват софтуера или комуникационните API?
  - Инсталиран ли е наскоро PTF? Ако проблемът е възникнал, когато потребител се е опитал да използва компонент, който не е използвал (или зареждал) на операционната система след инсталирането, определете последното PTF ниво на IBM и заредете това ниво *след* инсталирането на компонента.
2. *Възниквала ли е тази грешка и преди?*
  - Документирано ли е как е решен проблемът при предишното възникване?
  - Кой са участниците и могат ли да осигурят информация за това какво може да се предприеме?
3. *Изследвали ли сте с помощта на командите на комуникационния софтуер, които връщат информация за мрежата?*
  - Имате ли средства, с които да проверите вашия SNA софтуер?
  - Ако използвате TCP/IP, може да получите ценна информация с помощта на TCP/IP команди и демони.

4. *Има ли информация, върната в SQLCA (SQL комуникационна област), която може да е полезна?*
  - В процедурите за обработка на проблеми трябва да са включени стъпки, в които да се проверява съдържанието на полетата SQLCODE и SQLSTATE.
  - Полетата SQLSTATE позволяват на приложните програмисти да тестват за класове от грешки, които са общи за DB2 фамилията продукти за бази данни. В мрежа от разпределени релационни база данни това поле може да осигури обща база. За допълнителна информация се обърнете към *Справочник на съобщенията*.
5. *На сървъра изпълнено ли е DB2START? Освен това се уверете, че променливата на обкръжението DB2COMM е настроена правилно за отдалечените клиентите, които имат достъп до сървъра.*
6. *Други машини, които изпълняват същата задача, могат ли да се свързват към сървъра? Може да е достигнат максималният брой клиенти, които могат да се опитват да се свържат към сървъра. Ако друг клиент прекъсне връзката си към сървъра, успява ли да се свърже клиентът, който преди това не е успявал?*
7. *Компютърът има ли правилно адресиране? Проверете дали машината е уникална в мрежата.*
8. *При отдалечено свързване предоставени ли са необходимите права на клиента? Свързването към потребителския модел може да е успешно, но предоставената оторизация да не позволява достъп на ниво база данни или таблици.*
9. *Това първата машина ли е, която се свързва отдалечено към базата данни? В разпределено обкръжение маршрутизаторите и мостовете между мрежите може да блокират комуникацията между клиента и сървъра. Например, когато използвате APPC, се уверете, че може да се установи сесия. Когато използвате TCP/IP, се уверете, че можете да изпълните PING за отдалечения хост.*

---

## Средства за диагностика

Когато срещнете проблем, може да използвате следното:

- Сервизният журнал на първи отказ, където диагностичната информация се събира и съхранява във формат, разрешен за четене. За допълнителна информация се обърнете към *Ръководство за отстраняване на проблеми*. Информация за съобщенията в журнала можете да намерите в *Справочник на съобщенията*.

- db2diag.log

Този файл се намира в /u/db2/sqllib/db2dump/db2diag.log на UNIX системи, където db2 представлява името на потребителския модел.

Този файл се намира в х:\sqllib\db2\db2diag.log на Intel системи, където х: е логическото устройство, а db2 е името на потребителския модел.

- db2alert.log (На същото място, както db2diag.log).
- Помощната програма за трасиране, както е описана в “Помощна програма за трасиране (ddcstrc)” на страница 143.
- На UNIX–базирани системи командата **ps**, която изпраща информация за състоянието на активните процеси към стандартния изход.
- На UNIX–базирани системи, основният файл, който се създава в текущата директория при възникване на сериозни грешки. Съдържа образ на паметта на терминирания процес и може да се използва, за да се определи коя функция е причинила грешката.
- На Windows NT и Windows 2000 системи използвайте Event Viewer.



Допълнителна информация за отстраняване на проблеми при TCP/IP свързвания (или други теми) можете да намерите в *Ръководство за отстраняване на проблеми* или потърсете “Technotes” в DB2 Техническа библиотека за продукти и услуги (вижте “Използване на World Wide Web” на страница 139).

---

## Помощна програма за трасиране (ddcstrc)

Помощната програма **ddcstrc** осигурява запис на данните, обменени между DB2 Connect работната станция (от името на клиент на базата данни) и системата за управление на хоста или AS/400 сървър на базата данни.

Като администратор на база данни (или разработчик на приложения) може да ви е полезно да разберете как работи този поток от данни, тъй като тези знания ще ви помогнат да определите произхода на определен проблем. Например, да кажем, че сте изпратили оператор CONNECT TO към хост или AS/400 сървър на база данни, но командата не е изпълнена успешно и сте получили код на връщане за неуспешно изпълнение. Ако разберете точно каква информация е предадена на системата за управление на хоста или AS/400 сървър, може да успеете да определите причината за грешката, дори когато информацията в кода на връщане е обща. Много от проблемите се причиняват от прости потребителски грешки.

В резултат от изпълнението на ddcstrc се представят потоците данни, обменени между DB2 Connect работната станция и системата за управление на хоста или AS/400 сървър на база данни. Данните, изпратени към хоста или AS/400 сървър, са обозначени като SEND BUFFER (изпратен буфер), а данните, получени от хоста или AS/400 сървър, са обозначени като RECEIVE BUFFER (получен буфер).

Ако полученият буфер съдържа SQLCA информация, ще се следва от форматирана интерпретация на тези данни и ще има етикет SQLCA. Полето SQLCODE на SQLCA е стойността *без преобразуване* – такава, каквато се връща от хоста или AS/400 сървър на база данни. (Повече за преобразуването вижте в Глава 11, “Преобразуване на SQLCODE” на страница 103.) Изпратеният и полученият буфер се подреждат от най-старото до най-последното в рамките на файла. Всеки буфер има:

- Идентификаторът на процеса
- Етикет SEND BUFFER, RECEIVE BUFFER или SQLCA. Първата DDM команда или обект в буфер е обозначена с DSS TYPE.

Останалите данни в изпратения и получения буфер се разделят на пет колони, състоящи се от:

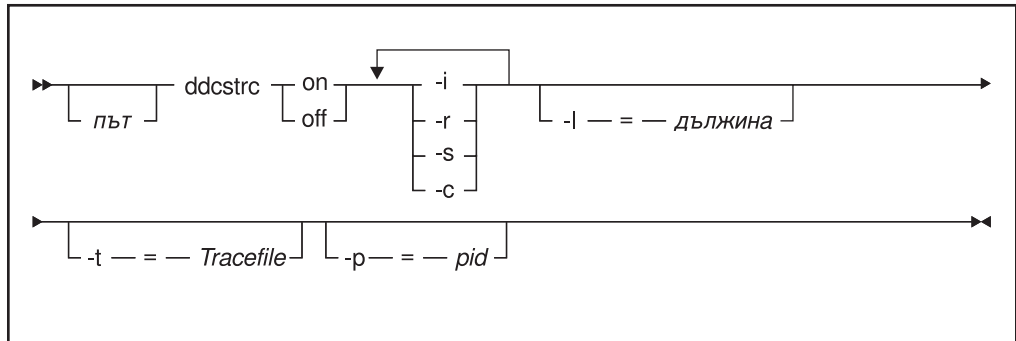
- Брой байтове.
- Колони 2 и 3 представляват DRDA потока данни, обменен между две системи, в ASCII или EBCDIC.
- ASCII представяне на колони 2 и 3.
- EBCDIC представяне на колони 2 и 3.

За допълнителна информация за DDM се обърнете към:

- *DB2 for OS/390 Reference for Remote DRDA Requesters and Servers*
- *Distributed Relational Database Reference*
- *Distributed Data Management Architecture Level 3: Reference*

## Синтаксис на трасиране

Тази команда се извиква от команден промпт на операционната система със следния синтаксис:



Фигура 9. Синтаксис на командата ddcstrc

**Забележка:** Синтаксисът на тази команда може малко да се различава в зависимост от използваната операционна система. Например / може да се използва вместо – на операционната система OS/2.

## Параметри на трасиране

- on** Включва DB2 Connect трасиране на DRDA потоци с хоста или AS/400 сървъра на база данни.
- off** Изключва DB2 Connect трасирането на DRDA потоци с хоста или AS/400 сървъра на база данни.
- i** В информацията от трасирането ще се включат и времеви индикатори.
- r** Трасира DRDA потоците данни, получени от хоста или AS/400 сървъра.
- s** Трасира DRDA потоците данни, изпратени на хоста или AS/400 сървъра на базата данни.
- c** Трасира SQLCA, получен от хоста или AS/400 сървъра на базата данни.

По подразбиране е -r, -s и -c.

**-l=дължина**

Определя размера на буфера, използван за съхраняване на информацията от трасирането. По подразбиране е 1M, а минималното е 64K.

**-t=файл на трасиране**

Определя разположението за трасирането; *файлът на трасирането* може да е името на файл или на стандартно устройство. Ако е определено име на файл, но не и пълната пътека до него, за липсващите части се използва работната пътека. Името на файла по подразбиране е ddcstrc.dmp.

**-p=pid** Трасира събитията само за този процес. Ако не е определена опцията -p, в изходния файл се записват всички процеси за потребителския модел.

**Забележка:** За отдалечен клиент *pid* може да се намери в полето Идентификатор на агент, върнато от системен монитор за базата данни.

За повече информация вижте Глава 8, “Системен монитор на базата данни” на страница 81.

## Резултат от трасиране

Помощната програма ddcstrc записва следната информация във *файла на трасирането*:

- -t
  - Тип на DRDA отговор/обект
  - Получен буфер
- -s
  - Тип на DRDA заявка
  - Изпратен буфер
- -c
  - SQLCA
- Информация за CFI-C грешка
  - Получен код на връщане от функция
  - Severity
  - Използван протокол
  - Използван API
  - Функция
  - CFI-C код на връщане
  - Номер на грешка
  - Вътрешен код на връщане.
- Информация за SNA грешка
  - Получен код на връщане от функция
  - Severity
  - Използван протокол
  - Функция
  - Име на LU партньор
  - Номер на грешка.
- Информация за TCP/IP грешка
  - Получен код на връщане от функция
  - Severity
  - Използван протокол
  - Използван API
  - Функция
  - Номер на грешка.

### Забележки:

1. Нулева стойност на кода за приключване посочва, че командата е приключила успешно, а различна от нула стойност показва, че не е.
2. Върнатите полета се различават в зависимост от използвания API. SNA API се използва само за свързвания 2PC SPM.
3. Върнатите полета се различават в зависимост от платформата, на която работи DB2 Connect, дори и при един и същи API.
4. Ако ddcstrc изпрати резултата от своето изпълнение във файл, който вече съществува, старият файл ще се изтрие, освен ако не е забранен за изтриване.

## Анализиране на изходния файл от трасирането

В следващите страници е представен примерен резултат, който илюстрира някои DRDA потоци данни, обменени между DB2 Connect работна станция и хост или AS/400 сървър на база данни. От гледна точка на потребителя е подадена командата CONNECT TO с помощта на процесор за обработка на команди.

Фигура 10 използва DB2 Connect Enterprise Edition Версия 7 и DB2 Universal Database за OS/390 версия 5.1 чрез APPC свързване.

Фигура 11 на страница 147 използва DB2 Connect Enterprise Edition Версия 7 и DB2 Universal Database за OS/390 версия 5.1 чрез TCP/IP свързване.

```
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...@.. ase1 ...]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 .....
```

Фигура 10 (Глава 1 от 2). Примерен резултат от трасиране (APPC свързване)

```

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  D640404040400005 211501000BD00300  .@@@...!.....   0      .....}...
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

```

Фигура 10 (Глава 2 от 2). Примерен резултат от трасиране (APPC свързване)

```

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  .....

```

Фигура 11 (Глава 1 от 2). Примерен резултат от трасиране (TCP/IP свързване)

```

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  D640404040400005 211501000BD00300  .@@@@@.!.....    0 .....}...
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

```

Фигура 11 (Глава 2 от 2). Примерен резултат от трасиране (TCP/IP свързване)

Следната информация е приета в трасиранията:

- Идентификаторът на процеса (PID – process ID) на приложението на клиента
- RDB\_NAME записано в директорията на услугите за връзка на базата данни (DCS)
- DB2 Connect CCSID
- CCSID на хоста или AS/400 сървъра на базата данни.
- Системата за управление на база данни на хоста или AS/400 сървъра, с която DB2 Connect системата комуникира.

Първият буфер съдържа командите EXCSAT (Exchange Server Attributes) и ACCRDB (Access RDB), изпратени към системата за управление на хоста или AS/400 сървъра. Изпраща ги като резултат от командата CONNECT TO.

Следващият буфер съдържа отговора, който DB2 Connect е получила от системата за управление на хоста или AS/400 сървъра на базата данни. Съдържа данните от отговора на командата за обмен на атрибутите на сървъра (EXCSATRD – Exchange Server Attributes Reply Data) и съобщението, получено в отговор на командата за достъп до RDB (ACCRDBRM – Access RDB Reply Message).

## Анализиране на EXCSAT и ACCRDB

Командата EXCSAT съдържа името на работната станция на клиента, определено от обекта Име на сървър (SRBAM – Server Name), който е кодов шаблон X'116D' според DDM спецификацията. Командата EXCSAT се намира в първия буфер. В рамките на командата EXCSAT стойностите X'116DA68581A28593' (кодирани в CCSID 500) се транслират до *weasel*, веднага след като се отстрани X'116D'.

Освен това командата EXCSAT съдържа обекта Външно име – EXTNAM (External Name), който често се поставя в диагностичната информация на системата за управление на хоста или AS/400 сървъра на базата данни. Състои се от 20-байтов идентификатор на приложението, следван от 8-байтов идентификатор на процеса (или 4-байтов идентификатор на процес и 4-байтов идентификатор на нишка). Представя се с кодов шаблон X'115E' и в този пример неговата стойност е db2br\_32, допълнена с празни места и следвана от 0000BE5C. На UNIX-базиран клиент на базата данни тази стойност може да се съпостави с командата **ps**, която изпраща към стандартния изход информация за състоянието на активните процеси.

Командата ACCRDB съдържа RDB\_NAME в обекта RDBNAM, който е кодов шаблон X'2110'. Командата ACCRDB е след командата EXCSAT в първия буфер. В рамките на командата ACCRDB стойностите X'2110E2C1D56DC6D9C1D5C3C9E2C3D6' се транслират до SAN\_FRANCISCO, след като се премахне X'2110'. Това съответства на полето с име на базата данни приемник в DCS директорията.

Низът на отчитане има кодов шаблон X'2104' (вижте “Реализиране на обратно такуване на DB2 Universal Database за OS/390” на страница 46).

Кодовият набор, конфигуриран за DB2 Connect работната станция, е показан с намирането в командата ACCRDB на CCSID обекта CCSIDSBC (CCSID for single-byte characters – CCSID за еднобайтови символи) с кодов шаблон X'119C'. В този пример CCSIDSBC е X'0352', което е 850.

Ако има допълнителни обекти CCSIDDBC (CCSID for double-byte characters – CCSID за двубайтови символи) и CCSIDMBC (CCSID for mixed-byte characters – CCSID за смесени символи), съответно с кодови шаблони X'119D' и X'119E', DB2 Connect работната станция е конфигурирана да поддържа DBCS кодови страници. Тъй като примерният изходен файл не съдържа двата допълнителни кодови шаблона, работната станция не е конфигурирана за DBCS.

**Забележка:** TCP/IP потоците съдържат две нови команди: ACCSEC, използвана за достъп до Мениджъра за защита и за обмен на поддържаните механизми за защита и SECCHK, която съдържа токените за автентичност, използвани при разпознаване на крайния потребител на свързването. ACCSEC и SECCHK се появяват само за TCP/IP свързвания и се намират между EXCSAT и ACCRDB.

## Анализиране на EXCSATRD и ACCRDBRM

CCSID стойностите се връщат също така и от хоста или AS/400 сървъра на базата данни в рамките на втория буфер в съобщението за отговор на команда за достъп до RDB – ACCRDBRM (Access RDB Reply Message). Този буфер съдържа EXCSATRD, следвана от ACCRDBRM. Примерният изходен файл съдържа CCSID стойност за хоста или AS/400 сървъра на базата данни от 500 (X'01F4', SBCS CCSID).

Ако DB2 Connect не разпознае кодовата страница, която идва от хоста или AS/400 сървъра на базата данни, ще върне на потребителя SQLCODE –332 с кодовите страници на източника и приемника. Ако хоста или AS/400 сървъра на базата данни не разпознае кодовия набор, изпратен от DB2 Connect, ще върне VALNSPRM (Parameter Value Not Supported – Не поддържана стойност на параметър с DDM кодов шаблон X'1252'), който се транслира за потребителя в SQLCODE –30073.

Освен това ACCRDBRM съдържа параметъра PRDID (Product-specific Identifier – Идентификатор специфичен за продукта с кодов шаблон X'112E'). Стойността е X'C4E2D5F0F5F0F1F0'. Този шестнайсетичен низ съответства на DSN05010 в

EBCDIC. Според стандартите DSN е DB2 за MVS/ESA или DB2 Universal Database за OS/390. Версията 5.1 също е посочена. ARI е DB2 за VSE и VM, SQL е DB2 Common сървър, а QSQ е DB2 Universal Database за AS/400.

## Анализиране на следващите буфери

За допълнителна информация можете да анализирате следващите изпратени и получени буфери. Третият буфер съдържа записване на промените. Командата **commit** инструктира системата за управление на хоста или AS/400 сървъра на базата данни да запише промените в текущата единица работа. Четвъртият буфер е получен от системата за управление на хоста или AS/400 сървъра на базата данни в резултат от операции комит или ролбек. Съдържа съобщението за край на единицата работа ENDUOWRM (End Unit of Work Reply Message), което показва, че е приключила текущата единица работа. В този пример съдържа null SQLCA, обозначен от DDM кодов шаблон X'2408', следван от X'FF'. Получаването на null SQLCA (X'2408FF') показва успех (SQLCODE 0). Когато полученият буфер съдържа SQLCA (възможно е null SQLCA), ddcstrc ще представи след него форматирана интерпретация на SQLCA информацията.

Фигура 12 показва пример на получен буфер, който съдържа SQLCA за грешка и форматирано представяне на SQLCA. Този SQLCA е получен в резултат от опит да се изтрият редове в несъществуваща таблица.

```

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 240800FFFFFFFF34F4 .e.....$_.....4. ..}.....^.....4
0010 F2F7F0F4C4E2D5E7 D6E3D34000E2C1D5 .....@.... 2704DSNXOTL .SAN
0020 6DC6D9C1D5C3C9E2 C3D64040404040FF m.....@... _FRANCISCO .
0030 FFFE0C0000000000 000000FFFFFFFFF00 .....
0040 0000000000000040 4040404040404040 .....@...
0050 4040000000FC4C4 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]: FFFFFFF0C, 00000000, 00000000, FFFFFFFF, 00000000, 00000000
      SQLWARN(0->A): , , , , , , , , ,
      SQLSTATE:  42704

```

Фигура 12. Примерен получен буфер

## Най-разпространени проблеми с DB2 Connect

В този раздел са представени най-разпространените симптоми на проблеми при свързване, срещани при използването на DB2 Connect. При всеки случай получавате:

- Комбинация от номер на съобщение и код на връщане (или специфичен за протокола код на връщане), свързан с това съобщение. Всяка комбинация от



съобщение и код на връщане има отделно заглавие, като заглавията са подредени по номера на съобщението и след това по кода на връщане.

- Осигурява се симптом, обикновено във вид на списък с примерни съобщения.
- Предлага се решение, като се посочва вероятната причина за грешката. В някои случаи може да се предложи повече от едно решение.

#### **Забележки:**

1. Обърнете се към ръководството Бърз старт за вашия продукт и най-новите Последни бележки, за да видите най-актуалната информация за препоръките относно софтуерните корекции.
2. При комбинациите от съобщение и код на връщане, специфични за APPC комуникациите, може да се посочи и SNA код на състояние. Засега информацията за SNA кода на състояние, свързан с определено съобщение, трябва да се получи от SNA подсистемата.

Понякога SNA кодове на състояние може да се видят при преглед на системните журнали. Дали това е конкретния случай зависи от използваната SNA подсистема, а в някои ситуации може да се наложи да създадете отново проблема, след като активирате SNA трасирането, за да получите информация за кодовете на състояние.

3. Терминът шлюз се отнася за DB2 Connect Enterprise Edition.

## **SQL0965 или SQL0969**

### **Симптом**

Съобщенията SQL0965 и SQL0969 може да се генерират с редица различни кодове на връщане от DB2 Universal Database за AS/400, DB2 Universal Database за OS/390, DB2 за MVS/ESA и DB2 за VM & VSE.

Когато срещнете някое от двете съобщения, трябва да погледнете оригиналния SQL код в документацията на сървъра на базата данни, генерирал съобщението.

### **Решение**

SQL кодът, получен от хост базата данни не може да се преведе. Коригирайте проблема на базата на кода за грешка и след това отново предайте неуспешната команда.

## **SQL1338 по време на CONNECT**

### **Симптом / Причина**

Не е дефинирано или не е дефинирано правилно името на символно предназначение.

Например това може да се случи, когато се използва APPC възел, а името на символно предназначение, определено в DB2 директорията на възлите, не отговаря на CRI-C запис в конфигурацията на локалната APPC комуникационна подсистема.

Друга причина може да е, че на компютъра е инсталиран повече от един SNA стек. Може да се наложи да проверите PATH и LIBPATH, за да се уверите, че най-напред е посочен стекът, който искате да използвате.

## Решения

1. Проверете дали името на CPIC Side информационния профил, определено в запис на DB2 директорията на възлите, отговаря на SNA конфигурацията (използването на главни и малки букви е от значение).
2. Може да се наложи да проверите PATH и LIBPATH, за да се уверите, че най-напред е посочен SNA стекът, който искате да използвате.

## SQL1403N по време на CONNECT

### Симптом

SQL1403N Посоченото име на потребител и/или парола са неправилни.

### Решение

1. На DB2 Connect работната станция не е разпознат потребителя. Проверете дали автентичността на потребителя трябва да се проверява на DB2 Connect работната станция.

Ако това е така, при необходимост проверете дали е въведена правилната парола в оператора CONNECT.

В противен случай записът в системната директория на базата данни трябва да е въведен неправилно със стойността AUTHENTICATION SERVER (това е стойността по подразбиране, ако изрично не е определен тип за разпознаване). В този случай трябва да запишете отново записа, като използвате тип AUTHENTICATION DCS или CLIENT.

2. Не е достъпна парола, която да се изпрати на сървъра на базата данни приемник. Ако записът в системната директория на базата данни е със стойност AUTHENTICATION DCS, тогава трябва да се предаде парола от DB2 клиент към сървъра на базата данни приемник. При някои платформи, например AIX, паролата може да се получи, само ако е осигурена с оператор CONNECT.

## SQL5043N

### Симптом

Поддръжката на един или повече комуникационни протокола не успя да се стартира успешно. Основните функции на мениджъра на базата данни обаче са стартирани успешно.

Вероятно TCP/IP протоколът не е стартирал на DB2 Connect шлюза. Преди това може да е имало успешно свързване на клиента.

Ако `diaglevel = 4`, тогава `db2diag.log` може да съдържа подобен запис:

```
1997-05-30-14.09.55.321092 Instance:svtdbm5 Node:000
PID:10296(db2tcpdm) Appid:none
common_communication sqlcctcpconnmgr_child Probe:46
DIA3205E адреса на сокет "30090", конфигуриран в TCP/IP сервисния файл и
необходим за TCP/IP поддръжката на сървъра се използва от друг процес.
```

### Решение

Това предупреждение е симптом, че DB2 Connect, който действа като шлюз за отдалечени клиенти, има проблеми при поддържането на един или повече

комуникационни протоколи. Тези протоколи може да са TCP/IP, APPC и други и обикновено съобщението посочва, че не е конфигуриран правилно един от дефинираните на DB2 Connect комуникационни протоколи.

Често причината може да се състои в това, че не е дефинирана или е дефинирана неправилно променливата на профила DB2COMM. Като цяло проблемът се получава в резултат от несъответствие между променливата DB2COMM и имената, дефинирани в конфигурацията на мениджъра на базата данни (например svcename, pname или trname).

Възможен сценарий е да сте имали преди това успешни свързвания и след това да получите съобщението за грешка SQL5043, без да сте променили конфигурацията. Ако се използва TCP/IP протокол, това може да възникне, когато отдалечената система неправилно прекрати свързването поради някаква причина. Когато това се случи, на клиента свързването може все още да изглежда, като че ли съществува и може да стане възможно да възстановите свързването без други интервенции, като се използват показаните по-долу команди.

Най-вероятно един от клиентите, свързващи се към шлюза, все още има указател на TCP/IP порта. На всеки компютър клиент, който е свързан към шлюза:

1. db2 terminate
2. db2stop

## SQL30020

### Симптом

SQL30020N Изпълнението не бе успешно поради разпределена протоколна грешка (Distributed Protocol Error), която ще повлияе на успешното изпълнение на следващите команди и SQL оператори.

### Решения

При тази грешка трябва да се обърнете към сервиз.

Проверете db2dump директорията за ffdc dump (pid.000). След това форматирайте този dump файл с db2fdump и в получения файл потърсете "ERROR". Тук може да е посочено MVS ABEND. В този случай проверете MVS конзолата за допълнителна информация и проверете кода abend в DB2 Ръководството за MVS съобщения и кодове.

## SQL30060

### Симптом

SQL30060N "<ID-за-оторизация>" не притежава необходимите права, за да изпълни операцията <операция>".

### Решение

При свързване към DB2 за MVS или DB2 за OS/390 не са обновени правилно комуникационните таблици на базата данни (CDB). Обърнете се към:

- DB2 Connect: Бърз старт или
- DB2 Приложение за свързваемост

## SQL30061

### Симптом

Свързване към грешен хост или AS/400 сървър на база данни – не е намерена база данни приемник.

### Решение

Може да е определено грешно име на сървър на базата данни в записа на DCS директорията. Когато това се случи, към приложението се връща SQLCODE –30061.

Проверете DB2 възела, базата данни и записите в директорията за DCS. Полето с името на базата данни приемник в записа на директорията за DCS трябва да съответства на името на базата данни, което зависи от платформата. Например за DB2 Universal Database за OS/390 база данни използваното име трябва да е същото като посоченото в полето "LOCATION=locname" в Boot Strap Data Set (BSDS), което също така се осигурява и в съобщението DSNL004I (LOCATION=location), когато се стартира помощното средство за разпределени данни DDF (Distributed Data Facility). Вижте също "Концепцията база данни" на страница 4 и Глава 6, "Обновяване на директории на бази данни" на страница 63.

Ръководството DB2 Connect Бърз старт освен това съдържа примери, които показват как да обновите DB2 каталозите. Вижте раздела "Обновяване на DB2 директории" във всяка глава, която описва SNA конфигурацията или вижте главата "Конфигуриране на хост или AS/400 база данни за DB2 Connect" и раздела "Конфигуриране на TCP/IP свързване".

Правилните команди за APPC или APPN възел са:

```
db2 catalog appc node <име_възел> remote <име_симв_назн> security program
db2 catalog dcs database <локално_име> as <истинско_db_име>
db2 catalog database <локално_име> as <псевдоним> at node <име_на_възел>
authentication dcs
```

Правилните команди за TCP/IP възел са:

```
db2 catalog tcpip node <име_на_възел> remote <адрес_или_име_на_хост>
server <номер_на_порт_или_име_на_услуга>
db2 catalog dcs database <локално_име> as <истинско_db_име>
db2 catalog database <локално_име> as <псевдоним> at node <име_на_възел>
authentication dcs
```

След това за да се свържете към базата данни, изпълнявате:

```
db2 connect to <псевдоним> user <име_на_потребител> using <парола>
```

## SQL30073 с код на връщане 119C при CONNECT

### Симптом

Генерира се съобщение SQL30073 с код на връщане 119C. Това се получава, когато сървърът на базата данни приемник не поддържа кодовата страница, използвана от DB2 клиента (който преминава през DB2 Connect). Кодовата таблица се получава от конфигурацията на операционната система, в която работи DB2 клиента.

Вижте *Ръководство за администриране* за допълнителна информация.

## Решение

Често този проблем може да се разреши, като се инсталира корекция в сървъра на базата данни приемник. Свържете се със съответната сервизна организация, за да получите и приложите корекцията, която може да ви препоръчат при този симптом.

Като временно решение потребителят може да замени кодовата страница по подразбиране, като настрои променливата на обкръжението DB2CODEPAGE. Проверете кода на географското положение или определете DB2CODEPAGE=850.

На UNIX платформи потребителят може да има възможност да превключи към различна кодова страница, ако въведе различна стойност за променливата на обкръжението LANG.

## SQL30081N с код на връщане 1

### Симптом

Симптомът е следното съобщение плюс SNA код на състояние:

```
db2 connect to <име на база данни> user <id на потребител>  
Въведете парола за <id на потребител>:  
SQL30081N Открита е комуникационна грешка.  
Използван комуникационен  
протокол: "APPC". Използван комуникационен API: "CPI-C".  
Място, където  
е открита грешката: "". Комуникационната функция, открила  
грешката:  
"smallc". Кодове за грешка, специфични за протокола: "1", "*",  
"0x10030021".  
SQLSTATE=08001
```

### Решение(я)

В този пример кодът на състояние е 10030021.

Най-често срещаните кодове на състояния, свързани с това съобщение за грешка, както и предлаганото решение при всеки отделен случай, са както следва:

1.

SQL30081N с код на връщане 1 и sna код на състояние 0877002C

Определено е грешно мрежово име.

2.

SQL30081N с код на връщане 1 и SNA код на състояние ffff0003

Определен е грешен MAC адрес или не е активна SNA връзката.

3.

SQL30081N с код на връщане 1 и SNA код на състояние 10030021

Има несъответствие между тип на LU.

4.

SQL30081N с код на връщане 1 и SNA код на състояние 084B6031

MAXDBAT в DSNZPARM (в DB2 за MVS или DB2 за OS/390 хост) е установен на 0

Други предположения:

1. При създаването на профила на локалната LU, дефинирайте LU като LU по подразбиране. Например в панела със списъка с компонентите на SNA в CM/2 направете едно от следните:
  - Маркирайте полето 'Използвай тази локална LU като псевдоним на локална LU по подразбиране' или
  - Определете профила или променливата на обкръжението APPCLLU на система шлюз DB2 Connect Enterprise Edition да бъде името на локалната LU. Например на OS/2 системи можете да направите това, като редактирате CONFIG.SYS, а на Windows NT системи чрез Control Panel.
2. Проверете дали SNA е стартирал на DB2 Connect шлюза
3. Ако използвате DB2 за MVS или DB2 за OS/390, проверете дали е стартирано адресното пространство на Помощното средство за разпределени данни DDF (Distributed Data Facility) и дали работи DB2.

## SQL30081N с код на връщане 2

### Симптом

Получено е съобщение SQL30081N с код на връщане 2 и SNA код на състояние 08120022.

### Решение

Параметърът NUMILU на NCP (хост страната на връзката) може да е установен на стойността по подразбиране (0). Проверете това. При необходимост променете NCP дефиницията и опитайте отново, след като влезе в сила направената промяна.

## SQL30081N с код на връщане 9

### Симптом

Симптомът е следното съобщение (SNA код на състояние не е необходим в този случай):

```
db2 connect to <база данни> user <id на потребител>
SQL30081N Открита е комуникационна грешка.
Използван комуникационен
протокол: "APPC". Използван комуникационен API: "CPI-C".
Място, където
е открита грешката: "". Комуникационната функция, открила
грешката:
"cmsend". Кодове за грешка, специфични за протокола: "9", "*",
"0x10086021".
SQLSTATE=08001
```

### Решение

Проблемът е, че в DB2 Connect системата не е дефинирано правилно името на Транзакционната програма. Например, може да сте обновили SNA конфигурацията, но все още да не сте я проверили на DB2 Connect шлюза. За допълнителни подробности се обърнете към ръководствата *DB2 Connect Enterprise Edition за OS/2 и Windows – Бърз старт* или *DB2 Connect Personal Edition: Бърз старт*.

## SQL30081N с код на връщане 10

### Симптом

Симптомът е следното съобщение (SNA код на състояние не е необходим в този случай):

```
SQL30081N Открита е комуникационна грешка.  
Използван комуникационен  
протокол: "APPC". Използван комуникационен API: "CPI-C".  
Място, където  
е открита грешката: "". Комуникационната функция, открила  
грешката:  
"cmrcsv". Кодове за грешка, специфични за протокола: "10", "*", "*".  
SQLSTATE=08001
```

### Решение

Проверете дали DB2 е инсталирана правилно.

Ако използвате DB2 Connect за OS/2 шлюз, може да видите следното, ако TP името не е дефинирано правилно:

```
Кодове за грешка, специфични за протокола: "10", "*", "0x084C0000".  
SQLSTATE=08001
```

Например в CM/2 в този случай трябва да е дефинирано както следва:

```
Име на транзакционна програма = 'tpname' (дефинирано от потребителя)  
Пътека и име на файл на OS/2 програма = notused
```

и (на следващия екран за CM/2 конфигурация)

```
Тип представяне - фонов режим  
Тип работа - На опашка, предварително заредени оператори
```

## SQL30081N с код на връщане 20

### Симптом

```
SQL30081N Открита е комуникационна грешка.  
Използван комуникационен  
протокол: "APPC". Използван комуникационен API: "CPI-C".  
Място, където  
е открита грешката: "". Комуникационната функция, открила  
грешката:  
"xcstp". Кодовете за грешка на протокола са: "20", "*", "*". SQLSTATE=08001  
SQLSTATE=08001
```

### Решение

Проверете дали SNA подсистемата е стартирала на DB2 Connect

## SQL30081N с код на връщане 27

### Симптом

Получено е съобщение SQL30081N с код на връщане 27 и SNA код на състояние 800Axxxx.

## Решение

Прекалено голям VTAM PIU (Path Information Unit).

## SQL30081N с код на връщане 79

### Симптом

SQL30081N Открита е комуникационна грешка.  
Използван комуникационен  
протокол: "TCP/IP". Използван комуникационен API: "SOCKETS".  
Място, където  
е открита грешката: "". Комуникационната функция, открила  
грешката:  
"connect". Кодове за грешка, специфични за протокола: "79", "\*", "\*".  
SQLSTATE=08001

### Решение(я)

Тази грешка може да възникне в случай, че отдалечен клиент не успее да се свърже към DB2 Connect шлюз. Освен това може да възникне при свързване от DB2 Connect шлюз към хост.

1. Променливата на профила DB2COMM може да е настроена неправилно на DB2 Connect шлюза. Проверете това. Например командата `db2set db2comm=tcpip` трябва да се появи в `sqllib/db2profile`, когато DB2 Extended Enterprise Edition работи на AIX.
2. Може да има несъответствие между името на TCP/IP услуга и/или номера на порт в спецификациите на DB2 клиента и DB2 Connect шлюза. Проверете записите във файловете на TCP/IP услугите и на двете машини.
3. Проверете дали DB2 е стартирана на DB2 Connect шлюза. Въведете стойност 4 за `diaglevel` на конфигурацията на Мениджъра на базата данни, като използвате командата:

```
db2 update dbm cfg using diaglevel 4
```

След като спрете и рестартирате DB2, погледнете във файла `db2diag.log`, за да проверите дали са стартирали DB2 TCP/IP комуникациите. Би трябвало да видите резултат, подобен на показания:

```
1998-02-03-12.41.04.861119 Instance:svtdbm2 Node:00  
PID:86496(db2sysc) Appid:none  
common_communication sqlcctcp_start_listen Probe:80  
DIA3000I Поддръжката на "TCP/IP" протокол е стартирана успешно.
```

## SQL30081N със специфичен за протокола код за грешка 10032

### Симптом

SQL30081N Открита е комуникационна грешка.  
Използван комуникационен  
протокол: "TCP/IP". Използван комуникационен API: "SOCKETS".  
Място, където  
е открита грешката: "9.21.85.159". Комуникационна функция,  
открила  
грешката: "send". Кодове за грешка, специфични за протокола: "10032",  
"\*, "\*".  
SQLSTATE=08001



## **Решение**

Това съобщение за грешка може да се получи, когато се опитвате да се откачите от машина, където TCP/IP комуникациите вече са прекъснати. Отстранете проблема с TCP/IP подсистемата.

На повечето машини начинът да се коригира проблема, е просто да рестартирате TCP/IP протокола за машината. Понякога може да е необходимо да се рециклира цялата машина.



---

## Част 3. Приложения



---

## Приложение А. Функции, характерни и за предишните версии

Като започва с последната версия и подверсия, този раздел предоставя обобщение на усъвършенстванията, въведени във всяка от тях.

---

### DB2 Connect версия 6 подверсия 1

DB2 Connect версия 6.1 включваше следните подобрения:

- Използване на TCP/IP комуникационен протокол
- Двухазов протокол за записване на промените (двухазов комит)
- Многоредови запомнени процедури
- DCE защита
- Поддръжка на DCE Cell Directory и хост системи
- Подобрено предупреждаване при неуспех поради защита
- Подобрено използване на System/390 SYSPLEX
- Оптимизиран достъп до каталози за ODBC и JDBC приложения
- Нови BIND опции
- Поддръжка на Microsoft Transaction Server
- Опростено управление на пароли
- Подобрения в информацията за клиент
- Поддръжка на двупосочни езици
- Следене на DB2 Connect приложения
- Подобрения в двухазовия комит
- Опростено конфигуриране на DB2 Мениджъра за синхронизация
- Поддръжка на допълнителни обекти данни и типове
- DB2 Connect за персонални комуникации

---

### DB2 Connect версия 5 подверсия 2

DB2 Connect версия 5.2 включваше следните подобрения по отношение на DRDA функциите за хост и AS/400:

- Поддръжка на DCE Cell директория
- Усъвършенстване на предупреждаването при неуспех поради защитата
- Подобрено използване на System/390 SYSPLEX
- Оптимизиран достъп до каталози за ODBC и JDBC приложения
- Поддръжка на Microsoft Transaction Server
- Нови опции на BIND (DYNAMICRULES)
- Подобрения в множеството API за информация за клиенти
- Поддръжка на SQLDescribeParam за DB2 Connect
- Поддръжка на двупосочни езици
- Усъвършенстване на следенето на системата
- Усъвършенстване на поддръжката на двухазовия комит
- Опростено конфигуриране на DB2 Мениджъра за синхронизация

- Поддръжка на операционната система SCO\*\*
- Поддръжка на типове Big Integer, Large Object, Row ID и дефинирани от потребителя отделни типове данни.

---

## DB2 Connect версия 5.0

- Нови пакети, които улесняват купувачите:
  - Всеки пакет DB2 Connect Personal Edition съдържа версиите на продукта за OS/2, Windows 3.1, Windows 95 и Windows NT. Този пакет съдържа всичко, което е необходимо да започнете, включително допълнително копие на Lotus Approach.
  - Всеки пакет DB2 Connect Enterprise Edition съдържа версиите за OS/2, Windows NT и всички версии за UNIX.
- Възможности:
  - Нов ODBC драйвер от ниво 3 с много подобрения
  - Обновен JDBC драйвер за по-добра поддръжка на Java
  - Поддръжка на запомнени процедури, които връщат резултатен набор от няколко реда и няколко резултатни набора (изисква DB2 Universal Database за OS/390 версия 5.1 или следваща )
  - Вградена поддръжка за репликация
  - Стандартна възможност за свързване: можете да определите всяка опция за свързване, поддържана от хост базата данни.
  - Използване на SYSPLEX (само DB2 Connect Personal Edition; изисква DB2 Universal Database за OS/390 Версия 5.1 или следваща )
- Приложимост:
  - Нов метод на инсталиране
  - Свързването към базата данни чрез TCP/IP е много по-лесно за конфигуриране (изисква DB2 Universal Database за OS/390 версия 5.1 или следваща или DB2 Universal Database за AS/400 версия 4.2)
  - Интегрирана поддръжка на SNA архитектура, която се конфигурира само с посочване и щракане (само DB2 Connect Personal Edition)
  - Нова помощна програма за конфигуриране на хост свързвания само с посочване и щракане с мишка.
  - Много по-лесен процес за свързване на настолни клиентски системи към DB2 Connect Enterprise Edition сървъри. Клиентите могат да открият DB2 Connect сървърите и всички бази данни, които са дефинирани на всеки сървър
  - Усъвършенствани ODBC следи с подробна информация за анализиране на производителността
  - Център за управление и други средства с графичен интерфейс, които опростяват редица от задачите на DBA
- Защита:
  - DCE защита (изисква DB2 Universal Database за OS/390 версия 5.1 или следваща)
  - Възможност да изпълнява ODBC приложения, без да е необходимо да дава права на всеки потребител за базовите таблици. Сега потребителите могат да свържат своя ODBC драйвер така, че да позволи на приложенията да се изпълняват под правата на този, който е свързал ODBC драйвера.
- Производителност:
  - По-бърз достъп до DB2 каталога за ODBC приложенията

- Намален мрежов трафик:
  - Ранно деактивиране на указателите
  - Отложена подготовка
  - Намален брой байтове в SQL блок
  - Редица други подобрения в мрежовия поток
  - Поддръжка на записване в ASCII на хост системата (изисква DB2 Universal Database за OS/390 версия 5.1 или следваща)
- Възможности за свързване:
  - Поддръжка на DRDA през TCP/IP връзки към други DRDA сървъри на приложения на IBM, когато те въведат поддръжка на TCP/IP.
  - SNA архитектура през TCP/IP чрез интегрирана поддръжка на MPTN (изисква AnyNet на хост системата).
  - Поддръжка на допълнителни възможности за SNA свързване:
    - IBM Communication Server за Windows NT
    - IBM Personal Communications
- Други:
  - Възможност за активиране на транзакции през TCP/IP с двуфазов комит (изисква DB2 Universal Database за OS/390 версия 5.1 или следваща)
  - Възможност настолните приложения да участват в транзакциите с двуфазов комит, без да е необходим шлюз (само TCP/IP, изисква DB2 Universal Database за OS/390 версия 5.1 или следваща)
  - Възможност да използва DB2 Universal Database за OS/390 за допълнителна надеждност при координиране на транзакциите (изисква DB2 Universal Database за OS/390 версия 5.1 или следваща и TCP/IP)
  - Много други подобрения и коригиране на грешки, които влияят върху всички страни на производителността, надеждността и приложимостта на системата.

---

## DDCS версия 2 подверсия 4

Distributed Database Connection Services (DDCS) за Windows за един потребител версия 2.4 въвежда:

- Средство за настройка на източник на данни, което ви помага бързо и лесно да дефинирате свързванията към хост и AS/400 сървъри.
- Wall Data Rumba – предоставя необходимата комуникационна поддръжка за осъществяването на тези свързвания.
- Помощна програма DB2 Password Expiration Maintenance (DB2PEM), която ви дава възможност да промените вашата DB2 за MVS/ESA парола, без да се свързвате към TSO.
- Усъвършенствания, които повишават производителността и гъвкавостта на DB2 Connect:
  - Отложена подготовка, която повишава производителността на ODBC и другите приложения с динамичен SQL, като прикрепва заявка PREPARE към следващата заявка, вместо да я изпраща отделно.
  - Асинхронен ODBC, който подобрява достъпа до ODBC приложенията. Преди това при някои мрежови ситуации те понякога изглеждаха, като че ли са забавени, когато се обработват дълги запитвания.
  - На AIX и OS/2 поддръжка на многопоточни приложения, които дават възможност на приложенията, които не са ODBC да обслужват няколко свързвания към бази данни със съответния контекст.

---

## DDCS версия 2 подверсия 3

Сред новите характеристики на DDCS версия 2 подверсия 3.1 са:

- Двухазов комит за DRDA свързвания с помощта на LU6.2 Syncpoint Manager (SPM) на OS/2 и AIX.

Сред новите характеристики на DDCS версия 2 подверсия 3.0 са:

- Производителността на клиентските приложения може да се повиши, като се изпълняват запомнени процедури на сървъри DB2 за MVS/ESA версия 4.1 и DB2 Universal Database за AS/400 версия 3.1. Вижте “Запомнени процедури” на страница 41.
- Възможност да работи с няколко бази данни в рамките на отделна транзакция.
- Възможност за повишаване на производителността чрез конкатенация на SQL изрази. Вижте “НЕ АТОМАРЕН SQL блок” на страница 43 и “Използване на средствата за импорт и експорт” на страница 92.
- Възможност за осъществяване на обратно таксуване чрез използването на низ за отчитане. Вижте “Реализиране на обратно таксуване на DB2 Universal Database за OS/390” на страница 46.
- Възможност за използване на много нови опции за свързване на приложения към DRDA сървър на приложения. Вижте “Командата BIND” на страница 79.
- При използване на DCE директория възможност за консолидиране в централно хранилище на информацията в директорията, която е необходима на всички клиенти. Вижте Приложение D, “Използване на DCE директорните услуги” на страница 173.
- По-голяма гъвкавост при обработката на SQLCODE. Вижте Глава 11, “Преобразуване на SQLCODE” на страница 103.
- Диагностичната информация се съхранява във формат, който може да се прочете и да се консолидира на едно място (Сервизният журнал за първи отказ). За допълнителна информация се обърнете към *Ръководство за отстраняване на проблеми*.
- Променливата на средата DDCSSETP е заменена от опции на BIND и PREPARE като SQLERROR CONTINUE и по този начин се опростяват операциите.
- Въведени са различни други подобрения на производителността.



## Приложение В. Работен лист за персонализиране на директории

Използвайте този работен лист, за да персонализирате вашите директории. Вижте “Обновяване на директориите” на страница 72 или се обърнете към *Справочник на командите* за синтаксиса на командите.

<i>Таблица 9. Параметри на директорията на възлите</i>		
Параметър	Пример	Ваша стойност
Име на възел	DB2NODE или MVSIPNOD	
Име на символно предназначение (APPC node)	DB2CPIC	
Име на отдалечен хост (TCP/IP възел)	MVSHOST	
Сървър (Име на TCP/IP услуга или номер на порт)	db2inst1c (или 446)	
Тип на защита	PROGRAM за APPC възли; NONE за TCP/IP възли.	
<b>Забележки:</b>		
1. Номерът на TCP/IP порта по подразбиране за DRDA е 446		
2. Освен ако не сте сигурни, че хоста или AS/400 сървъра на базата данни поддържа SECURITY SOCKS, не определяйте SECURITY за TCP/IP възел.		

<i>Таблица 10. Параметри на DCS директория</i>		
Параметър	Пример	Ваша стойност
Име на базата данни	DB2DB	
Име на база данни приемник	NEW_YORK3	
Средство за обработка на запитвания		
Параметричен низ	" ,,,,,,LOCALDATE='\"'YYMMDD'\"'\""	

<i>Таблица 11. Параметри на Системната директория на базата данни</i>		
Параметър	Пример	Ваша стойност
Име на базата данни	DB2DB	
Псевдоним на базата данни	NYC3	
Име на възел	DB2NODE	
Разпознаване	DCS	



---

## Приложение С. Съображения за поддръжка на национални езици

DB2 Connect има следните съображения при поддръжка на национални езици (NLS):

- DB2 Connect съобщенията се превеждат на определени езици. Информация за това как да получите достъп до преведени съобщения, потърсете в книгата *Бърз старт* за вашата платформа.
- DB2 Connect поддържа голям брой езици и кодови страници. За списък от тези кодови страници се обърнете към *Ръководство за администриране*.
- Когато се прехвърлят данни между DB2 Connect и хост или AS/400 сървър на базата данна, обикновено те се преобразуват от кодовата страница на работната станция към CCSID и обратно. Вижте “Преобразуване на символни данни.”

Допълнителна информация за използването на DB2 Connect може да се намери в книгите *DB2 Connect Бърз старт*, включително:

- Формати за дата и час.
- Кои езици се поддържат от DB2 Connect Enterprise Edition и DB2 Connect Personal Edition.
- Как да персонализирате вашата DB2 Connect работна станция за вашето езиково обкръжение.
- Как да персонализирате на хост компютъра настройката Идентификатор на кодиран набор от символи (CCSID).

---

## Преобразуване на символни данни

Когато между машини се прехвърлят символни данни, те трябва да бъдат преобразувани във вид, който приемащата машина може да ползва.

Например, когато се прехвърлят данни между DB2 Connect работна станция и хост или AS/400 сървър на база данни, обикновено те се преобразуват от кодовата страница на работната станция към CCSID и обратно. Ако двете машини използват различни кодови страници или CCSID, кодовите шаблони се трансформират от едната кодова страница или CCSID в другата. Това преобразуване винаги се извършва в приемащия.

Изпратени символни данни към база данни, състояща се от SQL оператори и входни данни. Изпратени символни данни от база данни, състояща се от изходни данни. Изходни данни, които се интерпретират като двоични (например: данни от колона, декларирана с клауза FOR BIT DATA), не се преобразуват. Всички други входни и изходни символни данни се преобразуват, ако двете машини имат различни кодови страници или CCSID.

Например, ако се използва DB2 Connect за достъп до DB2 Universal Database за OS/390 или DB2/MVS данни, се случва следното:

1. DB2 Connect изпраща един SQL оператор и входни данни към OS/390 или MVS.
2. DB2 Universal Database за OS/390 преобразува данните към EBCDIC CCSID и ги обработва.
3. DB2 Universal Database за OS/390 изпраща резултата обратно към DB2 Connect работната станция.

4. DB2 Connect преобразува резултата в ASCII или ISO кодова страница и го връща на потребителя.

Таблицата, която следва, показва поддържаните преобразуванията, между кодови страници (на работната станция) и CCSID (на хоста).

За допълнителна информация относно поддържаните преобразувания на кодови страници, се обърнете към *Ръководство за администриране*.

*Таблица 12. Преобразуване на кодова страница на работна станция към CCSID на хоста*

<b>CCSID на хоста</b>	<b>Кодова страница</b>	<b>Държави</b>
037, 273, 277, 278, 280, 284, 285, 297, 500, 871, 1140–1149	437, 819, 850, 858, 860, 863, 1004, 1051, 1252, 1275	Албания, Австралия, Австрия, Белгия, Бразилия, Канада, Дания, Финландия, Франция, Германия, Исландия, Ирландия, Италия, Латинска Америка, Холандия, Нова Зеландия, Норвегия, Португалия, Южна Африка, Испания, Швеция, Швейцария, Обединеното кралство, САЩ
423, 875	737, 813, 869, 1253, 1280	Гърция
870	852, 912, 1250, 1282	Хърватска, Чехия, Унгария, Полша, Румъния, Сърбия/Черна гора (Латиница), Словакия, Словения
1025	855, 866, 915, 1251, 1283	България, СР Македония, Русия, Сърбия/Черна гора (Кирилица)
1026	857, 920, 1254, 1281	Турция
424	862, 916, 1255	Израел – вижте забележка 3 по-долу
420	864, 1046, 1089, 1256	Арабските страни – вижте забележка 3 по-долу
838	874	Тайланд
930, 939, 5026, 5035	932, 942, 943, 954, 5039	Япония
937	938, 948, 950, 964	Тайван
933, 1364	949, 970, 1363	Корея
935, 1388	1381, 1383, 1386	Народна Република Китай
1112, 1122	921, 922	Естония, Латвия и Литва
1025	915, 1131, 1251, 1283	Беларус
1123	1124, 1125, 1251	Украйна

### Забележки:

1. Кодова страница 1004 се поддържа като кодова страница 1252.
2. По принцип, данни могат да бъдат преобразувани от една кодова страница в CCSID и обратно в същата кодова страница, без промяна. Следват единствените изключения на горното правило:
  - В кодови страници с набор от двубайтови символи (DBCS) може да се изгубят данни, съдържащи дефинирани от потребителя символи.
  - При еднобайтови кодови страници, дефинирани в рамките на смесени кодови страници, и при някои по-нови еднобайтови кодови страници символите, които не съществуват в източника и в приемника, може да се трансформират в заместващи символи и те да бъдат загубени, когато данните се преобразуват обратно в първоначалната кодова страница.
3. За двупосочните езици от IBM са дефинирани специални "BiDi CCSIDS" и те се поддържат от DB2 Connect Версия 7.

Ако двупосочните атрибути на сървъра на базата данни са различни от тези на клиента, може да използвате тези специални CCSIDS, за да се справите с различията.

Обърнете се към *Ръководство за администриране* за детайли относно тези специални CCSID. Вижте информацията в последните производствени забележки на DB2 Connect Версия 7 за детайлна информация за това, как да ги настроите за връзки на DRDA хост.



---

## Приложение D. Използване на DCE директорните услуги

С помощта на DCE Cell Directory Services (CDS) можете да съхраните информация за сървъра в CDS вместо да я съхранявате на всеки клиент. CDS се поддържа за всички клиенти на DB2 Universal Database и DB2 Connect Enterprise Edition на всички платформи.

**Забележка:** Ако искате да използвате поддръжка на DCE Cell Directory Services в DB2 Connect при свързване към DB2 за MVS/ESA през SNA връзки, тогава трябва да приложите DB2 за MVS/ESA PTF UN73393, който поддържа използването на DB2DRDA като име на отдалечена транзакционна програма (RTPN).

Ако искате да използвате DCE директория, трябва да създадете следното:

- *Обект в базата данни*, който съдържа информация за база данни.
- *Обект локатор в базата данни*, който съдържа информация за свързването от отдалечени клиенти към DB2 Connect работна станция.
- *Обект с маршрутизираща информация*, който осъществява съответствието между обектите и обектите локатори в базата данни.

За всеки хост или AS/400 сървър на база данни, с който се свързвате, преди да създадете тези обекти, трябва:

- Да се уверите, че следните DCE атрибути са добавени към файла със `cds` атрибути на работната станция, от която създавате обектите.

### На AIX система

Името на файла е `/etc/dce/cds_attributes`.

### На OS/2 система

Името на файла е `x:\opt\dcelocal\etc\cds_attr`, където `x`: е името на устройството.

### На 32-битова Windows система

името на файла е `x:\root\dcelocal\etc\cds_attributes`, където `x`: е името на устройството, а `root` е името на директорията, където сте инсталирали DCE.

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

- Уверете се, че сте влезли в DCE с такива права за достъп, че да можете да създавате обекти. Следващата DCE команда може да се използва, за да влезете в UNIX или Windows 2000 система:

```
dce_login principal-id password
```

Следващата DCE команда може да се използва, за да влезете в OS/2 система:

```
dcelogin principal-id password
```

**Забележка:** Преди да се свържете към базите данни с помощта на тези обекти, трябва да конфигурирате комуникациите между хоста или AS/400 сървъра на базата данни и работните станции. Тази информация е представена в съответната книга *Бърз старт*.

---

## Създаване на обект в базата данни

Обектът в базата данни дефинира хоста или AS/400 сървъра на базата данни за DB2 Connect; винаги трябва да е дефиниран. За всеки хост или AS/400 сървър на база данни, към който се свързвате, използвайте DCE командата **cdscp create object**, за да създадете обекта в базата данни. Например:

```
cdscp create object глобално_име_на_БД
```

Добавете следните атрибути към обекта:

### **DB\_Object\_Type**

D за база данни

### **DB\_Product\_Name**

Релационната база данни. Например, DB2\_за\_MVS или DB2\_за\_OS390.

### **DB\_Native\_Database\_Name**

Името на базата данни на хоста или AS/400 сървъра на базата данни, както следва:

#### **MVS или OS/390**

Стойността LOCATION

#### **VSE или VM**

Името на базата данни

#### **OS/400**

Името на релационната база данни

### **DB\_Database\_Protocol**

DRDA

### **DB\_Authentication**

SERVER, CLIENT или DCE, както е описано в “Защита с DCE директорни услуги” на страница 179.

### **DB\_Principal**

Ако методът за разпознаване, определен с параметъра Authentication, е DCE, в този атрибут въведете DCE Principal.

### **DB\_Communication\_Protocol**

Следната информация за комуникационния протокол между DB2 Connect сървъра и хоста или AS/400 сървъра на базата данни:

- За комуникационния протокол APPC:
  1. Комуникационният протокол (APPC)
  2. Мрежовият идентификатор на хоста или AS/400 сървъра на базата данни
  3. LU името за хоста или AS/400 сървъра на базата данни.
  4. Името на транзакционната програма за свързването към хоста или AS/400 сървъра на базата данни. При DB2 за MVS/ESA определете DB2DRDA. При всички други операционни системи определете валидна стойност, която не е в шестнайсетично десетичен формат.
  5. Името на режим
  6. Типът защита, както е описано в “Защита с DCE директорни услуги” на страница 179. Например:



APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;PROGRAM

- За комуникационния протокол TCP/IP:
  1. Комуникационният протокол (TCP/IP)
  2. TCP/IP името на хоста на предназначението (за хоста или AS/400 сървър на базата данни).
  3. TCP/IP номерът на порта.
  4. Типът на свързването (дали се използва SOCKS или NONE). Това е допълнително. Ако не е определено, се използва NONE. Например, следващите са стойностите на атрибутите за комуникационния протокол TCP/IP:

tcPIP;jaguar;19713;NONE

За да създадете обект в базата данни със системна защита, можете да поставите следните инструкции във файл:

```
create object /.../cdsCell11/subsys/database/DBMVS01
add object /.../cdsCell11/subsys/database/DBMVS01 DB_Object_Type=D
add object /.../cdsCell11/subsys/database/DBMVS01 DB_Product_Name=DB2_for_MVS
add object /.../cdsCell11/subsys/database/DBMVS01 DB_Database_Protocol=DRDA
add object /.../cdsCell11/subsys/database/DBMVS01 DB_Native_Database_Name=\
NEW_YORK
add object /.../cdsCell11/subsys/database/DBMVS01 DB_Authentication=SERVER
add object /.../cdsCell11/subsys/database/DBMVS01 DB_Communication_Protocol=\
APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;PROGRAM
```

След това въведете командата:

```
cdscp < име на файл
```

**Забележка:** Във файла въведете наклонена черта (\), ако искате изразът да продължи на следващия ред.

За да създадете обект в базата данни с DCE защита, можете да поставите следните инструкции във файл:

```
create object /.../cdsCell11/subsys/database/DBMVS02
add object /.../cdsCell11/subsys/database/DBMVS02 DB_Object_Type=D
add object /.../cdsCell11/subsys/database/DBMVS02 DB_Product_Name=DB2_for_MVS
add object /.../cdsCell11/subsys/database/DBMVS02 DB_Database_Protocol=DRDA
add object /.../cdsCell11/subsys/database/DBMVS02 DB_Native_Database_Name=\
NEW_YORK
add object /.../cdsCell11/subsys/database/DBMVS02 DB_Authentication=DCE
add object /.../cdsCell11/subsys/database/DBMVS02 DB_Principal=\
/.../cdsCell11/principal_name
add object /.../cdsCell11/subsys/database/DBMVS02 DB_Communication_Protocol=\
APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;NONE
```

След това въведете командата:

```
cdscp < име на файл
```

---

## Създаване на обект локатор в базата данни

Обектът локатор в базата данни се използва за дефиниране на DB2 Connect Enterprise Edition сървър за неговите клиенти. За DB2 Connect работната станция използвайте DCE командата **cdscp create object**, за да създадете обект локатор в базата данни. Например:

```
cdscp create object глобално_име_на_обект
```

Добавете следните атрибути към обекта:

## **DB\_Object\_Type**

L за обект локатор

## **DB\_Communication\_Protocol**

Ще се нуждаете от следната информация, за да настроите всеки комуникационен протокол между DB2 Connect работната станция и отдалечените клиенти.

Следва обобщение на поддръжката на протоколи от различните платформи:

- На OS/2: APPC, IPX, NETBIOS и TCP/IP
- На Windows 32-битова операционна система: APPC, IPX, NETBIOS, NPIPE и TCP/IP
- На UNIX: APPC и TCP/IP

За APPC:

1. Комуникационният протокол (APPC)
2. Мрежовият идентификатор на DB2 Connect работната станция
3. LU името на DB2 Connect работната станция
4. Името на транзакционната програма за свързването от отдалечените клиенти
5. Името на режим
6. Типът защита, както е описано в “Защита с DCE директорни услуги” на страница 179.

За TCP/IP:

1. Комуникационният протокол (TCP/IP)
2. Името на хоста на DB2 Connect работната станция
3. Портът, използван от DB2 Connect работната станция, при свързвания от отдалечени клиенти
4. Типът на свързването (дали се използва SOCKS или NONE). Това не е задължително. Ако не е определено, се използва NONE.

За IPX/SPX:

1. Комуникационният протокол (IPX/SPX)
2. Името на файловия сървър. Използвайте \* при директно адресиране
3. Името на обекта. Използвайте междумрежовия адрес при директно адресиране.

За именувани конвейери

1. Комуникационният протокол (NPIPE)
2. Името на компютъра на DB2 Connect работната станция.
3. Името на потребителския модел

При NETBIOS:

1. Комуникационният протокол (NETBIOS)
2. NNAME за сървъра и DB2 Connect Enterprise Edition шлюза.

Например можете да поставите следващите редове във файл:

```

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=\
TCP/IP;AIX001;3700
add object /.../cdscell1/subsys/database/DBAIX01 DB_Communication_Protocol=\
APPC;SPIFNET;NYX1GW01;NYSERVER;IBMRDB;NONE

```

След това въведете командата:

```
cdscp < име на файл
```

На Windows 32-битова операционна система можете по подобен начин да определите именуван конвейер. Например:

```
add object /.../cdscell1/subsys/database/DBAIX01 DB_Communication_Protocol=\
NPIPE;име_на_компютър;име_на_потребителски_модел
```

На OS/2 можете да определите протокола в атрибута DB\_Communication\_Protocol. Например:

```

- IPXSPX;fileserv;objectname
- NETBIOS;nname

```

---

## Създаване на обект с маршрутизираща информация

Обектът с маршрутизираща информация трябва да се дефинира в DCE и се извлича от DB2 клиента. Използвайте DCE командата **cdscp create object**, за да създадете обект с маршрутизираща информация. Например:

```
cdscp create object глобално_име_на_обект
```

Добавете атрибута **DB\_Object\_Type** със стойност R.

За всеки обект в базата данни добавете по един параметър **DB\_Target\_Database\_Info** атрибут. Всеки атрибут **DB\_Target\_Database\_Info** се състои от следните параметри:

### База данни

Името на обекта в базата данни, като е включена пълната пътека. Определете \*OTHERDBS, за да включите всички други бази данни, които не са изрично посочени.

### Изходящ протокол

Протоколът на базата данни при свързвания на хост или AS/400 сървър на база данни (DRDA)

### Входящ протокол

Протоколът на база данни за свързвания на отдалечени клиенти (DB2RA),

### Разпознаване на шлюз

0 (за Не) или 1 (за Да), както е описано в “Защита с DCE директорни услуги” на страница 179.

### Параметричен низ за шлюз

Низът, съдържащ параметрите, които ще се използват на шлюза. Съдържанието е специфично за шлюза. За низове, специфични за DB2 Connect шлюз се обърнете към “DCS Директория” на страница 65.

### Локаатор на база данни

Името на обекта локаатор на базата данни, който представлява DB2 Connect работна станция.

Например можете да поставите следните редове във файл:

```

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

```

След това въведете командата:

```
cdscsp < име на файл
```

---

## Настройка на конфигурационните параметри

Обновете Конфигурацията на мениджъра на базата данни на клиента, както следва:

```

DB2 UPDATE DATABASE MANAGER CONFIGURATION USING
[DIR_PATH_NAME пътека]
DIR_OBJ_NAME обект_локатор
DIR_TYPE DCE
[ROUTE_OBJ_NAME обект_маршрутизатор]
[DFT_CLIENT_COMM протокол]
[DFT_CLIENT_ADPT 0-15]

```

където:

- *пътека* представлява пътеката по подразбиране, използвана за образуването на пълното име на базата данни приемник (по подразбиране `././subsys/database/`)
- *обект\_локатор* определя клиента в DCE пространството с имена
- `DIR_TYPE DCE` определя, че DCE директориите се използват от приложението на клиента
- *обект\_маршрутизатор* представлява името на обекта с маршрутизираща информация (например ROUTE1).
- *протокол* представлява комуникационният протокол между клиента и DB2 Connect работната станция (APPC или TCPIP при UNIX; APPC, IPXSPX, NETBIOS, или TCP/IP при OS/2; при Windows 32-битова операционна система, APPC, TCPIP, IPXSPX, NETBIOS и NPIPE).
- Адаптер на клиент по подразбиране от 0 до 15 за NETBIOS. Ако протоколът е NETBIOS и номерът на адаптера на клиента не е стойността по подразбиране 0, определете номера на адаптера на клиента.

**Забележка:** Следните променливи на обкръжението могат да заменят съответните променливи, посочени по-горе.

- DB2DIRPATHNAME може да замени DIR\_PATH\_NAME
- DB2ROUTE може да замени ROUTE\_OBJ\_NAME
- DB2CLIENTCOMM може да замени DFT\_CLIENT\_COMM
- DB2CLIENTADPT може да замени DFT\_CLIENT\_ADPT

---

## Каталогизиране на базата данни

Ако базата данни е на различна пътека от тази по подразбиране или ако искате да използвате псевдоним, който е различен от името на базата данни, можете да каталогизирате глобалната база данни. Можете да използвате командата на Процесор за обработка на команди CATALOG GLOBAL DATABASE, както следва:

```
db2 CATALOG GLOBAL DATABASE име_на_глобална_база_данни
AS псевдоним USING DIRECTORY DCE
```

Псевдонимът ще се използва от приложените програми при достъп до базата данни.

Например:

```
db2 CATALOG GLOBAL DATABASE /.../cdscell2/subsys/database/dbmvs12 AS NYC3
USING DIRECTORY DCE
```

## Защита с DCE директорни услуги

Като администратор на DB2 Connect можете да определите къде да се проверява валидността на имената и паролите на потребителите. В случая с DCE директориите правите това, като настроите следното:

- Типът защита на комуникационния протокол в обекта локатор на базата данни, представляващ DB2 Connect работната станция. Използвайте тип защита NONE.
- Типът разпознаване на обекта в базата данни.
- Типът защита на комуникационния протокол в обекта в базата данни.
- Параметърът за разпознаване на шлюза в обекта с маршрутизираща информация.

В Таблица 13 и Таблица 14 на страница 180 са показани възможните комбинации на тези стойности и къде се проверява валидността при всяка от тях. Само показаните в тази таблица комбинации се поддържат от DB2 Connect с DCE директорни услуги.

<i>Таблица 13 (Страница 1 от 2). Валидни сценарии за защита с DCE при APPC свързване</i>				
Случай	Обект сървър в базата данни		Маршрут. обект	Проверка за валидност
	Разпознаване	Защита	Разпознаване от DB2 Connect шлюз (1=true, 0=false)	
1	CLIENT	SAME	0	Отдалечен клиент (или DB2 Connect работна станция)
2	CLIENT	SAME	1	DB2 Connect работна станция
3	SERVER	PROGRAM	0	хост или AS/400 сървър на база данни
4	SERVER	PROGRAM	1	DB2 Connect работна станция и хост или AS/400 сървър на базата данни

*Таблица 13 (Страница 2 от 2). Валидни сценарии за защита с DCE при APPC свързване*

Случай	Обект сървър в базата данни		Маршрут. обект	Проверка за валидност
	Разпознаване	Защита	Разпознаване от DB2 Connect шлюз (1=true, 0=false)	
5	DCE	NONE	Няма	На DCE сървър за защита

**Забележка:** Ако отдалечен клиент е свързан към DB2 Connect Enterprise Edition работна станция шлюз чрез APPC свързване, определете тип защита NONE в DCE обекта локатор на шлюза.

*Таблица 14. Валидни сценарии за защита с DCE при TCP/IP свързване*

Случай	Обект сървър в базата данни	Маршрут. обект	Проверка за валидност
	Разпознаване	Разпознаване от DB2 Connect Enterprise Edition шлюз (1=true, 0=false)	
1	CLIENT	0	Отдалечен клиент (или DB2 Connect работна станция)
2	CLIENT	1	DB2 Connect работна станция
3	SERVER	0	хост или AS/400 сървър на база данни
4	N/A	N/A	Няма
5	DCE	N/A	На DCE сървър за защита

Всяка комбинация е описана по-подробно по-надолу:

- В първия случай валидността на името и паролата на потребителя се проверява само на отдалечения клиент. (За локален клиент валидността на името и паролата на потребителя се проверява само на DB2 Connect работната станция.)

Очаква се, че потребителите ще бъдат разпознати там, където за първи път са въвели регистрацията си. През мрежата се изпраща идентификаторът на потребителя, но не и неговата парола. Използвайте този тип защита, само ако всички работни станции клиенти имат адекватни средства за защита.

- Във втория случай валидността на името и паролата на потребителя се проверява само на DB2 Connect работната станция. Паролата се изпраща през мрежата от отдалечения клиент до DB2 Connect сървъра, но не и към хоста или AS/400 сървъра на база данни.

- В третия случай валидността на името и паролата на потребителя се проверява само на хоста или AS/400 сървъра на база данни. Паролата се изпраща през мрежата от отдалечения клиент до DB2 Connect сървъра и от DB2 Connect работната станция до хоста или AS/400 сървъра на база данни.
- В четвъртия случай валидността на името и паролата на потребителя се проверява както на DB2 Connect работната станция, така и на хоста или AS/400 сървъра на база данни. Паролата се изпраща през мрежата от отдалечения клиент до DB2 Connect сървъра и от DB2 Connect сървъра до хоста или AS/400 сървъра на база данни.

Тъй като валидността се проверява на две места, трябва да се поддържа едно и също множество от имена и пароли на потребители както на DB2 Connect сървъра, така и на хоста или AS/400 сървъра на база данни.

- В петия случай се получава DCE билет от DCE сървъра за защита.

#### **Забележки:**

1. При AIX системи всички потребители, които използват тип защита SAME, трябва да принадлежат към AIX **системната** група.
2. При AIX системи с отдалечени клиенти, потребителският модел на DB2 Connect, който работи на DB2 Connect сървъра, трябва да принадлежи на AIX **системната** група.
3. Достъпът до хоста или AS/400 сървъра на базата данни се контролира от негови собствени механизми или подсистеми; например Virtual Telecommunications Access Method (VTAM) и Resource Access Control Facility (RACF). Достъпът до защитени обекти на база данни се управлява от SQL операторите **GRANT** и **REVOKE**.





---

## Приложение Е. Средства за свързване на клиенти от предишни версии

Ако имате отдалечени клиенти от предишна версия, може да се наложи да свържете помощните програми на тези клиенти към хоста или AS/400 сървър на базата данни:

- Ако старият клиент е използван с предишна версия на DB2 Connect спрямо същия хост или AS/400 сървър на база данни, не е необходимо да изпълнявате никакви допълнителни стъпки.
- Ако старият клиент не е използван с DB2 Connect (например ако няколко OS/2 машини са свързани помежду си, но без връзка към хост или AS/400 сървър на база данни), направете следното:
  1. Ако имате клиент DB2 за OS/2 версия 1.0 или 1.2, създайте файл със списъка за свързване, който да съдържа следните редове:

```
sqlabind.bnd+
sqlueiwi.bnd+
sqluigsi.bnd+
sqluiici.bnd+
sqluiict.bnd+
sqluexpm.bnd+
sqluimpm.bnd+
sqlurexp.bnd+
sqlarxcs.bnd+
sqlarxrr.bnd+
sqlarxur.bnd
```

и копирайте всеки от тези файлове от единия от вашите клиенти на DB2 Connect работната станция.

2. Ако имате Разрешител на приложения на клиенти версия 1.0 или 1.2, създайте файл със списъка за свързване със следните редове:

```
db2ajgrt.bnd+
db2clics.bnd+
db2clpcs.bnd+
db2clpr.r.bnd+
db2clpur.bnd+
db2ueiwi.bnd+
db2uigsi.bnd+
db2uiici.bnd+
db2uiict.bnd+
db2uexpm.bnd+
db2uimpm.bnd+
db2urexp.bnd
```

и копирайте всеки от тези файлове от единия от вашите клиенти на DB2 Connect работната станция.

3. На DB2 Connect сървър свържете всеки файл със списък за свързване към всяка база данни на хост или AS/400 сървър на база данни. Генерирайте команди, подобни на:

```
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
```

където *DBALIAS*, *USERID* и *PASSWORD* се прилагат върху базата данни на хоста или AS/400 сървъра на база данни, *bindfile* представлява името на файла със списъка за свързване, а *path* показва мястото, където се намира той.

Можете да използвате опцията за предоставяне на права на командата **bind**, за да предоставите права EXECUTE на PUBLIC или на определен идентификатор на потребител или на група. Ако не използвате опцията за предоставяне на права на командата **bind**, трябва да изпълните отделно за всеки пакет командата GRANT EXECUTE (RUN).

За да разберете имената на пакетите за файловете за свързване, въведете следната команда:

```
ddcspkgn @bindfile.lst
```

---

## Приложение F. Настройка работата на CLI/ODBC приложение с помощта на ключовата дума CLISCHEMA

Този раздел съдържа нова информация, която да ви помогне да настроите работата на вашите ODBC/CLI приложения с помощта на инициализиращата ключова дума CLISCHEMA. Не съдържа обща информация за настройка производителността на мрежата или базата данни (вижте Глава 12, “Производителност” на страница 109). Ето обобщение на информацията, която следва:

- “Обкръжение приемник”
- “CLI/ODBC”
- “DB2 инициализиращата ключова дума CLISCHEMA” на страница 186
- “Предлаган подход” на страница 188
- “Допълнителни препоръки и съвети” на страница 188
- “db2ocat оптимизатор за каталози” на страница 189
- “Допълнителни източници на информация” на страница 189

---

### Обкръжение приемник

Изложената тук информация е предназначена основно за потребители на DB2 Universal Database за OS/390, като обкръжението приемник се състои от:

- CLI/ODBC приложение, изпълнявано с DB2 Universal Database клиент
- DB2 Connect версия 5 или следваща (Personal Edition или Enterprise Edition)
- DB2 Universal Database за OS/390 версия 5.1 или следваща (или DB2 за MVS/ESA освен където е посочено друго).

---

### CLI/ODBC

CLI/ODBC е SQL интерфейс за приложно програмиране, който може да се използва от вашите приложения за бази данни. Прехвърля изрази с динамичен SQL като обръщения към функции на бази данни. За разлика от вградения SQL не изисква хост променливи или предкомпилиране.

Когато приложна програма се обърне към CLI/ODBC, първото нещо е да направи SQL обръщения към някои от таблиците на системния каталог на базата данни приемник, за да получи информация за съдържанието на другата база данни. CLI/ODBC приложенията винаги се обръщат към таблиците на системния каталог по този начин. Има десет API обръщения, които могат да се използват, за да се събере информация за базата данни, към която е осъществено свързване. Тези API обръщения включват:

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns.



- SERGE.TABLES
- SERGE.COLUMNS
- SERGE.SPECIALCOLUMNS
- SERGE.TSTATISTICS
- SERGE.PRIMARYKEYS
- SERGE.FOREIGNKEYS
- SERGE.TABLEPRIVILEGES
- SERGE.COLUMNTABLES
- SERGE.PROCEDURES
- SERGE.PROCEDURESCOLUMNS.

Тези потребителски таблици трябва да са изградени от администратора на базата данни, преди да използвате CLISCHEMA.

**Забележка:** DataPropagator осигурява поддръжка за CLISCHEMA, така че администраторът на базата данни може да изпълни тази задача по три възможни начина:

1. С помощта на db2cli.exe на клиента.
2. Автоматично на сървъра с помощта на DataPropagator.
3. Ръчно на сървъра.

Следва обяснение за това как тази задача може да се изпълни на клиента.

## Помощните програми db2cli и bldschem

Помощната програма, с която да се настроят необходимите за CLISCHEMA потребителски таблици, се осигурява под формата на по-рано недокументираната команда bldschem на интерфейса за команден ред (CLI) и може да се намери като: /samples/cli/db2cli.exe. Документация за db2cli.exe, която не включва командата bldschem, може да се намери в: /samples/cli/INTCLI.DOC.

Например, за да изградите множеството от потребителски таблици, необходими за работа с CLISCHEMA='SERGE', за името на таблица STAFF, притежавано от собственика на схема (създател) USERID в базата данни SAMPLE, ще изпълните следната команда след изпълнението на db2start и след регистрирането на базата данни към ODBC/CLI:

```
db2cli < addstaff.txt
```

Където "addstaff.txt" съдържа следния скрипт:

```
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
```

В резултат от този скрипт ще се създаде множеството таблици SERGE.\* както е показано горе, с индекси, попълнени с помощта на данните в таблицата на системния каталог за таблицата USERID.STAFF. Например SERGE.TABLES ще се попълни с нов ред за всеки запис, за който има съответствие. В резултат от допълнителните обръщения bldschem, се допълват съществуващите таблици SERGE.\*, като се заменят съществуващи редове.

Накратко, синтаксисът на командата bldschem е:

```
bldschem <номер_указател> <стойност_CLISCHEMA> <собственик_схема> <име_таблица>
```

Където:

- <номер\_указател> трябва да е 1
- <стойност\_CLISCHEMA> трябва да е същата като името на схемата, определено с ключовата дума CLISCHEMA
- <собственик\_схема> е създателят на таблицата
- <име\_таблица> може да е име на потр. таблица, изглед, псевдоним, синоним или име на системна таблица.  
(Позволени са глобални символи).

Ако впоследствие изпълните следващия пример чрез db2cli.exe, към създадените в предишния пример потребителски таблици SERGE.\* ще добавите редове, които отразяват данните в таблиците на системния каталог за всяка таблица, за която FRED и BERT са собствениците на схема.

```
bldschem 1 SERGE FRED %  
bldschem 1 SERGE BERT %
```

Когато впоследствие CLI/ODBC ключовата дума CLISCHEMA се установи на SERGE, при обработката на ODBC/CLI приложения спрямо базата данни SAMPLE ще се изпълняват обръщения към множеството таблици SERGE.\* вместо към таблиците от системния каталог.

---

## Предлаган подход

В повечето производствени среди търсенето по подразбиране в таблиците на системния каталог връща огромно количество данни, така че всеки път, когато CLI/ODBC приложение отвори база данни, може да има значително забавяне. Дори на типична тестова база данни забавянето може да е от порядъка на 25 секунди.

Измерете това забавяне първоначално, без да сте използвали посоченото горе множество от CLI ключови думи, като не забравите да извадите времето за свързване и особено дългите забавяния, които може да възникнат, когато DB2 клиент генерира първото си свързване към нова база данни – често за автоматичното свързване може да са необходими няколко минути.

Как ще продължите нататък зависи от структурата на вашите данни и организация. В някои случаи може да използвате комбинация от DBNAME, SCHEMALIST и TABLETYPE, за да ограничите търсенето, използвано от определено приложение или група приложения. Например, ако производствени DBA клиенти обикновено осъществяват достъп до таблици под дадено DBNAME и схема, тогава това е лесно да се определи.

За повечето потребители CLISCHEMA осигурява най-големи предимства по отношение на производителността. Поради тази причина като цяло ще препоръчаме да се използва CLISCHEMA в производствена среда, тъй като е много по-лесно да се настроят и модифицират потребителските таблици на CLISCHEMA чрез процесора за обработка на команди CLI (db2cli.exe) и командата bldschem. Вижте също “db2ocat оптимизатор за каталози” на страница 189.

---

## Допълнителни препоръки и съвети

Ключовата дума CLISCHEMA трябва да се добави към файла db2cli.ini в рамките на раздела за DSN името или раздела COMMON. Раздел е текст в квадратни скоби. Разделът COMMON може да се разпознае по текста "COMMON" в квадратни

скоби. Отбележете, че за ключовите думи и имената на разделите главните и малките букви не са от значение.

При свързване всяка възможна ключова дума се проверява най-напред под DSN името, а след това в раздела COMMON, ако не се намери там. Така има възможност да се използват както специфични DSN ключови думи, така и глобални (клиентски) ключови думи.

Освен това ключовата дума DBALIAS може да се използва за създаване на различни DSN (ODBC източници на данни) които се преобразуват към същата база данни. (DSN името може да е с дължина до 255 символа и се трансформира към 8-буквеното име на база данни).

В следващия пример всеки път, когато потребител се свърже към TESTDB или DSN, който не е изброен във файла, ще се използва clischema=ODBCCAT. При свързване към TestDBcat2, ще се използва clischema=odbccat2, но отново свързването ще е към базата данни testdb.

Примерен файл db2cli.ini:

```
[TESTDB]

[COMMON]
clischema=odbccat

[TestDBcat1]
DBALIAS=testdb
clischema=odbccat1

[TestDBcat2]
DBALIAS=testdb
clischema=odbccat2
```

---

## db2ocat оптимизатор за каталози

За 32-битовите Windows операционни системи е предоставено новото средство – db2ocat, с което да се оптимизира търсенето в системните каталози от ODBC и JDBC приложенията.

Можете да получите db2ocat оптимизатора за каталози, като изтеглите db2ocat.zip от:

<ftp://ftp.software.ibm.com/ps/products/db2/tools>.

---

## Допълнителни източници на информация

Следващите допълнителни източници на информация може да представляват интерес:

- Доклад за автоматичното разпространение на каталози с помощта на DataPropagator:  
<http://www.ibm.com/software/data/db2/os390/odbcatl1g.html>
- Доклад за ръчен подход, без да се използва DataPropagator:  
<http://www.ibm.com/software/data/db2/os390/odbcmanu.html>





---

## Приложение G. Източници с допълнителна информация

---

### Други свързани издания

Номер	Заглавие на книга
SG24-2006	<i>Мигриране до DB2 Universal Database версия 5</i>
SG24-2213	<i>Въпроси за производителността на DB2 за OS/390 версия 5</i>
SG24-4893	<i>DB2 среща NT</i>
SG24-4894	<i>Универсално ръководство за свързване на DB2</i>
SG24-4693	<i>Как да започнем със запомнените процедури на DB2</i>
SG24-2212	<i>DRDA Поддръжка за TCP/IP in DB2 Universal Database за OS/390 V5.1 и DB2 Universal Database V5.0</i>
SC33-0814	<i>CICS за AIX Ръководство за приложно програмиране</i>
SC33-0931	<i>CICS за AIX Ръководство за персонализиране и работа</i>
GC09-2952	<i>DB2 Connect Enterprise Edition за UNIX – Бърз старт</i>
GC09-2953	<i>DB2 Connect Enterprise Edition за OS/2 и Windows – Бърз старт</i>
GC09-2967	<i>DB2 Connect Personal Edition: Бърз старт</i>
GG24-4155	<i>Архитектура на разбрелени реляционни бази данни: Използване на DRDA поддръжката на DDCS за AIX с DB2 за MVS/ESA и DB2 Universal Database за AS/400</i>
GG24-4311	<i>Междуплатформено свързване и приложение на Архитектура на разпределени реляционни бази данни</i>
SC23-2443	<i>Преглед на семейството продукти Encina за AIX</i>



---

## Приложение Н. Забележки

IBM може да не предлага продуктите, услугите или компонентите, дискутирани в този документ, във всички страни. Информация за продуктите и услугите, които се предлагат във вашата област можете да получите от местния представител на IBM. Споменаването на продукт, програма или услуга на IBM не е предназначено да твърди или внушава, че само този продукт, програма или услуга на IBM може да се използва. Всеки функционално еквивалентен продукт, програма или услуга, който не нарушава лицензионните права на IBM, може да се използва като заместител. Потребителят сам носи отговорността да прецени и провери работата на всеки продукт, програма или услуга, които не са на IBM.

IBM може да има патенти или заявки за патенти относно обекти, споменати в този документ. Предоставянето на този документ не дава право на никакъв лиценз върху тези патенти. Може да изпращате писмени запитвания за патенти на адрес:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

За запитвания за лиценз относно двубайтова (DBCS) информация се свържете с Отдела за лицензни права на IBM във вашата страна или изпратете писмени запитвания на адрес:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

**Следващия параграф не се отнася за Великобритания, както и всяка друга страна, където такива изключения са несъвместими с местния закон: INTERNATIONAL BUSINESS MACHINES CORPORATION ОСИГУРЯВА ТОВА ИЗДАНИЕ ВЪВ ВИДА, В “КОЙТО Е” БЕЗ ГАРАНЦИЯ ОТ НИКАКЪВ ВИД, ДИРЕКТНА ИЛИ КОСВЕНА, ВКЛЮЧИТЕЛНО, НО НЕ И САМО КОСВЕНИТЕ ГАРАНЦИИ ЗА НЕ-НАРУШЕНИЕ, ПРИГОДНОСТ ЗА ПРОДАЖБА ИЛИ ПРИЛОЖИМОСТ ЗА НИКАКВА ОПРЕДЕЛЕНА ЦЕЛ. В някои страни не се позволява отхвърляне на директните или косвени гаранции в определени случаи, следователно това твърдение може да не се отнася за вас.**

Тази информация може да включва технически неточности или печатни грешки. Периодично информацията тук се променя; тези промени се вмъкват в новите издания на публикацията. По всяко време и без предупреждение IBM може да направи подобрения и/или промени в продукта(ите) и/или програмата(ите), описани в тази публикация.

Всички препратки в тази информация към страници в Web, които не са на IBM, са само за удобство и по никакъв начин не служат като препоръка за тези страници в Web. Материалите на тези страници в Web не са част от материалите за този продукт на IBM и използването на тези страници в Web е изцяло на ваш риск.

Когато изпращате информация до IBM, вие предоставяте на IBM правото да ползва или разпространява тази информация по всякакъв начин, който фирмата счита за подходящ, без това да води до никакви задължения към вас.

Притежатели на лиценз за тази програма, които желаят да получат информация за нея във връзка с (i) осъществяването на обмен на информация между независимо създадени програми и други програми (включително и тази) и (ii) взаимното използване на обменената информация, трябва да се свържат с:

IBM Canada Limited  
Office of the Lab Director  
1150 Eglinton Ave. East  
North York, Ontario  
M3C 1H7  
CANADA

Такава информация може да е достъпна в съответствие с определени директиви и условия, включващи в някои случаи заплащане или такса.

Лицензионната програма, описана в тази информация и всички налични лицензионни материали са осигурени от IBM под условията на IBM Customer Agreement, IBM International Program License Agreement или някое еквивалентно споразумение между нас.

Всички данни за производителност, които се представят тук са определени в контролирана среда. Следователно резултатите, получени в друга работна среда може значително да се различават. Някои измервания може да са направени в системи на ниво разработка и няма гаранция, че тези измервания ще са същите при стандартните системи. Още повече, че някои измервания може да са оценени чрез екстраполация. Действителните резултати може да се различават. Потребителите на този документ трябва да проверят дали данните са приложими за тяхната специфична среда.

Информацията относно продуктите, които не са на IBM, е получена от доставчиците на тези продукти, техни публикации или други обществено достъпни източници. IBM не е тествал тези продукти и не може да потвърди точността на производителността, съвместимостта или другите твърдения, свързани с продуктите, които не са на IBM. Въпросите за възможностите на продуктите, които не са на IBM, трябва да се отправят към доставчиците на тези продукти.

Всички твърдения относно бъдещи насоки или намерения на IBM могат да се променят или отхвърлят без предупреждение и представляват само цели.

Тази информация може да съдържа примери за данни и отчети, използвани във всекидневни бизнес операции. За по-пълното им илюстриране примерите съдържат имена на индивиди, компании, марки и продукти. Тези имена са измислени и всички съвпадения с имена и адреси, използвани от реални бизнес агенти, са напълно случайни.

#### ЛИЦЕНЗ ЗА ПРАВА ЗА КОПИРАНЕ:

Тази информация може да съдържа примерни приложни програми в съответния програмен код, които илюстрират техники за програмиране за различни платформи. Можете да копирате, промените или разпространявате тези примерни програми в произволен вид без заплащане на IBM при разработка, използване, маркетинг или разпространение на приложни програми, които са в съответствие с интерфейса за приложно програмиране за платформата, за която са написани примерните програми. Тази примери не са тествани изцяло и при всички възможни условия. Следователно IBM не може да гарантира или потвърди надеждността, възможностите за обслужване или функционирането на тези програми.

Всяко копие или всяка част от тези примерни програми или техни производни трябва да включва следния знак за запазени права:

© (името на вашата компания) (година). Части от този код са производни от Примерни програми на IBM Corp. © Copyright IBM Corp. \_въведете годината или годините\_. Всички права запазени.

---

## Търговски марки

Следващите термини, които може да са отбелязани със звездичка(\*), са търговски марки на International Business Machines Corporation в САЩ, други страни или и двете.

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager	OS/390
CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/DS
DB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational	SystemView
Database Architecture	VisualAge
DRDA	VM/ESA
eNetwork	VSE/ESA
Extended Services	VTAM
FFST	WebExplorer
First Failure Support Technology	WIN-OS/2

Следните термини са търговски марки или регистрирани търговски марки на други компании:

Microsoft, Windows и Windows NT са търговски марки и регистрирани търговски марки на Microsoft Corporation.

Java и всички базирани на Java търговски марки и фирмени знаци, както и Solaris са търговски марки на Sun Microsystems, Inc. в САЩ и други страни или и двете.

Tivoli и NetView са търговски марки на Tivoli Systems Inc. в САЩ, в други страни или и двете.

UNIX е регистрирана търговска марка в САЩ, други страни или и двете и е лицензирана изключително чрез X/Open Company Limited.

Имена на други фирми, продукти или услуги, които може да са отбелязани с две звездички(\*\*), може да са търговски марки или марки на услуги на други.



# Индекс

## Специални СИМВОЛИ

, (запетая) в параметричен низ 66  
,, (запетая запетая) в параметричен  
низ 66  
" (двойни кавички) в CLP за  
AIX 92  
\* (звездичка) в CLP за AIX 92  
\ (наклонена черта) в OS/2 144  
&& във файла за SQLCODE  
преобразуване 104

## A

access RDB команда 148  
*Вижте също* ACCRDB команда  
ACCRDB команда 148, 149  
ACCRDBRM команда 148, 149  
ACCSEC 149  
ACQUIRE оператор 45  
AGENTPRI параметър 118  
ampersand, двоен (&&) във файла  
за SQLCODE преобразуване 104  
API  
обновяване на директории на  
база данни 72  
APPC  
име на символно  
предназначение 167  
AR име 65  
ARI (DB2 за VSE и VM) 35  
AS име на база данни  
приемник 65  
AS/400  
DRDA 7  
ASCII  
ред за сортиране 38  
смесени данни 34  
AUTHENTICATION параметър 95  
AUTHENTICATION=CLIENT 101

## B

BIDI параметър 69  
BIND команда  
синтаксис 79  
BINDADD право на достъп 76  
bldschem 187  
синтаксис 188

blocking 36  
Bootstrap dataset на DB2 за  
MVS/ESA или DB2 Universal  
Database за OS/390  
BSDS параметри 64

## C

CALL USING DESCRIPTOR  
оператор (OS/400) 41  
CALL оператори  
различни платформи 41  
CCSID 171  
CGI програмиране  
ограничения 21  
предимства 21  
CHAR тип данни 126  
CHGPWD\_SDN параметър 68  
CICS 33  
CLI  
помощни програми 187  
CLI/ODBC приложения  
CURRENTPACKAGESET 101  
CLIENT тип разпознаване 95  
CLISchema, ключова дума 186,  
187, 188, 189  
COMMIT WORK RELEASE  
оператор  
не се поддържа 45  
commit команда 150  
COMMIT оператор  
статично свързан 116  
CREATE IN COLLECTION  
NULLID 76  
CREATE STORGROUP оператор  
поддръжка 34  
CREATE TABLESPACE оператор  
поддръжка 34  
CURRENTPACKAGESET 101

## D

D (disconnect–прекъсване на  
връзката) параметър 66  
data control language (DCL) 35  
data definition language (DDL) 34  
data manipulation language  
(DML) 34  
DataPropagator 187  
DB\_Authentication параметър 174  
DB\_Communication\_Protocol  
параметър 174, 176

DB\_Database\_Protocol  
параметър 174  
DB\_Native\_Database\_Name  
параметър 174  
DB\_Object\_Type параметър 174,  
176  
DB\_Principal параметър 174  
DB\_Product\_Name параметър 174  
DB\_Target\_Database\_Info  
параметър 177  
DB2 Connect Enterprise Edition  
като сървър за свързване 20  
DB2 Connect версия 5.0  
подобрения 164  
DB2 Connect версия 5.2  
подобрения 163  
DB2 Connect версия 6.1  
подобрения 163  
DB2 Universal Database за OS/390  
DYNAMICRULES(BIND) 101  
TCP/IP предварителна  
проверка 101  
DB2 за MVS/ESA или DB2  
Universal Database за OS/390  
bootstrap dataset 64  
RESPORT 64  
TCPREPORT 64  
ДОМЕЙН 64  
DB2CONNECT\_IN\_APP\_PROCESS  
променлива на обкръжението 81  
db2ocat  
оптимизатор за каталог 189  
DBALIAS, ключова дума 189  
DCE  
директорни услуги 173  
задължителен софтуер 95  
защита 95  
информация за настройка 95  
тип разпознаване 96  
DCL (data control language) 35  
DCS  
тип разпознаване 96  
DCS директория 65  
AR име 65  
AS име на база данни  
приемник 65  
BIDI параметър 69  
CHGPWD\_SDN параметър 68  
LOCALDATE параметър 67  
SYSPLEX параметър 67  
име на база данни  
приемник 65  
име на базата данни 65

DCS директория (*продължение*)  
 Име на средство за обработка  
 на запитвания 65  
 определяне на параметричния  
 низ 70

DCS\_ENCRYPT тип  
 разпознаване 96

dcs1ari.mar файл 103

dcs1dsn.mar файл 103

dcs1qsq.mar файл 103

DDCS версия 2.3  
 подобрения 166

DDCS версия 2.4  
 подобрения 165

ddestrc.dmp файл 144

DDL (data definition language) 34

DECLARE PROCEDURE оператор  
 (OS/400) 41

DECLARE STATEMENT  
 поддръжка 45

DECLARE оператор 45

DELETE  
 поддръжка 34

DESCRIBE оператор 45, 116  
 поддръжка 45

DIR\_CACHE параметър 117

DML (data manipulation  
 language) 34

DSN (DB2 Universal Database за  
 OS/390) 35

DSS тип (трасиране) 143

**E**

EBCDIC  
 ред за сортиране 38  
 смесени данни 34

end unit of work reply message  
 (ENDUOWRM) 150

ENDUOWRM съобщение 150

EXCSAT команда 148

EXCSATRD команда 148, 149

EXCSQLSTT команда 45

EXECUTE IMMEDIATE  
 оператор 116

EXTNAM обект 149

**F**

floating point data type 125

FOR FETCH ONLY в оператор  
 SELECT 115

FORCE, команда 36, 85

**G**

GROUP BY клауза  
 ред за сортиране 38

**I**

IBM WebSphere  
 запознаване 24

ID за оторизация (монитор) 85

ID на клиентско приложение  
 (монитор) 87

ID на хост програма  
 (монитор) 87

INSERT оператор  
 поддръжка 34, 35

integer data type 125

INTERRUPT\_ENABLED  
 (прекъсване на връзка) –  
 параметър 66

ISO/ANS SQL92 38

**J**

Java  
 изпълнение на програми 56  
 сървър на приложения  
 използване на DB2  
 Connect 23

JDBC  
 изпълнение на програми 56  
 оптимизатор за каталог 189  
 производителност на  
 приложение 185

**L**

LABEL ON оператор 45

LANGLEVEL SQL92E  
 предкомпилиране 38

LIST DCS APPLICATIONS  
 команда 85

LOB тип данни  
 поддържан от DB2 Connect  
 Версия 7 35

LOCALDATE параметър 67

**M**

MAX\_COORDAGENTS  
 параметър 121

MAXAGENTS параметър 118, 121

MAXDARI параметър 118

Microsoft Windows приложения 19

Microsoft мениджър за ODBC  
 драйвери 53

MVS  
 DRDA 7

**N**

Net.Data

Net.Data (*продължение*)  
 запознаване 24  
 характеристики 24

NNAME на клиент (монитор) 87

NOMAP параметър 66, 103

NONE тип защита 97

NULLID за OS/400 76

NUM\_INITAGENTS  
 параметър 121

NUM\_POOLAGENTS  
 параметър 121

NUMDB параметър 118

**O**

ODBC  
 изпълнение на програми 52  
 интерфейс 19  
 производителност на  
 приложение 185  
 регистриране на мениджъра за  
 драйвери 53

ODBC приложения  
 CURRENTPACKAGESET 101

ODBC/CLI  
 оптимизатор за каталог 189

odbcad32.exe 53

ORDER BY клауза  
 ред за сортиране 38

OS/390  
 DRDA 7

OS/400  
 DRDA 7

**P**

packed decimal data type 125

PC/IXF файлов формат 92

PIU 133

PRDDTA параметър (DRDA) 46

PRDID параметър 149

PREPARE оператор 116  
 поддръжка 45

PROGRAM тип защита 97

ps (process status–състояние на  
 процеси) помощна  
 програма 142, 149

PUT оператор  
 не се поддържа 45

**Q**

QSQ (DB2 Universal Database за  
 AS/400) 35



## R

RACF 181  
RDBNAM обект 149  
REBIND PACKAGE команда (CLP) 79  
REVOKE оператор  
защита 98, 181  
оператор 35  
ROLLBACK WORK RELEASE  
не се поддържа 45  
ROLLBACK команда  
статично свързан 116  
ROLLBACK оператор 36  
RQRIOBLK параметър 117  
RQRIOBLK поле 36

## S

SAME тип защита 97  
SECCHK 149  
SELECT оператор 115, 116  
поддръжка 34  
SERVER тип разпознаване 96  
SERVER\_ENCRYPT тип  
разпознаване 96  
SET CURRENT  
PACKAGESET 101  
SET CURRENT оператор  
поддръжка 45  
SHOW DETAIL опция на  
монитор 86  
SNA производителност  
съвети за настройка 130  
SOCKS  
задължителни променливи на  
обкръжението 64  
SQL  
динамични 115  
статичен 115  
SQL блок  
HE АТОМАРЕН 43, 113  
SQL на IBM 5  
SQL оператори  
вграден 13  
категории 33  
поддръжка 45  
SQL/DS  
DRDA 7  
SQL1338 код на връщане 64  
SQL92 38  
SQLCA  
SQLCODE поле 143  
SQLERRMC поле 36, 43  
SQLERRP поле 35  
буфери с данни 143  
SQLCODE  
поле в SQLCA 143

SQLCODE (*продължение*)  
преобразуване 103  
разлики между платформи 39  
самостоятелен 38  
SQLDA, най-добрия размер, който  
да се задели 116  
SQLERRMC поле на SQLCA 36,  
43  
SQLERRP поле за SQLCA 35  
sqlsact API 47  
SQLSTATE  
в поле SQLERRMC на  
SQLCA 43  
кодове за класове 104  
разлики 39  
самостоятелен 38  
SRVNAM обект 148  
Stored Procedure Builder  
запознаване 42  
характеристики 42  
SYSIBM.SYSPROCEDURES  
каталог (OS/390) 41  
SYSPLEX параметър 67

## T

TCP/IP  
ACCSEC 149  
RESPORT 64  
SECCHK 149  
TCP/PORT 64  
ДОМЕЙН 64  
защитата вече проверена 101  
име на TCP/IP услуга или номер  
на порт 167  
име на услуга 64  
име на хост 167  
конфигуриране на свързванията  
към хост компютъра 18  
отдалечено име на хост 64, 167  
порт за повторна  
синхронизация 64  
Tuxedo  
с DB2 Connect 31

## U

UPDATE оператор  
поддръжка 34

## V

VALIDATE RUN  
DB2 Connect поддръжка 36  
VALNSPRM стойност 149  
VARCHAR тип данни 126  
VM  
DRDA 7

VSE  
DRDA 7  
VTAM 181

## W

web приложения  
запознаване 21  
WebSphere  
advanced edition 25  
enterprise edition 25  
standard edition 25  
запознаване 24  
характеристики 24  
Windows приложения 19

## X

X/Open Distributed Transaction  
Processing (DTP) модел  
запознаване 31  
XA интерфейси  
дефиниция 31  
XA концентратор  
примери 121  
XA съвместим Мениджър на  
транзакции  
XA съвместим Мениджър на  
ресурси 31  
дефиниция 31

## Z

zoned decimal data type 125

## A

Архитектура за представяне на  
символни данни (CDRA) 8  
Архитектура за управление на  
SNA услуги (MSA) 8  
Архитектура на обекти  
форматирани данни (FD:OCA) 8  
Архитектура на разпределена  
релационна база данни (DRDA)  
архитектури, използвани от 8  
издания 15  
понятие 7  
поток данни 7  
средство за обработка на  
запитвания 7  
сървър на приложения 7  
АТОМАРЕН SQL блок  
не се поддържа 43  
не се поддържа в DB2  
Connect 113  
атрибути на идентификатор на  
колекция

атрибути на идентификатор на  
колекция (*продължение*)  
DB2 Universal Database за  
AS/400 37  
пакет 37  
атрибути на квалификатор  
пакет 37  
различни платформи 37  
атрибути на пакети  
квалификатор 37  
собственик 37  
създател 37  
атрибути на собственик  
пакет 37  
атрибути на създател  
пакет 37

## Б

база данни  
групиране на заявки 114  
име 167  
настройка 123  
обект 173  
обект локатор 173  
псевдоним 167  
средства за оценка на  
производителността 112

## В

вграден SQL 13  
време за отговор 109  
външен ключ 38

## Г

грешки 139  
групиране на заявки  
база данни 114

## Д

двубайтови символи 171  
двуфазов протокол за записване на  
промените 10  
порт за повторно  
синхронизиране, използван при  
TCP/IP връзки 64  
дефиниран указател 36  
дефинирана от потребителя  
последователност 38  
дефинирани от потребителя типове  
поддържани от DB2 Connect 35  
динамичен SQL 5, 115  
CURRENTPACKAGESET 101  
DB2 Connect поддръжка 33

динамични указатели 36  
директен достъп до база данни 18  
директно свързване  
към хост бази данни 18  
директори на база данни  
обновяване 63  
директория на база данни  
възел 63  
система база данни 63  
Услуги за връзка с база данни  
(DCS) 63  
директория на възлите 63  
Директория на Услугите за връзка  
с база данни (DCS) 63  
допълнителни оператори от  
динамичен SQL  
не се поддържа 45  
дълги полета 35

## Е

единица работа  
отдалечена 8  
разпределен 10  
експорт  
ограничения 93

## З

заклучване  
ниво ред 39  
ниво страница 39  
таймаут 39  
заклучване на ниво ред 39  
заклучване на ниво страница 39  
заместване на символи 171  
запетая в параметричен низ 66  
запетая запетая в параметричния  
низ 66  
запомнени процедури  
запознаване 22  
общо 41  
защита 179  
APPC 97  
DCE 95  
GRANT 181  
NONE 97  
PROGRAM 97  
REVOKE 181  
SAME 97  
обсъждане 95  
оператор GRANT 98  
оператор REVOKE 98  
тип 97, 167  
типове 64, 179  
защита на ниво ред 39  
заявки за маршрутизиране 8

## И

Идентификатор на клиентско  
приложение (монитор) 86  
идентификатор на кодиран набор  
символи (CCSID) 169  
Идентификатор на кодова  
страница (монитор) 87  
Идентификатор на хост  
приложение (монитор) 86  
издания за прехвърляне 49  
изключения при конвертиране 171  
изключения при конвертиране  
между кодови страници 171  
използване на DB2 Connect  
Tuxedo 31  
с ХА съвместим мениджър на  
транзакции 31  
с монитори за обработка на  
транзакции 29  
сценарии 17  
изпратен буфер (трасиране) 143  
изпълнение на приложения  
клиент на базата данни 51  
източник на данни 9  
име на база данни приемник 65,  
167  
име на базата данни 65, 71  
име на възел 64, 71, 167  
име на потребител 179  
име на приложение (монитор) 85  
име на символно  
предназначение 167  
зависи от главни и малки  
букви 64  
Име на средство за обработка на  
запитвания 65  
име на хост базата данни  
(монитор) 87  
импорт  
ограничения 93  
инсталиране  
DB2 Connect 5  
интерактивен режим на вход  
(CLP) 91  
информация за кеш  
директория 117  
информация за програмиране 33

## К

каскадност 39  
ключове  
външен 38  
първичен 38  
ключови думи  
CLISHEMA 186, 187, 188, 189  
DBALIAS 189

- код на държава
  - в поле SQLERRMC на SQLCA 36
- кодов набор
  - в поле SQLERRMC на SQLCA 36
- кодова страница 169
  - в поле SQLERRMC на SQLCA 36
- колекции 37
- колони с променлива дължина на символите 93
- команда за край (CLP) 91
- команда за обмен на атрибутите на сървъра 148
  - Вижте също* EXCSAT команда
- команда за прекратяване (CLP) 91
- команди
  - ACCRDB 148, 149
  - ACCRDBRM 148, 149
  - BIND 79
  - commit 150
  - EXCSAT 148
  - EXCSATRD 148, 149
  - EXCSQLSTT 45
  - FORCE 36
  - REBIND PACKAGE 79
  - край 91
  - прекратяване 91
- компоненти на многосайтово обновяване
  - тестване 15
- конвертиране
  - символ 34
- конвертиране на символи 34
- конкурентно използване на системни ресурси 129
- конфигурационен параметър DFT\_ACCOUNT\_STR 47
- конфигуриране
  - DB2 Connect 5
  - ODBC драйвер 54, 56
- концентратор за връзки
  - запознаване 119
  - натоварване при свързване 120
  - ограничения 122
  - поддръжка на XA транзакция 121
  - примери 121
- концентратор за свързване
  - конфигурационни параметри 121

## M

- метод за виртуален телекомуникационен достъп (VTAM) 98, 181

- многосайтово обновяване 10
  - поддръжка 44
  - Център за управление 14
- монитори за обработка на транзакции
  - примери 30
- мрежа
  - адаптер или комуникационен контролер 128
  - надеждност 129
  - настройка 126
  - топология 128
  - трафик 129

## N

- настройка
  - DB2 Connect 5
  - база данни 123
  - производителност на мрежата 126
  - производителност на приложение 185
- НЕ АТОМАРЕН SQL блок 43, 113
- необходими права за свързване 76
- неопределен указател 36
- неявна връзка 35
- нива на изолация 40
- ниво на изолация 40
- низ за отчитане 46
- низове с променлива дължина 34
- номер на раздел 45
- номер последователност за клиент (монитор) 86

## O

- обект с маршрутизираща информация 173
- обновяване на директории на база данни 63
- обработка на транзакциите
  - характеристики 29
- обратно таксуване
  - DB2 Universal Database за OS/390 46
  - дефиниция 46
- обръщащи се към себе си таблици 38
- общ SQL 5
- ограничения
  - импорт и експорт 93
  - концентратор за връзки 122
- оператор GRANT
  - защита 98, 181

- оператори
  - ACQUIRE 45
  - call 41
  - COMMIT 116
  - COMMIT WORK RELEASE 45
  - CREATE STORGROUP 34
  - CREATE TABLESPACE 34
  - DECLARE 45
  - DELETE 34
  - DESCRIBE 45, 116
  - EXECUTE IMMEDIATE 116
  - FOR FETCH ONLY 115
  - GRANT 35
  - INSERT 34, 35
  - LABEL ON 45
  - PREPARE 45, 116
  - ROLLBACK 36, 116
  - SELECT 34, 115, 116
  - SET CURRENT 45
  - UPDATE 34
  - свързване 35
- определяне на проблеми 139
- основен файл 142
- отдалечена единица работа 8

## P

- пакет
  - атрибути 37
  - създаден на хост или AS/400 сървър на база данни 77
- параметри
  - AGENTPRI 118
  - AUTHENTICATION 95
  - BIDI 69
  - DB\_Authentication 174
  - DB\_Communication\_Protocol 174, 176
  - DB\_Database\_Protocol 174
  - DB\_Native\_Database\_Name 174
  - DB\_Object\_Type 174, 176
  - DB\_Principal 174
  - DB\_Product\_Name 174
  - DB\_Target\_Database\_Info 177
  - DFT\_ACCOUNT\_STR 47
  - DIR\_CACHE 117
  - INTERRUPT\_ENABLED (прекъсване на връзка) 66
  - LOCALDATE 67
  - MAX\_COORDAGENTS 121
  - MAXAGENTS 118, 121
  - MAXDARI 118
  - NOMAP 66
  - NUM\_INITAGENTS 121
  - NUM\_POOLAGENTS 121
  - NUMDB 118
  - PRDDTA 46
  - PRDID 149

параметри (*продължение*)  
 RQRIOBLK 117  
 SYSPLEX 67

параметричен низ 167

пароли  
 DCE директорни услуги 179

пейджинг  
 размер на блок 117

поддръжка на XA транзакция  
 концентратор за връзки 121

Поддръжка на двупосочен  
 CCSID 69

поддръжка на двупосочни  
 езици 171

поддръжка на зони за дата и  
 час 67

поддръжка на кодовата страница  
 на страната 169

поддръжка на национални езици  
 (NLS)  
 конвертиране на символни  
 данни 169  
 обсъждане 169  
 смесени едно- и двубайтови  
 данни 34, 93

поддръжка на часови зони 67

подобрения  
 DB2 Connect версия 5.0 164  
 DB2 Connect версия 5.2 163  
 DB2 Connect версия 6.1 163  
 DDCS версия 2.3 166  
 DDCS версия 2.4 165

полета на низ на отчитане 46

получен буфер (трасиране) 143

помагала за администриране 6

помощна програма db2cli.exe 187

помощна програма ddcstrc 143  
 изходен файл 143  
 резултат 145  
 синтаксис 144

помощна програма за състоянието  
 на процесите 142, 149

помощна програма за  
 трасиране 143  
 изходен файл 143  
 резултат 145  
 синтаксис 144

помощни програми  
 bldschem 187  
 db2cli 187  
 db2ocat 189  
 ddcspkgn 78, 80  
 ddcstrc 143  
 ps 149  
 администриране 6, 91  
 експорт 92  
 импорт 92  
 свързване 51, 75

помощни програми (*продължение*)  
 системен монитор за базата  
 данни 6  
 състояние на процесите 149  
 трасиране 143

помощник за многосайтово  
 обновяване 14

помощници  
 многосайтово обновяване 14

помощно средство за управление  
 достъпа до ресурси (RACF) 98,  
 181

последователност  
 EBCDIC и ASCII 38

поток данни 7, 109

предикатна логика 114

предкомпилиране  
 DB2 Connect поддръжка 36  
 поддръжка 34

преобразуване 169

преобразуване на SQLCODE 103

преобразуване на данни 125  
 CCSID 169  
 изключения 171  
 кодови страници 169

препълване при числено  
 конвертиране 40

прехвърляне на приложения 33

приложения  
 свързване 75

примери  
 XA концентратор 121  
 концентратор за връзки 121

проектиране на приложения 113

производителност  
 CLI приложения 185  
 CLISCHHEMA, ключова  
 дума 185  
 DB2 за OS/390 124  
 SNA критерии за  
 настройка 133  
 мрежов хардуер 128  
 мрежови средства 112  
 настройка 124  
 настройка работата на ODBC и  
 JDBC приложение 185  
 общо 109  
 отстраняване на проблеми 129  
 Процесор за обработка на  
 команди 116  
 размер на PIU 133  
 размер на RU 133  
 сравнителни  
 характеристики 111  
 средства 111  
 съвети за настройка на  
 SNA 130  
 тесни места 110

производителност (*продължение*)  
 транзакция 109

производителност на CLI/ODBC  
 приложение  
 производителност на  
 приложение 185

производителност на приложение  
 CLISCHHEMA, ключова  
 дума 185

производни таблици  
 системни каталози 39

Променлива на обкръжението  
 DB2ACCOUNT 47

променливи на обкръжението  
 DB2ACCOUNT 47

Процесор за обработка на команди  
 (CLP) 6, 91  
 REBIND PACKAGE  
 команда 79  
 производителност 116

псевдоним на база данни 71

псевдоним на клиент в БД  
 (монитор) 86

първичен ключ 38

## Р

разлики в SQLCODE и  
 SQLSTATE 39

разлики между различни DB2  
 продукти 33

разлики между хост или AS/400  
 сървър и работна станция 45

размер на DB2 Connect  
 RQRIOBLK 133

размер на RQRIOBLK 133

размер на RU 133

размер на блок 117

разновидности на SQL 5

разпознаване 71, 167  
 валидност 95

разпределена заявка 9

разпределена среда 33

Разпределено управление на данни  
 (DDM) 8

разработка на приложения 33,  
 113  
 използване на Клиент за  
 разработка на DB2  
 приложения 19  
 като се използва ODBC 19, 51

разширение на данни в хост или  
 AS/400 сървър 34

регистриране  
 мениджър за ODBC  
 драйвери 53

ред за сортиране  
 дефиниране 38

ред за сортиране (*продължение*)  
последователност 38  
референциален интегритет 38  
решаване на проблеми 139  
препълване при числено  
конвертиране 40

## С

свързване  
CONNECT RESET оператор 35  
CONNECT TO оператор 35  
null CONNECT 35  
необходими права 76  
неявна връзка 35  
пакети 77  
помощни програми 51  
помощни програми и  
приложения 75  
свързвания към DRDA хост  
компютри  
директно към DRDA хост  
компютър 18  
сервизен журнал за първи  
отказ 142  
символ escape 70  
символи shift-out и shift-in 34, 93  
синтаксис  
bldschm 188  
системен каталог  
използване 39  
системен монитор за базата  
данни 81  
системен монитор за базата данни  
помощна програма 6  
системна директория на базата  
данни 63, 71  
системни ресурси  
конкурентно използване 129  
скорост на трансфер на данни 109  
производителност 128  
следене  
свързвания към DB2 Connect  
шлюз 81  
смесени едно- и двубайтови  
данни 34  
експорт 93  
импорт 93  
софтуер, който трябва да е  
инсталиран предварително  
DCE 95  
списък за свързване 75, 183  
сравнителни характеристики  
производителност 111  
средства  
използване на паметта 111  
Използване на процесора 111  
производителност 111

средства за диагностика 142  
средства за експорт  
общо описание 92  
средства за използване на  
паметта 111  
Средства за използване на  
процесора 111  
средства за обработка на  
запитвания  
DRDA определение 7  
параметри 167  
средства за оценка на  
производителността на  
мрежата 112  
средство за импорт  
общо описание 92  
статичен SQL 5, 115  
DB2 Connect поддръжка 33  
събития  
трасиране 144  
създаване на блокове  
данни 115  
създаване на блокове данни 115  
създаване на пул за връзките  
запознаване 22  
съображения за конфигуриране  
промяна на парола 102  
съображения при  
програмиране 33  
в хост или AS/400 среда 33  
сървъри за свързване  
DB2 Connect Enterprise  
Edition 20  
сървъри на приложения  
DB2 Connect поддръжка 27  
DRDA определение 7  
запознаване 25  
конфигуриране 28  
развой 28

## Т

таймаут на заключване 39  
територия  
в поле SQLERRMC на  
SQLCA 36  
тесни места  
транзакция 110  
тестване  
компоненти на многосайтово  
обновяване 15  
тип данни 64-битов integer  
(BIGINT)  
поддържан от DB2 Connect  
Версия 7 35  
тип данни ROWID  
поддържан от DB2 Connect  
Версия 7 35

типове  
ROWID 35  
защита 97  
разпознаване 95  
типове данни  
CHAR 126  
floating point 125  
integer 125  
packed decimal 125  
VARCHAR 126  
zoned decimal 125  
преобразуване 125  
числени 34  
типове разпознаване  
CLIENT 95  
DCE 96  
DCE директорни услуги 179  
DCS 96  
DCS\_ENCRYPT 96  
SERVER 96  
SERVER\_ENCRYPT 96  
по подразбиране 95  
токени и SQLCODE 103  
транзакция  
производителност 109  
трансфер на данни  
между хост и работна  
станция 92

## У

Указател на приложение 85  
указатели  
дефиниран 36  
динамични 36  
неопределен 36  
Управление изтичане срока на  
валидност на паролите(PEM) 68  
управление на разпределени  
данни 143

## Ф

файл db2cli.ini  
настройване работата на  
ODBC и JDBC  
приложение 186  
файл ddc400.lst 75  
файл ddcsmvs.lst 75  
файл ddcsvm.lst 75  
файл ddcsvse.lst 75  
файл за преобразуване на  
SQLCODE  
\* (звездичка) 104  
cc 104  
i 105  
P 104  
s 105

файл за преобразуване на  
SQLCODE (*продължение*)  
U 104  
W 104  
звездичка 104  
Синтаксис 104

## **Х**

хардуер  
производителност на  
мрежата 128

## **Ч**

числени типове данни 34

## **Э**

Э на изходяща последователност  
(монитор) 87

---

## Свързване с IBM

Ако имате технически проблем, моля отделете време да прегледате и изпълните действията, предложени в *Ръководството за решаване на проблеми*, преди да се свържете с отдела за поддръжка на клиенти на DB2. От това ръководство ще разберете каква информация ще е хубаво да имате, така че отдела за поддръжка на клиенти на DB2 да ви обслужи по-добре.

За да получите информация или да поръчате някой от продуктите на DB2 Universal Database, обърнете се към представителството или локалния офис на IBM във вашата страна или към оторизиран дилър на софтуер на IBM.

Ако живеете в САЩ, можете да позвъните на един от следните номера:

- 1-800-237-5511 за поддръжка на клиенти
- 1-888-426-4343, за да научите за възможните опции за обслужване

---

## Информация за продукти

Ако живеете в САЩ, можете да позвъните на един от следните номера:

- 1-800-IBM-CALL (1-800-426-2255) или 1-800-3IBM-OS2 (1-800-342-6672), за да поръчате продукти или да получите обща информация.
- 1-800-879-2755, за да получите издания.

<http://www.ibm.com/software/data/>

Страниците за DB2 в World Wide Web предоставят съвременна информация за DB2, свързана с новости, описания на продукти, графици за образователни курсове и др.

<http://www.ibm.com/software/data/db2/library/>

DB2 Product and Service Technical Library предоставя достъп до често задавани въпроси, поправени грешки, книги и най-нова техническа информация за DB2.

**Забележка:** Възможно е тази информация да е само на английски.

<http://www.elink.ibm.com/pbl/pbl/>

Страниците в Web за поръчка на международни публикации осигурява информация за това как да поръчате книги.

<http://www.ibm.com/education/certify/>

Програмата Professional Certification Program от страниците на IBM в Web осигурява информация за тестове получаване на сертификати за множество продукти на IBM, включително DB2.

<ftp://software.ibm.com>

Включете се като anonymous. В директорията /ps/products/db2 можете да намерите демонстрации, поправени грешки, информация и помощни средства, отнасящи се до DB2 и много други продукти.

<comp.databases.ibm-db2>, <bit.listserv.db2-1>

Тези интернет групи от новини са на разположение на потребителите, за да обсъждат опита си в работата с DB2 продукти.

**В Compuserve: GO IBMDB2**

Въведете тази команда, за да осъществите достъп до фамилията форуми IBM DB2. Всички DB2 продукти се поддържат чрез тези форуми.

Информация за това как да се свържете с IBM извън САЩ, можете да получите от Приложение А на *Наръчник за поддръжка на софтуер на IBM*. За достъп до този документ отидете на следната страница в Web: <http://www.ibm.com/support/>, и след това изберете връзката IBM Software Support Handbook в долната част на тази страница.

**Забележка:** В някои страни оторизираните дилъри на IBM трябва да се свържа с тяхната структура за поддръжка на дилърите, вместо с Центъра за поддръжка на IBM.



Отпечатано в ЕО

SH26-6171-00

