

**Ръководство на потребителя за DB2 Connect  
Обновено Март 2008**





**Ръководство на потребителя за DB2 Connect  
Обновено Март 2008**

**Забележка**

Преди да използвате тази информация, и продукта, който се поддържа от нея, прочетете общата информация под Приложение В, “Забележки”, на страница 169.

**Редакционна забележка**

Този документ съдържа информация, която е собственост на IBM. Тя се предоставя според лицензно споразумение и се защитава от законите за авторското право. Информацията, съдържаща се в тази публикация, не съдържа никакви гаранции за продукти, и всички изявления, осигурени в настоящото ръководство, не трябва да бъдат интерпретирани като такива.

Можете да поръчате IBM публикации онлайн или чрез местния представител на IBM.

- За да поръчате публикации онлайн, идете на IBM Центъра за публикации на [www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)
- За да намерите местен представител на IBM, посетете IBM Указател за контакти за цял свят на [www.ibm.com/planetwide](http://www.ibm.com/planetwide)

За да поръчате DB2 публикации от DB2 Маркетинг и продажби в САЩ или Канада, позвънете на 1-800-IBM-4YOU (426-4968).

Когато изпращате информация на IBM, предоставяте на IBM неизключително право да използва или разпространява информацията по всеки начин, по който бъде преценено за уместно, без от това да произлизат никакви задължения към вас.

© Авторско право International Business Machines Corporation 1993, 2008. Всички права запазени.

© Copyright International Business Machines Corporation 1993, 2008. All rights reserved.

# Съдържание

Относно тази книга . . . . . v

## Раздел 1. Концепции в DB2 Connect 1

### Глава 1. DB2 Connect . . . . . 3

Предложения за DB2 Connect продукти . . . . . 3  
Функции, доставени във Версия 9 и предни издания . . . 3  
Хост бази данни . . . . . 6  
DB2 Connect и SQL изрази . . . . . 6  
Помощни административни средства на DB2 Connect 7  
WebSphere Federation Server и DB2 Connect . . . . . 7

### Глава 2. Архитектура на разпределена реляционна база данни . . . . . 9

DRDA и достъп до данни . . . . . 9  
DB2 Connect и DRDA . . . . . 9  
Отдалечена единица работа . . . . . 11  
Разпределени заявки . . . . . 12

### Глава 3. DB2 Connect сценарии. . . . . 15

Пряк достъп до хост бази данни. . . . . 15  
Достъпване на хост или System i DB2 данни посредством DB2 Connect Personal Edition. . . . . 17  
DB2 Connect сървърни продукти като сървъри на свързваемост . . . . . 18  
DB2 Connect и web приложения . . . . . 19  
DB2 Connect и IBM WebSphere . . . . . 21  
DB2 Connect като Java сървър на приложения . . . . . 21  
DB2 Connect на web сървъра . . . . . 22  
DB2 Connect и сървъри на приложения. . . . . 23  
DB2 Connect и наблюдатели на обработката на транзакции . . . . . 26

## Раздел 2. DB2 Connect справочник . . . . . 29

### Глава 4. Обновяване на директории на бази данни . . . . . 31

Стойности на системната директория на базата данни 31  
Стойности на директорията на възлите . . . . . 32  
Стойности на DCS директорията . . . . . 33  
Работен лист за персонализиране на директория . . . 37  
Дефиниране на множество записи за една и съща база данни . . . . . 38  
Боравене с ViDi данни . . . . . 38

### Глава 5. Защита на DB2 Connect . . . . . 41

Доверени свързвания чрез DB2 Connect . . . . . 41  
Създаване и прекратяване на доверено свързване чрез CLI . . . . . 42  
Превключване на потребители на доверено свързване чрез CLI . . . . . 44

Съображения за разпознаване за DB2 Connect . . . . 46  
Поддръжка на Kerberos . . . . . 47  
Съвети и тънкости за OS/390 и z/OS защитата . . . . 48  
Типове защита, поддържани в DB2 Connect . . . . . 49

### Глава 6. Свързване на приложения и помощни програми (DB2 Connect) . . . 51

### Глава 7. Многосайтови обновявания 55

Активиране на многосайтови обновявания посредством Центъра за управление . . . . . 56  
Тестване на многосайтови обновявания посредством Центъра за управление. . . . . 56  
Многосайтово обновяване и мениджър на точка на синхронизация . . . . . 57  
Конфигуриране на DB2 Connect с XA съвместим мениджър на транзакции . . . . . 57  
DB2 Connect поддръжка за свободно съчетани транзакции . . . . . 58

### Глава 8. Пренасяне на данни с DB2 Connect . . . . . 59

### Глава 9. SQLCODE преобразуване . . . 63

Изключване на SQLCODE преобразуването . . . . . 63  
Персонализиране на SQLCODE преобразуването . . . 63

### Глава 10. Наблюдение на система база данни и DB2 Connect . . . . . 69

Наблюдение на свързванията за отдалечени клиенти 69  
Наблюдение на производителността чрез Windows Performance Monitor . . . . . 69  
Използване на командите GET SNAPSHOT . . . . . 70  
Статус на DCS приложение . . . . . 72  
Здравен монитор и тревоги . . . . . 76  
Общ преглед на здравния монитор на DB2 за z/OS 76  
Стартиране, спиране и опресняване на здравен монитор на DB2 за z/OS . . . . . 77  
Преглеждане, подаване и запазване на препоръчани действия . . . . . 78  
Преглеждане на обобщения за здравни тревоги. . . 81  
Преглеждане на обекти на здравни тревоги . . . . . 82

## Раздел 3. Висока достъпност и DB2 Connect . . . . . 85

### Глава 11. Висока достъпност и балансиране на натоварването за свързваемост на хост база данни . . . 87

<b>Глава 12. Описание и настройка на автоматичното пренасочване на клиент (DB2 Connect) . . . . .</b>	<b>89</b>
---	-----------

<b>Глава 13. Конфигуриране на автоматично пренасочване на клиент за технология на разпределител на клиентски свързвания . . . . .</b>	<b>91</b>
---	-----------

---

<b>Раздел 4. Настройка и DB2 Connect. . . . .</b>	<b>93</b>
---	-----------

<b>Глава 14. Съображения за производителност за DB2 Connect . . . . .</b>	<b>95</b>
---	-----------

<b>Глава 15. Оптимизиране на ODBC достъпа . . . . .</b>	<b>99</b>
---	-----------

<b>Глава 16. Дизайн на приложение . . . . .</b>	<b>101</b>
---	------------

<b>Глава 17. Управление на свързвания . . . . .</b>	<b>105</b>
---	------------

Пулиране на свързвания . . . . .	105
Концентратор на свързвания . . . . .	107
Пулове и концентратор на свързвания . . . . .	112
Концентратор на свързвания, изискван с WebSphere MQ Transaction Manager и DB2 за OS/390 . . . . .	112

<b>Глава 18. DB2 Connect Sysplex поддръжка . . . . .</b>	<b>115</b>
--	------------

Съображения за OS/390 и zSeries SYSPLEX употребата . . . . .	115
DB2 Sysplex употреба . . . . .	116
Конфигурационни параметри за Sysplex . . . . .	117

<b>Глава 19. Настройка на DB2 Connect . . . . .</b>	<b>119</b>
---	------------

Фина настройка на хост база данни . . . . .	120
Съображения при фината настройка на мрежата . . . . .	121
Конфликт в системните ресурси . . . . .	122
Отстраняване на проблеми с производителността на DB2 Connect . . . . .	123
Настройка на DB2 за OS/390 и z/OS . . . . .	123
Увеличаване на нивата на трансфер на данни на DB2 Connect . . . . .	123
Допълнителен блок на заявка . . . . .	124
Мащабиране на RFC-1323 прозорец . . . . .	125
Преобразуване на хост данни . . . . .	126
Типове символни данни . . . . .	126
Мрежов хардуер . . . . .	126

<b>Глава 20. Настройка на производителност на CLI/ODBC приложение. . . . .</b>	<b>129</b>
--	------------

---

<b>Раздел 5. Отстраняване на проблеми . . . . .</b>	<b>131</b>
---	------------

<b>Глава 21. Отстраняване на проблеми . . . . .</b>	<b>133</b>
---	------------

Събиране на свързана информация . . . . .	133
Началното свързване не е успешно . . . . .	133
Проблеми, срещани след първоначалното свързване . . . . .	134
Диагностични средства . . . . .	135

<b>Глава 22. DB2 проследявания в DB2 Connect. . . . .</b>	<b>137</b>
---	------------

Получаване на DB2 проследяване посредством db2trc . . . . .	137
Дъмп на DB2 файл на проследяване . . . . .	138
Форматиране на файл на DB2 проследяване . . . . .	139

<b>Глава 23. Файлове от DRDA проследяване . . . . .</b>	<b>141</b>
---	------------

Помощна програма за проследяване . . . . .	141
Извеждане на проследяването . . . . .	142
Анализ на изходния файл от проследяването . . . . .	142
Примерни изходни файлове от проследяването . . . . .	144
Последваща буферна информация за DRDA проследявания . . . . .	148

---

<b>Раздел 6. Съобщения . . . . .</b>	<b>151</b>
--------------------------------------	------------

<b>Глава 24. Общи DB2 Connect проблеми . . . . .</b>	<b>153</b>
--	------------

---

<b>Раздел 7. Приложения . . . . .</b>	<b>157</b>
---------------------------------------	------------

<b>Приложение А. Общ преглед на DB2 техническата информация . . . . .</b>	<b>159</b>
---	------------

DB2 техническа библиотека на хартия или в PDF формат . . . . .	159
Поръчване на отпечатани DB2 книги . . . . .	162
Извеждане на помощ за SQL състояние от процесор за обработка на команди. . . . .	163
Достъпване на различни версии на DB2 Център за информация . . . . .	163
Извеждане на теми на предпочитания ви език в DB2 Център за информация . . . . .	163
Обновяване на DB2 Център за информация, инсталиран на компютъра ви или на intranet сървър . . . . .	164
DB2 самоучители . . . . .	166
DB2 Информация за отстраняване на проблеми . . . . .	166
Срокове и условия . . . . .	167

<b>Приложение В. Забележки . . . . .</b>	<b>169</b>
--	------------

<b>Индекс . . . . .</b>	<b>173</b>
-------------------------	------------

---

## Относно тази книга

*Ръководството за потребителя на DB2 Connect* осигурява цялата информация, която трябва да научите, за да използвате DB2 Connect продукта. DB2 Connect концепцията е представена с типичен сценарий, показващ взаимоотношенията между DB2 Connect и другите части на мрежовото обкръжение. Разглеждат се съображения, включващи директории на бази данни, защита между системи, многосайтови обновявания, преместване на данни и наблюдение на DB2 Connect. Представено е как DB2 Connect поддържа висока достъпност във вашето мрежово обкръжение. Въвежда се как да се осигури добра производителност от DB2 Connect и в мрежата, както и някои теми, свързани с отстраняването на възможни проблеми.

### **За кого е предназначена тази книга?**

Системни администратори, администратори на бази данни и специалисти по системни комуникации биха се заинтересували от част или от цялата тази книга.





---

## **Раздел 1. Концепции в DB2 Connect**



---

## Глава 1. DB2 Connect

DB2 Connect осигурява бърза и надеждна свързваемост към хост и System i бази данни за e-business и други приложения, изпълнявани под Linux, UNIX, и Windows операционни системи.

DB2 Connect Personal Edition осигурява пряка свързваемост към хост и System i DB2 сървъри, докато DB2 Connect сървърните продукти осигуряват непряка свързваемост, която позволява на клиентите да достъпват хост и System i DB2 сървъри чрез DB2 Connect шлюза. Множество DB2 Connect сървърни продукти предоставят уникални решения за пакетирание и лицензиране, които ви позволяват да изберете продукт, подходящ за вашето обкръжение.

---

### Предложения за DB2 Connect продукти

DB2 Connect има няколко решения за свързване, включително DB2 Connect Personal Edition, и множество DB2 Connect сървърни продукти:

- DB2 Connect Enterprise Edition
- DB2 Connect Application Server Edition
- DB2 Connect Unlimited Edition за zSeries
- DB2 Connect Unlimited Edition за i5/OS

За подробна информация относно предложенията на DB2 Connect продукти, вижте <http://www.ibm.com/support/docview.wss?rs=73&uid=swg21219983>

---

### Функции, доставени във Версия 9 и предни издания

Този раздел предоставя обобщение на подобренията, въведени във всяка версия и издание.

#### Функции, доставяни в DB2 Connect Версия 9

DB2 Connect Версия 9 включва следните подобрения:

- Клиентска поддръжка за доверени свързвания  
Клиент може да създаде доверени свързвания посредством ODBC, XA, или новите Java методи към сървъри на бази данни (понастоящем само DB2 за z/OS), които поддържат доверени контексти. Потребителското име на клиента може да бъде превключено, без да се налага сървърът на база данни да разпознава напълно новото име.
- Поддръжка на BINARY, VARBINARY, и DECFLOAT типове данни  
DB2 за z/OS вече поддържа типовете данни BINARY, VARBINARY, и DECFLOAT. Поддръжката за тези типове данни е добавена към DB2 CLI и IBM Data Server Provider за .NET. Вашите приложения, използващи DB2 Connect за достъпване на DB2 за z/OS, могат да използват DB2 CLI и IBM Data Server Provider за .NET, за да използват предимствата на новите типове данни. Нова настройка за свързване на име SQL\_ATTR\_DECFLOAT\_ROUNDING\_MODE позволява на клиента да укаже какъв тип закръгляване следва да се изпълни, ако някоя операция на сървъра изисква закръгляване на десетична плаваща стойност.
- Комуникационните протоколи NetBIOS и SNA вече не се поддържат

Клиентите, използващи тези протоколи, трябва да прекаталогизират своите възли и бази данни, като използват поддържан протокол като TCP/IP.

- Добавена е поддръжка на комуникационния протокол IPv6  
Добавена е поддръжка за Internet Protocol Version 6 (IPv6), така че вече можете да се свързвате със сървъри, като използвате IPv4 или IPv6 адреси.
- Ограничението от 64 KB на Процесора за обработка на команди (CLP) за SQL изрази е премахнато  
Новото ограничение за процесора за обработка на команди (CLP) от приблизително 2 MB за SQL изрази и за CLP команди, съдържащи компоненти на SQL изрази, е съизмеримо с ограниченията на другите DB2 средства. Приложенията ви, използващи DB2 Connect, вече могат да използват предимствата на това ново ограничение.
- Подобрения в IBM Data Server Provider за .NET, включително поддръжката на .NET Framework 2.0  
Тези поддръжка и подобрения ще ви помогнат да разработвате по-мощни .NET приложения за използване с DB2 Connect. Някои от тези нови способности включват:
  - Приложенията могат да поднасят конкретен набор редове, вместо да се налага придвижване в целия резултатен набор.
  - Приложенията могат да изпълняват операция по копиране на насипни данни.
  - Приложенията могат да определят броя SQL изрази, които да събират, преди да ги използват като последователност от команди на DB2 сървъра на база данни. Това ще доведе до по-малко индивидуални предавания на данни между клиентското приложение и сървъра на базата данни.
- Двухазов комит за източници данни на различни производители, когато се използва WebSphere Federation Server  
DB2 Connect приложенията могат да използват WebSphere Federation Server за достъпване на източници на данни, предлагани от много IBM и не-IBM доставчици.
- Поддръжка на таймаут на свързвания за приложения на базата данни  
Можете да ограничите периода от време, който вашите DB2 Connect приложения на база данни трябва да изчака за свързване. Това е особено полезно, когато целевият сървър на база данни е недостъпен.
- DB2 Connect Personal Edition е по-лесен за надстрояване  
Можете да надстроите DB2 Connect Personal Edition на Windows и Linux операционни системи, като осигурите подходящия файл на електронен сертификат. Вече не е нужно да се изпълнява цяла инсталация при надстрояването.
- Изменения в DB2 лицензната поддръжка  
Измененията в пакетирането на DB2 Connect продукта са част от измененията в Лицензния център и командата на Средството за управление на лицензи (db2licm).

#### **Функции, доставяни в DB2 Connect Версия 8 Издание 2**

DB2 Connect Версия 8.2 включва следните подобрения:

- Автоматично пренасочване на клиент  
Ако TCP/IP свързване към сървър или DB2 Connect Server бъде изгубено, клиентът автоматично ще опита отново да установи свързването, ако

съществува алтернативен сървър. Алтернативният сървър се указва на потребителския модел на сървъра, и разположението му се изпраща на клиента по време на свързването.

- Криптиране на данни  
Комуникацията клиент/сървър вече предлага криптиране на потребителските данни за пренасянето им по мрежата.

#### **Функции, доставяни в DB2 Connect Версия 8 Издание 1 (включително всички FixPaks и нива на модификации)**

DB2 Connect Версия 8.1 включва следните подобрения:

- Поддръжка за по-дълги SQL изрази (до 2MB)  
SQL изрази до 2 MB могат да преминават през CLI и JDBC приложения. Ограничението на вградения интерфейс обаче остава 64K.
- Диагностична информация, която идентифицира произхода на SQL израз  
Осигурява възможността да се определи коя приложна програма е подала даден израз в кеша за динамични SQL изрази на DB2 за z/OS.
- Поколонен входов масив  
Позволява на приложенията да осигуряват множество набори параметри за единичен SQL израз.
- Наблюдение на мрежово време  
Използват се нови мониторинжни елементи за получаване на по-добра идея за активността в базата данни и мрежовия трафик на ниво база данни или приложение.
- DB2 CLI поддръжка на динамичен скролируем указател  
Динамичните скролируеми указатели вече се поддържат в DB2 CLI, при достъпване на сървъри, които са DB2 Universal Database (UDB) за z/OS Версия 8.1 или по-нови.
- eWLM поддръжка  
Осигурява способността да наблюдава от край до край единици работа чрез междинни групи, за да определя места на потенциални задръствания.
- Подобрения в DB2 командата ping  
DB2 командата ping вече поддържа спецификацията на размер на пакет на запитване и отговор.

**Забележка:** DB2 Connect не поддържа командата PING, когато тя се подава от Клиент от версия 7 през шлюз от Версия 9 на хоста.

#### **Функции, доставяни в DB2 Connect Версия 7 Издание 2**

DB2 Connect Версия 7.2 включва следните подобрения:

- Подобрена поддръжка за Microsoft Transaction Server (MTS) и COM+ технологии
- DB2 Connect Web Starter Kit
- DB2 Connect за Linux на S/390

#### **Функции, доставяни в DB2 Connect Версия 7 Издание 1**

DB2 Connect Версия 7.1 включва следните подобрения:

- ХА концентратор
- Подобрения в многосайтовото обновяване

---

## Хост бази данни

Терминът *база данни* се използва в този документ за описване на система за управление на реляционни бази данни (СУРБД). Други системи, с които комуникира DB2 Connect, могат да използват термина база данни, за да описват леко различна концепция. Освен това DB2 Connect терминът база данни може да се отнася и за:

### OS/390 или z/OS

DB2 Universal Database (UDB) за OS/390 и z/OS Версия 7 или DB2 UDB за z/OS Версия 8. DB2 Universal Database за z/OS и OS/390 подсистема, идентифицирана по нейното LOCATION NAME. LOCATION NAME 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). LOCATION NAME поддържа до 8 псевдонима на имена на разположения, което позволява на приложенията да използват различни dbalias имена за достъп до Версия 8 z/OS сървър. Използвайте z/OS -display ddf командата, за да получите името на разположение на DB2 сървър, името на домейн, IP адреса и порта.

**VSE** DB2 за VSE, изпълнявана на дял на базата данни, идентифициран чрез неговото DBNAME

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

### OS/400

DB2 за i5/OS, интегрална част на OS/400 операционната система. Само една база данни може да съществува на System i сървър, освен ако системата е конфигурирана да използва независими спомагателни пулове на съхранение.

---

## DB2 Connect и SQL изрази

DB2 Connect препраща SQL изразите, подадени от приложните програми, на хост или System i сървърите на бази данни.

DB2 Connect може да препраща почти всеки валиден SQL израз, както и поддържаните DB2 API (интерфейси за приложно програмиране):

- JDBC
- SQLJ
- ADO.NET
- OLE DB
- ODBC
- Perl
- PHP
- DB2 CLI
- Вграден SQL

### Вградена SQL поддръжка

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 изразът се подава на хост или System i сървър на база данни. Динамичният SQL е по-гъвкав, но потенциално е по-бавен. Приложният програмист взима решението дали да използва статичен или динамичен SQL. И двата типа се поддържат от DB2 Connect.

Различните хост или System i сървъри на бази данни реализират различно SQL. DB2 Connect напълно поддържа общия IBM SQL, както и DB2 за OS/390 и z/OS, DB2 сървър за VSE & VM (доскоро SQL/DS), и DB2 за System i реализациите на SQL. IBM SQL силно се препоръчва за поддържане на независимостта на база данни.

---

## Помощни административни средства на DB2 Connect

Следните помагала са в помощ на администраторите на DB2 Connect:

- Процесор за обработка на команди (CLP) ви позволява да подавате SQL изрази срещу хост или System i сървъри на бази данни. Насочва SQL изразите към определената от вас база данни.
- DB2 Командният център осигурява графичен интерфейс за Процесор за обработка на команди (CLP).
- Помощните програми за импортиране и експортиране ви позволяват да зареждате, импортирате и експортирате данни към и от файл на работна станция и хост или System i 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.
- Ако изпълнявате DB2 Connect сървърен продукт, можете да използвате Event Viewer и Монитор на производителността. 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 администратите и наблюдавате всички аспекти на DB2 Connect сървърите. Той също позволява на администраторите да работят с обекти на DB2 за OS/390 или z/OS бази данни, като таблици, производни таблици, буферни пулове и нишки.
- Помощната програма системен монитор за базата данни дава възможност на системния администратор да следи свързванията на системата. Тази функция е достъпна само, когато DB2 Connect функционира като сървър. Помощната програма също помага на системния администратор да определи източника на грешка. Системният администратор може да обвърже клиентските приложения със съответните задания, изпълнявани на хост или System i сървър на база данни.

**Забележка:** В предни издания DB2 графичните административни средства като Център за управление са били поддържани на всички платформи. Считано от Версия 9, DB2 графичните административни средства се поддържат само на Windows x86, Windows x64 (AMD64/EM64T), Linux на x86, и Linux на AMD64/EM64T. За всички платформи, можете да използвате DB2 Процесора за обработка на команди (CLP) за административни цели.

---

## WebSphere Federation Server и DB2 Connect

WebSphere Federation Server е отделно продуктово предложение, което осигурява достъп до и интеграция на данни спрямо източници на данни на множество производители, докато DB2 Connect ви позволява да боравите с големи обеми данни, разположени в съществуващи хост или средни сървъри.

WebSphere Federation Server помага за интегрирането на информацията, като позволява набор източници на данни да бъде преглеждан и манипулиран като единичен източник. Това прави достъпа до източници на данни напълно прозрачен за извикващото приложение. WebSphere Federation Server работи в съчетание с DB2 Connect сървърните продукти. WebSphere Federation Server осигурява собствен достъп за четене и запис в бази данни на семейството DB2 продукти, Informix, Oracle, Sybase, Teradata, и Microsoft SQL Server. WebSphere Federation Server също осигурява достъп за четене към нерелационни източници и източници на данни от естествени науки, като BLAST, Documentum, Entrez, IBM Lotus Extended Search, файлове с таблична структура, и XML. Можете да го използвате за формулиране на запитвания към данни в обединена система.



---

## Глава 2. Архитектура на разпределена реляционна база данни

Distributed Relational Database Architecture (DRDA) представлява набор протоколи, които позволяват на множество системи бази данни, както на IBM, така и други, както и на приложни програми, да работят заедно. Всяка комбинация от продукти за управление на реляционни бази данни, която използва DRDA, може да бъде свързана, за да оформи разпределена система за управление на реляционни бази данни. DRDA координира комуникациите между системите, като дефинира какво трябва да бъде обменено и как трябва да бъде обменено.

### 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.

### Разпределена единица работа

*Разпределена единица работа (DUOW)*, също известна като многосайтово обновяване, включва повече от един сървър на база данни в единица работа. DUOW има 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.
- Възможно е да има множество 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.

---

## DRDA и достъп до данни

Въпреки че DRDA дефинира комуникационните протоколи на базата данни, не дефинира програмните интерфейси, или API, които следва да бъдат използвани от програмистите на приложения. Като цяло DRDA може да се използва от приложната програма, за да се предават заявки, които DRDA сървърът-приемник може да изпълни. Всички познати днес DRDA сървъри могат да изпълняват SQL заявки, препратени от приложна програма посредством DB2 Connect.

IBM предоставя на приложните програмисти средства за генериране на SQL заявки за Windows, UNIX, и Linux операционни системи. Тези средства са част от DB2 клиента. DB2 мениджърът на база данни поддържа няколко програмни интерфейса: ADO.NET, JDBC, SQLJ, PHP, Perl DBI, вграден SQL, DB2 Call Level Interface (DB2 Call Level Interface), и OLE DB. Тези API могат да се използват от програмистите за създаване на програми на множество програмни езици.

---

## DB2 Connect и DRDA

DB2 Connect реализира DRDA архитектурата за намаляване на цената и сложността на достъпването на данни, съхранявани в DB2 Universal Database (UDB) за iSeries, DB2 за System i, DB2 UDB за OS/390 и z/OS, DB2 за z/OS, DB2 сървър за VSE & VM, и други DRDA-съвместими сървъри на бази данни. Чрез пълно използване на DRDA

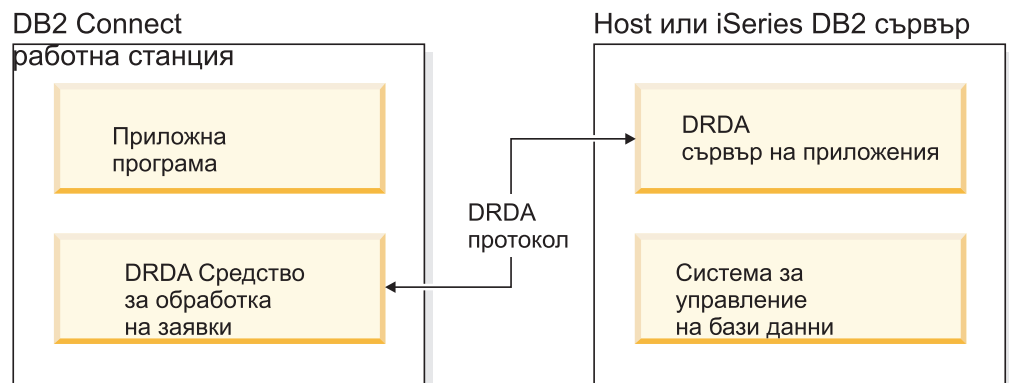
архитектурата, DB2 Connect предлага решение с добра производителност и ниска цена, с изискваните от купувачите характеристики за управление на системите.

В DRDA терминологията *рикуестър на приложение (AR)* е кодът, който обработва разпределено свързване от страната на приложението. AR е приложението, което заявява данни. DB2 Connect действа като риквестър на приложение от името на приложни програми, които могат да бъдат локални за DB2 Connect работната станция, или на отделен клиент, който е отдалечен за DB2 Connect.

*Сървър на приложения (AS)* е кодът, който се грижи от страна на базата данни за отдалеченото свързване.

DRDA също поддържа многослойни свързвания между риквестър на приложение и сървър. В тази топология сървърът, към който се свързва риквестърът на приложение, е сървър на приложения, но всеки друг сървър нататък в последователността се нарича сървър на база данни (DS), тъй като не взаимодейства пряко с риквестъра на приложение. В добавка, за да се акцентира на ролята му като различна от системата, от която произлиза заявката към базата данни, или от системата, която изпълнява функцията на база данни за заявката, всеки сървър на приложение или сървър на база данни между риквестъра на приложение, и крайния сървър на база данни, се нарича също междинен сървър. Употребата на сървъри на бази данни и междинни сървъри се поддържа от DB2 Connect.

Фигура 1 показва потока данни между DB2 Connect работната станция и хост или System i сървъра в случая, когато има само локални клиенти.



Фигура 1. Поток данни между DB2 Connect сървър и хост или System i сървър

За да реализира свързванията между DRDA сървърни системи за управление на бази данни и Клиент на IBM сървър на данни, DRDA използва следните архитектури:

- Character Data Representation Architecture (CDRA)
- Distributed Data Management Architecture (DDM)
- Formatted Data Object Content Architecture (FD:OCA)
- Transmission Control Protocol/Internet Protocol (TCP/IP).

Тези архитектури са използвани като изграждащи блокове. Поточите данни, които протичат в мрежата, се указват от DRDA архитектурата, която документира протокол за поток данни, поддържащ достъп до разпределени реляционни бази данни.

Заявката се пренасочва към правилното местоназначение посредством директории, които съдържат различни типове комуникационна информация и името на достъпвания DRDA сървър на база данни.

## Отдалечена единица работа

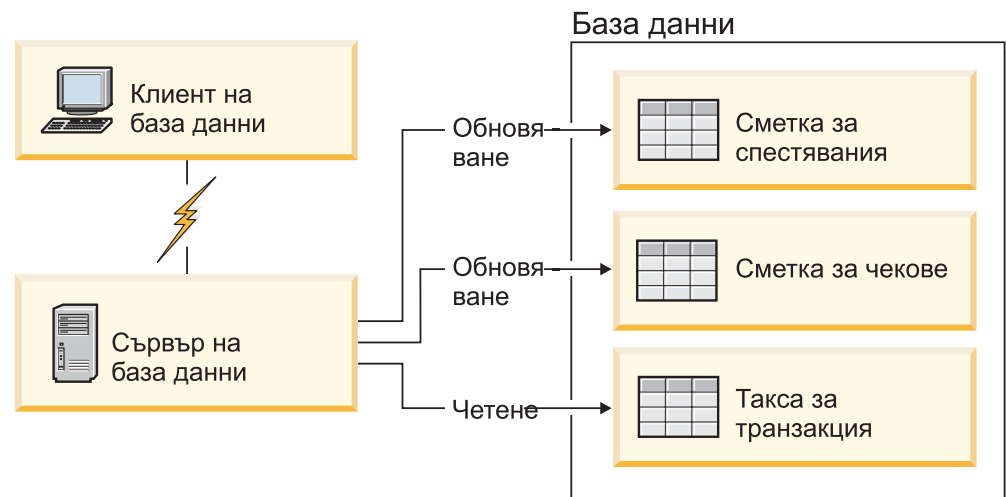
*Отдалечената единица работа* позволява на потребителя или приложната програма да прочете или обнови данни от едно място на всяка единица работа. Поддържа достъп до една база данни в рамките на единица работа. Въпреки че приложна програма може да обнови няколко отдалечени бази данни, тя може да осъществи 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. В някои случаи на грешка, сървърът на базата данни или DB2 Connect може да изпълни ролбек на единицата работа.

Например, Фигура 2 показва клиент на база данни, който изпълнява приложение за превод на средства, което достъпва база данни, съдържаща таблици за сметки за чекове и спестявания, както и план за таксуване на преводите. 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.
- Да запише операцията (единицата работа).



Фигура 2. Използване на една база данни в транзакция

За да настроите такова приложение, трябва да направите следното:

1. Създайте таблиците за сметката за спестявания, чековата сметка и плана за таксуване на преводите в една база данни.

2. Ако физически е отдалечен, настройте сървъра на базата данни да използва подходящия комуникационен протокол.
3. Ако физически е отдалечен, каталогизирайте възела и базата данни, за да идентифицирате базата данни на сървъра на бази данни.
4. Прекомпилирайте приложната си програма, за да укажете свързване от тип 1; с други думи, укажете CONNECT(1) на командата PREP.

---

## 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 СУБД или бази данни в един израз. Например, свързване между две таблици в две различни DB2 за OS/390 или z/OS подсистеми.

DB2 Connect осигурява поддръжка за разпределени заявки между бази данни и СУБД. Например, можете да изпълните операция UNION между DB2 таблица и производна таблица на Oracle. Поддържаните СУБД включват членове на DB2 семейството (като DB2 Database за Linux, UNIX и Windows, DB2 за OS/390 и z/OS, и DB2 за System i) и Oracle. Поддръжката на множество производители е достъпна, когато се използва DB2 Connect в съчетание с WebSphere Federation Server.

Разпределеното запитване осигурява *прозрачност за мястото* по отношение на обектите в базата данни. Ако се премести информация (в таблици) обръщанията към тази информация (наречени *прякори*) могат да се обновят, без да се променят приложенията, за които е необходима тази информация. Разпределената заявка също осигурява *компенсация* за СУБД, които не поддържат изцяло DB2 SQL диалекта, или определени способности за оптимизация. Операциите, които не могат да бъдат изпълнени в такава СУБД (като например рекурсивен SQL), се изпълняват под DB2 Connect.

Разпределеното запитване действа по *почти автономен* начин. Например, могат да се подават DB2 заявки, съдържащи препратки към Oracle обекти, докато Oracle приложения достъпват същия сървър. Разпределеното запитване не монополизира и не ограничава достъпа (освен ограниченията за интегритет и заключване) до обекти на Oracle или на други СУБД.

Реализацията на функцията за разпределена заявка се състои от DB2 Connect потребителски модел, база данни, която ще служи като обединена база данни, 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 СУБД и данни. Приложенията се свързват към обединената база данни като към всяка друга DB2 база данни. DB2 Connect обединената база данни не е лицензирана 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 заявки към една обединена база данни, която после извлича данни от системи от DB2 семейството или Oracle системи, според необходимостта. Потребителят и приложенията определят прякори в запитванията; тези прякори осигуряват обръщания 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 разпредените 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 засегне решения за потискане на операции по оценка на източници на данни.



---

## Глава 3. DB2 Connect сценарии

DB2 Connect може да осигури множество решения за нуждите за достъп до вашата хост или System i база данни. Тази тема очертава няколко сценария, които могат да се приложат към вашите конкретни нужди или среда.

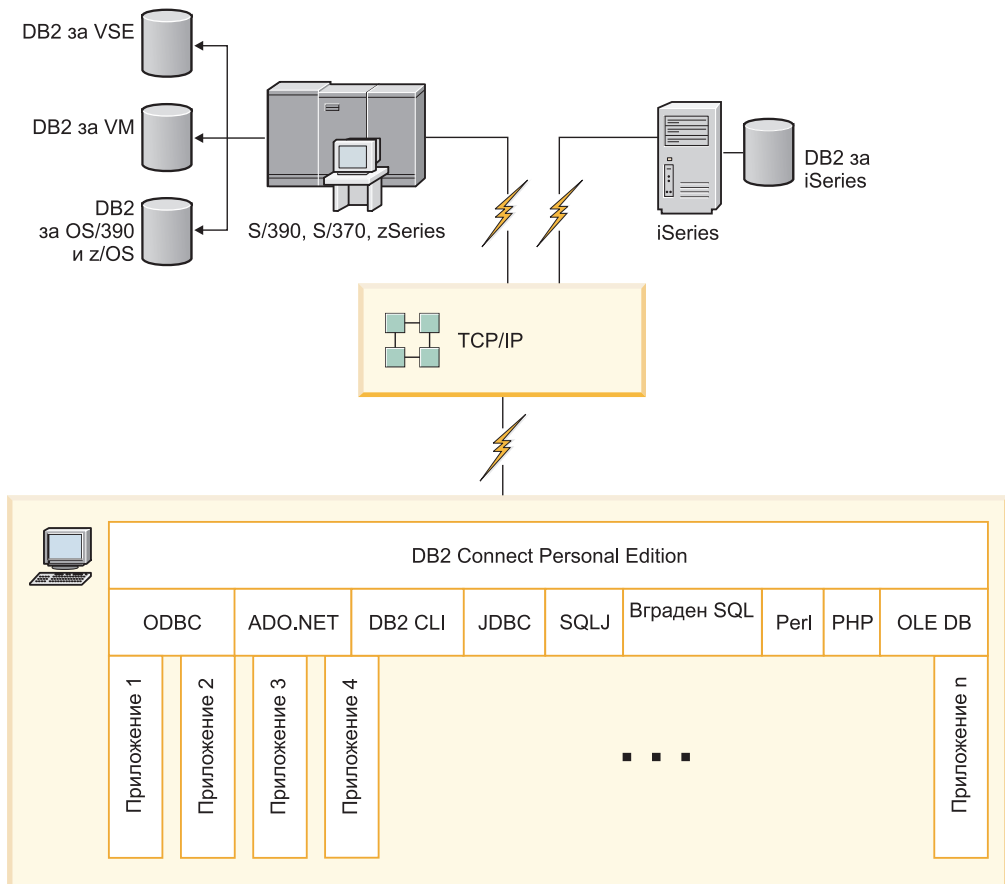
---

### Пряк достъп до хост бази данни

Основната характеристика на DB2 Connect е предоставянето на директно свързване към хост база данни от настолни приложения, изпълнявани на Windows, Solaris или Linux работни станции. DB2 Connect Personal Edition е най-простият начин да се осигури такова решение.

Всяка работна станция, която има инсталиран DB2 Connect Personal Edition, може да установи пряко TCP/IP свързване към DB2 Universal Database (UDB) за OS/390 и z/OS, DB2 за z/OS, DB2 UDB за iSeries, DB2 за i5/OS, и DB2 Database за Linux, UNIX и Windows сървъри. В добавка, приложенията могат да се свързват с, и да обновяват множество бази данни от DB2 семейството в една транзакция с пълен интегритет на данните, осигуряван от протокола за двуфазов комит.

Фигура 3 на страница 16 показва директно свързване към хост или System i сървър на база данни от работна станция с инсталирана DB2 Connect Personal Edition.



Фигура 3. Пряко свързване между DB2 Connect и хост или System i сървър на база данни

**Забележка:**

1. Не е необходимо DB2 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.
2. IBM Data Server Client вече е част от DB2 Connect пакета и може да бъде инсталиран, ако клиентът иска да го използва за разработка на приложения. В добавка DB2 Connect сега включва Stored Procedure Builder, който може да се използва за изграждане, тестване и разгръщане на запомнени процедури за DB2 за OS/390 и z/OS.
3. С програмистите, разработващи Windows приложения, които 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 посредством езика за програмиране Java, могат да използват произволна среда за Java разработка.
4. Ако свързване към DB2 за z/OS сървър на база данни с активирано използване на Sysplex бъде изгубено, клиентът автоматично ще опита да установи отново свързването.



---

## Достъпване на хост или System i DB2 данни посредством DB2 Connect Personal Edition

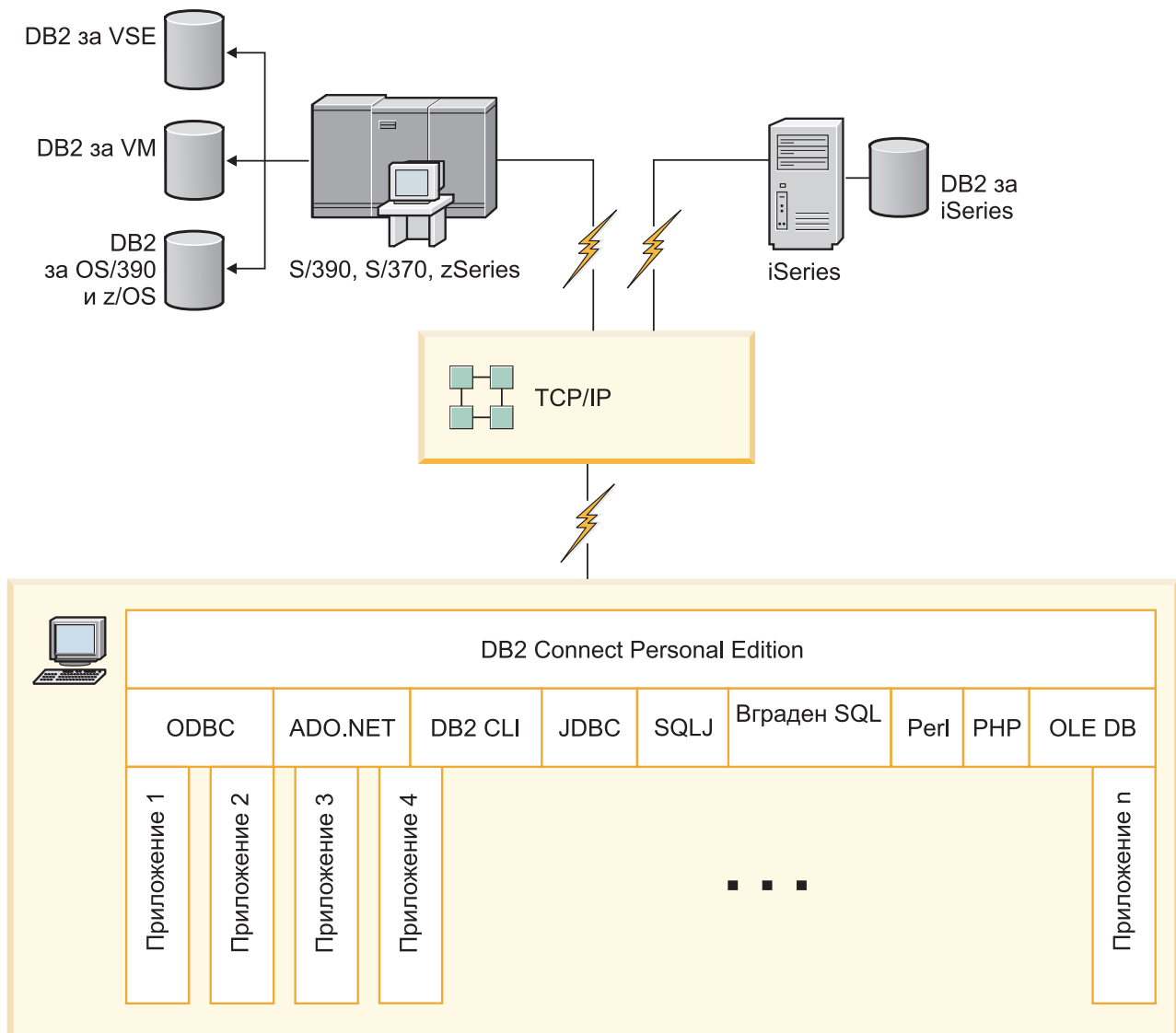
Пряката връзка без междинни сървъри е много удобна и желана конфигурация. Това важи особено за ситуации, в които хостът или System i сървърът на база данни поддържа TCP/IP свързваемост. В такива ситуации, всяка DB2 Connect работна станция установява пряко свързване с хоста или System i сървъра на база данни.

TCP/IP свързваемостта изисква хост или System i базата данни да поддържа TCP/IP. Следните версии поддържат собствени TCP/IP свързвания:

- DB2 Universal Database (UDB) за OS/390 и z/OS Версия 7.1 или по-късна
- DB2 UDB за iSeries Версия 5 Издание 1 или по-късна, и
- DB2 Server за VSE & VM Версия 7 или по-късна

За да се свържете към IBM хост или System i сървър на база данни, ви трябва лицензиран DB2 Connect продукт. Не можете да се свържете пряко към IBM хост или System i сървър на данни посредством Клиент на IBM сървър на данни.

Фигура 4 на страница 18 показва работна станция с инсталиран DB2 Connect Personal Edition, свързан пряко с хост или System i сървър на база данни.

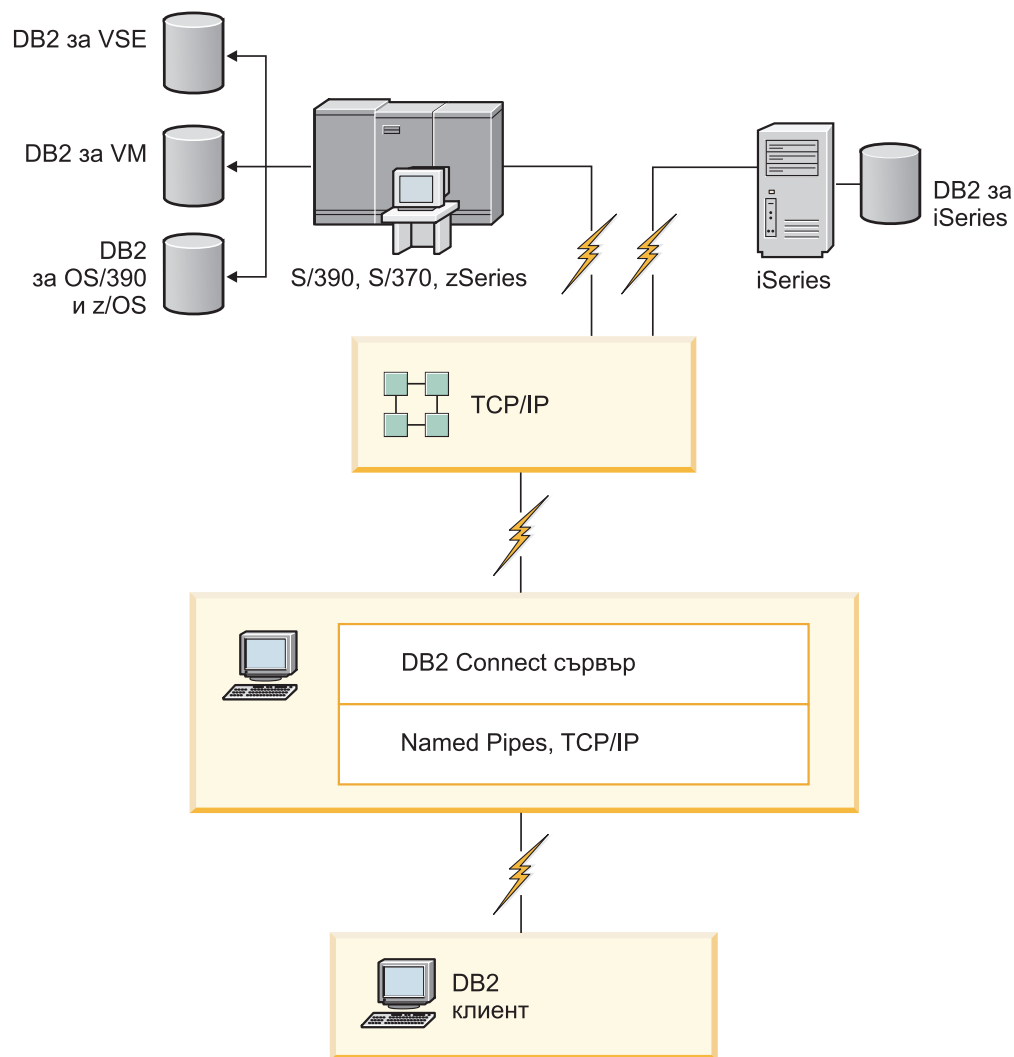


Фигура 4. Пряко свързване между DB2 Connect и хост или System i сървър на база данни

## DB2 Connect сървърни продукти като сървъри на свързваемост

Един DB2 Connect сървър позволява на множество клиенти да се свързват към хост или System i данни, и може да намали значително усилието, което се изисква за установяване и поддържане на достъпа до корпоративни данни. Фигура 5 на страница 19 илюстрира решението на IBM за среди, в които искате DB2 клиент да направи индиректно свързване към хост или System i сървър на база данни чрез DB2 Connect сървърен продукт, какъвто е DB2 Connect Enterprise Server Edition.

**Забележка:** Непреки свързвания се поддържат само с DB2 клиенти или JCC клиенти, изпълняващи се на Linux, UNIX, или Windows. Опитът за свързване към хост или System i сървър на база данни чрез DB2 Connect сървърен продукт посредством всеки друг клиент води до SQL1334 грешка.



Фигура 5. DB2 Connect Enterprise Server Edition

Ако TCP/IP свързване към DB2 Connect сървър бъде изгубено, клиентът ще опита автоматично да установи повторно свързването. Клиентът първо ще опита да установи отново свързването към оригиналния сървър. Ако свързването не бъде установено отново, клиентът ще се обърне в случай на отказ към алтернативен DB2 Connect сървър. (Алтернативният сървър се указва на потребителския модел на сървъра, и разположението му се връща на клиента по време на свързването.) Ако свързването към алтернативния сървър не бъде установено повторно, клиентът ще опита да установи отново свързването към оригиналния сървър. Клиентът ще продължи опитите да установи отново свързването, като сменя оригиналния сървър и алтернативния сървър, докато свързването бъде установено, или се надхвърли допустимия брой опити.

## DB2 Connect и web приложения

Web браузърът бързо се превръща в стандартен интерфейс за всичко от онлайн каталози до интранет приложения. За прости web приложения може да е достатъчен и само web сървър. За обемни приложения, които изискват достъп до бази данни и обработки на транзакции, IBM предлага решения, които използват DB2 Connect за управление на много голям брой едновременни транзакции чрез web.

## Предимства и ограничения на традиционното CGI програмиране

Приложенията за е-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 брауъра на потребителя. Например, онлайн каталог, в който потребителят може да провери достъпността и текущата цена на конкретни 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 програмите могат да бъдат написани на C++, или на скриптов език, какъвто е Perl или PHP.

Въпреки че CGI може да изглежда като идеалното решение за web-базирани приложения, има и някои значителни недостатъци. Средата за програмиране за CGI не е толкова комплексна, колкото другите API. В добавка, скалируемостта може да стане проблемна при мащабни e-commerce операции. При всяко обръщение към CGI приложение на web сървъра се създава нов процес. Всеки процес трябва да направи свое собствено свързване към базата данни, и да подаде своя собствена заявка. 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 позволяват на DB2 Connect потребителите да намалят броя на изпращаните към базата данни заявки.

Пуловете на свързвания намаляват честотата на свързванията и изключванията от и към базата данни.

## Използване на PHP като модул на Web сървър или плъгин

Въпреки че PHP може да се използва за CGI програмиране, обикновено се използва като модул на web сървър или плъгин. В многопроцесен web сървър като Apache, IBM DB2 драйверът за PHP може да се използва за облекчаване на проблема със скалируемостта. В многопроцесен web сървър има пул с процеси за повторната им употреба за обслужване на заявки на web сървъра. За да премахнете нуждата от изграждане на свързване към база данни за всяка web заявка, може да бъде създадено постоянно свързване. В тази среда, постоянно свързване може да съществува извън обхвата на отделния PHP скрипт. Свързването ще бъде използвано повторно, ако се изисква идентично свързване от последваща web заявка.

---

## DB2 Connect и IBM WebSphere

IBM WebSphere осигурява по-пълно e-business решение от възможните чрез традиционните средства за скриптиране, например PHP. WebSphere Application Servers не само изпълняват скриптовите възможности на PHP, но също позволяват да осигурявате комплексни и висококачествени услуги чрез web, като използвате сървлети, Active Server Pages, и Enterprise JavaBeans, и включват поддръжка за Web-базирани технологии като Java, TCP/IP, HTTP, HTTPS, HTML, DHTML, XML, MIME, SMTP, POP, и X.509, както и други. С WebSphere можете:

- Да използвате индустриалните стандарти за ускоряване на разработката и максимизиране на взаимодействието
- Да включвате технологиите на средства и рамкови приложения на трети страни
- Да анализирате производителността и използването на съдържанието на Web сайт
- Да мащабирате с лекота сайта си за повече потребители и да поддържате пропускателната способност
- Да разгръщате с лекота в множество основни операционни обкръжения (AIX, HP-UX, Linux, Novell NetWare, OS/390, z/OS, OS/400, Solaris операционна система, Microsoft Windows)
- 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. Сърцето на WebSphere решението е WebSphere сървърът на приложения.

WebSphere сървърът на приложения осигурява средата за три типа обекти. Единият са Java сървърните страници, които са аналог на Active Server Pages. Вторият компонент се състои от Java сървлети, а третият са enterprise JavaBeans. Enterprise JavaBeans е изгряващият стандарт за разгръщане на широкомащабни стабилни корпоративни приложения.

WebSphere приложенията могат да се разгръщат на същата платформа като web сървъра и DB2. В случая с DB2 Universal Database (UDB) за OS/390 и z/OS, DB2 за z/OS, DB2 за VM, DB2 за VSE, DB2 UDB за iSeries, и DB2 за i5/OS, WebSphere е разгрънат на същата платформа като DB2 Connect сървърния продукт.

Има няколко WebSphere решения, както и Rational Application Developer (RAD). За повече подробности идете на <http://www.ibm.com/software/webservers/appserv/was/>

---

## DB2 Connect като Java сървър на приложения

Много от недостатъците, асоциирани със скриптовите езици, могат да бъдат преодоляни, като вместо тях се използва Java. IBM осигурява аплети и приложения, които ви позволяват да използвате Java на всеки етап на web транзакция. Решенията, предоставяни от IBM, позволяват смесването на техники, което означава, че можете да използвате скриптовите решения като Perl DBI или Microsoft Active Server Pages с DB2, или да се придвижите към по-стабилна реализация, осигурявана от Java сървър на приложения като IBM WebSphere.

Има два интерфейса за приложно програмиране (API- Application Programming Interfaces) за Java програмисти. Първият, JDBC, се поддържа за използване на Java при разработка на Java аплети за данни, Java приложения, както и Java сървлети, 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 указва SQL в рамките на Java програма. DB2 може да използва и двата API, както от клиентската страна, така и от страна на сървъра на web транзакция.

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

За DB2 за OS/390 и z/OS, DB2 за VSE и VM, DB2 Universal Database (UDB) за iSeries, и DB2 за i5/OS, има два различни начина да се разгърне Java приложение. Можете да използвате преките възможности за свързване, предоставени от DB2 Connect Personal Edition с TCP/IP, или да изберете преминаване през DB2 Connect сървърен продукт, който ще осигури свързваемост към хост или System i сървър на 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 сървърен продукт и произволен стандартен Web сървър. Ако web сървърът и DB2 Connect не са физически на един и същи компютър, на web сървъра трябва да е инсталиран Клиент на IBM сървър на данни.

За DB2 за OS/390 и z/OS, ключовият компонент е DB2 Connect сървърен продукт, изпълняван на междинен сървър. Този компонент осигурява задействането на JDBC на сървъра в добавка към свързването с DB2 за OS/390 и z/OS, DB2 за VSE и VM, DB2 Universal Database (UDB) за iSeries, и DB2 за i5/OS сървъра. Отново не е необходим никакъв специален софтуер за web брауъра на клиента.

IBM осигурява изчерпателна поддръжка и инструментариум за разработката на Java приложения и аплети. За разработка на приложения в базите данни, DB2 Database Enterprise Developer Edition осигурява Rational Web Developer, IBM Data Studio, DB2 Embedded Application Server, Cloudscape Version 10.2, както и DB2 и DB2 Connect за тестване. Средствата на трети страни като NetBeans, Borland JBuilder или Symantec Visual Cafe също ще работят с решенията за база данни на IBM.

---

## DB2 Connect на web сървъра

IBM осигурява HTTP (Web) сървъри с всички DB2 Connect продукти. DB2 Connect сървърните продукти, като DB2 Connect Enterprise Server Edition, осигуряват готова поддръжка за Apache или Lotus Domino Go web сървърите, и могат да работят също с всеки друг web сървър, като Microsoft Internet Information Server или Netscape Enterprise Server.

Ако работите с DB2 семейството бази данни, изпълнявано на zSeries, System i, VM, и VSE системи, се изисква DB2 Connect сървърен продукт на Web сървъра. DB2 Connect сървърните продукти ще осигурят библиотеките и комуникационните интерфейси, необходими за активирането на достъпа на Web сървъри до тези хост и System i платформи. TCP/IP може да се използва за комуникации между Web сървъра и базата данни, изпълнявана на zSeries, System i, VM или VSE.

**Забележка:** Web решенията на IBM осигуряват възможността да се работи с няколко бази данни в рамките на един и същи Common Gateway Interface (CGI) скрипт (като PHP), или в рамките на една и съща транзакция в CGI скрипт.

### 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 Database за Linux, UNIX и Windows, и DB2 Universal Database (UDB) на OS/390 и z/OS, DB2 за z/OS, DB2 UDB за iSeries, DB2 за i5/OS, и DB2 за VSE споделят една и съща конвенция за параметри за извикване на запомнени процедури.

Както с редовните web интерфейсни скриптове, web браузърът подава формуляра на web сървъра, където се изпълнява web интерфейсният скрипт. 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 израза, които в противен случай щяха да се изпълняват отделно. Запомнените процедури намаляват броя съобщения, протичащи в двете посоки между web интерфейсният скрипт и подлежащата база данни.

Основното им предимство е, че намаляват мрежовия трафик между HTTP сървъра и DB2 базата данни.

---

## DB2 Connect и сървъри на приложения

Възходът на клиент-сървър приложенията позволи на дизайнерите на приложения да подобрят използваемостта и да намалят разходите за обучение, като предоставиха на приложенията графични потребителски интерфейси на платформи като Windows. В същото време позволява гъвкавостта на прехвърлянето на функции по управлението на бази данни на стабилни сървъри на бази данни на 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. DB2 Connect предоставя пълна поддръжка за двуслойни клиент-сървър приложения, където сървърите на база данни са DB2 Universal Database (UDB) за OS/390 и z/OS, DB2 за z/OS, DB2 UDB за iSeries, DB2 за i5/OS, 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 да се намали от разширено използване на запомнени процедури, другите недостатъци не могат да се решат лесно без промени в модела.



## Решение със 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. Клиентският слой обаче се допълва от един или повече средни слоя; типично един, отгук и названието *3-слоен*.

В 3-слоения модел клиентът се свежда до обработка на взаимодействието с потребителя и не съдържа никаква бизнес логика. Междинният слой 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 двуслойния 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 клиентския слой са силно намалени. На практика клиентският слой в 3-слоения 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-слоен сървър на приложения могат да се посочат в практиката днес. Почти всички доставчици на Enterprise Resource Planning (ERP) реализират своите приложения с помощта на 3-слоен 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

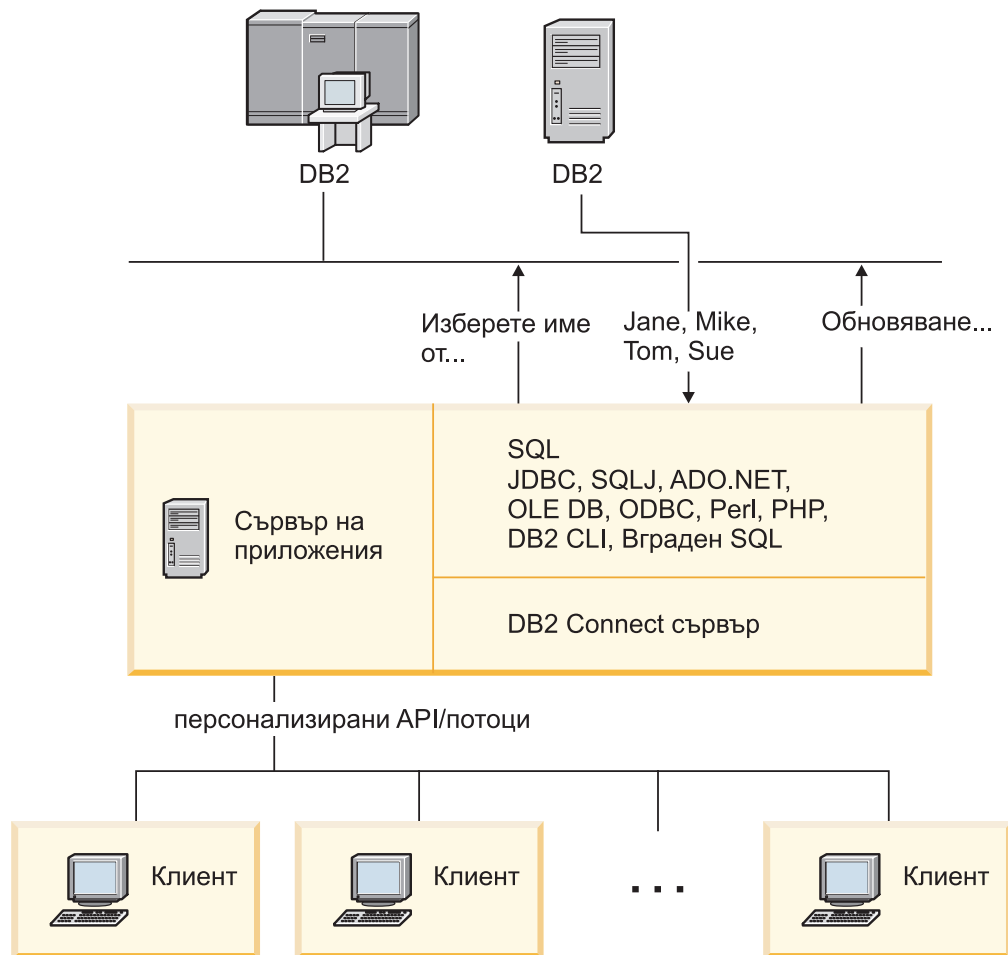
DB2 Connect сървърните продукти предоставят изчерпателна поддръжка за разгръщане на многослойни приложения. Поддръжката, предоставяна от DB2 Connect, включва разнообразни API, които могат да се използват за разработка на логика на приложение (ODBC, ADO.NET, DB2 CLI, вграден SQL, JDBC, SQLJ, Perl, PHP, и OLE DB), както и пълна комуникационна инфраструктура за взаимодействие със сървъри на бази данни от DB2 семейството.

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

Поддръжката за двуфазов протокол за комит, предоставяна от DB2 Connect, подsigурява интегритета на такива разпределени транзакции. Например, приложение може да обновява данни в DB2 за OS/390, и z/OS база данни и DB2 Database за Linux, UNIX и Windows в една и съща транзакция. Ако поддръжката на разпределена заявка е инсталирана и активирана, приложението може да чете от Oracle база данни и да обновява база данни от DB2 семейството в една и съща транзакция.

В следната диаграма, API, както и механизмът за свързваемост между между сървъра на приложения и сървърите на базата данни под него, се осигуряват чрез DB2 Connect сървърен продукт, какъвто е DB2 Connect Enterprise Server Edition.





Фигура 6. DB2 Connect поддръжка за сървъри на приложения

Разширените характеристики на DB2 Connect, като пулирането на свързвания, значително намаляват изискванията за ресурси на приложенията, и опростяват сървърната реализация на приложение.

### Конфигурации на DB2 Connect и сървър на приложения

Изисква се DB2 Connect сървърен продукт за употреба със сървъри на приложения. DB2 Connect Personal Edition не се поддържа, и не е лицензирано за употреба със сървъри на приложения. В добавка, купувачите, реализиращи сървъри на приложения, трябва да прегледат условията, осигурени с техните копия на DB2 Connect, за да разберат броя на потребителските лицензи, които трябва да бъдат закупени.

Има два метода за разгръщане на DB2 Connect в обкръжение на сървър на приложения. Може да се инсталира DB2 Connect сървърен продукт на всяко от следните:

- Машината на сървъра на приложения
- Отделна машина на сървър за комуникации

В повечето случаи, инсталирането на копие на DB2 Connect на същия сървър като сървъра на приложения, е предпочитаното решение. Инсталирането на DB2 Connect на сървъра на приложения му позволява да участва във всяка схема за възстановяване след срив или балансиране на натоварването, която може да се реализира от сървъра на приложения. Тази настройка може

потенциално да осигури по-добра производителност, тъй като премахва допълнителните мрежови точки, които се изискват, когато DB2 Connect се инсталира на отделен сървър. В добавка, администрирането може да бъде опростено, тъй като няма нужда от инсталиране и поддържане на допълнителен сървър.

Инсталирането на DB2 Connect на отделен сървър е добра опция в ситуации, когато вашият DB2 Connect сървърен продукт не е достъпен за операционната система или хардуерната платформа, на която се изпълнява сървърът на приложения.

---

## DB2 Connect и наблюдатели на обработката на транзакции

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 монитори.

### Обработка на транзакции

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.

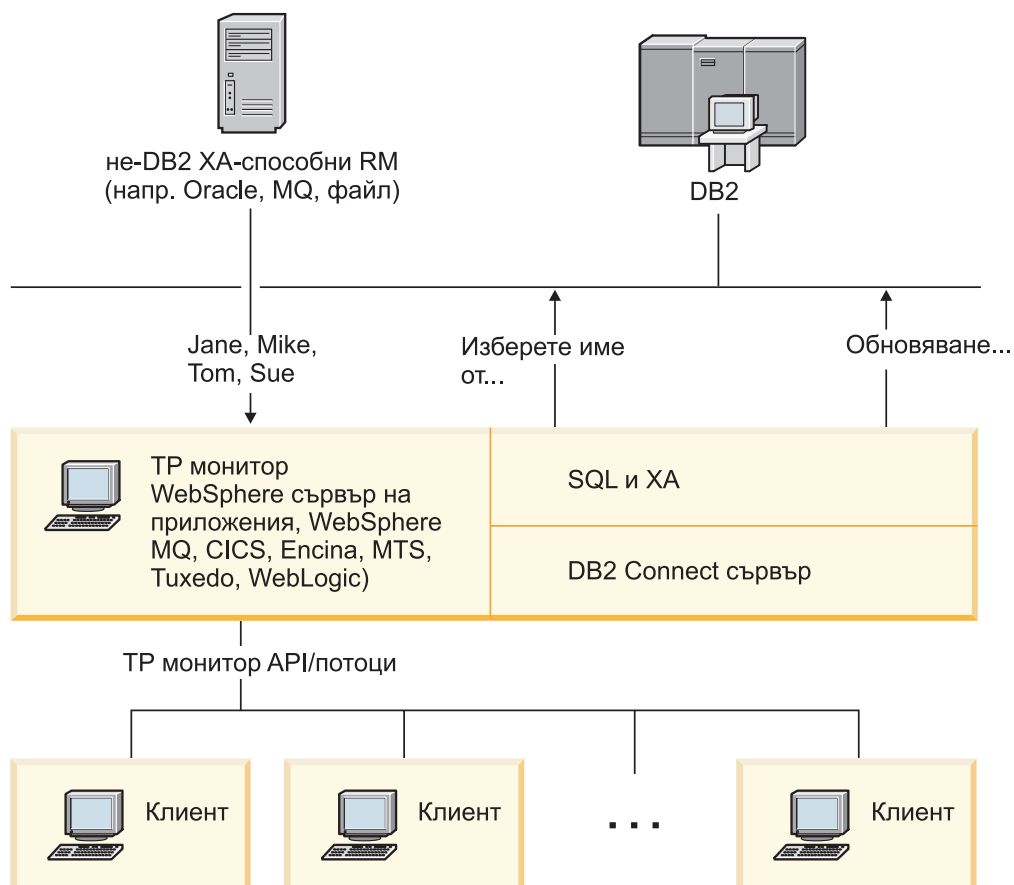
#### Интегритет на данни

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 транзакция представлява пряката цена на правене

на бизнес, цената на системата трябва да бъде минимална. DB2 Connect позволява на приложения под управлението на сървър на приложения, изпълняван на Linux, UNIX, и Windows, да изпълняват транзакции срещу отдалечени LAN, хост и System i сървъри на бази данни, и да координират тези транзакции чрез TP монитор.



Фигура 7. DB2 Connect поддръжка за TP монитори

В Фигура 7, API, както и механизмът за свързваемост между сървъра на приложения и сървърите на бази данни под него, се осигуряват от DB2 Connect сървърен продукт, какъвто е DB2 Connect Enterprise Server Edition.

## Примери за наблюдатели на обработката на транзакции

Най-разпространените TP монитори на пазара днес са:

- IBM WebSphere сървър на приложения
- IBM WebSphere MQ
- IBM TxSeries CICS
- IBM TxSeries Encina Monitor
- BEA Tuxedo
- BEA WebLogic
- Microsoft Transaction Server (MTS)

Отдалечените System i, zSeries, и LAN сървъри на бази данни могат да се използват в транзакции, координирани от тези TP монитори.

## **X/Open модел за разпределена обработка на транзакции (DTP)**

Може да се изисква от приложение, изпълняващо бизнес логика, да обновява множество ресурси в отделна 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 и z/OS, и другата е база данни на 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 интерфейса, се нарича *XA съвместим мениджър на транзакции (TM)*. 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. Отдалечените хост, System i, и DB2 LAN-базирани бази данни, когато се достъпват чрез DB2 Connect, са XA съвместими RM. Следователно, всеки TP монитор, който има XA съвместим TM, може да използва хост, System i, и LAN-базирани DB2 бази данни в бизнес приложения, изпълняващи транзакции.

---

## Раздел 2. DB2 Connect справочник



---

## Глава 4. Обновяване на директории на бази данни

DB2 Connect използва следните директории за управление на информацията за свързване на базата данни:

- *директория на системната база данни*, съдържаща информация за име, възел и разпознаване за всяка база данни, достъпвана от DB2 Connect.
- *директория на възли*, която съдържа информация за мрежов адрес и комуникационен протокол за всеки хост или System i сървър на база данни, който се достъпва от DB2 Connect.
- *database connection services (DCS) директория*, която съдържа информация, специфична за базата данни на хост или System i сървър на бази данни.

### **Забележка:**

1. Преди да обновите тези директории, трябва да конфигурирате комуникациите на хоста или System i сървъра на бази данни и работните станции.
2. Директориите на бази данни могат да бъдат обновени посредством Configuration Assistant (CA).

За да обновите директории на база данни:

1. Съберете информация за директорията на базата данни, като използвате работния лист за персонализиране на директория
2. Отнесете се към темата “Обновяване на директориите с информация за отдалечените машини на сървъри на бази данни” в Центъра за управление

---

## Стойности на системната директория на базата 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 System i сървъра на база 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**

Указва къде ще се изпълнява проверката на потребителското име и парола за свързвания, произлизащи от DB2 Connect сървър. Валидните опции са: SERVER, SERVER\_ENCRYPT, CLIENT, KERBEROS, и DATA\_ENCRYPT. Няма поддръжка за тип разпознаване GSSPLUGIN в системната директория на базата данни.

---

## Стойности на директорията на възлите

Можете да определите следната информация в директорията на възлите:

### Име на възел

Прякор за системата на хост или System i сървър на база данни, на която е разположена 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 еднобайтови буквено-цифрови символи, включително числен знак (#), знака at (@), знака за долар (\$), и долно тире (\_). Не може да започва с долно тире или число.

### Протокол

Трябва да бъде TCP/IP.

### Тип на защита

Типът на защитните проверки, които ще се изпълнят. 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.

### 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.

За DB2 за OS/390 и z/OS отдалечени хостове, името на хост се появява в DSNL004I съобщението (DOMAIN=име-на-хост), когато се стартира Distributed Data Facility (DDF). Командата -DISplay DDF може също да бъде използвана.

Ако достъпвате z/OS група за поделяне на данни, името на домейн трябва да съответства на динамичния VIPA адрес на DB2 групата. Този адрес води до най-малко заредения DB2 член. За достъпване на специфичен член използвайте специфичния динамичен VIPA адрес на DB2 член и изключете sysplex маршрутизирането. Съобщението DSNL004I на всеки член извежда специфичното за члена име на домейн.

### 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.

За DB2 за OS/390 и z/OS отдалечени хостове, номерът на порт е дефиниран в Boot Strap Data Set (BSDS) като PORT и също се предоставя в DSNL004I съобщението (TCPPOINT=номер-на-порт), когато се стартира Distributed Data Facility (DDF). Командата -DISplay DDF може също да бъде използвана.

Ако достъпвате z/OS група за поделяне на данни, името на домейн трябва да съответства на динамичния VIPA адрес на DB2 групата. Този адрес води до най-малко заредения DB2 член. За достъпване на специфичен член използвайте специфичния динамичен VIPA адрес на DB2 член и изключете sysplex маршрутизирането. Съобщението DSNL004I на всеки член извежда специфичното за члена име на домейн.



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

---

## Стойности на DCS директорията

Можете да определите следната информация в DCS директорията:

### Име на базата данни

Дефиниран от потребителя прякор за хост или System i сървър на бази 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 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 база данни приемник

Базата данни на системата на хост или System i сървър на база данни, като следва:

#### OS/390 и z/OS

DB2 Universal Database за z/OS и OS/390 подсистема, идентифицирана от своето LOCATION NAME или от един от псевдонимите за LOCATION имена, дефинирани на z/OS сървъра.

LOCATION NAME 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
```

множествени LOCATION NAME също се дефинират в Boot Strap Data Set (BSDS), както и DSNL004I съобщение (LOCATION=разположение), което се изписва при стартирането на Помощно средство за разпределени данни (DDF). Командата -DISplay DDF може също да бъде използвана.

Ако достъпвате z/OS група за поделяне на данни, името на домейн трябва да съответства на динамичния VIPA адрес на DB2 групата. Този адрес води до най-малко заредения DB2 член. За достъпване на специфичен член използвайте специфичния динамичен VIPA адрес на DB2 член и изключете sysplex маршрутизирането. Съобщението DSNL004I на всеки член извежда специфичното за члена име на домейн.

#### VSE или VM

Името на базата данни (DBNAME)

#### OS/400 и z/OS

Името на релационна база данни (RDBNAME)

**Други** За Windows, Linux, и UNIX операционни системи, псевдонимът на базата данни е разположен в директорията на базата данни.

### Параметричен низ

Ако искате да промените настройките по подразбиране, определете следните параметри в посочения ред.

#### *файл-преобразуване*

Името на файла със SQLCODE съответствията, който заменя SQLCODE съответствията по подразбиране. За да изключите SQLCODE съответствията, определете **NOMAP**.

**Забележка:** При обработка на заявка, DRDA сървърът връща данни във вид на набор редове, които представляват резултатния набор. С всеки ред се връща също SQLCA, обикновено съдържащ нула или положителен sqlcode (като +12 или +802). Ако използвате персонализиран файл за преобразуване на DB2 Connect сървъра, тези положителни sqlcode няма да бъдат преобразувани, ако се съдържат в персонализирания файл за преобразуване и имат модифицирани преобразувания (например, преобразуват се в различен sqlcode или имат модифицирани преобразувания на токени).

Важно е да се наблегне, че:

1. Положителните sqlcode-ове представляват предупреждения, за разлика от отрицателните, които указват условия на грешка. Всички отрицателни sqlcode ще бъдат преобразувани при всякакви обстоятелства, независимо кой файл за преобразуване се използва. Всички положителни sqlcode, включени в потребителския файл за преобразуване и насочени към себе си без изменения, също ще бъдат винаги преобразувани. Също така, положителните sqlcode-ове, които не се съдържат в персонализирания файл за преобразуване на DB2 Connect сървъра, ще бъдат винаги преобразувани.
2. Ако използвате файла за преобразуване по подразбиране или се свързвате пряко към хост базата данни, преобразуването винаги ще бъде изпълнявано за всички sqlcode-ове.

**,D** Това е вторият позиционен параметър. Ако е указан, приложението ще се изключи от базата данни на хост или System i сървър на база данни, когато бъде върнат един от следните SQLCODES:

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.

INTERRUPT\_ENABLED се прилага само, ако крайният сървър не поддържа прекъсвания. Ако сървърът поддържа DRDA прекъсване на потока, DB2 Connect просто ще предаде заявката за прекъсване на сървъра.

Ако INTERRUPT\_ENABLED е конфигурирано в DCS директорията на DB2 Connect работната станция, и клиентско приложение подаде заявка за прекъсване, докато е свързано към хост или System i сървър на база данни, DB2 Connect ще изпълни прекъсването, като отстрани свързването и изпълни ролбек на единицата работа. Това поведение при прекъсване се поддържа на AIX и Windows.

Приложението ще получи sqlcode (-30081), който показва, че е прекъсната връзката към сървъра. Приложението трябва да установи нова връзка с хост или System i сървъра на база данни, за да обработи допълнителни заявки на базата данни. На платформи, различни от AIX V5.2 и по-късни, и Windows, DB2 Connect не поддържа опцията за автоматично прекъсване на свързването, когато използващото я приложение получи заявка за прекъсване.

**Забележка:** Тази поддръжка работи на TCP/IP свързвания на всички платформи. Клиентът може да прекъсне сокета, но - в зависимост от реализацията на сървъра - може да има или не получени неща. DB2 Universal Database за z/OS и OS/390 използва асинхронни извиквания на сокет и следователно може да открива загубата на връзка и да изпълнява ролбек на продължителни SQL изрази в процес на изпълнение.

### „„„„SYSPLEX

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

### „„„„LOCALDATE=<стойност>

Този параметър, седмият позиционен параметър, се използва за активиране поддръжка на формат за датата от DB2 Connect. Това е постигнато чрез използване на маска за данни за <стойност>, както следва:

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

```
catalog TCP/IP node nynode remote myhost server myport
catalog dcs database nydb1 as new_york
catalog database nydb1 as newyork1 at node nynode
сървър за разпознаване
```

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. В този случай, тъй като се използва CLP, и самият параметричен низ е указан с двойни кавички, стойността LOCALDATE трябва да бъде указана между два чифта двойни кавички. Забележете използването на символа за освобождаване на операционната система "\" (обратно наклонена черта), за да се подсигури, че двойните кавички няма да бъдат премахнати от LOCALDATE спецификацията.

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

Псевдонимът на базата данни *newyork2* ви дава достъп до същата база данни на хост, но в добавка има указана 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 JIS).
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.

,,,,,,**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 сървър на базата данни BiDi CCSID. Например:

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

където xyz представлява замяната на CCSID.

## Работен лист за персонализиране на директория

Работният лист за персонализиране на директория показва информацията, която трябва да съберете. Може да сметете за удобно да направите копие на работния лист, и да въведете стойностите за вашата система.

### Параметри на директорията на възлите

Таблица 1. Параметри на директорията на възлите

Параметър	Пример	Ваша стойност
Име на възел	DB2NODE	
Име на отдалечен хост (TCP/IP възел)	ZOSHOST	
Сървър (Име на TCP/IP услуга или номер на порт)	db2inst1c (или 446)	

#### Забележка:

1. Подразбирацията се номер на TCP/IP порт за DRDA е 446
2. Освен ако знаете, че хост или System i сървърът на базата данни поддържа SECURITY SOCKS, не указвайте SECURITY за TCP/IP възел.

### Параметри на DCS директория

Таблица 2. Параметри на DCS директория

Параметър	Пример	Ваша стойност
Име на базата данни	DB2DB	
Име на база данни приемник	NEW_YORK3	
Средство за обработка на запитвания		

Таблица 2. Параметри на DCS директория (продължение)

Параметър	Пример	Ваша стойност
Параметричен низ	" ,,,,,,LOCALDATE=\\\"YYMMDD\\\"\\\""	

## Параметри на Системната директория на базата данни

Таблица 3. Параметри на Системната директория на базата данни

Параметър	Пример	Ваша стойност
Име на базата данни	DB2DB	
Псевдоним на базата данни	NYC3	
Име на възел	DB2NODE	
Разпознаване	SERVER	

## Дефиниране на множество записи за една и съща база данни

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 искате да изключите SQLCODE преобразуването за приложения, пренесени от хост или System i сървър на база данни, но да приемете преобразуването по подразбиране за приложенията, разработени за клиент/сървър обкръжението. 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.

## Боравене с ViDi данни

Следният раздел се отнася само за OS/390 и z/OS сървъри. Тази характеристика трябва да не се активира за DB2 за i5/OS сървър, тъй като вече е осигурена пълна ViDi поддръжка.

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

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

Тъй като подразбиращите се стойности са различни на различните платформи, се появяват проблеми при изпращане на DB2 данни от една платформа на друга. Например, Windows платформите използват LOGICAL UNSHAPED данни, докато OS/390 или z/OS данните обикновено са в SHAPED VISUAL формат. Следователно,

без никаква поддръжка за BiDi атрибути, данните, изпращани от DB2 за OS/390 и z/OS към DB2 Connect на Windows, се извеждат неправилно.

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. Понастоящем обаче никой хост DB2 продукт не поддържа BiDi-специфични CCSID или BiDi трансформация на разположение. 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 BiDi трансформация на разположение на изходящите данни към сървър на база данни, трябва да бъде заменен BiDi CCSID на базата данни на сървъра. 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.

Представете си Клиент на IBM сървър на данни на иврит, изпълняващ CCSID 62213 (BiDi низов тип 5), и че искате да достъпите DB2 хост база данни, изпълняваща CCSID 424 (BiDi низов тип 4). Знаете обаче, че данните, съдържащи се в DB2 хост базата данни, вместо това са базирани на CCSID 62245 (BiDi низов тип 10).

В тази ситуация има два проблема. Първият е, че DB2 хост базата данни не прави разлика между BiDi низови типове с CCSIDs 424 и 62245. Вторият проблем е, че DB2 хост базата данни не разпознава CCSID от 62213 на Клиент на IBM сървър на данни. Тя поддържа само CCSID 62209 (BiDi низ тип 10), който е базиран на същата кодова страница като CCSID 62213.

Ще трябва да проверите дали данните, изпратени на DB2 хост базата данни, са във формат BiDi низ тип 6, като начало, и да уведомите DB2 Connect, че трябва да изпълни BiDi трансформация на разположение върху данните, които получава от DB2 хост базата данни. Ще използвате следното каталогизиране за DB2 хост базата данни:

```
catalog dcs database nydb1 as TELAVIV parms ",,,,,,,,,BIDI=62245"
```

Това указва на DB2 Connect да замени CCSID на DB2 хост базата данни от 424 с 62245. Тази замяна включва следните обработки:

1. DB2 Connect ще се свърже с DB2 хост базата данни, като използва CCSID 62209 (BiDi низов тип 10).
2. DB2 Connect ще изпълни BiDi трансформация на разположение на данните, които са за прашане на DB2 хост база данни от CCSID 62213 (BiDi низов тип 5) до CCSID 62209 (BiDi низов тип 10).
3. DB2 Connect ще изпълни BiDi трансформация на разположение върху данните, които получава от DB2 хост базата данни от CCSID 62245 (BiDi низов тип 10) до CCSID 62213 (BiDi низов тип 5).

#### **Забележка:**

1. Променливата на обкръжението или регистърната стойност DB2BIDI трябва да се установи на YES, за да влезе в сила параметърът BiDi. DB2BIDI трябва да бъде настроена на DB2 Connect работната станция, където се каталогизира записът за

DCS директория на база данни. За приложения, изпълняващи се на клиент, отдалечен за DB2 Connect сървър, променливата DB2BIDI трябва да бъде настроена също и на този клиент.

2. Ако искате DB2 Connect да изпълнява трансформация на разположение на данните, които ще изпраща на DB2 хост базата данни, дори ако не трябва да подменяте нейното CCSID, пак трябва да добавите BIDI параметъра в полето PARMS на DCS директорията на базата данни. В този случай CCSID, което трябва да предоставите, ще бъде подразбиращото се за DB2 хост базата данни CCSID.
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 вместо тях използвайте хост променливи, където е възможно.

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.

## Спецификации на параметрични низове

Следното са примери за DCS параметри (всеки ред е набор параметри):

```
NOMAP
/u/username/sql1lib/map/dcs1new.map,D
,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:** Трябва да използвате символа за освобождаване на операционната система "\ (обратно наклонена черта), когато използвате CLP от командния ред на операционната система на UNIX системи, поради нуждата от указване на два чифта кавички, когато се указва LOCALDATE маска в параметричния низ. 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



---

## Глава 5. Защита на DB2 Connect

Разпознаването на потребители е важно, когато се използва DB2 Connect, защото потребителите могат да бъдат локални или отдалечени спрямо DB2 Connect и спрямо базата данни, която те желаят да достъпят. Доверените свързвания и поддръжката на Kerberos са представени, наред със съображенията за защита за бази данни на хост машини.

---

### Доверени свързвания чрез DB2 Connect

Някои DB2 сървъри на бази данни поддържат доверени контексти. Един *доверен контекст* позволява на администратора на база данни, наред с други неща, да дефинира условия, според които на клиентско приложение да се позволява да създаде доверено свързване. На *доверено свързване* се позволява да прави неща, които нормалното свързване не може.

Има два типа доверени свързвания, явни и неявни. Когато създавате свързване, дали получавате явно доверено свързване, неявно доверено свързване, или обикновено свързване, зависи от това дали сте заявили доверено свързване, и дали свързването отговаря на критериите, дефинирани в доверения контекст на сървъра, както са обобщени в Таблица 4.

Таблица 4. Какви типове свързване се получават от различни комбинации от действия

	Свързването отговаря на критериите на сървъра за доверено	Свързването не отговаря на критериите на сървъра за доверено
Заявили сте доверено свързване	Явно доверено свързване	Обикновено свързване и се връща предупреждение SQL20360W (SQLSTATE 01679).
Не сте заявили доверено свързване	Неявно доверено свързване	Обикновено свързване

Едно *неявно доверено свързване* е идентично на обикновеното свързване, с изключение на това, че предоставя временни права за роля на потребителя, докато се използва това свързване. Правата за роля, които се предоставят (ако има такива), се посочват в доверения контекст, който прави свързването доверено.

Неявни доверени свързвания могат да се създават от всяко приложение, което се свързва посредством DB2 Connect. Неявните доверени свързвания се създават и използват по същия начин, по който се създават и използват обикновените свързвания. Това означава, че не са необходими изменения в кода, за да може съществуващо приложение да използва предимствата на неявните доверени свързвания, стига приложението да се свързва чрез DB2 Connect.

Едно *явно доверено свързване* предоставя временни права за роля на потребителя по същия начин, по който и неявното доверено свързване. В добавка, явното доверено свързване ви позволява да промените използвания идентификатор за оторизация при изпълнение на действия по това свързване. Промяната на идентификатор за оторизация на явно доверено свързване се споменава още като *превключване на потребители*. Идентификаторите за оторизация, към които можете да превключвате,

и дали даден идентификатор за оторизация изисква парола при превключване към него, се дефинира като част от доверения контекст, който е позволил създаването на довереното свързване.

Превключването на потребители може значително да намали натоварването при поделяне на свързване между няколко потребителя, особено за потребителски имена, които не изискват парола, защото в този случай сървърът на базата данни не разпознава идентификатор за оторизация. Когато използвате тази характеристика обаче, трябва да сте напълно уверени, че приложението ви не позволява превключване на идентификатор за оторизация, без да валидира и разпознае този идентификатор за оторизация. В противен случай създавате пробив в сигурността на вашата система.

Явните доверени свързвания могат да бъдат създавани, и потребителят може да бъде превключван, когато се извършва свързване чрез DB2 Connect с използване на CLI или JDBC, включително XA установени свързвания. Създаването на явно доверено свързване и превключването на потребители изисква настройването на специални атрибути на свързване. Това означава, че съществуващи приложения ще трябва да бъдат модифицирани, за да използват предимствата на явните доверени свързвания.

Освен вече споменатите разлики, можете да използвате доверено свързване (независимо явно или неявно) по същия начин, по който бихте използвали обикновено свързване. Трябва със сигурност обаче да прекъснете изрично явно доверено свързване, когато свършите да го използвате, дори ако е в увредено или прекъснато състояние. В противен случай използваните от свързването ресурси може да не бъдат освободени. Това не е проблем при неявните доверени свързвания.

#### **Забележка:**

1.

**Важно:** Превключването на потребители без предоставяне на парола заобикаля разпознаването на сървъра на базата данни. Вашето приложение не трябва да позволява превключване на идентификатор за оторизация без парола, освен ако приложението вече е валидирало и разпознало този идентификатор за оторизация. В противен случай би се създаде пробив в защитата.

2. Явните доверени свързвания не следва да използват CLIENT разпознаване. Това не се отнася за неявните доверени свързвания.
3. Приложенията, които използват явни доверени свързвания, трябва да се изпълняват на защитени машини, предпазвани с парола, и достъпни само за оторизиран персонал. Това не се отнася за неявните доверени свързвания.

## **Създаване и прекратяване на доверено свързване чрез CLI**

Ако сървърът на база данни, към който се свързвате, е конфигуриран да го позволява, можете да създадете явно доверено свързване, когато се свързвате чрез CLI.

Тази процедура приема, че не използвате мениджър на XA транзакции. Ако използвате мениджър на XA транзакции, трябва само да се уверите, че мениджърът на транзакции е конфигуриран да настройва конфигурационната стойност TCTX на TRUE, когато извиква ха\_orep. Ако това се направи, всяко свързване, което може да бъде явно доверено свързване, ще бъде. За да проверите, дали едно свързване е явно доверено свързване, вижте стъпка 3.

- Базата данни, към която се свързвате, трябва да поддържа доверени свързвания.

- Трябва да бъде дефиниран доверен контекст, който ще разпознае клиента като благонадежден.
- Трябва да знаете системния идентификатор за оторизация, който е указан в доверения контекст. Системният идентификатор за оторизация на доверено свързване е идентификаторът за оторизация, който предоставяте на сървъра като потребителско име при създаване на свързването. За да бъде доверено вашето свързване от определен доверен контекст, системният идентификатор за оторизация трябва да бъде посоченият в този доверен контекст. Попитайте вашия защитен администратор за валиден системен идентификатор за оторизация и за паролата за този идентификатор.

Примерите в тези инструкции използват езика C, и допускат, че conn е показалец към валиден, но не свързан, указател за свързване. Приема се, че променливата rc има тип данни SQLRETURN.

1. В добавка към задаването на всякакви атрибути за свързване, които бихте задали за обикновено свързване, настройте атрибута на свързване SQL\_ATTR\_USE\_TRUSTED\_CONTEXT на SQL\_TRUE с обръщение към функцията SQLSetConnectAttr.
 

```
rc = SQLSetConnectAttr(
    conn,
    SQL_ATTR_USE_TRUSTED_CONTEXT, SQL_TRUE, SQL_IS_INTEGER
);
```
2. Свържете се към базата данни, както бихте при нормално свързване, като извикате функцията SQLConnect за потребителския модел. Използвайте системния идентификатор за оторизация като потребителско име и паролата му като парола. Не забравяйте да проверите за грешки и предупреждения, особено тези, изброени в таблицата Таблица 5.

Таблица 5. Грешки, указващи неуспех при създаването на доверено свързване

SQLCODE	SQLSTATE	Значение
SQL20360W	01679	Свързването не може да бъде установено като доверено. Вместо това е установено като обикновено свързване.

Ако няма грешки или предупреждения, които да посочат различно, свързването е установено, и е явно доверено свързване.

3. (По желание) Можете да се уверите, че установено свързване е явно доверено свързване, като проверите стойността на атрибута за свързване SQL\_ATTR\_USE\_TRUSTED\_CONTEXT посредством функцията SQLGetConnectAttr. Ако е настроен на SQL\_TRUE, свързването е явно доверено свързване.
4. Когато свършите да използвате свързването, трябва да не забравяте изрично да го прекъснете, дори ако е в увредено или прекъснато състояние. Ако не прекъснете изрично доверено свързване, някои от ресурсите, използвани от свързването, може да не бъдат освободени.

**Забележка:**

1. Явните доверени свързвания не следва да използват CLIENT разпознаване. Това не се отнася за неявните доверени свързвания.
2. Приложенията, които използват явни доверени свързвания, трябва да се изпълняват само на защитени компютри, предпазвани с парола, и достъпни само за оторизиран персонал. Това не се отнася за неявните доверени свързвания.

## Превключване на потребители на доверено свързване чрез CLI

Можете да превключвате потребители на явно доверено свързване чрез интерфейса на команден ред (CLI). За описание какво означава да се превключват потребители с използване на доверено свързване, вижте темата в свързаните връзки.

- Свързването трябва да е създадено успешно като доверено свързване.
- Явното доверено свързване не трябва да е в транзакция.
- Довереният контекст, който позволява явното доверено свързване да бъде създадено, трябва да бъде конфигурирано да позволява превключване на идентификатора за оторизация, към който превключвате.

Примерите в тези инструкции използват езика C, и допускат, че `conn` е показалец към свързано явно доверено свързване. Приема се, че променливата `rc` има тип данни `SQLRETURN`. Променливата `newuser` се приема да е показалец към символен низ, който държи идентификатора за разпознаване на потребителя, към който искате да се превключите. Приема се, че променливата `passwd` е показалец към символен низ, който съдържа паролата за този идентификатор за оторизация.

1. Извикайте функцията `SQLSetConnectAttr`, за да настроите атрибута `SQL_ATTR_TRUSTED_CONTEXT_USERID`. Настройте го на идентификатора за оторизация, на който искате да превключите.

```
rc = SQLSetConnectAttr(  
    conn,  
    SQL_ATTR_TRUSTED_CONTEXT_USERID, нов-потребител, SQL_NTS  
);  
//Проверете за грешки
```

Не забравяйте да проверите за грешки и предупреждения, особено тези, изброени в таблицата Таблица 6.

Таблица 6. Грешки, показващи неуспех при задаването на нов идентификатор за оторизация при превключване на потребители

SQLCODE	Значение
CLI0106E	Свързването не е свързано.
CLI0197E	Свързването не е доверено свързване.
CLI0124E	Има проблем с предоставената стойност. Проверете например дали не е null или твърде дълга.
CLI0196E	Свързването е въввлечено в единица работа, която не позволява превключването на потребители. За да можете да превключвате потребители, свързването трябва да не е в транзакция.

2. (По желание, освен ако довереният контекст, който е позволил това доверено свързване, изисква парола за идентификатора за оторизация, на който превключвате) Извикайте функцията `SQLSetConnectAttr`, за да настроите атрибута `SQL_ATTR_TRUSTED_CONTEXT_PASSWORD`. Настройте го на паролата за новия идентификатор за оторизация.

```
rc = SQLSetConnectAttr(  
    conn,  
    SQL_ATTR_TRUSTED_CONTEXT_PASSWORD, парола, SQL_NTS  
);  
//Проверете за грешки
```

Не забравяйте да проверите за грешки и предупреждения, посочени в таблицата Таблица 6, и в таблицата Таблица 7 на страница 45.

Таблица 7. Грешки, показващи неуспех при задаването на парола при превключване на потребители

SQLCODE	Значение
CLI0198E	Атрибутът SQL_ATTR_TRUSTED_CONTEXT_USERID още не е бил зададен.

- Продължете, както при обикновено свързване. Ако използвате мениджър на ХА транзакции, превключването на потребител се опитва като част от следващата заявка, в противен случай превключването на потребител се опитва точно преди задействането на следващото извикване на функция, която достъпва базата данни (например, SQLExecDirect). При всеки случай, в добавка към грешките и предупрежденията, които нормално би следвало да проверите, не забравяйте да проверите за грешките, изброени в Таблица 8. Грешките в Таблица 8 показват, че превключването на потребител е неуспешно.

Таблица 8. Грешки, показващи неуспех при превключване на потребител

SQLCODE	Значение
SQL1046N	Довереният контекст, който е позволил това доверено свързване, не е конфигуриран да позволява превключване към идентификатор за оторизация, към който опитвате да превключите. Няма да можете да превключвате на този идентификатор за оторизация, докато доверения контекст не бъде променен.
SQL30082N	Предоставената парола не е правилна за идентификатора за оторизация, към който превключвате.
SQL0969N със собствена грешка -20361	Има ограничение на ниво база данни, което не ви позволява да превключите към потребителя.

Ако превключването на потребител не успее, свързването ще бъде в несвързано състояние, докато успешно превключите на друг потребител. Можете да превключвате потребители на доверено свързване в несвързано състояние, но не можете да достъпите сървъра на база данни с него. Свързване в несвързано състояние ще остане в него, докато успешно превключите потребители на него.

Забележки:

- Важно:** Превключването на потребители без предоставяне на парола заобикаля разпознаването на сървъра на базата данни. Вашето приложение не трябва да позволява превключване на идентификатор за оторизация без парола, освен ако приложението вече е валидирано и разпознало този идентификатор за оторизация. В противен случай би се създаде пробив в защитата.
- Указването на стойност NULL за атрибута SQL\_ATTR\_TRUSTED\_CONTEXT\_USERID е еквивалентно на указването на системния идентификатор за оторизация за довереното свързване (потребителския идентификатор, използван, когато е било създадено явното доверено свързване).
- Когато настроите успешно стойността на атрибута на свързване SQL\_ATTR\_TRUSTED\_CONTEXT\_USERID на явно доверено свързване, свързването автоматично се рестартира. Резултатът от рестартирането е както при създаване на ново свързване с оригиналните атрибути на това свързване. Това рестартиране се случва дори, ако стойността, на която сте настроили атрибута за свързване, е системния идентификатор за оторизация, или същата стойност, която атрибутът съдържа понастоящем.

4. Ако атрибутът `SQL_ATTR_TRUSTED_CONTEXT_PASSWORD` е настроен, паролата ще бъде разпозната при обработката на превключване на потребител, дори ако довереният контекст, който е позволил довереното свързване, не изисква разпознаване на превключения потребител за този идентификатор за оторизация. Това води до ненужно допълнително натоварване. Това правило не се отнася до системния идентификатор за оторизация на доверения контекст. Ако системният идентификатор за оторизация на доверения контекст не изисква разпознаване, когато превключвате към него, той не се проверява дори, ако е осигурена парола.

---

## Съображения за разпознаване за DB2 Connect

Като DB2 Connect администратор, в сътрудничество с администратора на вашата хост или System i база данни, можете да определите къде се проверяват потребителските имена и пароли:

- При клиента
- На хост или System i сървър
- Еднократно влизане и разпознаване чрез система на трета страна (Kerberos).

**Забележка:** Ако отдалеченият клиент не е указал тип разпознаване, клиентът приема по подразбиране `SERVER_ENCRYPT`. Ако този тип не се поддържа от сървъра, клиентът ще опита повторно, като използва подходящата стойност, върната от сървъра. За да помогнете при оптимизирането на производителността, винаги указвайте типа разпознаване на клиента, за да избегнете допълнителния мрежов трафик.

Считано от DB2 Connect Версия 8.2.2 (еквивалентна на Версия 8.1 FixPak 9) шлюзът вече не е пасивен участник в преговорите за разпознаване. Вместо това шлюзът поема активна роля. Типът разпознаване, посочен в записа на директорията на базата данни за шлюза подменя типа разпознаване, каталогизиран на клиента. Клиентът, шлюзът и сървърът трябва да посочат съвместими типове. Ако каталогизираният тип разпознаване на шлюза не е бил посочен в записа на директорията на базата данни, разпознаването `SERVER` ще бъде подразбиращият се тип, изискван на сървъра. Въпреки това, ще се изпълни преговаряне между клиента и сървъра, ако сървърът не поддържа `SERVER` разпознаване. Това поведение е противоположно на клиентското, което по подразбиране е `SERVER_ENCRYPT`, ако не е бил посочен тип разпознаване.

Каталогизираният на шлюза тип разпознаване не се използва, ако опцията `DB2NODE` или `SQL_CONNECT_NODE` на Set Client API е била зададена на клиента. В тези случаи преговарянето е стриктно между клиента и сървъра.

Следните типове разпознаване са позволени с DB2 Connect:

### **CLIENT**

Потребителското име и паролата се проверяват на клиента.

### **SERVER**

Потребителското име и паролата се проверяват на хост или System i сървър на база данни.

### **SERVER\_ENCRYPT**

Що се отнася до `SERVER` разпознаването, потребителското име и парола се проверяват на хоста или System i сървър на базата данни, но прехвърлените ключови думи са криптирани на клиента.

### **DATA\_ENCRYPT**

Предоставя възможността да се криптират потребителски данни при комуникации клиент/сървър.

## KERBEROS

Позволява на клиента да влезе в сървъра, като използва Kerberos разпознаване вместо традиционната комбинация от идентификатор и парола. При този тип разпознаване е необходимо едновременно и за сървъра, и за клиента да е активирана Kerberos защита.

Kerberos разпознаването е уникално в това, че клиентът не изпраща директно на сървъра идентификатор и парола на потребител. Вместо това Kerberos действа като механизъм за разпознаване на друга фирма. Потребителят въвежда еднократно идентификатор и парола на клиентския терминал, а Kerberos проверява валидността на тази регистрация. След това Kerberos автоматично и сигурно изпраща оторизацията на потребителя на всички заявени локални и мрежови услуги. Това означава, че потребителят не трябва да въвежда повторно идентификатор и парола, за да влезе в отдалечения DB2 сървър. Способността за еднократно влизане, обезпечавана от Kerberos разпознаването, изисква от DB2 Connect и от сървъра на базата данни, към който се осъществява свързване, да предоставят Kerberos поддръжка.

**Забележка:** Няма поддръжка за тип разпознаване GSSPLUGIN.

## Поддръжка на Kerberos

Kerberos слоят за разпознаване, който управлява системата за билети, е интегриран в механизма Active Directory на Windows 2000. Клиент и сървър страните на приложение комуникират с клиент и сървър модулите на Kerberos SSP (Security Support Provider). Интерфейсът Security Support Provider (SSPI) осигурява интерфейс от високо ниво към Kerberos SSP и други защитни протоколи.

### Типична настройка

За да се конфигурира DB2 с Kerberos разпознаване, настройте:

- Полица за оторизация за DB2 (като услуга) в Active Directory, която е поделена в мрежата, и
- Доверителни взаимоотношения между центровете Kerberos Key Distribution Centers (KDC)

В най-простия сценарий, има поне едно KDC доверено отношение за конфигуриране, което е между KDC, управляващ клиентската работна станция, и System i, OS/390 или z/OS системата. OS/390 Версия 2 Издание 10 или z/OS Версия 1 Издание 2 осигурява на Kerberos билетна обработка чрез своето RACF средство, което позволява на хоста да функционира като UNIX KDC.

DB2 Connect осигурява, както обикновено, функционалност на маршрутизатор в 3-слойното обкръжение. DB2 не поема роля в разпознаването, когато се използва Kerberos защита. Вместо това, само предава защитния токен на клиента на DB2 за i5/OS или на DB2 за OS/390 и z/OS. Не е нужно DB2 Connect шлюзът да е член на Kerberos средата на клиента или хоста.

### Обратна съвместимост

Минимални изисквания на DB2 за Kerberos поддръжка:

**Клиент на IBM сървър на данни:**

Версия 8

**DB2 Connect:**

Версия 8



## DB2 Universal Database (UDB) за OS/390 и z/OS:

Версия 7

### Съвети и тънкости за OS/390 и z/OS защитата

Тази тема предоставя някои насоки и съвети за защитата на DB2 Connect при свързване към DB2 за OS/390 и z/OS сървър на база данни.

#### Поле за допълнителна защита

Уверете се, че DB2 OS/390 и z/OS полето за допълнителна защита е настроено на YES. Това поле се появява в DSNTIPR панела на DB2 за OS/390 и z/OS.

#### Кодове за допълнителна защита

До версия 5.1 на DB2 Universal Database за z/OS и OS/390 заявките за свързване, които съдържаха идентификатор или парола на потребителя, можеха да приключат неуспешно с код за причина 0 SQL30082, но без никакви други признаци за това какво може да е неправилно.

Във версия 5.1 на DB2 Universal Database за z/OS и OS/390 беше въведено подобрението, което осигурява поддръжка на кодове за разширена защита. Ако се използва разширена защита, освен кода за причина се осигурява и допълнителна диагностична информация, като (PASSWORD EXPIRED).

За да се използва това, инсталационният параметър за разширена защита DB2 Universal Database за z/OS и OS/390 ZPARM трябва да бъде настроен на стойността YES. Използвайте инсталационния панел на DB2 Universal Database за z/OS и OS/390 DSN6SYSP, за да определите EXTSEC=YES. Освен това можете да използвате и DDF панел 1 (DSNTIPR). Стойността по подразбиране е EXTSEC=NO. В случай на изтекла парола, Windows, Linux, UNIX, и Web приложенията, използващи DB2 Connect, ще получат SQL30082 съобщение за грешка.

#### вече проверена TCP/IP защита

Ако искате да осигурите поддръжка за DB2 защитната опция AUTHENTICATION=CLIENT, използвайте DB2 Universal Database за z/OS и OS/390 инсталационния панел DSNTIP4 (DDF панел 2) за настройка на вече проверената TCP/IP защита на YES.

#### Защита на настолни ODBC и Java приложения

ODBC и Java приложенията на работни станции използват динамичен SQL. Това може да създаде защитни проблеми при някои инсталации. DB2 Universal Database за z/OS и OS/390 въвежда нова опция за свързване DYNAMICRULES(BIND), която позволява изпълнението на динамичен SQL под оторизацията на собственика или на този, който е изпълнил свързването.

DB2 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. След всяко свързване към приложението автоматично ще се генерира SQL оператор SET CURRENT PACKAGESET схема.

Използвайте ODBC мениджъра, за да обновите DB2CLI.INI.



## Поддръжка на промяна на паролата

Ако парола на потребителски идентификатор е изтекла, SQL CONNECT израз връща съобщение за грешка, като например SQLCODE -30082 код на причина 1. С DB2 Connect е възможно паролата да бъде променена отдалечено. Чрез DRDA, DB2 Universal Database за z/OS и OS/390 може да промени паролата от ваше име, като подаде следния CONNECT израз:

```
CONNECT TO <база-данни> USER <потребителски-идентификатор> USING <парола>  
NEW <нова_парола> CONFIRM <нова_парола>
```

Диалогът "Промяна на парола" на DB2 Асистента за конфигуриране също може да се използва за смяна на паролата.

## Типове защита, поддържани в DB2 Connect

Тази тема изброява различните комбинации разпознаване и защитни настройки, които се поддържат с DB2 Connect.

### Типове защита за TCP/IP свързвания

TCP/IP комуникационният протокол не поддържа възможност за защита на ниво мрежов протокол. Типът разпознаване определя къде се извършва разпознаването. Само показаните в тази таблица комбинации се поддържат от DB2 Connect. Настройката за разпознаване е в запис на директорията на базата данни на DB2 Connect сървър.

Таблица 9. Валидни защитни сценарии

Сценарий	Настройка за разпознаване	Проверка за валидност
1	CLIENT	Клиент
2	SERVER	Хост или System i сървър на база данни
3	SERVER_ENCRYPT	Хост или System i сървър на база данни
4	KERBEROS	Kerberos защита
5	DATA_ENCRYPT	Хост или System i сървър на база данни

### Обсъждане на типовете защита

Следната дискусия се отнася за свързванията, описани по-горе и изброени в Таблица 9. Всеки сценарий е описан по-подробно, както следва:

- В сценарий 1, потребителското име и парола се проверяват само на отдалечения клиент. За локален клиент, потребителското име и парола се проверяват само на DB2 Connect сървър.  
Очаква се потребителите да бъдат разпознавани на разположението, в което влизат. 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 сценарий 2, потребителското име и паролата се валидират само на хоста или System i сървър на базата данни. Потребителският идентификатор и паролата се изпращат по мрежата от отдалечения клиент на DB2 Connect сървър, и от DB2 Connect сървър до хост или System i сървър на база данни.
- Сценарий 3 е същият, като сценарий 2, освен че потребителският идентификатор и паролата са криптирани.
- В сценарий 4 клиентът получава Kerberos билет от Kerberos KDC. Билетът се предава непроменен през DB2 Connect на сървър, където се валидира от сървър.

- Сценарий 5 е същият, като сценарий 3, освен че потребителските данни също са криптирани.

---

## Глава 6. Свързване на приложения и помощни програми (DB2 Connect)

Приложните програми, разработени с вграден SQL, трябва да се свържат с всяка база данни, с която ще работят. На платформи, на които тези функции са достъпни, можете да направите това чрез Command Center и Configuration Assistant.

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. тези плановете за достъп се доставят от разработчици на приложения, и се съдържат във *файловете за обвързване*, които се създават по време на предкомпилирането. Обвързването е процесът по обработката на тези файлове за обвързване от хост или System i сървър на базата данни.

Тъй като някои от помощните програми, доставяни с DB2 Connect, са разработени посредством вграден SQL, те трябва да бъдат обвързани към хост или System i сървър на база данни, преди да могат да бъдат използвани с тази система. Ако не използвате DB2 Connect помощни програми и интерфейси, не се налага да ги обвързвате към всеки ваш хост или System i сървър на бази данни. Списъците с файловете за свързване, необходими за тези помощни програми, се съдържат в следните файлове:

- ddcsmvs.lst за OS/390 или z/OS
- 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 сървърен продукт, DB2 Connect помощните програми трябва да бъдат обвързани към всеки хост или System i сървър на бази данни, преди да могат да бъдат използвани с тази система. Ако приемете, че клиентите са на еднакво fix pack ниво, трябва да обвържете помощните програми само веднъж, независимо от броя въввлечени клиентски платформи.

Например, ако имате 10 Windows клиента и 10 AIX клиента, свързващи се към DB2 Universal Database (UDB) за OS/390 и z/OS чрез DB2 Connect Enterprise Server Edition на Windows сървър, направете едно от следните:

- Обвържете ddcsmvs.lst от един от Windows клиентите.
- Обвържете ddcsmvs.lst от един от AIX клиентите.
- Обвържете ddcsmvs.lst от DB2 Connect сървъра.

Този пример приема, че:

- Всички клиенти са на едно сервизно ниво. Ако не са, в добавка може да се наложи да обвържете всеки клиент от дадено сервизно ниво
- Сървърът е на същото сервизно ниво като клиентите. Ако не е, трябва да изпълните обвързването също и от сървъра.

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 System i сървър на база данни, към който изпълнявате обвързване, направете следното:

1. Уверете се, че имате достатъчни права за системата за управление на хост или System i сървъра на бази данни:

#### **OS/390 или z/OS**

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.

**А)** Ако сте изпълнили оригиналното обвързване, и изпълнявате обвързването отново, наличието на всяко от изброените по-горе права ще ви позволи да завършите обвързването.

**В)** Ако вашето оригинално обвързване е направено от някой друг, и вие изпълнявате второто обвързване, ще се нуждаете от права SYSADM или SYSCTRL, за да завършите обвързването. 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 таблица to nullid with grant option
```

## OS/400

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

### 2. Генерирайте команди, подобни на:

```
db2 connect to ПСЕВДОНИМ-НА-БД user ПОТР-ID using ПАРОЛА
db2 bind пътека@ddcsmvs.lst blocking all
      sqlerror continue messages ddcsmvs.msg grant public
db2 connect reset
```

Където *ПСЕВДОНИМ-НА-БД*, *ПОТР-ID*, и *ПАРОЛА* се прилагат за хост или System i сървъра на бази данни, *ddcsmvs.lst* е файлът със списък за обвързвания за z/OS, и *пътека* представлява разположението на файла със списък за обвързвания. Например, *устройство:\sqllib\bnd\* се отнася за всички Windows операционни системи, и *INSTHOME/sqllib/bnd/* се отнася за всички Linux и UNIX операционни системи, където *устройство* представлява устройството, на което е инсталиран DB2 Connect, и *INSTHOME* представлява 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 @файл-за-обвързване.lst
```

Например:

```
ddcspkgn @ddcsmvs.lst
```

може да генерира следния резултат:

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

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
```

### Забележка:

- Използването на опцията за обвързване *sqlerror continue* е задължително; тази опция обаче автоматично се указва от ваше име, когато обвързвате приложения посредством DB2 средствата или Процесора за обработка на команди (CLP). 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. На свой ред, това позволява един файл за обвързване да бъде използван срещу множество сървъри, дори когато дадена реализация на сървър може да обозначи SQL синтаксиса на друг като невалидна. Поради тази причина, обвързването на всеки от списъчните файлове *ddcsxxx.lst* срещу даден хост или System i сървър на база данни може да се очаква да изведе предупреждения.
- Ако се свързвате към DB2 база данни чрез DB2 Connect, използвайте списъка за обвързване *db2ubind.lst*, и не посочвайте *sqlerror continue*, което е валидно само при свързване с хост или System i сървър на база данни. Също

така за свързване към DB2 база данни, се препоръчва 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 от предно издание на DB2, може да се наложи да обвържете помощните средства на тези клиенти към DB2 Connect.

---

## Глава 7. Многосайтови обновявания

Функцията Многосайтово обновяване, позната също и като Разпределена единица работа (distributed unit of work -DUOW) и Двухазов протокол за записване на промените, е функция, която позволява на вашите приложения да обновяват данни в множество отдалечени сървъри на бази данни като се гарантира целостта на данните. Характерен пример за многосайтово обновяване е банкова транзакция, която включва трансфер на пари от една сметка в друга, намираща се на 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 изчерпателна поддръжка за многосайтови обновявания. Тази поддръжка е достъпна за приложения, използващи обикновен SQL, както и за приложения, които използват монитори за обработка на транзакции (TP монитори), които реализират X/Open XA спецификацията за интерфейс. Примерите за такива TP мониторни продукти включват IBM TxSeries (CICS и Encina), IBM Message and Queuing Series, IBM Component Broker Series, IBM San Francisco Project, както и Microsoft Transaction Server (MTS), BEA Tuxedo, и няколко други. Изискванията за настройка са различни в зависимост от това дали се използва многосайтово обновяване на базата на собствен SQL или на TP Монитор.

И двете програми за многосайтово обновяване - на базата на собствен SQL и на TP Монитор трябва да се прекомпилират 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, че ще координира транзакцията (както се указва чрез DB2, получаваща xa\_orep извиквания от TP монитора за установяване на свързване към базата данни), тогава DB2 софтуерът ще се използва за координиране на транзакцията.

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

Многосайтовото обновяване с TP монитор може да координира транзакция, която достъпва едновременно DB2 и не-DB2 мениджъри на ресурси, като Oracle, Informix или SQLServer. Собственото SQL многосайтово обновяване се използва само с DB2 сървъри.

За да работи транзакция с многосайтово обновяване, всяка от участващите в разпределената транзакция база данни трябва да бъде способна да поддържа разпределена единица работа (DUOW). Понастоящем, следните DB2 сървъри осигуряват DUOW поддръжка, която им позволява да участват в разпределени транзакции:

- DB2 за Linux, UNIX и Windows Версия 8, или по-късна
- DB2 Universal Database (UDB) за OS/390 и z/OS Версия 7

- DB2 за z/OS Версия 8
- DB2 за i5/OS изисква OS/400 Версия 5 Издание 1 или по-късна

Разпределената транзакция може да актуализира всяка комбинация от поддържани сървъри на бази данни. Например, приложението ви може да обнови няколко таблици в DB2 база данни на Windows, DB2 за OS/390 и z/OS база данни, и DB2 за i5/OS база данни, всичко това в рамките на единична транзакция.

---

## Активиране на многосайтови обновявания посредством Центъра за управление

Можете да използвате Центъра за управление, за да осъществите многосайтово обновяване.

За да разрешите многосайтови обновявания:

1. Стартирайте Центъра за управление.
2. Натиснете знака [+], за да разгънете изображението на дървото.
3. С десния бутон на мишката изберете потребителския модел, който искате да конфигурирате. Отваря се контекстно меню.
4. Изберете **Многосайтово обновяване** → **Конфигуриране**. Отваря се Помощникът за многосайтово обновяване.
5. Изберете **Използване на посочения по-долу TP монитор** и задайте Монитор на Процесор на транзакции (TP монитор). Това поле ще ви покаже подразбиращите се стойности за Монитора на транзакции, който сте активирали. Ако не искате да използвате Монитор на транзакции, изберете **Не използвай TP Монитор**. Щракнете върху **Следващ**.
6. Ако използвате TP монитор, укажете настройките на мениджъра на точка на синхронизация. Ако не използвате TP монитор, укажете базата данни на вашия мениджър на транзакции.
7. Щракнете **Край**.

---

## Тестване на многосайтови обновявания посредством Центъра за управление

Можете да тествате настройката си за многосайтово обновяване посредством Центъра за управление.

За да тествате многосайтовото обновяване:

1. Изберете потребителския модел с десния бутон на мишката и изберете опцията на меню **Многосайтово обновяване** → **Тест** от контекстното меню. Отваря се прозорецът Тестване на многосайтово обновяване.
2. Изберете базите данни, които искате да тествате, от достъпните бази данни в списъка **Достъпни**. Можете да използвате бутоните със стрелки (> и >>) в средата за местене на избраните към и от списъка **Избрани**. Можете също да променят избраните потребителски идентификатор и парола, като ги редактирате пряко в списъка **Избрани**.
3. Когато завършите избора си, щракнете **ОК**. Отваря се прозорецът Резултати от тестване на многосайтово обновяване.
4. Прозорецът Резултат от тест на многосайтово обновяване показва кои от избраните бази данни са преминали, и кои не са преминали теста за обновяване. В



прозореца са показани и SQL кодовете и съобщенията за грешки за базите данни, които не са преминали успешно теста. Щракнете **Затвори**, за да затворите прозореца.

- Щракнете **Затвори**, за да затворите прозореца Тест на многосайтово обновяване.

## Многосайтово обновяване и мениджър на точка на синхронизация

Хост и System i сървърите на бази данни изискват DB2 Connect, за да участват в разпределени транзакции, произлизащи от Linux, Windows, UNIX, и web приложения. Освен това, множество от сценариите на многосайтово обновяване, включващи хост и System i сървъри на база данни, изискват да бъде конфигуриран компонентът мениджър на точка на синхронизация (SPM). Когато се създава DB2 потребителски модел, DB2 SPM автоматично се конфигурира със стойностите по подразбиране.

Нуждата от SPM се налага от избора на протокол (TCP/IP) и използването на TP монитор. В следващата таблица е дадено резюме на сценариите, които изискват използването на SPM. Таблицата също показва дали DB2 Connect се изисква за всякакъв достъп до хост или System i от Intel или UNIX машини. За многосайтови обновявания е необходим SPM компонентът на DB2 Connect, ако използвате TP монитор.

Таблица 10. Сценарии на многосайтови обновявания, които изискват SPM – TCP/IP

Използва ли се монитор на процесора на транзакции?	Необходим ли е мениджър на точка на синхронизация?	Необходим продукт (изберете един)	Поддържани хост и System i бази данни
Да	Да	DB2 Connect сървърен продукт DB2 Enterprise Server Edition с приложен DB2 Connect лиценз	DB2 Universal Database (UDB) за OS/390 и z/OS B7 DB2 UDB за z/OS B8 или по-късна
Не	Не	DB2 Connect Personal Edition DB2 Connect сървърен продукт DB2 Enterprise Server Edition с приложен DB2 Connect лиценз	DB2 UDB за OS/390 и z/OS B7 DB2 UDB за z/OS B8 или по-късна

**Забележка:** Разпределената транзакция може да актуализира всяка комбинация от поддържани сървъри на бази данни. Например, приложението ви може да обновява няколко таблици в DB2 база данни на Windows, база данни на DB2 за OS/390 и база данни на DB2 за i5/OS в рамките на единична транзакция.

## Конфигуриране на DB2 Connect с XA съвместим мениджър на транзакции

Тази тема описва конфигурационните стъпки, необходими за използването на S/390, System i, и zSeries сървъри на бази данни във вашия TP монитор.

Трябва да имате работещ TP монитор, да сте инсталирали DB2 Connect, и също така да сте конфигурирали и тествали свързване към хост или System i сървър на база данни.

За да конфигурирате DB2 Connect да използва S/390, System i, и zSeries сървъри на бази данни във вашия TP монитор, изпълнете следните стъпки:

1. Конфигурирайте TP монитора така, че да може да достъпи DB2 XA Switch. DB2 XA Switch осигурява на TP монитора адресите на DB2 Connect XA API. Всеки TP монитор изпълнява това по различен начин.
2. Конфигурирайте TP монитора с низа XA\_OPEN на DB2. Всеки TP монитор има свой собствен начин да направи това. 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 се изисква, модифицирайте подразбиращите се конфигурационни параметри на DB2 Connect мениджъра на точка на синхронизиране (SPM). Хост и System i (Версия 5 Издание 3 и по-ранни) сървърите на бази данни засега не поддържат XA интерфейса. System i Версия 5 Издание 4 и следващите имат пълна XA поддръжка. SPM е компонент на DB2 Connect, който преобразува XA двуфазовия протокол за комит в двуфазовия протокол за комит, използван от хост и System i сървърите на бази данни. По подразбиране DB2 потребителският модел има предварително дефинирани стойности за SPM конфигурационните параметри. 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 и z/OS, не трябва да промените никоя от настройките по подразбиране. В този случай не е необходима SPM конфигурация, тъй като вече работи.

---

## DB2 Connect поддръжка за свободно съчетани транзакции

Поддръжката в DB2 Connect за свободно съчетани транзакции е предвидена за потребители, които реализират XA разпределени приложения, достъпващи DB2 Universal Database (UDB) за i5/OS Версия 5 Издание 4 или по-късна; и DB2 UDB за OS/390 и z/OS Версия 7 или по-късна. Тази поддръжка позволява на различни клонове на една глобална транзакция да поделят пространство за заключвания на DB2 UDB за OS/390 и z/OS.

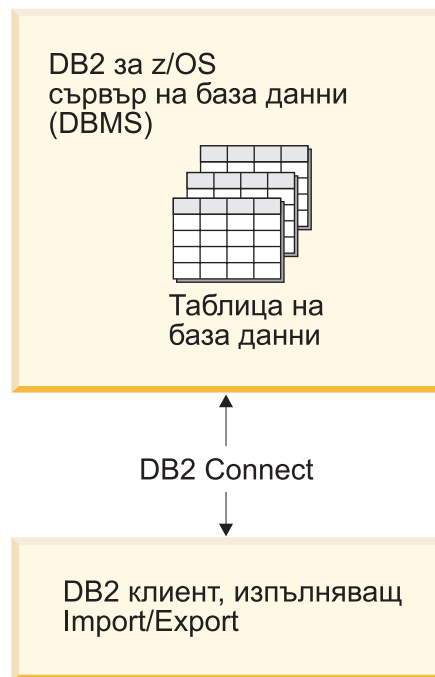
Поддръжката за свободно съчетани транзакции е предвидена за .NET и COM+ приложения.

Тази характеристика намалява интервала между срещането на таймаут при заключване от едно разклонение на разпределена транзакция или блокиране поради заключване, като резултат от друго разклонение в същата глобална транзакция.

---

## Глава 8. Пренасяне на данни с DB2 Connect

Ако работите в сложна среда, в която трябва да премествате данни между хост система на база данни и работна станция, можете да използвате DB2 Connect, шлюза за пренос на данни между хоста и работната станция (вижте Фигура 8).



Фигура 8. Импортиране/експортиране чрез DB2 Connect

Помощните програми за експорт и импорт на DB2 ви позволяват да премествате данни от база данни на хост или System i сървър във файл на DB2 Connect работна станция, и обратното. Можете после да използвате данните с всяко друго приложение или релационна система за управление на бази данни, която поддържа този експортен или импортен формат. Например, можете да експортирате данни от база данни на хост или System i сървър в PC/IXF файл, и после да го импортирате в база данни на DB2 за Windows.

Можете да изпълнявате операции по експортиране и импортиране от клиент на базата данни или от DB2 Connect работната станция.

### Забележка:

1. Данните, които ще се експортират или импортират, трябва да отговарят на ограниченията за размер и тип данни, които са приложими и за двете бази данни.
2. За да подобрите производителността на импортиране, можете да използвате съставни заявки. Укажете модификатора на файлов тип compound в помощната програма за импортиране, за да групирате посочен брой изрази на заявки в блок. Това може да намали мрежовото натоварване, и да подобри времето за реакция.

С DB2 Connect операциите по импорт и експорт трябва да отговарят на следните условия:

- Файловият тип трябва да бъде PC/IXF.
- Преди да можете да импортирате, на сървъра приемник трябва да бъде създадена таблица приемник с атрибути, които да са съвместими с данните. Помощната програма db2look може да се използва за получаване на атрибутите на таблицата източник. Импортирането чрез DB2 Connect не може да създаде таблица, защото INSERT е единствената поддържана опция.

Ако някое от тези условия не е посрещнато, операцията е неуспешна, и се връща съобщение за грешка.

**Забележка:** Дефинициите на индекс не се съхраняват при експорт и не се използват при импорт.

Ако експортирате или импортирате смесени данни (колони, съдържащи едновременно еднобайтови и двубайтови данни), вземете предвид следното:

- На системи, които съхраняват данни в EBCDIC (MVS, OS/390, OS/400, VM, и VSE), символите shift-out и shift-in маркират началото и края на двубайтовите данни. Когато дефинирате дължини на колони за таблиците на вашата база данни, не пропускайте да оставите достатъчно място за тези символи.
- Препоръчват се колони с променлива символна дължина, освен ако данните в колоната имат последователен шаблон.

### Преместване на данни от работна станция на хост сървър

За да преместите данни в база данни на хост или System i сървър:

1. Експортирайте данните от DB2 таблица в PC/IXF файл
2. Като използвате опцията INSERT, импортирайте PC/IXF файла в съвместима таблица в базата данни на хост сървъра.

За да преместите данни от база данни на хост сървър на работна станция:

1. Експортирайте данните от таблицата на базата данни на хост сървър в PC/IXF файл.
2. Импортирайте PC/IXF файла в DB2 таблица.

### Пример

Следният пример илюстрира как да се пренасят данни от работна станция на база данни на хост или System i сървър.

Експортирайте данните във външен IXF формат, като подадете следната команда:

```
db2 export to staff.ixf of ixf select * from userid.staff
```

Подайте следната команда, за да установите DRDA свързване към целевата DB2 база данни:

```
db2 connect to cbc664 user admin using xxx
```

Ако не е вече създадена, създайте таблицата приемник на целевия потребителски модел на DB2 базата данни:

```
CREATE TABLE mydb.staff (ID SMALLINT NOT NULL, NAME VARCHAR(9),
DEPT SMALLINT, JOB CHAR(5), YEARS SMALLINT, SALARY DECIMAL(7,2),
COMM DECIMAL(7,2))
```

За да импортирате данните, подайте следната команда:

```
db2 import from staff.ixf of ixf insert into mydb.staff
```

Всеки ред данни ще бъде прочетен от файла в IXF формат, и ще бъде подаден SQL израз INSERT за вмъкване на реда в таблицата mydb.staff. Единичните редове ще бъдат вмъквани, докато всички данни бъдат преместени в таблицата приемник.

Подробна информация е достъпна в "Moving Data Across the DB2 Family," публикация от IBM Redbooks. Тази Redbooks публикация може да бъде намерена на следния URL: <http://www.redbooks.ibm.com/redbooks/SG246905>.



---

## Глава 9. SQLCODE преобразуване

Различните IBM продукти за релационни бази данни не винаги генерират еднакви SQLCODE за подобни грешки. Дори ако SQLCODE е еднакъв, може да бъде придружен от токени, които се указват различно. Списъкът с токени се предава в полето SQLERRMC на SQLCA. По подразбиране, DB2 Connect преобразува SQLCODE и токени от всеки хост или System i сървър на база данни в подходящите DB2 SQLCODE.

Ако искате да изключите SQLCODE преобразуването, укажете **NOMAP** в низовия параметър на DCS директорията.

Ако пренесете директно приложение от хост или System i сървър на база данни, като DB2 Universal за OS/390 и z/OS, може да пожелаете да изключите SQLCODE преобразуването. Така ще можете да използвате приложението, без да промените използвания код SQLCODE.

---

### Изключване на SQLCODE преобразуването

Ако искате да изключите SQLCODE преобразуването, укажете **NOMAP** в низовия параметър на DCS директорията.

Ако пренесете директно приложение от хост или System i сървър на база данни, като DB2 Universal Database (UDB) за OS/390 и z/OS, може да пожелаете да изключите SQLCODE преобразуването. Така ще можете да използвате приложението, без да промените използвания код SQLCODE.

---

### Персонализиране на SQLCODE преобразуването

По подразбиране, DB2 Connect преобразува SQLCODE и токени от всеки хост или System i сървър на база данни в подходящите DB2 SQLCODE. Следните файлове са копия на SQLCODE преобразуването по подразбиране:

- dcs1dsn.map преобразува SQLCODE на DB2 Universal Database (UDB) за OS/390 и z/OS и DB2 за z/OS.
- dcs1ari.map преобразува SQLCODE на DB2 UDB за VSE и VM.
- dcs1qsq.map преобразува SQLCODE на DB2 UDB за iSeries и DB2 за i5/OS.

Не се изисква преобразуване за DB2 на Linux или UNIX операционни системи.

1. Ако искате да подмените подразбиращото се SQLCODE преобразуване, или ако използвате хост или System i сървър на база данни, който няма SQLCODE преобразуване (не-IBM сървър на база данни), можете да копирате един от тези файлове, и да го използвате като основа за вашия нов файл за SQLCODE преобразуване. Като копирате файла, вместо директно да го редактирате, подsigурявате възможността винаги да направите справка с оригиналното SQLCODE преобразуване, ако е необходимо.
2. Посочете името на файл на вашия нов файл за SQLCODE преобразуване в параметричния низ на DCS директорията.
3. 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 `map` 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:

входящ\_код [, изходящ\_код [, списък\_токени]]

*входящ\_код* представя едно от следните:

*sqlcode*

SQLCODE от хост или System i 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.

**csnn** SQLSTATE код на клас от хост или System i сървър на база данни. *nn* е едно 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 **сс** 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* представя изходния 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 има формата на число (*n*), по желание предшествано от **с**, по желание следвано от **с** или **i**. Интерпретира се, както следва:

- с** Типът на данни на токена в тази позиция е CHAR (по подразбиране). Ако **с** е преди *n*, се отнася за входящия токен; ако идва след *n*, се отнася за изходящия токен. Символът **с** трябва да е с малки букви.
- i** Типът на данни на токена в тази позиция е INTEGER. Ако **i** е след *n*, се отнася за изходния токен. **i** не трябва да се появява преди *n*, защото IBM хост или System i сървър продуктите за бази данни поддържат само CHAR токени. Символът **i** трябва да е малка буква.
- n* Число или числа, указващи кои токени на хост или System i сървър на база данни се използват. Те се подреждат в желанния ред за мястото в изходящия SQLCA. Числото указва токена на хост или System i сървър на база данни; подредбата указва реда, в който токените ще бъдат поставени в SQLCA.

Например, хост или System i сървърът на база данни може да върне два токена, 1 и 2. Ако искате токен 2 да се появява преди токен 1 в изходния SQLCA, укажете (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 9 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 9. Файл за преобразуване на SQLCODE

Следните описания отговарят на съответния номер на ред в предната фигура:

1. SQLCODE се преобразува от -007 до -007. Първият входен токен, получен от хост или System i сървър на база данни, се използва като първи изходен 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. Първият входен токен, получен от хост или System i сървър на база данни, се отхвърля. 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 токени, получени от хост или System i сървър на база 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 входен токен, получен от хост или System i сървър на база 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 входни токени, получени от хост или System i сървър на база данни, са CHAR, и се използват като първи и втори токени в изходния 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 10. Наблюдение на система база данни и DB2 Connect

Разглеждат се няколко начина за наблюдение на свързванията и производителността в обкръжение, използващо DB2 Connect. Типът наблюдение, който се изпълнява, е специфичен за операционната система.

---

### Наблюдение на свързванията за отдалечени клиенти

Можете да използвате системен монитор за базата данни с DB2 Connect сървърен продукт, като DB2 Connect Enterprise Edition, за наблюдаване на отдалечени клиентски свързвания. За да наблюдавате клиенти, които са локални за DB2 Connect сървъра, изпълнявайки се на самия сървър, трябва да настроите следната променлива:

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

Например, когато възниква грешка на хост или System i система, системният администратор може да определи дали проблемът е бил на DB2 Connect работната станция. Системният монитор на базата данни е свързан с:

- DRDA токена за корелация (CRRTKN), за незащитен обмен.
- Идентификатор на единица работа (UOWID), за двуфазови свързвания, защитени от DRDA-3 супсройнт мениджър (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.

### Представяне на състоянието на параметрите за наблюдение

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

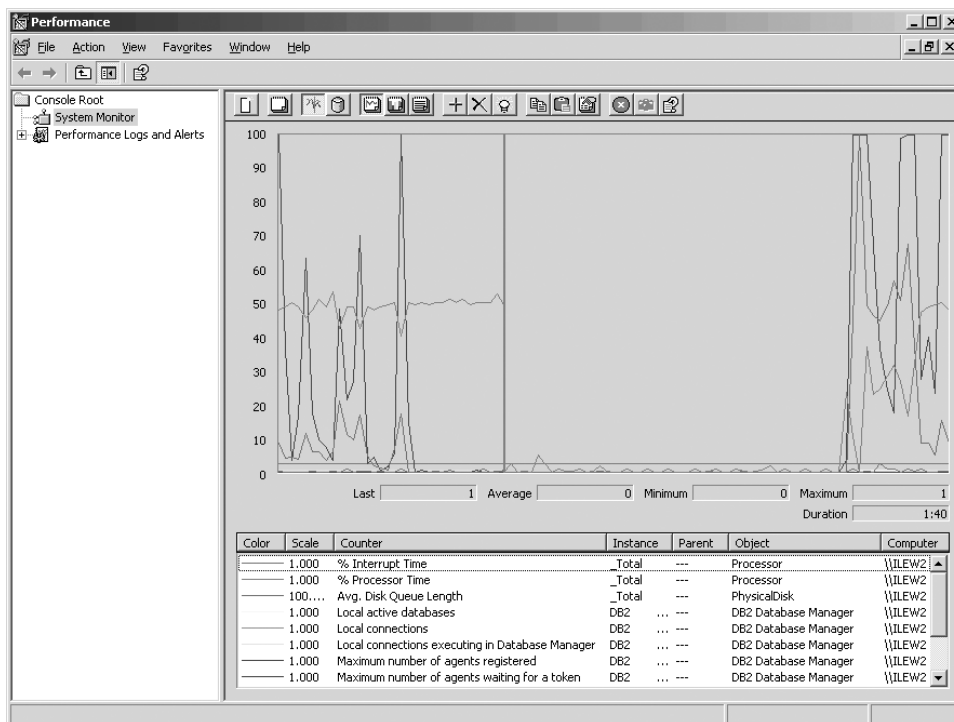
---

### Наблюдение на производителността чрез Windows Performance Monitor

Windows операционните системи осигуряват полезно средство за наблюдение на производителността на вашите DB2 приложения. Performance Monitor, който е едно от административните средства на Windows, извежда графично представяне на системната производителност. 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 PMS, откъдето по-късно ще можете да ги вземете.

Например, във фигурата по-долу, няколко DB2 показателя са показани графично срещу CPU употребата. Наборът от представени стойности са записани във файла db2chart.rmc. Можете да запазите колкото желаете PMS файлове, като всеки отразява различен профил на системната производителност.



Фигура 10. Монитор на производителността

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

## Използване на командите GET SNAPSHOT

DB2 мониторът поддържа текуща сметка за ценната 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 база 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 псевдоним\_на\_БД
- GET SNAPSHOT FOR DCS APPLICATIONS ON псевдоним\_на\_БД

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 DCSDDB 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 = 12-15-2001 10:28:24.596495
Последно изменило 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
Време на промяна в състоянието   = 12-15-2001 10:29:06.707086
Възел на клиент                   = sys143
Версия на клиента                 = SQL06010
Платформа на клиента              = AIX
Протокол на клиент                = 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 базата 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
Времеви индикатор на първо свързване = 12-15-2001 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     =
Времеви индикатор на UOW завършване = 12-15-2001 10:28:25.592631
Времеви индикатор на UOW започване  = 12-15-2001 10:29:06.142790
Час на спиране на UOW              =
Изминало време от последната uow (sec.ms) = 0.034396

Последна изпълнена операция        = Execute Immediate
Времеви индикатор на начало на посл. опер. = 12-15-2001 10:29:06.142790
Времеви индикатор на край на посл. опер.  = 12-15-2001 10:29:06.707053

Оператор                            = Execute Immediate
Номер на раздел                      = 203
Създател на приложение              = NULLID
Име на пакет                         = SQLC2C07
Очаквана ст-ст в таймерони SQL компилатор = 0
Очаквана кардиналност на SQL компилатор = 0
Времеви индикатор на начало на израз  = 12-15-2001 10:29:06.142790
Времеви индикатор на край на израз    = 12-15-2001 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)

```

## Статус на DCS приложение

System Monitor осигурява три форми на командата LIST DCS APPLICATIONS, както следва:

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

В изхода, който следва, форматът за Идентификатор на хост приложение и Идентификатор на клиентско приложение може да се различава в зависимост от версията на хост или System i базата данни и нивото на поддръжка на TCP/IP.

Таблица 11. Формат на идентификатор на приложение на база версия на хост и ниво на TCP/IP поддръжка

Сценарий	Формат на идентификатор на приложение
Клиенти, достъпващи сървъри на данни с поддръжка на ниво на RDB Manager, по-малко от 7	G91A0D3A.P8BC.060306212019



Таблица 11. Формат на идентификатор на приложение на база версия на хост и ниво на TCP/IP поддръжка (продължение)

Сценарий	Формат на идентификатор на приложение
Клиенти, достъпващи сървъри на данни с поддръжка на ниво по-голямо от 8 на RDB Manager по TCP/IP v4	9.26.13.61.65289.060306213816
Клиенти, достъпващи сървъри на данни с поддръжка на ниво по-голямо от 8 на RDB Manager по TCP/IP v6	2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741

## LIST DCS APPLICATIONS

За да прегледате информацията, осигурена от Монитора на ниво приложение, изпълнете командата DB2 LIST DCS APPLICATIONS.

Връща следната информация за TCP/IP свързване (DB2 Connect и DB2 Universal Database за z/OS и OS/390):

Id от.	Име приложение	Указ.	ID на хост приложение прил.
-----	-----	-----	-----
NEWTON	db2cli.exe	7	G91A0D3A.P8BC.060306212019
NEWTON	db2cli.exe	25	9.26.13.61.65289.060306213816
NEWTON	db2cli.exe	20	2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741

### Id за оторизация

Идентификаторът за разпознаване, използван за влизане в хост или System i сървъра на база 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.

### Указател на прил.

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. Идентификаторът на агента е необходим и когато използвате командата FORCE USERS или API.

### Идентификатор на хост приложение

Един от следните:

- DRDA токена за корелация (CRRTKN), за незащитен обмен.
- 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 свързва с хост или System i сървъра на база 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. за оторизация	Име на приложение		Прил. прил.	Идентификатор на клиентско приложение	
NEWTON	db2cli.exe		37	2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224	

Пор#	Клиент Псевд.БД	Клиент Възел	Клиент Издание	Клиент Код. стр.	Идент. хост приложение
00001	MDB	SAYYID	SQL09000	1252	G91A0D3A.P982.060306214231

Пор#	Хост	Име на БД	Хост Издание
00001	MEXICO		DSN08015

### 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.

Тази стойност ви позволява да корелирате свързвания от клиенти към DB2 Connect работната станция и от DB2 Connect работната станция към хост или System i сървър на база данни.

### Пореден номер на клиент (Пор#)

Поредният номер на клиент е поредният номер на транзакцията. Използва се за съпоставяне на транзакция, разпространена в рамките на различни системи.

### Псевдоним на клиентска 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 база данни сървъра.

### NNAME на клиент (възел)

Идентифицира възела, на който се изпълнява клиентското приложение. Информацията зависи от използвания от клиента протокол. 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. Идентификаторите на клиентски продукти ще бъдат:

- SQL07010 за версия 7.1 на DB2 Universal Database и DB2 Connect и техните клиенти.
- SQL08010 за версия 8.1 на DB2 Universal Database и DB2 Connect и техните клиенти.
- SQL08020 за версия 8.2 на DB2 Universal Database и DB2 Connect продукти и техните клиенти.
- SQL09120 за Версия 9.1 на DB2 продуктите, DB2 Connect продуктите, 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 убедите, че се поддържа преобразуване на данните между кодовата страница на приложението и кодовата страница на базата данни (или за базата данни на хост или System i сървър на база данни, CCSID за хост или System i сървър на база данни).

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

### Изходящ пореден номер

Представява изходящия пореден номер. 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 сървърния продукт на хост или System i базата данни (например, DSN за DB2 Universal Database за z/OS и OS/390, ARI за DB2 сървър за VSE & VM, или QSQ за DB2 за i5/OS)
- VV** Представява двуцифрен номер на версия, като 08.
- RR** Представява двуцифрен номер на издание, като 01.
- M** Представява едносимволно ниво на модификация (0-9 или A-Z).

## 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

Идент. на клиентско приложение	= 2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224
Пореден номер	= 00001
Идентификатор за оторизация	= NEWTON
Идентификатор за доверено св.	=
Име на приложение	= db2cli.exe
Указател на приложение	= 37
Състояние на приложение	= изчаква заявка
Час за промяна на състоянието	= Не е отчетен
Възел на клиент	= SAYYID
Ниво на издание на клиент	= SQL09000
Платформа на клиент	= NT
Протокол на клиент	= TCP/IP
Кодова страница на клиент	= 1252
ID процес клиентско приложение	= 1192
ID за включване на клиента	= ISAYYID
ID на хост приложение	= G91A0D3A.P982.060306214231
Пореден номер	= 00001
Псевдоним на база данни на шлюза	= MDB
Име на DCS база данни	= MDB
Име на хост база данни	= MEXICO
Ниво на издание на хост	= DSN08015
Хост CCSID	= 1208

Полето за състоянието на приложението съдържа една от следните стойности:

1. изчакване на свързване - външно. Това означава, че заявката за свързване към хост или System i база данни е била подадена и DB2 Connect изчаква свързването да бъде осъществено.
2. чакане на заявка. Това означава, че свързването с хост или System i базата данни е осъществено и DB2 Connect изчаква SQL израз от клиентското приложение
3. изчакване на отговор. Това означава, че SQL изразът е изпратен на хост или System i базата данни.

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.

---

## Здравен монитор и тревоги

Здравният монитор на DB2 за z/OS периодично оценява политиките за поддръжка на обект. Ако здравният монитор определи, че има нужда от поддръжка на даден обект, се създават здравни тревоги. Действията в отговор на здравни тревоги мога да се прегледат, подават за изпълнение и записват.

### Общ преглед на здравния монитор на DB2 за z/OS

На z/OS системи, здравният монитор на DB2 за z/OS се стартира като дейност за всяка DB2 подсистема, която ще се наблюдава или на заделен член на групата за поделяне на данни.

Здравният монитор на DB2 за z/OS задейства оценките на политики за поддръжка на обекти в насрочени часове и интервали, както е дефинирано в политиката.

Политиките за поддръжка на обекти се създават посредством помощника Създаване на политика за поддръжка на обект на DB2 Центъра за управление. По време на всяка оценка на политика, критериите за препоръчване на поддръжка се проверяват спрямо праговете, заложи в политиката за поддръжка на обект, за да се определи необходимостта от поддръжка, тоест, дали се изискват COPY, REORG, RUNSTATS, STOSPACE, ALTER TABLESPACE, или ALTER INDEX, и за идентифициране на ограничени състояния, като CHKР, на пространство за таблици, обекти на индекс и на групи за съхранение, където е приложимо. Когато обектите са идентифицирани да са в състояние на предупреждение при оценка, контактите от политиката за здравно оповестяване се уведомяват на техните e-mail адреси или номера на пейджъри. Списъкът на контакти за здравни предупреждения за всяка DB2 подсистема се дефинира в и управлява от Центъра за управление.

Моментното изображение на плана за оценка на политиките, което се използва от здравния монитор за определяне кога да се задействат оценките на политики, отначало се сваля от здравния монитор при стартирането му. Това моментно изображение на план се опреснява в часа за опресняване, указан при стартирането на здравния монитор, или когато здравният монитор получи команда за опресняване. Всяка промяна в плана за оценка на политики се прихваща от здравния монитор, когато настъпи опресняване на плана.

Здравният монитор се стартира и спира от конзолата посредством MVS съответните системни команди START и STOP.

Примерна каталогизирана процедура (DSNHMONP), която стартира DB2 здравния монитор, и примерна каталогизирана процедура (DSNHMONA), която стартира множество DB2 здравни монитори в MVS система или Parallel Sysplex, са поставени в процедурната библиотека от инсталационното задание DSNTIJHM.

Производни таблици, таблици, набори данни, каталогизирани процедури, запомнени процедури, потребителски дефинирани функции и таблица за резултатен набор, които се използват от db2 здравния монитор или свързаните дейности, изброени по-долу, се създават и инсталират от инсталационните задания DSNTIJCC и DSNTIJHM. DSNTIJCC и DSNTIJHM се доставят с FMID JDB771D и JDB881D.

#### **Журнал от оценка на политика**

Оценките на политики, задействани от DB2 здравния монитор, се журналират в таблицата DSNACC.HM\_EVAL\_LOG. Записи се журналират, когато започва и когато завършва оценка на политика. Журналните записи се пазят 7 дни, след което се изтриват от таблицата. DB2 производната таблица DSNACC.HM\_ALERT\_PO\_EV, която е създадена на тази таблица от инсталационното задание DSNTIJCC, може да бъде използвана за показване на всички политики, за които последното изпълнение на оценка не е било успешно.

## **Стартиране, спиране и опресняване на здравен монитор на DB2 за z/OS**

На z/OS система, здравният монитор на DB2 за z/OS се стартира като дейност за всяка DB2 подсистема, която ще се наблюдава или на заделен член на групата за поделене на данни.

- За да стартирате здравния монитор на DB2, подайте следната START MVS системна команда:

```
S име-на-член,DB2SSN=ssid,JOBNAME=HMONssid,TRACE=проследяване,REFRESH=nn
```

Параметрите TRACE и REFRESH са по желание.

*име-на-член*

Указва член на библиотеката процедури, който се изпълнява за стартиране на DB2 здравния монитор, тоест, DSNHMONP. Тази каталогизирана процедура се създава от DSNTIJHM инсталационното задание.

*ssid*

Указва името или идентификатора на DB2 подсистемата, която да се наблюдава.

*проследяване*

Указва флага за проследяване. Възможните стойности са:

- ON - Включва проследяването. Записите от проследяване се записват в SYSOUT
- OFF - Без включване на проследяване

По подразбиране е OFF.

*np*

Указва часа (като се използва 24-часов формат за час), когато здравният монитор опреснява моментното изображение на разписанието за оценка, което използва за задействане на оценките на политика. По подразбиране е 22.

- За да стартирате множество DB2 здравни монитори, подайте следната START MVS системна команда:

*S име-на-член*

*име-на-член*

Член на библиотеката процедури, който се изпълнява за стартиране на множество DB2 здравни монитори, тоест, DSNHMONA.

**Забележка:** Преди да стартирате множество DB2 здравни монитори с една START команда посредством DSNHMONA, HMONPARM наборът данни, посочен в процедурата DSNHMONA, трябва да бъде попълнен със списък на подсистемите, които да бъдат наблюдавани. Каталогизираната процедура и наборът данни се създават от инсталационното задание DSNTIJHM.

- За да опресните моментното изображение на оценка на политика, използвано от DB2 здравния монитор за определяне кога да бъдат задействани оценките на политика, подайте следната MODIFY MVS системна команда:

*F HMONssid,APPL=REFRESH*

*ssid*

Името или идентификатора на DB2 подсистемата, която се наблюдава от DB2 здравния монитор, който опреснявате.

- За да спрете здравния монитор на DB2, подайте следната STOP MVS системна команда:

*STOP HMONssid или P HMONssid*

*ssid*

Името или идентификатора на DB2 подсистемата, която се наблюдава от DB2 здравния монитор, който спирате.

## Преглеждане, подаване и запазване на препоръчани действия

За да прегледате, подавате и запазвате действията, препоръчани за обекти на тревога, идентифицирани при оценката на политика, извикайте DB2 запомнената процедура SYSPROC.DSNACCHR, която се създава от инсталационното задание DSNTIJC. DSNACCHR е запомнена процедура, която определя препоръчаните действия за обекти на тревога, идентифицирани при оценката на политика, и която генерира JCL задание, което ще изпълни препоръчаните действия.

Следната синтактична диаграма показва SQL CALL израза за извикване на DSNACCHR. Тъй като конвенцията за свързване за DSNACCHR е GENERAL WITH NULLS, ако подавате параметри в хост променливи, трябва да включите null индикатор с всяка хост променлива. Null индикаторите за входни хост променливи трябва да бъдат инициализирани, преди да изпълните израза CALL.

### Синтаксис

```

▶—CALL DSNACCHR (—query-type,—health-ind,—policy-id,—work-set,—
dataset-name,—member-name,—save-opt,—trace-flag,—
NULL NULL NULL
—job-id,—jobname,—jcl-proc-time,—trace-flag,—last-statement,—
▶—return-code,—error-msg )—▶

```

#### *query-type*

Указва какво искате да направите с действията, препоръчани за обектите, идентифицирани като пребиваващи в състояние на тревога по време на оценката на политика. Възможните стойности са:

- 0 - Преглеждане на препоръчани действия на обекти на тревога като JCL задание
- 1 - Подаване на JCL задание, което изпълнява препоръчаните действия на обектите на тревога
- 2 - Подаване на JCL задание, което изпълнява препоръчаните действия на обектите на тревога, и поставя заданието на опашката за задържане
- 3 Записване на препоръчани действия на обекти на тревога в JCL задание в член на библиотека

*query-type* е входен параметър от тип INTEGER.

#### *health-ind*

Указва типа тревога, който се включва от DSNACCHR в JCL заданието. Възможните стойности са:

- RS - Ограничено състояние
- EX - Надхвърлени екстенти
- RR - Изисква се REORG
- CR - Изисква се COPY
- RT - Изисква се RUNSTATS
- SS - Изисква се STOSPACE

*health-ind* е входен параметър от тип VARCHAR(4).

#### *policy-id*

Указва политика за поддръжка на обект. *policy-id* е входен параметър от тип VARCHAR(7).

#### *work-set*

Указва работния набор на политиката за поддръжка на обект, която е идентифицирала обектите на тревога, включени от DSNACCHR в JCL заданието. Този работен набор трябва да бъде идентифициран с политиката и типа на тревогата, указани в параметрите *policy-id* и *health-ind. work-set* е входен параметър от тип INTEGER.

#### *dataset-name*

Указва име на напълно квалифициран набор разделени данни (PDS) или разширен набор разделени данни (PDSE). Тази стойност трябва да бъде указана, ако *query-type* е 3. *dataset-name* е входен параметър от тип VARCHAR(44).

#### *member-name*

Указва член на набора разделени данни (PDS) или разширения набор разделени данни (PDSE), указан в параметъра *dataset-name*, където ще бъде запазено JCL заданието за поддръжка на обект. Тази стойност трябва да бъде указана, ако *query-type* е 3. *member-name* е входен параметър от тип VARCHAR(8).

#### *save-opt*

Указва как да бъде запазено JCL заданието за поддръжка на обект. Тази стойност трябва да бъде указана, ако *query-type* е 3. Възможните стойности са:

- R - Подмяна
- A - Добавяне
- NM - Нов член

*save-opt* е входен параметър от тип VARCHAR(2).

#### *trace-flag*

Указва дали проследяването да бъде включено или изключено. Възможните стойности са:

- Y - Включване на проследяването
- N - Изключване на проследяването

*trace-flag* е входен параметър от тип CHAR(1).

#### *job-ID*

Когато *query-type* е 1 или 2, указва идентификатора на подаденото задание. *job-id* е изходен параметър от тип VARCHAR(8).

#### *jobname*

Когато *query-type* е 1 или 2, указва името на подаденото задание. *jobname* е изходен параметър от тип VARCHAR(8).

#### *jcl-proc-time*

Указва часа на обработка на заявката. *jcl-proc-time* е изходен параметър от тип TIMESTAMP.

#### *last-statement*



Когато DSNACCHR връща тежка грешка (код на връщане 12), това поле съдържа SQL израза, който е бил изпълняван при възникване на грешката. *last-statement* е изходен параметър от тип VARCHAR(2500).

*return-code*

Кодът на връщане от изпълнението на DSNACCHR. Възможните стойности са:

- 0 - DSNACCHR е изпълнена успешно
- 12 - DSNACCHR е приключила с тежки грешки. Параметърът *error-msg* съдържа съобщение, което описва грешката. Параметърът *last-statement* съдържа SQL израза, който е бил изпълняван при възникване на грешката.

*return-code* е изходен параметър от тип INTEGER.

*error-msg*

Когато DSNACCHR връща тежка грешка (код на връщане 12), това поле съдържа съобщенията за грешки, включително форматирания SQLCA. *error-msg* е изходен параметър от тип VARCHAR(1331).

DSNACCHR връща един резултатен набор, когато параметърът *query-type* е 0. Резултатният набор съдържа JCL заданието, генерирано от DSNACCHR. Таблицата на DSNACCHR резултатния набор се създава от инсталационното задание DSNTIJCC. Таблица 12 показва формата на резултатния набор.

Таблица 12. Формат на DSNACCHR резултатен набор

Име на колона	Тип данни	Описание
JCLSEQNO	INTEGER	Пореден номер на табличен ред (1,...,n)
JCLSTMT	VARCHAR(80)	Указва JCL израз

## Преглеждане на обобщения за здравни тревоги

Функцията HEALTH\_OVERVIEW връща информация от Health Alert Summary VSAM KSDS набора данни като DB2 таблица. Този набор данни се създава от инсталационното задание DSNTIJHM.

Health Alert Summary наборът данни съдържа информация за състоянието на DB2 здравния монитор и статистики за обобщения на тревоги за всяка DB2 подсистема, наблюдавана преди или понастоящем от здравния монитор на тази MVS система или Parallel Sysplex. Тази информация се връща на клиента с ред за всяка препоръка за DB2 подсистема и тревога.

Резултатът от функцията е DB2 таблица със следните колони:

*ip-addr*

IP адресът на DB2 сървъра. Това е колона от тип VARCHAR(40).

*db2-ssid*

Идентификаторът на DB2 подсистема. Това е колона от тип VARCHAR(4).

*health-ind*

Тип на тревогата. Възможните стойности са:

- RS - Ограничено състояние
- EX - Надхвърлени екстенти
- RR - Изисква се REORG
- CR - Изисква се COPY
- RT - Изисква се RUNSTATS
- SS - Изисква се STOSPACE
- PO - Неуспешна оценка на пол.
- HM - Съст. на здр. монитор

*health-ind* е колона от тип VARCHAR(4).

*host-name*

Напълно квалифицираното име на домейн на DB2 сървър. Това е колона от тип VARCHAR(255).

*summary-stats*

Състоянието на DB2 здравния монитор, ако *health-ind* е 'HM'. Възможните стойности са:

- 0 Здравният монитор не е стартиран
- 1 Здравният монитор е стартиран
- -1 Състоянието на здравния монитор е неизвестно

Иначе е общият брой обекти на тревога с посочения в *health-ind* тип на тревога. Това е колона от тип INTEGER.

*alert-state*

Състоянието на тревога, указано в *health-ind*. Възможните стойности са:

- 5 - Тревога
- 4 - Внимание
- 3 - Предупр.
- 0 - Нормално

*alert-state* винаги е 0, когато *health-ind* е 'HM'. Това е колона от тип INTEGER.

Името на външна програма за функцията е HEALTH\_OVERVIEW, и специфичното име е DSNACC.DSNACCNO. Тази функция се създава от инсталационното задание DSNTIJCC.

**Пример:** Намерете общия брой обекти на тревога, изискващи COPY за DB2 подсистемата 'ABCD':

```
SELECT SUMMARYSTATS FROM TABLE (DSNACC.HEALTH_OVERVIEW()) AS T
WHERE DB2SSID = 'ABCD'
AND HEALTHIND = 'CR';
```

## Преглеждане на обекти на здравни тревоги

Обектите на тревоги, идентифицирани при последното успешно изпълнение на оценка на политика, се записват в тези хранилищни таблици за обекти на тревога, според техния тип на обект.

Обектите на тревоги са:

- DSNACC.HM\_MAINT\_TS за пространства за таблици
- DSNACC.HM\_MAINT\_IX за индекси
- DSNACC.HM\_MAINT\_SG за групи за съхранение

DB2 създава множество производни таблици на тези хранилищни таблици за обекти на тревоги. Производните таблици и хранилищните таблици за обекти на тревоги се създават от инсталационното задание DSNTIJC. Таблица 13 изброява таблиците, на които е дефинирана всяка производна таблица, и описанията на производните таблици. Всички имена на производни таблици и имена на таблици имат квалификатора DSNACC.

*Таблица 13. Производни таблици на обекти на здравни тревоги*

<b>Име на производна таблица</b>	<b>На таблица</b>	<b>Описание на производна таблица</b>
HM_ALERT_TS_RS	HM_MAINT_TS	Показва всички пространства за таблици в ограничено състояние
HM_ALERT_TS_EX	HM_MAINT_TS	Показва всички пространства за таблици, чиито екстенти са надхвърлили указано от потребителя ограничение
HM_ALERT_TS_RR	HM_MAINT_TS	Показва всички пространства за таблици, които изискват REORG
HM_ALERT_TS_CR	HM_MAINT_TS	Показва всички пространства за таблици, които изискват COPY
HM_ALERT_TS_RT	HM_MAINT_TS	Показва всички пространства за таблици, които изискват RUNSTATS
HM_ALERT_IX_RS	HM_MAINT_IX	Показва всички индекси, които са в ограничено състояние
HM_ALERT_IX_EX	HM_MAINT_IX	Показва всички индекси, чиито екстенти са надхвърлили указано от потребителя ограничение
HM_ALERT_IX_RR	HM_MAINT_IX	Показва всички пространства за индекси, които изискват REORG
HM_ALERT_IX_CR	HM_MAINT_IX	Показва всички индекси, които изискват COPY
HM_ALERT_IX_RT	HM_MAINT_IX	Показва всички индекси, които изискват RUNSTATS
HM_ALERT_SG_SS	HM_MAINT_SG	Показва всички групи за съхранение, които изискват STOSPACE



---

## Раздел 3. Висока достъпност и DB2 Connect

Има специфични съображения по отношение на високата достъпност в обкръжение, което използва DB2 Connect. Ако поради някаква причина сървър на база данни в мрежа стане недостъпен за използване, способността за пренасочване на клиентска работна станция към алтернативен сървър на база данни е важна.

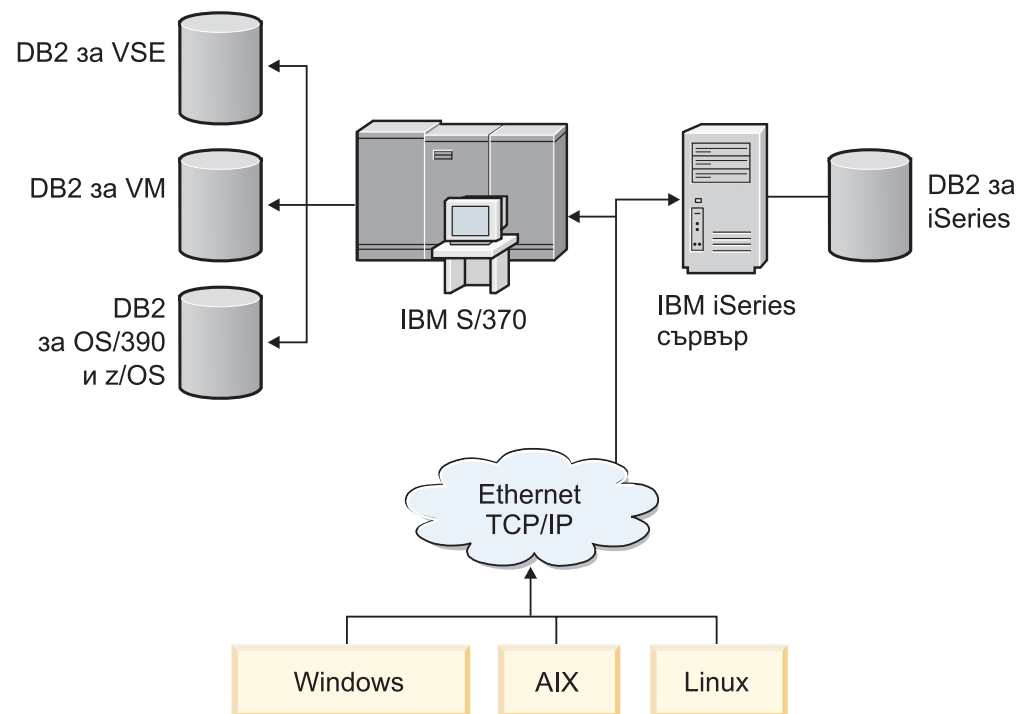


## Глава 11. Висока достъпност и балансиране на натоварването за свързваемост на хост база данни

В днешния пазар на информационни технологии има голямо търсене за денонощна достъпност на данни. Това изискване трябва да бъде посрещнато, за да се поддържа конкурентноспособността на даден бизнес и да се поддържа растежа му. Много от днешните web, e-business приложения и електронни таблици изискват достъп до корпоративни данни. Трябва да бъде установена надеждна, бърза и защитена връзка към хост или System i бази данни. Тази връзка трябва да бъде постоянно достъпна, и да може да поема високите изисквания за свързвания в условия на критично натоварване. Как може да бъде изградена такава връзка?

### Сценарий на висока достъпност

Компанията има няколко работни станции и сървъра, работещи под Windows, Linux, и UNIX. Тези машини изискват достъп до данни, разположени на няколко хост и System i бази данни. Приложенията, изпълнявани на тези машини, изискват бързи и надеждни свързвания към базите данни. Цялата система е свързана от Ethernet мрежа посредством TCP/IP.



Фигура 11. Примерен мрежов сценарий

За да може работните станции и сървърите на приложения да имат достъп до хост и System i бази данни, ви трябва междинен компонент за свързване. Този компонент трябва да предоставя високо достъпна, стабилна и бърза връзка към хост или System i бази данни. Трябва също да е мащабируем, за да може да посрещне бъдещия растеж в обема на свързванията.

Използвайте свързаните връзки от тази тема, за да прегледате подробности по отношение на решение, използващо DB2 Connect и характеристиката за автоматично пренасочване на клиент.



---

## Глава 12. Описание и настройка на автоматичното пренасочване на клиент (DB2 Connect)

Главната цел на характеристиката за автоматично пренасочване на клиент е да позволява на клиентските приложения на IBM Data Server Client да се възстановяват от загуба на комуникации, така че приложението да може да продължи работата си с минимално прекъсване. Както подсказва името, пренасочването е ключово за поддръжката на продължителна работа. Пренасочването обаче е възможно само, когато има алтернативно разположение, което се идентифицира на клиентското свързване. В различно от DB2 Connect обкръжение с висока достъпност, достъпваната база данни по принцип се синхронизира между оригиналния DB2 сървър и алтернативния DB2 сървър чрез някое от множество средства, каквито са High availability cluster multiprocessor (HACMP) или High availability disaster recovery (HADR).

В случая с DB2 Connect сървъра обаче, тъй като няма изискване за синхронизиране на локални бази данни, е необходимо само да се уверите, че и оригиналният, и алтернативният DB2 Connect сървъри имат целевата хост или System i база данни каталогизирана по такъв начин, че да е достъпна чрез използването на идентичен псевдоним на базата данни.

**Забележка:** В обкръжение на DB2 Connect сървър може да бъде указан алтернативен DB2 Connect сървър за активиране на автоматичното пренасочване между клиент и DB2 Connect сървъра. За да възникне пренасочване между DB2 Connect лични или сървърни продукти и хост или System i сървър на база данни, отдалеченият сървър трябва да предостави един или повече алтернативни адреса за себе си. В случая на DB2 за z/OS са известни множество адреси, ако базата данни е Sysplex обкръжение за поделене на данни.

Способността за пренасочване за Sysplex може да бъде конфигурирана между DB2 Connect и хост сървъра на база данни, ако е активирана Sysplex поддръжката. Способността за пренасочване за Sysplex е DB2 Connect характеристика, която позволява на DB2 Connect да опитва повторно свързване към други членове на Sysplex групата след загуба на комуникация с оригиналния член. Не е нужно алтернативният сървър да бъде каталогизиран в директорията на базата данни, за да се активира способността за пренасочване за Sysplex на DB2 Connect. По подразбиране способността за пренасочване за Sysplex е активирана, ако е активирана Sysplex поддръжката.

За да може IBM Data Server Client да има способността да се възстановява от загуба на комуникации към DB2 Connect сървър чрез автоматично пренасочване на клиент, трябва да бъде указано разположение на алтернативен DB2 Connect сървър, преди да възникне загубата на комуникация. Командата UPDATE ALTERNATE SERVER FOR DATABASE се използва за дефиниране на разположението на алтернативния DB2 Connect сървър за конкретна хост или System i база данни. Алтернативните име на хост и номер на порт се дават като част от командата. Разположението се съхранява в файла на системната директория на базата данни на DB2 Connect сървъра. За да се подsigури прилагането на посоченото алтернативно разположение на DB2 Connect сървър към тази база данни на всички клиенти, алтернативното разположение на сървър трябва да бъде указано от страната на DB2 Connect сървъра. Алтернативният сървър се игнорира, ако е зададен на клиентския потребителски модел.

Например, ако допуснем, че хост или System i база данни се каталогизират чрез псевдоним db1 на DB2 Connect сървъра S1 (с име на хост db2conn1 и номер на порт 122). Администраторът на базата данни би искал да укаже алтернативен DB2 Connect сървър S2 на име на хост db2conn2 с номер на порт 123. Това е командата, която би следвало администраторът на базата данни да изпълни на DB2 Connect сървъра S1:

```
db2 update alternate server for database db1 using hostname db2conn2 port 123
```

След като сте посочили разположението на алтернативен DB2 Connect сървър за псевдоним на база данни db1 на DB2 Connect сървър S1, информацията за разположение на алтернативния сървър се връща на IBM Data Server Client като част от процеса на свързване. Ако комуникацията между IBM Data Server Client и DB2 Connect сървъра S1 бъде изгубена по някаква причина (най-често комуникационна грешка, като SQL code -30081 или SQL code -1224), IBM Data Server Client ще опита да се свърже отново с db1 чрез оригиналния DB2 Connect сървър (S1) или алтернативния DB2 Connect сървър (S2), като редува опитите между двата сървъра. Времевият интервал между опитите в началото е кратък, и после постепенно се удължава при всеки опит.

След като едно свързване е успешно, се връща SQL code -30108, за да укаже, че е било установено повторно свързване към базата данни след отказа на комуникациите. Връщат се името на хост и IP адресът или номерът на порт. IBM Data Server Client връща само грешката за оригиналния отказ на комуникации към приложението, ако повторното установяване на клиентските комуникации не е възможно нито към оригиналния, нито към алтернативния сървър.

Следните съображения, включващи свързваемост към алтернативен сървър в обкръжение на DB2 Connect сървър, също следва да бъдат отбелязани:

- Когато се използва DB2 Connect сървър за осигуряване на достъп до хост или System i база данни от името на отдалечени и локални клиенти, може да настъпи объркване по отношение на информацията за свързваемост към алтернативен сървър в запис в системната директория на базата данни. За да минимизирате това объркване, обмислете каталогизирането на два записа в системната директория на базата данни, които да представят една и съща хост или System i база данни. Каталогизирайте един запис за отдалечени клиенти, и един запис за локални клиенти.
- Всяка SYSPLEX информация, която се връща от целевия сървър на DB2 за z/OS, се запазва само в кеша на DB2 Connect сървъра. Само един алтернативен сървър се записва на диск. Когато съществуват множество алтернативни или множество активни сървъри, информацията се поддържа само в паметта и бива изгубена, когато процесът прекъсне.

---

## Глава 13. Конфигуриране на автоматично пренасочване на клиент за технология на разпределител на клиентски свързвания

Технологиите за разпределител или диспечер като WebSphere EdgeServer разпределят заявките за повторно свързване на клиентски приложения в предварително дефиниран набор системи, ако първичният сървър на базата данни откаже. Ако използвате технологията на разпределител с DB2 автоматичното пренасочване на клиент, трябва да идентифицирате самия разпределител като алтернативен сървър за DB2 автоматичното пренасочване на клиенти.

Можете да използвате технологията за разпределител в обкръжение, подобно на следното:

Клиент → разпределител технология → (DB2 Connect сървър 1 или DB2 Connect сървър 2) → DB2 z/OS

където:

- Компонентът с технология на разпределител има TCP/IP име на хост **DThostname**
- DB2 Connect сървър 1 има TCP/IP име на хост **GWYhostname1**
- DB2 Connect сървър 2 има TCP/IP име на хост **GWYhostname2**
- DB2 z/OS сървърът има TCP/IP име на хост **zOShostname**

Клиентът е каталогизиран посредством **DThostname**, за да се използва технологията на разпределител за достъпване на който и да е от DB2 Connect сървърите. Участващата технология на разпределител взема решението да се използва **GWYhostname1** или **GWYhostname2**. Щом решението се вземе, клиентът има пряко сокет свързване към един от тези два DB2 Connect шлюза. Щом сокет свързваемостта се установи към избрания DB2 Connect сървър, имате типична свързваемост клиент към DB2 Connect сървър към DB2 z/OS.

Например, да приемем, че разпределителят избере **GWYhostname2**. Това води до следното обкръжение:

Клиент → DB2 Connect сървър 2 → DB2 z/OS

Разпределителят не опитва повторно никое от свързванията, ако има комуникационен отказ. Ако искате да активирате характеристиката за автоматично пренасочване на клиент за база данни в такова обкръжение, алтернативният сървър за асоциираната база данни или бази данни в DB2 Connect сървъра (DB2 Connect сървър 1 или DB2 Connect сървър 2) трябва да бъде настроен на разпределителя (**DThostname**). Тогава, ако DB2 Connect сървър 1 блокира поради някаква причина, се задейства автоматичното пренасочване на клиент, и клиентското свързване се опитва отново с с разпределителя като едновременно първичен и алтернативен сървър. Тази опция ви позволява да комбинирате и поддържате способностите на разпределител с DB2 характеристиката за автоматично пренасочване на клиент. Настройването на алтернативен сървър на хост, различен от хост името на разпределителя, продължава да осигурява на клиентите характеристиката за автоматично пренасочване на клиенти. Клиентите обаче ще установят преки свързвания към дефинирания алтернативен сървър, и ще заобиколят технологията на разпределител, което елиминира разпределителя и неговата стойност.

Характеристиката за автоматично пренасочване на клиенти прихваща следните SQL кодове:

- sqlcode -20157
- sqlcode -1768 (код на причина = 7)

**Забележка:** Пренасочването на клиенти може да не бъде информирано своевременно за отказ на сокет, ако настройката на конфигурационния параметър на операционната система "TCP Keepalive" е твърде висока. (Отбележете си, че името на този конфигурационен параметър варира според платформата.)

---

## Раздел 4. Настройка и DB2 Connect

При едно обкръжение на база данни, което използва DB2 Connect за преместване на заявки на базата данни и реакции между клиентски работни станции и сървъри на бази данни, има отделни съображения, когато се обмислят проблемите с производителността. Има няколко начина да се подобри или да се поддържа производителността в такова обкръжение.



---

## Глава 14. Съображения за производителност за DB2 Connect

*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 множество различни *показатели за производителност*, включително:

### **Време за отговор**

Интервалът време между момента, когато приложение изпрати заявка към база данни до момента, когато получи отговор.

### **Капацитет 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.

### **Равнище на трансфера на данни**

Броят байтове данни, прехвърлен между DB2 Connect приложението и хост или System i базата 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, подсистемите за страниране, mbuf за AIX, са примери за софтуерни ресурси.

### **Потоци данни**

Фигура 12 на страница 96 показва пътя на данните, протичащи между хост или System i сървъра на базата данни и работната станция чрез DB2 Connect.



Фигура 12. Потоци данни в DB2 Connect

- Хостът или System i базата данни и част от комуникационната подсистема В обикновено се изпълняват 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 други програми могат да поделят тези компоненти, съперничеството за ресурси може да причини проблеми с производителността.
- Мрежата се състои от комбинация от кабели, концентратори, комуникационни линии, превключватели и други комуникационни контролери. Например, хардуерният мрежов интерфейс В може да бъде комуникационни контролери като 3745 или 3172, или token ring адаптер за System i сървър. 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.
- DB2 Connect и комуникационната подсистема А обикновено са разположени в една система. За целите на тази дискусия се приема, че приложението е също 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.

## Сравнителни характеристики

*Сравнителни характеристики* сравнява производителността в едно обкръжение с производителността в 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.
- Няма активни функции или приложения в системата, различни от измерваните, освен ако сценарият включва друга дейност, имаща място 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

Следните таблици изреждат някои от средствата, които могат да ви помогнат да измерите системната производителност. Тъй като тези средства също използват системни ресурси, може да предпочетете да ги оставите активни през цялото време.

Таблица 14. Средства за измерване на производителност за употреба на CPU и памет

Система	Средство	Описание
AIX	vmstat, time, ps, tprof	Осигурява информация за проблеми при конкурентно използване на процесора и паметта на DB2 Connect работната станция и отдалечени клиенти.
HP-UX	vmstat, time, ps, монитор и наблюдение, ако са достъпни	
Windows	Microsoft Performance Monitor	

Таблица 15. Средства за измерване на производителност за активност в базата данни

Система	Средство	Описание
Всички	Монитор на базата данни	Определя дали проблемът е причинен от базата данни.

Таблица 15. Средства за измерване на производителност за активност в базата данни (продължение)

Система	Средство	Описание
OS/390 или zSeries	DB2PM (IBM), OMEGAMON/DB2 (Candle), TMON (Landmark), INSIGHT (Goal Systems) и DB2AM (BMC)	
Windows	Microsoft Performance Monitor	

Таблица 16. Средства за измерване на производителност за мрежова активност

Система	Средство	Описание
AIX	netrmon	Отчита мрежови статистики на ниско ниво, включително TCP/IP статистики като брой пакети или фреймове, получени за секунда.
Мрежов контролер като 3745	NetView Performance Monitor	Отчита употребата на комуникационен контрол и VTAM.
Linux и UNIX	netstat	Предназначено за TCP/IP трафик.

---

## Глава 15. Оптимизиране на ODBC достъпа

DB2 базата данни предоставя специална оптимизация, предназначена да подобри производителността на комуникация чрез ODBC. Тези подобрения са достъпни за Microsoft Access, Lotus Approach, или Visual Basic. Можете да получите предимствата на по-бързия ODBC, като използвате DB2 Асистента за конфигуриране (CA).

За да активирате оптимизирана ODBC:

- Ако дефинирате ново свързване:
  1. Стартирайте DB2 CA.
  2. Отворете менюто Избрано и изберете Добавяне на база данни чрез помощник...
  3. Следвайте страниците на помощника, докато достигнете страницата **Източник на данни**.
  4. Маркирайте **Регистриране на тази база данни за CLI/ODBC**.
  5. Укажете как следва да бъдат регистрирани CLI/ODBC приложенията, достъпващи тази база данни:
    - **Като системен източник на данни** означава, че базата данни е достъпна на всички потребители на системата.
    - **Като потребителски източник на данни** означава, че вие сте единственият потребител, който може да достъпва базата данни.
    - **Като файлов източник на данни** означава, че ще бъде създаден файл, съдържащ информация за източника на данни. Този файл за източника на данни може да бъде поделен с други работни станции, ако имате TCP/IP свързване. В противен случай, файлът може да бъде използван само на този компютър
  6. Въведете **Име на източник на данни**.
  7. (По избор) Изберете приложение от списъка **Оптимизиране за приложение**, за да оптимизирате настройките на източник на данни за конкретно приложение.
  8. Щракнете върху **ОК** и излезте от CA.
- Ако обновявате съществуващо свързване:
  1. Стартирайте DB2 CA.
  2. Щракнете два пъти на псевдонима на базата данни, която искате да оптимизирате.
  3. Щракнете на **Източник на данни**.
  4. Маркирайте **Регистриране на тази база данни за CLI/ODBC**.
  5. Укажете как следва да бъдат регистрирани CLI/ODBC приложенията, достъпващи тази база данни:
    - **Като системен източник на данни** означава, че базата данни е достъпна на всички потребители на системата.
    - **Като потребителски източник на данни** означава, че вие сте единственият потребител, който може да достъпва базата данни.
    - **Като файлов източник на данни** означава, че ще бъде създаден файл, съдържащ информация за източника на данни. Този файл за източника на данни може да бъде поделен с други работни станции, ако имате TCP/IP свързване. В противен случай, файлът може да бъде използван само на този компютър
  6. Въведете **Име на източник на данни**.

7. (По избор) Изберете приложение от списъка **Оптимизиране за приложение**, за да оптимизирате настройките на източник на данни за конкретно приложение.
8. Щракнете върху **ОК** и излезте от СА.

---

## Глава 16. Дизайн на приложение

Когато създавате приложение, можете да подобрите производителността по няколко начина.

### Съставен SQL 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
- Издание
- Describe
- Rollback
- Disconnect
- Set connection
- execute immediate

Запомнените процедури помагат да се намали мрежовия трафик, като поставят програмна логика на сървъра. Можете автоматично да изпълнявате комит при излизане от процедурата. Освен това можете да върнете резултатен набор, който намалява логиката на приложението от страната на клиента.

### 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.

Можете също да използвате ключови думи като IN и BETWEEN, за да намалите броя върнати редове. В добавка можете да използвате ключовите думи WHERE, IN, и BETWEEN на изрази 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 само първият ред на TABLEA с ROW\_ID=1 е действително необходим, или ако са нужни само колона 1 и колона 2.

### Блокове данни

Трябва да създадете блокове данни, ако очаквате голямо количество данни от сървъра. Създаването на блокове подобрява употребата на мрежовата честотна лента и намалява натоварването на CPU на хост или System i сървър на база данни, и на DB2 Connect сървър. 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.

В рамките на DB2 Connect, можете да контролирате обема данни, които се прехвърлят в рамките на всеки блок.

За да извикате създаването на блокове, използвайте опцията BLOCKING на командата rgr или bind. Създаването на блокове е включено, ако:

- Указателят е само за четене или
- Указателят е двусмислен, и е указано създаване на блокове по време на rgr или bind.

**Забележка:** Когато се използва динамичен SQL, указателят винаги е неопределен.

### SQL изрази с BLOCKING

Обновяемите SELECT изрази (при използване на UPDATE/DELETE WHERE CURRENT OF изрази) не са създаващи блокове заявки, затова следва да ги използвате само, ако е абсолютно необходимо.

Обновяемият SELECT подsigурява, че редът не е бил променен между завършването на SELECT и подаването на UPDATE/DELETE. Ако това ниво на едновременност 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.

За SELECT само за четене, укажете 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 изразът е SELECT и трябва да бъде подготвен, 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 последователността PREPARE и DESCRIBE. Използването на израза PREPARE . . . . INTO осигурява по-добра производителност.

- Изпълнете статично свързани оператори SQL COMMIT или ROLLBACK вместо динамични оператори COMMIT или ROLLBACK.
- Ако не е SELECT, COMMIT, или ROLLBACK израз, подайте EXECUTE IMMEDIATE, за да изпълните израза вместо последователността PREPARE и EXECUTE.
- ODBC приложения използват динамичен SQL. Можете да използвате CLI/ODBC характеристиката за статично профилиране за подобряване на производителността. Тази функция ви позволява да прихванете и конвертирате ODBC обръщения в статични оператори, записани в пакет на база данни. Действителната производителност, която ще получите, зависи от сложността на вашето приложение.

### Други SQL съображения

Използването на Процесор за обработка на команди (CLP) е, като цяло, по-бавно от използването на динамичен SQL в програмата, защото CLP трябва да преобразува въведеното, преди да подаде SQL към машината на базата данни. CLP също форматира данни, когато ги получи, което може да не е необходимо за вашето приложение.

SQL изразите в интерпретиран език, какъвто е REXX, са значително по-бавни от същите SQL изрази в компилиран език, какъвто е 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. По тази причина, използването на свързване тип 2 може да подобри производителността за приложения, достъпващи повече от една база данни.





---

## Глава 17. Управление на свързвания

Управлението на свързвания се състои от две операции: пулиране на свързвания и концентратор на свързвания. Пулирането на свързвания намалява натоварването на свързванията към базата данни и управлява обема на свързвания. Концентраторът на свързвания увеличава скалируемостта във вашето работно обкръжение, като оптимизира ресурсите, използвани от повечето хост сървъри на бази данни. И двете операции се разглеждат тук.

---

### Пулиране на свързвания

DB2 Connect сървърните продукти като DB2 Connect Enterprise Edition често предоставят свързвания към база данни за хиляди едновременни клиентски заявки. Установяването и прекъсването на свързвания към сървъра на базата данни може да бъде извънредно ресурсоемък процес, който засяга силно производителността едновременно на сървъра на базата данни и на DB2 Connect сървъра.

Този проблем е особено очевиден в web обкръженията, където всяко посещение на web страница може да изисква изграждането на ново свързване към сървъра на базата данни, изпълнението на заявката и прекратяването на свързването. За да се намали това натоварване, DB2 Connect сървърните продукти използват пулиране на свързванията, за да поддържат отворени свързвания към базата данни в готов за достъп пул.

Повечето базирани на web технологии приложения изпълняват голям обем къси транзакции. Типична web транзакция се изпълнява като част от собственото й свързване. С други думи, изпълнението на транзакция означава установяване на свързване към базата данни, и прекъсване на това свързване само след няколко SQL израза. Този процес по установяване и прекратяване на връзка е много скъп. Това включва създаването на DB2 Connect агент, установяването на мрежово свързване между този агент и DB2 сървъра, и създаването на DB2 нишка на сървъра. За по-дълго изпълняваните свързвания разходите се амортизират върху всички изпълнени върху това свързване транзакции, но за типична web транзакция тези разходи типично надхвърлят стойността на изпълнението на самата транзакция.

Пулът на свързванията е техника, която позволява повторната употреба на инфраструктурата на установеното свързване за последващи свързвания. Когато се стартира DB2 Connect потребителски модел, се създава пул с координиращи агенти. Когато пристигне заявка за свързване, ѝ се присвоява агент. Агентът ще се свърже към DB2 сървъра, и в DB2 ще бъде създадена нишка. Когато приложението подаде заявка за прекъсване, агентът няма да препредаде тази заявка на DB2 сървъра. Вместо това, агентът се оставя обратно в пула. Агентът в пула продължава да притежава свързването към DB2 сървъра и съответстващата му DB2 нишка. Когато друго приложение подаде заявка за свързване, този агент се присвоява на това ново приложение. За да се осигури сигурно функциониране, информацията за идентифициране на потребителя се подава на DB2 нишката, която на свой ред изпълнява разпознаването на потребителя.

Създаването на пул на свързванията 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.

Пулът за връзките е прозрачен за приложенията, които се свързват към хоста чрез DB2 Connect. Когато приложение заяви прекъсване на свързването към хоста, DB2 Connect прекъсва входящата връзка с приложението, но запазва в пул изходящото свързване към хоста. Когато ново приложение заяви свързване, DB2 Connect използва едно от свързванията в съществуващия пул. Използването на вече установена връзка намалява общото време за свързване, както и високото натоварване на процесора при свързване към хоста.

DB2 Connect агентите могат да бъдат в едно от двете състояния: свободни или активни. 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 посредством 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. По подразбиране този параметър се настройва на AUTOMATIC със стойност 100. Като е настроен на AUTOMATIC, DB2 Connect автоматично управлява броя свободни агенти в пула за свободни агенти.

DB2 Connect не установява свързване към базата данни, преди да получи първата заявка от отдалечен клиент. Като алтернатива, можете да попълните пула от свободните агенти, преди някой клиент да направи заявка. Пулът може да бъде запълнен при стартиране посредством 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 сървъра 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 сървъра на базата данни. Ако не успее, ще се опита да намери достъпен агент в пула със свободни агенти. Ако пулът е празен, DB2 Connect ще създаде нов агент.

Можете да управлявате максималния брой агенти, които могат да са активни едновременно, като използвате 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.) По подразбиране този параметър се настройва на AUTOMATIC със стойност 200. Като е настроен на AUTOMATIC, DB2 Connect автоматично управлява броя координиращи агенти.

DB2 регистърната променлива DB2CONNECT\_IN\_APP\_PROCESS позволява на приложения, изпълнявани на същата машина като DB2 Connect сървърния продукт, да изпълняват DB2 Connect в процеса на приложенията, както е по подразбиране, или да насочат приложението да се свърже с DB2 Connect сървърния продукт, и после да изпълнят свързването към хост в агент. За да може приложение да използва пулирането, свързванията към хоста трябва да бъдат правени от вътрешността на агентите на DB2 Connect сървърния продукт, и затова DB2CONNECT\_IN\_APP\_PROCESS трябва да бъде настроен на NO.

## DB2 Connect пул на свързванията срещу пул на свързванията на сървъра на приложения

Пулът на свързванията е задължителен за всяко базирано на web технологии приложение, което трябва да поддържа голям обем транзакции. Повечето web сървъри на приложения сега предлагат собствени начини да се изградят пулове на свързванията към база данни. Например, и Microsoft MTS (COM+), и IBM WebSphere осигуряват пулиране на свързванията.

Механизмите за пулове на свързвания, реализирани от тези сървъри, се различават значително от предоставените от DB2 Connect сървърите. Тъй като сървърите на приложения пулират свързвания само за собствена употреба, те по принцип изхождат от предпоставката, че потребителският идентификатор, парола, нива на изолация и прочие, са еднакви за всички свързвания. Дори по-важно, сървърите на приложения поставят в пул само свързванията, инициирани от един и същи процес. Това означава, че свързванията от други машини, потребители или процеси не попадат в пула. Докато тези техники за пул на свързванията на сървърите на приложения са ефективни за повторно използване на свързванията от същия потребителски модел или приложение, те са напълно неефективни за пул на свързвания от различни потребители, сървъри, и т.н.

Пулът на свързванията, предоставен от DB2 Connect сървърите, е напълно независим от приложението, машината и потребителя. Свързванията от множество клиенти, сървъри на приложения, всичките с различни потребителски идентификатори, могат да използват повторно свързванията едно на друго, което води до много по-добро използване на ресурсите в пула.

Кой тип пул на свързвания е правилно да се използва? И двата. Като цяло, използването едновременно на пул за свързвания на DB2 Connect и използването на пул за свързвания на сървъра на приложения е добра стратегия, тъй като те не си въздействат взаимно. Дори когато е активиран пул за свързвания на сървъра на приложения, DB2 Connect пулът за свързвания може да осигури повторно използване на свързването за множество сървъри на приложения, както и за други клиенти, използващи DB2 Connect сървър.

---

## Концентратор на свързвания

Концентраторът на свързвания намалява ресурсите, изисквани на DB2 за OS/390 и z/OS сървъри на бази данни за поддръжка на големи количества работни станции и web потребители. Тази функция може драстично да увеличи скалируемостта на вашето DB2 за OS/390 и z/OS и DB2 Connect решение, като осигурява също подсигурана срещу откази работа и балансиране на натоварването на ниво транзакция в DB2 за OS/390 и z/OS среди с поделяне на данни.

Концентраторът на свързвания позволява на приложенията да остават свързани, без да се консумират ресурси на DB2 хост сървъра. Можете да имате хиляди активни потребители в приложенията, и само няколко активни нишки на DB2 хост сървъра.

Технологията на DB2 Connect за *концентратор на свързвания* позволява на DB2 Connect сървърните продукти, като DB2 Connect Enterprise Edition, да предоставят поддръжка на хиляди потребители, изпълняващи едновременно бизнес транзакции, като същевременно драстично намаляват изискванията за ресурси на S/390 хост или System i сървърите на бази данни. Тази цел се постига чрез концентриране на натоварването от всички приложения в много по-малък брой свързвания към S/390 хост или System i сървър на база данни. Въпреки че това може да изглежда подобно на функцията за пулиране на свързвания, описана по-горе, всъщност представлява

по-изтънчен подход към намаляване на консумацията на ресурси за високообемни OLTP (On-line Transaction Processing) приложения.

Концентраторът на свързвания взема концепцията за агент, и я разделя на две единици:

- Логически агент, който представлява свързване на приложение.
- Координиращ агент, който притежава DB2 свързването и нишката, и изпълнява заявки на приложения.

Когато ново приложение опита свързване към хоста, му се приписва логически агент. За да се предаде SQL на базата данни, се изисква координиращ агент, който се приписва веднага, щом се задейства нова транзакция. Ключът към тази архитектура е фактът, че координиращият агент е:

- Деасоцииран от логическия агент
- Връзан в пула, щом транзакцията завърши поради комит или ролбек

Друга ключова характеристика е методът на приписване на координиращи агенти към нови транзакции в обкръжение със споделяне на данни. DB2 Connect реализира усъвършенстван алгоритъм за планиране, който използва информация от OS/390 и z/OS Work Load Manager (WLM). Тази информация се използва за разпределяне на натоварването между членовете на група за поделяне на данни според заложените в WLM критерии. WLM не само следи натоварването на всеки член, но и тяхната достъпност. Това позволява на DB2 Connect прозрачно да пренасочва работата от отказалите или претоварени членове към членове, които са функционални и недостатъчно натоварени. Концентраторът на свързвания на DB2 Connect се активира, когато настроите броя на максимални логически агенти (*max\_connections*) да е по-висок от броя на координиращите агенти (*max\_coordagents*).

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.

Като алтернатива концентраторът на свързвания позволява на DB2 Connect да прави свързване достъпно за приложение, щом друго приложение завърши транзакция, и не изисква другото приложение да се изключи. Всъщност, свързването към сървър на база данни, и асоциираните с него 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.

В предните версии на DB2 Connect, всяко активно приложение имаше Engine Dispatchable Unit (EDU), който управляваше свързването на базата данни, както и всички заявки на приложения. Тази EDU обикновено се разглежда като *агент координатор*. Всеки агент координатор проследяваше състоянието или контекста на приложението и EDU. Всяка EDU заема значителен обем памет, когато броят на свързванията се увеличава, и превключването на контексти между агентите води до 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. Тоест, взаимовръзката на свързванията (X) към EDU (Y) вече 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. Работните агенти се асоциират с логически агенти за изпълнението на транзакции, и в края на транзакцията прекъсват асоциирането и се връщат в достъпния пул.

Единица, известна като *диспечер* приписва работни агенти към логическите агенти. Ограниченията в броя на отворените файлови указатели на някои изчислителни платформи могат да доведат до повече от един модел на планировчик.

## Ограничения за концентратора на свързвания

Има известно количество важни ограничения при използването на DB2 Connect сървърния концентратор. Разгледайте следващата информация в нейната цялост, преди да се опитате да използвате концентратор за връзки във вашата система.

Общи ограничения:

- Концентраторът разчита на TCP/IP протокола, 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 приема свързвания чрез други комуникационни протоколи като *named pipes*, но няма да можете да използвате неговите характеристики за ХА концентрация с това свързване.
- За поддръжка на ХА тясно съчетани транзакции, всички приложения, които участват в една ХА транзакция, трябва да използват DB2 Connect сървърен потребителски модел за свързване към хоста.
- Само приложения, които затварят задържаните ресурси (като задържани указатели) на границите на транзакции, могат да извлекат ползва 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.
- Ако декларирате глобални временни таблици, те трябва изрично да бъдат затворени на границата на транзакция или клон. Незатварянето на таблиците ще изключи концентрирането на свързванията, но приложението ще продължи да работи.
- Всички приложения, участващи в една ХА транзакция, трябва да имат еднакви CCSID и да използват един потребителски идентификатор за осъществяване на свързването.
- Ако изходящо свързване е установено да поддържа двуфазово свързване, агентът на това свързване може да се използва само за поддръжка на двуфазови свързвания. По подобен начин, агентите, установени да поддържат еднофазово свързване, могат да поддържат само еднофазови свързвания.
- Концентраторът поддържа само динамичен SQL от интерфейса за вградени SQL оператори (CLI). CLI приложенията също не трябва да ползват KEEPDYNAMIC, защото концентраторът зависи от това изразите да се подготвят предварително на всяка граница на транзакция.
- 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.

При работа с DB2 Версия 9 или Версия 8 FixPak 13 (или по-висок), активирането на DB2 Connect поддръжката на концентратор изисква System i Version 5 Release 4 (PTF

SI23726). В противен случай се поддържа само XA частта на концентратора на свързвания.

## Активиране на концентратора на свързванията

Конфигурационният параметър на мениджъра на базата данни *max\_coordagents* задава максималния брой логически агенти. Можете да активирате характеристиката за концентратор, като настроите стойността на *max\_connections* на всяко число, по-голямо от стойността по подразбиране. Стойността по подразбиране за *max\_connections* е еквивалентна на стойността на *max\_coordagents*. Тъй като всяко приложение ще има един логически агент, *max\_connections* в действителност контролира броя на приложенията, които могат да бъдат свързани към потребителски модел на базата данни, докато *max\_coordagents* контролира броя входящи свързвания, които могат да бъдат активни в даден момент. *max\_connections* ще бъде в числения обхват от *max\_coordagents* до 64,000. Броят по подразбиране на логическите агенти е равен на *max\_coordagents*.

И *max\_connections*, и *max\_coordagents* могат да бъдат настроени на AUTOMATIC. Ако *max\_connections* е настроен на AUTOMATIC, броят на свързванията може да бъде увеличен над основната конфигурирана стойност. Ако *max\_connections* и *max\_coordagents* са едновременно настроени на AUTOMATIC, *max\_connections* може да бъде увеличен над базовата стойност, и *max\_coordagents* се увеличава автоматично, за да се запазва съотношението за концентрация между свързванията и координиращите агенти.

Редица съществуващи конфигурационни параметри се използват за конфигуриране на агенти. Тази параметри са както следва:

### *max\_coordagents*

Максималният брой на активни агенти координатори.

### *num\_poolagents*

Размер на пула за агенти. Пултът за агенти включва неактивните агенти и свободните агенти. За подобрена производителност, *num\_poolagents* следва да бъде конфигурирано да е равно на средния брой клиенти.

### *num\_initagents*

Първоначалният брой на работещите агенти в пула. Това ще са свободните агенти.

## поддръжка на XA транзакция

Архитектурата на концентратора на свързвания позволява на DB2 Connect да предоставя поддръжка на тясно съчетана XA транзакция на DB2 за OS/390 и z/OS и DB2 за System i. 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 прикрепи 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 за поддръжка на ХА транзакции

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 ще изисква DB2 Connect да работи като отделен шлюз, тоест, базата данни и DB2 Connect системата са на отделни машини.

DB2 Connect сървърната система може да не е в състояние да поддържа 4,000 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_CONNECTIONS = 4,000
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. Тези агенти, които не са в покой могат да поддържат свързване към базата данни, но не участват в конкретна транзакция, поради което са свободни за всеки логически агент (приложение), който изисква свързване.

Случаят с ХА транзакциите е донякъде различен. За целите на този пример, приемете, че TP монитор се използва с DB2 Connect шлюз, и zSeries или System i база данни. Когато приложение заяви свързване, концентраторът или ще превключи неактивен агент, за да обслужи тази заявка, или ще създаде нов работещ агент. Допускаме, че приложението изисква ХА транзакция. Създава се ХИД 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 ХИД на транзакцията. Сега може да обслужва заявки само за транзакции с неговия асоцииран ХИД.

В този момент, друго приложение може да направи заявка за не-ХА транзакция. Дори ако няма други достъпни работещи агенти, агентът, асоцииран с ХИД, 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, заявяващи транзакцията, асоциирана с ХИД на първия агент, могат да се прикрепят и отделят от този агент, който изпълнява заделената му ХА транзакция от тяхно име. 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 ХИД 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.

---

## Пулове и концентратор на свързвания

Въпреки че пулт на свързвания и концентраторът на свързвания изглеждат подобни, те се различават в реализацията си и разрешават различни въпроси. Пулт на свързвания помага да се намали прекомерното натоварване на свързванията на базата данни и боравенето с обема на свързванията. Концентраторът на свързвания подобрява скалируемостта на вашето DB2 за OS/390 и z/OS и DB2 Connect решение, като оптимизира използването на вашите хост сървъри на бази данни.

Когато се използва пул на свързвания, свързването става достъпно за повторна употреба, след като притежаващото го приложение подаде заявка за прекратяване на свързването. В много двуслойни клиент-сървър приложения потребителите не прекъсват връзката в продължение на работния ден. По подобен начин, повечето сървъри на приложения в многослойни приложения установяват връзката при стартиране на сървъра и не я освобождават, докато не бъдат затворени.

В такива обкръжения от пул на свързванията има малко, ако има изобщо, полза. В случаите с web и клиент-сървър обкръженията, обаче, където честотата на свързвания и прекратявания е по-висока, пулт на свързванията ще доведе до значителни подобрения в производителността. Концентраторът на свързвания разпределя ресурсите на хост базата данни само за продължението на SQL транзакция, като потребителските приложения се поддържат активни. Това позволява броят DB2 нишки и ресурсите, които те използват, да бъде много по-малък, отколкото ако всяко свързване на приложение има собствена нишка.

Когато се стигне до безпогрешна работа и балансиране на натоварването, концентраторът на свързвания е явно правилният избор, тъй като позволява преразпределение на работата с всяка нова транзакция. От друга страна, пулт на свързвания позволява много ограничено балансиране, и то само по време на свързване.

Пулът на свързвания и концентраторът на свързвания трябва да бъдат използвани заедно, въпреки че разрешават различни проблеми.

---

## Концентратор на свързвания, изискван с WebSphere MQ Transaction Manager и DB2 за OS/390

Когато изпълнявате приложения в IBM WebSphere MQ (известно преди като IBM MQSeries) обкръжение, WebSphere MQ може да действа като XA-съвместим мениджър на транзакции, координиращ всички разпределени транзакции с двуфазов комит. Когато WebSphere MQ действа като мениджър на транзакции по този начин, и източниците на данни са от семейството DB2 продукти, има няколко конфигурационни изисквания.

Повечето от конфигурационните изисквания в такова обкръжение на мениджър на транзакции вече са документирани другаде. Например, можете да настроите конфигурационния параметър TP\_MON\_NAME на DB2 на "MQ" на DB2 изпълнимия клиент.

Има обаче конфигурационно изискване, което липсва. Изискването е специфично за DB2 Connect при свързване към източници на данни, които са DB2 за OS/390 сървъри: когато се използва WebSphere MQ за координиране на разпределени транзакции, включващи DB2 за z/OS и DB2 за i5/OS сървъри, характеристиката на концентратор на свързвания на DB2 Connect трябва да бъде активирана на шлюза. Концентраторът



на свързвания е активиран, когато стойността на конфигурационния параметър **max\_connections** е по-голяма от стойността на конфигурационния параметър **max\_coordagents**.

Ако не активирате концентратора на свързвания, това ще доведе до неочаквано поведение на транзакциите.



---

## Глава 18. DB2 Connect Sysplex поддръжка

Sysplex е набор от System z сървъри, които си сътрудничат, използвайки хардуер и софтуер, за обработка на натоварване. Sysplex координира съдействието, като увеличава броя на работещите заедно процесори, което увеличава обема работа, който може да бъде обработен. В добавка към увеличението на възможността за обработка, един Sysplex може да осигури гъвкавост чрез смесване на нива на хардуер и софтуер, и чрез динамично добавяне на системи.

Sysplex позволява на DB2 Connect прозрачно да балансира свързвания между различни членове на група за поделяне на данни. Sysplex също осигурява на DB2 Connect средствата да опитва алтернативни членове в случай, че възникне отказ на някой член. Способността за пренасочване за Sysplex е характеристика на DB2 Connect. DB2 Connect поддръжката за Sysplex е активирана по подразбиране, както и способността за пренасочване за Sysplex. Sysplex поддръжката на хост база данни може да бъде изключена чрез премахването на параметъра SYSPLEX от нейния запис в DCS директорията, но самият DCS запис не трябва да бъде премахван, дори ако няма указан друг параметър.

Със способността за автоматично пренасочване на клиент за Sysplex, подразбиращото се поведение за Sysplex активирано свързване е да опита отново свързването, когато има комуникационен отказ. Стойностите на специални регистри до последната успешна транзакция, която не задържа ресурси, се преиграват, когато DB2 Connect е свързана към сървър на DB2 за z/OS.

Можете да конфигурирате точното поведение за автоматичен повторен опит за пренасочване на клиент, включително деактивиране, като използвате регистърните променливи DB2\_MAX\_CLIENT\_CONNRETRIES и DB2\_CONNRETRIES\_INTERVAL. Регистърната стойност за таймаут на свързване е DB2TCP\_CLIENT\_CONTIMEOUT.

---

### Съображения за OS/390 и zSeries SYSPLEX употребата

DB2 Connect осигурява балансиране на натоварването и толерантност към грешки при пренасочване на свързвания към множество Sysplex-и. Когато е свързан към DB2 за OS/390 и z/OS сървър на база данни, изпълняван в обкръжение с поделяне на данни, DB2 Connect ще разпределя натоварването между различните DB2 подсистеми, съставляващи групата за поделяне на данни, на база информацията за натоварване, осигурена от Workload Manager (WLM).

DB2 Connect получава приоритизиран списък Sysplex членове от WLM. Всеки Sysplex връща претеглена информация за приоритет за всеки адрес на свързване. Този списък после се използва от DB2 Connect за боравене с входящите CONNECT заявки, като се разпределят между Sysplex членовете с най-високите присвоени приоритети. За балансиране на натоварването, списъкът на Sysplex претеглената информация за приоритети се получава при всяко свързване. Ако DB2 Connect концентраторът на свързвания е активиран, този списък се използва също за определяне къде да се изпрати всяка транзакция.

**Забележка:** OS/390 и z/OS Distributed Data Facility (DDF) конфигурацията не трябва да бъде променяна, за да се използват предимствата на DB2 Connect Sysplex употребата.

DB2 Connect също осигурява толерантност към грешки чрез опитване на свързване към алтернативна sysplex машина в случай на отказ на свързване. На приложението ще бъде върната грешка само, ако всички известни свързвания са отказали.

DB2 Connect Sysplex е разработена с взето предвид пулиране на агентите. С активиран Sysplex, DB2 Connect пренасочва свързванията към друг DDF член в случай, че свързването към участващ член бъде изгубено. Пренасочването се постига чрез списък Sysplex сървъри.

С добавянето на концентратора, DB2 Connect вече има възможността да балансира натоварванията на границите на транзакции. DB2 Connect концентраторът трябва да бъде активиран, за да може това да работи.

---

## DB2 Sysplex употреба

В типичен сценарий, DB2 Connect сървър (сървър А) ще бъде в обмен със Sysplex, съдържащ два DB2 за OS/390 и z/OS сървъра (сървъри В и С).

Sysplex сървър В	Sysplex сървър С
HOST_NAME=MVSHOST	HOST_NAME=MVSHOST1

Да допуснем, че в този сценарий приложение подава:

```
db2 connect to aliasb user xxxxxxxx using xxxxxxxx
```

Свързването към база данни MVSHOST е установено. Тъй като Sysplex употребата е активирана за DB2 Connect сървъра и за записа в DCS директория, DB2 за OS/390 и z/OS идентифицира мрежовия адрес до DB2 Connect за всеки Sysplex участник (MVSHOST и MVSHOST1. DRDA4 протоколите и потоците съобщения се използват за връщане на тази информация). След като бъде осъществено началното свързване, върнатият списък с адреси се кешира на DB2 Connect работната станция. След като началният CONNECT бъде подаден за TCP/IP възел, се връщат IP адресите.

### Информация за приоритет, използвана за балансиране на натоварване и толерантност към грешки

Списъкът с адреси, осигурен от DB2 за OS/390 и z/OS, включва също информация за приоритет, включително броя свързвания за всеки мрежов адрес. Списъкът се опреснява винаги, когато се направи ново свързване от DB2 Connect. Тази допълнителна информация се използва за целите на балансиране на натоварването, както и за толерантност към грешки.

### Кеширан списък с адреси, използван от DB2 Connect

Ако свързването към базата данни ALIASB откаже, се подава съобщение за грешка SQL30081N, и свързването ще бъде отстранено. Ако се получи допълнителна заявка за свързване за ALIASB, DB2 Connect прави следното:

1. Опитва сървъра с най-висок приоритет от кеширания списък с адреси на база информацията за приоритети, върната от DB2 за OS/390 и z/OS. Тази стратегия винаги се използва от DB2 Connect, и е средството, чрез което се постига балансирането на натоварването.
2. Ако този опит за свързване откаже, се опитват другите адреси в списъка в низходящ ред по приоритета им, както са върнати от DB2 за OS/390 и z/OS. По този начин DB2 Connect използва Sysplex информацията за постигане на толерантност към грешки.

3. Ако всички останали опити за свързване са неуспешни, DB2 Connect ще опита отново свързването към ALIASB, като използва адреса, съдържащ се в директорията на каталогизирания възел.

Командата `db2pd` с параметъра `sysplex` (`db2pd -sysplex`) може да бъде използвана за извличане на информация за сървъри, асоциирани със Sysplex обкръжение.

---

## Конфигурационни параметри за Sysplex

Sysplex употребата няма да се използва за дадена база данни, освен ако записът в DCS директория за тази база данни съдържа Sysplex (без разлика малки или големи букви) в 6-тия позиционен параметър.



---

## Глава 19. Настройка на DB2 Connect

За настройка на DB2 Connect могат да се използват различни параметри от конфигурационния файл на мениджъра на базата данни.

### RQRIOBLK

Параметърът **RQRIOBLK** задава максималния размер на мрежови I/O блокове. По-големият размер на блок може да подобри производителността на обемистите 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 блок обикновено изисква повече памет на DB2 Connect сървъра. Това увеличава размера на работния набор, и може да причини големи обеми страниране на малки работни станции.

Използвайте подразбиращия се DRDA размер на блок (32767), ако той не причинява прекалено много страниране при изпълнение на приложението ви. В противен случай намалете размера на входно/изходния блок, докато не се получава страниране. Ако се получи страниране, производителността забележително ще се влоши. Използвайте средства за наблюдение на производителността (като средството vmstat за Linux и UNIX операционни системи), за да определите дали на системата ви се проявява страниране.

### DIR\_CACHE

Параметърът **DIR\_CACHE** определя дали информацията за директория се кешира. С кеширане (DIR\_CACHE=YES), файловете на директория се прочитат и кешират в паметта, за да се намали натоварването от създаване на вътрешна структура на директорията и прочитане на файловете й всеки път, щом се установи свързване.

Без кеширане (DIR\_CACHE=NO), винаги, когато се свържете към база данни, подходящата директория се прочита от диск, и после се изпълнява търсене. След като бъдат намерени заявените записи, цялата памет, отделена за търсене в директорията, се освобождава.

С кеширане, кешът на поделена директория се изгражда по време на db2start обработка, и се освобождава, когато DB2 спре. Този кеш се използва от всички DB2 сървърни процеси (db2agent). Също така се изгражда собствен кеш на директория, когато приложение подаде първа заявка за свързване към база данни, и се освобождава, когато приложението завърши.

Всеки кеш осигурява копие на системната директория на базата данни, на директорията на услугите за свързване към базата данни и на директорията на възлите. Кешът намалява стойността на свързване, като премахва I/O на файлове в директорията и минимизира търсенията в директорията.

Ако кеширана директория бъде обновена, измененията не се разпространяват незабавно в кешовете. Ако записът в директория не бъде намерен в кеша, се претърсва оригиналната директория.

Кеширането увеличава собствената памет, която е необходима за живота на едно приложение. Без кеширане тази памет е необходима само, когато се обработва

претърсване в директория. Цялостната употреба на поделена памет от DB2 се увеличава леко, защото информацията за директорията, която се поделя между агентите на базата данни, се премества в поделената памет. Размерът на необходимата за кеширане памет зависи от броя записи, дефинирани във всяка директория.

## NUMDB

Поведението на DB2 Connect не се влияе от конфигурационния параметър **NUMDB** в предни версии, но това е променено, считано от Версия 8. Този параметър указва максималния брой бази данни, към които могат да се свързват клиенти чрез DB2 Connect сървъра. По-конкретно, максималният брой различни псевдоними бази данни, които могат да бъдат каталогизирани на DB2 Connect сървъра.

## Други DB2 Connect параметри

**AGENTPRI** и **MAXAGENTS** са отхвърлени във Версия 9.5

Командите за обновяване на стойността за **MAXAGENTS** ще продължат да работят, така че съществуващите приложения да не са повредени, но стойностите ще се игнорират. Името на параметъра няма да се появява в никой конфигурационен списък. В миналото общият брой агенти, позволени за създаване на даден DB2 дял, се контролираше чрез конфигурационния параметър **MAXAGENTS**. Сега имате възможността да автоматизирате конфигурирането на агенти.

По подразбиране **NUM\_POOLAGENTS** ще бъде настроен на **AUTOMATIC** с подразбираща се стойност 100. Също по подразбиране **MAX\_COORDAGENTS** ще бъде настроен на **AUTOMATIC** с подразбираща се стойност 200.

За да се изпращат отчитащи низове от клиентските ви приложения към DB2 Connect сървъра, използвайте специфичните за API средства за задаване на информация за отчитане. Специфичните за API средства работят по-бързо от настройването на променливата на обкръжение **DB2ACCOUNT**.

### **IBM Data Server драйвер за JDBC и SQLJ**

`com.ibm.db2.jcc.DB2BaseDataSource.clientAccountingInformation` свойство

### **DB2 .NET Data Provider**

`DB2Connection.ClientAccountingInformation` свойство

### **CLI/ODBC**

`ClientAcctStr` CLI/ODBC конфигурационна ключова дума

### **Вграден SQL (C, C++, и COBOL)**

`sqlsact` функция

Ако нямате свързан файл за **SQLCODE** преобразуване, можете да увеличите производителността, като използвате **SQLCODE** преобразуването по подразбиране или да го изключите. Подразбиращият се файл за преобразуване е вграден в библиотеката DB2 Connect; персонализиращият файл за преобразуване трябва да бъде прочитан от диска, което засяга производителността.

---

## Фина настройка на хост база данни

Системната производителност ще бъде повлияна от производителността на хост или System i 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. Проверете своята документация на за системна производителност на хост или System i сървър на база данни за повече информация.

Възможно е да успеете да подобрите производителността, като използвате опциите за обвързване за четене без комит (UR) или без комит (NC), където е достъпно, за да избегнете журналирането.

**Забележка:** Когато използвате UR, нежурналираните данни могат само да бъдат четени, без обновяване, а и тогава само, ако създаването на блокове е настроено на ALL.

В зависимост от сървъра на приложения, и грануларността на заключения, която той предлага, нивото на изолация, използвано за заявка или приложение, 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.

---

## Съображения при фината настройка на мрежата

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. В частност, UNIX системите имат по принцип подразбираща се дълбочина на Transmit или Receive опашка от 32. 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 настроен на 500, но опитът показва, че стойността 3992 работи най-добре, ако преместите големи обеми данни, особено за канални свързвания като ESCON или 3172.

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. В token ring или FDDI мрежа тази стойност трябва да бъде 4400 байта, и ако използвате ESCON адаптер с TCP/IP, размерът на сегмент трябва винаги да бъде 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.

Обобщение на методите за настройка на мрежовата производителност се предоставя в Таблица 17.

Таблица 17. Методи за настройка на мрежова производителност

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 за ESCON или друг канален адаптер.
Буфери	RUSIZE	Оптимален размер - 4096.	Настройването на RUSIZE и RQRIOBLK на един размер може да даде най-добрата производителност.
Буфери	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.
TCP настройки	Размери на сегменти	1500 в Ethernet, 4400 в token ring и FDDI.	ESCON адаптерите, използвани за TCP/IP, трябва винаги да са настроени на 4096.
TCP настройки	Размери на пространства за Изпращане/ Получаване	Трябва да е 64К и за двете.	По подразбиране е само 8192 за Windows. Може да бъде зададен в Windows регистъра.

## Конфликт в 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 с производителността на DB2 Connect

Ако DB2 Connect потребителите се сблъскват с дълги периоди на изчакване по време на големи заявки от хост или System i сървъри, следните области следва да бъдат прегледани за възможна причина за проблем с производителността:

1. За заявки, които водят до връщане на големи блокове данни от хоста или System i сървъра (обикновено 32К данни или повече), се уверете, че конфигурационният параметър на мениджъра на базата данни RQRIOBLK е настроен 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. Уверете се, че максималният размер на RU, определен в дефиницията на IBMRDB режим, е установен на подходяща стойност. Препоръчва се размерът да не е по-малък от 4К за свързвания, използващи Token-ring хардуер. За свързвания, използващи Ethernet хардуер, вземете предвид максималния размер на фрейм в Ethernet от 1536 байта, което може да е ограничаващ фактор.

---

## Настройка на DB2 за OS/390 и z/OS

Можете да оптимизирате обработката на неактивни нишки в OS/390 и z/OS. Във B5 ви е позволено да имате до 25,000 едновременно свързани клиента. 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.

---

## Увеличаване на нивата на трансфер на данни на DB2 Connect

Освен създаването на блокове редове за резултатен набор на заявка, DB2 за OS/390 и z/OS може да връща множество такива блокови заявки в отговор на OPEN или FETCH заявка към отдалечен клиент, какъвто е DB2 Connect. Вместо клиентът многократно да подава заявки към DB2 за OS/390 и z/OS сървъра, изисквайки по един блок поредови данни на заявка, клиентът сега може да заяви сървърът да му изпрати известен брой блокови заявки освен тази, която винаги би върнал. Такива добавъчни блокове се наричат допълнителни блокове на заявки.

Тази нова характеристика позволява на клиента да минимизира броя мрежови обиколки, които съставят голям разход за мрежовата производителност.

Намаляването на броя на изпращаните от клиента до сървъра заявки за блокове води до значителен прираст в производителността. Този прираст в производителността се дължи на факта, че превключването между изпращане и получаване е скъпа от гледна точка на производителността операция. DB2 Connect сега може да използва това подобрение, като заявява допълнителни блокове заявки от DB2 за OS/390 и z/OS сървъра по подразбиране.

За да използвате напълно предимствата от връщането на допълнителните блокове (всеки от които може да бъде дълъг до 32Кбайта) за предпочитания мрежов протокол TCP/IP, са активирани разширения за мащабиране на прозорци под RFC-1323 в DB2 Connect. Тази характеристика позволява на TCP/IP динамично да настройва размерите на прозорци за изпращане и приемане, за да поемат ефективно потенциално големите обеми данни, връщани посредством допълнителните блокове.

---

## Допълнителен блок на заявка

Поддръжката на допълнителни блокове на заявки на сървъри с DB2 Universal Database (UDB) за OS/390 и z/OS Версия 7 или по-късна, се конфигурира чрез параметъра EXTRA BLOCKS SRV на DB2 DDF инсталационния панел. Тази поддръжка се конфигурира, като се контролира максималния брой на допълнителни блокове на заявка, които DB2 може да изпрати обратно на изискалия ги клиент. Можете да настроите този параметър на стойност между 0 и 100. Настройването на параметъра на 0 забранява връщането на допълнителни блокове на заявка. Стойността по подразбиране от 100 трябва да бъде използвана винаги, за да се извлече максимум облага от тази характеристика, като изключим някои особености в мрежата, които биха направили тази настройка различна от идеалната.

От страната на клиента, където приложението се обръща към DB2 за z/OS или пряко чрез инсталация на DB2 Connect в същото разположение, или чрез отделна инсталация на DB2 Connect сървър, има различни средства за активиране на съответната DB2 Connect поддръжка на база указател или израз:

- Използването на размера на редови набор на заявка за указател
- Използването на клаузата 'OPTIMIZE for N ROWS' на select изрза, асоцииран с указател
- Използването на клаузата 'FETCH FIRST N ROWS ONLY' на select изрза, асоцииран с указател

DB2 Connect може да активира поддръжката на допълнителни блокове на заявки, като използва различни SQL API:

### Вграден SQL

- Потребителят може да активира поддръжката на допълнителни блокове на заявка, като укаже или клаузата 'OPTIMIZE for N ROWS', или клаузата 'FETCH FIRST N ROWS ONLY', или и двете, на самия израз select.
- С клаузата 'OPTIMIZE for N ROWS', DB2 за OS/390 и z/OS ще опита да направи блок от желания брой редове, за да го върне на DB2 Connect, според низа на DDF инсталационния параметър EXTRA BLOCKS SRV. Приложението може да избере да подаде повече от N реда, тъй като DB2 за z/OS не ограничава общия брой редове, които в крайна сметка могат да бъдат върнати за резултатния набор, до N.
- Клаузата 'FETCH FIRST N ROWS ONLY' работи по сходен начин, освен че резултатният набор е ограничен до N реда от DB2 за OS/390 и z/OS. Извличането отвъд N реда ще доведе до SQL код +100 (край на данните).

### CLI/ODBC

- Потребителят може да извика поддръжка на допълнителни блокове за заявка чрез атрибута SQL\_MAX\_ROWS на израз.
- Клаузата 'FETCH FIRST N ROWS ONLY' се използва като замяна за DB2 UDB за OS/390 и z/OS 7.1 или по-късен сървър.
  - За Версия 7, резултатният набор е ограничен до N реда от DB2 за OS/390 и z/OS. Извличането отвъд N реда ще доведе до SQL\_NO\_DATA\_FOUND.
  - За Версия 8 или по-късна, CLI подsigурява, че само първите N реда се връщат на приложението чрез клиентския Cursor Manager.

**JDBC** Потребителят може да извика поддръжка на допълнителни блокове за заявка чрез метода setMaxRows. Подобно на активирането на CLI/ODBC, DB2 Connect ще постави като етикет клаузата 'OPTIMIZE for N ROWS' за DB2 за OS/390 и z/OS 6.x сървъра. DB2 Connect също ще постави като етикет клаузата 'FETCH FIRST N ROWS ONLY' за DB2 за z/OS 7.1 или по-горен сървър.

---

## Мащабиране на RFC-1323 прозорец

Мащабирането на прозорец се поддържа на всички Windows, Linux, и UNIX платформи, които поддържат RFC-1323 разширенията за TCP/IP. Можете да активирате този компонент на DB2 за Windows, Linux, или UNIX, като използвате DB2 регистърната променлива DB2SORCVBUF. За да включите мащабирането на прозорци, тази регистратурна променлива трябва да бъде настроена на произволна стойност над 64К. Например, на DB2 за Windows, Linux, или UNIX, можете да зададете db2set DB2SORCVBUF=65537.

Максималните размери на буфери за изпращане и получаване зависят от конкретната операционна система. За да се убеди, че конфигурираните размери на буфери са приети, потребителят може да настрои конфигурационния параметър на мениджъра на базата данни DIAGLEVEL на 4 (информационно) и да провери журналния файл за административно оповестяване за съобщения.

За да има ефект мащабирането на прозорци, то трябва да бъде активирано и на двата края на свързването - както на работната станция, така и на хоста, пряко чрез TCP/IP стека на операционната система, или непряко чрез DB2 продукта. В частност, за DB2 за z/OS, мащабирането на прозорци понастоящем може да бъде активирано само чрез операционната система, посредством настройване на TCPRCVBUFRSIZE на произволна стойност над 64К. Ако използвате отдалечен Клиент на IBM сървър на данни за достъпване на хост или System i DB2 база данни чрез работна станция на DB2 Connect сървър, можете да активирате мащабирането на прозорци и на клиента. Под същия токен, можете също да активирате мащабирането на прозорци между отдалечен Клиент на IBM сървър на данни и работна станция на DB2, когато не е въввлечена хост или System i DB2 база данни.

Въпреки че мащабирането на прозорци е проектирано да подобри мрежовата производителност, важно е да се знае, че очакваното подобрене в мрежовата производителност не винаги се материализира. Взаимодействието между фактори като размерът на рамка, използван за ethernet или token ring LAN адаптер, размерът на IP MTU и други настройки на маршрутизатори по комуникационната линия могат да доведат дори до спад в производителността след активирането на мащабиране на прозорци. Следователно, по подразбиране мащабирането на прозорци е деактивирано с размери на буферите за изпращане и получаване от 64К.

Трябва да бъдете подготвени да прецените въздействието на мащабирането на прозорци и да изпълните необходимите донастройвания на системата. За въвеждане в настрояването на мрежата за подобрена мрежова производителност, се отнесете към <http://www.networking.ibm.com/>.

---

## Преобразуване на хост данни

Когато информация се прехвърля между различни среди (като Intel [Windows], IEEE [Linux и UNIX операционни системи], zSeries [VM, VSE, z/OS], System i [OS/400]), може да се наложи преобразуването на числените типове данни (като десетични, целочислени, с плаваща запетая). Това преобразуване може да засегне производителността.

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). Разработчикът на приложение 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 бъде използвано при архивирането на данни на хост или System i 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.

---

## 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

**Високоскоростен 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

Нивото на трансфер на данни се ограничава от най-бавния носител в пътеката към хост или System i сървър на база 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 мрежа в друга, вземете предвид времето за преминаване. Мостовите, маршрутизаторите и шлюзовете ще увеличат необходимото време. Например намаляването на броя на мостовите, 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.





---

## Глава 20. Настройка на производителност на CLI/ODBC приложение

CLI/ODBC е SQL интерфейс за приложно програмиране, който може да се използва от вашите приложения за бази данни. CLI функциите извикват DB2 запомнени процедури, които на свой ред достъпват системните каталожни таблици.

Някои приложения използват ODBC API за събиране на метаданни, които се използват в по-нататъшната обработка. Десетте обръщания за метаданни към API, които могат да бъдат направени, са:

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns

Определени CLI/ODBC приложения, които използват API за метаданни, изброени по-горе, могат да запитват всички приложения в базата данни. Например, SQLTables извикванията изискват метаданни за всички таблици в базата данни. На голяма система, тези заявки могат да доведат до много мрежов трафик, да отнемат значително време, и да консумират значително количество сървърни ресурси.

Някои CLI/ODBC ключови думи за инициализиране могат да се използват за ограничаване на обема данни, който ще бъде върнат от началните API извиквания по време на етапа "събиране на информация" след първото свързване към базата данни. Тези ключови думи могат да се определят от:

1. Ръчно редактиране на файла db2cli.ini.
2. Променяне на ODBC/CLI настройките за базата данни с помощта на Асистента за конфигуриране на клиент (на платформите, които го поддържат).
3. Обновяване на CLI конфигурацията на базата данни с помощта на DBA интерфейса за обработка на команди.

Ключовите думи са:

- DBName
- TableType
- SchemaList
- SysSchema
- GrantorList
- GranteeList



---

## Раздел 5. Отстраняване на проблеми



---

## Глава 21. Отстраняване на проблеми

DB2 Connect обкръжението включва няколко софтуерни, хардуерни и комуникационни продукта. Към отстраняване на проблеми се подхожда най-добре чрез процеса на елиминиране и пресяване на достъпните данни, за да стигнете до заключение (местоположението на грешката).

След като съберете съответната информация и изберете подходящата тема, преминете към посочения раздел.

---

### Събиране на свързана информация

Отстраняването на проблеми включва стесняване на обхвата на проблема и разследване на възможните причини. 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?
- Дали комуникационният срив е между един или повече клиенти и DB2 Connect Server (шлюз); между DB2 Connect шлюз и хост или System i сървър на база данни; или между DB2 Connect Personal Edition и хост или System i сървър на база данни?
- Какво можете да определите от съдържанието на съобщението и токените, върнати в съобщението?
- Дали използването на диагностични средства като db2trc, db2pd, или db2support ще осигури съдействие на този етап?
- Другите машини, които изпълняват подобни задачи, работят ли правилно?
- При отдалечена задача - успешно ли щеше да се изпълни, ако беше локална?

---

### Началното свързване не е успешно

Прегледайте следните въпроси, и се уверете, че стъпките на инсталацията са следвани:

1. *Успешно ли приключи инсталационният процес?*
  - 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. *За UNIX операционни системи, бил ли е създаден 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, била ли е правилно обработена лицензната информация?*
  - За UNIX операционни системи, редактирали ли сте nodelock файла, и въвеждали ли сте осигурената от IBM парола?

4. *Били ли са комуникациите на хоста и System i сървър на базата данни правилно конфигурирани?*
  - Има три конфигурации, които трябва да се разгледат:
    - a. Конфигурацията на хост или System i сървър на база данни идентифицира риквестъра на приложение пред сървъра. Системата за управление на хоста или System i сървъра на базата данни ще имат записи в системния каталог, които ще дефинират риквестъра като разположение, мрежов протокол и защита.
    - b. Конфигурацията на DB2 Connect работна станция дефинира популацията на клиента пред сървъра, и хост или System i сървъра пред клиента.
    - c. Конфигурацията на клиентската работна станция трябва да съдържа името на работната станция, а комуникационният протокол трябва да е дефиниран.
  - Анализът на проблем за неосъществяване на начално свързване включва проверка дали имената на PU (физическа единица) са правилни и пълни, или проверка за TCP/IP свързвания, дали са указани правилни номер на порт и име на хост.
  - Администраторът на хост или System i сървър на база данни и мрежовите администратори имат достъпни средства за диагностициране на проблеми.
5. *Имате ли нивото на права, изисквано от системата за управление на хост или System i сървър на бази данни, за да използвате хост или System i сървър на бази данни?*
  - Разгледайте правата за достъп на потребителя, правилата за квалификаторите на таблиците и очакваните резултати.
6. *Ако опитате да използвате Процесор за обработка на команди (CLP) за подаване на SQL изрази срещу хост или System i сървър на бази данни, успявате ли?*
  - Следвахте ли процедурата за обвързване на CLP към хост или System i сървър на бази данни?

---

## Проблеми, срещани след първоначалното свързване

Следните въпроси се предлагат като отправна точка, за да ви помогнат да стесните възможните причини за проблема.

1. *Има ли специални или необичайни работни обстоятелства?*
  - 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 скоро fix pack? Ако проблемът възниква, когато потребителят опитва да използва характеристика, която не е била използвана (или заредена) на тяхната операционна система, откакто е била инсталирана, определете най-новия fix pack от IBM и го заредете *след* инсталирането на характеристиката.
2. *Тази грешка възниквала ли е и преди?*
  - Има ли документирани разрешения на предни състояния за грешки?
  - Кои са били участниците и могат ли да осигурят наблюдения върху възможната посока за действия?

3. *Изследвали ли сте използването на командите на комуникационен софтуер, които връщат информация за мрежата?*
  - TCP/IP може да има извлечена стойностна информация от използването на TCP/IP команди и помощни програми.
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.
  - SQLSTATE позволяват на програмистите на приложения да тестват за класове на грешки, които са общи за DB2 семейството продукти. В мрежа на разпределена реляционна база данни това поле може да осигури обща основа.
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? Възможно е да е достигнат максималния брой клиенти, които се опитват да се свържат към сървъра. Дали ако друг клиент се изключи от сървъра, клиентът, който преди това не е можел да се свърже, вече може?*
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 оторизацията може да не е била дадена на ниво базата данни или таблица.*
9. *Това първата машина ли е, която се свързва отдалечено към базата данни? В разпределени обкръжения маршрутизаторите или мостовете между мрежи могат да блокират комуникацията между клиента и сървъра. Например, когато използвате TCP/IP, уверете се, че можете да изпълните PING към отдалечения хост.*

---

## Диагностични средства

Когато срещнете проблем, може да използвате следното:

- Всички диагностични данни, включително дъмп файлове, трап файлове, журнали на грешки, файлове на оповестявания и журнали за тревоги, се намират в пътеката, указана чрез конфигурационния параметър за пътека до диагностични данни (**diagpath**) на мениджъра на базата данни:

Ако стойността за този конфигурационен параметър е null, диагностичните данни се записват в една от следните директории или папки:

- За Linux и UNIX обкръжения: INSTHOME/sqllib/db2dump, където *INSTHOME* е собствената директория на потребителския модел.
- За поддържани Windows обкръжения:
  - Ако променливата на обкръжение **DB2INSTPROF** не е настроена, се използва *x:\SQLLIB\DB2INSTANCE*, където *x:\SQLLIB* е препратката към устройство и директория, посочени в регистърната променлива **DB2PATH**, и стойността на **DB2INSTANCE** има името на потребителския модел.

**Забележка:** Не е нужно директорията да е наименувана SQLLIB.

- Ако променливата на обкръжение **DB2INSTPROF** е настроена, се използва *x:\DB2INSTPROF\DB2INSTANCE*, където **DB2INSTPROF** е името на директорията на профил на потребителския модел, и **DB2INSTANCE** е името на потребителския модел (по подразбиране, стойността на **DB2INSTDEF** на Windows 32-битови операционни системи).

- За Windows операционни системи, можете да използвате Event Viewer за преглеждане на журнала за административно оповестяване.
- Достъпните диагностични средства, които могат да бъдат използвани, включват **db2trc**, **db2pd** и **db2support**.
- За Linux и UNIX операционни системи, командата **ps**, която връща информация за състоянието на процес за активните процеси на стандартния изход.
- За UNIX операционни системи, файлът на ядрото, който се създава в текущата директория, когато възникват тежки грешки. Съдържа образ на паметта на терминирания процес и може да се използва, за да се определи коя функция е причинила грешката.



---

## Глава 22. DB2 проследявания в DB2 Connect

Проследяването на действия и операции според случването им във вашето обкръжение може да предостави полезна информация при отстраняване на проблем. Можете да добиете, да изпълните дъмп и да форматирате проследяване, взето от DB2 сървърния продукт на база данни. Помощната програма за проследяване се предоставя като част от DB2 сървърния продукт на база данни.

---

### Получаване на DB2 проследяване посредством db2trc

Командата **db2trc** управлява помощната програма за проследяване, предоставяна с DB2. Помощната програма за проследяване записва информация за операции и форматира тази информация в четим вид.

Имайте предвид, че съществува добавено натоварване, когато се изпълнява проследяване, затова активирането на помощната програма за проследяване може да засегне производителността на системата ви.

В общи линии екипите на DB2 поддръжката и разработката използват DB2 проследяванията за отстраняване на проблеми. Можете да изпълните проследяване за събиране на информация относно проблем, който разследвате, но употребата му е доста ограничена без познаване на изходния код на DB2.

При все това е важно да се знае как правилно да се включи проследяване и как да се дъмпват файловете от проследяване, в случаи, че бъдете помолени да ги предоставите.

**Забележка:** Ще се нуждаете от едно от SYSADM, SYSCTRL или SYSMAINT правата, за да използвате db2trc

За да получите обща идея за достъпните опции, изпълнете командата db2trc без никакви параметри:

```
C:\>db2trc
```

```
Употреба: db2trc (chg|clr|dmp|flw|fmt|inf|off|on) опции
```

За повече информация относно конкретен параметър на командата db2trc, използвайте опцията -u. Например, за да видите повече информация за включването на проследяване, изпълнете следната команда:

```
db2trc on -u
```

Това ще осигури информация за всички допълнителни опции (обозначени като "помощни средства"), които могат да бъдат указани при включване на DB2 проследяване.

При включването на проследяване, най-важната опция е -L. Това задава размера на буфера на паметта, който ще се използва за съхраняване на информацията от проследяването. Размерът на буфера може да бъде указан в байтове или в мегабайтове. (За да укажете мегабайтове, добавете "M" или "m" след стойността). Размерът на буфер за проследяване трябва да е степен на две мегабайта. Ако укажете размер, който не отговаря на това изискване, размерът на буфер ще бъде автоматично закръглен надолу до най-близката степен на две.

Ако буферът е твърде малък, може да бъде изгубена информация. По подразбиране, ако буферът се запълни, се запазва само най-актуалната информация от проследяването. Ако буферът е твърде голям, може да се окаже трудно да се изпрати файлът на екипа по поддръжката на DB2.

Ако се проследява операция, която е относително кратка (като свързване към база данни), обикновено е достатъчен размер от приблизително 8MB:

```
C:\> db2trc on -l 8M
Проследяването е включено
```

Ако обаче проследявате по-голяма операция, или ако едновременно се изпълнява много работа, може да се наложи по-голям размер на буфера на проследяването.

На повечето платформи проследяването може да бъде включено по всяко време и работи, както е описано по-горе. Има обаче някои ситуации, които да се вземат предвид:

1. На системи с множество дялове на базата данни, трябва да изпълните проследяване за всеки физически (противопоставено на логически) дял на базата данни.
2. На HP-UX, Linux и Solaris платформи, ако проследяването бъде изключено след стартирането на потребителския модел, при следващото стартиране на проследяването ще бъде използван много малък буфер, независимо от зададения размер. Например, вчера сте включили проследяване, като сте използвали `db2trc on -l 8m`, получили сте резултата, и после сте изключили проследяването (`db2trc off`). Днес искате да стартирате проследяване с настроен размер на буфера от 32 мегабайта (`db2trc on -l 32m`), без да спирате потребителския модел и да рестартирате. Ще видите, че в този случай проследяването ще получи само малък буфер. За да изпълните ефективно проследяване на тези платформи, включете проследяването с желания от вас размер на буфер преди да стартирате потребителския модел, и “изчистете” буфера според необходимостта впоследствие.

---

## Дъмп на DB2 файл на проследяване

След като помощната програма за проследяване бъде активирана посредством опцията `on`, всяка следваща работа, изпълнявана от потребителския модел, ще бъде проследявана.

Докато проследяването се изпълнява, можете да използвате опцията `clr` за изчистване на буфера на проследяване. Цялата съществуваща информация в буфера на проследяване ще бъде премахната.

```
C:\>db2trc clr
Проследяването е изчистено
```

След като проследяването операция завърши, използвайте опцията `dmp`, следвана от име на файл на проследяването, за да изпълните дъмп на паметта на буфера на диска. Например:

```
C:\>db2trc dmp trace.dmp
Проследяването е записано във файл
```

Помощната програма за проследяване ще продължи да се изпълнява след изпълнението на дъмп на буфера на проследяване на диска. За да изключите проследяването, използвайте опцията `off`:

```
C:\>db2trc off
Проследяването е изключено
```

---

## Форматиране на файл на DB2 проследяване

Дъмп файлът, създаван от командата `db2trc dmp`, е в двоичен формат и не е четим.

За да се уверите, че файл от проследяване може да бъде прочетен, форматирайте двоичния файл от проследяване да показва управлението на потока, и изпратете форматирания изход на `null` устройство. Следният пример показва командата за изпълнение на тази дейност:

```
db2trc flw example.trc nul
```

където `example.trc` е двоичен файл, получен посредством опцията `dmp`.

Изходът за командата ще ви покаже явно дали има проблем при прочитане на файла и дали проследяването е огънато.

На този етап, дъмп файлът може да бъде изпратен на DB2 поддръжката. Те ще го форматира на база нивото на вашата DB2 услуга. Понякога обаче може да бъдете помолени да форматира дъмп файла в ASCII формат, преди да го изпратите. Това се постига чрез опциите `flw` и `fmt`. Трябва да предоставите името на двоичния дъмп файл заедно с името на ASCII файла, който искате да създадете:

```
C:\>db2trc flw trace.dmp trace.flw
C:\Temp>db2trc flw trace.dmp trace.flw
Общ брой записи от проследяване      : 18854
Подрязване на проследяване           : NO
Огъване на проследяване              : NO
Брой форматиран записи от просл.     : 1513 (pid: 2196 tid 2148 node: -1)
Брой форматиран записи от просл.     : 100 (pid: 1568 tid 1304 node: 0)
...
```

```
C:\>db2trc fmt trace.dmp trace.fmt
C:\Temp>db2trc fmt trace.dmp trace.fmt
Подрязване на проследяване           : NO
Огъване на проследяване              : NO
Общ брой записи от проследяване     : 18854
Брой форматиран записи от просл.     : 18854
```

Ако този изход указва, че "Огъване на проследяване" е "YES", тогава това означава, че буферът на проследяване не е бил достатъчно голям да помести цялата събрана по време на проследяването информация. Огънатото проследяване може да е достатъчно, в зависимост от ситуацията. Ако се интересувате от най-актуалната информация (това е по подразбиране запазваната информация, освен ако е посочена опцията `-i`), тогава наличното във файла на проследяването може да е достатъчно. Ако обаче се интересувате какво се е случило в началото на периода на проследяване, или ако се интересувате от всичко случило се, може да се наложи да изпълните отново проследяването с по-голям буфер на проследяване.

Има достъпни опции при форматирането на двоичен файл в четим текстов файл. Например, можете да използвате `db2trc fmt -xml trace.dmp trace.fmt` за преобразуване на двоичните данни и извеждане на резултата в `xml` обработваем формат. Допълнителни опции са показани в подробното описание на командата за проследяване (`db2trc`).

Друго нещо, за което трябва да сте наясно, е, че на Linux и UNIX операционни системи DB2 ще изпълни автоматично дъмп на буфера на проследяване на диска, когато затваря потребителския модел поради тежка грешка. По този начин, ако проследяването е активирано при неправилно приключване на потребителски модел, ще бъде създаден файл в диагностичната директория, и неговото име ще бъде

db2trdmp.###, където ### е номера на дял на базата данни. Това не се случва на Windows платформи. В тези ситуации трябва да изпълните ръчно дъмп на проследяването.

За обобщаване, следното е пример на типична последователност на db2trc команди:

```
db2trc on -l 8M
db2trc clr
<Изпълняване на команди за пресъздаване на проблема>
db2trc dump db2trc.dmp
db2trc off
db2trc flw db2trc.dmp <име-на-файл>.flw
db2trc fmt db2trc.dmp <име-на-файл>.fmt
db2trc fmt -c db2trc.dmp <име-на-файл>.fmtc
```

---

## Глава 23. Файлове от DRDA проследяване

Преди да анализирате DRDA проследяванията, трябва да разберете, че DRDA е отворен стандарт за дефиниране на структури за данни и комуникации. Например, DRDA се състои от набор правила как следва да бъдат организирани данните за предаване, и как следва да бъде изпълнена комуникацията на тази информация. Тези правила са дефинирани в следните справочници:

- DRDA V3 Vol. 1: Distributed Relational Database Architecture
- DRDA V3 Vol. 2: Formatted Data Object Content Architecture
- DRDA V3 Vol. 3: Distributed Data Management Architecture

PDF версии на тези справочници са достъпни на [www.opengroup.org](http://www.opengroup.org).

Помощната програма **db2drdat** записва данните, обменени между DRDA риквестър на приложение (AR) и DB2 DRDA сървър на приложения (AS) (например, между DB2 Connect и хост или Series i сървър на база данни).

---

### Помощна програма за проследяване

Помощната програма **db2drdat** записва данните, обменени между DB2 Connect сървъра (от името на Клиент на IBM сървър на данни) и хост или System i сървър на базата данни.

Като администратор на база данни (или разработчик на приложения), може да сметете за полезно да разберете как работи този поток данни, тъй като това знание може да ви помогне да определите произхода на даден проблем. Допуснете, че сте в следната ситуация: подавате израз `CONNECT TO` база данни за хост или System i сървър на база данни, но командата отказва и вие получавате код за връщане при неуспех. Ако разбирате точно каква информация е подавана на системата за управление на хост или System i сървър на база данни, може да успеете да определите причината за отказа, дори ако информацията от кода на връщане е твърде обща. Много от проблемите се причиняват от прости потребителски грешки.

Изходът от **db2drdat** извежда потоците данни, обменяни между DB2 Connect работната станция и системата за управление на хост или System i сървър на база данни. Данните, изпратени към хост или System i сървър на база данни, се обозначават със `SEND BUFFER`, и данните, получени от хост или System i сървър на база данни, се обозначават с `RECEIVE BUFFER`.

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 *непреобразуваната* стойност, както е върната от хост или System i сървър на база данни. 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, обменян между двете системи, в ASCII или EBCDIC.
- ASCII представяне на колони 2 и 3.
- EBCDIC представяне на колони 2 и 3.

---

## Извеждане на проследяването

Помощната програма db2drdat записва следната информация във *файл-проследяване*:

- -t
  - Тип на DRDA отговор/обект
  - Получен буфер
- -s
  - Тип на DRDA заявка
  - Изпратен буфер
- -c
  - SQLCA
- Информация за TCP/IP грешка
  - Получен код на връщане от функция
  - Тежест
  - Използван протокол
  - Използван API
  - Функция
  - Номер на грешка.

### Забележка:

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.
3. Върнатите полета се различават в зависимост от платформата, на която работи DB2 Connect, дори и при един и същи API.
4. Ако командата db2drdat изпраща извеждането към файл, който вече съществува, старият файл ще бъде изтрит, освен ако правата върху него не позволяват той да бъде изтрит.

---

## Анализ на изходния файл от проследяването

Следната информация се прихваща в db2drdat проследяването:

- 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 хост или System i сървър на база данни
- Система за управление на хост или System i сървър на база данни, с която комуникира DB2 Connect системата.

Първият буфер съдържа Exchange Server Attributes (EXCSAT) и Access RDB (ACCRDB) командите, изпратени на системата за управление на хост или System i сървъра на база данни. Той подава тези команди като резултат от командата на базата данни CONNECT TO. Следващият буфер съдържа отговора, който DB2 Connect е получила от системата за управление на хост или System i 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).

### EXCSAT

Командата EXCSAT съдържа името за работната станция на клиента, указан чрез обекта Server Name (SRVNAM), който е кодов шаблон X'116D', според DDM спецификацията. Командата EXCSAT е намерена в първия буфер. В командата EXCSAT стойностите X'9481A292' (кодирани в CCSID 500) се превеждат до *mask*, след като X'116D' се премахне.

Командата EXCSAT също съдържа EXTNAM (External Name) обекта, който често се поставя в диагностична информация на системата за управление на хост или System i 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 чрез кодовия шаблон X'115E', и в този пример стойността е db2bp, заобиколена с интервали, следвани от 000C50CC. На Linux или UNIX Клиент на IBM сървър на данни, тази стойност може да се корелира с командата ps, която връща информация за състояние на процес за активните процеси към стандартния изход.

### ACCRDB

Командата ACCRDB съдържа RDB\_NAME в обекта RDBNAM, който е кодов шаблон X'2110'. Командата ACCRDB следва командата EXCSAT в първия буфер. В командата ACCRDB стойностите X'E2E3D3C5C3F1' се обръщат в STLEC1, 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'.

Кодовият набор, конфигуриран за DB2 Connect работната станция, е показан чрез разполагането на CCSID обекта CCSIDSBC (CCSID за еднобайтови символи) с кодов шаблон X'119C' в командата ACCRDB. В този пример CCSIDSBC в X'0333', което е 819.

Допълнителните обекти CCSIDDBC (CCSID за двубайтови символи) и CCSIDMBC (CCSID за символи със смесена байтова дължина), с кодови шаблони съответно X'119D' и X'119E', също са представени в командата ACCRDB. В този пример CCSIDDBC е X'04B0', което е 1200, и CCSIDMBC е X'0333', което съответно е 819.

### EXCSATRD и ACCRDBRM

CCSID стойностите също се връщат от хост или System i сървъра на базата данни в Access RDB Reply Message (ACCRDBRM) във втория буфер. Този буфер съдържа EXCSATRD, следвана от ACCRDBRM. Примерният изходен файл съдържа две CCSID стойности за системата на хост или System i сървъра на база данни. Стойностите са 1208 (за двубайтови символи и за символи със смесена байтова дължина) и 1200 (за двубайтови символи).

Ако DB2 Connect не разпознава кодовата страница, връщаща се от хост или System i сървъра на база данни, на потребителя ще бъде върнат SQLCODE -332 с кодовата страница на източник и на приемник. Ако хост или System i сървър на база данни не разпознае кодовия набор, изпратен от DB2 Connect, ще върне VALNSPRM (Стойността на параметъра не се поддържа, с DDM кодов шаблон X'1252'), който се преобразува в SQLCODE -332 за потребителя.

ACCRDBRM съдържа също параметъра PRDID (продуктово специфичен идентификатор, с кодов шаблон X'112E'). Стойността е

X'C4E2D5F0F8F0F1F5', което е DSN08015 в EBCDIC. Според стандартите, DSN е DB2 Universal Database за z/OS и OS/390. Номерът на версия също се указва. ARI е DB2 Server за VSE & VM, SQL е DB2 база данни или DB2 Connect, и QSQ е DB2 за i5/OS.

---

## Примерни изходни файлове от проследяването

Следните фигури показват примерно извеждане, илюстриращо някои DRDA потоци данни, обменяни между DB2 Connect работни станции и хост или System i сървър на база данни. От гледна точка на потребителя е подадена командата CONNECT TO с помощта на процесор за обработка на команди (CLP).

Фигура 13 на страница 145 използва DB2 Connect Enterprise Edition Версия 9.1 и DB2 Universal Database (UDB) за z/OS Версия 8 по TCP/IP свързване.



1 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 0 probe 100  
 bytes 16

Data1 (PD\_TYPE\_UINT,8) unsigned integer:  
 233

2 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.1177)  
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 19532 probe 1177  
 bytes 250

SEND BUFFER(AR):

	EXCSAT RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	00C3D041000100BD 1041007F115E8482	...A.....A...^..	.C}.....".;db
0010	F282974040404040 4040404040404040	...@@@@@@@@@@@@	2bp
0020	4040F0F0F0C3F5F0 C3C3F0F0F0000000	@@.....	000C50CC000...
0030	0000000000000000 0000000000000000	.....	.....
0040	0000000000000000 000000000060F0F0	.....	.....-00
0050	F0F1A2A495404040 4040404040404040	....@@@@@@@@@@	01sun
0060	4040404040404040 4040404040404040	@@@@@@@@@@@@@@	
0070	C4C5C3E5F8404040 F0A2A49540404040	....@@@...@@@	DECV8 0sun
0080	4040404040404040 4000181404140300	@@@@@@@@@.....	.....
0090	0724070008147400 05240F0008144000	.\$...t.\$...@.	.....
00A0	08000E1147D8C4C2 F261C1C9E7F6F400	...G...a.....	.....QDB2/AIX64.
00B0	08116D9481A29200 0C115AE2D8D3F0F9	..m.....Z.....	.._mask...]SQL09
00C0	F0F0F0	...	000

	ACCSEC RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0026D00100020020 106D000611A20003	.&.... .m.....	..}....._s...
0010	00162110E2E3D3C5 C3F1404040404040	..!.....@@@@@	...STLECI
0020	40404040404040	@@@@@	

3 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110546200 probe 100  
 bytes 12

Data1 (PD\_TYPE\_UINT,4) unsigned integer:  
 105

4 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.1178)  
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110549755 probe 1178  
 bytes 122

RECEIVE BUFFER(AR):

	EXCSATRD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0059D04300010053 1443000F115EE5F8	.Y.C...S.C...^..	..}.....;V8
0010	F1C14BE2E3D3C5C3 F100181404140300	..K.....	1A.STLECI.....
0020	0724070007147400 05240F0007144000	.\$...t.\$...@.	.....
0030	0700081147D8C4C2 F20014116DE2E3D3	...G.....m...	.....QDB2..._STL
0040	C5C3F14040404040 4040404040000C11	...@@@@@@@@@...]	EC1 ...
0050	5AC4E2D5F0F8F0F1 F5	Z.....	]DSN08015

	ACCSECRD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0010D0030002000A 14AC000611A20003	.....	..}.....s...

5 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110656806 probe 100  
 bytes 16

Data1 (PD\_TYPE\_UINT,8) unsigned integer:  
 233

Фигура 13. Примерен резултат от трасиране (TCP/IP свързване)

6 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.1177)  
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110659711 probe 1177  
 bytes 250

SEND BUFFER(AR):

	SECCHK RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	003CD04100010036	106E000611A20003	..<.A...6.n.....
0010	00162110E2E3D3C5	C3F1404040404040	..!.....@@@
0020	40404040404000C	11A1D9858799F485	@@@.....
0030	A599000A11A09585	A6A39695	.....vr....newton

	ACCRDB RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	00ADD001000200A7	20010006210F2407	.....!.\$.
0010	00172135C7F9F1C1	F0C4F3C14BD7C1F8	..!5.....K...
0020	F806030221064600	162110E2E3D3C5C3	...!.F.!.....
0030	F140404040404040	4040404040000C11	..@.....1
0040	2EE2D8D3F0F9F0F0	F000D002FD8E3C4	...../...SQL0900....QTD
0050	E2D8D3C1E2C30016	00350006119C0333	.....5.....3
0060	0006119D04B00006	119E0333003C2104	.....3.

7 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 259908001 probe 100  
 bytes 12

Data1 (PD\_TYPE\_UINT,4) unsigned integer:  
 176

8 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.1178)  
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 259911584 probe 1178  
 bytes 193

RECEIVE BUFFER(AR):

	SECCHKRM RPYDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0015D0420001000F	1219000611490000	...B.....I..
0010	000511A400		.....u.

	ACCRDBRM RPYDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	009BD00200020095	2201000611490000	....."....I..
0010	000D002FD8E3C4E2	D8D3F3F7F0000C11	.../.....QTDSQL370...
0020	2EC4E2D5F0F8F0F1	F500160035000611	.....5....DSN08015.....
0030	9C04B80006119E04	B80006119D04B000	.....
0040	0C11A0D5C5E6E3D6	D540400006212524	.....@...!%\$
0050	34001E244E000624	4C00010014244D00	4..\$N..\$L...\$M.
0060	06244FFFFF000A11	E8091E768301BE00	.\$0.....v....
0070	2221030000000005	68B3B8C7F9F1C1F0	"!.....h.....
0080	C4F3C1D7C1F8F840	4040400603022106	.....@...!.D3APA88
0090	46000A11E8091E76	831389	F.....v....Y....c.i

9 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 364420503 probe 100  
 bytes 16

Data1 (PD\_TYPE\_UINT,8) unsigned integer:  
 10

Фигура 14. Примерен резултат от трасиране (TCP/IP свързване) продължение

10 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.1177)  
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 364440751 probe 1177  
 bytes 27

SEND BUFFER(AR):

	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	..... .	..}.....

11 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 475009631 probe 100  
 bytes 12

Data1 (PD\_TYPE\_UINT,4) unsigned integer:  
 54

12 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.1178)  
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 475014579 probe 1178  
 bytes 71

RECEIVE BUFFER(AR):

	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	00162110E2E3D3C5 C3F1404040404040	..!.....@#@#@#@	....STLEC1
0020	4040404040400005 211501	@#@#@@...!..	.....

	SQLCARD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	000BD00300010005 2408FF	.....\$..	..}.....

13 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 721710319 probe 100  
 bytes 16

Data1 (PD\_TYPE\_UINT,8) unsigned integer:  
 126

14 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.1177)  
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 721727276 probe 1177  
 bytes 143

SEND BUFFER(AR):

	EXCSQLIMM RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0053D0510001004D 200A00442113E2E3	..S.Q...M ..D!...	..}....(.....ST
0010	D3C5C3F140404040 4040404040404040	....@#@#@#@#@#@	LEC1
0020	D5E4D3D3C9C44040 4040404040404040	.....@#@#@#@#@	NULLID
0030	4040E2D8D3C3F2C6 F0C1404040404040	@@.....@#@#@#@	SQLC2F0A
0040	4040404041414141 41484C5600CB0005	@#@@AAAAAHLV....	.....<.....
0050	2105F1	!..	..1

	SQLSTT OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	002BD00300010025 2414000000001B64	..+....%\$......d	..}.....
0010	656C657465206672 6F6D206464637375	elete from ddcsu	%......?_.....
0020	73312E6D79746162 6C65FF	s1.mytable.	..._`./.%..

15 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 832901261 probe 100  
 bytes 12

Data1 (PD\_TYPE\_UINT,4) unsigned integer:  
 102

Фигура 15. Примерен резултат от трасиране (TCP/IP свързване) продължение

16 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)  
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 832906528 probe 1178  
 bytes 119

RECEIVE BUFFER(AR):

	SQLCARD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0066D00300010060 240800FFFFFF3434	.f.....`\$. ....44	..}....-.....
0010	3237303444534E58 4F544C2000FFFFFFE	2704DSNXOTL ....	.....+!.<.....
0020	0C00000000000000 00FFFFFFF000000	.....	.....
0030	00000000000572020 2057202020202020	....W W	.....
0040	001053544C454331 2020202020202020	..STLEC1	.....<.....
0050	2020000F44444353 5553312E4D595441	..DDCSUS1.MYTA	.....(...
0060	424C450000FF	BLE...	.....<.....

17 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 833156953 probe 100  
 bytes 16

Data1 (PD\_TYPE\_UINT,8) unsigned integer:  
 10

18 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)  
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 833159843 probe 1177  
 bytes 27

SEND BUFFER(AR):

	RDBRLLBCK RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	000AD00100010004 200F	.....	..}.....

19 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.100)  
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 943302832 probe 100  
 bytes 12

Data1 (PD\_TYPE\_UINT,4) unsigned integer:  
 54

20 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)  
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 943306288 probe 1178  
 bytes 71

RECEIVE BUFFER(AR):

	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	00162110E2E3D3C5 C3F1404040404040	..!.....@#@#@@	....STLEC1
0020	4040404040400005 211502	@#@#@@...!..	.....

	SQLCARD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	000BD00300010005 2408FF	.....\$..	..}.....

Фигура 16. Примерен резултат от трасиране (TCP/IP свързване) продължение

## Последваща буферна информация за DRDA проследявания

За допълнителна информация можете да анализирате следващите изпратени и получени буфери. Следващата заявка съдържа комит. Командата commit инструктира хост или System i сървър на система за управление на бази данни да изпълни комит на текущата единица работа. Четвъртият буфер се получава от хост или System i сървър на система за управление на бази данни като резултат от 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, запис от проследяване 12 съдържа празен SQLCA, указан от DDM кодов шаблон X'2408', следван от X'FF'. Незначещ SQLCA (X'2408FF') показва успех (SQLCODE 0).

Фигура 13 на страница 145 показва пример за буфер на получаване, съдържащ SQLCA на грешка на запис от проследяване 16.



---

## Раздел 6. Съобщения





---

## Глава 24. Общи DB2 Connect проблеми

Тази тема изброява най-честите симптоми на проблеми със свързванията, срещани при използването на DB2 Connect. При всеки случай разполагате с:

- 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.
- Симптом, обикновено във вид на списък с примерни съобщения.
- Решение, което посочва вероятната причина за грешката. В някои случаи, може да бъде предоставено повече от едно предложение за разрешаване.

### SQL0965 или SQL0969

#### Симптом

Съобщенията SQL0965 и SQL0969 могат да се подават с множество различни кодове на връщания от DB2 за i5/OS, DB2 за z/OS, 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 или i5/OS базата данни, не може да бъде преведен. Коригирайте проблема въз основа на кода за грешка и след това отново подайте неуспешната команда.

### 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.

Може би TCP/IP протоколът не е стартиран на DB2 Connect сървъра. Възможно е да е имало успешно свързване на клиент преди това.

Ако `diaglevel = 4`, тогава `db2diag.log` може да съдържа подобен запис, например:

```
2001-05-30-14.09.55.321092 Instance:svtdbm5 Node:000
PID:10296(db2tcpdm) Appid:none
common_communication sqlcctcpconnmgr_child Probe:46
DIA3205E адресът на сокет "30090", конфигуриран в TCP/IP
сервизния файл и
необходим за TCP/IP поддръжката на сървъра се използва от друг
процес.
```

#### Решение

Това предупреждение е симптом, който показва, че DB2 Connect, действайки като сървър за отдалечени клиенти, среща проблеми при боравенето с един или повече комуникационни протоколи на клиент. Тези протоколи могат да бъдат TCP/IP и други, и обикновено съобщението показва, че един от комуникационните протоколи, дефиниран за DB2 Connect, не е конфигуриран правилно.

Често причината може да бъде, че променливата на профила DB2COMM не е дефинирана, или е дефинирана неправилно. В общия случай, проблемът е

резултат от несъответствие между променливата DB2COMM, и имената, дефинирани в конфигурацията на мениджъра на базата данни (например, svcename или pname).

Един възможен сценарий е при наличие на свързване, което е било успешно преди, да се получи съобщението за грешка SQL5043, без да е променена конфигурацията. Ако се използва TCP/IP протокол, това може да се случи, когато отдалечената система неправилно прекрати свързването поради някаква причина. Когато това се случи, свързването може да продължава да изглежда като съществуващо на клиента, и може да е възможно да се възстанови свързването без допълнителна намеса чрез подаване на командите, показани по-долу.

Най-вероятно, един от клиентите, свързващи се към DB2 Connect сървър, продължава да държи TCP/IP порт. На всяка клиентска машина, която е свързана към DB2 Connect сървъра, въведете следните команди:

```
db2 terminate
db2stop
```

## SQL30020

### Симптом

SQL30020N Изпълнението е неуспешно поради грешка в разпределения протокол, която ще засегне успешното изпълнение на следващи команди и SQL изрази.

### Решения

При тази грешка трябва да се обърнете към сервиз. Изпълнете командата db2support, преди да се обадите в сервиза.

## SQL30060

### Симптом

SQL30060N "<идентификатор-за-оторизация>" няма права за изпълнения на операцията "<операция>".

### Решение

При свързване с DB2 за OS/390 и z/OS, таблиците на комуникационната база данни (CDB) не са били правилно обновени.

## SQL30061

### Симптом

Свързване към неправилно разположение на хост или System i сървър на бази данни - не може да бъде намерена целева база данни.

### Решение

Възможно е да е посочено неправилно име на сървър на база данни в записа в DCS директорията. Когато това се случи, към приложението се връща SQLCODE -30061.

Проверете записите на DB2 възел, база данни и DCS директория. 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 z/OS и OS/390 база данни, името, което ще се използва, следва да е същото като използваното в Boot Strap Data Set (BSDS) полето "LOCATION=*име-разположение*", което също се предоставя в DSNL004I съобщението (LOCATION=*разположение*), когато е стартирана Distributed Data Facility (DDF).

Правилните команди за TCP/IP възел са:

```
db2 catalog tcpip node <име_възел> remote <хост_име_или_адрес>  
server <номер_на_порт_или_име_на_услуга>  
db2 catalog dcs database <локално_име> as <реално_име_на_БД>  
db2 catalog database <локално_име> as <псевдоним> at <node_име_възел>  
authentication server
```

След това за да се свържете към базата данни, изпълнявате:

```
db2 connect to <псевдоним> user <име_на_потребител> using <парола>
```

## 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 с DB2 Connect сървъра. Това може да се случи също при свързване от DB2 Connect сървър към хост или System i сървър на база данни.

1. Променливата на профила DB2COMM може да бъде настроена неправилно на DB2 Connect сървър. Проверете това. Например, командата db2set db2comm=tcpip трябва да се появи в sqllib/db2profile при изпълнение на DB2 Enterprise Server Edition на AIX.
2. Възможно е да има несъответствие между спецификациите за името на TCP/IP услуга и номера на порт на Клиент на IBM сървър на данни и на DB2 Connect сървъра. Проверете записите във файловете на TCP/IP услугите и на двете машини.
3. Проверете дали DB2 е стартирана на DB2 Connect сървъра. Въведете стойност 4 за diaglevel на конфигурацията на Мениджъра на базата данни, като използвате командата:

```
db2 update dbm cfg using diaglevel 4
```

След спирането и рестартирането на DB2, вижте във файла db2diag.log, за да проверите дали DB2 TCP/IP комуникациите са били стартирани. Би трябвало да видите резултат, подобен на показания:

```
2001-02-03-12.41.04.861119 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 бъде получено при опит за прекъсване на

свързване от машина, където TCP/IP комуникациите вече са отказали. Отстранете проблема с TCP/IP подсистемата.

На повечето машини начинът да се коригира проблемът, е просто да рестартирате TCP/IP протокола за машината. Понякога може да се наложи рециклиране на цялата машина.

## **SQL30082 RC=24 по време на CONNECT**

### **Симптом**

SQLCODE -30082 Осигурените потребителско име или парола не са правилни.

### **Решение**

Проверете дали е въведена правилната парола в оператора CONNECT. Не е достъпна парола, която да се изпрати на сървъра на базата данни приемник. Трябва да бъде изпратена парола от Клиент на IBM сървър на данни към сървъра приемник на база данни. На определени платформи, например AIX, паролата може да бъде получена само, ако се осигури на CONNECT израза.

---

## Раздел 7. Приложения



---

## Приложение А. Общ преглед на DB2 техническата информация

DB2 техническа информация е достъпна чрез следните средства и методи:

- DB2 Център за информация
  - Теми (Теми за дейност, концепция и справка)
  - Помощ за DB2 средствата
  - Примерни програми
  - Самоучители
- DB2 книги
  - PDF файлове за изтегляне)
  - PDF файлове (от DB2 PDF DVD)
  - отпечатани книги
- Помощ от командния ред
  - Помощ за команди
  - Помощ за съобщения

**Забележка:** Темите на DB2 Център за информация се обновяват по-често от PDF книгите или отпечатаните книги. За да получите най-актуалната информация, инсталирайте обновленията на документацията с обявяването им, или се отнесете към DB2 Центъра за информация на [ibm.com](http://ibm.com).

Можете да достъпвате допълнителна техническа информация за DB2 като технически бележки, информационни бюлетини и IBM Redbooks публикации онлайн на [ibm.com](http://ibm.com). Достъпете сайта на DB2 библиотеката за софтуер за управление на информация на <http://www.ibm.com/software/data/sw-library/>.

### Обратна връзка за документация

Ние ценим вашите отзиви за DB2 документацията. Ако имате предложения как да бъде подобрена DB2 документацията, изпратете email на [db2docs@ca.ibm.com](mailto:db2docs@ca.ibm.com). Екипът за DB2 документация чете всички ваши отзиви, но не може да отговаря пряко. Осигурете конкретни примери, когато е възможно, за да можем да разберем по-добре вашите затруднения. Ако предоставяте отзив за конкретна тема или помощен файл, включете заглавието и URL на темата.

Не използвайте този email адрес за контакт с DB2 поддръжката за клиенти. Ако имате технически проблем с DB2, който документацията ви не може да разреши, се свържете с местния сервизен център на IBM за съдействие.

---

## DB2 техническа библиотека на хартия или в PDF формат

Следните таблици описват DB2 библиотеката, достъпна от IBM Центъра за публикации на [www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order). Ръководствата за DB2 Версия 9.5 в PDF формат на английски език, както и преведените им версии, могат да бъдат изтеглени от [www.ibm.com/support/docview.wss?rs=71&uid=swg2700947](http://www.ibm.com/support/docview.wss?rs=71&uid=swg2700947).

Въпреки че таблиците идентифицират книгите, достъпни в печатен вид, книгите може да не са достъпни във вашата страна или регион.

Номерът на формуляр се увеличава всеки път, когато едно ръководство бъде обновено. Проверете дали четете най-новата версия на ръководствата, както са изброени по-долу.

**Забележка:** DB2 Центърът за информация се обновява по-често от PDF книгите или отпечатаните книги.

Таблица 18. DB2 техническа информация

Име	Номер	Достъпна в печатен вид
<i>Справочник за административни API</i>	SC23-5842-01	Да
<i>Административни рутинни процедури и производни таблици</i>	SC23-5843-01	Не
<i>Call Level Interface Ръководство и справочник, част 1</i>	SC23-5844-01	Да
<i>Call Level Interface Ръководство и справочник, Част 2</i>	SC23-5845-01	Да
<i>Справочник на командите</i>	SC23-5846-01	Да
<i>Ръководство и справочник за помощни програми за преместване на данни</i>	SC23-5847-01	Да
<i>Ръководство и справочник за Data Recovery and High Availability</i>	SC23-5848-01	Да
<i>Ръководство за сървъри на данни, бази данни и обекти на базите данни</i>	SC23-5849-01	Да
<i>Ръководство за защита на база данни</i>	SC23-5850-01	Да
<i>Разработка на ADO.NET и OLE DB приложения</i>	SC23-5851-01	Да
<i>Разработка на приложения с вграден SQL</i>	SC23-5852-01	Да
<i>Разработка на Java приложения</i>	SC23-5853-01	Да
<i>Разработка на Perl и PHP приложения</i>	SC23-5854-01	Не
<i>Разработка на потребителски дефинирани рутинни процедури (SQL и външни)</i>	SC23-5855-01	Да
<i>Как да започнем с разработката на приложения за базата данни</i>	GC23-5856-01	Да
<i>Как да започнем с DB2 инсталиране и администриране на Linux и Windows</i>	GC23-5857-01	Да
<i>Ръководство за интернационализация</i>	SC23-5858-01	Да
<i>Справочник на съобщенията, Част 1</i>	GI11-7855-00	Не
<i>Справочник на съобщенията, Част 2</i>	GI11-7856-00	Не
<i>Ръководство за миграция</i>	GC23-5859-01	Да



Таблица 18. DB2 техническа информация (продължение)

Име	Номер	Достъпна в печатен вид
<i>Ръководство за администриране и ръководство за потребителя на Net Search Extender</i>	SC23-8509-01	Да
<i>Ръководство за разделяне и клъстеризиране</i>	SC23-5860-01	Да
<i>Ръководство за администриране и на потребителя за Query Patroller</i>	SC23-8507-00	Да
<i>Бърз старт за клиенти на IBM сървър на данни</i>	GH26-6176-01	Не
<i>Бърз старт за DB2 свъръри</i>	GC23-5864-01	Да
<i>Ръководство на потребителя и справочник за Spatial Extender и Geodetic Data Management Feature</i>	SC23-8508-01	Да
<i>SQL Справочник, Част 1</i>	SC23-5861-01	Да
<i>SQL Справочник, Част 2</i>	SC23-5862-01	Да
<i>Ръководство и справочник за системния монитор</i>	SC23-5865-01	Да
<i>Ръководство за отстраняване на проблеми</i>	GI11-7857-01	Не
<i>Настройка на производителност на база данни</i>	SC23-5867-01	Да
<i>Visual Explain самоучител</i>	SC23-5868-00	Не
<i>Какво е новото</i>	SA12-6674-01	Да
<i>Ръководство и справочник за Workload Manager</i>	SC23-5870-01	Да
<i>Ръководство за rpgXML</i>	SC23-5871-01	Да
<i>XQuery справочник</i>	SC23-5872-01	Не

Таблица 19. DB2 Connect-специфична техническа информация

Име	Номер	Достъпна в печатен вид
<i>Бърз старт за DB2 Connect Personal Edition</i>	GA12-6672-01	Да
<i>Бърз старт за DB2 Connect свъръри</i>	GA12-6673-01	Да
<i>Ръководство на потребителя за DB2 Connect</i>	SA12-6671-01	Да

Таблица 20. Information Integration техническа информация

Име	Номер	Достъпна в печатен вид
<i>Information Integration: Ръководство за администриране за обединени системи</i>	SC19-1020-01	Да

Таблица 20. Information Integration техническа информация (продължение)

Име	Номер	Достъпна в печатен вид
Information Integration: Справочник на ASNCLP програмата за репликация и публикуване на събития	SC19-1018-02	Да
Information Integration: Ръководство за конфигуриране за обединени източници на данни	SC19-1034-01	Не
Information Integration: Ръководство и справочник за SQL репликация	SC19-1030-01	Да
Information Integration: Въведение в репликация и публикуване на събития	SC19-1028-01	Да

## Поръчване на отпечатани DB2 книги

Ако се нуждаете от отпечатани DB2 книги, можете да ги закупите онлайн в много, но не всички страни или региони. Винаги можете да поръчате отпечатани DB2 книги от местния представител на IBM. Не забравяйте, че някои от книгите в електронен формат на DVD DB2 PDF документация не са достъпни в отпечатан вид. Например, никой от томовете на DB2 Справочник на съобщенията не се предлага като отпечатана книга.

Отпечатаните версии на много от DB2 книгите, достъпни на DVD DB2 PDF Документация могат да бъдат поръчани срещу такса от IBM. В зависимост от това къде подавате поръчката си, може да има възможност да поръчате книги онлайн от IBM Publications Center. Ако онлайн поръчката не е достъпна във вашата страна или регион, винаги можете да поръчате отпечатани DB2 книги от вашия местен IBM представител. Отбележете си, че не всички книги на DVD DB2 PDF Документация са достъпни в печатен вид.

**Забележка:** Най-актуалната и пълна DB2 документация се поддържа в DB2 Център за информация на <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5>.

За да поръчате отпечатани DB2 книги:

- За да проверите дали можете да поръчате отпечатани DB2 книги онлайн във вашата страна или регион, проверете IBM Publications Center на <http://www.ibm.com/shop/publications/order>. Трябва да изберете страна, регион или език, за да достъпите информацията за поръчване на публикации, и после да следвате инструкциите за поръчка от вашето местоположение.
- За да поръчате отпечатани DB2 книги от местния представител на IBM:
  1. Намерете информацията за контакт за местния ви представител от един от следните Web сайтове:
    - IBM указателят за контакти за цял свят на [www.ibm.com/planetwide](http://www.ibm.com/planetwide)
    - Web сайтът IBM Publications на <http://www.ibm.com/shop/publications/order>. Ще трябва да изберете вашите страна, регион или език, за да достъпите собствената страница на подходящите за вашето разположение публикации. От тази страница, проследете връзката "About this site".
  2. Когато се обадите, уточнете, че желаете да поръчате DB2 публикация.

3. Предоставете на вашия представител заглавията и номерата на формуляри на книгите, които искате да поръчате. За заглавия и номера на форми вижте “DB2 техническа библиотека на хартия или в PDF формат” на страница 159.

---

## Извеждане на помощ за SQL състояние от процесор за обработка на команди

DB2 връща SQLSTATE стойност за условия, които могат да са резултат от SQL израз. SQLSTATE помощта обяснява значенията на SQL състоянията и кодовете на класове на SQL състояния.

За да извикате помощ за SQL състояние, отворете процесора за обработка на команди и въведете:

```
? sqlstate или ? клас код
```

където *sqlstate* е валидно петцифрено SQL състояние и *клас код* е първите две цифри на SQL състоянието.

Например, ? 08003 показва помощ за SQL състояние 08003, и ? 08 показва помощ за код на клас 08.

---

## Достъпване на различни версии на DB2 Център за информация

За DB2 Версия 9.5 теми, URL на DB2 Център за информация е <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>

За DB2 Версия 9 теми, URL на DB2 Център за информация е <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>

За теми за DB2 Версия 8, идете на URL на Версия 8 Център за информация на: <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>

---

## Извеждане на теми на предпочитания ви език в DB2 Център за информация

DB2 Център за информация опитва да изведе информацията на предпочитания език, зададен в настройките на браузъра ви. Ако една тема не е превеждана на предпочитания от вас език, DB2 Центърът за информация извежда темата на английски.

- За да извеждате теми на предпочитания от вас език в браузъра Internet Explorer:

1. В Internet Explorer щракнете на **Tools** —> **Internet Options** —> бутон **Languages....** Отваря се прозорецът Language Preferences.
2. Проверете дали предпочитаният от вас език е посочен като първи запис в списъка с езици.
  - За да добавите нов език в списъка, щракнете на бутона **Add...**

**Забележка:** Добавянето на език не гарантира, че компютърът има необходимите шрифтове за извеждане на темите на предпочитания език.

- За да преместите език в началото на списъка, изберете езика и щракайте на бутона **Move Up**, докато езикът застане на първа позиция в списъка с езици.
3. Изчистете кеша на браузъра и презаредете страницата, за да изведете DB2 Център за информация на предпочитания от вас език.

- За да извеждате теми на предпочитания от вас език в браузъра Firefox или Mozilla:
  1. Изберете бутона в раздела **Languages** section на диалога **Tools** → **Options** → **Advanced**. Панелът Languages се извежда в прозореца Preferences.
  2. Проверете дали предпочитаният от вас език е посочен като първи запис в списъка с езици.
    - За да добавите нов език в списъка, щракнете на бутона **Add...**, за да изберете език от прозореца Add Languages.
    - За да преместите език в началото на списъка, изберете езика и щраквайте на бутона **Move Up**, докато езикът застане на първа позиция в списъка с езици.
  3. Изчистете кеша на браузъра и презаредете страницата, за да изведете DB2 Център за информация на предпочитания от вас език.

При някои комбинации от браузър и операционна система може да се наложи също да промените регионалните настройки на операционната си система до избраните от вас код на географско разположение и език.

---

## Обновяване на DB2 Център за информация, инсталиран на компютъра ви или на intranet сървър

Ако сте инсталирали локално DB2 Център за информация, можете да получите и инсталирате обновления на документацията от IBM.

Обновяването на вашия локално инсталиран DB2 Център за информация изисква от вас да:

1. Спрете DB2 Център за информация на вашия компютър, и рестартирайте Центъра за информация в самостоятелен режим. Изпълнението на Център за информация в самостоятелен режим не позволява на други потребители в мрежата ви да достъпват Центъра за информация, и ви позволява да прилагате обновления. Non-Administrative и Non-Root DB2 Центровете за информация винаги се изпълняват в самостоятелен режим. .
2. Използвайте характеристиката Обновяване, за да видите дали има достъпни обновления. Ако има обновления, които бихте искали да инсталирате, можете да използвате характеристиката Обновяване за получаването и инсталирането им

**Забележка:** Ако обкръжението ви изисква инсталирането на обновления на DB2 Център за информация на машина, която не е свързана към internet, трябва да направите огледално копие на сайта за обновления на локална файлова система, като използвате машина, която е свързана към internet и има инсталиран DB2 Център за информация. Ако множество потребители във вашата мрежа ще инсталират обновленията в документацията, можете също да намалите времето, необходимо на отделните индивиди за изпълнение на обновлението, като направите огледално копие на сайта за обновления локално, и създадете прокси за сайта за обновления.

Ако има достъпни пакети с обновления, използвайте характеристиката Обновяване за получаване на пакетите. Характеристиката Обновяване обаче е достъпна само в самостоятелен режим.

3. Спрете самостоятелния Център за информация и рестартирайте DB2 Център за информация на вашия компютър.

**Забележка:** На Windows Vista, посочените по-долу команди трябва да бъдат изпълнявани от администратор. За да стартирате команден ред или графично средство с пълни администраторски права, щракнете с десен бутон на иконата за ускорен достъп и после изберете **Run as administrator**.

За да обновите DB2 Център за информация, инсталиран на компютъра ви или на intranet сървър:

1. Спрете DB2 Център за информация.
  - На Windows, щракнете на **Start** → **Control Panel** → **Administrative Tools** → **Services**. После щракнете с десен бутон на услугата **DB2 Център за информация**, и изберете **Stop**.
  - На Linux, въведете следната команда:  
`/etc/init.d/db2icdv95 stop`
2. Стартирайте Центъра за информация в самостоятелен режим.
  - На Windows:
    - a. Отворете команден прозорец.
    - b. Придвигете се до пътеката, в която е инсталиран Центърът за информация. По подразбиране DB2 Център за информация се инсталира в директорията <Program Files>\IBM\DB2 Information Center\Version 9.5, където <Program Files> представлява разположението на директорията Program Files.
    - c. Придвигете се от инсталационната директория до директорията doc\bin.
    - d. Изпълнете файла help\_start.bat:  
`help_start.bat`
  - На Linux:
    - a. Придвигете се до пътеката, в която е инсталиран Центърът за информация. По подразбиране DB2 Център за информация се инсталира в директорията /opt/ibm/db2ic/V9.5.
    - b. Придвигете се от инсталационната директория до директорията doc/bin.
    - c. Изпълнете скрипта help\_start:  
`help_start`

Web браузърът по подразбиране се стартира, за да изведе самостоятелния Център за информация.
3. Щракнете на бутона **Обновяване** (🔄). В десния панел на Център за информация щракнете на **Намиране на обновления**. Извежда се списък на обновленията за съществуващата документация.
4. За да задействате инсталационния процес, проверете изборите, които искате да инсталирате, после щракнете на **Инсталиране на обновленията**.
5. След като инсталационният процес приключи, щракнете на **Завършване**.
6. Спрете самостоятелния Център за информация:
  - На Windows се придвигете до директорията doc\bin в инсталационната директория и изпълнете файла help\_end.bat:  
`help_end.bat`  
**Забележка:** Файлът с последователност от команди help\_end съдържа изискваните команди за безопасно прекратяване на процесите, които са били стартирани с файла с последователност от команди help\_start. Не използвайте Ctrl-C или друг метод за прекъсване на help\_start.bat.
  - На Linux се придвигете до директорията doc/bin в инсталационната директория, и изпълнете скрипта help\_end:  
`help_end`  
**Забележка:** Скриптът help\_end съдържа изискваните команди за безопасно прекратяване на процесите, които са били стартирани с скрипта help\_start. Не използвайте друг метод за прекъсване на скрипта help\_start.
7. Рестартирайте DB2 Център за информация.

- На Windows, щракнете на **Start** → **Control Panel** → **Administrative Tools** → **Services**. После щракнете с десен бутон на услугата **DB2 Център за информация**, и изберете **Start**.
- На Linux, въведете следната команда:  

```
/etc/init.d/db2icdv95 start
```

Обновеният DB2 Център за информация извежда новите и обновените теми.

---

## DB2 самоучители

DB2 самоучителите ви помагат да научите повече за различните аспекти на DB2 продуктите. Уроците предоставят инструкции стъпка по стъпка.

### Преди да започнете

Можете да преглеждате XHTML версията на самоучителя от Центъра за информация на <http://publib.boulder.ibm.com/infocenter/db2help/>.

Някои уроци използват примерни данни или код. Вижте самоучителя за описание на необходимите условия за специфичните му дейности.

### DB2 самоучители

За да прегледате самоучителя, щракнете на заглавието.

#### “pureXML” in *Ръководство за pureXML*

Настройване на DB2 база данни за съхранение на XML данни и за изпълнение на основни операции със собственото съхранение на XML данни.

#### “Visual Explain” в *Visual Explain самоучител*

Анализирайте, оптимизирайте и настройте SQL изрази за по-добра производителност, като използвате Visual Explain.

---

## DB2 Информация за отстраняване на проблеми

Широко многообразие информация за определяне и разрешаване на проблеми е достъпно, за да ви съдейства при ползването на DB2 продуктите.

### DB2 документация

Информация за отстраняването на проблеми може да бъде намерена в DB2 Ръководство за отстраняване на проблеми или раздела Поддръжка и отстраняване на проблеми на DB2 Център за информация. Там ще намерите информация как да изолирате и идентифицирате проблеми чрез DB2 диагностичните средства и помощни програми, разрешения за някои от често срещаните проблеми, и други съвети как да разрешавате проблеми, които е възможно да срещнете с вашите DB2 продукти.

### Web сайт за DB2 техническа поддръжка

Отнесете се към Web сайта за DB2 техническа поддръжка, ако изпитвате затруднения, и се нуждаете от помощ за откриването на възможни причини и разрешения. Сайтът за техническа поддръжка има връзки към последните DB2 публикации, технически бележки, Оторизирани отчети от анализ на програми (Authorized Program Analysis Reports - APAR), fix packs, и други ресурси. Можете да търсите в тази информационна база за намиране на възможни решения за проблемите си.

Достъпете Web сайта за DB2 техническа поддръжка на <http://www.ibm.com/software/data/db2/udb/support.html>

---

## Срокове и условия

Позволенията за използване на тези публикации се предоставят според следните условия.

**Лична употреба:** Можете да възпроизвеждате тези публикации за лична некомерсиална употреба, при условие, че се запазват всички забележки за собственост. Не можете да разпространявате, показвате или да извличате производни продукти от тези публикации или от части от тях без изричното съгласие на IBM.

**Комерсиална употреба:** Можете да възпроизвеждате, разпространявате и показвате тези публикации единствено в рамките на предприятието ви, при условие, че всички бележки за собственост се запазват. Не можете да извличате производни продукти от тези публикации, или да възпроизвеждате, разпространявате или показвате тези публикации или части от тях извън предприятието ви без изричното съгласие на IBM.

Освен ако е изрично упоменато в това разрешение, не се предоставят други позволени, лицензи или права, независимо дали явни или неявни, върху публикациите, или всякакви други информация, данни, софтуер или други включени в тях предмети на интелектуална собственост.

IBM си запазва правото да оттегли предоставените тук позволени по всяко време, по което прецени, че употребата на тези публикации противоречи на интересите, или, ако се определи от IBM, че горните инструкции не са били следвани подобаващо.

Не можете да изтеглите, експортирате или преекспортирате тази информация, освен в пълно съответствие с всички приложими закони и разпоредби, включително всички закони и разпоредби за износа на САЩ.

IBM НЕ ГАРАНТИРА ЗА СЪДЪРЖАНИЕТО НА ТЕЗИ ПУБЛИКАЦИИ. ПУБЛИКАЦИИТЕ СЕ ПРЕДОСТАВЯТ "КАКТО СА", И БЕЗ ГАРАНЦИИ ОТ ВСЯКАКЪВ ВИД, НЕЗАВИСИМО ДАЛИ ЯВНИ ИЛИ НЕЯВНИ, НО НЕ ОГРАНИЧЕНИ ДО НЕЯВНИТЕ ГАРАНЦИИ ЗА ПРОДАВАЕМОСТ, НЕНАРУШАВАНЕ НА ПРАВА, И ПРИГОДНОСТ ЗА ОПРЕДЕЛЕНА ЦЕЛ.





---

## Приложение В. Забележки

Тази информация е разработена за продукти и услуги, предлагани в САЩ.

Възможно е IBM да не предлага продуктите, услугите или компонентите, разгледани в този документ, в други страни. Консултирайте се с локалния представител на IBM за информация за продуктите и услугите, предлагани във вашата област.

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. Обаче потребителят 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/регион или изпратете писмени запитвания на адрес:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

**Следният 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, ПРЯКА ИЛИ КОСВЕНА, ВКЛЮЧИТЕЛНО, НО НЕ 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. IBM може да направи подобрения и/или промени в продукта(ите) и/или програмата(ите), описани в тази публикация, по всяко време и без предизвестие.

Този документ може да предоставя връзки или споменавания на не-IBM Web сайтове и ресурси. IBM не извършва представяне, не дава гаранции и не поема каквито и да било ангажменти по отношение на всички не-IBM Web сайтове или ресурси на трети страни, които могат да бъдат споменавани в, достъпни от или свързани с този

документ. Връзка към не-IBM Web сайт не означава, че IBM подкрепя съдържанието или употребата на този Web сайт или неговия собственик. В добавка, IBM не е страна по и не носи отговорност за каквито и да било сделки, които може да сключите с трети страни, дори ако сте научили за тях (или сте последвали връзка към тях) от IBM сайт. Съответно, вие приемате и се съгласявате, че IBM не е отговорна за достъпността на такива външни сайтове и ресурси, и не носи отговорност за каквото и да е съдържание, услуги, продукти или други материали, разположени на или достъпни чрез тези сайтове или ресурси. Всеки осигурен от трета страна софтуер е предмет на условията на лиценза, който го придружава.

Когато изпращате информация до IBM, вие предоставяте правото на ползване или разпространение на 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  
8200 Warden Avenue  
Markham, Ontario  
L6G 1C7  
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 този документ 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 тествала 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. Тези 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). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. *въведете годината или годините*. All rights reserved.

### Търговски марки

Следните термини са търговски марки или регистрирани търговски марки на International Business Machines Corporation в Съединените щати, други страни, или и двете.

pureXML	Distributed Relational Database Architecture
Informix	ESCON
DB2	AIX
Candle	VTAM
System z	i5/OS
Parallel Sysplex	Encina
WebSphere	OS/390
DB2 Connect	DB2 Universal Database
Redbooks	z/OS
System i	CICS
IBM	RACF
zSeries	Cloudscape
SQL/DS	Lotus
S/390	Rational
HACMP	DRDA
MVS	OS/400
Approach	Domino
ibm.com	NetView
iSeries	

Следните термини са търговски марки или регистрирани търговски марки на други компании

- Linux е търговска марка на Линус Торвалдс в Съединените щати, в други страни, или и двете.
- Java и всички базирани на Java търговски марки са търговски марки на Sun Microsystems, Inc. Съединените щати, в други страни, или и двете.

- UNIX е регистрирана търговска марка на The Open Group в Съединените щати и в други страни.
- Intel е регистрирана търговска марка на Intel Corporation в Съединените щати и в други страни.
- Microsoft и Windows са търговски марки на Microsoft Corporation в Съединените щати, в други страни, или и двете.

Други имена на компании, продукти или услуги могат да бъдат търговски марки или марки на услуги на други.

# Индекс

## A

AS име на база данни приемник 33  
ATOMIC съставен SQL  
не се поддържа в DB2 Connect 101

## B

BINDADD право на достъп  
права за обвързване 51  
bootstrap data set (BSDS) параметри  
Z/OS и OS/390 32

## C

CCSID (coded character set identifier)  
двупосочна поддръжка  
описание 33  
CGI (Common Gateway Interface)  
програмиране  
ограничения 19  
предимства 19  
CHAR тип данни  
описание 126  
character data representation architecture  
(CDRA) 9  
CLI (call level interface)  
доверени свързвания 41  
преглед 129  
приложения  
CURRENTPACKAGESET 48  
CLIENT тип разпознаване  
DB2 Connect съображения 46  
Common Gateway Interface (CGI)  
програмиране  
ограничения 19  
предимства 19  
CPU  
средства за използване 95  
CREATE IN COLLECTION NULLID  
права 51

## D

D (disconnect) параметър 33  
DATA\_ENCRYPT тип разпознаване 46  
DB2 Connect  
Enterprise Edition  
API 21  
JDBC 21  
SQLJ 21  
Web приложения 19  
Web сървъри 22  
XA-съвместими мениджъри на  
транзакции 58  
монитори на обработка на  
транзакции 26  
сървъри за свързваемост 18  
Sysplex поддръжка 115  
System i поддръжка 15

DB2 Connect (*продължение*)  
защита 41  
концентратори на свързвания 112  
поддръжка на хост 15  
подобрения 3  
преглед 3  
пренасяне на данни 59  
продукти 3  
сценарии за сървър на  
свързваемост 15  
DB2 Universal Database за OS/390 и z/OS  
bootstrap набор данни 32  
BSDS параметри 32  
DOMAIN 32  
RESPORT 32  
TCPPORT 32  
защита 48  
опция DYNAMICRULES (BIND) 48  
стойности на директория на възли 32  
DB2 Център за информация  
версии 163  
езици 163  
обновяване 164  
преглеждане на различни езици 163  
DCS (Database Connection Services)  
директория  
вижте Database Connection Services  
(DCS) директория 33  
DDM (Distributed Data Management)  
db2drdat изход 141  
Distributed Relational Database  
Architecture (DRDA) архитектура 9  
Distributed Data Management (DDM)  
db2drdat изход 141  
Distributed Relational Database  
Architecture (DRDA) 9  
Distributed Relational Database Architecture  
(DRDA)  
достъп до данни 9  
преглед 9

## F

FORCE команда  
идентификатор на агент за 72  
Formatted Data Object Content Architecture  
(FDOCA) 9

## I

IBM WebSphere 21  
INTEGER тип данни 126  
INTERRUPT\_ENABLED (disconnect)  
параметър 33  
iSeries  
DRDA 9

## J

Java  
сървъри на приложения  
API 21  
DB2 Connect 21  
JDBC 21  
SQLJ 21

## K

Kerberos протокол за разпознаване  
DB2 Connect 46  
OS/390 и z/OS 47

## L

LIST DCS APPLICATIONS команда 72  
LOCALDATE параметър 33

## M

Microsoft Windows  
приложения 15

## N

NONE типове защита 49  
NOT ATOMIC съставен SQL  
дизайн на приложение 101  
NULLID  
OS/400 51  
NUMDB  
параметър 119

## O

ODBC (open database connectivity)  
интерфейс 15  
оптимизиране на достъп 99  
преглед 129  
приложения  
CURRENTPACKAGESET 48  
OS/390  
DRDA 9

## P

PROGRAM тип защита 49

## R

Relational Connect  
описание на продукт 7  
resource access control facility (RACF)  
защита 49

## S

SAME тип защита 49  
SERVER тип разпознаване 46  
SERVER\_ENCRYPT тип разпознаване 46  
SHOW DETAIL опция на монитор 72  
SOCKS  
    възел  
        задължителни променливи на обкръжение 32  
SQL (Structured Query Language)  
    динамичен 101  
    статичен 101  
SQL изрази  
    DB2 Connect 6  
    извеждане на помощ 163  
SQL\_ATTR\_  
    TRUSTED\_CONTEXT\_PASSWORD  
        употреба 44  
    TRUSTED\_CONTEXT\_USERID  
        употреба 44  
    USE\_TRUSTED\_CONTEXT  
        употреба 42  
SQL/DS  
    DRDA 9  
SQL0965 код на грешка 153  
SQL0969 код на грешка 153  
SQL1338 код на грешка 32, 153  
SQL30020 код на грешка 153  
SQL30060 код на грешка 153  
SQL30061 код на грешка 153  
SQL30073 код на грешка 153  
SQL30081N код на грешка 153  
SQL30082 код на грешка 153  
SQL5043N код на грешка 153  
SQLCA (SQL communication area)  
    буфери на данни 141  
    поле SQLCODE 141  
SQLCODE  
    поле в SQLCA 141  
    преобразуване 63  
SQLDA (SQL descriptor area)  
    размер на заделяне 101  
SQLSTATE  
    кодове на клас 63  
START MVS системна команда 76  
STOP MVS системна команда 76  
Sysplex  
    DB2 Connect поддръжка 115  
    балансиране на натоварване 116  
    изисквания за конфигурация 117  
    използване 116  
    информация за приоритет 116  
    параметър 33  
    съображения за zSeries 115  
    толеранс за грешка 116  
System i  
    DRDA 9

## T

TCP\_KEEPALIVE  
    конфигурационен параметър на операционна система 91  
TCP/IP  
    DOMAIN 32  
    RESPORT 32

TCP/IP (*продължение*)  
    RFC-1323 разширения  
        мащабиране на прозорци 125  
    TCP/PORT 32  
    защита  
        проверена 48  
        сценарии 49  
    имена на услуги 32  
    команда ACCSEC 142  
    команда SECCHK 142  
    конфигурация  
        хост свързвания 17  
    номера на портове 37  
    отдалечени хост имена 32, 37  
    порт за ресинхронизиране 32  
    хост имена 37  
Tuxedo  
    DB2 Connect Enterprise Edition 26

## V

VARCHAR тип данни  
    описание 126  
virtual telecommunications access method (VTAM) 49  
Visual Explain  
    самоучител 166  
VM  
    DRDA  
        и DB2 Connect 9  
VSE  
    DRDA 9

## W

Web приложения  
    DB2 Connect 19  
    запомнени процедури 22  
Web сървъри  
    DB2 Connect 22  
WebSphere  
    преглед 21  
WebSphere Federation Server  
    преглед 7  
WebSphere MQ  
    DB2 Connect 112  
Windows операционни системи  
    Монитор на производителността  
        наблюдение на DB2 приложения 69

## X

X/Open distributed transaction processing (DTP) модел 26  
XA  
    доверени свързвания 41  
    мениджъри на ресурсите 26  
    примери за концентратор 107  
XA мениджъри на транзакции  
    описание 26

## Z

z/OS  
    DRDA 9

## A

автоматично пренасочване на клиент  
    настройка за DB2 Connect 89  
    описание 89  
    откази на свързване 91  
амперсанд (двойно &)  
    файл за SQLCODE преобразуване 63

## Б

бази данни  
    групиране на заявки 101  
    имена  
        DCS директория 33  
        обект RDBNAM 142  
        работен лист за персонализиране на директория 37  
        системна директория на база данни 31  
        настройване 120  
        псевдоними  
            работен лист за персонализиране на директория 37  
            системна директория на база данни 31  
        средства за производителност 95  
        хост бази данни 6  
    бази данни приемници  
        име 33, 37  
    блокинг на данни 101  
    блокове на заявка  
        увеличаване на DB2 Connect равнища на пренос на данни 123  
    буфер за получаване 141

## В

висока достъпност  
    DB2 Connect 85  
време за реакция 95  
възли  
    директории  
        обновяване 31  
        стойности 32  
    име  
        работен лист за персонализиране на директория 37  
        стойности в системна база данни 31  
        стойности на директория на възли 32  
възстановяване на комуникация  
    клиентски приложения 89

## Г

грешки  
    отстраняване на проблеми 133

## Д

данни  
    блокинг 101  
    источници  
        разпределена заявка 12  
    потоци  
        DB2 Connect 9, 95  
    пренос  
        между хостове и работни станции 59  
        производителност 126  
        равнища 95, 126

дати  
    поддръжка на часова зона 33

двуфазов комит  
    активирание 55  
    порт за ресинхронизиране, използван от TCP/IP свързвания 32

диагностична информация  
    преглед 135

динамичен SQL  
    CLI/ODBC конфигурационен параметър  
        CURRENTPACKAGESET 48  
    ефекти на обработка 6  
    производителност  
        техники 101

директории  
    работни листове за персонализиране 37  
    системна база данни  
        обновяване 31  
        стойности 31

директории на базата данни  
    Database Connection Services (DCS) 31  
    възел 31  
    множество записи 38  
    обновяване 31

директория на услуги на свързване към база данни (DCS)  
    обновяване на записи 31  
    стойности 33

доверен контекст  
    DB2 Connect поддръжка 41  
    чрез CLI/ODBC 42

доверени отношения  
    доверени контексти и доверени отношения 41

доверени свързвания 41  
    превключване на потребители чрез CLI/ODBC 44  
    чрез CLI/ODBC 42

документация  
    PDF 159  
    обща информация 159  
    отпечатан 159  
    срокове и условия на употреба 167

допълнителни блокове на заявки  
    EXTRA BLOCKS SRV параметър 124  
    преглед 124

дъмп на проследяване във файл  
    преглед 138

## Е

единици работа (UOW)  
    дефиниция 9  
    отдалечена 11  
    разпределени 55

## Ж

журнал за оценка на политика  
    здравен монитор на DB2 за z/OS 76

журнали  
    Оценка на политики 76

## З

забележки 169

задръствания  
    производителност 95  
    транзакции 95

задръжане  
    системни ресурси 122

запомнени процедури  
    преглед 22

защита  
    DB2 Connect  
        поддръжка 49  
    Kerberos 47  
    TCP/IP 49  
    въведение 41  
    израз GRANT 49  
    израз REVOKE 49  
    насоки 48  
    разширени кодове OS/390 и z/OS 48  
    стойности на директория на възли 32  
    съвети 48  
    типове 37

заявки към база данни  
    групиране за производителност 101

здравен монитор  
    DB2 за z/OS 76

здравен монитор на DB2 за z/OS  
    преглед 76  
    преглеждане на обекти на тревоги 83  
    преглеждане на обобщения за тревоги 81  
    Преглеждане, подаване, запазване на препоръчани действия 79  
    стартиране, спиране, опресняване 77

здравни тревоги 76

зониран десетичен  
    типове данни 126

## И

идентификатор за оторизация 72

идентификатор на клиентски продукт 72

идентификатор на кодова страница 72

Идентификатор на хост приложение 72

изпратен буфер  
    данни от проследяване 141

израз COMMIT  
    статично обвързан 101

израз DESCRIBE  
    производителност с израз PREPARE 101  
    съставни SQL изрази 101

израз EXECUTE IMMEDIATE  
    дизайн на приложение 101

израз GRANT  
    защита 49

израз PREPARE  
    дизайн на приложение 101  
    ефект върху производителност 101

израз REVOKE  
    защита 49

израз ROLLBACK  
    статично обвързан 101

израз SELECT  
    FOR FETCH ONLY на 101  
    в дизайн на приложение 101  
    обновяем 101

израз SET CURRENT PACKAGESET 48

изрази  
    COMMIT 101  
    DESCRIBE 101  
    EXECUTE IMMEDIATE 101  
    FOR FETCH ONLY 101  
    GRANT 49  
    PREPARE 101  
    REVOKE 49  
    ROLLBACK  
        дизайн на приложение 101  
    SELECT 101

изходящ последователен номер  
    команда DB2 LIST DCS APPLICATIONS 72

име на символно предназначение 37

чувствителност към размер 32

## К

клауза FOR FETCH ONLY  
    израз SELECT 101

клиентски приложения  
    ID 72  
    възстановяване на комуникация 89

клиентско NNNAME 72

ключова дума CURRENTPACKAGESET  
    CLI/ODBC 48

книги  
    отпечатани  
        поръчване 162

команда ACCRDB 142

команда ACCRDBRM 142

команда ACCSEC 142

команда db2drdat  
    изходен файл 141

команда db2trc  
    дъмп на изход от проследяване 138  
    преглед 137  
    форматиране на изход от проследяване 139

команда EXCSAT 142

команда EXCSATRD 142

Команда GET SNAPSHOT 70

команда ps  
    обект EXTNAM 142  
    преглед 135

команда SECCHK 142



команда за комит  
буфери за извеждане от  
проследяване 142

команда за обмен на атрибути на  
сървър 142

команди  
ACCRDB 142  
ACCRDBRM 142  
ACCSEC 142  
db2drdat 141  
db2trc 137, 139  
EXCSAT 142  
EXCSATRD 142  
GET SNAPSHOT 70  
SECCHK 142  
комит 142

конфигурационен параметър  
MAXDARI 119

конфигурационен параметър за поддръжка  
на кеш на директория  
Настройка на DB2 Connect 119

конфигурационен параметър на мениджър  
на база данни agentpri 119

конфигурационен параметър на мениджър  
на база данни max\_coordagents 105, 107

конфигурационен параметър на мениджър  
на база данни maxagents 119

конфигурационен параметър на мениджър  
на база данни num\_initagents 105, 107

конфигурационен параметър на мениджър  
на база данни num\_poolagents 105, 107

конфигурационни параметри  
agentpri 119  
DIRCACHE 119  
max\_coordagents 105, 107  
MAXDARI 119  
num\_initagents 105, 107  
num\_poolagents 105, 107  
numdb 119  
rqrioblk 119  
TCP\_KEEPLIVE 91

конфигуриране  
съображения, смяна на парола 48  
хост свързвания 15

концентратор на свързвания 105

концентратори на свързвания  
DB2 Connect 112  
диспечер 107  
конфигурационен параметър  
MAX\_COORDAGENTS 107  
конфигурационен параметър  
NUM\_INITAGENTS 107  
конфигурационен параметър  
NUM\_POOLAGENTS 107  
конфигурационни параметри 107  
логически агенти 107  
натоварване 107  
ограничения 107  
поддръжка на XA транзакция 107  
преглед 105  
примери 107  
пулиране 107  
работни агенти 107  
реализация 107  
сравнени с пулиране на  
свързвания 112

## M

мащабиране на прозорци  
RFC-1323 разширения 125

мениджър на точка на синхронизация  
(SPM)  
параметри по подразбиране 58  
сценарии 57

мениджъри на XA транзакции  
концентратори на свързвания 107

многосайтови обновявания  
активиране 55  
мениджър на точка на  
синхронизация 57  
разпределена единица работа  
(DUOW) 55  
тестване 56  
Център за управление 56

монитор  
производителност 69  
свързвания 69

мониторен елемент на име на  
приложение 72

монитори на обработка на транзакции  
OLTP 26  
Tuxedo 26  
многосайтови обновявания 55  
примери 26  
транзакции 26  
характеристики на използване 26

мрежа  
настройване 121  
равнища на пренос на данни 126  
средства за производителност 95

## N

наблюдение  
Windows монитор на  
производителността 69  
свързвания  
DB2 Connect сървър 69

настройване  
DB2 Connect 93  
DB2 за OS/390 и z/OS 123  
параметри  
AGENTPRI 119  
DIRCACHE 119  
MAXAGENTS 119  
MAXDARI 119  
NUMDB 119  
RQRIOBLK 119  
производителност  
база данни 120  
мрежа 121

низове на параметър  
двойни запетаи 33  
запетаи 33

## O

обвързване  
имена на пакети 51  
пакети 51  
помощни програми и приложения 51

обвързване (*продължение*)  
права  
маркери на параметри с  
отместване 51

обединени бази данни  
разпределена заявка 12

обект EXTNAM 142  
обект SRVNAM 142  
обекти на тревога  
преглеждане 83

обновления  
директории на базата данни 31  
обновявания  
DB2 Център за информация 164

обобщения за тревоги  
преглеждане 81

ограничения  
концентратор на свързвания 107

определяне на проблеми  
диагностични средства  
преглед 135  
достъпна информация 166  
самоучители 166  
свързване 133  
след свързване 134

опресняване  
здравен монитор на DB2 за z/OS 77

отдалечена единица работа  
преглед 11  
пример 11  
характеристики 11

откази на свързване  
автоматично пренасочване на  
клиент 91

относно тази книга v  
отстраняване на проблеми  
DB2 Connect 153  
Информация, достъпна по електронен  
път 166  
помощни средства за  
проследяване 137  
DRDA 144, 148  
преглед 133  
производителност 123  
самоучители 166  
свързване 133, 134  
събиране на информация 133

оценка на производителност  
производителност 95

## P

пакети  
System i сървъри на база данни 51  
хост сървъри на база данни 51  
пакетиран десетичен тип данни 126

параметри  
MAXAGENTS 119  
PRDID 142  
SYSPLEX 33  
директории 37  
низове 38  
параметър DIRCACHE 119  
параметър NOMAP  
SQL CODE преобразуване 33  
изключване на SQL преобразуване  
параметри на DCS директория 63



параметър PRDID 142  
 параметър RQRIOBLK  
   настройване 119  
 пароли  
   промяна  
     OS/390 и z/OS 48  
 персонализиране  
   директории, работни листове за 37  
 плаваща точка  
   тип данни 126  
 поддръжка на двупосочен CCSID  
   BIDI параметър 33  
 подобрения в издание 3  
 помощ  
   SQL изрази 163  
   конфигуриране на език 163  
 помощна програма ddestrc 142  
 помощна програма export  
   пренос на данни между хостове и  
   работни станции 59  
 помощна програма import  
   пренос на данни между хост и работна  
   станция 59  
 помощна програма за администриране  
   DB2 Connect 7  
 помощна програма за проследяване  
   (db2drdat) 141  
 помощна програма за състояние на  
   обработка 135, 142  
 помощни програми  
   db2drdat 141  
   ddcspkgn 51  
   ps (състояние на процес) 135, 142  
   администриране, DB2 Connect 7  
   обвързване 51  
   проследяване 141  
   системен монитор за базата данни 7  
   състояние на обработка 142  
 помощник за Конфигуриране на  
   многосайтово обновяване 56  
 помощници  
   Многосайтово обновяване 56  
   помощно средство за проследяване  
     DB2 проследявания 137, 138, 139  
     DRDA проследявания 144, 148  
     преглед 137  
   поръчване на DB2 книги 162  
   последователен номер на клиент 72  
 права  
   обвързване 51  
 предикати  
   производителност на логика 101  
 пренасяне на данни  
   използване на DB2 Connect 59  
 преобразуване  
   SQLCODE 63  
   параметър NOMAP 63  
 преобразуване на SQLCODE  
   параметър NOMAP 63  
   пригаждане 63  
 преобразуване на данни  
   хост 126  
 преобразувания  
   хост данни 126  
 препоръчани действия  
   преглеждане, подаване, запазване 79

приложения  
   Web  
     използване на DB2 Connect 19  
     запомнени процедури 101  
     обвързване 51  
     производителност 101  
     създаване на дизайн 101  
     съставен SQL 101  
 примери  
   XA концентратори 107  
   концентратори на свързвания 107  
 производителност  
   DB2 Connect  
     настройване 93  
     отстраняване на проблеми 123  
     преглед 95  
     увеличаване на равнища на  
     пренос 123  
   ODBC достъп 99  
   OS/390 123  
   z/OS 123  
   Въздействие върху Процесор за  
     обработка на команди (CLP) 101  
   концентратор на свързвания 112  
   мрежа  
     хардуер 126  
     приложения 101  
     пулиране на свързвания 112  
     системни ресурси 122  
 пропускаемост  
   транзакции 95  
 проследявания  
   DRDA  
     интерпретиране 141  
     буферна информация за DRDA  
     проследявания 148  
     данни между DB2 Connect и  
     сървъра 141  
     изходен файл 141, 142  
     примери за изходни файлове 144  
   процесор за обработка на команди (CLP)  
     SQL изрази 7  
     производителност 101  
   псевдоним на клиентска база данни 72  
   пулиране на свързвания 105  
   преглед 105  
   сравнено с концентратор на  
   свързвания 112

**Р**  
 работни листове  
   персонализиране на директория 37  
 размер на блок 119  
 размер на блок за страниране 119  
 разпознаване 37  
   валидиране 46  
   преглед 46  
   типове  
     CLIENT 46, 48  
     DATA\_ENCRYPT 46  
     KERBEROS 46  
     SERVER 46  
     SERVER\_ENCRYPT 46  
   по подразбиране 46  
 разпределена единица работа  
   двуфазов комит 55

разпределена единица работа  
   (продължение)  
   многосайтово обновявания 55  
   поддържани сървъри 55  
   преглед 9  
 разпределени заявки  
   DB2 Connect поддръжка 12  
   компенсация 12  
   обединени бази данни 12  
   преглед 12  
   прозрачност на разположение 12  
 разработка на приложения 101  
   DB2 AD клиент 15  
   ODBC 15  
 риквестъри на приложения (AR)  
   DRDA дефиниция 9  
   параметри 37

**С**  
 самоучители  
   Visual Explain 166  
   определяне на проблеми 166  
   отстраняване на проблеми 166  
 свързваемост  
   сървъри, DB2 Connect Enterprise  
   Edition 18  
 свързвания  
   DB2 Connect Enterprise Edition 18  
   концентратори, виж концентратори на  
   свързвания 107  
   повторно установяване  
     DB2 Connect Enterprise Edition 18  
     пряко към хост 15  
   пряко към хост 15  
   пряко към хост или System i данни 17  
   пулиране  
     концентратори на свързвания 107  
     преглед 105  
     предимства 107  
   символни типове данни 126  
   система за поддържане на решения  
   (DSS) 141  
   системен монитор за базата данни  
   отдалечени клиенти 69  
   преглед 7  
   системна директория на база данни  
   обновяване 31  
   стойности 31  
   системни ресурси  
   задържане 122  
   системно състояние  
   Команда GET SNAPSHOT 70  
 спиране  
   здравен монитор на DB2 за z/OS 77  
   списък за обвързване 51  
   списък на кеширани адреси 116  
 справки  
   дефиниране на множество записи в  
   база данни 38  
 средства  
   CPU употреба 95  
   производителност 95  
   употреба на памет 95  
 средства за употреба на паметта 95  
 срокове и условия  
   употреба на публикации 167

- стартване
  - здравен монитор на DB2 за z/OS 77
- статичен SQL
  - ефекти на обработка 6
  - производителност 101
- стойност за разпознаване 31
- Стойности на DCS директорията 33
- сценарии
  - TCP/IP защита 49
- съобщение за отговор при край на единица
  - работа (ENDUOWRM) 142
- съобщения за грешки
  - DB2 Connect 153
- сървъри
  - приложение
    - DB2 Connect EE 23
- сървъри на приложения (AS)
  - 2-слоен и 3-слоен модели 23
  - DB2 Connect поддръжка 23
  - DRDA дефиниция 9
  - конфигурация 23
  - преглед 23
  - пълни клиенти 23
  - разгръщане 23
- съставен SQL
  - NOT ATOMIC 101

## T

- тестване
  - многосайтови обновявания 56
- типове данни
  - CHAR 126
  - INTEGER
    - преобразуване на данни на хост 126
  - VARCHAR
    - преглед 126
  - зониран десетичен 126
  - пакетиран десетичен 126
  - плаваща точка
    - преобразуване на данни на хост 126
  - преобразуване
    - ефект върху производителност 126
  - символ 126
- токени
  - SQLCODE 63
- транзакции
  - DB2 Connect Enterprise Edition 26
  - XA разпределени приложения 58
  - двуфазов комит 9
  - единица работа (UOW) 9
  - многосайтови обновявания 9, 55
  - монитори на обработка на транзакции 26
  - поддръжка 58
  - пропускателна способност 95
  - разпределени
    - поддържани сървъри 55

## У

- управление на свързвания 105

## Ф

- файл dcs1ari.map 63
- файл dcs1dsn.map 63
- файл dcs1qsq.map 63
- файл ddc400.lst 51
- файл ddcsmvs.lst 51
- файл ddcsvm.lst 51
- файл ddcsvse.lst 51
- файл за SQLCODE преобразуване 63
- файлове на ядро
  - определяне на проблеми 135

## X

- хардуер
  - мрежова производителност 126
- хост бази данни
  - достъпване посредством DB2 Connect Personal Edition 15
- свързваемост
  - балансиране на натоварване 87
  - висока достъпност 87

## Ц

- Център за управление
  - многосайтови обновявания 56

## Ч

- часови зони
  - преглед 33





Отпечатано в САЩ

SA12-6671-01



Spine information:

DB2 Connect Версия 9.5

Ръководство на потребителя за DB2 Connect

