







**注意**

使用此信息及其支持的产品前，请先阅读第 101 页的附录 B、『声明』下的常规信息。

**修订版声明**

此文档包含 IBM 的所有权信息。它在许可协议中提供，且受版权法的保护。本出版物中包含的信息不包括对任何产品的保证，且提供的任何语句都不需要如此解释。

您可在线或通过当地的 IBM 代表处订购 IBM 出版物。

- 要在线订购出版物，请转至 IBM 出版物中心，网址为：[www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)
- 要查找当地的 IBM 代表处，请转至 IBM 全球联系人目录，网址为：[www.ibm.com/planetwide](http://www.ibm.com/planetwide)

要从美国或加拿大的 DB2 市场和销售部订购 DB2 出版物，请致电 1-800-IBM-4YOU (426-4968)。

当您向 IBM 发送信息时，即同意授予 IBM 独一无二的权力以它认为适当且不会对您造成任何影响的方式使用或分发该信息。

© Copyright International Business Machines Corporation 1993, 2007. All rights reserved.

# 目录

关于本书	v
<b>第 1 章 了解有关故障诊断的更多信息</b>	<b>1</b>
故障诊断简介	1
关于首次出现数据捕获	3
根据常见中断问题收集诊断信息	3
配置诊断信息的自动收集	3
作为 FODC 的一部分收集的数据及其放置	4
自动 FODC 数据生成	4
首次出现数据捕获信息	5
DB2 Query Patroller 和首次出现数据捕获 (FODC)	6
使用首次出现数据捕获 (FODC) 的监视和审计工具	7
使用首次出现数据捕获 (FODC) 的图形工具	8
关于管理通知日志文件	9
管理通知日志	9
设置管理通知日志文件的错误捕获级别	10
解释管理通知日志文件条目	10
关于 DB2 诊断日志 (db2diag.log) 文件	11
设置诊断日志文件错误捕获级别	11
解释 db2diag.log 文件的信息记录	11
解释诊断日志文件条目	12
db2cos (调出脚本) 输出文件	15
转储文件	16
陷阱文件	17
格式化陷阱文件 (Windows)	17
特定于平台的错误日志信息	17
系统核心文件 (Linux 和 UNIX)	18
访问系统核心文件信息 (Linux 和 UNIX)	18
访问事件日志 (Windows)	19
导出事件日志 (Windows)	20
访问 Dr. Watson 日志文件 (Windows)	20
组合 DB2 数据库和操作系统诊断	20
<b>第 2 章 DB2 故障诊断</b>	<b>25</b>
收集 DB2 数据	25
收集关于安装问题的数据	26
收集关于数据移动问题的数据	26
收集关于 DAS 和实例管理问题的数据	27
分析 DB2 数据	27
分析安装问题的数据	28
将数据提交给 IBM 软件支持机构	28
当前发行版故障诊断指导	29
故障诊断优化准则和概要文件	29
存储键故障诊断支持	32
未自动创建数据压缩字典	32
诊断全局变量问题	33
诊断工作负载管理	35
诊断脚本	35
诊断数据不一致问题	36
诊断索引与数据不一致问题	36
诊断分区数据库环境	36

<b>第 3 章 DB2 Connect 故障诊断</b>	<b>39</b>
故障诊断	39
收集相关信息	39
初始连接不成功	39
初始连接后遇到的问题	40
诊断工具	41
DB2 Connect 常见问题	41
<b>第 4 章 用于故障诊断的工具</b>	<b>45</b>
了解更多关于内部返回码的内容	45
db2dart 工具概述	46
比较 INSPECT 和 db2dart	47
使用 db2diag 分析 db2diag.log 文件	49
使用 db2greg 显示和改变全局注册表 (UNIX)	50
标识产品的版本和服务级别	51
使用 db2look 模拟数据库	51
列出系统上安装的 DB2 产品 (Linux 和 UNIX)	54
使用 db2pd 进行监视和故障诊断	55
使用 db2support 收集环境信息	63
基本跟踪诊断	65
DB2 跟踪	66
DRDA 跟踪	69
控制中心跟踪	77
JDBC 跟踪	77
CLI 跟踪文件	79
特定于平台的工具	84
诊断工具 (Windows)	84
诊断工具 (Linux 和 UNIX)	84
<b>第 5 章 如何有效地搜索已知问题</b>	<b>87</b>
故障诊断资源	87
<b>第 6 章 获取 DB2 产品修订</b>	<b>89</b>
应用修订包	89
修订包、测试修订和 APAR	91
应用测试修订	91
<b>附录 A. DB2 技术信息概述</b>	<b>93</b>
硬拷贝或 PDF 格式的 DB2 技术库	93
订购印刷版的 DB2 书籍	95
从命令行处理器显示 SQL 状态帮助	96
访问不同版本的 DB2 信息中心	96
在 DB2 信息中心中以您的首选语言显示主题	97
更新安装在您的计算机或内部网服务器上的 DB2 信息中心	97
DB2 教程	99
DB2 故障诊断信息	99
条款和条件	99
<b>附录 B. 声明</b>	<b>101</b>

索引 . . . . . 105

---

## 关于本书

此指南提供了一些信息，使您可以开始着手解决 DB2® 数据库客户机和服务器的问题。它可帮助您执行下列操作：

- 简要地确定问题与错误
- 根据症状解决问题
- 了解可用的诊断工具
- 制定日常操作的故障诊断策略

### 本书的读者用户

此指南适用于 DB2 数据库客户机和服务器的客户、用户、系统管理员、数据库管理员（DBA）、通信专家、应用程序开发者和技术支持代表。要使用本书，您应熟悉以下方面：

- 通信、关系数据库和局域网（LAN）概念
- 硬件和软件要求与选项
- 网络的整体配置
- 在网络上运行的应用程序和其他功能
- 基本的 DB2 数据库管理任务
- 有关适用于您安装的产品《快速入门》指南中所描述的安装和早期任务的信息





---

## 第 1 章 了解有关故障诊断的更多信息

使用 DB2 数据库产品时，您有时可能会遇到问题。此问题可能由数据库管理器以及针对数据库运行的应用程序报告，也可能由用户向您提供数据库“某些地方不太正常”反馈来报告此问题。此处展示的概念和工具将向您介绍对数据库操作中真实的或察觉到的问题进行故障诊断的任务，并帮助您执行该任务。这里强调了在正确时间捕获正确数据的重要性，因此首次出现数据捕获是第一个讨论的工具。将展示数据库管理器用于捕获有关数据库操作的数据的其他日志和文件，同时还提及了操作系统诊断工具。

---

### 故障诊断简介

为了准确地分析问题，第一步要做的就是完整地描述问题。如果没有问题描述，您就不知道从什么地方开始调查造成问题的原因。此步骤包括询问自己如下基本问题：

- 症状是什么？
- 问题是在哪里发生的？
- 问题是在何时发生的？
- 发生问题的条件是什么？
- 问题是否可以再现？

通过回答上述及其他问题，就得到了对大多数问题的准确描述，并且也是找出问题解决方案的最好办法。

#### 症状是什么？

开始描述问题时，最明显的问题是“发生了什么问题？”。这看起来像一个很直观的问题；但是，它可以分为若干其他问题，从而更好地描述该问题。这些问题包括：

- 是什么人或什么工具报告该问题的？
- 错误代码和错误消息是什么？
- 怎么失败的？例如：循环、中止、崩溃、性能下降或结果错误。
- 对企业有什么影响？

#### 问题是在哪里发生的？

确定问题发生的位置并不总是那么容易，但它是解决问题的最重要步骤之一。报告组件和失败组件之间可能存在多层技术。网络、磁盘和驱动程序只是调查问题时要考虑的几个部分。

- 是在特定平台上还是在多个平台上都发生了该问题？
- 是否支持当前环境和配置？
- 应用程序是在数据库服务器本地运行还是在远程服务器上运行？
- 是否涉及网关？
- 数据库是驻留在各个磁盘上，还是驻留在 RAID 磁盘阵列上？

这些类型的问题可帮助您隔离问题层，并且是确定问题来源所必需的。记住，不能只因为某层报告问题而总是断定那就是问题根源所在。

标识发生问题的位置时应了解发生问题的环境。总是应该花一些时间来完整描述问题环境，包括操作系统、操作系统版本、所有相应软件及版本和硬件信息。确认您正在配置受支持的环境中运行，这是因为许多问题会解释为发现若干软件级别不能在一起运行，或者未在一起经过完整测试。

## 问题是在何时发生的？

问题分析中的另一个必需步骤是创建导致故障的事件的详细时间线，对于从前发生的那些案例尤其如此。可以通过回溯工作过程很轻松地完成此任务：从报告错误时开始（尽可能精确，甚至精确到毫秒），然后通过可用日志和信息回溯工作过程。通常您只需要进行观察，直到您在任何诊断日志中发现的第一个值得怀疑的事件，但是，这并不总是那么容易，通常需要您有实践经验。如果同时存在多层技术，每层都有自己的诊断信息，那么要知道停止的时间就特别困难。

- 问题是否仅在日间或夜间的特定时间发生？
- 问题多久发生一次？
- 导致问题的事件的发生顺序是什么？
- 问题是否发生在环境改变（如升级现有软件或硬件或者安装新的软件或硬件）之后？

回答这类问题可帮助您创建事件的详细时间线，并且为您提供用来进行调查的参考框架。

## 发生问题的条件是什么？

要完整地描述问题，知道发生问题时还有什么别的软件在运行是非常重要的。如果问题是在特定环境或特定条件下发生的，那么这可能是找出问题原因的关键线索。

- 问题是否总是在执行同一任务时发生？
- 事件是否要按一定顺序发生，问题才会再现？
- 是否存在其他应用程序同一时间失败？

回答这些类型的问题可帮助您说明发生问题的环境，并且能够将所有从属项关联起来。记住，不能只因为同一时间发生了多个问题就表示它们总是相关的。

## 问题是否可以再现？

从问题描述和调查角度来看，“理想”的问题是可以再现的。对于可再现的问题，您几乎总是可以将它们与提供的一大堆工具或过程配合使用，以帮助进行调查。因此，可再现问题通常比较容易调试和解决。

但是，可再现问题也有缺点：如果该问题对企业有很严重的影响，那么您可能不想让它再现。在这种情况下，最好尽可能在测试或开发环境中再现该问题。

- 能否在测试机器上再现问题？
- 是否多个用户或应用程序都遇到同一类型的问题？
- 能否通过运行单个命令、一组命令、特定应用程序或独立应用程序来再现问题？
- 能否通过从 DB2 命令行输入等价命令/查询来再现问题？

因为在调查时测试或开发环境有更好的灵活性，也更便于控制，所以最好在测试或开发环境中再现容易发生的单个问题。

---

## 关于首次出现数据捕获

首次出现数据捕获 (FODC) 是适用于错误发生时 DB2 数据库产品自动捕获的诊断信息集合的术语。该信息使您不需要重现错误即可获取诊断信息。

与 FODC 有关的主题介绍了诊断信息的重要性以及收集该信息的不同方法。存在多个与数据库工具、实用程序和 DB2 产品相关联的日志，在您对数据库问题进行故障诊断时，这些日志将非常有用。

### 根据常见中断问题收集诊断信息

发生影响实例的中断时，可自动将诊断信息收集到包中。包中的信息也可手动创建。

如果使用 DB2 实例和数据库时发生问题，应收集发生问题时的数据。首次出现数据收集 (FODC) 是一个术语，用于描述 DB2 环境中发生问题时应采取的操作。应使用 db2pdcfg 工具设置 DB2FODC 注册表变量中的选项，以控制中断期间收集的数据。应使用 db2pdcfg -fodc 来更改 DB2FODC 注册表变量选项。这些选项会影响 FODC 情况中有关数据捕获的数据库系统行为。

#### 自动收集诊断信息

数据库管理器会调用 db2fodc 命令以自动执行首次出现数据捕获 (FODC)。

为了使中断与 DB2 诊断日志和其他故障诊断文件相关，会同时将诊断消息写至管理通知日志和 db2diag.log。诊断消息包括 FODC 目录名和创建 FODC 目录时的时间戳记。FODC 包描述文件在新的 FODC 目录中。

#### 诊断信息的手动收集

首次出现数据收集命令 (db2fodc) 用于收集有关可能中止的信息或出现严重性能问题时的信息。运行 db2fodc 命令后，会在当前诊断路径下自动创建新目录 FODC\_hang\_<timestamp>。会运行 db2cos\_hang 脚本。此脚本控制将收集并放在 FODC 子目录中的数据集合。FODC 子目录是否存在取决于 db2fodc 命令的运行方式或 DB2 注册表变量的配置。

### 配置诊断信息的自动收集

必须先指示数据库管理器要采取的操作，数据库管理器才能自动执行操作。

系统会设置一些标志，来指示数据库操作期间遇到错误或警告时数据库管理器应采取的操作。要采取的操作包括：

- 在 db2diag.log 中生成堆栈跟踪。(缺省)
- 运行标注脚本 db2cos。(缺省)
- 停止跟踪 (db2trc) 命令。

#### 更改首次出现数据捕获 (FODC) 选项

使用“配置 DB2 数据库的问题确定行为” (db2pdcfg) 命令来更改首次出现数据捕获 (FODC) 选项。FODC 选项是使用 db2pdcfg 工具在 DB2FODC 注册表变量中设置的。这些选项会影响 FODC 情况中有关数据捕获的数据库系统行为。

## 作为 FODC 的一部分收集的数据及其放置

根据实例内的中断类型，首次出现数据捕获（FODC）会导致创建子目录及收集的特定内容。将创建一系列子目录及文件和日志的集合。

将在 FODC 目录下创建下列一个或多个子目录：

- DB2CONFIG，包含 DB2 配置输出和文件
- DB2PD，包含 db2pd 输出或输出文件
- DB2SNAPS，包含 DB2 快照
- DB2TRACE，包含 DB2 跟踪
- OSCONFIG，包含操作系统配置文件
- OSSNAPS，包含操作系统监视器信息
- OSTRACE，包含操作系统跟踪

根据 DB2FODC 的配置或 db2fodc 的运行方式，这些目录并非一直存在。

根据中断类型，可在 FODC 目录和子目录中找到以下内容：

- 陷阱文件
- 中断时进行数据捕获期间生成的和不同组件完成的所有不同二进制转储文件和纯文本转储文件
- db2evlog 的事件日志文件
- 中断时启动了跟踪的情况下生成的 DB2 跟踪转储
- 包含核心文件的目录
- DB2FODC 日志文件：
  - 只有一个“日志”文件用于手动 FODC：db2fodc\_hang.log（用于中止）或 db2fodc\_badpage.log（用于错误页面）
- 数据毁坏的相关信息
  - 进程信息：ps（在 UNIX 上）和 db2pd -edus 输出
  - 当前由 db2support 收集的其他信息（可选）：
    - errpt -a output（在 AIX 上）
    - UNIX 平台上的系统日志。例如，/var/adm/messages（用于 SunOS）和 /var/adm/syslog.log（在 HP/UX 上）。只要能够收集这些文件，就会完成此操作（在 Linux 上，必须有 root 用户访问权才能复制 syslog 文件）。

## 自动 FODC 数据生成

出现中断并自动启用首次出现数据捕获（FODC）后，会根据症状收集数据。收集的数据特定于诊断该中断所需的信息。

用于标记中断原因的一条或多条消息，包括定义为“关键”的消息。

陷阱文件包含如下信息：

- 可用虚拟内存量
- 发生陷阱时与产品的配置参数及注册表变量相关的值
- 发生陷阱时 DB2 产品使用的估计内存量
- 提供中断上下文的信息

原始堆栈转储可能包括在 ASCII 陷阱文件中。

特定于数据库管理器内的组件的转储文件存储在相应 FODC 包目录中。

## 首次出现数据捕获信息

首次出现数据捕获 (FODC) 是用来捕获有关 DB2 实例的基于场景的数据的过程。DB2 用户可根据特定症状手动调用 FODC, 也可在检测到预定义场景或症状时自动调用 FODC。此信息减少了再现错误以获取诊断信息的需要。

可在下列文件中找到 FODC 信息:

### 管理通知日志 (“instance\_name.nfy”)

- 操作系统: 所有操作系统
- 缺省位置:
  - Linux® 和 UNIX®: 位于 *diagpath* 数据库管理器配置参数所指定的目录中。
  - Windows®: 使用事件查看器工具 (“开始”>“控制面板”>“管理工具”>“事件查看器”)
- 创建实例时自动创建。
- 发生重大事件时, DB2 将信息写入管理通知日志。该信息供数据库和系统管理员使用。记录在此文件中的消息类型由 *notifylevel* 配置参数确定。

### DB2 诊断日志 (“db2diag.log”)

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所标识的目录中。
- 创建实例时自动创建。
- 此文本文件包含关于实例遇到的错误和警告的诊断信息。此信息用于故障诊断,且旨在供 IBM® 软件支持机构使用。记录在此文件中的消息类型由 *diaglevel* 数据库管理器配置参数确定。

### DB2 管理服务器 (DAS) 诊断日志 (“db2dasdiag.log”)

- 操作系统: 所有操作系统
- 缺省位置:
  - Linux 和 UNIX: 位于 DASHOME/das/dump 中, 其中 DASHOME 是 DAS 所有者的主目录
  - Windows: 位于 DAS 主目录的“dump”文件夹中。例如, C:\Program Files\IBM\SQLLIB\DB2DAS00\dump
- 创建 DAS 时自动创建。
- 此文本文件包含关于 DAS 遇到的错误和警告的诊断信息。

### DB2 事件日志 (“db2eventlog.xxx”, 其中 xxx 是数据库分区号)

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所指定的目录中。
- 创建实例时自动创建。
- DB2 事件日志文件是数据库管理器中发生的基础结构级事件的循环日志。该文件大小固定, 并且充当在实例运行时记录的特定事件的循环缓冲区。每次

停止实例时，就会替换先前的事件日志，而不是追加。如果实例捕获，那么还会生成 `db2eventlog.XXX.crash` 文件。这些文件旨在供 IBM 软件支持机构使用。

### DB2 调出脚本 (db2cos) 输出文件

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所指定的目录中。
- 出现应急启动、陷阱或分段违例时自动创建。还可以按使用 `db2pdcfg` 命令指定的那样在出现特定问题时创建。
- 缺省 `db2cos` 脚本将调用 `db2pd` 命令以打开方式收集信息。根据 `db2cos` 脚本中包含的命令，`db2cos` 输出文件的内容会有所不同。
- `bin/` 目录中提供了 `db2cos` 脚本。在 UNIX 上，此目录是只读的。要对此脚本创建您自己的可修改版本，请将 `db2cos` 脚本复制至 `adm/` 目录。您可自由修改脚本的此版本。如果脚本在 `adm/` 目录中，那么会运行该版本。否则，会运行 `bin/` 目录中的缺省版本。

### 转储文件

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所指定的目录中。
- 出现特定问题情况时自动创建。
- 对于某些错误情况，会将附加信息记录在以失败进程标识命名的二进制文件中。这些文件旨在供 IBM 软件支持机构使用。

### 陷阱文件

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所指定的目录中。
- 实例异常结束时自动创建。还可以使用 `db2pd` 命令创建。
- 如果数据库管理器由于陷阱、分段违例或异常而不能继续处理，那么会生成陷阱文件。

### 核心文件

- 操作系统: Linux 和 UNIX
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所指定的目录中。
- DB2 实例异常终止时由操作系统创建。
- 此外，核心映像将包括大多数或全部 DB2 内存分配，这可能是问题描述所必需的。

为改进陷阱、应急启动、数据毁坏或中止中断情况下（不能使用数据库或实例时）的故障诊断，与特定中断类型相关联的 DB2 诊断文件会发送至新“首次出现数据捕获”（FODC）目录。此目录是在实例诊断路径下创建的。

注: 此 FODC 目录名为 `FODC_<symptom>_<timestamp>`。DB2 诊断文件将在 FODC 包中。只有 AIX 和 Linux 操作系统才能创建此目录和包。

## DB2 Query Patroller 和首次出现数据捕获 (FODC)

如果发现需要调查 DB2 Query Patroller 问题，有一些日志会包含有关可能遇到的问题可探测原因的信息。



### qpdiag.log

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所标识的目录中。
- Query Patroller 系统变为活动状态时自动创建。
- 包含 Query Patroller 的参考记录和诊断记录。此信息用于故障诊断,且旨在供 IBM 软件支持机构使用。

### qpmigrate.log

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所标识的目录中。
- 由 qpmigrate 实用程序自动创建。安装 Query Patroller (如果指定要运行 Query Patroller 的现有数据库) 时可隐式运行 qpmigrate 命令, 或安装之后显式运行该命令。
- 将 Query Patroller 从一个版本迁移到另一个版本时捕获信息和错误消息。它由 Query Patroller 管理员使用。

### qpsetup.log

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所标识的目录中。
- 由 qpsetup 实用程序自动创建。安装 Query Patroller (如果指定要运行 Query Patroller 的现有数据库) 时可隐式运行 qpsetup 命令, 或安装之后显式运行该命令。
- 捕获 qpsetup 实用程序运行时的信息和错误消息。它由 Query Patroller 管理员使用。

### qpuser.log

- 操作系统: 所有操作系统
- 缺省位置: 位于 *diagpath* 数据库管理器配置参数所标识的目录中。
- Query Patroller 系统变为活动状态时自动创建。
- 包含有关 Query Patroller 的参考消息; 例如, 指示 Query Patroller 何时停止和启动。它由 Query Patroller 管理员使用。

## 使用首次出现数据捕获 ( FODC ) 的监视和审计工具

如果发现需要调查监视和审计工具问题, 有一些日志会包含有关可能遇到的问题的可探测原因的信息。

### DB2 审计日志 (“db2audit.log”)

- 操作系统: 所有操作系统
- 缺省位置:
  - Windows: 位于  $\$DB2PATH\instance\_name\security$  目录中
  - Linux 和 UNIX: 在  $\$HOME\sqllib\security$  目录中, 其中  $\$HOME$  是实例所有者的主目录
- 启动 db2audit 工具时创建。
- 包含 DB2 审计工具为一系列预定义的数据库事件生成的审计记录。

### DB2 控制器日志 (“mylog.x”, 其中 x 是控制器在其上运行的数据库分区号)

- 操作系统: 所有操作系统
- 缺省位置:
  - Windows: 位于 `$DB2PATH\instance_name\log` 目录
  - Linux 和 UNIX: 位于 `$HOME\sqllib\log` 目录中, 其中 `$HOME` 是实例所有者的主目录
- 使用控制器实用程序时创建。在 `db2gov` 命令中指定了基本日志文件名。
- 有关控制器守护程序执行的操作 (例如, 强制执行一个应用程序、读取控制器配置文件、启动或结束该实用程序) 的记录信息以及错误和警告。

#### 事件监视器文件 (例如, “00000000.evt”)

- 操作系统: 所有操作系统
- 缺省位置: 创建文件事件监视器时, 所有事件记录都将被写入 `CREATE EVENT MONITOR` 语句指定的目录中。
- 事件发生时由事件监视器生成。
- 包含和事件监视器相关联的事件记录。

## 使用首次出现数据捕获 (FODC) 的图形工具

如果发现需要调查命令编辑器、数据仓库中心或信息目录中心问题, 有一些日志会包含有关可能遇到的问题的可探测原因的信息。

#### 命令编辑器日志

- 操作系统: 所有操作系统
- 缺省位置: 使用 DB2 工具栏的命令编辑器页指定此日志文件的名称和位置。如果未指定路径, 那么此日志将存储在 Windows 上的 `$DB2PATH\sqllib\tools` 目录中或 Linux 和 UNIX 上的 `$HOME/sqllib/tools` 目录中, 其中 `HOME` 是实例所有者的主目录。
- 当选择命令编辑器中的文件的日志命令历史记录并指定文件和位置时创建。
- 包含命令编辑器的命令和语句执行历史记录。

#### 数据仓库中心 IWH2LOGC.log 文件

- 操作系统: 所有操作系统
- 缺省位置: 位于 `VWS_LOGGING` 环境变量指定的目录中。在 Windows 上, 缺省路径为 `$DB2PATH\sqllib\logging` 目录; 在 Linux 和 UNIX 上, 缺省路径为 `$HOME/sqllib/logging` 目录, 其中 `HOME` 是实例所有者的主目录。
- 记录器停止时由数据仓库中心自动创建。
- 包含记录器停止时无法发送的由数据仓库中心和 OLE 服务器编写的消息。可使用数据仓库中心中的“日志查看器”窗口查看此日志。

#### 数据仓库中心 IWH2LOG.log 文件

- 操作系统: 所有操作系统
- 缺省位置: 位于 `VWS_LOGGING` 环境变量指定的目录中。在 Windows 上, 缺省路径为 `$DB2PATH\sqllib\logging` 目录; 在 Linux 和 UNIX 上, 缺省路径为 `$HOME/sqllib/logging` 目录, 其中 `HOME` 是实例所有者的主目录。
- 数据仓库中心无法启动或激活了跟踪时由数据仓库中心自动创建。



- 包含数据仓库中心记录器无法启动并无法写入数据仓库中心日志（IWH2LOGC.log）时的诊断信息。可使用数据仓库中心中的“日志查看器”窗口查看此日志。

#### 数据仓库中心 IWH2SERV.log 文件

- 操作系统: 所有操作系统
- 缺省位置: 位于 VWS\_LOGGING 环境变量指定的目录中。在 Windows 上, 缺省路径为 \$DB2PATH\sqllib\logging 目录; 在 Linux 和 UNIX 上, 缺省路径为 \$HOME/sqlib/logging 目录, 其中 HOME 是实例所有者的主目录。
- 由数据仓库中心服务器跟踪工具自动创建。
- 包含服务器跟踪工具创建的数据仓库中心启动消息和日志消息。可使用数据仓库中心中的“日志查看器”窗口查看此日志。

#### 信息目录中心标记文件 EXPORT 日志

- 操作系统: 所有操作系统
- 缺省位置: 信息目录中心中“导出”工具的选项选项卡指定的已导出标记文件路径和日志文件名
- 由信息目录中心中的“导出”工具生成
- 包含标记文件导出信息, 例如导出过程开始和停止的时间和日期。它还包括在导出操作期间遇到的任何错误消息。

#### 信息目录中心标记文件 IMPORT 日志

- 操作系统: 所有操作系统
- 缺省位置: 信息目录中心中“导入”工具指定的已导入标记文件路径和日志文件名
- 由信息目录中心中的“导入”工具生成
- 包含标记文件导入历史记录信息, 例如导入过程开始和停止的时间和日期。它还包括在导入操作期间遇到的任何错误消息。

---

## 关于管理通知日志文件

管理通知日志文件简介, 它描述了已记录的错误类型的不同级别以及如何建立错误捕获级别。一旦创建日志, 将提供有关如何理解日志记录的指导, 以便您可以了解有关特定数据库问题的更多信息。

### 管理通知日志

DB2 数据库管理器将下列类型的信息写入管理通知日志: DB2 实用程序 (如 REORG 和 BACKUP) 的状态; 客户机应用程序错误、服务类更改、许可证发放活动、日志文件路径和存储问题、监视活动并为活动建立索引, 以及表空间问题。数据库管理员可以使用这些信息来诊断问题、调整数据库或仅监视数据库。

还会使用标准化消息格式将管理通知日志消息记录至 db2diag.log。

通知消息提供了其他信息以补充提供的 SQLCODE。事件的类型和收集的信息的详细级别是由 NOTIFYLEVEL 配置参数确定的。

## 设置管理通知日志文件的错误捕获级别

DB2 记录在管理通知日志中的信息由 NOTIFYLEVEL 设置确定。

- 要检查当前设置，发出 GET DBM CFG 命令。

查找以下变量：

通知级别 (NOTIFYLEVEL) = 3

- 要改变该设置，使用 UPDATE DBM CFG 命令。 例如：

```
DB2 UPDATE DBM CFG USING NOTIFYLEVEL X
```

其中 X 是期望的通知级别。

## 解释管理通知日志文件条目

使用文本编辑器来查看怀疑发生了问题的机器上的管理通知日志文件。记录的最新事件在文件的最后面。通常，每个条目包含下列部分：

- 时间戳记
- 报告错误的位置。应用程序标识允许您匹配在服务器和客户机的日志上与应用程序有关的条目。
- 说明错误的诊断消息（通常以“DIA”或“ADM”开头）。
- 任何可用的支持数据，例如，SQLCA 数据结构和指向任何其他转储文件或陷阱文件的位置的指针。

管理日志与数据库中的所有日志一样会不断增长。根据每个文件中记录的内容不同，某些日志的增长速度会比其他日志的增长速度快。日志过大时，应对其进行备份并擦除。下一次系统需要新日志时会自动生成。

以下示例显示样本日志条目的标题信息，且标识了日志的所有部分。

注：不是所有日志条目都包括所有这些部分。

```
2006-02-15-19.33.37.630000 1 实例: DB2 2 节点: 000 3  
PID: 940(db2syscs.exe) TID: 660 4 Appid: *LOCAL.DB2.020205091435 5  
恢复管理器 6 sqlpresr 7 探测点: 1 8 数据库: SAMPLE 9  
ADM1530E 10 已启动崩溃恢复。 11
```

图注：

1. 消息的时间戳记。
2. 生成该消息的实例的名称。
3. 对于多分区系统，此项为生成该消息的数据库分区。在非分区数据库中，该值为“000”。
4. 进程标识 (PID)，后跟进程名称，再后跟负责生成消息的线程标识 (TID)。
5. 进程正在为其工作的应用程序的标识。在本示例中，生成消息的进程代表标识为 \*LOCAL.DB2.020205091435 的应用程序工作。

此值与 *appl\_id* 监视元素数据相同。有关如何解释此值的详细信息，请参阅 *appl\_id* 监视元素的文档。

要标识关于特定应用程序标识的信息，执行下列其中一项操作：

- 在 DB2 服务器上使用 LIST APPLICATIONS 命令或在 DB2 Connect™ 网关上使用 LIST DCS APPLICATIONS 命令来查看应用程序标识列表。可以根据此列表确定有关遇到错误的客户机的信息，例如其节点名以及其 TCP/IP 地址。
  - 使用 GET SNAPSHOT FOR APPLICATION 命令查看应用程序标识列表。
6. 写入消息的 DB2 组件。对于由使用 db2AdminMsgWrite API 的用户应用程序编写的消息，该组件将读取“用户应用程序”。
  7. 提供消息的函数的名称。此函数在写入消息的 DB2 组件中运行。对于由使用 db2AdminMsgWrite API 的用户应用程序编写的消息，该函数将读取“用户函数”。
  8. 唯一内部标识。此数字允许 DB2 客户支持和开发在报告了消息的 DB2 源代码中找到相应位置。
  9. 发生错误的数据库。
  10. 以十六进制代码指示错误类型和编号的消息（如果有的话）。
  11. 说明记录的事件的消息文本（如果有的话）。

---

## 关于 DB2 诊断日志 (db2diag.log) 文件

包含不同级别日志记录内容的描述的 DB2 诊断日志文件内容将与如何设置错误捕获级别的指示一起介绍。

### 设置诊断日志文件错误捕获级别

DB2 诊断日志是包含 DB2 记录的文本信息的文件。此信息可用于故障诊断，但主要用于 DB2 客户支持。

DB2 在 db2diag.log 中记录的信息由 DIAGLEVEL 设置确定。

- 要检查当前设置，发出 GET DBM CFG 命令。

查找以下变量：

```
诊断错误捕获级别 (DIAGLEVEL) = 3
```

- 要动态更改该值，请使用 UPDATE DBM CFG 命令。

要联机更改数据库管理器配置参数，请使用以下命令：

```
db2 attach to <instance-name>
db2 update dbm cfg using <parameter-name> <value>
db2 detach
```

例如：

```
DB2 UPDATE DBM CFG USING DIAGLEVEL X
```

其中 X 是期望的通知级别。如果要诊断可再现的问题，支持人员建议您在执行故障诊断时使用 DIAGLEVEL 4。

### 解释 db2diag.log 文件的信息记录

db2diag.log 中的第一个消息应该总是参考记录。

参考记录的示例如下所示：

```

2006-02-09-18.07.31.059000-300 I1H917          级别: 事件
PID: 3140          TID: 2864          PROC: db2start.exe
实例: DB2          节点: 000
函数: DB2 UDB, RAS/PD 组件, _pdlogInt, 探测点: 120
开始: 新的诊断日志文件
DATA #1: 构建级别, 124 字节
实例"DB2"使用"32"位和 DB2 代码发行版"SQL09010", 其级别标识为"01010107"。
参考标记为"DB2 v9.1.0.190"、"s060121"、""和修订包"0"。
DATA #2 : 系统信息, 1564 字节
系统: WIN32_NT MYSRVR Service Pack 2 5.1 x86 系列 15, 模型 2, 步骤 4
CPU: 总计: 1 联机: 1 每个套接字的核心数: 1 每个核心的线程化程度: 1
物理内存 (MB): 总计: 1024 空闲: 617 可用: 617
虚拟内存 (MB): 总计: 2462 空闲: 2830
交换内存 (MB): 总计: 1438 空闲: 2213
此记录中的信息仅在创建此文件时有效 (请参阅此记录的时间戳记)

```

参考记录是“db2start”在每个逻辑分区上的输出。这会生成多个参考记录：每个逻辑分区一个参考记录。因为参考记录包含的内存值在每个分区上都不同，所以此信息可能很有用。

## 解释诊断日志文件条目

使用 db2diag.log 分析工具 (db2diag) 来过滤并格式化 db2diag.log 文件。虽然已使用标准化消息格式将管理通知日志消息记录至 db2diag.log, 但还是建议先查看 db2diag.log 以了解数据库所发生的情况。

除了使用 db2diag 之外, 还可使用文本编辑器来查看怀疑发生了问题的机器上的诊断日志文件。记录的最新事件在文件的最后面。

**注:** 管理和诊断日志会不断增加。当它们变得太多时, 对其进行备份然后擦除。下一次系统需要它们时, 会自动生成一组新的文件。

以下示例显示样本日志条目的标题信息, 且标识了日志的所有部分。

**注:** 不是所有日志条目都包括所有这些部分。只有开头的一些字段 (时间戳记至 TID) 和函数才会显示在所有 db2diag.log 记录中。

```

2007-05-18-14.20.46.973000-240 1 I27204F655 2 LEVEL: Info 3
PID : 3228 4 TID : 8796 5 PROC : db2syscs.exe 6
INSTANCE: DB2MPP 7 NODE : 002 8 DB : WIN3DB1 9
APPHDL : 0-51 10 APPID: 9.26.54.62.45837.070518182042 11
AUTHID : UDBADM 12
EDUID : 8796 13 EDUNAME: db2agntp 14 (WIN3DB1) 2
FUNCTION: 15 DB2 UDB, data management, sqldInitDBCBC, probe:4820
DATA #1 : 16 String, 26 bytes
Setting ADC Threshold to:
DATA #2 : unsigned integer, 8 bytes
1048576

```

**图注:**

1. 消息的时间戳记和时区。

**注:** db2diag.log 中的时间戳记包含时区。例如: 2006-02-13-14.34.35.965000-300, 其中“-300”是 UTC (全球标准时间, 以前称为 GMT) 与应用程序服务器中的当地时间 (以分钟为单位) 之间的偏差。因此, -300 表示 UTC - 5 小时, 如 EST (东部标准时间)。

2. 记录标识字段。对于创建 DB2 诊断日志的平台, db2diag.log 的记录标识指定要记录的当前消息的文件位移 (如“27204”)和消息长度 (如“655”)。

3. 与错误消息相关联的诊断级别。例如，参考、警告、错误、严重或事件。
4. 进程标识。
5. 线程标识。
6. 进程名称。
7. 生成该消息的实例的名称。
8. 对于多分区系统，此项为生成该消息的数据库分区。在非分区数据库中，该值为“000”。
9. 数据库名称。
10. 应用程序句柄。此值与 db2pd 输出和锁定转储文件中使用的值相对应。它包括后跟协调程序索引编号并且用破折号分开的协调程序分区号。
11. 进程正在为其工作的应用程序的标识。在本示例中，生成消息的进程代表标识为 9.26.54.62.45837.070518182042 的应用程序工作。

TCP/IP 生成的应用程序标识由三个部分组成：

1. **IP 地址**：它表示为 32-bit 位数字，最大显示为 8 位十六进制字符。
2. **端口号**：显示为 4 位十六进制字符。
3. 此应用程序的实例的**唯一标识**。

**注**：当 IP 地址或端口号的十六进制版本以 0 至 9 开头，那么它们将分别转换为 G 至 P。例如，“0”映射为“G”，“1”映射为“H”，以此类推。IP 地址 AC10150C.NA04.006D07064947 表示为如下所示：IP 地址仍为 AC10150C，它将转换为 172.16.21.12。端口号为 NA04。第一个字符为“N”，它将映射为“7”。因此，端口号的十六进制为 7A04，它将转换为十进制格式 31236。

此值与 *appl\_id* 监视元素数据相同。有关如何解释此值的详细信息，请参阅 *appl\_id* 监视元素的文档。

要标识关于特定应用程序标识的信息，执行下列其中一项操作：

- 在 DB2 服务器上使用 LIST APPLICATIONS 命令或在 DB2 Connect 网关上使用 LIST DCS APPLICATIONS 命令来查看应用程序标识列表。可以根据此列表确定有关遇到错误的客户机的信息，例如其数据库分区名以及其 TCP/IP 地址。
- 使用 GET SNAPSHOT FOR APPLICATION 命令查看应用程序标识列表。
- 使用 db2pd -applications -db <sample> 命令。

12. 授权标识。
13. 引擎可调度单元标识。
14. 引擎可调度单元的名称。
15. 产品名（“DB2”）、组件名（“数据管理”）、写入消息的函数名（“sqlInitDBCB”）以及函数内的探测点（“4820”）。
16. 被调用函数返回的信息。可能会返回多个数据字段。

既然已经看到了样本 db2diag.log 条目，以下是所有可能字段的列表：

```
<timestamp><timezone>          <recordID>          级别: <level> (<source>)
PID: <pid>                      TID: <tid>          PROC: <procName>
实例: <instance>                节点: <node>       数据库: <database>
```

```

APPIDL: <appHandle>    APPID: <appID>
AUTHID  : <authID>
EDUID   : <eduID>      EDUNAME: <engine dispatchable unit name>
函数: <prodName>, <compName>, <funcName>, 探测点: <probeNum>
消息: <messageID> <msgText>
已调用: <prodName>, <compName>, <funcName>  OSERR: <errorName> (<errno>)
返回码: <type>=<retCode> <errorDesc>
ARG #N: <typeTitle>, <typeName>, <size> bytes
... 参数 ...
DATA #N: <typeTitle>, <typeName>, <size> bytes
... 数据 ...

```

尚未在示例中说明的字段包括:

•

<source> 指示已记录错误的源头。(可在样本的第一行结尾找到它。)可能的值包括:

- origin - 导致错误的函数(起始点)已记录消息
- OS - 操作系统已生成错误
- received - 已从另一个进程(客户机/服务器)接收到错误
- sent - 错误已发送至另一进程(客户机/服务器)

•

MESSAGE 包含要记录的消息。它包括:

- <messageID> - 消息号, 如 ECF=0x9000004A 或 DIA8604C
- <msgText> - 错误描述

如果还有被调用(CALLED)字段,那么 <msgText> 是被调用(CALLED)函数返回的错误对记录消息的函数(在“函数”字段中指定)的影响

•

被调用(CALLED) 这是返回错误的函数。它包括:

- <prodName> - 产品名: “OS”、“DB2”、“DB2 Tools”或“DB2 Common”
- <compName> - 组件名称(如果是系统调用则为“-”)
- <funcName> - 被调用函数名

• OSERR 这是被调用(CALLED)的系统调用返回的操作系统错误。(可在 CALLED 所在行的结尾找到它。)它包括:

- <errorName> - 特定于系统的错误名称
- <errno> - 操作系统错误号

• ARG 本节列示返回错误的函数调用的自变量。它包括:

- <N> - 对“被调用”函数的调用中的自变量的位置
- <typeTitle> - 与第 N 个自变量类型名相关联的标注
- <typeName> - 要记录的自变量的类型的名称
- <size> - 要记录的自变量的大小

• DATA 它包含记录函数转储的所有其他数据。它包括:

- <N> - 要转储的数据对象的序号
- <typeTitle> - 要转储的数据的标注



- <typeName> - 要记录的数据字段的类型的名称，如 PD\_TYPE\_UINT32 和 PD\_TYPE\_STRING
- <size> - 数据对象的大小

---

## db2cos ( 调出脚本 ) 输出文件

在缺省情况下，当数据库管理器因为应急启动、陷阱、分段违例或异常而不能继续处理时，将会调用 db2cos 脚本。每个缺省 db2cos 脚本将调用 db2pd 命令以打开方式收集信息。db2cos 脚本的名称的格式为 db2cos\_hang、db2cos\_trap 等等。每个脚本的行为方式类似，但 db2cos\_hang 则有所不同，它是通过 db2fodc 工具调用的。

缺省 db2cos 脚本在 bin 目录中。在 UNIX 操作系统上，此目录是只读的。可将 db2cos 脚本复制至 adm 目录，必要时可在该位置修改该文件。如果 db2cos 脚本在 adm 目录中，那么会运行该脚本，否则会运行 bin 目录中的脚本。

在多分区配置中，将仅对遇到陷阱的分区上的陷阱代理程序调用该脚本。如果需要从其他分区收集信息，可更新 db2cos 脚本以使用 db2\_all命令，或者如果所有分区在同一台机器上，那么在 db2pd 命令上指定 -alldbpartitionnums 选项。

还可通过 db2pdcfg -cos 命令来配置触发 db2cos 调用的信号类型。缺省配置用于要在发生应急启动或陷阱时运行的 db2cos 脚本。但是，在缺省情况下，生成的信号不会启动 db2cos 脚本。

发生应急启动、陷阱、分段违例或异常时，事件顺序如下所示：

1. 创建陷阱文件
2. 调用信号处理程序
3. 调用 db2cos 脚本（取决于启用的 db2cos 设置）
4. 在管理通知日志中记录相应条目
5. 在 db2diag.log 中记录相应条目

db2cos 脚本中的 db2pd 命令收集到的缺省信息包括有关操作系统、已安装 DB2 产品的版本和服务级别、数据库管理器和数据库配置的详细信息，以及有关以下各项的状态的信息：代理程序、内存池、内存块、应用程序、实用程序、事务、缓冲池、锁定、事务日志、表空间和容器。此外，它还会提供有关下列各项的信息：动态、静态和目录高速缓存的状态、表和索引统计信息、恢复状态以及重新优化的 SQL 语句及活动语句列表。如果需要收集进一步的信息，那么只需使用附加命令更新 db2cos 脚本。

调用缺省 db2cos 脚本时，它将在 DIAGPATH 数据库管理器配置参数指定的目录中生成输出文件。这些文件名为 XXX.YYY.ZZZ.cos.txt，其中 XXX 是进程标识（PID），YYY 是线程标识（TID），而 ZZZ 是数据库分区号（对于单分区数据库则为 000）。如果存在多线程陷阱，那么会对每个线程单独调用 db2cos 脚本。如果 PID 和 TID 组合多次出现，那么该数据将追加至文件。还会显示时间戳记，所以您可以区分输出的迭代。

根据 db2cos 脚本中指定的命令，db2cos 输出文件将包含不同信息。如果未改变缺省脚本，那么会显示类似以下的条目（后跟详细 db2pd 输出）：

```
2005-10-14-10.56.21.523659
PID: 782348          TID: 1          PROC: db2cos
实例: db2inst1      节点: 0        数据库: SAMPLE
APPHDL :           APPID: *LOCAL.db2inst1.051014155507
```

函数: 操作系统服务, sqloEDUCodeTrapHandler, 探测点: 999  
事件: 从操作系统服务 sqloEDUCodeTrapHandler 调用  
      /home/db2inst1/sqllib/bin/db2cos 捕获到陷阱  
实例 db2inst1 使用 64 位和 DB2 代码发行版 SQL09010

...  
操作系统信息:

操作系统名称: AIX  
节点名: n1  
版本: 5  
发行版: 2  
机器: 000966594C00

...

db2diag.log 还将包含与发生位置有关的条目。例如:

```
2005-10-14-10.42.17.149512-300 I19441A349          级别: 事件
PID: 782348          TID: 1          PROC: db2sysc
实例: db2inst1      节点: 000
函数: DB2 UDB, 跟踪服务, pdInvokeCalloutScript, 探测点: 10
开始: 从操作系统服务 sqloEDUCodeTrapHandler 调用 /home/db2inst1/sqllib/bin/db2cos
2005-10-14-10.42.23.173872-300 I19791A310          级别: 事件
PID: 782348          TID: 1          PROC: db2sysc
实例: db2inst1      节点: 000
函数: DB2 UDB, 跟踪服务, pdInvokeCalloutScript, 探测点: 20
停止: 完成 /home/db2inst1/sqllib/bin/db2cos 调用
```

```
2005-10-14-10.42.23.519227-300 E20102A509          级别: 严重
PID: 782348          TID: 1          PROC: db2sysc
实例: db2inst1      节点: 000
函数: DB2 UDB, 操作系统服务, sqloEDUCodeTrapHandler, 探测点: 10
消息: ADM0503C 发生了意外内部处理错误。已关闭所有与此实例相关联的
      DB2 进程。
      已经记录了诊断信息。有关进一步的帮助, 与"IBM 支持机构"联系。
```

```
2005-10-14-10.42.23.520111-300 E20612A642          级别: 严重
PID: 782348          TID: 1          PROC: db2sysc
实例: db2inst1      节点: 000
函数: DB2 UDB, 操作系统服务, sqloEDUCodeTrapHandler, 探测点: 20
DATA #1: 接收到信号编号, 4 字节
11
DATA #2: 信号信息, 64 字节
0x0FFFFFFFFFD5C0 : 0000 000B 0000 0000 0000 0009 0000 0000 .....
0x0FFFFFFFFFD5D0 : 0000 0000 0000 0000 0000 0000 0000 0000 .....
0x0FFFFFFFFFD5E0 : 0000 0000 0000 0000 0000 0000 0000 0000 .....
0x0FFFFFFFFFD5F0 : 0000 0000 0000 0000 0000 0000 0000 0000 .....
```

---

## 转储文件

转储文件是在发生错误时创建的, 它包含将有助于诊断问题(例如, 内部控制块)的其他信息。写至转储文件的每个数据项都具有与其相关联的时间戳记, 以帮助进行问题确定。转储文件使用二进制格式, 目的是供 DB2 客户支持代表使用。

创建或追加转储文件时, 会在 db2diag.log 中建立一个条目, 指示所写数据的时间和类型。这些 db2diag.log 条目类似于以下所示:

---

```
2007-05-18-12.28.11.277956-240 I24861950A192 LEVEL: Severe
PID:1056930 TID:225448 NODE:000 Title: dynamic memory buffer
Dump File:/home/svtdbm5/sqllib/db2dump/1056930.225448.000.dump.bin
```

---

**注:** 对于分区数据库环境, 文件扩展名标识分区号。例如, 以下条目指示转储文件是在分区 10 上运行的 DB2 进程创建的:



---

## 陷阱文件

如果 DB2 由于陷阱、分段违规或异常而不能继续处理，它就会生成陷阱文件。

DB2 接收到的所有信号或异常都会记录在陷阱文件中。陷阱文件还包含发生错误时正在运行的函数序列。此序列有时又称“函数调用堆栈”或“堆栈回溯”。陷阱文件还包含有关捕获到信号或异常时进程的状态的其他信息。

文件位于由 `DIAGPATH` 数据库管理器配置参数指定的目录中。

在所有平台上，陷阱文件名以进程标识（PID）开头，后跟线程标识（TID），再后跟分区号（在单分区数据库上为 000），并以“.trap.txt”结尾。

还有一些诊断陷阱，它们是在发生某些特定条件（这些条件不一定会使实例崩溃）时由代码生成的，在查看堆栈时非常有用。这些陷阱是使用 PID 以十进制格式命名的，后跟分区号（在单分区数据库中为 0）。

示例:

- 6881492.2.000.trap.txt 是进程标识（PID）为 6881492，线程标识（TID）为 2 的陷阱文件。
- 6881492.2.010.trap.txt 是进程和线程在分区 10 上运行的陷阱文件。

可将 `db2pd` 命令与 `-stack all` 或 `-dump` 选项配合使用，以根据需要生成陷阱文件。尽管一般情况下只有在 DB2 支持机构请求时才应完成此任务。

可使用 `db2pd -stacks` 或 `db2pd -dumps` 命令生成堆栈回溯文件。这些文件与陷阱文件的内容相同，但仅供诊断使用。它们的名称类似于 `6881492.2.000.stack.txt`。

## 格式化陷阱文件（Windows）

一个称为 `db2xpert.exe` 的工具，它允许您格式化陷阱文件（\*.TRP）。它会将 DB2 数据库二进制陷阱文件格式化为人们可以阅读的 ASCII 文件。

该工具使用 DB2 符号文件来格式化陷阱文件。这些 .PDB 文件的子集与 DB2 数据库产品包括在一起。

如果在 `DIAGPATH` 中生成了陷阱文件“DB30882416.TRP”，那么可以按如下方式进行格式化:

```
db2xpert DB30882416.TRP DB30882416.FMT
```

---

## 特定于平台的错误日志信息

在 DB2 之外也提供了许多其他文件和实用程序，可用于帮助分析问题。通常，在确定问题的根本原因时，它们和 DB2 文件一样重要。其他文件和实用程序允许访问与下列各方面有关的日志和跟踪中包含的信息。

- 操作系统
- 应用程序和第三方供应商
- 硬件

除此描述的一些领域之外，还有许多其他领域包含这种信息，这取决于您的操作环境。因此，在调试系统中的问题时，要了解需要调查的所有潜在领域。

## 操作系统

每个操作系统都有一组自己的诊断文件，用于跟踪活动和故障。最常见的（通常也是最有用的）诊断文件是错误报告或事件日志。以下是收集此信息的方法列表：

- AIX®: 使用 `/usr/bin/errpt -a` 命令
- Solaris: `/var/adm/messages*` 文件或 `/usr/bin/dmesg` 命令
- Linux: `/var/log/messages*` 文件或 `/bin/dmesg` 命令
- HP-UX: `/var/adm/syslog/syslog.log` 文件或 `/usr/bin/dmesg` 命令
- Windows: 使用系统日志文件、安全性日志文件、应用程序事件日志文件以及 `windir\drwtsn32.log` 文件（其中，`windir` 是 Windows 安装目录）

每个操作系统总是有更多的跟踪和调试实用程序。请参阅您的操作系统文档和支持资料，以确定提供了哪些进一步的信息。

## 应用程序和第三方供应商

每个应用程序都应该有它自己的记录和诊断文件。这些文件将补充 DB2 信息集，以便您更准确地了解潜在问题领域。

## 硬件

硬件设备通常将信息记录到操作系统错误日志中。但是，有时需要其他信息。在这些情况下，需要确定哪些硬件诊断文件和实用程序可用于环境中的某个硬件。例如，在 DB2 报告页面不正确或某种类型的毁坏时。通常由于磁盘问题而报告此类错误，在这种情况下，需要调查硬件诊断文件。请参阅硬件文档和支持材料，以确定提供了哪些进一步的信息。

某些信息（如硬件记录中的信息）有时效性。发生错误时，应尽快从相关源收集尽可能多的信息。

总之，要完全了解并评估问题，需要从 DB2、应用程序、操作系统和底层硬件收集所有可用的信息。`db2support` 工具将自动收集您需要的大多数 DB2 和操作系统信息，但您仍然应该了解除此之外可能有助于调查的任何信息。

## 系统核心文件（Linux 和 UNIX）

如果程序异常终止，那么系统会创建一个核心文件以存储已终止进程的内存映像。诸如内存地址违规、非法指令、总线错误和用户生成的退出信号之类的错误会使核心文件转储。

核心文件名为“core”，除非使用 `DB2FODC` 注册表变量中的值进行配置，否则缺省情况下该文件放在 `diagpath` 目录中。注意，系统核心文件与 DB2 陷阱文件不同。

## 访问系统核心文件信息（Linux 和 UNIX）

`dbx` 系统命令帮助您确定哪个函数导致创建系统核心文件。这是一个简单的检查，它将帮助您确定是数据库管理器有错误，或者帮助确定是操作系统还是应用程序错误导致该问题。

- 必须已安装 dbx 命令。此命令特定于操作系统：在 AIX 和 Solaris 上使用 dbx；在 HP-UX 上使用 xdb，在 Linux 上则使用 gdb。
- 在 AIX 上，使用 chdev 命令或 smitty 以确保启用了完整的核心选项。

可使用以下步骤以确定导致发生核心文件转储的函数。

1. 从 UNIX 命令提示符处输入以下命令：

```
dbx program_name core_filename
```

*program\_name* 是异常终止的程序的名称，*core\_filename* 是包含核心文件转储的文件的名称。*core\_filename* 参数是可选的。如果不指定它，那么会使用缺省名称“core”。

2. 检查核心文件中的调用堆栈。可通过 UNIX 命令提示符发出 `man dbx` 来获取有关如何执行此操作的信息。
3. 要结束 dbx 命令，在 dbx 提示符下输入 `quit`。

以下示例显示如何使用 dbx 命令来读取称为“main”的程序的核文件。

1. 在命令提示符下，输入：

```
dbx main
```

2. 屏幕上将出现类似如下的输出：

```
dbx version 3.1 for AIX.
Type 'help' for help.
reading symbolic information ...
[using memory image in core]
segmentation.violation in freeSegments at line 136
136      (void) shmdt((void *) pcAddress[i]);
```

3. 使核心转储的函数的名称为“freeSegments”。在 dbx 提示符下输入 `where` 以显示故障点的程序路径。

```
(dbx) where
freeSegments(numSegs = 2, iSetId = 0x2ff7f730, pcAddress = 0x2ff7f758, line
136
in "main.c"
main (0x1, 2ff7f7d4), line 96 in "main.c"
```

在本示例中，freeSegments 的第 136 行出错，它是从 main.c 中的第 96 行调用的。

4. 要结束 dbx 命令，在 dbx 提示符下输入 `quit`。

## 访问事件日志 ( Windows )

Windows 事件日志还会提供有用的信息。虽然系统事件日志在发生 DB2 崩溃或有关系系统资源的其他不明错误时好像最有用，但获取全部三种类型的事件日志仍是值得的：

- 系统
- 应用程序
- 安全性

使用 Windows 事件查看器查看事件日志。根据您使用的 Windows 操作系统的不同，用于启动查看器的方法将会不同。

例如，要在 Windows XP 上打开“事件查看器”，请单击**开始** —> **控制面板**。选择**管理工具**，然后双击**事件查看器**。

## 导出事件日志 ( Windows )

可从 Windows 事件查看器导出两种格式的事件日志：日志文件格式和文本或逗号分隔的格式。

使用 Windows 事件查看器导出事件日志。

- 日志文件格式 (\*.evt) 允许您将数据装入回到事件查看器（例如，在另一台机器上）。此格式易于使用，这是因为您可以使用该查看器来切换时间顺序、过滤特定事件以及提前和推后事件。
- 文本或逗号分隔的格式（分别为 \*.txt 和 \*.csv）允许您在大多数文本编辑器中打开日志。它们还避免了关于时间戳记的潜在问题。以 .evt 格式导出事件日志时，时间戳记将使用全球标准时间，并且会在查看器中转换为机器的当地时间。如果不够仔细，可能会因为时差忽略关键事件。文本文件还易于搜索。

## 访问 Dr. Watson 日志文件 ( Windows )

Dr. Watson 日志 drwtsn32.log 是系统上发生的所有异常的编年表。尽管 Dr. Watson 在评估整体系统稳定性及描述 DB2 陷阱的历史记录时很有帮助，但 DB2 陷阱文件比 Dr. Watson 日志更实用。

找到 Dr. Watson 日志文件。缺省路径为 <install\_drive>\Documents and Settings \All Users\Documents\DrWatson

---

## 组合 DB2 数据库和操作系统诊断

在诊断与内存、交换文件、CPU、磁盘存储器和其他资源有关的一些问题时，需要完整地理解给定操作系统管理这些资源的方式。至少在确定与资源有关的问题时需要知道对于每个用户而言，该资源存在的限制及限制程度。（相关限制通常用于 DB2 实例所有者的用户标识。）

以下是需要获取的一些重要配置信息：

- 操作系统补丁级别、已安装软件和升级历史
- CPU 数目
- RAM 量
- 交换和文件高速缓存设置
- 用户数据和文件资源限制及每个用户的进程极限
- IPC 资源限制（消息队列、共享内存段和信号量）
- 磁盘存储器类型
- 机器还有什么功能？DB2 是否争用资源？
- 认证在何处进行？

大多数平台有直接的命令可用来检索资源信息。但是，您很少需要手动获取该信息，原因是 db2support 实用程序会收集此数据及更多其他信息。当指定了 -s 和 -m 选项时，db2support 生成的 detailed\_system\_info.html 文件包含用来收集此信息的许多操作系统命令的语法。

下列练习用于帮助您发现 DB2 诊断文件中的系统配置和用户环境信息。第一个练习让您熟悉运行 db2support 实用程序涉及的步骤。后续练习涉及陷阱文件，这些文件会提供更多 DB2 生成的数据，而这些数据在了解用户环境和资源限制时非常有用。

### 练习 1: 运行 db2support 命令

1. 使用 db2start 命令启动 DB2 实例。
2. 假定您已具有可用的 SAMPLE 数据库，请创建一个目录以存储 db2support 的输出。
3. 切换至该目录并发出：  

```
db2support <directory> -d sample -s -m
```
4. 复查控制台输出，特别是所收集信息的类型。

在 Windows 上运行时，应该看到类似以下的输出：

```
...
正在收集"系统文件"
    "db2cache.prf"
    "db2cos9402136.0"
    "db2cos9402840.0"
    "db2dbamr.prf"
    "db2diag.bak"
    "db2eventlog.000"
    "db2misc.prf"
    "db2nodes.cfg"
    "db2profile.bat"
    "db2system"
    "db2tools.prf"
    "HealthRulesV82.reg"
    "db2dasdiag.log"
...
```

```
正在收集"详细的操作系统和硬件信息"
正在收集"系统资源信息(磁盘、CPU和内存)"
正在收集"操作系统和级别"
正在收集"JDK 级别"
正在收集"DB2 发行版信息"
正在收集"DB2 安装路径信息"
正在收集"注册表信息"
...
```

```
正在创建最终输出归档
    "db2support.html"
    "db2_sql1lib_directory.txt"
    "detailed_system_info.html"
    "db2supp_system.zip"
    "dbm_detailed.supp_cfg"
    "db2diag.log"
db2support 现在已完成。
已生成归档文件"db2support.zip"
```

5. 现在使用 Web 浏览器来查看 detailed\_system\_info.html 文件。在每个系统上标识下列信息：
  - CPU 数目
  - 操作系统级别
  - 用户环境
  - 使用资源限制 (UNIX ulimit 命令)

### 练习 2: 查找 DB2 陷阱文件中的环境信息

1. 确保 DB2 实例已启动，然后发出  

```
db2pd -stack all
```

调用堆栈放在数据库管理器配置参数 `DIAGPATH` 中定义的诊断目录的文件中。

2. 在其中一个陷阱文件中查找下列内容:

- `DB2` 代码级别
- 数据段顶部 (这是所需的最大专用地址空间)
- 当前数据大小 (这是最大专用地址空间限制)
- 当前核心大小 (这是最大核心文件限制)
- 信号处理程序 (此信息可能不会出现在所有陷阱文件中)
- 环境变量 (此信息可能不会出现在所有陷阱文件中)
- 映射输出 (显示装入的库)

Windows 中的示例陷阱文件 (被截断):

```
...
<DB2TrapFile version="1.0">
<Trap>
<Header>
DB2 build information: DB2 v9.1.0.190 s060121 SQL09010
timestamp: 2006-02-17-14.03.43.846000
uname: S:Windows
comment:
process id: 940
thread id: 3592
</Header>
<SystemInformation>
Number of Processors: 1
Processor Type: x86 Family 15 Model 2 Stepping 4
OS Version: Microsoft Windows XP, Service Pack 2 (5.1)
Current Build: 2600
</SystemInformation>
<MemoryInformation>
<Usage>
Physical Memory:    1023 total,    568 free.
Virtual Memory :    2047 total,    1882 free.
Paging File   :    2461 total,    2011 free.
Ext. Virtual   :         0 free.
</Usage>
</MemoryInformation>
<EnvironmentVariables>
<![CDATA[
[e] DB2PATH=C:\Program Files\IBM\SQLLIB
[g] DB2_EXTSECURITY=YES
[g] DB2SYSTEM=MYSRVR
[g] DB2PATH=C:\Program Files\IBM\SQLLIB
[g] DB2INSTDEF=DB2
[g] DB2ADMINSERVER=DB2DAS00
]]></EnvironmentVariables>
```

## 使 **DB2** 和系统事件或错误相关

通常会忽略系统消息和错误日志。如果在问题定义和调查的初始阶段花时间执行一个简单任务,就可以在解决问题时节省几小时、几天甚至几星期的时间。该任务将会比较不同日志中的各个条目,并且记录看起来与时间和各个条目引用的资源相关的所有内容。

虽然并非总是与问题诊断有关,但许多情况下系统日志中会提供最好的线索。如果可将报告的系统问题与 `DB2` 错误相关,通常就能确定直接导致 `DB2` 症状的原因。很明显的示例包括磁盘错误、网络错误和硬件错误。并不那么明显的示例包括在不同机器 (如域控制器) 上报告的错误,机器不同会影响连接时间或认证。

可以调查系统日志以评估稳定性，特别是在全新的系统上报告问题时尤其如此。在常用应用程序中间歇发生陷阱可能表示存在底层硬件问题。

以下是系统日志提供的一些其他信息。

- 重要事件，如重新引导系统的时间
- 系统上发生 DB2 陷阱（及失败的其他软件中的错误、陷阱或异常）的时间
- 内核应急启动、文件系统空间不足和交换空间不足错误（可能导致系统无法创建或派生新进程）

系统日志可帮助您在 `db2diag.log` 中排除作为考虑因素的崩溃条目。如果在 DB2 管理通知或 DB2 诊断日志中发现崩溃恢复但先前没有任何错误，那么 DB2 崩溃恢复可能是系统关闭造成的。

这一关联信息原则扩展至来自任何源的日志和所有可标识用户症状。例如，它在标识和描述来自另一应用程序的日志的关联条目时很有用，即使您不能完整地解释它们也是如此。

总结此信息就能够很完整地了解服务器和发生问题时进行的各种事件。





---

## 第 2 章 DB2 故障诊断

通常，故障诊断要求您先找出并确定问题，然后再寻求解决方法。本节将提供关于 DB2 产品特定功能的故障诊断信息。

对于已知常见问题，找出的解决方法在本节以核对表的形式提供。如果您无法在该核对表中找到解决方法，那么可以自己收集其他诊断数据并进行分析，或者将该数据提交给 IBM 软件支持机构进行分析。

下列问题将引导您执行相应的故障诊断任务：

1. 是否已经应用了所有已知修订包？如果尚未应用，请考虑《DB2 服务器快速入门》中的“应用修订包”。
2. 是否在以下情况下出现问题：
  - 安装 DB2 数据库服务器或客户机时？如果是，请参阅本书其他位置处的“收集有关安装问题的数据”主题。
  - 创建、删除、更新或迁移实例或 DB2 管理服务器（DAS）时？如果是，请参阅本书其他位置处的“收集有关 DAS 和实例管理问题的数据”主题。
  - 使用 EXPORT、IMPORT、LOAD 或 db2move 命令移动数据时？如果是，请参阅本书其他位置处的“收集有关数据移动问题的数据”主题。

如果您的问题不属于上面任一类别并且要与 IBM 软件支持机构联系，那么可能仍需要基本的诊断数据。应参阅本书其他位置处的“收集有关 DB2 的数据”主题。

---

### 收集 DB2 数据

有时只对症状进行故障诊断不能解决问题。在这种情况下，需要收集诊断数据。需要收集的诊断数据以及所收集的数据的来源取决于正在调查的问题类型。这些步骤表示如何收集在将问题提交给 IBM 软件支持机构时通常需要提供的基本信息集。

要获取最完整的输出，实例所有者应调用 db2support 实用程序。

要收集基本诊断信息集并将其存储在压缩的文件归档中，请输入 db2support 命令：

```
db2support <output_directory> -s -d <database_name> -c
```

使用 -s 会给出有关使用的硬件和操作系统的系统详细信息。使用 -d 会给出有关指定数据库的详细信息。使用 -c 允许尝试连接至指定数据库。

输出的收集非常方便，并且会存储在压缩的 ZIP 归档 db2support.zip 中，以便可以很轻松地任何系统上对其进行传送和解压缩。

对于特定症状或产品的特定部分中的问题，可能需要收集其他数据。请参阅特定于问题的“收集数据”文档。

接下来，您可以执行下列任何任务：

- 分析数据
- 将数据提交给 IBM 软件支持机构

---

## 收集关于安装问题的数据

如果您遇到安装问题并且不能确定问题的原因，请收集您或 IBM 软件支持机构可用来诊断和解决问题的诊断数据。

要收集关于安装问题的诊断数据：

1. 可选：在启用跟踪的情况下重复安装尝试。例如：

```
db2setup -t trace.out  
或  
setup -t trace.out
```

2. 找到安装日志文件。

- 在 Windows 上，缺省文件名为“DB2-ProductAbbreviation-DateTime.log”。例如：DB2-ESE-Wed Jun 21 11\_59\_37 2006.log。安装日志的缺省位置为“我的文档”\DB2LOG\ 目录。
- 在 Linux 和 UNIX 上，缺省文件名为 db2setup.log、db2setup.his 和 db2setup.err。

如果在启用跟踪（或调试方式）后问题再次出现，那么可能会创建其他文件，例如：dascrt.log、dasdrop.log、dasupdt.log、db2icrt.log.PID、db2idrop.log.PID、db2imigr.log.PID 和 db2iupdt.log.PID（其中 PID 是进程标识）。

所有这些文件的缺省位置都是 /tmp 目录。

3. 可选：如果打算将数据提交给 IBM 软件支持机构，还请收集 DB2 数据。

---

## 收集关于数据移动问题的数据

如果您在执行数据移动命令时遇到问题并且不能确定问题的原因，请收集您或 IBM 软件支持机构可用来诊断和解决问题的诊断数据。

- 要收集与 db2move 命令相关的数据，请转至发出了此命令的目录。根据在此命令中指定的操作找到下列文件：
  - 对于 COPY 操作，查找名为 COPY.timestamp.ERR 和 COPYSHEMA.timestamp.MSG 的文件。如果还指定了 LOAD\_ONLY 或 DDL\_AND\_LOAD 方式，还需查找名为 LOADTABLE.timestamp.MSG 的文件。
  - 对于 EXPORT 操作，查找名为 EXPORT.out 的文件。
  - 对于 IMPORT 操作，查找名为 IMPORT.out 的文件。
  - 对于 LOAD 操作，查找名为 LOAD.out 的文件。
- 要收集与 EXPORT、IMPORT 或 LOAD 命令相关的数据，确定命令是否包括 MESSAGES 参数。如果包括此参数，那么收集输出文件。如果您不指定其他目录和驱动器，那么这些实用程序使用当前目录和缺省驱动器作为目标。
- 要收集与 REDISTRIBUTE 命令相关的数据，在 Linux 和 UNIX 上，查找名为“*dbname.database\_partition\_groupname.timestamp*”的文件；在 Windows 上，查找名为“*dbname.database\_partition\_groupname.date.time*”的文件。文件分别位于 \$HOME/sql/lib/db2dump 目录或 \$DB2PATH/sql/lib/redist 目录中，其中 \$HOME 是实例所有者的主目录。

---

## 收集关于 DAS 和实例管理问题的数据

如果您在执行 DB2 管理服务器 (DAS) 或实例管理时遇到问题并且不能确定问题的原因, 请收集您或 IBM 软件支持机构可用来诊断和解决问题的诊断数据。

这些步骤仅适用于您在 Linux 或 UNIX 上使用 DB2 时重现问题的情况。

要收集关于 DAS 或实例管理问题的诊断数据:

1. 在启用跟踪或调试方式的情况下重复失败的命令。示例命令:

```
db2setup -t trace.out
dascrt -u DASUSER -d
dasdrop -d
dasmigr -d
dasupdt -d
db2icrt -d INSTNAME
db2idrop INSTNAME -d
db2imigr -d INSTNAME
db2iupdt -d INSTNAME
```

2. 找到诊断文件。可能会有多个文件, 请比较时间戳记以确保获取所有合适的文件。

缺省情况下, 输出内容将出现在 /tmp 目录中。

示例文件名为:

dascrt.log、dasdrop.log、dasupdt.log、db2icrt.log.PID、db2idrop.log.PID、db2imigr.log.PID 和 db2iupdt.log.PID, 其中 PID 是进程标识。

3. 将诊断文件提供给 IBM 软件支持机构。

如果是因为 db2start 或 START DATABASE MANAGER 命令失败而导致出现问题, 请在 insthome/sqllib/log 目录 (其中 insthome 是实例所有者的主目录) 中查找名为 db2start.timestamp.log 的文件。同样, 如果是因为 db2stop 或 STOP DATABASE MANAGER 命令失败而导致出现问题, 那么查找名为 db2stop.timestamp.log 的文件。仅当数据库管理器未在 start\_stop\_time 数据库管理器配置参数所指定的时间内响应命令时, 才会出现这些文件。

---

## 分析 DB2 数据

在收集数据后, 您需要确定这些数据如何帮助您解决特定问题。分析类型取决于正在调查的问题类型和已收集的数据。这些步骤表示如何开始调查任何基本 DB2 诊断数据。

要分析诊断数据, 请执行下列操作:

- 弄清楚各个数据相互之间的关系。例如, 如果数据在多个系统上, 那么有条理地组织好数据, 以便知道每个数据的来源。
- 通过检查时间戳记确认每个诊断数据与问题的计时有关。注意, 来自不同源的数据可能采用不同的时间戳记格式; 一定要了解每种时间戳记格式中不同元素的序列, 以便知道不同事件的发生时间。
- 确定最有可能包含与问题有关的信息的数据源, 并从那里开始分析。例如, 如果问题与安装有关, 那么从安装日志文件 (如果有的话) 开始分析, 而不是从一般产品或操作系统日志文件开始分析。

- 每个数据源的特定分析方法都是唯一的，但大多数跟踪文件和日志文件都适用的一个技巧是，一开始就要确定数据中出现问题的时间点。在确定此时间点后，您可以从此时间点往前分析数据，以便弄清楚问题的根本原因。
- 如果您正在调查一个问题，并且拥有两个环境中几乎一样多的数据（其中一个环境正在运行，而另一个环境未运行），那么应从比较每个环境的操作系统和产品配置详细信息开始。

## 分析安装问题的数据

收集有关安装问题的诊断数据后，可分析此数据以确定产生问题的原因。这些步骤是可选的。如果不能轻易确定问题原因，请将数据提交给 IBM 软件支持机构。

这些步骤假定您以具有收集安装问题的数据中描述的文件。

- 确保您正在查看相应的安装日志文件。请检查文件的创建日期或文件名中包含的时间戳记（在 Windows 上）。
- 确定是否已成功完成安装。
  - 在 Windows 上，安装日志文件的底部出现类似以下的消息指示安装成功：
 

```
Property(C): INSTALL_RESULT = 已成功完成安装
=== 日志记录停止时间: 6/21/2006 16:03:09 ===
MSI (c) (34:38) [16:03:09:109]:
产品: DB2 企业服务器版 - DB2COPY1 -- 已成功完成安装操作。
```
  - 在 Linux 和 UNIX 上，安装日志文件（缺省情况下为 db2setup.log）底部的消息指示安装成功。
- 可选：确定是否发生了错误。如果安装成功完成，但您在安装过程中接收到错误消息，那么在安装日志文件中找到这些错误。
  - 在 Windows 上，大多错误都以“错误：”或“警告：”开头。例如：
 

```
1: 错误: 运行命令"D:\IBM\SQLLIB\bin\db2.exe CREATE TOOLS CATALOG SYSTOOLS
USE EXISTING DATABASE TOOLSDB FORCE"以启动和/或迁移 DB2 工具目录数据库时出现
错误。返回值为"4"。
1: 警告: 在此计算机上安装"DB2 企业服务器版 - DB2COPY1"时
发生了小错误。某些功能可能不会正确地起作用。
```
  - 在 Linux 和 UNIX 上，如果 Java™ 返回了任何错误（如异常和陷阱信息），那么将显示缺省名为 db2setup.err 的文件。

如果您启用了安装跟踪，安装日志文件中将具有更多的条目且条目会更详细。

如果分析此数据不能帮您解决问题，且如果您与 IBM 签有维护合同，那么可发出问题报告。IBM 软件支持机构将让您提交收集的任何数据，并可能需要您执行的任何分析。

如果您的调查未解决问题：

- 将数据提交给 IBM 软件支持机构。

---

## 将数据提交给 IBM 软件支持机构

下列步骤假定您已向 IBM 软件支持机构开放问题管理记录（PMR）。

可使用下列其中一种方法将日志文件和配置文件之类的诊断数据发送至 IBM 软件支持机构：

- FTP
- 电子服务请求（ESR）工具

- 要通过 FTP 向增强中央客户端数据存储库 (EcuRep) 提交文件:
  1. 将收集的数据文件打包成 ZIP 或 TAR 格式, 并根据问题管理记录 (PMR) 标识指定压缩包的名称。

文件必须使用以下命名约定以便与 PMR 正确关联: xxxxx.bbb.ccc.yyy.yyy, 其中 xxxxx 是 PMR 号, bbb 是 PMR 的分支号, ccc 是 PMR 的地域代码, 而 yyy.yyy 是文件名。

2. 使用 FTP 实用程序连接至服务器 ftp.emea.ibm.com。
3. 作为用户标识“anonymous”登录并输入您的电子邮件地址作为密码。
4. 转至 toibm 目录。例如, cd toibm。
5. 转至其中一个特定于操作系统的子目录。例如, 子目录包括 aix、linux、unix 或 windows。
6. 切换至二进制方式。例如, 在命令提示符下输入 bin。
7. 使用 put 命令将文件放在服务器上。使用以下文件命名约定来指定文件的名称并将其放到服务器上。您的 PMR 将更新, 以使用以下格式列示文件的存储位置: xxxx.bbb.ccc.yyy.yyy。(xxx 是 PMR 号, bbb 是分支, ccc 是地域代码, 而 yyy.yyy 是 tar.Z 或 xyz.zip 之类的文件类型的描述。) 可将文件发送至 FTP 服务器, 但不能更新它们。如果以后任何时间需要更改该文件, 那么需要创建新的文件名。
8. 输入 quit 命令。

- 要使用 ESR 工具提交文件, 请执行下列操作:

1. 登录 ESR。
2. 在“欢迎”页上的输入报告编号字段中输入 PMR 号, 然后单击**执行**。
3. 下滚至**附加相关文件**字段。
4. 单击**浏览**以查找要提交至 IBM 软件支持机构的日志、跟踪或其他诊断文件的位置。
5. 单击**提交**。文件将通过 FTP 传输至 IBM 软件支持机构, 并且与您的 PMR 相关联。

有关 EcuRep 服务的更多信息, 请参阅 IBM EMEA 中央客户数据存储服务。

有关 ESR 的更多信息, 请参阅电子服务请求 (ESR) 帮助。

---

## 当前发行版故障诊断指导

对在使用新增和已更改数据库函数与功能时可能遇到的潜在问题, 提供了相关指导。对数据库操作的新近更改可能导致您忘记或忽视了利用新增或已更改功能的最佳可能方法。下列故障诊断主题对可能的问题区域和解决方法提供了建议和指导, 以便您可以充分利用本发行版介绍的新增和已更改数据库函数与功能。

### 故障诊断优化准则和概要文件

通过 EXPLAIN 表提供了对优化准则的诊断支持 (由优化概要文件传递)。

如果优化器未应用优化准则, 您将接收到 SQL0437W 警告, 原因码为 13。详细说明未应用优化准则的原因的诊断信息将被添加到 EXPLAIN 表中。有两个 EXPLAIN 表用于接收优化器诊断输出:

- `EXPLAIN_DIAGNOSTIC` - 此表中的每个条目都表示一条与特定语句的优化相关的诊断消息。每条诊断消息都使用一个数字代码表示。
- `EXPLAIN_DIAGNOSTIC_DATA` - 此表中的每个条目都是与 `EXPLAIN_DIAGNOSTIC` 表中的特定诊断消息相关的诊断数据。

这些表的 DDL 显示在第 31 页的图 1 中。

以下步骤可以帮助您诊断使用优化准则时出现的问题:

1. 《调整数据库性能》中的“验证已使用的优化准则”。
2. 使用内置 *Administrative Routines and Views* 中的“`EXPLAIN_GET_MSGS` 表函数”检查完整错误消息。

如果您完成上述步骤后仍不能确定问题所在, 请开始收集诊断数据并考虑与 IBM 软件支持机构联系。

```

CREATE TABLE EXPLAIN_DIAGNOSTIC
( EXPLAIN_REQUESTER VARCHAR(128) NOT NULL,
  EXPLAIN_TIME       TIMESTAMP    NOT NULL,
  SOURCE_NAME        VARCHAR(128) NOT NULL,
  SOURCE_SCHEMA      VARCHAR(128) NOT NULL,
  SOURCE_VERSION     VARCHAR(64)  NOT NULL,
  EXPLAIN_LEVEL      CHAR(1)      NOT NULL,
  STMTNO             INTEGER       NOT NULL,
  SECTNO             INTEGER       NOT NULL,
  DIAGNOSTIC_ID     INTEGER       NOT NULL,
  CODE              INTEGER       NOT NULL,
  PRIMARY KEY (EXPLAIN_REQUESTER,
               EXPLAIN_TIME,
               SOURCE_NAME,
               SOURCE_SCHEMA,
               SOURCE_VERSION,
               EXPLAIN_LEVEL,
               STMTNO,
               SECTNO,
               DIAGNOSTIC_ID),
  FOREIGN KEY (EXPLAIN_REQUESTER,
               EXPLAIN_TIME,
               SOURCE_NAME,
               SOURCE_SCHEMA,
               SOURCE_VERSION,
               EXPLAIN_LEVEL,
               STMTNO,
               SECTNO)
  REFERENCES EXPLAIN_STATEMENT ON DELETE CASCADE);

```

```

CREATE TABLE EXPLAIN_DIAGNOSTIC_DATA
( EXPLAIN_REQUESTER VARCHAR(128) NOT NULL,
  EXPLAIN_TIME       TIMESTAMP    NOT NULL,
  SOURCE_NAME        VARCHAR(128) NOT NULL,
  SOURCE_SCHEMA      VARCHAR(128) NOT NULL,
  SOURCE_VERSION     VARCHAR(64)  NOT NULL,
  EXPLAIN_LEVEL      CHAR(1)      NOT NULL,
  STMTNO             INTEGER       NOT NULL,
  SECTNO             INTEGER       NOT NULL,
  DIAGNOSTIC_ID     INTEGER       NOT NULL,
  ORDINAL            INTEGER       NOT NULL,
  TOKEN              VARCHAR(1000),
  TOKEN_LONG         BLOB(3M) NOT LOGGED,
  FOREIGN KEY (EXPLAIN_REQUESTER,
               EXPLAIN_TIME,
               SOURCE_NAME,
               SOURCE_SCHEMA,
               SOURCE_VERSION,
               EXPLAIN_LEVEL,
               STMTNO,
               SECTNO,
               DIAGNOSTIC_ID)
  REFERENCES EXPLAIN_DIAGNOSTIC ON DELETE CASCADE);

```

注: EXPLAIN\_REQUESTER、EXPLAIN\_TIME、SOURCE\_NAME、SOURCE\_SCHEMA、SOURCE\_VERSION、EXPLAIN\_LEVEL、STMTNO 和 SECTNO 列同时包括在两个表中, 以形成 EXPLAIN\_STATEMENT 表的外键及 EXPLAIN\_DIAGNOSTIC 与 EXPLAIN\_DIAGNOSTIC\_DATA 之间的父子关系。

图 1. 用于创建诊断说明表的 DDL

此 DDL 包括在 sqllib 目录的 misc 子目录下的 EXPLAIN.DDL 文件中。



## 存储键故障诊断支持

存储保护键（线程级别的硬件键）可以避免无效内存访问，以使 DB2 引擎具有更大的弹性。遵循以下步骤来诊断启用此功能部件时遇到的问题或启用此功能部件后遇到的陷阱。

### 诊断注册表变量错误

设置注册表变量『DB2\_MEMORY\_PROTECT』和『DB2\_THREAD\_SUSPENSION』（在《数据服务器、数据库和数据库对象指南》中）时，返回了无效值（DBI1301E）错误。因为以下原因之一发生了此错误：

- 为注册表变量给定的值无效。请参阅注册表变量用法以获取适用的变量 DB2\_MEMORY\_PROTECT 或 DB2\_THREAD\_SUSPENSION。
- 如果在设置 DB2\_MEMORY\_PROTECT 变量时发生错误，那么硬件和操作系统可能不支持存储保护键，并且此功能部件无法启用。存储保护键在 POWER6 处理器中可用并且受具有 5300-06 技术级别的 AIX 5L V5.3 操作系统的支持。
- 如果将 DB2\_THREAD\_SUSPENSION 变量设置为 ON 时发生错误，那么在设置 DB2\_THREAD\_SUSPENSION 变量前未设置 DB2\_MEMORY\_PROTECT 变量。使用 **db2set** 命令将 DB2\_MEMORY\_PROTECT 变量设置为 YES。下一步，使用 **db2set** 命令将 DB2\_THREAD\_SUSPENSION 变量设置为 ON。最后，停止然后重新启动 DB2 实例以激活注册表变量更改。

### 诊断陷阱

DB2 实例将为遇到的陷阱准备首次出现数据捕获（FODC）包。如果已将 DB2 实例配置为具有更大的数据库弹性，DB2\_MEMORY\_PROTECT 变量设置为 YES 且 DB2\_THREAD\_SUSPENSION 变量设置为 ON，那么 DB2 实例还确定了陷阱是否是持久的。术语“持久”的意思是捕获的 DB2 引擎线程已被暂挂或终止并且 DB2 实例将继续运行。执行下列这些步骤：

1. 使用文本编辑器来查看管理通知日志文件。如果陷阱可持续并且 DB2 实例仍在运行，那么您会见到错误消息『ADM14010C』（在《消息参考，第 1 卷》中）。否则，您会见到错误消息『ADM14011C』（在《消息参考，第 1 卷》中）并且 DB2 实例已关闭。
2. 注意，FODC 信息的目录名在上述相应错误消息中指定。
3. 如果陷阱是持久的，尽快停止 DB2 实例。因为当陷阱为持久时 DB2 引擎将暂挂，如果 **db2stop** 和 STOP DATABASE MANAGER 命令用于停止 DB2 实例时也将挂起。因而，必须使用 **db2\_kill** 命令来停止 DB2 实例和除去暂挂的 DB2 引擎线程。
4. 使用 **db2start** 或 START DATABASE MANAGER 命令重新启动 DB2 实例。
5. 联系 IBM 客户支持机构并请参阅 FODC 诊断信息以解决导致陷阱的原因。

## 未自动创建数据压缩字典

表很大，但未创建数据压缩字典。您希望了解未按预期创建数据压缩字典的原因。

您可能会处于以下情况：

- 某个表中的 COMPRESS 属性设置为 YES。
- 该表已存在一段时间并且曾添加和除去数据。



- 表的大小接近线程大小（大约为 1 到 2 MB）。您希望自动创建数据压缩字典。
- 您运行了期望的表数据填充操作（如 INSERT、LOAD INSERT 或 REDISTRIBUTE），这会使表大小增加至超过阈值大小。
- 未自动创建数据压缩字典。未创建数据压缩字典并将其放入表中。您期望压缩该时间点之后添加至表的数据，但数据仍为解压缩状态。

为什么未执行表数据压缩？

尽管表大小超过允许自动创建数据压缩字典的阈值大小，但仍会检查另一条件。该条件是：表中必须有足够的数据以允许创建该字典。针对表数据的过去活动可能也包括删除或除去数据。表内可能有若干不包含数据的大块。这样一来，大表可能超出或刚好符合表大小阈值，但表中也可能没有足够的数据来创建字典。

如果对表执行了大量活动，那么需要定期重组表。否则，表大小可能会很大，但填充的数据会很稀疏。重组表会消除碎片数据并压缩表中的数据。重组之后，表会变小并且数据的填充会更紧密。重组表可以更准确地体现表中的数据量，并且会小于允许自动创建数据压缩字典的阈值大小。

使用 REORGCHK 命令来确定是否需要重组表。

## 诊断全局变量问题

一般来说，如果遇到问题的用户有权读取全局变量，他们就可以诊断与全局变量有关的应用程序。只需要具有读许可权，就可以通过发出 VALUES(Global Variable Name) 语句来知道全局变量的值。有时运行应用程序的用户无权读取全局变量。

第一个场景说明引用全局变量时的可能问题有一个简单的解决方案。第二个场景说明更有可能出现的情况：需要对适当用户授予针对全局变量的读许可权。

### 场景 1

对全局变量的引用必须正确限定。可能存在名称相同但模式不同的变量，并且之前在 PATH 寄存器值中遇到了不正确的模式。一个解决方案是确定对全局变量的引用使用的是标准名称。

### 场景 2

应用程序开发者（developerUser）根据只有他才有读许可权的一些全局变量的值创建了很复杂的一系列过程、视图、触发器等等。应用程序的最终用户（finalUser）登录并开始使用 developerUser 创建的环境发出 SQL。finalUser 向 developerUser 抱怨他看不到应该允许看到的数据。在诊断此问题时，developerUser 将其授权标识切换为 finalUser 的授权标识，作为 finalUser 登录并尝试发出 finalUser 所发出的 SQL。developerUser 发现 finalUser 所说的问题是事实。

developerUser 必须确定 finalUser 是否看到了他所看到的全局变量。developerUser 运行 SET SESSION USER 以查看 finalUser 所看到的全局变量值。以下是确定并解决此问题的建议方法。

developerUser 要求安全管理员（secadmUser）授予其许可权以作为 finalUser 使用 SET SESSION USER。然后，developerUser 以本来身份登录并使用 SET SESSION AUTHORIZATION 语句以将 SESSION\_USER 专用寄存器设置为 finalUser 的专用寄存器。运

行出现该问题的 SQL 后，他使用 SET SESSION AUTHORIZATION 语句切换到 developerUser 身份。developerUser 现在发出 VALUES 语句并会看到全局变量的实际值。

下面是一个样本 SQL，显示 developerUser 在数据库中采取的操作。

```
#####
# developerUser connects to database and creates needed objects
#####

db2 "connect to sample user developerUser using xxxxxxxx"

db2 "create table security.users \
(userid varchar(10) not null primary key, \
firstname varchar(10), \
lastname varchar(10), \
authlevel int)"

db2 "insert into security.users values ('ZUBIRI', 'Adriana', 'Zubiri', 1)"
db2 "insert into security.users values ('SMITH', 'Mary', 'Smith', 2)"
db2 "insert into security.users values ('NEWTON', 'John', 'Newton', 3)"

db2 "create variable security.gv_user varchar(10) default (SESSION_USER)"
db2 "create variable security.authorization int default 0"

# Create a procedure that depends on a global variable
db2 "CREATE PROCEDURE SECURITY.GET_AUTHORIZATION() \
SPECIFIC GET_AUTHORIZATION \
RESULT SETS 1 \
LANGUAGE SQL \
SELECT authlevel INTO security.authorization \
FROM security.users \
WHERE userid = security.gv_user"

db2 "grant all on variable security.authorization to public"
db2 "grant execute on procedure security.get_authorization to public"
db2 "terminate"

#####
# secadmUser grants setsessionuser
#####
db2 "connect to sample user secadmUser using xxxxxxxx"
db2 "grant setsessionuser on user finalUser to user developerUser"
db2 "terminate"

#####
# developerUser will debug the problem now
#####

echo "-----"
echo " Connect as developerUser "
echo "-----"
db2 "connect to sample user developerUser using xxxxxxxx"

echo "-----"
echo " SET SESSION AUTHORIZATION = finalUser "
echo "-----"
db2 "set session authorization = finalUser"

echo "--- TRY to get the value of gv_user as finalUser (we should not be able to)"
db2 "values(security.gv_user)"

echo "--- Now call the procedure---"
db2 "call security.get_authorization()"

echo "--- if it works it should return 3 ---"
```

```

db2 "values(security.authorization)"

echo "-----"
echo " SET SESSION AUTHORIZATION = developerUser "
echo "-----"

db2 "set session authorization = developerUser"

echo "--- See what the variable looks like ----"
db2 "values(security.gv_user)"

db2 "terminate"

```

## 诊断工作负载管理

工作负载是一个数据库对象，由用于组合数据库中的一个或多个工作单元（UOW）的用户定义条件组成。工作负载存象由与工作负载相关联的数据库连接内的一个或多个 UOW 组成。如果同一工作负载与另一数据库连接的另一组 UOW 相关联，那么该组 UOW 被视为另一工作负载存象。

对于每个工作负载，可能有多个工作负载存象在系统上并行运行。

如果工作负载或工作负载存象出现问题或困难，您可能想要知道有关每一项的信息类型。信息类型包括：

- 工作负载存象列表。请使用 `WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURENCES` 表函数。此函数的输出包括应用程序在系统范围内的唯一标识以及唯一 UOW 标识。
- 工作负载及工作负载存象的标识。请使用 `WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES` 表函数。此函数的输出包括唯一 UOW 标识和 UOW 内的唯一活动标识。
- 当前在工作负载存象内运行的所有活动（和请求）的列表。请使用 `WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES` 表函数。此函数的输出包括唯一 UOW 标识和 UOW 内的唯一活动标识。
- 有关当时内存中的工作负载信息的内容。请使用 `db2pd -workloads` 命令。此命令的输出包括工作负载列表、工作负载权限信息以及表示每个工作负载的并行工作负载存象数目的计数器。

此外，如果在诊断工作负载管理问题时需要与 IBM 联系，那么需要收集与此情况相关联的 `db2agent` 的相关信息。请使用 `db2pd -stack` 收集所需的信息。与 IBM 联系后，IBM 可能要求您执行 DB2 跟踪（`db2trc`）。

## 诊断脚本

您可能有一些内部工具或脚本以数据库引擎中运行的进程为基础。因为所有代理程序、预取程序和页面清除程序现在被视为单个多线程进程内的线程，所以这些工具或脚本可能不再生效。

内部工具和脚本必须进行修改才能用于线程化进程。例如，某些脚本可能会调用 `ps` 命令来列示进程名称；然后对特定代理进程执行任务。这些脚本需要重写。

问题确定数据库命令 `db2pd` 会有一个新选项 `-edu`（“引擎可调度单元”的缩写），用于列示所有代理程序名称及其线程标识。`db2pd -stack` 命令继续与线程化引擎配合使用，以转储各个 EDU 堆栈或当前节点的所有 EDU 堆栈。

## 诊断数据不一致问题

诊断数据库中出现数据不一致问题的位置是非常重要的。确定数据不一致的一种方法是使用 `INSPECT` 命令中的输出来标识存在问题的位置。发现不一致时，必须确定如何处理该问题。

一旦确定存在数据一致性问题，有两个选择：

- 与 DB2 服务机构联系，并要求他们帮助您从数据不一致的情况进行恢复
- 删除并重建存在数据一致性问题的数据库对象。

您将使用 `INSPECT` 命令的 `INSPECT CHECK` 变体来检查存在数据不一致的数据库、表空间或表。一旦生成 `INSPECT CHECK` 命令的结果，就应使用 `db2inspf` 命令格式化检查结果。

如果 `INSPECT` 命令未完成，请与 DB2 服务机构联系。

## 诊断索引与数据不一致问题

索引必须是准确的，才能快速访问表中的正确数据，除非数据库已毁坏。

可使用 `INSPECT` 命令并通过在交叉对象检索子句中使用 `INDEXDATA` 选项，以对索引与数据不一致问题执行联机检查。使用 `INSPECT` 命令时，缺省情况下不会执行索引数据检查；必须明确请求才会执行此检查。

如果因为 `INSPECT` 执行 `INDEXDATA` 检查时索引数据不一致而发现错误，那么会返回错误消息 `SQL1141N`。在返回此错误消息的同时，会收集数据诊断信息并将其转储至 `db2diag.log`。还会在管理通知日志中记录紧急消息。请使用 `db2diag.log` 分析工具来过滤并格式化 `db2diag.log` 文件的内容。

### 锁定影响

使用 `INSPECT` 命令和 `INDEXDATA` 选项检查索引与数据不一致问题时，仅以 `IS` 方式锁定被检查的表。

指定 `INDEXDATA` 选项后，缺省情况下只会使用明确指定级别子句选项的值。对于未明确指定的任何级别子句选项，缺省级别（`INDEX NORMAL` 和 `DATA NORMAL`）会从 `NORMAL` 改写为 `NONE`。

## 诊断分区数据库环境

当您的环境具有分区数据库时，有一些独特的故障诊断注意事项。如果主机文件具有当您在非分区数据库环境中工作时有效的特定条目，那么快速通信管理器（FCM）将遇到问题。

### 与 127.0.0.2 有关的 FCM 问题（Linux 和 UNIX）

在分区数据库环境中，如果 `/etc/hosts` 文件中有对应 `127.0.0.2` 的条目，那么快速通信管理器（FCM）会遇到问题。

### 症状

根据情况，可能会出现不同的错误消息。例如，创建数据库时可能会发生以下错误：`SQL1229N` 当前事务已回滚，因为发生了系统错误。SQLSTATE=40504

## 原因

jk该问题是由于 `/etc/hosts` 文件中存在对应 IP 地址 `127.0.0.2` 的条目而导致的，其中 `127.0.0.2` 映射至机器的标准主机名。例如：

```
127.0.0.2 ServerA.ibm.com ServerA
```

其中“`ServerA.ibm.com`”是标准主机名。

## 环境

该问题仅限于带有DB2 数据库分区功能部件的 DB2 企业服务器版。

## 解决问题

从 `/etc/hosts` 文件中除去该条目，或者将其转换为注释。例如：

```
# 127.0.0.2 ServerA.ibm.com ServerA
```



---

## 第 3 章 DB2 Connect 故障诊断

对于非本地数据库通常涉及的问题，使用 DB2 Connect 的系统环境具有唯一的可能性。收集有关问题的信息还必须涉及通信协议和硬件。此外，远程数据库也可能是问题的根源，您需要将它考虑在内。

---

### 故障诊断

DB2 Connect 环境涉及到多个软件、硬件和通信产品。通过排除和提炼可用的数据以得出结论（找出发生错误的位置）是故障诊断的最佳方法。

在收集相关信息之后，根据您选择的适当主题，转到可参考的章节。

### 收集相关信息

故障诊断包括缩小问题的范围和调查可能的原因。正确的出发点是收集相关信息，确定您知道哪些信息，哪些数据尚未收集，以及可以排除哪些途径。至少要回答下列问题：

- 最初的连接已经成功了吗？
- 硬件运行正常吗？
- 通信路径正常吗？
- 有任何通信网络更改会使先前的目录条目无效吗？
- 已经启动数据库了吗？
- 一个或多个客户机与 DB2 Connect 服务器（网关）之间的通信、DB2 Connect 网关与主机或 System i™ 数据库服务器之间的通信或 DB2 Connect 个人版与主机或 System i 数据库服务器之间的通信中断了吗？
- 利用消息中所返回的消息内容和标记可以确定哪些问题？
- 此时将使用诊断工具（例如，db2trc、db2pd 或 db2support）来提供任何帮助吗？
- 执行类似任务的其他机器运行正确吗？
- 如果这是一个远程任务，那么在本地能成功执行它吗？

### 初始连接不成功

复查下列问题并确保已遵循了安装步骤：

1. 成功完成了安装过程吗？
  - 所有必备软件产品都可用吗？
  - 内存和磁盘空间足够用吗？
  - 安装了远程客户机支持吗？
  - 完成了通信软件的安装并且没有任何错误情况吗？
2. 对于 UNIX 操作系统，创建了产品的实例吗？
  - 作为 root 用户，您创建了将作为实例所有者和 sysadm 组的用户和组了吗？
3. 成功地处理了许可证信息吗？（如果适用的话）
  - 对于 UNIX 操作系统，您编辑了节点锁定文件并输入了由 IBM 提供的密码吗？



4. 是否正确地配置了主机或 *System i* 数据库服务器和 workstation 通信？
  - 必须考虑的三个配置：
    - a. 主机或 *System i* 数据库服务器配置向服务器标识应用程序请求器。主机或 *System i* 服务器数据库管理系统将具有系统目录条目，这些条目将根据位置、网络协议和安全性来定义请求器。
    - b. DB2 Connect workstation 配置向服务器定义客户机成员并向客户机定义主机或 *System i* 服务器。
    - c. 客户机 workstation 配置必须定义工作站的名称和通信协议。
  - 关于未执行最初连接的问题分析包括：验证 PU（物理单元）名称都是完整的且正确的，或者，对于 TCP/IP 连接，验证是否已经指定了正确的端口号和主机名。
  - 主机或 *System i* 服务器数据库管理员和“网络”管理员都有用来诊断问题的实用程序。
5. 您是否具有主机或 *System i* 服务器数据库管理系统在使用主机或 *System i* 服务器数据库时需要的权限级别？
  - 考虑用户的访问权限、表限定符的规则以及预期的结果。
6. 如果尝试使用命令行处理器（CLP）对主机或 *System i* 数据库服务器发出 SQL 语句，是否没有成功？
  - 您是否遵循了将 CLP 绑定至主机或 *System i* 数据库服务器的过程？

## 初始连接后遇到的问题

提供下列问题作为起始点，以帮助缩小问题的范围。

1. 有任何特殊的或不常见的操作环境吗？
  - 这是新应用程序吗？
  - 正在使用新过程吗？
  - 最近是否执行了可能会影响系统的更改？例如，自从上次成功地运行了应用程序或方案之后，是否更改了某些软件产品或应用程序？
  - 对于应用程序，是使用哪个应用程序编程接口（API）来创建该应用程序的？
  - 是否有使用该软件或通信 API 的其他应用程序在用户系统上运行？
  - 最近安装了修订包吗？如果当用户试图在他们的操作系统上使用（或装入）一个自从安装以来从未使用过的功能部件时发生问题，那么应确定 IBM 的最新修订包，并在安装该功能部件后装入该修订包。
2. 以前发生过此错误吗？
  - 是否有关于先前错误状态的已记录的解决方案？
  - 谁是参与者？他们能否提供对可能的操作方法的深入见解？
3. 您探索过使用通信软件命令来返回有关网络的信息吗？
  - TCP/IP 可能会在使用 TCP/IP 命令和守护程序时检索到有用的信息。
4. 在 *SQLCA*（SQL 通信区）中是否返回了有帮助的信息？
  - 问题处理过程应该包括检查 *SQLCODE* 和 *SQLSTATE* 字段内容的步骤。
  - *SQLSTATE* 允许应用程序员对 DB2 系列数据库产品的常见错误类进行测试。在分布式关系数据库网络中，此字段可以提供一个公共基础。
5. 在服务器中执行了 *DB2START* 吗？另外，对于访问远程服务器的客户机，要确保正确地设置了 *DB2COMM* 环境变量。



6. 执行同一任务的其他机器能够成功与服务器相连吗？试图与服务器相连的客户机数目可能已经达到了最大值。如果另一个客户机与服务器断开连接，那么先前不能连接的客户机现在能连接吗？
7. 机器是否有正确的地址？验证该机器在网络中是否是唯一的。
8. 当远程连接时，已经为客户机授予了正确的权限吗？可能成功与实例进行了连接，但是，可能未在数据库级别或表级别授予权限。
9. 这是第一台与远程数据库进行连接的机器吗？在分布式环境中，网络之间的路由器或桥接器可能会阻塞客户机与服务器之间的通信。例如，当使用 TCP/IP 时，应确保可以对远程主机执行 PING。

## 诊断工具

当您遇到问题时，可以使用下列设施：

- 包括转储文件、陷阱文件、错误日志、通知文件和警报日志在内的所有诊断数据都在诊断数据目录路径（**diagpath**）数据库管理器配置参数指定的路径中：

如果此配置参数的值为 `null`，那么诊断数据将写入下列目录或文件夹之一：

- 对于 Linux 和 UNIX 环境：`INSTHOME/sqllib/db2dump`，其中 `INSTHOME` 是实例的主目录。
- 对于受支持的 Windows 环境：
  - 如果未设置 `DB2INSTPROF` 环境变量，那么将使用 `x:\SQLLIB\DB2INSTANCE`，其中 `x:\SQLLIB` 是驱动器引用和在 `DB2PATH` 注册表变量中指定的目录，而 `DB2INSTANCE` 的值为实例的名称。

注：该目录没必要命名为 `SQLLIB`。

- 如果设置了 `DB2INSTPROF` 环境变量，那么将使用 `x:\DB2INSTPROF\DB2INSTANCE`，其中 `DB2INSTPROF` 是实例概要文件目录的名称，`DB2INSTANCE` 是实例的名称（缺省情况下，在 Windows 32 位操作系统上为 `DB2INSTDEF` 的值）。
- 对于 Windows 操作系统，可以使用“事件查看器”来查看管理通知日志。
- 可以使用的变量诊断工具包括 `db2trc`、`db2pd` 和 `db2support`。
- 对于 Linux 和 UNIX 操作系统，`ps` 命令将关于活动进程的进程状态信息返回到标准输出中。
- 对于 UNIX 操作系统，核心文件是在发生服务器错误时，在当前目录中创建的。它包含已终止的进程的内存映像，可用于确定哪个功能导致该错误。

## DB2 Connect 常见问题

本主题列示使用 DB2 Connect 时遇到的常见的连接问题症状。在每种情况下，都为您提供：

- 消息号和与该消息相关联的返回码（或特定于协议的返回码）的组合。每个消息和返回码组合都具有独立的标题，这些标题是先按消息号，再按返回码来排序的。
- 症状，通常采用样本消息列表的格式。
- 建议的解决方案，指示错误的可能原因。在某些情况下，可能会提供多种建议的解决方案。

## SQL0965 或 SQL0969

**症状** 可以发出消息 SQL0965 和 SQL0969 并带来自 DB2 i5/OS 版、DB2 z/OS® 版以及 DB2 VM 和 VSE 版的许多不同返回码。

当您遇到任何其中一个消息时，应该在发出该消息的数据库服务器产品的文档中查找原始 SQL 代码。

### 解决方案

不能转换从主机或 i5/OS 数据库接收到的 SQL 代码。根据错误代码来更正该问题，然后重新提交失败的命令。

## SQL5043N

**症状** 未能成功启动对一个或多个通信协议的支持。但是，已成功启动了核心数据库管理器功能。

DB2 Connect 服务器上可能未启动 TCP/IP 协议。先前可能已经有成功的客户机连接。

如果 diaglevel = 4，那么 db2diag.log 可能包含类似的条目，例如：

```
2001-05-30-14.09.55.321092 Instance:svtdbm5 Node:000
PID:10296(db2tccpm) Appid:none
common_communication sqlcctcpconnmgr_child Probe:46
DIA3205E 在 TCP/IP services 文件中配置的且 TCP/IP
服务器支持所需的套接字地址"30090"正被另一进程使用。
```

### 解决方案

此警告是一种症状，它表示充当远程客户机的服务器的 DB2 Connect 在处理一个或多个客户机通信协议时遇到问题。这些协议可以是 TCP/IP 和其他协议，消息通常会指出为 DB2 Connect 定义的某个通信协议未正确配置。

原因通常是未定义 DB2COMM 概要文件变量，或者该变量未正确定义。该问题通常是 DB2COMM 变量与数据库管理器配置中定义的名称（例如，svcname 或 nname）之间不匹配的结果。

一种可能的情况是先前有成功的连接，然后得到 SQL5043 错误消息，但未更改任何配置。使用 TCP/IP 协议时，当远程系统因为某些原因而异常终止了连接，就可能会发生这种情况。发生这种情况时，客户机上可能仍然显示连接存在，通过发出下面所显示的命令，就可复原连接，而无须进一步的操作。

与 DB2 Connect 服务器相连接的某个客户机很有可能在 TCP/IP 端口上仍有句柄。在与 DB2 Connect 服务器相连接的每一台客户机上，输入下列命令：

```
db2 terminate      db2stop
```

## SQL30020

**症状** SQL30020N 执行失败，原因是“分布式协议错误”，该错误将影响后续命令和 SQL 语句的成功执行。

### 解决方案

遇到此错误时，应该与服务中心联系。与服务中心联系之前请先运行 db2support 命令。

## SQL30060

**症状** SQL30060N "<authorization-ID>" 没有执行操作 "<operation>" 的特权。

## 解决方案

连接至 DB2 OS/390® 版和 z/OS 版时，未正确地更新“通信数据库”（CDB）表。

## SQL30061

**症状** 连接至错误的主机或 System i 数据库服务器位置 - 找不到目标数据库。

## 解决方案

可能在 DCS 目录条目中指定了错误的服务器数据库名称。当发生此情况时，将对应用程序返回 SQLCODE -30061。

检查 DB2 节点、数据库和 DCS 目录条目。DCS 目录条目中的目标数据库名称字段必须与基于平台的数据库名称相对应。例如，对于 DB2 通用数据库™ z/OS 版和 OS/390 版数据库，要使用的名称应该与“引导数据集”（BSDS）“LOCATION=*locname*”字段中的名称相同，当启动了“分布式数据设施”（DDF）时，在 DSNL004I 消息（LOCATION=*location*）中也提供了该名称。

TCP/IP 节点的正确命令是：

```
db2 catalog tcpip node <node_name> remote <host_name_or_address>
      server <port_no_or_service_name>
db2 catalog dcs database <local_name> as <real_db_name>
db2 catalog database <local_name> as <alias> at <node node_name>
      authentication server
```

要连接至数据库，您应该发出：

```
db2 connect to <alias> user <user_name> using <password>
```

## 带有返回码 79 的 SQL30081N

### 症状

SQL30081N 已检测到通信错误。  
所使用的通信协议: "TCP/IP"。  
所使用的通信 API: "SOCKETS"。  
位置 检测到错误的位置: ""。错误的  
通信功能:  
"连接"。特定于协议的错误代码: "79"、"\*"和"\*"。  
SQLSTATE=08001

## 解决方案

当远程客户机未能与 DB2 Connect 服务器连接时，就可能发生此错误。当从 DB2 Connect 服务器连接至主机或 System i 数据库服务器时，也可能发生此错误。

1. 在 DB2 Connect 服务器上，DB2COMM 概要文件变量可能设置得不正确。检查此变量。例如，当在 AIX 上运行 DB2 企业服务器版时，命令 db2set db2comm=tcpip 应该出现在 sqllib/db2profile 中。
2. IBM 数据服务器客户机 中指定的 TCP/IP 服务名称和端口号可能与 DB2 Connect 服务器中指定的不匹配。在以上两种机器中验证 TCP/IP services 文件中的各个条目。
3. 检查在 DB2 Connect 服务器上是否启动了 DB2。使用以下命令将“数据库管理器配置”diaglevel 设置为 4:

```
db2 update dbm cfg using diaglevel 4
```

在停止并重新启动 DB2 后，查看 db2diag.log 文件，以便检查是否已经启动了 DB2 TCP/IP 通信。您应该看到与下列信息类似的输出：

```
2001-02-03-12.41.04.861119 Instance:svtdbm2 Node:00
PID:86496(db2sysc) Appid:none
common_communication sqlcctcp_start_listen Probe:80
DIA3000I "TCPIP"协议支持已成功启动。
```

## 带有特定于协议的错误代码 10032 的 SQL30081N

### 症状

```
SQL30081N 已检测到通信错误。
    所使用的通信协议: "TCP/IP"。
    所使用的通信 API: "SOCKETS"。
位置 检测到错误的位置: "9.21.85.159"。 检测到
    错误的通信功能:
    "发送"。特定于协议的错误代码: "10032"、"*"、"*"。
SQLSTATE=08001
```

### 解决方案

当试图与 TCP/IP 通信已经发生故障的机器断开连接时，可能会接收到此错误消息。用 TCP/IP 子系统更正该问题。

在大多数机器上，只须对机器重新启动 TCP/IP 协议就可以更正该问题。有时可能需要重新启动整个机器。

## 连接 (CONNECT) 期间出现 SQL30082 RC=24

**症状** SQLCODE -30082 提供的用户名或密码不正确。

### 解决方案

确保在 CONNECT 语句上提供了正确的密码（必要时）。未提供要发送到目标服务器数据库的密码。必须将密码从 IBM 数据服务器客户机发送到目标服务器数据库中。在某些平台上，例如 AIX，仅当密码是在 CONNECT 语句中提供的时候，才能获得该密码。

---

## 第 4 章 用于故障诊断的工具

当对问题进行故障诊断时，数据库管理器内部的返回码、作为 DB2 产品一部分的工具、不同类型的跟踪以及作为操作系统一部分的工具都将会被使用。当调查数据库问题时，每个工具均会提供数据和信息来帮助您和 DB2 支持人员。

---

### 了解更多关于内部返回码的内容

有两种类型的内部返回码：ZRC 值和 ECF 值。这些返回码一般仅在供 IBM 软件支持机构使用的诊断工具中可视。例如，它们出现在 DB2 跟踪输出和 db2diag.log 文件中。

ZRC 和 ECF 值基本上具有相同的作用，但格式上稍有不同。每个 ZRC 值都具有以下特征：

- 类名
- 组件
- 原因码
- 相关联的 SQLCODE
- SQLCA 消息标记
- 描述

但是，ECF 值由以下部分组成：

- 集名
- 产品标识
- 组件
- 描述

ZRC 和 ECF 值通常为负数，并用于表示错误状况。ZRC 值根据它们表示的错误类型进行分组。这些分组称为“类”。例如，具有以“SQLZ\_RC\_MEMHEP”开头的名称的 ZRC 值通常是与内存不足相关的错误。相似地，ECF 值被分组为“集”。

以下是包含 ZRC 值的 db2diag.log 条目的示例：

```
2006-02-13-14.34.35.965000-300    I17502H435    级别: 错误
PID: 940                        TID: 660      PROC: db2syscs.exe
实例: DB2                        节点: 000     数据库: SAMPLE
APPHDL: 0-1433                   APPID: *LOCAL.DB2.050120082811
函数: DB2 UDB, 数据包含, sqlpsize, 探测点: 20
返回码: ZRC=0x860F000A=-2045837302=SQL0_FNEX"找不到文件。"
        DIA8411C 找不到文件 ""。
```

可使用 db2diag 命令获取有关此 ZRC 值的完整详细信息，例如：

```
c:\>db2diag -rc 0x860F000A
```

输入 ZRC 字符串"0x860F000A"被解析为 0x860F000A (-2045837302)。

要映射的 ZRC 值: 0x860F000A (-2045837302)  
V7 等同 ZRC 值: 0xFFFFE60A (-6646)

ZRC 类:  
关键介质错误 (类索引: 6)

组件: SQL0; 操作系统服务 ( 组件索引: 15 )  
原因码: 10 ( 0x000A )  
标识: SQL0\_FNEX  
SQL0\_MOD\_NOT\_FOUND  
标识 ( 不带组件 ): SQLZ\_RC\_FNEX  
描述: 未找到文件。  
相关信息: Sqlcode -980  
SQL0980C 发生磁盘错误。无法处理后续的 SQL 语句。  
sqlca 标记数: 0  
对话框消息号: 8411

如果发出命令 `db2diag -rc -2045837302` 或 `db2diag -rc SQL0_FNEX`, 将返回相同信息。

ECF 返回码的输出示例如下:

```
c:\>db2diag -rc 0x90000076
```

输入 ECF 字符串"0x90000076"被解析为 0x90000076 ( -1879048074 )

要映射的 ECF 值: 0x90000076 ( -1879048074 )

ECF 集: setecf ( 集索引: 1 )

产品: DB2 Common

组件: OSSe

代码: 118 ( 0x0076 )

标识: ECF\_LIB\_CANNOT\_LOAD

描述: 不能装入指定的库。

`db2diag` 命令输出中最有价值的故障诊断信息是描述和相关信息 ( 仅对于 ZRC 返回码 )。

要获取 ZRC 或 ECF 值的完整列表, 请分别使用命令 `db2diag -rc zrc` 和 `db2diag -rc ecf`。

---

## db2dart 工具概述

可以使用 `db2dart` 命令来验证数据库及其对象的体系结构是否正确。还可以使用它来显示数据库控制文件的内容, 以便从其他情况下可能无法访问的表中抽取数据。

要显示所有可能的选项, 只需发出不带任何参数的 `db2dart` 命令。如果命令行中未显式指定一些需要参数的选项 ( 如表空间标识 ), 那么会提示输入这些参数。

缺省情况下，db2dart 实用程序将创建名为 databaseName.RPT 的报告文件。对于单一分区数据库分区环境，将在当前目录中创建该文件。对于多分区数据库分区环境，将在诊断目录的子目录中创建该文件。该子目录称为 DART####，其中 #### 是数据库分区号。

db2dart 实用程序通过直接从磁盘中读取数据库中的数据和元数据来访问它们。因此，决不能对仍具有活动连接的数据库运行该工具。如果存在活动连接，那么该工具将不知道缓冲池中的页面或内存中的控制结构（此处是举例说明），可能会报告假的错误结果。同样，如果对需要进行崩溃恢复或尚未完成前滚恢复的数据库运行 db2dart，由于磁盘上的数据性质不一致，可能会导致类似的不一致情况。

## 比较 INSPECT 和 db2dart

INSPECT 命令允许检查数据库的体系结构完整性，并检查数据库页面以了解页面是否一致。INSPECT 命令会检查表对象的结构和表空间的结构是否有效。交叉对象验证会创建索引与数据一致性的联机检查。db2dart 命令会检查数据库以了解体系结构是否正确，并报告遇到的所有错误。

INSPECT 命令与 db2dart 命令的相似之处在于它允许您检查数据库、表空间和表。这两个命令之间的主要差别是，在运行 db2dart 之前需要取消激活数据库，而 INSPECT 需要与数据库连接，并且可以在同时有多个活动数据库连接时运行。

如果不取消激活数据库，那么 db2dart 将产生不可靠的结果。

下列各表列示了 db2dart 命令和 INSPECT 命令执行的测试之间的差别。

表 1. 针对表空间比较 db2dart 和 INSPECT 的功能

执行的测试	db2dart	INSPECT
<b>SMS 表空间</b>		
检查表空间文件	YES	NO
验证内部页标题字段的内容	YES	YES
<b>DMS 表空间</b>		
检查多个对象指向的扩展数据块映像	YES	NO
检查每个扩展数据块映像页以找出一致性位错误	NO	YES
检查每个空间映射页以找出一致性位错误	NO	YES
验证内部页标题字段的内容	YES	YES
验证扩展数据块映像是否与表空间映射一致	YES	NO

表 2. 针对数据对象比较 db2dart 和 INSPECT 的功能

执行的测试	db2dart	INSPECT
检查数据对象以找出一致性位错误	YES	YES
检查特殊控制行的内容	YES	NO
检查可变长度列的长度和位置	YES	NO



表 2. 针对数据对象比较 *db2dart* 和 *INSPECT* 的功能 (续)

执行的测试	<b>db2dart</b>	<b>INSPECT</b>
检查表行中的 LONG VARCHAR、LONG VARCHAR 和大对象 (LOB) 描述符	YES	NO
检查总计页数、已使用的页数和可用空间百分比	NO	YES
验证内部页标题字段的内容	YES	YES
验证每行的记录类型及其长度	YES	YES
验证行是否未重叠	YES	YES

表 3. 针对索引对象比较 *db2dart* 和 *INSPECT* 的功能

执行的测试	<b>db2dart</b>	<b>INSPECT</b>
检查一致性位错误	YES	YES
检查索引键的位置和长度以及是否存在重叠	YES	YES
检查索引中键的顺序	YES	NO
确定总计页数和已使用的页数	NO	YES
验证内部页标题字段的内容	YES	YES
验证唯一键的唯一性	YES	NO
检查给定索引条件的数据行是否存在	NO	YES
验证数据值的每个键	NO	YES

表 4. 针对块映射对象比较 *db2dart* 和 *INSPECT* 的功能

执行的测试	<b>db2dart</b>	<b>INSPECT</b>
检查一致性位错误	YES	YES
确定总计页数和已使用的页数	NO	YES
验证内部页标题字段的内容	YES	YES

表 5. 针对长整型字段和 LOB 对象比较 *db2dart* 和 *INSPECT* 的功能

执行的测试	<b>db2dart</b>	<b>INSPECT</b>
检查分配结构	YES	YES
确定总计页数和已使用的页数 (仅适用于 LOB 对象)	NO	YES

此外, 还可以使用 *db2dart* 命令执行下列操作:

- 格式化和转储数据页
- 格式化和转储索引页
- 将数据行格式化为定界 ASCII
- 将索引标记为无效

*INSPECT* 命令不能用于执行这些操作。

## 使用 db2diag 分析 db2diag.log 文件

供数据库和系统管理员使用的主日志文件为管理通知日志。db2diag.log 文件可供 DB2 支持机构用于进行故障诊断。

还会使用标准化消息格式将管理通知日志消息记录至 db2diag.log。

db2diag 工具用于对 db2diag.log 中提供的大量信息进行过滤和格式化。过滤 db2diag.log 记录可缩短诊断问题时查找所需记录的时间。

**示例 1:** 按数据库名称过滤 db2diag.log:

如果实例中有若干数据库，并且您只希望显示与数据库“SAMPLE”有关的信息，那么可以按如下所示过滤 db2diag.log:

```
db2diag -g db=SAMPLE
```

因此，将仅显示包含“DB: SAMPLE”的 db2diag.log 记录，如:

```
2006-02-15-19.31.36.114000-300 E21432H406          级别: 错误
PID: 940                TID: 660          PROC: db2syscs.exe
实例: DB2                节点: 000          数据库: SAMPLE
APPHDL: 0-1056          APPID: *LOCAL.DB2.060216003103
函数: DB2 UDB, 基本系统实用程序, sqliteDatabaseQuiesce, 探测点: 2
消息: ADM7507W 数据库停顿请求成功完成。
```

**示例 2:** 按进程标识过滤 db2diag.log:

以下命令可用于显示进程标识 (PID) 为 2200，并且在分区 0、1、2 或分区 3 上运行的进程生成的所有严重错误消息:

```
db2diag -g level=Severe,pid=2200 -n 0,1,2,3
```

注意，此命令可能以不同方式写入，包括 db2diag -l severe -pid 2200 -n 0,1,2,3。还要注意，-g 选项指定区分大小写的搜索，所以此处“Severe”会起作用，但如果使用了“severe”则会失败。满足如下要求时，这些命令将成功检索 db2diag.log 记录:

```
2006-02-13-14.34.36.027000-300 I18366H421          级别: 严重
PID      : 2200                TID : 660          PROC : db2syscs.exe
实例: DB2                节点: 000          数据库: SAMPLE
APPHDL: 0-1433          APPID: *LOCAL.DB2.060213193043
函数: DB2 UDB, 数据管理, sqlPoolCreate, 探测点: 273
返回码: ZRC=0x8002003C=-2147352516=SQLB_BAD_CONTAINER_PATH
          "错误的容器路径"
```

**示例 3:** 格式化 db2diag 工具输出。

以下命令过滤 2006 年 1 月 1 日之后发生，并且包含分区 0、1 或 2 上记录的所有非严重错误和严重错误的所有记录。它会输出匹配的记录，因此时间戳记、分区号和级别将出现在第一行上，而 pid、tid 和实例名将出现在第二行上，之后是错误消息:

```
db2diag -time 2006-01-01 -node "0,1,2" -level "Severe, Error" |
db2diag -fmt "Time: %{ts}
分区: %node Message Level: %{level} \nPid: %{pid} Tid: %{tid}
实例: %{instance}\nMessage: @{msg}\n"
```

生成的输出示例如下所示:

```
时间: 2006-02-15-19.31.36.099000 分区: 000 消息级别: 错误
Pid: 940 Tid: 40 实例: DB2
消息: ADM7506W 已经请求了数据库停顿。
```

有关更多信息，请发出下列命令：

- `db2diag -help` 提供了所有可用选项的简短描述
- `db2diag -h brief` 提供对所有不带示例的选项的描述
- `db2diag -h notes` 提供用法说明和限制
- `db2diag -h examples` 提供一小组示例以帮助您入门
- `db2diag -h tutorial` 提供所有可用选项的示例
- `db2diag -h all` 提供最完整的选项列表

**示例 4：** 过滤来自不同工具的消息。

下列示例显示如何仅查看来自数据库管理器中的特定工具（或所有工具）的消息。受支持的工具包括：

- ALL，这会返回来自所有工具的记录
- MAIN，这会返回来自 DB2 常规诊断日志的记录，如 `db2diag.log` 和管理通知日志
- OPSTATS，这会返回与优化器统计信息有关的记录

要读取来自 MAIN 工具的消息，请使用以下命令：

```
db2diag -facility MAIN
```

要显示来自 OPSTATS 工具的消息并滤出级别为 Severe 的记录，请使用以下命令：

```
db2diag -fac OPSTATS -level Severe
```

要显示来自所有可用工具的消息并滤出实例为 harmistr 级别为 Error 的记录，请使用以下命令：

```
db2diag -fac all -g instance=harmistr,level=Error
```

要显示 OPSTATS 工具中级别为 Error 并且以特定格式输出时间戳记和 PID 字段的所有消息，请使用以下命令：

```
db2diag -fac opstats -level Error -fmt " Time :%{ts} Pid :%{ts}"
```

---

## 使用 db2greg 显示和改变全局注册表（UNIX）

全局注册表只存在于UNIX 和 Linux 平台上：

- 对于 root 安装，全局注册表文件为 `/var/db2/global.reg`（HP-UX 上为 `/var/opt/db2/global.reg`）。
- 对于非 root 安装，全局注册表文件为 `$HOME/sqllib/global.reg`，其中 `$HOME` 是非 root 用户的主目录。

全局注册表由以下三种不同记录类型组成：

- “服务”：服务记录包含产品级别的信息，如版本和安装路径。
- “实例”：实例记录包含实例级别的信息，如实例名、实例路径、版本和“引导时启动”标志。
- “变量”：变量记录包含变量级别的信息，如变量名、变量值和注释。

可使用 `db2greg` 工具查看全局注册表。此工具位于 `sqllib/bin` 和 `bin` 下的 `install` 目录中（以供作为 root 用户登录时使用）。

可使用db2greg 工具编辑全局注册表。在 root 安装中编辑全局注册表需要 root 权限。

如果DB2 客户支持机构要求使用 db2greg 工具，那么应该使用该工具。

---

## 标识产品的版本和服务级别

db2level 命令帮助您确定 DB2 实例的版本和服务级别（构件级别和修订包级别）。要确定 DB2 实例是否为最新服务级别，比较 db2level 输出和 DB2 支持机构 Web 站点 <http://www.ibm.com/software/data/db2/udb/support.html> 上的修订包下载页面中的信息。

在 Windows 系统上运行 db2level 命令的典型结果为：

```
DB21085I 实例"DB2"使用"32"位和 DB2 代码发行版"SQL09010"，其级别标识为"01010107"。
```

```
参考标记为"DB2 v9.1.0.189"、"n060119"、""和修订包"0"。  
产品安装在"c:\SQLLIB"中，DB2 副本名称为"db2build"。
```

这四个参考标记的组合唯一地标识 DB2 实例的精确服务级别。在与 IBM 支持机构联系以获取帮助时，此信息非常重要。

对于 JDBC 或 SQLJ 应用程序，如果要使用用于 SQLJ 和 JDBC 的 IBM DB2 驱动程序，可通过运行 db2jcc 实用程序来确定驱动程序的级别：

```
db2jcc -version
```

```
IBM DB2 JDBC Driver Architecture 2.3.63
```

---

## 使用 db2look 模拟数据库

如果能够创建结构类似另一数据库的数据库，有时会非常有利。例如，与在生产系统上测试新应用程序或恢复计划相比，创建结构和数据类似的测试系统然后对其进行测试更有意义。这样生产系统不会受到测试的负面性能影响，或者因为错误的应用程序导致的意外数据毁坏而造成的影响。而且，在调查问题（如无效结果、性能问题等）时，会更容易在与生产系统完全相同的测试系统上调试该问题。

可使用 db2look 工具来抽取必需的 DDL 语句，这些语句是在一个数据库中再现另一个数据库中的数据库对象所需的。该工具还可生成将统计信息从一个数据库复制至另一个数据库所需的 SQL 语句，以及复制数据库配置、数据库管理器配置和注册表变量所需的语句。因为新数据库可能未包含与原始数据库完全相同的一组数据，但您仍然要对两个系统选择相同的访问计划，所以这一点是非常重要的。

db2look 工具在 *DB2 Command Reference* 中作了详细描述，但您可以通过执行该工具（不带任何参数）来查看选项列表。可使用 -h 选项来显示更多详细用法。

### 使用 db2look 来模拟数据库中的表

要在数据库中抽取表的 DDL，使用 -e 选项。例如，创建 SAMPLE 数据库的副本 SAMPLE2，这样会在新数据库中创建第一个数据库中的所有对象。

```
C:\>db2 create database sample2  
DB20000I 成功完成了 CREATE DATABASE 命令。  
C:\>db2look -d sample -e > sample.ddl  
-- 用户为:  
-- 为表创建 DDL
```

```
-- 正在自动绑定程序包 ...
-- 绑定成功
-- 正在自动绑定程序包 ...
-- 绑定成功
```

**注:** 如果还想对用户定义的空间、数据库分区组和缓冲池生成 DDL, 那么在上述命令中 -e 的后面添加 -l 标志。将不会抽取缺省数据库分区组、缓冲池和表空间。这是因为在缺省情况下它们已经存在于每个数据库中。如果希望模拟它们, 那么必须手动对其进行更改。

在文本编辑器中打开 sample.ddl 文件。因为您要对新数据库执行此文件中的 DDL, 所以必须将 CONNECT TO SAMPLE 语句更改为 CONNECT TO SAMPLE2。如果使用了 -l 选项, 那么可能需要改变与表空间命令相关联的路径, 这样它们也会指向适当的路径。请抽空查看文件内容的余下部分。您应该会看到样本数据库中的所有用户表的 CREATE TABLE、ALTER TABLE 和 CREATE INDEX 语句:

```
...
-----
-- 表 "DB2"."ORG" 的 DDL 语句
-----

CREATE TABLE "DB2"."ORG" (
  "DEPTNUMB" SMALLINT NOT NULL ,
  "DEPTNAME" VARCHAR(14) ,
  "MANAGER" SMALLINT ,
  "DIVISION" VARCHAR(10) ,
  "LOCATION" VARCHAR(13) )
  IN "USERSPACE1" ;
...

```

更改连接语句后, 执行以下语句:

```
C:\>db2 -tvf sample.ddl > sample2.out
```

查看 sample2.out 输出文件 - 每一项都应该已经成功执行。如果发生了错误, 错误消息应指示问题所在。修正这些问题并再次执行这些语句。

就像您在输出中看到的那样, 将导出所有用户表的 DDL。这是缺省行为, 但对于包括的表, 还有更具体的其他选项可用。例如, 如果希望仅包括 STAFF 和 ORG 表, 那么使用 -t 选项:

```
C:\>db2look -d sample -e -t staff org > staff_org.ddl
```

如果希望仅包括带有模式 DB2 的表, 那么使用 -z 选项:

```
C:\>db2look -d sample -e -z db2 > db2.ddl
```

## 模拟表的统计信息

如果测试数据库用于测试性能或调试性能问题, 那么对两个数据库生成的访问方案应该完全相同。优化器根据统计信息、配置参数、注册表变量和环境变量来生成访问方案。如果两个系统间的这些设置完全相同, 那么访问方案很可能是相同的。

如果将完全相同的数据装入到两个数据库中, 并且对它们两个执行相同的 RUNSTATS 选项, 那么统计信息也应该完全相同。但是, 如果数据库包含不同数据, 或者如果测试数据库中只使用数据的子集, 那么统计信息可能会有显著差别。在这种情况下, 可

使用 db2look 来从生产数据库收集统计信息并将它们放在测试数据库中。通过对可更新表的 SYSSTAT 集合创建 UPDATE 语句并对所有表创建 RUNSTATS 命令来执行此操作。

用于创建统计信息语句的选项为 -m。返回至 SAMPLE/SAMPLE2 示例，从 SAMPLE 收集统计信息并将其添加至 SAMPLE2:

```
C:\>db2look -d sample -m > stats.dml
-- 用户为:
-- 以模拟方式运行 db2look
```

像以前一样，必须编辑输出文件以便将 CONNECT TO SAMPLE 语句更改为 CONNECT TO SAMPLE2。而且再查看一下文件内容的余下部分，以了解某些 RUNSTATS 和 UPDATE 语句的情况:

```
...
-- 模拟表 ORG
RUNSTATS ON TABLE "DB2"."ORG" ;
UPDATE SYSSTAT.INDEXES
SET NLEAF=-1,
    NLEVELS=-1,
    FIRSTKEYCARD=-1,
    FIRST2KEYCARD=-1,
    FIRST3KEYCARD=-1,
    FIRST4KEYCARD=-1,
    FULLKEYCARD=-1,
    CLUSTERFACTOR=-1,
    CLUSTERRATIO=-1,
    SEQUENTIAL_PAGES=-1,
    PAGE_FETCH_PAIRS='',
    DENSITY=-1,
    AVERAGE_SEQUENCE_GAP=-1,
    AVERAGE_SEQUENCE_FETCH_GAP=-1,
    AVERAGE_SEQUENCE_PAGES=-1,
    AVERAGE_SEQUENCE_FETCH_PAGES=-1,
    AVERAGE_RANDOM_PAGES=-1,
    AVERAGE_RANDOM_FETCH_PAGES=-1,
    NUMRIDS=-1,
    NUMRIDS_DELETED=-1,
    NUM_EMPTY_LEAFS=-1
WHERE TABNAME = 'ORG' AND TABSCHEMA = 'DB2  ';
...
```

使用抽取 DDL 的 -e 选项的同时，可使用 -t 和 -z 选项来指定一组表。

## 抽取配置参数和环境变量

优化器根据统计信息、配置参数、注册表变量和环境变量来选择计划。可将统计信息与 db2look 一起使用来生成必需的配置更新和设置语句。此操作使用 -f 选项完成。例如:

```
c:\>db2look -d sample -f>config.txt
-- 用户为: DB2INST1
-- 正在自动绑定程序包 ...
-- 绑定成功
-- 正在自动绑定程序包 ...
-- 绑定成功
```

config.txt 包含类似如下的输出:

```
-- 此 CLP 文件是使用 DB2LOOK 版本 9.1 创建的
-- 时间戳记: 2/16/2006 7:15:17 PM
-- 数据库名称: SAMPLE
```

```

-- 数据库管理器版本: DB2/NT 版本 9.1.0
-- 数据库代码页: 1252
-- 数据库整理顺序为: UNIQUE

CONNECT TO SAMPLE;

-----
-- 数据库和数据库管理器配置参数
-----

UPDATE DBM CFG USING cpuspeed 2.991513e-007;
UPDATE DBM CFG USING intra_parallel NO;
UPDATE DBM CFG USING comm_bandwidth 100.000000;
UPDATE DBM CFG USING federated NO;

...

-----
-- 环境变量设置
-----

COMMIT WORK;

CONNECT RESET;

```

注: 将仅包括影响 DB2 编译器的那些参数和变量。如果影响编译器的注册表变量设置为其缺省值, 那么它不会显示在“环境变量设置”下面。

---

## 列出系统上安装的 DB2 产品 (Linux 和 UNIX)

root 用户必须至少安装了一个 DB2 版本 9 产品, /usr/local/bin 目录中才会存在与 db2ls 命令的符号链接。

由于能够在系统上安装 DB2 产品的多个副本, 并且能够灵活地在您选择的路径中安装 DB2 产品和功能部件, 所以需要使用一个工具来帮助跟踪已经安装了哪些 DB2 产品及其安装位置。在受支持的 Linux 和 UNIX 操作系统上, db2ls 命令将列示安装在系统上的 DB2 产品和功能部件, 还会列示 DB2 版本 9 HTML 文档。

可以使用 db2ls 命令来列示:

- DB2 产品在系统上的安装位置, 并且会列示 DB2 产品级别
- 特定安装路径中的所有或特定 DB2 产品和功能部件

根据所使用的标识, db2ls 命令列示的输出会有所不同:

- 当使用 root 用户权限运行 db2ls 命令时, 仅查询 root 用户 DB2 安装。
- 当使用非 root 用户标识运行 db2ls 命令时, 将查询 root 用户 DB2 安装和匹配的非 root 用户标识拥有的非 root 用户安装。不查询由其他非 root 用户标识拥有的 DB2 安装。

db2ls 命令是可用来查询 DB2 产品的唯一方法。您不能使用 Linux 或 UNIX 操作系统本机实用程序 (例如, pkginfo、rpm、SMIT 或 swlist) 来查询 DB2 产品。包含用于查询 DB2 安装并与其进行交互的本机安装实用程序的任何现有脚本将需要进行更改。

在 Windows 操作系统上不能使用 db2ls 命令。



要列示 DB2 产品在系统上的安装路径和 DB2 产品级别，请输入以下命令：

```
db2ls
```

该命令将列示安装在系统上的每个 DB2 产品的下列信息：

- 安装路径
- 级别
- 修订包
- 特殊安装编号。此列由 IBM DB2 支持人员使用。
- 安装日期。此列显示上次修改 DB2 产品的时间。
- 安装程序用户标识。此列显示用于安装 DB2 产品的用户标识。

要列示有关特定安装路径中的 DB2 产品或功能部件的信息，必须指定 *q* 参数：

```
db2ls -q -p -b baseInstallDirectory
```

其中：

- *q* 指定您要查询产品或功能部件。此参数是必需的。如果查询了 DB2 版本 8 产品，那么会返回空白值。
- *p* 指定列表显示产品而不是列示功能部件。
- *b* 参数指定产品或功能部件的安装目录。此参数是固定的，如果不是从安装目录运行该命令。

根据所提供的参数，该命令将列示以下信息：

- 安装路径。安装路径只需指定一次，无须对每个功能部件都指定它。
- 将显示以下信息：
  - 已安装功能部件的响应文件标识，或者如果指定了 *p* 选项，已安装产品的响应文件标识。例如，ENTERPRISE\_SERVER\_EDITION。
  - 功能部件名称，或者如果指定了 *p* 选项，产品名。
  - 产品版本、发行版、修改级别和修订包级别（VRMF）。例如，9.5.0.0
  - 修订包（如果有的话）。例如，如果安装了修订包 1，那么显示的值是 1。这包括临时修订包，例如修订包 1a。
- 如果任何产品的 VRMF 信息不匹配，在输出列表末尾都将显示一条警告消息。该消息建议应用此修订包。

---

## 使用 db2pd 进行监视和故障诊断

因为 db2pd 工具可从 DB2 内存集合迅速返回即时信息，所以该工具可用于故障诊断。

该工具不需要获得任何锁存器或使用任何引擎资源就可以收集信息。因此，在 db2pd 收集信息时，有可能（并且预计）会检索到正在更改的信息；这样，数据可能不是十分准确。如果遇到正在更改的内存指针，可使用信号处理程序来防止 db2pd 异常终止。这可能会导致输出中出现诸如以下的消息：“正在更改的数据结构已强制终止命令”。虽然如此，该工具对于故障诊断却非常有用。在不锁存的情况下收集信息有两个好处：检索速度更快并且不会争用引擎资源。

如果要在出现特定 SQLCODE、ZRC 代码或 ECF 代码时捕获关于数据库管理系统的信息，那么可以使用 db2pdcfg -catch 命令完成此操作。捕获到错误时，将启动 db2cos（调

出脚本)。db2cos 文件可以自动改变，以便运行解决问题所需的任何 db2pd 命令、操作系统命令或任何其他命令。在 UNIX 和 Linux 上，模板文件 db2cos 位于 sqllib/bin 中。在 Windows 操作系统上，db2cos 位于 \$DB2PATH\bin 目录中。

以下是使用 db2pd 快速故障诊断的一组示例。

### 场景 1: 诊断锁定等待。

使用 db2pd -db <database name> -locks -transactions -applications -dynamic 命令来获取下列结果:

锁定:

Address	TranHdl	Lockname	Type	Mode	Sts	Owner	Dur	HldCnt
Att ReleaseFlg								
0x07800000202E5238	3	00020002000000040000000052	Row	..X	G	3	1	0
0x0000 0x40000000								
0x07800000202E4668	2	00020002000000040000000052	Row	..X	W*	2	1	0
0x0000 0x40000000								

对于使用 -db 数据库名称选项指定的数据库，开头的结果会显示该数据库的锁定。您会发现 TranHdl 2 正在等待 TranHdl 3 挂起的锁定。

事务:

Address	AppHandl	[nod-index] TranHdl	Locks	State	Tflag	Tflag2
FirstIsn	LastIsn	LogSpace	SpaceReserved	TID	AxRegCnt	GXID
0x0780000020251B80	11	[000-00011]	2	4	READ	0x00000000 0x00000000
0x000000000000	0x000000000000	0	0		0x0000000000B7	1 0
0x0780000020252900	12	[000-00012]	3	4	WRITE	0x00000000 0x00000000
0x000000FA000C	0x000000FA000C	113	154		0x0000000000B8	1 0

您会发现 TranHdl 2 与 AppHandl 11 相关联，而 TranHdl 3 与 AppHandl 12 相关联。

应用程序:

Address	AppHandl	[nod-index] NumAgents	CoorPid	Status
C-AnchID	C-StmtUID	L-AnchID	L-StmtUID	Appid
0x07800000006879E0	12	[000-00012]	1	1073336 UOW-Waiting
0	0	17	1	*LOCAL.jmcmahon.060303225602
0x0780000000685E80	11	[000-00011]	1	1040570 UOW-Executing
17	1	94	1	*LOCAL.jmcmahon.060303225601

您会发现 AppHandl 12 最后运行动态语句 17, 1。AppHandl 11 是当前正在运行的动态语句 17, 1，而最后运行的语句是 94, 1。

动态 SQL 语句:

Address	AnchID	StmtUID	NumEnv	NumVar	NumRef	NumExe	Text
0x07800000209FD800	17	1	1	1	2	2	update pdtest set c1 = 5
0x07800000209FCCC0	94	1	1	1	2	2	set lock mode to wait 1

您会发现，文本列显示与锁定超时相关联的 SQL 语句。

### 场景 2: 在考虑锁定问题时使用调出脚本

查找 db2cos 输出文件。该文件的位置由数据库管理器配置参数 DIAGPATH 控制。输出文件的内容将随您在 db2cos 文件中输入的命令不同而不同。当 db2cos 文件包含 db2pd -db sample -locks 命令时，提供的输出示例如下所示:

```

捕获到锁定超时
Thu Feb 17 01:40:04 EST 2006
实例 DB2 数据库: SAMPLE
分区号: 0
PID: 940
TID: 2136

```

```

函数: sqlplnfd
组件: 锁管理器
探测点: 999
时间戳记: 2006-02-17-01.40.04.106000
AppID: *LOCAL.DB2...
AppHdl:
...
数据库分区 0 -- 数据库 SAMPLE -- 活动 -- 正常运行 0 天 00:06:53
锁定:
Address TranHdl Lockname Type Mode Sts Owner Dur HldCnt Att Rlse
0x402C6B30 3 00020003000000040000000052 Row ..X W* 3 1 0 0 0x40

```

仅查找“W\*”，因为这是经历超时的锁定。当锁定转换为更高级的方式时，也会发生锁定超时。在这种情况下，您只会在输出中见到“C\*”，而不会见到“W\*”。但是，在此特定情况下发生了锁定等待。可使用 db2cos 文件中的其他 db2pd 命令提供的输出将结果映射至事务、应用程序、代理程序甚至是 SQL 语句。可以缩小输出范围或使用其他命令来收集需要的信息。例如，可以更改 db2pd 命令选项以使用 -locks wait 选项，后一个选项仅打印具有等待状态的锁定。如果需要 -app 和 -agent 选项，也可以输入它们。

### 场景 3: 将应用程序映射至动态 SQL 语句

db2pd -applications 命令报告动态 SQL 语句的当前和最后一个锚点标识和语句唯一标识。这允许直接从应用程序映射至动态 SQL 语句。

```
db2pd -app -dyn
```

```

应用程序:
Address          AppHdl [nod-index] NumAgents  CoorPid Status
0x00000002006D2120 780 [000-00780] 1 10615 UOW-Executing

C-AnchID C-StmtUID L-AnchID L-StmtUID Appid
163 1 110 1 *LOCAL.jmcmahon.050202200412

```

```

动态 SQL 语句:
Address          AnchID StmtUID NumEnv NumVar NumRef NumExe
Text
0x0000000220A02760 163 1 2 2 1 CREATE VIEW MYVIEW
0x0000000220A0B460 110 1 2 2 1 CREATE VIEW YOURVIEW

```

### 场景 4: 监视内存使用情况。

尝试了解内存使用情况时，db2pd -memblock 命令非常有用。例如：

DBMS 集中的所有内存块。

```

地址          池标识      池名称      块存活时间 大小 (字节) I LOC 文件
0x0780000000740068 62      resynch     2          112      1 1746 1583816485
0x07800000000725688 62      resynch     1          108864   1 127 1599127346
0x078000000001F4348 57      ostrack     6          5160048 1 3047 698130716
0x078000000001B5608 57      ostrack     5          240048   1 3034 698130716
0x078000000001A0068 57      ostrack     1          80       1 2970 698130716
0x078000000001A00E8 57      ostrack     2          240      1 2983 698130716
0x078000000001A0208 57      ostrack     3          80       1 2999 698130716
0x078000000001A0288 57      ostrack     4          80       1 3009 698130716
0x07800000000700068 70      apmh        1          360      1 1024 3878879032
0x078000000007001E8 70      apmh        2          48       1 914 1937674139
0x07800000000700248 70      apmh        3          32       1 1000 1937674139
...

```

接下来是已排序的“性能池”输出：

```

按 ostrack 池的大小排序的内存块:
池标识      池名称      总大小 (字节)      总计数      LOC      文件
57          ostrack     5160048            1           3047     698130716

```

57	ostrack	240048	1	3034	698130716
57	ostrack	240	1	2983	698130716
57	ostrack	80	1	2999	698130716
57	ostrack	80	1	2970	698130716
57	ostrack	80	1	3009	698130716

ostrack 池的总大小: 5400576 字节

按 apmh 池的大小排序的内存块:

池标识	池名称	总大小 (字节)	总计数	LOC	文件
70	apmh	40200	2	121	2986298236
70	apmh	10016	1	308	1586829889
70	apmh	6096	2	4014	1312473490
70	apmh	2516	1	294	1586829889
70	apmh	496	1	2192	1953793439
70	apmh	360	1	1024	3878879032
70	apmh	176	1	1608	1953793439
70	apmh	152	1	2623	1583816485
70	apmh	48	1	914	1937674139
70	apmh	32	1	1000	1937674139

apmh 池的总大小: 60092 字节

...

最后一部分输出对整个 DBMS 集的内存使用者进行排序:

DBMS 内存集中的所有内存使用者:

池标识	池名称	总大小 (字节)	字节 %	总计数	计数 %	LOC	文件
57	ostrack	5160048	71.90	1	0.07	3047	698130716
50	sqlch	778496	10.85	1	0.07	202	2576467555
50	sqlch	271784	3.79	1	0.07	260	2576467555
57	ostrack	240048	3.34	1	0.07	3034	698130716
50	sqlch	144464	2.01	1	0.07	217	2576467555
62	resynch	108864	1.52	1	0.07	127	1599127346
72	eduah	108048	1.51	1	0.07	174	4210081592
69	krcbh	73640	1.03	5	0.36	547	4210081592
50	sqlch	43752	0.61	1	0.07	274	2576467555
70	apmh	40200	0.56	2	0.14	121	2986298236
69	krcbh	32992	0.46	1	0.07	838	698130716
50	sqlch	31000	0.43	31	2.20	633	3966224537
50	sqlch	25456	0.35	31	2.20	930	3966224537
52	kerh	15376	0.21	1	0.07	157	1193352763
50	sqlch	14697	0.20	1	0.07	345	2576467555

...

在 UNIX 和 Linux 上, 还可以报告专用内存的内存块。例如:

db2pd -memb pid=159770

专用内存集中的所有内存块。

地址	池标识	池名称	块存活时间	大小 (字节)	I	LOC	文件
0x0000000110469068	88	private	1	2488	1	172	4283993058
0x0000000110469A48	88	private	2	1608	1	172	4283993058
0x000000011046A0A8	88	private	3	4928	1	172	4283993058
0x000000011046B408	88	private	4	7336	1	172	4283993058
0x000000011046D0C8	88	private	5	32	1	172	4283993058
0x000000011046D108	88	private	6	6728	1	172	4283993058
0x000000011046EB68	88	private	7	168	1	172	4283993058
0x000000011046EC28	88	private	8	24	1	172	4283993058
0x000000011046EC68	88	private	9	408	1	172	4283993058
0x000000011046EE28	88	private	10	1072	1	172	4283993058
0x000000011046F288	88	private	11	3464	1	172	4283993058
0x0000000110470028	88	private	12	80	1	172	4283993058
0x00000001104700A8	88	private	13	480	1	1534	862348285
0x00000001104702A8	88	private	14	480	1	1939	862348285
0x0000000110499FA8	88	private	80	65551	1	1779	4231792244

内存集总大小: 94847 字节

按大小排序的内存块:

池标识	池名称	总大小 (字节)	总计数	LOC	文件
88	private	65551	1	1779	4231792244
88	private	28336	12	172	4283993058
88	private	480	1	1939	862348285
88	private	480	1	1534	862348285

内存集总大小: 94847 字节

### 场景 5: 确定哪个应用程序用完了表空间

通过使用 `db2pd -tcbstats`, 可以标识对一个表执行插入操作的次数。以下是用户定义的全局临时表 `TEMP1` 的样本信息:

TCB 表信息:

Address	TbSpaceID	TableID	PartID	MasterTbs	MasterTab	TableName	SchemaNm
ObjClass	DataSize	LfSize	LobSize	XMLSize			
0x0780000020B62AB0	3	2	n/a	3	2	TEMP1	SESSION
Temp	966	0	0	0			

TCB 表状态:

Address	TableName	Scans	UDI	PgReorgs	NoChgUpdts	Reads
FscrUpdates	Inserts	Updates	Deletes	OvFlReads	OvFlCrtes	
0x0780000020B62AB0	TEMP1	0	0	0	0	0
0	43968	0	0	0	0	

然后, 可以通过 `db2pd -tablespaces` 命令获取表空间 3 的信息。样本输出如下所示:

表空间 3 配置:

Address	Type	Content	PageSz	ExtentSz	Auto	Prefetch	BufID	BufIDDisk	FSC
NumCnts	MaxStripe	LastConsecPg	Name						
0x0780000020B1B5A0	DMS	UsrTmp	4096	32	Yes	32	1	1	On
1	0	31	TEMPSPACE2						

表空间 3 统计信息:

Address	TotalPgs	UsablePgs	UsedPgs	PndFreePgs	FreePgs	HWM
State	MinRecTime	NQuiescers				
0x0780000020B1B5A0	5000	4960	1088	0	<b>3872</b>	1088
0x00000000	0	0				

表空间 3 自动调整大小统计信息:

Address	AS	AR	InitSize	IncSize	IIP	MaxSize	LastResize
LRF							
0x0780000020B1B5A0	No	No	0	0	No	0	None
No							

容器:

Address	ContainNum	Type	TotalPgs	UseablePgs	StripeSet
Container					
0x0780000020B1DCC0	0	File	5000	4960	0

/home/db2inst1/tempspace2a

您会注意到通过引用 `FreePgs` 列填满的空间。因为可用页数值下降, 所以可用空间减少。还应注意, `FreePgs` 加上 `UsedPgs` 的值将等于 `UsablePgs` 的值。

一旦了解这一点, 就可以标识使用表 `TEMP1` 的动态 SQL 语句:

```
db2pd -db sample -dyn
```

```
数据库分区 0 -- 数据库 SAMPLE -- 活动 -- 正常运行 0 天 00:13:06
```

动态高速缓存:

当前使用的内存	1022197
堆总大小	1271398
高速缓存溢出标志	0
引用数	237
语句插入数目	32
语句删除数目	13

```
变动插入数目          21
语句数目              19
```

动态 SQL 语句:

```
Address          AnchID StmtUID NumEnv NumVar NumRef NumExe Text
0x0000000220A08C40 78      1      2      2      3      2      declare
    global temporary table temp1 (c1 char(6)) not logged
0x0000000220A8D960 253    1      1      1      24     24     insert
    into session.temp1 values('TEST')
```

最后, 可以将它映射至 db2pd -app 输出以标识应用程序。

应用程序:

```
Address          AppHandl [nod-index] NumAgents CoorPid Status
0x0000000200661840 501     [000-00501] 1          11246  UOW-Waiting
```

```
C-AnchID C-StmtUID L-AnchID L-StmtUID Appid
0         0         253     1         *LOCAL.db2inst1.050202160426
```

针对先前使用的 db2pd 中的动态 SQL 语句的请求产生的锚点标识 (AnchID) 将与针对关联应用程序的请求一起使用。应用程序结果显示最后一个锚点标识 (L-AnchID) 值与该锚点标识 (AnchID) 相同。一次运行 db2pd 产生的结果将在下一次运行 db2pd 时使用。

db2pd -agent 的输出将显示应用程序读取的行数 (Rowsread 列) 和写入的行数 (Rowswrtn 列)。这些值将显示应用程序已完成的部分及尚未完成的部分。

```
Address          AppHandl [nod-index] AgentPid Priority Type DBName
0x0000000200698080 501     [000-00501] 11246    0      Coord SAMPLE
```

```
State          ClientPid Userid ClientNm Rowsread Rowswrtn LkTmOt
Inst-Active 26377      db2inst1 db2bp    22      9588    NotSet
```

db2pd -agent 请求中的 AppHandl 和 AgentPid 的值可映射回 db2pd -app 请求中的 AppHandl 和 CoorPid 的对应值。

如果您怀疑内部临时表占满了表空间, 那么这些步骤会稍有不同。仍然可以使用 db2pd -tcbstats 来标识具有最大插入数目的表。以下是隐式临时表的样本信息:

TCB 表信息:

```
Address          TbspaceID TableID PartID MasterTbs MasterTab TableName
SchemaNm ObjClass DataSize ...
0x0780000020CC0D30 1      2      n/a    1      2      TEMP (00001,00002)
<30>      <JMC Temp 2470    ...
0x0780000020CC14B0 1      3      n/a    1      3      TEMP (00001,00003)
<31>      <JMC Temp 2367    ...
0x0780000020CC21B0 1      4      n/a    1      4      TEMP (00001,00004)
<30>      <JMC Temp 1872    ...
```

TCB 表状态:

```
Address          TableName Scans UDI PgReorgs NoChgUpdts Reads
FscrUpdates Inserts ...
0x0780000020CC0D30 TEMP (00001,00002) 0      0      0      0      0      0
43219 ...
0x0780000020CC14B0 TEMP (00001,00003) 0      0      0      0      0      0
42485 ...
0x0780000020CC21B0 TEMP (00001,00004) 0      0      0      0      0      0
0      ...
```

在此示例中, 使用命名约定“TEMP (TbspaceID, TableID)”的表中有大量插入。这些是隐式临时表。SchemaNm 列中的值的命名约定为 AppHandl 的值与 SchemaNm 的值并置, 这使得它能够标识正在工作的应用程序。

然后，可以将这些信息映射至 db2pd -tablespaces 产生的输出，以查看表空间 1 的已使用空间。在表空间统计信息中记下已使用的页数与可用页数之间的关系。

表空间配置:

Address	Id	Type	Content	PageSz	ExtentSz	Auto	Prefetch	BufID	BufIDDisk
FSC NumCntrs	MaxStripe	LastConsecPg	Name						
0x07800000203FB5A0	1	SMS	SysTmp	4096	32	Yes	320	1	1
On 10	0	31	TEMPSPACE1						

表空间统计信息:

Address	Id	TotalPgs	UsablePgs	UsedPgs	PndFreePgs	FreePgs	HWM
State	MinRecTime	NQuiescers					
0x07800000203FB5A0	1	6516	6516	6516	0	0	0
0x00000000 0	0	0					

表空间自动调整大小统计信息:

Address	Id	AS	AR	InitSize	IncSize	IIP	MaxSize	LastResize
LRF								
0x07800000203FB5A0	1	No	No	0	0	No	0	None

否  
容器:  
...

然后，可以使用命令 db2pd -app 标识应用程序句柄 30 和 31（因为它们出现在 -tcbstats 输出中）:

应用程序:

Address	AppHandl	[nod-index]	NumAgents	CoorPid	Status
C-AnchID	C-StmtUID	L-AnchID	L-StmtUID	Appid	
0x07800000006FB880	31	[000-00031]	1	4784182	UOW-Waiting
0	0	107	1	*LOCAL.db2inst1.051215214142	
0x07800000006F9CE0	30	[000-00030]	1	8966270	UOW-Executing
107	1	107	1	*LOCAL.db2inst1.051215214013	

最后，使用 db2pd -dyn 命令将它映射至动态 SQL:

动态 SQL 语句:

Address	AnchID	StmtUID	NumEnv	NumVar	NumRef	NumExe
Text						
0x0780000020B296C0	107	1	1	1	43	43
select c1, c2 from test group by c1,c2						

## 场景 6: 监视恢复。

命令 db2pd -recovery 显示了几个可用于验证是否正在进行恢复的计数器。当前日志和当前 LSN 提供了日志位置。已完成的工作计算目前已完成的字节数。

恢复:

恢复状态	0x00000401
当前日志	S0000005.LOG
当前 LSN	000002551BEA
作业类型	ROLLFORWARD RECOVERY
作业标识	7
作业开始时间	(1107380474) Wed Feb 2 16:41:14 2005
作业描述	数据库前滚恢复
调用程序类型	用户
总阶段	2
当前阶段	1

进度:

地址	阶段号	描述	开始时间	已完成的工作	总工作
0x0000000200667160	1	向前	Wed Feb 2 16:41:14 2005	2268098 字节	未知
0x0000000200667258	2	向后	NotStarted	0 字节	未知

## 场景 7: 确定事务正在使用的资源量



命令 `db2pd -transactions` 提供了锁定数、第一个日志序号 (LSN)、最后一个 LSN、已使用的日志空间和保留空间。这对于了解事务行为很有用。

事务:

Address	AppHandl	[nod-index]	TranHdl	Locks	State	Tflag
0x000000022026D980	797	[000-00797]	2	108	WRITE	0x00000000
0x000000022026E600	806	[000-00806]	3	157	WRITE	0x00000000
0x000000022026F280	807	[000-00807]	4	90	WRITE	0x00000000

Tflag2	Firstlsn	Lastlsn	LogSpace	SpaceReserved
0x00000000	0x000001072262	0x0000010B2C8C	4518	95450
0x00000000	0x000001057574	0x0000010B3340	6576	139670
0x00000000	0x00000107CF0C	0x0000010B2FDE	3762	79266

TID	AxRegCnt	GXID
0x000000000451	1	0
0x0000000003E0	1	0
0x000000000472	1	0

**场景 8:** 监视日志使用情况。

命令 `db2pd -logs` 对于监视数据库的日志使用情况很有用。通过观察已写入的页面输出，可以确定是否正在使用日志。

日志:

当前日志编号	2
已写入的页面	846
方法 1 归档状态	成功
方法 1 下一个要归档的日志	2
方法 1 首次失败	不适用
方法 2 归档状态	成功
方法 2 下一个要归档的日志	2
方法 2 首次失败	不适用

Address	StartLSN	State	Size	Pages	Filename
0x000000023001BF58	0x000001B58000	0x00000000	1000	1000	S0000002.LOG
0x000000023001BE98	0x000001F40000	0x00000000	1000	1000	S0000003.LOG
0x0000000230008F58	0x000002328000	0x00000000	1000	1000	S0000004.LOG

使用此输出可以确定两个问题:

1. 如果归档存在问题，那么“归档状态”将设置为值“失败”，表示最近的日志归档失败。或者，如果正在发生的归档失败导致根本无法归档日志，那么将设置“首次失败”。
2. 如果日志归档速度非常慢，那么“下一个要归档的日志”值将小于“当前日志编号”值。这可能导致填满日志路径，从而在完全填满日志路径时防止数据库中出现任何数据更改。

**场景 9:** 查看综合系统列表

如果不使用 `db2pd -sysplex` 命令，那么报告综合系统 (sysplex) 列表的唯一方法是通过 DB2 跟踪。

综合系统 (sysplex) 列表:

别名:	HOST			
位置名:	HOST1			
计数:	1			
IP 地址	端口	优先级	连接状态	PRDID
1.2.34.56	400	1	0	0

**场景 10:** 生成堆栈跟踪

可对 Windows 操作系统使用 `db2pd -stack all` 命令（对 UNIX 操作系统使用时为 `-stack` 命令）来对当前数据库分区中的所有进程生成堆栈跟踪。如果您怀疑某个进程或线程正在循环或正被挂起，那么可能要反复使用此命令。

可以通过发出命令 `db2pd -stack <eduid>` 来获取特定引擎可调度单元（EDU）的当前调用堆栈。例如：

```
db2pd -stack 137
```

正在尝试转储 eduid 137 的堆栈跟踪。  
有关陷阱文件，请参阅当前 DIAGPATH。

如果需要所有 DB2 进程的调用堆栈，请使用 `db2pd -stack all` 命令，例如，在 Windows 系统上：

```
db2pd -stack all
```

尝试转储实例的所有堆栈跟踪。 有关陷阱文件，请参阅当前 DIAGPATH。

如果您正在使用具有多个物理节点的分区数据库环境，那么通过使用命令 `db2_all "; db2pd -stack all"` 可以获取所有分区中的信息。但是，如果分区是同一机器上的所有逻辑分区，那么使用 `db2pd -alldbp -stacks` 时的速度会更快。

---

## 使用 db2support 收集环境信息

对 DB2 问题收集信息时，您需要运行的最重要的 DB2 实用程序是 `db2support`。 `db2support` 实用程序用于自动收集所有可用的 DB2 诊断信息和系统诊断信息。它还有一个可选的交互式“问与答”会话，该会话会提出有关问题的详情。

使用 `db2support` 实用程序可以避免可能的用户错误，这是因为您不必手动输入“GET DATABASE CONFIGURATION FOR <database name>”或“LIST TABLESPACES SHOW DETAIL”之类的命令。而且，您不需要有关要运行的命令或要收集的文件的指示信息，因此收集数据的速度会比较快。

- 执行 `db2support -h` 命令以生成完整的命令选项列表。
- 使用相应的 `db2support` 命令收集数据。

`db2support` 实用程序应该由具有 `SYSADM` 权限的用户（如实例所有者）运行，以便该实用程序可以收集所有必需的信息而不发生错误。如果没有 `SYSADM` 权限的用户运行 `db2support`，那么在实用程序运行“`QUERY CLIENT`”或“`LIST ACTIVE DATABASE`”之类的命令时，可能会产生 SQL 错误（`SQL1092`）。

如果使用 `db2support` 实用程序来帮助将信息传送至 IBM 支持机构，那么在系统遇到问题时运行 `db2support`。这样工具就会及时地收集信息，例如操作系统性能详细信息。如果在出现问题时无法运行实用程序，您仍可以在问题停止之后发出 `db2support` 命令，因为会自动生成某些首次出现数据捕获（`FODC`）诊断文件。

对于调试问题所需的大多数信息的收集而言，以下基本调用通常已经足够（注意，如果使用 `-c` 选项，那么该实用程序将建立与数据库的连接）：

```
db2support <output path> -d <database name> -c
```

输出的收集非常方便，并且会存储在压缩的 ZIP 归档 `db2support.zip` 中，以便可以很轻松地任何系统上对其进行传送和解压缩。

db2support 捕获的信息类型取决于调用命令的方式、是否启动了数据库管理器以及能否连接至数据库。

db2support 实用程序在所有条件下收集以下信息:

- db2diag.log
- 所有陷阱文件
- 锁定列表文件
- 转储文件
- 各种与系统有关的文件
- 各种系统命令的输出
- db2cli.ini

根据情况, db2support 实用程序还有可能收集以下信息:

- 活动日志文件
- 缓冲池和表空间 (SQLSPCS.1 和 SQLSPCS.2) 控制文件 (使用 -d 选项)
- db2dump 目录的内容
- 扩展系统信息 (使用 -s 选项)
- 数据库配置设置 (使用 -d 选项)
- 数据库管理器配置设置文件
- 日志文件头文件 (使用 -d 选项)
- 恢复历史记录文件 (使用 -d 选项)

HTML 报告 db2support.html 将总是包括下列信息:

- 问题记录 (PMR) 编号 (如果指定了 -n)
- 操作系统和级别 (如 AIX 5.1)
- DB2 发行版信息
- 是 32 位还是 64 位环境的指示信息
- DB2 安装路径信息
- db2nodes.cfg 的内容
- CPU 和磁盘数目及内存量
- 实例上的数据库列表
- 注册表信息和环境, 包括 PATH 和 LIBPATH
- UNIX 的当前文件系统和索引节点的磁盘可用空间
- Java SDK 级别
- 数据库管理器配置
- 数据库恢复历史记录文件列表
- sqllib 目录的 ls -lR 输出 (或 Windows 上的相应项)
- LIST NODE DIRECTORY 命令的结果
- LIST ADMIN NODE DIRECTORY 命令的结果
- LIST DCS DIRECTORY 命令的结果
- LIST DCS APPLICATIONS EXTENDED 命令的结果
- 所有已安装软件的列表

指定 -s 时，以下信息将出现在 db2support.html 文件中：

- 详细的磁盘信息（分区布局、类型、LVM 信息等等）
- 详细的网络信息
- 内核统计信息
- 固件版本
- 其他特定于操作系统的命令

如果已经启动了 DB2，那么 db2support.html 文件包含下列附加信息：

- 客户机连接状态
- 数据库和数据库管理器配置（数据库配置需要 -d 选项）
- CLI 配置
- 内存池信息（大小和耗用大小）。如果使用 -d 选项，那么会收集完整数据
- LIST ACTIVE DATABASES 命令的结果
- LIST DCS APPLICATIONS 命令的结果

如果已指定 -c 并且已成功连接至数据库，那么 db2support.html 文件包含以下信息：

- 用户表的数目
- 数据库数据的大概大小
- 数据库快照
- 应用程序快照
- 缓冲池信息
- LIST APPLICATIONS 命令的结果
- LIST COMMAND OPTIONS 命令的结果
- LIST DATABASE DIRECTORY 命令的结果
- LIST INDOUBT TRANSACTIONS 命令的结果
- LIST DATABASE PARTITION GROUPS 命令的结果
- LIST DBPARTITIONNUMS 命令的结果
- LIST ODBC DATA SOURCES 命令的结果
- LIST PACKAGES/TABLES 命令的结果
- LIST TABLESPACE CONTAINERS 命令的结果
- LIST TABLESPACES 命令的结果
- LIST DRDA<sup>®</sup> IN DOUBT TRANSACTIONS 命令的结果

---

## 基本跟踪诊断

如果遇到与 DB2 相关的反复出现并且可再现的问题，那么执行跟踪有时能够捕获有关该问题的其他信息。在正常情况下，仅当 DB2 客户支持机构要求时，才应该使用跟踪。执行跟踪的过程包括设置跟踪工具、再现错误与收集数据。

跟踪收集的信息量增长得相当快。当您执行跟踪时，请尽可能地只捕获错误状态并避免任何其他活动。在执行跟踪时，请尽可能地使用最省力的方案来再现问题。

收集跟踪信息会对 DB2 实例的性能带来负面影响。性能的降低程度取决于问题的类型以及使用了多少资源来收集跟踪信息。

在请求跟踪时，DB2 客户支持机构应提供下列信息：

- 简单的逐步过程
- 进行每次跟踪的位置的说明
- 跟踪内容的说明
- 请求跟踪的理由的说明
- 回退过程（例如，如何禁用所有跟踪）

尽管您应该听从 DB2 客户支持机构有关获取哪些跟踪的建议，但以下仍然提供要求您获取特定跟踪时的一些一般准则：

- 如果安装期间发生问题，并且缺省安装日志不足以确定问题原因，那么比较适合进行安装跟踪。
- 如果在其中一个 GUI（图形用户界面）工具中发生问题，并且通过 DB2 命令窗口中的一些显式命令成功地执行了某些操作，那么比较适合进行控制中心跟踪。注意，这只会使用可从控制中心启动的工具捕获问题。
- 如果 CLI 应用程序中出现问题，并且该问题不能在该应用程序之外再现，那么比较适合进行 CLI 跟踪。
- 如果 JDBC 应用程序中出现问题，并且该问题不能在该应用程序之外再现，那么比较适合进行 JDBC 跟踪。
- 如果该问题与要在 DRDA 层进行通信的信息直接有关，那么进行 DRDA 跟踪比较适合。
- 对于可进行跟踪的所有其他情况，最适合进行 DB2 跟踪。

在诊断错误时，跟踪信息并不总是对您有帮助。例如，在下列情况下可能无法捕获错误情况：

- 您指定的跟踪缓冲区大小不够大，无法存放完整的跟踪事件集，当跟踪停止写入文件或合并时，丢失了有用的信息。
- 跟踪的方案没有重新创建错误状态。
- 重新创建了错误状态，但对问题发生位置的假定不正确。例如，跟踪是在客户机工作站上收集到的，但实际错误发生在服务器上。

## DB2 跟踪

将讨论如何使用内部实用程序来获取 DB2 跟踪。一旦根据跟踪缓冲区中的跟踪数据来创建 DB2 跟踪文件，您将需要对该输出内容编排格式，使其容易读取。DB2 支持人员可使用该文件中的信息来解决特定的问题。

### 使用 db2trc 获取 DB2 跟踪

**db2trc** 命令控制随 DB2 提供的跟踪工具。跟踪工具记录有关操作的信息并将此信息格式化为可读格式。

记住，运行跟踪时会增加开销，所以启用跟踪工具可能会影响系统性能。

通常 DB2 支持机构和开发团队使用 DB2 跟踪故障诊断。您可以运行跟踪来获取有关正在调查的问题的信息，但如果您不了解 DB2 源代码的话，它的用途将十分有限。

即使只要求您获取跟踪文件，您也应该知道如何正确地启动跟踪以及如何转储跟踪文件。

**注：**您需要 SYSADM、SYSCTRL 或 SYSMANT 权限的其中一种权限来使用 db2trc。

要大致了解可用的选项，执行不带任何参数的 db2trc 命令：

```
C:\>db2trc
用法: db2trc (chg|clr|dmp|flw|fmt|inf|off|on) options
```

有关特定 db2trc 命令参数的更多信息，请使用 -u 选项。例如，要查看有关启动跟踪的更多信息，请执行以下命令：

```
db2trc on -u
```

这会提供有关可在启动 DB2 跟踪时指定的所有附加选项（标注为“工具”）的信息。

启用跟踪时，最重要的选项是 -L。它指定用于存储跟踪信息的内存缓冲区的大小。缓冲区大小可以字节或兆字节为单位来指定。要指定兆字节，那么在值后面追加“M”或“m”。跟踪缓冲区大小必须是 2 兆字节的幂。如果指定的大小不符合此要求，那么缓冲区大小将自动舍入为最接近的 2 的幂。

如果缓冲区太小，那么信息可能会丢失。在缺省情况下，如果缓冲区变满，那么只会保留最新的跟踪信息。如果缓冲太大，可能难以将文件发送至 DB2 支持团队。

如果跟踪时间相对较短的操作（如数据库连接），那么大概 8MB 大小通常就已足够：

```
C:\> db2trc on -l 8M
已启动跟踪
```

但是，如果跟踪规模较大的操作或者同时在进行大量工作，那么可能需要较大的跟踪缓冲区。

在大多数平台上，跟踪可按如上所述随时打开并工作。但是，有些特定情况需要注意：

1. 在多数数据库分区系统上，必须对每个物理（相对于逻辑）数据库分区运行跟踪。
2. 在 HP-UX 上，Linux 和 Solaris 平台上，如果在启动实例后关闭跟踪，那么不管指定大小如何，下一次启动跟踪时都会使用非常小的缓冲区。例如，昨天您通过使用 db2trc on -l 8m 启动了跟踪，然后收集了跟踪信息，然后停止了跟踪（db2trc off）。今天您希望在不关闭并重新启动实例的情况下，运行跟踪并将内存缓冲区设置为 32 MB（db2trc on -l 32m）。您会发现在此情况下，跟踪仅获得很小的缓冲区。为了在这些平台上有效地运行跟踪，应在启动实例前以所需大小缓冲区启动跟踪，并在以后必要时“清除”缓冲区。

## 转储 DB2 跟踪文件

使用 on 选项启用跟踪工具后，将跟踪实例所作的所有后续工作。

跟踪运行时，可使用 clr 选项来清除跟踪缓冲区。将除去跟踪缓冲区中的所有现有信息。

```
C:\>db2trc clr
已清除跟踪
```

要跟踪的操作完成后，使用后跟跟踪文件名的 dmp 选项将内存缓冲池转储至磁盘。例如：



```
C:\>db2trc dmp trace.dmp
跟踪已转储至文件
```

跟踪缓冲区转储至磁盘后，跟踪工具将继续运行。要关闭跟踪，使用 `off` 选项：

```
C:\>db2trc off
已关闭跟踪
```

## 格式化 DB2 跟踪文件

`db2trc dmp` 命令创建的转储文件为二进制格式，并且不可读取。

要验证是否可读取跟踪文件，请对二进制跟踪文件进行格式化以显示流量控制并将格式化输出发送至空设备。下面的示例显示了用于执行此任务的命令：

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

其中 `example.trc` 使用 `dmp` 选项生成的二进制文件。

此命令的输出将显式地指出读取文件是否有问题，以及跟踪是否已合并。

此时，可将转储文件发送至 DB2 支持机构。他们会根据您的 DB2 服务级别来对其进行格式化。但有时可能会要求您在发送转储文件之前将其转换为 ASCII 格式。这是通过 `flw` 和 `fmt` 选项完成的。必须提供二进制转储文件的名称及要创建的 ASCII 文件的名称：

```
C:\>db2trc flw trace.dmp trace.flw
C:\Temp>db2trc flw trace.dmp trace.flw
总跟踪记录数: 18854
截断跟踪: NO
合并跟踪: NO
格式化的跟踪记录数: 1513 (pid: 2196 tid 2148 节点: -1)
格式化的跟踪记录数: 100 (pid: 1568 tid 1304 节点: 0)
...
```

```
C:\>db2trc fmt trace.dmp trace.fmt
C:\Temp>db2trc fmt trace.dmp trace.fmt
截断跟踪: NO
合并跟踪: NO
总跟踪记录数: 18854
格式化的跟踪记录数: 18854
```

如果此输出指示“合并跟踪”为“YES”，那么表示跟踪缓冲区不够大，无法包含跟踪时间段收集的所有信息。根据情况，合并跟踪也许是可行的。如果您关心的是最新信息（除非指定了 `-i` 选项，否则这是缺省情况下保留的信息），那么跟踪文件中的内容可能已经足够。但是，如果您关心的是跟踪时间段开始时发生的情况，或者如果您关心的是发生的所有情况，那么您可能想要使用更大的跟踪缓冲区重做该操作。

将二进制文件格式化为可读文本文件时，有一些选项可用。例如，可使用 `db2trc fmt -xml trace.dmp trace.fmt` 转换二进制数据并以 XML 可解析格式输出结果。其他选项显示在跟踪命令（`db2trc`）的详细描述中。

要注意的另一件事是：在 Linux 和 UNIX 操作系统上，DB2 在因为严重错误而关闭实例时会自动将跟踪缓冲区转储至磁盘。因此，如果实例异常结束时启用了跟踪，那么会在诊断目录中创建一个文件，其名称为 `db2trdmp.###`，其中 `###` 是数据库分区号。Windows 平台上不会发生这种情况。在这些情况下，您必须手动转储跟踪。

为进行总结，以下是 `db2trc` 命令的常用顺序示例：



```
db2trc on -l 8M
db2trc clr
<Execute problem recreation commands>
db2trc dump db2trc.dmp
db2trc off
db2trc flw db2trc.dmp <filename>.flw
db2trc fmt db2trc.dmp <filename>.fmt
db2trc fmt -c db2trc.dmp <filename>.fmtc
```

## DRDA 跟踪

Distributed Relational Database Architecture™ (DRDA) 定义如何在公共结构链接的数据库之间传递数据。DB2 Connect 使用 DRDA 来将数据库客户机工作站链接至主机或 System i™ 机器上的数据库。如果遇到此类环境的问题，那么需要了解 DRDA 跟踪文件、用于创建跟踪的实用程序、所生成的输出和如何分析输出。另外，DRDA 跟踪样本也会显示出来供您参考。最后，建议您检查后续发送和接收缓冲区，这是因为它们可能包含与您遇到的问题相关的信息。

### DRDA 跟踪文件

在分析 DRDA 跟踪之前，需要了解 DRDA 是数据和通信结构的定义的开放式标准。例如，DRDA 包含有关数据传输的组织方式以及有关该信息的通信方式的一组规则。这些规则是在以下参考手册中定义的：

- DRDA V3 Vol. 1: 分布式关系数据库体系结构
- DRDA V3 Vol. 2: 格式化数据对象内容体系结构
- DRDA V3 Vol. 3: 分布式数据管理体系结构

这些手册的 PDF 版本可从 [www.opengroup.org](http://www.opengroup.org) 获取。

**db2drdat** 实用程序记录 DRDA Application Requestor (AR) 与 DB2 DRDA Application Server (AS) 之间（如 DB2 Connect 与主机或 Series i 数据库服务器之间）的数据交换。

### 跟踪实用程序

db2drdat 实用程序记录 DB2 Connect 服务器（代表 IBM 数据服务器客户机）与主机或 System i 数据库服务器之间交换的数据。

作为数据库管理员（或应用程序开发者），您可能会发现了解此数据流如何工作是很很有用的，因为这些知识可以帮助您确定特定问题的起源。例如，如果您对主机或 System i 数据库服务器发出了 CONNECT TO 数据库语句，但是该命令失败了并且您接收到一个不成功的返回码。如果您确实了解到哪些信息被传送到主机或 System i 数据库服务器管理系统，那么您就可能能够确定故障的原因，即使返回码信息是一般信息。很多故障是由简单的用户错误造成的。

来自 db2drdat 的输出列示了 DB2 Connect 工作站与主机或 System i 数据库服务器管理系统之间交换的数据流。发送到主机或 System i 数据库服务器的数据标记为 SEND BUFFER，而从主机或 System i 数据库接收到的数据标记为 RECEIVE BUFFER。

如果接收缓冲区中包含 SQLCA 信息，那么它将后接此数据的已格式化的解释并标记为 SQLCA。SQLCA 的 SQLCODE 字段是主机或 System i 数据库服务器所返回的未映射的值。在文件中，发送缓冲区和接收缓冲区是按从最旧到最新的顺序来排列的。每个缓冲区都具有：

- 进程标识。
- SEND BUFFER、RECEIVE BUFFER 或 SQLCA 标号。缓冲区中的第一个 DDM 命令或对象被标记为 DSS TYPE。

在发送缓冲区和接收缓冲区中的其他数据被分成五列，包括：

- 字节数。
- 第 2 列和第 3 列表示在两个系统之间所交换的 DRDA 数据流，采用 ASCII 或 EBCDIC 格式。
- 第 2 列和第 3 列的 ASCII 表示。
- 第 2 列和第 3 列的 EBCDIC 表示。

## 跟踪输出

db2drdat 实用程序将下列信息写入到跟踪文件中：

- -r
  - DRDA 应答/对象的类型
  - 接收缓冲区
- -s
  - DRDA 请求的类型
  - 发送缓冲区
- -c
  - SQLCA
- TCP/IP 错误信息
  - 接收函数返回码
  - 严重性
  - 使用的协议
  - 使用的 API
  - 功能
  - 错误号

注：

1. 出口码的值为零，指示成功完成了命令，非零值指示没有成功完成命令。
2. 返回的字段将随使用的 API 不同而不同。
3. 返回的字段将随运行 DB2 Connect 的平台不同而不同，即使对于同一 API 也是如此。
4. 如果 db2drdat 命令将输出发送至已存在的文件，那么将擦除旧文件，除非文件的许可权不允许擦除。

## 跟踪输出文件分析

在 db2drdat 跟踪中捕获到下列信息：

- 客户机应用程序的进程标识 (PID)
- 在数据库连接服务 (DCS) 目录中编目的 RDB\_NAME
- DB2 Connect CCSID
- 主机或 System i 数据库服务器 CCSID

- DB2 Connect 系统正在与其通信的主机或 System i 数据库服务器管理系统。

首个缓冲区中包含发送到主机或 System i 数据库服务器管理系统的“交换服务器属性” (EXCSAT) 和“访问 RDB” (ACCRDB) 命令。它将这些命令作为 CONNECT TO 数据库命令的结果来发送。下一个缓冲区中包含 DB2 Connect 从主机或 System i 数据库服务器管理系统中接收到的应答。它包含“交换服务器属性应答数据” (EXCSATRD) 和“访问 RDB 应答消息” (ACCRDBRM)。

### EXCSAT

EXCSAT 命令包含由“服务器名” (SRVNAM) 对象指定的客户机的工作站名, 按照 DDM 规范, 它是代码点 X'116D'。EXCSAT 命令在第一个缓冲区中。在 EXCSAT 命令内, 一旦除去了 X'116D', 值 X'9481A292' (按 CCSID 500 编码) 就被转换为掩码。

EXCSAT 命令还包含 EXTNAM (外部名) 对象, 通常将该对象放在主机或 System i 数据库管理系统上的诊断信息中。它由 20 个字节的应用程序标识、后接 8 个字节的进程标识 (或者是 4 个字节的进程标识和 4 个字节的线程标识) 组成。它由代码点 X'115E' 表示, 在此示例中, 其值为 db2bp 并用空格填充, 后接 000C50CC。在 Linux 或 UNIX IBM 数据服务器客户机上, 可以将此值与 ps 命令相关联, 该命令将与活动进程有关的进程状态信息返回到标准输出中。

### ACCRDB

ACCRDB 命令包含 RDBNAM 对象中的 RDB\_NAME, 它是代码点 X'2110'。在第一个缓冲区中, ACCRDB 命令跟在 EXCSAT 命令后面。在 ACCRDB 命令内, 一旦除去了 X'2110', 值 X'E2E3D3C5C3F1' 就会被转换为 STLECL1。这对应于 DCS 目录中的目标数据库名称字段。

记帐字符串具有代码点 X'2104'。

通过在 ACCRDB 命令中查找代码点为 X'119C' 的 CCSID 对象 CCSIDSBC (用于单字节字符的 CCSID), 就可显示为 DB2 Connect 工作站配置的代码集。在此示例中, CCSIDSBC 为 X'0333', 代码集为 819。

在 ACCRDB 命令中还存在附加对象 CCSIDDBC (双字节字符的 CCSID) 和 CCSIDMBC (混合字节字符的 CCSID), 它们的代码点分别为 X'119D' 和 X'119E'。在此示例中, CCSIDDBC 为 X'04B0', 代码集为 1200; CCSIDMBC 为 X'0333', 代码集为 819。

### EXCSATRD 和 ACCRDBRM

CCSID 值也是从主机或 System i 数据库服务器的第二个缓冲区内的“访问 RDB 应答消息” (ACCRDBRM) 中返回的。此缓冲区中包含 EXCSATRD, 后接 ACCRDBRM。样本输出文件中包含主机或 System i 数据库服务器系统的两个 CCSID 值。这两个值分别为 1208 (对于单字节字符和混合字节字符) 和 1200 (对于双字节字符)。

如果 DB2 Connect 不能识别从主机或 System i 数据库服务器返回的代码页, 那么将对用户返回 SQLCODE -332 以及源和目标代码页。如果主机或 System i 数据库服务器不识别从 DB2 Connect 发送的代码集, 那么它将返回 VALNSPRM (不支持参数值, DDM 代码点为 X'1252'), 并对用户转换为 SQLCODE -332。

ACCRDBRM 还包含参数 PRDID (特定产品标识, 代码点为 X'112E')。值为 X'C4E2D5F0F8F0F1F5', 用 EBCDIC 表示为 DSN08015。按照标准, DSN 为

DB2 通用数据库 z/OS 版和 OS/390 版。还指示了版本号。ARI 是 DB2 服务器 VSE 版和 VM 版，SQL 是 DB2 数据库或 DB2 Connect，QSQ 是 DB2 i5/OS 版。

## 跟踪输出文件样本

下列各图显示了样本输出，说明在 DB2 Connect 工作站与主机或 System i 数据库服务器之间交换的一些 DRDA 数据流。从用户的角度来看，已经使用命令行处理器（CLP）发出了 CONNECT TO 数据库命令。

第 73 页的图 2 使用 DB2 Connect 企业版版本 9.1 和 DB2 通用数据库（UDB）z/OS 版版本 8 来进行 TCP/IP 连接。

```
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	..!.....@@@@  ....STLEC1	
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 1443000F115E5F8	.Y.C...S.C...^.. ..}.....;V8	
0010 F1C14BE2E3D3C5C3 F100181404140300	..K..... 1A.STLEC1.....	
0020 0724070007147400 05240F0007144000	.\$...t..\$...@. ....	
0030 0700081147D8C4C2 F20014116DE2E3D3	...G.....m... ....QDB2..._STL	
0040 C5C3F14040404040 404040404000C11	...@@@@@@@@... 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
```

图 2. 跟踪输出的示例 (TCP/IP 连接)

```
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..... ..}>.....>...s..
0010 00162110E2E3D3C5 C3F1404040404040 ..!.....@..... ..STLECI
0020 404040404040000C 11A1D9858799F485 @..... ..Regr4e
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 ..... ..!$. ..}....x.....
0010 00172135C7F9F1C1 F0C4F3C14BD7C1F8 ..!5.....K... ..G91A0D3A.PA8
0020 F806030221064600 162110E2E3D3C5C3 .....!.F..!..... 8.....STLEC
0030 F140404040404040 4040404040000C11 .@.....@..... 1 ..
0040 2EE2D8D3F0F9F0F0 F0000D002FD8E3C4 ...../... .SQL09000....QTD
0050 E2D8D3C1E2C30016 00350006119C0333 .....5.....3 SQLASC.....
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.. ..}....n.....
0010 000D002FD8E3C4E2 D8D3F3F7F0000C11 ..!/..... ..QTDSQL370...
0020 2EC4E2D5F0F8F0F1 F500160035000611 .....5... .DSN08015.....
0030 9C04B80006119E04 B80006119D04B000 .....
0040 0C11A0D5C5E6E3D6 D540400006212524 .....@...!%$ ...NEWTON .....
0050 34001E244E000624 4C00010014244D00 4..$.N..$.L...$.M. ....+...<.....(.
0060 06244FFFFFF000A11 E8091E768301BE00 .$.O.....v.... ..!.....Y....c...
0070 2221030000000005 68B3B8C7F9F1C1F0 "!.....h..... ..G91A0
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

图 3. 跟踪输出的示例 (TCP/IP 连接) - 续

```
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  @##### SQLC2FOA
0040    4040404041414141 41484C5600CB0005  @#####<.....
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

图 4. 跟踪输出的示例 (TCP/IP 连接) - 续



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	240800FFFFFFF3434										.f.....~\$.....44	..}.....-.....
0010	3237303444534E58	4F544C2000FFFFFFE										2704DSNXOTL ....	.....+!.<.....
0020	0C00000000000000	00FFFFFFF000000										.....	.....
0030	0000000000572020	2057202020202020										....W W	.....
0040	001053544C454331	2020202020202020										..STLECI	.....<.....
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										..!.....@...@	....STLECI
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										.....\$..	..}.....

图 5. 跟踪输出的示例 (TCP/IP 连接) - 续

## 有关 DRDA 跟踪的后续缓冲区信息

可以分析后续发送和接收缓冲区以获取附加信息。下一个请求包含一个落实。commit 命令指示主机或 System i 数据库服务器管理系统来落实当前工作单元。第四个缓冲区是

作为落实或回滚的结果从主机或 System i 数据库服务器数据库管理系统中接收到的。它包含“结束工作单元应答消息”（ENDUOWRM），它指示当前工作单元已经结束。

在此示例中，跟踪条目 12 包含一个空的 SQLCA，由后接 X'FF' 的 DDM 代码点 X'2408' 指示。空的 SQLCA（X'2408FF'）表示成功（SQLCODE 0）。

第 73 页的图 2 在跟踪条目 16 中显示了包含错误 SQLCA 的接收缓冲区的示例。

## 控制中心跟踪

尝试在控制中心中跟踪问题之前，建议先确保通过从 DB2 命令提示符处输入显式命令执行操作时未发生同样的问题。通常当您在控制中心（或可从控制中心启动的一种其他的 GUI 工具）中执行任务时，您会看到“显示命令”按钮，它将提供该工具使用的命令的确切语法。如果从 DB2 命令提示符处成功执行了这一确切命令，但在 GUI 工具中执行该命令时失败了，那么应该获取控制中心跟踪。

为了获取只能在控制中心中再现的问题的跟踪，按如下所示启动控制中心：

```
db2cc -tf filename
```

这会打开控制中心跟踪并且将跟踪输出保存至指定的文件。输出文件将保存至 Windows 上的 <DB2 install path>\sqllib\tools 以及 UNIX 和 Linux 上的 /home/<userid>/sqllib/tools。

**注：**启动控制中心并且启用跟踪后，使用尽可能少的步骤来再现问题。尽量避免在工具中单击不必要或不相关的项。再现问题后，关闭控制中心（以及为再现问题而打开的任何其他 GUI 工具）。

生成的跟踪文件需要发送至 DB2 支持机构以便进行分析。

## JDBC 跟踪

根据使用的 JDBC 驱动程序的类型，有不同方法来获取正在运行的应用程序或存储过程的跟踪文件。此处介绍了这些不同的方法。

### 获取使用 DB2 JDBC 2 类驱动程序 Linux 版、UNIX 版和 Windows 版的应用程序的跟踪

此类型的跟踪适用于以下遇到问题的情况：

- 使用 DB2 JDBC 2 类驱动程序 Linux 版、UNIX 版和 Windows 版（DB2 JDBC 2 类驱动程序）的 JDBC 应用程序
- DB2 JDBC 存储过程

**注：**有许多关键字可添加至 db2cli.ini 文件并影响应用程序行为。这些关键字可能解决应用程序问题，也可能是造成应用程序问题的原因。还有一些关键字在 CLI 文档中并未讨论。这些关键字只能从 DB2 支持机构处获取。如果 db2cli.ini 文件包含未讨论的关键字，那么它们可能是 DB2 支持机构建议使用的关键字。DB2 JDBC 2 类驱动程序在内部使用 DB2 CLI 驱动程序进行数据库访问。例如，DB2 JDBC 2 类驱动程序在内部将 Java getConnection() 方法映射至 DB2 CLI SQLConnect() 函数。因此，Java 开发者可能会发现 DB2 CLI 跟踪是 DB2 JDBC 跟踪的一个很好的补充。

1. 为跟踪文件创建路径。应该创建每个用户都有写入权限的路径。

例如，在 Windows 上：

```
mkdir c:\temp\trace
```

在 Linux 和 UNIX 上：

```
mkdir /tmp/trace
chmod 777 /tmp/trace
```

2. 更新 CLI 配置关键字。解决此冲突有两种方法：

- 手动编辑 db2cli.ini 文件。
  - a. 在纯文本编辑器中打开 db2cli.ini 文件。

缺省情况下，DB2 CLI/ODBC 配置关键字文件 (db2cli.ini) 在 Windows 平台上的 sqllib 目录中和 UNIX 平台上运行 CLI/ODBC 应用程序的数据库实例的 sqllib/cfg 目录中。如果使用 ODBC 驱动程序管理器在 Windows 平台上配置用户数据源，那么会在用户主 (概要文件) 目录中创建 db2cli.ini。还可使用环境变量 DB2CLIINIPATH 来覆盖缺省值并对该文件指定另一位置。

- b. 将以下一段添加至该文件 (如果 COMMON 段已存在，那么只追加变量)：

```
[COMMON]
JDBCTrace=1
JDBCTracePathName=<path>
JDBCTraceFlush=1
```

例如，其中 <path> 在 Windows 上为 C:\temp\trace，在 Linux 或 UNIX 操作系统上为 /tmp/trace。

- c. 保存该文件时至少保存文件结尾的一个空白行。(这可以避免某些语法分析错误。)
- 使用 UPDATE CLI CFG 命令更新 db2cli.ini 文件。发出下列命令：

```
db2 UPDATE CLI CFG FOR SECTION COMMON USING JDBCTrace 1
db2 UPDATE CLI CFG FOR SECTION COMMON USING JDBCTracePathName <path>
```

例如，其中 <path> 在 Windows 上为 C:\temp\trace，在 Linux 或 UNIX 操作系统上为 /tmp/trace。

```
db2 UPDATE CLI CFG FOR SECTION COMMON USING JDBCTraceFlush 1
```

当您使用跟踪工具来诊断应用程序问题时，记住它对应用程序性能会有影响，并且会影响所有应用程序而不仅仅是测试程序。这就是一定要记住在标识问题后要关闭跟踪工具的原因。

3. 发出以下命令来验证是否设置并选择了正确的关键字：

```
db2 GET CLI CFG FOR SECTION COMMON
```

4. 重新启动应用程序。

仅当应用程序启动时才读取 db2cli.ini 文件，因此，要使任何更改生效，都必须重新启动应用程序。

如果要跟踪 JDBC 存储过程，那么表示要重新启动 DB2 实例。

5. 捕获错误。运行该应用程序直到生成错误，然后终止该应用程序。尽可能减少这种情况，在跟踪时只运行与问题再现有关的 JDBC 应用程序。这会使跟踪文件更加清晰明了。
6. 禁用 JDBC 跟踪。

在 db2cli.ini 的 [COMMON] 段手动设置 JDBCTrace=0 关键字，或者发出：

```
db2 UPDATE CLI CFG FOR SECTION COMMON USING Trace 0
db2 UPDATE CLI CFG FOR SECTION COMMON USING JDBCTrace 0
```

7. 重新启动正在运行和跟踪的所有应用程序。
8. 收集跟踪文件。

JDBC 跟踪文件将写至 JDBCTracePathName 关键字指定的路径。生成的文件名都会以 .trc 扩展名结尾。问题再现时在跟踪路径中生成的所有文件都是必需的。

## 获取使用 DB2 通用 JDBC 驱动程序的应用程序的跟踪

如果已经有正在使用 DB2 通用 JDBC 驱动程序的 SQLJ 或 JDBC 应用程序，那么可使用以下几种不同方法来启用 JDBC 跟踪：

- 如果使用 DataSource 接口连接至数据源，那么使用 DataSource.setTraceLevel() 和 DataSource.setTraceFile() 方法来启用跟踪。
- 如果使用 DriverManager 接口连接至数据源，那么启用跟踪的最简单方法是获取连接之前在 DriverManager 上设置 logWriter。

例如：

```
DriverManager.setLogWriter(new PrintWriter(new FileOutputStream("trace.txt")));
```

- 如果在使用 DriverManager 接口，在装入驱动程序时还可以在 URL 中指定 traceFile 和 traceLevel 属性。

例如：

```
String databaseURL =
"jdbc:db2://hal:50000/sample:traceFile=c:/temp/foobar.txt;" ;
```

## CLI 跟踪文件

CLI 跟踪捕获有关访问 DB2 CLI 驱动程序的应用程序的信息。

CLI 跟踪基本上不提供有关 DB2 CLI 驱动程序的内部工作情况的信息。

此类型的跟踪适用于以下遇到问题的情况：

- CLI 应用程序
- ODBC 应用程序（因为 ODBC 应用程序使用 DB2 CLI 接口来访问 DB2）
- DB2 CLI 存储过程
- JDBC 应用程序和存储过程

在诊断 ODBC 应用程序时，通常最容易确定问题的方法就是使用 ODBC 跟踪或 DB2 CLI 跟踪。如果在使用 ODBC 驱动程序管理器，那么可以进行 ODBC 跟踪。查阅驱动程序管理器文档以确定启用 ODBC 跟踪的方式。DB2 CLI 跟踪是特定于 DB2 的，并且通常包含的信息比一般 ODBC 跟踪要多。这两种跟踪通常极为相似，都会列示某个应用程序中所有 CLI 调用的入口点和出口点；包括这些调用的所有参数和返回码。

DB2 JDBC 2 类驱动程序 Linux 版、UNIX 版和 Windows 版（DB2 JDBC 2 类驱动程序）取决于访问数据库的 DB2 CLI 驱动程序。因此，Java 开发者可能还想启用 DB2 CLI 跟踪，以获取有关应用程序如何通过各个软件层与数据库交互的其他信息。尽管都是在 db2cli.ini 文件中设置的，但 DB2 JDBC 和 DB2 CLI 跟踪选项相互独立。

## 启用 CLI 跟踪

通过将特定条目添加至 db2cli.ini 文件来启用 CLI 跟踪。

**注：**有许多关键字可添加至 db2cli.ini 文件并影响应用程序行为。这些关键字可能解决应用程序问题，也可能是造成应用程序问题的原因。还有一些关键字在 CLI 文档中并未讨论。这些关键字只能从 DB2 支持机构处获取。如果 db2cli.ini 文件包含未讨论的关键字，那么它们可能是 DB2 支持团队建议使用的关键字。

缺省情况下，DB2 CLI/ODBC 配置关键字文件的位置在 Windows 操作系统上为 sqllib 目录，在 Linux 和 UNIX 操作系统上为运行 CLI/ODBC 应用程序的数据库实例的 sqllib/cfg 目录。如果使用 ODBC 驱动程序管理器在 Windows 上配置用户数据源，那么会在用户主（概要文件）目录中创建 db2cli.ini。还可使用环境变量 DB2CLIINIPATH 来覆盖缺省值并对该文件指定另一位置。

### 1. 为跟踪文件创建路径。

应该创建每个用户都有写入权限的路径。例如，在 Windows 上：

```
mkdir c:\temp\trace
```

在 Linux 和 UNIX 上：

```
mkdir /tmp/trace
chmod 777 /tmp/trace
```

### 2. 更新 CLI 配置关键字。

可通过手动编辑 db2cli.ini 文件或使用 UPDATE CLI CFG 命令来执行此操作。

选项 A：手动编辑 db2cli.ini 文件。

- a. 在纯文本编辑器中打开 db2cli.ini 文件。
- b. 将以下一段添加至该文件（或者如果 COMMON 段已存在，那么只追加变量）：

```
[COMMON]
Trace=1
TracePathName=<path>
TraceComm=1
TraceFlush=1
TraceTimeStamp=1
TraceScript=1
```

例如，其中 <path> 在 Windows 上为 C:\temp\trace，在 Linux 或 UNIX 上为 /tmp/trace。

- c. 保存该文件时至少保存文件结尾的一个空白行。（这可以避免某些语法分析错误。）

选项 B：使用 UPDATE CLI CFG 命令更新 db2cli.ini 文件。发出下列命令：

```
db2 UPDATE CLI CFG FOR SECTION COMMON USING Trace 1
db2 UPDATE CLI CFG FOR SECTION COMMON USING TracePathName <path>
```

例如，其中 <path> 在 Windows 上为 C:\temp\trace，在 Linux 或 UNIX 上为 /tmp/trace。

```
db2 UPDATE CLI CFG FOR SECTION COMMON USING TraceComm 1
db2 UPDATE CLI CFG FOR SECTION COMMON USING TraceFlush 1
db2 UPDATE CLI CFG FOR SECTION COMMON USING TraceTimeStamp 3
```

### 3. 确认 db2cli.ini 配置。

发出以下命令来验证是否设置并选择了正确的关键字:

```
db2 GET CLI CFG FOR SECTION COMMON
```

4. 重新启动应用程序。

仅当应用程序启动时才读取 db2cli.ini 文件, 因此, 要使任何更改生效, 都必须重新启动应用程序。

如果要跟踪 CLI 存储过程, 那么表示要重新启动 DB2 实例。

5. 捕获错误。

运行该应用程序直到生成错误, 然后终止该应用程序。如果可以减少这种情况, 在跟踪时只运行与问题再现有关的应用程序, 那么跟踪分析会清晰得多。

6. 禁用 CLI 跟踪。

在 db2cli.ini 的 [COMMON] 段手动将 **Trace** 关键字设置为值 0, 或者发出:

```
db2 UPDATE CLI CFG FOR SECTION COMMON USING Trace 0
```

然后重新启动可能正在运行和跟踪的所有应用程序。

7. 收集跟踪信息。

CLI 跟踪文件将写至 **TracePathName** 关键字指定的路径。生成的文件名的格式为 <pid>t<tid>.cli, 其中 <pid> 是操作系统指定的进程标识, 而 <tid> 是应用程序进程生成的每个线程的数字计数器 (从 0 开始)。例如, p1234t1.cli。如果要与 DB2 支持机构配合工作来诊断问题, 他们会想要查看跟踪路径中生成的所有文件。

当您使用跟踪工具来诊断应用程序问题时, 记住它对应用程序性能会有影响, 并且会影响所有应用程序而不仅仅是测试程序。这就是一定要记住在标识问题后要关闭跟踪工具的原因。

### 解释 CLI 跟踪文件中的输入和输出参数

与任何常规函数一样, DB2 CLI 函数也有输入和输出参数。在 DB2 CLI 跟踪中, 可以看到这些输入和输出参数, 它们提供有关每个应用程序如何调用特定 CLI API 的大量详细信息。CLI 跟踪中显示的任何 CLI 函数的输入和输出参数都可与文档的“CLI 参考”部分中的 CLI 函数的定义进行比较。

以下是 CLI 跟踪文件中的某个片段:

```
SQLConnect( hDbc=0:1, szDSN="sample", cbDSN=-3, szUID="",
            cbUID=-3, szAuthStr="", cbAuthStr=-3 )
----> Time elapsed - +6.960000E-004 seconds
```

```
SQLRETURN  SQLConnect      (
            SQLHDBC          ConnectionHandle, /* hdbc */
            SQLCHAR          *FAR ServerName,   /* szDSN */
            SQLSMALLINT      NameLength1,      /* cbDSN */
            SQLCHAR          *FAR UserName,     /* szUID */
            SQLSMALLINT      NameLength2,      /* cbUID */
            SQLCHAR          *FAR Authentication, /* szAuthStr */
            SQLSMALLINT      NameLength3);     /* cbAuthStr */
```

CLI 函数的初始调用显示输入参数和指定给它们的值 (如果适当的话)。

在 CLI 函数返回时, 它们将显示作为结果的输出参数, 例如:



```
SQLAllocStmt( phStmt=1:1 )
<--- SQL_SUCCESS Time elapsed - +4.444000E-003 seconds
```

在此情况下，CLI 函数 SQLAllocStmt() 将返回输出参数 phStmt，其值为“1:1”（连接句柄 1，语句句柄 1）。

## 分析 CLI 跟踪中的动态 SQL

DB2 CLI 跟踪还会显示如何通过声明并在 SQLPrepare() 和 SQLBindParameter() 中使用参数标记来执行动态 SQL。这使得您能够确定运行时执行的 SQL 语句。

以下跟踪条目显示 SQL 语句的编译过程（“?”表示参数标记）：

```
SQLPrepare( hStmt=1:1, pszSqlStr=
"select * from employee where empno = ?",
cbSqlStr=-3 )
---> Time elapsed - +1.648000E-003 seconds
( StmtOut="select * from employee where empno = ?" )
SQLPrepare( )
<--- SQL_SUCCESS Time elapsed - +5.929000E-003 seconds
```

以下跟踪条目显示将参数标记绑定为最大长度为 7 的 CHAR：

```
SQLBindParameter( hStmt=1:1, iPar=1, fParamType=SQL_PARAM_INPUT,
fCType=SQL_C_CHAR, fSQLType=SQL_CHAR, cbColDef=7, ibScale=0,
rgbValue=&00854f28, cbValueMax=7, pcbValue=&00858534 )
---> Time elapsed - +1.348000E-003 seconds
SQLBindParameter( )
<--- SQL_SUCCESS Time elapsed - +7.607000E-003 seconds
```

这就执行了动态 SQL 语句。rgbValue="000010" 显示应用程序在运行时用来取代参数标记的值：

```
SQLExecute( hStmt=1:1 )
---> Time elapsed - +1.317000E-003 seconds
( iPar=1, fCType=SQL_C_CHAR, rgbValue="000010" - X"30303030303130",
pcbValue=6, piIndicatorPtr=6 )
sqlccsend( ulBytes - 384 )
sqlccsend( Handle - 14437216 )
sqlccsend( ) - rc - 0, time elapsed - +1.915000E-003
sqlccrecv( )
sqlccrecv( ulBytes - 1053 ) - rc - 0, time elapsed - +8.808000E-003
SQLExecute( )
<--- SQL_SUCCESS Time elapsed - +2.213300E-002 seconds
```

## 解释 CLI 跟踪中的计时信息

有几种方法可用来从 DB2 CLI 跟踪中收集计时信息。在缺省情况下，CLI 跟踪会捕获应用程序中自上次对此特定线程执行 CLP API 调用后所花的时间。它包括客户机与服务器之间的网络时间以及 DB2 中所花的时间。例如：

```
SQLAllocStmt( hDbc=0:1, phStmt=&0012ee48 )
---> Time elapsed - +3.964187E+000 seconds
```

（此时间值指示应用程序中自上次调用 CLI API 后所花的时间）

```
SQLAllocStmt( phStmt=1:1 )
<--- SQL_SUCCESS Time elapsed - +4.444000E-003 seconds
```

（自函数完成后，此时间值指示 DB2 中所花的时间，包括网络时间）



捕获计时信息的另一个方法是使用 CLI 关键字 TraceTimeStamp。此关键字将对每次调用和 DB2 CLI API 调用的结果生成时间戳记。关键字有 4 个显示选项：没有时间戳记信息、处理器标记和 ISO 时间戳记、处理器标记或者 ISO 时间戳记。

在处理与计时有关的问题（如 CLI0125E - 函数顺序错误）时，它会非常有用。如果要在使用多线程应用程序时尝试确定最先发生的事件，它也会非常有用。

### 解释 CLI 跟踪中的未知值

DB2 CLI 函数可返回“未知值”作为 CLI 跟踪中的输入参数。如果 DB2 CLI 驱动程序要查找特定于该输入参数的内容，而应用程序提供了另一个值，那么可能发生这种情况。例如，如果遵循 CLI 函数的过时定义或使用已经不推荐使用的 CLI 函数，那么可能会发生这种情况。

您还可能会看到 CLI 函数调用返回“选项值已更改”或“键集解析器返回码”。这是显示消息的键集游标的结果，如游标因为某些特定原因降级为静态游标。

```
SQLExecDirect( hStmt=1:1, pszSqlStr="select * from org", cbSqlStr=-3 )
    ---> Time elapsed - +5.000000E-002 seconds
( StmtOut="select * from org" )
( COMMIT=0 )
( StmtOut=" SELECT A.TABSCHEMA, ..... )
( StmtOut=" SELECT A.TABSCHEMA, ..... )
( Keyset Parser Return Code=1100 )

SQLExecDirect( )
    <--- SQL_SUCCESS_WITH_INFO Time elapsed - +1.06E+001 seconds
```

在上述 CLI 跟踪中，键集解析器指示返回码为 1100，这表示该表没有唯一索引或主键，因此未能创建键集游标。这些返回码未外部化，所以此时如果您想要有关返回码的含义的更多信息，那么需要与 DB2 支持机构联系。

调用 SQLError 或 SQLDiagRec 将指示游标类型已更改。那么，该应用程序就应该查询游标类型和并行性以确定更改的属性。

### 解释多线程 CLI 跟踪输出

CLI 跟踪可跟踪多线程应用程序。跟踪多线程应用程序的最好方法是使用 CLI 关键字 TracePathName。这将生成跟踪文件 p<pid>t<tid>.cli，其中 <tid> 是应用程序的实际线程标识。

如果需要知道实际线程标识，可在 CLI 跟踪头中看到此信息：

```
[ Process: 3500, Thread: 728 ]
[ Date & Time:          02/17/2006 04:28:02.238015 ]
[ Product:             QDB2/NT DB2 v9.1.0.190 ]
...
```

还可使用 CLI 关键字 TraceFileName 将多线程应用程序跟踪写至一个文件。此方法将生成一个您选择的文件，但阅读此文件会很困难，这是因为在执行一个线程中的特定 API 的同时会执行另一个线程中的另一个 API，因此可能会导致复查跟踪时造成混乱。

通常建议将 TraceTimeStamp 设置为 ON 以便您可以通过查看执行特定 API 的时间来确定事件的真实顺序。对于调查一个线程在另一个线程中导致问题的情况，这会非常有用（如 CLI0125E - 函数顺序错误）。

---

## 特定于平台的工具

存在多种故障诊断命令、性能监视实用程序和其他收集诊断信息的方法，它们均与您正在使用的平台相关。这些工具作为 Windows 操作系统或 Linux 和 UNIX 操作系统的一部分介绍。

### 诊断工具 ( Windows )

下列诊断工具可用于 Windows 操作系统:

#### 事件查看器、性能监视器和其他管理工具

“管理工具”文件夹提供了各种诊断信息，包括访问事件日志和访问性能信息。

#### 任务管理器

“任务管理器”显示正在 Windows 服务器上运行的所有进程以及有关内存使用情况的详细信息。使用此工具来查明哪些 DB2 进程正在运行并诊断性能问题。通过使用此工具，可以确定进程的内存使用情况、内存限制、已使用的交换文件空间以及内存泄漏。

要打开“任务管理器”，按 **Ctrl + Alt + Delete**，并从可用选项中单击**任务管理器**。

#### Dr. Watson

Dr. Watson 实用程序是在“常规故障保护”(GPF)事件中调用的。它记录可能有助于诊断问题的数据，并将此信息保存至文件。必须通过在命令行上输入 `drwatson` 以启动此实用程序。

### 诊断工具 ( Linux 和 UNIX )

本节描述有关在 Linux 和 UNIX 平台上进行故障诊断和性能监视的一些基本命令。有关其中任何一个命令的详细信息，请在命令行上输入它并在前面加上“man”。使用这些命令来收集和處理数据，这些数据有助于确定系统问题的原因。一旦收集了数据，就可由熟悉该问题的人检查它，或者，如果 DB2 支持机构要求，就交给 DB2 支持机构。

#### 故障诊断命令 ( AIX )

下列 AIX 系统命令对于 DB2 故障诊断很有用:

**errpt** `errpt` 命令报告系统错误，如硬件错误和网络故障。

- 要获取每个错误显示一行的概述，使用 `errpt`
- 要获取每个错误显示一页的更为详细的视图，使用 `errpt -a`
- 要获取错误号为“1581762B”的错误，使用 `errpt -a -j 1581762B`
- 要查明过去是否用光了调页空间，使用 `errpt | grep SYSVMM`
- 要查明是否存在令牌环卡或磁盘问题，检查 `errpt` 输出中的短语“disk”和“tr0”

**lspcs** `lspcs -a` 命令监视并显示调页空间的使用情况。

**lsattr** 此命令显示各种操作系统参数。例如，使用以下命令来查明数据库分区上的实内存量:

```
lsattr -l sys0 -E
```

#### xmperf

对于使用 Motif 的 AIX 系统，此命令启动用于收集和显示与系统相关的性能

数据的图形监视器。对于每个数据库分区，监视器都会在单个窗口中显示三维图，这对于高级监视有好处。但是，如果活动很低，那么此监视器的输出将是有限值。

### **spmon**

如果使用系统分区作为 Parallel System Support Programs (PSSP) 的一部分，那么可能需要检查 SP 交换机是否在所有工作站上运行。要查看所有数据库分区的状态，从控制工作站使用下列其中一个命令：

- `spmon -d`，用于 ASCII 输出
- `spmon -g`，用于图形用户界面

或者，从数据库分区工作站使用命令 `netstat -i` 来查看是否关闭了该交换机。如果已关闭高性能交换机，那么数据库分区名旁边会有一个星号 (\*)。例如：

```
css0* 65520 <Link>0.0.0.0.0.0
```

如果启动了高性能交换机，那么不会出现星号。

## **故障诊断命令 (Linux 和 UNIX)**

下列系统命令用于所有 Linux 和 UNIX 系统 (包括 AIX)，除非另行说明。

**df** `df` 命令让您查看文件系统是否已满。

- 要查看所有文件系统 (包括已安装的文件系统) 中有多少可用空间，使用 `df`
- 要查看名称包含“dev”的所有文件系统中有多少可用空间，使用 `df | grep dev`
- 要查看 `home` 文件系统中有多少可用空间，使用 `df /home`
- 要查看文件系统“tmp”中有多少可用空间，使用 `df /tmp`
- 要查看机器上是否有足够的可用空间，检查下列命令的输出：`df /usr`、`df /var`、`df /tmp` 和 `df /home`

**truss** 此命令对于跟踪一个或多个进程中的系统调用很有用。

### **pstack**

可用于 Solaris 2.5.1 或更新版本，`/usr/proc/bin/pstack` 命令显示堆栈回溯信息。`/usr/proc/bin` 目录包含用于调试以暂挂状态出现的进程的其他工具。

## **性能监视工具**

下列工具可用于监视系统的性能。

### **vmstat**

此命令对于确定某个对象是处于暂挂状态还是在长时间运行是很有用的。您可以监视在页面调进 (pi) 和页面调出 (po) 列中找到的页面调度速率。其他很重要的列是已分配虚拟存储器 (avm) 数量和可用虚拟存储器 (fre) 数量。

**iostat** 此命令对于监视 I/O 活动很有用。您可以使用读写速率来估计某些 SQL 操作所需的时间量 (如果它们是系统上唯一的活动)。

### **netstat**

此命令让您了解每个数据库分区上的网络流量以及遇到的错误信息包的数目。它对于隔离网络问题是很有用的。

### **system file**

可用于 Solaris 操作系统，`/etc/system` 文件包含内核配置限制的定义，例如系统

上允许同时存在的最大用户数、每个用户的最大进程数以及有关资源的大小和数量的进程间通信（IPC）限制。这些限制非常重要，因为它们会影响 Solaris 操作系统机器上的 DB2 性能。

---

## 第 5 章 如何有效地搜索已知问题

有许多资源可用于描述已知问题，包括 DB2 APAR、白皮书、IBM Redbooks™（红皮书）出版物、技术说明和手册。所以，有效地搜索这些（及其他）资源以迅速确定您遇到的问题是否已有解决方案是非常重要的。

在搜索之前，应清楚地了解问题的情况。

在清楚地了解问题的情况之后，需要编译搜索关键字列表以增加找到现有解决方案的机率。以下是一些技巧：

1. 在搜索中使用多个字词。您使用的搜索项越切合，搜索结果越准确。
2. 从特定结果开始，然后在必要时扩大搜索结果的范围。例如，如果返回的结果太少，那么除去一些不太切合的搜索项，然后重试。或者，如果不确定要使用的关键字，可使用若干关键字进行广泛搜索，查看接收到的搜索结果类型，您就能够更准确地选择其他关键字。
3. 有时搜索特定短语会更有效。例如，如果输入 "administration notification file"（加上引号），那么只会得到包含内容与您的输入内容和顺序完全相同的短语。（相对于包含这三个单词的任何组合的所有文档）。
4. 使用通配符。如果遇到特定 SQL 错误，那么搜索"SQL5005<wildcard>"，其中 <wildcard> 是您要搜索的资源的相应通配符。这比只搜索"SQL5005"或"SQL5005c"返回的结果要多。
5. 如果遇到实例异常结束并且生成陷阱文件的情况，那么使用陷阱或核心文件的堆栈回溯中的前两个或前三个函数来搜索已知问题。如果返回的结果太多，那么尝试添加关键字"trap"、"abend"或"crash"。
6. 如果搜索特定于操作系统的关键字（如信号编号或错误值），那么尝试针对常量名称而不是值进行搜索。例如，搜索"EFGIG"而不搜索错误号 27。

一般情况下，成功搜索到的搜索项通常包括：

- 描述所运行命令的词汇
- 描述症状的词汇
- 诊断中的标记

---

### 故障诊断资源

提供有大量故障诊断信息，可帮助您使用 DB2 数据库产品。

#### DB2 文档

在 DB2 信息中心以及构成 DB2 资料库的 PDF 书籍中可以找到故障诊断信息。可参阅 DB2 信息中心导航树（在浏览器窗口的左窗格中）的“支持和故障诊断”分支，以查看 DB2 故障诊断文档的完整列表。

## DB2 技术支持 Web 站点

如果您遇到了问题并且想要获取查找可能的原因和解决方案的帮助，请参阅 DB2 技术支持 (DB2 Technical Support) Web 站点。该“技术站点”具有指向最新 DB2 出版物、技术说明、授权程序分析报告 (APAR)、修订包的链接、内部 DB2 错误代码的最新列表以及其他资源。可搜索此知识库并查找问题的可能解决方案。

访问位于以下网址的 DB2 技术支持 Web 站点: <http://www.ibm.com/software/data/db2/udb/support>

---

## 第 6 章 获取 DB2 产品修订

修订包中包含适用于 IBM 在产品测试期间发现的问题以及客户使用产品时发现的问题的代码更新和修订。将讨论如何找到最新修订包以及如何对数据库环境应用这些修订。

---

### 应用修订包

建议使 DB2 环境始终在最新修订包级别运行，以确保操作不会出现问题。要成功安装修订包，应执行安装前后必须执行的所有任务。

DB2 修订包包含对 IBM 公司在测试期间发现的问题（“授权程序分析报告”或“APAR”）所作的更新和修订以及对客户报告的问题所作的修订。每个修订包都包含一个 APARLIST.TXT 文件，用于描述该修订包包含的修订。

修订包是不断累积的。这意味着任何给定版本的 DB2 的最新修订包都包含同一版本的 DB2 的先前修订包中的所有更新。

可用的修订包映像为：

- 单一服务器映像。

单一服务器映像包含所有 DB2 服务器产品和 IBM 数据服务器客户机所需的新的和更新的代码。如果多个 DB2 服务器产品已安装在一个位置，那么 DB2 服务器修订包会对所有已安装的 DB2 服务器产品应用维护代码更新。数据服务器客户机修订包已包含在一个 DB2 服务器修订包中（即，可维护以下任一服务器产品的修订包中：企业服务器版、工作组服务器版、易捷版、个人版、Connect 企业版、Connect 应用程序服务器版、适用于 zSeries® 的 Connect 无限制版和适用于 i5/OS® 的 Connect 无限制版）。您可以使用 DB2 服务器修订包来升级数据服务器客户机。

单一服务器映像还可用于安装处于特定修订包级别且缺省情况下具有 DB2 “先试后买”许可证的任何 DB2 数据库服务器产品。

- 其他每个 DB2 数据库产品的修订包。

仅在安装了非服务器数据库产品或附件产品的情况下才使用此修订包。例如，IBM 数据服务器运行时客户机或 Query Patroller。

如果已安装的 DB2 产品只有 DB2 服务器产品或数据服务器客户机，那么切勿使用此类修订包。而是应使用单一服务器映像修订包。

对于 Windows 平台，如果在单个 DB2 副本中安装了多个 DB2 数据库产品（包括至少一个非数据服务器客户机或 DB2 服务器的产品），那么在开始修订包安装过程之前，必须下载并解压缩所有特定于相应产品的修订包。

- 通用修订包（仅适用于 Linux 或 UNIX 平台）。

在已经安装了多个 DB2 数据库产品的情况下进行安装时，就可以使用通用修订包。



如果已安装的 DB2 产品只有 DB2 服务器产品或 数据服务器客户机，那么不需要通用修订包。在此情况下，应使用单一服务器映像修订包。

### 限制

- DB2 版本 9.5 修订包只能应用于 DB2 版本 9.5 一般可用性 (GA) 或修订包级别副本。
  - 在安装修订包之前，必须停止所有 DB2 实例、DAS 以及与正在更新的 DB2 副本相关的应用程序。
  - 如果使用数据库分区功能 (DPF)，那么在安装修订包之前，必须停止所有节点上的数据库管理器。您必须在拥有节点的实例以及所有其他分区节点上安装修订包。必须将参与实例的所有计算机都升级到同一修订包级别。
  - 在 Linux 或 UNIX 操作系统上：
    - 如果在网络文件系统 (NFS) 上具有 DB2 产品，那么必须确保在安装修订包之前完全停止下列各项：所有实例、DB2 管理服务器 (DAS)、进程间通信 (IPC) 和其他使用相同 NFS 安装的机器上的应用程序。
    - 如果系统命令 fuser 或 lsof 不可用，那么 installFixPack 命令无法检测已装入的 DB2 文件。您必须确保未装入 DB2 文件，并提供覆盖选项来安装修订包。在 UNIX 上，需要 fuser 命令来检查已装入的文件。在 Linux 上，需要 fuser 命令或 lsof 命令。
- 有关覆盖选项的详细信息，请参阅 installFixPack 命令。
- 在客户机应用程序上，在应用修订包之后，用户必须具有绑定权限，才能执行应用程序自动绑定。
  - 安装 DB2 修订包将不维护数据服务器管理者工具或数据服务器开发者工具。

对于 Linux 或 UNIX 上的非 root 用户安装，可以使用 db2rfe 命令来启用基于 root 用户的功能（例如，高可用性和基于操作系统的认证）。如果在安装 DB2 产品之后启用基于 root 用户的功能，那么您每次应用修订包时都必须重新运行 db2rfe 命令，以便重新启用这些功能。有关详细信息，请参阅以下非 root 用户相关的链接。

在 Linux 或 UNIX 操作系统上，如果安装了本地语言，那么还需要一个单独的本地语言修订包。不能单独安装本地语言修订包。必须同时应用通用修订包或特定于产品的修订包，并且这两种修订包必须处于相同修订包级别。例如，如果要在 Linux 或 UNIX 上将通用修订包应用于非英语版的 DB2 数据库产品，那么必须同时应用通用修订包和本地语言修订包来更新 DB2 数据库产品。

如果同一系统上有多个 DB2 副本，那么这些副本可以处于不同的版本和修订包级别。如果要将修订包应用于一个或多个 DB2 副本，那么必须在那些 DB2 副本上逐个安装该修订包。

要安装修订包：

1. 通过从 DB2 支持 Web 站点（网址为 <http://www.ibm.com/software/data/db2/udb/support.html>）选择修订包下载的连接，访问并下载最新的 DB2 修订包。
2. 检查修订包先决条件，然后在安装修订包之前执行必需的任务。如果已经安装了 DB2 数据库产品，那么还必须停止各种 DB2 进程。请参阅修订包自述文件以了解详细信息。
3. 选择修订包安装方法并安装修订包。

检查日志文件以获得任何安装后的步骤或错误消息和建议的操作。

---

## 修订包、测试修订和 APAR

授权程序分析报告 (APAR) 是当前未更改发行版的 IBM 程序中可疑缺陷导致的问题的书面报告。APAR 描述 IBM 测试期间发现的问题以及客户报告的问题。

可在修订包或测试修订中发布解决了 APAR 中所描述问题的已修改 DB2 代码。

**修订包** 修订包是 APAR 修订的累积集合。修订包尤其说明 DB2 的新发行版之间出现的 APAR。它们允许您上移至特定维护级别。修订包具有以下特征:

- 它们是累积的。特定 DB2 发行版的修订包将替代或包含所有此发行版先前修订包发出的所有 APAR 修订。
- 它们包含多个 APAR。
- 它们发布在 DB2 技术支持 Web 站点上, 且通常可供购买了符合 Passport Advantage® 程序的产品的用户使用。
- 它们已经过 IBM 的完整测试。
- 它们包含自述文件和一系列发行说明。
  - 修订包自述文件中提供了有关安装和除去该修订包的指示信息。
  - 发行说明中包含有关对产品所作的更改的信息。

**注:** 当 APAR 修订提供在修订包中时, APAR 的状态将从“Open”改为“Closed”。可通过检查 DB2 技术支持 Web 站点上的 APAR 描述来确定单个 APAR 的状态。

### 测试修订

测试修订是针对报告的问题提供给特定用户进行测试的临时解决方案。测试修订具有下列特征:

- 它们通常包含单个 APAR。
- 它们通过 DB2 支持机构获得且通常不对公众开放。
- 它们经过了 IBM 的有限测试。
- 它们包含最少的文档记录, 包括应如何应用测试修订的描述、修订的 APAR 以及除去测试修订的指示信息。

当未解决新问题、问题没有变通方法或无法等到下一修订包可用时, 就会提供测试修订。例如, 如果问题导致对业务产生极大影响, 将提供测试修订以缓解这种状况直到在修订包中描述 APAR。

建议使 DB2 环境始终在最新修订包级别运行, 以确保操作不会出现问题。要接收可用的新修订包的通知, 请向 DB2 技术支持 Web 站点 (<http://www.ibm.com/software/data/db2/udb/support>) 上的 My Support 电子邮件更新进行预订。

---

## 应用测试修订

测试修订是针对报告的问题提供给特定用户进行测试的临时修订。每个测试修订都有自述文件。测试修订自述文件提供了有关安装和卸载该测试修订的指示信息以及包括在测试修订中的 APAR 列表 (如果有的话)。

每个测试修订都有特定的先决条件。有关详细信息, 请参阅该测试修订附带提供的自述文件。

有两种类型的测试修订:

- 单个 DB2 产品的测试修订。这些测试修订可应用于该产品的现有安装,也可用在未安装 DB2 的情况下执行完整产品安装。
- 通用测试修订(仅适用于 Linux 和 UNIX)。在已经安装了多个 DB2 产品的情况下进行安装时,就可以使用通用测试修订。

如果安装了本地语言,那么可能还需要一个单独的本地语言测试修订。仅当本地语言测试修订与已安装的 DB2 产品处于同一测试修订级别时,才能应用该本地语言测试修订。如果您要应用通用测试修订,那么必须同时应用通用测试修订和本地语言测试修订来更新 DB2 产品。

从 DB2 客户支持机构获取测试修订,并遵循自述文件中有关安装、测试和除去(如果需要的话)该测试修订的指示信息。

当在一个多分区数据库分区环境中安装测试修订时,该系统必须处于脱机状态,并且必须将参与实例的所有计算机都升级到同一测试修订级别。

---

## 附录 A. DB2 技术信息概述

可以通过下列工具和方法获取 DB2 技术信息:

- DB2 信息中心
  - 主题 (任务、概念和参考主题)
  - DB2 工具的帮助
  - 样本程序
  - 教程
- DB2 书籍
  - PDF 文件 (可下载)
  - PDF 文件 (在 DB2 PDF DVD 中)
  - 印刷版书籍
- 命令行帮助
  - 命令帮助
  - 消息帮助

**注:** DB2 信息中心主题的更新频率比 PDF 书籍或硬拷贝书籍的更新频率高。要获取最新信息, 请安装可用的文档更新, 或者参阅 [ibm.com](http://www.ibm.com)<sup>®</sup> 上的 DB2 信息中心。

可以在线访问 [ibm.com](http://www.ibm.com) 上的其他 DB2 技术信息, 如技术说明、白皮书和 IBM Redbooks 出版物。访问位于以下网址的 DB2 信息管理软件库站点: <http://www.ibm.com/software/data/sw-library/>。

### 文档反馈

我们非常重视您对 DB2 文档的反馈。如果您想就如何改善 DB2 文档提出建议, 请将电子邮件发送至 [db2docs@ca.ibm.com](mailto:db2docs@ca.ibm.com)。DB2 文档小组会阅读您的所有反馈, 但不能直接答复您。请尽可能提供具体的示例, 这样我们才能更好地了解您所关心的问题。如果您要提供有关具体主题或帮助文件的反馈, 请加上标题和 URL。

请不要用以上电子邮件地址与 DB2 客户支持机构联系。如果您遇到文档不能解决的 DB2 技术问题, 请与您当地的 IBM 服务中心联系以获得帮助。

---

## 硬拷贝或 PDF 格式的 DB2 技术库

下列各表描述 IBM 出版物中心 (网址为 [www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) 提供的 DB2 资料库。可以从 [www.ibm.com/support/docview.wss?rs=71&uid=swg2700947](http://www.ibm.com/support/docview.wss?rs=71&uid=swg2700947) 下载 PDF 格式的英文 DB2 版本 9.5 手册和已翻译的版本。

尽管这些表标识书籍有印刷版, 但可能未在您所在国家或地区提供。

表 6. DB2 技术信息

书名	书号	是否提供印刷版
<i>Administrative API Reference</i>	SC23-5842-00	是

表 6. DB2 技术信息 (续)

书名	书号	是否提供印刷版
<i>Administrative Routines and Views</i>	SC23-5843-00	否
<i>Call Level Interface Guide and Reference, Volume 1</i>	SC23-5844-00	是
<i>Call Level Interface Guide and Reference, Volume 2</i>	SC23-5845-00	是
<i>Command Reference</i>	SC23-5846-00	是
《数据移动指南和参考》	S151-0617-00	是
《数据恢复及高可用性指南与参考》	S151-0619-00	是
《数据服务器、数据库和数据库对象指南》	S151-0612-00	是
《数据库安全性指南》	S151-0614-00	是
<i>Developing ADO.NET and OLE DB Applications</i>	SC23-5851-00	是
<i>Developing Embedded SQL Applications</i>	SC23-5852-00	是
<i>Developing Java Applications</i>	SC23-5853-00	是
<i>Developing Perl and PHP Applications</i>	SC23-5854-00	否
<i>Developing User-defined Routines (SQL and External)</i>	SC23-5855-00	是
<i>Getting Started with Database Application Development</i>	GC23-5856-00	是
《Linux 和 Windows 上的 DB2 安装和管理入门》	G151-0623-00	是
《国际化指南》	S151-0616-00	是
《消息参考, 第 1 卷》	G151-0632-00	否
《消息参考, 第 2 卷》	G151-0633-00	否
《迁移指南》	G151-0622-00	是
《Net Search Extender 管理和用户指南》	S151-0761-00	是
注: DB2 信息中心中并不包含此文档的内容		
《分区和集群指南》	S151-0615-00	是
<i>Query Patroller Administration and User's Guide</i>	SC23-8507-00	是
《IBM 数据服务器客户机快速入门》	G151-0625-00	否
《DB2 服务器快速入门》	G151-0624-00	是
《Spatial Extender 和地理数据管理功能部件用户指南和参考》	S151-0760-00	是
<i>SQL Reference, Volume 1</i>	SC23-5861-00	是
<i>SQL Reference, Volume 2</i>	SC23-5862-00	是

表 6. DB2 技术信息 (续)

书名	书号	是否提供印刷版
《系统监视器指南和参考》	S151-0618-00	是
《文本搜索指南》	S151-0620-00	是
《故障诊断指南》	G151-0621-00	否
《调整数据库性能》	S151-0613-00	是
《Visual Explain 教程》	S151-0634-00	否
《新增内容》	S151-0629-00	是
<i>Workload Manager Guide and Reference</i>	SC23-5870-00	是
《pureXML 指南》	S151-0630-00	是
《XQuery 参考》	S151-0631-00	否

表 7. 特定于 DB2 Connect 的技术信息

书名	书号	是否提供印刷版
《DB2 Connect 个人版快速入门》	G151-0627-00	是
《DB2 Connect 服务器快速入门》	G151-0628-00	是
《DB2 Connect 用户指南》	S151-0626-00	是

表 8. Information Integration 技术信息

书名	书号	是否提供印刷版
<i>Information Integration: Administration Guide for Federated Systems</i>	SC19-1020-01	是
<i>Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i>	SC19-1018-02	是
<i>Information Integration: 《联合数据源配置指南》</i>	S151-0468-00	否
<i>Information Integration: 《SQL 复制指南和参考》</i>	S151-0475-00	是
<i>Information Integration: Introduction to Replication and Event Publishing</i>	SC19-1028-01	是

## 订购印刷版的 DB2 书籍

如果您需要印刷版的 DB2 书籍，可以在许多（但不是所有）国家或地区在线购买。无论何时都可以从当地的 IBM 代表处订购印刷版的 DB2 书籍。请注意，DB2 PDF 文档 DVD 上的某些软拷贝书籍没有印刷版。例如，DB2 消息参考的任何一卷都没有提供印刷版书籍。

只要支付一定费用，就可以从 IBM 获取 DB2 PDF 文档 DVD，该 DVD 包含许多 DB2 书籍的印刷版。根据您下订单的位置，您可能能够从 IBM 出版物中心在线订购书籍。如果在线订购在您所在国家或地区不可用，您总是可以从当地的 IBM 代表处订购印刷版 DB2 书籍。注意，并非 DB2 PDF 文档 DVD 上的所有书籍都有印刷版。

注：最新最完整的 DB2 文档保留在网址如下的 DB2 信息中心中：<http://publib.boulder.ibm.com/infocenter/db2luw/v9r5>。

要订购印刷版的 DB2 书籍：

- 要了解您是否可从所在国家或地区在线订购印刷版的 DB2 书籍，可查看 IBM 出版物中心站点，网址为：<http://www.ibm.com/shop/publications/order>。必须先选择国家、地区或语言才能访问出版物订购信息，然后再按照针对您所在位置的订购指示信息进行订购。
- 要从当地的 IBM 代表处订购印刷版的 DB2 书籍：
  1. 从下列其中一个 Web 站点找到当地代表处的联系信息：
    - IBM 全球联系人目录，网址为 [www.ibm.com/planetwide](http://www.ibm.com/planetwide)
    - IBM 出版物 Web 站点，网址为 <http://www.ibm.com/shop/publications/order>。必须先选择国家、地区或语言才能访问对应您的所在地的出版物主页。在此页面中访问“关于此站点”链接。
  2. 请在致电时说明您想订购 DB2 出版物。
  3. 请向您当地的代表处提供想要订购的书籍的书名和书号。有关书名和书号的信息，请参阅第 93 页的『硬拷贝或 PDF 格式的 DB2 技术库』。

---

## 从命令行处理器显示 SQL 状态帮助

DB2 返回描述 SQL 语句执行结果的 SQLSTATE。SQLSTATE 帮助说明 SQL 状态和 SQL 状态类代码的含义。

要调用 SQL 状态帮助，打开命令行处理器并输入：

```
? sqlstate or ? class code
```

其中，*sqlstate* 表示有效的 5 位 SQL 状态，*class code* 表示该 SQL 状态的前 2 位。

例如，? 08003 显示 08003 SQL 状态的帮助，而 ? 08 显示 08 类代码的帮助。

---

## 访问不同版本的 DB2 信息中心

对于 DB2 版本 9.5 主题，DB2 信息中心 URL 为 <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>

对于 DB2 版本 9 主题，DB2 信息中心 URL 为 <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>

对于 DB2 版本 8 主题，请访问以下版本 8 信息中心 URL：<http://publib.boulder.ibm.com/infocenter/db2luw/v8/>



---

## 在 DB2 信息中心中以您的首选语言显示主题:

DB2 信息中心尝试以您在浏览器首选项中指定的语言显示主题。如果未提供主题的首选语言翻译版本,那么 DB2 信息中心将显示该主题的英文版。

- 要在 Internet Explorer 浏览器中以您的首选语言显示主题:

1. 在 Internet Explorer 中,单击工具 —> **Internet 选项** —> **语言...**按钮。“语言首选项”窗口打开。
2. 确保您的首选语言被指定为语言列表中的第一个条目。
  - 要将新语言添加至列表,单击**添加...**按钮。

**注:** 添加语言并不能保证计算机具有以首选语言显示主题所需的字体。

- 要将语言移至列表顶部,选择该语言并单击**上移**按钮直到该语言成为语言列表中的第一个条目。
3. 清除浏览器高速缓存然后刷新页面以便以首选语言显示 DB2 信息中心。
- 要在 Firefox 或 Mozilla 浏览器中以首选语言显示主题:
  1. 在**工具** —> **选项** —> **高级对话框**中的**语言**部分中选择按钮。“语言”面板将显示在“首选项”窗口中。
  2. 确保您的首选语言被指定为语言列表中的第一个条目。
    - 要将新语言添加至列表,单击**添加...**按钮以从“添加语言”窗口中选择一种语言。
    - 要将语言移至列表顶部,选择该语言并单击**上移**按钮直到该语言成为语言列表中的第一个条目。
  3. 清除浏览器高速缓存然后刷新页面以便以首选语言显示 DB2 信息中心。

在某些浏览器和操作系统组合上,可能还必须将操作系统的区域设置更改为您选择的语言环境和语言。

---

## 更新安装在您的计算机或内部网服务器上的 DB2 信息中心

如果已经在本地安装了 DB2 信息中心,您可以下载并安装 IBM 提供的更新。

更新在本地安装的 DB2 信息中心要求您:

1. 停止计算机上的 DB2 信息中心,然后以独立方式重新启动信息中心。如果以独立方式运行信息中心,那么网络上的其他用户将无法访问信息中心,因而您可以下载和应用更新。
2. 使用“更新”功能部件来查看可用的更新。如果有您希望安装的更新,那么请使用“更新”功能部件来下载并安装这些更新。

**注:** 如果您的环境要求在一台未连接至因特网的机器上安装 DB2 信息中心更新,那么必须使用一台已连接至因特网的机器将更新站点镜像至本地文件系统并安装 DB2 信息中心。如果网络中有许多用户将安装文档更新,那么可以通过在本地也为更新站点建立镜像并为更新站点创建代理来缩短每个人执行更新所需要的时间。

如果提供了更新包,那么使用更新功能来下载这些更新包。但是,只有在单机方式下才能使用更新功能。

3. 停止独立信息中心,然后在计算机上重新启动 DB2 信息中心。

**注：**在 Windows Vista 上，必须以管理员身份才能运行下面所列示的命令。要启动具有所有管理员特权的命令提示符或图形工具，右键单击快捷方式，然后选择**以管理员身份运行**。

要更新安装在您的计算机或内部网服务器上的 DB2 信息中心：


1. 停止 DB2 信息中心。

- 在 Windows 上，单击**开始** → **控制面板** → **管理工具** → **服务**。右键单击 **DB2 信息中心** 服务，并选择**停止**。
- 在 Linux 上，输入以下命令：  
`/etc/init.d/db2icdv95 stop`

2. 以独立方式启动信息中心。

- 在 Windows 上：
  - a. 打开命令窗口。
  - b. 浏览至信息中心的安装位置。缺省情况下，DB2 信息中心安装在 <Program Files>\IBM\DB2 Information Center\Version 9.5 目录中，其中 <Program Files> 表示 Program Files 目录的位置。
  - c. 从安装目录浏览至 doc\bin 目录。
  - d. 运行 help\_start.bat 文件：  
`help_start.bat`
- 在 Linux 上：
  - a. 浏览至信息中心的安装位置。缺省情况下，DB2 信息中心安装在 /opt/ibm/db2ic/V9.5 目录中。
  - b. 从安装目录浏览至 doc/bin 目录。
  - c. 运行 help\_start 脚本：  
`help_start`

系统缺省 Web 浏览器将启动以显示独立信息中心。

3. 单击“更新”按钮 (  )。在信息中心的右边面板上，单击**查找更新**。将显示现有文档的更新列表。
4. 要启动下载进程，请检查您想要下载的选项，然后单击**安装更新**。
5. 在完成下载和安装进程后，单击**完成**。
6. 停止独立信息中心。

- 在 Windows 上，浏览至安装目录的 doc\bin 目录并运行 help\_end.bat 文件：  
`help_end.bat`

**注：**help\_end 批处理文件包含安全终止用 help\_start 批处理文件启动的进程所需的命令。不要使用 Ctrl-C 或任何其他方法来终止 help\_start.bat。

- 在 Linux 上，浏览至安装目录的 doc/bin 目录并运行 help\_end 脚本：  
`help_end`

**注：**help\_end 脚本包含安全终止用 help\_start 脚本启动的进程所需的命令。不要使用任何其他方法来终止 help\_start 脚本。

7. 重新启动 DB2 信息中心。

- 在 Windows 上，单击开始 → 控制面板 → 管理工具 → 服务。右键单击 **DB2 信息中心** 服务，并选择启动。
- 在 Linux 上，输入以下命令：  
`/etc/init.d/db2icdv95 start`

更新后的 DB2 信息中心将显示新的主题和更新后的主题。

---

## DB2 教程

DB2 教程帮助您了解 DB2 产品的各个方面。这些课程提供了逐步指示信息。

### 开始之前

可从信息中心查看 XHTML 版的教程：<http://publib.boulder.ibm.com/infocenter/db2help/>。

某些课程使用了样本数据或代码。有关其特定任务的任何先决条件的描述，请参阅教程。

### DB2 教程

要查看教程，请单击标题。

《*pureXML 指南*》中的『**pureXML™**』

设置 DB2 数据库以存储 XML 数据以及如何对本机 XML 数据存储执行基本操作。

《*Visual Explain 教程*》中的『**Visual Explain**』

使用 Visual Explain 来分析、优化和调整 SQL 语句以获取更好的性能。

---

## DB2 故障诊断信息

很多故障诊断和问题确定信息可帮助您使用 DB2 产品。

### DB2 文档

故障诊断信息可在 DB2 信息中心的“DB2 故障诊断指南”或“支持和故障诊断”部分找到。可在该处找到有关如何使用 DB2 诊断工具和实用程序隔离和找出问题的信息、某些最常见问题的解决方案以及有关如何解决使用 DB2 产品时可能遇到的问题建议。

### DB2 技术支持 Web 站点

如果您遇到了问题并且想要获取查找可能的原因和解决方案的帮助，请参阅 DB2 技术支持 Web 站点。该“技术支持”站点具有指向最新 DB2 出版物、技术说明、授权程序分析报告（APAR 或错误修订）、修订包和其他资源的链接。可搜索此知识库并查找问题的可能解决方案。

访问位于以下网址的 DB2 技术支持 Web 站点：<http://www.ibm.com/software/data/db2/udb/support.html>

---

## 条款和条件

如果符合以下条款和条件，那么授予您使用这些出版物的准用权。

**个人使用:** 只要保留所有的专有权声明, 您就可以为个人、非商业使用复制这些出版物。未经 IBM 明确同意, 您不可以分发、展示或制作这些出版物或其中任何部分的演绎作品。

**商业使用:** 只要保留所有的专有权声明, 您就可以仅在企业内复制、分发和展示这些出版物。未经 IBM 明确同意, 您不可以制作这些出版物的演绎作品, 或者在您的企业外部复制、分发或展示这些出版物或其中的任何部分。

除非本准用权中有明确授权, 不得把其他准用权、许可或权利(无论是明示的还是暗含的)授予其中包含的出版物或任何信息、数据、软件或其他知识产权。

当使用这些出版物损害了 IBM 的利益, 或者根据 IBM 的规定, 未正确遵守上述指导说明时, 那么 IBM 保留自主决定撤销本文授予的准用权的权利。

您不可以下载、出口或再出口本信息, 除非完全遵守所有适用的法律和法规, 包括所有美国出口法律和法规。

IBM 对这些出版物的内容不作任何保证。这些出版物“按现状”提供, 不附有任何种类的(无论是明示的还是暗含的)保证, 包括但不限于暗含的关于适销和适用于某种特定用途的保证。

---

## 附录 B. 声明

本信息是为在美国提供的产品和服务编写的。

IBM 可能在其他国家或地区不提供本文中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，那么由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往：

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

有关双字节（DBCS）信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

**本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区：** International Business Machines Corporation“按现状”提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些国家或地区在某些交易中不允许免除明示或暗含的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本资料中描述的产品和 / 或程序进行改进和 / 或更改，而不另行通知。

本文档可提供非 IBM Web 站点和资源的链接和引用。IBM 对本文中引用、提供或链接到的任何非 IBM Web 站点或第三方资源不承担任何责任、保证或其他义务。对非 IBM Web 站点的链接并不意味着 IBM 认可这些 Web 站点或其拥有者的内容或用途。此外，IBM 并不是您使用第三方可进入的任何事务的一方，也不会为这些事务负责，即使您是通过 IBM 站点了解或链接到第三方。因此，您承认并认可 IBM 不会为这些外部站点或资源的可用性负责，也不会为它们提供的任何内容、服务、产品或其他事务承担责任或义务。第三方提供的任何软件都遵守与该软件相关的许可条款和条件。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：（i）允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换，以及（ii）允许对已经交换的信息进行相互使用，请与下列地址联系：

IBM Canada Limited  
Office of the Lab Director  
8200 Warden Avenue  
Markham, Ontario  
L6G 1C7  
CANADA

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或任何同等协议中的条款提供。

此处包含的任何性能数据都是在受控环境中测得的。因此，在其他操作环境中获得的数据可能会有明显的不同。有些测量可能是在开发级的系统上进行的，因此不保证与一般可用系统上进行的测量结果相同。此外，有些测量是通过推算而估计的，实际结果可能会有差异。本文档的用户应当验证其特定环境的适用数据。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。

所有关于 IBM 未来方向或意向的声明都可随时更改或收回，而不另行通知，它们仅仅表示了目标和意愿而已。

本信息可能包含在日常业务操作中使用的数据和报告的示例。为了尽可能完整地说明这些示例，示例中可能会包括个人、公司、品牌和产品的名称。所有这些名称都是虚构的，与实际商业企业所用的名称和地址的任何雷同纯属巧合。

版权许可：

本信息可能包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口（API）进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此，IBM 不能担保或暗示这些程序的可靠性、可维护性或功能。

凡这些样本程序的每份拷贝或其任何部分或任何衍生产品，都必须包括如下版权声明：

©（贵公司的名称）（年）。此部分代码是根据 IBM 公司的样本程序衍生出来的。© Copyright IBM Corp.（输入年份）。All rights reserved.

## 商标

DB2 版本 9.5 文档库的各个文档中标识的公司、产品或服务名称可能是 International Business Machines Corporation 或其他公司的商标或服务标记。有关 IBM Corporation 在美国和 / 或其他国家的商标的信息在 <http://www.ibm.com/legal/copytrade.shtml> 中。

下列各项是其他公司的商标或注册商标, 且已在 DB2 文档库中的至少一份文档中使用:

Microsoft<sup>®</sup>、Windows、Windows NT<sup>®</sup> 和 Windows 徽标是 Microsoft Corporation 在美国和 / 或其他国家或地区的商标。

Intel<sup>®</sup>、Intel 徽标、Intel Inside<sup>®</sup> 徽标、Intel Centrino<sup>®</sup>、Intel Centrino 徽标、Celeron<sup>®</sup>、Intel Xeon<sup>®</sup>、Intel SpeedStep<sup>®</sup>、Itanium<sup>®</sup> 和 Pentium<sup>®</sup> 是 Intel Corporation 在美国和 / 或其他国家或地区的商标。

Java 和所有基于 Java 的商标是 Sun Microsystems,Inc. 在美国和 / 或其他国家或地区的商标。

UNIX 是 The Open Group 在美国和其他国家或地区的注册商标。

Linux 是 Linus Torvalds 在美国和其他国家或地区的注册商标。

Adobe<sup>®</sup>、Adobe 徽标、PostScript<sup>®</sup> 和 PostScript 徽标是 Adobe Systems Incorporated 在美国和 / 或其他国家或地区的注册商标或商标。

其他公司、产品或服务名称可能是其他公司的商标或服务标记。





# 索引

## [ A ]

- 安装
  - 错误日志 26
  - 手动
    - 列示产品 54
- 安装问题
  - 分析 28

## [ B ]

- 帮助
  - 显示 97
  - 有关 SQL 语句 96

## [ C ]

- 参数
  - PRDID 70
- 操作系统
  - 故障诊断工具 84
- 测试修订
  - 类型 91
  - 描述 91
  - 应用 91
- 存储键
  - 故障诊断 32
- 错误
  - 故障诊断 39
- 错误消息
  - DB2 Connect 41

## [ D ]

- 当前发行版
  - 故障诊断 29
- 订购 DB2 书籍 95

## [ F ]

- 发送缓冲区
  - 跟踪数据 69
- 返回码
  - 内部 45
- 分布式关系数据库体系结构 (DRDA)
  - 跟踪 69
- 分布式数据管理 (DDM) 69
- 分区数据库
  - 故障诊断 36
- 分析诊断数据 27

## [ G ]

- 跟踪
    - 概述 65
    - 输出文件 69, 70
    - 输出文件样本 72
    - 有关 DRDA 跟踪的缓冲区信息 76
    - CLI 79
      - 分析 81, 82, 83
    - DB2 connect 与服务器之间的数据 69
    - DRDA
      - 解释 69
  - 跟踪工具
    - 故障诊断概述 65
    - 控制中心跟踪 77
    - CLI 应用程序 80
    - DB2 跟踪 66, 67, 68
    - DRDA 跟踪 72, 76
    - JDBC 应用程序 79
    - 跟踪选项配置 77
  - 更新
    - 信息中心 97
    - DB2 信息中心 97
  - 工具
    - 诊断 41, 84
      - Windows 84
  - 工作负载管理
    - 故障诊断 35
  - 故障诊断 1, 20
    - 存储键 32
    - 当前发行版 29
    - 概述 1, 39
    - 跟踪工具 65, 66
      - 控制中心跟踪 77
    - CLI 和 ODBC 应用程序 79, 80
    - DRDA 72, 76
    - JDBC 应用程序 77, 79
  - 工具 45
  - 简介 1
  - 教程 99
  - 联机信息 99
  - 连接 39, 40
  - 描述 25
  - 收集信息 25, 28, 39, 51, 55, 63
  - 搜索问题的解决方案 87
  - 问题重现 51
  - 资源 87
    - DB2 Connect 41
- 关于本书 v
- 管理日志文件 9
- 管理通知日志 5

## 管理通知日志 (续)

- 简介 9
- 解释 10

## [ H ]

### 核心文件

- 标识 19
- 问题确定 41
- Linux 系统 18
- UNIX 系统 18

## [ J ]

### 教程

- 故障诊断和问题确定 99
- Visual Explain 99
- 交换服务器属性命令 70
- 脚本
  - 故障诊断 35
- 接收缓冲区 69
- 结束工作单元应答消息 (ENDUOWRM) 70
- 进程状态实用程序 41, 70

## [ K ]

### 控制中心

- 跟踪 77

## [ L ]

### 落实命令

- 跟踪输出缓冲区 70

## [ M ]

### 命令

- 落实 70
- ACCRDB 70
- ACCRDBRM 70
- ACCSEC 70
- EXCSAT 70
- EXCSATRD 70
- SECCHK 70
- 命令编辑器
  - 故障诊断 8

## [ Q ]

### 全局变量

- 故障诊断 33

### 全局注册表

- 改变 50

## [ R ]

### 日志文件

- 管理 9

## [ S ]

### 声明 101

### 实用程序

- 跟踪 69
- 进程状态 70
- db2drdat 69
- ps (进程状态) 41, 70

### 事件监视器

- 故障诊断 7

### 收集数据 25, 26, 27

### 收集诊断信息

- 配置 3
- 手动 3
- 自动 3

### 首次出现数据捕获

- 简介 3

### 首次出现数据捕获 (FODC)

- 描述 5
- 数据生成 4
- 子目录 4

### 首次故障服务日志 41

### 首次故障数据捕获 (FFDC)

- 特定于平台的 17
- 陷阱文件 17
- 转储文件 16

### 授权程序分析报告 (APAR) 91

### 数据不一致 36

### 数据仓库中心

- 故障诊断 8
- 信息目录中心
  - 故障诊断 8

### 数据库

- 毁坏 36
- name
  - RDBNAM 对象 70

### 数据库分析和报告工具命令

- 故障诊断概述 46

### 数据库弹性

- 故障诊断 32

### 搜索

- 技巧 87

### 索引数据

- 一致性 36
- 索引数据一致性 36

## [ T ]

### 条款和条件

- 出版物的使用 99

通知级别配置参数  
更新 10

## [ W ]

文档  
    使用条款和条件 99  
    PDF 或印刷的 93  
文档概述 93  
问题确定  
    教程 99  
    联机信息 99  
    连接后问题 40  
    连接问题 39  
    诊断工具 41

## [ X ]

系统核心文件  
    Linux 18  
    UNIX 18  
系统命令 (HP-UX)  
    xdb 19  
系统命令 (Linux)  
    gdb 19  
系统命令 (UNIX)  
    dbx 19  
线程 35  
陷阱文件 17  
    格式化 (Windows) 17  
信息中心  
    版本 96  
    更新 97  
    以各种语言查看 97  
修订包  
    获取 89  
    描述 91  
    应用 89

## [ Y ]

压缩字典  
    未创建 32  
引擎进程 35  
印刷版书籍  
    订购 95  
优化  
    准则  
        故障诊断 29  
优化概要文件  
    故障诊断 29  
与 IBM 联系 28

## [ Z ]

诊断工具 84  
    问题确定 41  
    Linux 18  
    UNIX 18  
    Windows 19, 84  
        访问 Dr. Watson 日志 20  
诊断日志文件  
    简介 11  
诊断信息 5, 17, 28  
    分析 27  
    获取 25, 26, 27  
    收集 3  
诊断 FCM 问题 36  
转储文件  
    错误报告 16

## A

ACCRDB 命令 70  
ACCRDBRM 命令 70  
ACCSEC 命令 70

## C

CLI (调用级接口)  
    跟踪  
        故障诊断概述 79  
    跟踪工具  
        启动 80  
CLI 应用程序  
    跟踪工具配置 80  
CLI/ODBC/JDBC  
    跟踪  
        故障诊断概述 79

## D

DB2  
    安装  
        应用修订包 89  
DB2 产品  
    除去  
        列示产品 54  
    手动安装  
        列示产品 54  
DB2 跟踪 66  
DB2 跟踪工具 (db2trc)  
    转储跟踪输出 67  
DB2 审计  
    故障诊断 7  
DB2 控制器  
    故障诊断 7

- DB2 信息中心
  - 版本 96
  - 更新 97
  - 以各种语言查看 97
- DB2 Connect
  - 故障诊断 39
- DB2 JDBC 驱动程序
  - 跟踪工具配置 79
- DB2 JDBC 2 类驱动程序
  - 跟踪工具配置 77
- DB2 Query Patroller
  - 故障诊断 7
- db2cli.ini 文件
  - 跟踪配置
  - CLI 和 ODBC 应用程序 80
- db2cos
  - 输出文件 15
- db2dart 命令 47
  - 故障诊断概述 46
- db2diag 命令
  - 示例 49
- db2diag.log 5, 9
  - 分析 11, 12, 49
  - 简介 11
- db2drdat 实用程序
  - 输出文件 69
- DB2FODC 注册表变量 3
- db2inspf 36
- db2level 命令
  - 用于故障诊断 51
- db2look 命令
  - 用于故障诊断 51
- db2pd 命令
  - 故障诊断示例 55
  - 缺省 db2cos 脚本收集的输出 15
- db2pdcfg -fodc 3
- db2support 命令
  - 故障诊断用法 20, 63
- db2trc (DB2 跟踪工具)
  - 格式化跟踪输出 68
  - 启动, 概述 66
- ddcstrc 实用程序
  - 输出文件 70
- diaglevel 配置参数
  - 更新 11
- DSS (分布式子节)
  - 类型, 跟踪 69

## E

- ECF 返回码
  - 描述 45
- EXCSAT 命令 70
- EXCSATRD 命令 70
- EXTNAM 对象 70

## F

- FFDC (首次故障数据捕获)
  - 陷阱文件 17
- FODC (首次出现数据捕获) 5

## I

- INSPECT 命令 47
- INSPECT CHECK 36

## J

- JDBC 跟踪 77
- JDBC 应用程序
  - 跟踪工具配置 77, 79

## O

- ODBC (开放式数据库连接)
  - 应用程序
  - 跟踪工具配置 80

## P

- PRDID 参数 70
- ps (进程状态) 实用程序 41, 70

## S

- SECCHK 命令 70
- SQL 语句
  - 显示帮助 96
  - SQL0965 错误代码 41
  - SQL0969 错误代码 41
  - SQL1338 错误代码 41
  - SQL30020 错误代码 41
  - SQL30060 错误代码 41
  - SQL30061 错误代码 41
  - SQL30073 错误代码 41
  - SQL30081N 错误代码 41
  - SQL30082 错误代码 41
  - SQL5043N 错误代码 41
- SQLCA (SQL 通信区)
  - 数据的缓冲区 69
  - SQLCODE 字段 69
- SQLCODE
  - SQLCA 中的字段 69
- SRVNAM 对象 70

## T

- TCP/IP
  - ACCSEC 命令 70

TCP/IP (续)  
SECCHK 命令 70  
Trace 实用程序 69

## U

UNIX  
除去  
列示产品 54

## V

Visual Explain  
教程 99

## Z

ZRC 返回码  
描述 45









中国印刷

G151-0621-00



Spine information:

**DB2 版本 9.5 Linux 版、UNIX 版和 Windows 版**

**故障诊断指南**

