



SQL Anywhere® 11 更改和升级

2009 年 2 月

11.0.1 版

版权和商标

版权所有 © 2009 iAnywhere Solutions, Inc. 部分版权所有 © 2009 Sybase, Inc. 保留所有权利。

本文档按原样提供，并不做任何形式的担保或承担任何责任（除非在您与 iAnywhere 达成的书面协议中另行规定）。

对本文档（全部或部分）的使用、打印、复制和分发须符合下列条件：1) 必须在整个或部分文档的所有副本中保留此声明和所有其它所有权声明，2) 不得修改本文档，3) 不得以任何形式表明您或 iAnywhere 之外的任何人是本文档的作者或提供者。

iAnywhere®、Sybase® 以及在 <http://www.sybase.com/detail?id=1011207> 上所列出商标均为 Sybase, Inc. 或其子公司的商标。® 表示在美国注册。

文中提及的所有其它公司和产品名可能是与其相关的各个公司的商标。

目录

关于本手册	vii
关于 SQL Anywhere 文档	viii
11.0.1 版中的新功能	1
SQL Anywhere 的新功能	2
SQL Anywhere 的行为更改和不建议使用的功能	8
MobiLink 的新功能	12
MobiLink 的行为更改和不建议使用的功能	13
QAnywhere 的新功能	14
UltraLite 的新功能	15
UltraLite 的行为更改和不建议使用的功能	16
UltraLiteJ 的新功能	17
UltraLiteJ 的行为更改和不建议使用的功能	18
管理工具新功能	19
管理工具的行为更改和不建议使用的功能	20
产品范围的新功能	21
文档增强	22
11.0.0 版中的新功能	23
SQL Anywhere	25
MobiLink	59
QAnywhere	64
SQL Remote	66
UltraLite	67
Sybase Central 和 Interactive SQL	73
文档增强	81
产品范围的功能	82
10.0.1 版中的新功能	85
SQL Anywhere	87

MobiLink	96
QAnywhere	98
SQL Remote	100
UltraLite	101
产品范围的功能	103
10.0.0 版中的新功能	107
SQL Anywhere	109
MobiLink	168
QAnywhere	187
SQL Remote	192
UltraLite	194
Sybase Central 和 Interactive SQL	206
文档增强	211
产品范围的功能	212
9.0.2 版中的新功能	215
9.0.2 版中的新功能	216
9.0.2 版中的行为更改	230
9.0.1 版中的新功能	237
9.0.1 版中的新功能	238
9.0.1 版中的行为更改	250
9.0.0 版中的新功能	253
9.0.0 版中的新功能	254
9.0 版中的行为更改	269
8.0.2 版中的新功能	277
8.0.2 版中的新功能	278
8.0.2 版的行为更改	286

8.0.1 版中的新功能	289
8.0.1 版中的新功能	290
8.0.1 版的行为更改	295
8.0.0 版中的新功能	297
版本 8 中的新功能	298
版本 8 中的行为更改	315
升级到 SQL Anywhere 11	323
升级 SQL Anywhere	324
升级 MobiLink	337
升级 QAnywhere	344
升级 UltraLite	345
升级 SQL Remote	353
索引	355

关于本手册

主题

本手册介绍 SQL Anywhere 11 以及该软件以前版本中的新功能。

有关 SQL Anywhere 版本 8 之前版本的新功能和行为更改的信息，请访问 http://www.iAnywhere.com/developer/product_manuals/sqlanywhere/1001/en/html/dbwnen10/dbwnen10.html。

读者

本手册面向的读者是该软件以前版本的用户，他们想要了解该软件的这一版本有什么新功能和不同之处。

关于 SQL Anywhere 文档

完整的 SQL Anywhere 文档以四种形式提供，但所包含信息均相同。

- **HTML 帮助** 联机帮助文档包含完整的 SQL Anywhere 文档，其中包括手册和 SQL Anywhere 工具的上下文相关帮助。

如果使用 Microsoft Windows 操作系统，则联机帮助文档以 HTML 帮助 (CHM) 格式提供。若要访问此文档，请选择 [开始] » [程序] » [SQL Anywhere 11] » [文档] » [联机手册]。

管理工具使用同一联机文档来实现帮助功能。

- **Eclipse** 在 Unix 平台上以 Eclipse 格式提供完整的联机帮助。要访问文档，请从 SQL Anywhere 11 安装的 *bin32* 或 *bin64* 目录下运行 *sadoc*。

- **DocCommentXchange** DocCommentXchange 是一个用于访问和讨论 SQL Anywhere 文档的社区。

使用 DocCommentXchange 可以执行以下任务：

- 查看文档
- 检查是否有用户对文档各部分所做出的阐明
- 提供建议和修正意见以在将来的版本中为所有用户改进文档

访问 <http://dcx.sybase.com>。

- **PDF** 整套 SQL Anywhere 手册会以一组 Portable Document Format (PDF) 文件的形式提供。您必须有 PDF 阅读器才能查看信息。要下载 Adobe Reader，请访问 <http://get.adobe.com/reader/>。

若要在 Microsoft Windows 操作系统上访问 PDF 文档，请选择 [开始] » [程序] » [SQL Anywhere 11] » 文档 » [联机手册 - PDF 格式]。

要在 Unix 操作系统上访问 PDF 文档，请使用 Web 浏览器打开 *install-dir/documentation/zh/pdf/index.html*。

关于文档集中的手册

SQL Anywhere 文档由以下手册组成：

- **SQL Anywhere 11 - 简介** 本手册介绍 SQL Anywhere 11，一个提供数据管理和数据交换技术的综合数据包，通过它可以为服务器环境、台式机环境、移动环境以及远程办公环境快速开发由数据库驱动的应用程序。
- **SQL Anywhere 11 - 更改和升级** 本手册介绍 SQL Anywhere 11 以及该软件以前版本中的新功能。
- **SQL Anywhere 服务器 - 数据库管理** 本手册介绍如何运行、管理及配置 SQL Anywhere 数据库。它介绍了数据库连接、数据库服务器、数据库文件、备份过程、安全性、高可用性、使用复制服务器进行复制以及管理实用程序和选项。

- **SQL Anywhere 服务器 - 编程** 本手册介绍如何使用 C、C++、Java、PHP、Perl、Python 和 .NET 编程语言（例如 Visual Basic 和 Visual C#）建立和部署数据库应用程序。其中介绍了各种编程接口，如 ADO.NET 和 ODBC。
- **SQL Anywhere 服务器 - SQL 参考** 本手册提供了系统过程和目录（系统表和视图）的参考信息。也介绍了 SQL 语言（搜索条件、语法、数据类型和函数）的 SQL Anywhere 实现。
- **SQL Anywhere 服务器 - SQL 的用法** 本手册介绍如何设计和创建数据库；如何导入、导出和修改数据；如何检索数据以及如何建立存储过程和触发器。
- **MobiLink - 入门** 本手册介绍基于会话的关系数据库同步系统 MobiLink。MobiLink 技术支持双向复制并且非常适用于移动计算环境。
- **MobiLink - 客户端管理** 本手册介绍如何设置、配置和同步 MobiLink 客户端。MobiLink 客户端可以是 SQL Anywhere 或者 UltraLite 数据库。本手册同时也介绍了 Dbmlsync API，通过它可以无缝地将同步集成到 C++ 或 .NET 客户端应用程序中。
- **MobiLink - 服务器管理** 本手册说明如何设置和管理 MobiLink 应用程序。
- **MobiLink - 服务器启动的同步** 本手册介绍 MobiLink 服务器启动的同步，这种功能允许 MobiLink 服务器启动同步或在远程设备上进行操作。
- **QAnywhere** 本手册介绍 QAnywhere，一个用于移动、无线、台式机和膝上型客户端的消息传递平台。
- **SQL Remote** 本手册介绍用于移动计算的 SQL Remote 数据复制系统，此系统支持使用电子邮件或文件传输等间接链接共享 SQL Anywhere 统一数据库和多个 SQL Anywhere 远程数据库之间的数据。
- **UltraLite - 数据库管理和参考** 本手册介绍适用于小型设备的 UltraLite 数据库系统。
- **UltraLite - C 及 C++ 编程** 本手册介绍 UltraLite C 和 C++ 编程接口。利用 UltraLite，可以开发数据库应用程序，并将它们部署到手持式设备、移动设备或嵌入式设备。
- **UltraLite - M-Business Anywhere 编程** 本手册介绍 UltraLite for M-Business Anywhere。利用 UltraLite for M-Business Anywhere，用户可以开发基于 Web 的数据库应用程序，并将它们部署到运行 Palm OS、Windows Mobile 或 Windows 的手持式设备、移动设备或嵌入式设备。
- **UltraLite - .NET 编程** 本手册介绍 UltraLite.NET。利用 UltraLite.NET，您可以开发数据库应用程序，并将它们部署到计算机、手持式设备、移动设备或嵌入式设备。
- **UltraLiteJ** 本手册介绍 UltraLiteJ。利用 UltraLiteJ，可以在支持 Java 的环境中开发和部署数据库应用程序。UltraLiteJ 支持 BlackBerry 智能手机和 Java SE 环境。UltraLiteJ 基于 iAnywhere UltraLite 数据库产品。
- **错误消息** 本手册提供了 SQL Anywhere 错误消息及其诊断信息的完整列表。

文档约定

本节列出了本文档中使用的约定。

操作系统

SQL Anywhere 可以在各种平台上运行。在大多数情况下，该软件在所有平台上的行为都是相同的，但也有变动或限制。这些变动或限制通常基于基础操作系统（Windows、Unix），很少基于特定变型（AIX、Windows Mobile）或版本。

为了简化对操作系统的提及，本文档按如下方式对支持的操作系统进行分组：

- **Windows** Microsoft Windows 系列包括 Windows Vista 和 Windows XP（主要用于服务器、台式计算机和膝上型计算机），以及 Windows Mobile（用于移动设备）。

除非另外指定，否则当本文档提及 Windows 时，是指所有基于 Windows 的平台，包括 Windows Mobile。

- **Unix** 除非另外指定，否则当本文档提及 Unix 时，是指所有基于 Unix 的平台，包括 Linux 和 Mac OS X。

目录和文件名

大部分情况下，对目录和文件名的引用在所有支持的平台上都是类似的，只需在不同形式之间进行简单的转换。这时需使用 Windows 约定。在细节更为复杂的情况下，文档显示所有相关形式。

下面是文档编写中用于简化目录和文件名的约定：

- **大写和小写目录名** 在 Windows 和 Unix 上，目录和文件名可以包括大写和小写字母。创建目录和文件时，文件系统会保留字母大小写。

在 Windows 上，对目录和文件的提及不区分大小写。混合使用大小写的目录和文件名很常见，但使用所有小写字母来提及目录和文件的形式也很常见。SQL Anywhere 安装包包含诸如 *Bin32* 和 *Documentation* 的目录。

在 Unix 上，对目录和文件的提及区分大小写。混合使用大小写的目录和文件名不常见。大多数的目录和文件名全部使用小写字母。SQL Anywhere 安装包包含诸如 *bin32* 和 *documentation* 的目录。

本文档采用 Windows 形式的目录名。大多数情况下，在 Unix 上可以将大小写混合形式的目录名转换成小写字母的等效目录名。

- **分隔目录和文件名的斜线** 文档使用反斜线作为目录分隔符。例如，PDF 格式的文档位于 *install-dir\Documentation\zh\PDF*（Windows 形式）。

在 Unix 上，用正斜线替换反斜线。PDF 文档位于 *install-dir/documentation/zh/pdf* 下。

- **可执行文件** 文档使用 Windows 约定显示可执行文件名（带有诸如 *.exe* 或 *.bat* 后缀）。在 Unix 上，可执行文件名没有后缀。

例如，在 Windows 上，网络数据库服务器是 *dbsrv11.exe*。在 Unix 上是 *dbsrv11*。

- **install-dir** 在安装过程中，选择 SQL Anywhere 的安装位置。创建环境变量 *SQLANY11*，用来表示此位置。文档中以 *install-dir* 表示此位置。

例如，本文档将此文件表示为 *install-dir\readme.txt*。在 Windows 上，这等同于 *%SQLANY11%\readme.txt*。在 Unix 上，这等同于 *\$SQLANY11/readme.txt* 或 *\${SQLANY11}/readme.txt*。

有关 *install-dir* 缺省位置的详细信息，请参见“SQLANY11 环境变量”一节《SQL Anywhere 服务器 - 数据库管理》。

- **samples-dir** 在安装过程中，选择 SQL Anywhere 随附的示例的安装位置。创建环境变量 SQLANY11，用来表示此位置。文档中以 *samples-dir* 表示此位置。

要在 *samples-dir* 中打开 Windows 资源管理器窗口，请在 [开始] 菜单中，选择 [程序] » [SQL Anywhere 11] » [示例应用程序和项目]。

有关 *samples-dir* 缺省位置的详细信息，请参见“SQLANY11 环境变量”一节《SQL Anywhere 服务器 - 数据库管理》。

命令提示符和命令 shell 语法

大多数操作系统都提供一种或多种使用命令 shell 或命令提示符来输入命令和参数的方法。Windows 命令提示符包括 Command Prompt (DOS 提示符) 和 4NT。Unix 命令 shell 包括 Korn shell 和 bash。每个 shell 都具有一些功能，其能力不仅仅局限于简单命令。这些功能通过特殊字符来驱动。特殊字符和功能随 shell 的不同而不同。如果没有正确使用这些特殊字符，通常会导致语法错误或意外行为。

本文档以普通形式提供命令行示例。如果这些示例中包含 shell 的特殊字符，则命令需要根据特定 shell 进行修改。修改方法不在本文档所述范围之内，但通常是在包含这些特殊字符的参数两旁加上引号，或是在特殊字符前面使用转义字符。

下面是命令行语法的一些示例，不同的平台可能会有不同的形式：

- **括号和大括号** 有些命令行选项需要一个参数，该参数将以列表形式接受详细的值指定。该列表通常用括号或大括号括起来。本文档使用括号。例如：

```
-x tcpip(host=127.0.0.1)
```

如果括号导致出现语法问题，用大括号替代：

```
-x tcpip{host=127.0.0.1}
```

如果两种形式都将产生语法问题，应按照 shell 的要求，用引号将整个参数括起来：

```
-x "tcpip(host=127.0.0.1)"
```

- **引号** 如果必须在参数值中指定引号，该引号可能会与用于括参数的引号的传统用法发生冲突。例如，要指定值中包含双引号的加密密钥，则可能必须用引号括起密钥，然后转义嵌入的引号：

```
-ek "my \"secret\" key"
```

在许多 shell 中，密钥的值为 my "secret" key。

- **环境变量** 本文档介绍设置环境变量。在 Windows shell 中，环境变量使用语法 %ENVVAR% 来指定。在 Unix shell 中，环境变量使用语法 \$ENVVAR 或 \${ENVVAR} 来指定。

图标

本文档中使用了下列图标。

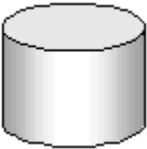
- 客户端应用程序。



- 数据库服务器，如 Sybase SQL Anywhere。



- 数据库。在某些高水平的图中，可以使用此图标表示数据库和管理该数据库的数据库服务器。



- 复制或同步中间件。用于帮助在数据库之间共享数据。例如 MobiLink 服务器和 SQL Remote 消息代理。



- 编程接口。



联系文档小组

我们欢迎您就本帮助文档提出意见、建议和反馈信息。

要提交意见和建议，请发送电子邮件到 SQL Anywhere 文档小组，地址为 iasdoc@sybase.com。虽然我们不对这些电子邮件进行回复，但您的反馈会帮助我们改进文档，因此我们真诚地欢迎您提出宝贵的意见和建议。

DocCommentXchange

也可以使用 DocCommentXchange 将意见或建议直接置于帮助主题中。DocCommentXchange (DCX) 是一个用于访问和讨论 SQL Anywhere 文档的社区。使用 DocCommentXchange 可以执行以下任务：

- 查看文档
- 检查是否有用户对文档各部分所做出的阐明
- 提供建议和修正意见以在将来的版本中为所有用户改进文档

访问 <http://dcx.sybase.com>。

查找详细信息并请求技术支持

附加信息和资源可从 Sybase iAnywhere 开发人员社区获得，网址是 <http://www.sybase.com/developer/library/sql-anywhere-techcorner>。

如果您有问题或是需要帮助，可将邮件发布到下面所列的 Sybase iAnywhere 新闻组。

当您向这些新闻组发布邮件时，请务必提供问题的详细信息，包括 SQL Anywhere 版本的内部版本号。可以通过运行以下命令找到此信息：**dbeng11 -v**。 **dbeng11 -v**。

新闻组位于 *forums.sybase.com* 新闻服务器上。

这些新闻组包括：

- [sybase.public.sqlanywhere.general](#)
- [sybase.public.sqlanywhere.linux](#)
- [sybase.public.sqlanywhere.mobilink](#)
- [sybase.public.sqlanywhere.product_futures_discussion](#)
- [sybase.public.sqlanywhere.replication](#)
- [sybase.public.sqlanywhere.ultralite](#)
- [ianywhere.public.sqlanywhere.qanywhere](#)

有关 Web 开发问题，请访问 <http://groups.google.com/group/sql-anywhere-web-development>。

新闻组免责声明

iAnywhere Solutions 没有义务为其新闻组提供解决方案、信息或建议，除提供系统操作员监控服务和确保新闻组的运行和可用性外，iAnywhere Solutions 也没有义务提供任何其它服务。

如果时间允许，iAnywhere 技术顾问以及其他员工也会对新闻组服务提供帮助。他们是在自愿的基础上提供帮助的，所以可能无法定期提供解决方案和信息。他们可以提供多少帮助取决于他们的工作量。

11.0.1 版中的新功能

目录

SQL Anywhere 的新功能	2
SQL Anywhere 的行为更改和不建议使用的功能	8
MobiLink 的新功能	12
MobiLink 的行为更改和不建议使用的功能	13
QAnywhere 的新功能	14
UltraLite 的新功能	15
UltraLite 的行为更改和不建议使用的功能	16
UltraLiteJ 的新功能	17
UltraLiteJ 的行为更改和不建议使用的功能	18
管理工具新功能	19
管理工具的行为更改和不建议使用的功能	20
产品范围的新功能	21
文档增强	22

不建议使用的功能的列表可能会发生更改

与所有前瞻性声明一样，不建议使用的功能的列表不保证其完整性，并可能会发生更改。

SQL Anywhere 的新功能

以下列出了 11.0.1 版中在 SQL Anywhere 数据库和数据库服务器方面新增的功能。

连接属性

在 SQL Anywhere 11.0.1 版中新增了以下连接属性：

- Authenticated
- IsDebugger
- QueryBypassedCosted
- QueryBypassedHeuristic
- QueryBypassedOptimized
- QueryOpened
- QueryDescribedBypass
- QueryDescribedOptimizer
- StatementDescribes
- StatementPostAnnotates
- StatementPostAnnotatesSimple
- StatementPostAnnotatesSkipped
- WaitStartTime
- WaitType

有关这些属性的说明，请参见“[连接属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

数据库属性

在 SQL Anywhere 11.0.1 版中新增了以下数据库属性：

- Authenticated
- Prepares
- QueryBypassedCosted
- QueryBypassedHeuristic
- QueryBypassedOptimized
- QueryOpened
- QueryDescribedBypass
- QueryDescribedOptimizer
- StatementDescribes
- StatementPostAnnotates
- StatementPostAnnotatesSimple
- StatementPostAnnotatesSkipped

有关这些属性的说明，请参见“[数据库属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

数据库服务器属性

以下列出了 SQL Anywhere 11.0.1 版中对数据库服务器属性的增强。

- **ServerEdition**

有关这些属性的说明，请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库实用程序

以下列出了 SQL Anywhere 11.0.1 版中对数据库实用程序的增强。

- **服务实用程序 (dbsvc) 增强** 现在在 Windows 上，可以使用服务实用程序为 MobiLink 中继服务器 (rshost)、中继服务器出站启动器 (Relay Server Outbound Enabler, 简称 RSOE) 和卷影复制服务 (dbvss11) 创建服务。请参见“用于 Windows 的服务实用程序 (dbsvc)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **卸载实用程序 (dbunload) 增强** dbunload 实用程序现在支持以下选项：
 - **-cm 选项** 针对所卸载的数据库显示 dbinit 命令或 CREATE DATABASE 语句。
 - **-l 选项** 为重建的数据库中的自动增量列保留下一个可用值。请参见“卸载实用程序 (dbunload)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **升级实用程序 (dbupgrad) 增强** 尝试使用升级实用程序升级正在参与数据库镜像的数据库现在将会导致错误。请参见“升级实用程序 (dbupgrad)”一节《SQL Anywhere 服务器 - 数据库管理》。

系统过程和函数

以下列出了在 SQL Anywhere 11.0.1 版中增加的对系统过程和函数的增强。

- **sa_get_table_definition 系统过程** 新增的 sa_get_table_definition 系统过程将返回一个 LONG VARCHAR 字符串，该字符串包含创建指定表及其索引、外键、触发器和授予权限所必需的 SQL 语句。请参见“sa_get_table_definition 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **FIRST_VALUE 函数 [Aggregate]** FIRST_VALUE 函数 [Aggregate] 现在包括一个 RESPECT NULLS 子句。请参见“FIRST_VALUE 函数 [Aggregate]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **LAST_VALUE 函数 [Aggregate]** LAST_VALUE 函数 [Aggregate] 现在包括一个 RESPECT NULLS 子句。请参见“LAST_VALUE 函数 [Aggregate]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_set_http_option 系统过程** 现在通过 AcceptCharset 选项可以对字符集选择进行更多的控制。请参见“sa_set_http_option 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

SQL 语句

以下列出了在 SQL Anywhere 11.0.1 版中引入的 SQL 增强。

- **新增的 DEFAULT VALUES 子句, INSERT 语句** INSERT 语句中新增的 DEFAULT VALUES 子句用于插入为所有列定义的缺省值。请参见“[INSERT 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **CREATE ENCRYPTED DATABASE 语句** 此语句会为现有数据库创建一份加密副本，包括所有的事务日志、镜像日志和 dbspace。还可使用此语句为启用了表加密的数据库创建一份副本。请参见“[CREATE ENCRYPTED DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

如果想对需要恢复的数据库进行加密（例如，要发送给技术支持部门），仍然必须使用 CREATE ENCRYPTED FILE 语句。请参见“[CREATE ENCRYPTED FILE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **CREATE DECRYPTED DATABASE 语句** 此语句会为现有数据库创建一份解密副本，包括所有的事务日志、镜像日志和 dbspace。请参见“[CREATE DECRYPTED DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

如果想对需要恢复的数据库进行解密（例如，要发送给技术支持部门），仍然必须使用 CREATE DECRYPTED FILE 语句。请参见“[CREATE DECRYPTED FILE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **ALTER DATABASE 语句增强** 尝试对当前正在镜像的数据库服务器执行 ALTER DATABASE UPGRADE 语句现在将会导致错误。请参见“[ALTER DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **创建或替换同名的函数、过程、触发器或视图** 新增的 OR REPLACE 子句用于创建或替换同名的函数、过程、触发器或视图。请参见：
 - “[CREATE FUNCTION 语句 \(Web 服务\)](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[CREATE PROCEDURE 语句 \(Web 服务\)](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[CREATE TRIGGER 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[CREATE VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
- **当某语句试图删除不存在的数据库对象时不发送错误消息** 新增的 IF EXISTS 子句用于指定在 DROP 语句试图删除不存在的数据库对象时您不希望返回错误。请参见：
 - “[DROP EVENT 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[DROP FUNCTION 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[DROP MATERIALIZED VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[DROP PROCEDURE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[DROP TABLE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[DROP TRIGGER 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[DROP VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
- **新增的 INTO LOCAL TEMPORARY TABLE 子句, SELECT 语句** SELECT 语句中新增的 INTO LOCAL TEMPORARY TABLE 子句用于使用 SELECT 语句的结果集创建和填充局部临时表。之前，使用 INTO 子句只能对以 # 开头的临时表完成此项任务。请参见“[SELECT 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **新增的 IF NOT EXISTS 子句, CREATE TABLE 语句** CREATE TABLE 语句中新增的 IF NOT EXISTS 子句用于在表尚未建立的情况下创建永久表、全局临时表和局部临时表。请参见“[CREATE TABLE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **在创建临时过程或函数时指定所有者** 现在通过 CREATE FUNCTION 和 CREATE PROCEDURE 语句可以指定所有者（非必需项）。请参见：
 - “[CREATE FUNCTION 语句（Web 服务）](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[CREATE PROCEDURE 语句（Web 服务）](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》

编程接口

以下列出了 SQL Anywhere 11.0.1 版中对编程接口的增强。

- **新增的 ASP.NET 提供程序** 以下新增的 ASP.NET 提供程序模拟标准 ASP.NET 提供程序的功能，但会将它们的数据存储在 SQL Anywhere 数据库中，而不是存储在 SQL Server 数据库中：
 - **成员资格** 用于登录和注销并管理用户及其密码
 - **角色** 用于将用户指派给各组，从而实现简单容易的权限管理
 - **配置文件** 存储用户变量
 - **Web 部件个性化** 管理 "Web 部件" 数据的存储，使用户可以将其视图个性化
 - **Web 事件** 与 "健康监视" 配合使用，将刷新的 Web 事件信息存储在数据库中以便进行日志记录。

请参见“[SQL Anywhere ASP.NET 提供程序](#)”《[SQL Anywhere 服务器 - 编程](#)》。

- **对 Ruby 的新支持** SQL Anywhere 现在支持 Ruby 开放源编程语言。请参见“[SQL Anywhere for Ruby](#)”《[SQL Anywhere 服务器 - 编程](#)》。
- **OLE DB 现在支持 CATALOGS 和 SCHEMATA 行集** 现在支持 OLE DB 的 CATALOGS 和 SCHEMATA 行集。由于 SQL Anywhere 不支持目录的概念，因此 SQL Anywhere OLE DB 提供程序将返回一个包含所有当前启动的数据库（及其位置）的 CATALOGS 结果集。同样，对于 SCHEMATA，数据库名称将被用作结果集中的目录。

杂类

以下列出了在 SQL Anywhere 11.0.1 版中引入的其它增强。

- **由于语句执行简单且开销较低而提高性能** 对于花费较少执行时间的语句，为其生成执行计划的开销有时要高于执行该语句本身的开销。凭借以下增强功能，SQL Anywhere 现在能够花费较少执行时间识别更多的简单语句，并使它们跳过优化程序。

有关详细信息，请参见“[跳过查询处理阶段的资格](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

- **新增的 FORCE NO OPTIMIZATION 子句** 以前，所有需要以开销为基础进行优化的语句都是由优化程序进行处理的。现在，通过使用新增的 FORCE NO OPTIMIZATION 子句，可以

指定希望语句跳过优化程序。如果语句过于复杂以至于无法以这种方式进行处理（可能是由于数据库选项的设置，或由于模式或查询的特性而导致），此语句会失败并且数据库服务器将返回错误。

以下语句支持新增的 FORCE NO OPTIMIZATION 子句：

- “DELETE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “INSERT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “SELECT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “UPDATE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》

○ 新连接属性

- QueryBypassedCosted
- QueryBypassedHeuristic
- QueryBypassedOptimized
- QueryOpened
- QueryDescribedBypass
- QueryDescribedOptimizer
- StatementDescribes
- StatementPostAnnotates
- StatementPostAnnotatesSimple
- StatementPostAnnotatesSkipped

○ 新数据库属性

- Prepares
- QueryBypassedCosted
- QueryBypassedHeuristic
- QueryBypassedOptimized
- QueryOpened
- QueryDescribedBypass
- QueryDescribedOptimizer
- StatementDescribes
- StatementPostAnnotates
- StatementPostAnnotatesSimple
- StatementPostAnnotatesSkipped

- **新增 [优化方法] 字段** [优化方法] 字段已经被添加到图形式计划的 [优化程序统计] 部分。该字段返回由查询优化程序所选择的执行策略。请参见“[优化程序统计字段说明](#)”一节 《SQL Anywhere 服务器 - SQL 的用法》。

- **防止数据库服务器成为缺省数据库服务器** -xd 服务器选项可防止数据库服务器监听缺省 TCP 端口，并且防止数据库服务器成为缺省数据库服务器。请参见“[-x 服务器选项](#)”一节 《SQL Anywhere 服务器 - 数据库管理》。
- **支持并行档案备份** SQL Anywhere 数据库服务器现在支持服务器端档案备份的并行备份。并行数据库备份利用物理 I/O 来并行（而不是按顺序）执行读写操作，从而提高了性能。

BACKUP DATABASE 语句中增加了两个支持并行档案备份的新子句：

- **WITH CHECKPOINT LOG [NO] COPY**
- **MAX WRITE { *n* | AUTO }**

11.0.0 版及更早版本的数据库服务器不能恢复通过 11.0.1 版数据库服务器生成的档案备份。11.0.1 版数据库服务器能够恢复由更早版本数据库服务器生成的备份。

请参见“BACKUP 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **在 Mac OS X 上运行开发人员版和评测版数据库服务器** 现在可以从 Mac OS X 上的管理工具自动启动开发人员版和评测版数据库服务器。

SQL Anywhere 的行为更改和不建议使用的功能

以下列出了 11.0.1 版中引入的 SQL Anywhere 数据库和数据库服务器更改（根据类别分组）。

行为更改

- **全文搜索** 对全文搜索做出了以下行为更改：
 - **现在应用运算符优先级** 以前，在查询字符串中不对运算符应用优先级。现在，可以应用以下运算符优先级：
 - NEAR、FUZZY 运算符
 - AND NOT 运算符
 - AND 运算符
 - OR 运算符

请参见“CONTAINS 搜索条件中的运算符优先级”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **NEAR 子句参数必须是术语或前缀术语** 在执行邻近搜索时，NEAR 子句的参数必须是术语或前缀术语。请参见“CONTAINS 搜索条件”一节《SQL Anywhere 服务器 - SQL 参考》和“邻近搜索”一节《SQL Anywhere 服务器 - SQL 的用法》。
 - **连字符的使用和 AND NOT 子句** 在短语内，连字符被视为术语中断符，而不是特殊字符。在短语之外，连字符的处理取决于连字符附近的语法。请参见“对于连字符 (-) 允许的语法”一节《SQL Anywhere 服务器 - SQL 参考》和“在全文搜索中使用 AND NOT 运算符”一节《SQL Anywhere 服务器 - SQL 的用法》。
 - **星号的使用和前缀搜索** 在执行前缀搜索时，必须将星号附加于术语，并且后面紧跟一个空格、查询字符串结尾或允许的特殊字符之一。请参见“对于星号 (*) 允许的语法”一节《SQL Anywhere 服务器 - SQL 参考》和“前缀搜索”一节《SQL Anywhere 服务器 - SQL 的用法》。
 - **创建重复文本索引现在将返回错误** 不再可以创建重复文本索引。如果以下设置与某现有文本索引的设置相同，则将文本索引视为重复项：
 - 所引用的基本表
 - 要建立索引的列（顺序不重要）
 - 所使用的配置对象的设置（TERM BREAKER、MINIMUM TERM LENGTH、MAXIMUM TERM LENGTH、STOPLIST、归类信息）

在 SQL Anywhere 11.0.1 之前的版本中创建的重复文本索引可以保留在数据库中，并且在通过 11.0.1 数据库服务器启动时不会导致错误。但是，如果将包含重复文本索引的数据库重新加载到 11.0.1 版或更高版本上，则会返回错误。

要识别现有数据库中的重复文本索引，请执行以下查询：

```
SELECT LIST( i.index_name )
FROM SYS.SYSIDX i
      JOIN SYS.SYSTEXTIDX t ON i.object_id = t.index_id AND t.sequence =
1
      JOIN SYS.SYSTEXTCONFIG F ON F.object_id = t.text_config
```

```

JOIN (
  SELECT table_id, index_id, LIST( column_id, ', ' ORDER BY
  column_id ) col_id
  FROM SYS.SYSDIXCOL
  GROUP BY table_id, index_id) x
ON x.table_id = i.table_id AND x.index_id = i.index_id
WHERE i.index_category=4
GROUP BY i.table_id, f.term_breaker, f.min_term_length,
f.max_term_length,
f.collation, ISNULL( f.char_stoplist, '-' ),
ISNULL( f.nchar_stoplist, '-' ), x.col_id
HAVING count(*) > 1

```

只有代表 STOPLIST 的字符串完全相同或者已指定 NO STOPLIST 时该查询才有效。例如，使用此查询时，不会将非索引字表 'a b c' 和 'a - b c' 视为是相同的非索引字表，但在文本索引创建期间的重复项检查过程中，它们会被视为相同。

- **正则表达式** 已经对 SIMILAR TO 和 REGEXP 搜索条件以及 REGEXP_SUBSTR 函数的行为进行了更改。更改的总体目的是继续使 SIMILAR TO 符合 ANSI/SQL 标准，同时使 REGEXP 和 REGEXP_SUBSTR 的行为符合 Perl。

- **数据库归类 and 匹配** 以前，REGEXP 和 REGEXP_SUBSTR 通过使用归类等同性和排序顺序来确定模式中的文字或字符类范围是否与字符串匹配。现在，REGEXP 和 REGEXP_SUBSTR 使用代码点值的二进制比较进行匹配和范围评估。进行这一更改是为了使行为符合 Perl 5.0。

SIMILAR TO 仍然使用数据库归类进行匹配和范围评估。请参见“LIKE、REGEXP 和 SIMILAR TO 搜索条件”一节《SQL Anywhere 服务器 - SQL 参考》。

- **数据库区分大小写以及 [[:upper:]] 和 [[:lower:]] 子字符类** SIMILAR TO 及 REGEXP [[:upper:]] 和 [[:lower:]] 子字符类在不区分大小写的数据库中是不区分大小写的。这在 11.0.1 中发生了变化，以便 [[:upper:]] 仅匹配大写字符，而 [[:lower:]] 仅匹配小写字符（无论数据库是否区分大小写）。
- **将脱字符 (^)、下划线 (_) 和百分号 (%) 作为元字符的处理方式** 下表说明了以前和现在将这些字符作为元字符的处理方式：

字符	以前的行为	最新的行为
_ (下划线)	对于 SIMILAR TO、REGEXP 和 REGEXP_SUBSTR，下划线被视为元字符：它与任意单字符匹配。	对于 SIMILAR TO，下划线被视为元字符：它与任意单字符匹配。 对于 REGEXP 和 REGEXP_SUBSTR，下划线不被视为元字符。 REGEXP 和 REGEXP_SUBSTR 使用句点 (.) 匹配任意单字符。

字符	以前的行为	最新的行为
%	对于 SIMILAR TO、REGEXP 和 REGEXP_SUBSTR，百分号被视为元字符：它与任意数量的任何字符匹配。	对于 SIMILAR TO，百分号被视为元字符：它与任意数量的任何字符匹配。 对于 REGEXP 和 REGEXP_SUBSTR，百分号不被视为元字符。 REGEXP 和 REGEXP_SUBSTR 使用点星号 (.) 匹配任意数量的任何字符。
^	对于 SIMILAR TO、REGEXP 和 REGEXP_SUBSTR，字符类内部的脱字符被视为它右侧任何字符的取非或相减字符：它被解释为不匹配。	对于 SIMILAR TO，脱字符被视为它右侧字符的取非或相减字符。例如，SIMILAR TO [a-d^c] 与 a、b 和 d 匹配，但不与 c 匹配。 对于 REGEXP 和 REGEXP_SUBSTR，只有当脱字符位于字符类中的第一个位置时，才会将它视为元字符：它被解释为字符类的取非。例如，REGEXP [^abc] 与 a、b 或 c 以外的任何单字符匹配，而 REGEXP [a-d^c] 与 a、b、c、d 和 ^ 匹配。

- **Mac OS X 不再需要 dbmodenv** 在以前的版本中，要使用 Mac OS X 上的图形管理工具，用户的 `$HOME/MacOSX/environment.plist` 文件必须将 SQL Anywhere 二进制内容和库位置添加到 PATH 和 DYLD_LIBRARY_PATH。可以使用 dbmodenv 工具来完成此操作。SQL Anywhere 不再依赖 `$HOME/MacOSX/environment.plist` 中的设置，您也不再需要在安装 SQL Anywhere 之后运行 dbmodenv 或注销并再次登录。
- **更改了从 dbisqlc OUTPUT 语句返回的空值的缺省值** 在以前的版本中，如果使用来自 dbisqlc 的 OUTPUT 语句，该语句将为空值返回值 [(NULL)]。现在，在缺省情况下，该语句将为空值返回空字符串。您可以通过设置 output_nulls 选项来改变 NULL 值的导出方式。请参见“output_nulls 选项 [Interactive SQL]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Endian 支持** 升级后，在 big-endian 计算机上创建的 11.0.1 版之前的文本索引需要截断和刷新（对于 MANUAL REFRESH 和 AUTO REFRESH 文本索引）或重新创建（对于 IMMEDIATE REFRESH 索引）。

- **Mac OS X 上的管理工具** 现在在 Mac OS X 上，SQL Anywhere 管理工具使用 64 位 JDK 1.6。管理工具只能在使用 Apple JDK 1.6 所支持的 64 位处理器的 Intel Macintosh 机（Mac OS X 10.5.2 或更高）上运行。如果要为 Mac OS X 部署管理工具，本地库位于 `$$SQLANY11/System/lib64` 下。请参见“在 Linux、Solaris 和 Mac OS X 上部署管理工具”一节《SQL Anywhere 服务器 - 编程》。
- **HTTP 客户端分块传输编码的新缺省大小** 先前，如果 HTTP client 发送超过 2048 字节的数据，缺省情况下（或者如果用户指定 `CREATE PROCEDURE ... SET 'HTTP(CH=auto)'` 时）会尝试分块传输编码。缺省大小已从 2048 字节更改为 8196 字节。此外，还新增了一个状态 411 **需要长度**，此状态在需重新发送未使用分块传输编码的请求时使用。请参见“**CREATE PROCEDURE 语句（Web 服务）**”一节《SQL Anywhere 服务器 - SQL 参考》。

不建议使用和停用的功能

- **COMMENT ON EXTERNAL ENVIRONMENT OBJECT *object-name*** 语法已更改为 **COMMENT ON EXTERNAL OBJECT *object-name***。目前仍接受旧语法，但在未来版本中可能不再支持该语法。请参见“**COMMENT 语句**”一节《SQL Anywhere 服务器 - SQL 参考》。
- **支持 ansi_substring 选项** ansi_substring 选项 11.0.0 版不建议使用，但是现在在 11.0.1 版中将受支持。请参见“**ansi_substring 选项 [兼容性]**”一节《SQL Anywhere 服务器 - 数据库管理》。

MobiLink 的新功能

以下列出了 11.0.1 版中引入的 MobiLink 新增功能。

- **mlsrv11 的 -vi 选项** 显示上载的每行的列值。请参见“[-v 选项](#)”一节《[MobiLink - 服务器管理](#)》。
- **mlsrv11 的 -vq 选项** 显示下载的每行的列值。请参见“[-v 选项](#)”一节《[MobiLink - 服务器管理](#)》。
- **mlsrv11 的 -vm 选项** 每当同步完成后将每个同步的持续时间以及每个同步阶段的持续时间打印到日志中。请参见“[-v 选项](#)”一节《[MobiLink - 服务器管理](#)》。
- **mlsrv11 的 -ppv 选项** 使 MobiLink 根据指定的时间段打印新的定期监控值。请参见“[-ppv 选项](#)”一节《[MobiLink - 服务器管理](#)》。
- **ml_ignore 前缀** MobiLink 服务器将以 --{ml_ignore} 为前缀的 SQL 脚本视为有意忽略的脚本。请参见“[忽略脚本](#)”一节《[MobiLink - 服务器管理](#)》。
- **dblsn 的 -sv 选项** 指定由 MobiLink 监听器用来对数据库进行验证的脚本版本。请参见“[-sv 选项](#)”一节《[MobiLink - 服务器启动的同步](#)》。
- **对 Oracle VArray 的支持** iAnywhere Solutions 11 - Oracle ODBC 驱动程序现在支持在存储过程中使用 Oracle VArray。请参见“[使用 Oracle varray](#)”一节《[MobiLink - 服务器管理](#)》。
- **轻量级轮询监听器关键字变量** 添加了 poll_connect、poll_notifier、poll_key 和 poll_every 监听器关键字以支持轻量级轮询。
- **轻量级轮询操作变量** 添加了 \$poll_connect、\$poll_notifier、\$poll_key 和 \$poll_every 操作变量以支持轻量级轮询。
- **使用通用访问卡的客户端验证** MobiLink 客户端现在支持使用通用访问卡 (CAC) 提供客户端身份进行验证。请参见“[identity_name](#)”一节《[MobiLink - 客户端管理](#)》。

需要单独授予许可的组成部分

此功能是 CAC 验证插件的一部分，需要单独的许可。请参见“[单独授权的组件](#)”一节《[SQL Anywhere 11 - 简介](#)》。

- **对 Microsoft SQL Server 的支持** MobiLink 同步服务器现在支持在 Microsoft SQL Server 2008 上运行的统一数据库。有关对应新 Microsoft DATE、TIME 和 DATETIME2 数据类型的信息，请参见“[Microsoft SQL Server 数据映射](#)”一节《[MobiLink - 服务器管理](#)》。
- **.NET DownloadTableData 接口的新方法** 返回表的上次下载时间的 getLastDownloadTime 方法。请参见“[GetLastDownloadTime 方法](#)”一节《[MobiLink - 服务器管理](#)》。
- **目标 MobiLink 用户和远程 ID 的日志详细程度** 现在可以为目标 MobiLink 用户或远程 ID 设置不同的日志详细程度。请参见“[目标 MobiLink 用户和远程 ID 的日志详细程度](#)”一节《[MobiLink - 服务器管理](#)》。

MobiLink 的行为更改和不建议使用的功能

以下列出了 11.0.1 版中引入的 MobiLink 更改。

- **mlsrv11 的 -sm 选项** 经过改进后，在结合非持久性 HTTP/HTTPS 使用时，mlsrv11 的 -sm 选项现已提供类似 -nc 选项的功能。请参见“-sm 选项”一节《MobiLink - 服务器管理》和“-nc 选项”一节《MobiLink - 服务器管理》。
- **Microsoft SQL Server 数据类型** 从 SQL Server 2005 开始，已不建议使用 TEXT、NTEXT 和 IMAGE 类型。而是使用 VARCHAR(max)、NVARCHAR(max) 和 VARBINARY(max)。请参见“Microsoft SQL Server 数据映射”一节《MobiLink - 服务器管理》。

QAnywhere 的新功能

以下列出了 11.0.1 版中引入的 QAnywhere 新增功能。

- **QAnywhere 独立客户端** QAnywhere 独立客户端提供了一个精简的客户端，该客户端允许您设置消息传递系统而不必为运行 QAnywhere 代理或管理数据库担忧。请参见“[QAnywhere 独立客户端简介](#)”一节《QAnywhere》。

UltraLite 的新功能

以下列出了 11.0.1 版中引入的 UltraLite 新增功能。

- **镜像文件** UltraLite 提供基本的数据库文件镜像功能，来改进对潜在不可靠存储系统的容错能力。这一效果通过使用镜像文件来实现。所有数据库写入操作在发送到主数据库文件同时也发送到镜像文件。请参见“[UltraLite MIRROR_FILE 连接参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **Unset ml_remote_id** 可以使用 SET OPTION 取消设置 ml_remote_id。请参见“[UltraLite SET OPTION 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》和“[ML_GET_SERVER_NOTIFICATION \[系统\]](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **ML_GET_SERVER_NOTIFICATION** 此函数允许 Ultralite 用户使用轻量级轮询，来查询 MobiLink 服务器上服务器启动的同步请求的通告程序。请参见“[ML_GET_SERVER_NOTIFICATION \[系统\]](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **SYNC_PROFILE_OPTION_VALUE 函数 [系统]** 此函数返回在同步配置文件中的指定选项的值。请参见“[SYNC_PROFILE_OPTION_VALUE 函数 \[系统\]](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite .NET API 的 StreamErrorParameters 属性** StreamErrorParameters 已添加到 SyncResult 类。此成员包含 StreamErrorCode 中报告的流错误代码对应的流错误参数列表（以逗号分隔）。请参见“[StreamErrorParameters 属性](#)”一节《[UltraLite - .NET 编程](#)》。
- **UltraLite for M-Business API 的 getStreamErrorParameters 方法** 此方法返回由同步流处理报告的错误参数列表（以逗号分隔）。请参见“[getStreamErrorParameters 方法](#)”一节《[UltraLite - M-Business Anywhere 编程](#)》。

UltraLite 的行为更改和不建议使用的功能

以下列出了 11.0.1 版中引入的 UltraLite 更改。

● **Palm 挂起功能** 请参见：

- “在 UltraLite Palm 应用程序中维护状态（不建议使用）”一节 《UltraLite - C 及 C++ 编程》
- “在 UltraLite Palm 应用程序中恢复状态（不建议使用）”一节 《UltraLite - C 及 C++ 编程》
- “GetSuspend 函数（不建议使用）”一节 《UltraLite - C 及 C++ 编程》
- “SetSuspend 函数（不建议使用）”一节 《UltraLite - C 及 C++ 编程》
- “GetSuspend 函数（不建议使用）”一节 《UltraLite - C 及 C++ 编程》
- “SetSuspend 函数（不建议使用）”一节 《UltraLite - C 及 C++ 编程》
- “SQLSetSuspend 函数（不建议使用）”一节 《UltraLite - C 及 C++ 编程》

UltraLiteJ 的新功能

以下列出了 11.0.1 版中引入的 UltraLiteJ 新增功能。

● **其它 SQL 支持** UltraLiteJ 提供的更多 SQL 支持如下：

- ALTER TABLE
- CREATE TABLE
- CREATE INDEX
- DROP INDEX
- DROP TABLE
- TRUNCATE TABLE

此外，以下限制已解除：

- 已支持 HAVING。
- 已支持集合函数中的 DISTINCT
- 已支持 CURRENT TIME、CURRENT TIMESTAMP 和 CURRENT DATE。

请参见“支持的 SQL 语句”一节《UltraLiteJ》。

- **新 DatabaseInfo 方法** DatabaseInfo 接口新增了两个新方法，`getPageReads()` 和 `getPageWrites`。这两个方法返回创建 DatabaseInfo 对象时读取和写入的页数。请参见“`getPageReads` 方法”一节《UltraLiteJ》和“`getPageWrites` 方法”一节《UltraLiteJ》。
- **更新的 UltraLiteJ 数据库传输实用程序** UltraLiteJ 数据库传输实用程序现在支持从客户端删除数据库、显示数据库信息，或查看和用电子邮件发送数据库传输日志以及传输数据库。请参见“UltraLiteJ 数据库传输实用程序 (ULjDbT)”一节《UltraLiteJ》。

UltraLiteJ 的行为更改和不建议使用的功能

以下列出了 11.0.1 版中引入的 UltraLiteJ 更改。

- **@@identity 全局变量** UltraLiteJ 不支持 **@@identity** 全局变量。

管理工具新功能

以下列出了 11.0.1 版中引入的 Sybase Central 和 Interactive SQL 新增功能。

Sybase Central 的新功能

以下列出了 11.0.1 版中引入的 Sybase Central 插件新增功能。

所有插件

- **[创建服务向导] 增强** 现在可以使用 [创建服务向导] 为 MobiLink 中继服务器 (rshost)、中继服务器出站启动器 (RSOE)、卷影复制服务 (dbvss11) 和 MobiLink 监听实用程序 (dblsn) 创建服务。请参见“可作为 Windows 服务运行的程序”一节《SQL Anywhere 服务器 - 数据库管理》。

SQL Anywhere 插件

- **查看 Sybase Central 中的事件状态** 现在用户可以在 Sybase Central 中看到各种事件的当前运行状态。在 [事件] 文件夹中，新增了 [运行] 列来显示事件的当前运行状态。在左窗格中选中该文件夹时，该文件夹的内容即被更新。此外，在 [事件属性] 窗口中也有 [运行] 属性。

如果事件正在运行，则运行值显示 [是]，如果未运行，则显示 [否]，或者当事件位于 SQL Anywhere 9.0.2 版或更早版本数据库中时显示 [未知]。

- **查看 Sybase Central 中的维护计划状态** 现在用户可以在 Sybase Central 中看到各种维护计划的当前运行状态。在 [维护计划] 文件夹中，新增了 [运行] 列来显示维护计划的当前运行状态。在左窗格中选中该文件夹时，该文件夹的内容即被更新。

如果维护计划正在运行，则运行值显示 [是]，如果未运行，则显示 [否]，或者当维护计划位于 SQL Anywhere 9.0.2 版或更早版本数据库中时显示 [未知]。

- **维护计划增强** 现在，在配置用于以电子邮件发送维护计划报告的设置时，可使用 [创建维护计划向导] 发送一个测试邮件。请参见“创建维护计划”一节《SQL Anywhere 服务器 - 数据库管理》。

- **创建过程向导和创建函数向导增强** 现在在 [创建函数向导] 和 [创建过程向导] 中，用户可以使用以下 SQL 方言或语言来在 Watcom-SQL、Transact-SQL、外部 C/C++ 或外部环境中编写过程。如果选择外部环境，在选择以下语言之一：C_ESQL32、C_ESQL64、C_ODBC32、C_ODBC64、CLR、JAVA、PERL 或 PHP。之前，只能在 Watcom-SQL 或 Transact-SQL 中进行选择。选择 C/C++ 或 Java 会使用 EXTERNAL NAME 子句为函数或过程生成代码框架。请参见“CREATE FUNCTION 语句（外部过程）”一节《SQL Anywhere 服务器 - SQL 参考》和“CREATE PROCEDURE 语句（外部过程）”一节《SQL Anywhere 服务器 - SQL 参考》。

MobiLink 插件

- **用于 Oracle 数据库的 MobiLink 模型增强** 现在，当使用 [创建同步模型向导] 创建使用 Oracle 统一数据库的同步模型向导时，可以选择加载整个模式，或选择需要使用其表同步的所有者的子集。选择所有者子集可降低模式加载时间。

管理工具的行为更改和不建议使用的功能

以下列出了 11.0.1 版中引入的 Sybase Central 和 Interactive SQL 更改。

Sybase Central 的行为更改和不建议使用的功能

以下列出了 11.0.1 版中引入的 Sybase Central 更改。

SQL Anywhere 插件

- **只读数据库增强** 现在当连接到只读数据库时会接收到警告。通过选择 [工具] » [SQL Anywhere 11] » [首选项] 可隐藏此警告。

Interactive SQL 的行为更改和不建议使用的功能

以下列出了 11.0.1 版中引入的 Interactive SQL 更改。

- **配置数据库锁的自动释放** 现在可以配置 Interactive SQL 在显示结果集时尝试释放其创建的数据库模式锁。要实现此结果，在 Interactive SQL 中选择 [工具] » [选项] » [SQL Anywhere] 并选择 [自动释放数据库锁]。

如果选择了此选项，则在执行返回结果集的语句之后，Interactive SQL 将检查连接是否具有任何未提交的数据库更改。如果没有，则 Interactive SQL 将释放您的模式锁；否则，Interactive SQL 将不会释放您的模式锁。即，如果您有任何未提交的数据库更改，则 Interactive SQL 将不会释放模式锁。

产品范围的新功能

以下列出了 11.0.1 版中引入的产品范围的新增功能。

- **新增的在线文档论坛 - DocCommentXchange (DCX)** 创建了一个名为 DocCommentXchange 的新在线论坛。DocCommentXchange 是一个用于访问和讨论 SQL Anywhere 文档的社区。

使用 DocCommentXchange 可以执行以下任务：

- 查看文档
- 检查是否有用户对文档各部分所做出的阐明
- 提供建议和修正意见以在将来的版本中为所有用户改进文档

访问 <http://dcx.sybase.com>。

- **SQL Anywhere 监控器** SQL Anywhere 监控器是一种基于浏览器的管理工具，可向您提供与 SQL Anywhere 数据库和 MobiLink 服务器的健康情况和可用性有关的信息。

对于 MobiLink 用户，此功能不会取代或覆盖现有 MobiLink 监控器中的功能。

请参见“SQL Anywhere 监控器”《SQL Anywhere 服务器 - 数据库管理》。

文档增强

- **SQL Anywhere 备份和恢复文档增强** SQL Anywhere 随附的关于使用备份和恢复工具的文档已针对此版本重新编写。请参见“[备份和数据恢复](#)”《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **全文搜索文档增强** 具有全文本功能的文档已经过重组，现在包含更多的示例和教程。请参见“[全文搜索](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **MobiLink 服务器启动的同步手册** 自 11.0.0 版开始，MobiLink 服务器启动的同步手册已经进行了改进。例如，改进了目录的编排，对服务器启动的同步进行了更详细的解释，并改进了参考资料的格式。请参见 [MobiLink - 服务器启动的同步](#)。
- **SQL Remote 手册** SQL Remote 手册已经过重新编写和组织，从而增强了导航功能和易用性。请参见 [SQL Remote](#)。

11.0.0 版中的新功能

目录

SQL Anywhere	25
MobiLink	59
QAnywhere	64
SQL Remote	66
UltraLite	67
Sybase Central 和 Interactive SQL	73
文档增强	81
产品范围的功能	82

有关 SQL Anywhere 版本 8 之前版本的新功能和行为更改的信息，请访问 http://www.iAnywhere.com/developer/product_manuals/sqlanywhere/1001/en/html/dbwnen10/dbwnen10.html。

SQL Anywhere

- “SQL Anywhere 的新功能” 一节第 25 页
- “SQL Anywhere 行为更改” 一节第 47 页
- “不建议使用和停用的 SQL Anywhere 功能” 一节第 55 页

MobiLink

- “MobiLink 的新功能” 一节第 59 页
- “MobiLink 的行为更改和不建议使用的功能” 一节第 63 页

QAnywhere

- “QAnywhere 的新功能” 一节第 64 页
- “QAnywhere 的行为更改和不建议使用的功能” 一节第 65 页

SQL Remote

- “SQL Remote 的新功能” 一节第 66 页
- “SQL Remote 的行为更改和不建议使用的功能” 一节第 66 页

UltraLite

- “UltraLite 的新功能” 一节第 67 页
- “UltraLite 的行为更改和不建议使用的功能” 一节第 71 页

Sybase Central 和 Interactive SQL

- “Sybase Central 和 Interactive SQL 的新功能” 一节第 73 页
- “Sybase Central 和 Interactive SQL 的行为更改和不建议使用的功能” 一节第 76 页

文档增强

- “文档增强” 一节第 81 页

不建议使用的功能的列表可能会发生更改

与所有前瞻性声明一样，不建议使用的功能的列表不保证其完整性，并可能会发生更改。

产品范围的功能

- “产品范围的新功能” 一节第 82 页
- “产品范围的行为更改” 一节第 82 页

SQL Anywhere

以下各部分介绍 11.0.0 版的 SQL Anywhere 中的新功能、行为更改和不建议使用的功能。

注意

在发行版本 10 之前，SQL Anywhere 数据库服务器称为 Adaptive Server Anywhere。

SQL Anywhere 的新功能

以下列出了 11.0.0 版中在 SQL Anywhere 数据库和数据库服务器方面新增的功能。

主要功能

以下列出了 SQL Anywhere 11.0.0 版中引入的主要功能。

- **支持合并表** 现在，通过 SQL Anywhere 可以将多个表、视图和系统过程结果合并到一个表或视图中。请参见“[MERGE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **支持登录策略** SQL Anywhere 现已支持登录策略。登录策略是一组用于定义在为数据库连接时所应用的规则的选项。SQL Anywhere 提供用于存储所有登录策略的缺省值的根策略。可以创建单独的登录策略以存储根登录策略选项的替换值。可以为每位用户指派一个登录策略，但不能通过组成员资格继承任何登录策略。可以根据需要修改对特定用户的策略指派。请参见“[管理登录策略概述](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
要利用此功能，需要将数据库升级。请参见“[升级 SQL Anywhere](#)”一节第 324 页。
- **支持全文搜索** SQL Anywhere 现在支持全文搜索。全文搜索可在数据库中快速查找某个术语（单词）的所有实例。全文搜索与使用谓语句（例如 LIKE、REGEXP 和 SIMILAR TO）进行的搜索不同，因为全文搜索是基于术语进行搜索并且使用文本索引而不是扫描表中的各行。请参见“[全文搜索](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
您必须升级数据库才能使用全文搜索功能。请参见“[升级 SQL Anywhere](#)”一节第 324 页。
- **支持正则表达式** SQL Anywhere 现在支持使用两个新搜索条件 REGEXP 和 SIMILAR TO 的正则表达式。请参见“[REGEXP 搜索条件](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[SIMILAR TO 搜索条件](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
另请参见“[正则表达式概述](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[LIKE、REGEXP 和 SIMILAR TO 搜索条件](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **现在在事务日志中记录数据库选项设置** 现在，在事务日志中记录在 LOAD 操作期间有效的数据库选项设置。这样，可确保在恢复期间应用事务日志而导致选项（例如 date_order 和 nearest_century）在原始 LOAD 操作与最终 LOAD 操作之间发生变化时，数据不会存在不一致。
- **增强了优化程序对于索引的使用** 在 SQL Anywhere 的索引功能方面进行了几处增强。您必须升级数据库才能使用这些新功能。请参见“[升级 SQL Anywhere](#)”一节第 324 页。
 - **支持多索引扫描** 将优化程序增强后，可以考虑多索引（多达四个），以便基于基本表上的多个谓语句从该表检索数据。以前，每次查询只能指定一个索引作为索引提示。通过

SELECT 语句的 WITH 子句中的一个新索引提示，可以指定使用多索引扫描。请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。

已经添加了一个新的表访问算法：多索引扫描。请参见“MultipleIndexScan 方法 (MultIdx)”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **支持仅索引检索** 已增强优化程序，使其支持仅索引检索。通过仅索引检索，仅使用索引中的数据而无需访问表中的各对应行就可以满足查询的要求。在可能的情况下，优化程序始终执行仅索引检索。可以使用 INDEX ONLY { ON | OFF } 提示控制是否执行仅索引检索。请参见“索引可用于满足谓词的要求”一节《SQL Anywhere 服务器 - SQL 的用法》和“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **对装载和卸载数据的增强** 在装载和卸载数据方面进行了以下增强：
 - **从客户端计算机上的文件装载数据及向其卸载数据** 通过 SQL 语句和函数，数据库服务器已经为读取和写入驻留在数据库服务器计算机上的数据提供了支持。已经实现的新功能针对驻留在客户端计算机上的文件扩展了此功能，即无需将客户端文件复制到数据库服务器计算机上。对客户端计算机上的数据提供安全性和访问控制的同时，实现了高效率地传输数据。

通过客户端库透明地完成对于客户端计算机上文件的实际读取，这意味着现有客户端应用程序可以立即开始从使用新 SQL 语言支持的新功能受益。

要从这些新功能受益，客户端和数据库服务器都必须为 SQL Anywhere 11.0.0 版，并且客户端必须使用命令序列通信协议 (CmdSeq)。

请参见“访问客户端计算机上的数据”一节《SQL Anywhere 服务器 - SQL 的用法》。
 - **将数据卸载到变量中** 将 UNLOAD 语句增强后，其中包含一个 INTO VARIABLE 子句，通过该子句可将数据卸载到变量中。请参见“UNLOAD 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **从另一个表的某列中装载数据** 将 LOAD TABLE 语句增强后，其中包含一个 USING COLUMN 子句，通过该子句可从另一个表的某列中装载数据。请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“使用 LOAD TABLE 语句导入数据”一节《SQL Anywhere 服务器 - SQL 的用法》。

要利用此新功能，需要将数据库升级。请参见“升级 SQL Anywhere”一节第 324 页。
 - **从某个值 (BLOB) 装载数据** 将 LOAD TABLE 语句增强后，其中包含一个 USING VALUE 子句，通过该子句可从某个值表达式（例如函数或系统过程的结果）装载数据。请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“使用 LOAD TABLE 语句导入数据”一节《SQL Anywhere 服务器 - SQL 的用法》。

要利用此新功能，需要将数据库升级。请参见“升级 SQL Anywhere”一节第 324 页。
 - **LOAD TABLE 语句恢复和镜像功能的增强** 以前，在被镜像的数据库配置中，不支持使用 LOAD TABLE 语句从某个文件装载数据，因为在事务日志中仅记录 LOAD TABLE 语句，而不记录正在被装载的数据。此外，在恢复数据库时，除非在恢复期间存在原始装载文件，否则使用 LOAD TABLE 语句装载的数据不可恢复。

在增强后的 LOAD TABLE 语句中包含以下三个新的记录选项子句：WITH CONTENT LOGGING、WITH ROW LOGGING 和 WITH FILE NAME LOGGING。通过这些子句，可以控制是否将装载的数据记录在事务日志中。在数据库镜像系统中，这些数据可用于装载镜像

数据库。此外，在恢复期间，不再需要存在装载文件。请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

要利用此功能，需要将数据库升级。请参见“升级 SQL Anywhere”一节第 324 页。

● **实例化视图的增强** 对实例化视图的支持增强如下：

- **支持快速实例化视图** 现在，可将实例化视图配置为当基础表中的数据变化影响实例化视图中的数据时立即刷新。具有此刷新类型的视图称作**快速视图**；不立即刷新的视图现在称为**手工视图**。将在此发行版本之前创建的实例化视图视为手工视图，并且将其作为创建新实例化视图时的缺省视图。

有关手动视图和快速视图的详细信息，请参见“使用实例化视图”一节《SQL Anywhere 服务器 - SQL 的用法》和“手动和快速实例化视图”一节《SQL Anywhere 服务器 - SQL 的用法》。

您必须升级数据库以使用支持此功能的那些系统过程。请参见“升级 SQL Anywhere”一节第 324 页。

- **立即刷新多个实例化视图的功能** 以前，每次只能刷新一个实例化视图。在每次刷新操作之间对基础数据的更改会导致实例化视图之间的不一致。现在，通过使用相同的数据刷新多个实例化视图，可以为 REFRESH MATERIALIZED VIEW 语句指定一个实例化视图的列表。请参见“REFRESH MATERIALIZED VIEW 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“刷新手动视图”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **用于 REFRESH MATERIALIZED VIEW 语句的新 WITH SHARE MODE 子句** 已将新 WITH SHARE MODE 子句添加到 REFRESH MATERIALIZED VIEW 语句中。发生刷新操作时，此模式赋予其它事务对基础表的读取权限。指定此子句后，在执行刷新操作之前，所有基础基本表都获得了共享表锁。现在，缺省模式为 WITH SHARE MODE，除非将实例化视图定义为 IMMEDIATE REFRESH，否则将为数据库启用快照隔离。有关缺省刷新行为的详细信息，请参见“REFRESH MATERIALIZED VIEW 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **支持查询文件或 BLOB 字符串的内容** 现在，通过使用 FROM 子句的新 OPENSTRING 子句，可以查询文件或 BLOB 字符串中的数据。通过 OPENSTRING 子句，可以指定要查询的对象以及数据的模式和其它分析信息。请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。

执行 OPENSTRING 操作时，会在执行计划中显示新计划项 OpenString。请参见“OpenString 算法 (OpenString)”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **改进了的对压缩索引的支持** 由于对压缩索引支持的改进工作，当通过卸载和重装来重建数据库时，重建的数据库可能会比原始数据库小。数据库大小的减小并不表示出现了问题或有数据丢失。

- **对在镜像服务器上运行的数据库的只读访问** 如果正在使用数据库镜像，现在则可以连接到在镜像服务器上运行的数据库。这样，可以将有可能耗费大量资源的报告操作下放给镜像服务器，同时使得主服务器处于可用状态。通过 -sm 服务器选项（可用于访问只读镜像数据库）提供数据库服务器名，可以连接到镜像数据库。请参见“-sm 数据库选项”一节《SQL Anywhere 服务器 - 数据库管理》和“配置在镜像服务器上运行的数据库的只读访问”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库连接

以下列出了 SQL Anywhere 11.0.0 版中对数据库连接的增强。

- **AppInfo 连接参数增强** AppInfo 连接参数现在支持 OSUSER 关键字。此关键字返回与客户端进程相关联的操作系统用户名。Linux 和 Solaris 现在支持 EXE 关键字。请参见“[AppInfo 连接参数 \[APP\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **Elevate 连接参数** Elevate 连接参数会提升在 Windows Vista 上自动启动的 SQL Anywhere 数据库服务器。请参见“[Elevate 连接参数](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **NewPassword 连接参数 [NEWPWD]** 通过 NewPassword 连接参数，用户可以更改口令（即使口令已经到期），而无需 DBA 干预。请参见“[NewPassword 连接参数 \[NEWPWD\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **对预取的增强** 更改了 PrefetchBuffer (PBUF) 连接参数的缺省值。在 Windows Mobile 上的缺省值现为 64 KB，而在所有其它平台上的缺省值现为 512 KB。此连接参数接受介于 64 KB 和 8 MB 之间的值。请参见“[PrefetchBuffer 连接参数 \[PBUF\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

在先前的版本中，预取的最大行数基于可预取的最大数据量。现在，预取的最大行数还考虑到实际预取的数据量以及由 PrefetchBuffer 连接参数指定的数据限制。这样，在某列中数据量远远小于主机变量长度和描述长度时可以获得更高的性能。

如果有可能导致改进性能，预取还可以动态地增加预取的行数。请参见“[预取行](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。

备份和恢复

以下列出了在 SQL Anywhere 11.0.0 版中引入的对备份和恢复的增强。

- **支持 Microsoft 卷影复制服务 (VSS)** SQL Anywhere 与 Microsoft 卷影复制服务 (Volume Shadow Copy Service, 简称 VSS) 兼容。要使用 VSS，必须重建所有现有数据库。请参见“[使用 SQL Anywhere 卷影复制服务 \(VSS\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

安全性

以下列出了在 SQL Anywhere 11.0.0 版中引入的安全增强。

- **现在，启用表加密后，会将 ISYSUSER 和 ISYSEXTERNLOGIN 系统表加密** 以前，加密数据库时或者在启用表加密的情况下创建数据库时，会自动将 ISYSCOLSTAT 系统表加密。现在，也将 ISYSUSER 和 ISYSEXTERNLOGIN 系统表加密，这样就提供了附加的安全性。
- **对审计的增强** 现在，可以通过 Sybase Central 控制审计。具有 DBA 权限的用户，可从 [数据库属性] 窗口启用审计、禁用审计以及指定需要审计的信息。在 Sybase Central 右侧窗格的 [审计] 选项卡中可以查看审计信息。请参见“[控制审计](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[检索审计信息](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

启用审计后，会记录连接失败的错误，并且指出失败的原因。

- **现在支持 256 位 AES 加密** SQL Anywhere 现在支持对数据库、表、文件和数据的 256 位 AES 加密。此增强影响到的几个方面如下所述：
 - **数据库和表加密** 现在，可为 CREATE DATABASE 语句的 ENCRYPTION 子句指定 AES256 和 AES256_FIPS。请参见“CREATE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - 还可以为初始化实用程序 (dbinit) 和卸载实用程序 (dbunload) 的 -ea 选项指定 AES256 和 AES256_FIPS。请参见“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》和“卸载实用程序 (dbunload)”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **FIPS 认可的算法** 现在，可以在启用了 FIPS 的平台上使用 256 位 FIPS 认可的 AES 算法。请参见“-fips 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **加密和解密数据** 现在，使用 ENCRYPT 和 DECRYPT 函数加密数据时，可以指定 AES256 和 AES256_FIPS。请参见“ENCRYPT 函数 [String]”一节《SQL Anywhere 服务器 - SQL 参考》和“DECRYPT 函数 [String]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **为数据库、事务日志和 dbspace 创建加密副本** 现在，使用 CREATE ENCRYPTED FILE 语句给加密或未加密的数据库、事务日志或 dbspace 创建加密副本时，可以指定 256 位 AES 算法 (AES256 或 AES256_FIPS)。请参见“CREATE ENCRYPTED FILE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **DBTool 支持 256 位 AES 加密** 已将 a_create_db 和 an_unload_db 的结构扩展，使其支持 AES256 和 AES256_FIPS 作为 encryption_algorithm 成员的值。请参见“a_create_db 结构”一节《SQL Anywhere 服务器 - 编程》和“an_unload_db 结构”一节《SQL Anywhere 服务器 - 编程》。
- 另请参见：
- “加密和解密数据库”一节《SQL Anywhere 服务器 - 数据库管理》
 - “数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》
- **jConnect 和 Open Client 支持的口令加密** 现在，jConnect 连接和 Open Client 连接支持口令加密。请参见：
 - “使用 jConnect JDBC 驱动程序”一节《SQL Anywhere 服务器 - 编程》
 - “部署 JDBC 客户端”一节《SQL Anywhere 服务器 - 编程》
 - “SQL Anywhere 的已知 Open Client 限制”一节《SQL Anywhere 服务器 - 编程》

数据库权限和特权

下面的列表包含 SQL Anywhere 中新的或增强的权限和特权。您必须升级数据库才能使用这些更改。请参见“升级 SQL Anywhere”一节第 324 页。

- **支持继承某些特权** 现在，SQL Anywhere 支持继承某些特权，如下所示：
 - PROFILE、READCLIENTFILE、READFILE 和 WRITECLIENTFILE 特权可以继承
 - DBA、BACKUP 和 RESOURCE 特权不可继承
 请参见“特权概述”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新 PROFILE 特权** 以前，用户需要具有 DBA 特权才能执行应用程序分析和诊断跟踪。现在，具有 PROFILE 特权的用户也可以执行这些任务。有关 PROFILE 特权的可用权限的完整列表，请参见“PROFILE 特权”一节《SQL Anywhere 服务器 - 数据库管理》。
- **新 READFILE 特权** READFILE 特权允许用户使用 SELECT 语句的 OPENSTRING 子句从文件中进行选择。请参见“READFILE 特权”一节《SQL Anywhere 服务器 - 数据库管理》和“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新 READCLIENTFILE 特权** READCLIENTFILE 特权允许用户读取客户端计算机上的文件。例如，要使用 LOAD TABLE 语句从客户端计算机上的文件装载数据，用户则需要具有 READCLIENTFILE 特权。请参见“READCLIENTFILE 特权”一节《SQL Anywhere 服务器 - 数据库管理》。
- **新 WRITECLIENTFILE 特权** WRITECLIENTFILE 特权允许用户写入客户端计算机上的文件。例如，要使用 UNLOAD TABLE 语句将数据卸载到客户端计算机上的文件中，用户则需要具有 WRITECLIENTFILE 特权。请参见“WRITECLIENTFILE 特权”一节《SQL Anywhere 服务器 - 数据库管理》。
- **dbspace 支持 CREATE 权限** 现在，SQL Anywhere 支持用于在指定 dbspace 上创建数据库对象的 CREATE ON 权限。可以将 CREATE 权限指派给用户，也可以通过组成员资格继承。请参见“GRANT 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“权限概述”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库实用程序

以下列出了 SQL Anywhere 11.0.0 版中对数据库实用程序的增强。

- **配置文件的增强** 现在，可在配置文件中使用和号 (&) 以表示其前面的标识会在下一行继续。请参见“使用配置文件”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **卸载实用程序 (dbunload) 增强** 对 dbunload 进行了以下增强：
 - 通过添加的新选项 -cp，可以导致 dbunload 压缩数据输出文件。
 - 以前，如果指定 -ck、-ep 或 -ea 加密选项而未同时指定 -an 或 -ar 重装选项时，会返回错误。但现在，dbunload 接受这些加密选项并且将它们应用于它创建的输出文件。
 - 现在，-g 选项会刷新定义为 MANUAL REFRESH 的文本索引。请参见“文本索引”一节《SQL Anywhere 服务器 - SQL 的用法》。
 - 缺省情况下，不会将定义为 MANUAL REFRESH 的文本索引作为重装的一部分进行初始化。如果需要初始化这些文本索引，可以指定 dbunload -g 选项。
 - 通过 -no 选项，可以按字母顺序卸载对象定义，并且按对象类型分组。这会有助于比较各数据库的 *reload.sql* 文件。
- 请参见“卸载实用程序 (dbunload)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **校验实用程序 (dbvalid) 的增强** 以前，缺省情况下，dbvalid 实用程序会校验所有表和实例化视图。现在，dbvalid 也执行 VALIDATE DATABASE 语句。

通过运行校验实用程序自动启动某个数据库时，该数据库会以只读方式启动。这样可以防止在作为备份和恢复计划的一部分校验数据库时，对该数据库进行更改。

请参见“校验实用程序 (dbvalid)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **翻译日志文件实用程序 (dbtran) 检测到损坏情况时维护生成的 .sql 文件** 如果由于事务日志文件中的损坏导致翻译日志文件失败时，通过翻译日志文件实用程序 -k 选项，可以指定不删除部分 .sql 文件。请参见“翻译日志文件实用程序 (dbtran)”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库选项

以下列出了 SQL Anywhere 11.0.0 版中对数据库选项的增强。

- **allow_read_client_file 选项** 此选项控制是否允许读取客户端计算机上的文件。请参见“allow_read_client_file 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **allow_write_client_file 选项** 此选项控制是否允许写入客户端计算机上的文件。请参见“allow_write_client_file 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **login_procedure 选项** 现在，可以发出到期口令错误消息。请参见“login_procedure 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **max_priority 选项** 此选项控制数据库连接的最大优先级级别。请参见“max_priority 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **priority 选项** 此选项控制优先级级别，根据这些级别执行来自连接的请求。请参见“priority 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **query_mem_timeout 选项** 此选项控制请求等待内存授予的时间长短。请参见“query_mem_timeout 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库服务器选项

以下列出了 SQL Anywhere 11.0.0 版中对数据库服务器选项的增强。

- **-es 服务器选项** SQL Anywhere 先前的版本中，如果使用 -ec 选项启动数据库服务器（以支持传送层安全性），但在允许的加密协议列表中不包含 NONE 或 SIMPLE，则不会启动共享内存端口，因为它不支持传送层安全性。这意味着所有到数据库服务器的连接必须使用高度加密通过 TCP/IP 进行。
-es 服务器选项指示数据库服务器通过共享内存允许未加密的连接。请参见“-es 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **-gb 服务器选项** 现在，在 Unix 和 Windows 上都支持用于控制服务器进程优先级级别的 -gb 服务器选项。请参见“-gb 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **-im 服务器选项** 如果您的应用程序容许丢失所有数据库操作，则可以完全在内存中运行数据库。此功能用于打算将 SQL Anywhere 用作快速、临时的数据存储器的情况下，此时会快速插入数据。请参见“-im 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **从客户端计算机上的文件读取和向其写入** 现在，通过 -sf 服务器选项可以控制从客户端计算机上的文件读取和向其写入的能力。请参见“-sf 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **-um 服务器选项** -um 选项允许用户连接到 *DBLauncher.app* 实例（如果该实例正在运行），并会在 *DBLauncher.app* 的新窗口中显示数据库服务器消息。该选项仅适用于 Mac OS X。请参见“-um 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Windows 性能监控器选项** 已经添加了以下服务器选项，以进一步配置 Windows 性能监控器：
 - **-ks 选项** 禁止创建性能监控器用于从数据库服务器收集计数器值的共享内存。请参见“-ks 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **-ksc 选项** 指定性能监控器可以监控的最大连接数。请参见“-ksc 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **-ksd 选项** 指定性能监控器可以监控的数据库的最大数目。请参见“-ksd 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

属性和性能监控器统计信息

以下列出了 SQL Anywhere 11.0.0 版中对属性和性能监控器统计信息的增强。

- **新连接属性** 本发行版本中添加了以下连接属性：

- allow_read_client_file
- allow_write_client_file
- AuthType
- CacheReadWorkTable
- ClientNodeAddress
- DiskReadWorkTable
- DiskSyncRead
- DiskSyncWrite
- DiskWaitRead
- DiskWaitWrite
- DiskWriteHint
- DiskWriteHintPages
- LockIndexID
- LockRowID
- max_priority
- OSUser
- priority
- query_mem_timeout
- QueryMemActiveCurr
- QueryMemExtraAvail
- QueryMemGrantFailed
- QueryMemGrantGranted
- QueryMemGrantWaiting
- QueryMemGrantRequested
- QueryMemWaited
- ServerNodeAddress
- ReadHint
- ReadHintScatter

有关这些属性的说明，请参见“连接属性”一节《SQL Anywhere 服务器 - 数据库管理》。

● **新数据库服务器属性** 本发行版本中添加了以下数据库服务器属性：

- DiskRetryRead
- DiskRetryReadScatter
- DiskRetryWrite
- EventTypeDesc
- EventTypeName
- HttpAddresses
- HttpsAddresses
- HttpNumActiveReq
- HttpNumConnections
- HttpNumSessions
- HttpsNumActiveReq
- HttpsNumConnections
- MaxEventType
- MaxRemoteCapability
- MessageCategoryLimit
- OptionWatchAction
- OptionWatchList
- QueryMemActiveCurr
- QueryMemActiveEst
- QueryMemActiveMax
- QueryMemExtraAvail
- QueryMemGrantBase
- QueryMemGrantBaseMI
- QueryMemGrantExtra
- QueryMemGrantFailed
- QueryMemGrantGranted
- QueryMemGrantWaiting
- QueryMemGrantRequested
- QueryMemPages
- QueryMemPercentOfCache
- QueryMemWaited
- ReadHintScatterLimit
- RemoteCapability
- StreamsUsed
- TcpIpAddresses
- WebClientLogFile
- WebClientLogging

有关这些属性的说明，请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。

● **新数据库属性** 本发行版本中添加了以下数据库属性：

- AlternateMirrorServerName
- CacheReadWorkTable
- DiskReadWorkTable
- DiskRetryReadScatter
- DiskSyncRead
- DiskSyncWrite
- DiskWaitRead
- DiskWaitWrite
- DiskWriteHint
- DiskWriteHintPages
- HasEndianSwapFix
- MirrorMode
- ReadHint
- ReadHintScatter

有关这些属性的说明，请参见“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新增性能监控器统计信息** 本发行版本中添加了以下性能监控器统计信息：

- 高速缓存读取：工作表
- 磁盘读取：工作表

系统过程和函数

以下列出了在 SQL Anywhere 11.0.0 版中添加的对系统过程和函数的增强。

- **sa_get_dtt_groupreads 系统过程** 通过新增的 sa_get_dtt_groupreads 系统过程，可以估计在数据库服务器上发出组读取的成本。请参见“sa_get_dtt_groupreads 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **PROPERTY_NAME 函数增强** 现在，通过为指定连接级别提供的属性 ID，返回该属性的名称。请参见“PROPERTY_NAME 函数 [System]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **READ_CLIENT_FILE 函数** 新增 READ_CLIENT_FILE 函数从客户端计算机上的指定文件读取数据。请参见“READ_CLIENT_FILE 函数 [String]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **WRITE_CLIENT_FILE 函数** 新增 WRITE_CLIENT_FILE 函数向客户端计算机上的指定文件写入数据。请参见“WRITE_CLIENT_FILE 函数 [String]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **REGEXP_SUBSTR 函数** 通过新增的 REGEXP_SUBSTR 函数，可在字符串中搜索其子串。此新增函数将正则表达式作为其参数。请参见“REGEXP_SUBSTR 函数 [String]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_char_terms 系统过程** 新增 sa_char_terms 系统过程将一个 CHAR 字符串拆分为多个语词，然后返回每个语词及其位置。请参见“sa_char_terms 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

- **sa_nchar_terms 系统过程** 新增 sa_nchar_terms 系统过程将一个 NCHAR 字符串拆分为多个语词，然后返回每个语词及其位置。请参见“sa_nchar_terms 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_refresh_text_indexes 系统过程** 新增 sa_refresh_text_indexes 系统过程刷新所有定义为 MANUAL REFRESH 或 AUTO REFRESH 的文本索引。请参见“sa_refresh_text_indexes 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_text_index_stats 系统过程** 新增 sa_text_index_stats 系统过程为数据库中的所有文本索引返回统计信息，其中包括上次刷新时间以及待执行更改的大小。请参见“sa_text_index_stats 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_text_index_vocab 系统过程** 新增 sa_text_index_vocab 系统过程列出所有在文本索引中出现的语词以及每个语词在其中出现的索引值的总数。请参见“sa_text_index_vocab 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

还增加了两个新系统过程 sa_internal_text_index_vocab 和 sa_internal_text_index_postings，但它们仅供 sa_text_index_vocab 系统过程使用。

- **sa_text_index_postings 系统过程** 此新增系统过程仅供内部使用。
- **sa_text_index_handles 系统过程** 此新增系统过程仅供内部使用。
- **sa_get_user_status 系统过程** 通过新增 sa_get_user_status 系统过程，可以确定某位用户当前的登录状态。请参见“sa_get_user_status 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **以调用者身份运行过程和函数** 现在，创建过程或函数时可以指定过程或函数的运行似乎由调用它的用户（调用者）调用或者似乎由创建它的用户（定义者）调用。要指定这一点，需使用 CREATE PROCEDURE 或 CREATE FUNCTION 语句的 SQL SECURITY 子句。请参见“CREATE FUNCTION 语句（Web 服务）”一节《SQL Anywhere 服务器 - SQL 参考》和“CREATE PROCEDURE 语句（Web 服务）”一节《SQL Anywhere 服务器 - SQL 参考》。

此更改也应用于外部过程和函数。

- **sa_disk_free_space 系统过程** 现在，sa_disk_free_space 系统过程返回一个新列 total_space，指示 dbspace 所驻留驱动器的可用磁盘空间总量。对于使用 SQL Anywhere 11.0.0 之前版本创建的数据库，必须升级后才能返回 total_space 列。请参见“sa_disk_free_space 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_external_library_unload 系统过程** 通过已添加的新系统过程 sa_external_library_unload，可以卸载未在使用中的外部库。请参见“sa_external_library_unload 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **现在 sa_index_density 系统过程返回偏差** 已增强的 sa_index_density 系统过程返回在索引中出现的偏差量。与匀称的索引相比，高度偏差会影响性能。请参见“减少索引碎片和分布偏差”一节《SQL Anywhere 服务器 - SQL 的用法》和“sa_index_density 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_materialized_view_info 系统过程增强** 已将由 sa_materialized_view_info 在 Status 列中返回的信息拆分为两列 Status 和 DataStatus。现在，Status 列返回有关是否启用或禁用视图的信息。新增的 DataStatus 列返回有关在视图中是否包含数据以及数据更新度的信息。添加的附加列

RefreshType 指示视图是手工视图还是快速视图。请参见“[sa_materialized_view_info 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **sa_materialized_view_can_be_immediate 系统过程** 缺省情况下，新创建的实例化视图为手工视图，但只要不违反快速视图的任何限制条件,就可以将其变更为快速视图。通过新增的 sa_materialized_view_can_be_immediate 系统过程，可以测试是否可将某个手工视图更改为快速视图。请参见“[sa_materialized_view_can_be_immediate 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[快速视图的附加限制](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **sa_post_login_procedure 系统过程** 添加的这一新系统过程允许您确定用户的口令将要到期时是否应发出警告。请参见“[sa_post_login_procedure 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **EVENT_PARAMETER 函数增强** 现在，EVENT_PARAMETER 函数支持将异常作为 DisconnectReason。这一新原因，指示所发生的断开连接是客户端应用程序在与数据库断开连接之前异常关闭的结果，或者是在客户端和服务器计算机之间发生通信故障的结果。请参见“[EVENT_PARAMETER 函数 \[System\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **sa_server_option 系统过程增强** 已向 sa_server_option 系统过程中添加了两个新属性 OptionWatchList 和 OptionWatchAction。使用这些属性可以监控为更改数据库选项设置所进行的尝试，并且可以指定要采取的操作。请参见“[监控选项设置](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[sa_server_option 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **sa_db_properties 系统过程增强** 现在，sa_db_properties 系统过程返回包含 NULL 值的有效属性。请参见“[sa_db_properties 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **sa_conn_properties 系统过程增强** 现在，sa_conn_properties 系统过程返回包含 NULL 值的有效属性。请参见“[sa_conn_properties 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

SQL 语句

以下列出了在 SQL Anywhere 11.0.0 版中引入的 SQL 增强。

- **ALTER DATABASE 语句的新 CALIBRATE GROUP READ 子句** 通过 ALTER DATABASE 语句的新 CALIBRATE GROUP READ 子句，可在临时 dbspace 上执行组读取校准。请参见“[ALTER DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **CREATE MATERIALIZED VIEW 语句的新 CHECK 子句** 通过 CREATE MATERIALIZED VIEW 语句的新 CHECK 子句，可在创建视图之前校验该语句。请参见“[CREATE MATERIALIZED VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **ALTER FUNCTION 语句的新 RECOMPILE 子句** 通过添加到 ALTER FUNCTION 语句中的新子句 RECOMPILE，可以重新编译用户定义的函数。请参见“[ALTER FUNCTION 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **ALTER PROCEDURE 语句的新 RECOMPILE 子句** 通过添加到 ALTER PROCEDURE 语句中的新子句 RECOMPILE，可以重新编译存储过程。请参见“[ALTER PROCEDURE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **ALTER MATERIALIZED VIEW 语句的新 REFRESH 子句** 通过添加到 ALTER MATERIALIZED VIEW 语句中的新子句 REFRESH, 可以指定实例化视图的刷新类型。请参见“ALTER MATERIALIZED VIEW 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **LOAD TABLE 语句在支持恢复和镜像功能方面的增强** 添加到 LOAD TABLE 语句中的以下子句用于支持恢复和镜像功能:
 - **WITH CONTENT LOGGING 子句** WITH CONTENT LOGGING 子句指示数据库服务器在事务日志中记录数据源的内容。由于通过 LOAD TABLE 处理输入, 所以使用小数据块记录数据。通过某个镜像数据库或者在从事务日志恢复数据时, 可将这些数据块重组到行中。如果不希望保留原始数据文件供以后恢复, 则 WITH CONTENT LOGGING 子句会很有用。请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **WITH ROW LOGGING 子句** WITH ROW LOGGING 子句指示数据库服务器象一系列 INSERT 语句那样记录正在装载的所有行。对于涉及同步的数据库以及正在装载的表中包含非确定性值(例如计算列)或 CURRENT TIMESTAMP 缺省值的情况下, 这种级别最为理想。请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **WITH FILE NAME LOGGING 子句** WITH FILE NAME LOGGING 子句指示数据库服务器仅记录 LOAD TABLE 语句。这是缺省行为, 该行为与 SQL Anywhere 先前版本中的记录行为一致。请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新客户端文件的装载和卸载子句** 对 LOAD TABLE 和 UNLOAD TABLE 语句的增强支持新客户端文件装载/卸载功能:
 - **LOAD TABLE 语句的新 USING CLIENT FILE 子句** 用于使用位于客户端计算机上的文件中的数据装载某个表。请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **UNLOAD TABLE 语句的新 INTO CLIENT FILE 子句** 用于指定将数据卸载到其中的客户端计算机上的文件。请参见“UNLOAD 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新登录策略子句** 添加了以下语句来支持新登录策略功能:
 - “CREATE LOGIN POLICY 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “ALTER LOGIN POLICY 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “DROP LOGIN POLICY 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “CREATE USER 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “ALTER USER 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “DROP USER 语句”一节《SQL Anywhere 服务器 - SQL 参考》
- **新全文搜索语句和子句** 添加了以下语句来支持新的全文搜索功能:
 - **新 CONTAINS 搜索条件** CONTAINS 搜索条件用于为存在的语词或短语的指定列表检查列的指定列表。CONTAINS 搜索条件返回 TRUE 或 FALSE。搜索多个语词或短语时, 可以使用不同的布尔运算符将它们组合在一起。请参见“CONTAINS 搜索条件”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **SELECT 语句的 FROM 子句中的新 CONTAINS 子句** 在 SELECT 语句的 FROM 子句中指定 CONTAINS 子句, CONTAINS 子句的工作方式与 CONTAINS 搜索条件的工作方式很相似, 但是它为每个匹配列返回一个分数并为每个匹配行返回一个总分数。请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **CREATE TEXT CONFIGURATION 语句** 此语句创建一个文本配置对象。文本配置对象是一组控制文本索引特性的配置设置。请参见“CREATE TEXT CONFIGURATION 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER TEXT CONFIGURATION 语句** 此语句变更一个文本配置对象。请参见“ALTER TEXT CONFIGURATION 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **DROP TEXT CONFIGURATION 语句** 此语句删除一个文本配置对象。请参见“DROP TEXT CONFIGURATION 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **CREATE TEXT INDEX 语句** 此语句创建一个文本索引。文本索引在每个索引列中存储每个语词的每个实例的完整定位信息。请参见“CREATE TEXT INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER TEXT INDEX 语句** 此语句变更一个文本索引。请参见“ALTER TEXT INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **DROP TEXT INDEX 语句** 此语句从数据库中删除文本索引。请参见“DROP TEXT INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **REFRESH TEXT INDEX 语句** 此语句刷新一个文本索引。请参见“REFRESH TEXT INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **TRUNCATE TEXT INDEX 语句** 此语句从文本索引中截断数据。请参见“TRUNCATE TEXT INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER EVENT 语句增强** 现在，可以使用 ALTER EVENT ... SET HIDDEN 语句来隐藏某个事件处理程序的定义。SET HIDDEN statement.此语句会导致模糊处理存储在 ISYSEVENT 系统表的操作列中的事件处理程序定义。请参见“ALTER EVENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **BEGIN SNAPSHOT 语句** BEGIN SNAPSHOT 语句用于针对快照隔离控制快照启动的时间。请参见“BEGIN SNAPSHOT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **CASE 语句和 CASE 表达式的增强功能** 为改进兼容性，现在 CASE 语句和 CASE 表达式可以使用 END 或 END CASE 结束。请参见“CASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“CASE 表达式”一节《SQL Anywhere 服务器 - SQL 参考》。
- **COMMENT 语句增强** 现在，可将注释添加到登录策略表和 dbspace 中。请参见：
 - “COMMENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “管理登录策略概述”一节《SQL Anywhere 服务器 - 数据库管理》
 - “使用附加 dbspace”一节《SQL Anywhere 服务器 - 数据库管理》
- **CREATE MATERIALIZED VIEW 语句增强** 现在，通过使用 CREATE MATERIALIZED VIEW 语句的新 IMMEDIATE REFRESH 子句，可以创建一个每当基础数据更改时都刷新的实例化视图。请参见“CREATE MATERIALIZED VIEW 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **DESCRIBE 语句增强** 现在，可以通过 Interactive SQL DESCRIBE 语句获取有关 Interactive SQL 所连接的数据库或数据库服务器的信息。请参见“DESCRIBE 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **IF 语句和 IF 表达式增强** 为改进兼容性，现在 IF 语句和 IF 表达式可以使用 ENDIF 或 END IF 结束。请参见“IF 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“IF 表达式”一节《SQL Anywhere 服务器 - SQL 参考》。
- **LOAD TABLE 语句增强** 现在使用 LOAD TABLE 语句时，可以通过新 COMPRESSED 或 ENCRYPTED 子句指定是否压缩和/或加密输入文件中的数据。请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **SELECT 语句增强**
 - **对 INDEX 子句的增强** 使用 INDEX 子句指定索引提示时，现在可指定多达四个数据库服务器必须使用的索引。请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **新 INDEX ONLY 子句** 使用 INDEX 子句指定索引提示时，可以选择指定 INDEX ONLY 子句来控制数据库服务器是否执行仅索引检索（即，仅使用索引数据来满足查询的需要）。请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **新 CROSS APPLY 和 OUTER APPLY 子句** 已将 SELECT 语句扩展为支持 FROM 子句中的 apply 表达式（具体地说，就是 CROSS APPLY 和 OUTER APPLY 子句）。apply 表达式是一种指定右侧取决于左侧的连接的简单方式。例如，可以使用一个 apply 表达式来计算过程或派生表，对表表达式中每行计算一次。请参见“从 apply 表达式生成的连接”一节《SQL Anywhere 服务器 - SQL 的用法》和“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **新 OPENSTRING 子句** 通过新 OPENSTRING 子句，现在可以使用一个 SELECT 语句来查询某个文件中的数据。请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **在创建、变更、删除或注释事件时指定所有者** 现在，通过 CREATE EVENT、ALTER EVENT、DROP EVENT 和 COMMENT ON EVENT 语句可以选择指定所有者。请参见：
 - “CREATE EVENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “ALTER EVENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “DROP EVENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “COMMENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》
- **UNLOAD 语句增强** 现在使用 UNLOAD 语句时，可以通过分别指定 COMPRESSED 或 ENCRYPTED 子句来指定是否压缩和/或加密正在卸载的数据。请参见“UNLOAD 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

使用这些子句压缩或加密的文件仅可以通过 SQL Anywhere 11.0.0 数据库服务器（或更新的版本）来装载（例如，使用 LOAD TABLE）。对于 SQL Anywhere 而言，使用其它工具压缩或加密的文件不可用。
- **UPDATE 语句增强** 对于搜索更新和定位更新，现在都可以使用 SET 子句将列值设置为其缺省值。请参见“UPDATE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“UPDATE（定位）语句 [ESQL] [SP]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **OPTION 子句的扩展** 现在, INSERT、UPDATE、DELETE、SELECT、UNION、EXCEPT 和 INTERSECT 语句中的 OPTION 子句可以替换 user_estimates 数据库选项的设置。请参见:
 - “INSERT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “UPDATE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “DELETE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “SELECT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “UNION 子句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “EXCEPT 子句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “INTERSECT 子句”一节 《SQL Anywhere 服务器 - SQL 参考》

编程接口

以下列出了在 SQL Anywhere 11.0.0 版中引入的编程接口增强。

- **新的 SQL Anywhere C API** SQL Anywhere C 应用程序编程接口 (API) 简化了为多种解释性编程语言创建 C 和 C++ 包装驱动程序的过程, 这类解释性编程语言包括 PHP、Perl、Python 和 Ruby。SQL Anywhere C API 处于 DBLIB 库的最上层, 采用嵌入式 SQL 实现。

虽然 SQL Anywhere C API 不能替代 DBLIB, 但可以简化使用 C 和 C++ 创建应用程序的过程。使用 SQL Anywhere C API 无需对嵌入式 SQL 具有很深的了解。请参见 “SQL Anywhere C API 参考” 《SQL Anywhere 服务器 - 编程》。

- **新的 Python 数据库 API (sqlanydb)** 通过 Python 数据库 API (sqlanydb), 可从使用 Python 编写的脚本访问 SQL Anywhere 数据库。sqlanydb 模块及扩展执行 Python Database API Specification 2.0 版。请参见 “SQL Anywhere Python 数据库支持” 《SQL Anywhere 服务器 - 编程》。

- **外部环境** SQL Anywhere 现在包括对以下六个外部运行时环境的支持: Java、Perl、PHP、CLR、嵌入式 SQL 和 ODBC。有一段时间, SQL Anywhere 曾有能力调用以 C 或 C++ 语言编写的已编译本地函数。但是, 当服务器运行这些过程时, 始终由数据库服务器装载动态链接库或共享对象, 并且始终由数据库服务器执行对本地函数的调用。这样做的风险是, 如果本地函数导致故障, 则会使数据库服务器崩溃。所以, 在数据库服务器的外部 (即在外环境中) 运行已编译的本地函数, 可以消除服务器的这些风险。请参见 “SQL Anywhere 外部环境支持” 《SQL Anywhere 服务器 - 编程》。

要利用此新功能, 需要将数据库升级。请参见 “升级 SQL Anywhere” 一节第 324 页。

- **对 PHP 外部环境的支持** SQL Anywhere 11.0.0 包含各个 PHP 版本 (包括 5.1.1 到 5.1.6 以及 5.2.0 到 5.2.6) 的大量预先生成的二进制文件。如果您已经在服务器计算机上安装了这些版本之一, 则应使用 SQL Anywhere 预先生成的二进制文件而不是自行构建 PHP 外部环境。请注意, 对于 Linux 和 Solaris, 同时提供 32 位和 64 位版本的这些二进制文件。对于 Windows 和其它系统, 仅提供 32 位版本。

如果已经安装了与上述 PHP 版本不同的版本, 则必须生成该软件, 或者将您的 PHP 版本转换为与 SQL Anywhere 预先生成的版本匹配的版本。有关生成 SQL Anywhere PHP 模块的说明, 请参见 “SQL Anywhere PHP API” 《SQL Anywhere 服务器 - 编程》。

- **对 Perl 外部环境的支持** 非常重要的一点是, 要在尝试使用 Perl 外部环境之前更新 SQL Anywhere Perl DBD 驱动程序的版本。如果未更新 Perl DBD 驱动程序, 则服务器端的 Perl 将不会工作。

此外，与 PHP 不同的是，我们没有为 Perl 的各个版本准备预先生成的二进制文件。SQL Anywhere Perl DBD 驱动程序的源代码位于 `install-dir\SDK\perl` 中。有关生成 SQL Anywhere Perl DBD 驱动程序的说明，请参见“SQL Anywhere Perl DBD::SQLAnywhere DBI 模块”《SQL Anywhere 服务器 - 编程》。

- **对于 UTF-8 URL 的 Web 服务器支持** 以前，Web 服务器将请求 URL 内部的百分比编码 (%encoded) 数据（或者请求主体中的 application/x-www-form-urlencoded 数据）解码到数据库字符集中。现在，将为 UTF-8 序列测试百分比编码 (%encoded) 数据的内容，并最大程度地将其转换为数据库字符集。对任何非 UTF-8 的编码数据进行解码，并且将这些数据视为似乎它们已在数据库字符集中。

客户端 HTTP 应用程序应以独占方式发送百分比编码 (%encoded) 的 UTF-8 数据。请注意，在 UTF-8 中 ASCII 的表示方法不变。例如，将空格编码为 %20。
- **新客户端回调 API** 添加的新客户端回调 API 支持新客户端装载和卸载数据的功能。有关嵌入式 SQL，请参见“db_register_a_callback 函数”一节《SQL Anywhere 服务器 - 编程》中的 DB_CALLBACK_VALIDATE_FILE_TRANSFER。有关 ODBC，请参见“SQL Anywhere 连接属性”一节《SQL Anywhere 服务器 - 编程》中的 SA_REGISTER_VALIDATE_FILE_TRANSFER_CALLBACK。
- **SQL_ATTR_CONNECTION_DEAD 立即检测失效连接** 如果连接已经删除（即使从连接删除时起未向服务器发出任何请求），使用 ODBC 的 SQLGetConnectAttr 调用来获取 SQL_ATTR_CONNECTION_DEAD 属性现在会获取值 SQL_CD_TRUE。确定删除的连接是否是在未向服务器发出请求的情况下完成的，并且可在几秒钟内检测到删除的连接。有多种原因可导致删除连接，例如空闲超时。在这一改变之前，如果连接断开或者如果连接删除后 ODBC 驱动程序向服务器发出请求（例如，通过调用 SQLExecDirect），则 SQL_ATTR_CONNECTION_DEAD 仅获得值 SQL_CD_TRUE。请参见“获取连接属性”一节《SQL Anywhere 服务器 - 编程》。
- **JDBC 驱动程序现在支持 ResultSet.getBlob().getBinaryStream()** 即使 ResultSet.getBlob() 方法在 JDBC 说明中是可选的，iAnywhere JDBC 驱动程序当前也支持此方法。已添加了对 ResultSet.getBlob().getBinaryStream() 方法的支持。请参见“iAnywhere JDBC 3.0 API 支持”一节《SQL Anywhere 服务器 - 编程》。
- **现在，除 jdbc:odbc 外，iAnywhere JDBC 驱动程序还接受将 jdbc:iAnywhere 作为 URL 标头** 以前，使用 URL 标头 jdbc:odbc 的应用程序非常确定：JDBC 驱动程序管理器将使用 iAnywhere JDBC 驱动程序通过此 URL 建立连接。但是，新版 Java VM 已开始将 Sun JDBC-ODBC bridge 注册为 JDBC 驱动程序，并且由于 Sun JDBC-ODBC bridge 也接受以 jdbc:odbc 开始的 URL，因而应用程序获得 Sun JDBC-ODBC bridge 而非 iAnywhere JDBC 驱动程序的几率非常高。要确保 JDBC 驱动程序管理器使用 iAnywhere JDBC 驱动程序而非 Sun JDBC-ODBC bridge，应用程序应改为使用 URL 标头 jdbc:iAnywhere。请参见“从 JDBC 客户端应用程序连接”一节《SQL Anywhere 服务器 - 编程》。
- **ODBC 驱动程序管理器现在接受 driver=iAnywhere Solutions 11 - Oracle** 现在，UNIX ODBC 驱动程序管理器接受 driver=iAnywhere Solutions 11 - Oracle，并且如果以线程方式执行应用程序，该管理器就会装载 Oracle 的线程 iAnywhere ODBC 驱动程序。如果以非线程方式执行应用程序，该管理器就不会装载驱动程序，因为 Oracle 的非线程 iAnywhere ODBC 驱动程序不受支持。请参见“iAnywhere Solutions Oracle 驱动程序”一节《MobiLink - 服务器管理》。
- **ODBC 驱动程序管理器现在接受 driver=UltraLite 11** Unix ODBC 驱动程序管理器已接受 driver=SQL Anywhere 10 并装载 SQL Anywhere ODBC 驱动程序（是否以线程方式执行，取决于

应用程序)。Unix ODBC 驱动程序管理器现在接受 driver=SQL Anywhere 11 和 driver=UltraLite 11。对于 UltraLite 驱动程序，驱动程序管理器仅装载 UltraLite ODBC 驱动程序的线程版本，因为只存在线程版本。

- **TDS 连接增强** 即使 Open Client 登录服务器名与缺省数据库的名称不匹配，SQL Anywhere 数据库服务器现在也允许与缺省数据库进行 TDS 连接，但前提是要假定在连接字符串中不包含涉及启动数据库的内容（即，不存在 DBF=...）并且假定数据库服务器仅运行一个数据库。
- **现在更易于重新部署管理工具启动程序** 现在更易于为数据库工具（Sybase Central、DBISQL、DBConsole、ML Monitor）重新部署启动程序可执行文件。不再需要用于 JAR 文件位置的注册表条目和一组目录结构。每个可执行文件都需要一个位于相同目录中的 .ini 文件（名称与可执行文件相同），在该文件中包含有关如何装载该工具的详细信息。请参见“部署管理工具”一节《SQL Anywhere 服务器 - 编程》。

- **SQL Anywhere .NET 数据提供程序现在支持分布式事务征用** .NET 2.0 framework 引入了一个新命名空间 System.Transactions，其中包含用于编写事务性应用程序的类。客户端应用程序可以通过一个或多个参与者创建及参与分布式事务。客户端应用程序可以使用 TransactionScope 类隐式地创建事务。该连接对象可以检测周围存在的由 TransactionScope 创建的事务并且自动征用。客户端应用程序还可以创建 CommittableTransaction 并且调用 EnlistTransaction 方法进行征用。

SQL Anywhere .NET 2.0 数据提供程序支持此功能。分布式事务具有很高的性能开销。建议对于非分布式事务使用数据库事务。请参见“事务处理”一节《SQL Anywhere 服务器 - 编程》。

- **SQL Anywhere .NET 数据提供程序现在支持命名参数** SQL Anywhere 提供程序现在支持 SACommand 中的命名参数。如果用户指定所有参数名，该提供程序会在执行该命令时映射数值。使用命名参数时，参数的顺序无需与主机变量的顺序相匹配。

```
SACommand cmd = new SACommand(
    "UPDATE MyTable SET name = :name WHERE id = :id", conn );

SAParameter p1 = new SAParameter(
    "id", SADBType.Integer );
p1.Direction = ParameterDirection.Input;
p1.Value = 1;
cmd.Parameters.Add( p1 );

SAParameter p2 = new SAParameter(
    "name", SADBType.Char, 40 );
p2.Direction = ParameterDirection.Input;
p2.Value = "asdasd";
cmd.Parameters.Add( p2 );

cmd.ExecuteNonQuery();
```

- **Web 服务增强** 本发行版本中进行了以下 Web 服务增强：
 - **扩展 HTTP:POST 类型的 Web 客户端服务过程以允许使用用户定义的主体** 已将 CREATE PROCEDURE 和 CREATE FUNCTION 语句的 TYPE 子句扩展为允许指定 mime 类型。请参见“CREATE FUNCTION 语句（Web 服务）”一节《SQL Anywhere 服务器 - SQL 参考》或“CREATE PROCEDURE 语句（Web 服务）”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **扩展 Web 服务客户端过程以支持 PUT、DELETE 和 HEAD HTTP 方法** Web 服务客户端过程和函数现在支持 PUT、DELETE 和 HEAD HTTP 方法。已将 CREATE PROCEDURE 和 CREATE FUNCTION 语句的 TYPE 子句扩展为支持这些方法。与 POST 方法类似，PUT 在

类型子句中需要一个内容类型扩展并且仅允许一个（非替代）参数。请参见“[CREATE SERVICE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》、“[CREATE FUNCTION 语句 \(Web 服务\)](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[CREATE PROCEDURE 语句 \(Web 服务\)](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **sa_http_php_page 和 sa_http_php_page_interpreted 系统过程** 新 Web 服务系统过程 sa_http_php_page 和 sa_http_php_page_interpreted 返回通过 PHP 解释器传递 PHP 脚本的结果。请参见“[sa_http_php_page 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[sa_http_php_page_interpreted 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **HTTP_BODY 系统函数** 添加了新的 Web 服务函数。HTTP_BODY 函数以二进制形式返回 HTTP 请求的主体。请参见“[HTTP_BODY 函数 \[HTTP\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **WSDL 支持生成 Web 服务客户端 SOAP 过程** 除了为 C# 和 JAVA 生成 QAnywhere 客户端 SOAP 接口外，WSDL 现在还支持为 SQL Anywhere 生成 SQL SOAP（Web 服务）客户端过程。WSDL 读取 WSDL1.1 兼容的 URL 或文件，然后使用相应的参数和子句生成过程（或函数），这些过程（或函数）映射到在 WSDL 内部列出的相应 SOAP 操作。会将生成的 SQL 语句写入某个 SQL 文件中。请参见“[使用 iAnywhere WSDL 编译器](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **使用 FORMAT 子句定义的 HTTP SOAP 服务可通过 EXPLICIT OFF 或 ON 进行进一步限定** 创建 HTTP SOAP 服务时，FORMAT 子句的缺省设置为 EXPLICIT ON。这意味着，由 DISH 服务生成的 WSDL 为在结果集中返回的每一列指定显式名称和数据类型。这就允许 SOAP 客户端工具箱自动生成代表该结果集的客户对象和接口，从而提供对列值的本地访问。在此功能出现前，只能将列值作为抽象 XML 数据元素进行访问。这一行为仍可通过指定 EXPLICIT OFF 来实现。
有关如何定义 EXPLICIT 响应对象或通用 SimpleDataset 的详细信息，请参见“[CREATE SERVICE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[教程：将数据类型与 JAX-WS 一起使用](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **支持 JSON Web 服务** SQL Anywhere 现在支持以 JSON 格式返回响应的 Web 服务。请参见“[CREATE SERVICE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **记录 Web 服务客户端** 数据库服务器现在支持将记录 Web 服务客户端连接到输出文件。可以指定 -zoc 服务器选项或者将 WebClientLogFile 和 WebClientLogging 属性与 sa_server_option 系统过程一起使用，以控制记录并指定 Web 服务客户端日志文件的位置。也可以通过 -sf 服务器选项禁用此功能。请参见：
 - “[-zoc 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
 - “[-sf 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
 - “[sa_server_option 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》

Windows Mobile 增强

以下列出了在 SQL Anywhere 11.0.0 版中引入的 Windows Mobile 增强。

- **支持 -gss 服务器选项** 在 Windows CE 4 (Pocket PC 2003) 及其更高版本上，可以使用 -gss 服务器选项来为内部执行线程指定缺省堆栈大小。请参见“[-gss 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **缺省情况下启用校验和** 在 Windows Mobile 上运行某个数据库时，无论是否为该数据库启用了校验和，数据库服务器都会自动启用校验和。要使用此功能，必须升级现有数据库或者新建数据库。请参见“[使用校验和检测损坏](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

Unix/Linux 增强

以下列出了在 SQL Anywhere 11.0.0 版中引入的对 Unix 和 Linux 的增强。

- **控制临时文件的权限** 在以前的发行版本中，通过数据库服务器和客户端创建的临时文件允许全局读取、写入和执行。现在，通过将 SATMP 环境变量设置为具有所需权限的某个目录，可以控制临时文件的权限。请参见“[SATMP 环境变量](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **SELinux 支持** SELinux 策略控制应用程序对系统资源的访问。可在 Red Hat Enterprise Linux 5 上将缺省策略和 SQL Anywhere 一起使用，但以这种方式运行时 SQL Anywhere 是不安全的。现在，SQL Anywhere 包含一个可在 Red Hat Enterprise Linux 5 上保护其自身安全的策略。但必须编译并安装该策略才能使其起作用。该策略源代码作为 SQL Anywhere 安装的一部分提供。
有关编译和安装 SQL Anywhere SELinux 策略的信息，请参见 *install-dir/selinux/readme*。
- **应用程序 Linux 上的菜单项** 在 Linux 上安装 SQL Anywhere 11 时，可选择创建 [应用程序] 菜单项。

Mac OS X 增强

- **现在在 Mac OS X 上支持加密** 现在，在 Mac OS X 上，数据库服务器和客户端都支持 RSA 通信加密。有关使用高度加密的信息，请参见“[传送层安全简介](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **现在在 Mac OS X 上支持 HTTPS** 现在，在 Mac OS X 上，数据库服务器支持 HTTPS 通信。有关使用 HTTPS 的信息，请参见“[-xs 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

杂类

以下列出了在 SQL Anywhere 11.0.0 版中引入的杂类增强。

- **变更无效视图** 以前，无法变更 INVALID 状态的常规视图，从而需要用户删除该视图然后重新创建视图。现在，可以变更无效视图来更改定义，以使该视图不再为无效状态。
- **添加了对 BIG-ENDIAN 和 LITTLE-ENDIAN UTF-16 编码的支持** 现在，无论平台的字节序如何，SQL Anywhere 在所有平台上都支持 BIG-ENDIAN 和 LITTLE-ENDIAN UTF-16 编码。UTF-16 编码可在 LOAD TABLE 和 UNLOAD 语句中使用，也可以和 CSCONVERT 函数一起使用。但是，不能将 UTF-16 编码用作某个连接或某个数据库的编码。请参见“[LOAD TABLE 语句](#)”一

节《SQL Anywhere 服务器 - SQL 参考》和“UNLOAD 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

要利用此功能，需要将数据库升级。请参见“升级 SQL Anywhere”一节第 324 页。

- **索引性能增强** 已经改进了索引性能，尤其是在使用全部高速缓存的情况下操作时。要受益于增强的索引性能，必须重建索引。实现此目的的最简单方法就是重建数据库。重建后，您会发现数据库文件比以前小多了。这是正常现象，不必为此担忧。
- **现在 INLINE 和 PREFIX 设置与压缩列相关** 以前，忽略为某列指定的 INLINE 和 PREFIX 设置并且在该列被压缩时将 these 设置视为 0。现在，即使将该列压缩，该列的这些设置仍然相关。请参见“存储 BLOB”一节《SQL Anywhere 服务器 - SQL 的用法》和“CREATE TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **现在允许在批处理中使用主机变量** 现在，允许在批处理中引用主机变量，但是具有一些限制。请参见“批处理简介”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **InList 算法的增强** 以前，仅在 IN 列表中的所有元素都是常量值或者可在优化时经计算得到常量值时，优化程序才使用 InList 算法。现在，在 IN 列表谓语中可以包含仅在打开时计算的值（例如 CURRENT DATE、CURRENT TIMESTAMP 或者非确定型系统函数和用户定义的函数）以及在查询块的一次执行中为常量的那些值（外部引用）。请参见“**InList 算法 (IN)**”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **简单 DML 语句的计划高速缓存** 计划高速缓存已扩展为包括可以跳过查询的 SELECT 语句（简单语句）。请参见“计划高速缓存”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **减小了新数据库的大小** 现在，已经压缩了以下系统表中的列以减少新（空）数据库的大小，减少的空间约为 200 KB。这有益于创建在 Windows Mobile 上使用的数据库。
 - ISYSEVENT.action
 - ISYSJARCOMPONENT.contents
 - ISYSPROCEDURE.proc_defn
 - ISYSSOURCE.source
 - ISYSTEXTCONFIG.char_stoplist
 - ISYSTEXTCONFIG.nchar_stoplist
 - ISYSTRIGGER.trigger_defn
 - ISYSVIEW.view_def
- **增加了缺省数据包大小和最小数据包大小** 已将所有操作系统（Windows Mobile 除外）上的缺省数据包大小增加到 7300 字节。在 Windows Mobile 上，缺省数据包大小仍为 1460 字节。最小数据包大小已增大到 500 字节。请参见“CommBufferSize 连接参数 [CBSIZE]”一节《SQL Anywhere 服务器 - 数据库管理》和“-p 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **远程数据访问支持新 ODBC 类** 添加了对以下 ODBC 类的支持：
 - msaccessodbc
 - mysqlodbc
 - ulodbc
 - adsodbc

有关详细信息，请参见“[基于 ODBC 的服务器类](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

迁移 Access 数据库

如果以前使用 SQL Anywhere for MS Access 迁移实用程序（upsizer 工具）将 Microsoft Access 数据库迁移到了 SQL Anywhere，则现在可使用 msaccessodbc 类。

- **数据库服务器消息增强** 现在，为来自数据库服务器的消息指派了类别和严重级。可以使用 sa_server_messages 系统过程访问此信息，并且可以配置通过 MessageCategoryLimit 属性进行维护的消息数量。请参见“[sa_server_messages 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **用于 a_validate_type 枚举的新 VALIDATE_COMPLETE 参数** a_validate_type 枚举包含一个新参数 VALIDATE_COMPLETE，该参数用于对数据库执行所有可能的校验。请参见“[a_validate_db 结构](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **外部卸载增强** 执行数据库的外部卸载时，在所生成的 reload.sql 的开头部分现在包括一条注释过的 CREATE DATABASE 语句。使用该语句，可以创建一个与正要卸载的数据库相当的数据库。

如果被卸载的数据库是使用 SQL Anywhere 版本 9 或更早的版本创建的，并且包含自定义归类，则 COLLATION 子句以如下形式出现：

```
COLLATION collation-label DEFINITION collation-definition
```

其中 *collation-definition* 是用于指定自定义归类的字符串。

如果被卸载的数据库是通过高度加密的方法创建的，则 CREATE DATABASE 语句中的 KEY 子句的值会显示为三个问号 (???)。

有关详细信息，请参见“[内部与外部卸载和重装](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **新 SQL Anywhere Extension Agent OID** 在本发行版本中添加了下列 OID：
 - saAgent.saRestart
 - saAgent.saInifile

有关详细信息，请参见“[SQL Anywhere MIB 参考](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **Deadlock 系统事件** 每当发生死锁时，即会触发 Deadlock 系统事件。事件处理程序可以使用 sa_report_deadlocks 过程以获得有关导致死锁的条件的信息。要使用 Deadlock 系统事件，必须升级现有数据库。请参见“[了解系统事件](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **数据库限制已提高** 已经提高了若干 SQL Anywhere 数据库限制。请参见“[SQL Anywhere 大小和数量限制](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **对执行计划的更改** 现在，通过优化程序生成的详细计划显示与总计划相关的以下条目：
 - **成本最佳计划** 优化程序找到的不同最佳访问计划的数目。
 - **成本计划** 优化程序考虑的不同访问计划的数目。

- **优化时间** 优化查询所用的时间。

请参见“执行计划缩写”一节《SQL Anywhere 服务器 - SQL 的用法》。

图形式计划现在显示以下条目：

- **成本最佳计划** 位于根节点的 [优化程序统计] 部分中的这个条目提供了优化程序找到的不同最佳访问计划的数目。
- **成本计划** 位于根节点的 [优化程序统计] 部分中的这个条目提供了优化程序考虑的不同访问计划的数目。
- **优化时间** 位于根节点的 [优化程序统计] 部分中的这个条目提供了优化查询所用的时间。
- **FirstRowRunTime** 位于任何 [节点统计] 部分中的这个条目提供了读取第一行所用的时间。
- **考虑的连接** 位于任何连接运算符的 [高级详细信息] 部分中的这个条目，列出了对连接运算符左右两侧的子树的优化过程中优化程序所考虑的所有连接运算符。
- **前置过滤器谓词** 位于 [详细信息] 窗格的新扫描节点部分中的这个条目，列出了所有在启动扫描之前被计算的谓词。
- **扫描谓词** 位于 [详细信息] 窗格的某个扫描节点部分中的这个条目，列出了所有作为从该行中读取的列来计算的谓词。如果某个扫描谓词拒绝了某行，则不能再继续读取列。扫描谓词是简单的单个列谓词，如 [T.x <= 3] 或 [T.x IS NULL]。
- **发布扫描谓词** 位于 [详细信息] 窗格的新扫描节点部分中的这个条目，列出了从表页中读取某行后立即计算的谓词。发布扫描谓词可以引用多个列并且可以使用函数或算术。
- **驻留谓词** 位于 [详细信息] 窗格的新扫描节点部分中的这个条目，列出了将一组行读入内存后计算的谓词。驻留谓词通常包含复杂运算，例如子查询或用户定义的函数，所以不能象扫描谓词或发布扫描谓词那样计算。
- **考虑的索引** 此条目位于 [高级详细信息] 窗格中，在对由这个扫描运算符引用的表进行优化的过程中，它列出了优化程序考虑的所有索引或表扫描。列表中每一项的格式与为在 [详细信息] 窗格的访问计划中使用的扫描运算符列出的详细信息的格式类似。
- **主键表** 此条目位于索引扫描运算符的 [索引] 部分中，提供主键表的名称。
- **主键表预计行数** 此条目位于索引扫描运算符的 [索引] 部分中，提供主键表中的行数。
- **主键列** 此条目位于索引扫描运算符的 [索引] 部分中，提供主键列的名称。
- **顺序转换** 此条目位于索引扫描运算符的 [索引] 部分中，提供为每个物理索引保存的统计信息，指示该索引是如何聚簇的。
- **随机转换** 此条目位于索引扫描运算符的 [索引] 部分中，提供为每个物理索引保存的统计信息，指示该索引是如何聚簇的。
- **键值** 此条目位于索引扫描运算符的 [索引] 部分中，提供该索引中唯一条目的数量。

SQL Anywhere 行为更改

以下列出了 11.0.0 版中引入的对 SQL Anywhere 的行为更改（根据类别分组）。

● **对分类目录进行的更改** 在下表中包含了 11.0.0 版对目录的更改。

您必须升级数据库才能获得这些更改。请参见“[升级 SQL Anywhere](#)”一节第 324 页。

表名称和/或视图名称	对更改的说明	其它信息
ISYSTAB/SYSTAB	<ul style="list-style-type: none"> ○ 添加了一个新列 <code>dbspace_id</code>，作为对于当前 <code>file_id</code> 列的最后替换。 ○ 不建议使用 <code>file_id</code> 列。请改用 <code>dbspace_id</code>。对于全局临时表，<code>SYSTAB.file_id</code> 现在指向临时 <code>dbspace</code> 而不是系统 <code>dbspace</code>。 ○ 已添加了新列 <code>last_modified_tsn</code>，用于为修改该表的事务存储序列号。 	请参见“ SYSTAB 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSIDX/SYSIDX	<ul style="list-style-type: none"> ○ 添加了一个新列 <code>dbspace_id</code>，作为对于当前 <code>file_id</code> 列的最后替换。 ○ 不建议使用 <code>file_id</code> 列。请改用 <code>dbspace_id</code>。 	请参见“ SYSIDX 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSFILE	<p>不建议使用此系统表。现在，可在（新）<code>ISYSDBSPACE</code> 系统表中找到除 <code>lob_map</code> 外的所有列。现在，可在（新）<code>ISYSDBFILE</code> 系统表中找到 <code>lob_map</code> 列。</p>	<p>请参见：</p> <ul style="list-style-type: none"> ○ “SYSDBSPACE 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》 ○ “SYSDBFILE 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》 ○ “SYSFILE 兼容性视图（不建议使用）”一节《SQL Anywhere 服务器 - SQL 参考》
ISYSDBFILE/SYSDBFILE	用于保存有关 <code>dbspace</code> 信息的新表。	请参见“ SYSDBFILE 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSDBSPACE/SYSDBSPACE	用于保存有关 <code>dbspace</code> 信息的新表。	请参见“ SYSDBSPACE 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

表名称和/或视图名称	对更改的说明	其它信息
SYSDSPACEPERM/ ISYSDSPACEPERM	用于保存 dbspace 权限的新表。	请参见“ SYSDSPACEPERM 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSOBJECT/SYSOBJECT	已将 file_id 列重命名为 dbspace_id。此外，在 object_type 列中可以包含两个新值：17（文本配置）和 18 (Dbspace)。	请参见“ SYSOBJECT 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSINDEXES	在 indextype 字段中，现在将外键和主键索引分别标识为 [主键] 和 [外键]，以便与其它索引相区别。	请参见“ SYSINDEXES 统一视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSCAPABILITYNAME	此表不再存在于目录中。相应的 SYSCAPABILITYNAME 系统视图仍然可用，但通过使用服务器属性生成该视图。	请参见“ SYSCAPABILITYNAME 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSEVENTTYPE	此表不再存在于目录中。相应的 SYSEVENTTYPE 系统视图仍然可用，但通过使用服务器属性生成该视图。	请参见“ SYSEVENTTYPE 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSVIEW	名为 mv_last_refreshed_tsn 的一个新列，用于为刷新实例化视图的事务存储序列号。	请参见“ SYSVIEW 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSLOGINMAP/ SYSLOGINMAP	用于保存有关登录策略信息的新表。	请参见“ SYSLOGINMAP 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSLOGINPOLICY/ SYSLOGINPOLICY	用于保存有关登录策略信息的新表。	请参见“ SYSLOGINPOLICY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

表名称和/或视图名称	对更改的说明	其它信息
ISYSLOGINPOLICYOPTION/ SYSLOGINPOLICYOPTION	用于保存有关登录策略信息的新表。	请参见“ SYSLOGINPOLICYOPTION 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSTEXTCONFIG/ SYSTEXTCONFIG	用于保存有关文本配置对象信息的新表。	请参见“ SYSTEXTCONFIG 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSTEXTIDX/ SYSTEXTIDX	用于保存有关文本索引信息的新表。	请参见“ SYSTEXTIDX 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
ISYSTEXTIDXTAB/ SYSTEXTIDXTAB	用于保存有关文本索引信息的新表。	请参见“ SYSTEXTIDXTAB 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

- **对 PHP 函数名的更改** 已经重命名所有 PHP 函数，其前缀改为 `sasql_` 而不是 `sqlanywhere_`。调用函数时，仍然允许在名称中存在 `sqlanywhere_` 前缀，但不建议使用。您应该更改应用程序以使用新前缀。
- **INSERT ...ON EXISTING UPDATE 语句现在触发触发器** Previously, when you executed an INSERT ...ON EXISTING UPDATE statement, triggers did not fire if data was updated.现在，数据库服务器会为更新触发语句级后触发器。
- **REFRESH MATERIALIZED VIEW 语句** 不再能为刷新将 STATEMENT SNAPSHOT 和 READONLY STATEMENT SNAPSHOT 指定为隔离级别，因为这些选项的效果与为隔离级别指定 SNAPSHOT 的效果相同。请参见“[REFRESH MATERIALIZED VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **REORGANIZE TABLE 语句** 现在，试图在同一个表上同时执行多个 REORGANIZE TABLE 语句会导致错误。
- **sa_validate 系统过程** `sa_validate` 的快速参数和校验和参数现已过时；指定这些参数不再产生效果。现在，缺省情况下执行校验和校验。此外，如果调用 `sa_validate` 系统过程时未指定任何参数，数据库服务器除了校验所有表、实例化视图和索引外，还会校验数据库本身，其中包括校验和。请参见“[sa_validate 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **-gss 服务器选项** 现在，在 Windows XP 及其更高版本上支持 `-gss` 服务器选项。在以前的发行版本中，在 Windows 操作系统上不支持此选项。请参见“[-gss 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不再支持 -gx 服务器选项** 此发行版本中已经删除了对 `-gx` 服务器选项的支持。启动 SQL Anywhere 数据库服务器时指定 `-gx` 选项会导致错误。

- **LazyClose 连接参数的缺省设置现在为 AUTO** 在以前的版本中，应用程序关闭游标时，如果未将 LazyClose 连接参数设置为 YES，必需往返经过数据库服务器。现在，缺省情况下，可以为许多游标对游标关闭请求进行排队，这样就消除了往返，从而改进了性能。LazyClose 连接参数现在接受三个值：YES、NO 和 AUTO（缺省值）。在以前的发行版本中，YES 为缺省设置。请参见“[LazyClose 连接参数 \[LCLOSE\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **嵌入式 SQL 导入库的更改** 不再包括 Watcom 和 Borland 版本的 DBLIB 导入库。这些导入库分别是 *dblibtw.lib* 和 *dblibtb.lib*。提供了一个重要的定义文件 (*dblib11.def* file) 来替换这些导入库。
- **数据库工具导入库的更改** 不再包括 Watcom 和 Borland 版本的数据库工具导入库。这些导入库分别是 *dbtlstw.lib* 和 *dbtlstb.lib*。提供了一个重要的定义文件 (*dbtool11.def* file) 来替换这些导入库。
- **未接收到任何行时定义的 DBLIB 指示符行为** 如果在读取或执行操作中，未从数据库服务器接收到任何行（由于出错或者到达结果集的末尾），则指示符值现在不会更改。请参见“[指示符变量](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **ODBC SQLGetConnectAttr** 现在，通过 ODBC 的 SQLGetConnectAttr 调用来获取 SQL_ATTR_CONNECTION_DEAD 属性，如果连接已经删除，即使从连接删除时起未向服务器发出任何请求，也可以获得值 SQL_CD_TRUE。确定删除的连接是否是在未向服务器发出请求的情况下完成的，并且可在几秒钟内检测到删除的连接。有多个原因可删除连接，例如空闲超时。

在以前的版本中，如果连接断开或者连接删除后如果 ODBC 驱动程序向服务器发出请求（例如，通过调用 SQLExecDirect），则 SQL_ATTR_CONNECTION_DEAD 仅获得值 SQL_CD_TRUE。
- **不能创建也不能启动名为 utility_db 的数据库** 现在，为 SQL Anywhere 服务器实用程序数据库保留名称 utility_db。如果试图创建一个名为 *utility_db.db* 的新数据库或者试图启动一个具有该名称的现有数据库，将会返回错误。如果现有数据库的名称为 utility_db，可以使用不同的名称来启动该数据库。请参见“[使用实用程序数据库](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **计算列依赖性** 以前，为使更新或插入操作能够无错进行，应用程序可以使用触发器向声明为 NOT NULL 的列指派非 NULL 值。这会影响到那些依赖于该列的计算列，因为会导致某个计算值不能反映所需的计算。现在，如果试图在计算列所依赖的 NOT NULL 列设置一个 NULL 值，结果会失败并返回错误消息，并且不会触发任何触发器。请参见“[插入和更新计算列](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **包含句点的 dbspace 名称会生成错误** 在以前的版本中，如果某个 dbspace 名称包含一个句点并且未加上引号，则服务器会以静默方式忽略该 dbspace 名称中位于句点前面的部分。现在，数据库服务器会为此类名称生成一个错误。
- **SQL Anywhere Web 服务器不再支持 SSL 版本 2.0** 使用 SQL Anywhere Web 服务器时，仅支持 SSL 版本 3.0 和 TLS 版本 1.0 连接。不支持 SSL 版本 2.0 连接。
- **CREATE SERVICE 选项 DATATYPE 缺省值已更改** 已将 DATATYPE 子句的缺省值从 OFF 更改为 ON。如果需要原来的行为，则必须在 CREATE SERVICE 定义中显式地包含 DATATYPE OFF。请参见“[CREATE SERVICE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **已重命名某些受保护的功能** 此发行版本已重命名了以下受保护的功能：

不建议使用的名称	新名称
xp_read_file	read_file
xp_write_file	write_file
unload_table	write_file
load_table	read_file

有关详细信息，请参见“-sf 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **校验和行为更改** 对于使用版本 11 创建的数据库或者升级到版本 11 的数据库，数据库服务器会自动为在如网络驱动器或可移动驱动器的介质上运行的数据库启用校验和。只要数据库驻留在此类设备上，校验和就保持启用状态。请参见“使用校验和检测损坏”一节《SQL Anywhere 服务器 - 数据库管理》。
- **HTTP 连接不会导致自动停止** 在以前的版本中，如果将某个数据库配置为自动停止，当某个 HTTP 连接断开并且在该数据库上不存在其它连接时，数据库会停止。现在，仅在最后一个命令序列或 TDS 连接断开连接时，数据库才会自动停止。
 如果与数据库的唯一连接是 HTTP 连接而且该数据库配置为自动停止，则当该 HTTP 连接断开时，数据库不会自动停止。同样，如果配置为自动停止的数据库具有 HTTP 连接和命令序列或 TDS 连接，则当最后一个命令序列或 TDS 连接断开时，该数据库就会自动停止，并且所有 HTTP 连接都将断开。请参见“-ga 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》和“AutoStop 连接参数 [ASTOP]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **数据库镜像行为更改** 在以前的版本中，如果在 -xp 选项中为主服务器或镜像服务器指定的连接参数无效，则数据库服务器会反复尝试连接，但是该连接决不会成功。在本发行版本中，如果在 -xp 选项中指定的连接参数无效，并且在服务器上运行多个数据库，则镜像的数据库启动失败并且不尝试重新连接。如果镜像数据库是在数据库服务器上运行的唯一数据库，则数据库服务器不会启动。
- **实例化视图的缺省刷新行为** 以前，实例化视图的缺省刷新行为采用 WITH EXCLUSIVE MODE。现在，缺省刷新行为取决于是否将实例化视图定义为 IMMEDIATE REFRESH，以及是否为数据库启用快照隔离级别。请参见“REFRESH MATERIALIZED VIEW 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **post_login_procedure 数据库选项行为更改** 现在，post_login_procedure 数据库选项的缺省设置为 sa_post_login_procedure 系统过程。请参见“post_login_procedure 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **non_keywords 数据库选项** 在以前的版本中，除了指定各个关键字之外，您还可以通过在关键字列表中使用下列特殊值来关闭特定发行版本后的所有关键字：

```
keywords_4_0_d, keywords_4_0_c, keywords_4_0_b, keywords_4_0_a,
keywords_4_0,
keywords_5_0_01, keywords_5_0
```

不再支持这些特殊值。但仍然可以关闭各个关键字。请参见“non_keywords 选项 [兼容性]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **与远程数据访问相关的 quoted_identifier 数据库选项设置** 现在，当使用远程数据访问时，quoted_identifier 选项的本地设置控制是否对 Adaptive Server Enterprise 和 Microsoft SQL Server 使用加引号的标识符。例如，如果在本地将 quoted_identifiers 选项设置为 [关闭]，则会为 Adaptive Server Enterprise 关闭加引号的标识符。请参见：
 - “服务器类 aseodbc” 一节 《SQL Anywhere 服务器 - SQL 的用法》
 - “服务器类 mssodbc” 一节 《SQL Anywhere 服务器 - SQL 的用法》
- **对于 precision 和 scale 数据库选项范围的更改** 在以前的发行版本中，可以为各位用户设置 precision 和 scale 数据库选项或者指定该设置具有临时范围。但是，这些设置会影响到数据库的可恢复性。如果在执行创建或变更表和域的 DDL 语句时，临时设置或用户级设置与相应的 PUBLIC 设置不同，则可能会在重建数据库时遇到问题。以下行为现在应用于 precision 和 scale 数据库选项：

数据库服务器版本	版本 10 或更早版本的数据库	版本 11 的数据库	升级到版本 11 的数据库	卸载版本 10 或更早版本的数据库
11	允许 PUBLIC 设置 允许用户设置 不允许临时设置	允许 PUBLIC 设置 不允许用户设置 不允许临时设置	允许 PUBLIC 设置 不允许用户设置 不允许临时设置	卸载 PUBLIC 设置 卸载时放弃用户设置
版本 10 或更早版本	允许 PUBLIC 设置 允许用户设置 允许临时设置	N/A	N/A	卸载 PUBLIC 设置 卸载用户设置

版本 10 和更早版本的数据库服务器仍然允许临时设置 scale 和 precision 选项，也允许为各位用户进行此类设置。

小心

建议对于 precision 和 scale 数据库选项不要依赖用户级或临时设置，因为会在重建数据库期间遇到潜在问题，而且可能会发生不可预知的数据库服务器行为。

请参见：

- “precision 选项 [数据库]” 一节 《SQL Anywhere 服务器 - 数据库管理》
- “scale 选项 [数据库]” 一节 《SQL Anywhere 服务器 - 数据库管理》

- **OPTION 子句行为更改** 现在，如果指定一个 INSERT、UPDATE、DELETE、SELECT、UNION、EXCEPT 和 INTERSECT 语句中的 OPTION 子句不支持的数据库选项，该子句会返回错误。请参见：
 - “INSERT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “UPDATE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “DELETE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “SELECT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “UNION 子句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “EXCEPT 子句”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “INTERSECT 子句”一节 《SQL Anywhere 服务器 - SQL 参考》

- **只读数据库的回退日志行为更改** 在以前的发行版本中，不针对涉及事务性临时对象的只读数据库的操作视为事务性的：不为它们保留回退日志信息。在本发行版本中，在只读数据库中的事务性临时对象具有完全的事务性语义。它们会被提交、回退以及回退到保存点。

- **支持 Itanium 64 位的平台更改** 在以前的版本中，该软件有面向基于 Itanium II 芯片的 Windows Server 2003 的完整 64 位版本，还有面向 64 位 Linux 和 HP-UX 操作系统的部署发行版本。

在本发行版本中，仅有面向 64 位 HP-UX 的部署发行版本。

- **卸载实用程序 (dbunload) 行为更改** 在以前的发行版本中，必须使用 -an 或 -ar 选项指定 dbunload -ea、-ek 和 -ep 选项，以便控制新数据库的加密。现在，如果卸载数据库或者数据库的任何一部分并且不再进行重装，则 -ea、-ek 和 -ep 选项会控制所创建的表数据文件的加密。使用这些文件从 Interactive SQL 重装数据库时，必须将加密密钥指定为 READ 语句的参数。请参见“卸载实用程序 (dbunload)”一节 《SQL Anywhere 服务器 - 数据库管理》。

还有，在以前的发行版本中，dbunload 用于抽取数据库的版本不必与运行数据库的数据库服务器的版本相同。现在，在将 dbunload 用于版本 10.0.0 或更高版本的数据库时，所使用的 dbunload 的版本必须与用于访问数据库的数据库服务器的版本相匹配。如果将较旧版本的 dbunload 和相对较新版本的数据库服务器一起使用，会报告出现错误，反之亦然。

- **抽取实用程序 (dbxtract) 行为更改** 在以前的发行版本中，dbxtract 用于抽取数据库的版本不必与运行数据库的数据库服务器的版本相同。现在，在将 dbxtract 用于版本 10.0.0 或更高版本的数据库时，所使用的 dbxtract 的版本必须与用于访问数据库的数据库服务器的版本相匹配。如果将较旧版本的 dbxtract 和相对较新版本的数据库服务器一起使用，会报告出现错误，反之亦然。

- **锁定行为中的更改** 在以前的发行版本中，在隔离级别 0 执行的 UPDATE 或 DELETE 语句会在不受该语句影响的行的行锁处阻塞。现在，UPDATE 或 DELETE 语句对某个不受其影响的行记录目标或进行排它锁定的可能性更小。开发应用程序时，如果将隔离级别 0 或 1 与 UPDATE 和 DELETE 语句一起使用时要谨慎，并且要确保应用程序可以接受该行为。请参见“更新过程中的锁定”一节 《SQL Anywhere 服务器 - SQL 的用法》和“删除过程中的锁定”一节 《SQL Anywhere 服务器 - SQL 的用法》。

- **对属性名称的更改** 本发行版本中重命名了以下属性：

旧名称	新名称
CacheHitsEng	CacheHits

旧名称	新名称
CacheReadEng	CacheRead
DiskReadEng	DiskRead
ReadHint	DiskReadHint
ReadHintScatter	DiskReadHintPages
ReadHintScatterLimit	DiskReadHintScatterLimit

有关详细信息，请参见“[连接属性、数据库属性和数据库服务器属性](#)”《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **语言选择实用程序 (dblang)** 在以前的版本中，只有在安装过程中选择了国际资源开发工具包 (International Resources Development Kit, 简称 IRDK) 时，才会安装此实用程序。而在该版本中，会始终安装所有国际资源和语言选择实用程序 (dblang)。
- **临时表及索引的缺省 dbspace** 临时表只能在 TEMPORARY dbspace 中创建。如果在 CREATE TABLE 语句的 IN 子句中指定 SYSTEM dbspace，则会忽略 IN 子句，而且会在临时 dbspace 中创建临时表。如果在 CREATE TABLE 语句的 IN 子句中指定用户定义 dbspace，则会返回错误。同样，在创建临时对象时，会忽略 default_dbpace 选项。
- **向临时表中装载数据** 将数据装载到临时表中时，可以不再装载处于 ON COMMIT DELETE 的本地临时表。在以前的版本中，可以将数据装载到使用 ON COMMIT DELETE ROWS 定义的本地临时表中。
现在，当运行 LOAD TABLE 语句时，会自动执行自动提交；在以前的版本中，并不总是这样。
- **数据库服务器选项** 现在，Mac OS X 支持服务器选项 -uc 和 -ui。以前，它们仅在 Linux 上受支持。在 Linux 上，-ui 服务器选项用于打开 [\[服务器启动选项\]](#) 窗口，显示数据库服务器消息窗口并启动数据库服务器（无论 X 窗口服务器是否启动）。如果在 Mac OS X 上没有可用的显示，-ui 会在新窗口内显示数据库服务器消息并以 shell 模式启动数据库服务器。-uc 服务器选项用于在 shell 模式下启动数据库服务器。请参见“[-uc 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[-um 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **远程数据访问不再使用不支持 UNICODE 调用的 ODBC 驱动程序** 远程数据访问不再使用不支持 UNICODE 调用的 ODBC 驱动程序。结果是，对于非 UNICODE ODBC 驱动程序，远程数据访问不再对来自 ODBC 驱动程序的数据执行任何字符集转换。
- **SYSFILE 系统视图** 现在 SYSFILE 兼容性视图中包括了临时文件的一行。请参见“[SYSFILE 兼容性视图（不建议使用）](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

不建议使用和停用的 SQL Anywhere 功能

- **对 LOAD TABLE、UNLOAD TABLE、INPUT 和 OUTPUT 语句不建议使用 FORMAT ASCII 子句** 对 LOAD TABLE、UNLOAD TABLE、INPUT 和 OUTPUT 语句不建议使用 FORMAT

ASCII 子句，已经用 FORMAT TEXT 替换了该子句。现在，实用程序（例如 dbunload）可以生成包含 FORMAT TEXT 而不是 FORMAT ASCII 的重装脚本。

对于 OUTPUT 语句，FORMAT TEXT 子句现在采用与 FORMAT ASCII 在先前版本中所用的相同文件格式写入数据。以前由 FORMAT TEXT 创建的输出不再可用。

● **数据库属性** 不支持以下数据库属性：

- MapPages
- PreserveSource
- UniqueIdentifier

● **服务器属性** 本发行版本中不建议使用以下服务器属性：

- MaxMessage
- Message
- MessageTime
- MessageText
- MessageWindowSize

● **不支持 SPX 协议** 本发行版本中已经删除了对 SPX 协议的支持。因此，停用以下协议选项：

- ExtendedName 协议选项 [ENAME]
- RegisterBindery 协议选项 [REGBIN]
- SearchBindery 协议选项 [BINSEARCH]

停用 SQL Anywhere .NET 数据提供程序的以下功能：

- SACommLinksOptionsBuilder 类：SpxOptionsBuilder 属性
- SACommLinksOptionsBuilder 类：SpxOptionsString 属性
- SASpxOptionsBuilder 类

● **不支持 dbinit -e 选项** 不再支持用于在创建数据库时指定简单加密的 dbinit -e 选项。使用 -ea simple 选项指定简单加密。请参见“[初始化实用程序 \(dbinit\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

● **停用的数据库选项** 在本发行版本中删除了对以下数据库选项及其相应数据库属性的支持。

选项	本发行版本中的行为
ansi_integer_overflow	现在，溢出始终会导致 SQLSTATE = 22003 - overflow 错误。 卸载或连接到包含实例化视图的较旧数据库时，会忽略此选项的设置。
ansi_substring	现在，SUBSTRING 函数的行为相当于 ANSI/ISO SQL/2003 行为。请参见“ SUBSTRING 函数 [String] ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

选项	本发行版本中的行为
automatic_timestamp	绝不会将 Transact-SQL 时间戳的缺省值赋给没有定义显式缺省值且数据类型为 TIMESTAMP 的新列。
divide_by_zero_error	现在，除以零会导致错误 SQLSTATE 22012。 卸载或连接到包含实例化视图的较旧数据库时，会忽略此选项的设置。
float_as_double	在 SQL Anywhere 中，FLOAT 关键字的行为绝不会像 Adaptive Server Enterprise 的 FLOAT 关键字没有指定精度时一样。SQL Anywhere 使用 FLOAT 值的方式与使用 DOUBLE 值的方式不同。 对于 Open Client 和 jConnect 连接，此行为与以前发行版本中的缺省行为不同。 卸载或连接到包含实例化视图的较旧数据库时，会忽略此选项的设置。
optimistic_wait_for_commit	不再支持此选项。
query_plan_on_open	对游标完成 OPEN 后，不再返回一个计划。可以使用 EXPLAIN 语句或 PLAN 函数获得更完整的说明。请参见“EXPLAIN 语句 [ESQL]”一节《SQL Anywhere 服务器 - SQL 参考》和“PLAN 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》。
ri_trigger_time	现在，在 UPDATE 或 DELETE 之后执行参照完整性操作。
truncate_with_auto_commit	现在，在执行 TRUNCATE TABLE 语句之前和之后都执行 COMMIT。
tsql_hex_constant	现在，将十六进制常量视为二进制类型的常量。
uuid_has_hyphens	现在，UUID 字符串包含四个连字符。 卸载或连接到包含实例化视图的较旧数据库时，会忽略此选项的设置。
percent_as_comment	在以前的发行版本中，取决于 percent_as_comment 数据库选项的设置，可将百分号 (%) 用作注释标记。现在，SQL Anywhere 将百分号 % 视为模运算符。请参见“MOD 函数 [Numeric]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **不支持 SQLANYSH10 环境变量** 在以前的发行版本中，某些 SQL Anywhere 软件安装在某个共享目录中。可以通过 SQLANYSH10 环境变量指定此位置。现在，不再将软件安装到某个共享目录作为安装过程的一部分，并且不再使用 SQLANYSH10 环境变量。

创建静默安装时，不再需要设置 SHARED_DIR 位置。请参见“使用静默安装进行部署”一节《SQL Anywhere 服务器 - 编程》。

- **sa_get_server_messages 系统过程已停用** 在以前的发行版本中，可以使用 sa_get_server_messages 系统过程以结果集的形式从数据库服务器消息窗口返回常量。现在，可以使用 sa_server_messages 系统过程获取此信息。请参见“sa_server_messages 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **不建议使用 background_priority 选项** 已经不建议使用 background_priority 选项。改用 priority 选项。请参见“priority 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **不支持 encrypt_aes_random_iv 选项** 本发行版本中已经删除了对 encrypt_aes_random_iv 数据库选项的支持。现在，始终使用随机 IV（初始化矢量）。
- **不支持 DLL 协议选项** 删除了对 DLL 协议选项的支持。Windows 数据库服务器和客户端使用 Winsock 2.2。Windows Mobile 客户端使用 Winsock 1.1。请参见“使用 TCP/IP 协议”一节《SQL Anywhere 服务器 - 数据库管理》。
- **重命名了 SQL Anywhere 广播转发器实用程序** 在版本 10 中，用于运行 SQL Anywhere 广播转发器实用程序的命令为 dbns10。在本发行版本中，该命令为 dbns11。请参见“广播转发器实用程序 (dbns11)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **重命名了 SQLANY10 和 SQLANYSAMP10 环境变量** SQLANY10 和 SQLANYSAMP10 环境变量被分别重命名为 SQLANY11 和 SQLANYSAMP11。请参见：
 - “SQLANY11 环境变量”一节《SQL Anywhere 服务器 - 数据库管理》
 - “SQLANYSAMP11 环境变量”一节《SQL Anywhere 服务器 - 数据库管理》

MobiLink

以下各节介绍 11.0.0 版的 MobiLink 中的新功能、行为更改和不建议使用的功能。

MobiLink 的新功能

以下列出了 11.0.0 版中引入的 MobiLink 新增功能。

统一数据库

- **现在支持 DB2 主机作为统一数据库** 长期以来，MobiLink 都支持 DB2 LUW（Linux、Unix 和 Windows）作为统一数据库。现在，它还支持 DB2 主机。
请参见“IBM DB2 主机统一数据库”一节《MobiLink - 服务器管理》。
- **现在将 MySQL 作为统一数据库来支持** MobiLink 现在将 MySQL 作为统一数据库来支持。
请参见“MySQL 统一数据库”一节《MobiLink - 服务器管理》。
- **MobiLink 系统数据库 (MLSD)** 添加了对保存 MobiLink 系统数据的单独数据库（MLSD - MobiLink 系统数据库）的支持。该功能必须与 Microsoft DTC（分布式事务处理协调器）一起使用。请参见“-cs 选项”一节《MobiLink - 服务器管理》。

新的系统对象

- **新的 MobiLink 服务器系统表和模式** 对 MobiLink 系统表进行了以下更改：
 - 添加了多个新的 MobiLink 系统表。请参见：
 - “ml_qa_delivery_archive”一节《MobiLink - 服务器管理》
 - “ml_qa_repository_archive”一节《MobiLink - 服务器管理》
 - “ml_qa_repository_props_archive”一节《MobiLink - 服务器管理》
 - “ml_qa_status_history_archive”一节《MobiLink - 服务器管理》
 - “ml_server”一节《MobiLink - 服务器管理》
 - “ml_active_remote_id”一节《MobiLink - 服务器管理》
 - “ml_passthrough”一节《MobiLink - 服务器管理》
 - “ml_passthrough_repair”一节《MobiLink - 服务器管理》
 - “ml_passthrough_script”一节《MobiLink - 服务器管理》
 - “ml_passthrough_status”一节《MobiLink - 服务器管理》

○ 添加了多个新的 MobiLink 系统过程。请参见：

- “ml_add_passthrough 系统过程”一节 《MobiLink - 服务器管理》
- “ml_add_passthrough_repair 系统过程”一节 《MobiLink - 服务器管理》
- “ml_add_passthrough_script 系统过程”一节 《MobiLink - 服务器管理》
- “ml_delete_passthrough 系统过程”一节 《MobiLink - 服务器管理》
- “ml_delete_passthrough_repair 系统过程”一节 《MobiLink - 服务器管理》
- “ml_delete_passthrough_script 系统过程”一节 《MobiLink - 服务器管理》

iAnywhere Solutions Oracle 驱动程序

- **Oracle DSN 可以存储加密口令** 现在，在 Windows [ODBC 管理器] 中创建 Oracle ODBC DSN 时，可以选择将存储在 DSN 中的口令加密。请参见 “iAnywhere Solutions Oracle 驱动程序”一节 《MobiLink - 服务器管理》。
- **Oracle ODBC 驱动程序支持 Microsoft 分布式事务** 现在，Oracle ODBC 驱动程序支持 Microsoft 分布式事务。在 Windows [ODBC 管理器] 中选择 [启用 Microsoft 分布式事务]，并且要确保已经与 Oracle 客户端一起安装了相应的 DLL。请参见 “iAnywhere Solutions Oracle 驱动程序”一节 《MobiLink - 服务器管理》。

MobiLink 服务器

- **中继服务器** 中继服务器是一组 Web 扩展，它使得移动设备和 MobiLink、Afaria 与 OneBridge 服务器之间通过 Web 服务器的通信安全且负载平衡。请参见 “中继服务器简介”一节 《MobiLink - 服务器管理》。
- **Sybase 中继服务器托管服务** Sybase 中继服务器托管服务是一个由 Sybase 托管的中继服务器群。该服务旨在便于开发使用 MobiLink 数据同步的移动应用程序，以及简化开发人员的评估过程，特别是在使用公共无线网络发送数据的情况下。请参见 “Sybase 中继服务器托管服务”一节 《MobiLink - 服务器管理》。
- **MobiLink 服务器群** 现在，可以将 MobiLink 服务器显式分组为相同服务器的服务器群。
现在，可以通过整个群自动检测来自相同远程 ID 的冗余和并发同步。这样，负载平衡器可以无需不断地向同一 MobiLink 服务器发送相同的远程 ID。
现在，您可以对每个 MobiLink 服务器进行相同的配置。
通告程序和 QAnywhere 连接器自动支持故障转移。群会自动指定一个 MobiLink 服务器来运行这些功能，如果所指定的第一个计算机出现故障，群还会选择一个新服务器来运行这些功能。
- **64 位平台** 现在，MobiLink 服务器在许多 64 位平台上都为完全 64 位。有关支持的 64 位平台的列表，请参见 <http://www.sybase.com/detail?id=1062617>。
- **Java DownloadTableData 接口的新成员** 返回表的上次下载时间的 getLastDownloadTime 方法。请参见 “getLastDownloadTime 方法”一节 《MobiLink - 服务器管理》。
- **SQL 直通** 通过 SQL 直通功能，可将 SQL 语句的脚本从统一数据库下载到 SQL Anywhere 或 UltraLite 客户端，并且可于适当的时间在该客户端上执行这些 SQL 语句。请参见 “SQL 直通简介”一节 《MobiLink - 客户端管理》。

- **信息消息监听** 已添加 Java 和 .NET API 方法，注册这些方法后，只要有信息行输出到日志，就可以收到通知。

请参见：

- “INFO 变量”一节 《MobiLink - 服务器管理》
- “addInfoListener 方法”一节 《MobiLink - 服务器管理》
- “removeInfoListener 方法”一节 《MobiLink - 服务器管理》
- “INFO 字段”一节 《MobiLink - 服务器管理》
- “LogCallback InfoListener 事件”一节 《MobiLink - 服务器管理》

新的 mlsrv11 功能

- **-cs 选项** 现在，可将 MobiLink 服务器系统对象（例如系统表、过程、触发器和视图）存储在除统一数据库外的数据库中。将存储 MobiLink 系统对象的数据库称为 MobiLink 系统数据库或 MLSD。

使用 -cs 为 MLSD 指定连接参数。请参见 “-cs 选项”一节 《MobiLink - 服务器管理》。

- **-lsc 选项** 指定本地服务器连接信息。会将此信息传递给服务器群中的其它服务器。请参见 “-lsc 选项”一节 《MobiLink - 服务器管理》。
- **-ss 选项** 使 MobiLink 服务器能够在服务器群中运行。请参见 “-ss 选项”一节 《MobiLink - 服务器管理》。

需要单独授予许可的组成部分

-ss 选项是 MobiLink 高可用性选项的一个功能，它需要单独的许可证。请参见 “单独授权的组件”一节 《SQL Anywhere 11 - 简介》。

- **-tc 选项** 为 SQL 脚本的执行设置倒数计时器。请参见 “-tc 选项”一节 《MobiLink - 服务器管理》。
- **-tf 选项** 当通过 -tc 选项指定的倒数计时器到期时，SQL 脚本执行失败（不用于 Oracle）。请参见 “-tf 选项”一节 《MobiLink - 服务器管理》。

新的 MobiLink 脚本编写功能

- **非阻塞下载 ACK 脚本** 请参见 “nonblocking_download_ack 连接事件”一节 《MobiLink - 服务器管理》和 “publication_nonblocking_download_ack 连接事件”一节 《MobiLink - 服务器管理》。

MobiLink 重定向器增强

- **不建议使用重定向器** 不建议使用重定向器。已经用中继服务器将其替换。请参见 “中继服务器” 《MobiLink - 服务器管理》。
- **新的中继服务器** 中继服务器是一组 Web 扩展，它使得移动设备和 MobiLink、Afaría 与 OneBridge 服务器之间通过 Web 服务器的通信安全且负载平衡。请参见 “中继服务器简介”一节 《MobiLink - 服务器管理》。

MobiLink 客户端

SQL Anywhere 客户端

- **用于 C++ 和 .NET 的 dbmsync API** Dbmsync API 提供了一个编程接口，允许使用 C++ 或 .NET 来编写 MobiLink 客户端应用程序，以启动同步并接收有关这些应用程序所请求的同步进度的反馈。请参见“[Dbmsync API](#)”《[MobiLink - 客户端管理](#)》。
- **同步配置文件** 现在，可将 dbmsync 命令行作为同步配置文件存储在数据库中。-sp dbmsync 选项使您能够将选项从同步配置文件添加到命令行同步选项。请参见“[-sp 选项](#)”一节《[MobiLink - 客户端管理](#)》。
- **LOAD TABLE 现在可选择将已装载的行记录为插入** 新子句 WITH ROW LOGGING 表示 SQL Anywhere 远程数据库现在可以使用 LOAD TABLE 语句来装载数据。以前，LOAD TABLE 在同步环境中并非总是有用，因为不记录行，所以 dbmsync 会忽略那些行。请参见“[使用 LOAD TABLE 语句导入数据](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **dbmsync 日志扫描优化** 现在，为非重叠发布而优化。

安全性

在本发行版本中添加了以下安全功能：

- **端对端加密** 目前，MobiLink 同步流和客户端支持协议级端对端加密。请参见“[端对端加密](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **密钥对生成器实用程序 (createkey)** 此实用程序创建与 MobiLink 端对端加密一起使用的 RSA 和 ECC 密钥对。请参见“[密钥对生成器实用程序 \(createkey\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

服务器启动的同步

- **支持 MobiLink 服务器群** 已将 SIS 增强为可在 MobiLink 服务器群环境中更好地运行。现在，可以在群中的每个 MobiLink 服务器上运行通告程序，并且通告程序共同确保不会向同一个监听器发出重复的通知。请参见“[-lsc 选项](#)”一节《[MobiLink - 服务器管理](#)》。

监听器增强

- **新 dblsn 选项** 已向 Windows 的 MobiLink 监听器添加了以下选项：
 - **-ni** 禁用 IP 跟踪。
 - **-pc-** 禁用持久连接。
 - **-nu** 禁用缺省 UDP 监听。

MobiLink 的行为更改和不建议使用的功能

以下列出了 11.0.0 版中引入的 MobiLink 更改。

MobiLink 服务器更改

- **对缺省下载确认的更改** 现在，-nba 选项的缺省值为 -nba+，这就使得非阻塞下载确认成为缺省设置。如果要在现有的部署中使用阻塞下载确认的下载，则必须执行以下两项之一：
 - 使用 -nba- 选项。
 - 查看同步脚本以使用 nonblocking_download_ack 和/或 publication_nonblocking_download_ack 脚本（建议使用）。
- **已重命名 certificate 和 certificate_password 协议选项** 已将 TLS 和 HTTPS 的 certificate 和 certificate_password 协议选项分别重命名为 identity 和 identity_password。请参见 MobiLink 服务器：“-x 选项”一节《MobiLink - 服务器管理》。

MobiLink 客户端更改

- **不建议使用 Dbmlsync 集成组件** 不建议使用 Dbmlsync 集成组件。已经用 dbmlsync API 将其替换。
请参见“Dbmlsync API”《MobiLink - 客户端管理》。
- **不再支持 dbmlsync StreamCompression 扩展选项** 不再支持此选项。
- **现在 -lt 扩展选项的缺省值为 OFF** 以前，dbmlsync 的 LockTables (-lt) 扩展选项的缺省值为 ON。现在其缺省值为 OFF，表示在缺省情况下，dbmlsync 不锁定同步表。请参见“LockTables (lt) 扩展选项”一节《MobiLink - 客户端管理》。

MobiLink 的行为更改杂项

服务器启动的同步

- **不再支持 Sierra Wireless Aircard** 不再支持 Sierra Wireless Aircard 的 SMS 监听库。
- **不建议使用 -g 选项** -g 监听器选项已由 -ni 选项替代。
- **不建议使用 -ga 选项** 已不建议使用 -ga 监听器选项。异步 IP 跟踪现在为隐式。
- **已更改 -gi 缺省值** 已将缺省轮询间隔从 10 秒更改为 60 秒。

QAnywhere

以下各节介绍 11.0.0 版的 QAnywhere 中的新功能、行为更改和不建议使用的功能。

QAnywhere 的新功能

以下列出了 11.0.0 版中引入的 QAnywhere 新增功能。

新 QAnywhere 客户端 API

- **.NET API** 添加了以下 .NET API:
 - CreateQAManager 方法
 - CreateQAManager(Hashtable String)
 - CreateQATransactionalManager()
 - CreateQATransactionalManager(Hashtable String)
- **Java QAnywhere API 包括一个新的接口类** 添加了一个新类。请参见“[QAMessageListener2 接口](#)”一节《QAnywhere》。
- **C# QAnywhere API 包括新的委派**
 - “[ExceptionListener 委派](#)”一节《QAnywhere》
 - “[ExceptionListener2 委派](#)”一节《QAnywhere》
 - “[MessageListener2 委派](#)”一节《QAnywhere》

QAnywhere 代理的新功能

- **UltraLite 用作消息存储库** 现在，QAnywhere 支持将 UltraLite 用作消息存储库。请参见“[qauagent 实用程序](#)”一节《QAnywhere》。
- **qastop -id 选项** 通过此选项可将消息存储库 ID 提供给 qastop，以便使其停止 QAnywhere 代理，该代理是通过给定的存储库 ID 连接到数据库的。
- **新的 qauagent 实用程序** UltraLite 的 QAnywhere 代理。请参见“[qauagent 实用程序](#)”一节《QAnywhere》。

其它 QAnywhere 功能增强

- **UltraLite 用作消息存储库** 现在，QAnywhere 支持将 UltraLite 用作消息存储库。请参见“[qauagent 实用程序](#)”一节《QAnywhere》。
- **对增量上载和下载的增强** 增量上载和下载可将大消息拆分为较小的消息段。请参见“[-iu 选项](#)”一节《QAnywhere》和“[-idl 选项](#)”一节《QAnywhere》。
- **将数据库类型指定为配置属性** 现在，可将客户端消息存储库的数据库类型指定为 QAnywhere Manager 配置属性。请参见“[QAnywhere Manager 配置属性](#)”一节《QAnywhere》。
- **只要到达最终状态就将消息归档** 目前，提供了一个档案消息存储库，其唯一的用途在于存储那些已达到最终状态并等待永久删除的消息。请参见“[档案消息存储库请求](#)”一节《QAnywhere》。

- **服务器删除规则** 现在，服务器删除规则应用于档案消息存储库。
- **对服务器管理请求的增强** 已将服务器管理请求扩展为包含一个 <actionsResponseId> 标记，可选择将此标记包含在一个 <actions> 标记中。

QAnywhere 的行为更改和不建议使用的功能

以下列出了 11.0.0 版中引入的 QAnywhere 更改。

QAnywhere 代理更改

- **qastop 实用程序** 如果未向 qastop 提供选项，则在同一计算机上运行的所有 QAnywhere 代理都会停止。
- **为 Windows Mobile 保留的堆栈大小** 对于 Windows Mobile，为在 *qaagent.exe*、*dblsn.exe* 和 *dbmlsync.exe* 中的所有线程保留的堆栈大小已减小。

其它 QAnywhere 更改

- **非阻塞下载确认** 现在，继同步后会立即在服务器中将成功下载的消息的状态从正在传输更新为已传输。

SQL Remote

以下各节介绍了 11.0.0 版的 SQL Remote 中的新功能、行为更改和不建议使用的功能。

SQL Remote 的新功能

- **对抽取实用程序 (dbxtract) -ea 选项的增强** dbxtract 的 -ea 选项现在接受 none 和 simple 作为加密类型。指定 none 会导致不加密。指定 simple 会导致简单加密。同样，缺省加密类型也已更改，它取决于 -ek、-et 或 -ep 选项是否与 -ea 一同指定。请参见“抽取实用程序 (dbxtract)”一节《SQL Remote》。
- **已向抽取实用程序 (dbxtract) 添加了 -ap、-er、-et 和 -nl 选项** 现在，可与 dbxtract 一起使用 -ap、-er、-et 和 -nl 选项。请参见“抽取实用程序 (dbxtract)”一节《SQL Remote》。

SQL Remote 的行为更改和不建议使用的功能

以下列出了 11.0.0 版中引入的 SQL Remote 更改。

- **LOAD TABLE 现在可选择将已装载的行记录为插入** 新子句 WITH ROW LOGGING 表示 SQL Anywhere 远程数据库现在可以使用 LOAD TABLE 语句来装载数据。以前，LOAD TABLE 在同步或复制环境中并非总是有用，因为不记录行，所以 dbremote 或 dbltm 会忽略那些行。请参见“使用 LOAD TABLE 语句导入数据”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **不支持 VIM 和 MAPI 消息类型** 在本发行版本中已删除对 VIM 和 MAPI 消息系统的支持。将使用 VIM 或 MAPI 的数据库升级到 SQL Anywhere 版本 11.0.0 时，必须将消息类型更改成 File、FTP 或 SMTP。如果消息类型是 MAPI 或 VIM，则 *Dbremote.exe* 不启动。

最简单的更改方法是切换到 SMTP/POP；更改消息类型可能需要更改邮件服务器以使其支持 SMTP/POP。

有关选择 SQL Remote 消息类型的详细信息，请参见“SQL Remote 消息系统”一节《SQL Remote》。

UltraLite

以下各节介绍了 11.0.0 版的 UltraLite 中的新功能、行为更改和不建议使用的功能。

UltraLite 的新功能

以下列出了 11.0.0 版中引入的 UltraLite 新增功能。

主要功能

- **支持同步配置文件** UltraLite 现在支持同步配置文件。可以使用同步配置文件来定义 UltraLite 数据库如何与 MobiLink 同步。请参见“[UltraLite CREATE SYNCHRONIZATION PROFILE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite SELECT 语句** 不显式包含 FOR 子句的 UltraLite SELECT 语句的缺省设置现在为 FOR READ ONLY。此更改容许 UltraLite 在不允许更新时为查询选择更佳的计划。请参见“[UltraLite SELECT 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite SYNCHRONIZE 语句** 用于同步某个 UltraLite 同步配置文件或某些特定同步选项的新语句。请参见“[UltraLite SYNCHRONIZE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite CREATE SYNCHRONIZATION PROFILE 语句** 用于创建 UltraLite 同步配置文件的新语句。同步配置文件定义 UltraLite 数据库如何与 MobiLink 服务器同步。请参见“[UltraLite CREATE SYNCHRONIZATION PROFILE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite ALTER SYNCHRONIZATION PROFILE 语句** 用于变更 UltraLite 同步配置文件的新语句。请参见“[UltraLite ALTER SYNCHRONIZATION PROFILE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite DROP SYNCHRONIZATION PROFILE 语句** 用于删除 UltraLite 同步配置文件的新语句。请参见“[UltraLite DROP SYNCHRONIZATION PROFILE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **支持 SQL Anywhere 直通脚本** UltraLite 实用程序现在包括对 SQL Anywhere 直通脚本的支持。这些更改应用于以下实用程序：
 - ulcond11
 - ulunload
 - ulload
 - ulinfo
 - ulsync

请参见：

- “SQL 直通的 UltraLite 支持” 一节 《UltraLite - 数据库管理和参考》
- “Palm OS 的 UltraLite HotSync 管道安装实用程序 (ulcond11)” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 卸载旧数据库实用程序 (ulunload)” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 装载 XML 到数据库实用程序 (ulload)” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 信息实用程序 (ulinfo)” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 同步实用程序 (ulsync)” 一节 《UltraLite - 数据库管理和参考》

- **UltraLite 数据库校验** 现在，可以使用 ulvalid 实用程序或 ValidateDatabase API 来校验 UltraLite 数据库。此校验测试数据库文件中的某些损坏类型，您可以使用命令行参数来改进结果。请参见“UltraLite 校验数据库实用程序 (ulvalid)” 一节 《UltraLite - 数据库管理和参考》和“校验 UltraLite 数据库” 一节 《UltraLite - 数据库管理和参考》。

UltraLite.NET 现在支持 ValidateDatabase 函数。现在可以使用或不使用连接来校验数据库或特定表。请参见“ULDatabaseManager 类” 一节 《UltraLite - .NET 编程》和“ULConnection 类” 一节 《UltraLite - .NET 编程》。

现在，可以使用 Sybase Central 中的 [校验数据库向导] 来校验 UltraLite 数据库。[工具] 菜单上提供了 [校验数据库] 选项。

- **支持事件和通知** UltraLite 现在支持事件和通知。发生事件时，系统会将通知消息发送到注册队列或连接。也可以由应用程序来定义和触发用户事件。为每种受支持的语言提供了事件和通知的 API。此外，还提供了一个 SQL 函数以访问 API 功能。
- **UltraLite 支持隔离级别** 现在，缺省情况下，系统会将连接相互隔离。其它连接和下载的未提交更改，在提交之前不可见。

现在，可将隔离级别设置为 READ_COMMITTED 或 READ_UNCOMMITTED。请参见“UltraLite 隔离级别” 一节 《UltraLite - 数据库管理和参考》和“UltraLite 事务处理” 一节 《UltraLite - 数据库管理和参考》。

UltraLite.NET 现在支持 ReadUncommitted 隔离级别。在自动提交模式中，连接的缺省隔离级别为 ReadCommitted。请参见“UltraLite 事务处理” 一节 《UltraLite - 数据库管理和参考》和“UltraLite 隔离级别” 一节 《UltraLite - 数据库管理和参考》。

- **UltraLite ALTER DATABASE SCHEMA FROM FILE 语句** 现在，可以使用 ALTER DATABASE SCHEMA FROM FILE 语句来变更 UltraLite 模式。ALTER DATABASE SCHEMA FROM FILE 语句替换了 9.0.2 模式升级功能，该功能通过现在已被删除的 UpgradeSchemaFromFile 或 ApplyFile 方法实现。使用 ulinit 或 ulunload 两个实用程序之一来确保所需的 DDL 语句的语法正确。

请参见：

- “UltraLite ALTER DATABASE SCHEMA FROM FILE 语句” 一节 《UltraLite - 数据库管理和参考》
- “部署 UltraLite 模式升级” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 初始化数据库实用程序 (ulinit)” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 卸载旧数据库实用程序 (ulunload)” 一节 《UltraLite - 数据库管理和参考》

- **[抽取数据库向导] 行为更改** 现在，可从抽取进程排除某些表，并且 [抽取数据库向导] 现在会忽略可用发布列表中具有重复名称的发布。请参见“[升级版本 9 及更早版本的 UltraLite 数据库](#)”一节第 346 页。
- **UltraLite 客户端版本和内部版本号已添加到 MobiLink 日志文件中** 同步期间，UltraLite 客户端现在会将它们的版本和内部版本号添加到 MobiLink 服务器日志中。请参见：
 - “[查看 MobiLink 服务器日志](#)”一节 《[MobiLink - 服务器管理](#)》
 - “[UltraLite 的同步参数](#)”一节 《[UltraLite - 数据库管理和参考](#)》
- **UltraLite LOAD TABLE 语句** 现在，可在台式计算机上执行 LOAD TABLE 语句。请参见 UltraLite LOAD TABLE 语句。请参见“[UltraLite LOAD TABLE 语句](#)”一节 《[UltraLite - 数据库管理和参考](#)》。
- **后台同步支持** 现在，可在应用程序中的任何点于单独的线程上开始同步，UltraLite 将仅上载在上载开始时已提交的行。现在可以在上载期间修改数据库，然后提交更改而不会影响到上载。上载会忽略当上载正在进行时提交的任何行。请参见“[UltraLite 中的并发](#)”一节 《[UltraLite - 数据库管理和参考](#)》。
- **增强了对 GUID 标识符的支持** 在 UltraLite 的以前版本中，运行时允许以 16 字节二进制或字符串形式输入和输出 UUID（通用唯一标识符）或 GUID（全局唯一标识符）标识符。endian 转换使得这些标识符与 GUID 结构兼容。在 UltraLite 11 中，可从运行时将 GUID 结构显式地输入和输出而无需进行 endian 转换。
- **ul_stream_error 结构** 在 UltraLite 11 中，删除了 stream_id、stream_context 和 error_string_length 字段。此外，已将 error_string 字段从用户提供的 char * 更改为 static char array。请参见“[ul_stream_error 结构](#)”一节 《[UltraLite - C 及 C++ 编程](#)》。

平台和设备

- **支持对 64 位 Windows 平台（64 位 XP 及更高版本）的本地 amd64/x64 ESQL 和 C++ 应用程序部署** UltraLite 现在支持向 64 位 Windows 平台（64 位 XP 及更高版本）部署本地 amd64/x64 ESQL 和 C++ 应用程序。但是，请注意，要在 64 位计算机上开发 UltraLite 应用程序，必须使用 32 位版本的 UltraLite 实用程序。此外，如果在 64 位计算机上连接 UltraLite，那么还需要使用 32 位版本的 DBISQL 和 Sybase Central。
- **UltraLite 实用程序端口连接到 Linux（32 位）** 请参见“[实用程序](#)”一节第 70 页。

安全性

- **端对端加密** 目前，UltraLite 支持协议级端对端加密。在密码块链锁 (CBC) 模式下使用 128 位 AES 将数据加密，并且通过 RSA 或 ECC 来处理密钥交换。

实用程序

- **UltraLite 卸载数据库到 XML 实用程序 (ulunload)** 现在，可以使用 -s 选项来卸载模式，并且可以与 SQL Anywhere 兼容的格式保存数据。请参见“[UltraLite 卸载旧数据库实用程序 \(ulunload\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite 初始化数据库实用程序 (ulinit)** 现在，可以使用 -d 选项从 SQL Anywhere 数据库将数据复制到新的 UltraLite 数据库中。请参见“[UltraLite 初始化数据库实用程序 \(ulinit\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **ulerase** 新实用程序，用于消除 UltraLite 数据库（包括与该数据库相关的所有临时文件）。此实用程序需要用户 ID 和口令，以便确认对该数据库的访问权限。请参见“[UltraLite 消除数据库 \(ulerase\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **ulvalid** 新实用程序，用于校验 UltraLite 数据库。此校验测试数据库文件中的某些损坏类型，您可以根据命令行参数来进行配置。请参见“[UltraLite 校验数据库实用程序 \(ulvalid\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **已将以下 UltraLite 实用程序的端口连接到 Linux（仅限于 32 位版本使用）**
 - **ulcreate** 创建一个新的空 UltraLite 数据库
 - **ulerase** 永久消除 UltraLite 数据库及与其相关联的检查点文件
 - **ulinfo** 显示有关现有 UltraLite 数据库的信息
 - **ulinit** 从 SQL Anywhere 参考数据库中的可用模式创建一个新 UltraLite 数据库
 - **ulload** 从由 ulunload 保存的 XML 数据创建并装载一个 UltraLite 数据库
 - **ulsync** 将 UltraLite 数据库与统一数据库同步，使用 MobiLink 作为传输代理
 - **ulunload** 将 UltraLite 数据库卸载到 XML
 - **ulvalid** 对 UltraLite 数据库执行有效性检查

在 Linux 上 ulunloadold 实用程序不可用。

SQL

- **对 IF 和 CASE 语句以及 CASE 表达式的增强** 为改进兼容性，现在 IF 表达式可以使用 ENDIF 或 END IF 结束。现在 CASE 表达式可以使用 END 或 END CASE 结束。请参见“[IF 表达式](#)”一节《[UltraLite - 数据库管理和参考](#)》和“[CASE 表达式](#)”一节《[UltraLite - 数据库管理和参考](#)》。

编程接口

一般改进

已经用发布列表替换了发布掩码。关键字 Publications 将采用以逗号分隔的发布名列表。

UltraLite C/C++

新方法 `UltraLite_Table* OpenTableEx()` 现在是 `UltraLite_Connection` 对象的一部分。此方法给予了非 SQL 应用程序以更为通用的方式打开表和直接扫描行。请参见“[使用直接页扫描](#)”一节《[UltraLite - 数据库管理和参考](#)》。

采用此方法可以指定以下方式之一来打开表：

- 要返回主键行，可使用 `ul_table_open_primary_key`。
- 要以任意顺序返回行，可使用 `ul_table_open_no_index`。
- 要以索引指定的顺序返回行，可使用 `ul_table_open_with_index`。

有关此方法的完整语法和引用的信息，请参见“[OpenTableEx 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。

UltraLite 嵌入式 SQL

- 与错误解释相关的两个函数的文档。请参见“[ULGetErrorParameter 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》和“[ULGetErrorParameterCount 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
- 已经更改函数 `ULGetLastDownloadTime`、`ULResetLastDownloadTime` 和 `ULCountUploadRows` 的语法，以便反映从发布掩码到发布列表的变化。
- 函数 `ULGetPublicationMask` 不再可用。

UltraLite.NET

- `ULDataReader` 类中包含一个新方法：`GetRowCount(threshold)` 用于检索行计数直至指定的最大行数。
- `ULDataReader` 类现在实现了 `IListSource` 接口。

UltraLite for M-Business Anywhere

- `Connection` 类中用于事件处理和通知的新方法：`cancelGetNotification`、`createNotificationQueue`、`declareEvent`、`destroyNotificationQueue`、`getNotification`、`getNotificationParameter`、`registerForEvent`、`sendNotification` 和 `triggerEvent`。

UltraLiteJ

UltraLiteJ 是支持 Java SE 和 Java ME 环境（包括 BlackBerry 智能手机）的 UltraLite 的 Java 实现。请参见“[UltraLiteJ 简介](#)”《[UltraLiteJ](#)》。

UltraLite 的行为更改和不建议使用的功能

以下列出了 11.0.0 版中引入的 UltraLite 更改。

不建议使用的平台

- UltraLite C++ 接口不再支持 Symbian OS。Symbian 的 UltraLite 应用程序开发人员应使用 UltraLiteJ。请参见“[UltraLiteJ 简介](#)”《[UltraLiteJ](#)》。
- UltraLite .NET 接口不再支持 .NET 1.0 组件。已将 .NET 1.0 API 参考从文档中删除。

版本 11 不支持 UltraLite for Appforge。

数据库属性

本发行版本中不建议使用以下数据库属性：

- CollationName

连接参数

本发行版本中不建议使用以下 UltraLite 连接参数。

- ORDERED_TABLE_SCAN

删除的实用程序

[[迁移 C++ 应用程序向导](#)] 在 Sybase Central 中不再可用。

删除的、不建议使用的以及已修改的功能

- 已将 ESQL 接口中的 ULGetPublicationMask 函数删除（已用发布列表替换了发布掩码）。
- 已经更改了 ESQL 接口中函数 ULgetLastDownloadTime、ULResetLastDownloadTime 和 ULCountUploadRows 的语法，以便反映从发布掩码到发布列表的变化。
- 在 M-Business Anywhere API 中不建议使用以下各项：
DatabaseSchema.getTableCountInPublications、DatabaseSchema.SYNC_ALL_DB、
DatabaseSchema.SYNC_ALL_PUBS、PublicationSchema.getMask、
SyncParms.getPublicationMask 和 SyncParms.setPublicationMask。
- 在 M-Business Anywhere 和 .NET API 中，以下方法过去接受发布掩码作为参数。该参数已更改为发布列表（字符串）。受影响的方法在 Connection 类中：countUploadRows、getLastDownloadTime 和 resetLastDownloadTime。
- ul_sync_info 结构已更改：已经删除字段 **disable_concurrency**、**checkpoint_store** 和 **table_order**。现在，在名为 **additional_parms** 的新字段中指定这些选项，该字段包含以分号分隔的“关键字-值”对。

杂类

- **运行 CustDB 的第二个实例** 在以前的版本中，运行 CustDB 应用程序的第二个实例会导致错误。现在，如果启动 CustDB 的第二个实例，就会将第一个实例推到前台，而第二个实例会退出。

Sybase Central 和 Interactive SQL

以下各节介绍了 11.0.0 版的 Sybase Central 和 Interactive SQL 中的新功能、行为更改以及不建议使用的功能。

Sybase Central 和 Interactive SQL 的新功能

以下列出了 11.0.0 版中引入的 Sybase Central 和 Interactive SQL 新增功能。

- **新快速启动程序策略** 以前，用户登录时会启动 Interactive SQL 和 Sybase Central 的快速启动程序。现在，只有启动 Interactive SQL 或 Sybase Central 后，才能启动快速启动程序。然后，快速启动程序在应用程序关闭后仍将继续运行 30 分钟（可进行配置）。这一新策略可加快 Interactive SQL 和 Sybase Central 的后续启动，使其在 30 分钟的窗口内完成。请参见“[使用快速启动程序选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **对 [连接] 窗口的增强和更改** 在 Sybase Central 和 Interactive SQL 中对 [连接] 窗口进行了以下增强：
 - **新增 [另存为 ODBC 数据源] 工具** [另存为 ODBC 数据源] 工具可用于通过在 [连接] 窗口中指定的连接参数来生成 ODBC 数据源。要使用此工具，可在 [连接] 窗口中单击 [工具] 按钮，选择 [另存为 ODBC 数据源]，然后按照屏幕指令进行操作。请参见“[使用 \[连接\] 窗口创建 ODBC 数据源](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
 - **新增 [将连接字符串复制到剪贴板] 工具** [将连接字符串复制到剪贴板] 工具用于将连接字符串复制到剪贴板，以便查看正在向数据库服务器传递的内容。要使用此工具，可在 [连接] 窗口中单击 [工具] 按钮，然后选择 [将连接字符串复制到剪贴板]。在文本编辑器中粘贴连接字符串，以便进行检查。
 - **连接助手** 现在，在 [连接] 窗口的右半部分中有一个 [连接助手] 工具。[连接助手] 功能类似于向导，旨在帮助您连接到数据库。
 - **对 [高级] 选项卡的增强** [高级] 选项卡现在显示一个属性表以及对于属性本身的简要说明。
 - **新增 [Networking] 选项卡** [Networking] 选项卡用于为受支持的协议指定选项：共享内存和 TCP/IP。此新增选项卡替换了由以前位于 [数据库] 选项卡上的 [Search Network] 选项提供的功能。
- **键盘快捷方式** 现在，在 Sybase Central 的 [代码编辑器] 中以及在 Interactive SQL 的 [SQL 语句] 窗格中，可以使用键盘快捷方式来增加和减少代码和注释的缩进。此外，还可以使用键盘快捷方式来添加和删除双连字符及双斜线注释指示符。请参见“[Sybase Central 键盘快捷方式](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

Sybase Central 的新功能

以下列出了 11.0.0 版中引入的 Sybase Central 插件新增功能。

SQL Anywhere 插件新功能

- **数据库 [一览表] 选项卡** 现在，可以获得数据库服务器及其功能的健康和统计信息的概况。请参见“[监视数据库健康和统计信息](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **数据库文档** 现在，可以使用 [\[数据库文档向导\]](#) 为数据库生成文档。请参见“[记录数据库](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **新向导** SQL Anywhere 插件现在包含以下新向导：
 - 创建登录策略向导
 - 数据库文档向导
 - 创建文本索引向导
 - 创建文本配置对象向导
 - 创建数据库向导
- **新属性窗口** SQL Anywhere 插件现在包含以下功能的属性窗口：
 - 文本索引
 - 文本对象配置
 - 外部环境
 - 实例化视图
 - 数据库审计
- **在 Sybase Central 内设置审计首选项** 现在，您可以使用 Sybase Central 为数据库审计设置首选项以及查看审计信息。请参见“[控制审计](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[检索审计信息](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **支持指定归类定制设置** 在 [\[数据库属性\]](#) 窗口的 [\[设置\]](#) 选项卡上，现在可以指定归类定制选项。也可以在使用 [\[创建数据库向导\]](#) 新建数据库时，指定归类定制选项。

UltraLite 插件新功能

- **新向导** UltraLite 插件现在包含以下新向导：
 - 校验数据库向导
 - 创建同步配置文件向导
- **新属性窗口** UltraLite 插件现在包含以下功能的属性窗口：
 - 同步配置文件

MobiLink 插件新功能

- **支持同步配置文件** 可以在 Sybase Central 中创建和管理同步配置文件。
- **新向导** MobiLink 插件现在包含以下向导：
 - 创建直通脚本向导
 - 创建直通下载向导

- **新属性窗口** MobiLink 插件现在包含以下功能的属性窗口：
 - 直通脚本
 - 直通下载

QAnywhere 插件新功能

- **支持 UltraLite** 现在，QAnywhere 插件支持将 UltraLite 用作客户端消息存储库，并且支持新增的消息归档功能。请参见“[设置客户端消息存储库](#)”一节《[QAnywhere](#)》。

Interactive SQL 的新功能

以下列出了 11.0.0 版中引入的 Interactive SQL 新增功能。

- **自动释放数据库锁** 即使您不会在 Interactive SQL 中修改所查看的表，数据库服务器也会在这些表上创建模式锁。

但是，现在 Interactive SQL 会在显示结果集时尝试释放其创建的数据库模式锁。

在执行返回结果集的语句之后，Interactive SQL 将检查连接是否具有任何未提交的数据库更改。如果没有，则 Interactive SQL 将释放您的模式锁；否则，Interactive SQL 将不会释放您的模式锁。即，如果您有任何未提交的数据库更改，则 Interactive SQL 将不会释放模式锁。
- **支持只读结果集** 现在，可在 Interactive SQL 中将结果集设置为只读。为此，可选择 [选项] » [结果]，然后选择 [禁用编辑] 选项。此设置应用于随后获取的结果。请参见“[从 Interactive SQL 结果集中编辑表值](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

部署 Interactive SQL 时，为防止用户更改此设置，可将 disableResultSetEditing 添加到 OEM.ini 文件的 lockedPreferences 的条目中。请参见“[配置管理工具](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **执行 SQL 语句时，一次执行一条语句** 以前，如果需要一次执行一条 SQL 语句，需要重复选择 SQL 语句，并且运行 [执行所选内容]。现在，可以运行 [单步] 来执行所选的语句，然后选择下一条要执行的语句。与 [执行所选内容] 相似，在 Interactive SQL 的 [SQL] 菜单上提供 [单步] 选项。请参见“[从 Interactive SQL 执行 SQL 语句](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **SQL 文件和数据库连接收藏夹** 现在，可以通过 Interactive SQL 中的 [收藏夹] 菜单来创建及维护一个数据库连接的收藏夹列表和一个 SQL 命令文件的收藏夹列表。请参见“[使用收藏夹](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **禁用结果集的表编辑** 部署 Interactive SQL 时，现在可在 Interactive SQL 中禁用 SQL Anywhere 和 UltraLite 结果集的表编辑。请参见“[配置管理工具](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **新的键盘快捷方式** 为 Interactive SQL 添加了新的键盘快捷方式。请参见“[Interactive SQL 键盘快捷方式](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **支持防止选项在客户端应用程序中更改** 现在，通过在 OEM.ini 文件中锁定某些 Interactive SQL 选项设置，可以防止用户更改这些选项设置。请参见“[配置管理工具](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。

- **dbisql 的新 -version 选项** 在命令提示符处，键入 `dbisql -version`，以查看 Interactive SQL 的版本号。请参见“[Interactive SQL 实用程序 \(dbisql\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[UltraLite 的 Interactive SQL 实用程序 \(dbisql\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。

Sybase Central 和 Interactive SQL 的行为更改和不建议使用的功能

以下列出了 11.0.0 版中引入的 Sybase Central 和 Interactive SQL 更改。

- **更易于重新部署数据库工具启动程序可执行文件** 现在更易于重新部署 Sybase Central、Interactive SQL、数据库控制台实用程序和 MobiLink 监控器的启动程序可执行文件。不再需要用于 JAR 文件位置的注册表条目和一组目录结构。在与 `filename.exe` 文件相同的目录中，每个可执行文件都需要相应的 `filename.INI` 文件（具有相同的名称）。`.INI` 文件中包含如何装载工具的详细信息。请参见“[部署管理工具](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **不再支持 OEM.ini 中的 [help] 部分** 不再支持 `OEM.ini` 文件中的 `[help]` 部分。有关详细信息，请参见“[配置管理工具](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。

Sybase Central 的行为更改和不建议使用的功能

以下列出了 11.0.0 版中引入的 Sybase Central 更改。

- **Sybase Central 配置文件已重命名** 现在，已将 `.screpository` 文件命名为 `.screpository600`。

SQL Anywhere 插件的已更改功能

- **属性窗口的增强** 已经更新了以下属性页面：
 - [Web 服务属性] 窗口
 - [用户属性] 窗口
 - [视图属性] 窗口
 - [实例化视图属性] 窗口
- **调试特定用户** 在 SQL Anywhere 插件中输入 [调试] 模式时，必须指定需要调试哪些用户。请参见“[教程：调试程序入门](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **已更新向导** 已经更新了以下向导：
 - 创建用户向导
 - 创建 Web 服务向导
 - 部署向导

MobiLink 插件的已更改功能

- **已更新向导** 已经更新了以下向导：
 - 部署同步模型向导

SQL Anywhere 插件的不建议使用功能

- **已删除属性选项卡** 已经更新了以下属性窗口：
 - [表属性] 窗口
 - [UltraLite 项目属性] 窗口
 - [UltraLite 语句属性] 窗口
 - [UltraLite 语句属性] 窗口

Interactive SQL 的行为更改和不建议使用的功能

以下列出了 11.0.0 版中引入的 Interactive SQL 更改。

- **现在，在 [计划查看器] 中可以查看图形式计划** 在 Interactive SQL 中，现在在名为 [计划查看器] 的大小可调整的单独窗口中查看 SQL Anywhere 数据库的图形式计划。此更改使得查看和比较计划更为容易，因为现在可以同时打开多个 [计划查看器] 窗口。要访问 [计划查看器]，可选择 [工具] » [计划查看器]。[计划查看器] 中还显示 UltraLite 数据库的文本计划。请参见“[使用 Interactive SQL 计划查看器查看计划](#)”一节《SQL Anywhere 服务器 - 数据库管理》。

此外，已不再支持 Interactive SQL 的 isql_plan 选项。

- **已删除了对查看详细计划和简要计划的支持** 在 Interactive SQL 中，不再能够查看 SQL Anywhere 数据库的文本计划。但是，仍然可以使用 EXPLANATION 和 PLAN 函数检索这些计划。在 Interactive SQL 中，仍然可以使用 [计划查看器] 来查看 UltraLite 数据库的文本计划。
- **打印执行计划和结果集** 现在，可以按 Ctrl+P 或从 [文件] 菜单中选择 [打印] 来打印 [SQL 语句] 窗格的内容和结果集。以前，只能打印 [SQL 语句] 窗格的内容。在 [计划查看器] 中，通过按 [打印] 按钮可以进行打印。请参见“[打印 SQL 语句、执行计划和结果集](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **已将行号添加到 [SQL 语句] 窗格** 现在，在 [SQL 语句] 窗格的左侧显示行号。这些行号有助于您标识语法错误的位置。
- **对 [执行 SQL 语句] 工具栏按钮的增强** 以前，在 Interactive SQL 工具栏上，[执行 SQL 语句] 按钮用于执行所有 SQL 语句。现在可以指定单击此按钮时是执行所有语句还是仅执行选定的语句。

要设置 [执行 SQL 语句] 按钮的行为，可在 [工具] 菜单上选择 [选项] » [工具栏]。请参见“[从 Interactive SQL 执行 SQL 语句](#)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **对执行批处理语句的增强**
 - 在执行语句的批处理时，Interactive SQL 提供了改进的反馈。在 [SQL 语句] 窗格中执行 SQL 语句时，所执行的语句现在会被选定并且滚动显示。通过 [文件] » [运行脚本] 来执行脚本文件时，会出现状态窗口显示脚本的进度。请参见“[执行多条 SQL 语句](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **对 [结果] 窗格的增强**
 - 在 [结果] 窗格中，现在通过按 Ctrl+A 可以选择所有结果。也可以选择整个结果集而不仅仅是当前读取的结果。[结果] 窗格不包含整个结果集时，会提示您读取其余结果。否则，只会选择当前已读取的结果。

- 现在，从 **[结果]** 选项卡复制单元格时，所复制的数据会根据以下 Interactive SQL 选项来设置格式：isql_field_separator、isql_quote 和 isql_escape_character。也可以从结果集将选定的值、行和列复制到剪贴板。请参见“从 Interactive SQL 结果集中复制行”一节《SQL Anywhere 服务器 - 数据库管理》。
- 现在，单击 **[结果]** 选项卡中的某个列标题时，会按该列对结果进行排序。**[结果]** 窗格不包含整个结果集时，会提示您读取其余结果。否则，只会对当前已读取的结果进行排序。
- 现在，可以生成基于结果集中所选行的 INSERT、DELETE 和 UPDATE 语句以及将它们复制到剪贴板。请参见“从结果集生成 SQL 语句”一节《SQL Anywhere 服务器 - 数据库管理》。
- 已将 Interactive SQL 中的 **[结果]** 窗格增强，使其包含在右击菜单时提供的以下功能：
 - **[复制] » [复制单元格]** 复制所选单元格的内容。
 - **[复制] » [复制列]** 从所选单元格所在的列复制单元格值。
 - **[生成] » [INSERT 语句]** 为每个选定的行生成一个 INSERT 语句，并且将这些语句复制到剪贴板。
 - **[生成] » [DELETE 语句]** 为每个选定的行生成一个 DELETE 语句，并且将这些语句复制到剪贴板。
 - **[生成] » [UPDATE 语句]** 为每个选定的行生成一个 UPDATE 语句，并且将这些语句复制到剪贴板。这些生成的语句将列值设置为它们的当前值。因此，执行这些语句不会实际改变列值。此功能对于提供可以在执行前先进行编辑的模板 UPDATE 语句很有用。

请参见“从 Interactive SQL 结果集中复制行”一节《SQL Anywhere 服务器 - 数据库管理》和“从结果集生成 SQL 语句”一节《SQL Anywhere 服务器 - 数据库管理》。

● 对 Interactive SQL 语句的增强

- **DESCRIBE 语句增强** 现在，DESCRIBE 语句可以返回有关连接到 Interactive SQL 的数据库或数据库服务器的信息。请参见“DESCRIBE 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **INPUT 和 READ 语句增强** 现在，INPUT 和 READ 语句尝试以两种方式解析相对路径。请参见“INPUT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》和“READ 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **INPUT 和 OUTPUT 语句增强**
 - **对于从 ODBC 数据源导入和导出到 ODBC 数据源的新支持** 现在，使用 INPUT 和 OUTPUT 语句可以指定导入数据库和从数据库导出时的 ODBC 数据源。为此，请使用新 USING 子句。请参见“INPUT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》和“OUTPUT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。

使用 **[导入向导]** 和 **[导出向导]** 时，也可以指定一个 ODBC 数据源。请参见“使用 **[导入向导]** 导入数据”一节《SQL Anywhere 服务器 - SQL 的用法》和“使用 **[导出向导]** 导出数据”一节《SQL Anywhere 服务器 - SQL 的用法》。
 - **对于字节顺序标记 (BOM) 的新支持** 现在，可以控制是否处理数据中的某个字节顺序标记 (BOM)。为此，请使用新 BYTE ORDER MARK 子句。请参见“INPUT 语句

[\[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - SQL 参考》和“[OUTPUT 语句 \[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - SQL 参考》。

- **已经更改了 INPUT 语句的受支持格式** INPUT 语句不再支持 dBase、Lotus、Excel 和 FoxPro 文件格式。但仍然支持 TEXT 和 FIXED。如果需要继续使用这些文件格式，则必须通过 ODBC 驱动程序来使用。请参见“[INPUT 语句 \[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - SQL 参考》。
- **已经更改了 OUTPUT 语句的受支持格式** OUTPUT 语句不再支持 dBase、Lotus、Excel 和 FoxPro 文件格式。但仍然支持 TEXT、FIXED、HTML、SQL 和 XML。请参见“[OUTPUT 语句 \[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - SQL 参考》。
- **对于 INPUT 和 OUTPUT 语句，已将 ASCII 格式重命名为 TEXT** INPUT 和 OUTPUT 语句现在使用 TEXT。仍然支持对 ASCII 的使用，但这是为了向后兼容。
- **对 [导入向导] 和 [导出向导] 的更改** [导入向导] 或 [导出向导] 完成时，由向导生成的 SQL 语句存储在命令历史记录中。要查看生成的 SQL 语句，请选择 [SQL] » [历史记录]。
- **Interactive SQL 选项**
 - **isql_allow_read_client_file 和 isql_allow_write_client_file** 这两个选项描述 Interactive SQL 如何响应读取和写入客户端文件的请求。请参见“[isql_allow_read_client_file 选项 \[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - 数据库管理》和“[isql_allow_write_client_file 选项 \[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - 数据库管理》。
 - **不建议使用 -codepage 选项** 如果需要 Interactive SQL 读取具有特定代码页的文件，请使用 INPUT、OUTPUT 或 READ 语句的 ENCODING 子句。请参见：
 - “[Interactive SQL 实用程序 \(dbisql\)](#)”一节 《SQL Anywhere 服务器 - 数据库管理》
 - “[UltraLite 的 Interactive SQL 实用程序 \(dbisql\)](#)”一节 《UltraLite - 数据库管理和参考》
 - “[INPUT 语句 \[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “[OUTPUT 语句 \[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “[READ 语句 \[Interactive SQL\]](#)”一节 《SQL Anywhere 服务器 - SQL 参考》
 - **不支持 isql_plan 选项** 不再支持 Interactive SQL 的 isql_plan 选项。如果试图设置该选项，系统会以静默方式忽略，以便向后兼容。请参见“[使用 Interactive SQL 计划查看器查看计划](#)”一节 《SQL Anywhere 服务器 - 数据库管理》。
 - **已删除 SET OPTION 语句 PUBLIC 关键字** 在使用 SET OPTION 语句设置 Interactive SQL 选项时，已删除了对 PUBLIC 关键字的支持。请参见“[Interactive SQL 选项](#)”一节 《SQL Anywhere 服务器 - 数据库管理》。
- **对 Interactive SQL 启动程序的更改** 以前，Interactive SQL 启动程序可执行文件的 Windows 版本名为 *dbisqlg.exe*，其命令行版本名为 *dbisql.exe*。现在，Interactive SQL 的 Windows 版本名为 *dbisql.exe*，其命令行版本名为 *dbisql.com*。

批处理脚本应调用 *dbisql* 或 *dbisql.com* 而不是 *dbisql.exe*。

SQL Anywhere 控制台实用程序行为更改

以下列出了 11.0.0 版中引入的 SQL Anywhere 控制台实用程序的更改。

- **新 Console 选项** 现在，可在 [选项] 窗口中指定数据库服务器的关闭日期和时间。从 [文件] 菜单中，选择 [选项] » [控制台]。

MobiLink 监控器的行为更改

以下列出了 11.0.0 版中引入的 MobiLink 监控器的更改。

MobiLink 监控器不能读取使用版本 9 或更早版本的 MobiLink 服务器创建的监控器文件；MobiLink 监控器仅应和相同主版本的 MobiLink 服务器一起使用。此外，已经删除了工作列，并且重命名了以下 MobiLink 监控器属性：

旧属性名称	新属性名称
preload_upload	sync_request
verify_upload	authenticate_user

文档增强

- **文档目录增强** 以前，文档位于 *install-dir\docs* 中；现在，文档驻留在 *install-dir\documentation* 中。各个 HTML 帮助文件和 PDF 文件的文件名也已更新。

HTML 帮助文件的新文件名如下：

- *dbadmin_en11.chm*
- *dbprogramming_en11.chm*
- *dbreference_en11.chm*
- *dbusage_en11.chm*
- *mlclient_en11.chm*
- *mlserver_en11.chm*
- *mlsisync_en11.chm*
- *mlstart_en11.chm*
- *qanywhere_en11.chm*
- *sachanges_en11.chm*
- *saerrors_en11.chm*
- *saintro_en11.chm*
- *scplugin_en11.chm*
- *sqlanywhere_en11.chm*
- *sqlremote_en11.chm*
- *uladmin_en11.chm*
- *ulc_en11.chm*
- *uldotnet_en11.chm*
- *ulj_en11.chm*
- *ulmbus_en11.chm*

- **现在通过 Web 站点访问所支持的平台页面** 以前，所支持的平台页面与软件一起安装。现在，所有支持平台信息均在以下 Sybase Web 站点上提供：<http://www.sybase.com/detail?id=1062617>。
- **[查找] 按钮** 现在，在 HTML 帮助浏览器中，可以使用 **[查找]** 按钮来查看当前帮助页位于目录中的何处。

产品范围的功能

以下各节介绍了会影响 11.0.0 版 SQL Anywhere 的所有组件的新功能、行为更改和不建议使用的功能。

产品范围的新功能

以下列出了 11.0.0 版中引入的产品范围的新增功能。

- **错误报告增强** 在 Windows、Windows Mobile 和 Linux 上，生成错误报告时会出现一个窗口，您可在选择是否将其提交给 iAnywhere 之前于该窗口中查看错误报告的内容。请参见“[SQL Anywhere 中的错误报告](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

现在，可以使用 `-ce` 选项配置支持实用程序 (dbsupport)，该选项用于当 dbsupport 正在监控某个应用程序并且该应用程序崩溃时发送电子邮件。请参见“[支持实用程序 \(dbsupport\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **附加的 Windows Mobile 平台支持** SQL Anywhere 现在支持 Windows Mobile 5 for smartphone 和 Windows Mobile 6 标准版。有关在 Windows Mobile 上运行 SQL Anywhere 服务器的信息，请参见“[SQL Anywhere for Windows Mobile](#)”《[SQL Anywhere 服务器 - 数据库管理](#)》和“[安装注意事项：Windows Mobile 5.0 for smartphone 的限制](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **SQL Anywhere 示例数据库 (demo.db) 中的新表** 已将新表 MarketingInformation 添加到 SQL Anywhere 示例数据库中。此表中的每行都包含一个 HTML 页，用于描述 Products 表中的某个产品。测试和试验功能时，添加的此表可为查询提供更丰富的字符数据。请参见“[SQL Anywhere 示例数据库](#)”《[SQL Anywhere 11 - 简介](#)》。

产品范围的行为更改

以下列出了 11.0.0 版中产品范围内的更改。

- **已将 Windows CE 更改为 Windows Mobile** 在文档和软件中已将名称 Windows CE 更改为 Windows Mobile，但在某些继续称 Windows CE 更为准确的地方除外。
- **已删除了 Readcert、gencert 和 reqtool** 已删除实用程序 readcert、gencert 和 reqtool。先前已经不建议使用这些实用程序。可以使用 createcert 和 viewcert 来代替它们。请参见“[证书实用程序](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **在 Mac OS X 上支持 createcert 和 viewcert 实用程序** 现在，在 Mac OS X 上支持 createcert 和 viewcert 证书实用程序。请参见“[证书实用程序](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **已重命名 certificate 和 certificate_password 协议选项** 已将 TLS 和 HTTPS 的 certificate 和 certificate_password 协议选项分别重命名为 identity 和 identity_password。请参见：
 - SQL Anywhere 数据库服务器：“-ec 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》
 - SQL Anywhere Web 服务器：“-xs 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》
 - SQL Anywhere 协议选项：“Identity 协议选项”一节《SQL Anywhere 服务器 - 数据库管理》和“Identity_Password 协议选项”一节《SQL Anywhere 服务器 - 数据库管理》
 - MobiLink 服务器：“-x 选项”一节《MobiLink - 服务器管理》
- **示例标识文件更改** 在本发行版本中，已将包含 TLS 的示例证书和相应专用密钥的标识文件重命名。已将文件 *rsaserver.crt* 重命名为 *rsaserver.id*，将文件 *sample.crt* 重命名为 *eccserver.id*。已将这两个标识文件的口令从 **tJ1#m6+W** 更改为 **test**。
- **对安装目录的更改** 现在，将 32 位软件安装到 *bin32* 目录中而不是 *win32* 目录中，将 64 位软件安装到 *bin64* 目录中而不是 *X64* 目录中。例如，在以前的版本中安装到 *C:\Program Files\SQL Anywhere 11\win32* 中的软件现在安装到 *C:\Program Files\SQL Anywhere 11\bin32* 中。
- **对示例数据库的 ODBC 数据源的更改** 在以前的发行版本中，使用该软件安装的示例数据库的 ODBC 数据源是用户数据源。现在，SQL Anywhere 11 Demo、SQL Anywhere 11 CustDB 和 QAnywhere 11 Demo 数据源是系统数据源。
- **不支持 .NET 1.0** SQL Anywhere 11 不支持 Visual Studio 2003，但支持 Visual Studio 2005 (.NET 2.0) 和 Visual Studio 2008 (.NET 3.x)。

10.0.1 版中的新功能

目录

SQL Anywhere	87
MobiLink	96
QAnywhere	98
SQL Remote	100
UltraLite	101
产品范围的功能	103

SQL Anywhere

- “新功能” 一节第 87 页
- “行为更改和不建议使用的功能” 一节第 92 页

MobiLink

- “新功能” 一节第 96 页
- “行为更改和不建议使用的功能” 一节第 97 页

QAnywhere

- “新功能” 一节第 98 页
- “行为更改和不建议使用的功能” 一节第 98 页

UltraLite

- “新功能” 一节第 101 页
- “行为更改和不建议使用的功能” 一节第 102 页

SQL Remote

- “行为更改和不建议使用的功能” 一节第 100 页

产品范围的功能

- “新功能” 一节第 103 页
- “行为更改” 一节第 104 页
- “Windows Vista 的支持问题” 一节第 104 页

不建议使用的功能的列表可能会发生更改

与所有前瞻性声明一样，不建议使用的功能的列表不保证其完整性，并可能会发生更改。

SQL Anywhere

以下各部分介绍 10.0.1 版的 SQL Anywhere 中的新功能、行为更改和不建议使用的功能。

注意

在发行版本 10 之前，SQL Anywhere 数据库服务器称为 Adaptive Server Anywhere。

新功能

以下列出了 10.0.1 版中在 SQL Anywhere 数据库和数据库服务器方面新增的功能。

加密增强

已进行了如下更改以增强对加密的支持。

- **对 CREATE DATABASE 语句 ENCRYPTION 子句的扩展** CREATE DATABASE 语句的 ENCRYPTION 子句的语法已得到了扩展，允许您指定 SIMPLE 作为加密类型。此外，您还可以按任意顺序指定加密密钥和算法。请参见“[CREATE DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **dbinit 和 dbunload -ea 选项的增强** dbinit 和 dbunload 的 -ea 选项现在接受 none 和 simple 作为加密类型。指定 none 会导致不加密。指定 simple 会导致简单加密。同样，缺省加密类型也已更改，它取决于 -ek、-et 或 -ep 选项是否与 -ea 一同指定。请参见“[初始化实用程序 \(dbinit\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[卸载实用程序 \(dbunload\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
不建议使用 -e 选项。请参见“[行为更改和不建议使用的功能](#)”一节第 92 页。
- **Mac OS X 上的高度加密** 您现在可以在 Mac OS X 上使用 RSA 加密，来加密客户端/服务器通信。请参见“[加密 SQL Anywhere 客户端/服务器通信](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

对客户端语句高速缓存的支持

客户端语句高速缓存现在受到支持并且会在缺省状态下启用，从而在重复准备和删除同一 SQL 文本时，客户端会高速缓存此语句，使它被应用程序删除后，在服务器上处于准备好状态。这样就节省了数据库服务器删除和重新准备语句的额外工作量。使用客户端语句高速缓存需要 10.0.1 版客户端库和 10.0.1 版数据库服务器。

对支持客户端语句高速缓存进行了以下更改。

- **max_client_statements_cached 选项** 此选项指定可在数据库服务器上保留为已高速缓存（准备好）的语句（即使已被应用程序删除）的最大数。请参见“[max_client_statements_cached 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **新连接和服务器属性** 已添加了 ClientStmtCacheHits、ClientStmtCacheMisses 和 max_client_statements_cached 属性。请参见“[连接属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[数据库服务器属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **新请求统计** 已添加了语句高速缓存命中和语句高速缓存未命中统计功能。请参见“请求统计”一节《SQL Anywhere 服务器 - SQL 的用法》。

SQL Flagger 增强

SQL Flagger 功能已得到了增强，它能够更好地检测兼容性，并且加入了对更新标准的支持。例如，您现在可以测试与某个特定 SQL 标准的兼容性，或与 UltraLite SQL 的兼容性。

为支持这些增强功能，进行了以下更改：

- **新 SQLFLAGGER 函数** 您可以使用新 SQLFLAGGER 函数来测试 SQL 语句是否符合指定的 SQL 标准，而无需实际运行此语句。请参见“SQLFLAGGER 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新 sa_ansi_standard_packages 系统过程** 使用新 sa_ansi_standard_packages 系统过程，您可以指定 SQL 标准和 SQL 语句，并获得语句执行期间可能会用到的非核心 SQL 扩展的列表。请参见“sa_ansi_standard_packages 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
您必须升级数据库才能使用此功能。请参见“升级版本 10 及更高版本数据库”一节第 333 页。
- **sql_flagger_error_level 和 sql_flagger_warning_level 数据库选项的新值** 为 sql_flagger_error_level 和 sql_flagger_warning_level 数据库选项提供了几个新值，以支持 SQL/1999 和 SQL/2003 标准。请参见“sql_flagger_error_level 选项 [兼容性]”一节《SQL Anywhere 服务器 - 数据库管理》和“sql_flagger_warning_level 选项 [兼容性]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **SQL 预处理器 (sqlpp) -e 和 -w 选项的新值** 为 SQL 预处理器 (sqlpp) 的 -e 和 -w 选项提供了几个新值，以支持 SQL/1999 和 SQL/2003 标准。请参见“SQL 预处理器”一节《SQL Anywhere 服务器 - 编程》。

SQL 语句

在 SQL 语句和函数方面进行了以下增强。

- **START DATABASE 语句增强** START DATABASE 语句目前支持 DIRECTORY 子句，此子句能够使您指定数据库的 dbspace 文件所在的目录。请参见“START DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **INSERT、UPDATE、DELETE、SELECT、UNION、EXCEPT 和 INTERSECT 语句包含一个 OPTION 子句** INSERT、UPDATE、DELETE、SELECT、UNION、EXCEPT 和 INTERSECT 语句支持 OPTION 子句，此子句控制语句使用实例化视图的方式、指定优化查询的方式，并且还能够替换以下数据库选项的设置：
 - isolation_level
 - max_query_tasks
 - optimization_goal
 - optimization_level
 - optimization_workload

请参见：

- “INSERT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “UPDATE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “DELETE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “SELECT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “UNION 子句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “EXCEPT 子句”一节 《SQL Anywhere 服务器 - SQL 参考》
- “INTERSECT 子句”一节 《SQL Anywhere 服务器 - SQL 参考》

- **HTML_DECODE 函数** HTML_DECODE 函数现在可以解码更多作为数字实体给出的 Unicode 代码点，如商标符号 (™)。如果代码点无法用数据库字符集表示，它将以代码点形式存在。从前，小于 0x7F 的代码点会转换为字符（对于某些字符集，小于 0xFF 的代码点会转换为字符），而所有其它代码点将继续以代码点形式存在。请参见“HTML_DECODE 函数 [Miscellaneous]”一节 《SQL Anywhere 服务器 - SQL 参考》。

对归类定制的支持

SQL Anywhere 目前支持在创建数据库时进行归类定制。为支持归类定制进行了如下更改：

- **CREATE DATABASE 语句的增强** 当使用 CREATE DATABASE 语句或初始化实用程序 (dbinit) 创建数据库时，您现在可以指定定制选项，以便能够对字符的排序和比较进行更多的控制。

对于 CREATE DATABASE 语句，可使用 COLLATION 和 NCHAR COLLATION 子句来支持归类定制。请参见“CREATE DATABASE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。

对于初始化实用程序，可使用 -z 和 -zn 选项来支持归类定制。请参见“初始化实用程序 (dbinit)”一节 《SQL Anywhere 服务器 - 数据库管理》。

注意

使用归类定制选项创建的数据库无法使用 10.0.1 之前版本的数据库服务器启动。

如果想要在现有数据库中使用归类定制，您必须创建支持归类定制的新 10.0.1 版数据库，卸载现有数据库，然后将数据库重装为新 10.0.1 版数据库。请参见“重建版本 10 及更高版本数据库”一节第 327 页。

- **新 HasCollationTailoring 数据库属性** HasCollationTailoring 是一个新数据库属性，表示创建数据库时是否启用了定制支持。请参见“数据库属性”一节 《SQL Anywhere 服务器 - 数据库管理》。
- **新增扩展属性值** 当查询 Collation、NcharCollation 和 CatalogCollation 数据库属性时，可使用以下新增的 DB_EXTENDED_PROPERTY 值：CaseSensitivity、AccentSensitive、PunctuationSensitivity、Properties 和 Specification。请参见“DB_EXTENDED_PROPERTY 函数 [System]”一节 《SQL Anywhere 服务器 - SQL 参考》。
- **SORTKEY 和 COMPARE 函数的增强** 除了将归类名作为参数接受外，SORTKEY 和 COMPARE 函数现在将相同的带括号归类定制选项集作为 CREATE DATABASE 语句接受。请参见“SORTKEY 函数 [String]”一节 《SQL Anywhere 服务器 - SQL 参考》和“COMPARE 函数 [String]”一节 《SQL Anywhere 服务器 - SQL 参考》。

Web 服务的增强

进行了以下增强来提高 HTTP 和 SOAP 标头的可配置性：

- **改善的可配置性** 使用 CREATE PROCEDURE 和 CREATE FUNCTION 语句的新 SET 子句，您能够修改 HTTP 和 SOAP 协议的以下选项：客户端使用的 HTTP 版本、是否使用分块、以及执行 SOAP 请求时要调用的 SOAP 操作名（如果与过程名或函数名不同）。请参见“[CREATE PROCEDURE 语句（Web 服务）](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **HTTP 标头规范** CREATE PROCEDURE 和 CREATE FUNCTION 语句的 HEADER 子句的语法已得到了扩展，这样一来您就能取消给定的 HTTP 请求标头，或为它提供一个空值。此功能可扩展到自动生成的 HTTP 请求标头，这些标头在之前的版本中是不能修改的。请参见“[CREATE PROCEDURE 语句（Web 服务）](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[修改 HTTP 标头](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **对 SOAP:RPC 客户端数据类型的支持** 可使用 CREATE SERVICE 语句的 DATATYPE 子句启用数据类型设置。数据类型信息包含在参数输入和结果集输出或所有 SOAP 服务格式的响应的 XML 编码中。这样一来便不再需要客户端代码将参数显式转换为字符串，从而简化了从 SOAP 工具箱进行参数传递。请参见“[使用数据类型](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **Mac OS X 上支持的 HTTPS** 在以前的版本中，Mac OS X 上只支持 HTTP 协议。现在，当将 SQL Anywhere 数据库服务器作为 Web 服务器在 Mac OS X 上运行时，可以使用 HTTPS。请参见“[-xs 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[SQL Anywhere Web 服务](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。

数据库镜像的增强

数据库镜像功能中增加了以下增强：

- **为数据库镜像指定首选数据库服务器** 您现在可以指定哪个数据库服务器应该在镜像系统中担当主服务器的角色。请参见“[-xp 数据库选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[指定首选数据库服务器](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **启动从主服务器到镜像服务器的数据库镜像故障转移** 现在使用 ALTER DATABASE 语句的 SET PARTNER FAILOVER 子句，可以启动从主服务器到镜像服务器的故障转移。请参见“[ALTER DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[启动主服务器上的故障转移](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

SQL Anywhere 插件

- **[触发器] 文件夹列名称已更改** 在 [触发器] 文件夹中，[表名称] 和 [表所有者] 列已被 [对象名称]、[对象所有者] 和 [对象类型] 列替换。缺省情况下不会显示 [对象类型] 列，但是可通过选择 [视图] » [选择列] 来显示它。
- **[触发器] 选项卡已添加到 [视图属性] 窗口** 非实例化视图的属性窗口现在具有列出视图的 INSTEAD OF 触发器的 [触发器] 选项卡。
- **INSTEAD OF 触发器支持已添加到 [创建触发器向导]** [创建触发器向导] 中新增了几处增强，以支持 INSTEAD OF 触发器，包括选择是否为表或非实例化视图创建触发器的选项。请参见“[创建触发器](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

- **归类定制支持已添加到 [创建数据库向导]** 如果所选数据库服务器是 10.0.1 或之后版本的服务器，或者如果已选择通过启动新数据库服务器在本地计算机上创建数据库，则 [创建数据库向导] 现在包含归类定制页。

其它增强

- **简单 DML 语句的计划高速缓存** 计划高速缓存已扩展为包括可以跳过查询的 INSERT、UPDATE 和 DELETE 语句（简单语句）。请参见“计划高速缓存”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **具有左和右外连接的实例化视图在基于开销的优化期间可以使用** 从前，允许在实例化视图的定义中使用左和右外连接。但是，这会导致无法在基于开销的优化中使用实例化视图。现在，则可以在基于开销的优化期间使用具有左或右外连接的实例化视图。请参见“使用实例化视图提高性能”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **对 INSTEAD OF 触发器的支持** BEFORE 或 AFTER 触发器分别在触发操作前或后触发。INSTEAD OF 触发器取代了触发操作。INSTEAD OF 触发器能够使您在进行插入、更新或删除操作期间，对触发器的行为进行更多的控制。请参见“CREATE TRIGGER 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **DBTools 增强** 现在您可以通过使用 DBCreatedVersion 函数，来确定数据库是使用 SQL Anywhere 10.0.0 版还是之前版本创建的，而无需启动数据库。请参见“DBCreatedVersion 函数”一节《SQL Anywhere 服务器 - 编程》。
- **OLAP 增强** 现在支持两个新增的窗口集合函数 FIRST_VALUE 和 LAST_VALUE。这两个函数会分别返回窗口的第一个值或最后一个值，从而不必再使用自连接返回这些值。然后您可以将这些值作为在此窗口执行的进一步计算中的基线。请参见“FIRST_VALUE 函数 [Aggregate]”一节《SQL Anywhere 服务器 - SQL 参考》和“LAST_VALUE 函数 [Aggregate]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **Unix 增强了对 IPv6 的支持** 在 Unix 上，您可以指定一个接口标识符或接口名称，作为 IPv6 地址的一部分。在 Linux（内核 2.6.13 及之后版本）上，当您在客户端或服务上指定 IP 地址时（例如，在使用 HOST=、MYIP= 或 BROADCAST= TCP 协议选项时），必须使用一个接口标识符。请参见“SQL Anywhere 中的 IPv6 支持”一节《SQL Anywhere 服务器 - 数据库管理》。
- **对 TDS DATE 和 TIME 数据类型的支持** TDS DATE 和 TDS TIME 数据类型最近已引入 TDS 客户端。使用 Open Client 15 或更新版本或者使用 jConnect 的 EBF 的应用程序，现在可将日期和时间列做为 TDS DATE 或 TDS TIME 值而非 TDS DATETIME 值读取。

SQL Anywhere 已进行了增强，因此基于 TDS 的应用程序可将日期和时间数据作为 TDS DATE 和 TDS TIME 值读取。使用旧版 Open Client 或 jConnect 的应用程序，将继续将日期和时间数据作为 TDS DATETIME 值读取。注意：那些不是基于 TDS 的应用程序（使用嵌入式 SQL、ODBC 或 iAnywhere JDBC 驱动程序的应用程序）始终能够将日期和时间数据作为日期和时间值读取。

- **列出可用字符集编码的新 dbinit 选项** 使用初始化实用程序 (dbinit) -le 选项列出数据库的可用字符集编码。请参见“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **新 -ds 服务器选项** -ds 服务器选项允许您指定数据库的 dbspace 文件所在的目录。请参见“-ds 数据库选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **SADbType.Xml 数据类型** 已将 SADbType.Xml 枚举常量添加到 SQL Anywhere .NET 提供程序。

- **SQL Anywhere SNMP Extension Agent 的动态陷阱支持单位** 设置动态陷阱时，在指定陷阱的数字值时，现在可使用 k、m、g 或 t 将单位分别指定为千字节 (KB)、兆字节 (MB)、千兆字节 (GB) 或千吉字节 (TB)。请参见“创建动态陷阱”一节《SQL Anywhere 服务器 - 数据库管理》。
- **为 iAnywhere Solutions Oracle 驱动程序创建 ODBC 数据源** 现在可以使用数据源实用程序 (dbdsn)，通过指定 -or 选项，为 iAnywhere Solutions Oracle 驱动程序创建 ODBC 数据源。请参见“数据源实用程序 (dbdsn)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **对提交错误报告的窗口的增强** 提示您向 iAnywhere 提交错误报告的 dbsupport 窗口现在包含一个 [查看错误报告] 按钮，因此您可以在提交错误报告前查看其中包含的信息。

行为更改和不建议使用的功能

以下列出了 10.0.1 版中引入的 SQL Anywhere 数据库和数据库服务器更改（根据类别分组）。

行为更改

- **对使用查询内并行机制的时机的更改** 查询内并行机制不再用于 background_priority 设置为 on 的连接。此外，如果当前正在处理请求的服务器线程的数量 (ActiveReq 服务器属性)，最近超出了授权该数据库服务器使用的计算机上的 CPU 内核的数量，则不使用查询内并行机制。请参见“查询执行期间的并行”一节《SQL Anywhere 服务器 - SQL 的用法》。

新服务器属性 ExchangeTasksCompleted 返回自数据库服务器启动以来用于查询内并行机制的内部任务总数。请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。

- **加密结果不再为确定型** 以前，使用 ENCRYPT 函数的加密值为确定型。如果输入两个相同的输入字符串和两个相同的加密密钥，则会返回相同的输出数据（密码文本）。现在，可以使用新 encrypt_aes_random_iv 数据库选项来控制加密是否为确定型。新的缺省行为是非确定型。

注意

不具有此数据库选项的数据库服务器（版本 10.0.0 和之前版本），无法从设置了此选项的数据库解密数据，即使在此选项设置为 Off 时也是如此。

- **ALTER DBSPACE RENAME 尝试打开未打开的 dbspace** 以前，如果启动了使用 dbspace 的数据库，并且无法找到其中一个 dbspace，对该 dbspace 执行 ALTER DBSPACE ...RENAME 语句会在此目录中更新该 dbspace 的名称，但不会尝试启动该 dbspace。现在，数据库服务器将在目录更新后，尝试打开该 dbspace。请参见“ALTER DBSPACE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **对 CREATE、ALTER 和 DROP DBSPACE 语句的更改** CREATE DBSPACE 和 DROP DBSPACE 语句不再接受预定义的 dbspace 的名称 (SYSTEM、TEMPORARY、TEMP、TRANSLOG 和 TRANSLOGMIRROR)。如果旧版本的 SQL Anywhere 数据库服务器创建的数据库中的一个用户 dbspace 与预定义的 dbspace 之一同名，则数据库服务器将始终引用该用户 dbspace。请参见“预定义 dbspace”一节《SQL Anywhere 服务器 - 数据库管理》。
- **对 sa_conn_info 系统过程的输出的更改** sa_conn_info 系统过程的输出已更改，以便能够给出有关连接正在等待的锁的更多信息。LockName 字段已删除。代替它的是两个新增字段 LockRowID 和 LockIndexID。如果连接正在等待与特定行标识符相关联的锁，LockRowID 会包

含此行标识符。如果连接正在等待与特定索引相关联的锁，LockIndexID 会包含此索引的标识符。请参见“[sa_conn_info 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

您必须升级数据库才能使用此功能。请参见“[升级版本 10 及更高版本数据库](#)”一节第 333 页。

- **某些系统过程不再需要 DBA 权限** sa_dependent_views、sa_get_dtt、sa_check_commit 和 sa_materialized_view_info 系统过程运行时不再需要 DBA 权限。
- **CREATE DATABASE 语句的缺省测量单位已删除** 当使用 CREATE DATABASE 语句创建数据库时，如果为 DATABASE SIZE 指定了一个值，则指定测量单位不再为可选。请参见“[CREATE DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **default_timestamp_increment 选项的新最大值** default_timestamp_increment 选项的最大值现在为 1000000（1 秒）。请参见“[default_timestamp_increment 选项 \[数据库\] \[MobiLink 客户端\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **dbdata10.dll 已删除** 已将 dbdata10 动态链接库提供的功能合并到 SQL Anywhere .NET 提供程序 DLL 中。因此，对于 Windows CE，现在有特定于平台的 SQL Anywhere .NET 提供程序 DLL 版本。
- **NetWare 上的缺省高速缓存大小已增加** NetWare 上数据库服务器高速缓存的缺省大小已从 2 MB 增加到 8 MB。请参见“[-c 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **客户端语句高速缓存导致的行为更改** 作为支持客户端语句高速缓存的结果，已引入以下行为更改：
 - 如果同一 SQL 语句之前描述为没有结果集，之后又确实有结果集，将会出现不正确描述。例如：


```
CREATE PROCEDURE p() NO RESULT SET BEGIN ... END
Prepare, Describe, Drop "call p"
ALTER PROCEDURE p() RESULT( ... ) BEGIN ... END
Prepare, Describe, Drop "call p" // describe returns no result set
```
 - 如果启用了客户端语句高速缓存并且启用了 RememberLastStatement（-zl 服务器选项），则当重新使用高速缓存的语句时，此 LastStatement 属性为空字符串。
 - 当启用客户端语句高速缓存时，如果使用 sa_get_request_times 或 sa_get_request_profile 处理请求级日志，则语句的执行次数可能会不正确。请参见“[max_client_statements_cached 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **语言实用程序 (dblang) 不再需要管理员权限** 在以前的 SQL Anywhere 版本中，用户必须以管理员身份登录才能通过使用 dblang 实用程序来更改 SQL Anywhere 本地化版本的语言设置。此要求现已删除。
- **DISH 服务名中的正斜线已不允许使用** 为了防止对 DISH 服务名的错误解释，不再允许使用正斜线 (/) 作为服务名的一部分。请参见“[CREATE SERVICE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **对 SACommand.UpdateRowSource 的缺省值的更改** 以前，SACommand.UpdatedRowSource 的缺省值是 UpdatedRowSource.Both。此值现已更改为 UpdatedRowSource.OutputParameters。请参见“[UpdatedRowSource 属性](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **对 PrefetchRows 连接参数的缺省值的更改** 当使用 .NET 数据提供程序时，为了提高性能，PrefetchRows 连接参数的缺省值已从 10 更改到 200。SAConnectionStringBuilder.PrefetchRow 的

缺省值也已更改为 200。如果结果集包含 BLOB 列，将禁用预取。请参见“[PrefetchRows 连接参数 \[PROWS\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **经过验证的应用程序现在使用 `authenticate.sql` 来代替 `saopts.sql`** 在 SQL Anywhere OEM 版的以前版本中，曾经建议您将验证语句存储在文件 `install-dir\scripts\saopts.sql` 中，以便在创建、重建或升级数据库时能够随时应用它。

现在建议您将验证字符串存储在文件 `install-dir\scripts\authenticate.sql` 中。请参见“[升级经过验证的数据库](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **HP-UX 上现已接受长主机名** 从 HP-UX 11i v2 2004 年 9 月的更新开始，系统管理员通过设置内核参数能够启用对 255 字节主机名的支持。但是，在已启用了长主机名的支持的 HP-UX 计算机上的 SQL Anywhere 服务器上，`MachineName` 属性和 `AppInfo` HOST 密钥最多返回 64 字节主机名。现在 `MachineName` 和 `AppInfo` 都能够返回 255 字节主机名。
- **iAnywhere JDBC 驱动程序 URL 标头** 在以前的版本中，当应用程序使用 iAnywhere JDBC 驱动程序连接到 SQL Anywhere 时，传递到 JDBC 驱动程序的 URL 以标头 `jdbc:odbc:` 开头。现在，URL 标头还可以用 `jdbc:iAnywhere:` 开头。建议使用 `jdbc:iAnywhere:`，以避免与 Sun JDBC-ODBC bridge 冲突。请参见“[将 URL 提供给驱动程序](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **当注释值长度大于 128 字符时，使用 `jConnect` 检索表的列表** 以前，如果 JDBC 应用程序连接（使用 `jConnect`）并请求表的列表，则即使表存在结果也可能为空。当 `string_truncation` 选项设置为 `On`，应用程序使用 `DatabaseMetaData.getTables` 方法，而且任何表的注释值都长于 128 字符时，会出现这种情况。现在，那些过长的注释值会截断为 128 字符，因此能够返回表的列表。您必须运行 `jcatalog.sql` 或升级数据库，才能使用此更改。请参见“[在数据库中安装 `jConnect` 系统对象](#)”一节《[SQL Anywhere 服务器 - 编程](#)》或“[升级实用程序 \(dbupgrad\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **比较 CHAR 和 NCHAR 值** 对于 SQL Anywhere 10.0.0，将 CHAR 和 NCHAR 域结合会导致 NCHAR 比较。但是，这意味着当使用绑定为 `SQL_C_WCHAR` 的主机变量时，升级到 10.0.0 的应用程序可能会得到不同的结果或性能降低。在 SQL Anywhere 10.0.0 中，绑定为 `SQL_C_WCHAR` 的变量表示为 NCHAR。在 SQL Anywhere 10.0.1 中，引入了新推导规则，以提高与现有应用程序的兼容性，以及在结合 CHAR 和 NCHAR 域时提供一致的、可预见的结果。请参见“[CHAR 和 NCHAR 之间的比较](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **2.6.12 之前的 Linux 内核上的异步 I/O 已禁用** 由于 2.6.12 之前的 Linux 内核中存在一个错误，因此在其中一个受影响的内核上运行 SQL Anywhere 数据库服务器时，在缺省条件下异步 I/O 将被禁用。如果要使用异步 I/O，则必须将内核升级到 2.6.12 或之后版本。
- **OEM 版文档已移走** 在以前版本的 SQL Anywhere OEM 版中，设置经过验证的应用程序的说明包含在单独的 `.pdf` 或 `.html` 文件中。现在，可在以下位置获得此信息：
 - “[运行经过验证的 SQL Anywhere 应用程序](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
 - “[connection_authentication 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
 - “[database_authentication \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
- **卸载实用程序 `-e` 和 `-t` 选项用于区分大小写的数据库时不再需要区分大小写的表名** 在以前的版本中，当使用 `dbunload` 实用程序和 `-e` 或 `-t` 选项卸载区分大小写的数据库时，这些选项需要区分大小写的表名。现在，表名不再区分大小写。
- **向临时表中装载数据** 向临时表中装载数据时的行为已更改。除了那些使用 `ON COMMIT DELETE ROWS` 定义的 `LOCAL TEMPORARY TABLE` 之外，现在在针对临时表执行 `LOAD`

TABLE 前后，将自动执行一个提交。现在，如果装载失败，临时表中的所有行将全部被删除，包括在装载之前已经存在的行。

对于带有 ON COMMIT DELETE ROWS 的 LOCAL TEMPORARY TABLE，则没有行为的更改；不会执行任何提交。这意味着装载期间如果出现故障而导致部分装载，此种类型的临时表将只包含某些已装载的行，同时还可能丢失装载前已存在的其它行。

同样，现在如果一个临时表包含另一个表的外键所引用的行，向此临时表中进行装载也会失败。

您无法向使用 ON COMMIT DELETE ROWS 定义的 GLOBAL TEMPORARY TABLE 进行装载。

- **使用 UCA 归类的日文数据库的区分大小写缺省值** 创建日文数据库时 UCA 归类的区分大小写和区分重音缺省值，现在为区分。日文数据库定义为在日文计算机（OS 语言或字符集为日文）上创建的任何数据库或使用日文 CHAR 归类（如 932JPN 或 EUC_JAPAN）创建的任何数据库。

创建非日文数据库时 UCA 归类的区分大小写缺省值仍为不区分。

区分大小写和区分重音缺省值仍然可以分别使用 dbinit -c 和 -a（或 -c- 和 -a-）选项来替换，替换时既可以使用归类定制语法，也可以使用 CREATE DATABASE STATEMENT 的 CASE 和 ACCENT 子句。请参见“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》和“CREATE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

不建议使用和停用的功能

- **SQL Flagger 对 SQL/1992 标准的支持** 不建议使用 SQL Flagger 对 SQL:1992（所有级别）的支持。
- **不建议使用 dbinit -e 选项** dbinit -e 选项用于在创建数据库时指定简单加密，目前不建议使用。使用 -ea 选项（尤其是 -ea simple）指定简单加密。请参见“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **SADbType.oldbit 数据类型已删除** 已将 SADbType.oldbit 枚举常量从 SQL Anywhere .NET 提供程序中删除。
- **不建议使用 -gx 服务器选项** 在 Windows 桌面平台上，数据库服务器调度程序现在试图保持请求的关联度，以便它能够使用 CPU 高速缓存。这样一来，在一个 CPU 上运行尽可能多的请求。同样，也不建议使用 -gx 服务器选项（它用于指定数据库服务器要使用的操作系统线程的数量）。现在这个选项被数据库服务器忽略。
- **不建议使用 CREATE DATABASE 语句的 CASE 和 ACCENT 子句** 由于使用 CREATE DATABASE STATEMENT 的 COLLATION 和 NCHAR COLLATION 添加了归类定制支持，因此不建议使用此语句的 CASE 和 ACCENT 子句。请参见“CREATE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

MobiLink

以下各节介绍 10.0.1 版的 MobiLink 中的新功能、行为更改和不建议使用的功能。

新功能

MobiLink 服务器

- **用于 Oracle 的新 ODBC 驱动程序** 现在为 Oracle 提供了一个名为 iAnywhere Solutions 10 - Oracle 的 ODBC 驱动程序，该程序是为 MobiLink 应用程序自定义定制的。
请参见“[用于 Oracle 的新 ODBC 驱动程序](#)”一节第 103 页。
- **已重新组织的加密层** 已将 MobiLink 加密层重新组织并加以改进。此更改对应用程序是透明的，应用程序无需做任何更改。
- **MobiLink 服务器日志文件查看器** 添加了一个支持查看 MobiLink 日志文件的新窗口。日志文件查看器提供了增强的功能，如过滤记录的信息以及查看汇总和统计信息。
请参见“[MobiLink 服务器日志文件查看器](#)”一节《[MobiLink - 服务器管理](#)》。
- **改进的内存使用** 在 Windows 上，MobiLink 服务器（32 位处理）现在在需要的情况下可使用更多内存。从前，可使用的内存被限制在 2 GB 内。但是现在如果需要更多内存，同时也有更多可用内存，服务器则可以使用多得多的内存。使用 `mlsrv10 -cm` 选项来设置服务器内存高速缓存的最大大小。内存的增加能够减少磁盘分页，从而提高性能。

Sybase Central 的 MobiLink 插件

- **[模型] 模式中的新功能**
 - **表映射** 简化了表映射和列映射的创建和更改。可以从表的 [映射方向] 列的下拉列表中进行选择，来启用表映射。同样，现在可以在 [列映射] 选项卡中指定是否对列进行映射。[新建表映射向导] 现已删除。
请参见“[修改表映射和列映射](#)”一节《[MobiLink - 入门](#)》。
 - **创建新远程表** 基于统一数据库模式创建新远程表更容易。可使用 [创建新远程表] 窗口来创建与统一数据库中的表同名并具有相同列的新远程表。此窗口还将远程数据库表映射到模型的统一表。
请参见“[修改模型创建的远程数据库](#)”一节《[MobiLink - 入门](#)》。
 - **删除远程数据库表和列** 现在可在 [映射] 选项卡中删除远程数据库表和列。在此模型中您可以从远程数据库模式中删除远程表或列，方法是选择表或列，然后选择 [编辑] » [删除]。
请参见“[修改模型创建的远程数据库](#)”一节《[MobiLink - 入门](#)》。

MobiLink 客户端

- **在 Windows CE 上指定网络名称的方式更简便** 您现在可以指定关键字 `default_internet` 或 `default_work` 作为 `network_name` 协议选项中的名称，这样一来缺省设置就是使用的名称。

请参见“[network_name](#)”一节《[MobiLink - 客户端管理](#)》。

- **delete_old_logs 的增强功能** 数据库选项 `delete_old_logs` 现在允许指定多天。然后，此时间段之后创建的日志将会删除。

请参见“[delete_old_logs 选项 \[MobiLink 客户端\] \[SQL Remote\] \[复制代理\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **新挂接** 新挂接 `sp_hook_dbmsync_set_ml_connect_info` 允许在 `dbmsync` 即将尝试连接到 MobiLink 服务器之前，设置网络协议和网络协议选项。

请参见“[sp_hook_dbmsync_set_ml_connect_info](#)”一节《[MobiLink - 客户端管理](#)》。

安全性

新增了两个用于管理传送层安全性的实用程序 `createcert` 和 `viewcert`。请参见：

- “[传送层安全](#)”一节第 103 页
- “[证书实用程序](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》

行为更改和不建议使用的功能

以下列出了 10.0.1 版中引入的 MobiLink 更改。

Sybase Central 的 MobiLink 插件

- **[添加表映射向导] 已删除** 已将新功能添加到 [模型] 模式，以向远程数据库模式添加新表。请参见“[修改表映射和列映射](#)”一节《[MobiLink - 入门](#)》。
- **缺省 dbmsync 批处理文件已改进** 当部署 MobiLink 模型时，创建名为 `model-name_dbmsync.bat` 的文件。以前，此文件中的缺省 `dbmsync` 命令包括 `-qc` 选项，此选项在同步完成后关闭 `dbmsync` 窗口。这意味着很难确定同步是否成功。现在已将 `-qc` 选项从缺省 `dbmsync` 命令行中删除。

安全性

现在不建议使用 TLS 实用程序 `readcert`、`gencert` 和 `reqtool`。替代它们的是实用程序 `createcert` 和 `viewcert`。请参见：

- “[证书实用程序](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》

QAnywhere

以下各节介绍 10.0.1 版的 QAnywhere 中的新功能、行为更改和不建议使用的功能。

新功能

以下列出了 10.0.1 版中引入的 QAnywhere 新增功能。

- **动态寻址** QAnywhere 代理现在可以检测活动网络并自动调整 MobiLink 服务器的通信协议和地址而无需重新启动。
请参见“[-xd 选项](#)”一节《QAnywhere》。
- **最大下载大小** 现在可设置消息下载的最大大小。
请参见“[-idl 选项](#)”一节《QAnywhere》和“[预定义的客户端消息存储库属性](#)”一节《QAnywhere》中的 `ias_MaxDownloadSize`。
- **QAnywhere 服务器日志文件查看器** 添加了一个支持查看 QAnywhere 服务器日志文件的新查看器。日志文件查看器提供了增强的功能，如过滤记录的信息以及查看汇总和统计信息。
请参见“[记录 QAnywhere 服务器](#)”一节《QAnywhere》。

客户端 API 增强

- **能够在处理过程中处理 .NET API 中消息监听器的异常** 已将 `ExceptionHandler` 委派添加到 .NET API。此功能在 Java API 中已存在。
请参见：
 - “[ExceptionHandler 委派](#)”一节《QAnywhere》
 - “[ExceptionHandler2 委派](#)”一节《QAnywhere》
- **向监听器传递消息的所有 QAManager 的能力** 向 .NET 和 Java API 添加了新接口，这些接口方便了在监听器内部调用 `QAManagerBase` API 调用。这很有用，如在确认消息时。新接口不需要引用 `QAManagerBase` 的全局实例或使用其它编码技术来将 `QAManagerBase` 传递到监听器。
请参见：
 - .NET API: “[MessageListener2 委派](#)”一节《QAnywhere》
 - Java API: “[QAMessageListener2 接口](#)”一节《QAnywhere》

行为更改和不建议使用的功能

以下列出了 10.0.1 版中引入的 QAnywhere 更改。

- **客户端消息存储库事务日志** 缺省情况下，客户端消息存储库事务日志的内容现在都将在检查点删除。
请参见“[-m 数据库选项](#)”一节《SQL Anywhere 服务器 - 数据库管理》。

可通过在 `qaagent -c` 选项中指定 `StartLine` 参数更改此行为。
请参见“-c 选项”一节 《QAnywhere》。

SQL Remote

以下各节介绍了 10.0.1 版的 SQL Remote 中的行为更改和不建议使用的功能。

行为更改和不建议使用的功能

以下列出了 10.0.1 版中引入的 SQL Remote 更改。

- **不建议使用 VIM 和 MAPI** 不建议在此版本中使用对 VIM 和 MAPI 消息系统的支持。仍然支持 file、ftp 和 SMTP 消息类型。请参见“[SQL Remote 消息系统](#)”一节《SQL Remote》。

UltraLite

以下各节介绍了 10.0.1 版的 UltraLite 中的新功能、行为更改和不建议使用的功能。

新功能

以下列出了 10.0.1 版中引入的 UltraLite 新增功能。

- **新平台和设备** 此版本现在支持 Windows Vista。
- **改进的 SQL 性能** 从前，如果 UltraLite 查询优化程序确定使用现有索引进行查询没有任何益处，UltraLite 将缺省使用主键索引。

在此版本中，UltraLite 现在将直接从数据库页访问行，而不是使用主键。这样一来，您会看到查询结果的返回顺序与以前版本不同，原因是行不再按主键索引排序。如果需要考虑数据的排序，可在查询中使用 ORDER BY 子句。请参见“使用直接页扫描”一节《UltraLite - 数据库管理和参考》和“使用索引扫描”一节《UltraLite - 数据库管理和参考》。
- **通过 SQL 访问数据库属性和选项** 以前的版本仅允许使用每个 UltraLite API 的方法来访问数据库属性和选项。现在，UltraLite SQL 引入了以下语句和函数，以便能够通过 SQL（包括 Interactive SQL）设置和检索属性和选项：
 - SET OPTION 语句。请参见“UltraLite SET OPTION 语句”一节《UltraLite - 数据库管理和参考》。
 - DB_PROPERTY 函数。请参见“DB_PROPERTY 函数 [System]”一节《UltraLite - 数据库管理和参考》。
- **并发连接的数目已增加** UltraLite 目前支持更多并发连接。单个数据库仍然支持 14 个连接。但是，并发数据库连接的总数已增加：
 - Palm OS 和 Symbian OS 有 8 个数据库连接和 16 个并发连接。
 - 所有其它平台有 32 个数据库连接和 64 个并发连接。

SQLCA 限制可进一步限制连接

能够用于连接引擎的 SQLCA 总数为 31。请记住这个数字，要知道某些组件在每个数据库管理器和每个连接上分别使用一个 SQLCA。这意味着，例如，如果应用程序正在使用 UltraLite.NET API，实际的连接限制减少到总共只有 30 个连接。

- **提交刷新操作** 从前，通过 COMMIT 语句或 API 调用执行的全部提交操作，只有在 UltraLite 将事务安全刷新到存储空间后才能完成。在这个版本中，现在可以配置这些行为，并在逻辑上将它们分离为不同的操作：
 - 现在逻辑提交使您能够在应用程序中回退事务。
 - 检查点目前可在故障出现后提供一个恢复点。这使您能够恢复到上次提交的已刷新到内存的事务。

但是，您还可以增强使用自动提交功能的 UltraLite 应用程序的性能—特别是如果您使用的是组提交刷新。请参见“[备份和恢复 UltraLite 中的数据](#)”一节《[UltraLite - 数据库管理和参考](#)》和“[刷新单个或分组的事务](#)”一节《[UltraLite - 数据库管理和参考](#)》。

这个新行为受以下功能支持：

- CHECKPOINT 语句。请参见“[UltraLite CHECKPOINT 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。

嵌入了 SQL API 和 C++ API 组件的 UltraLite 的 Checkpoint 方法。其它语言必须改用 CHECKPOINT 语句。请参见：

- UltraLite 嵌入式 SQL：“[ULCheckpoint 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》
- UltraLite C++：“[Checkpoint 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》
- COMMIT_FLUSH 连接参数。请参见“[UltraLite COMMIT_FLUSH 连接参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- commit_flush_timeout 和 commit_flush_count 数据库选项。请参见“[UltraLite commit_flush_timeout 选项 \[临时性\]](#)”一节《[UltraLite - 数据库管理和参考](#)》和“[UltraLite commit_flush_count 选项 \[临时性\]](#)”一节《[UltraLite - 数据库管理和参考](#)》。

行为更改和不建议使用的功能

以下列出了 10.0.1 版中引入的 UltraLite 更改。

- **ulafreg** 已将 ulafreg 更改，因此标准输出不再可用。您仍然可以从命令行使用所需选项运行此实用程序。但是，您现在必须单击 [\[编辑\]](#) » [\[复制\]](#) 将输出复制到剪贴板。之后，您可使用所选文本编辑器将它保存到文件。
- **Windows CE 上的 FIPS 支持** 您再也无需运行以下可执行文件来支持 Windows CE 设备上的 FIPS：*install-dir\ultralite\ce\arm\fips\setup.exe*。现在，您只需部署以下文件：*install-dir\ultralite\ce\arm\sbgse2.dll*。

产品范围的功能

以下各节介绍了会影响 10.0.1 版 SQL Anywhere 的所有组件的新功能、行为更改和不建议使用的功能。

新功能

以下列出了 10.0.1 版中引入的产品范围的新增功能。

用于 Oracle 的新 ODBC 驱动程序

现在有一个用于 Oracle 的本地 ODBC 驱动程序（称为 iAnywhere Solutions 10 - Oracle）。

以前，iAnywhere 对第三方创建的 Oracle 驱动程序进行了品牌再造。现在，iAnywhere 拥有了自己的 Oracle ODBC 驱动程序，可供 SQL Anywhere 应用程序使用。使用这个驱动程序，您能够在处理国际字符数据时，更快速地修复错误并且体验功能的改进。如果您使用 Oracle 统一数据库部署 MobiLink 或使用 OMNI 来连接 Oracle，强烈建议您改用此新驱动程序。

请参见“[iAnywhere Solutions Oracle 驱动程序](#)”一节《[MobiLink - 服务器管理](#)》。

传送层安全

- **新证书实用程序** createcert 和 viewcert 是两个新增的实用程序，它们使您能够制作、修改和查看安全证书。以前，实用程序 gencert、reqtool 和 readcert 用于此目的（目前不建议使用这些实用程序）。

Viewcert 允许您查看几种类型的 PKI 对象。以前，只能查看证书。Viewcert 还允许您查看 PEM 和 DER 对象；以前，只能查看 PEM 对象。通过 viewcert，还可以在 PEM 和 DER 间进行转换，以及加密密码或解密密码。

Createcert 将原有的 gencert 和 reqtool 实用程序的功能相结合，并且还提供了新功能。现在创建 ECC 曲线时，可以选择自己的曲线；从前，必须使用 sect163k1。现在可以使用的密钥大小从 512 到 16384 位不等；从前，大小限制为 512 到 2048 位。现在有一个缺省 GUID 序列号；以前，无缺省值。现在还可以选择创建可以签署其它证书的证书。还可以指定确定证书的专用密钥使用方式的高级选项。最后，现在可以使用未加密的专用密钥；以前，全部专用密钥都必须具有口令。

请参见“[证书创建实用程序 \(createcert\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[证书查看器实用程序 \(viewcert\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **在 Windows CE 上已升级 Certicom Security Builder for FIPS 支持** SQL Anywhere 产品可以使用两个均来自 Certicom 的经 FIPS 认可的模块之一进行加密：
 - 在 Palm OS 上，仍必须使用 Security Builder Government Services Edition v1.0.1。
 - 在 Windows CE 上，现在必须使用 Security Builder Government Services Edition v2.0.0，原因是这些库可以签署在 Windows Mobile 上使用。

请参见“[FIPS 认可的加密技术](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

行为更改

以下列出了 10.0.1 版中产品范围内的更改。

传送层安全

- **不建议使用 gencert、readcert 和 reqtool** 现在不建议使用安全实用程序 gencert、reqtool 和 readcert。Gencert 和 reqtool 由 createcert 替代。Readcert 由 viewcert 替代。

请参见“证书实用程序”一节《SQL Anywhere 服务器 - 数据库管理》。

授权

- **服务器授权信息现在存储在 .lic 文件中** 在以前的版本中，SQL Anywhere 个人数据库服务器、SQL Anywhere 网络数据库服务器以及 MobiLink 服务器的授权信息均保存在服务器可执行文件中。此信息现在保存在一个 .lic 文件中，该文件与服务器可执行文件位于相同的目录中。如果可执行文件的 .lic 无法找到，则不会启动此可执行文件。

由于这一更改，a_dblic_info 结构的 exename 成员现在指定可执行文件名或许可文件名。

请参见：

- “服务器授权实用程序 (dblic)”一节《SQL Anywhere 服务器 - 数据库管理》
- “部署数据库服务器”一节《SQL Anywhere 服务器 - 编程》
- “部署 MobiLink 服务器”一节《MobiLink - 服务器管理》
- “a_dblic_info 结构”一节《SQL Anywhere 服务器 - 编程》

Windows Vista 的支持问题

SQL Anywhere 10.0.1 版支持 Windows Vista 操作系统。以下是在 Vista 上运行 SQL Anywhere 软件的一些有关问题：

- **Windows Vista 安全性** Windows Vista 并入了新安全模型。用户帐户控制 (User Account Control, 简称 UAC) 将在默认条件下启用，它可能会影响预期能够写入文件的程序的行为，尤其是在计算机支持多个用户的情况下。一位用户创建的某个文件可能具有禁止其他用户对该文件进行读写的权限，这取决于创建文件和目录的位置和方式。如果在缺省目录中安装了 SQL Anywhere，则会相应建立需要多个用户的读/写访问权限的文件和目录。
- **SQL Anywhere 操作代理已提升** 在 Vista 中，某些操作需要提升权限，才能在用户帐户控制下运行时执行。以下程序在 SQL Anywhere 中可能需要提升：*dbdsn.exe*、*dbelevate10.exe*、*dblic.exe*、*dbsvc.exe*、*installULNet.exe*、*mlasinst.exe*、*SetupVSPackage.exe*、*ulcond10.exe* 和 *ulafreg.exe*。

以下 DLL 在注册或注销时需要提升：*dbctrs10.dll*、*dbodbc10.dll*、*dboledb10.dll* 和 *dboledba10.dll*。

在已激活用户帐户控制的 Vista 系统上，可能会收到 SQL Anywhere 已提升操作代理的提升提示。此提示由 Vista 用户帐户控制系统发出，以确认要继续运行已标识的程序（如果以管理员身份登录）或提供管理员证书（如果以非管理员身份登录）。

- **部署更改** 程序 *dbelevate10.exe* 由 SQL Anywhere 组件内部使用，以执行需要提升权限的操作。这个可执行文件必须包含在 SQL Anywhere 部署中。
- **ActiveSync 支持** Vista 中不支持 Microsoft ActiveSync 实用程序。它由 Windows 移动设备中心所代替。您可以将 SQL Anywhere ActiveSync 提供程序安装实用程序与 Windows 移动设备中心结合使用。
- **SQL Anywhere 可执行文件已签名** Vista 上的 SQL Anywhere 可执行文件由 iAnywhere Solutions, Inc. 签名
- **新许可文件** 10.0.1 的安装包括为 SQL Anywhere 创建新许可文件的过程。现有安装的许可信息会在可执行文件中的原有位置抽取，然后移动到新位置（与可执行文件位于相同目录的 *dbsrv10.lic*、*dbeng10.lic* 和 *mlsrv10.lic* 文件）。
请参见“[服务器授权实用程序 \(dblic\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **示例** 示例现在会正确处理包含一个或多个空格的 SQL Anywhere 安装路径名。
- **Windows 服务** 符合 Vista 的服务不允许与桌面交互。在 Windows Vista 上，没有 SQL Anywhere 服务与桌面交互（即使服务定义中启用了 [允许服务与桌面交互] 也是如此）。可使用 *dbconsole* 实用程序或者从 Sybase Central 来监控 SQL Anywhere 数据库服务器。
Sybase Central 禁用此选项，以使服务在 Windows Vista 上运行时能够与桌面交互。
- **使用 AWE 高速缓存** 要在 Windows Vista 上使用 AWE 高速缓存，必须以管理员身份运行数据库服务器。使用 AWE 高速缓存启动未提升的数据库服务器，将会出现警告，提醒您必须以管理员身份运行数据库服务器才能使用 AWE。请参见“[-cw 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **PowerDesigner、InfoMaker 和 DataWindow .NET** 随 SQL Anywhere 提供的 PowerDesigner、InfoMaker 和 DataWindow .NET 组件在 Windows Vista 上未得到正式支持。因此，在 Vista 环境中运行这些组件时，可能会遇到问题。请参考相应的产品文档，以获得如何在 Vista 环境中运行以及如何获得 Vista 支持的这些产品的版本的说明。

10.0.0 版中的新功能

目录

SQL Anywhere	109
MobiLink	168
QAnywhere	187
SQL Remote	192
UltraLite	194
Sybase Central 和 Interactive SQL	206
文档增强	211
产品范围的功能	212

SQL Anywhere

- “新功能” 一节第 109 页
- “行为更改” 一节第 138 页
- “不建议使用和停用的功能” 一节第 161 页

MobiLink

- “新功能” 一节第 168 页
- “行为更改和不建议使用的功能” 一节第 178 页

QAnywhere

- “新功能” 一节第 187 页
- “行为更改和不建议使用的功能” 一节第 189 页

SQL Remote

- “新功能” 一节第 192 页
- “行为更改和不建议使用的功能” 一节第 192 页

UltraLite

- “新功能” 一节第 194 页
- “行为更改和不建议使用的功能” 一节第 203 页

Sybase Central 和 Interactive SQL

- “新功能” 一节第 206 页
- “行为更改和不建议使用的功能” 一节第 208 页

文档增强

- “文档增强” 一节第 211 页

不建议使用的功能的列表可能会发生更改

与所有前瞻性声明一样，不建议使用的功能的列表不保证其完整性，并可能会发生更改。

SQL Anywhere

以下各部分描述了 10.0.0 版的 SQL Anywhere 中的新功能、行为更改和不建议使用的功能。

注意

在发行版本 10 之前，SQL Anywhere 数据库服务器称为 Adaptive Server Anywhere。

新功能

以下列出了 10.0.0 版中在 SQL Anywhere 数据库和数据库服务器方面新增的功能。

主要功能

- **支持并行机制以提高性能** 数据库服务器目前支持使用多个处理器处理单个查询。当同时执行的查询的数目少于可用处理器的数目时，采用查询内并行机制将带来帮助。请参见“[查询执行期间的并行](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **支持数据库镜像** SQL Anywhere 目前支持数据库镜像，这是一种提高数据库可用性的机制。它需要使用在不同的计算机上运行并在同步或异步模式下相互通信的两台或三台数据库服务器。请参见“[数据库镜像简介](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

添加了以下功能来支持数据库镜像：

- “[synchronize_mirror_on_commit 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
- 数据库服务器的替代服务器名。请参见“[-sn 数据库选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[START DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- ServerName 属性
- AlternateServerName 属性
- RetryConnectionTimeout 属性
- ALTER DATABASE *dbname* FORCE START。请参见“[ALTER DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- MirrorServerDisconnect 和 MirrorFailover 系统事件。请参见“[了解系统事件](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- “[-xf 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
- “[-xp 数据库选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
- SQL Anywhere SNMP Extension Agent 的新陷阱。请参见“[使用陷阱](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- 向 EVENT_PARAMETER 函数添加的 MirrorServerName 参数。请参见“[EVENT_PARAMETER 函数 \[System\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

除数据库镜像之外，SQL Anywhere 现在还为数据库和数据库服务器都提供了 Veritas 聚簇服务器代理（分别是 SADatabase 代理和 SAServer 代理）。请参见“使用 SQL Anywhere Veritas 聚簇服务器代理”一节《SQL Anywhere 服务器 - 数据库管理》。

- **支持快照隔离** 使用快照隔离时，数据库将在用户更改原始数据时保存一份原始数据副本，使想要读取该原始数据的其他用户可以访问它。快照隔离对于用户是完全透明的，并可有助于减少死锁数和锁争用。请参见“快照隔离”一节《SQL Anywhere 服务器 - SQL 的用法》。

添加或增强了以下功能来支持快照隔离：

- “allow_snapshot_isolation 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》
- “isolation_level 选项 [数据库] [兼容性]”一节《SQL Anywhere 服务器 - 数据库管理》
- “sa_snapshots 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》
- “sa_transactions 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》
- LockCount、SnapshotCount、SnapshotIsolationState 和 VersionStorePages 数据库属性
- allow_snapshot_isolation、LockCount 和 SnapshotCount 连接属性
- 版本存储页数性能监控器统计信息
- “SET 语句 [T-SQL]”一节《SQL Anywhere 服务器 - SQL 参考》
- “OPEN 语句 [ESQL] [SP]”一节《SQL Anywhere 服务器 - SQL 参考》
- “ValuePtr 参数”一节《SQL Anywhere 服务器 - SQL 的用法》

- **支持应用程序分析和诊断跟踪** 现有的应用程序分析功能（如存储过程分析和请求记录）已集成到 Sybase Central 的 SQL Anywhere 插件的一个单独统一互动接口中。当您从 Sybase Central 分析应用程序时，将向您提供有助于提高数据库性能的建议。

有关在 Sybase Central 中进行应用程序分析的详细信息，请参见“应用程序分析”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **支持实例化视图** 在某些环境中，数据库较大，并且频繁的查询造成对大量数据的重复性聚合和接合操作，为提高这些环境的性能，SQL Anywhere 现在支持实例化视图。请参见“使用实例化视图”一节《SQL Anywhere 服务器 - SQL 的用法》。

数据库服务器已增强为自动根据开销确定可使用哪些实例化视图应答查询各部分，而不是使用该查询直接引用的基表。请参见“使用实例化视图提高性能”一节《SQL Anywhere 服务器 - SQL 的用法》。

增加了两个新的系统表（ISYSMVOPTION 和 ISYSMVOPTIONNAME）以存储有关实例化视图的信息。请参见“SYSMVOPTION 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》和“SYSMVOPTIONNAME 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》。

- **支持 NCHAR 数据** SQL Anywhere 现在支持 NCHAR 数据类型。NCHAR 数据类型用于存储 Unicode 字符数据。请参见“NCHAR 数据类型”一节《SQL Anywhere 服务器 - SQL 参考》。

添加了以下新函数来支持 NCHAR:

- “UNISTR 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “CONNECTION_EXTENDED_PROPERTY 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “UNICODE 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “NCHAR 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “TO_CHAR 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “TO_NCHAR 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》

以下函数（SORTKEY 和 COMPARE 函数）具有用来支持 NCHAR 数据类型的新参数:

- “SORTKEY 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “COMPARE 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》

SQL Anywhere 现在在使用 Unicode 归类算法 (UCA) 时会将多字节字符集进行正确排序。

初始化实用程序 (dbinit) 和卸载 (dbunload) 实用程序也具有用来支持 NCHAR 数据类型的新选项。请参见“初始化实用程序 (dbinit)”一节 《SQL Anywhere 服务器 - 数据库管理》和“卸载实用程序 (dbunload)”一节 《SQL Anywhere 服务器 - 数据库管理》。

SQL Anywhere 现在使用 Unicode 国际组件 (International Components for Unicode, 简称 ICU) 来支持 Unicode。请参见“国际语言和字符集”一节 《SQL Anywhere 服务器 - 数据库管理》。

为支持 ICU 以及对 NCHAR 数据的处理, 进行了以下属性更改:

- 添加了一个新的 NcharCharSet 数据库和连接扩展属性。此属性返回数据库或连接正在使用的 NCHAR 字符集。
- 添加了一个新的 AccentSensitive 数据库属性。此属性返回区分重音功能的状态。
- CharSet 数据库和连接属性现在是扩展属性。

请参见“数据库属性”一节 《SQL Anywhere 服务器 - 数据库管理》和“连接属性”一节 《SQL Anywhere 服务器 - 数据库管理》。

- **内部性能增强** 为提高数据库服务器性能, 采用了虚拟机技术来重新设计 SQL 表达式的表示和求值, 从而显著提高了吞吐量。
- **支持视图依赖性** 系统表现在存储关于视图依赖性的信息。系统表特别要跟踪数据库中每个视图所依赖的视图、表和列。当您对其视图所依赖的对象进行更改时, 数据库服务器将自动执行附加操作以确保该视图定义不会保留在可能返回不正确结果的状态。请参见“视图依赖性”一节 《SQL Anywhere 服务器 - SQL 的用法》。

添加了两个新表 (ISYSDEPENDENCY 和 ISYSOBJECT) 以存储关于系统对象及其依赖性的信息。请参见“SYSDEPENDENCY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和“SYSOBJECT 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》。

- **改进了检查点算法** 数据库服务器现在可以在发生检查点时启动该检查点并执行其它操作。以前, 在发生检查点时所有活动都将停止。如果某检查点已在进行中, 则启动新检查点的任何操作 (如 ALTER TABLE 或 CREATE INDEX) 都必须等到当前检查点结束才能执行。请参见“了解检查点日志”一节 《SQL Anywhere 服务器 - 数据库管理》。
- **锁定增强** 对于锁定功能进行了以下增强:

- **锁的分类** SQL Anywhere 现在支持四种不同种类的锁：模式锁、表锁、行锁和位置锁。sa_locks 系统过程已修改为更清晰地记录每个事务所持有的锁类型，以更准确地分析锁定问题。请参见“锁定的工作方式”一节《SQL Anywhere 服务器 - SQL 的用法》和“sa_locks 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **支持意图锁** 已针对表锁和行锁引入了一种新类型的锁，称为意图锁。意图锁由应用程序用来表明其要更新某表或该表中的某组行的意图。意图锁目前是在应用程序使用 SELECT FOR UPDATE 语句或 FETCH FOR UPDATE 语句（或它们在各种编程接口中的相应构造）时获取。意图锁会阻塞其它意图锁和写锁，但不阻塞读锁。此支持为那些将锁定用作显式并发控制机制的应用程序提供了更程度的并发性。请参见“使用游标”一节《SQL Anywhere 服务器 - 编程》和“意图锁”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **在某些情况下消除的键范围锁定** 对索引维护算法所做的更改现在允许数据库服务器在单独的索引条目（而不是某一范围的键）上放置写锁。这将改进并发性并消除多种情况下因并发的 INSERT 操作引起的不必要阻塞。请参见“锁定的工作方式”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **索引增强** 在索引功能方面进行了以下增强：
 - **新索引实现** 在 SQL Anywhere 的先前发行版本中包含两个不同的索引实现，它们根据索引列的声明大小自动选择。SQL Anywhere 10 中从头至尾使用的都是压缩 B 树索引的新实现，已淘汰了原有的 B 树索引技术。新索引将压缩格式的索引键值存储在索引条目中，与行中的值分开且不相同。这对快照隔离支持是必需的。
 - **支持快照隔离** 在 SQL Anywhere 的先前发行版本中，索引条目在 UPDATE 或 DELETE 语句上立即删除。为支持快照隔离，有可能若干个索引条目指向含有不同索引键值的同一逻辑行。这些多索引条目由数据库服务器管理，以便任何一个连接都只能看到任何给定行的一个条目。当不再需要这些额外的索引条目时，服务器内部的守护程序会将它们定期物理删除（在事务执行 COMMIT 或 ROLLBACK 时）。保留未提交的 DELETE 的索引条目还提高了 SQL Anywhere 的并发控制机制的语义一致性。请参见“快照隔离”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **改进了 BLOB 存储控制和性能** 现在可以控制存储在表行中（内置）的 BLOB 值的量。还可以控制是否对 BLOB 值进行索引处理。这些增强改进了对 BLOB 的搜索和访问，并通过 CREATE TABLE 语句和 ALTER TABLE 语句中的三个新子句实现：INLINE、PREFIX 和 [NO] INDEX。现在可在同一个表的各行内部或之间共享 BLOB 值，消除了存储重复 BLOB 值的需要，从而降低了存储要求。请参见“存储 BLOB”一节《SQL Anywhere 服务器 - SQL 的用法》、“CREATE TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“ALTER TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **支持列压缩** 现在可以分别压缩表中的各个列。使用 deflate 压缩算法实现压缩。此算法是 COMPRESS 函数使用的同一压缩，也是在 Windows .zip 文件中使用的算法。请参见“CREATE TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“ALTER TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **支持表加密** 您现在可以分开加密数据库中的各个表，而不必加密整个数据库来保护数据。表加密必须在初始化数据库后在数据库中启用。请参见“表加密”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库连接

- **支持单引号或双引号** 连接字符串中的值现在可以用单引号或双引号括起来。这就允许在连接字符串值中使用空格和分号之类的字符。请参见“以连接字符串形式传递的连接参数”一节《SQL Anywhere 服务器 - 数据库管理》和“解决连接参数冲突”一节《SQL Anywhere 服务器 - 数据库管理》。
- **连接字符串现在允许将 T、Y、F 和 N 用作布尔值** 现在可以在指定连接字符串中的连接参数和协议选项时指定 T 或 Y 表示“真”，指定 F 或 N 表示“假”。请参见“连接参数”一节《SQL Anywhere 服务器 - 数据库管理》。
- **某些连接字符串和协议选项现在接受带有 k、m 和 g 后缀的值** 下面的连接参数和协议选项现在将 k、m 和 g 分别作为表示千字节、兆字节和吉字节的后缀接受：
 - “CommBufferSize 连接参数 [CBSIZE]”一节《SQL Anywhere 服务器 - 数据库管理》
 - “CompressionThreshold 连接参数 [COMPTH]”一节《SQL Anywhere 服务器 - 数据库管理》
 - “PrefetchBuffer 连接参数 [PBUF]”一节《SQL Anywhere 服务器 - 数据库管理》
 - “LogMaxSize 协议选项 [LSIZE]”一节《SQL Anywhere 服务器 - 数据库管理》
 - “MaxRequestSize 协议选项 [MAXSIZE]”一节《SQL Anywhere 服务器 - 数据库管理》
 - “ReceiveBufferSize 协议选项 [RCVBUFSZ]”一节《SQL Anywhere 服务器 - 数据库管理》
 - “SendBufferSize 协议选项 [SNDBUFSZ]”一节《SQL Anywhere 服务器 - 数据库管理》
- **AppInfo 返回 Windows 客户端的 IP 地址** 在以前的版本中，AppInfo 连接参数只返回 Unix 客户端和 NetWare 客户端上客户端计算机的 IP 地址。现在还会返回 Windows 客户端的 IP 地址。请参见“AppInfo 连接参数 [APP]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **审计单个连接** 当在登录过程中设置 conn_auditing 临时数据库选项时，该选项允许您对特定连接启用或禁用审计。添加了 auditing 数据库属性以帮助您获取关于数据库审计状态的信息。请参见“conn_auditing 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **RetryConnectionTimeout 连接参数** RetryConnectionTimeout (RetryConnTO) 连接参数指示客户端库在指定时段内重试连接尝试（只要没有找到服务器）。请参见“RetryConnectionTimeout 连接参数 [RetryConnTO]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **支持 IPv6** 现在，Windows、Linux、Mac OS X、Solaris、AIX 和 HP-UX 上都支持 IPv6。在这些操作系统上运行的服务器现在监听所有可用的 IPv4 和 IPv6 地址，并且在客户端或服务端上只要能指定 IP 地址（如 HOST=、MYIP= 和 BROADCAST= TCP 协议选项）的地方，现在就可以指定 IPv6 地址。请参见“SQL Anywhere 中的 IPv6 支持”一节《SQL Anywhere 服务器 - 数据库管理》。
- **用于 LDAP 注册的新参数** read_authdn 和 read_password 这两个参数可用于在数据库服务器为 Active Directory 服务器的情况下将该数据库服务器注册到 LDAP。请参见“使用 LDAP 服务器进行连接”一节《SQL Anywhere 服务器 - 数据库管理》。

备份和恢复

- **为关键数据库页自动计算校验和** 数据库服务器将记录关键数据库页的校验和，而无论是否为该数据库启用了校验和。因此，当您校验数据库时，即使该数据库没有启用校验和，您也可能会

看到关于校验和违规的警告。此外，数据库服务器在试图访问损坏的关键页面时将关闭，并返回一个致命错误。请参见“[校验校验和](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **启动进行恢复时应用多个事务日志** 缺省情况下，恢复数据库时必须按正确顺序分别应用各个事务日志。如果启动数据库服务器时指定了新的 `-ad`、`-ar` 和 `-as` 恢复选项，则无需手工指定将事务日志应用于数据库的顺序。由于在应用事务日志时数据库服务器和数据库正在运行，因此，服务器的高速缓存仍处于预热状态，从而缩短了总恢复时间。请参见“[-ad 数据库选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》、“[-ar 数据库选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[-as 数据库选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **支持并行数据库备份** SQL Anywhere 数据库服务器现在支持服务器端映像备份的并行备份。并行数据库备份利用物理 I/O 来并行（而不是按顺序）执行读写信息，从而提高了性能。可通过以下任一方式执行并行备份：
 - “[备份实用程序 \(dbbackup\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
 - “[BACKUP 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[db_backup 函数](#)”一节《[SQL Anywhere 服务器 - 编程](#)》
- **跟踪有关上次备份的信息** 向 ISYSHISTORY 系统表中添加了新列 LAST_BACKUP 以用于存储有关上次备份的信息。请参见“[SYSHISTORY 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

安全性

本部分介绍了对 SQL Anywhere 所进行的改进安全性的增强。

- **现在随 SQL Anywhere 提供了 RSA** 您不必再去购买单独的许可就可使用 RSA 加密。请参见“[单独授权的组件](#)”一节《[SQL Anywhere 11 - 简介](#)》。
- **在 FIPS 支持方面的增强** 对数据库服务器进行了以下与 FIPS 相关的更改：
 - 已将 FIPS DLL 从 `dbrsa10f.dll` 重命名为 `dbfips10.dll`。
 - HASH 函数现在接受两种新算法：SHA1_FIPS 和 SHA256_FIPS。这两种算法与 SHA1 算法和 SHA256 算法相同，但属于经过 FIPS 校验的 Certicom 版本。
 - 如果指定了 `-fips` 服务器选项并向 HASH 函数提供了一个非 FIPS 算法，则数据库服务器将使用 SHA1_FIPS 而不用 SHA1，使用 SHA256_FIPS 而不用 SHA256，并在使用了 MD5（MD5 不是 FIPS 算法）的情况下返回一个错误。
 - 如果指定了 `-fips` 选项，则数据库服务器将使用 SHA256_FIPS 进行口令散列处理。另外，现在有更多的平台可以使用 `-fips` 选项和 FIPS 功能。要查看支持 `-fips` 选项的平台的列表，请参见“[支持的平台](#)”一节《[SQL Anywhere 11 - 简介](#)》。
- **Kerberos 验证** SQL Anywhere 现在支持 Kerberos 验证。Kerberos 验证允许使用 Kerberos 证书连接到数据库而不必指定用户 ID 或口令。请参见“[Kerberos 验证](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **添加了新权限** 添加了以下权限：
 - **BACKUP 权限** 可以为用户指派 BACKUP 权限（而不是授予用户 DBA 权限）以便他们可以执行备份。请参见“[BACKUP 特权](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **VALIDATE 权限** 添加了一个用于校验操作 VALIDATE 的权限。要执行由不同的 VALIDATE 语句执行的操作（如数据库校验、表校验、索引校验和校验和校验），需要具备 VALIDATE 权限。请参见“VALIDATE 特权”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **保护数据库服务器的功能** -sf 数据库服务器选项允许为数据库服务器上所运行的数据库指定受保护的（禁用的）功能或功能组。请参见“-sf 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- sk 服务器选项与 secure_feature_key 数据库选项联用时，允许您指定一个可用来启用禁用功能的键。您还可以使用 sa_server_option 系统过程 SecureFeatures 属性来更改禁用功能的功能集。请参见“-sk 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库实用程序

- **@filename 可重用于若干个实用程序** 现在可以有选择地为使用参数文件的实用程序分析命令参数文件。分析基于参数文件中放置的简单条件指令完成。请参见“在配置文件中使条件分析”一节《SQL Anywhere 服务器 - 数据库管理》。
- **数据源实用程序 (dbdsn) 增强** 向 dbdsn 实用程序添加了以下选项：
 - **-dr** 在列出用于创建数据源的命令时包含参数 DRIVER=。这使您可以重新创建数据源，以便这些数据源使用另一个版本的 ODBC 驱动程序（而不是当前版本软件随附的 ODBC 驱动程序）。
 - **-f** 显示所使用的系统信息文件的名称（通常为 *.odbc.ini*）。
 - **-ns** 指示 dbdsn 不要搜索系统信息文件（通常为 *.odbc.ini*），而要使用现有环境变量确定该文件应在的位置。当由一个或多个环境变量指定的文件不存在且正在创建 ODBC 数据源时，此选项十分有用。
 - **-pe** 将数据源中的口令字段加密。
 请参见“数据源实用程序 (dbdsn)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **直方图实用程序 (dbhist) 增强** Excel 输出文件内由 dbhist 创建的工作表现在的命名方式能够体现出这些工作表所应用到的列名，而不是 Sheet1、Sheet2 等等。请参见“直方图实用程序 (dbhist)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **信息实用程序 (dbinfo) 增强** -u 选项现在包含关于实例化视图的信息。请参见“信息实用程序 (dbinfo)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **初始化实用程序 (dbinit) 增强** 初始化实用程序 (dbinit) 现在支持以下新选项：
 - **-a** 将区分重音用于 UCA 字符串比较
 - **-af** 将区分法语重音规则用于 UCA 字符串比较。
 - **-dba** 更改新数据库中缺省 DBA 数据库用户的用户 ID 和/或口令。
 - **-dbs** 指定数据库文件的初始大小。
 - **-ze** 指定 CHAR 数据类型的字符集编码。
 - **-zn** 指定 NCHAR 数据类型的归类序列。

请参见“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **日志传送管理器 (LTM) 增强** 将日志传送管理器 (LTM) 实用程序（也称为复制代理）与 Replication Server 15.0 和 Open Server/Open Client 15.0 联用时，该复制代理现在对表名、列名、过程名、函数名和参数名最多支持 128 个字节的标识符。在该软件的早期版本中，标识符长度仅限定于 30 个字节。请参见“复制服务器中的标识符”一节《SQL Anywhere 服务器 - 数据库管理》。

现在，由 dbltm 生成的信息性消息、警告消息和错误消息中的时间戳都使用非模糊的 ISO 8601 日期时间格式：{I|W|E} yyyy-mm-dd hh:mm:ss message。

- **Ping 实用程序 (dbping) 增强** 您可以使用 Ping 实用程序 (dbping) 通过指定 -s 或 -st 选项来获取嵌入式 SQL 连接的工作情况和网络性能的相关信息。这些选项将报告运行 dbping 的计算机与运行数据库服务器的计算机之间的工作情况的相关统计信息。请参见“测试嵌入式 SQL 连接的性能”一节《SQL Anywhere 服务器 - 数据库管理》。

-pd 选项现在允许您指定想要从中获取属性值的数据库的名称。请参见“Ping 实用程序 (dbping)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **服务器枚举实用程序 (dblocate) 增强** 服务器枚举实用程序 (dblocate) 现在支持若干个用于搜索数据库的新选项：

- **-d** 显示服务器名称和地址，以及运行于每个服务器上的所有数据库的列表（以逗号分隔）。
- **-dn** 只有服务器运行具有指定名称的数据库时，才显示服务器名称和地址。
- **-dv** 显示服务器名称和地址，并在单独的行上列出运行于每个服务器上的所有数据库。
- **-p** 显示使用指定 TCP/IP 端口号的服务器。
- **-s** 显示具有指定名称的服务器。
- **-ss** 显示包含指定子串的服务器名称。

请参见“服务器枚举实用程序 (dblocate)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **服务实用程序 (dbsvc) 增强** 服务实用程序 (dbsvc) 支持 DBLTM 服务类型和 dbsln 服务类型，前者允许您管理日志传送管理器的服务，后者允许您管理监听器实用程序的服务。

服务实用程序还支持 -o 选项，此选项允许您将输出数据从该实用程序记录到文件。请参见“用于 Windows 的服务实用程序 (dbsvc)”一节《SQL Anywhere 服务器 - 数据库管理》和“用于 Linux 的服务实用程序 (dbsvc)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新的 SQL Anywhere 广播转发器实用程序 (dbns10)** SQL Anywhere 广播转发器实用程序允许 SQL Anywhere 客户端在其它子网上穿过防火墙（UDP 广播通常不会到达这些位置）查找 SQL Anywhere 数据库服务器，而无需使用 HOST 参数或 LDAP。请参见“广播转发器实用程序 (dbns11)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新的报告提交实用程序 (dbsupport)** 通过新的支持实用程序 (dbsupport) 可以提交错误报告和统计信息、查询更新（EBF 的可用性）并检查是否修复了以前提交的问题。请参见“支持实用程序 (dbsupport)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **卸载实用程序 (dbunload) 增强** 对 dbunload 进行了以下增强：

- 在 `dbunload` 遇到失败时记录未处理的语句。请参见“失败的卸载”一节《SQL Anywhere 服务器 - 数据库管理》。
- 在卸载的表中支持二进制数据。
- 进行了多处增强来提高卸载数据库的性能

添加了以下新选项：

- **-dc** 重新计算数据库中所有计算列的值。
- **-g** 重装期间初始化实例化视图。
- **-k** 创建用于支持跟踪的辅助表。指定此选项可填充 `sa_diagnostic_auxiliary_catalog` 表。此选项在创建跟踪数据库时很有用。
- **-nl** 创建一个 `reload.sql` 文件，该文件包含用于每个表的 `LOAD TABLE` 语句和 `INPUT` 语句，但不包含数据。

请参见“卸载实用程序 (`dbunload`)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **校验实用程序 (`dbvalid`)** 添加了一个新的数据库校验选项 `-d`。此选项执行的数据库校验包括校验和校验、对孤立表页和孤立 BLOB 的检查以及结构检查。对数据不进行检查。请参见“校验实用程序 (`dbvalid`)”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库选项

以下是添加的或已增强功能的数据库选项：

- **ansi_substring 数据库选项** 此选项用于控制 `SUBSTRING` 函数的行为。现在，`SUBSTRING` 函数的缺省行为相当于 ANSI/ISO SQL/99 行为。如果起始偏移为负或零，则视为字符串左侧用非字符填补，并且在提供负值长度的情况下会出现错误。
- **collect_statistics_on_dml_updates 数据库选项** 在执行 DML 语句（`INSERT`、`DELETE` 和 `UPDATE`）期间控制统计信息的收集。请参见“`collect_statistics_on_dml_updates` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **default_dbpace 选项** 此选项允许指定作为表的创建位置的缺省 `dbpace`。请参见“`default_dbpace` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **http_session_timeout 选项** `http_session_timeout` 选项将提供变量会话超时控制。该选项设置以分钟为单位。缺省情况下，公共设置为 30 分钟。最小值为 1 分钟，最大值为 525600 分钟（365 天）。请参见“`http_session_timeout` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **max_temp_space 数据库选项** 您可以使用 `max_temp_space` 选项指定连接可以使用的最大临时空间量。请参见“`max_temp_space` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **materialized_view_optimization 数据库选项** 此选项用于控制优化程序对实例化视图的使用。请参见“`materialized_view_optimization` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **oem_string 数据库选项** 您可以使用 `oem_string` 数据库选项将信息存储在数据库文件的头页中。此字符串可由各应用程序访问并可用于多种用途，如存储版本信息或校验数据库文件是否专用于您的应用程序。请参见“[oem_string 选项 \[数据库\]](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **request_timeout 数据库选项** 此选项允许您指定单个请求可以运行的最长时间，以帮助防止连接长时间耗用大量服务器资源。请参见“[request_timeout 选项 \[数据库\]](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **synchronize_mirror_on_commit 选项** 此选项用于控制何时保证在异步或异步满页模式下运行数据库镜像时，数据库更改已发送到镜像服务器。请参见“[synchronize_mirror_on_commit 选项 \[数据库\]](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **uuid_has_hyphens 数据库选项** 此选项用于控制将 `uniqueidentifier` 的值转换为字符串时所采用的格式。
- **verify_password_function 数据库选项** `verify_password_function` 数据库选项允许指定可用于实现口令规则的函数。该函数在 `GRANT CONNECT` 语句上调用。请参见“[verify_password_function 选项 \[数据库\]](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **用于 Java 支持的新数据库选项** 添加了以下数据库选项：
 - “[java_location 选项 \[数据库\]](#)”一节《SQL Anywhere 服务器 - 数据库管理》
 - “[java_main_userid 选项 \[数据库\]](#)”一节《SQL Anywhere 服务器 - 数据库管理》
 - “[java_vm_options 选项 \[数据库\]](#)”一节《SQL Anywhere 服务器 - 数据库管理》

数据库服务器选项

除了在“新功能”部分中描述的所有服务器选项之外，另外还添加了以下新的服务器选项：

- **-cm 服务器选项** 此服务器选项允许指定在 Windows 上为 Address Windowing Extensions (AWE) 分配的地址空间量。请参见“[-cm 服务器选项](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **-dh 服务器选项** 当针对服务器运行服务器枚举实用程序 (`dblocate`) 时，此服务器选项会使数据库变为不可检测。请参见“[-dh 数据库选项](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **-dt 服务器选项** 此服务器选项允许指定存储临时文件的目录。对于 Unix 上使用共享内存连接的数据库服务器不可以指定此选项。请参见“[-dt 服务器选项](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **-gtc 服务器选项** 此选项允许控制可在 CPU 上并发运行的线程数。请参见“[-gtc 服务器选项](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **-ot 服务器选项** 如果指定此服务器选项，数据库服务器消息日志文件将在有消息写入之前被截断。请参见“[-ot 服务器选项](#)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **-su 服务器选项** 此选项允许在连接到实用程序数据库时为 DBA 用户指定口令。您应使用 `-su` 而不是 `util_db.ini` 文件。请参见“[-su 服务器选项](#)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **-zp 服务器选项** 如果指定此服务器选项，将存储每个连接的最近一次使用的查询优化计划。请参见“-zp 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

属性和性能监控器统计信息

为帮助您管理数据库，添加了几个新的连接属性、服务器属性和数据库属性以及新的性能监控器统计信息。

- **连接属性** 本发行版本中添加了以下连接属性：

- allow_snapshot_isolation
- ansi_substring
- ApproximateCPUTime
- conn_auditing
- default_dbpace
- ExprCacheAbandons
- ExprCacheDropsToReadOnly
- ExprCacheEvicts
- ExprCacheHits
- ExprCacheInserts
- ExprCacheLookups
- ExprCache
- GetData
- HeapsCarver
- HeapsLocked
- HeapsQuery
- HeapsRelocatable
- HttpServiceName
- http_session_timeout
- java_location
- java_main_userid
- LastPlanText
- LockCount
- LockedCursorPages
- LockTableOID
- materialized_view_optimization
- max_query_tasks
- max_temp_space
- MultiPageAllocs
- NcharCharSet
- oem_string
- post_login_procedure
- QueryHeapPages
- ReqCountActive
- ReqCountBlockContention
- ReqCountBlockLock
- ReqCountBlockIO
- ReqCountUnscheduled
- ReqTimeActive
- ReqTimeBlockContention
- ReqTimeBlockIO
- ReqTimeBlockLock
- ReqTimeUnscheduled
- ReqStatus
- RequestsReceived
- RetryConnectionTimeout
- SessionCreateTime

- SessionID
- SessionLastTime
- SnapshotCount
- synchronize_mirror_on_commit
- tsq_l_outer_joins
- verify_password_function

有关连接属性的详细信息，请参见“[连接属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **服务器属性** 本发行版本中添加了以下服务器属性：

- CachePinned
- CacheReadEng
- CacheSizingStatistics
- CarverHeapPages
- ConsoleLogMaxSize
- CollectStatistics
- DebuggingInformation
- DefaultNcharCollation
- DiskReadEng
- ExchangeTasks
- FirstOption
- FunctionMaxParms
- FunctionMinParms
- HeapsRelocatable
- HeapsLocked
- HeapsQuery
- HeapsCarver
- IsEccAvailable
- IsRsaAvailable
- LastConnectionProperty
- LastDatabaseProperty
- LastOption
- LastServerProperty
- MapPhysicalMemoryEng
- MaxConnections
- MultiProgrammingLevel
- NumLogicalProcessors
- NumLogicalProcessorsUsed
- NumPhysicalProcessors
- NumPhysicalProcessorsUsed
- QueryHeapPages
- RememberLastPlan
- RemoteputWait
- RequestFilterConn
- RequestFilterDB
- RequestLogMaxSize
- RequestsReceived
- ServerName
- StartDBPermission

有关这些属性的详细信息，请参见“[数据库服务器属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **数据库属性** 本发行版本中添加了以下数据库属性：

- AccentSensitive
- AlternateServerName
- ArbiterState
- AuditingTypes
- CleanablePagesAdded
- CleanablePagesCleaned
- EncryptionScope
- http_session_timeout
- IOParallelism
- JavaVM
- LockCount
- MirrorState
- NcharCollation
- NcharCharSet
- NextScheduleTime
- PartnerState
- ReceivingTracingFrom
- SendingTracingTo
- SnapshotCount
- SnapshotIsolationState
- VersionStorePages
- XPathCompiles

有关这些属性的详细信息，请参见“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。

● **性能监控器统计信息属性** 本发行版本中添加了以下性能监控器统计信息：

- 高速缓存:多页分配
- 高速缓存:混乱
- 高速缓存:已访问搜寻
- 高速缓存:搜寻
- 高速缓存页:已分配结构
- 高速缓存页:文件
- 高速缓存页:脏文件
- 高速缓存页:空闲
- 通信:收到的请求
- 堆:Carver
- 堆:查询处理
- 堆:被锁可重定位
- 堆:可重定位
- 记忆页:Carver
- 记忆页:已固定游标
- 记忆页:查询处理
- 版本存储页数

有关这些统计信息的详细信息，请参见“性能监控器统计”一节《SQL Anywhere 服务器 - SQL 的用法》。

系统过程和函数

以下是几个新的系统过程和函数以及对现有系统过程和函数的新扩展。

- **对所有过程和函数进行的支持 DEFAULT 子句的增强** 对于过程和用户定义的函数，如果相应参数是使用缺省值定义的，则可以将值 DEFAULT 作为参数提供。如果过程具有若干个参数且使用缺省值的参数不全都在结尾处，则在参数列表中指定 DEFAULT 可能要比使用命名参数更简单。此外，函数调用中不允许使用命名参数。
- **新系统过程** 添加了以下系统过程：
 - **sa_clean_database 系统过程** 按照指定时间运行数据库清理程序。请参见“sa_clean_database 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_column_stats 系统过程** sa_column_stats 系统过程返回指定列的与字符串相关的统计信息。请参见“sa_column_stats 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_conn_list 系统过程** sa_conn_list 系统过程返回连接 ID。请参见“sa_conn_list 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_conn_options 系统过程** sa_conn_options 系统过程返回与数据库选项对应的连接属性的属性信息。请参见“sa_conn_options 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_db_list 系统过程** sa_db_list 系统过程返回数据库 ID。请参见“sa_db_list 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_describe_query 系统过程** sa_describe_query 系统过程对于每列都返回一个数据行，并描述结果表达式的域及其为空性。此过程相当于对每列执行 EXPRTYPE 函数。请参见“sa_describe_query 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_get_bits 系统过程** sa_get_bits 系统过程用于将位字符串解码，对于位字符串中的每个位都返回一个数据行，表示相应位的值。请参见“sa_get_bits 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_make_object 系统过程** 现在可以将事件指定为 sa_make_object 系统过程的一个对象类型。请参见“sa_make_object 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_materialized_view_info 系统过程** sa_materialized_view_info 系统过程返回有关指定的实例化视图的信息，例如视图的状态和所有者。请参见“sa_materialized_view_info 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_refresh_materialized_views 系统过程** sa_refresh_materialized_views 系统过程用于刷新数据库中当前处于未初始化状态的所有实例化视图。请参见“sa_refresh_materialized_views 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_remove_tracing_data 系统过程** 此过程用于从诊断跟踪表中永久删除给定记录会话的所有记录。请参见“sa_remove_tracing_data 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

- **sa_save_trace_data 系统过程** 此过程用于将临时跟踪表中的数据保存到基表。请参见“sa_save_trace_data 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_set_tracing_level 系统过程** 针对要分析的数据库设置要生成的跟踪数据的级别。请参见“sa_set_tracing_level 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_snapshots 系统过程** 返回数据库的当前活动快照的列表。请参见“sa_snapshots 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_split_list 系统过程** 采用一个代表值列表的字符串并返回一个包含该列表的结果集。请参见“sa_split_list 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_table_stats 系统过程** 返回有关每个表中已读取的页数的信息。请参见“sa_table_stats 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_transactions** 返回当前针对数据库运行的事务的列表。请参见“sa_transactions 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_unload_cost_model 和 sa_load_cost_model 系统过程** 现在可以分别使用新系统过程 sa_unload_cost_model 和 sa_load_cost_model 从某一数据库卸载开销模型并将其装载到另一个数据库中。当存在大量类似的硬件安装时，这可以避免重复且耗时的重新校准活动。请参见“sa_unload_cost_model 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》和“sa_load_cost_model 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新增函数** 添加了以下函数：
- **BIT_LENGTH 函数** 返回数组中存储的位数。请参见“BIT_LENGTH 函数 [Bit array]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **BIT_SUBSTR 函数** 返回位数组的子数组。请参见“BIT_SUBSTR 函数 [Bit array]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **BIT_AND 函数** 采用两个位数组并使用下面的逻辑返回其参数的按位逻辑“与”运算结果。对于进行比较的每个位，如果两个位都为 1，则返回 1；否则，返回 0。请参见“BIT_AND 函数 [Aggregate]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **BIT_OR 函数** 采用两个位数组并使用下面的逻辑返回其参数的按位逻辑“或”运算结果。对于进行比较的每个位，如果其中一个位（或两个位）为 1，则返回 1；否则，返回 0。请参见“BIT_OR 函数 [Aggregate]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **BIT_XOR 函数** 采用两个位数组并使用下面的逻辑返回其参数的按位“异或”运算结果。对于进行比较的每个位，如果只有一个位（而不是两个位）为 1，则返回 1；否则，返回 0。请参见“BIT_XOR 函数 [Aggregate]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **COUNT_SET_BITS 函数** 返回数组中设为 1 (TRUE) 的位数的计数。请参见“COUNT_SET_BITS 函数 [Bit array]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **GET_BIT 函数** 返回位数组中指定位的值（1 或 0）。请参见“GET_BIT 函数 [Bit array]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **REVERSE 函数** 此新函数用于返回字符表达式的相反值。请参见“REVERSE 函数 [String]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **SET_BIT 函数** 设置位数组中特定位的值。请参见“SET_BIT 函数 [Bit array]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **SET_BITS 函数** 创建一个位数组，其中与某组行的值相对应的特定位设置为 1 (TRUE)。请参见“[SET_BITS 函数 \[Aggregate\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **TRACED_PLAN 函数** 跟踪查询时，使用跟踪数据和有关优化程序条件的信息为该查询生成一个图形式计划。请参见“[TRACED_PLAN 函数 \[Miscellaneous\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **对各种系统过程和函数的增强** 以下系统过程和函数进行了下述增强：
 - **属性函数增强** 属性函数现在可以返回 LONG VARCHAR。
请参见：
 - “[CONNECTION_PROPERTY 函数 \[System\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[DB_PROPERTY 函数 \[System\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[PROPERTY 函数 \[System\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - **DB_EXTENDED_PROPERTY 函数增强** 现在可以将 DB_EXTENDED_PROPERTY 函数与 NextScheduleTime 数据库属性联用以获取事件的下次调度执行时间。还可以使用该函数返回有关 CHAR 字符集的扩展信息。请参见“[DB_EXTENDED_PROPERTY 函数 \[System\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
 - **新 CONNECTION_EXTENDED_PROPERTY 函数** 可以使用 CONNECTION_EXTENDED_PROPERTY 函数查找某些连接参数的扩展信息。请参见“[CONNECTION_EXTENDED_PROPERTY 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
 - **sa_procedure_profile 系统过程** 现在，sa_procedure_profile 系统过程的输出可以保存到某文件，并应用新语法、需要更少的参数和具有新用途。请参见“[sa_procedure_profile 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
 - **sa_procedure_profile_summary 系统过程** 现在，sa_procedure_profile_summary 系统过程支持将其输出保存到某文件，并应用新语法、接受更少的参数和具有新用途。请参见“[sa_procedure_profile_summary 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
 - **sa_server_option 系统过程** sa_server_option 系统过程允许您在数据库服务器仍然运行时更改它的设置。现在可以更改以下设置：
 - **CacheSizingStatistics 属性** 每当高速缓存大小发生变化时，数据库服务器消息窗口就会显示高速缓存信息。
 - **CollectStatistics 属性** 收集数据库服务器的性能监控器统计信息。
 - **ConsoleLogFile 属性** 指定记录数据库服务器消息窗口信息的输出文件的名称。
 - **ConsoleLogMaxSize 属性** 指定用于记录数据库服务器消息窗口信息的输出文件大小的上限。
 - **DebuggingInformation 属性** 显示诊断性通信消息和其它消息以供故障排除使用。
 - **IdleTimeout 服务器选项** 断开未在指定分钟内提交请求的 TCP/IP 或 SPX 连接。
 - **ProfileFilterConn 属性** 捕获特定连接 ID 的分析信息而不妨碍其它连接使用数据库。
 - **RequestFilterDB 属性** 可以使用 sa_server_option 系统过程过滤与单个数据库建立的用于请求记录的连接。

- **RequestLogging 属性** 请求日志现在可以记录阻塞事件和解除阻塞事件、计划信息、过程以及触发器。

- **RequestTiming 属性** 打开请求计时将指示数据库服务器维护每个请求的计时信息。

有关详细信息，请参见“[sa_server_option 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **xp_startsmtp 系统过程增强** xp_startsmtp 系统过程支持三个新参数：smtp_user_name、smtp_auth_username 和 smtp_auth_password。请参见“[xp_startsmtp 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **xp_sendmail 系统过程增强** xp_sendmail 系统过程现在使用 SMTP 发送邮件时通过新的 include_file 参数来支持附件。此外，xp_sendmail 在使用 SMTP 邮件时通过新的 content_type 参数来支持 MIME 内容。请参见“[xp_sendmail 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **sa_conn_info 系统过程现在返回几个新属性值** sa_conn_info 系统过程现在返回以下附加属性：ClientPort、ServerPort 和 LockTable。该过程不再返回 LastIdle 属性，并且 UncmtOps 值已重命名为 UncommitOps。请参见“[sa_conn_info 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **sa_performance_diagnostics 返回更多信息** 现在使用快照隔离时，sa_performance_diagnostics 系统过程会返回 LockCount 和 SnapshotCount。请参见“[sa_performance_diagnostics 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **HASH 函数增强** HASH 函数现在接受以下新算法：SHA256、SHA1_FIPS 和 SHA256_FIPS。与 FIPS 相关的算法仅供用于使用 FIPS 认证软件的系统。请参见“[HASH 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **COMPRESS 和 DECOMPRESS 函数支持新算法。** gzip 算法现在可供用于压缩和解压缩函数中的字符串。请参见“[COMPRESS 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[DECOMPRESS 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

SQL 语句

以下是几个新 SQL 语句以及现有 SQL 语句语法的新扩展。这些新功能是本文档先前有关功能的部分中所列的语句更改之外的附加内容。

- **支持实例化视图的 SQL 语句** 为支持实例化视图，添加了以下 SQL 语句或对其语法和功能进行了扩展：
 - “[ALTER MATERIALIZED VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[COMMENT 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[CREATE INDEX 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[CREATE MATERIALIZED VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[DROP MATERIALIZED VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[REFRESH MATERIALIZED VIEW 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》
 - “[VALIDATE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》

可使用 SELECT 语句中新的 OPTION 子句来替换 materialized_view_optimization 数据库选项。请参见“SELECT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **支持诊断跟踪和应用程序分析的新 SQL 语句** 下面列出了支持应用程序分析功能的新 SQL 语句：
 - “ATTACH TRACING 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “DETACH TRACING 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “REFRESH TRACING LEVEL 语句”一节《SQL Anywhere 服务器 - SQL 参考》
- **新 VALIDATE DATABASE 语句** 现在可以使用 VALIDATE DATABASE 语句校验数据库。请参见“VALIDATE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新 VALIDATE MATERIALIZED VIEW 语句** 现在可以使用 VALIDATE MATERIALIZED VIEW 语句校验实例化视图。请参见“VALIDATE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新 ALTER STATISTICS 语句** 现在可以使用 ALTER STATISTICS 语句控制是否自动更新列统计信息。使用显式 CREATE STATISTICS 语句或 DROP STATISTICS 语句仍可以对已禁用自动更新的列强制更新统计信息。请参见“ALTER STATISTICS 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER INDEX 语句增强** 现在可以使用 ALTER INDEX 语句的 REBUILD 子句重建一个索引。请参见“ALTER INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER TABLE 和 CREATE TABLE 语句增强** 现在可以使用 MATCH 子句更精细地控制在引用表的外键与被引用表的主键之间构成匹配的元素。您还能够将外键声明为唯一键，从而无需再另外单独声明唯一性。请参见“CREATE TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“ALTER TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER DATABASE 语句的新 CALIBRATE PARALLEL READ 子句** 使用 ALTER DATABASE 语句的新 CALIBRATE PARALLEL READ 子句检测能够进行并行输入和输出的硬件。您可以通过使用 DB_EXTENDED_PROPERTY 函数查询新的 IOParallelism 扩展数据库属性来检索 dbspace 的校准结果。请参见“ALTER DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **COMMENT 语句的新 PRIMARY KEY ON 子句** 现在可以使用 COMMENT 语句的新 PRIMARY KEY ON 子句创建主键的注释。请参见“COMMENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **用于更改加密密钥的 CREATE ENCRYPTED FILE 语句增强** 使用 CREATE ENCRYPTED FILE 语句的扩展，现在可以在不卸载或重装数据库的情况下更改用于加密数据库、事务日志或 dbspace 的加密密钥。如果数据库未加密但启用了表加密，则可以使用 CREATE ENCRYPTED FILE 语句更改用于表加密的密钥。请参见“CREATE ENCRYPTED FILE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **CREATE DATABASE 语句增强** 增加了三个新子句（ENCODING、NCHAR COLLATION 和 ACCENT）以改进对字符集的处理。此外，还添加了 DATABASE SIZE 子句，这样您就可以指定数据库的初始大小。请参见“CREATE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **SELECT 语句增强** FOR UPDATE 子句（通过游标在更新行中使用）已扩展为允许列列表限定哪些列可以使用随后定位的 UPDATE 语句进行修改。请参见“SELECT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

SELECT 语句的 FROM 子句已扩展为支持 READPAST 表提示和 UPDLOCK 表提示，前者指示数据库服务器忽略锁定的行，而后的行为与 XLOCK 类似。请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。

SELECT 语句已扩展为支持 OPTION 子句，以控制该特定语句的查询优化方面。OPTION 子句包含通过此特定 SELECT 语句的 MATERIALIZED VIEW OPTIMIZATION 子句控制实例化视图的匹配的语法。第二个子句 FORCE OPTIMIZATION 指示数据库服务器对查询执行优化，即使该查询符合忽略基于开销的优化的条件。请参见“SELECT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **LOAD TABLE 和 UNLOAD TABLE 语句增强** LOAD TABLE 语句的 STRIP 子句现在接受允许您控制在插入不带引号的值之前是否去除前导空格的选项。其它 STRIP 选项允许您对数据的去除方式进行微调。

LOAD TABLE 语句还扩展为支持 COMMENTS INTRODUCED BY 选项。此选项允许指定用于在输入数据中标识注释的字符串。装载操作期间将忽略输入数据中以指定字符串为开头的任何行。

LOAD TABLE 和 UNLOAD TABLE 语句都已扩展为支持以下选项：

- **ENCODING 选项** 用于指定装载或卸载数据时所使用的编码。
- **ROW DELIMITED BY 选项** 用于指定在批量装载或卸载数据时表示输入记录结尾的字符串。
- **QUOTE 选项** 类似于 Interactive SQL 中 OUTPUT 语句的 QUOTE 选项。请参见“OUTPUT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。

请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“UNLOAD 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **VALIDATE INDEX 语句增强** VALIDATE INDEX 的语法已增强为支持索引说明。请参见“VALIDATE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **用于重命名主键的 ALTER INDEX 语句增强** 现在可以使用 ALTER INDEX 语句重命名主键。请参见“ALTER INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新 CONTINUE 语句** 此语句用于重新启动循环。将跳过循环中 CONTINUE 语句之后的语句。请参见“CONTINUE 语句 [T-SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新 BREAK 语句 [T-SQL]** 此语句用于退出复合语句或循环。请参见“BREAK 语句 [T-SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **用于控制 INSERT 期间缺省值更新的 INSERT 语句增强** 您可以在已存在某行的情况下使用 DEFAULTS ON|OFF 子句控制是否在 INSERT 期间更新缺省值。该新功能没有扩展到以下缺省字段：DEFAULT TIMESTAMP、DEFAULT UTC TIMESTAMP 和 DEFAULT LAST USER；这些字段始终都会更新。请参见“INSERT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **用于支持 ORDER BY 子句的 DELETE 语句增强** DELETE 语句现在支持 ORDER BY 子句，这允许您指定从数据库删除行的顺序。请参见“DELETE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **START DATABASE 语句增强** START DATABASE 语句现在可以在语句无法指明数据库启动失败原因时返回更大范围的错误消息。此外，现在还可以按任意顺序指定 START DATABASE 子句。请参见“START DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **用于支持仅记录到事件或系统日志的 MESSAGE 语句增强** 除了可以打开或关闭记录功能之外，您还可以指定是否仅记录到事件或系统日志。MESSAGE 语句的语法已扩展为允许在 TO LOG 子句内使用可选子句 [EVENT | SYSTEM]。例如，TO EVENT LOG 将导致仅记录到事件日志。请参见“MESSAGE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **FOR OLAP WORKLOAD 选项** CREATE INDEX、CREATE TABLE 和 ALTER TABLE 语句的语法已扩展为在外键定义中支持 FOR OLAP WORKLOAD 选项。此选项指示数据库服务器执行某些优化并收集有关键的统计信息，以提高 OLAP 性能。请参见“CREATE INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》、“CREATE TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》、“ALTER TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“ClusteredHashGroupBy 算法 (GrByHClust)”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **支持临时存储过程** 现在可以使用 CREATE PROCEDURE 语句的扩展创建临时存储过程。临时存储过程仅对创建它们的连接可见，并在该连接删除时会随之自动删除。请参见“CREATE PROCEDURE 语句 (Web 服务)”一节《SQL Anywhere 服务器 - SQL 参考》。
- **支持本地临时表** 现在可以使用 CREATE LOCAL TEMPORARY TABLE 语句创建本地临时表。以此方式创建的本地临时表在连接关闭时将被删除。请参见“CREATE LOCAL TEMPORARY TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **临时表的增强** 现在可以使用 CREATE GLOBAL TEMPORARY TABLE 语句的 SHARE BY ALL 子句创建其中数据可由数据库的所有连接共享的全局临时表。请参见“CREATE TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

数据类型

- **支持 CHAR 和 VARCHAR 数据类型的字符长度语义** 指定 CHAR 或 VARCHAR 列时，现在可以使用字符长度语义。字符长度语义允许您以字符（而不是字节）为单位来表示长度。请参见“CHAR 数据类型”一节《SQL Anywhere 服务器 - SQL 参考》和“VARCHAR 数据类型”一节《SQL Anywhere 服务器 - SQL 参考》。
- **支持位数组数据类型** SQL Anywhere 现在支持 VARBIT 和 LONG VARBIT 数据类型。这些数据类型用于存储位数组。请参见“位数组数据类型”一节《SQL Anywhere 服务器 - SQL 参考》。

添加了以下函数以与位数组数据类型联用：

- “BIT_LENGTH 函数 [Bit array]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “BIT_SUBSTR 函数 [Bit array]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “BIT_AND 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “BIT_OR 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “BIT_XOR 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “COUNT_SET_BITS 函数 [Bit array]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “GET_BIT 函数 [Bit array]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “SET_BIT 函数 [Bit array]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “SET_BITS 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》

编程接口

- **ADO.NET 2.0 支持** ADO.NET 驱动程序已更新为支持 2.0 版的 .NET Framework。为实现此支持，添加了几个新类和新方法。请参见“iAnywhere.Data.SQLAnywhere 命名空间 (.NET 2.0)”一节 《SQL Anywhere 服务器 - 编程》。
- **SQL Anywhere Explorer** SQL Anywhere Explorer 允许您从 Visual Studio .NET 内部连接到 SQL Anywhere 数据库。此外，您还可以直接从 Visual Studio .NET 打开 Sybase Central 和 Interactive SQL。请参见“在 Visual Studio 中使用数据库连接”一节 《SQL Anywhere 服务器 - 编程》。
- **iAnywhere JDBC 驱动程序支持 JDBC 3.0** iAnywhere JDBC 驱动程序现在支持 JDBC 3.0 调用。iAnywhere JDBC 驱动程序不再支持 JDBC 2.0。iAnywhere.ml.jdbcodbc.IDriver 和 iAnywhere.ml.jdbcodbc.jdbc3.IDriver 两个类仍然受到支持，以允许现有应用程序不经过修改即可继续运行，但现在两个驱动程序是相同的，且仅实现 JDBC 3.0。不再可以将早于 1.4 版本的 JRE 与 iAnywhere JDBC 驱动程序一起使用。请参见“JDBC 简介”一节 《SQL Anywhere 服务器 - 编程》。
- **iAnywhere JDBC 驱动程序支持 SQL Server Native Client ODBC 驱动程序** iAnywhere JDBC 驱动程序现在检查 ODBC 驱动程序是否是 Microsoft SQL Server Native Client ODBC 驱动程序并相应地设置缺省的结果集类型和其它属性。
- **支持 PreparedStatement.addBatch 方法** iAnywhere JDBC 驱动程序现在支持 PreparedStatement.addBatch 方法。此方法对于执行成批插入（或大范围插入）十分有用。
- **向 ODBC 驱动程序添加了对 SQL_GUID 的支持** 现在已向 SQL Anywhere ODBC 驱动程序添加了对 UNIQUEIDENTIFIER 列的支持。UNIQUEIDENTIFIER 列现在可以设置为 SQL_GUID 类型。
- **向 ODBC 驱动程序添加了对 GUID 转义序列的支持** 已向 SQL Anywhere ODBC 驱动程序添加了对 GUID 转义序列的支持。可以在通过 ODBC 准备并执行的 SQL 语句中使用 GUID 转义序列。GUID 转义序列的形式为 {guid 'nnnnnnnn-nnnn-nnnn-nnnn-nnnnnnnnnnnnn'}。
- **现在 ODBC 消息回调对应单个连接** 自 Adaptive Server Anywhere 9.0.0 版发行之，ODBC 就一直支持消息回调，但所有连接的消息都使用的是一个单独的回调函数。从 9.0.2 版开始，当您指定某个消息回调函数时，它将只应用于单个连接。这与 DBLIB 的工作原理一致。所有消息现在都要流经 ODBC 驱动程序中的一个单个函数，该函数将按连接过滤这些消息，并仅为具有回调函数的那些连接调用连接的回调函数。

● **向 SQL Anywhere PHP 模块添加的新函数** 向 SQL Anywhere PHP 模块添加了以下新函数：

- sqlanywhere_execute
- sqlanywhere_error
- sqlanywhere_errorcode
- sqlanywhere_insert_id

此外，还向 sqlanywhere_set_option 函数添加了两个新选项：verbose_errors 和 row_counts。请参见“SQL Anywhere PHP API 参考”一节《SQL Anywhere 服务器 - 编程》。

- **db_locate_servers_ex 函数增强** db_locate_servers_ex 函数支持两个新标志：DB_LOOKUP_FLAG_ADDRESS_INCLUDES_PORT 和 DB_LOOKUP_FLAG_DATABASES，前者返回传递给回调函数的 a_server_address 结构中的 TCP/IP 端口号，后者则指明对于找到的每个数据库或数据库服务器调用一次回调函数。请参见“db_locate_servers_ex 函数”一节《SQL Anywhere 服务器 - 编程》。
- **Perl DBI 模块的 Perl DBD::ASAny 驱动程序已重命名** Perl 驱动程序已从 DBD::ASAny 重命名为 DBD::SQLAnywhere。必须将使用 SQL Anywhere 的 Perl 脚本更改为使用新的驱动程序名。返回本地 SQL Anywhere 类型的游标属性 ASATYPE 还未更改，类型名称（ASA_STRING、ASA_FIXCHAR、ASA_LONGVARCHAR 等等）也未更改。请参见“SQL Anywhere Perl DBD::SQLAnywhere DBI 模块”一节《SQL Anywhere 服务器 - 编程》。
- **SQL 预处理器 (sqlpp) -o 选项值** sqlpp -o 选项现在对于 Microsoft Windows 接受的是 WINDOWS 而不是 WINNT。此外，还可以为受支持的 64 位 Unix 操作系统指定 UNIX64。请参见“SQL 预处理器”一节《SQL Anywhere 服务器 - 编程》。

新 ODBC 驱动程序管理器和 ODBC 驱动程序增强

- **ODBC 驱动程序管理器增强** ODBC 驱动程序管理器现在支持：所有 ODBC 3.x 调用、宽 CHAR 入口点、连接跟踪。此外，ODBC 驱动程序管理器现在还能够在线程或线程 SQL Anywhere 驱动程序间进行切换。
- **线程应用程序和非线程应用程序现在都可以使用 ODBC 驱动程序管理器** 线程应用程序和非线程应用程序现在都可以使用 ODBC 驱动程序管理器。

部署

- **部署向导** 添加了 [部署向导] 以用于创建 SQL Anywhere for Windows 的部署。可以使用 [部署向导] 来创建 Microsoft Windows 安装程序包文件和 Microsoft Windows 安装程序合并模块文件。随先前版本 SQL Anywhere 提供的 InstallShield 合并模块和模板将不再提供。将改为使用 [部署向导] 创建 SQL Anywhere 部署。请参见“使用 [部署向导]”一节《SQL Anywhere 服务器 - 编程》。

Windows CE 增强

- **Windows CE 上支持高速缓存大小的动态调整** Windows CE 现在支持高速缓存大小的动态调整。与 Windows 和 Unix 一样，在 Windows CE 上，数据库服务器高速缓存的大小会根据数据库服务器上的负载以及对系统内存的其它需求来进行增减。此功能在许多情况下免除了选择显式高速缓存大小的需要，并且还可以提升性能。

- **为 Windows CE 创建代理端口** 在该软件的先前发行版本中，必须修改注册表中的条目才能将 ActiveSync 配置为使用代理端口连接到 Windows CE 设备上的数据库。Interactive SQL、Sybase Central 和 SQL Anywhere 控制台实用程序的 [连接] 窗口现在包含一个 [设置 Windows CE 代理端口] 工具，该工具允许在不编辑注册表的情况下创建用于连接到 Windows CE 设备上的数据库的代理端口。
- **可以在 Windows CE 上重建数据库** 您现在可以使用一个两步式过程在 Windows CE 上重建数据库，在此过程中使用 dbunload 实用程序将数据库卸载到文件，然后再将其重装到新数据库中。请参见“在 Windows Mobile 上重建数据库”一节《SQL Anywhere 服务器 - 数据库管理》。
- **dbrunsql 实用程序** 脚本执行实用程序 (dbrunsql) 允许您在 Windows CE 上键入 SQL 命令或运行命令文件。请参见“脚本执行实用程序 (dbrunsql)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Windows Mobile 5 的已签名 .cab 文件** SQL Anywhere 安装包包含由 VeriSign 签名的预置的 .cab 文件。使用这些 .cab 文件时，不会接收到未知发布者警告。

Unix/Linux 增强

- **可在 Unix 平台上使用的新 ODBC 驱动程序管理器** 现在可以在 Unix 平台上将 libdbodm10 共享对象用作 ODBC 驱动程序管理器。使用 iAnywhere ODBC 驱动程序管理器的应用程序必须将使用的 ODBC 限定于 3.0 版及更高版本。请参见“在 Unix 上使用 SQL Anywhere ODBC 驱动程序管理器”一节《SQL Anywhere 服务器 - 编程》。
- **-uf 服务器选项** -uf 选项允许指定在 Unix 上发生致命错误时数据库服务器要执行的操作。请参见“-uf 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Linux 现在支持服务实用程序** 现在可在 Linux 上使用服务实用程序来创建、删除、列出、启动和停止 SQL Anywhere 服务。
- **Unix 支持的附加示例** Unix 现在支持以下 SQL Anywhere 示例：
 - **DBTools** 此示例是一个通过备份 SQL Anywhere 示例数据库来说明如何调用和编译数据库工具库的数据库工具应用程序。
 - **DiskFull** 此示例说明了一个 [磁盘已满] 回调 DLL 示例。
 - **HTTP** HTTP 目录中的示例演示了各种 Web 服务功能，包括如何使用 Web 服务在 SQL 过程内设置和检索客户端 cookie、如何在 HTTP Web 服务过程内处理二进制数据、如何使用窗体和 HTML 表显示简单日历、如何使用 xp_read_file 从本地磁盘检索映像并将它们作为对 HTTP 请求的响应返回，以及如何创建、使用和删除 HTTP 会话。
 - **oemString** 此示例说明如何确定是否为 OEM 软件设置了数据库文件。
 - **PerformanceFetch** 此示例说明如何使用 fetchtest 测试随机查询的读取速率。
 - **PerformanceInsert** 此示例说明如何使用 instest 测试向某表插入数据的插入速率。
- **Linux 上对线程的支持（线程关联度）和改进的并发处理** SQL Anywhere for Linux 数据库服务器引进了一个新的协同例程处理模型，类似于 Windows 线程的处理模型。此处理模型使服务器可以更好地控制各任务/例程间的上下文切换，实现了与数据库操作之间更好的关联度。

- **对 Linux 的直接 I/O 支持** SQL Anywhere for Linux 现在支持 Linux 2.6 O_DIRECT 功能，此功能可以提高 I/O 性能，因为文件系统不会对 I/O 进行高速缓存处理。
- **对 Linux 的异步 I/O 支持** SQL Anywhere for Linux 现在支持 Linux AIO 功能，通过此功能，即使单个应用程序线程也可以将 I/O 操作与其它提高 IO 性能的处理重叠进行。
- **Linux 桌面操作系统 GUI** SQL Anywhere for Linux 现在为个人数据库服务器、网络数据库服务器、MobiLink 服务器、MobiLink 同步客户端和 SQL Remote 提供了可选的桌面操作系统 GUI。如果安装了 GTK 库，则可以用 -ui 选项调用此 GUI。
- **Linux 桌面操作系统图标** SQL Anywhere for Linux 现在针对 Linux 桌面操作系统提供了可选安装图标，使运行 Linux 桌面操作系统的用户更便于启动和管理数据库服务器、Sybase Central、Interactive SQL、SQL Anywhere 控制台实用程序和 MobiLink 监控器。
- **对 IBM AIX、HP-UX 和 Sun Solaris 的 64 位客户端支持** SQL Anywhere 客户端库现在可供 64 位内存模型使用，使您在 IBM AIX、HP-UX 和 Sun Solaris 上可以利用应用程序内更大的内存。

Web 服务

- **Web 服务器符合 HTTP 1.1** 为遵从 HTTP 1.1，Web 服务器现在接受以下项目：
 - HTTP 请求的管道操作，实现了多个 HTTP 请求（如 GET 和 HEAD）的同步处理
 - 绝对 URI（以前仅支持相对 URI）
 - 100-continue 请求标头字段，使客户端在发送整个请求主体之前可以确定服务器能否接受请求（根据请求标头）。
 - Accept-Charset 请求标头字段中的 quality 值（这些值以前被忽略）
- **HTTP 客户端符合 HTTP 1.1** 现在实现了以下与 HTTP 相关的增强：
 - **支持 HTTP 字符串内存池机制** HTTP 字符串不再存储在连续内存中。高速缓存被用作后台存储区。
 - **客户端块模式** HTTP 客户端现在可以使用 HTTP 分块模式发送 POST 请求。
 - **HTTP 会话** HTTP 连接可创建一个 HTTP 会话以维护 HTTP 请求之间的状态。
- **HTTP 服务器支持保持活动状态** 数据库服务器现在在 HTTP 客户端发出请求时支持保持活动状态选项。HTTP 连接将在每个请求和响应后保持打开状态，而不再在每个请求后关闭连接，从而可以在同一连接上执行多个请求。请参见“[处理 HTTP 标头](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。

已添加了 KeepaliveTimeout 协议选项，用于支持此功能。请参见“[KeepaliveTimeout 协议选项 \[KTO\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **新 HttpServiceName 连接属性** 已添加了新的连接属性 HttpServiceName，从而使 Web 应用程序能够确定其服务名起源。该属性对于错误报告和控制流很有用。请参见“[连接属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **sa_set_http_option 增强** 现在可以基于请求 Accept-Charset 请求标头字段使用 sa_set_http_option 系统过程控制 HTTP 响应中使用的字符集。请参见“[sa_set_http_option 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **支持 SOAP 服务的数据键入** 对 CREATE SERVICE 和 ALTER SERVICE 语句进行了扩展，可支持新的 DATATYPE 子句。此子句仅与 SOAP 服务一起使用，并且控制数据键入是否支持输入参数和输出响应。请参见“CREATE SERVICE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“ALTER SERVICE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_set_soap_header 系统过程** 使用 sa_set_soap_header 系统过程为 SOAP 服务设置响应标头。请参见“sa_set_soap_header 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **SOAP_HEADER 和 NEXT_SOAP_HEADER 函数** 使用 SOAP_HEADER 函数获取 SOAP 服务的请求标头。请参见“SOAP_HEADER 函数 [SOAP]”一节《SQL Anywhere 服务器 - SQL 参考》。
使用 NEXT_SOAP_HEADER 函数获取 SOAP 标头中的下一个标头条目。请参见“NEXT_SOAP_HEADER 函数 [SOAP]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **CREATE PROCEDURE、ALTER PROCEDURE、CREATE FUNCTION 和 ALTER FUNCTION 语句中的 HEADER 子句** 已将新的 HEADER 子句添加到上述语句中，以便在创建 HTTP Web 服务客户端过程和函数时使用。此子句允许您添加或修改 HTTP 请求标头条目。
请参见“CREATE PROCEDURE 语句 (Web 服务)”一节《SQL Anywhere 服务器 - SQL 参考》、“CREATE FUNCTION 语句 (Web 服务)”一节《SQL Anywhere 服务器 - SQL 参考》和“处理 HTTP 标头”一节《SQL Anywhere 服务器 - 编程》。
- **CREATE PROCEDURE、ALTER PROCEDURE、CREATE FUNCTION 和 ALTER FUNCTION 语句中的 SOAPHEADER 子句** 已将新的 SOAPHEADER 子句添加到上述语句中，以便在创建 SOAP Web 服务客户端过程和函数时使用。此子句允许您使用 IN (IN/OUT) 替代参数来指定发送的 SOAP 标头条目和接收的 SOAP 标头数据。
请参见“CREATE PROCEDURE 语句 (Web 服务)”一节《SQL Anywhere 服务器 - SQL 参考》、“CREATE FUNCTION 语句 (Web 服务)”一节《SQL Anywhere 服务器 - SQL 参考》和“处理 SOAP 标头”一节《SQL Anywhere 服务器 - 编程》。

杂项

- **索引增强** 为了在此版本中进行索引，进行了以下的增强：
 - **支持索引共享** 在创建主键、辅助键、外键或唯一约束时，现在可以创建指向物理索引（磁盘上的实际索引结构）的逻辑索引。数据库服务器自动确定是否需要新的物理索引来满足逻辑索引。此模型允许共享物理索引并防止创建和维护重复的物理索引，重复的物理索引会浪费磁盘空间。请参见“使用逻辑索引共享索引”一节《SQL Anywhere 服务器 - SQL 用法》。
 - **改进了索引信息的存储** 对数据库中索引信息的组织方法有几种改进。例如，所有索引（包括主键索引和外键索引）的列表现在存储在单个系统表 ISYSIDX 中。
三个新的系统表 ISYSPHYSIDX、ISYSIDXCOL 和 ISYSFKEY 提供与 ISYSIDX 中列出的索引有关的附加信息。请参见：

- “目录中的索引信息”一节 《SQL Anywhere 服务器 - SQL 的用法》
 - “SYSIDX 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “SYSPHYSIDX 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “SYSIDXCOL 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
 - “SYSFKEY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- **索引顾问增强** 对索引顾问进行了增强，提高了针对聚簇索引、负载中的数据库和服务器状态以及完整的负载统计信息报告的建议水平。该功能已集成到了 [应用程序分析] 工具中。
- **改进了对索引创建方式的控制** 当应用程序创建参照完整性约束（主键约束、外键约束或唯一约束）时，数据库服务器通过在组成约束键的各列上隐式创建一个索引来强制执行该约束。数据库服务器现在允许您指定该索引的创建方式。您可以指定约束键中各列的顺序以及索引中每一列的值的排序方式（升序或降序）。此外，不要求外键中的列的顺序和排序方式与相应的主键或唯一约束一致。
- 其它的功能改进包括：
- 现在不必对表中各列重新排序即可更改主键顺序。
 - 可按照应用程序要求指定所有约束索引中的列的排序方式。
 - 现在可按照应用程序对外键表的要求定制外键索引而不必考虑主表设计。
 - 外键现在可以具有唯一约束。
- **新增了外连接消除重写优化** 如果结果查询在语义上等同于原始查询，外连接将在执行前从查询中消除。请参见“语义查询转换”一节 《SQL Anywhere 服务器 - SQL 的用法》。
 - **日期格式字符串现在使用字符长度的语义** 日期格式字符串现在使用字符长度的语义控制替代格式指定符的文本的数量；以前，使用字节长度的语义。例如，当格式日期使用字符串时，过去表示使用 3 个字节存储月份的 MMM，现在则表示使用 3 个字符来存储月份。
 - **目录访问服务器** 现在可以通过创建目录访问服务器来创建一个远程服务器，该服务器用来访问运行数据库服务器的计算机的目录结构。请参见“使用目录访问服务器”一节 《SQL Anywhere 服务器 - SQL 的用法》。
 - **挪威语归类** 添加了 1252NOR，支持挪威语。在挪威语 Windows 系统中，如果未指定任何归类，则数据库服务器将选择 1252NOR 作为新数据库的缺省归类。请参见“支持的归类和替代归类”一节 《SQL Anywhere 服务器 - 数据库管理》。
 - **UTF8BIN 归类** 添加了 UTF8BIN 归类，提供二进制数据的改进的排序。此新归类替换现在已不建议使用的 UTF8 归类。请参见“支持的归类和替代归类”一节 《SQL Anywhere 服务器 - 数据库管理》。
 - **数据库服务器消息窗口增强** 针对数据库服务器消息窗口，进行了以下增强：
 - **窗口标题栏的新右击选择** 现在，在所有支持的 Windows 平台（Windows CE 除外）上，右击数据库服务器消息窗口的标题栏时，可以选择 [关于] 或 [清除消息区域]。选择 [关于] 将显示关于数据库服务器的信息，而选择 [清除消息区域] 将消除数据库服务器消息窗口中的所有消息。此窗口（数据库服务器消息日志文件、Sybase Central 的 [服务器消息和已执行的 SQL] 窗格和 SQL Anywhere 控制台实用程序）的复本不受清除操作的影响。

- **可以将数据库服务器使用的环境变量记录到数据库服务器消息窗口中** `-ze` 服务器选项在数据库服务器消息窗口中显示数据库服务器的环境变量列表。此功能不能用于 NetWare 或 Windows CE。请参见“[-ze 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **启动时控制窗口最小化** 缺省情况下，数据库服务器消息窗口在数据库服务器启动后会最小化。如果不想让数据库服务器消息窗口在数据库服务器启动后最小化，可以指定 `-qn` 选项。请参见“[-qn 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **能够跟踪表上次更新的时间** 数据库服务器现在能够跟踪表上次更新的时间。使用 SYSTAB 系统视图中的新 `last_modified_at` 列可以实现这一点。请参见“[SYSTAB 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **镜像过程中更改为另一台服务器时发送 SNMP 陷阱** 现在，SQL Anywhere SNMP Extension Agent 在其连接到镜像所涉及的服务器上时发送陷阱，原连接断开，重新建立一个连接到其它服务器的新连接。
此陷阱表示原始服务器已关闭，而充当镜像的服务器成为主服务器。请参见“[使用陷阱](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **对请求记录进行的更改** 现在以逗号分隔的文本格式存储请求日志，从而将其减少到约原始大小的三分之一。此外，只要符合条件，现在就不再采用正常的时间条目来记录时间，而是将时间记录为一个等号 (=)，这表示与日志中前一条目相同，或记录为 `+nnn`，其中，`nnn` 是日志中前一条目之后的毫秒数。现在还记录附加信息。例如，对于查询，现在记录的信息有隔离级别、读取的行数和游标类型。对于 INSERT、UPDATE 和 DELETE 语句，现在记录受影响的行数和触发的触发器数。请参见“[请求记录](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
`sa_get_request_times` 系统过程仅支持新的请求日志格式。但是，`tracetime` Perl 脚本 `tracetime.pl` 可处理旧请求日志格式和新请求日志格式。`tracetime` 脚本还可以在新格式的日志上执行得更快，而且在大的请求日志上更是如此。
- **ODBC 驱动程序增强** 在 SQL Anywhere 连接到 Adaptive Server Enterprise 和 DB2 数据库时，所使用的是用于远程数据访问的新驱动程序。请参见：“[对 MobiLink、QAnywhere 和远程数据访问所使用的 ODBC 驱动程序进行的更改](#)”一节第 214 页。
- **SQLANYSAMP10 环境变量** SQLANYSAMP10 环境变量指定包含 SQL Anywhere 10 示例的目录的位置，其中包括 `demo.db` 和 `custdb.db` 示例数据库。请参见“[SQLANYSAMP11 环境变量](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

版本支持

本节包含对支持的和不支持的第三方组件的版本号进行的更改。

- **支持版本 8 和更早的数据库** 使用 Sybase Central、Interactive SQL 或 SQL Anywhere 控制台实用程序时，将不再支持使用早于 Adaptive Server Anywhere 8.0.0 的软件版本创建的数据库。这包括使用旧版软件创建再使用新版软件进行升级的数据库。尝试装载此类数据库将导致数据库启动时产生错误。可以从 Sybase Central 连接到这些数据库，以将它们卸载到新版 10 数据库中。请参见“[升级 SQL Anywhere](#)”一节第 324 页。
- **支持 jConnect 版本 5.5 和 6.0.5** SQL Anywhere 现在支持 jConnect 版本 5.5 和 6.0.5（9.0.2 支持版本 5.5）连接到数据库服务器。jConnect 可以通过 <http://www.sybase.com/products/>

[informationmanagement/softwaredeveloperkit/jconnect](#) 作为单独的下载使用。有关所支持功能的信息，请查阅 [jConnect](#) 文档。

- **Intel x86 体系结构上的处理器要求** 在 Intel x86 体系结构上，SQL Anywhere 仅支持 Pentium 类和更新的处理器，在以前的处理器（例如 80386 或 80486）上将不能启动。

行为更改

以下列出了 10.0.0 版中引入的对 SQL Anywhere 数据库的更改（根据类别分组）。

杂项

- **重命名了 Adaptive Server Anywhere** 在 10.0.0 版中，Adaptive Server Anywhere 已重命名为 SQL Anywhere。
- **升级更改** [升级数据库] 向导、升级实用程序 (dbupgrad) 和 ALTER DATABASE UPGRADE 语句不能用于将版本 9.0.2 和更早版本的数据库升级到版本 10。要将早期版本的数据库升级到版本 10，必须通过执行卸载和重装来重建数据库。请参见“[升级 SQL Anywhere](#)”一节第 324 页。
- **口令更改** 在新建的数据库中，所有口令均区分大小写，与数据库是否区分大小写无关。新数据库的缺省 DBA 口令是 **sql**。

重建现有数据库时，按以下条件确定口令的大小写：

- 如果口令最初是在不区分大小写的数据库中输入的，则目前的口令依然不区分大小写。
- 如果口令最初是在区分大小写的数据库中输入的，则大写和混合大小写的口令依然区分大小写。不过，如果口令全部以小写输入，则口令不区分大小写。
- 对现有口令和新口令进行的更改均区分大小写。

数据库服务器现在使用 SHA256 来散列口令。旧（专有）散列算法仍然支持从旧数据库重装的口令，但所有新口令将使用 SHA256。

现在口令的存储格式是 UTF-8，因此如果数据库被重装入带有不同字符集的数据库时，口令将继续有效。

在先前的版本中，当从嵌入式 SQL 进行连接时，有可能使用 DBA 权限连接到数据库，然后在不指定口令的情况下，为任何用户成功建立到同一数据库的第二个连接。现在必须指定每个连接的口令。

- **空格填充更改** 在 SQL Anywhere 先前的版本中，带有空格填充的数据库的字符串比较的语义就好像是用无限数量的空格填充两个正在比较的字符串。在 10 版中，对这些语义进行了更改，从而可以通过忽略每个字符串中的尾随空白来执行这种比较。

对于等于 (=) 和不等 (<>) 比较，语义方面没有变化；两种技术（空格填充和忽略尾随空白）会生成相同的结果。但是，对于不等式比较是有差别的。例如，假定您有双字节字符串值 'a*'，其中 '*' 代表数据库归类中的一个字符，该字符小于空格的值。在 SQL Anywhere 先前的版本中，比较谓语句 'a*' < 'a' 返回 TRUE。在版本 10 中，该谓语句生成 FALSE，这是因为较短的字符串在被比较之前并没有用空格进行填充。

有关空格填充的详细信息，请参见“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》和“CREATE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **属性返回值的大小写** 由 PROPERTY 函数返回的服务器属性在以前的版本中返回 YES 或 NO，现在则返回 Yes 或 No。由 DB_PROPERTY 函数返回数据库属性和由 CONNECTION_PROPERTY 函数返回的连接属性在以前的版本中返回 ON 或 OFF，现在则返回 On 或 Off。此更改可能会影响区分大小写的数据库或应用程序，它们使用区分大小写的字符串比较。
- **对服务器属性返回值进行的更改** 在先前的版本中，服务器属性 ConnsDisabled 和 RememberLastStatement 返回值 ON 和 OFF。它们现在返回值 Yes 和 No。请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **sasrv.ini 文件的缺省位置已更改** 在 Windows 上 *sasrv.ini* 的缺省位置是 %ALLUSERSPROFILE%\Application Data\SQL Anywhere 10，而在 Unix 上为 ~/.sqlanywhere10。在先前的版本中，该 Unix 文件称为 .sasrv.ini。在所有的平台上，该文件现在均称为 *sasrv.ini*。
- **连接到名称较长的数据库服务器** 在 Windows 和 Unix 上，9.0.2 版及更早版本的客户端无法连接到名称长于 40 个字节的 10.0.0 版和更高版本的数据库服务器。
- **数据库服务器始终启用字符集转换** 不再支持用于启用和禁用字符集转换的 -ct 数据库服务器选项。数据库服务器始终启用字符集转换，但如果数据库服务器决定不再需要此功能，则不再使用它。指定 CharSet=none，就可以从客户端禁用字符集转换。请参见“CharSet 连接参数 [CS]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Windows CE 不支持字符集转换** Windows CE 不支持字符集转换。在先前的版本中，Windows CE 的数据库服务器禁用字符集转换，因此您可以对数据库使用任意字符集。现在，您必须使用操作系统字符集或 UTF-8 创建 Windows CE 的数据库。请参见“安装注意事项：在 Windows Mobile 上使用 ICU”一节《SQL Anywhere 服务器 - 数据库管理》。
- **对系统过程和函数进行的更改** 以下是对系统过程和函数进行的更改的列表：
 - **若干系统过程已成为内部过程** 外部系统过程 xp_read_file、xp_write_file、xp_sprintf、xp_scanf 和 xp_cmdshell 现在是内部系统过程。
 - **sa_validate 系统过程** sa_validate 系统过程现在需要 VALIDATE 权限。请参见“sa_validate 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_reset_identity 系统过程** 现在需要 table-name 参数。此外，如果不指定 owner-name 参数，则 table-name 参数必须唯一标识数据库中的某个表。请参见“sa_reset_identity 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **sa_locks 系统过程** sa_locks 系统过程的输出已更改为返回其它信息，其中包括连接 ID、用户 ID、表名、锁类和锁持续时间。请参见“sa_locks 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **RAND 函数** 以前，每个连接都带有相同的种子值，从而 RAND 函数将为每个连接返回相同的序列。现在，每个连接都被提供唯一的种子，从而每个连接将得到不同的随机序列。请参见“RAND 函数 [Numeric]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **DB_CALLBACK_START 和 DB_CALLBACK_FINISH 回调函数** 现在，所有的平台都支持 DB_CALLBACK_START 和 DB_CALLBACK_FINISH 回调函数（以前，只有 Windows 平台

才支持它们)。请参见“[db_register_a_callback 函数](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。

- **分散读取不再用于使用 UNC 名称指定的文件** 分散读取不再用于远程计算机上的文件，也不再用于使用 UNC 名称（如 `\\mycomputer\myshare\mydb.db`）指定的文件。请参见“[使用适当的页大小](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **主键和外键约束中的列排序** 创建主键约束时，可以指定列的任意顺序，与列在表中出现的顺序无关。另外，倘若指定了外键列和主键列之间的映射，现在还可以创建外键列顺序不同于其所引用的主键的外键。请参见“[CREATE TABLE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》中的 PRIMARY KEY 子句。
- **索引中不允许有重复的列名称** 以前，索引中允许对列的重复引用（主键、外键和唯一约束说明除外）。现在，该行为在所有类型的索引中都一致；指定重复列名称就会返回一个错误。此外，如果较旧数据库包含一个带有重复列引用的索引，则 `dbunload` 实用程序将在生成 `reload.sql` 时删除该索引的重复列。请参见“[CREATE TABLE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **Encryption 数据库属性** 现在，执行 `SELECT DB_PROPERTY('Encryption')` 可能会返回 None 以外的值，即使在数据库未加密的情况下也是如此。这种情况将在为数据库启用表加密时发生。如果您的应用程序执行此命令，将其作为检查数据库是否加密的方法，则将改用 `SELECT DB_PROPERTY('EncryptionScope')`。请参见“[数据库属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **使用 FIPS 启动 HTTPS 的语法中的更改** 以前，您将指定 `-xs HTTPS_FIPS(...)`。现在，您必须指定 `-xs HTTPS(FIPS=yes;...)`。仍然支持以前的语法，但不建议使用。请参见“[-xs 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **用户 ID 的最大长度为 128 字节** 在先前的版本中，当某语句需要用户 ID 时，数据库服务器将在使用长度超过 128 个字节的用户 ID 之前，先将其截断。如果设置了 `string_rtruncation` 选项，将返回一个截断错误。现在，如果指定了长度超过 128 个字节的用户 ID，则无论 `string_rtruncation` 选项的设置是什么，数据库服务器都将返回一个错误。请参见“[标识符](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **服务器名称的最大长度** TCP/IP 和共享内存连接上的数据库服务器名称的最大长度已经从 40 字节增加到 250 字节。请参见“[-n 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **对标识符中可接受字符进行的更改** 标识符中不再允许使用双引号和反斜线。请参见“[标识符](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **重命名了 LicensesInUse 服务器属性** 服务器属性 `LicensesInUse` 已重命名为 `UniqueClientAddresses`。请参见“[数据库服务器属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **SQL Anywhere OLE DB 提供程序名称已更改** SQL Anywhere OLE DB 提供程序（以前有 `ASAProv`、`ASAProv.90`、`ASAProv.80`）现在称为 `SAOLEDB`。版本 10 提供程序可以由名称 `SAOLEDB.10` 专门引用。
- **SQL Anywhere 示例数据库 ODBC DSN 已更改** ODBC 数据源（以前是 `ASA 9.0 Sample`）现在称为 `SQL Anywhere 10 Demo`。
- **对连接字符串进行的更改** 对于 ODBC 和 OLE DB 连接，现在，连接参数的优先级为：连接字符串、`SQLCONNECT` 环境变量、数据源。以前，在 ODBC 和 OLE DB 中，数据源的优先级高

于 SQLCONNECT。请参见“连接参数的语法规则”一节《SQL Anywhere 服务器 - 数据库管理》。

- **现在空值连接参数视为未指定的参数** 对于所有的 API，使用空值指定的连接参数被视为未指定该参数。在先前的版本中，根据指定空值的位置和所使用的 API，将空值视为未指定的参数或一个空字符串。请参见“连接参数的语法规则”一节《SQL Anywhere 服务器 - 数据库管理》。
- **当开启审计时无法关闭事务日志** 在该软件先前的版本中，当某数据库的审计已打开时，您可以停止该数据库的事务日志。现在，如果打开了数据库的审计，就必须使用事务日志。如果想要停止使用事务日志，则必须关闭审计。
- **不能以只读模式启动打开了审计的数据库** 在该软件先前的版本中，您可以以只读模式启动打开了审计的数据库。现在，不能以只读模式启动开启了审计的数据库。
- **现在带符号 BIGINT 列的精度是 19 而不是 20** 以前，当 ODBC 应用程序使用 SQL_BIGINT 描述带符号的 BIGINT 列时，会返回精度值 20，这个值并不正确。现在，则返回值 19。您需要更改任何依赖先前（不正确的）值的应用程序。
- **Java VM 增强** SQL Anywhere 不再将 Java 选项作为单独授权的组件提供。数据库中的 Java 现在使用外部 VM（而不是使用内部 VM）运行 Java 代码。因此，您现在可以使用任何想要的 Java VM，并且不再局限于特定的 JDK 版本或 Java 目标。新初始化的数据库始终启用 Java。

这将产生以下的更改：

- **不支持的数据库选项** SQL Anywhere 不再支持以下的选项：

- describe_java_format
- java_heap_size
- java_namespace_size
- java_page_buffer_size
- java_input_output
- return_java_as_string

- **不支持的属性** 不再支持以下的属性：

- 数据库属性：

- JDKVersion
- JavaHeapSize
- JavaNSSize

- 数据库服务器属性：

- IsJavaAvailable
- JavaGlobFix

- 连接属性：

- JavaHeapSize
- java_input_output

- **新的 JavaVM 属性** JavaVM 数据库属性返回数据库服务器在数据库中执行 Java 时所使用的 Java VM 的路径。

- **不支持的兼容性视图列** 以下的列不再可用于系统兼容性视图：
 - SYSINFO.classes_version
 - SYSJAVACLASS.replaced_by
 - SYSJAVACLASS.type_id
- **数据库实用程序不建议使用的 Java 选项** 以下的数据库实用程序选项已不建议使用：
 - 初始化实用程序 (dbinit): -ja, -jdk
 - 卸载实用程序 (dbunload): -jr
 - 升级实用程序 (dbupgrad): -ja, -jdk, -jr, -j
- **Java 支持 CREATE DATABASE 和 ALTER DATABASE 语句中不建议使用的某些与 Java 相关的子句** CREATE DATABASE 语句不再支持 JAVA ON|OFF 和 JDK 版本子句, ALTER DATABASE 语句不再支持 REMOVE JAVA 子句。
- **新的 Java 文件** 除了所提到的更改外, 还增加了以下文件: *java\sajvm.jar*。
- **Ping 实用程序 (dbping)** 以前, 如果数据库服务器返回的某属性值为 NULL, 则 Ping 实用程序 (dbping) 报告一个错误。现在, dbping 在属性值未知时输出 NULL 并退出, 同时返回成功返回代码。如果想要 dbping 在属性值未知时退出并返回失败的返回代码, 可以指定 -en 选项。请参见“[Ping 实用程序 \(dbping\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **重命名的环境变量** 此版本已重命名了以下的环境变量:

以前的名称	新名称
ASTMP	SATMP
ASDIR	SADIR
ASLOGDIR	SALOGDIR
ASLANG	SALANG
ASCHARSET	SACHARSET

- **对 PHP 模块文件名称进行的更改** PHP 模块文件的命名约定已更改。在先前的版本中, 这些文件命名为 *phpX_sqlanywhereY.dll*, 其中 *X* 为 PHP 主版本号, *Y* 为 SQL Anywhere 主版本号。现在, PHP 模块文件命名为 *php-a.b.c_sqlanywhereY.dll*, 其中 *a.b.c* 是构建文件时依据的 PHP 源的完全版本号, 而 *Y* 是 SQL Anywhere 主版本号。例如, *php-5.0.2_sqlanywhere10.dll*。
- **指定 PrefetchBuffer 连接参数的值** 为了向后兼容, PrefetchBuffer 连接参数现在将小于 16384 的值解释为千字节。不建议使用没有 k 后缀的千字节。如果因为 PrefetchBuffer 的值在有效范围之外或在没有 k 后缀的千字节中进行定义而使该值得到了调整, 则客户端日志文件将显示使用的实际 PrefetchBuffer 值。请参见“[PrefetchBuffer 连接参数 \[PBUF\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **无法删除系统定义的域** 不再可以从数据库中删除系统定义的域 (如 MONEY 或 UNIQUEIDENTIFIERSTR)。请参见“[DROP DOMAIN 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **对数据库实用程序进行的更改** 以下为对数据库实用程序所做的更改的列表，如下所述：
 - **服务实用程序 (dbsvc) 可以授予 "作为服务登录" 的特权。** 如果使用 -a 选项，并且尝试用没有启用 "作为服务登录" 的特权的帐户运行服务，则服务实用程序 (dbsvc) 将提示您授予 "作为服务登录" 的特权。如果使用 -y 选项，则 dbsvc 将尝试授予 "作为服务登录" 的特权而不提示您。请参见“用于 Windows 的服务实用程序 (dbsvc)”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **可以针对远程服务器使用卸载实用程序 (dbunload) -an 选项** 在进行此更改之前，您可以仅针对同一计算机上的服务器运行 dbunload -an。现在可以针对不同计算机上运行的服务器运行 dbunload -an。请参见“卸载实用程序 (dbunload)”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **服务器枚举实用程序 (dblocate) 主机名或 IP 地址格式** 主机名或 IP 地址可以使用任何格式，与是否指定 -n 无关。例如，如果某个服务器运行在 myhost.mycompany.com（其 IP 地址为 1.2.3.4）上，要从使用 mycompany.com 域的任何计算机中仅列出在此计算机上运行的服务器，则可以使用 dblocate myhost、dblocate myhost.mycompany.com 或 dblocate 1.2.3.4 中的任意一个。在先前的版本中，只有 dblocate myhost.mycompany.com 或 dblocate -n 1.2.3.4 会起作用，因为给定的主机名或 IP 地址必须匹配 dblocate 所显示的地址字符串（不包括端口号）。请参见“服务器枚举实用程序 (dblocate)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **缺省相关的更改** 对缺省设置进行了以下的更改：
 - **为个人数据库服务器更改了缺省 TCP/IP 监听地址** 在 Windows 上，个人数据库服务器现在监听 127.0.0.1（而不是 0.0.0.0）上的连接。此更改意味着运行 SQL Anywhere 且启用了 Windows 防火墙的用户在能够使用 dbeng10 之前无需将其添加到异常列表中。

由于这一更改，同 LINKS=tcpip (HOST=hostname; DOBROADCAST=none) 相连接的尝试在 *hostname* 是计算机的实际主机名或 IP 地址的情况下将不起作用。但是，使用主机名 **localhost** 或 **127.0.0.1** 将有效。
 - **缺省数据库页面大小更改为 4096** SQL Anywhere 数据库的缺省数据库页面大小已从 2048 字节更改为 4096 字节。此页面大小在许多环境中已证实能提高性能。请参见“CREATE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

如果没有指定 -gp 选项并在启动数据库服务器没有装载数据库，则数据库服务器的缺省页面大小为 4096。
 - **对缺省最大高速缓存大小进行的更改** 已增加了 Windows（非 AWE）的缺省最大高速缓存大小。现在将缺省最大高速缓存大小限制为下列各项中较小者：
 - 90% of (total_physical_memory - 4 MB)，但不小于 2 MB
 - (available address space - 512 MB)
 - **Unix 高速缓存大小** 计算 Unix 上最大高速缓存大小的方法已更改。现在，缺省的最大高速缓存大小的计算方法如下：
 - 在 32 位 Unix 平台上，它是总物理内存的 90% 或 1,834,880 KB 的较小者。
 - 在 64 位 Unix 平台上，它是总物理内存的 90% 和 8,589,672,320 KB 的较小者。

请参见“限制高速缓存内存使用”一节《SQL Anywhere 服务器 - SQL 的用法》和“-ch 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **Unix 存储过程** 升级现有的 Unix 应用程序时，如果您使用的是 64 位数据库服务器，则必须将任何现有的外部存储过程更改为 64 位。
- **将 NULL 常量转换为 NUMERIC 或字符串数据类型时的缺省大小** 将 NULL 常量转换为 NUMERIC 数据类型或字符串数据类型（如 CHAR 和 VARCHAR）时，此长度现在设置为 0（而不是 32767）。
- **openxml 系统过程的缺省 URI 已更改** 使用 openxml 系统过程时，如果未指定命名空间声明，则缺省情况下将前缀 mp 绑定到统一资源定位符 (URI)。在该软件的先前版本中，此 URI 为 urn:ianywhere-com:asa-xpath-metaprop。缺省 URI 已重命名为 urn:ianywhere-com:saxpath-metaprop。请参见“openxml 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **对服务器选项 -c、-ch 和 -cl 的高速缓存大小百分比计算进行的更改** 使用 -c、-ch 或 -cl 的 P（百分比）时，系统现在将针对物理系统内存的数量或可用地址空间的数量（取较低者）计算百分比。这就消除了为高速缓存分配的内存多于寻址可用的内存的风险。请参见“-c 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》、“-ch 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》和“-cl 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **重命名了 Procedure_profiling 服务器选项** 用于控制过程分析的服务器选项的正确名称现在为 ProcedureProfiling。仍接受以前的格式 Procedure_profiling，但未来的版本将不会支持它。请参见“sa_server_option 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **在 HP-UX 上，当连接到不使用缺省端口的数据库服务器时，TCP/IP 端口号无需由客户端指定** 在该软件先前的版本中，如果在 HP-UX 上启动数据库服务器，而且缺省端口 (2638) 已被使用或不想使用此缺省端口，则必须使用 ServerPort [PORT] 协议选项来指定一个端口号。

在 HP-UX 上，当在一台计算机上启动多台数据库服务器时，就不再需要 TCP/IP ServerPort 协议选项。在 Mac OS X 上，当启动网络服务器时如果已有一个服务器在同一计算机上运行，则仍必须指定 TCP/IP ServerPort 选项。请参见“ServerPort 协议选项 [PORT]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **SOAP CONCRETE 响应从 ASADataset 重命名为 SimpleDataset** CONCRETE 响应已从 ASADataset 重命名为 SimpleDataset。请参见“CREATE SERVICE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **对 [卸载数据库向导] 行为进行的更改** 不再可以将数据库卸载到早于版本 10 的数据库版本中。在将版本 9.0.2 或更早版本的数据库卸载到版本 10 数据库时，一旦完成了重建，您就无法自动连接到数据库。
- **对 [抽取数据库向导] 行为进行的更改** 无法抽取版本 9.0.2 和更早版本的数据库。必须从版本 10 数据中抽取。
- **在 Solaris 上不支持服务器选项 -ui 和 -ux** 在 Solaris 上不再支持 -ui 和 -ux 选项。它们在 Linux 上仍然可用。
- **转换数值数据类型** 将 DOUBLE 类型转换为 NUMERIC 时，SQL Anywhere 现在使用一种更加精确的算法（近似于原始的 DOUBLE 值）。有了这些更改，具有 15 个或更少有效位的 DOUBLE 值将被精确地转换为 NUMERIC。在某些情况下，这可能会导致不同于先前 SQL Anywhere 版本的答案。请参见“数字集之间的转换”一节《SQL Anywhere 服务器 - SQL 参考》。

- **对 sa_validate 系统过程进行的更改** 不再需要 sa_validate 系统过程的数据、索引和完整选项，也不建议使用它们。除非您请求的是快速校验或校验和校验，否则现在将缺省执行使用以前的数据、索引和完整选项执行的检查。请参见“sa_validate 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **对 a_validate_type 枚举进行的更改** 不再需要 a_validate_type 枚举的 VALIDATE_DATA、VALIDATE_INDEX 和 VALIDATE_FULL 参数，也不建议使用它们。当指定 VALIDATE_NORMAL 时，现在将缺省执行由这些选项执行的校验。请参见“a_validate_type 枚举”一节《SQL Anywhere 服务器 - 编程》。
- **SQLPATH 环境变量语法更改** SQLPATH 环境变量的语法在 Unix 上已发生更改。在以前的版本中，所有操作系统的路径元素都由分号 (;) 分隔。在 SQL Anywhere 10 中，在 Unix 平台上由冒号 (;) 分隔路径元素，在其它平台上由分号分隔。

对数据库选项进行的更改

- **区分大小写选项和数据库选项** SET OPTION 语句和 CONNECTION_PROPERTY 函数使用的选项名称区分大小写。但是，在使用土耳其语归类或区分大小写的数据库中，查询中引用的选项名称应使用“按字母顺序排列的选项列表”一节《SQL Anywhere 服务器 - 数据库管理》中指定的大小写进行书写。

在这两种情况中的任何一种情况下，如果选项名称使用错误的大小写，在 SYSOPTION 上执行查询或执行类似以下的查询都可能匹配不到任何行：

```
SELECT *
FROM sa_conn_properties()
WHERE propname = 'BLOCKING'
```

- **使用嵌入式 SQL 时将 ansi_blanks 设置为 On** 对于嵌入式 SQL，如果 ansi_blanks 设置为 On 并且数据库为空格所填充，则在提供数据类型为 DT_STRING 的值时，必须将 sqlen 字段设置为包含该值的缓冲区的长度（至少为该值的长度加上终止空字符的空间）。
数据库以空白填补时，如果正在读取的表达式是 CHAR 或 NCHAR（不是 VARCHAR 或 NVARCHAR）并且正在被读取到 char 或 nchar（不是 VARCHAR 或 NVARCHAR）主机变量，则 ansi_blanks 选项控制发送到客户端的截断警告。请参见“ansi_blanks 选项 [兼容性]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **已更改 ansi_integer_overflow 选项缺省设置** 创建新数据库时，ansi_integer_overflow 数据库选项的缺省值为 On。在该软件先前的版本中，此选项的缺省值为 Off。
- **对 date_format 选项进行的更改** 指定格式字符串时，date_format 选项不再支持以下值：
 - hh 两位数表示的小时数
 - nn 两位数表示的分钟数
 - ss[.ss.] 秒及一秒的小数部分
 - aa 上午/下午指示符（上午或下午，12 小时制）
 - aaa[a...] 上午/下午指示符（上午或下午，12 小时制）
 - pp 下午指示符，如有必要（下午，12 小时制）

- **ppp[p...]** 下午指示符，如有必要（下午，12 小时制）

此外，如果字符数据为多字节，则现在每个符号的长度都反映字符数。例如，'mmm' 符号指定月份的长度为三个字符。在先前的版本中，符号的长度反映的是字节数。请参见“[date_format 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **login_mode 数据库选项** 对 login_mode 数据库选项不建议使用值 [Mixed]。指定 [Standard]、[Integrated] 以允许标准登录和集成登录。请参见“[login_mode 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **更改了 string_rtruncation 选项缺省设置** 创建新数据库时，string_rtruncation 数据库选项的缺省值为 On。在该软件先前的版本中，此选项的缺省值为 Off。请参见“[string_rtruncation 选项 \[兼容性\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
如果使用 CAST 函数来截断字符串，则必须将 string_rtruncation 数据库选项设置为 Off；否则会出错。
建议您使用 LEFT 函数来截断字符串。请参见“[LEFT 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **更改了 temp_space_limit_check 选项缺省设置** temp_space_limit_check 选项的缺省设置已更改为 On。现在，缺省情况下，如果连接请求的临时文件空间量超出其限额，则请求将失败并返回 SQLSTATE_TEMP_SPACE_LIMIT 错误。请参见“[temp_space_limit_check 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **对 timestamp_format 选项进行的更改** timestamp_format 选项不再支持使用法式日期和月份。此外，如果字符数据为多字节，则现在每个符号的长度都反映字符数。例如，'mmm' 符号指定月份的长度为三个字符。在先前的版本中，符号的长度反映的是字节数。请参见“[timestamp_format 选项 \[兼容性\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **已删除 truncate_date_values 选项** 已删除 truncate_date_values 选项。在先前的版本中，此选项允许在使用 DATE 数据类型定义的列中包括时间。在此版本中，使用 DATE 定义的列只能包含日期。如果想要存储日期和时间，请使用 TIMESTAMP 数据类型。请参见“[TIMESTAMP 数据类型](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

对服务器选项进行的更改

- **用于高度加密类型的 -ec 服务器选项和 -xs 服务器选项 TLS 语法已更改** -ec 服务器选项、-xs 服务器选项和 Encryption 连接参数中高度加密类型的语法已更改。现在只有 3 种可用类型：none、simple 和 tls。代替指定密钥交换算法用作加密类型，现在指定 tls 作为加密类型并使用新的协议选项 tls_type 来指定算法。请参见“[-ec 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》、“[-xs 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[Encryption 连接参数 \[ENC\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **-os 服务器选项** 在先前的版本中，-os 数据库服务器选项将日志文件重命名为 *current-filename.old*。现在这是 -on 选项的行为。-os 数据库服务器选项现在指定输出日志的最大大小，达到此大小时将重命名日志。以前，使用 -os 将产生两个日志文件，但现在将不限制日志文件的数目。请参见“[-os 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[-on 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

对分类目录进行的更改

10.0.0 版中分类目录已经进行了较大更改。最明显的更改是系统表的名称前增加了一个 I。如果尝试访问系统表，您将接收到权限被拒绝的错误。通过系统视图可以提供系统表中的信息。每个系统表都有一个系统视图，而且，为了向后兼容，系统视图名称与来自 SQL Anywhere 先前版本的表名称相符。例如，在版本 9.0.2 中，有一个名为 SYS.SYSARTICLE 的系统表。在版本 10.0.0 中，该系统表现在名为 SYS.ISYSARTICLE，有一个相应的系统视图 SYS.SYSARTICLE。

该分类目录现在还包含统一视图。这些统一视图提供来自两个或更多表或视图的通常需要的连接。大多数统一视图在先前的版本中显示为系统视图。

某些系统表和系统视图已不建议使用或已从分类目录中删除。不过，在多数情况下提供了兼容性视图。

下表提供了从 Adaptive Server Anywhere 9.0.2 到 SQL Anywhere 10.0.0 的目录的完整映射。第一列 **9.0.2 系统表/视图** 显示了 9.0.2 系统表的名称，后跟一条正斜线 (/)，斜线后是 9.0.2 关联视图的名称。中间列 **10.0.0 系统表** 包含 10.0.0 表名。最后列 **10.0.0 系统视图** 包含关联的 10.0.0 视图名称以及兼容性说明。

注意

任意列中的横线 (-) 都表示没有对应的对象。例如，10.0.0 版本的分类目录中的一个新表将在 9.0.2 列中的相应表位置对应一个横线。

9.0.2 系统表/视图	10.0.0 系统表	10.0.0 系统视图
DUMMY / -	DUMMY	-
RowGenerator / -	RowGenerator	-
SYSARTICLE / SYSARTICLES	ISYSARTICLE	“SYSARTICLE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 用于 10.0.0 以前的版本的兼容性： “SYSARTICLES 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》。
SYSARTICLECOL / SYSARTICLECOL	ISYSARTICLECOL	“SYSARTICLECOL 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 用于 10.0.0 以前的版本的兼容性： “SYSARTICLECOLS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》。
SYSATTRIBUTE / -	ISYSATTRIBUTE	-
SYSATTRIBUTENAME / -	ISYSATTRIBUTENAME	-

9.0.2 系统表/视图	10.0.0 系统表	10.0.0 系统视图
SYSCAPABILITY / SYSCAPABILITIES	ISYSCAPABILITY	“SYSCAPABILITY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 “SYSCAPABILITIES 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSCAPABILITYNAME / -	ISYSCAPABILITYNAME	“SYSCAPABILITYNAME 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSCATALOG		“SYSCATALOG 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSCHECK / -	ISYSCHECK	“SYSCHECK 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSCOLAUTH	-	“SYSCOLAUTH 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSCOLLATION / -	-	“SYSCOLLATION 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSCOLLATIONMAPPINGS / -	-	“SYSCOLLATIONMAPPINGS 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSCOLPERM / -	ISYSCOLPERM	“SYSCOLPERM 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSCOLSTAT / SYSCOLSTATS	ISYSCOLSTAT	“SYSCOLSTAT 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSCOLSTATS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSCOLUMN / SYSCOLUMNS	ISYSTABCOL	“SYSTABCOL 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSCOLUMNS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》 用于 10.0.0 以前的版本的兼容性： “SYSCOLUMN 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSCONSTRAINT / -	ISYSCONSTRAINT	“SYSCONSTRAINT 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》

9.0.2 系统表/视图	10.0.0 系统表	10.0.0 系统视图
- / -	ISYSDEPENDENCY	“SYSDEPENDENCY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSDOMAIN / -	ISYSDOMAIN	“SYSDOMAIN 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSEVENT / -	ISYSEVENT	“SYSEVENT 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSEVENTTYPE / -	ISYSEVENTTYPE	“SYSEVENTTYPE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSEXTENT / -	-	-
SYSEXTERNLOGINS / -	ISYSEXTERNLOGIN	“SYSEXTERNLOGIN 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSFILE / -	ISYSFILE	“SYSFILE 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSFKCOL / -	ISYSIDXCOL	“SYSIDXCOL 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 用于 10.0.0 以前的版本的兼容性： “SYSFKCOL 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSFOREIGNKEY / SYSFOREIGNKEYS	ISYSFKEY	“SYSFKEY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 和 “SYSFOREIGNKEYS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》。 用于 10.0.0 以前的版本的兼容性： “SYSFOREIGNKEY 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSGROUPS	ISYSGROUP	“SYSGROUP 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 和 “SYSGROUPS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSHISTORY / -	ISYSHISTORY	“SYSHISTORY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》

9.0.2 系统表/视图	10.0.0 系统表	10.0.0 系统视图
SYSINDEX / SYSINDEXES	ISYSIDX	“SYSIDX 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSINDEXES 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》 用于 10.0.0 以前的版本的兼容性： “SYSINDEX 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSINFO / -	-	“SYSINFO 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSIXCOL / -	ISYSIXCOL	“SYSIXCOL 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 用于 10.0.0 以前的版本的兼容性： “SYSIXCOL 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSJAR / -	ISYSJAR	“SYSJAR 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSJARCOMPONENT / -	ISYSJARCOMPONEN T	“SYSJARCOMPONENT 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSJAVACLASS / -	ISYSJAVACLASS	“SYSJAVACLASS 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSLOGIN / -	ISYSLOGINMAP	“SYSLOGINMAP 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSOPTBLOCK / -	-	仅系统使用
- / -	ISYSMVOPTION	“SYSMVOPTION 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / -	ISYSMVOPTIONNAM E	“SYSMVOPTIONNAME 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / -	ISYSOBJECT	“SYSOBJECT 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSOPTION / SYSOPTIONS	ISYSOPTION	“SYSOPTION 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSOPTIONS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》

9.0.2 系统表/视图	10.0.0 系统表	10.0.0 系统视图
SYSOPTJOINSTRATEGY / SYSOPTJOINSTRATEGIES	-	仅系统使用
SYSOPTORDER / SYSOPTORDERS	-	仅系统使用
SYSOPTQUANTIFIER / -	-	仅系统使用
SYSOPTREQUEST / -	-	仅系统使用
SYSOPTREWRITE / -	-	仅系统使用
SYSOPTSTAT / -	ISYSOPTSTAT	“SYSOPTSTAT 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
-	ISYSPHYSIDX	“SYSPHYSIDX 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSPROCAUTH	-	“SYSPROCAUTH 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSPROCEDURE / SYSPROCEDURES	ISYSPROCEDURE	“SYSPROCEDURE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 SYSPROCEDURES 视图已重命名为 SYSPROCS。请参见“SYSPROCS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》。
SYSROCPARM / SYSPROCPARMS	ISYSROCPARM	“SYSPROCPARM 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSPROCPARMS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSROCPERM / -	ISYSROCPERM	“SYSPROCPERM 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
-	ISYSROXYTAB	“SYSPROXYTAB 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》

9.0.2 系统表/视图	10.0.0 系统表	10.0.0 系统视图
SYS PUBLICATION / SYS PUBLICATIONS	ISYS PUBLICATION	“SYS PUBLICATION 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYS PUBLICATIONS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / -	ISYS REMARK	“SYS REMARK 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYS REMOTE OPTION / SYS REMOTE OPTIONS 、 SYS REMOTE OPTION2	ISYS REMOTE OPTIO N	“SYS REMOTE OPTION 系统视图”一节 《SQL Anywhere 服务器 - SQL 参 考》、“SYS REMOTE OPTION2 统一视 图”一节 《SQL Anywhere 服务器 - SQL 参 考》和 “SYS REMOTE OPTIONS 统一视 图”一节 《SQL Anywhere 服务器 - SQL 参 考》
SYS REMOTE OPTION T YPE / -	ISYS REMOTE OPTIO N TYPE	“SYS REMOTE OPTION TYPE 系统视图”一 节 《SQL Anywhere 服务器 - SQL 参考》
SYS REMOTE TYPE / SYS REMOTE TYPES	ISYS REMOTE TYPE	“SYS REMOTE TYPE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYS REMOTE TYPES 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYS REMOTE USER / SYS REMOTE USERS	ISYS REMOTE USER	“SYS REMOTE USER 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYS REMOTE USERS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYS SCHEDULE / -	ISYS SCHEDULE	“SYS SCHEDULE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYS SERVERS / -	ISYS SERVER	“SYS SERVER 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / -	ISYS SOURCE	“SYS SOURCE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYS SQL SERVER TYPE / -	ISYS SQL SERVER TYP E	“SYS SQL SERVER TYPE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYS SUBSCRIPTION / SYS SUBSCRIPTIONS	ISYS SUBSCRIPTION	“SYS SUBSCRIPTION 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYS SUBSCRIPTIONS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》

9.0.2 系统表/视图	10.0.0 系统表	10.0.0 系统视图
SYSSYNC / SYSSYNCS、 SYSSYNC2	ISYSSYNC	“SYSSYNC 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》、“SYSSYNC2 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSSYNC2 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
-	ISYSSYNCSCRIPT	“SYSSYNCSCRIPT 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSSYNCSCRIPTS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSSYNCSUBSCRIPTIONS	-	“SYSSYNCSUBSCRIPTIONS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSSYNCUSERS	-	“SYSSYNCUSERS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSTABAUTH	-	“SYSTABAUTH 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSTABLE / -	ISYSTAB	“SYSTAB 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 用于 10.0.0 以前的版本的兼容性： “SYSTABLE 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
-	ISYSTABCOL	“SYSTABCOL 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSTABLEPERM / -	ISYSTABLEPERM	“SYSTABLEPERM 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSTRIGGER / SYSTRIGGERS	ISYSTRIGGER	“SYSTRIGGER 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSTRIGGERS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSTYPEMAP / -	ISYSTYPEMAP	“SYSTYPEMAP 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
-	ISYSUSER	“SYSUSER 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》

9.0.2 系统表/视图	10.0.0 系统表	10.0.0 系统视图
- / SYSUSERAUTH	ISYSUSERAUTHORITY	“SYSUSERAUTHORITY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSUSERAUTH 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSUSERLIST		“SYSUSERAUTHORITY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSUSERLIST 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSUSERMESSAGES / -	ISYSUSERMESSAGE	“SYSUSERMESSAGE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSUSEROPTIONS	-	“SYSUSEROPTIONS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSUSERPERM / SYSUSERPERMS	-	数据现在位于 ISYSUSER 和 ISYSUSERAUTHORITY 系统表中。请参见：“SYSUSER 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSUSERAUTHORITY 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》 用于 10.0.0 以前的版本的兼容性：“SYSUSERPERM 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSUSERPERMS 兼容性视图（不建议使用）”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSUSERTYPE / -	ISYSUSERTYPE	“SYSUSERTYPE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》
- / SYSVIEWS	ISYSVIEW	“SYSVIEW 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》和 “SYSVIEWS 统一视图”一节 《SQL Anywhere 服务器 - SQL 参考》
SYSWEBSERVICE / -	ISYSWEBSERVICE	“SYSWEBSERVICE 系统视图”一节 《SQL Anywhere 服务器 - SQL 参考》

新视图汇总表

系统视图名称	链接到详细信息
SYSDEPENDENCY	SYSDEPENDENCY 系统视图中的每行描述两个数据库对象间的依赖性。请参见“ SYSDEPENDENCY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSFKEY	SYSFKEY 系统视图的每行描述系统中的一个外键约束。请参见“ SYSFKEY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSIDX	SYSIDX 系统表的每行定义数据库中的一个逻辑索引。请参见“ SYSIDX 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSIDXCOL	SYSIDXCOL 系统视图的每行描述在 SYSIDX 系统视图中描述的索引的一列。请参见“ SYSIDXCOL 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSLOGINMAP	SYSLOGINMAP 系统视图包含可用于使用集成登录或 Kerberos 登录连接到数据库的所有用户名。请参见“ SYSLOGINMAP 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSMVOPTION	SYSMVOPTION 系统视图的每行描述实例化视图的一个选项值的设置。请参见“ SYSMVOPTION 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSMVOPTIONNAME	SYSMVOPTIONNAME 系统视图的每行包含 SYSMVOPTION 系统视图中定义的选项的名称。请参见“ SYSMVOPTIONNAME 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSOBJECT	SYSOBJECT 系统视图的每行描述一个对象。数据库对象的示例包括表、视图、列、索引和过程。请参见“ SYSOBJECT 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSPHYSIDX	SYSPHYSIDX 系统视图的每行定义数据库中的一个物理索引。请参见“ SYSPHYSIDX 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSPROCS	SYSPROCS 系统视图替换以前的 SYSPROCEDURES 视图。请参见“ SYSPROCS 统一视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSPROXYTAB	SYSPROXYTAB 系统视图的每行描述一个代理表的远程参数。请参见“ SYSPROXYTAB 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSREMARK	SYSREMARK 系统视图的每行描述一个对象的一个注释。请参见“ SYSREMARK 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

系统视图名称	链接到详细信息
SYSSOURCE	SYSSOURCE 系统视图的每行包含在 ISYSOBJECT 系统表中列出的一个对象的源对象。请参见“ SYSSOURCE 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSSYNCSCRIPT	SYSSYNCSCRIPT 系统视图的每行标识一个 MobiLink 脚本式上载的存储过程。请参见“ SYSSYNCSCRIPT 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSTABCOL	SYSTABCOL 系统视图为数据库中每个表和视图的每列都包含一行。请参见“ SYSTABCOL 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSUSER	SYSUSER 系统视图的每行描述数据库中的一个用户。请参见“ SYSUSER 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSUSERAUTHORITY	SYSUSERAUTHORITY 系统视图的每行描述授予一个用户 ID 的一个权限。请参见“ SYSUSERAUTHORITY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

不建议使用的表或视图的汇总表

以下为不建议使用的目录对象的列表。在多数情况下，该对象在先前版本中为一个表，但现在却是一个兼容性视图。引用这些对象不会产生错误；但是，为了将来的兼容性，鼓励您更改应用程序，以改为指向建议的对象。

不建议使用的表或视图	转换信息
SYSCOLLATION 系统表	归类映射信息现在存储为数据库属性。请参见“ SYSCOLLATION 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSCOLLATIONMAPPINGS 系统表	归类映射信息现在存储为数据库属性。请参见“ SYSCOLLATIONMAPPINGS 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSCOLUMN 系统表	改用 SYSTABCOL 系统视图。请参见“ SYSTABCOL 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSCOLUMN 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSFKCOL 系统表	改用 SYSFKEY 系统视图。请参见“ SYSFKEY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSFKCOL 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

不建议使用的表或视图	转换信息
SYSFOREIGNKEY 系统表	改用 SYSFKEY 系统视图。请参见“ SYSFKEY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSFOREIGNKEY 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSINDEX 系统表	改用 SYSIDX 系统视图。请参见“ SYSIDX 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSINDEX 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSIXCOL 系统表	改用 SYSIDXCOL 系统视图。请参见“ SYSIDXCOL 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSIXCOL 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSTABLE 系统表	改用 SYSTAB 系统视图。请参见“ SYSTAB 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSTABLE 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSUSERAUTH 系统视图	改用 SYSUSERAUTHORITY 系统视图。请参见“ SYSUSERAUTHORITY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSUSERAUTH 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSUSERPERM 系统表	改用 SYSUSERAUTHORITY 系统视图。请参见“ SYSUSERAUTHORITY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSUSERPERM 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSUSERLIST 系统视图	改用 SYSUSERAUTHORITY 系统视图。请参见“ SYSUSERAUTHORITY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSUSERLIST 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。
SYSUSERPERMS 系统视图	改用 SYSUSERAUTHORITY 系统视图。请参见“ SYSUSERAUTHORITY 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》和“ SYSUSERPERMS 兼容性视图（不建议使用） ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

删除的或重命名的表或视图的汇总表

以下为不再在目录中显示的目录对象的列表。引用这些对象将产生错误。

删除的表或视图	转换信息
SYSATTRIBUTE 系统表	改用 SYSTAB 和 SYSPHYSIDX 系统视图。现在在 ISYSTAB 系统表中维护有关可用百分比和聚簇索引的信息。有关键值、键距离、叶页和深度的信息现在存储在 ISYSPHYSIDX 系统表中。请参见“SYSTAB 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》和“SYSPHYSIDX 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》。
SYSATTRIBUTENAME 系统表	改用 SYSIDX 和 SYSPHYSIDX 系统视图。请参见“SYSIDX 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》和“SYSPHYSIDX 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》。
SYSEXTENT 系统表	SYSEXTENT 表在 SQL Anywhere 版本 10.0.0 及更高版本的目录中不再可用。此表以前未被使用。
SYSEXTERNLOGINS	重命名为 SYSEXTERNLOGIN。请参见“SYSEXTERNLOGIN 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》。
SYSLOGIN 系统表	SYSLOGIN 表已由 SYSLOGINMAP 系统视图所代替，并发生了某些更改。请参见“SYSLOGINMAP 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》。
SYSOPTBLOCK	此表过去仅供内部使用。
SYSOPTJOINSTRATEGY	此表过去仅供内部使用。
SYSOPTJOINSTRATEGIES	此视图过去仅供内部使用。
SYSOPTORDER	此表过去仅供内部使用。
SYSOPTORDERS	此视图过去仅供内部使用。
SYSOPTQUANTIFIER	此表过去仅供内部使用。
SYSOPTREQUEST	此表过去仅供内部使用。
SYSOPTREWRITE	此表过去仅供内部使用。
SYS PROCEDURES 视图	改用 SYSPROCS 统一视图。请参见“SYSPROCS 统一视图”一节《SQL Anywhere 服务器 - SQL 参考》。
SYSSERVERS	重命名为 SYSSERVER。请参见“SYSSERVER 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》。

删除的表或视图	转换信息
SYSUSERMESSAGES	重命名为 SYSUSERMESSAGE。请参见“ SYSUSERMESSAGE 系统视图 ”一节《 SQL Anywhere 服务器 - SQL 参考 》。

对系统表和系统视图中各列进行的更改

对系统表和系统视图中各列已进行了许多更改。除下面的信息外，所有的更改都包括添加新列或删除未使用的列，二者都不会对应用程序产生影响。

- **SYSCOLUMN 视图和 SYSCOLUMNS 视图** 这两个视图的 width 列都已从 SMALLINT 更改为 UNSIGNED INT。请参见“[SYSCOLUMN 兼容性视图（不建议使用）](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[SYSCOLUMNS 统一视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSCONSTRAINT 视图** 以前的 SYSCONSTRAINT 系统表已替换为新的系统表 ISYSCONSTRAINT，还包含相应的 SYSCONSTRAINT 系统视图。现在引用 SYSCONSTRAINT 将使用相应的新的系统视图，这是此版本的明显不同之处。要了解 SYSCONSTRAINT 系统视图的内容，请参见“[SYSCONSTRAINT 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSREMOPTION 视图** 您不再可以从 SYSREMOPTION 中进行选择。改用 SYSREMOPTIONS 或 SYSREMOPTION2。请参见“[SYSREMOPTIONS 统一视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》或“[SYSREMOPTION2 统一视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSJAR、SYSJARCOMPONENT 和 SYSJAVACLASS 视图** create_time 列已被删除。但是，在 SYSOBJECT.create_time 中提供创建时间信息。请参见“[SYSOBJECT 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSFILE 系统视图** store_type 列现在是 INTEGER 类型。请参见“[SYSFILE 兼容性视图（不建议使用）](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSROCPARM 视图和 SYSLOGINMAP 视图** remarks 列已从这些视图中删除。此外，SYSROCPARM 中的 width 列已从 SMALLINT 更改为 UNSIGNED INT。请参见“[SYSROCPARM 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[SYSLOGINMAP 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSROCPARMS 视图** SYSROCPARM.width 已从 SMALLINT 更改为 UNSIGNED INT。请参见“[SYSROCPARMS 统一视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSREMOUSER 视图** log_send、log_sent、confirm_sent、log_received 和 confirm_received 列现在是 UNSIGNED BIGINT。请参见“[SYSREMOUSER 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSUBSCRIPTION 视图** created 和 started 列现在是 UNSIGNED BIGINT。请参见“[SYSUBSCRIPTION 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SYSYNC 视图** progress、created 和 log_sent 列现在是 UNSIGNED BIGINT。请参见“[SYSYNC 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

SQL 语句

- **REVOKE CONNECT 语句** 当执行 REVOKE CONNECT 语句删除用户时，也会将指定用户所拥有的所有对象与该用户一并删除。如果数据库包含其他用户所拥有的活动视图，且该视图依赖于正在被删除用户所拥有的对象，则 REVOKE CONNECT 语句立即会返回一个错误。请参见“REVOKE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **派生表的键连接限制** 包含 TOP N、START AT、FIRST、ORDER BY、窗口函数、FOR XML 或递归表的派生表不允许使用键连接。请参见“视图和派生表的键连接”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **ALTER SERVER 语句和 CREATE SERVER 语句** ASAJDBC 和 ASAODBC 服务器类已分别重命名为 SAJDBC 和 SAODBC。请参见“ALTER SERVER 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“CREATE SERVER 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER 语句** 所有的 ALTER 语句现在都使用 ALTER 而不是 MODIFY 作为子句。如果应用程序使用的是 MODIFY 子句，则应该将其更改为使用 ALTER 子句。仍然支持 MODIFY 语法，但不建议使用。这将影响以下的语句：
 - “ALTER DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “ALTER EVENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》
 - “ALTER PUBLICATION 语句 [MobiLink] [SQL Remote]”一节《SQL Anywhere 服务器 - SQL 参考》
 - “ALTER SYNCHRONIZATION SUBSCRIPTION 语句 [MobiLink]”一节《SQL Anywhere 服务器 - SQL 参考》
 - “ALTER SYNCHRONIZATION USER 语句 [MobiLink]”一节《SQL Anywhere 服务器 - SQL 参考》
 - “ALTER TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》
- **BACKUP 语句** 在先前的版本中，您可以使用 TRANSACTION LOG RENAME 或 TRANSACTION LOG TRUNCATE 子句来指定 DBFILE ONLY 子句。现在使用这些 TRANSACTION LOG 子句中的任何一个来指定 DBFILE ONLY 都会产生错误，因为两个备份的类型相互排斥。请参见“BACKUP 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **COMMENT 语句** 不再支持语法 COMMENT ON LOGIN。改用语法 COMMENT ON INTEGRATED LOGIN。请参见“COMMENT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **INSERT 语句** 在 SQL Anywhere 10 中，使用 ON EXISTING SKIP 和 ON EXISTING ERROR 子句时，如果表包含缺省列，则服务器会计算缺省值，即使对于已存在的行也会如此。因此，缺省值（例如 AUTOINCREMENT）即使对跳过的行也会产生副作用。对于本例中的 AUTOINCREMENT，这导致在 AUTOINCREMENT 序列中跳过某些值。在以前的版本中，不会为跳过的行在缺省列上执行这些计算。请参见“INSERT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **VALIDATE 语句** 所有校验活动（如执行 VALIDATE 语句或运行校验实用程序 (dbvalid)）现在都需要 VALIDATE 权限；不再接受 REMOTE DBA 权限用于执行校验活动。

VALIDATE TABLE 语句以及 VALIDATE MATERIALIZED VIEW 检查孤立的 BLOB。

VALIDATE INDEX 的语法已更改为与 ALTER INDEX 语句的语法相一致。仍然支持旧语法，但不建议使用。如果应用程序当前使用的是 VALIDATE INDEX 语句，则应该更改为新语法。

有关这些更改的详细信息，请参见“[VALIDATE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

不建议使用和停用的功能

- **校验实用程序 (dbvalid) 的 -f、-fi、-fd、-fn 选项不建议使用** 已经简化了 dbvalid 实用程序的语法。以前，没有指定选项时，dbvalid 实用程序在校验表时执行快速校验。现在，dbvalid 实用程序在缺省情况下执行完全校验，就像指定了 -f、-fi- 和 -fd 选项。因此不建议使用这些选项，并且为了对表执行快速校验，必须指定 -fx 选项。
此外，不再支持 -fn 选项，该选项使用来自 9.0.0 版本及早期版本的算法执行校验。
有关校验实用程序的详细信息，请参见“[校验实用程序 \(dbvalid\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不建议使用的 VALIDATE TABLE 语句选项** 已经简化了 VALIDATE TABLE 语句的语法。以前，没有指定选项时，VALIDATE TABLE 语句执行常规校验。现在，VALIDATE TABLE 语句在缺省情况下执行完全校验（好像指定了 WITH FULL CHECK 选项一样）。因此，不建议使用 WITH FULL CHECK、WITH INDEX 和 WITH DATA 选项。请参见“[VALIDATE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **不建议使用 Transact-SQL 外连接** 此版本中已不建议使用 Transact-SQL 外连接，并且在 SQL Anywhere 的将来版本中对其将不支持。新 tsql_outer_joins 数据库选项启用或禁用在当前连接的 DML 语句和视图中使用 Transact-SQL 外连接运算符 *= 和 =* 的功能。此选项在缺省情况下设置为 Off。请参见“[tsql_outer_joins 选项 \[兼容性\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不再支持 WITH HASH SIZE 子句** 随着原有的 B 树索引技术的淘汰，现在不再支持 CREATE INDEX 语句的 WITH HASH SIZE 子句。
- **不支持的属性** 不再支持 NumProcessorsAvail 和 NumProcessorsMax 服务器属性。可以改用 NumLogicalProcessors、NumLogicalProcessorsUsed、NumPhysicalProcessors 和 NumPhysicalProcessorsUsed 服务器属性。请参见“[数据库服务器属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不建议使用 LOAD TABLE 的 STRIP ON 子句** 在 SQL Anywhere 10.0.0 中增强了去除前导空白和尾随空白进而允许您微调去除行为的情况下，不建议使用 STRIP ON。如果仍要仅去除尾随空白（在先前的版本中指定 STRIP ON 时的缺省行为），请改用 STRIP RTRIM。请参见“[LOAD TABLE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **不建议使用 UTF8 归类** 不建议使用 UTF8 归类。改用 UTF8BIN 归类。请参见“[支持的归类和替代归类](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不再支持 jConnect 4.5** 以前使用 jConnect 4.5 进行连接的应用程序仍然有效，但建议使用 jConnect 5.5 或 jConnect 6.0.5。请参见“[使用 jConnect JDBC 驱动程序](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **不再支持 SQLLOCALE 环境变量** 不再支持 SQLLOCALE 环境变量。该变量已由 SALANG 和 SACHARSET 环境变量代替。请参见“[SALANG 环境变量](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[SACHARSET 环境变量](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **不再支持命名管道** 不再支持命名管道协议。必须将以前使用命名管道的应用程序更改为使用共享内存。请参见“[选择通信协议](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不建议使用数据源实用程序 (dbdsn) 的 -o 选项** 数据源实用程序的 -o 选项已不建议使用。如果想要将输出消息写入某文件，可以在连接字符串中指定 LogFile 连接参数。请参见“[Logfile 连接参数 \[LOG\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不支持创建自定义归类** 不再支持创建自定义归类。不再支持 [创建自定义归类] 向导、归类实用程序 (dbcollat)、DBCcollate 函数和 a_db_collation 结构。请参见“[选择归类](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

如果利用自定义归类重建数据库，则在一个步骤中重建时会保留该归类。如果您选择卸载数据库，然后将模式和数据装载到您创建的数据库中，则必须使用所提供归类中的一种。请参见“[为版本 11 重建版本 9 及更早版本的数据库](#)”一节第 329 页。
- **不支持服务器授权实用程序 -p 选项** 在以前的版本中，服务器授权实用程序支持 -p 选项，该选项用来指定数据库服务器被授权使用的操作系统。不再支持此选项。
- **不支持数据库服务器 -d 选项** 不再支持 -d 数据库服务器选项，在 NetWare 上使用时，该选项强制使用 POSIX I/O 而不是 DFS（直接文件系统）I/O。
- **不支持数据库服务器 -y 选项** 因为不再支持 Windows 95/98/Me，所以也不再支持用于在这些操作系统上作为 Windows 服务来运行数据库服务器的 -y 数据库服务器选项。要在任何支持的平台上作为服务来运行数据库服务器，请使用 dbsvc 实用程序。请参见“[用于 Windows 的服务实用程序 \(dbsvc\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不支持 -sc 选项** SQL Anywhere 7.0 荣获美国政府授予的 TCSEC（可信计算机系统评估标准）C2 安全等级。-sc 服务器选项允许您以相当于 C2 认证环境的方式运行 SQL Anywhere 的当前版本。已经在 10.0.0 版中删除了 -sc 选项的支持以及 C2 服务器属性。
- **不支持 max_work_table_hash_size 数据库选项** 不再支持 max_work_table_hash_size 选项。查询优化程序基于表中的数据分布为内部临时表分配散列大小。
- **不支持 max_hash_size 数据库选项** 不再支持 max_hash_size 选项。
- **不支持压缩数据库和写文件** 因此，以下功能不再可用：
 - **文件扩展名** 不再支持以下文件扩展名：
 - 用于标识写文件的 .wrt 扩展名
 - 用于标识压缩数据库文件的 .cdb 扩展名
 - **NetWare 上的数据库服务器行为** 当指定不带扩展名的数据库文件时，数据库服务器不再查找扩展名为 .wrt 的数据库文件。请参见“[SQL Anywhere 数据库服务器](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
 - **在只读介质上部署数据库** 不能再提供写文件以记录对只读介质（如 CD-ROM）上提供的数据库的更改。但是，如果以只读模式运行这些数据库，您仍可以在只读介质上部署它们。请参见“[在只读介质上部署数据库](#)”一节《[SQL Anywhere 服务器 - 编程](#)》和“[-r 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **数据库实用程序** 不再支持以下的实用程序和向导：
 - 压缩数据库向导
 - 创建写文件向导
 - 解压缩数据库向导
 - 解压缩实用程序 (dbexpand)
 - 压缩实用程序 (dbshrink)
 - 写文件实用程序 (dbwrite)
- **SQL 语句** 不再支持以下 SQL 语句：
 - ALTER WRITEFILE
 - CREATE WRITEFILE
 - CREATE COMPRESSED DATABASE
 - CREATE EXPANDED DATABASE
- **DBTools 结构** 不再支持以下的结构或结构成员：
 - **a_backup_db 结构** 此结构用于保存使用 DBTools 库执行备份任务所需的信息。
backup_writefile 成员现在显示为 _unused。
 - **a_compress_db 结构** 此结构已被删除。
 - **a_compress_stats 结构** 此结构用于保存使用 DBTools 库执行数据库压缩任务所需的信息。
 - **a_db_info 结构** 此结构用于保存使用 DBTools 库返回 dbinfo 信息所需的信息。
wrbufsize 成员现在显示为 _unused1, wrnamebuffer 成员现在显示为 _unused2, 而 compressed 成员现在显示为 _unused3。
 - **an_expand_db 结构** 此结构用于保存使用 DBTools 库扩展数据库所需的信息。
 - **a_stats_line 结构** 此结构用于保存使用 DBTools 库进行数据库压缩和扩展所需的信息。
 - **a_writefile 结构** 此结构用于保存有关使用 DBTools 库进行数据库写文件管理所需的信息。
- **DBTools 函数** 不再支持以下函数：
 - DBChangeWriteFile
 - DBCompress
 - DBCreateWriteFile
 - DBExpand
 - DBStatusWriteFile
- **数据库属性** 不再支持以下数据库属性：
 - Compression
 - FileSize *writefile*
 - FreePages *writefile*
- **DB_BACKUP_WRITEFILE** 不再支持此嵌入式 SQL 函数。

- **删除了对未使用的 ASE 兼容性视图和过程的支持** 删除了对 SQL Anywhere 数据库中以下未使用的 Adaptive Server Enterprise 视图的支持：

视图名称	视图名称
SYSALTERNATES	SYSLOGINROLES
SYSAUDITOPTIONS	SYSLOGS
SYSAUDITS	SYSMESSAGES
SYSCHARSETS	SYSPROCEDURES
SYSCONFIGURES	SYSPROCESSES
SYSCONSTRAINTS	SYSPROTECTS
SYSCURCONFIGS	SYSREFERENCES
SYSDATABASES	SYSREMOTEOLOGINS
SYSDEPENDS	SYSROLES
SYSDEVICES	SYSSEGMENTS
SYSENGINES	SYSSERVERS
SYSKEYS	SYSSRVROLES
SYSLANGUAGES	SYSTHRESHOLDS
SYSLOCKS	SYSUSAGES

删除了对 SQL Anywhere 数据库中以下未使用的 Adaptive Server Enterprise 过程的支持：

过程名称	过程名称
sp_addalias	sp_helpindex
sp_addauditrecord	sp_helpjoins
sp_addlanguage	sp_helpkey
sp_addremotelogin	sp_helplanguage
sp_addsegment	sp_helplog
sp_addserver	sp_helpremotelogin
sp_addthreshold	sp_helpprotect

过程名称	过程名称
sp_adddumpdevice	sp_helpsegment
sp_auditdatabase	sp_helpserver
sp_auditlogin	sp_helpsort
sp_auditobject	sp_helpthreshold
sp_auditoption	sp_helpuser
sp_auditsproc	sp_indsuspect
sp_bindefault	sp_lock
sp_bindmsg	sp_locklogin
sp_bindrule	sp_logdevice
sp_changedbowner	sp_modifylogin
sp_checknames	sp_modifythreshold
sp_checkreswords	sp_monitor
sp_clearstats	sp_placeobject
sp_commonkey	sp_primarykey
sp_configure	sp_procxmode
sp_cursorinfo	sp_recompile
sp_dboption	sp_remap
sp_dbremap	sp_remoteoption
sp_depends	sp_rename
sp_diskdefault	sp_renamedb
sp_displaylogin	sp_reportstats
sp_dropalias	sp_role
sp_dropdevice	sp_serveroption
sp_dropkey	sp_setlangalias

过程名称	过程名称
sp_droplanguage	sp_spaceused
sp_dropremotelogin	sp_syntax
sp_dropsegment	sp_unbindefault
sp_dropserver	sp_unbindmsg
sp_dropthreshold	sp_unbindrule
sp_estspace	sp_volchanged
sp_extendsegment	sp_who
sp_foreignkey	sp_column_privileges
sp_help	sp_databases
sp_helpconstraint	sp_datatype_info
sp_helpdb	sp_server_info
sp_helpdevice	sp_table_privileges
sp_helpgroup	

- **删除了 SYSINDEX 系统视图中的 index_type 和 index_owner 列** 从 SYSINDEX 视图中删除了 index_type 和 index_owner 列。这些列以前分别包含缺省值 USER 和 SA。索引信息现在存储在 ISYSIDX 和 ISYSIDXCOL 系统视图中。请参见“[SYSIDX 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[SYSIDXCOL 系统视图](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **删除了服务器上的 DLL 协议选项** DLL 协议选项现在仅适用于在 Windows 32 位平台上运行的客户端。由于数据库服务器仅使用 Winsock 2.2，已将 DLL 协议选项从数据库服务器删除。类似地，由于 Windows CE 客户端仅使用 Winsock 1.1，已将 DLL 协议选项从 Windows CE 客户端删除。
所有 Windows 平台上的数据库服务器都要求 Winsock 2.2。
- **重命名了 ASANY 和 ASANYSH 环境变量** ASANY 和 ASANYSH 环境变量被分别重命名为 SQLANY10 和 SQLANYSH10。请参见“[SQLANY11 环境变量](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[SQLANYSAMP11 环境变量](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **不建议使用 PreserveSource 属性** 不建议在本版本中使用 PreserveSource 数据库属性，查询该属性的设置时，始终返回值 On。
- **不支持的数据库属性** 在本版本中删除了以下数据库属性：

- BlobArenas
 - ClusteredIndexes
 - CompressedBTrees
 - FileVersion
 - FreePageBitMaps
 - Histograms
 - HistogramHashFix
 - IndexStatistics
 - LargeProcedureIDs
 - NamedConstraints
 - SeparateCheckpointLog
 - SeparateForeignKeys
 - StringHistogramsFix
 - TableBitMaps
 - TablesQualTriggers
 - TransactionsSpanLogs
 - UniqueIdentifier
 - VariableHashSize
- **不支持的系统过程** 不再支持 `sa_conn_properties_by_name` 和 `sa_conn_properties_by_conn` 系统过程。可以使用新的 `sa_conn_options` 系统过程获取此信息。请参见“[sa_conn_options 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
 - **从查询优化计划中删除的算法** 从查询优化计划中删除了 [锁] 算法、[嵌套块连接] 算法、[排序块] 算法和 [JNBO] 算法，并且计划中不再显示锁节点。可以在计划中查看扫描节点的锁定信息。
 - **不建议使用 util_db.ini 文件** 不建议连接到实用程序数据库时使用 `util_db.ini` 文件为 DBA 用户指定口令。可以改用 `-su` 服务器选项。请参见“[对网络数据库服务器使用 util_db.ini（不建议使用）](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[-su 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
 - **不建议使用 WindowsCE 平台** 已取消对 Windows CE MIPS 处理器的支持。有关受支持的平台的列表，请参见“[支持的平台](#)”一节《[SQL Anywhere 11 - 简介](#)》。

MobiLink

以下各节描述了 10.0.0 版的 MobiLink 中的新功能、行为更改和不建议使用的功能。

新功能

以下列出了 10.0.0 版中引入的 MobiLink 新增功能。

主要的 MobiLink 新功能

Sybase Central 的 MobiLink 插件的增强

现在，通过使用向导创建 **synchronization model file**，可以更容易地建立 MobiLink 应用程序。此文件包含您所输入的有关远程表和统一表及如何同步它们的信息。准备好模型后，可以使用另一向导进行部署，这将生成应用程序所需的脚本和表。

- **创建同步模型向导** 使用新的 [创建同步模型向导]，可以快速创建和部署 MobiLink 应用程序。此向导可以创建远程数据库或使用现有的远程数据库。它将自动创建同步脚本，并可以自动处理下载删除、冲突解决及其它复杂的同步问题。

请参见“[创建模型](#)”一节《[MobiLink - 入门](#)》。

- **模型模式** 使用 [创建同步模型向导] 后，可以在部署同步项目前使用模型模式对其进行自定义。如果在模型模式下，则您以脱机方式工作。模型模式将同步模型存储为 XML 文件。

请参见“[模型模式](#)”一节《[MobiLink - 入门](#)》。

- **部署向导** 自定义模型后，可以使用新的 [部署] 向导进行部署。[部署] 向导将脚本、用户、脚本版本等添加到统一数据库上的 MobiLink 系统表中。尽管可以多次部署相同的模型，但对统一数据库所进行的任何更改都无法回退到模型模式。

请参见“[部署模型](#)”一节《[MobiLink - 入门](#)》。

- **管理模式** 在 10.0.0 版之前存在的 MobiLink 插件现在称为管理模式。为使管理模式更易于使用，已对其进行了许多增强。处于管理模式时，您将连接到统一数据库并使更改处于活动状态。可以使用管理模式修改所有的 MobiLink 统一数据库。

请参见“[模型模式](#)”一节《[MobiLink - 入门](#)》。

同步到任何数据源

使用称为**直接行处理**的新功能几乎可以同步到任何数据源。例如，您可以同步到应用程序服务器、Web 服务器、Web 服务、文本文件、Excel 电子表格、J2ME 设备，或使用基于 SQL 的行处理同步到无法用作统一数据库的 RDBMS。您仍需要一个统一数据库，用来保存 MobiLink 系统表和要让 MobiLink 管理的数据。可以将一个或多个新数据源完全集成到同步进程中。

对 MobiLink 服务器 API 进行了扩展以支持直接行处理，并添加了两个新事件。请参见：

- “直接行处理” 《MobiLink - 服务器管理》
- “handle_DownloadData 连接事件” 一节 《MobiLink - 服务器管理》
- “handle_UploadData 连接事件” 一节 《MobiLink - 服务器管理》

此外，一个称为 **mobile web services** 的新功能还提供了对可与远程应用程序集成的优化的移动异步 Web 服务的支持。

请参见“移动 Web 服务” 《QAnywhere》。

更易于部署

现在，您可以使用 [部署] 向导部署 MobiLink 服务器、SQL Anywhere 客户端、MobiLink 监控器和加密组件。不再提供以前版本附带的 InstallShield 合并模块和模板。请参见：

- “使用 [部署向导]” 一节 《SQL Anywhere 服务器 - 编程》
- “部署 MobiLink 应用程序” 《MobiLink - 服务器管理》

统一数据库

新的设置过程

- **SQL Anywhere 统一数据库所需的安装脚本** 现在，将 SQL Anywhere 数据库用作 MobiLink 统一数据库之前必须运行安装脚本，并且使用安装脚本创建的 MobiLink 系统表归运行安装脚本的用户所有。此行为与其它统一数据库类型一致。在 MobiLink 的以前版本中，MobiLink 系统表归 SQL Anywhere 统一数据库中的 DBO 所有。

请参见“SQL Anywhere 统一数据库” 一节 《MobiLink - 服务器管理》。

- **简化的设置过程** 现在，可以在 Sybase Central 中设置统一数据库，或使用安装脚本设置统一数据库。以前，您必须运行安装脚本。此外，每种类型的统一数据库现在都只有一个安装脚本。没有更多的版本特定的安装脚本（如 *syncase125.sql*）。

请参见“MobiLink 统一数据库” 《MobiLink - 服务器管理》。

新的系统对象

- **清理统一数据库的 MobiLink 系统表的新方法** 添加了新的系统过程，可帮助您执行以下操作：

- 从 MobiLink 系统表清除过时远程数据库的相关信息。

请参见“ml_delete_sync_state_before 系统过程” 一节 《MobiLink - 服务器管理》。

- 删除未使用的或不需要的同步状态信息。

请参见“ml_delete_sync_state 系统过程” 一节 《MobiLink - 服务器管理》。

- 重置同步状态信息。

请参见“ml_reset_sync_state 系统过程” 一节 《MobiLink - 服务器管理》。

- **新的 MobiLink 服务器系统表和模式** 对 MobiLink 系统表进行了以下更改：

- 添加了多个新的 MobiLink 系统表。请参见：
 - “ml_database” 一节 《MobiLink - 服务器管理》
 - “ml_column” 一节 《MobiLink - 服务器管理》
 - “ml_qa_clients” 一节 《MobiLink - 服务器管理》
- ml_subscription 的内容发生了很大变化。以前存储在 ml_user.commmmit_state 中的 UltraLite 同步序列号现在存储在 ml_subscription.progress 中。progress 列还存储 SQL Anywhere 远程同步进度。
请参见 “ml_subscription” 一节 《MobiLink - 服务器管理》。
- ml_user 的内容发生了很大变化。
请参见 “ml_user” 一节 《MobiLink - 服务器管理》。
- 将 checksum 列添加到了 ml_script 表。
请参见 “ml_script” 一节 《MobiLink - 服务器管理》。
- ml_listening 的 ml_user 列已更改为 name 列。
请参见 “ml_listening” 一节 《MobiLink - 服务器管理》。
- 添加了一个新的系统表，该表由 Sybase Central 内部用于服务器启动的同步。
请参见 “ml_sis_sync_state” 一节 《MobiLink - 服务器管理》。
- 对多个 QAnywhere 系统表进行了更改。请参见：
 - “ml_qa_delivery” 一节 《MobiLink - 服务器管理》
 - ml_qa_delivery_client
 - “ml_qa_global_props” 一节 《MobiLink - 服务器管理》
 - ml_qa_global_props_client
 - “ml_qa_repository” 一节 《MobiLink - 服务器管理》
 - ml_qa_repository_client
 - “ml_qa_repository_props” 一节 《MobiLink - 服务器管理》
 - ml_qa_repository_props_client
 - “ml_qa_repository_staging” 一节 《MobiLink - 服务器管理》
- **新系统过程 ml_add_column** 使用命名的行参数时，您可能在某些情况下需要使用此新的系统过程，以使用远程数据库上列的相关信息来填充 ml_column MobiLink 系统表。
请参见 “ml_add_column 系统过程” 一节 《MobiLink - 服务器管理》。

MobiLink 服务器

新的 mlsrv10 功能

- **服务器名更改为 mlsrv10** MobiLink 服务器现在称为 mlsrv10。以前称为 dbmlsrv9。
- **mlsrv10 -x 的新语法** mlsrv10 -x 选项（用于为 MobiLink 客户端设置网络协议选项）发生了更改。
请参见 “-x 选项” 一节 《MobiLink - 服务器管理》。

- **用于旧客户端的新 -xo 选项** 要将 MobiLink 服务器连接到版本 8 客户端或版本 9 客户端，您应该使用 `mlsrv10 -xo` 选项，该选项与 `dbmlsrv9 -x` 选项的作用相同。您可以支持版本 8 和版本 9 客户端以及版本 10 客户端（通过一个 `mlsrv10` 实例），但要执行此操作，您需要使用两个不同的端口。

HTTP 和 HTTPS 的 `-xo` 选项包含一个新选项 `session_key`，如果无法将 `JSESSIONID` 用于跟踪连接，该选项会很有用。

请参见“[-xo 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **改进的高速缓存处理** MobiLink 服务器不再为不同任务维护单独的内存池。所有的高速缓存都由所有的同步共享。您可以使用新的 `mlsrv10 -cm` 选项设置高速缓存大小。用于设置高速缓存大小的其它选项（`-bc`、`-d`、`-dd` 和 `-u`）已被删除。

请参见“[-cm 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **ignore 选项现在影响所有流** 现在，在所有的 `-x` 流上都忽略指定要忽略的每个主机名或 IP 地址。以前（并仍然使用 `-xo`），只在指定忽略的主机所在的流上忽略该主机。

请参见“[-x 选项](#)”一节《[MobiLink - 服务器管理](#)》中的“`ignore`”。

- **强制上载脚本** 使用 `mlsrv10 -zus` 选项可以强制 MobiLink 服务器调用表的上载脚本，即使该表没有要上载的数据。

请参见“[-zus 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **新的详细程度选项** 使用新的详细程度选项 `e` 可以捕获系统事件脚本。指定了 `-ve` 后，MobiLink 服务器将显示用于维护 MobiLink 系统表的所有系统事件脚本，以及定义上载流的 SQL 语句。

请参见“[-v 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **文件传输目录** 添加了一个新选项，使用该选项可以将目录用于文件传输。

请参见“[-ftr 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **设置并发同步的最大数量** 现在可以设置可活动处理的最大同步数，从而提高性能。

请参见“[-sm 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **限制并发网络连接** 使用新 `-nc` 选项，您可以指定对并发网络连接数量的限制。

请参见“[-nc 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **mlsrv10 现在将 ISO 8601 日期时间格式用于消息时间戳** 现在，信息性消息、警告消息和错误消息中的时间戳都使用明确的 ISO 8601 日期时间格式：`{I|W|E}.yyyy-mm-dd hh:mm:ss message`。

新的 MobiLink 脚本编写功能

- **命名的脚本参数** 现在，MobiLink 事件参数有了名称。以前，必须将脚本参数指定为问号。现在，问号是可选的。您可以从一组预定义的命名参数中进行选择或创建自己的命名参数，或采用以上两种方式。当 RDBMS 不支持变量时，用户定义的命名参数很有用。与问号不同的是，您可以以任意顺序指定命名参数，并可以使用可用参数的任何子集。此外，在多数情况下，您可以在同一脚本中多次使用同一命名参数。

请参见“[脚本参数](#)”一节《[MobiLink - 服务器管理](#)》。

- **新冲突检测事件** 增加了一个新事件，可将其写入脚本以检测列级冲突。此事件与 `upload_fetch` 事件作用相同，后者用于检测行级冲突。
请参见“[upload_fetch_column_conflict 表事件](#)”一节《[MobiLink - 服务器管理](#)》。
- **全局脚本版本** 现在可以创建全局脚本版本。您只要定义一次与全局脚本版本关联的脚本，以后就可以在所有同步中自动使用这些脚本，除非您在要用来同步的脚本版本中为相同事件指定了脚本。当您使用多个脚本版本时，这就意味着您可以避免复制连接级脚本。
请参见“[ml_global 脚本版本](#)”一节《[MobiLink - 服务器管理](#)》。

性能增强

- **改进的 MobiLink 体系结构** 为提高吞吐量、灵活性和可维护性，重新设计了 MobiLink 服务器。出于相同原因，增强了内部 MobiLink 客户端/服务器协议。
请参见“[MobiLink 性能](#)”《[MobiLink - 服务器管理](#)》。

其它服务器增强

- **快照隔离** 对于 SQL Anywhere 版本 10 和 Microsoft SQL Server 2005 及更高版本的统一数据库，快照隔离现在是下载的缺省选项，同时还是上载的一个选项。添加了一些 MobiLink 服务器选项以帮助您控制此行为。
请参见：
 - “[MobiLink 隔离级别](#)”一节《[MobiLink - 服务器管理](#)》
 - “[-dsd 选项](#)”一节《[MobiLink - 服务器管理](#)》
 - “[-dt 选项](#)”一节《[MobiLink - 服务器管理](#)》
 - “[-esu 选项](#)”一节《[MobiLink - 服务器管理](#)》
- **同步 ID** 现在每个同步都由 1 到 4294967295 之间的一个整数标识。MobiLink 服务器的每个实例都维护其自己的同步 ID。启动 MobiLink 服务器时，同步 ID 被重置为 1。在输出文件中记录此 ID。
- **改进的 MobiLink 网络层** 网络层现在包括压缩、持久连接（因此您可以在同一连接上多次同步）、IPv6 支持和改进的错误检测、活动检测以及调试。

MobiLink 监控器增强

- **实用程序名更改为 mlmon** MobiLink 监控器现在称为 mlmon。以前，它称为 dbmlmon。
请参见“[启动 MobiLink 监控器](#)”一节《[MobiLink - 服务器管理](#)》。
- **多个 MobiLink 监控器** 现在可以同时将多个 MobiLink 监控器连接到同一 MobiLink 服务器。这使得多个用户可以跟踪同一服务器上的同步。
请参见“[启动 MobiLink 监控器](#)”一节《[MobiLink - 服务器管理](#)》。
- **网络选项** 现在，MobiLink 监控器允许使用与 MobiLink 客户端相同的网络选项。
请参见“[启动 MobiLink 监控器](#)”一节《[MobiLink - 服务器管理](#)》。
- **新运用图形** [运用图形] 窗格显示 MobiLink 服务器中的队列长度。
请参见“[\[运用图形\] 窗格](#)”一节《[MobiLink - 服务器管理](#)》。

- **在 [图表] 窗格中查看数据** 在 [图表] 窗格中，您仍可以按用户查看数据，也可以选择 [精简视图] 中查看数据，该视图以尽可能少的行显示所有活动的同步。[工作] 视图已被删除，因为同步不再绑定到单个工作线程。
请参见 “[图表] 窗格” 一节 《MobiLink - 服务器管理》。
- **新 [示例属性] 窗口** 新 [示例属性] 对话框显示在所选项期间内一个一秒间隔或所有一秒间隔的平均值的数据。
请参见 “示例属性” 一节 《MobiLink - 服务器管理》。
- **增强的 [会话属性] 窗口** [会话属性] 对话框现在包含一个详细的 [统计] 选项卡。
请参见 “会话属性” 一节 《MobiLink - 服务器管理》。
- **能够监控启用了 FIPS 的服务器** MobiLink 监控器现在可以监控正在运行 FIPS 认可的加密的 MobiLink 服务器。以前，它做不到这一点。
- **对统计属性进行了更改** 请参见 “MobiLink 服务器更改” 一节第 179 页中的 “对统计属性进行了更改”。

MobiLink 重定向器增强

- **重定向器支持 MobiLink 服务器组** 对于某些重定向器，您现在可以创建 MobiLink 服务器组。服务器组可用于同时支持版本 10 客户端和版本 8 或版本 9 客户端（通过一个重定向器），或用于其它目的。有关哪些重定向器支持服务器组的信息，请参见 “支持的平台” 一节 《SQL Anywhere 11 - 简介》。
请参见 “MobiLink 服务器组” 一节 《MobiLink - 服务器管理》。
支持服务器组的重定向器还对其配置设置进行了其它增强。此外，它们还使用名为 *redirector_server_group.config* 的新示例配置文件。
请参见 “配置重定向器属性（适用于支持服务器组的重定向器）” 一节 《MobiLink - 服务器管理》。
- **HTTPS 支持** 在 SQL Anywhere 的以前版本中，当 HTTPS 用于远程数据库和 Web 服务器之间的连接时，Web 服务器会对 HTTPS 进行解密并通过重定向器将 HTTP 发送到 MobiLink。现在，对于某些 Web 服务器，重定向器将流重新加密为 HTTPS，并将其发送到 MobiLink 服务器。重定向器配置文件中的 ML 指令有了新语法。有关哪些 Web 服务器具有 HTTPS 支持的信息，请参见 “支持的平台” 一节 《SQL Anywhere 11 - 简介》。
请参见 “配置重定向器属性（适用于支持服务器组的重定向器）” 一节 《MobiLink - 服务器管理》。

Unix/Linux 增强

- **MobiLink 服务器消息窗口** Linux 安装现在有了一个消息窗口，可显示 dbmlsync 和 mlsrv10 的日志信息。
请参见 “-ux 选项” 一节 《MobiLink - 客户端管理》和 “-ux 选项” 一节 《MobiLink - 服务器管理》。
- **更一致的字符转换** 改进了 Unix/Linux 和 Windows 之间字符转换的一致性。

MobiLink 客户端

- **新远程 ID** MobiLink 现在使用称为远程 ID 的新标识符来唯一标识远程数据库。以前版本使用的是 MobiLink 用户名。远程 ID 存储在远程数据库中。MobiLink 在远程数据库第一次进行同步时（或遇到远程 ID 的 NULL 值时）生成远程 ID。远程 ID 自动创建为 GUID，但您可以将其设置为对您具有意义的任何字符串。远程 ID 允许同一 MobiLink 用户与多个远程数据库进行同步。在 UltraLite 远程数据库中，远程 ID 也可用于允许多个 MobiLink 用户与同一远程数据库进行同步。

将 MobiLink 用户名作为参数接受的每个脚本现在也接受 `remote_id` 参数。只有在使用命名参数时，`remote_id` 参数才可用。

为了帮助您更改远程 ID，将一个名为 `ml_remote_id` 的新数据库选项添加到了 SQL Anywhere 和 UltraLite 数据库。

请参见：

- “远程 ID”一节 《MobiLink - 客户端管理》
- “MobiLink 用户名和远程 ID”一节第 184 页
- SQL Anywhere 客户端：“设置远程 ID”一节 《MobiLink - 客户端管理》
- UltraLite 客户端：“UltraLite `ml_remote_id` 选项”一节 《UltraLite - 数据库管理和参考》
- **新文件传输功能** 新功能帮助您通过同步数据时所使用的同一网络路径向远程设备传输文件。SQL Anywhere 客户端可以使用新 `mlfiletransfer` 实用程序，而 UltraLite 客户端则可以使用新 `MLFileTransfer` 方法。填充新的远程数据库或升级软件时，此功能特别有用。添加了新的 MobiLink 事件来验证文件传输（如果需要）。请参见：
 - SQL Anywhere 客户端：“MobiLink 文件传输实用程序 (`mlfiletransfer`)”一节 《MobiLink - 客户端管理》
 - UltraLite 客户端：“使用 MobiLink 文件传输”一节 《UltraLite - 数据库管理和参考》
 - MobiLink 服务器：“`authenticate_file_transfer` 连接事件”一节 《MobiLink - 服务器管理》
- **SendColumnNames 已更改** 以前使用 `SendColumnNames dbmlsync` 扩展选项和 UltraLite 同步参数 `Send Column Names` 来上载有关远程数据库列的信息，以便 MobiLink 服务器可以生成示例同步脚本。示例同步脚本的创建已被删除（并由 [创建同步模型向导] 所取代）。现在只有直接行处理使用 `SendColumnNames`。请参见：
 - “直接行处理” 《MobiLink - 服务器管理》
 - “`SendColumnNames (scn)` 扩展选项”一节 《MobiLink - 客户端管理》
 - “`Send Column Names` 同步参数”一节 《UltraLite - 数据库管理和参考》
- **简化的活动超时设置** 活动超时现在由客户端进行控制。引入了一个称为 `timeout` 的新网络协议选项，用于替换 `liveness_timeout`、`contd_timeout`、`unknown_timeout` 和 `network_connect_timeout`。请参见“`timeout`”一节 《MobiLink - 客户端管理》。
- **Buffer_size 增强** 使用 `buffer_size` 网络协议选项，现在可以控制 TCP/IP 协议的写入缓冲区以及 HTTP 协议的 HTTP 主体大小。缺省值也已更改。请参见“`buffer_size`”一节 《MobiLink - 客户端管理》。

UltraLite 客户端

- **Palm 支持 network_leave_open** 在 Palm 设备上，您现在可以选择同步完成后网络连接是否保持打开状态。在以前的版本中，在其它平台上提供此功能。

请参见“network_leave_open”一节《MobiLink - 客户端管理》。

- **UltraLite 增强** 有关其它 UltraLite 增强的信息，请参见“同步”一节第 199 页。

SQL Anywhere 客户端

- **脚本式上载** 在常规同步中，dbmsync 使用事务日志创建上载，因此将同步自上次上载以来远程数据库上更改的所有相关数据。现在，您可以编写存储过程来确切定义要上载的行，从而跳过事务日志的使用。这些存储过程可以执行 DML 并上载结果集，因而这些行可以根据需要动态创建。

请参见“脚本式上载”一节《MobiLink - 客户端管理》。

为支持脚本式上载，对 SQL Anywhere 系统对象进行了以下更改：

- 在 ISYSPUBLICATION 系统表中添加新列 (sync_type)。请参见“SYSPUBLICATION 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》。

- 在 ISYSSYNCSCRIPT 系统表中添加新目录对象以跟踪同步脚本。

请参见“SYSSYNCSCRIPT 系统视图”一节《SQL Anywhere 服务器 - SQL 参考》。

- 新建了用于转换进度值的系统过程。请参见：

- “sa_convert_ml_progress_to_timestamp 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》
- “sa_convert_timestamp_to_ml_progress 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》

- **dbmsync 的新调度选项** 使用 EVERY 和 INFINITE 调度选项时，您现在可以指定启动 dbmsync 时不发生同步。

请参见“NoSyncOnStartup (nss) 扩展选项”一节《MobiLink - 客户端管理》。

- **仅下载发布** 现在可以创建只下载数据的发布。仅下载发布不使用日志文件。

请参见“仅下载发布”一节《MobiLink - 客户端管理》。

- **错误处理增强** 添加了新的的事件挂接，允许您在客户端上处理 dbmsync 报告的错误。

请参见：

- “处理事件挂接过程中的错误和警告”一节《MobiLink - 客户端管理》
- “sp_hook_dbmsync_all_error”一节《MobiLink - 客户端管理》
- “sp_hook_dbmsync_communication_error”一节《MobiLink - 客户端管理》
- “sp_hook_dbmsync_misc_error”一节《MobiLink - 客户端管理》
- “sp_hook_dbmsync_sql_error”一节《MobiLink - 客户端管理》

- **阻止 dbmsync 强制实施表顺序** 缺省情况下，如果在上载父表之前先上载子表，dbmsync 会发出一个错误。新的扩展选项允许您替换此行为。

请参见“[TableOrderChecking \(toc\) 扩展选项](#)”一节《[MobiLink - 客户端管理](#)》。

- **持久连接** 现在，您可以指定 dbmsync 应使与 MobiLink 服务器的连接在多个同步之间处于打开状态。

请参见“[-pc 选项](#)”一节《[MobiLink - 客户端管理](#)》。

- **跟踪同步的新方法** 只针对 SQL Anywhere 远程数据库，您现在可以在 begin_publication 或 end_publication 脚本中指定 subscription_id 参数。此值在 SYSSYNC 系统表中称为 sync_id。这是一个帮助您跟踪同步信息的高级功能。请参见：

- “[begin_publication 连接事件](#)”一节《[MobiLink - 服务器管理](#)》

- “[end_publication 连接事件](#)”一节《[MobiLink - 服务器管理](#)》

- **dbmsync 现在将 ISO 8601 日期时间格式用于消息时间戳** 现在，信息性消息、警告消息和错误消息中的时间戳都使用明确的 ISO 8601 日期时间格式：`{I|W|E}.yyyy-mm-dd`

`hh:mm:ss message.`

- **#hook_dict 中的扩展值** dbmsync 实用程序提供挂接并通过称为 #hook_dict 的临时表以名称/值的形式传递值。过去，#hook_dict 表中的值被定义为 VARCHAR(255)。此值已增大到 VARCHAR(10240)。

安全性

- **现在随 SQL Anywhere 提供了 RSA** 您不必再去购买单独的许可就可使用 RSA 加密，除非您在使用 FIPS。

- **可同时使用 ECC 加密和 HTTPS** 现在可以在使用 HTTPS 时使用 ECC 加密。

- **新 mlsrv10 -fips 选项** 现在，您可以在启动 MobiLink 服务器时指定 -fips，从而强制所有安全连接都使用 FIPS 认可的算法。此设置不会影响非安全流。

请参见“[-fips 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **新 mluser -fips 选项** MobiLink 用户验证实用程序现在提供一个选项，以强制使用 FIPS 安全功能。

请参见“[MobiLink 用户验证实用程序 \(mluser\)](#)”一节《[MobiLink - 服务器管理](#)》。

- **更多的平台支持 FIPS 安全功能** 现在更多的平台都支持 FIPS 安全功能。有关受支持的平台的列表，请参见“[支持的平台](#)”一节《[SQL Anywhere 11 - 简介](#)》。

- **简化了指定安全流的方法** 通过将安全选项视为单独的网络协议，简化了服务器和客户端上指定安全选项的语法。现在支持以下协议：TCP/IP、TLS（使用 TLS 安全性的 TCP/IP 同步）、HTTP 和 HTTPS。删除了 UltraLite 的 Security 参数。

请参见：

- MobiLink 服务器：[“-x 选项”](#)一节《[MobiLink - 服务器管理](#)》

- MobiLink 客户端：[“MobiLink 客户端网络协议选项汇总”](#)一节《[MobiLink - 客户端管理](#)》

- **与操作系统集成** 缺省情况下，MobiLink 客户端现在信任客户端在其上运行的操作系统已经信任的证书。

服务器启动的同步

易于使用

- **可以更容易地设置服务器启动的同步** 为了更快速地设置服务器启动的同步应用程序，进行了以下的增强：
 - **Sybase Central 支持** 现在，可以在 Sybase Central 模型模式下设置通告程序和监听器，并允许有用的通知服务的子集。在模型模式下，您可以标识用于服务器启动的同步的表，并且 `download_cursor` 将自动用于确定哪些数据用于通知目的。当下载游标中标识的数据发生变化时，将发送一个通知。[部署向导] 生成相应的 [监听器] 选项文件。
请参见“在 [模型] 模式中设置服务器启动的同步”一节《MobiLink - 入门》。
 - **新缺省网关** 使用称为 SYNC 网关的新网关，您可以在用于 MobiLink 同步的相同类型的通信路径上建立持久连接。SYNC 网关现在是缺省的设备跟踪器网关，这意味着通知将首先尝试通过 SYNC 网关，然后回退到 UDP，最后是 SMTP 网关。
请参见“网关和运营公司”一节《MobiLink - 服务器启动的同步》。

通告程序增强

- **共享连接** 多个通告程序现在可以共享相同的数据库连接，从而降低了统一数据库中的争用和必需的服务器资源。
请参见“通告程序事件”一节《MobiLink - 服务器启动的同步》。
- **通告程序使用远程设备的字符集** 现在可以使用远程设备的字符集将通知发送到远程设备。在将设备跟踪信息应用到统一数据库之前对其进行转换。
- **自定义确认处理** 现在，可以在 SQL 中实现用于处理推式请求的确认并返回其状态的通告程序属性。
请参见“`confirmation_handler` 事件”一节《MobiLink - 服务器启动的同步》。
- **自定义错误处理** 现在，可以在 SQL 中实现用于处理错误（如推式请求未传递、未确认或未正确确认）的通告程序属性。
请参见“`error_handler` 事件”一节《MobiLink - 服务器启动的同步》。

监听器增强

- **持久连接** Windows 监听器现在支持持久连接。缺省情况下，Windows 监听器现在维护与 MobiLink 服务器的持久连接，以进行设备跟踪、通知和确认。此功能显著增强了以前版本的性能。可以使用 `dblsn -pc` 选项禁用此功能。
请参见“用于 Windows 设备的监听器实用程序”一节《MobiLink - 服务器启动的同步》。
- **新的或更改的 Windows 监听器选项** 监听器现在支持以下选项：

选项	说明
<code>-ni</code>	使用 <code>-x</code> 时停止跟踪 UDP 地址。以前，它称为 <code>-g</code> 。
<code>-pc{+ -}</code>	启用/禁用用于通知的持久连接。

选项	说明
-ns	禁用 Windows Mobile 2003 及更高版本的 Phone Edition 上的缺省 SMS 监听。
-nu	禁用缺省 UDP 监听。
-r	注册由 \$remote_id 变量所使用的远程 ID 文件。
-v	当设置为 1 或更大值时，详细程度选项现在显示并记录命令行选项。

- **远程 ID 文件** 在监听器命令行上，现在可以使用远程 ID 文件访问新 MobiLink 远程 ID（缺省情况下为 GUID）。使用新 dblsn 选项 **-r** 和新监听器操作变量 \$remote_id 执行此操作。
请参见“用于 Windows 设备的监听器实用程序”一节《MobiLink - 服务器启动的同步》和“用于 Windows 的监听器操作命令”一节《MobiLink - 服务器启动的同步》。
- **用于验证的新监听器操作变量** 有几个用于消息处理程序中的新操作变量：\$ml_user 和 \$ml_password。
请参见“用于 Windows 的监听器操作命令”一节《MobiLink - 服务器启动的同步》。
- **用于连接参数的新监听器操作变量** 新的 \$ml_connect 操作变量扩展到使用 dblsn -x 选项指定的 MobiLink 连接参数。
请参见“用于 Windows 的监听器操作命令”一节《MobiLink - 服务器启动的同步》。
- **监听器现在将 ISO 8601 日期时间格式用于消息时间戳** 现在，信息性消息、警告消息和错误消息中的时间戳都使用明确的 ISO 8601 日期时间格式：`{I|W|E}.yyyy-mm-dd hh:mm:ss message`。
- **监听器可以使用 TLS** Windows 监听器现在可以将所有网络选项用作其它 MobiLink 客户端，从而连接到 MobiLink 服务器。这允许您将安全功能应用于设备跟踪和通知。
请参见“用于 Windows 设备的监听器实用程序”一节《MobiLink - 服务器启动的同步》中的 -x。

增加了设备支持

- **支持 Treo 600 和 Treo 650** Palm 监听器现在支持 Treo 600 和 Treo 650。
- **CE Phone Edition 支持** Windows 监听器现在支持 Windows Mobile 2003 Phone Edition 上的 SMS。

行为更改和不建议使用的功能

以下列出了 10.0.0 版中引入的 MobiLink 更改。

MobiLink 服务器更改

对 MobiLink 脚本的更改

- **删除了基于游标的上载** 以下脚本在 9.0.0 版中不建议使用，现在已将它们删除：`upload_cursor`、`new_row_cursor` 和 `old_row_cursor`。您应该改用基于语句的脚本。
请参见“编写用于上载行的脚本”一节《MobiLink - 服务器管理》。
- **无法识别的脚本会导致同步失败** 如果 MobiLink 服务器遇到任何无法识别的表级或连接级脚本，它将中止同步。在以前的版本中，无法识别的脚本只产生警告消息。这意味着基于游标的上载脚本的存在会导致同步中止。
- **上载或下载脚本中的错误会导致同步失败** 现在，MobiLink 服务器遇到上载或下载脚本中的错误时会始终中止同步。以前，MobiLink 服务器不会始终中止同步。
- **handle_error 和 handle_odbc_error 事件在更受限制的方式下工作** 现在，只有当 MobiLink 在上载事务期间处理插入、更新或删除脚本或者当其读取下载行时发生 ODBC 错误，才调用 `handle_error` 和 `handle_odbc_error` 脚本。如果 ODBC 错误在其它时间发生，则 MobiLink 服务器调用 `report_error` 或 `report_odbc_error` 脚本并中止同步。
- **提交验证脚本** 如果没有错误，MobiLink 服务器将始终在调用 `authenticate_user`、`authenticate_user_hashed` 或 `authenticate_parameters` 脚本后提交事务，即使验证失败。以前，涉及失败验证的事务被回退，因此可能没有失败的验证尝试的记录。请参见：
 - “authenticate_user 连接事件”一节《MobiLink - 服务器管理》
 - “authenticate_user_hashed 连接事件”一节《MobiLink - 服务器管理》
 - “authenticate_parameters 连接事件”一节《MobiLink - 服务器管理》
- **对 authenticate_user_hashed 脚本的更改** 现在可以在用户验证序列过程中多次调用 `authenticate_user_hashed` 脚本。
请参见“authenticate_user_hashed 连接事件”一节《MobiLink - 服务器管理》。
- **调用开始脚本后，无论同步是否成功，都将调用其结束脚本** 有一些 MobiLink 脚本具有开始和结束格式，如 `begin_connection` 和 `end_connection`。过去，如果同步失败，则通常不执行结束脚本。现在，如果调用了开始脚本，则始终调用结束脚本（如果已定义），即使同步有错误。
请参见“同步事件”一节《MobiLink - 服务器管理》。
- **没有要上载的数据时不调用表的上载脚本** 在以前的版本中，可以使用 `-us` 选项防止 MobiLink 服务器在没有要上载的数据时调用上载脚本。现在删除了 `-us` 选项，并且缺省情况下，MobiLink 服务器仅在上载流包含要上载的数据时才调用上载脚本。可以使用 `-zus` 选项恢复为原来的行为。
请参见“-zus 选项”一节《MobiLink - 服务器管理》。
- **SQL Anywhere 10 和 Microsoft SQL Server 2005 统一数据库不应更改 begin_connection 脚本中的隔离级别** 对于 SQL Anywhere 版本 10 和 Microsoft SQL Server 2005 及更高版本的统一数据库，下载的缺省隔离级别现在是快照。这意味着可以在开始下载事务时更改隔离级别，此种情况下将覆盖 `begin_connection` 脚本中的任何设置。因此，您应该在 `begin_download` 脚本中更改下载的隔离级别，或使用新的 `mlsrv10 -dsd` 选项来禁用快照隔离。以前的文档建议在

`begin_connection` 脚本中更改隔离级别，对于不使用快照隔离的统一数据库而言，这仍是一个好的做法。

请参见“[MobiLink 隔离级别](#)”一节《[MobiLink - 服务器管理](#)》。

- **删除了 `example_upload_cursor`、`example_upload_delete`、`example_upload_insert` 和 `example_upload_update` 表事件** 由于删除了 `-za` 和 `-ze` MobiLink 服务器选项，将不再生成 `example_upload_cursor`、`example_upload_delete`、`example_upload_insert` 和 `example_upload_update` 表事件。现在可以使用 [创建同步模型向导] 生成脚本。

请参见“[创建模型](#)”一节《[MobiLink - 入门](#)》。

mlsrv10 更改

- **`-w` 和 `-wu` 选项发生了变化** `-w` 选项和 `-wu` 选项分别用于设置数据库工作线程数和上载数据库工作线程的最大数。在以前的版本中，这些工作线程执行同步的每个方面，包括读取和写入网络、压缩和解压缩协议字节和行数据、运行脚本以及更新和读取统一数据库中的行。

现在 `-w` 和 `-wu` 选项仅影响数据库工作线程。这些数据库工作线程仅负责所有的数据库活动，不负责其它活动。其它线程负责网络活动、压缩和解压缩以及所有其它 MobiLink 服务器活动。

`-w` 选项和 `-wu` 选项的新行为与网络活动无关。在以前的版本中，具有长等待时间的网络可能会导致工作线程阻塞，从而需要某些部署来指定大的工作线程数。新的 MobiLink 体系结构取消了此需求。

要限制 MobiLink 服务器对统一数据库施加的负载，最简单的方法仍然是使用 `-w` 和 `-wu` 选项。测试 `-w` 和 `-wu` 值可以帮助您找到同步系统的最佳吞吐量。请参见：

- “[-w 选项](#)”一节《[MobiLink - 服务器管理](#)》
- “[-wu 选项](#)”一节《[MobiLink - 服务器管理](#)》
- “[MobiLink 性能](#)” 《[MobiLink - 服务器管理](#)》

- **删除了用于设置高速缓存大小的选项** 删除了以下 `mlsrv10` 选项：

- `-bc`
- `-d`
- `-dd`
- `-u`

这些选项已被 `mlsrv10 -cm` 选项所替代，该选项为所有同步设置高速缓存。

请参见“[-cm 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **删除了用于设置超时的选项** 以下 `mlsrv10` 选项不再需要并已被删除：

- `contd_timeout`
- `unknown_timeout`

还删除了 `mlsrv10 liveness_timeout` 选项。它由用于同步客户端的 `timeout` 选项所替代。

请参见“[timeout](#)”一节《[MobiLink - 客户端管理](#)》。

- **不再需要 `backlog` 选项** 不再需要 `mlsrv10 backlog` 选项并将其删除。

- **更改了协议名称和网络选项 security** 删除了以下网络协议关键字：`https_fips`、`rsa_tls`、`rsa_tls_fips`、`ecc_tls`；以及网络协议选项 `security`。网络协议未被删除，但现在可以不同的方式指定这些协议。`mlsrv10 -x` 语法发生了如下变化：

旧语法	10.0.0 版中的新语法	说明
<code>-x https_fips</code>	<code>-x https(fips=y;...)</code>	具有 FIPS 的 HTTPS
<code>-x rsa_tls</code>	<code>-x tls(tls_type=rsa;...)</code>	具有使用 RSA 加密的 TLS 的 TCP/IP
<code>-x rsa_tls_fips</code>	<code>-x tls(tls_type=rsa;...)</code>	具有使用 RSA 加密的 TLS 和 FIPS 的 TCP/IP
<code>-x ecc_tls</code>	<code>-x tls(tls_type=ecc;...)</code>	具有使用 ECC 加密的 TLS 的 TCP/IP
<code>-x tcpip(security=...)</code>	<code>-x tcpip</code>	TCP/IP
<code>-x http(security=...)</code>	<code>-x http</code>	HTTP

请参见“[-x 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **更改了 -bn 选项** `mlsrv10 -bn` 选项在冲突检测过程中对 BLOB 字节进行比较。以前，会对 LONGVARCHAR 类型的数据进行字符比较。现在，进行比较的单位始终是二进制和 LONGVARCHAR BLOB 的字节。

请参见“[-bn 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **更改了详细程度输出** `mlsrv10` 选项 `-vr`、`-vt` 和 `-vu` 输出的信息都稍有不同：

- `-vr` 现在，`-vr` 只返回上载和下载行值。以前，还返回上载和下载脚本的名称和内容。
- `-vt` 现在，`-vt` 只返回转换的脚本的内容。以前，还返回原始脚本内容。
- `-vu` 现在，`-vu` 在需要调用脚本时返回所有未定义的表脚本。其中包括统计脚本。

请参见“[-v 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **删除了 MobiLink 服务器选项 -za 和 -ze** 删除了由 MobiLink 服务器 `-za` 选项和 `-ze` 选项提供的自动脚本生成功能。现在可以使用 [创建同步模型向导] 生成脚本。

请参见“[创建模型](#)”一节《[MobiLink - 入门](#)》。

- **删除了 -zac 和 -zec** 不建议使用用于生成基于游标的脚本的 MobiLink 服务器选项 `-zac` 和 `-zec`，现在已将它们删除。

- **删除了 MobiLink 服务器选项 -oy** `mlsrv10 -oy` 选项（用于显示时间戳中的年）已被删除。现在始终在信息性消息、警告消息和错误消息的时间戳中显示年。

统计属性

- **对统计属性进行了更改** 有两个常规更改：

- 上载和下载的字节计数的含义发生了变化。现在，字节计数反映了 MobiLink 服务器中用于存储上载和下载的内存量。以前，是指上载和下载在发送到或接收自 MobiLink 服务器时的

大小（以字节为单位）。新的计数方法可以更好地指示同步对服务器内存的影响，所以更有用。此外，使用 HTTP、加密或压缩时，以前的计数方法可能不可靠。

- 在本文档的以前版本中，属性说明不解释在不同模式下（正常上传模式或强制冲突模式）使用属性时，属性会返回不同值。对此进行了更正（请参见以下内容）。

以下统计属性已更改：

统计属性	说明
conflicted_deletes	<p>在正常上传模式下，其值始终为零。</p> <p>在强制冲突模式下，它返回使用 <code>upload_old_row_insert</code> 脚本成功插入到统一数据库中的上传删除行的总数。</p> <p>以前，它返回检测到有冲突的上传删除行数。</p>
conflicted_inserts	<p>在正常上传模式下，其值始终为零。</p> <p>在强制冲突模式下，它返回使用 <code>upload_new_row_insert</code> 脚本成功插入到统一数据库中的上传插入行的总数。</p> <p>以前，它返回检测到有冲突的上传插入行数。</p>
conflicted_updates	<p>在正常上传模式下，它返回导致冲突的更新行的总数。</p> <p>在强制冲突模式下，它返回使用 <code>upload_new_row_insert</code> 或 <code>upload_old_row_insert</code> 脚本成功应用的上传更新行的总数。</p> <p>以前，它返回检测到有冲突的上传更新行数。</p>
download_bytes	<p>它返回 MobiLink 服务器内用来存储下载的内存量。</p> <p>以前，它返回下载的字节数。</p>
ignored_deletes	<p>在正常上传模式下，如果定义了 <code>handle_error</code> 或 <code>handle_odbc_error</code> 并返回 1000，或者对给定表没有定义 <code>upload_delete</code> 脚本，此属性会返回调用 <code>upload_delete</code> 脚本时导致出错的上载删除行的总数。</p> <p>在强制冲突模式下，如果定义了 <code>handle_error</code> 或 <code>handle_odbc_error</code> 并返回 1000，或者对给定表没有定义 <code>upload_old_row_insert</code> 脚本，此属性将返回调用 <code>upload_old_row_insert</code> 脚本时导致出错的上载删除行的总数。</p> <p>以前，它返回忽略的上载删除行数。</p>
ignored_inserts	<p>在正常上传模式下，如果定义了 <code>handle_error</code> 或 <code>handle_odbc_error</code> 并返回 1000，或者对给定表没有定义 <code>upload_insert</code> 脚本，此属性将返回调用 <code>upload_insert</code> 脚本时导致出错的上载插入行的总数。</p> <p>在强制冲突模式下，如果定义了 <code>handle_error</code> 或 <code>handle_odbc_error</code> 并返回 1000，或者对给定表没有定义 <code>upload_new_row_insert</code> 脚本，此属性将返回调用 <code>upload_new_row_insert</code> 脚本时导致出错的上载插入行的总数。</p> <p>以前，它返回忽略的上载插入行数。</p>

统计属性	说明
ignored_updates	<p>在正常上载模式下，如果定义了 handle_error 或 handle_odbc_error 并返回 1000，或者对给定表没有定义 upload_update 脚本，此属性将返回调用 upload_update 脚本时导致出错的上载更新行的总数。</p> <p>在强制冲突模式下，如果定义了 handle_error 或 handle_odbc_error 并返回 1000，此属性将返回调用 upload_new_row_insert 或 upload_old_row_insert 脚本时导致出错的上载更新行的总数。</p> <p>以前，它返回忽略的上载更新行数。</p>
upload_bytes	<p>它返回 MobiLink 服务器内用来存储上载的内存量。</p> <p>以前，它返回上载的字节数。</p>
upload_deleted_rows	<p>在正常上载模式下，它返回从统一数据库中成功删除的行的总数。</p> <p>在强制冲突模式下，其值始终为零。</p> <p>以前，它返回从同步客户端上载的行删除的数目。</p>
upload_inserted_rows	<p>在正常上载模式下，它返回成功插入到统一数据库中的行的总数。</p> <p>在强制冲突模式下，其值始终为零。</p> <p>以前，它返回从同步客户端上载的行插入的数目。</p>
upload_updated_rows	<p>在正常上载模式下，它返回统一数据库中成功更新的行的总数。</p> <p>在强制冲突模式下，其值始终为零。</p> <p>以前，它返回从同步客户端上载的行更新的数目。</p>

请参见“MobiLink 统计信息属性”一节《MobiLink - 服务器管理》。

其它 MobiLink 服务器更改

- **空值字符现在可与远程数据库中 CHAR 或 NCHAR 数据类型的列进行同步** 以前在 MobiLink 中，包含空值字符的 VARCHAR 和 CHAR 列值可能导致同步失败。现在，您可以对远程数据库中数据类型为 CHAR、VARCHAR、LONG VARCHAR、NCHAR、NVARCHAR 和 LONG NVARCHAR 的列中的空值字符进行同步。
- **用于记录信息性、警告和错误消息的新格式** 以前，MobiLink 服务器以以下格式记录消息：

```
T.mm/dd hh:mm:ss. thread_id User_name: message
```

现在，MobiLink 服务器采用以下格式记录消息：

```
T. yyyy-mm-dd hh:mm:ss. synchronization_id: message
```

对于每个同步，日志中的第一条消息都显示远程 ID、用户名、脚本版本和客户端名称（UltraLite 或 SQL Anywhere）。

新格式在没有减少提供的信息的情况下减少了输出日志的大小。

- **系统过程中用于 Oracle 的新数据类型** 在用于注册脚本的 MobiLink 系统过程中，脚本内容参数现在将 CLOB 数据类型用于 Oracle 统一数据库。在 ml_add_property 系统过程中，prop_value 参数现在为用于 Oracle 的 CLOB。以前，这些参数的类型为 VARCHAR。

请参见“MobiLink 服务器系统过程”《MobiLink - 服务器管理》。

MobiLink 客户端更改

MobiLink 用户名和远程 ID

MobiLink 在远程数据库第一次进行同步时（或遇到远程 ID 的值为 NULL 时）生成称为远程 ID 的唯一 ID。MobiLink 用户名不再必须唯一。现在，可以将 MobiLink 用户名视为用于验证的真实用户名。

在以前的版本中，为 MobiLink 用户名存储同步进度。现在，为 SQL Anywhere 远程数据库的远程 ID 和预订以及 UltraLite 远程数据库的远程 ID 和发布存储同步进度。

请参见“ml_subscription”一节《MobiLink - 服务器管理》。

以前，使用 MobiLink 用户名来唯一标识远程数据库。当您希望 MobiLink 用户与多个远程数据库进行同步时，可使用远程 ID 标识远程数据库。在 UltraLite 远程数据库中，远程 ID 也可用于允许多个 MobiLink 用户与同一远程数据库进行同步。

请参见“远程 ID”一节《MobiLink - 客户端管理》。

UltraLite 客户端

请参见“行为更改和不建议使用的功能”一节第 203 页。

SQL Anywhere 客户端

- **不建议使用下载错误挂接** 不建议使用以下错误挂接：sp_hook_dbmlsync_download_com_error、sp_hook_dbmlsync_fatal_sql_error 和 sp_hook_dbmlsync_sql_error。它们已被替换。

请参见“处理事件挂接过程中的错误和警告”一节《MobiLink - 客户端管理》。

- **只有在 dbmlsync 期望另一个同步时才会调用 sp_hook_dbmlsync_log_rescan** 以前，在每个同步结束时调用 sp_hook_dbmlsync_log_rescan 挂接。这会导致在 dbmlsync 断开与 MobiLink 服务器的连接之后但在日志中显示 [同步已完成] 消息之前出现暂停。现在，只有在 dbmlsync 期望另一同步时（例如，当在命令行中多次指定 dbmlsync -n 选项或启用调度时）才调用此挂接。

请参见“sp_hook_dbmlsync_log_rescan”一节《MobiLink - 客户端管理》。

- **简化了活动超时选项** 在客户端上删除了网络连接协议选项 liveness_timeout 和 network_connect_timeout。改用 timeout 连接选项。

请参见“timeout”一节《MobiLink - 客户端管理》。

- **不压缩意味着不进行模糊处理** 现在，如果将 compression 设置为 none，数据完全是非模糊的。如果需要考虑安全性，应使用传送层安全来加密数据。

请参见“Compression”一节《MobiLink - 客户端管理》。

- **删除了版本 7 语法和实用程序** 不建议使用以下 SQL 语句和实用程序，现在已将其删除：

- MobiLink 客户端数据库抽取实用程序 (mlxtract)
- CREATE SYNCHRONIZATION SITE 语句
- CREATE SYNCHRONIZATION DEFINITION 语句
- CREATE SYNCHRONIZATION TEMPLATE 语句
- **用于 ActiveSync 的新网络协议选项** 指定 CommunicationAddress 扩展选项或在 SQL 语句中指定 ADDRESS 子句时，ActiveSync 用户不必再指定 ActiveSync 协议。对于 ActiveSync，您只需指定用于在 ActiveSync 的 MobiLink 提供程序和 MobiLink 服务器间进行通信的协议和协议选项。
请参见“MobiLink 客户端网络协议选项汇总”一节《MobiLink - 客户端管理》。
- **关闭 dbmlsync 的新方法** 不建议使用 dbmlsync -k 选项，该选项已由 -qc 选项代替。
请参见“-qc 选项”一节《MobiLink - 客户端管理》。

MobiLink 的行为更改

版本支持

- **已删除对版本 8.0.0 之前的客户端的支持** MobiLink 服务器不再支持版本 8.0.0 之前的 SQL Anywhere 客户端。要将以前版本的数据库与版本 10 MobiLink 服务器一起使用，需要执行升级过程。
请参见“升级 SQL Anywhere”一节第 324 页。

名称更改

以下实用程序名称已更改：

旧实用程序名称	新实用程序名称
dbmlsrv9	mlsrv10
dbmluser	mluser
dbmlmon	mlmon
dbmlstop	mlstop
dbasinst	mlasinst

以下文件名称已更改：

旧文件名称	新文件名称
<i>dbmlsv9.dll</i>	<i>mlodbc10.dll</i>
<i>dbasdesk.dll</i>	<i>mlasdesk.dll</i>

旧文件名称	新文件名称
<i>dbasdev.dll</i>	<i>mlasdev.dll</i>
<i>dbmlsrv.mle</i>	<i>mlsrv10.mle</i>
<i>syncasa.sql</i>	<i>syncsa.sql</i>

ODBC 驱动程序增强

现在，MobiLink 将新驱动程序用于 Adaptive Server Enterprise 和 DB2 统一数据库。

请参见：“对 MobiLink、QAnywhere 和远程数据访问所使用的 ODBC 驱动程序进行的更改”一节第 214 页。

服务器启动的同步

- **Windows SDK 已删除** 用于创建对多个 Windows 设备的支持的 SDK 已被删除。它由对 SMS 的改进支持所代替。但 Palm SDK 仍保留。
- **监听器 -g 选项被替换** dbsln -g 选项被替换为 dblsn -ni 选项。

其它 MobiLink 行为更改

- **删除了对 Windows 性能监控器的支持** MobiLink 不再支持 Windows 性能监控器。应改用 MobiLink 监控器。
请参见“MobiLink 监控器”《MobiLink - 服务器管理》。
- **服务器启动的同步不再支持 Kyocera 设备** 不存在用于 Kyocera 设备的 Palm SDK。Palm 监听器继续支持 Treo 设备。
请参见“用于 Palm 设备的 MobiLink 监听器 C API”《MobiLink - 服务器启动的同步》。

QAnywhere

以下各节介绍 10.0.0 版的 QAnywhere 的新功能、行为更改和不建议使用的功能。

新功能

以下是 10.0.0 版中引入的 QAnywhere 新增功能的列表。

移动 Web 服务

移动 Web 服务提供了对优化的移动异步 Web 服务的支持。这样一来，移动应用程序便可提出 Web 服务请求—即使在应用程序处于离线状态时—并让这些请求排队等待以后进行传输。这些请求将被作为使用 QAnywhere 的消息进行传送。服务器端的 Web 服务连接器将会收到客户端请求，并将其转发到 Web 服务。然后，连接器会收到来自 Web 服务的响应，并将其作为消息返回到客户端。提供了一个 WSDL 编译器，该编译器有助于通过 .NET 或 Java 应用程序使用移动 Web 服务。

请参见“[移动 Web 服务](#)”《QAnywhere》。

用于 Sybase Central 的新 QAnywhere 插件

Sybase Central 现在包括一个 QAnywhere 插件，此插件提供用于创建和管理 QAnywhere 应用程序的、易于使用的图形界面。使用 QAnywhere 插件，可以执行以下操作：

- 创建客户端和服务器消息存储库。
- 为 QAnywhere 代理创建和维护配置文件。
- 浏览 QAnywhere 代理日志文件。
- 创建或修改目标别名。
- 创建 JMS 连接器和 Web 服务连接器。
- 创建和维护传输规则文件。
- 远程浏览消息存储库。
- 跟踪消息。

尽管 Unix 平台不支持 QAnywhere，但现在您可以在 Unix 上使用 Sybase Central 跟踪消息。

新 QAnywhere 客户端 API

- **新 SQL API** QAnywhere SQL API 是使开发人员能够轻松地利用 QAnywhere 消息传递功能的一组 SQL 存储过程。利用此 API，存储过程可以通过简单的方法来发送或接收消息，从而补充了现有的数据库应用程序。这样一来，应用程序的功能便会变得非常强大，可以在单个事务中将数据库操作和消息传递操作结合起来。例如，某存储过程可以将行插入数据库，并将消息发送到另一应用程序—并且这两个操作均作为同一事务的一部分进行提交。

请参见“[QAnywhere SQL API 参考](#)”《QAnywhere》。

- **新 Java 客户端 API** 用于 Java 的新 QAnywhere 客户端 API 有助于您在 Java 中创建消息传递客户端应用程序。Windows（包括 Windows CE）当前支持用于 Java 的 QAnywhere 客户端 API。

请参见“[QAnywhere Java API 参考](#)” [《QAnywhere》](#)。

QAnywhere 客户端 API 增强

QAnywhere 客户端 API 已增加以下功能：

- **消息选择程序** 现在您可以使用类似于 SQL 的表达式，来有选择地浏览或接收来自队列的消息。创建消息选择程序的语法与用于传输规则中的条件的语法完全相同。
请参见“[浏览 QAnywhere 消息](#)”一节 [《QAnywhere》](#)。
- **浏览消息的新方法** 现在您可以基于 ID 或消息选择程序，浏览多个队列中的消息或消息子集。
请参见“[使用选择程序浏览消息](#)”一节 [《QAnywhere》](#)。
- **枚举消息存储库属性名** 现在您可以枚举消息存储库属性名。
请参见“[枚举客户端消息存储库属性](#)”一节 [《QAnywhere》](#)。
- **无法传送的消息** 您可以使用新的消息存储库属性 `ias_MaxDeliveryAttempts`，来设置 QAnywhere 客户端在其认为某消息无法传送之前尝试接收该消息的最大次数。
请参见“[规则变量](#)”一节 [《QAnywhere》](#)。
- **取消消息** 现在您可以在消息发送之前将其取消。
请参见“[取消 QAnywhere 消息](#)”一节 [《QAnywhere》](#)。
- **查询消息状态** 现在，可以使用以下两个新的预定义消息属性来查询消息的状态：`ias_Status` 和 `ias_StatusTime`。还可以使用 `ias_Originator` 和 `ias_DeliveryCount`，分别查询消息的发出方以及将消息传送到接收方的次数。
请参见“[预定义的消息属性](#)”一节 [《QAnywhere》](#)。
- **可设置上载增量的新的消息存储库属性** `ias_MaxUploadSize` 可用于更改上载增量。
请参见“[预定义的客户端消息存储库属性](#)”一节 [《QAnywhere》](#)。

QAnywhere 代理的新功能

- **在单个设备上运行多个代理** 以前，在一个设备上只能运行一个 QAnywhere 代理实例。此限制现已取消。
请参见“[启动 QAnywhere 代理](#)”一节 [《QAnywhere》](#)。
- **增加了设置故障转移的选项** 提供了两个新的 QAnywhere 代理选项 `-fd` 和 `-fr`，可帮助您自定义故障转移发生的方式。
请参见“[-fd 选项](#)”一节 [《QAnywhere》](#) 和“[-fr 选项](#)”一节 [《QAnywhere》](#)。
- **持久连接** 增加了新选项 `-pc+` 以启用消息传输的持久连接。新选项 `-push` 替代了 `-push_notifications`，该选项现在允许您指定是否希望推式通知使用持久连接。
请参见：
 - “[-pc 选项](#)”一节 [《QAnywhere》](#)
 - “[-push 选项](#)”一节 [《QAnywhere》](#)

- **新升级过程** 新选项 `-sur` 可用于将客户端消息存储库从 SQL Anywhere 的先前版本进行升级。请参见“[-sur 选项](#)”一节《QAnywhere》。
- **QAnywhere 代理现在将 ISO 8601 日期时间格式用于消息时间戳** 现在，信息性消息、警告消息和错误消息中的时间戳都使用明确的 ISO 8601 日期时间格式：`{I|W|E} yyyy-mm-dd hh:mm:ss message`。

其它 QAnywhere 增强

- **目标别名** 现在您可以定义表示一组 QAnywhere 目标的目标别名。发送到某目标别名的消息将被发送给该别名的每个成员。请参见“[目标别名](#)”一节《QAnywhere》。
- **服务器管理请求** 现在您可以使用服务器管理请求来管理和监控活动，例如创建目标别名或监控、启动和停止 JMS 连接器。您在客户端创建服务器管理请求，然后将其发送到服务器消息存储库进行处理。请参见“[服务器管理请求](#)”一节《QAnywhere》。
- **改进了服务器传输规则的维护** 现在，可以更改缺省的服务器传输规则，并且此更改会自动应用于所有客户端。以前，要更改缺省的服务器传输规则，必须为每个客户端手工定义传输规则。请参见“[服务器传输规则](#)”一节《QAnywhere》。
- **增加了消息属性** QAnywhere 还设置了一些其它的预定义消息属性，从而为您在处理消息时提供更大的灵活性、在调试时提供更详细的信息、在解决消息状态问题时提供更多的帮助。请参见“[消息属性](#)”一节《QAnywhere》。
- **可以在 JMS 目标中嵌入反斜线** 现在 JMS 目标可以包括需要用反斜线作为分隔符的子上下文。请参见“[向 JMS 连接器发送 QAnywhere 消息](#)”一节《QAnywhere》。
- **新传输规则函数** 增加了以下传输规则函数以改进对日期的处理：
 - DATEADD(datepart, count, datetime)
 - DATEPART(datepart, date)
 - DATETIME(string)
 - LENGTH(string)
 - SUBSTR(string, start, length)
 请参见“[规则函数](#)”一节《QAnywhere》。
- **在传输规则中为属性加上前缀** 现在，当在传输规则中使用消息属性名和消息存储库属性名时，您可以为它们加上前缀，这样就可以忽略为名称相同的传输规则变量提供的优先级。请参见“[将属性用作规则变量](#)”一节《QAnywhere》。

行为更改和不建议使用的功能

以下列出了 10.0.0 版中引入的 QAnywhere 更改。

QAnywhere 客户端更改

- **客户端消息存储库 ID 已经更改** 现在，客户端消息存储库 ID 是一个 MobiLink 远程 ID。而以前它则是 MobiLink 用户名。您不必向统一数据库注册远程 ID。但是，您仍然需要向服务器消息存储库注册 MobiLink 用户名。如果未指定 MobiLink 用户名，则该用户名将缺省为客户端消息存储库 ID。

提供了几个新的 `qaagent` 选项以管理您的 MobiLink 用户名。请参见：

- “`-mn` 选项”一节 《QAnywhere》
- “`-mp` 选项”一节 《QAnywhere》
- “`-mu` 选项”一节 《QAnywhere》

- **新 ODBC 驱动程序** 用于连接到 Adaptive Server Enterprise 和 DB2 服务器消息存储库的 iAnywhere Solutions ODBC 驱动程序已经更改。

请参见“对 MobiLink、QAnywhere 和远程数据访问所使用的 ODBC 驱动程序进行的更改”一节第 214 页。

QAnywhere 代理更改

- **`qaagent -port` 已删除** `-port` 选项指定了 QAnywhere 代理监听来自监听器的通信的端口号。现在已不再需要此选项，并已将其删除。现在，将自动使用自由端口。

- **`qaagent -la_port` 已被替代** `-la_port` 选项已由 `-lp` 选项替代。

请参见“`-lp` 选项”一节 《QAnywhere》。

- **`qaagent -push_notifications` 已重命名** 此选项现在名为 `-push`。现在，该选项允许您使用（或不使用）持久连接来启用推式通知。

请参见“`-push` 选项”一节 《QAnywhere》。

- **对策略缺省值进行了更改** 现在缺省策略为 [自动]。以前，缺省策略为 [调度]。现在，缺省的调度时间间隔是 900 秒（15 分钟）。而以前则是 10 秒。

请参见“`-policy` 选项”一节 《QAnywhere》。

- **不使用或维护事务日志** QAnywhere 代理不再使用事务日志或管理其大小。因此，对于大多数应用程序而言，应使用 `dbinit -n` 选项创建客户端消息存储库，这个选项将在不使用事务日志的情况下初始化数据库。

请参见“设置客户端消息存储库”一节 《QAnywhere》。

其它 QAnywhere 更改

- **不建议使用服务器端属性文件** 现在，将属性存储在文件中，而不是数据库中。

- **`getPropertyNames`** `getPropertyNames` 函数已从 C++ 客户端 API 中删除。取而代之的是 `beginEnumPropertyNames`、`nextPropertyName` 和 `endEnumPropertyNames`。

请参见“`QAMessage` 类”一节 《QAnywhere》。

● **传输规则中的日期处理** 以下传输规则消息存储库变量已删除:

- ias_CurrentDayOfWeek
- ias_CurrentDayOfMonth
- ias_CurrentMonth
- ias_CurrentYear

可以使用 ias_CurrentTimestamp 或 DATEPART 来代替它们。

请参见“规则变量”一节《QAnywhere》。

● **QAnywhere Central 已被替代** QAnywhere Central 已被 Sybase Central 的 QAnywhere 插件所替代。该插件提供了许多功能上的增强。

SQL Remote

以下各节介绍了 10.0.0 版的 SQL Remote 中的新功能、行为更改和不建议使用的功能。

新功能

以下列出了 10.0.0 版中引入的 SQL Remote 新增功能。

- **invalid_extensions 选项** 增加了一个新的消息传递选项，您可使用此选项阻止 SQL Remote 在 FILE 和 FTP 消息传递中使用某些文件扩展名。
请参见“[SET REMOTE OPTION 语句 \[SQL Remote\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **在 Unix 和 Linux 平台上，消息代理 (dbremote) 具有图形用户界面。** 请参见“[消息代理 \(dbremote\)](#)”一节《[SQL Remote](#)》中的 -ux 选项。
- **dbremote 现在将 ISO 8601 日期时间格式用于消息时间戳** 现在，信息性消息、警告消息和错误消息中的时间戳都使用明确的 ISO 8601 日期时间格式：`{I|W|E} yyyy-mm-dd hh:mm:ss message`。
- **dbremote 的新选项** 要在完成时关闭窗口，请使用新 -qc 选项。
请参见“[消息代理 \(dbremote\)](#)”一节《[SQL Remote](#)》。

行为更改和不建议使用的功能

以下列出了 10.0.0 版中引入的 SQL Remote 更改。

- **对 Adaptive Server Enterprise 数据库的支持已删除** SQL Remote 不再支持 Adaptive Server Enterprise 统一数据库。这表示 ssxtract、ssremote、ssqueue，以及所有其它 SQL Remote for Adaptive Server Enterprise 实用程序和文件均已从安装中删除。
要同步 Adaptive Server Enterprise 数据库，应使用 MobiLink。
有关从 SQL Remote 升级到 MobiLink 的信息，请参见 http://www.iAnywhere.com/whitepapers/migrate_to_ml.html。
- **[抽取数据库向导] 不再抽取旧数据库** [抽取数据库向导] 仅适用于版本 10 数据库。
- **#hook_dict 中的扩展值** dbxtract 和 dbremote 实用程序提供挂接，并通过名为 #hook_dict 的临时表将值作为名称/值对来传递。过去，#hook_dict 表中的值被定义为 VARCHAR(255)。此值已增大到 VARCHAR(10240)。
- **对抽取实用程序进行的更改** dbxtract 发生了以下几项更改：
 - -j、-k 和 -x 选项已删除。
 - 增加了新的 -al 和 -xh 选项。请参见“[抽取实用程序 \(dbxtract\)](#)”一节《[SQL Remote](#)》。

- **消息代理 (dbremote) 不建议使用的选项** 不建议使用在完成时关闭窗口的 `-k` 选项。要在完成时关闭窗口，请使用新 `-qc` 选项。

UltraLite

以下各节介绍了 10.0.0 版的 UltraLite 中的新功能、行为更改和不建议使用的功能。

新功能

以下列出了 10.0.0 版中引入的 UltraLite 新增功能。

主要功能

UltraLite 现在已成为一个功能完备的关系数据库管理系统，在对其进行设计时已将易于管理性和 SQL Anywhere 兼容性考虑在内。尽管增加了许多非常实用的新功能，但是 UltraLite 仍然保持着资源占用较少的优点。有关此发行版本的 UltraLite 限制的完整列表，请参见“[UltraLite 限制](#)”一节《[UltraLite - 数据库管理和参考](#)》。

此发行版本的主要功能包括：

- **数据库限制已提高** UltraLite 数据库限制已显著提高。特别是，表中的最大行数已增加到 1600 万行。有关当前数据库的其它限制，请参见“[UltraLite 限制](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **集成模式** UltraLite 现在是一个独立的 RDBMS，不再需要单独的模式文件来定义数据库的逻辑结构。在此发行版本中，UltraLite 模式已完全与数据库集成。有关内部数据库模式的详细信息，请参见“[UltraLite 数据库模式](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **统一文件格式** 在 10 版的 UltraLite 中，文件格式已经统一。这表示现在大多数平台都可以共享一个数据库文件。如果您需要的归类中没有定义您所需的字符，则现在您应选择使用 UTF-8 对您的数据库进行编码。有关详细信息，请参见“[字符集编码的 UltraLite 平台要求](#)”一节《[UltraLite - 数据库管理和参考](#)》和“[UltraLite utf8_encoding 创建参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **数据库性能和数据完整性已提高** 总的来说，UltraLite 数据库性能和数据完整性已随着索引和数据库页管理的多项改进而得到了提高。
- **索引可以使用散列** 现在可以指定索引使用散列。可基于每个索引指定散列大小。散列大小可提高索引查找的性能，且可能影响数据库文件的大小。请参见“[UltraLite 性能与优化](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **直接创建数据库** 现在您可以直接创建 UltraLite 数据库文件；数据库模式文件或参考数据库文件不必作为 UltraLite 数据库的来源。相反，您可使用 Sybase Central 或命令行实用程序独立创建 UltraLite 数据库，甚至也可通过编程方式从应用程序创建该数据库。
现有 UltraLite 用户将无法再通过与先前版本相同的方式来创建数据库。请参见“[升级 UltraLite](#)”一节第 345 页。
- **直接支持 Windows CE** 在此发行版本中，桌面操作系统上的 UltraLite 应用程序可直接与部署到 Windows CE 设备中的数据库相连接。可通过指定路径和名称并加上 **WCE:** 前缀，来指定 UltraLite 数据库。所有客户端应用程序和管理工具（包括 Sybase Central 和 Interactive SQL）都支持这种直接访问。请参见“[Windows Mobile](#)”一节《[UltraLite - 数据库管理和参考](#)》。

- **将嵌入式 SQL 用作动态 SQL 编程接口** 在先前版本中，嵌入式 SQL 是静态接口。而在此发行版本中，它是 UltraLite 动态 SQL 接口，并且不需要 SQL Anywhere 数据库。嵌入式 SQL 还支持动态 ESQL 语句及主机变量占位符的使用。此外，ESQL 应用程序现在也可使用 `uleng10` 来运行。这可以通过链接到 `ulrtc.lib` 而不是 `ulrt.lib` 来实现。

进行此更改后您可能会注意到：简单的嵌入式 SQL 应用程序的大小会增加，而复杂的应用程序可能会变得更小。请参见“升级 UltraLite”一节第 345 页和“开发嵌入式 SQL 应用程序”一节《UltraLite - C 及 C++ 编程》。

平台和设备

平台支持已修改。有关受支持的平台的列表，请参见“支持的平台”一节《SQL Anywhere 11 - 简介》。

要注意的重要增强包括：

- **部署平台** 平台增强包括：
 - **Palm OS** 以下这些更改增强了 UltraLite 对 Palm OS 设备的支持：
 - 对 Palm OS v4.x 和更高版本的运行时支持。
 - 对 CodeWarrior 的部署支持已升级至版本 9。请注意，不再建议使用 CodeWarrior 8。
 - 支持多个数据库和通用文件名。现在，您可以使用 DBF 连接参数为多个设备指定数据库，以确保根据您使用的是基于记录的存储还是基于文件的存储，来设置正确的文件名。请参见“UltraLite DBF 连接参数”一节《UltraLite - 数据库管理和参考》和“在 UltraLite 连接参数中指定文件路径”一节《UltraLite - 数据库管理和参考》。
 - 支持 NVFS 设备和 VFS 设备。
 - **Symbian OS** 此版本的 UltraLite 新增了对 Symbian OS 的支持。UltraLite 支持 UIQ 手机（2.0 和 2.1 版）、Nokia S60（第二版）和 Series 80 设备上的 Symbian OS 7.0 和 8.0 版。
 - **Windows Mobile 2005** 如果使用的是 Embedded Visual C++ 3.0 或 4.0，则可继续使用现有运行时。然而，如果使用 Visual Studio 2005 来构建应用程序，则需要新的运行时（安装在 `ralite\ce\arm.50` 下）。
- **开发环境支持已增强** 开发工具和语言已更新，如下所示：
 - UltraLite 现在支持 Visual Studio 2003 中的 ADO.NET 1.0 开发，以及 Visual Studio 2005 中的 ADO.NET 2.0 开发。
 - UltraLite 现在支持用于 Visual Basic 和 C# 开发的 AppForge Crossfire 5.6 版。您可以将 AppForge 的应用程序部署到 Palm OS、Symbian OS 和 Windows CE 平台。
 - C++ 组件开发。

安全性

- **加密类型** 如果您正通过 TLS 进行压缩，UltraLite 现在支持 ECC 和 RSA 加密。RSA 加密不再是单独的产品。请参见“配置 UltraLite 客户端以使用传送层安全”一节《SQL Anywhere 服务器 - 数据库管理》。
- **FIPS 安全性** 现在您可以通过 FIPS 安全性确保 MobiLink 服务器通信的安全。
- **安全流已简化** 您现在可以将加密流定义为网络协议或流类型，而不使用单独的安全参数。以下是一组完整的支持流类型：TCP/IP、TLS（用于 RSA、ECC 和 FIPS）、HTTP 和 HTTPS。请参见“Stream Type 同步参数”一节《UltraLite - 数据库管理和参考》。

数据库管理

重要的新功能和增强包括：

- **口令更改** 无论数据库是否区分大小写，所有口令均区分大小写。新数据库将用 **DBA** 这一缺省的用户 ID 及 **sql** 口令进行创建。因此，在将数据库从先前版本升级时，将不会保留用户 ID、口令和受信任的根证书。
- **数据库属性和连接参数已改进** 已增强和简化了数据库属性和连接参数，使您能够更轻松描述数据库和连接行为。有关您可以为此发行版本数据库设置的数据库属性和连接参数的完整列表，请参见“UltraLite 数据库属性”《UltraLite - 数据库管理和参考》和“UltraLite 连接参数”《UltraLite - 数据库管理和参考》。
- **索引性能已提高** 此发行版本已增强了 UltraLite 索引性能。其中一项重大的改进是引入了索引散列。现在，UltraLite 具有可配置的索引散列大小。通过将散列大小设置为 1 至 32 字节的值，UltraLite 将部分或全部索引值存储在索引页中。这便减少了所需的行查找次数。请参见“UltraLite max_hash_size 创建参数”一节《UltraLite - 数据库管理和参考》。
- **校验和校验** 现在，当数据库页存储在磁盘上时，您可以在这些页上包括校验和以校验其数据完整性。请参见“UltraLite checksum_level 创建参数”一节《UltraLite - 数据库管理和参考》。
- **BLOB 支持已扩展** 现在，UltraLite 数据库扩展了对 BLOB 的支持。UltraLite 允许您对这些 BLOB 进行更新、转换其数据类型以及获取其长度。

管理工具

此发行版本已对管理工具进行了改进。为确保正确使用各个工具，请务必查看与其对应的文档。

图形管理工具

- **Sybase Central** 您现在可以在图形用户界面中使用 Sybase Central 来创建、修改和管理 UltraLite 数据库。这将取代 ulview 模式编辑实用程序。

Sybase Central 提供的向导列表中包括：

- [创建数据库向导]，用于构建新的 UltraLite 数据库。此向导具有与 ulcreate 实用程序相同的功能。

- [消除数据库向导], 用于消除现有的 UltraLite 数据库。没有等效于此向导的实用程序。
- [抽取数据库向导], 用于从 SQL Anywhere 参考数据库初始化新的 UltraLite 数据库。此向导具有与 ulinit 实用程序相同的功能。
- [装载数据库] 向导, 用于将 XML 文件装载到 UltraLite 数据库中。此向导具有与 ulload 实用程序相同的功能。
- [迁移 C++ API] 向导, 用于迁移使用已删除的 ulgen 实用程序创建的 C/C++ 代码。没有等效于此向导的实用程序。
- [同步数据库] 向导, 用于同步 UltraLite 数据库。此向导具有与 ulsync 实用程序相同的功能。
- [升级数据库向导], 用于将现有 UltraLite 数据库从先前版本进行升级。此向导在与 ulload 实用程序配合使用时, 具有与 ulunloadold 实用程序相同的功能。
- [卸载数据库], 用于将数据/模式信息从 UltraLite 数据库卸载到 XML、SQL 或另一数据库。此向导具有与 ulunload 实用程序相同的功能, 同时还具有 ulcreate 和 ulload 实用程序的附加功能。
- **Interactive SQL** 现在可以使用 Interactive SQL 来开发和测试 SQL 语句以及 UltraLite 数据库。Interactive SQL 取代了在先前版本中使用的 ulisql 实用程序。请参见“[UltraLite 的 Interactive SQL 实用程序 \(dbisql\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。

命令行管理工具

以下是 UltraLite 新增的命令行实用程序:

- **卸载旧的数据库实用程序** 新的 ulunloadold 命令行实用程序可帮助您将现有的 8.0.2 或 9.x 版 UltraLite 数据库 (模式和数据) 或模式文件卸载到 XML 文件中。您可以随后使用 ulload 命令行实用程序的输出, 重建一个 10 版的 UltraLite 数据库。请参见“[UltraLite 卸载旧数据库实用程序 \(ulunloadold\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **信息实用程序** 新的 ulinfo 实用程序可显示有关 UltraLite 数据库的信息。它还可以更改和/或删除数据库选项 ID, 如 global_id 或 ml_remote_id。请参见“[UltraLite 信息实用程序 \(ulinfo\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。

同样, 由于已增强了现有命令行实用程序以支持 UltraLite 10 中的新 RDBMS 功能, 因此这些实用程序的选项已从先前版本进行了更改。为确保正确使用新的实用程序, 请务必在开始使用前先查阅参考文档。有关完整的实用程序参考说明, 请参见“[UltraLite 实用程序](#)”《[UltraLite - 数据库管理和参考](#)》。

- **错误报告功能已增强** 现在, UltraLite 实用程序可以报告错误, 且与其它 SQL Anywhere 实用程序一致。
- **数据库创建参数已扩展** 现在, 所有数据库创建实用程序 (例如 ulcreate 和 ulload) 都支持使用扩展的创建参数。这些扩展的创建参数是使用 -o 在命令行上配置的, 利用这些选项您可以配置与使用 Sybase Central 向导配置的功能相同的一组功能。请参见“[为 UltraLite 选择数据库创建参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **卸载行为已增强** 现在您可以使用 ulunload 将 UltraLite 数据库模式卸载为动态 SQL 语句序列。请参见“[UltraLite 卸载旧数据库实用程序 \(ulunload\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。

- **ulsync 行为已增强** ulsync 允许您直接从此实用程序设置网络协议选项和扩展的同步参数。有关完整列表，请参见“[UltraLite 同步参数和网络协议选项](#)”一节《[UltraLite - 数据库管理和参考](#)》。此外，ulsync 现在不仅允许您命名发布掩码，而且还允许您命名发布。关键字 **Publications** 将获取以逗号分隔的发布名列表。有关详细信息，请参见“[UltraLite 同步实用程序 \(ulsync\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **管道安装已增强** HotSync 管道安装实用程序 (ulcond10) 现在支持管道扩展、连接字符串和多个数据库。有关详细信息，请参见“[Palm OS 的 UltraLite HotSync 管道安装实用程序 \(ulcond11\)](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **ulmbreg** 注册 UltraLite for AppForge 的 ulmbreg 实用程序已重命名为 ulafreg。此实用程序现在安装在 `install-dir\win32` 目录下。

ULSQLCONNECT

以前，所有 UltraLite 实用程序均从命令行接收连接信息。现在，如果您要传递缺省用户 ID 和口令以外的信息，则可在主机上设置 ULSQLCONNECT 环境变量。请参见“[使用 ULSQLCONNECT 环境变量存储 UltraLite 参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。

SQL

- **SQL 语句** UltraLite 支持几个新语句。这些新语句包括：
 - **ALTER TABLE** 除了使用 UltraLite SQL 创建表以外，现在您还可使用此语句变更表定义。请参见“[UltraLite ALTER TABLE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
 - **ALTER/CREATE/DROP PUBLICATION** 现在，UltraLite 支持使用这三个语句添加、创建和删除发布。请参见“[UltraLite ALTER PUBLICATION 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》、“[UltraLite CREATE PUBLICATION 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》和“[UltraLite DROP PUBLICATION 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
 - **START/STOP SYNCHRONIZATION DELETE** UltraLite SQL 包括这两个语句。使用这两个语句可控制如何为 MobiLink 同步记录删除操作。请参见“[UltraLite START SYNCHRONIZATION DELETE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》和“[UltraLite STOP SYNCHRONIZATION DELETE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **为约束命名** 现在可以在 ALTER TABLE 和 CREATE TABLE 语句中为表约束命名。这使您可以通过更改单个约束来修改表和列的约束，而不必修改整个表约束。请参见“[UltraLite ALTER TABLE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》和“[UltraLite CREATE TABLE 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **其它 SELECT 语句增强** SELECT 语句已扩展：
 - 现在 SELECT 语句可以包含 START AT 作为 TOP 子句的一部分。START AT 在显式限制结果集的查询中提供了更多的灵活性。请参见“[UltraLite SELECT 语句](#)”一节《[UltraLite - 数据库管理和参考](#)》。
 - 现在，DISTINCT 子句已经增强，可与集合函数（如 SUM、AVERAGE、MAX 等）配合使用。如果将集合函数与此子句配合使用，则执行时间会显著增加。请参见“[UltraLite](#)

SELECT 语句”一节 《UltraLite - 数据库管理和参考》和“SQL 函数” 《SQL Anywhere 服务器 - SQL 参考》。

- **UNION 运算符** UNION 运算符允许将来自两个或更多个查询的各结果集组合到一个结果集中。缺省情况下，UNION 运算符可以从结果集中删除重复行。请参见“使用 UNION 语句组合集合”一节 《SQL Anywhere 服务器 - SQL 的用法》。

同步

- **可为 HotSync 管道同步配置和增加缺省的高速缓存大小** 以前，如果 Palm OS 基于文件的数据存储库中同步的数据超过一定数量，则会对同步速度造成负面影响。因此，UltraLite 管道的缺省高速缓存大小（在桌面操作系统上）已增加到 4 MB。增加的高速缓存大小减少了不必要的文件 I/O 操作，从而显著缩短了同步时间。但是，您也可配置一个不同的缺省高速缓存大小（如果您选择这样做）。请参见“Palm OS 的 UltraLite HotSync 管道安装实用程序 (ulcond11)”一节 《UltraLite - 数据库管理和参考》。
- **发布谓语句** UltraLite 的同步发布现在允许使用谓语句。如果您要有选择性地将条件表达式与逻辑运算符 AND 和 OR 组合在一起，则现在可在 WHERE 或 HAVING 子句中定义此条件集。在 SQL Anywhere 中，所求得值为 UNKNOWN 的谓语句将被解释为 FALSE。“UltraLite CREATE PUBLICATION 语句”一节 《UltraLite - 数据库管理和参考》和“UltraLite ALTER PUBLICATION 语句”一节 《UltraLite - 数据库管理和参考》。
- **MobiLink 客户端网络层已改进** 对客户端网络层进行了以下改进：
 - 所有协议都支持同步压缩。
 - 持久连接，因此您可在同一连接上进行多次同步。
 - 引入了 IPv6 支持。
 - 改进了错误检测和调试。有关将 UltraLite 用作 MobiLink 客户端的详细信息，请参见“UltraLite 客户端” 《UltraLite - 数据库管理和参考》。
- **为同步设置表顺序** 从 UltraLite 客户端进行的同步现在可指定表顺序，以免在上载表过程中出现参照完整性问题。如果要为同步指定表顺序，请使用 table_order 同步参数。请参见“Additional Parameters 同步参数”一节 《UltraLite - 数据库管理和参考》或参见以下任何内容：
 - UltraLite.NET: “ULSyncParams 类”一节 《UltraLite - .NET 编程》
 - UltraLite C/C++: “ul_synch_info_a 结构”一节 《UltraLite - C 及 C++ 编程》
 - UltraLite for M-Business Anywhere: “SyncParams 类”一节 《UltraLite - M-Business Anywhere 编程》
 - UltraLite for Embedded SQL: “ULGetSynchResult 函数”一节 《UltraLite - C 及 C++ 编程》

编程接口

一般改进

- **游标更新** UltraLite 应用程序现在支持在处理游标时修改数据库数据的功能。在 SQL Anywhere 数据库中，并非所有查询结果集都允许更新和删除游标。请确保您了解允许和执行更新的各种情况。请参见“[读取数据](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
- **连接字符串已简化** 由于 UltraLite 始终会提供缺省的用户 ID DBA 以及口令 sql，因此您现在只需在连接字符串中指定数据库即可进行连接。此外，大多数数据库都可使用 DBF 连接参数进行设置。请参见“[连接到 UltraLite 数据库](#)”《[UltraLite - 数据库管理和参考](#)》。
- **引入了 MFileTransfer 函数** 将此文件传输函数与 MobiLink 文件传输实用程序配合使用可下载文件。要下载的文件可特定于一个 Mobilink 用户名或一个缺省文件。例如，应用程序可选择下载预先配置的空数据库文件，来替换本地数据库（在月初或处理周期开始时）。请参见“[MobiLink 文件传输实用程序 \(mlfiletransfer\)](#)”一节《[MobiLink - 客户端管理](#)》。
 - UltraLite for C/C++: “[MFileTransfer 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》
 - UltraLite.NET: “[ULFileTransfer 类](#)”一节《[UltraLite - .NET 编程](#)》和“[ULFileTransferProgressData 类](#)”一节《[UltraLite - .NET 编程](#)》
 - UltraLite for M-Business Anywhere: 不适用
- **数据库创建** 现在，UltraLite 模式是数据库的一部分，而不存在于一个独立的 .usm 文件之中。这意味着，应用程序创建新数据库的方式不能与先前版本所支持的方式相同。

请参见以下任何内容：

- UltraLite for C/C++: “[ULCreateDatabase 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》
- UltraLite .NET: “[ULDatabaseManager 成员](#)”一节《[UltraLite - .NET 编程](#)》
- UltraLite for M-Business Anywhere: “[createDatabase 方法](#)”一节《[UltraLite - M-Business Anywhere 编程](#)》

UltraLite C/C++

- **支持 Symbian OS** 现在通过使用 Codewarrior 或 Carbide C++ 开发环境，为 Symbian OS 平台提供了 UltraLite C/C++ 支持。
- **新增函数** 此发行版本增添了许多新函数。其中包括：
 - GetPublicationMask 函数，用于获取给定发布名的发布掩码。请参见“[IsCaseSensitive 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
 - 现在必须先调用适当的 ULEnable*Synchronization 函数，然后才能通过指定的网络协议进行同步。请参见“[ULEnableHttpSynchronization 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》、“[ULEnableHttpsSynchronization 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》、“[ULEnableTcpiSynchronization 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》、“[ULEnableTlsSynchronization 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》、“[ULEnableZlibSyncCompression 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》或“[ULEnableEccSyncEncryption 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。

- **改进的宽字符和窄字符 (ASCII) 支持** 尽管现在 UltraLite 只有一种数据库文件格式（窄字符），应用程序仍可以使用 TCHAR 的宽定义。宽字符可转换为其 MBCS 等效形式，也可根据需要将其等效形式转换为宽字符。
- **增强的函数更改** 现有函数的增强包括：
 - 如果您的应用程序不需要 SQL 支持，则使用 `ULInitDatabaseManagerNoSQL` 函数代替 `ULInitDatabaseManager` 可显著减小应用程序的大小。请参见“[ULInitDatabaseManagerNoSQL](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
 - 已增强的 `SetReadPosition` 函数可以采用第二个参数 (`offset_in_chars`)，其指明偏移量是以字节为单位还是以字符为单位。请参见“[SetReadPosition 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
 - `SetSynchInfo` 现在执行自动提交，以使所有同步信息被立即保存。
 - `ULStoreDefragInit`、`ULStoreDefragFini` 和 `ULStoreDefragStep` 将不再需要。UltraLite 现在在内部管理数据库存储库碎片整理。
 - 不建议使用 `UEnableUserAuthentication` 函数，因为用户验证会一直启用。UltraLite 现在允许最多定义四个可连接到数据库的用户名（缺省用户名为 "DBA"，口令为 "sql"）。
 - 现在，MobiLink 同步代码必须先调用 `UEnableTcpiSynchronization`，然后才能调用 `InitSynchInfo`。请参见“[InitSynchInfo 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。

UltraLite 嵌入式 SQL

UltraLite 嵌入式 SQL 不再是静态 API，并且不再需要参考数据库。而 SQL 预处理器仅需要源文件。它生成可将 SQL 语句发送到 UltraLite 的函数。先前版本所支持的一些 SQL 语句将不受 UltraLite SQL 支持。版本 10 支持动态 SQL 语句，而先前版本并不支持该语句。

- **新增函数** 此发行版本增添了许多新函数。其中包括：
 - `UEnableZlibSyncCompression` 函数在同步过程中启用 zlib 压缩。请参见“[UEnableZlibSyncCompression 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》和“[MobiLink 客户端网络协议选项](#)”《[MobiLink - 客户端管理](#)》。

注释

Palm OS 或 Symbian OS 不支持 zlib 压缩。

- **增强的函数** 现有函数的增强包括：
 - 将 `GetSQLColumnName` 添加到 `UltraLite_RowSchema_iface` 中。根据模式的类型，该函数将返回不同的结果：
 - 与 `UltraLite_TableSchema` 一起使用时，此函数返回 `column_id` 参数指定的列名。
 - 与 `UltraLite_ResultSetSchema` 一起使用时，此函数返回以下两个值之一：
 - 别名（如果为正在讨论的结果集列指定了别名）。
 - 列名（如果结果集列代表一个表列）。
 - 在所有其它情况下，均会返回空字符串。

UltraLite.NET

UltraLite 现在支持 Visual Studio 2003 中的 ADO.NET 1.0 开发，以及 Visual Studio 2005 中的 ADO.NET 2.0 开发。

- **新方法** 此发行版本增添了许多新函数。其中包括：
 - ExecuteResultSet 方法执行一条 SQL SELECT 语句，并返回可更新的结果集作为 ULResultSet 类。请参见“ExecuteResultSet 方法”一节《UltraLite - .NET 编程》。
 - ULResultSet 类包括以下方法：Append*、Set*、Delete、Update。有关这些方法的详细信息，请参见“ULResultSet 类”一节《UltraLite - .NET 编程》。
 - UltraLite.NET 现在在 TCP/IP 同步期间支持 TLS。
 - ConnectionString 属性和 ULConnectionParms 对象已被增强，以支持受限制的引用。请参见“ULConnectionParms 类”一节《UltraLite - .NET 编程》。
 - GetPublicationPredicate 方法返回指定发布的发布谓语句字符串。如果此发布不存在，则设置 SQLE_PUBLICATION_NOT_FOUND。请参见“GetPublicationPredicate 方法”一节《UltraLite - .NET 编程》。
 - SignalSyncIsComplete 方法用信号通知 ActiveSync 的 MobiLink 提供程序某个应用程序已完成同步。请参见“SignalSyncIsComplete 方法”一节《UltraLite - .NET 编程》。
 - SetDatabaseOption 方法为指定数据库选项设置值。请参见“SetDatabaseOption 方法”一节《UltraLite - .NET 编程》。
- **增强方法** 现有方法的增强包括：
 - ULSyncParms 类现在利用 TableOrder 顺序属性，来指定将表上载到统一数据库时所应遵循的顺序。请参见“AdditionalParms 属性”一节《UltraLite - .NET 编程》。
 - GetSchemaTable 方法现在返回扩展的表元数据。有关完整列表，请参见“GetSchemaTable 方法”一节《UltraLite - .NET 编程》。
 - 现在，当表位于 UL_TABLE_ACCESS_MODE_NONE 或 UL_TABLE_ACCESS_MODE_FIND_AGAIN 中时，UpdateBegin 方法在 ResultSet 级别为可选项。过去，必须要进行此更改，才能使 UltraLite.NET API 与 ADO.NET 2.0 结果集兼容。请参见“UpdateBegin 方法”一节《UltraLite - .NET 编程》。
 - GetDatabaseProperty 方法现在可以识别更多属性。请参见“GetDatabaseProperty 方法”一节《UltraLite - .NET 编程》。
 - ULSyncProgressData 类现在包括一个 Flags 属性。请参见“Flags 属性”一节《UltraLite - .NET 编程》。

UltraLite for AppForge Crossfire

UltraLite for AppForge 现在支持 Symbian OS 平台。对 UltraLite 引擎的支持已添加到此发行版本中，从而允许多个应用程序同时访问单个数据库。

- **新方法** OnWaiting 方法提供了一种机制，使用户应用程序能够处理 GUI 事件并可以取消当前操作。

UltraLite for M-Business Anywhere

- **新方法** 此发行版本增添了许多新方法。其中包括：
 - setMBA_ServerWithMoreParms 方法在使用一键同步时设置代理服务器信息。这一新方法通过添加一个名为 **additional** 的新字符串参数，增强现有 setMBA_Server 方法。
 - getPublicationMask 方法获取给定发布名的发布掩码。
 - getPublicationPredicate 方法返回指定发布的发布谓词字符串。如果此发布不存在，则设置 SQLE_PUBLICATION_NOT_FOUND。
- **增强方法** 以下现有方法的增强包括：
 - setStream 方法现在支持用于 TLS (Transport Layer security, 传送层安全性) 的 ECC (Elliptic curve cryptography, 椭圆曲线加密算法)。请参见“SyncParms 类”一节《UltraLite - M-Business Anywhere 编程》。

注意

ECC 加密并非在所有平台上均可用。有关受支持的平台的列表，请参见“支持的平台”一节《SQL Anywhere 11 - 简介》。

行为更改和不建议使用的功能

以下列出了 10.0.0 版中引入的 UltraLite 更改。

不建议使用的平台

- 此发行版本不再支持 PocketPC 2000 OS。
- 已取消对 CodeWarrior 8 的支持。必须使用 Code Warrior 9 来代替。
- 已取消对 Windows CE MIPS 处理器的支持。

删除的组件、模块、命名空间

以下编程接口已从此发行版本中删除：

- **UltraLite for ActiveX** 所有应用程序必须使用支持的 API 重写。
- **Static Java API** 所有应用程序必须使用支持的 API 重写。
- **Native UltraLite for Java** 所有应用程序必须使用支持的 API 重写。
- **Static C++ API 和 Static 嵌入式 SQL** 希望编写 C++ 应用程序的开发人员必须使用动态 C++ 接口来编程。如果使用先前版本的静态 C++ 库来编写一个应用程序，则 UltraLite 10 将包括一个迁移实用程序以简化向此新库进行的移动过程。请参见“升级 UltraLite”一节第 345 页。
- **iAnywhere.UltraLite 命名空间** 在 UltraLite.NET 中，不再支持此命名空间。相反，必须使用 iAnywhere.Data.UltraLite 命名空间来重写您的应用程序。

删除的实用程序

- **模式管理器** 因为不再需要模式文件来创建 UltraLite 数据库，所以 [模式管理器] 工具已删除。
- **数据库转换工具** 不再支持数据库转换工具 (ulconv 实用程序)。对于 ulconv 功能，可使用 ulcreate、ulload、ulsync 和 ulunload 实用程序。
- **ulxml 实用程序** 不再支持可将模式文件转换为 XML 的 ulxml 实用程序。相反，对于类似的 ulxml 功能，可使用 ulload 和 ulunload 将数据库转换为 XML。
- **ulisql** 不再支持 ulisql 实用程序。相反，Interactive SQL (dbisql) 现在支持 UltraLite。
- **ulgen** 不再支持 ulgen 实用程序。对于使用此实用程序的 UltraLite 部署，需要相应地升级您的数据库和 C/C++ 应用程序。请参见“[升级 UltraLite](#)”一节第 345 页。

删除的、不建议使用的以及已修改的功能

- **UltraLite for C/C++ API** 对 C/C++ API 中的函数和宏进行的更改包括：
 - 无法再连接到数据库模式，也无法再动态升级数据库模式，因为 .usm 文件不复存在。与 UltraLite 的这个以前功能相关的所有类和函数均已删除。
 - ULEnablePalmRecordDB 和 ULEnableFileDB 已在此版本中删除。
 - 所有 ULEnableXXXX 函数现在都必须使用初始化的 SQLCA 进行调用。
 - 版本 10 不再支持宏 UL_STORE_PARMS。调用 OpenConnection 或 CreateDatabase 时，在适当的参数中指定连接和创建选项。
 - 在此发行版本中，不建议使用 ULSecureCerticomTLSStream 和 ULSecureRSATLSStream。可以使用 ULEccTlsStream 和 ULRsaTlsStream 来代替它们。
 - 已删除 ul_synch_info 的 security 和 security_parms 字段。相反，将 stream 字段设置为适当的字符串值：tcpip、http、https 或 tls。此外，将 security 参数与其它 stream 参数组合在一起。TCPIP 始终是基础传输机制，同时不再支持 TLS over HTTP。相反，可以使用 HTTPS 同步流。请参见“[UltraLite 同步参数和网络协议选项](#)”《[UltraLite - 数据库管理和参考](#)》。
 - 已对 ULSocketStream、ULHTTPStream 和 ULHTTPSSStream 进行了更改，以返回现在需要的适当字符串值。
 - 为 Windows CE 设备删除 ULActiveSyncStream。已使用 ActiveSync 提供程序注册的 UltraLite 应用程序，在其 Windows 消息处理程序中接收同步消息时，必须改用其中一个标准同步流。
- **嵌入式 SQL** 对 C/C++ API 的嵌入式 SQL 接口中的函数进行的更改包括：
 - 无法再动态升级数据库模式，因为 .usm 文件不复存在。与 UltraLite 的这个以前的功能相关的所有类和函数均已删除。
- **UltraLite.NET API** 对 UltraLite.NET API 中的函数进行的更改包括：
 - 无法再连接到数据库模式，也无法再动态升级数据库模式，因为 .usm 文件不复存在。与 UltraLite 的这个以前的功能相关的所有类和方法均已删除。
 - ParamsUsed 属性已在 ULConnectionParams 类中重命名为 ToString。
 - GetSQLColumnName 已重命名为 GetColumnSQLName。

- ULStreamType 的成员 UNKNOWN 和 ACTIVE_SYNC 已从此枚举中删除。缺省值现在是 ULStreamType.TCPIP。
- **UltraLite for MobileVB API** MobileVB API 中的更改方法包括:
 - 无法再连接到数据库模式，也无法再动态升级数据库模式，因为 .usm 文件不复存在。与 UltraLite 的这个以前的功能相关的所有类和方法均已删除。
- **UltraLite for M-Business Anywhere API** 对 M-Business Anywhere API 中的功能进行的更改包括:
 - 无法再连接到数据库模式，也无法再动态升级数据库模式，因为 .usm 文件不复存在。与 UltraLite 的这个以前的功能相关的所有类和方法均已删除。

名称更改

- **ULUtil** 用于 Palm OS 的 ULUtil 实用程序已重命名为 ULDBUtil。
- **ulmbreg** ulmbreg 已重命名为 ulafreg。此实用程序现在安装在 *install-dir\win32* 目录下。

杂项

- **ulcond.log** UltraLite HotSync 管道安装程序版本 10 (ulcond10) 不再向此日志文件写入消息。有关更新后的 ulcond10 实用程序的用法，请参见“[Palm OS 的 UltraLite HotSync 管道安装实用程序 \(ulcond11\)](#)”一节《UltraLite - 数据库管理和参考》。

Sybase Central 和 Interactive SQL

以下各节介绍了 10.0.0 版的 Sybase Central 和 Interactive SQL 中的新功能、行为更改以及不建议使用的功能。

新功能

以下列出了 10.0.0 版中引入的 Sybase Central 和 Interactive SQL 新增功能。

Sybase Central

本节介绍了 Sybase Central 中的新功能。对 Sybase Central 插件所做的更改和补充在以下各节中进行说明：

- [“SQL Anywhere 插件”一节第 206 页](#)
- [“用于 Sybase Central 的新插件”一节第 207 页](#)
- **Sybase Central [任务] 列表** 可选择在 Sybase Central 的左窗格（而非数据库的树形结构）中查看任务。[任务] 列表显示与当前选择的对象相关的常用任务。[任务] 列表包括常用任务、导航选项和文档链接。请参见 [“浏览 Sybase Central”一节《SQL Anywhere 服务器 - 数据库管理》](#)。
- **新连接菜单** 在先前版本中，需要按 F11 键来打开 [新建连接] 窗口，此窗口使您可以选择要与其建立连接的插件。Sybase Central 现在提供一个 [连接] 菜单，可在该菜单中选择要使用的插件。[新建连接] 窗口已删除，但可通过按下操作打开 [连接] 菜单，然后在该菜单中选择要用于您的连接的插件。
- **连接配置文件增强** 现在可以向 Sybase Central 连接配置文件中添加说明。也可将连接配置文件导入和导出。
- **插件搜索** 现在，可通过选择 [查看] » [搜索窗格]，来搜索包含 Sybase Central 中的插件和数据库内的特定文本的对象。
- **上下文下拉列表** 新的上下文下拉列表显示当前在对象树中选择的对象，从而使得浏览插件变得更加容易，尤其是对象树未在左窗格中打开时。

SQL Anywhere 插件

- **[死锁] 数据库选项卡** 在 Sybase Central 中连接到 SQL Anywhere 数据库时，可以在 [死锁] 选项卡上查看有关死锁的信息。请参见 [“从 Sybase Central 查看死锁”一节《SQL Anywhere 服务器 - SQL 的用法》](#)。
- **实体关系图** Sybase Central 显示数据库的实体关系图，以显示数据库表及其外键关系。请参见 [“从 SQL Anywhere 插件查看实体关系图”一节《SQL Anywhere 服务器 - 数据库管理》](#)。
- **新的 [创建调度向导] 和 [事件] 文件夹的其它增强** 对于每个调度事件，[事件] 文件夹现在显示该事件被触发的下一调度时间。对于每个条件事件，此文件夹显示系统事件，并有选择地显示触发该事件的触发条件。通过使用新的调度创建向导创建调度。创建新的调度事件时，[事件] 向导允许您创建单个调度，但是以后可以根据需要向事件中添加其它调度。请参见 [“定义调度”一节《SQL Anywhere 服务器 - 数据库管理》](#)。

- **维护计划** 可设置调度以自动校验和备份数据库，并可使输出日志以电子邮件的形式发送给您。请参见“[创建维护计划](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **新的 [恢复数据库向导]** 现在可以使用 [恢复数据库向导] 从档案备份恢复数据库。请参见“[从档案备份恢复](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **Sybase Central 和 Interactive SQL 的编辑器增强** 在 [选项] 窗口的 [格式] 选项卡上，可以选择在 Sybase Central 和 Interactive SQL 的编辑器窗口中使用的字体。
已为数据库对象名将键入完成选项添加到编辑器中。请参见“[使用文本完成](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

用于 Sybase Central 的新插件

- **QAnywhere 插件** QAnywhere 插件提供一个易于使用的图形界面，可用于创建和管理 QAnywhere 应用程序。
请参见“[用于 Sybase Central 的新 QAnywhere 插件](#)”一节第 187 页。
- **UltraLite 插件** UltraLite 插件允许您在图形用户界面中创建、修改和管理您的 UltraLite 数据库。
请参见“[图形管理工具](#)”一节第 196 页。
- **MobiLink [创建同步模型向导] 和 [模型] 模式** 现在可以使用向导来创建同步模型，并使用新的 [模型] 模式在 MobiLink 插件中编辑您的模型。还可以建立 MobiLink 服务器启动的同步。MobiLink 插件的旧功能也已被增强，并在 [管理] 模式中得以保留。
请参见“[Sybase Central 的 MobiLink 插件的增强](#)”一节第 168 页。

Interactive SQL

- **Interactive SQL 可以连接到 UltraLite 数据库** 现在可以使用 Interactive SQL 来开发和测试 SQL 语句以及 UltraLite 数据库。已不建议使用 ulisql 实用程序。请参见“[图形管理工具](#)”一节第 196 页。
- **Interactive SQL 与第三方源代码控制系统相集成** Interactive SQL 可与第三方源代码控制系统相集成，以允许您在 Interactive SQL 中对文件执行许多常见的源代码控制操作，如签入、签出以及与旧版本进行比较。请参见“[使用源代码控制集成](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **新的 Interactive SQL 选项** isql_maximum_displayed_rows 选项允许您指定 Interactive SQL 结果集中显示的行数，而 isql_show_multiple_result_sets 选项则指定 Interactive SQL 的 [结果] 窗格是否可显示多个结果集。请参见“[isql_maximum_displayed_rows 选项 \[Interactive SQL\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[isql_show_multiple_result_sets \[Interactive SQL\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **文本完成** Interactive SQL 现在包括一个键入完成选项，该选项可以填写以下对象类型的名称：表、视图、列、存储过程和系统函数。请参见“[使用文本完成](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **DESCRIBE 语句现在受 Interactive SQL 支持** DESCRIBE 语句使您能够获得以下有关指定表或过程的信息：

- 在表中找到的所有列
- 在表中找到的所有索引
- 与存储过程一起使用的所有参数

请参见“DESCRIBE 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **Interactive SQL 支持 @data 选项** 从命令提示符启动 Interactive SQL 时，可以指定 @data 选项以从指定环境变量或配置文件读入选项。请参见“Interactive SQL 实用程序 (dbisql)”一节《SQL Anywhere 服务器 - 数据库管理》。

SQL Anywhere 控制台实用程序

- **SQL Anywhere 控制台支持 @data 选项** 从命令提示符启动 SQL Anywhere 控制台时，可以指定 @data 选项以从指定环境变量或配置文件读入选项。请参见“SQL Anywhere 控制台实用程序 (dbconsole)”一节《SQL Anywhere 服务器 - 数据库管理》。

行为更改和不建议使用的功能

以下列出了 10.0.0 版中引入的 Sybase Central 和 Interactive SQL 更改。

- **Interactive SQL 不再将 quoted_identifier 选项设置为 On** 在此软件的先前版本中，Interactive SQL 将 quoted_identifier 选项设置为 On。它现在使用数据库的此选项的设置（缺省情况下，此选项被设置为 On）。请参见“quoted_identifier 选项 [兼容性]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **isql_plan 选项不再支持 NONE 参数** NONE 参数不再受 isql_plan 选项支持。
- **Interactive SQL 可以返回不受限制的结果集** 在软件的先前版本中，如果执行一个返回多个结果集的查询，则 Interactive SQL 最多只会显示 10 个结果集。现在，Interactive SQL 显示该查询返回的所有结果集。请参见“isql_show_multiple_result_sets [Interactive SQL]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Interactive -f 选项行为更改** 使用 -f 选项启动 Interactive SQL 时，不会自动建立与数据库之间的连接。以前，会自动打开一个连接。
- **在 Interactive SQL 中访问和保存图形式计划**
 - [打开计划] 和 [保存计划] 这两个新菜单选项可从 Interactive SQL 的 [文件] 菜单中获得，它们可用于打开和保存图形式计划（以前，这项操作是使用与打开和保存 SQL 语句相同的 [打开] 和 [保存] 菜单项来完成的）。
 - 以前，图形式计划使用 .xml 文件扩展名进行保存。现在，它们以扩展名 .saplan 进行保存。不过，先前的 .xml 文件扩展名在显示以该扩展名存储的图形式计划时仍受支持。
 - 在 [选项] 窗口中，现在可以使 Interactive SQL 成为 .sql 和 .saplan（图形式计划）文件的缺省编辑器。

- **不建议使用 SET OPTION 语句 PUBLIC 关键字** 在使用 SET OPTION 语句设置 Interactive SQL 选项时，不建议使用 PUBLIC 关键字。请参见“[Interactive SQL 选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **EXIT 语句现在只关闭当前 Interactive SQL 窗口** 在先前版本中，从 Interactive SQL 执行 EXIT 语句会关闭所有 Interactive SQL 窗口。现在，只关闭执行该语句的窗口。请参见“[EXIT 语句 \[Interactive SQL\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SQLE_ENGINE_NOT_MULTUSER 的新错误代码** 为处理 SQLE_ENGINE_NOT_MULTUSER 而被编程的应用程序，现在需要检查新的错误代码。以前，如果应用程序在数据库上尝试编写操作，而同时另一线程正在向 MobiLink 发送上载，则运行时将返回 SQLE_ENGINE_NOT_MULTUSER。现在，运行时将返回一个新的、更为准确的错误代码：SQLE_ULTRALITE_WRITE_ACCESS_DENIED。请参见“[ULSQLCode 枚举](#)”一节《[UltraLite - .NET 编程](#)》。
- **Sybase Central 插件的 [实用程序] 选项卡已删除** SQL Anywhere、MobiLink 和 UltraLite 插件中的 [实用程序] 选项卡已替换为 [工具] 按钮。还可以从 Sybase Central 的 [工具] 菜单访问实用程序。

不建议使用和停用的功能

- **不再支持使用 jConnect 连接到 Sybase Central、Interactive SQL 或 SQL Anywhere 控制台** Sybase Central、Interactive SQL 和 SQL Anywhere 控制台实用程序 (dbconsole) 不再支持通过 jConnect 连接到 SQL Anywhere 数据库。但是，仍可使用 iAnywhere JDBC 驱动程序从这些应用程序连接到数据库。作为这一更改的结果，以下功能已被删除：
 - 用于 Interactive SQL 实用程序 (dbisql) 的 -jconnect 和 -odbc 选项已被删除。
 - 用于 SQL Anywhere 控制台实用程序 (dbconsole) 的 -jconnect 和 -odbc 选项已被删除。
 - 用于连接到 Interactive SQL、SQL Anywhere 控制台以及 Sybase Central 中的 SQL Anywhere 和 MobiLink 插件的 [连接] 窗口，不再允许您指定是使用 jConnect 还是 iAnywhere JDBC 驱动程序。iAnywhere JDBC 驱动程序将用于所有连接。
- **UltraLite 计划现在显示在 Interactive SQL 中的 [计划] 选项卡上** 在先前版本中，Interactive SQL [结果] 窗格具有一个显示 UltraLite 计划优化策略的 [UltraLite 计划] 选项卡。[UltraLite 计划] 选项卡已删除，所以现在当您从 Interactive SQL 连接到 UltraLite 数据库时，计划将显示在 [计划] 选项卡上。
- **Sybase Central SQL Anywhere 插件不再支持版本 7 的数据库** 对版本 7 数据库服务器的支持以及使用版本 7 软件创建的数据库，已从 SQL Anywhere 插件中删除。仍可以连接到使用版本 5、6 或 7 软件创建而运行在版本 8 或更新版本的数据库服务器上的数据库，以便将数据库卸载并重装到重装文件或者新的（或现有的）数据库中。请参见“[为版本 11 重建版本 9 及更早版本的数据库](#)”一节第 329 页。
- **不建议使用 isql_log 选项** 不建议使用 isql_log 选项，该选项用于记录在 Interactive SQL 会话期间执行的语句。使用 START LOGGING 和 STOP LOGGING 语句来代替。请参见“[记录命令](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **Sybase Central 文件名更改** 除了所提到的 Sybase Central 的文件更改外，还增加了以下文件：

新名称	旧名称
<i>asaplugin.jar</i>	<i>saplugin.jar</i>

注册表项也已更改以反映新版本的 Sybase Central，其现在的形式如下：

`HKLM\SOFTWARE\Sybase\Sybase Central\5.0 (registry entries)`

文档增强

在几个方面对预先存在功能的文档进行了增强，其中包括：

- **SQL 语句的上下文相关帮助** 在 Interactive SQL 中，现在可以右击 SQL 语句名以打开该语句的参考主题。
- **MobiLink 入门** 已增加了新的入门手册，可帮助新用户了解如何使用 MobiLink 开发分布式应用程序。
请参见 [MobiLink - 入门](#)。
- **《数据库管理指南》中有关 SQL Anywhere 管理工具的新章节** 新增加的章节着重介绍如何使用 Sybase Central、Interactive SQL 和 SQL Anywhere 控制台实用程序。
请参见“[SQL Anywhere 图形管理工具](#)” [《SQL Anywhere 服务器 - 数据库管理》](#)。
- **为 EBF 更新的支持平台信息** 现在存在一些与产品一起安装且涉及您所安装的软件的支持平台页面。这些页面都安装在您的 SQL Anywhere 安装目录的 `documentation\zh` 子目录中。这些页面在 EBF 中进行更新。文档会在需要时链接到这些页面。
- **SQL Anywhere 安全功能已删除** *SQL Anywhere Studio 安全指南* 这个手册已删除。SQL Anywhere 安全功能现在在 [SQL Anywhere 服务器 - 数据库管理](#) 中予以介绍。
- **ODBC 驱动程序手册已删除** *用于 MobiLink 的 ODBC 驱动程序手册* 已删除。

产品范围的功能

以下各节介绍了会影响 10.0.0 版 SQL Anywhere 的所有组件的新功能、行为更改和不建议使用的功能。

新功能

以下列出了 10.0.0 版中引入的产品范围新增功能。

- **将产品名更改为 SQL Anywhere 10** SQL Anywhere Studio 已重命名为 SQL Anywhere 10，而 Adaptive Server Anywhere 已重命名为 SQL Anywhere。许多文件名、目录名、服务名和可执行文件名都因此发生了变化，主要反映在从 ASA 到 SA 的变化。这些更改在本章的相关 [行为更改] 主题中进行了详细介绍。
- **用于 DataWindow .NET 的新安装** DataWindow .NET 是一个自定义控件工具，对于使用 Visual Studio 的数据库开发人员非常有用。它在 SQL Anywhere 安装期间作为可选组件予以提供。完整文档将在安装中提供。
- **现在随 SQL Anywhere 提供了 RSA** 您不必再去购买单独的许可就可使用 RSA 加密。请参见“[传送层安全](#)”《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **功能统计信息集合** SQL Anywhere 10 将跟踪有关运行此软件的计算机的信息（例如操作系统、用于启动数据库服务器的选项、正在使用的 SQL Anywhere 10 软件的内部版本等），以及有关正在使用的 SQL Anywhere 10 功能的信息。如果发生致命错误，软件将自动提示您向 iAnywhere 发送任何有关该问题的信息，以及 SQL Anywhere 功能统计信息。如果您打开技术支持案例，则可使用此信息帮助诊断问题。请参见“[SQL Anywhere 中的错误报告](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

您也可随时选择使用支持实用程序 (dbsupport) 来发送功能统计信息。此功能统计信息可帮助 iAnywhere 了解产品的使用情况以改进软件。请参见“[支持实用程序 \(dbsupport\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

SADIAGDIR 环境变量将指定存储崩溃报告和功能统计信息的目录位置。请参见“[SADIAGDIR 环境变量](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **错误报告** 当在 Sybase Central、Interactive SQL 或 SQL Anywhere 控制台实用程序中发生内部错误时，错误日志将被写入硬盘驱动器并出现一个窗口，您可在其中选择向 iAnywhere 发送错误报告。

此外，如果在以下各项中发生致命错误：个人服务器、网络服务器、MobiLink 服务器、dbmsync、MobiLink 监听器、QAnywhere 代理、SQL Remote 或复制代理，则错误日志将被写入硬盘驱动器并出现一个窗口，您可在其中选择向 iAnywhere 发送错误报告。可将支持实用程序 (dbsupport) 配置为自动提交这些错误报告。请参见“[支持实用程序 \(dbsupport\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

如果选择不提交错误报告，则该文件将保留在硬盘上的诊断目录中。SADIAGDIR 环境变量将指定存储崩溃报告和功能统计信息的目录位置。请参见“[SADIAGDIR 环境变量](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **控制管理工具中的可用功能** 现在，可以通过使用与管理工具的 .jar 文件位于同一目录的 OEM.ini 文件，来控制管理工具中可供用户使用的功能。

- **在 Windows Vista 上使用 SQL Anywhere 10** 在 Windows Vista 上存在以下已知问题：
 - 如果对包含服务器可执行文件的目录不具有管理员权限或写入权限，则使用服务器授权实用程序 (*dblic.exe*) 更新许可信息将会失败。
 - SQL Anywhere 作为服务或无权限帐户运行时，存在一个使用共享内存的已知问题。此问题正在研究中。或者，使用 TCP/IP 来代替。

行为更改

以下是 10.0.0 版中产品范围的更改列表。

- **示例目录的位置已更改** SQL Anywhere 10 随附的示例不再安装于 SQL Anywhere 10 安装目录之下。作为此更改的结果，示例数据库现在安装在以下位置：

示例数据库	位置
SQL Anywhere 示例数据库	<i>samples-dir\demo.db</i>
MobiLink CustDB 示例统一数据库应用程序	<i>samples-dir\MobiLink\CustDB\</i>
UltraLite CustDB 示例数据库	<i>samples-dir\UltraLite\CustDB\</i>

有关每个支持的操作系统的 *samples-dir* 缺省位置列表，请参见“示例目录”一节《SQL Anywhere 服务器 - 数据库管理》。

- **不支持的平台** 不再支持以下平台：

- Windows 95
- Windows 98
- Windows Me
- Windows NT
- Compaq Tru64

有关平台支持的其它更改，请参见“支持的平台”一节《SQL Anywhere 11 - 简介》。

- **示例数据库已修订并已重命名** SQL Anywhere 示例数据库现在的名称是 *demo.db*。诸如表、列、视图和索引之类的对象现在具有全字名称，这对使用屏幕读取器的用户非常有益。请参见“SQL Anywhere 示例数据库”《SQL Anywhere 11 - 简介》。
- **其它文件名更改** 除了单个产品涉及的文件名更改外，以下文件名也进行了更改：

旧名称	新名称
<i>asa.cvf</i>	<i>sqlany.cvf</i>
<i>asaldap.ini</i>	<i>saldap.ini</i>
<i>asasrv.ini</i>	<i>sasrv.ini</i>

旧名称	新名称
<i>asa_config.sh</i>	<i>sa_config.sh</i>
<i>install-dir/SYBSasa9/lib</i>	<i>sqlanywhere10/lib32</i> 或 <i>sqlanywhere10/lib64</i>
<i>install-dir/SYBSasa9/bin</i>	<i>sqlanywhere10/bin32</i> 或 <i>sqlanywhere10/bin64</i>

为了反映新的产品名，某些注册表项已进行了更改。它们现在是：

```
HKLM\SYSTEM\CurrentControlSet\Services\Eventlog\Application\SQLANY
HKLM\SYSTEM\CurrentControlSet\Services\Eventlog\Application\SQLANY 10.0
HKLM\SYSTEM\CurrentControlSet\Services\Eventlog\Application\SQLANY 10.0
Admin
HKLM\SOFTWARE\Sybase\Sybase Central\5.0 (registry entries)
HKLM\SOFTWARE\Sybase\SQL Anywhere\10.0 (registry entries)
```

除了已提到的单个产品的文件更改外，还增加了以下文件：*ulbase.lib*。

以下服务组名已更改：

说明	旧名称	新名称
网络服务器	ASANYServer	SQLANYServer
个人服务器	ASANYEngine	SQLANYEngine
MobiLink 同步客户端	ASANYMLSync	SQLANYMLSync
复制代理	ASANYLTM	SQLANYLTM

对 MobiLink、QAnywhere 和远程数据访问所使用的 ODBC 驱动程序进行的更改

- **Sybase Adaptive Server Enterprise 驱动程序** SQL Anywhere 不再包括用于 Adaptive Server Enterprise 的 iAnywhere Solutions ODBC 驱动程序。相反，Adaptive Server Enterprise 本地驱动程序经过测试与 MobiLink 协同工作。不再支持 iAnywhere Solutions 9 - Adaptive Server Enterprise Wire Protocol 驱动程序。

请参见 http://www.iAnywhere.com/developer/technotes/odbc_mobilink.html。

- **IBM UDB DB2 驱动程序** SQL Anywhere 不再包括用于 DB2 的 iAnywhere Solutions ODBC 驱动程序。相反，IBM DB2 8.2 CLI 驱动程序经过测试与 MobiLink 协同工作。此本地 DB2 驱动程序支持 DB2 版本 8.1 和 8.2。不再支持以下驱动程序：IBM DB2 7.2 ODBC 驱动程序和 iAnywhere Solutions 9 - DB2 Wire Protocol 驱动程序。

请参见 http://www.iAnywhere.com/developer/technotes/odbc_mobilink.html。

- **Oracle 驱动程序** iAnywhere Solutions 10 - Oracle Wire Protocol 驱动程序，可通过单独下载获得。

请参见 http://www.iAnywhere.com/developer/technotes/odbc_mobilink.html。

9.0.2 版中的新功能

目录

9.0.2 版中的新功能	216
9.0.2 版中的行为更改	230

9.0.2 版中的新功能

本节列出在 SQL Anywhere Studio 9.0.2 版组件中引入的新功能。

Adaptive Server Anywhere 的新功能

本节介绍 Adaptive Server Anywhere 9.0.2 版中的新功能。它提供主要和次要新功能的详尽列表，以及指向详细叙述每个功能的各章节位置的交叉引用。

SQL 增强

- **UNIQUEIDENTIFIER 本机数据类型** UNIQUEIDENTIFIER 数据类型现在是本机数据类型，而不是在 BINARY(16) 上定义的域。因此，Adaptive Server Anywhere 会根据需要自动执行类型转换，因而处理 UNIQUEIDENTIFIER 值时不需要 STRTOUUID 和 UUIDTOSTR 转换函数。

要在使用以前版本创建的数据库上使用 UNIQUEIDENTIFIER 数据类型，必须通过卸载和重装该数据库来升级数据库文件格式。

请参见“UNIQUEIDENTIFIER 数据类型”一节《SQL Anywhere 服务器 - SQL 参考》。

- **RESOLVE UPDATE 触发器的冲突函数** CONFLICT 函数可用于冲突解决触发器，以确定某一特定列是否是在 SQL Remote 统一数据库上执行的 UPDATE 的冲突源。

请参见“CONFLICT 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **过程分析增强** 现在，通过使用 sa_server_option 存储过程，可以针对每个用户和每个连接过滤分析信息。

请参见“使用 sa_server_option 启用分析”一节《SQL Anywhere 服务器 - SQL 的用法》和“sa_server_option 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

- **创建或修改远程服务器之前，可以先进行测试** Sybase Central 中的 [创建远程服务器向导] 中有一个 [测试连接] 按钮，在创建远程服务器之前，使用此按钮可以测试远程服务器定义中提供的连接信息是否能够连接成功。

Sybase Central 中的 [远程服务器] 属性窗口中也有一个 [测试连接] 按钮，如果更改了远程服务器的属性，使用此按钮可以测试是否能够成功连接到远程服务器。

请参见“使用 Sybase Central 创建远程服务器”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **INPUT 和 OUTPUT 语句接受 ESCAPES 子句** 使用 ESCAPES 子句使您可以指定让数据库服务器将字符识别并解释为特殊字符。

请参见“INPUT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》和“OUTPUT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **WAITFOR 接收到来自另一连接的消息时会被唤醒** WAITFOR 语句现在可在使用 MESSAGE 语句接收到来自另一连接的消息时被唤醒。

请参见“WAITFOR 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **派生表显示在执行计划中** 现在，派生表在查询执行计划中显示为节点。

- **ALTER DOMAIN 语句** 使用 ALTER DOMAIN 语句可以重命名用户定义的域和数据类型。

请参见“ALTER DOMAIN 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **过程的 NO RESULT SET 子句** 当外部环境需要知道存储过程不返回结果集时，就可以声明存储过程的 NO RESULT SET 子句。

请参见“CREATE PROCEDURE 语句 (Web 服务)”一节《SQL Anywhere 服务器 - SQL 参考》。

- **在索引创建期间更新列统计信息** CREATE INDEX 语句现在有一个副作用，即会更新索引列的列统计信息。

请参见“CREATE INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

编程接口增强

- **PHP 模块** SQL Anywhere PHP 模块允许用 PHP 脚本语言访问 Adaptive Server Anywhere 数据库。

请参见“SQL Anywhere PHP API”《SQL Anywhere 服务器 - 编程》。

- **Web 服务客户端** 除充当 Web 服务提供程序外，Adaptive Server Anywhere 现在还能充当 Web 服务客户端，使创建访问 Adaptive Server Anywhere Web 服务及 Internet 上提供的标准 Web 服务的存储过程和存储函数成为可能。

请参见“SQL Anywhere Web 服务”《SQL Anywhere 服务器 - 编程》。

- **支持多种 Web 服务格式** 现在，可以选择 DISH 服务提供的 WSDL 文件的格式及部分 SOAP 响应返回的数据载荷的格式，以最好地满足客户端应用程序的需要。现在，您可以在 Microsoft .NET 的 DNET、用于自动生成接口的客户端的 CONCRETE 以及通用 XML 格式中进行选择。

请参见“创建 SOAP 和 DISH Web 服务”一节《SQL Anywhere 服务器 - 编程》。

- **odbc_describe_binary_as_varbinary 选项** 此选项允许您选择是将所有 BINARY 和 VARBINARY 列作为 BINARY 还是作为 VARBINARY 提供给应用程序。

请参见“odbc_describe_binary_as_varbinary [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新的 prefetch 选项值** prefetch 选项现在增加了一个值 Always。此值意味着即使是 SENSITIVE 游标类型和涉及代理表的游标结果也要进行预读。

请参见“prefetch 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **db_locate_servers_ex 函数** 此函数提供对 dblocate -n 选项显示的信息的程式访问（列出特定主机上的所有 Adaptive Server Anywhere 数据库服务器）。

请参见“db_locate_servers_ex 函数”一节《SQL Anywhere 服务器 - 编程》。

管理增强

- **SNMP 代理** 现在可以通过简单网络管理协议 (SNMP) 应用程序监控 Adaptive Server Anywhere。

请参见“SQL Anywhere SNMP Extension Agent”《SQL Anywhere 服务器 - 数据库管理》。

- **死锁报告** 现在，您可以使用新的数据库选项 `log_deadlocks` 和新的系统存储过程 `sa_report_deadlocks` 获取涉及死锁的连接的有关信息。打开 `log_deadlocks` 选项时，数据库服务器会在内部缓冲区中记录有关死锁的信息。您可以通过调用 `sa_report_deadlocks` 从此内部缓冲区获取死锁信息。

请参见“确定被阻塞的连接”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **新归类** 在本版本中添加了下列归类：
 - **1252SWEFIN** 已添加，用于支持瑞典语和芬兰语。在瑞典语和芬兰语系统中，如果未指定任何归类，则数据库服务器将选择 1252SWEFIN 作为新数据库的缺省归类。
 - **1255HEB** 已添加，用于支持希伯来语。在希伯来语 Windows 系统中，如果未指定任何归类，则数据库服务器将选择 1255HEB 作为新数据库的缺省归类。
 - **1256ARA** 已添加，用于支持阿拉伯语。在阿拉伯语 Windows 系统中，如果未指定任何归类，则数据库服务器将选择 1256ARA 作为新数据库的缺省归类。
 - **950ZHO_HK 和 950ZHO_TW** 已添加，用于支持中文。950ZHO_HK 提供对 Windows 繁体中文字符集 cp950 及中国香港特别行政区补充字符集 (HKSCS) 的支持。950ZHO_TW 归类提供对 Windows 繁体中文字符集 cp950 的支持，但不支持 HKSCS。排序基于繁体中文字符的字符顺序。这些归类取代了不建议使用的 950TWN 归类。
 - **1252SPA** 已添加，用于支持西班牙语。在西班牙语 Windows 系统中，如果未指定任何归类，则数据库服务器将选择 1252SPA 作为新数据库的缺省归类。
 - **874THAIBIN** 已添加，用于支持泰语。此为 Windows 和 UNIX 系统中建议使用的泰语归类。

请参见“支持的归类和替代归类”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新的服务实用程序 (dbsvc) 选项** 服务实用程序 (dbsvc) 支持以下新选项：
 - **-cm 选项** 此选项显示用于创建指定服务的命令。部署服务或将服务恢复到原始状态时可以使用此选项。
 - **-sd 选项** 此选项允许您提供服务的说明，该说明显示在 Windows 服务管理器中。
 - **-sn 选项** 此选项允许您提供服务的名称，该名称显示在 Windows 服务管理器中。

请参见“用于 Windows 的服务实用程序 (dbsvc)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新的数据源 (dbdsn) 实用程序选项** 数据源实用程序 (dbdsn) 支持以下新选项：
 - **-cm 选项** 此选项显示用于创建指定数据源的命令。部署数据源或将数据源恢复到原始状态时可以使用此选项。
 - **Driver 连接参数** 在 Windows 上使用数据源实用程序 (dbdsn) 创建数据源时，可以使用 Driver 连接参数指定 ODBC 数据源的驱动程序。在 UNIX 上，如果不指定 Driver 连接参数，则数据源实用程序会根据 ASANY9 环境变量的设置，自动添加一个 Driver 条目，其中包含 Adaptive Server Anywhere ODBC 驱动程序的完整路径。

请参见“数据源实用程序 (dbdsn)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **磁盘已满回调支持** `-fc` 数据库服务器选项允许您指定一个包含回调函数的 DLL，该回调函数在文件系统已满的情况下，可通知用户，还可能会采取纠正操作。
请参见“[-fc 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **[校验数据库] 向导增强** 在 Sybase Central 中使用 [校验数据库向导] 校验数据库时，该向导指示当前正在校验的表以及校验操作的总体进度。此外，对于启用了校验和的数据库，可以同时对比和校验和进行校验。
请参见“[校验数据库](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **在 Sybase Central 中卸载表数据** 现在，在 Sybase Central 中，使用 [卸载数据] 窗口一步就可以从一个或多个表卸载数据。
请参见“[使用 \[卸载数据\] 窗口导出数据](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **添加到 `sa_index_density` 和 `sa_index_levels` 的新列** 向 `sa_index_density` 和 `sa_index_levels` 存储过程返回的结果集添加了三个新列：`TableId`、`IndexId` 和 `IndexType`。如果要恢复到这些存储过程的旧行为，可以先删除存储过程，再使用以前版本软件的结果集中包含的列来重新创建存储过程。
请参见“[sa_index_density 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[sa_index_levels 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **BACKUP 和 RESTORE DATABASE 语句的 HISTORY 选项** 使用 HISTORY 选项可以控制是否在 `backup.syb` 文件中记录 BACKUP 和 RESTORE DATABASE 操作。
请参见“[BACKUP 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[RESUME 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **支持使用 Windows 用户组的集成登录** 除为 Windows NT/2000/XP 上的单个用户创建集成登录外，现在还可以创建到 Windows NT/2000/XP 上的用户组的集成登录映射。建议您在使用此功能前先升级数据库。
请参见“[为 Windows 用户组创建集成登录](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **管理请求日志的大小** 使用 `-zn` 数据库服务器选项可以指定应保留多少个请求日志文件。
请参见“[-zn 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **当备份重命名了事务日志时，位于事务日志末尾的空闲页将被删除** 为获得更好的性能，事务日志文件以固定大小的增量增长。作为备份过程的一部分，当事务日志被重命名时，会删除日志末尾的空闲页，这有助于释放磁盘空间。
- **远程服务器连接现在可以显式关闭** 在以前的版本中，只有当用户断开与 Adaptive Server Anywhere 的连接时，从 Adaptive Server Anywhere 到远程服务器的连接才会断开。现在，可以使用 ALTER SERVER 语句的新的 CONNECTION CLOSE 子句，将 Adaptive Server Anywhere 从远程服务器显式断开。
请参见“[ALTER SERVER 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

安全性增强

- **可使用 `dbfhide` 对初始化文件进行模糊处理** 文件隐藏实用程序 (`dbfhide`) 现在可用于对 Adaptive Server Anywhere 及其实用程序所使用的 `.ini` 文件的内容进行模糊处理。

请参见“文件隐藏实用程序 (dbfhide)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **FIPS 认证的安全性** 在所有支持的 Windows 平台（Windows CE 除外）上，现在可以使用 Certicom 的 FIPS 140-2 认证软件进行安全通信。

请参见“启动支持传送层安全的数据库服务器”一节《SQL Anywhere 服务器 - 数据库管理》。

使用 Certicom 的 FIPS140-2 认证软件的高度数据库加密在支持的 32 位 Windows 平台上也可用。

请参见“加密和解密数据库”一节《SQL Anywhere 服务器 - 数据库管理》。

其它增强

- **新连接属性** 添加了以下连接属性：

- ClientPort
- LoginTime
- ServerPort

请参见“连接属性”一节《SQL Anywhere 服务器 - 数据库管理》。

- **正确设置事件查看器消息的格式** 当部署 Adaptive Server Anywhere 数据库时，应设置一个注册表条目，用于控制事件查看器的消息的格式。

请参见“部署数据库服务器”一节《SQL Anywhere 服务器 - 编程》。

- **log_deadlocks 选项** 此选项允许您控制数据库服务器是否在内部缓冲区中记录有关死锁的信息。此选项可搭配 sa_report_deadlocks 过程使用，以获取有关死锁的信息。

请参见“log_deadlocks 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **rollback_on_deadlock 选项** 此选项允许您控制事务遇到死锁时是否自动回退。

请参见“rollback_on_deadlock [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **temp_space_limit_check 选项** 此选项允许您控制当连接请求的临时文件空间超过其限额时的操作。

请参见“temp_space_limit_check 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新系统存储过程** 添加了多个新的系统存储过程：

- sa_rowgenerator 过程** sa_rowgenerator 系统过程是作为 RowGenerator 表的备选过程提供的，用于返回指定起始值和结束值之间的行的结果集。

您可以使用此过程完成如下任务：生成一个结果集，其中的行数为某一范围中的值的总数，或为结果集中已知数目的行生成测试数据，等等。

请参见“sa_rowgenerator 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

- sa_send_udp 存储过程** 此过程向指定地址发送一个 UDP 包，并可用在 MobiLink 服务器启动的同步中，以唤醒监听器实用程序 (dblsn.exe)。

请参见“sa_send_udp 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

- sa_verify_password 存储过程** 此过程由 sp_password 存储过程用来验证当前用户的口令。

请参见“sa_verify_password 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

请参见“[sa_verify_password 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **Windows CE 上的最大高速缓存大小** 在 SQL Anywhere Studio 的以前版本中，Windows CE 上的最大高速缓存大小为 32 MB。此限制已取消，高速缓存大小现在由设备上的可用内存量限制。
- **UNIX 的新数据库服务器选项** 已为 UNIX 添加了以下数据库服务器选项：
 - **-uc** 在 UNIX 上以 shell 模式启动数据库服务器。
请参见“[-uc 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
 - **-ui** 在支持 X 窗口的 Linux 和 Solaris 上启动数据库服务器时，尝试显示 [服务器启动选项] 窗口和数据库服务器消息窗口。如果服务器找不到可用的显示，则服务器以 shell 模式启动。
请参见“[-ui 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
 - **-ux** 在支持 X 窗口的 Linux 和 Solaris 上启动数据库服务器时，显示 [服务器启动选项] 窗口和数据库服务器消息窗口。
请参见“[-ux 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

MobiLink 的新功能

以下列出了 9.0.2 版中引入的对软件的更改和补充。

- **新重定向器**
 - 提供了新的适用于 Apache 的本机重定向器，可在 Windows、Solaris 和 Linux 上使用。
 - 现在提供了 M-Business Anywhere 重定向器，可在 Windows、Solaris 和 Linux 上使用。
请参见“[M-Business Anywhere 重定向器（不建议使用）](#)”一节《[MobiLink - 服务器管理](#)》。
 - NSAPI 重定向器现在可在 Solaris 上使用。它以前只能在 Windows 上使用。
请参见“[Windows 上 Netscape/Sun Web 服务器的 NSAPI 重定向器（不建议使用）](#)”一节《[MobiLink - 服务器管理](#)》。
- **现在可以配置协议以忽略指定主机** 新选项 ignore 用来指定 MobiLink 服务器在连接时应忽略的主机。
请参见“[-x 选项](#)”一节《[MobiLink - 服务器管理](#)》中的 **ignore**。
- **阻止客户端在 MobiLink 服务器正忙时等待同步** 现在可以阻止客户端在 MobiLink 服务器正忙时等待同步。
请参见“[-x 选项](#)”一节《[MobiLink - 服务器管理](#)》中的 **backlog**。
- **版本存储在统一数据库中** SQL Anywhere Studio 的版本号和内部版本号现在存储在 MobiLink 系统表 ml_property 中。对于这些条目，component_name 为 **ML**，property_set_name 为 **server_info**，property_name 为 **release_version**，property_value 的形式为 *version.build*（如 9.0.2.1234）。
有关 MobiLink 系统表的详细信息，请参见“[ml_property](#)”一节《[MobiLink - 服务器管理](#)》。

- **MobiLink 服务器支持新的 uniqueidentifier 数据类型** UNIQUEIDENTIFIER 数据类型现在是本机数据类型，而不是在 BINARY(16) 上定义的域。因此，现在 MobiLink 远程数据库会在需要时自动执行类型转换，从而处理 UNIQUEIDENTIFIER 值时不再需要 String 到 UUID 和 UUID 到 String 的转换函数。

有关此数据类型到支持的统一数据库的映射的信息，请参见“[远程数据库和统一数据库之间的 MobiLink 数据映射](#)”《[MobiLink - 服务器管理](#)》。

安全性增强

- **FIPS 认证的安全流** 在 Windows 设备上，现在可以通过 Certicom 的 FIPS 140-2 认证软件进行安全通信。
请参见“[启动支持传送层安全的 MobiLink 服务器](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **连接选项现在显示在输出日志中** MobiLink 现在在输出日志中显示连接字符串和选项，口令用星号代替。
- **不建议使用的安全功能** 请参见“[MobiLink 的行为更改](#)”一节第 231 页。

MobiLink 客户端增强

- **UltraLite 的新同步设置工具** UltraLite 模式管理器现在可生成 MobiLink 同步脚本，以及 Adaptive Server Anywhere 统一数据库的数据库表和触发器。
- **现在，删除和重新创建远程数据库更为容易** Adaptive Server Anywhere 客户端预订的第一次同步现在始终有效。
请参见“[进度偏移](#)”一节《[MobiLink - 客户端管理](#)》。
- **到 MobiLink 的连接失败时调用新的 dbmlsync 挂接** 添加了新的事件挂接 sp_hook_dbmlsync_connect_failed，可以使用该事件挂接编写从失败的同步连接恢复的方法。
请参见“[sp_hook_dbmlsync_ml_connect_failed](#)”一节《[MobiLink - 客户端管理](#)》。
- **改进了 MobiLink 客户端与 HTTP 基础结构的集成** 现在可以在代理服务器和/或 Web 服务器要求下进行 RFC 2617 基本或摘要验证时使用 HTTP 进行同步。

请参见：

- “[http_password](#)”一节《[MobiLink - 客户端管理](#)》
- “[http_userid](#)”一节《[MobiLink - 客户端管理](#)》
- “[http_proxy_password](#)”一节《[MobiLink - 客户端管理](#)》
- “[http_proxy_userid](#)”一节《[MobiLink - 客户端管理](#)》

此外，还有两个新的客户端连接参数，允许您指定自定义 HTTP 标头和自定义 Cookie。为了区分会话 Cookie，HTTP 客户端现在可以识别在服务器回复中接收的所有 Set-Cookie 和 Set-Cookie2 HTTP 标头，并在以后的所有 HTTP 请求中发送这些 Cookie 的备份。如果 Cookie 与某个现有的 Cookie 名称相同，则客户端会用新值替换旧值。同步之间不会记住 Cookie：同步结束时 Cookie 即被清除。

请参见“[custom_header](#)”一节《[MobiLink - 客户端管理](#)》和“[set_cookie](#)”一节《[MobiLink - 客户端管理](#)》。

- **检测连接错误帮助** MobiLink 客户端现在会在指定了无效连接参数时发出警告消息。
- **镜像日志位置** 当 dbmlsync 不是在远程数据库所在的计算机上运行时，或者当镜像日志与镜像事务日志不是位于同一目录时，如果使用此新扩展选项指定了旧镜像日志的位置，dbmlsync 现在就可自动删除旧的日志文件。

请参见“[MirrorLogDirectory \(mld\) 扩展选项](#)”一节《[MobiLink - 客户端管理](#)》。

服务器启动的同步增强

- **连接启动的同步的增强功能** 除 `_BEST_IP_CHANGED_` 外，Windows 监听器现在还生成内部消息 `_IP_CHANGED_`，以帮助您在连接更改时启动同步。

请参见“[连接启动的同步](#)”一节《[MobiLink - 服务器启动的同步](#)》。

- **监听器发布操作增强** 指定监听器发布操作时，您现在可以选择使用 Windows 消息 ID 来指定窗口消息，也可以选择使用窗口标题来代替窗口类。如果消息或标题包含非字母数字字符（如空格或标点符号），则您也可以在窗口类名或消息两边使用单引号。

请参见“[用于 Windows 设备的监听器实用程序](#)”一节《[MobiLink - 服务器启动的同步](#)》中的 `post`。

- **新的操作变量** 有几个新的操作变量：

- `$request_id`
- `$best_ip`
- `$best_adapter_name`
- `$best_adapter_mac`
- `$best_network_name`

请参见“[用于 Windows 的监听器操作命令](#)”一节《[MobiLink - 服务器启动的同步](#)》。

- **更多设备支持** Palm 监听器现在支持 Kyocera 7135 Smartphone 和 Treo 600 Smartphone。

请参见“[用于 Palm 设备的实用程序](#)”一节《[MobiLink - 服务器启动的同步](#)》。

SQL Remote 的新功能

以下列出了 9.0.2 版中引入的对软件的更改和补充。

- **镜像日志位置** 当 dbremote 不是在远程数据库所在的计算机上运行时，或者当镜像日志与镜像事务日志不是位于同一目录时，如果使用新的 `-ml` 选项指定了旧镜像日志的位置，dbremote 就可自动删除旧的日志文件。

请参见“[消息代理 \(dbremote\)](#)”一节《[SQL Remote](#)》。

- **RESOLVE UPDATE 触发器的冲突函数** `CONFLICT` 函数可用于冲突解决触发器，以确定某一特定列是否是在 SQL Remote 统一数据库上执行的 `UPDATE` 的冲突源。

请参见“[CONFLICT 函数 \[Miscellaneous\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

UltraLite 的新功能

以下列出了 9.0.2 版中引入的对软件的更改和补充。

组件新功能

- **UltraLite.NET 中的 ADO.NET 接口** UltraLite.NET 现在在新的命名空间 `iAnywhere.Data.UltraLite` 中支持 ADO.NET 编程接口。ADO.NET 向 UltraLite 提供符合行业标准的接口，还为大型应用程序提供了迁移到 Adaptive Server Anywhere 的简单途径。

推荐使用 ADO.NET 接口，而不是以前的 UltraLite.NET 接口（`iAnywhere.UltraLite` 命名空间），后者现在已不建议使用。

请参见“[教程：构建 UltraLite.NET 应用程序](#)”《[UltraLite - .NET 编程](#)》。
- **UltraLite for MobileVB 增强** UltraLite for MobileVB 现在支持使用 AppForge Crossfire 进行 Visual Basic .NET 编程。
- **UltraLite for M-Business Anywhere 增强** 对 UltraLite for M-Business Anywhere 进行了以下增强：
 - UltraLite for M-Business Anywhere 现在支持客户端/服务器 UltraLite 引擎。您的应用程序可以使用 `DatabaseManager.runtimeType` 属性检查正在使用的是引擎还是运行时库。
 - UltraLite for M-Business Anywhere 应用程序现在通过一个操作就可以同时同步数据和 Web 内容。
请参见“[一键同步](#)”一节《[UltraLite - M-Business Anywhere 编程](#)》。
 - 可以使用 `MobiLink` 重定向器通过单个 M-Business Anywhere 服务器同步数据和 Web 内容。对于从防火墙外部进行的同步，这样可以减少所需的可访问端口数。
请参见“[通过 M-Business Anywhere 同步数据](#)”一节《[UltraLite - M-Business Anywhere 编程](#)》和“[M-Business Anywhere 重定向器（不建议使用）](#)”一节《[MobiLink - 服务器管理](#)》。
 - 现在，在 Windows XP 上支持 M-Business Anywhere 5.5 平台。连接参数 `databaseOnDesktop` 和 `schemaOnDesktop` 支持此环境。
 - API 中还添加了其它方法，这些方法可使用列 ID（而不是列名）来收集有关数据的信息。
请参见“[ResultSetSchema 类](#)”一节《[UltraLite - M-Business Anywhere 编程](#)》和“[TableSchema 类](#)”一节《[UltraLite - M-Business Anywhere 编程](#)》。
请参见 [UltraLite - M-Business Anywhere 编程](#)。
- **Native UltraLite for Java 增强** 已对 Native UltraLite for Java 进行了以下增强：
 - 列模式信息还可通过列 ID 访问而非只能通过名称访问。
 - 新的 `SyncProgressData ErrorMessage` 属性和改进的同步错误报告。
 - 添加了 `PreparedStatement.[get]Plan`。
 - 使用时，`ResultSet` 和 `ResultSetSchema` 使 `PreparedStatement` 保持为活动状态。
- **UltraLite.NET 组件增强** UltraLite.NET 支持以下函数。建议将这些函数用作 ADO.NET 接口（`iAnywhere.Data.UltraLite` 命名空间）的一部分。

- 新的 ULCursorSchema.Name 和 ULResultSetSchema.Name 只读属性。
- 新的 ULSyncProgressData ErrorMessage 属性和改进的同步错误报告。
- ULCommand.Plan 只读属性。

请参见“UltraLite .NET 2.0 API 参考”《UltraLite - .NET 编程》。

- **Palm 开发人员现在可以使用独立于版本的前缀文件** 在以前的版本中，UltraLite 前缀文件取决于您开发程序时所使用的 Palm OS 版本。现在，*ulpalms.h* 可用于任何版本的 Palm OS。
请参见“使用用于 CodeWarrior 的 UltraLite 插件”一节《UltraLite - C 及 C++ 编程》。
- **Palm 开发人员现在可以使用扩展模式** CodeWarrior 支持一种称为 **expanded mode** 的代码生成模式，该模式改进了全局数据的内存使用。现在您可以使用 UltraLite 运行时库的扩展模式版本。
请参见“构建扩展模式应用程序”一节《UltraLite - C 及 C++ 编程》。
- **可从永久存储中检索受信任证书** 在此软件的以前版本中，用于安全同步的受信任证书是嵌入在数据库模式中的。在 Windows 和 Windows CE 平台上，它现在可以存储在外部，并可通过 *trusted_certificates* 选项进行访问。
请参见“*trusted_certificates*”一节《MobiLink - 客户端管理》。

SQL 和运行时增强

- **动态 SQL 增强** 已对 UltraLite 动态 SQL 支持进行了以下增强：
 - **查询优化改进** 在此软件的以前版本中，表的访问顺序为这些表在查询中显示的顺序。在此版本中，UltraLite 优化查询以找到表的有效访问顺序。只要您在数据库中定义了适当的索引，优化程序就能帮助改进查询的执行性能。
 - **执行计划查看** 为清楚起见，现在执行计划包括的是索引名称而不是索引号。例如，可以使用新的 UltraLite Interactive SQL 实用程序查看访问计划。
 - **IF 和 CASE 表达式** 通过添加 IF 和 CASE 条件表达式，扩展了 UltraLite 支持的表达式范围。
请参见“IF 表达式”一节《UltraLite - 数据库管理和参考》和“CASE 表达式”一节《UltraLite - 数据库管理和参考》。
 - **表名称可以带有所有者名称** UltraLite 表没有所有者。为了方便使用现有的 SQL 和以编程方式生成的 SQL，已增加对 *owner.table-name* 的支持。UltraLite 接受 *owner*，但将其忽略。
- **引入 UNIQUEIDENTIFIER 数据类型** UNIQUEIDENTIFIER 数据类型现在是本机数据类型，而不是在 BINARY(16) 上定义的域。因此，需要时 UltraLite 会自动执行类型转换，这样就不需要 String 到 UUID 和 UUID 到 String 的转换函数来处理 UNIQUEIDENTIFIER 值。
请参见“UNIQUEIDENTIFIER 数据类型”一节《SQL Anywhere 服务器 - SQL 参考》。
- **增强了 UltraLite 执行计划描述** 增强了 UltraLite 执行计划描述（可在 UltraLite Interactive SQL 中查看），使之更易于读取，从而能更好地诊断性能问题。

管理增强

- **UltraLite Interactive SQL 实用程序** 现在提供 UltraLite Interactive SQL 实用程序，用于针对 UltraLite 数据库测试 SQL 语句和用于修改 UltraLite 数据。它还显示执行计划以便您可以诊断性能问题。
- **数据库管理命令行实用程序** 一组命令行实用程序简化了 Windows 计算机上 UltraLite 文件的数据库管理任务。这些实用程序在应用程序开发过程中尤其有用。

每一个新的实用程序执行 ulconv 实用程序提供的任务的一个子集。在本软件的以后版本中，ulconv 实用程序将由这些较新的单任务实用程序取代。

请参见：

- “UltraLite 创建数据库实用程序 (ulcreate)” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 装载 XML 到数据库实用程序 (ulload)” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 同步实用程序 (ulsync)” 一节 《UltraLite - 数据库管理和参考》
- “UltraLite 卸载旧数据库实用程序 (ulunload)” 一节 《UltraLite - 数据库管理和参考》

同步增强

- **改进了 MobiLink 客户端与 HTTP 基础结构的集成** 两个新的客户端连接参数允许您指定自定义标头和自定义 Cookie。

请参见 “custom_header” 一节 《MobiLink - 客户端管理》和 “set_cookie” 一节 《MobiLink - 客户端管理》。

- **从模式管理器生成同步脚本** UltraLite 模式管理器现在提供生成 Adaptive Server Anywhere 统一数据库同步脚本的功能。这一功能使得将 UltraLite 应用程序扩展到同步体系结构更为简单。
- **参照完整性违规的同步通知** 支持同步回调函数报告参照完整性违规 - 当前，不符合参照完整性的行在无提示的情况下被删除。

QAnywhere 的新功能

- **故障转移服务器** QAnywhere 代理现在可采用一个 MobiLink 服务器连接协议选项的列表，而不是仅采用一个选项。
请参见 “设置故障转移机制” 一节 《QAnywhere》。
- **模拟器支持** Pocket PC 2002 和 Pocket PC 2003 上的 QAnywhere 客户端应用程序现在支持 x86 模拟器。这些模拟器只支持 QAnywhere 代理的 "scheduled" 策略。
- **支持新的 RDBMS 作为服务器消息存储库** 所有支持的 MobiLink 统一数据库现在都可在 QAnywhere 应用程序中用作服务器消息存储库：Adaptive Server Anywhere、Adaptive Server Enterprise、Microsoft SQL Server、Oracle 和 DB2。
- **.NET Compact Framework 的 QAnywhere .NET 客户端库现在支持消息监听器** .NET Compact Framework 的 QAnywhere .NET 客户端库现在支持消息监听器。

传输规则增强

- **现在同步了远程消息存储库属性** 现在当您设置远程消息存储库属性时，这些属性会与服务器消息库同步以便在传输规则中使用。
- **增强了消息存储库属性** `ias_Network` 属性现在包含可用于访问详细网络信息的字段。
请参见“客户端消息存储库属性”一节《QAnywhere》。
此外，您现在还能创建自定义的消息存储库属性。
- **删除消息的规则** 现在您可以为消息存储库中消息的持久性指定传输规则。您可以在客户端和服务端删除消息。
请参见“传输规则”一节《QAnywhere》。

QAnywhere 代理增强

- **连接字符串** 若要启动本地消息存储库，您现在可以使用 `qaagent -c` 选项指定连接字符串。这样您就可以使用 Adaptive Server Anywhere 连接字符串参数。
请参见“-c 选项”一节《QAnywhere》。
- **安静模式** QAnywhere 代理现在支持两种风格的安静模式，安静模式可避免某些 Windows CE 设备上的问题。
请参见“-q 选项”一节《QAnywhere》和“-qi 选项”一节《QAnywhere》。
- **QAstop 实用程序** 当使用 `-qi` 选项以安静模式启动 QAnywhere 代理时，您必须使用新的 `qastop` 实用程序停止它。
请参见“-qi 选项”一节《QAnywhere》。
- **增强了详细程度** 现在您可以使用 `-o` 或 `-ot` 选项来指定输出日志文件名，还可以使用 `-os` 和 `-ot` 选项来控制输出文件的大小。此外，`-v` 选项替换了旧的 `-verbose` 选项。使用 `-v`，您对记录输出具有更大的控制权。
请参见：
 - “-o 选项”一节《QAnywhere》
 - “-ot 选项”一节《QAnywhere》
 - “-on 选项”一节《QAnywhere》
 - “-os 选项”一节《QAnywhere》
 - “-v 选项”一节《QAnywhere》
- **初始化数据库以用作远程消息存储库** 您可以使用新的 `qaagent -si` 选项来设置远程消息存储库。
请参见“-si 选项”一节《QAnywhere》。
- **从 9.0.1 版升级** QAnywhere 代理有一个新的选项 `-su`，可将远程消息存储库从 9.0.1 版升级到 9.0.2 版。
请参见“-su 选项”一节《QAnywhere》。

QAnywhere MobiLink 系统表

所有 QAnywhere MobiLink 系统表现在都属于 `ml_qa_user_group`。此前，它们属于 DBO。

添加了两个新的 MobiLink 系统表。请参见：

- “[ml_qa_delivery](#)” 一节 《MobiLink - 服务器管理》
- [ml_qa_delivery_client](#)

多个 MobiLink 系统表的模式发生了更改。请参见：

- “[ml_qa_global_props](#)” 一节 《MobiLink - 服务器管理》
- [ml_qa_global_props_client](#)
- “[ml_qa_repository](#)” 一节 《MobiLink - 服务器管理》
- [ml_qa_repository_client](#)
- “[ml_qa_repository_props](#)” 一节 《MobiLink - 服务器管理》
- [ml_qa_repository_props_client](#)

以下生成的 MobiLink 系统表不用于 9.0.2 客户端：

- [ml_qa_repository_staging_client](#)
- [ml_qa_status_staging_client](#)

以下生成的 MobiLink 系统表不用于 9.0.2 服务器：

- [ml_qa_repository_content](#)

文档增强

本节介绍在 9.0.2 版中对 Adaptive Server Anywhere 文档的外观、结构和导航方法所作的增强，详尽地列出了所做的主要更改。

新文档

在几个方面对现有功能的文档进行了增强，包括：

- **SNMP 代理文档** 增加了介绍 Adaptive Server Anywhere SNMP 代理的新手册。
请参见 “[SQL Anywhere SNMP Extension Agent](#)” 《SQL Anywhere 服务器 - 数据库管理》。
- **Windows CE 入门** 增加了向 Windows CE 用户提供入门资料的章节。
请参见 “[SQL Anywhere for Windows Mobile](#)” 《SQL Anywhere 服务器 - 数据库管理》。
- **MobiLink 同步客户端的 DBTool 接口** 增加了如何在 DBTool 中使用 dbmlsync 的一个示例及其它信息。
请参见 “[dbmlsync 的 DBTools 接口](#)” 《MobiLink - 客户端管理》。
- **QAnywhere 增强** 扩展了 QAnywhere 文档，它现在新增了有关如何将消息传递与 JMS 消息系统和 MobiLink 数据同步进行集成的信息，并完善了有关设置 QAnywhere 应用程序的信息。
请参见 [QAnywhere](#)。
- **服务器启动的同步 SDK** 扩展了 SDK 的文档，并新增了有关 Palm 监听器 SDK 的章节。

请参见“用于 Palm 设备的 MobiLink 监听器 C API”《MobiLink - 服务器启动的同步》。

文档增强

- **MobiLink 重组** 重组了 MobiLink 手册，从而现在有客户端指南、管理指南和教程手册。除了 Adaptive Server Anywhere 客户端，客户端指南还介绍 UltraLite 客户端的同步参数和同步连接参数，这部分内容以前在《UltraLite 数据库用户指南》中介绍。
- **UltraLite API 和 QAnywhere API 参考** 现在，UltraLite.NET、UltraLite C++ API、QAnywhere .NET 和 QAnywhere C++ API 参考材料也以与其余文档相同的形式提供。因此，这些参考材料同时以 PDF 和基于 HTML 的文档提供。

9.0.2 版中的行为更改

本节列出了 SQL Anywhere Studio 9.0.2 版的组件中引入的行为更改。它还列出了不建议使用的功能，即当前软件支持这些功能，但 SQL Anywhere Studio 的下一主版本将不再支持。

不建议使用功能的列表可能会发生更改

与所有前瞻性声明一样，不建议使用的功能的列表不保证已完全列出，列表可能会发生更改。

Adaptive Server Anywhere 的行为更改

不建议使用和停用的功能

下面的列表包含不再支持或不建议使用的功能以及可能影响现有应用程序的功能。

- **删除了 min_table_size_for_histogram 选项** 数据库服务器不再使用 min_table_size_for_histogram 选项。在本软件以前的版本中，可使用此选项指定能为其创建直方图的最小表大小。现在 Adaptive Server Anywhere 自动为所有具有 5 行或更多行的表创建直方图。您可以使用 CREATE STATISTICS 语句为任何大小的表创建直方图。

请参见“更新列统计信息以提高优化程序性能”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **不建议使用的数据库选项** 不再支持下面的数据库选项：
 - truncate_date_values
 - assume_distinct_servers
- **不建议使用的旧数据库格式** 在 SQL Anywhere Studio 的下一主版本中，不支持在本软件旧版本中创建的数据库。将提供迁移工具。
- **已不建议使用 UNIX 非线性 DBTool 库** 不建议使用 UNIX 非线性 DBTools 库 (*libdbtool9.so*)：当前软件完全支持，但 SQL Anywhere Studio 的下一主要版本将不再支持。
- **不再支持 950TWN 归类** 950TWN 已由以下归类替代：950ZHO_HK 和 950ZHO_TW。

请参见“支持的归类和替代归类”一节《SQL Anywhere 服务器 - 数据库管理》。

其它行为更改

下面是对该软件以前版本的行为更改列表。

- **删除事务日志时对事务日志实用程序 (dblog) 的限制** 使用 -n 选项删除事务日志时，您必须还要指定相应的忽略事务日志偏移选项（日志传送管理器使用 -il，SQL Remote 使用 -ir，dbmsync 使用 -is）。
- 有关信息，请参见“事务日志实用程序 (dblog)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **运行于安静模式的数据库实用程序** 如果在指定了 -q 选项（安静模式）的情况下执行以下任一操作，则还必须指定 -y 选项：
 - 使用服务创建 (dbsvc) 实用程序修改或删除服务

- 使用数据源 (dbdsn) 实用程序修改或删除数据源
- 使用消除 (dberase) 实用程序消除文件
- 使用翻译日志文件 (dbtran) 实用程序翻译事务日志文件
- **使用 ECC_TLS 或 RSA 加密时必须提供证书名和口令** 已删除 `certificate`、`certificate_password` 和 `trusted_certificates` 参数的缺省值。这些缺省值使用 SQL Anywhere Studio 安装的 `win32` 目录中提供的示例证书。这些示例证书仅用于测试和开发，并不提供安全性。
此外，不再支持 `-ec all` 服务器选项。
请参见“[-ec 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **-xs 服务器选项更改** 不再支持 `-xs all` 服务器选项监听 HTTP 端口和 HTTPS 端口的 Web 请求。
请参见“[-xs 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **未使用缺省端口时，必须在 Mac OS X、HP-UX 和 Tru64 上指定网络数据库服务器的 TCP/IP 端口号** 如果在 Mac OS X、HP-UX 或 Tru64 上启动数据库服务器，而且缺省端口 (2638) 已被使用或不想使用此缺省端口，则必须使用 `ServerPort [PORT]` 协议选项来指定一个端口号。
请参见“[ServerPort 协议选项 \[PORT\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **更改了使用卸载实用程序 (dbunload) 卸载和重装的数据库的 Dbspace 文件名** 如果数据库是使用卸载实用程序 (dbunload) 的 `-an` 选项进行卸载和重装的，则会在旧 `dbspace` 文件名末尾附加一个 **R** 作为新数据库的 `dbspace` 文件名。这样做是为了防止将新的 `dbspace` 文件放到原始 `dbspace` 文件所在的目录中出现命名冲突。当您使用 Sybase Central 的 [卸载数据库向导] 卸载和重装数据时，`Dbspace` 文件名后面也会附加 **R**。
请参见“[卸载实用程序 \(dbunload\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **属性函数现在返回 LONG VARCHAR 值** 以前，以下函数返回 `VARCHAR(254)` 值。现在，它们返回 `VARCHAR(maxprossize)` 值，其中 `maxprossize` 取决于为服务器指定的最大页面大小。
 - `CONNECTION_PROPERTY`
 - `DB_EXTENDED_PROPERTY`
 - `DB_PROPERTY`
 - `EVENT_PARAMETER`
 - `PROPERTY`
- **STRTOUUID 函数更改** 在以前的版本中，如果向 `STRTOUUID` 传递一个无效的 UUID 值，它会返回 `NULL`。现在它返回转换错误，除非 `conversion_error` 选项被设置为 `OFF`，在这种情况下，它返回 `NULL`。

MobiLink 的行为更改

以下是对该软件先前版本的行为更改列表。

安全性行为更改

- **不建议使用 HTTP+TLS 安全性，赞成使用 HTTPS** 对于通过 HTTP 连接的客户端，不建议使用传送层安全。若要使用基于 HTTP 的传送层安全，您应该使用 HTTPS。

有关服务器端安全性的信息，请参见“启动支持传送层安全的 [MobiLink 服务器](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

有关客户端安全性的信息，请参见“配置 [MobiLink 客户端](#)以使用传送层安全”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **在 MobiLink 中使用 ECC_TLS 或 RSA 加密时，必须提供证书名和口令** 已删除 certificate、certificate_password 和 trusted_certificates 同步参数的缺省值。这些缺省值使用 SQL Anywhere Studio 安装的 win32 目录中提供的示例证书。这些示例证书仅用于测试和开发，并不提供安全性。

请参见“-x 选项”一节《[MobiLink - 服务器管理](#)》。

其它 MobiLink 行为更改

- **UDP 监听没有轮询间隔** 在监听器上，现在 UDP 连接没有轮询间隔。监听器会即时处理消息。请参见“[MobiLink 监听器实用程序](#)”《[MobiLink - 服务器启动的同步](#)》中的 **-I option**。
- **不建议使用 Treo 180 Smartphone 和 Kyocera 6035 Smartphone 上对 MobiLink Palm 监听器的支持** 有关 Palm 监听器支持的设备的信息，请参见“[用于 Palm 设备的实用程序](#)”一节《[MobiLink - 服务器启动的同步](#)》。

SQL Remote 行为更改

不建议使用和停用的功能

下面的列表包含不再支持或不建议使用的功能以及可能影响现有应用程序的功能。

- **不建议使用 SQL Remote for Adaptive Server Enterprise** 在 SQL Anywhere Studio 的下一主版本中，不再提供 SQL Remote for Adaptive Server Enterprise。MobiLink 将为 Adaptive Server Enterprise 和 Adaptive Server Anywhere 数据库之间的数据同步提供更灵活、更具有伸缩性的解决方案。

其它行为更改

以下是对该软件先前版本的行为更改列表。

- **抽取 (dbxtract) 实用程序** 当使用 dbxtract 抽取一个远程数据库时，如果指定了 -q 选项（安静模式），则还必须指定 -y 选项，以使 dbxtract 无需确认就能自动替换现有的命令文件。请参见“[抽取实用程序 \(dbxtract\)](#)”一节《[SQL Remote](#)》。
- **IPM_Receive 消息控制参数** MAPI IPM_Receive 消息控制参数的缺省值已更改为 YES。将此值设置为 YES 可确保 SQL Remote 能拾取到 IPC 和 IPM 消息。

UltraLite 的行为更改

UltraLite 的下一主版本将使用行业标准 API 来增强开发，并使用组件模型（而不是原有的静态接口）来增强开发。这些更改将为用户带来一些好处，包括能更轻松地使用 UltraLite 开发应用程序。

这些计划的结果是，有些 UltraLite API 建议不要在本版本中使用，即在当前软件中会继续为它们提供完全支持，但在下一主版本中不再提供支持。在下一主要版本中，将为迁移使用了不建议使用的接口的应用程序提供帮助。

与所有前瞻性声明一样，此处提供的不建议使用和停用的功能的列表可能会发生更改。

不建议使用和停用的功能

不建议使用或停用下面的功能。

- **不建议使用静态接口** SQL Anywhere Studio 的下一主版本将不支持静态 C++ API 或静态 Java API。将提供嵌入式 SQL 接口，但不是通过当前生成的代码机制。
- **UltraLite.NET 组件接口由 ADO.NET 替代** 在本版本中，UltraLite.NET 支持新的 iAnywhere.Data.UltraLite 命名空间中的 ADO.NET 开发。ADO.NET 既提供了行业标准接口的优点，又为大型应用程序提供了迁移到 Adaptive Server Anywhere 的简单途径。UltraLite.NET 组件 API (iAnywhere.UltraLite 命名空间) 在本版本中不建议使用，并且下一主要版本中将不再提供。
- **Native UltraLite for Java 组件接口由 JDBC 替代** 计划使用 JDBC 接口替代当前的 Native UltraLite for Java 接口。

其它行为更改

以下是对该软件先前版本的行为更改列表。

- **下载过程中对参照完整性删除的新警告** UltraLite 在下载过程中为维持参照完整性会自动删除某些行。它现在会在以这种方式删除每一行时都发出警告。
请参见“[参照完整性与同步](#)”一节《[MobiLink - 入门](#)》。
- **Native UltraLite for Java 行为更改** `Cursor.getRowCount()` 方法更改为返回 `int`。不需要更改应用程序。
- **UltraLite.NET 组件行为更改** `Cursor.getRowCount()` 方法更改为返回 `int`。不需要更改应用程序。
- **处理无效的同步参数** 在以前的版本中，UltraLite 运行时忽略所有无效的同步参数。因此会将拼写错误的参数忽略，而使用缺省值代替。
在本版本中，如果运行时遇到无效的参数，同步会失败，并设置 SQL 代码 `SQL_UNRECOGNIZED_OPTION`。如果提供了错误回调，则将为每个无效的参数调用一次错误回调。仍然忽略重复参数。
- **安全同步的新库** 同步的安全性组件已移到单独的库中。如果您为加密的同步使用 `ULSecureCerticomTLSStream` 或 `ULSecureRSATLSStream` 安全性组件，现在则必须针对相应的静态库分别进行链接，或提供单独的 DLL。

- **UltraLite for MobileVB 与 Crossfire 的集成** 如果现有项目使用了本软件较早版本的 UltraLite for MobileVB 与 Crossfire 的集成，则必须将对 *Interop.UltraLiteAFLib.dll* 的引用更改为对 *iAnywhere.UltraLiteForAppForge.dll* 的引用。

QAnywhere 行为更改

不建议使用和停用的功能

- **QAnywhere 代理选项** 以下是不建议使用的 QAnywhere 代理 (qaagent) 选项和替代的选项。

不建议使用的 qaagent 选项...	替代的 qaagent 选项...
-agent_id <i>id</i>	-id <i>id</i>
-dbauser <i>user</i>	-c "UID=<i>user</i>"
-dbeng <i>name</i>	-c "ENG=<i>name</i>"
-dbfile <i>filename</i>	-c "DBF=<i>filename</i>"
-dbname <i>name</i>	-c "DBN=<i>name</i>"
-ek <i>key</i>	-c "DBKEY=<i>key</i>"
-password <i>password</i>	-c "PWD=<i>password</i>"
-sv	-c "Start={ <i>dbeng9</i> <i>dbsrv9</i> }"
-verbose	-v[<i>levels</i>]

此外，下面的 qaagent 选项不再必需且不建议使用：

- **-e**
- **-rb**

请参见“[qaagent 实用程序](#)”一节《[QAnywhere](#)》。

- **QAnywhere 代理不再创建客户端消息存储库** 现在，您必须在运行 qaagent 之前自己创建消息存储库数据库。提供了新的选项 **-si**，它使用 QAnywhere 要求的系统对象初始化数据库。请参见“[设置客户端消息存储库](#)”一节《[QAnywhere](#)》。
- **带有 InReplyToID 的 QAnywhere 消息** QAnywhere 内置标头 InReplyToID 不再映射到 JMS 属性 `ias_ReplyToAddress`。这意味着该标头不再复制到 JMSCorrelationID 中。请参见“[从 JMS 连接器向 QAnywhere 客户端发送消息](#)”一节《[QAnywhere](#)》。

其它行为更改

- **减少了 QAnywhere 客户端库消息的开销** 不再为常规消息设置 `ias_MessageType` 属性。但仍为网络状态和发送到系统队列的其它系统消息设置这一属性。

9.0.1 版中的新功能

目录

9.0.1 版中的新功能	238
9.0.1 版中的行为更改	250

9.0.1 版中的新功能

本节列出在 SQL Anywhere Studio 9.0.1 版的组件中引入的新功能。

Adaptive Server Anywhere 的新功能

本节介绍 Adaptive Server Anywhere 9.0.1 版中的新功能。它提供主要和次要新功能的详尽列表，以及指向详细叙述每个功能的各章节位置的交叉引用。

OLAP 增强

- **OLAP 查询扩展** 可以通过一组联机分析处理（On-Line Analytical Processing，简称 OLAP）功能对数据库中的数据进行更详细的分析。增强的内容包括使用 CUBE 和 GROUPING SETS 以灵活的方式向结果集中添加小计行的功能，以及提供移动平均数和其它高级功能的窗口函数。

查询编辑器中置入了对这些 OLAP 功能的支持，使得您可以建立使用 ROLLUP、CUBE 和 GROUPING SETS 操作的查询。

请参见：

- “OLAP 支持” 一节 《SQL Anywhere 服务器 - SQL 的用法》
- “使用 ROLLUP” 一节 《SQL Anywhere 服务器 - SQL 的用法》
- “使用 CUBE” 一节 《SQL Anywhere 服务器 - SQL 的用法》
- “GROUP BY GROUPING SETS” 一节 《SQL Anywhere 服务器 - SQL 的用法》
- “窗口集合函数” 一节 《SQL Anywhere 服务器 - SQL 的用法》

- **新统计函数** 添加了多个统计函数。

请参见：

- “COS 函数 [Numeric]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “COVAR_POP 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “COVAR_SAMP 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “CUME_DIST 函数 [Ranking]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “DENSE_RANK 函数 [Ranking]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “PERCENT_RANK 函数 [Ranking]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “RANK 函数 [Ranking]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_AVGX 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_AVGY 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_COUNT 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_INTERCEPT 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_R2 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_SLOPE 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_SXX 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_SXY 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “REGR_SYY 函数 [Aggregate]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “ROW_NUMBER 函数 [Miscellaneous]” 一节 《SQL Anywhere 服务器 - SQL 参考》

● **新字符串函数** 添加了下列字符串函数：

- “BASE64_DECODE 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “BASE64_ENCODE 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “COMPRESS 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “DECOMPRESS 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “DECRYPT 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “ENCRYPT 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》
- “HASH 函数 [String]” 一节 《SQL Anywhere 服务器 - SQL 参考》

语句增强

- **LOAD TABLE 增强** LOAD TABLE 语句现在增加了可以对所创建的统计信息加以限制的子句，该子句能够加快表的装载。它还有一个 SKIP 选项，可以通过它忽略文件的前几行。

请参见“LOAD TABLE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。

- **SELECT ...INTO base-table** 此新 SELECT 语法创建一个基表并用查询中的数据填充该表。

请参见“SELECT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。

- **SQL 语句中变量的扩展支持** 允许在某些位置使用变量以及常量使几个语句的灵活性得到提高。这在存储过程和批处理（在其中可以声明和使用变量）中尤其有用。它提供了以前只能在 EXECUTE IMMEDIATE 中以更麻烦的形式使用的功能。

以下语句含有这种对变量的扩展支持：

- 现在 SELECT 语句的 TOP 子句可以引用整数变量以及常量。请参见“SELECT 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。
 - BACKUP 语句 *backup-directory* 和 *archive-root*。请参见“BACKUP 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。
 - RESTORE 语句 *filename*、*archive-root* 和新的 *dbspace-name*。请参见“RESTORE DATABASE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。
 - LOAD TABLE 语句 *filename*。请参见“LOAD TABLE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。
 - UNLOAD 语句和 UNLOAD TABLE 语句 *filename*。
- **SET 语句增强** 为兼容 Microsoft SQL Server，SET 语句现在可以接受 `ansi_nulls` 选项（等效于 `ansinull` 选项）。
- 有关信息，请参见“SET 语句 [T-SQL]”一节 《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER TABLE 语句增强** 现在 ALTER TABLE 可以向非空表添加具有缺省值的 NOT NULL 列。此功能提高了修改现有表时的灵活性。
- 请参见“ALTER TABLE 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER VIEW 语句增强** 现在 ALTER VIEW 语句支持 RECOMPILE 子句，该子句允许您在基表中的列被修改后重新创建视图定义。
- 请参见“ALTER VIEW 语句”一节 《SQL Anywhere 服务器 - SQL 参考》。

- **MESSAGE 语句增强** MESSAGE 语句中添加了 FOR CONNECTION 子句。

此外，MESSAGE 语句中还添加了 DEBUG ONLY 子句。当 `debug_messages` 选项设置为 ON 时，会为包含带有 DEBUG ONLY 子句的 MESSAGE 语句的所有存储过程和触发器显示调试消息。

请参见“MESSAGE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“`debug_messages` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

安全性增强

- **数据库页校验和** 数据库页校验和用于检测是否已在磁盘上修改了数据库页。创建启用了校验和的数据库时，在将每一页写入磁盘前会为其计算校验和。从磁盘中读取页面时，会再次计算其校验和并与存储的校验和进行比较。如果值不同，则该页面在磁盘上时已被修改或已损坏。必须卸载现有的数据库，然后重装到启用了校验和的数据库中才能使用此功能。使用 `Checksum` 属性可以检查是否为数据库启用了校验和。

有关创建具有校验和的数据库的详细信息，请参见“CREATE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》、“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》和“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。

校验和还可用来校验数据库。请参见“VALIDATE 语句”一节《SQL Anywhere 服务器 - SQL 参考》、“校验实用程序 (dbvalid)”一节《SQL Anywhere 服务器 - 数据库管理》和“`sa_validate` 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

性能增强

为提高多种任务（包括复杂查询）的性能，进行了许多增强。其中一些增强是纯内部的。以下列出了其它增强：

- **并行索引扫描** 在有多个磁盘转轴（如硬件或软件 RAID 阵列）的卷上，查询优化器现在可以使用索引并行地扫描表。
请参见“`ParallelIndexScan` 方法”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **聚簇散列 Group By 算法** Adaptive Server Anywhere 查询优化器可以使用新算法来提高性能，该算法对 GROUP BY 查询（其中 HAVING 子句返回行的一小部分）的特定类尤其有用。
请参见“`ClusteredHashGroupBy` 算法 (GrByHClust)”一节《SQL Anywhere 服务器 - SQL 的用法》和“`optimization_workload` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **数据库服务器高速缓存预热** 添加了三个新的数据库服务器命令行选项来支持高速缓存预热。高速缓存预热使用上次启动数据库时所引用的数据库页面预装载数据库服务器的高速缓存，从而帮助减少针对数据库的初始查询的执行时间。在每次启动数据库对其执行相同的查询时，使用高速缓存预热可以提高性能。
请参见“-cc 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》、“-cr 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》、“-cv 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》和“使用高速缓存预热”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **优化程序提示** WITH (XLOCK) 是 FROM 子句中一个新的表提示功能。XLOCK 指示将独占锁定提示的表中的语句所处理的行。在事务结束前，受影响的行将保持锁定状态。它在所有隔离级别都起作用。

WITH INDEX 提示会强制优化程序在查询优化期间使用指定的索引。这是一项高级功能，使用不正确时可能会导致性能不良，因此只应由有经验的用户使用。

请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》和“经常被搜索的列的索引”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **增加 NetWare 上的内部执行线程的缺省堆栈大小** NetWare 上内部执行线程的缺省堆栈大小已增加到 128 KB。

请参见“-gss 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

编程接口增强

- **Perl 接口** Perl DBI 模块的 Perl 新 DBD::ASAny 驱动程序允许您从 Perl 脚本访问和修改 Adaptive Server Anywhere 数据库。
- **InstallShield 项目** SQL Anywhere Studio 现在加入了 InstallShield 合并模块项目和对象项目。这些项目使得 InstallShield 能够生成合并模块和对象，可以使用这些模块和对象来重新部署计算机上当前安装的软件。旧版本的 SQL Anywhere 包含合并模块和对象。这使得您可以重新部署原有软件，但对于应用了 EBF 后的部署并没有提供便利的方法。

管理增强

- **BACKUP 增强** 现在 BACKUP 语句加入了 ON EXISTING ERROR 子句来进行映像备份。指定此子句后，如果要在备份期间创建的任何文件已存在，便会出现错误。
BACKUP 语句的档案备份形式已被扩展至支持以前只能用于映像备份的选项。
请参见“BACKUP 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
现在备份实用程序可以在服务器计算机上创建备份。以前该实用程序只能在客户端计算机上创建备份。
请参见“备份实用程序 (dbbackup)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **卸载实用程序 (dbunload) 增强** 现在卸载数据库时，卸载实用程序会自动处理视图依赖性。已不建议使用以前版本的软件中使用的 -j 选项，该选项用于将视图定义多次输出到 *reload.sql* 文件。现在，卸载实用程序会自动处理依赖其它视图的视图定义的卸载。
卸载到新数据库中时，卸载实用程序还允许您更改数据库页面大小。
请参见“卸载实用程序 (dbunload)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **服务器枚举实用程序 (dblocate) 增强** 现在，可以通过服务器枚举实用程序 (dblocate) 提供主机名或 IP 地址，将对数据库服务器的搜索范围限定为特定计算机。此外，该实用程序还支持 -n 选项，该选项指定不将 IP 地址解析为计算机名，而这会提高性能。
请参见“服务器枚举实用程序 (dblocate)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Adaptive Server Anywhere 控制台实用程序支持集成登录** 当您连接到 Windows NT/2000/XP 上的 Adaptive Server Anywhere 控制台 (dbconsole) 实用程序时，可以通过 [连接] 窗口使用集成登录连接到数据库。
- **可以更改请求日志文件的大小而不需要重新启动数据库服务器** 启动数据库服务器时，可以使用 -zs 服务器选项指定请求日志文件的大小。可以使用 sa_server_option 系统过程更改请求日志文件的大小，而不必重新启动数据库服务器。

请参见“[sa_server_option 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **为分析系统触发器添加的附加信息** 在数据库中打开过程分析时，sa_procedure_profile 系统过程和 sa_procedure_profile_summary 系统过程现在会返回关于系统触发器的额外信息。

请参见“[sa_procedure_profile 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[sa_procedure_profile_summary 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **新系统表** 添加了一个新系统表，用于维护关于数据库用于启动的软件和平台的不同版本的信息。

请参见“[ISYSHISTORY 系统表](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **新归类** 可供使用的新归类有两个：一种用于支持立陶宛语（1257LIT、ANSI 代码页 1257），另一种用于支持土耳其语（1254TRKALT）。此土耳其语归类对 I-dot 和 I-no-dot 不加区分。

请参见“[支持的归类和替代归类](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[替代土耳其语归类 1254TRKALT](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **dedicated_task 选项** 指定后，将有一个请求处理任务专用于处理来自某单个连接的请求。在数据库服务器意外地不做响应时，可以通过此预先建立的连接收集有关该数据库服务器状态的信息。

请参见“[dedicated_task 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

Interactive SQL 增强

- **Interactive SQL 可以指定用于读写文件的编码** 现在 Interactive SQL READ、INPUT 和 OUTPUT 语句支持一个可选编码子句，可以通过它指定读取或写入文件时使用的字符编码。添加了 default_isql_encoding 选项，可以通过它指定用于后续 READ、INPUT 和 OUTPUT 语句的字符编码。

请参见“[default_isql_encoding 选项 \[Interactive SQL\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》、“[READ 语句 \[Interactive SQL\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》、“[INPUT 语句 \[Interactive SQL\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[OUTPUT 语句 \[Interactive SQL\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

使用 Interactive SQL 导入和导出向导时，您也可以指定用于读写文件的字符编码。

请参见“[导入和导出数据](#)”《[SQL Anywhere 服务器 - SQL 的用法](#)》。

- **Interactive SQL 支持集成登录** 当您连接到 Windows NT/2000/XP 上的 Interactive SQL 时，可以通过 [连接] 窗口使用集成登录连接到数据库。
- **Interactive SQL 允许您配置用于显示结果集的字体** 可以为显示在 Interactive SQL 的 [结果] 窗口中的数据选择字体、字形和磅值大小。
- **Interactive SQL 允许您指定用于文件浏览的初始文件夹** 在 Interactive SQL 中浏览文件时，可以指定 Interactive SQL 是使用当前目录（由操作系统定义）还是上一次打开文件时所在的文件夹作为初始目录。

Sybase Central 增强

- **Sybase Central 允许您配置用于显示结果集的字体** 选中一个表后，可以为显示在 Sybase Central 的 [数据] 选项卡上的数据选择字体、字形和磅值大小。

- **[创建远程服务器向导] 现在支持为当前用户创建外部登录** [创建远程服务器向导] 现在允许为当前用户创建外部登录，这样就不必在创建远程服务器之前创建外部登录。
请参见“使用 Sybase Central 创建远程服务器”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **Sybase Central 支持集成登录** 当您连接到 Windows NT/2000/XP 上的 Sybase Central 时，可以通过 [连接] 窗口使用集成登录连接到数据库。
- **可以在 Sybase Central 中使用 [查看] 菜单为列排序** Sybase Central 的 [查看] 菜单中有 [排序] 菜单项，可以用来为右窗格中的列排序，另一方法是在右窗格中单击列标题。
- **可以在外键属性窗口中修改外键设置** 在 Sybase Central 中可通过 [外键] 属性窗口更改外键设置。
- **[代理表] 向导现在显示主键列信息** 以前在使用 [代理表] 向导创建代理表时，无法确定哪些列属于远程表的主键。现在，向导中对主键中的列进行了标识。
- **可以取消实用程序向导** [升级数据库向导]、[备份数据库向导]、[恢复数据库向导]、[校验数据库向导]、[压缩数据库向导]、[解压缩数据库向导] 和 [创建备份映像向导] 都可以取消。它们还加入了一个消息窗口，其中显示有关操作是否成功的状态信息。
- **创建和编辑服务时，Sybase Central 支持 domainuser 形式的帐户名** 现在创建和编辑服务时，可以通过 [创建服务向导] 和 [服务] 属性窗口输入 *domainuser* 形式的帐户名。可以在 [服务] 属性窗口 [帐户] 选项卡的 [其它帐户] 字段或 [创建服务向导] 中输入帐户名。

其它增强

- **在 Linux 平台上，数据库服务器使用异步 I/O** 在 Linux 上运行数据库服务器时，数据库服务器在缺省情况下会尽可能使用异步 I/O。通过 `-ua` 数据库服务器选项可以禁用异步 I/O。
请参见“-ua 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **openxml 支持等式谓词** 可以通过 openxml 函数在 XPath 表达式中使用等式谓词。通过此功能可以使用属性值找到 XML 文档内的节点。
请参见“openxml 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **TransactionStartTime 连接属性** 该属性返回数据库在 COMMIT 或 ROLLBACK 操作后第一次修改的时间。
请参见“连接属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **UserAppInfo 属性** 该属性返回连接字符串中用 AppInfo 连接参数指定的部分。
请参见“连接属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **ConsoleLogFile 服务器属性** 此属性返回指定 `-o` 服务器选项时记录数据库服务器消息窗口中消息的文件的名称。
请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **UNIX 平台上的 DriveType 数据库属性** DriveType 数据库属性已被扩展至 UNIX 平台。
请参见“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **连接 ID 从 1 开始并为数据库服务器的每个新连接递增** 数据库服务器启动时，系统会为至该服务器的每个连接指派一个连接 ID（起始连接 ID 为 1），每当新建一个至该服务器的连接时，

连接号便会增加 1。连接 ID 记录在 -z 服务器输出和 LogFile 连接参数输出中。CONNECTION_PROPERTY、NEXT_CONNECTION、NEXT_DATABASE 和 DROP CONNECTION 函数以及请求记录也会使用它们。

请参见“CONNECTION_PROPERTY 函数 [System]”一节《SQL Anywhere 服务器 - SQL 参考》、“NEXT_CONNECTION 函数 [System]”一节《SQL Anywhere 服务器 - SQL 参考》、“NEXT_DATABASE 函数 [System]”一节《SQL Anywhere 服务器 - SQL 参考》和“请求记录”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **NetWare 和 UNIX 中改进的高速缓存管理** 当用 -c 指定的高速缓存大小大于 UNIX 或 NetWare 上的可用内存量时，数据库服务器即会根据可用内存计算最大高速缓存大小。

有关数据库服务器如何在上述情况下计算最大高速缓存大小的详细信息，请参见“-c 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **odbc_distinguish_char_and_varchar 选项** odbc_distinguish_char_and_varchar 选项控制 Adaptive Server Anywhere ODBC 驱动程序描述 CHAR 列的方式。

请参见“odbc_distinguish_char_and_varchar 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

MobiLink 的新功能

以下列出了 9.0.1 版中引入的对软件的更改和补充。

- **QAnywhere 消息传递** MobiLink QAnywhere 提供了应用程序间的消息传递功能。通过该功能可以编写应用程序来与运行 Windows 或 Windows CE 操作系统的多种设备上的远程应用程序交换消息。

请参见 [QAnywhere](#)。

- **外部验证** MobiLink 用户验证得到了增强，使您可以轻松地使用 LDAP 服务器和其它外部源（如 POP3 电子邮件服务器）对用户进行验证。

请参见“向外部服务器验证”一节《MobiLink - 客户端管理》。

- **新 MobiLink 系统表** 新增了几个 MobiLink 系统表。新 MobiLink 系统表的存在意味着您需要升级 Adaptive Server Anywhere 数据库，并需要为其它统一数据库运行升级脚本。

请参见“MobiLink 服务器系统表”《MobiLink - 服务器管理》。

- **可配置脚本版本** 您可以使用新的 ml_property MobiLink 系统表来存储脚本版本的属性。

请参见“ml_property”一节《MobiLink - 服务器管理》和“ml_add_property 系统过程”一节《MobiLink - 服务器管理》。

- **iAnywhere ODBC 驱动程序** 现在 Windows 上提供了用于 DB2 的 iAnywhere ODBC 驱动程序。这是一个 Wire Protocol 驱动程序，因此不需要 DB2 客户端软件。

- **IBM DB2 安装脚本具有版本号** 现在多个版本的 MobiLink 服务器可以使用同一 DB2 服务器实例。这是因为 MobiLink 用于 DB2 存储过程的两个 Java 类文件现在包含 SQL Anywhere Studio 版本号。它们在 9.0.1 发行版本中称为 *SyncDB2_901.class* 和 *SyncDB2Long_901.class*。

请参见“IBM DB2 LUW 统一数据库”一节《MobiLink - 服务器管理》。

- **新 -us 服务器选项** 这一新服务器选项通过防止服务器调用不必要的表脚本提高了性能。
- **ActiveSync 提供程序现在生成一个活动日志文件** 现在 ActiveSync 提供程序可以为其活动生成日志。
请参见“[ActiveSync 提供程序的安装实用程序 \(mlasinst\)](#)”一节《[MobiLink - 客户端管理](#)》。

Adaptive Server Anywhere 客户端增强

- **Adaptive Server Anywhere 客户端改进的应用程序集成** dbmsync 的一个新集成组件为您在 Windows 平台上通过 Adaptive Server Anywhere 远程数据库创建应用程序提供了一种更容易且自定义化程度更高的方法。
请参见“[Dbmsync 集成组件（不建议使用）](#)”《[MobiLink - 客户端管理](#)》。
- **恢复失败的下载** 现在当 Adaptive Server Anywhere 和 UltraLite 远程数据库下载都失败时，可以避免让数据进行长时间的重新传输。传输了部分下载后，可以继续传输失败的下载。
请参见“[恢复失败的下载](#)”一节《[MobiLink - 服务器管理](#)》。
- **Adaptive Server Anywhere 客户端的事务级上载** 您现在可以选择在上载中保留远程数据库上的事务，而且可以分别对每次同步执行此操作。
请参见“[-tu 选项](#)”一节《[MobiLink - 客户端管理](#)》。
- **指定扩展选项的新方式** 您可以使用挂接 `sp_hook_dbmsync_set_extended_options` 以编程方式自定义即将发生的同步的行为。
请参见“[sp_hook_dbmsync_set_extended_options](#)”一节《[MobiLink - 客户端管理](#)》。

服务器启动的同步

- **自动设备跟踪** 自动设备跟踪简化了通知过程。
请参见“[设备跟踪网关](#)”一节《[MobiLink - 服务器启动的同步](#)》。
- **Sybase Central 配置** 现在可以使用 Sybase Central MobiLink 插件配置通知。
请参见“[使用 Sybase Central 配置服务器端设置](#)”一节《[MobiLink - 服务器启动的同步](#)》。
- **可选传送确认** 您现在可以配置通知，以便在收到消息后自动向统一数据库发送确认信息。
请参见“[用于 Windows 设备的监听器实用程序](#)”一节《[MobiLink - 服务器启动的同步](#)》。
- **新监听器选项** 监听器有一些新的命令行选项，例如同时监听多个通道的功能。
请参见“[MobiLink 监听器实用程序](#)”《[MobiLink - 服务器启动的同步](#)》。
- **Palm 配置** 现在有一个可简化 Palm 设备的配置的 Palm 监听器配置实用程序 (dbsncfg)。
请参见“[用于 Palm 设备的监听器配置实用程序](#)”一节《[MobiLink - 服务器启动的同步](#)》。

MobiLink 监控器

- **导出监控器数据** 现在可以将监控器数据导出到关系数据库或 Excel 文件。
- **监控器的增强信息** 现在您可以自定义在表中显示的列。此外，还新增了一些列，它们包含以前只能从同步属性窗口获得的信息，并有一个新增的列用于对监控器会话的同步进行唯一标识。

- **改进的排序** MobiLink 监控器表的排序得到了改进。现在添加或更新数据时排序顺序会得到保留。
- **增强的用户界面** 新增了一些菜单和工具栏按钮，用于在图表中放大、缩小或缩放到选择区域。此外，暂停功能现在可以控制图表是否自动滚动。

请参见“[MobiLink 监控器](#)” 《[MobiLink - 服务器管理](#)》。

新的 Web 服务器支持

- **重定向器支持 Apache Web 服务器** 新增了用于 Apache Web 服务器的本机重定向器。现在不需要 Tomcat 也可以使用 Apache Web 服务器。

请参见“[重定向器（不建议使用）](#)” 《[MobiLink - 服务器管理](#)》。

UltraLite 的新功能

UltraLite 9.0.1 引入了一些新功能：

- **UltraLite.NET 控件** Visual Studio.NET 2003 工具箱中添加了一组控件，从而便于在 UltraLite.NET 应用程序中指定连接参数和监控同步。
请参见“[教程：构建 UltraLite.NET 应用程序](#)” 《[UltraLite - .NET 编程](#)》。
- **UltraLite for M-Business Anywhere** 提供了一个可用于 iAnywhere M-Business Anywhere 的新组件，该组件以前称为 AvantGo M-Business Server。
请参见 [UltraLite - M-Business Anywhere 编程](#)。
- **动态 SQL 中的 CREATE 和 DROP 语句** 现在可以在动态 SQL 中使用 CREATE/DROP TABLE 和 CREATE/DROP INDEX 语句。对 UltraLite 组件的用户而言，这些语句提供了一种更改 UltraLite 数据库模式的方法。
请参见“[UltraLite CREATE INDEX 语句](#)”一节 《[UltraLite - 数据库管理和参考](#)》和“[UltraLite CREATE TABLE 语句](#)”一节 《[UltraLite - 数据库管理和参考](#)》。
- **动态 SQL 中的事务控制** 现在可以在动态 SQL 中使用 COMMIT 和 ROLLBACK 语句。对于使用 UltraLite 组件用户，这些语句为使用 SQL 语句控制事务提供了一种方法。它们可以替代连接对象上的 commit 和 rollback 方法。
请参见“[UltraLite COMMIT 语句](#)”一节 《[UltraLite - 数据库管理和参考](#)》和“[UltraLite ROLLBACK 语句](#)”一节 《[UltraLite - 数据库管理和参考](#)》。
- **动态 SQL 的 SELECT 增强** 可以在搜索条件的 WHERE 子句或 HAVING 子句中使用子查询。它们也可用作 FROM 子句中的派生表。
请参见“[UltraLite SELECT 语句](#)”一节 《[UltraLite - 数据库管理和参考](#)》和“[UltraLite 中的搜索条件](#)”一节 《[UltraLite - 数据库管理和参考](#)》。
现在支持 HAVING 子句。请参见“[UltraLite SELECT 语句](#)”一节 《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite 的 ODBC 接口** UltraLite 现在支持部分 ODBC 编程接口。

- **混合 C++ 接口** 基于 UltraLite C/C++ 的接口（嵌入式 SQL、静态 C++ API 和 C++ 组件）可用于同一应用程序中。

这对在现有的嵌入式 SQL 或静态 C++ API 应用程序中添加 C++ 组件动态 SQL，或在主要基于 C++ 组件的应用程序中使用嵌入式 SQL 执行常规 SQL 特别有用。

新函数包括“[db_start_database 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》（嵌入式 SQL）和 StartDatabase 方法。

- **用于 UltraLite C++ 组件的 CodeWarrior 模板** Palm OS UltraLite C++ Component App 模板作为 CodeWarrior 的 UltraLite 插件的一部分提供。在使用面向 Palm OS 的 CodeWarrior 构建 C++ 组件应用程序时，它会提供协助。

在 UltraLite 安装过程中，会将用于 CodeWarrior 的 UltraLite 插件的文件放在磁盘上，但是如果并没有附加的安装步骤，则无法使用该插件。

请参见“[开发用于 Palm OS 的 UltraLite 应用程序](#)”《[UltraLite - C 及 C++ 编程](#)》。

- **改进的 UltraLite C/C++ 错误处理** 现在所有 UltraLite C/C++ 接口都支持错误回调。回调使应用程序能够获知所有错误，从而在开发过程中为开发人员提供宝贵的信息。

请参见“[ULRegisterErrorCallback 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》和“[ULRegisterErrorCallback 的回调函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。

- **UltraLite 组件可使用引擎** UltraLite 数据库引擎现在可作为备选部署选项用于 UltraLite 组件（该引擎可接受来自多个应用程序的连接）。

此选项在 UltraLite for MobileVB 或 UltraLite ActiveX 中不可用。

- **数据库转换工具** ulconv 实用程序是一个用于对 UltraLite 数据库执行多项操作的命令行工具，例如将数据库卸载到 XML 文件、从 XML 文件装载新数据库以及转换数据库格式。

- **附加同步进度事件** 当发生错误并且正在回退已下载的更改时，有另外一个事件可用于同步观察器。

请参见正在使用的 API 的 ULSyncState 结构或对象。

- **模式升级监控** 模式升级的运行时间可能会很长。新的模式升级事件为应用程序提供了对模式升级进度进行监控的机制。

- **可重新启动的下载** 现在 UltraLite 可以通过同步观察器重新启动由于通信错误或用户取消引起的失败的下载。

请参见“[恢复失败的下载](#)”一节《[MobiLink - 服务器管理](#)》。

- **新的 Windows CE 平台支持** 现在 UltraLite 支持 Smartphone 2002 平台。此平台不支持 ActiveSync 同步。

UltraLite 还支持基于 ARM 芯片、以 V4T ("thumb") 模式运行的 Windows CE 4.1。

- **多数据库支持** UltraLite 组件可以通过发出多个指定不同数据库文件名或创建者 ID 的连接请求来从一个应用程序处理多个数据库。

此功能为连接参数带来了一些扩展。

请参见“[UltraLite DBN 连接参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。

- **不再需要 ULPalmLaunch 和 ULPalmExit** UltraLite 现在支持更多与连接有关的基元，这些基元简化了对应用程序关闭时进行状态维护的支持。这些新功能使得 Palm OS 应用程序不再需要特殊的 Palm 特定的基元（包括 ULPalmLaunch 和 ULPalmExit）。

请参见“在 UltraLite Palm 应用程序中维护状态（不建议使用）”一节《UltraLite - C 及 C++ 编程》和“连接到数据库”一节《UltraLite - C 及 C++ 编程》。

- **UltraLite 数据库属性** 现在 UltraLite 组件应用程序可以使用 UltraLite 数据库的属性。用于全局自动增量值的区分大小写特性、归类和数据库 ID 都可作为 Connection 对象的属性或方法使用，但要视 API 而定。

文档增强

本节将介绍 9.0.1 版 Adaptive Server Anywhere 文档的外观、结构和导航方法，详尽地列出了所做的主要更改。

新文档

- **MobiLink 服务器启动的同步** MobiLink 服务器启动的同步的特性已经归集到它自己的手册中。请参见 [MobiLink - 服务器启动的同步](#)。
- **QAnywhere 消息传递** 现在有一本手册介绍新的 MobiLink 消息传递应用程序 QAnywhere。请参见 [QAnywhere](#)。
- **新 MobiLink 教程** 提供了介绍如何对 Adaptive Server Anywhere 远程数据库使用 Java 和 .NET 脚本逻辑的新教程。请参见“教程：使用 Java 同步逻辑”《MobiLink - 入门》和“教程：使用 .NET 同步逻辑”《MobiLink - 入门》。
- **新增介绍 dbmlsync 集成组件的一章** 请参见“Dbmlsync 集成组件（不建议使用）”《MobiLink - 客户端管理》。
- **组合的 UltraLite C/C++ 手册** UltraLite C/C++ 接口（嵌入式 SQL、静态 C++ API 和 C++ 组件）的文档合并到了一本手册中，其中还包括新的 ODBC 接口。请参见 [UltraLite - C 及 C++ 编程](#)。
- **UltraLite for M-Business Anywhere** 新的 UltraLite for M-Business Anywhere 有其自己的手册。请参见 [UltraLite - M-Business Anywhere 编程](#)。

文档增强

- **《Adaptive Server Anywhere 入门手册》和《SQL Anywhere Studio 介绍》合并** 《Adaptive Server Anywhere 入门手册》和《SQL Anywhere Studio 介绍》已合并为一本手册。新手册的名称是《SQL Anywhere Studio 介绍》。
- **增强的 PDF 手册** 有人发现 PDF 可作为基于 HTML 的联机手册的一种有益替代，对于概念性材料尤其如此。现在在缺省情况下会安装 PDF 版本的联机手册，可以从 Windows 的 [开始] 菜单或通过从在联机手册中单击每个主题顶部的 PDF 项进行访问。

PDF 文件以可单击链接为特色，这些链接不仅可以指向手册中的内容，还可以指向其它手册以及 Web 站点上的材料。这些功能的行为取决于是从浏览器插件还是直接从 Acrobat Reader 查看文件。要获得最佳体验，请直接使用 Acrobat Reader。

- **关于帮助的帮助信息** 新增了一节，内容是介绍 Windows HTML 帮助格式与 PDF 格式联机手册之间的差异，还介绍了如何使用各种帮助功能来浏览文档和访问要寻找的信息。

9.0.1 版中的行为更改

本节列出在 SQL Anywhere Studio 9.0.1 版的组件中引入的行为更改。

Adaptive Server Anywhere 的行为更改

不建议使用和停用的功能

此列表包含不再支持或不建议使用的功能以及可能影响现有应用程序的功能。

- **MDSR 加密停用** 以前 MDSR 和 AES 高度加密在 Adaptive Server Anywhere 都得到了支持。现在唯一支持的高度加密为 AES 加密。这一更改意味着通过初始化、抽取或卸载实用程序使用加密功能时，`-ea` 选项不再是必需的选项。此外，还删除了 Create Database 语句和 Create Encrypted File 语句中的 `algorithm` 参数。
- **不再支持写文件** 此版本中不再支持写文件。
- **不再支持压缩数据库文件** 此版本中不再支持使用压缩数据库文件。
- **不再支持卸载实用程序 (dbunload) 的 -j 选项** 对 Unload 实用程序的增强使得卸载实用程序的 `-j` 选项不再受到支持。
- **不建议使用语言选择实用程序 (dblang) 的 -d 选项** 不再支持语言选择实用程序的 `-d` 选项，该选项以前用于更改 Adaptive Server Anywhere 的注册表设置。

其它行为更改

下面是对该软件以前版本的行为更改列表。

- **添加了 CURRENT_TIMESTAMP 和 CURRENT_USER 特殊值** 为了与 Microsoft SQL Server 兼容，添加了 CURRENT_TIMESTAMP（等效于 CURRENT_TIMESTAMP）和 CURRENT_USER（等效于 CURRENT_USER）特殊值。
请参见“CURRENT_TIMESTAMP 特殊值”一节《SQL Anywhere 服务器 - SQL 参考》和“CURRENT_USER 特殊值”一节《SQL Anywhere 服务器 - SQL 参考》。
- **如果数据库服务器已经在运行，则 db_start_engine 返回 SQLCODE 0** 现在，如果数据库服务器已经在运行，db_start_engine 会返回非零值并将 SQLCODE 设置为 0。以前，db_start_engine 也会返回非零值，但会将 SQLCODE 设置为 SQLE_ENGINE_ALREADY_RUNNING。
请参见“db_start_engine 函数”一节《SQL Anywhere 服务器 - 编程》。
- **如果数据库服务器已经在运行，db_start_database 返回非零值和 SQLCODE 0** 现在，如果数据库已经在运行，db_start_database 会返回非零值并将 SQLCODE 设置为 0。以前，db_start_database 会返回 0（表示失败）并将 SQLCODE 设置为 SQLE_ALIAS_CLASH。
请参见“db_start_database 函数”一节《SQL Anywhere 服务器 - 编程》和“db_start_database 函数”一节《UltraLite - C 及 C++ 编程》。
- **更改了校验实用程序的缺省算法** 现在，校验实用程序 (dbvalid) 缺省情况下会使用快速检查（`-fx` 选项）算法。对于有多个索引且不能将表完全装入服务器高速缓存的大型表，快速检查算法

的执行速度要快很多。如果想要使用以前版本的 Adaptive Server Anywhere 中使用的校验算法，可以指定 `-fn` 选项。

请参见“[校验实用程序 \(dbvalid\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **在 [连接] 窗口中输入多字节字符** 为了遵从 Windows 的安全最佳做法，您不能再使用输入法编辑器 (IME) 在 Sybase Central、Interactive SQL 和 Adaptive Server Anywhere 控制台实用程序中使用的 [连接] 窗口的 [口令] 字段内键入日文和其它亚洲语言的多字节字符。

如果您现有数据库的口令中包含这些字符，可以在 [连接] 窗口和 [ODBC 配置] 窗口 [高级] 选项卡的 [其它连接参数] 字段中键入口令。不过，应注意在 [高级] 选项卡中键入口令时，口令不会隐藏，而是可见的纯文本。建议在升级数据库时更改口令，以使口令中不包含多字节字符。

- **不能在 DECLARE LOCAL TEMPORARY TABLE 语句中指定所有者名称** 在以前版本的软件中，如果在 DECLARE LOCAL TEMPORARY TABLE 中指定了所有者名并且所有者与当前用户不同，则可以创建多个同名的临时表。现在，如果指定所有者名，会发生语法错误。
- **更改了 min_table_size_for_histogram 选项的缺省设置** `min_table_size_for_histogram` 选项指定创建的直方图所对应的表的最小大小。缺省值现已更改为 100 行。在以前版本的软件中，缺省值为 1000 行。在使用以前版本的软件创建的数据库中，可以通过 SET OPTION 语句更改此设置。
- **NULL 常量数据类型转换更改** 在以前版本中，将 NULL 常量转换为 CHAR、VARCHAR、LONG VARCHAR、BINARY、VARBINARY 或 LONG BINARY 类型时，如果未提供长度，系统会将列的大小初始化为 32767。现在，该大小初始化为 0。

例如，以前以下查询会返回描述长度为 32767 的一列：

```
SELECT CAST( NULL AS CHAR )
-- This now returns a CHAR(0) column

SELECT 'abc'
UNION ALL
SELECT NULL
-- This now returns a CHAR(3) column

SELECT ''
UNION ALL
SELECT NULL
-- This now returns a CHAR(0) column

SELECT IF 1=1 THEN 'abc' ELSE NULL ENDIF
-- This now returns a CHAR(3) column
```

- **UPDATE 语句以及 ORDER BY 子句使用序号值时的错误** 包含使用序号值的 ORDER BY 子句的 UPDATE 语句现在返回语法错误。
- **标识符的限制** 标识符中不能再使用双反斜线或双引号。只有作为转义字符时才允许在标识符中使用反斜线。

请参见“[标识符](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **EXECUTE IMMEDIATE 语句缺省设置 WITH RESULT SET 子句** 指定 WITH RESULT SET ON 子句时，EXECUTE IMMEDIATE 语句可以返回结果集。缺省设置是 WITH RESULT SET OFF。请参见“[EXECUTE IMMEDIATE 语句 \[SP\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

MobiLink 的行为更改

以下是对该软件先前版本的行为更改列表。

- **挂接调用顺序的更改** 调用事件挂接的顺序进行了更改。这意味着现在增量上载的效率会更高，因为上载事件是同步序列中唯一在每次进行增量更新时都要重复执行的部分。

请参见“同步事件挂接序列”一节《MobiLink - 客户端管理》。

- **对于 Adaptive Server Anywhere 客户端发布的按行分区的修正** 包含 WHERE 子句的发布现在只复制满足 WHERE 条件的行。从 8.0.0 版到 9.0.0 版一直存在一个错误，当 WHERE 子句计算得到未知值时，该错误会使行被复制。例如，如果发布 WHERE 子句中有 "WHERE val = 1"，则 val 为 NULL 的行也将被复制。此错误对 SQL Remote 和 Adaptive Server Anywhere MobiLink 客户端都有影响。

UltraLite 的行为更改

以下是对该软件先前版本的行为更改列表。

- **Palm OS 状态管理** 对于使用嵌入式 SQL 或静态 C++ API 的 Palm OS 应用程序，管理状态和同步信息时不再需要使用 `ULPalmExit (ULData::PalmExit)` 和 `ULPalmLaunch (ULData::PalmLaunch)` 函数，现在也不建议使用这些函数。同样不建议使用 `ULData` 和 `ULConnection Reopen` 方法。

Palm OS 上的应用程序现在使用与其它应用程序相同的初始化、连接和关闭函数序列。`ULSetSynchInfo` 方法控制 HotSync 同步。

请参见“将 HotSync 同步添加到 Palm 应用程序”一节《UltraLite - C 及 C++ 编程》。

- **不再支持 Palm OS 3.0** 此版本中受支持的最早版本是 Palm OS 3.5。
- **不再支持 ULEnableGenericSchema 函数** 要求模式升级的 UltraLite C/C++ 应用程序不再需要调用 `ULEnableGenericSchema`。请改为使用 `ULRegisterSchemaUpgradeObserver` 函数。
- **UltraLite 组件 Table API** `Table` 对象的 `Delete` 方法在删除行后不再自动刷新行。要保持以前的行为，请在 `Delete` 操作后使用 `Relative(0)` 重新读取该行。
- **不再需要 Native UltraLite for Java 对列 ID 和参数 ID 的转换** 所有接受列 ID 和参数 ID 的方法和部分接受其它短类型参数的方法已更改为可以接受整数。这样便无需在代码中转换数字常量。例如，现在可以使用 `table.getString(1);`，而不使用 `table.getString((short)1);`。

此更改使得 Native UltraLite for Java 应用程序必须经过重新编译才能与 9.0.1 版软件配合使用。不需要更改代码。

9.0.0 版中的新功能

目录

9.0.0 版中的新功能	254
9.0 版中的行为更改	269

9.0.0 版中的新功能

本节列出在 SQL Anywhere Studio 9.0.0 版的组件中引入的新功能。

Adaptive Server Anywhere 的新功能

本节介绍 Adaptive Server Anywhere 9.0.0 版中的新功能。它提供主要和次要新功能的详尽列表，并提供交叉引用指向详细叙述各功能的章节。

主要新功能

- **XML 支持** Adaptive Server Anywhere 9.0.0 对 XML 提供了广泛的支持，包括存储 XML 文档、以 XML 格式导出关系数据、导入 XML 以及从对关系数据的查询中返回 XML。

- **FOR XML 子句** SELECT 语句支持使用 RAW、AUTO 和 EXPLICIT 这三种模式的 FOR XML 子句，它们使您可以获得 XML 文档格式的查询结果。可以通过不同模式对生成的 XML 格式进行不同级别的控制。

请参见“[以 XML 格式获取查询结果](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》和“[SELECT 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **for_xml_null_treatment 选项** 可以使用 for_xml_null_treatment 选项来控制包含 FOR XML 子句的查询如何返回 NULL 值。

请参见“[for_xml_null_treatment 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **openxml 过程** 请参见“[openxml 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **SQL/XML 支持** SQL/XML 是一项描述如何将 SQL 与 XML 联合使用的标准草案。作为对 SQL/XML 支持的一部分，Adaptive Server Anywhere 提供了一种 XML 数据类型，可以使用该数据类型在数据库中存储 XML 文档。

请参见“[XML 数据类型](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

Adaptive Server Anywhere 还支持以下 SQL/XML 函数，这些函数可作为 FOR XML 子句的替代方法从关系数据中生成 XML 文档：

- **XMLAGG 函数** 该集合函数从 XML 元素的集合生成一个 XML 元素林。

请参见“[XMLAGG 函数 \[Aggregate\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **XMLCONCAT 函数** 该函数通过将传入的 XML 值串联起来生成一个 XML 元素林。

请参见“[XMLCONCAT 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **XMLELEMENT 函数** 该函数生成一个 XML 元素，您可以选择为该元素指定元素内容、属性和属性内容。

请参见“[XMLELEMENT 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **XMLFOREST 函数** 该函数生成一个 XML 元素林。

请参见“[XMLFOREST 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **XMLGEN 函数** 该函数基于一个 XQuery 构造函数生成 XML 值。

请参见“XMLGEN 函数 [String]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **数据库中的 HTTP 服务器** Adaptive Server Anywhere 数据库服务器可以充当 Web 服务器，这使您只需使用一个 Adaptive Server Anywhere 数据库和您选择的 Web 浏览器就可以编写和运行基于 Web 的应用程序。

该功能使数据库服务器可以处理标准 HTTP 和 HTTPS 请求以及标准 SOAP 请求。可用的服务类型有 HTTP、HTTPS、XML、RAW、SOAP 和 DISH。DISH 是一个 SOAP 服务处理程序。

对于在此发行版本发布之前创建的数据库，要获得此增强所带来的益处，必须使用升级实用程序升级数据库。

请参见“SQL Anywhere Web 服务”《SQL Anywhere 服务器 - 编程》。

- **索引顾问** 索引顾问是一个协助您正确选择索引的工具。它对一个查询或一组操作进行分析，然后建议应将索引添加到数据库中还是将其从数据库中删除。

对于在此发行版本发布之前创建的数据库，要获得此增强所带来的益处，必须使用升级实用程序升级数据库。

请参见“索引顾问”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **64 位版本可用** 该软件有面向基于 Itanium II 芯片的 Windows Server 2003 的完整 64 位版本，还有面向 64 位 Linux 和 HP-UX 操作系统的部署发行版本。

SQL 增强

- **现在可在 select 语句之前使用 WITH 子句指定公用表表达式** 公用表表达式是只存在于 SELECT 语句作用域之内的临时视图定义。它们可以是递归的，也可以是非递归的。有时，使用公用表表达式可以令您以更整洁的方式编写查询。它们还允许您在单个查询中执行多级聚合。它们只能用在顶级 SELECT 语句、视图定义中的顶级 SELECT 语句或 INSERT 语句中的顶级语句内。

请参见“公用表表达式”《SQL Anywhere 服务器 - SQL 的用法》。

- **现在可以使用特殊格式的公用表表达式执行递归联合** 通过递归公用表表达式，可以编写递归查询。当查询表示分层数据结构或有向图的表时，这些公用表表达式特别有用。每个递归公用表表达式都包含一个先执行的初始子查询和一个递归子查询。对视图的引用必须位于递归子查询的 FROM 子句中，它引用在上次迭代中添加到视图的行。如果您查询的数据结构可能包含循环，则在提供停止递归的条件时必须特别小心。

请参见“递归公用表表达式”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **现在支持 INTERSECT 和 EXCEPT 语句** 这些语句计算两个或更多个结果集之间的交集和差集。它们是 UNION 语句的补充。

有关详细信息，请参见以下内容：

- “使用 EXCEPT 和 INTERSECT”一节《SQL Anywhere 服务器 - SQL 的用法》
- “集合运算符和 NULL”一节《SQL Anywhere 服务器 - SQL 的用法》
- “EXCEPT 子句”一节《SQL Anywhere 服务器 - SQL 参考》
- “INTERSECT 子句”一节《SQL Anywhere 服务器 - SQL 参考》

- **SELECT 语句可对存储过程结果集进行操作** 在 SELECT 语句中，存储过程调用可以出现在允许基表或视图的任何地方。

如果想要存储有关存储过程调用的统计信息，您必须使用升级实用程序升级数据库。如果没有统计信息，则您在尝试连接存储过程调用的结果时可能得不到很好的计划。

请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新增的在线分析处理功能** 受支持的 SQL 语言中添加了几种 OLAP 功能：
 - **ROLLUP 操作** 对于带 GROUP BY 子句的查询，ROLLUP 操作会将小计行添加到结果集中。每个小计行提供 GROUP BY 结果集中一组行的总计。

请参见“使用 ROLLUP”一节《SQL Anywhere 服务器 - SQL 的用法》。
 - **LIST 函数可以提供有序列表** LIST 函数已经得到扩展，现在可以提供经排序的项目列表。

请参见“LIST 函数 [Aggregate]”一节《SQL Anywhere 服务器 - SQL 参考》。
 - **更多集合函数** 增加了用于计算基于样本和基于总体的标准差与方差的函数。

请参见“集合函数”一节《SQL Anywhere 服务器 - SQL 参考》。
- **CREATE INDEX 语句允许对内置函数创建索引** 该功能为在表中新添一个计算列和对该列创建索引提供了一种方便的方法。

请参见“CREATE INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“创建索引”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **所有上下文中都允许使用 ORDER BY 子句** 在以前的版本中，在视图定义、子查询或 UNION 语句中的许多 SELECT 语句都不允许使用 ORDER BY 子句。此限制现在已经取消。

在某些情况下（特别是在与 FIRST 或 TOP 子句结合使用时），使用带有 ORDER BY 子句的 SELECT 确实会影响视图定义或集操作的结果。在其它环境中，也允许使用 ORDER BY 子句，但对操作不产生任何影响。
- **现在 SELECT 语句可以包含 START AT 作为 TOP 子句的一部分** START AT 在显式限制结果集的查询中提供了更多的灵活性。

请参见“SELECT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **现在可以对约束命名** 现在可以为检查约束、唯一约束和参照完整性约束指定名称。这使您可以通过更改单个约束来修改表和列的约束，而不必修改整个表约束。

对于在此版本之前创建的数据库，要获得此增强所带来的益处，必须通过卸载后再重装该数据库来升级数据库文件格式。

请参见“ALTER TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》、“CREATE TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“使用表和列约束”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **横向派生表允许在 FROM 子句中使用外部引用** 现在可以在 FROM 子句中从派生表和存储过程进行外部引用。为表示正在进行外部引用，会使用 LATERAL 关键字。

请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **EXECUTE IMMEDIATE 允许更灵活地处理转义字符** 通过新选项 WITH ESCAPES OFF 可以取消对转义字符的处理。此功能使构建包含文件路径的动态语句变得更加容易。

请参见“EXECUTE IMMEDIATE 语句 [SP]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **EXECUTE IMMEDIATE 支持返回结果集的查询** 这个新功能可使您在存储过程中更加动态地构建语句。

请参见“在过程中使用 EXECUTE IMMEDIATE 语句”一节《SQL Anywhere 服务器 - SQL 的用法》和“EXECUTE IMMEDIATE 语句 [SP]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **CREATE FUNCTION 和 ALTER FUNCTION 现在允许使用 Transact-SQL 语法** 现在，您可以使用 Transact-SQL 方言创建用户定义的函数，以向调用环境返回标量值。

请参见“CREATE FUNCTION 语句 (Web 服务)”一节《SQL Anywhere 服务器 - SQL 参考》。

- **现在可在插入多个行时使用自动增量列的值** 通过对值敏感的游标（即由键集决定的游标）插入行时，新插入的行出现在游标结果集的末尾处。

这一变化的结果是，通过选择游标中的最后一行可以找到最后插入的行的自动增量列的值。例如，在嵌入式 SQL 中，可以使用 [FETCH ABSOLUTE -1 cursor-name] 来获取该值。

请参见“通过游标修改行”一节《SQL Anywhere 服务器 - 编程》。

- **[远程数据访问] 现在可以处理 UUID/GUID 列** [远程数据访问] 现在可以管理 Microsoft SQL Server 的唯一标识符列。

请参见“数据类型转换: Microsoft SQL Server”一节《SQL Anywhere 服务器 - SQL 的用法》和“UNIQUEIDENTIFIERSTR 数据类型”一节《SQL Anywhere 服务器 - SQL 参考》。

- **[远程数据访问] 现在可以命名远程连接** 现在为通过 ODBC 建立的 [远程数据访问] 连接指定了名称，因此可以删除这些连接。

请参见“管理通过 ODBC 执行的远程数据访问连接”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **新增的函数可以返回表达式的数据类型** EXPRTYPE 函数返回表达式的数据类型。

请参见“EXPRTYPE 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **增强的 EXIT 语句** Interactive SQL EXIT 语句现在可以为 Interactive SQL 设置退出代码。

请参见“EXIT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **OUTPUT 语句接受 ASIS 关键字** 指定 ASIS 后，值在写入文件时不进行转义处理。

请参见“OUTPUT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **可以变更索引和外键** 通过 ALTER INDEX 语句，可以对索引和外键进行重命名，还可以将用户创建的索引以及主键或外键索引的索引类型更改为聚簇索引或非聚簇索引。

要在此版本之前创建的数据库上获得聚簇索引的优点，必须通过卸载和重装该数据库来升级数据库文件格式。

请参见“ALTER INDEX 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **查询中允许使用多个不同的集合** 集合函数可以将 [DISTINCT column-name] 作为参数。在该软件的以前版本中，查询中只能使用一个带 DISTINCT 参数的集合函数。现在可以使用多个这样的函数。版本 9 软件中允许使用以下查询，但该查询在 9 之前版本的软件中不允许使用：

```
SELECT count( DISTINCT first_name ),
       count( DISTINCT last_name )
FROM contact
```

- **在事件调度支持的所有语言中都可以识别完整的和简写的日期名** 创建事件时，数据库服务器可以识别以 Adaptive Server Anywhere 支持的任何语言表示的完整日期名和缩写日期名。在以前的版本中，非英语的语言表示的调度要求使用完整日期名。

请参见“[CREATE EVENT 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **隐藏过程文本以保密您的逻辑** 可以使用 SET HIDDEN 选项隐藏存储过程、函数、触发器和视图中包含的逻辑。这样在分发应用程序和数据库时便不会泄漏存储过程、函数、触发器和视图中的逻辑。

对于在此发行版本发布之前创建的数据库，要获得此增强所带来的益处，必须使用升级实用程序升级数据库。

请参见“[隐藏过程、函数、触发器和视图的内容](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

管理和可伸缩性的增强

- **校验实用程序给出了更详细的返回代码** 校验实用程序 (dbvalid) 可给出更具体的返回代码，用于说明发生故障的原因。

请参见“[校验实用程序 \(dbvalid\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **两个新的服务器属性** 增加了两个新的服务器属性。CommandLine 提供了用于启动服务器的行，CompactPlatformVer 提供了精简版本的 PlatformVer 服务器属性。

请参见“[数据库服务器属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **新增 sp_remote_primary_keys 存储过程** 为了获得有关使用远程数据访问的远程表的主键信息，现在新增了一个名为 sp_remote_primary_keys 的存储过程。

对于在此发行版本发布之前创建的数据库，要获得此增强所带来的益处，必须使用升级实用程序升级数据库。

请参见“[sp_remote_primary_keys 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **新增的 connection_property 可返回连接的通信链接的名称** 新增的 CommNetworkLink 连接属性可返回连接的通信链接的名称。

请参见“[连接属性](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **NetWare 现在支持完整的字符集转换** 在 8.x 中，NetWare 支持单字节到单字节的字符转换，但在 9.0 中，NetWare 支持其它平台所支持的所有字符集。

- **卸载实用程序可以卸载列列表** 现在，卸载实用程序 (dbunload) 可以卸载它在 reload.sql 文件中生成的 LOAD TABLE 语句的列列表，这使得对表中的列进行重新排序变得更加容易。

请参见“[卸载实用程序 \(dbunload\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **向 LDAP 注册数据库服务器** 现在，数据库服务器可以在 LDAP 服务器中对自身进行注册，以便客户端和定位实用程序 (dblocate) 可以通过查询 LDAP 服务器来查找它。这可以让运行在 WAN 上或穿过防火墙的客户端查找这些服务器，而无需指定 IP 地址。LDAP 仅与 TCP/IP 一起使用，而且仅在网络服务器上使用。

请参见“使用 LDAP 服务器进行连接”一节《SQL Anywhere 服务器 - 数据库管理》或“LDAP 协议选项 [LDAP]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **增强的处理大量连接的能力** 当连接数超过 200 时，活动超时值现在将自动增加，以更好地处理大量连接。

请参见“-tl 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》和“LivenessTimeout 连接参数 [LTO]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **请求日志过滤与主机变量支持** 现在可以对写入请求日志的输出进行过滤，使其只包含来自特定连接或针对特定数据库的请求。此外，现在还可将主机变量值输出到请求日志中。

请参见“sa_server_option 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》、“提高数据库性能”《SQL Anywhere 服务器 - SQL 的用法》、“sa_get_request_times 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》和“-zr 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **BACKUP 语句和 dbbackup 允许重命名日志副本** 可以使用 BACKUP 语句和备份实用程序 (dbbackup) 来重命名日志副本。

请参见“备份实用程序 (dbbackup)”一节《SQL Anywhere 服务器 - 数据库管理》和“BACKUP 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **START DATABASE 语句允许检查点上的日志截断和只读模式** START DATABASE 语句现在允许数据库启动时启用检查点上的日志截断或以只读模式启动。

请参见“START DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **Adaptive Server Anywhere 支持不同的审计选项** 在以前版本的 Adaptive Server Anywhere 中，您可以选择启用或禁用审计功能。现在，可以指定想要审计的选项。

请参见“sa_disable_auditing_type 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》或“sa_enable_auditing_type 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **可以给 event_parameter 函数传递三个新值** 可以给 event_parameter 函数传递三个新值。ScheduleName 返回触发事件的调度的名称。AppInfo 返回导致事件的连接的 connection_property('AppInfo') 的值。DisconnectReason 返回表示连接终止原因的字符串。

请参见“EVENT_PARAMETER 函数 [System]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新增的服务器属性可确定连接到网络服务器的并发用户的数量** 新增的 LicensesInUse 属性用于确定当前连接到网络服务器的并发用户数量。并发用户数由连接到服务器的唯一客户端网络地址数来确定，而不是由连接数确定。例如，如果有三个客户端计算机连接到服务器，而每个客户端计算机有两个连接，则 select property('LicensesInUse') 返回 '3'。

有关详细信息，请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **创建服务 [dbsvc] 实用程序现在可以启动和停止服务** 创建服务 [dbsvc] 实用程序中增加了两个新选项。Dbsvc -u service_name 用于启动名为 service_name 的服务，dbsvc -x service_name 用于停止名为 service_name 的服务。

请参见“用于 Windows 的服务实用程序 (dbsvc)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **网络服务器支持 LocalOnly 协议选项 [LOCAL]** 可以在服务器上使用 LocalOnly 协议选项 [LOCAL]。将 LocalOnly 协议选项设置为 YES 后运行网络服务器，可以将其作为个人服务器运行，而不受连接或 CPU 限制。

请参见“LocalOnly 协议选项 [LOCAL]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新增使用 Address Windowing Extensions 时的最小数据库服务器高速缓存大小** 现在，在 Windows 2000、Windows XP 和 Windows Server 2003 上使用 Address Windowing Extensions (AWE) 时的最小数据库服务器高速缓存大小为 2 MB。在以前的版本中，使用 AWE 时的最小高速缓存大小为 3 GB-256 MB。

请参见“-cw 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **新增的数据库属性可指定驱动器类型** 新增的 DriveType 数据库属性可提供有关数据库文件所在的驱动器的信息。

请参见“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。

- **Adaptive Server Anywhere NetWare 现在运行更快** 现在，面向 NetWare 的 Adaptive Server Anywhere 服务器使用 LibC 而不是 CLIB。LibC 是一个 C 运行时库，与旧 CLIB 库相比，它可以更好地与 NetWare 操作系统的新内核进行交互。所有 NetWare 客户端软件（如 dblib、dbisql、dbconsole 和 dbremote）仍然使用 CLIB。这样做的优点如下：可以将 NetWare 上的最大文件大小增加到与 NTFS 上的最大文件大小相同；有多个 CPU 时可以使用多个 CPU；允许 TCP 和 SPX 使用 Winsock，这样做可使软件的运行速度快于以前的版本。

请参见“物理限制”《SQL Anywhere 服务器 - 数据库管理》和“9.0 版中的行为更改”一节第 269 页。

- **NetWare 上的外部函数增强** NetWare 上的外部函数或外部存储过程现在可以使用多个 NLM，而不会产生命名冲突。

- **连接可以指定错误消息的语言** 现在，每个到数据库服务器的连接都可以请求使用数据库服务器报告错误消息和各种其它字符串时使用的语言。连接所使用的语言与服务器使用的语言无关。数据库服务器还使用连接请求的语言来解释日期字符串。

- **两个新服务器属性可标识处理器类型** 增加了两个新的服务器级属性。ProcessorArchitecture 用于标识处理器类型，NativeProcessorArchitecture 用于在可以模拟处理器的平台上标识本机处理器类型。

请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。

- **数据库口令是否区分大小写与数据库是否区分大小写无关** 可以通过 CREATE DATABASE 语句、初始化 [dbinit] 实用程序和 [创建数据库向导] 指定口令是否区分大小写。口令的区分大小写设置与用于字符串比较的数据库区分大小写设置无关。新增的 CaseSensitivePasswords 数据库属性用于检查数据库的口令设置是否区分大小写。

请参见“CREATE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》。

性能增强（查询优化）

此处列出的新功是无需用户进行任何操作就可以使用的查询优化增强。它们无需用户干预就可生效。如果您研究一下查询执行计划，可能就会看到这些优化的效果。

这些优化增强不要求对数据库进行升级，但它们在使用版本 9 软件创建的数据库上可获得最大的运行效率。

- **基于成本的子查询优化** 优化程序大大扩展了可以对子查询进行优化的范围。在以前的版本中，子查询要么在语义查询优化期间被重写为连接，要么独立于查询的其余部分单独进行优化。现在，由于过于复杂而无法以连接形式重写的子查询仍然可以作为查询应有的一部分进行优化。
- **缓冲行读取提高了顺序扫描的性能** 从数据库页中读取行以进行顺序表扫描时，Adaptive Server Anywhere 现在可以先将各行复制到缓冲区中，然后再将它们返回给用户。视查询的复杂程度而定，这可能会显著地节约时间。
- **Top N 查询的执行效率更高** 为执行含有 TOP N 子句的查询提供了一种新算法可提高执行速度。请参见“SortTopN 算法 (SrtN)”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **用于确定直方图中保留了哪些频率的新算法** 以前，列直方图会为选择性大于 1% 的值创建单一域桶。现在，创建单一域桶的条件已放宽，改为由直方图尽量使单一域桶的数量保持在最低水平。请参见“优化程序估计值和列统计信息”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **QueryCachedPlans 属性可显示当前高速缓存了多少执行计划** 新属性 QueryCachedPlans 显示当前为给定连接或所有连接高速缓存了多少个查询执行计划。它可以与 QueryCachePages、QueryOptimized、QueryBypassed 和 QueryReused 组合使用，以确定 max_plans_cached 选项的最佳设置。请参见“连接属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **更快地缓存过程语句计划** 将访问计划加入高速缓存的语句的范围得到了扩展，现在将包括存储过程内的查询，这些查询的结果集由存储过程返回到调用环境中。此增强使得不必再对某些语句进行重新优化。请参见“计划高速缓存”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **更新每个索引时维护索引统计信息** 由于更新每个索引时都会维护所有索引（包括那些目录表中的索引）的统计信息，因此可以在几乎没有任何性能开销的情况下为优化程序提供准确的统计信息。统计信息以索引的每项统计信息占一行的形式持久性地保留在 SYSATTRIBUTE 中。对于在此版本之前创建的数据库，要获得此增强所带来的益处，必须通过卸载后再重装该数据库来升级数据库文件格式。请参见“ISYSATTRIBUTE 系统表”一节《SQL Anywhere 服务器 - SQL 参考》。

性能增强（服务器操作）

- **新增性能监控统计信息** 添加了两个新的性能监控统计信息：[通信:使用中的许可] 和 [连接计数]，用户可以通过它们跟踪使用中的连接的数目。请参见“通信统计”一节《SQL Anywhere 服务器 - SQL 的用法》和“杂项统计”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **在 UNLOAD 语句中添加了 APPEND { ON | OFF } 选项** 新增的 APPEND 选项允许将卸载的数据附加到指定文件的末尾。

- **现在可将临时表声明为 NOT TRANSACTIONAL** 使用 NOT TRANSACTIONAL 时，该表不受 COMMIT 或 ROLLBACK 的影响。当不带 COMMIT 对访问表的过程进行重复调用时，此扩充会有帮助。

请参见“CREATE TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》和“DECLARE LOCAL TEMPORARY TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **持久索引统计** 对有关候选索引物理属性的统计信息进行准确的维护有助于优化程序基于开销就使用哪些索引做出决定。统计信息现在被保存在 SYSATTRIBUTE 中，并在每个索引被更新时进行维护。此外，VALIDATE 语句还会校验指定索引的统计信息的准确性，并在其不准确时生成错误消息。这可以为优化程序提供准确的统计信息，而几乎不产生任何性能开销。

对于在此版本之前创建的数据库，要获得此增强所带来的益处，必须通过卸载后再重装该数据库来升级数据库文件格式。

请参见“ISYSATTRIBUTE 系统表”一节《SQL Anywhere 服务器 - SQL 参考》和“VALIDATE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **新增了 optimistic_wait_for_commit 选项** 此选项专用于当事务在主键行之前添加外键行时模拟 5.x 的锁定行为。虽然该选项不具有通用性，但在将 5.x 版应用程序迁移到 8.x 或更高版本应用程序时，该选项会有帮助。

- **新增扩展属性函数** 新增的 DB_EXTENDED_PROPERTY 函数与 DB_PROPERTY 类似，不同的是它还允许指定一个可选的属性特定的字符串参数。

请参见“DB_EXTENDED_PROPERTY 函数 [System]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **两个新增属性** 添加了两个新的属性：FileSize 和 FreePages。这两个属性都可以带一个可选参数，该参数用于指定正请求该属性提供的 dbspace。

请参见“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。

- **增强的服务器安静模式** 服务器的安静模式和错误记录选项得到了增强，现在允许服务器取消各种消息。此外，-qw 选项替代了 -q 选项，-qi 选项替代了 -Q 选项。

开发和管理工具

- **Adaptive Server Anywhere 插件更改** 用于 Sybase Central 的 Adaptive Server Anywhere 插件已进行了重组。以前位于左窗格中的属性窗口、窗口框和文件夹中的很多信息现在在右窗格中的选项卡上提供。例如，若要查看有关外键的信息，您现在可以在左窗格中选择包含该外键的表，然后在右窗格中选择 [外键] 选项卡。在以前的版本中，左窗格中有一个单独的 [外键] 文件夹。

还对插件做了另外几处更改，包括：

- 表编辑器不再是一个单独的窗口。现在您可以在 Sybase Central 的右窗格中直接编辑表。
- 您可以在 Sybase Central 的右窗格中编辑存储过程、函数、触发器和事件，如果您想同时打开多个窗口，也可以在单独的代码编辑器窗口中进行编辑。
- 工具栏按钮现在更改为包含特定于所选对象的选项。
- 现在可以直接在 Sybase Central 主窗口中查看 [SQL 语句] 日志和服务器消息（与数据库服务器消息窗口中显示的信息相同）。要查看此信息，请在 Sybase Central 中选择 [文件] » [服务

器消息和已执行的 SQL]。[服务器消息和已执行的 SQL] 窗格显示在 Sybase Central 主窗口的底部。

- Adaptive Server Anywhere 插件提供了几个新向导，指导您完成各种任务（包括创建表、唯一约束和 Web 服务）。
- **Adaptive Server Anywhere 插件中增强的剪贴板支持** Adaptive Server Anywhere 插件中的剪贴板支持得到了增强，您现在可以将 Sybase Central 内的大部分对象复制并粘贴到其它应用程序（如 Interactive SQL 或文本编辑器）中。当您将对象复制到其它应用程序时，将会根据您选择的对象显示对象名或对象的 SQL。例如，如果在 Sybase Central 中复制某个索引，然后将其粘贴到文本编辑器中，则会出现与该索引相应的 CREATE INDEX 语句。
请参见“在 SQL Anywhere 插件中复制数据库对象”一节《SQL Anywhere 服务器 - 数据库管理》。
- **调试程序更改** 用于对存储过程和 Java 类进行调试的调试程序已集成到 Sybase Central 中。用户界面经过了重新设计。
请参见“调试过程、函数、触发器和事件”《SQL Anywhere 服务器 - SQL 的用法》。
- **Sybase Central、Interactive SQL 和 Adaptive Server Anywhere 控制台实用程序加入了一个用于自动检查软件更新的选项** 可对 Sybase Central、Interactive SQL 和 Adaptive Server Anywhere 控制台实用程序进行配置，使之自动检查是否有软件更新。可在 Interactive SQL 和 Adaptive Server Anywhere 控制台实用程序中的 [选项] 窗口中设置此选项，如果装载了 Adaptive Server Anywhere 插件，也可以从 Sybase Central 的 [帮助] 菜单中进行设置。在以前的发行版本中，您必须访问 Web 站点来获取此信息。
- **对 Adaptive Server Anywhere 控制台实用程序的增强** 已对 Adaptive Server Anywhere 控制台实用程序进行了很多改进，包括界面的更改、对多个连接的支持、排序和拖放。
- **快速启动 Sybase Central 和 Interactive SQL** 在 Windows 上，Sybase Central 和 Interactive SQL 加入了一个快速启动程序，旨在启动 Sybase Central 或 Interactive SQL 时缩短应用程序的启动时间。运行 Adaptive Server Anywhere 9.0.0 会启动两个后台进程：一个 *dbisqlg.exe* 的实例和一个 *scjview.exe* 的实例，它们分别是 Interactive SQL 和 Sybase Central 的快速启动程序进程。用户登录时这两个可执行文件都会启动。
请参见“使用快速启动程序选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Interactive SQL 中的语法突出显示编辑器** 可以使用 Interactive SQL 的 [选项] 窗口配置在 Interactive SQL 的 [SQL 语句] 窗格中键入的语法的外观。
- **从 Interactive SQL 中打印** 可以在 Interactive SQL 中打印 [SQL 语句] 窗格中的内容和图形式计划的内容。
请参见“浏览 Interactive SQL”一节《SQL Anywhere 服务器 - 数据库管理》。
- **图形式计划增强** 图形式计划的显示从以下几方面进行了增强：
 - 从一个运算符传给另一个运算符的行数由可变的行的粗细来指明。
 - 速度较慢的操作由红色边线突出显示。
 - 对统计显示进行了扩展和重组。
 - 现在可以打印访问计划。

- **数据库实用程序可接受 @filename 参数** 现在，除 Interactive SQL (dbisql)、语言选择实用程序 (dblank) 和 Adaptive Server Anywhere 控制台实用程序 (dbconsole) 之外的所有数据库管理实用程序都接受包含在文件内、使用 @file 语法的参数。文件名可以出现在配置行中的任何位置，文件中包含的参数将插入到该位置。可以指定多个文件，而且文件指定符可以与命令行选项一起使用。请注意，@file 语法不能递归使用。

请参见“@data 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **在 Interactive SQL 中，可在结果旁显示行号** Interactive SQL 有一个可在结果旁显示行号的选项。可在 Interactive SQL 的 [选项] 窗口的 [结果] 选项卡上设置此选项。

- **可将 Interactive SQL 设置为 .SQL 文件的缺省编辑器** 在 Windows 平台上，可以为 .SQL 文件创建一个文件关联，这样在双击该文件时就会使用 Interactive SQL 来将其打开。

请参见“Interactive SQL 实用程序 (dbisql)”一节《SQL Anywhere 服务器 - 数据库管理》。

- **Interactive SQL [命令历史记录] 窗口的增强** 现在可以在 Interactive SQL 的 [命令历史记录] 窗口中复制和删除命令，也可以在该窗口中选择多个命令。现在，命令历史记录会在 Interactive SQL 会话期间持续存在。

请参见“打印 SQL 语句、执行计划和结果集”一节《SQL Anywhere 服务器 - 数据库管理》。

- **警告消息现在具有 W 前缀** 在版本 9.0 之前，所有警告和错误消息的前缀均为 I 或 E。现在，警告消息的前缀为 W。此更改影响 dbmlsrv9、dbmlsync、dbremote、ssremote、dbltm 和 ssqueue。

MobiLink 的新功能

以下列出了 9.0.0 版中引入的对软件的更改和补充。

- **服务器启动的同步** 可通过服务器启动的同步，从统一数据库中启动 MobiLink 同步。这意味着您可以将数据更新推送到远程数据库。MobiLink 组件（通告程序）提供可编程的选项，用于确定统一数据库中的哪些更改将启动同步，以及如何选择远程设备接收更新消息。远程组件（监听器）决定远程设备的响应方式。

请参见“服务器启动的同步简介”一节《MobiLink - 服务器启动的同步》。

- **基于文件的下载** 下载现在可以作为文件进行处理，可以通过任何分发文件的方式（如电子邮件、ftp、磁盘或多路广播文件分发）对其进行分发。在此发行版本中，此功能只能用于 Adaptive Server Anywhere 远程数据库。

请参见“MobiLink 基于文件的下载”一节《MobiLink - 服务器管理》。

MobiLink 服务器的增强

- **新增连接脚本 begin_publication 和 end_publication** 添加了两个新的脚本。它们的用途之一是实现基于文件的下载。

请参见“begin_publication 连接事件”一节《MobiLink - 服务器管理》和“end_publication 连接事件”一节《MobiLink - 服务器管理》。

- **新增连接脚本 authenticate_parameters** 添加了一个支持自定义验证的新脚本。验证期间将在调用 begin_synchronization 脚本之前调用这个新脚本。

请参见“authenticate_parameters 连接事件”一节《MobiLink - 服务器管理》。

- **新增的选项可删除字符串的填补空白** 对于 VARCHAR 或 LONG VARCHAR 类型的列，dbmlsrv9 -b 选项在同步过程中删除字符串的尾随空白。
请参见“-b 选项”一节《MobiLink - 服务器管理》。
- **选项打开带有 .old 扩展的新日志文件** dbmlsrv9 -on 选项允许您为 MobiLink 服务器日志使用的磁盘空间大小设置一个硬性限制。
请参见“-on 选项”一节《MobiLink - 服务器管理》。
- **记录进度偏移** MobiLink 服务器现在可以报告进度偏移、上次上载时间和上次下载时间。要获得此信息，请使用 dbmlsrv9 选项 -vp 或 -v+。
请参见“-v 选项”一节《MobiLink - 服务器管理》。
- **在 .NET 和 Java 同步逻辑中处理错误和警告** 您现在可以添加逻辑来处理 MobiLink 服务器上出现的错误和警告。
请参见“使用 Java 处理 MobiLink 服务器错误”一节《MobiLink - 服务器管理》和“使用 .NET 处理 MobiLink 服务器错误”一节《MobiLink - 服务器管理》。
- **MobiLink 系统表的新功能** ml_user 和 ml_subscription 表都添加了两个新列。它们分别是 last_upload_time 和 last_download_time。缺省设置是 NOT NULL，并且缺省时间为 January 1, 1900 00:00:00。
此外，ml_subscription 表中还添加了一个 subscription_id 列。publication_name 列现在包含发布名。
请参见“ml_user”一节《MobiLink - 服务器管理》和“ml_subscription”一节《MobiLink - 服务器管理》。

Adaptive Server Anywhere 客户端增强

- **仅上载同步** 您现在可以选择执行仅上载同步。
请参见 dbmlsync “-uo 选项”一节《MobiLink - 客户端管理》。
- **仅下载同步** 您现在可以选择执行仅下载同步。
请参见“-ds 选项”一节《MobiLink - 客户端管理》和“DownloadOnly (ds) 扩展选项”一节《MobiLink - 客户端管理》。
- **窗口消息可以启动同步** 现在，可以通过将窗口消息注册为 dbas_synchronize 并将其发送到 dbmlsync 顶级窗口，来唤醒 dbmlsync 并执行同步。
- **启动时装载 Dll（适用于 Windows CE）** 新增的 dbmlsync 选项 -pd 指定应在启动时装载的 DLL。在 Windows CE 上使用 dbmlsync 的每个用户都应使用此选项。
请参见“-pd 选项”一节《MobiLink - 客户端管理》。
- **升级或修订模式的新方法** 增加了挂接 sp_hook_dbmlsync_schema_upgrade 存储过程来替代 dbmlsync 选项 -i 和扩展选项 SiteScriptName (sn)。
请参见“sp_hook_dbmlsync_schema_upgrade”一节《MobiLink - 客户端管理》。

- **MobiLink 退出代码** 为帮助您跟踪和记录同步的成功与失败（特别是当您在 dbmlsync 会话中运行多个同步时），新增了一个客户端事件挂接过程 `sp_hook_dbmlsync_process_exit_code`。此外，`#hook_dict` 表中还为 `sp_hook_dbmlsync_abort` 挂接设置了一个新值（退出代码）。

请参见“[sp_hook_dbmlsync_process_exit_code](#)”一节《MobiLink - 客户端管理》和“[sp_hook_dbmlsync_abort](#)”一节《MobiLink - 客户端管理》。

- **调度的增强** 如果指定了调度，那么您可以通过使用新的扩展选项 `HoverRescanThreshold (hrt)` 或新的挂接 `sp_hook_dbmlsync_log_rescan` 来减少花费在扫描日志上的时间。

请参见“[HoverRescanThreshold \(hrt\) 扩展选项](#)”一节《MobiLink - 客户端管理》和“[sp_hook_dbmlsync_log_rescan](#)”一节《MobiLink - 客户端管理》。

非英语的语言现在支持在调度中使用缩写的日期名。在以前的版本中，非英语的语言表示的调度要求使用完整日期名。

调度语法中增加了两个新的关键字：**INFINITE** 指示 dbmlsync 无限等待下一个同步的信号，**0**（作为月中的一天）指定月的最后一天。

请参见“[Schedule \(sch\) 扩展选项](#)”一节《MobiLink - 客户端管理》。

UltraLite 客户端增强

- **HotSync 管道附加故障排除帮助** 您可以对 HotSync 管道进行配置，使之将故障排除信息记录到 HotSync 日志中。

性能监控器增强

- **dbmlsync 在没有模式更改时可提供更佳的性能** 缺省情况下，Dbmlsync 在每次同步前不再装载模式信息。在速度较慢的手持式设备上，这通常会使用同步时间缩短 20 秒。
请参见“[-sc 选项](#)”一节《MobiLink - 客户端管理》。
- **Windows CE 上更佳的 dbmlsync 性能** 在 Windows CE 上，Dbmlsync 不再使用 `dbtool9.dll`。这意味着它使用的内存会减少。
- **MobiLink 监控器命令行选项** 现在可以从命令行使用各种选项启动 MobiLink 监控器。
请参见“[启动 MobiLink 监控器](#)”一节《MobiLink - 服务器管理》。
- **重定向器的增强** 新增的参数 `LOG_LEVEL` 允许您控制详细级别。
请参见“[配置重定向器属性（适用于支持服务器组的重定向器）](#)”一节《MobiLink - 服务器管理》。
- **改进的活动** 通过 TCP/IP 进行连接时，可以更快速地检测到删除的连接。这样就可以在连接被删除时更快速地释放 MobiLink 工作线程，从而提高吞吐量。

杂项

- **警告消息现在具有 W 前缀** 在 9.0 之前的版本中，所有警告和错误消息的前缀均为 I 或 E。现在，警告消息的前缀为 W。此更改会影响 `dbmlsrv9`、`dbmlsync`、`dbremote`、`ssremote`、`dbltm` 和 `ssqueue`。

SQL Remote 的新功能

SQL Remote 9.0.0 版包括下列新功能。

- **警告消息现在具有 W 前缀** 在 9.0 之前的版本中，所有警告和错误消息的前缀均为 I 或 E。现在，警告消息的前缀为 W。此更改会影响 dbmlsrv9、dbmlsync、dbremote、ssremote、dbltm 和 ssqueue。

UltraLite 的新功能

UltraLite 开发可以使用两种编程接口实现：

- **UltraLite 组件** UltraLite 组件为使用快速应用程序开发工具的用户提供了 UltraLite 数据库和同步功能。它们为每种受支持的开发工具都提供一个熟悉的接口。UltraLite 组件提供了一个简单的、基于表的数据访问接口，还为更复杂的查询提供了动态 SQL。

UltraLite 组件是在 8.0.2 版中引入的。

- **静态开发模型** 嵌入式 SQL、静态 C++ API 和静态 Java API 仍然可用。现在，在文档中将它们称为静态接口，以区别于组件。

请特别注意以下所列各项：

- **Native UltraLite for Java** 是一个 UltraLite 组件，使用 C/C++ UltraLite 运行时。UltraLite **静态 Java API** 是一个纯 Java 解决方案（在以前的发行版本中提供），在该解决方案中查询必须在编译时进行指定。
- **UltraLite for C++** 是一个组件接口。UltraLite **静态 C++ API** 是一个静态接口（在以前的发行版本中提供），使用该接口时查询必须在编译时进行指定。
- **Embedded SQL** 是一个静态接口，使用该接口时查询必须在编译时进行指定。

以下列出了 9.0 版中引入的对软件的更改和补充。

- **新组件** 除了用于 AppForge MobileVB、eMbedded Visual Basic 和 Java 的组件之外，还引入了以下组件：
 - **UltraLite .NET** 一个针对使用 Visual Studio .NET 环境的开发而提供的组件。可以将使用此组件构建的应用程序部署到支持 .NET Compact Framework（1.05.0000 或更高版本）的设备上。
请参见“[UltraLite.NET 简介](#)”《[UltraLite - .NET 编程](#)》。
 - **C++ 组件** 一个为使用 C++ 编译器进行开发而提供的组件。
请参见“[UltraLite for C/C++ 开发人员](#)”《[UltraLite - C 及 C++ 编程](#)》。
- **Pocket IE 支持** eMbedded Visual Basic 组件已升级为 ActiveX 组件。添加了对使用 JScript 进行开发的支持，用于构建从 Windows CE 设备上的 Pocket IE 运行的应用程序。
- **动态 SQL** 除了 8.0.2 版中提供的基于表的数据访问接口之外，UltraLite 组件现在可以使用动态 SQL 构建更复杂的查询（例如多表连接）。

- **连接参数** UltraLite 组件（C++ 除外）的连接参数现在作为单独的属性而不是一个字符串进行公开。这种设计使调试连接问题变得更加容易，并使连接管理变得更加简明。
请参见“[UltraLite 连接参数](#)” 《[UltraLite - 数据库管理和参考](#)》。
- **拖放 MobileVB 组件** 现在可以将 MobileVB 组件拖动到表单上。可以在设计环境中以及代码中设置组件的属性。
- **多进程访问** C++ 组件支持从多个进程进行访问。要使用此模型开发应用程序，必须将单独的 UltraLite 数据库引擎和应用程序链接到不同的 UltraLite 运行时库。
请参见“[编译和链接应用程序](#)”一节 《[UltraLite - C 及 C++ 编程](#)》。
- **并发同步** 在以前的发行版本中，同步期间会禁止所有对数据的访问。现在，在同步的下载阶段允许对数据进行完全访问。上载阶段期间提供了只读访问。
请参见“[UltraLite 中的并发](#)”一节 《[UltraLite - 数据库管理和参考](#)》。
- **Palm OS 增强** 在 Palm OS 上，UltraLite 代码的结构经过了重组，以更好地利用 Palm 数据库段。
- **扩展的错误信息** 使用 UltraLite 组件构建的应用程序有更多可用的错误信息。
- **Unicode 库在 Windows NT/2000/XP 上可用** 为嵌入式 SQL 和静态 C++ API 应用程序提供了一个 Unicode 版本的 UltraLite 运行时库。此版本可供 UltraLite 组件使用。使用此库时，UltraLite 数据库文件兼容 Windows CE 和桌面操作系统。
- **支持 Windows XP 作为部署平台** Windows XP 上现在支持 UltraLite 应用程序部署。

9.0 版中的行为更改

本节列出了在 SQL Anywhere Studio 9.0 版的组件中引入的行为更改。

Adaptive Server Anywhere 的行为更改

下面是对该软件以前版本的行为更改列表。

- **不支持数据库中的 Java 对象** 不再支持将数据作为 Java 对象进行存储。保留了对 Java 存储过程的支持。
请参见“SQL Anywhere 中的 Java 支持”《SQL Anywhere 服务器 - 编程》。
- **为 Windows 环境新增希腊语归类** 以前版本中提供了 OEM/DOS 字符集的希腊语归类，但现在又添加了一个针对 Windows 的新的希腊语归类 1253ELL。在希腊语 Windows 环境中创建新数据库时，如果未指定归类，系统会自动选择 1253ELL。
请参见“支持的归类和替代归类”一节《SQL Anywhere 服务器 - 数据库管理》。
- **新的连接限制** 数据库服务器现在允许在连接限制之外存在一个额外的 DBA 连接，以允许 DBA 在发生有意的或意外的拒绝服务时进行连接并删除其它连接。
请参见“-gm 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。
- **将个人数据库服务器限制为使用一个处理器** 在以前版本的软件中，个人数据库服务器最多可以使用两个 CPU 来处理请求。现在，个人服务器能够使用的处理器数量限制为一个。
- **FROM 子句中的前置表表达式引用现在可以在嵌套外连接的 ON 子句中使用** 在以前的发行版本中，允许在 ON 短语中使用外部引用。此类外部引用现在必须使用 LATERAL 关键字予以指明。此限制强制要求进行明确，它符合 SQL/99 标准。

下面的示例查询将不再有效，因为它包含不带 LATERAL 关键字的外部引用（突出显示部分）。

```
SELECT *
FROM T1,
      T2 LEFT OUTER JOIN
      ( T3 LEFT OUTER JOIN T4 ON T1.col1 = T2.col2 )
      ON T1.col2 = T2.col2
```

请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **有多个匹配项的非限定表引用被报告为语法错误** 在以前的发行版本中，如果查询包含未指定所有者名的表引用（即非限定表引用），而该表上可能存在多个匹配项，则会使用第一个找到的匹配项。现在，非限定的表引用会导致错误。
请参见“表名 '%1' 不明确”一节《错误消息》。
- **带有 NULL 转义字符的 LIKE 运算符现在计算为 NULL** 现在，包含 NULL 转义字符的 LIKE 谓语句计算为 NULL。以前，带有 NULL 转义字符的 LIKE 谓语句按没有转义字符进行计算。新行为符合 ISO/ANSI 规范。

- **删除的属性和统计信息** 删除了数据库属性 ServerIdleWaits、连接属性 TaskSwitch 和 CurrTaskSwitch，一并删除的还有它们相应的性能监控统计信息：[上下文开关]、[服务器空闲等待/秒] 和 [请求队列等待/秒]。
- **对 INSERT/UPDATE/DELETE 更新列统计信息** 现在，如果执行的 INSERT、UPDATE 或 DELETE 语句导致大量数据被更改，统计信息将被更新。
- **恢复期间不再更新统计信息** 恢复期间或执行简单的 DELETE 和 UPDATE 语句时，服务器不再更新统计信息。简单语句是指那些不进行优化便由服务器直接执行的语句。
- **直方图范围显示为正确的数据类型** 以前，sa_get_histogram() 系统过程和直方图 [dbhist] 实用程序将输出的范围显示为散列值。现在，输出的直方图范围与相应列中的数据相匹配，并以正确的数据类型进行显示。
请参见“直方图实用程序 (dbhist)”一节《SQL Anywhere 服务器 - 数据库管理》和“sa_get_histogram 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **每个远程数据库只允许有一个统一用户** 不能再为同一远程数据库定义多个统一用户。
请参见“GRANT CONSOLIDATE 语句 [SQL Remote]”一节《SQL Anywhere 服务器 - SQL 参考》或“REVOKE CONSOLIDATE 语句 [SQL Remote]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **如果不显式指定，CommLinks 连接参数将使用共享内存** 现在，没有指定 CommLinks 连接参数的连接总是尝试通过共享内存进行连接。
请参见“CommLinks 连接参数 [LINKS]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **CommLink 连接参数总是先尝试使用共享内存协议** 如果指定 CommLinks=all，则 Adaptive Server Anywhere 总是先尝试使用共享内存协议进行连接，然后再尝试使用其它协议。
请参见“CommLinks 连接参数 [LINKS]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **连接错误中断进程** 以前，系统会逐一尝试 CommLink 连接参数中列出的连接协议，直到连接成功。现在，如果在尝试过程中发生了连接错误，系统会立即中止连接进程，无论是否已尝试过列出的所有协议。
请参见“CommLinks 连接参数 [LINKS]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **prevent_article_pkey_update 的缺省值发生了变化** prevent_article_pkey_update 数据库选项的缺省值已更改为 On，以反映应避免更新主键值这一事实。这一新的缺省设置禁止对作为发布的一部分的主键进行更新。可以通过将该值设置为 OFF 来禁用此功能。
请参见“prevent_article_pkey_update 选项 [数据库] [MobiLink 客户端]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **有些函数被当作非确定型函数处理** RAND、NEWID 和 GET_IDENTITY 函数被视为非确定型函数。结果，查询执行期间将不对这些函数进行高速缓存。
有关详细信息，请参见“函数高速缓存”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **性能消息现在显示数据库名称** 引擎性能提示消息现在显示数据库名。这在运行多个数据库时特别有用。同样，以注意开头的消息也表示它们是提示消息。
- **使用 9.0.0 之前版本 Adaptive Server Anywhere 的 NetWare 客户端需要进行升级** Adaptive Server Anywhere 中对 NetWare 的支持得到增强的结果是，使用 9.0.0 以前版本 Adaptive Server

Anywhere 的 NetWare 客户端只有在安装了特定 EBF 后才能使用共享内存连接到 9.0.0 服务器。内部版本号是：7.0.4.3400、8.0.0.2358、8.0.1.3088 和 8.0.2.4095。内部版本号小于上述版本号的客户端根本找不到 9.x 服务器。

- **ALTER DATABASE CALIBRATE 语法的更改** ALTER DATABASE CALIBRATE TEMPORARY DBSPACE 的语法更改为 ALTER DATABASE CALIBRATE DBSPACE TEMPORARY，以使该语法与其它类似语句相一致。

请参见“ALTER DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **对动态高速缓存大小的调整更有力度** 现在，在启动新数据库后或在文件显著增大时可以更积极地调整动态高速缓存的大小。在做出此项更改之前，最快也要每分钟才能执行一次对统计信息的抽样和对高速缓存大小的调整操作。现在，在启动数据库后或在文件显著增大时，在三十秒内每五秒即可执行一次统计信息抽样和高速缓存大小调整操作。

请参见“使用高速缓存提高性能”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **确定接口和消息的语言** 新增了两个环境变量（ASLANG 和 ASCHARSET），它们控制界面（如 Sybase Central 或 Interactive SQL）和消息中使用的语言。ASLANG 指定语言，ASCHARSET 指定字符集。

请参见“SALANG 环境变量”一节《SQL Anywhere 服务器 - 数据库管理》或“SACHARSET 环境变量”一节《SQL Anywhere 服务器 - 数据库管理》。

- **Rowcount 设置现在限制返回的行** 现在，rowcount 设置会限制游标从顶部返回的行。不再可能使用绝对读取定位到结果的开始位置。

如果需要，您可以使用新功能 TOP N / START AT 来模拟此行为。

请参见“SortTopN 算法 (SrtN)”一节《SQL Anywhere 服务器 - SQL 的用法》。

不建议使用和不支持的功能

不再支持 Adaptive Server Anywhere -d 服务器选项 Adaptive Server Anywhere 中对 NetWare 的支持得到增强的结果是，现已不再支持 -d 服务器选项。

不再支持 NetWare 4.x Adaptive Server Anywhere 中对 NetWare 的支持得到增强的结果是，Adaptive Server Anywhere 将只能在 NetWare 5.1 版 SP6（或更高版本）或 6.0 版 SP3（或更高版本）上运行。必须安装正确的服务包，否则 Adaptive Server Anywhere 服务器会显示一条错误消息。

不再支持 SQLLOCALE 环境变量 SQLLOCALE 环境变量已被两个新环境变量 ASLANG 和 ASCHARSET 替代。

请参见“9.0 版中的行为更改”一节第 269 页。

MobiLink 的行为更改

以下是对该软件先前版本的行为更改列表。

- **不再支持 dbmlsync 选项 -i 和扩展选项 SiteScriptName** 不再支持 dbmlsync -i 和 dbmlsync -e sc。它们已被名为 sp_hook_dbmlsync_schema_upgrade 的新挂接所替换。

请参见“sp_hook_dbmlsync_schema_upgrade”一节《MobiLink - 客户端管理》。

- **下载确认现在缺省为 OFF** 如果是 Adaptive Server Anywhere 远程数据库，SendDownloadAck 扩展选项现在的缺省值为 OFF。如果是 UltraLite 远程数据库，ul_synch_info 结构的 send_download_ack 字段现在的缺省值是 ul_false。

升级到版本 9 时，如果应用程序必须知道在提交下载事务之前远程数据库是否已应用了下载，则必须显式地将此选项设置为 On。请参见：

- “SendDownloadACK (sa) 扩展选项” 一节 《MobiLink - 客户端管理》
- “发送下载确认同步参数” 一节 《UltraLite - 数据库管理和参考》

- **有些 dbmsync 挂接在缺省情况下可能无法在 Windows CE 设备上工作** dbmsync 扩展选项 LockTables 已进行了修改，现在可以通过它指定是以共享模式还是独占模式来锁定表。LockTables 的缺省设置是 ON，此设置继续为除 Windows CE 之外的所有平台以共享模式锁定表。但在 Windows CE 设备上，ON 现在意味着以排它模式锁定表。此项更改显著地增强了 Windows CE 应用程序的性能。

dbmsync 事件挂接 sp_hook_dbmsync_download_com_error、sp_hook_dbmsync_download_fatal_sql_error 和 sp_hook_dbmsync_download_log_ri_violation 都在不同的连接上执行。如果它们尝试访问以排它模式锁定的任何同步表，则无法正确执行。如果您的部署在 Windows CE 上使用这些挂接中的任何一个，那么您可能需要将 LockTables 设置为 SHARE。请参见：

- “LockTables (lt) 扩展选项” 一节 《MobiLink - 客户端管理》
- “sp_hook_dbmsync_download_com_error (不建议使用)” 一节 《MobiLink - 客户端管理》
- “sp_hook_dbmsync_download_fatal_sql_error (不建议使用)” 一节 《MobiLink - 客户端管理》
- “sp_hook_dbmsync_download_log_ri_violation” 一节 《MobiLink - 客户端管理》

- **MobiLink 服务器错误代码** MobiLink 服务器现在会提供更多有关错误的信息。所有 MobiLink 服务器错误代码都小于 -10000，从 -10001 开始。对于 dbmsync，错误显示在 GUI 和输出文件中。对于 UltraLite，错误以字符串形式显示在 ul_synch_info 结构中。

请参见 “MobiLink 服务器错误消息” 《错误消息》。

- **不再支持上载光标** 不建议使用以下脚本：upload_cursor、new_row_cursor 和 old_row_cursor。您应该为上载流使用基于语句的脚本。

请参见 “编写用于上载行的脚本” 一节 《MobiLink - 服务器管理》。

- **不建议使用 -zac 和 -zec** 已不建议使用用于生成基于游标的脚本的 MobiLink 服务器选项 -zac 和 -zec。

- **删除 -zd** MobiLink 服务器选项 -zd 已被删除，因为它会导致最后才传递 last_download 时间戳。现在始终首先传递此参数。

- **不建议使用 mlxtract** 不建议使用 mlxtract 实用程序。

请参见 “创建远程数据库” 一节 《MobiLink - 客户端管理》。

- **总是调用 end_synchronization 脚本** 在 9.0 版之前，同步失败时可能不会调用 end_synchronization 脚本。现在，只要调用了 begin_synchronization 脚本，就会调用 end_synchronization 脚本。这意味着无论同步成功与否，您放置在 end_synchronization 脚本中的所有清理活动都会得到执行。

此外，`end_synchronization` 脚本还有一个新参数 `sync_ok`，它指示同步已成功 (1) 还是已失败 (0)。

请参见“[end_synchronization 连接事件](#)”一节《MobiLink - 服务器管理》和“[end_synchronization 表事件](#)”一节《MobiLink - 服务器管理》。

- **流 DII 和共享对象被重命名** 流 DII 和共享对象的名称进行了更改，以改善与 Adaptive Server Anywhere 的一致性。下表详细列出了这些更改：

旧名称	新名称
dbhttp9	dbmlhttp9
dbhttps9	dbmlhttps9
dbjrsa9	dbmljrsa9
dbjtls9	dbmljtls9
dbrsa9	dbmlrsa9
dbsock9	dbmlsock9
dbtls9	dbmltls9

请参见“[部署 MobiLink 应用程序](#)”一节《MobiLink - 服务器管理》。

- **不再支持 ScoutSync** 不再支持 ScoutSync。
- **每次同步时不再重装模式信息** 在 9.0 版之前，`dbmlsync` 在每次同步前重新从数据库中装载模式信息。现在只在 `dbmlsync` 启动时重装模式信息。可以使用 `dbmlsync -sc` 选项恢复为原来的行为。如果不使用 `-sc`，则在对远程数据库进行任何更改前应先关闭 `dbmlsync`。不关闭 `dbmlsync` 就更改模式可能会导致发生同步错误或出现其它意外行为。

请参见“[-sc 选项](#)”一节《MobiLink - 客户端管理》。

- **如果丢失关键脚本，同步现在会中断** 在 9.0 版之前，即使缺少某些可能会导致数据丢失的脚本，同步仍会继续。现在 MobiLink 会在这种情况下中断。您可以使用 `dbmlsrv9 -fr` 选项指定生成错误而不是使其失败。

请参见“[-fr 选项](#)”一节《MobiLink - 服务器管理》。

- **删除了 keep_alive 同步参数** 现在，TCP/IP 和 HTTP 协议的 `keep_alive` 同步参数已不再有效；实际上该参数现在始终设置为 ON。这是以前的缺省设置。若要控制 TCP/IP 连接的活动，您可以使用 `liveness_timeout` 参数。

请参见“[CommunicationAddress \(adr\) 扩展选项](#)”一节《MobiLink - 客户端管理》或“[-x 选项](#)”一节《MobiLink - 服务器管理》中的 `liveness_timeout` 参数。

UltraLite 的行为更改

以下是对该软件先前版本的行为更改列表。

- **支持的平台的更改** 对支持的 UltraLite 部署平台进行了以下更改。
 - **不再支持 ScoutSync** 取消了对 ScoutSync 同步软件的支持。
 - **不再支持 VxWork** 不再支持 VxWorks 操作系统
 - **纯 Java UltraLite 需要 JDK 1.1.8** 纯 Java 静态开发模型需要 JDK 1.1.8 或更高版本，而不是 JDK 1.1.4 或更高版本。
 - **Palm OS 更改** 对面向 Palm OS 的 UltraLite 体系结构的更改使得在较新设备上的性能得到了提升。这使得 UltraLite 与以前的版本相比需要更多的动态内存。除非数据库非常小，否则建议使用 Palm OS 3.5 或更高版本，并至少具有 4 MB 内存。
 - **不再支持 MobileBuilder 和 PRC 工具** PenRight 不再支持 UltraLite 开发！MobileBuilder 平台的 UltraLite 开发。也不支持使用 GNU PRC Tool 套件进行的开发。
- **开发平台更改** 现在只支持在 Windows NT/2000/XP 上进行 UltraLite 组件的应用程序开发。另外还在 Windows 98 SE 上支持使用静态接口进行开发。不支持将 Windows 95/98/Me 系列的其它操作系统作开发用途。

现在支持的 Metrowerks CodeWarrior 版本是 8 和 9。

- **文档术语更改** 引入 UltraLite 组件需要使用新的名称以区分不同的接口。较旧的 UltraLite 接口（嵌入式 SQL、C++ API 和 Java API）现在命名为**静态接口**，因为它们使用的查询必须在编译时进行指定。通过这些组件可实现对**动态 SQL** 的访问。
- **Windows NT/2000/XP 上的 UltraLite 运行时库** 现在，ActiveX 和 MobileVB 组件在 Windows 上使用 Unicode 运行时库。此运行时库兼容面向 Windows 的 8.0.2 版 UltraLite 数据库 (.udb) 文件，但不兼容在其它 Windows 操作系统上构建的 8.0.2 版 UltraLite 数据库文件。
- **file_name 参数** 在以前版本的软件中，如果未提供平台特定的参数，则用于在桌面上指定 UltraLite 数据库文件名的 file_name 参数还会用于在设备上指定文件名。现在 file_name 参数将被忽略，但在桌面操作系统上除外。
- **静态 Java API 更改** 静态 Java API 已经更改。原来在 JdbcDatabase 对象上的以下方法已移动到 JdbcConnection 对象上：

- countUploadRows
- getLastDownloadTimeDate
- getLastDownloadTimeLong

在 JdbcConnection 中添加了授予和撤消方法，以供没有显式 JdbcManager 对象的应用程序使用。

- **错误代码更改** 有些 UltraLite 错误代码已更改为更为具体且有用的值。如果您在应用程序中对个别错误代码进行测试，请在升级后检查这些新代码。

例如，如果您在连接到数据库时检查 SQLE_DATABASE_NOT_FOUND（或它在一个 UltraLite 接口中的等价项），则应将它更改为 SQLE_ULTRALITE_DATABASE_NOT_FOUND。

有关错误代码的列表，请参见您正在使用的接口中的 SQL 错误对象。

- **为嵌入式 SQL 所做的 UL_STORE_PARMS 更改** 现在将在 EXEC SQL CONNECT 语句执行过程中对 UL_STORE_PARMS 宏进行计算。数据库不再在 dbinit 调用期间启动，而是在连接时启

动。这意味着，如果使用了多个连接，UL_STORE_PARMS 被计算的次数可能不同。这还意味着必须先定义 UL_STORE_PARMS，然后再定义任何 EXEC SQL CONNECT 语句。

8.0.2 版中的新功能

目录

8.0.2 版中的新功能	278
8.0.2 版的行为更改	286

8.0.2 版中的新功能

本节列出在 SQL Anywhere Studio 8.0.2 版的组件中引入的新功能。

Adaptive Server Anywhere 的新功能

本节介绍 Adaptive Server Anywhere 8.0.2 版中的新功能。它提供主要和次要新功能的详尽列表，并提供交叉引用指向详细叙述各功能的章节。

主要新功能

- **聚簇索引支持** 在表上创建聚簇索引将使该表中行的存储顺序几乎与它们在索引中出现的顺序相同。可以使用 LOAD TABLE 语句以聚簇顺序将信息装载到表中。随着将信息不断插入表中，该表的聚簇特性将逐渐下降。可以使用 REORGANIZE TABLE 语句重新建立聚簇顺序。聚簇索引可以提高性能。

要在此版本之前创建的数据库上使用聚簇索引，必须通过卸载和重装数据库来升级数据库文件格式。

有关详细信息，请参见“使用聚簇索引”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **唯一标识符支持** Adaptive Server Anywhere 支持唯一标识符（UUID 和 GUID）。UUID（通用唯一标识符）和 GUID（全局唯一标识符）是一种唯一地标识行的机制，甚至可以跨同步环境中不同的数据库。

有关详细信息，请参见“NEWID 缺省值”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **使用 ON EXISTING 子句更新现有行** 对于具有主键的表，可以使用 INSERT 语句的 ON EXISTING 子句用新值更新现有行。

有关详细信息，请参见“使用 INSERT 更改数据”一节《SQL Anywhere 服务器 - SQL 的用法》或“INSERT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **Windows CE 上支持 BACKUP 语句** Adaptive Server Anywhere 允许您创建在 Windows CE 平台上运行的数据库的映像备份，或者重命名或截断该数据库的事务日志。

有关详细信息，请参见“备份类型”一节《SQL Anywhere 服务器 - 数据库管理》或“BACKUP 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **图形式计划增强** 增强了图形式计划，加入了详细信息，使其界面焕然一新。

有关详细信息，请参见“读取图形式计划”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **现在显式使用工作表** 工作表的使用现在已推迟到计划中尽可能晚的阶段。使用工作表时，它们现在显式出现在图形式计划中。

有关详细信息，请参见“读取图形式计划”一节《SQL Anywhere 服务器 - SQL 的用法》或“在查询处理中使用工作表（使用 All-rows 优化目标）”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **添加了新的连接** 此版本中添加了新的连接，包括嵌套循环半连接、嵌套循环反半连接、散列半连接和散列反半连接。

有关详细信息，请参见“连接算法”一节《SQL Anywhere 服务器 - SQL 的用法》。

函数增强

- **获取特定游标类型的 SQL 查询的计划** 现在可以使用 PLAN、EXPLANATION 和 GRAPHICAL_PLAN 函数，根据 SQL 查询的游标类型获取其计划。
有关详细信息，请参见“GRAPHICAL_PLAN 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》、“EXPLANATION 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》或“PLAN 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **字符集转换函数** 提供了一个新的 CSCONVERT 函数，用于在字符集之间转换字符串。
有关详细信息，请参见“CSCONVERT 函数 [String]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **变量测试函数** 提供了一个新的 VAREXISTS 函数，用于测试是否已用给定的名称创建或声明过某个用户定义的变量。进行过此测试后，可以创建该变量（如有必要），然后安全地使用该变量。
有关详细信息，请参见“VAREXISTS 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》。

语句增强

- **隐藏过程文本以保密您的逻辑** 可以使用 SET HIDDEN 选项隐藏存储过程、函数、触发器和视图包含的逻辑。这样在分发应用程序和数据库时便不会泄漏存储过程、函数、触发器和视图中的逻辑。
有关详细信息，请参见“隐藏过程、函数、触发器和视图的内容”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **LOAD TABLE 现在接受长度大于 1 个字节的分隔符** LOAD TABLE 语句现在支持最长 255 个字节的分隔符。
有关详细信息，请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **新的语句为 Adaptive Server Enterprise 和 Microsoft SQL Server 提供了兼容性** 可以使用 DEALLOCATE 语句释放与游标相关联的资源。此语句为 Adaptive Server Enterprise 和 Microsoft SQL Server 提供了兼容性。
有关详细信息，请参见“DEALLOCATE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **ALTER DATABASE 语句具有 dblog 实用程序的功能** 可以使用 ALTER DATABASE 语句更改与数据库文件相关联的事务日志和镜像日志的名称。以前只能使用事务日志 (dblog) 实用程序进行更改。
有关详细信息，请参见“ALTER DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **全局临时表和局部临时表都可以使用 LOAD TABLE 语句** Adaptive Server Anywhere 现在支持针对声明的局部临时表使用 LOAD TABLE 语句。以前只支持全局临时表。
有关详细信息，请参见“LOAD TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **SET 语句可用于给变量赋值** 现在可以在 Transact-SQL 过程中使用 SET 语句给变量赋值。

- **INSERT 语句现在支持 WITH AUTO NAME** 如果在 INSERT 语句中指定 WITH AUTO NAME, SELECT 列表中各项的名称将决定值与目标列的关联。
有关详细信息, 请参见“INSERT 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **增强的 EXIT 语句** Interactive SQL EXIT 语句现在可以为 Interactive SQL 设置退出代码。
有关详细信息, 请参见“EXIT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **在 FROM 子句中指定查询的优化目标** 可以使用 FASTFIRSTROW 表提示来设置查询的优化目标, 而无需将 optimization_goal 选项设置为 first-row。
有关详细信息, 请参见“FROM 子句”一节《SQL Anywhere 服务器 - SQL 参考》。

安全性增强

- **使用新的实用程序可以隐藏文件的内容** 配置文件 (也称作命令文件) 有时包含口令。为增强安全性, Adaptive Server Anywhere 提供了一个称为文件隐藏实用程序的新实用程序, 使您可以使用简单的加密隐藏配置文件的内容。
有关详细信息, 请参见“文件隐藏实用程序 (dbfhide)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **Certicom 加密更改** 增强了安全性, 支持两种类型的 Certicom 加密, ECC_TLS 和 RSA_TLS。在 Adaptive Server Anywhere 以前的版本中称作 Certicom 加密的加密功能现在称作 ECC_TLS 加密。仍接受 Certicom 参数, 并且该参数等效于 ECC_TLS 加密。Adaptive Server Anywhere 现在还支持 RSA_TLS 加密。
有关详细信息, 请参见“-ec 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》或“Encryption 连接参数 [ENC]”一节《SQL Anywhere 服务器 - 数据库管理》。

性能增强

- **新的连接参数可以改善网络响应** LazyClose 和 PrefetchOnOpen 网络连接参数可以在等待时间较长, 或应用程序需要处理大量请求的网络上提高性能。
有关这些参数的信息, 请参见“LazyClose 连接参数 [LCLOSE]”一节《SQL Anywhere 服务器 - 数据库管理》和“PreFetchOnOpen 连接参数”一节《SQL Anywhere 服务器 - 数据库管理》。
- **现在 Windows NT/2000/XP 上使用散读** 以前, 对较大的表进行顺序扫描时, 首先将页复制到一个 64 KB 缓冲区, 然后再复制到高速缓存。现在, 如果您的运行环境是 Windows NT Service Patch 2 或更高版本, 或 Windows 2000/XP, 而且页大小最小为 4 KB, 则散读功能将直接把页复制到高速缓存, 从而节省时间和提高性能。
有关详细信息, 请参见“使用适当的页大小”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **提高了请求记录的时间分辨率** 使用过程分析或请求记录获取的时间的分辨率现在为 1 毫秒。这一变更主要影响在 Windows 操作系统上运行的服务器。
- **运行多个版本的性能监控器** 如果您同时运行多个版本的 Adaptive Server Anywhere, 您也可以同时运行多个版本的 Windows 性能监控器。
有关 Windows 性能监控器的详细信息, 请参见“使用 Windows 性能监控器监控统计信息”一节《SQL Anywhere 服务器 - SQL 的用法》。

其它增强

- **通过注册表设置更改服务器的临时文件夹** 在 Windows CE 平台上，可以使用注册表来指定服务器使用哪个临时目录。
有关详细信息，请参见“[Windows Mobile 上的注册表设置](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **新 iAnywhere JDBC 驱动程序** 此可靠的高性能 JDBC 驱动程序便于用户使用 ODBC 数据源和命令序列客户端/服务器协议。可以用它替代 jConnect JDBC 驱动程序。
有关 iAnywhere JDBC 驱动程序的信息，请参见“[使用 iAnywhere JDBC 驱动程序](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
有关选择 JDBC 驱动程序的信息，请参见“[选择 JDBC 驱动程序](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **触发器可以区分导致触发器触发的操作** 现在可以根据导致触发器触发的是 UPDATE、INSERT 还是 DELETE 操作，来采取不同的操作。利用此功能，可以在一个触发器中的不同事件间共享逻辑，同时还可以以与操作相关的方式来采取一些操作。
有关详细信息，请参见“[触发器操作条件](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **return_date_time_as_string 不再特定于 TDS** 现在，所有连接都可以使用 return_date_time_as_string 选项。
有关此选项的详细信息，请参见“[return_date_time_as_string 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **将空间添加到 dbspace 时可以指定单位** 可以特定大小为单位扩展数据库文件。可以指定的单位包括，页、KB、MB、GB 或 TB。
有关详细信息，请参见“[ALTER DBSPACE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **sa_make_object 系统过程** 可在 SQL 脚本中使用此系统过程，以确保执行提供实际定义的 ALTER 语句前存在对象的框架实例。
有关详细信息，请参见“[sa_make_object 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **与 Microsoft SQL Server 兼容的新全局变量** 引入了一个新的全局变量，实现与 Microsoft SQL Server 的兼容。除了以不同的值返回最近的读取状态外，@@fetch_status 全局变量与 @@sqlstatus 全局变量完全相同。
有关详细信息，请参见“[全局变量](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **在 NetWare 上支持字符集转换** NetWare 现在支持字符集转换。
- **信息实用程序报告安装的 Java 类的版本** dbinfo 实用程序和 a_db_info 结构现在报告数据库中安装的 Java 类的版本。
有关详细信息，请参见“[信息实用程序 \(dbinfo\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[a_db_info 结构](#)”一节《[SQL Anywhere 服务器 - 编程](#)》。
- **取消读取操作警告** 在进行读取时，数据库服务器 8.0 及更高版本与其早期版本相比，会返回范围更广的读取警告消息。通过 [ODBC Configuration for Adaptive Server Anywhere] 窗口可以取消

从数据库服务器返回的警告消息，以确保为使用早期版本的软件部署的应用程序正确地处理这些警告消息。

- **控制对主键列进行更新** 将新 `prevent_article_pkey_update` 选项设置为 On 时，禁止对作为发布的一部分的表的主键列进行更新。该选项有助于确保数据完整性，特别是在复制和同步环境中。

有关详细信息，请参见“[prevent_article_pkey_update 选项 \[数据库\] \[MobiLink 客户端\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

MobiLink 的新功能

下面是 8.0.2 版中引入的对软件的更改和补充的列表。

- **支持 .NET** MobiLink 现在支持使用 Visual Studio .NET 编程语言编写同步脚本。
有关详细信息，请参见“[使用 .NET 编写同步脚本](#)”《[MobiLink - 服务器管理](#)》、“[-sl dnet 选项](#)”一节《[MobiLink - 服务器管理](#)》、“[ml_add_dnet_connection_script 系统过程](#)”一节《[MobiLink - 服务器管理](#)》和“[ml_add_dnet_table_script 系统过程](#)”一节《[MobiLink - 服务器管理](#)》。
- **启动类** 现在可以编写在 MobiLink 服务器启动 Java 虚拟机或 CLR 时（在第一次同步之前）执行的 Java 和 .NET 代码。
有关详细信息，请参见“[用户定义的启动类](#)”一节《[MobiLink - 服务器管理](#)》。
- **使用 UUID 维护唯一的主键** 通过通用唯一 ID（UUID，也称作 GUID）引入了一种在远程服务器上维护唯一的主键的新方法。
有关详细信息，请参见“[使用 UUID](#)”一节《[MobiLink - 服务器管理](#)》。
- **引入新的方法来处理参照完整性违规** 引入了两个新的客户端事件挂接（`sp_hook_dbmsync_download_ri_conflict` 和 `sp_hook_dbmsync_download_log_ri_conflict`），用于帮助您在下载时管理参照完整性违规。
有关详细信息，请参见“[sp_hook_dbmsync_download_ri_violation](#)”一节《[MobiLink - 客户端管理](#)》和“[sp_hook_dbmsync_download_log_ri_violation](#)”一节《[MobiLink - 客户端管理](#)》。
- **提供了更简单的方法来删除远程表中的所有行** 现在可以通过在 `download_delete_cursor` 中加入每个主键列为 NULL 的一行来删除远程表中的所有数据。
有关详细信息，请参见“[编写 download_delete_cursor 脚本](#)”一节《[MobiLink - 服务器管理](#)》。

性能监控器增强

- **MobiLink 监控器** 引入了一个图形工具（MobiLink 监控器）可用于查看同步的各个过程所用的时间，并且可以按照 MobiLink 用户或工作线程进行排序。
有关详细信息，请参见“[MobiLink 监控器](#)”《[MobiLink - 服务器管理](#)》。
- **用户可以使用 dbmsync 估计要上载的行数** 提供了一个新的 `dbmsync` 命令行选项 `-urc`。使用该选项可以估计将要上载的行数，从而提高同步性能。
有关详细信息，请参见“[-urc 选项](#)”一节《[MobiLink - 客户端管理](#)》。

- **用户可以指定持久性 HTTP/HTTPS 连接** 可以使用 `persistent` 选项让 MobiLink 尝试使用相同的连接处理同步中的所有 HTTP 请求。这个设置可以提高性能。但只有在直接连接到 MobiLink，而不通过中间代理（如代理或重定向器）的情况下，才应使用此选项。

有关详细信息，请参见“[CREATE SYNCHRONIZATION USER 语句 \[MobiLink\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **提供了新的方法来控制警告消息** 已创建了三个新的 `dbmlsrv9` 命令行选项：`-zw`、`-zwd` 和 `-zwe`。使用 `-zw` 可以控制要报告哪个级别的警告消息。使用 `-zwd` 禁用特定的警告代码。使用 `-zwe` 可以启用通过 `-zw` 禁用的特定警告。

有关详细信息，请参见“-[zw 选项](#)”一节《[MobiLink - 服务器管理](#)》、“-[zwd 选项](#)”一节《[MobiLink - 服务器管理](#)》和“-[zwe 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **新的详细记录选项** 变更并扩展了 `dbmlsync -v` 命令行选项。现在，单独使用 `-v` 选项只能获得最低的详细程度。若要获得最高的详细程度，应使用 `-v+`。还有几个新的级别可以指定，用于调节记录的信息。这些选项也是作为扩展选项提供的。

有关详细信息，请参见“-[v 选项](#)”一节《[MobiLink - 客户端管理](#)》。

连接增强

- **Ping 支持** 远程数据库现在可以强制 MobiLink 服务器回应。

有关详细信息，请参见“-[pi 选项](#)”一节《[MobiLink - 客户端管理](#)》和“[Ping 同步参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。

- **新的同步流** MobiLink 现在支持 HTTPS 协议。这个新的流通过 SSL/TLS（使用 RSA 加密）实现 HTTP，并且与所有其它 HTTPS 服务器都兼容。

有关详细信息，请参见“-[x 选项](#)”一节《[MobiLink - 服务器管理](#)》和“[CREATE SYNCHRONIZATION USER 语句 \[MobiLink\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **新的 `buffer_size` 选项** 现在可以使用 `buffer_size` 选项为固定长度的 HTTP 消息指定最大缓冲区大小。

有关详细信息，请参见“[CREATE SYNCHRONIZATION USER 语句 \[MobiLink\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **MobiLink 客户端的自动拨号功能** 运行在 Pocket PC 2002 或 Windows 台式计算机上的 MobiLink 客户端现在可以通过拨号网络连接进行连接。通过使用调度，您的远程计算机可以在无人照管的情况下进行同步。新的同步流参数是 `network_name`、`network_connect_timeout` 和 `network_leave_open`。

有关详细信息，请参见“[CREATE SYNCHRONIZATION USER 语句 \[MobiLink\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

新的 Web 服务器支持

- **Servlet 重定向器** MobiLink 现在支持使用支持 Java Servlet API 2.2 的 Web 服务器，包括 Apache Tomcat。

有关详细信息，请参见“[重定向器（不建议使用）](#)”《[MobiLink - 服务器管理](#)》。

安全增强

- **支持 RSA 编码器套件** 现在，既可以使用现有的椭圆曲线加密，也可以使用 RSA 加密，来确保同步安全性。gencert 和 readcert 实用程序既支持椭圆曲线证书，也支持 RSA 证书。
有关详细信息，请参见“加密 MobiLink 客户端/服务器通信”一节《SQL Anywhere 服务器 - 数据库管理》。
- **gencert 可以为预生成的证书请求签名** 证书生成实用程序 gencert 具有一个新的命令行选项，可以为预生成的证书请求签名。

SQL Remote 的新功能

SQL Remote 8.0.2 版包括以下新功能。

- **错误日志发送到统一数据库** 为更好地排除远程站点上出现的故障，可在统一数据库收集日志信息。
有关详细信息，请参见“报告并处理复制错误”一节《SQL Remote》。

UltraLite 的新功能

UltraLite 8.0.2 引入了几个新功能：

- **UltraLite 组件** 现在，可以在新的开发平台上使用 UltraLite 数据库技术，而且使用起来非常简便。UltraLite 组件为 eMbedded Visual Basic、AppForge MobileVB 和 Java 的用户提供了 UltraLite 技术。用于 Java 的组件可以替代本书中介绍的 UltraLite for Java。该组件并不是 100% 纯 Java 实现，而是使用了本地类，以获得更高的性能。
在联机手册中有 UltraLite 组件文档。有关起点，请参见 [UltraLite - 数据库管理和参考](#)。
- **升级 UltraLite 数据库** 部署应用程序的新版本时，现在可以选择将 UltraLite 数据库的模式升级到新应用程序的模式。
在 9.0.1 版中，ULEnableGenericSchema 被 ULRegisterSchemaUpgradeObserver 代替。
- **Java 运行时是线程安全的** UltraLite Java 运行时现在是线程安全的，因此，可以开发多线程 UltraLite 应用程序。
- **删除 UltraLite 数据库文件** 可以使用 ULDropDatabase 函数从应用程序中删除 UltraLite 数据库文件。
有关详细信息，请参见：
 - 嵌入式 SQL：“[ULDropDatabase 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》
 - C++ API: Drop 方法
- **通用唯一标识符** UltraLite 数据库现在可以使用 UNIQUEIDENTIFIER Adaptive Server Anywhere 数据类型。此类型是一个用于存储通用唯一标识符（UUID 或 GUID）的 BINARY(16) 数据类型。使用 NEWID 函数作为缺省值的 UNIQUEIDENTIFIER 列可以确保主键在整个 MobiLink 安装中是唯一的，相当于 GLOBAL AUTOINCREMENT。

有关详细信息，请参见“NEWID 缺省值”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **新的同步安全性组件** 此版本引入了两个新的安全同步协议。HTTPS 是在传送层安全协议上实现的 HTTP；RSA 是一种在 HTTP 或 TCP/IP 网络上使用的传送层安全加密方式。

这些安全选项使用 Certicom 技术。要使用 Certicom 技术，必须获取单独授权的 SQL Anywhere Studio 安全性选件，并且受出口法规约束。有关此选项的详细信息，请参见“SQL Anywhere 11 组件”一节《SQL Anywhere 11 - 简介》。

有关 HTTPS 同步的详细信息，请参见“Stream Type 同步参数”一节《UltraLite - 数据库管理和参考》。

- **重置上次下载时间** 若要重新同步以前下载的数据，例如，将应用程序设置为干净的状态，可以重置上次下载时间戳。

有关详细信息，请参见“ULResetLastDownloadTime 函数”一节《UltraLite - C 及 C++ 编程》。

- **排除以前同步的故障** 现在提供了用于获取最近一次同步的结果（成功或失败）的函数。此功能对于使用 HotSync 的 Palm OS 应用程序特别有用。在这种情况下，在应用程序之外执行同步。

有关详细信息，请参见“ULGetSynchResult 函数”一节《UltraLite - C 及 C++ 编程》。此功能目前还不可用于 UltraLite Java 应用程序。

- **生成更多、更小的文件** -x 选项使 UltraLite 生成器为 C/C++ 项目写出更多、更小的文件。在生成的代码太大，编译器无法在一个文件中对其进行处理时，可以使用此选项。

- **改进的同步观察器** 增强了同步观察器功能。界面中添加了更多状态和字段，可以设计更具响应性和信息性的同步窗口。

8.0.2 版的行为更改

本节列出在 SQL Anywhere Studio 8.0.2 版的组件中引入的行为更改。

Adaptive Server Anywhere 的行为更改

以下是对该软件先前版本的行为更改列表。

- **不再支持 Windows CE 2.11** 已取消了对 Windows CE 2.11 平台的支持。
- **不再支持 SH3 和 SH4 芯片** 已取消了对使用 SH3 和 SH4 芯片的 Windows CE 设备的支持。
- **optimization_goal 设置** optimization_goal 选项的缺省设置是 **All-rows** 而不是 **first-row**。这将影响为某些查询选择的执行计划，因此也会更改性能特性。

有关详细信息，请参见“[optimization_goal 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **xp_cmdshell 在 Windows 操作系统上显示命令窗口** 现在，可以控制 xp_cmdshell 是否启动一个新窗口。此行为更改适用于使用 8.0.2 版或更高版本创建的数据库，或升级到 8.0.2 版或更高版本的数据库。在以前版本的数据库上，仍将保持原来不显示命令窗口的行为。新的行为与其它数据库（如 Adaptive Server Enterprise 和 Microsoft SQL Server）兼容。

在对 xp_cmdshell 的调用中指定另一个参数，可以隐藏该命令窗口。

有关详细信息，请参见“[xp_cmdshell 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **无论数据库服务器使用哪种语言，都可以识别全长英语日期名** 创建事件时，无论数据库服务器使用哪种语言（德语、中文，等等），都可以识别完整的英文日期名称。这意味着使用不同语言运行的服务器可以识别重载脚本中的事件定义。

但使用除英语之外的其它语言的服务器不能识别缩写英语日期名（Mon、Tue，等等）。

有关详细信息，请参见“[CREATE EVENT 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **对 OPTION 设置进行校验** 现在，系统对具有最小值和最大值的整型选项进行校验。如果设置的选项值无效，将出现以下错误消息：“[选项 '%1' 的设置无效](#)”一节《[错误消息](#)》。

如果您在卸载了包含无效选项设置的数据库后又重载了该数据库，将把那些选项设置为最接近的有效值。

下表列出了相关选项。方括号表示包含的范围。

选项	范围
isolation_level	[0, 3]
precision	[0, 127]
scale	[0, 127]
nearest_century	[0, 100]

选项	范围
max_hash_size	[2, 64]
max_work_table_hash_size	[2, 64]
first_day_of_week	[1, 7]
default_timestamp_increment	[1, 60000000]

- **重命名的连接** 有两个连接的名称已更改。这两个连接都是图形式计划中的连接，并且都在文档中进行了说明。嵌套循环不存在连接 (JNE) 现在称作嵌套循环反半连接 (JNLA)；而嵌套循环存在连接 (JE) 现在称作嵌套循环半连接 (JNLS)。

有关详细信息，请参见“连接算法”一节《SQL Anywhere 服务器 - SQL 的用法》。

不再支持和不支持的功能

此列表包括不再支持的和影响现有应用程序的功能。

- **Windows 上不再支持 -d 服务器选项** 在 NetWare 上使用时，-d 选项强制使用 POSIX I/O 而不使用 DFS (Direct File System) I/O。在 Windows 中，仍可以在命令行上使用该选项，但它将被忽略。

MobiLink 的行为更改

以下是对该软件以前版本的行为更改列表。

- **不再支持串行通信协议** 不再支持串行协议。可以使用 HTTP、HTTPS 或 TCP/IP 代替该协议。
- **Certicom 不再是证书颁发机构** 不能再从 Certicom 获取传送层安全性证书。但是，可以继续使用该 Certicom reqtool 实用程序生成证书请求，并且可从许多其它机构（包括 VeriSign 和 Entrust Technologies）购买证书。
有关详细信息，请参见 <http://www.verisign.com/> 或 <http://www.entrust.net/ssl-certificate-services/managed.htm>。
- **不再支持 dbmlsrv -vw 选项** 不建议使用用于取消警告消息的 -vw dbmlsrv 命令行选项。可以使用 -zw 或 -zwd 代替该选项。
有关详细信息，请参见“-zw 选项”一节《MobiLink - 服务器管理》和“-zwd 选项”一节《MobiLink - 服务器管理》。
- **dbmlsync -v 选项的行为已更改** 已变更和扩展了 -v dbmlsync 命令行选项。现在，单独使用 -v 选项只能获得最低的详细程度。
有关详细信息，请参见“-v 选项”一节《MobiLink - 客户端管理》。
- **无论同步服务器使用哪种语言，都可以识别全长英语日期名** 为 MobiLink 用户、发布和预订创建调度时，或在 dbmlsync 命令行上指定调度信息时，如果希望使用除英语之外的其它语言的同步服务器也能识别该调度，则必须使用全长英语日期名（例如，Monday）。

使用除英语之外的其它语言的同步服务器不能识别使用缩写英语日期名（例如，Mon）的调度。有关详细信息，请参见“[CREATE SYNCHRONIZATION USER 语句 \[MobiLink\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **dbmlsync 对长数据的支持更好** 构建上载流时，DBMLSync 现在处理 BLOB 的方式要比以前高效得多。现在，它将 BLOB 分块读入内存，因此处理长 BLOB 的能力不再受可用内存的限制。同时同步多个发布时，BLOB 数据只存储一次，并在上载流之间共享。输出日志现在输出 BLOB 的大小和它的前 32 个字节。
- **删除了 HTTP use_cookies 选项** 删除了 use_cookies 选项。如果使用该选项，它将被忽略。需要时，MobiLink 现在会自动检测 Cookie。

UltraLite 的行为更改

以下列出了对该软件以前版本的行为更改。

- **不再支持 Windows CE 2.11** 已取消了对 Windows CE 2.11 平台的支持。
- **不再支持 SH3 和 SH4 芯片** 已取消了对使用 SH3 和 SH4 芯片的 Windows CE 设备的支持。
- **不再支持串行通信协议** 不再支持串行协议。串行同步主要由 Palm 计算平台上的客户端使用。这些客户端可以改用 HotSync 同步。
- **VxWorks 上没有传送层安全性** VxWorks 操作系统上不再支持为同步提供传送层安全性的 Certicom 库。
- **不支持 VxWorks 5.5** 支持的 VxWorks 操作系统为 VxWorks 5.3 和 5.4。

版本 9 中不支持 VxWorks

版本 9 中完全停止了对 VxWorks 平台的支持。

- **Certicom 库要求 JDK 1.2** 此版本已更新了 Certicom 安全库。用于 Java 应用程序的新库要求 JDK 1.2 而不是 JDK 1.1.4。

8.0.1 版中的新功能

目录

8.0.1 版中的新功能	290
8.0.1 版的行为更改	295

8.0.1 版中的新功能

本节列出在 SQL Anywhere Studio 8.0.1 版的组件中引入的新功能。

Adaptive Server Anywhere 的新功能

本节介绍 Adaptive Server Anywhere 8.0.1 版中的新功能。它提供了主要和次要新功能的详尽列表，并提供了交叉引用指向详细叙述各功能的章节。

- **指定要在表页中保留的空间** 您可以通过指定应在表页中保留的可用空间的百分比来减少表的碎片。
有关详细信息，请参见“减少表碎片”一节《SQL Anywhere 服务器 - SQL 的用法》和“ALTER TABLE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
要在该版本之前创建的数据库上指定要分配的空间百分比，您必须通过卸载并重装数据库来升级数据库的文件格式。
- **新系统表** 已经添加了两个新系统表 SYSATTRIBUTE 和 SYSATTRIBUTENAME。
有关详细信息，请参见“ISYSATTRIBUTE 系统表”一节《SQL Anywhere 服务器 - SQL 参考》和“ISYSATTRIBUTENAME 系统表”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_disk_free_space 系统过程** 本过程允许您确定可用于 dbspace、临时文件、事务日志和事务日志镜像的空间。
有关详细信息，请参见“sa_disk_free_space 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **sa_flush_statistics 系统过程** 数据库管理员可以使用此过程来确保清除只存在于数据库服务器高速缓存中的成本模型统计。
有关详细信息，请参见“sa_flush_statistics 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
- **获取服务器消息窗口内容的新方法** 提供了一个新系统过程和三个新属性，可返回数据库服务器消息窗口中的信息。
- **确定非 ANSI 语句的 ANSI 等价语句** REWRITE 函数接受新参数 ANSI，该参数导致该函数返回与任何 SELECT、UPDATE 或 DELETE 语句等价的 ANSI 语句。
有关详细信息，请参见“REWRITE 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》。
- **允许在 UPDATE 语句中对变量赋值** 除了更新表以外，UPDATE 语句的 SET 子句现在还可用于为变量赋值。此功能与 Adaptive Server Enterprise 兼容。
有关详细信息，请参见“UPDATE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **自动增量的替代方法** 所提供的 GET_IDENTITY 函数作为给自动增量列分配标识值的替代方法。

有关详细信息，请参见“[GET_IDENTITY 函数 \[Miscellaneous\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **方括号可界定标识符** 您可以使用方括号来界定标识符。无论 `quoted_identifier` 选项的设置如何，始终可使用方括号。

有关详细信息，请参见“[标识符](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **在 FROM 子句中指定隔离级别** 您可以使用 `WITH table-hint` 参数为特定的 `SELECT`、`UPDATE` 或 `DELETE` 语句指定特定表或视图的锁定方法。

有关详细信息，请参见“[FROM 子句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **[数据迁移] 向导** [数据迁移] 向导允许您将远程表从 Sybase Central 迁移到 Adaptive Server Anywhere 数据库。

如果目标数据库是 8.0.0 版或更早版本，则不能迁移外键。要迁移外键，必须通过卸载并重装目标数据库来升级该数据库的文件格式。

有关详细信息，请参见“[将数据库迁移到 SQL Anywhere](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

- **从 Sybase Central 卸载 5.x 或 6.x 版数据库** Sybase Central 现在允许您连接到 5.x 或 6.x 版数据库，以便使用 [卸载数据库向导] 升级数据库的文件格式。为此，您必须在 8.0.0 或更高版本的服务器上运行该数据库。
- **从 [升级数据库向导] 备份和关闭数据库** 现在可以使用 Sybase Central 的 [升级数据库向导] 备份数据库文件，包括备份主数据库文件、事务日志和 `dbspace`。该向导还允许您在升级完成时关闭数据库。
- **sa_migrate 增强** `sa_migrate` 过程有一个可选参数 `migrate_fkeys`，该参数允许您指定是否希望在从远程数据库迁移表时迁移外键映射。在以前的版本中，当您使用 `sa_migrate` 过程时，总是迁移外键映射。

有关详细信息，请参见“[sa_migrate 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

要在此版本之前创建的数据库上使用该功能，必须通过卸载并重装数据库来升级数据库文件格式。

- **新 sort_collation 数据库选项** `sort_collation` 数据库选项允许在 `ORDER BY` 表达式上隐式使用 `SORTKEY` 函数。如果将此选项的值设置为有效的 `collation name` 或 `collation ID`，则在处理 `ORDER BY` 子句中的任何字符串表达式时都会象已经调用了 `SORTKEY` 函数那样处理。

有关详细信息，请参见“[sort_collation 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **使用 IP 地址/端口连接到服务器** 您可以使用 `VerifyServerName=NO` 协议选项来跳过服务器名验证，并允许 Adaptive Server Anywhere 客户端在只知道 IP 地址/端口的情况下连接到 Adaptive Server Anywhere 服务器。`VerifyServerName` 参数只在指定了 `DoBroadcast=NONE` 的情况下使用。

有关详细信息，请参见“[VerifyServerName 协议选项 \[VERIFY\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **新 LocalOnly 协议选项控制广播** 您可以使用 LocalOnly 协议选项只连接到本地计算机上的服务器（如果存在）。如果设置 LocalOnly=YES，则会使用常规的广播机制，但是将忽略来自其它计算机上的服务器的广播响应。
有关详细信息，请参见“LocalOnly 协议选项 [LOCAL]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **指定用于驻留游标的高速缓存的量** 您可以使用 pinned_cursor_percent_of_cache 选项调整可用于固定游标的高速缓存量。在内存较少的环境中，降低该限制可以提高性能。
有关详细信息，请参见“pinned_cursor_percent_of_cache 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **监控数据库文件和日志文件的碎片** 您可以使用 DBFileFragments 和 LogFileFragments 数据库属性来选择监控文件的碎片。通常不会特别关注事务日志文件的碎片；但是，数据库文件的碎片会导致性能降低，有必要使用磁盘碎片整理程序。
有关详细信息，请参见“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **新连接属性** 添加了两个新连接属性。LivenessTimeout 返回连接的活动超时，IdleTimeout 返回连接的空闲超时。
有关详细信息，请参见“连接属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **新服务器属性** 新的 IdleTimeout 服务器属性返回缺省的空闲超时值。
有关详细信息，请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **不确定性函数** 有些函数会修改基础数据，有些函数会依赖可能在执行查询的过程中发生更改的基础数据，可将这些函数声明为 NOT DETERMINISTIC。在执行查询的过程中，以这种方式声明的函数在每次被调用时都被重新取值。否则，函数值会被高速缓存并被重新使用以提高性能。
有关详细信息，请参见“CREATE FUNCTION 语句（Web 服务）”一节《SQL Anywhere 服务器 - SQL 参考》。
- **确保完成备份中的所有事务** 在缺省情况下，BACKUP 语句不等打开的事务完成就重命名或截断事务日志。现在，可通过指定 WAIT AFTER END 子句来确保完成备份中包含的所有事务。
有关详细信息，请参见“BACKUP 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

MobiLink 的新功能

以下列出了 8.0.1 版中引入的软件更改和新增功能。

- **错误上下文完整报告** 现在，当同步过程中出现错误时，MobiLink 服务器会在其输出文件中显示完整的错误上下文。
有关详细信息，请参见“-o 选项”一节《MobiLink - 服务器管理》。
- **用户 ID 映射** 现在，使用 MobiLink 可以更容易地查找数据库用户 ID 或将 MobiLink 用户名映射到用户 ID。
有关详细信息，请参见“modify_user 连接事件”一节《MobiLink - 服务器管理》。

- **将地址和类型设置为客户端选项** MobiLink 客户端现在允许在命令行上指定通信类型和地址以连接到 MobiLink 服务器。
有关详细信息，请参见“[MobiLink SQL Anywhere 客户端扩展选项](#)”《[MobiLink - 客户端管理](#)》。
- **记录 MobiLink 发出的 ODBC 语句** 您可以命令 MobiLink 将 MobiLink 发出的所有 ODBC 语句记录到 ODBC 输出文件中。
- **修改下载时间戳** 您可以在两个新事件中修改上次下载时间戳或下一个上次下载时间戳。
有关详细信息，请参见“[modify_last_download_timestamp 连接事件](#)”一节《[MobiLink - 服务器管理](#)》和“[modify_next_last_download_timestamp 连接事件](#)”一节《[MobiLink - 服务器管理](#)》。
- **自动时间戳冲突容错** 如果统一数据库和远程数据库发生时间戳冲突，利用此选项可将精度比最低精度高的时间戳值用来检测冲突。
有关详细信息，请参见“[-zp 选项](#)”一节《[MobiLink - 服务器管理](#)》。

SQL Remote 的新功能

SQL Remote 8.0.1 版包括以下新功能。

- **SMTP 用户验证** 为在使用 SMTP/POP 消息系统时，在 SMTP 服务器上验证单独用户提供了参数。
有关详细信息，请参见“[SMTP 消息系统](#)”一节《[SQL Remote](#)》。

UltraLite 的新功能

UltraLite 8.0.1 引入了几个新功能：

- **支持 CodeWarrior 8** 此版本支持 CodeWarrior 版本 8。
- **支持多线程应用程序** 在支持这种应用程序的平台上，UltraLite 应用程序现在可以是多线程的。
- **Pocket PC 2002 支持** Pocket PC 2002 已被添加到受支持的平台列表中。
- **添加了 JDBC ResultSet 方法** 现在支持 `ResultSet.findColumn` 和 `ResultSet.getType` 方法。
- **从 UltraLite Java 访问信息** 使用 `JdbcConnection.getLastIdentity` 方法、`getLastDownloadTime` 方法和 `JdbcDatabase.countUploadRows` 方法可访问有用的信息。这些功能以前只在 C/C++ 应用程序中可用。
- **UltraLite Java 中的用户验证** UltraLite 的 Java 版本现在支持用户验证。
- **显示 HotSync 同步进度** 现在，台式计算机上 [HotSync 进度] 窗口的状态字段会显示与 UltraLite 应用程序同步的进度。
- **HotSync 配置** 可以从 Palm Desktop 配置 HotSync 管道。
- **从 UltraLite 应用程序自动编写脚本** 现在，UltraLite 应用程序可为 MobiLink 服务器提供列名，从而能够自动生成同步脚本。

- **从 C++ API 获取列的 SQL 数据类型** `GetColumnSQLType` 方法返回列的数据类型。
- **同步过程中的可选检查点** 需要下载大量更新的同步可能导致 UltraLite 数据库的大小急剧增加。通过在同步过程中执行检查点可以限制这种增长。新的 `checkpoint_store` 同步参数控制检查点操作。缺省情况下，不执行任何检查点。

有关详细信息，请参见“[Additional Parameters 同步参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。

8.0.1 版的行为更改

本节列出在 SQL Anywhere Studio 8.0.1 版的组件中引入的行为更改。

Adaptive Server Anywhere 的行为更改

下面是对该软件以前版本的行为更改列表。

- **重命名的事务日志文件的新命名约定** 备份过程中重命名的事务日志文件末尾的两位数字已更改为两个字符。例如，如果首次备份是在 2000 年 12 月 10 日，则现在重命名的日志文件名为 *001210AA.log* 而非 *00121001.log*。前两位数字表示年份，第二个两位数字表示月份，第三个两位数字表示该月的第几天，最后两个字符区分在同一天进行的不同备份。这样会将一天内可能的备份数量从 100 增加到 676。
- **LOAD TABLE 现在对计算列重新计算** 现在，LOAD TABLE 检测计算列，并针对插入到表中的每一行对计算列进行求值。
- **Adaptive Server Anywhere 控制台实用程序 (dbconsole) 现在允许进行多次重新连接。** 以前的 Adaptive Server Anywhere 控制台实用程序 (dbconsole) 会话仅允许进行一次连接。现在，可在不退出应用程序的情况下进行断开和重新连接。

不再使用和 supports 的功能

此列表包括不再支持的和影响现有应用程序的功能。

- **不再支持 DEBUG 连接参数** DEBUG 连接参数已不再使用。您仍可以使用 LOG 参数创建包含调试信息的日志文件。从 8.0.1 版开始，LOG=filename 执行 DEBUG=YES;LOG=filename 过去执行的功能。
有关详细信息，请参见“连接参数”一节《SQL Anywhere 服务器 - 数据库管理》。
- **不再支持 AGENT 连接参数** AGENT 连接参数已不再使用。您可以结合使用 CommLinks 参数和相应的协议选项来实现相同的行为。
有关详细信息，请参见“连接参数”一节《SQL Anywhere 服务器 - 数据库管理》。
- **已删除端口连接属性** 端口连接参数已被删除。
- **已删除 [Adaptive Server Anywhere 转换驱动程序]** 不再建议使用转换驱动程序。服务器自动处理字符集转换。
- **先尝试 SharedMemory** 过去，在 LINKS= 连接参数中指定的端口按照指定它们时的顺序进行尝试。现在，如果指定了 sharedmemory (shmem) 端口，则会先尝试该端口，然后按出现的顺序尝试指定的其它端口。
- **GLOBAL AUTOINCREMENT** 缺省值已从 0 更改为 2147483647。global_database_id 现在可以设置为 0 并且将导致从 1 开始生成值。

MobiLink 的行为更改

- **时间戳不匹配通知** 当统一数据库和远程数据库之间的时间戳不一致时，MobiLink 服务器将对每次同步记录一个警告。
- **GLOBAL AUTOINCREMENT** 缺省值已从 0 更改为 2147483647。global_database_id 现在可以设置为 0 并且将导致从 1 开始生成值。

仍会出现下列情况：如果未设置 global_database_id 或者设置为缺省值，则如果进行会导致生成全局自动增量值的尝试，将产生 NULL。这通常在尝试将该值插入到非空的主键列中时报告错误，并可指示尚未设置 global_database_id 选项。

如果不允许将 global_database_id 设置为 0，则可阻止从 1 开始生成值，而是从为该列指定的分区大小开始。

有关详细信息，请参见“[global_database_id 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **dbmlstop 执行软关机** 在缺省情况下（如果 -w、-f、-h 或 -t 均未指定），dbmlstop 执行软关机。这意味着它在当前的同步完成时停止接受新连接并退出。

有关详细信息，请参见“[MobiLink 停止实用程序 \(mlstop\)](#)”一节《[MobiLink - 服务器管理](#)》。

UltraLite 的行为更改

- **Palm 数据库备份** 在以前的版本中，如果 ULUtil 应用程序用于备份数据库，该数据库将在执行每次后续 HotSync 操作时进行备份。

大多数 UltraLite 数据都能通过同步进行有效备份。由于显式备份的最常见目的在于创建一个用于部署的原始数据库，所以在大多数情况下，持续地在 HotSync 上创建备份不是所需的行为。现在，每次 UltraLite 应用程序启动时，都会禁止在以后执行 HotSync 操作时进行备份。

如果您希望显式地要求在每次执行 HotSync 时进行数据库备份，可以在 UL_STORE_PARMS 宏中设置 palm_all_backup 参数。

不再支持和不支持的功能

在使用 ScoutSync 技术的 Palm 计算平台上，不再为同步提供 UltraLite 支持。8.0.x 版仍然支持 ScoutSync 3.6 版或更低版本，但 SQL Anywhere Studio 的下一主要版本不支持 ScoutSync。

8.0.0 版中的新功能

目录

版本 8 中的新功能	298
版本 8 中的行为更改	315

版本 8 中的新功能

本节列出了 SQL Anywhere Studio 8 版的组件中引入的新功能。

Adaptive Server Anywhere 的新功能

本节介绍了 Adaptive Server Anywhere 8.0 版中的新功能。它提供主要和次要新功能的详尽列表，并提供交叉引用以指向手册中详细叙述各功能的章节。

如果您有本书的印刷版本，但没有完整的 SQL Anywhere Studio 文档集，则应当查看联机文档以获得每个功能的详细描述。

有些新功能需要您将数据库升级到版本 8，或通过卸载并重装数据库来升级数据库文件格式。如果需要进行数据库升级或文件格式升级才能使用某一特定功能，那么这项要求会在下面的叙述中指出。

有关如何执行这些任务的信息，请参见“升级到 SQL Anywhere 11”第 323 页。

Adaptive Server Anywhere 的新功能是按下列标题分组的：

- “查询处理和数据库性能”一节第 298 页
- “安全性”一节第 300 页
- “SQL 功能”一节第 300 页
- “开发和管理工具”一节第 301 页
- “应用程序开发”一节第 302 页
- “管理和故障排除”一节第 303 页
- “客户端/服务器连接”一节第 306 页
- “数据库中的 Java”一节第 307 页
- “文档”一节第 307 页
- “杂项”一节第 308 页

查询处理和数据库性能

- **查询处理已得到改进** 这一版提供了对查询执行引擎和优化程序的增强，从而使性能大大提高，尤其是对于复杂的查询。对 Adaptive Server Anywhere 查询处理的增强包括以下几个方面：

- 对连接的内部处理更加完善。
- 改进了用于评估其它可选访问计划的优化程序的成本模型。
- 改进了执行模型。

这些更改中大多数都是内部更改。文档在“[查询优化与执行](#)”《[SQL Anywhere 服务器 - SQL 的用法](#)》中提供。

这些更改使得输出结果的效率大大提高。使用临时工作表可能是一种执行查询非常高效的方式。有关详细信息，请参见“[在查询处理中使用工作表（使用 All-rows 优化目标）](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

现在，优化程序可基于成本选择索引，而且不再像旧版本那样仅仅依赖于谓语的选择性。

许多改进的查询处理不需要将数据库升级。要在此版本之前创建的数据库上使用新的开销模型，必须通过卸载和重装该数据库来升级数据库文件格式。

- **新索引类型** 添加了一个新的索引类型，从而提高了多列索引和包括宽列的索引的性能。该索引是一个压缩 B 树索引。

Adaptive Server Anywhere 自动基于索引宽度（索引中所有列的宽度之和）创建合适类型的索引。当索引宽度大于 9 个字节并小于页面大小的 1/8（最大为 256 个字节）时，将会创建压缩 B 树索引；否则，Adaptive Server Anywhere 将会创建散列 B 树索引。

不支持 CREATE INDEX 语句的 WITH HASH SIZE 子句。

要在此版本之前创建的数据库上使用新的索引类型，必须通过卸载和重装该数据库来升级数据库文件格式。

新增了一条限制：外键索引的大小和类型必须与相应的主键索引相同。

如果最初使用缺省值 (WITH HASH SIZE 10) 指定了散列大小，*dbunload* 现在将忽略指定的散列大小。

- **新数据库选项 optimization_goal** 确定优化查询处理的意图：是迅速返回第一行，还是为最大程度地降低返回整个结果集的开销。缺省设置为优化返回第一行的速度。

有关详细信息，请参见“[optimization_goal 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **表扫描的性能增强** 对于在 Adaptive Server Anywhere 8.0 中创建的页面为 2K、4K 或 8K 的数据库，当执行需要顺序表扫描的查询时其性能会得到增强。Adaptive Server Anywhere 将为大表创建位图，也称为页映射。位图列出了包含一个给定表的数据的所有页。有了该功能，只需要一个 I/O 操作即可搜索庞大的表。

有关详细信息，请参见“[表大小和页面大小](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

对于在此发行版本发布之前创建的数据库，要获得此增强功能所带来的益处，必须通过卸载后再重装该数据库来升级数据库文件格式。

- **改进了检查点日志的存储** 现在，检查点日志存储在数据库文件末尾处的连续几页中。这样就可以允许对检查点日志的内容进行顺序扫描和多页写入，从而改善性能。

有关检查点日志的详细信息，请参见“[了解检查点日志](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

对于在此版本之前创建的数据库，要获得此增强所带来的益处，必须通过卸载后再重装该数据库来升级数据库文件格式。

- **计划高速缓存** 现在，Adaptive Server Anywhere 可为在存储过程、用户定义的函数和触发器中执行的查询和 INSERT、UPDATE 和 DELETE 语句高速缓存执行计划。要高速缓存的最大计划数量用选项设置 max_plans_cached 来指定。要禁用计划高速缓存，请将此选项设置为 0。

有关详细信息，请参见“[计划高速缓存](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

- **替换缺省 I/O 成本模型** 现在，您可以使用带有 CALIBRATE 子句的 ALTER DATABASE 语句来替换缺省的 I/O 成本模型。

有关详细信息，请参见“[ALTER DATABASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **新数据库选项 max_plans_cached** 设置存储在高速缓存中的执行计划的最大数量。
有关详细信息，请参见“[max_plans_cached 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **新数据库选项 min_table_size_for_histogram** 此选项设置用来创建直方图的表的最小大小。直方图存储有关一个列中值的分布信息，优化程序利用它们来选择一个高效的执行计划。

安全性

- **TCP/IP 上的高度加密** 现在，Adaptive Server Anywhere 支持在 Solaris、Linux、NetWare 和所有受支持的 Windows 操作系统（Windows CE 除外）上对 TCP/IP 端口进行基于证书的加密。当网络数据包在客户端和服务器之间进行传输时，使用高度加密可保护它们的保密性和完整性。这种加密也称作传送层安全（Transport Layer Security，简称 TLS）。

数据库服务器 `-ec` 命令行选项可用来设置服务器的连接参数并替换 Adaptive Server Anywhere 的早期版本中的 `-e` 命令行选项。您可以设置带有加密连接参数的客户端连接参数。

有关详细信息，请参见“[-ec 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[Encryption 连接参数 \[ENC\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

要使用此功能，必须在客户端和服务器上都使用版本 8 软件。但不需要升级数据库。

- **数据库文件的高度加密** 现在可以对数据库文件本身进行高度加密，以提高安全性，尤其是在容易被盗窃的笔记本电脑和膝上型电脑上。

有关详细信息，请参见下列位置的内容：

- “[初始化实用程序 \(dbinit\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
- “[-ek 数据库选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
- “[-ep 服务器选项](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》
- “[CREATE DECRYPTED FILE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》

要创建加密的数据库文件，必须使用版本 8 软件。

SQL 功能

- **完全外连接** 现在支持完全外连接。此外，关键字 OUTER 现在对于右外连接、左外连接和完全外连接是可选的。
有关详细信息，请参见“[外连接](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
- **CASE 语句** ANSI 标准允许有两种形式的 CASE 语句。Adaptive Server Anywhere 8.0 支持这两种语法。
有关详细信息，请参见“[CASE 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **WAITFOR 语句** 此语句将对当前连接的处理延迟指定的时间长度或延迟到一个给定时间。
有关详细信息，请参见“[WAITFOR 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **RAISERROR 语句允许禁止连接** 现在此语句可以用于禁止或限制连接。

有关详细信息，请参见“[RAISERROR 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **时区调整** 为了便于在各时区之间调整日期/时间值，添加了下列新功能：
 - **当前 UTC 时间戳** 按照服务器的时区调整值来调整时区值。
 - **缺省 UTC 时间戳** 为 INSERT 指定缺省值并将更新的列设置为该值。
 - **TimeZoneAdjustment 属性** 返回为显示新的本地时间而必须添加到协调通用时间（Coordinated Universal Time，简称 UTC）的分钟数。
 - **time_zone_adjustment 选项** 允许修改连接的时区调整值。
- **新归类函数** SORTKEY 函数生成可用于对字符数据进行排序的值。SORTKEY 可用于在 Adaptive Server Anywhere 归类的缺省行为之外执行排序。

COMPARE 函数可用于基于替代归类规则直接比较两个字符串。

有关详细信息，请参见“[SORTKEY 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[COMPARE 函数 \[String\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **ERRORMSG 函数** 新的 SQL 函数 ERRORMSG 可用于获取错误消息。

有关详细信息，请参见“[ERRORMSG 函数 \[Miscellaneous\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **数据类型转换函数** ISDATE 和 ISNUMERIC 函数分别测试字符串是可以转换为日期还是可以转换为数字。

有关详细信息，请参见“[ISDATE 函数 \[Data type conversion\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[ISNUMERIC 函数 \[Miscellaneous\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

开发和管理工具

- **可访问性功能** SQL Anywhere Studio 符合《美国联邦康复条例》中的第 508 节。用户界面和文档都是遵从该条例制作的。辅助功能支持组件提供了可以实现使用辅助功能工具的软件。在缺省情况下不会安装辅助功能支持组件。

有关详细信息，请参见“[SQL Anywhere 11 组件](#)”一节《[SQL Anywhere 11 - 简介](#)》。
- **查询编辑器** Interactive SQL 中添加了一个图形查询编辑器。通过查询编辑器，您无需使用 SQL 代码就可以创建或编辑 SELECT 语句。在 Interactive SQL 中，您可以通过单击 [工具] » [编辑查询] 来打开 [查询编辑器]。
- **Interactive SQL 和 Sybase Central 中的可编辑数据** 您可以通过编辑 Interactive SQL 结果集以及编辑 Sybase Central 中的表和视图来更新数据库。可以复制、编辑、插入和删除行值。

Sybase Central 中显示的数据可以复制到剪贴板中。

有关详细信息，请参见“[在 Interactive SQL 中编辑结果集](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **Interactive SQL 支持 SQL 转义语法处理** Interactive SQL 现在支持 JDBC 转义语法，该语法允许您访问由 JDBC 驱动程序实现的函数库。

有关详细信息，请参见“使用 JDBC 转义语法”一节《SQL Anywhere 服务器 - 编程》。

- **过程分析** Sybase Central 包含一个 [分析] 选项卡，上面显示了有关存储过程、函数、事件和触发器的调用次数和执行次数的信息。另外，可以查看有关一个过程内每一行的执行速度的信息。通过 Sybase Central 和 SQL 存储过程可以获得分析信息。

有关在 Sybase Central 中查看过程分析信息的详细信息，请参见“使用系统过程进行过程分析”一节《SQL Anywhere 服务器 - SQL 的用法》。

有关使用 SQL 存储过程获取过程分析信息的详细信息，请参见“sa_procedure_profile_summary 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》和“sa_procedure_profile 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

要使用此功能，必须升级数据库。

- **访问计划的信息得到改进** 查看计划有两种新方式，即图形显示和带有统计数据的图形显示。这些新计划提供了有关查询的处理开销的详细信息，并可使您检查查询的子集的开销。现在缺省访问计划是图形计划。详细计划和简要计划现在都基于 Adaptive Server Enterprise 使用的 Ariadne 语法，并且有新的缩写。

有关详细信息，请参见“读取执行计划”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **[结果] 窗格显示查询执行计划** 现在，Interactive SQL [结果] 窗格中新增了一个 [结果] 选项卡。[结果] 选项卡将显示查询的结果，而 [计划] 选项卡将显示查询的执行计划。以前，查询执行计划显示在 Interactive SQL [消息] 窗格中。

- **[结果] 窗格显示 UltraLite 计划** Interactive SQL [结果] 窗格中现在包含一个 [UltraLite 计划] 选项卡。该选项卡以 XML 格式将 UltraLite 计划优化策略显示为字符串。

- **使用 OUTPUT 语句以 XML 格式导出** 可以将查询结果以 XML 格式导出。输出结果带有嵌入式 DTD。二进制值以 CDATA 块进行编码，二进制数据显示为两位数十六进制字符串。

有关详细信息，请参见“OUTPUT 语句 [Interactive SQL]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **Interactive SQL 批处理选项** 在运行批处理文件时，通过 -codepage 和 -onerror 命令行选项可为 Interactive SQL 赋予更多的控制功能。此外，-d1 命令行选项也提供了对调试批处理文件有用的反馈。

有关详细信息，请参见“Interactive SQL 实用程序 (dbisql)”一节《SQL Anywhere 服务器 - 数据库管理》。

应用程序开发

- **新游标类型** Adaptive Server Anywhere 提供的游标已得到增强，可提供更清楚的语义，从而更好地匹配新的游标类型（如由键集决定的游标），并利用新的查询优化可能性。

有关详细信息，请参见“SQL Anywhere 游标”一节《SQL Anywhere 服务器 - 编程》。

- **读取长列得到改进** 可在单一操作中读取的数据量已从 32 KB 提高到了一个可配置的值（缺省值为 256 KB）。在 ODBC 中，可使用 SQL_ATTR_MAX_LENGTH 语句属性来设置该值。在嵌入式 SQL 中，可使用 DT_LONGVARCHAR 和 DT_LONGBINARY 类型。

有关详细信息，请参见“检索数据”一节《SQL Anywhere 服务器 - 编程》和“发送和检索 Long 型值”一节《SQL Anywhere 服务器 - 编程》。

- **用于获取数据库属性的新嵌入式 SQL 函数** 函数 `db_get_property` 可用于获取数据库属性。
有关详细信息，请参见“`db_get_property` 函数”一节《SQL Anywhere 服务器 - 编程》。有关数据库属性的信息，请参见“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。
- **blocking_timeout 选项** 新的 `blocking_timeout` 选项可用于控制事务获取锁需要等待的时间。
有关详细信息，请参见“`blocking_timeout` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **return_date_time_as_string 选项** `return_date_time_as_string` 选项可用于控制通过 `jConnect` 和 `Open Client` 返回日期、时间和时间戳值的方式。
有关详细信息，请参见“`return_date_time_as_string` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

管理和故障排除

除了上文列出的 Sybase Central 中添加的管理增强外，版本 8 还包括了以下管理增强。

- **在不中断访问的情况下提高表性能** 由于要求连续访问数据库而无法完全重建数据库时，可使用 `REORGANIZE TABLE` 语句提高性能。使用该语句可对一个表中的行进行碎片整理，或压缩由于 `DELETE` 变得稀疏的索引。除了可以减少索引树中的级别数之外，它还可以减少用于存储表及其索引的总页数。
要基于主键、外键或索引对表进行重组，数据库必须是 `Adaptive Server Anywhere 7` 或更高版本。
有关详细信息，请参见“`REORGANIZE TABLE` 语句”一节《SQL Anywhere 服务器 - SQL 参考》。
- **快速数据库校验** 添加了一个新的校验检查类型，可减少数据库进行校验所需要的时间。该选项对于需要用小的高速缓存大小校验大数据库的用户特别有用。受影响的工具包括 `sa_validate` 系统过程、校验实用程序 (`dbvalid`) 和 `VALIDATE TABLE` 语句。
有关详细信息，请参见“在校验数据库时提高性能”一节《SQL Anywhere 服务器 - 数据库管理》。
要在此版本之前创建的数据库上使用该功能，必须通过卸载并重装数据库来升级数据库文件格式。
- **备份不必等到未完成的事务完成** 如果备份指令要求将事务日志截断或重命名，则会将未提交的事务转移到新的事务日志中。这就意味着，在开始备份之前，服务器不再需要等待提交或回退未完成的事务。
有关详细信息，请参见“翻译日志文件实用程序 (`dbtran`)”一节《SQL Anywhere 服务器 - 数据库管理》。
要在此版本之前创建的数据库上使用该功能，必须通过卸载并重装数据库来升级数据库文件格式。

- **获取碎片统计** 文件、表以及索引碎片都会降低性能。在 Adaptive Server Anywhere 8.0 中，当您在 Windows NT 上启动一个数据库时，服务器将自动显示有关每一个 dbspace 中的文件碎片数量的信息。

新的系统过程 `sa_table_fragmentation` 和 `sa_index_density` 可使数据库管理员获取有关一个数据库的表和索引的碎片的信息。

有关文件碎片的详细信息，请参见“减少文件碎片”一节《SQL Anywhere 服务器 - SQL 的用法》。

有关表碎片的详细信息，请参见“减少表碎片”一节《SQL Anywhere 服务器 - SQL 的用法》和“`sa_table_fragmentation` 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

有关索引碎片的详细信息，请参见“减少索引碎片和分布偏差”一节《SQL Anywhere 服务器 - SQL 的用法》和“`sa_index_density` 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

- **获取最近预准备 SQL 语句以进行连接** 数据库服务器 `-zl` 命令行选项用于启用捕获功能，为服务器上的每个数据库连接捕获最近准备好的 SQL 语句。您还可以使用 `sa_server_option` 存储过程结合 `remember_last_statement` 设置启用此功能。

当此功能启用之后，`LastStatement` 属性函数和 `sa_conn_activity` 系统过程将分别为服务器上数据库的当前连接和所有连接返回最近准备好的 SQL 语句。

有关详细信息，请参见“`-zl` 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》、“`sa_conn_activity` 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》和“`sa_server_option` 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。

- **`-cw` 命令行选项** 此服务器选项可让您在 Windows 2000、Windows XP 和 Windows Server 2003 上最多使用 64 GB 的高速缓存大小。

有关详细信息，请参见“`-cw` 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **`-qp` 选项** 使用该服务器选项，可取消数据库服务器消息窗口中有关性能的消息。

有关详细信息，请参见“`-qp` 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **调试服务器日志得到改进** 连接调试程序中记录的信息已得到改进，可以提供有关正在尝试的连接部分的更多上下文、删除 `CONN:`前缀以及增加 TCP/IP 消息的数量。

- **数据库可以保存更多过程** `SYSPROCEDURE`、`SYSPROCPARM`、`SYSPROCPERM` 和 `SYSTRIGGER` 系统表的主键值已经从 `SMALLINT` 更改为 `UNSIGNED INT`。这种更改增加了数据库可以保存的过程的数量。

有关数据库可以保存的过程数量的详细信息，请参见“SQL Anywhere 大小和数量限制”一节《SQL Anywhere 服务器 - 数据库管理》。

要使用此功能，必须将数据库文件格式升级。

- **监控查询性能** 提供了新的系统过程和实用程序来测量查询性能。

有关详细信息，请参见“`sa_get_request_profile` 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》、“`sa_get_request_times` 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》和“监控查询性能”一节《SQL Anywhere 服务器 - SQL 的用法》。

- **新的诊断属性** 这些属性可使您获取有关连接、数据库和当前数据库服务器的信息。本发行版本中添加了以下连接属性：

- UtilCmdsPermitted 属性
- TempTablePages 属性
- LastStatement 属性
- PacketSize 属性
- max_plans_cached 属性
- QueryCachePages 属性
- QueryLowMemoryStrategy 属性

有关详细信息，请参见“连接属性”一节《SQL Anywhere 服务器 - 数据库管理》。

在本版本中添加了下列数据库属性：

- DBFileFragments 属性
- LogFileFragments 属性
- BlobArenas 属性
- SeparateForeignKeys 属性
- VariableHashSize 属性
- TableBitMaps 属性
- FreePageBitMaps 属性
- SeparateCheckpointLog 属性
- Histograms 属性
- LargeProcedureIDs 属性
- PreserveSource 属性
- TransactionsSpanLogs 属性
- Capabilities 属性
- TempTablePages 属性
- CompressedBTrees 属性
- ProcedurePages 属性
- QueryCachePages 属性
- QueryLowMemoryStrategy 属性

有关详细信息，请参见“数据库属性”一节《SQL Anywhere 服务器 - 数据库管理》。

在本版本中添加了下列服务器属性：

- MachineName 属性
- IsJavaAvailable 属性
- PlatformVer 属性

有关详细信息，请参见“数据库服务器属性”一节《SQL Anywhere 服务器 - 数据库管理》。

- **其它性能监控统计** 在此版本中还添加了多项性能监控统计。
有关详细信息，请参见“性能监控器统计”一节《SQL Anywhere 服务器 - SQL 的用法》。
- **登录过程允许禁止连接** `login_procedure` 选项允许为每个新连接调用一个存储过程。此过程现在可用于禁止建立数据库连接。
有关详细信息，请参见“`login_procedure` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **dbsvc 增强** 用于管理 Windows 服务的 `dbsvc` 实用程序已得到扩展，可以列出用于启动和停止服务（通过系统 `net start` 和 `net stop` 命令）的服务名称，并且可处理对其它服务和组的依赖性。
有关详细信息，请参见“用于 Windows 的服务实用程序 (dbsvc)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **为存储过程保留源格式** 源格式（包括空格和换行符）现在作为注释存储在数据库中。该注释用于过程分析。

客户端/服务器连接

- **缓冲区大小变化得到改进** 现在可以分别为客户端和服务器指定缓冲区大小。
要使用此功能，必须在客户端和服务器上都使用版本 8 软件。但不需要升级数据库。
- **通信压缩** 如果在带宽有限的网络（包括某些无线网络、某些调制解调器、串行链接和某些广域网）上传输数据，可以使用新的通信压缩类型以改善性能。
有关详细信息，请参见“调整通信压缩设置以改善性能”一节《SQL Anywhere 服务器 - 数据库管理》。
要使用此功能，必须在客户端和服务器上都使用版本 8 软件。但不需要升级数据库。
- **增强的 dbping** `dbping` 实用程序有更多的选项帮助诊断连接问题。这些选项包括：使用 ODBC 进行连接；在连接之后报告连接、数据库和服务器属性。
有关详细信息，请参见“Ping 实用程序 (dbping)”一节《SQL Anywhere 服务器 - 数据库管理》。
- **抑制 TDS 调试选项** `suppress_tds_debugging` 选项控制 TDS 调试信息是否出现在数据库服务器消息窗口中。
有关详细信息，请参见“`suppress_tds_debugging` 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **PrefetchBuffer 连接参数** 该连接参数可让您指定用于存储预读行的最大内存量。
有关详细信息，请参见“PrefetchBuffer 连接参数 [PBUF]”一节《SQL Anywhere 服务器 - 数据库管理》。
- **PrefetchRows 连接参数** `PrefetchRows` 连接参数可让您指定查询数据库时预读的最大行数。在某些情况下，增加客户端从数据库服务器中预读的行数可以提高查询性能。

有关详细信息，请参见“PrefetchRows 连接参数 [PROWS]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **客户端可以指定空闲超时** 每个客户端都可以使用 IDLE 连接参数指定其自身的空闲超时。以前，与服务器的所有连接都使用由 -ti 服务器命令行选项指定的同一空闲超时。

有关详细信息，请参见“Idle 连接参数”一节《SQL Anywhere 服务器 - 数据库管理》。

数据库中的 Java

数据库中的 Java 提供了下列新功能：

- **Java 2 支持** 数据库中的 Java 现在可以使用 Java 2 (JDK 1.2 和 1.3) 和 Java 中的类。
要使用此功能，必须使用 ALTER DATABASE 或通过使用 dbupgrad 实用程序并提供 -jdk 选项将数据库升级。
- **JDBC 2.0** 数据库中的 Java 类现在可以使用 JDBC 2.0 接口访问数据。
要使用此功能，必须使用 ALTER DATABASE 或通过使用 dbupgrad 实用程序并提供 -jdk 选项将数据库升级。
- **诊断过程** 新的系统过程 sa_java_loaded_classes 列出了 Java 虚拟机装载的所有类。
有关详细信息，请参见“sa_java_loaded_classes 系统过程”一节《SQL Anywhere 服务器 - SQL 参考》。
要使用此功能，必须升级数据库。
- **安全管理器** 您可以使用内部安全管理器或自己的机制，以控制对安全敏感型 Java 功能的访问。
有关详细信息，请参见“Java 的安全管理”一节《SQL Anywhere 服务器 - 编程》。

文档

Adaptive Server Anywhere 文档集中添加了多个新功能，可帮助您更快地查找、访问和使用信息。

- **重新组织的书** 自上一版本以来文档集发生了两个主要更改：
 - *Replication and Synchronization Guide* 一书被一分为二，分别介绍这两种同步技术。这两本新书是 *MobiLink 同步用户指南* 和 *SQL Remote 用户指南*。
 - Adaptive Server Anywhere *用户指南*、*编程接口指南* 和 *参考手册* 已被 *数据库管理指南*、*SQL 用户指南*、*SQL 参考手册* 和 *编程指南* 所取代。数据库错误消息已经归集到各自的书中。新的组织形式使得每本书的打印大小更加便于管理。
- **新的上下文相关帮助** 所有用户接口工具（包括 Sybase Central、Interactive SQL、Adaptive Server Anywhere 调试程序以及查询编辑器）都共享一个公用的、跨平台上下文相关帮助系统，并带有指向联机手册的链接。
- **增强的联机手册** 联机手册的 HTML 帮助版本中提供了一个菜单栏，用于快速访问 SQL Anywhere Web 链接、教程、过程等等。

杂项

- **在休眠期间保持连接** 现在，当计算机处于休眠状态时，来自嵌入式 SQL、ODBC 或 OLE DB 客户端的连接仍可持续存在。以前，如果计算机的休眠时间大于活动或空闲超时时间，那么，当计算机从休眠状态被唤醒时，这台计算机上的客户端和服务器之间的 TCP/IP 连接将被切断。
- **查看当前的许可信息** `dblic` 实用程序现在接受一个参数，您可以使用该参数查看一个服务器可执行文件的当前许可信息，而无需启动服务器。

有关详细信息，请参见“[服务器授权实用程序 \(dblic\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **查看自定义归类的归类标签和名称** `dbinfo` 实用程序现在可返回自定义归类的归类标签和名称。此外，`dbtools.h` 中的 `a_db_info` 结构中添加了两个新字段：`collationnamebuffer` 和 `collationnamebufsize`。
- **sp_remote_tables 系统过程** `sp_remote_tables` 存储过程中添加了一个新参数 `tabletype`。此参数可返回远程表的类型。

有关 `tabletype argument` 的详细信息，请参见“[sp_remote_tables 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **-ct 命令行选项** 通过 `-ct` 命令行选项，您可以启用或关闭字符集转换。现在，字符集转换在缺省情况下是启用的，要将它关闭，可以指定 `-ct-`。要启用字符集转换，请使用 `-ct+`。
- **获取远程表外键信息** 两个新的存储过程 `sp_remote_exported_keys` 和 `sp_remote_imported_keys` 可用于获取有关远程表的外键和它们对应的主键的信息。

有关详细信息，请参见“[sp_remote_exported_keys 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[sp_remote_imported_keys 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **xp_sendmail** 此版本中新增了用于通过 SMTP 以及 MAPI 发送电子邮件的扩展存储过程。有关详细信息，请参见“[xp_startsmtp 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》和“[xp_stopsmtp 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

`xp_sendmail` 存储过程现在可接受任何长度的消息。该过程的 `LONG VARCHAR` 参数的长度受系统上可用内存量的限制。

有关详细信息，请参见“[xp_sendmail 系统过程](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **日志传送管理器的复制服务器 12 功能** LTM 配置文件中的 `qualify_table_owners` 参数为复制服务器 12 功能提供支持，允许主数据库中的表名、所有者和列名不同于复制数据库。

有关详细信息，请参见“[LTM 配置文件](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

- **ASANYSH8 环境变量** 添加了一个新的环境变量 `ASANYSH8`。Interactive SQL、Sybase Central、Adaptive Server Anywhere 控制台实用程序和调试程序使用该环境变量来定位共享的组件目录。

MobiLink 的新功能

以下列出了 8.0 版中引入的对软件的更改和补充。

灵活性

- **Java 同步逻辑** 同步脚本现在可以用 Java 来实现，以代替 SQL 语言或作为 SQL 语言的补充。这些脚本使用 MobiLink Java 环境在外部 JRE 中运行。
有关详细信息，请参见“使用 Java 语言编写同步脚本”《MobiLink - 服务器管理》。
- **使用发布进行同步** MobiLink 客户端中的所有数据不再需要同时进行同步。数据可以组织到发布中，每个发布都可以独立地进行同步。为发布和同步预订提供了一个新语法，该语法比以前的语法更简单、更精确。
有关详细信息，请参见“SQL Anywhere 客户端”《MobiLink - 客户端管理》。
- **配置 Web 服务器以处理 MobiLink 同步** 现在可以与防火墙背后的 MobiLink 服务器进行 HTTP 同步。一个适用于常用 Web 服务器的 Web 服务器插件可用来通过 Web 服务器进行 HTTP 同步。
有关详细信息，请参见“重定向器（不建议使用）”《MobiLink - 服务器管理》。
- **对 Windows CE 客户端的 ActiveSync 支持** Adaptive Server Anywhere 和 UltraLite Windows CE MobiLink 客户端都可以使用 Windows CE ActiveSync 同步软件。
有关详细信息，请参见“使用 ActiveSync 同步”一节《MobiLink - 客户端管理》。
- **增强的客户端命令行功能** 您可以在 CREATE/ALTER SYNCHRONIZATION SUBSCRIPTION 语句和命令行中指定扩展选项。
有关详细信息，请参见“dbmsync 语法”一节《MobiLink - 客户端管理》。
- **扩展选项可以存储在数据库中** 通过使用 CREATE/ALTER SYNCHRONIZATION SUBSCRIPTION 语句，可以将扩展选项和连接参数存储在数据库中并将其与预订、用户或发布关联。*Dbmsync* 现在从数据库中读取此信息。
有关详细信息，请参见“dbmsync 语法”一节《MobiLink - 客户端管理》。

性能

- **基于语句的上载** MobiLink 现在允许进行基于语句的上载，这种上载方式不仅比基于游标的上载更直观，而且还能大大提高速度。基于语句的上载采用 `upload_insert`、`upload_delete`、`upload_update`、`upload_new_row_insert` 和 `upload_old_row_insert` 事件。解决冲突时使用 `upload_fetch` 脚本。
有关详细信息，请参见“编写用于上载行的脚本”一节《MobiLink - 服务器管理》。
- **多处理器管理** MobiLink 有一个新选项，用于设置要使用的处理器的最大数量。`-zt` 选项可以对 MobiLink 服务器使用的资源进行更大的控制。它还可以帮助查找和/或解决存在多处理器问题的 ODBC 驱动程序。
有关详细信息，请参见“`-zt` 选项”一节《MobiLink - 服务器管理》。
- **可选的下载确认** MobiLink 同步客户端现在无需进行下载确认就可以同步，因此 MobiLink 服务器工作线程不必等待客户端应用下载，从而客户端可更快地释放工作线程以进行下一次同步。下载确认现在是一个可选项。去掉下载确认可以提高吞吐量，特别是对于速度较慢的客户端。请注意，在没有下载确认的情况下，统一端在下次同步之前不会知道下载是否成功。

有关详细信息，请参见“[MobiLink SQL Anywhere 客户端扩展选项](#)”《[MobiLink - 客户端管理](#)》。

- **缓冲下载流** MobiLink 服务器现在在下载高速缓存中缓冲下载流。由于客户端不需要确认即可提交下载事务，经过缓冲的下载流会在提交之后发送到客户端，这样，下载事务就不会再因网络延迟而有可能延缓。

下载流还可以在 Adaptive Server Anywhere 客户端中进行缓冲。可用缓冲区的大小可以使用 *dbmlsync* *DownloadBufferSize* 扩展选项进行设置。

有关详细信息，请参见“[MobiLink SQL Anywhere 客户端扩展选项](#)”《[MobiLink - 客户端管理](#)》。

- **连接和表脚本的批量加载** 特定表进行 *version_id* 配对所要求的第一个连接或表脚本会使得所有脚本批量加载到高速缓存中。结果是，通过批量获取所有脚本而不是分别获取这些脚本，从而改善了性能。

- **MobiLink 服务器关机增强** 您可以让 *dbmlstop* 等到 MobiLink 服务器完全关机然后再继续。还可以使用 *dbmlstop* 按名称停止特定的 MobiLink 服务器。

有关详细信息，请参见“[MobiLink 停止实用程序 \(mlstop\)](#)”一节《[MobiLink - 服务器管理](#)》。

- **连接超时** 在指定时间长度内未使用的 MobiLink 数据库连接现在由服务器自动断开连接。超时可以使用 *-ct* (连接超时) 命令行选项来设置。

有关详细信息，请参见“[-ct 选项](#)”一节《[MobiLink - 服务器管理](#)》。

- **并发上传程序的最大数量选项** *-wu* 命令行选项可以设置并发上传所允许的最大工作线程数，从而可以使某些部署的吞吐量提高。

有关详细信息，请参见“[-wu 选项](#)”一节《[MobiLink - 服务器管理](#)》。

安全性

- **MobiLink 用户验证** 使用基于口令的系统来验证用户，使 MobiLink 安装更加安全。现在，您可以使用 *-zu* 在未定义 *authenticate_user* 脚本的情况下自动添加用户。这样，用户模式信息也可用来进行 MobiLink 验证。

有关详细信息，请参见“[MobiLink 用户](#)”《[MobiLink - 客户端管理](#)》。

- **MobiLink 用户管理** *dbmluser* 实用程序已经过扩展，除了可以向系统添加用户之外，还可以从系统中删除用户。此实用程序还作了其它改进。*dbmluser* 命令行选项 *-pf*、*-pp* 和 *-pu* 已不再受支持，并分别由 *-f*、*-p* 和 *-u* 代替。

有关详细信息，请参见“[MobiLink 用户验证实用程序 \(mluser\)](#)”一节《[MobiLink - 服务器管理](#)》。

增强的报告功能

- **统计脚本** MobiLink 现在有了用于跟踪同步统计数据的脚本。收集完毕后，这些同步统计数据可用于对同步性能进行监控。

有关详细信息，请参见“[synchronization_statistics 连接事件](#)”一节《[MobiLink - 服务器管理](#)》、“[synchronization_statistics 表事件](#)”一节《[MobiLink - 服务器管理](#)》、“[upload_statistics 连接事件](#)”一节《[MobiLink - 服务器管理](#)》和“[upload_statistics 表事件](#)”一节《[MobiLink - 服务器管理](#)》。

- **详细的网络错误信息** MobiLink 服务器和客户端现在可显示详细的错误信息以及错误代码，以帮助更好地解决出现的任何错误。您将看到报告错误的网络层、正在执行的网络操作、错误本身以及系统特定的错误代码。
- **在发生错误时远程 Adaptive Server Anywhere 输出日志被发送到 MobiLink 服务器** 当远程日志和 MobiLink 服务器日志都可用于检查时，解决同步时出现的问题是最简单的。当客户端发生错误时，该新功能会将 Adaptive Server Anywhere 远程的输出日志一直发送到 MobiLink 服务器。
有关详细信息，请参见“[-e 选项](#)”一节《[MobiLink - 服务器管理](#)》。
- **日志消息标识工作线程** MobiLink 服务器日志上显示的消息现在可指出记录该消息的工作线程。这样便可以区别由尝试并发同步的同一用户而引发的消息，它也有助于区别同一用户连续同步两次时引发的消息。
- **详细记录** 您可以在 MobiLink 服务器 -v 命令行选项上使用更多的修饰符来配置 MobiLink 服务器日志记录。
有关详细信息，请参见“[-v 选项](#)”一节《[MobiLink - 服务器管理](#)》。
- **被忽略的行将报告到客户端** 如果 MobiLink 服务器由于缺少脚本而忽略任何上载的行，则将会向客户端返回一条消息。该消息会由 Adaptive Server Anywhere 客户端显示为警告，也会在 UltraLite 客户端的 `ignored_rows` 同步参数中显示。
有关详细信息，请参见“[Ignored Rows 同步参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。

易于使用

- **上次下载的时间戳** 上次下载的时间戳将自动写入到 MobiLink 客户端数据库中。
- **自动生成脚本** 可以指示 MobiLink 生成适合用于快照同步的脚本。-za 选项可控制这些脚本的创建和激活。
- **生成示例脚本** 可以指示 MobiLink 生成示例同步脚本。-ze 命令行选项用于控制是否要生成示例脚本。

适应性

- **支持常用的 RDBMS** 作为统一数据库，MobiLink 现在支持 Oracle 8i 和 9i、Microsoft SQL Server 7、Microsoft SQL Server 2000、IBM 的 DB2 等等。
有关详细信息，请参见“[MobiLink 支持的 ODBC 驱动程序](#)”一节《[MobiLink - 服务器管理](#)》。
- **TCP/IP 流中的活动检测** 现在，在 MobiLink 同步过程中使用的基于 TCP/IP 的流在客户端和服务器端都可以接受一个名为 `keep_alive` 的新参数，该参数可启用活动检查。
有关详细信息，请参见“[-x 选项](#)”一节《[MobiLink - 服务器管理](#)》。

UltraLite 的新功能

UltraLite 8.0 引入了下列新功能：

安全性

- **用户验证** 在以前的版本中，UltraLite 数据库没有用以管理访问的用户验证机制。在此版本中，提供了一个内部用户验证机制。与大多数关系数据库管理系统的用户 ID 不同的是，UltraLite 用户 ID 不意味着对表和其它数据库对象的任何所有权。
- **数据库加密** 可以通过加密数据库来提高数据的安全性。这可以通过两种方法实现。
- **高度加密** 可以使用高度加密算法对数据库进行加密，以最大限度地提高安全性。但是，这种安全性可能会导致性能下降。这种加密是基于密钥的，并使用了 AES 128 位算法。
- **数据库模糊处理** 您可以通过对数据库进行模糊处理来提高数据的安全性。如不经模糊处理，使用十六进制编辑器之类的工具就可以查看数据库中的数据。模糊处理可防止偶然查看数据的尝试，但不能提供高度加密那样严密的保护。模糊处理不会像高度加密那样造成性能下降。
有关详细信息，请参见“[保护 UltraLite 数据库](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite Java 应用程序的安全同步** 使用 Certicom 传送层安全的安全同步以前只能在 C/C++ UltraLite 应用程序中才可以进行。现在从 UltraLite Java 应用程序中也可以使用。

同步

- **ActiveSync 同步** Windows CE 设备上的 UltraLite 应用程序可以使用 ActiveSync 进行同步。
有关详细信息，请参见“[向应用程序添加 ActiveSync 同步](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
- **同步更灵活** 添加了多个新功能，可以更高效、灵活地选择要同步的数据：
 - 您可以使用发布将数据划分为不同的集，然后分别同步这些数据集。这样，可以对时效性要求较高的数据（可能通过慢速连接传输）进行高效同步，而其它数据可以在比较方便的时间同步。
 - 只下载同步允许您将只读表添加到 UltraLite 数据库，从而使用只下载同步来有效地同步这些表。
 - 您可以每次标记要同步的表，而无论表中的数据是否已经更改。使用该功能可以维护 UltraLite 客户端上用于控制同步且可由用户配置的信息。
- **全局自动增量缺省列值** 该功能提供了在同步数据库中保持主键唯一性的简单方法。
有关详细信息，请参见“[替换自动增量列的分区大小](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- **UltraLite 生成器的其它控制** 为 ulgen 和 sqlpp 可执行文件添加了新的命令行选项：
 - **脚本版本** 您可以将一个脚本版本与生成的同步脚本进行关联。
 - **日志查询执行计划** 可以导出并在 Interactive SQL 中显示所生成查询的查询执行计划。
- **错误报告** `ul_synch_info` 结构中的 `stream_error` 字段可用于确定同步错误的原因。
有关详细信息，请参见“[Stream Error 同步参数](#)”一节《[UltraLite - 数据库管理和参考](#)》。

数据库管理

- **现有数据库的重新使用** 在 UltraLite 的以前版本中，对数据库应用程序的任何更改都要求对数据库进行重建和同步。而采用这一版本，只要数据库模式没有改变，您就可以继续将 UltraLite

数据库用于应用程序的新版本。对查询的更改本身不要求新数据库，除非它们引用了新列并因此改变了生成的数据库的模式。

- **数据库碎片整理** UltraLite 存储区能够有效地重复使用可用空间，因此在正常情况下不必进行显式碎片整理。对于有非常严格的空间要求的应用程序，提供了显式碎片整理功能。
- **可选择页面大小** 您可以选择使用 2 KB 页面大小来替代缺省的 4 KB 页面大小。

开发功能

- **CodeWarrior 7 支持** CodeWarrior 的 UltraLite 插件现在支持 CodeWarrior 版本 7。
- **eMbedded Visual C++** 支持使用该工具进行开发，并为 CustDB 示例应用程序提供了一个 eMbedded Visual C++ 项目。
- **Palm OS 4.0 和基于文件的数据存储** UltraLite 现在支持 Palm Computing Platform 的 4.0 版本。从 Palm 4.0 开始，引入了各种辅助存储方案。您可以在 Palm 4.0 设备的扩展卡上使用基于文件的 UltraLite 数据存储。
- **Palm Computing Platform 的同步得到改进** Palm Computing Platform 上 HotSync 和 ScoutSync 同步的新简化同步机制与以前的同步机制相比具有多个优点：
 - 启动和退出速度快。
 - 在同步过程中，Palm 设备上不需要额外的存储空间。
 - 应用程序可以同步若干次而无需启动。
 - 不需要指定流参数。

不建议使用 `ULPalmDBStream` 和 `ULConduitStream` 函数。

有关详细信息，请参见“[将 HotSync 同步添加到 Palm 应用程序](#)”一节《[UltraLite - C 及 C++ 编程](#)》。

- **在 Palm Computing Platform 上部署更方便** 您可以给最终用户部署 UltraLite 数据库的初始副本，这样，第一次同步就不必为每个用户下载数据的初始副本。
有关详细信息，请参见“[部署 Palm 应用程序](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
- **Palm 段的处理得到改进** 当为 Palm Computing Platform 开发时，应用程序代码必须分成若干个大小有限的段。

此软件的早期版本中提供的分段方法不允许用户对 UltraLite 生成代码的分段进行控制，从而经常造成所指定的段数过多（可能会降低性能）。新机制生成的段数更少，并允许客户控制段数的指定。

- **嵌入式 SQL 中的 LONG 值** 您可以通过 `DECL_LONGVARCHAR` 和 `DECL_LONGBINARY` 对于 Long 型值（介于 32 KB 和 64 KB 之间）使用主机变量。
- **参考数据库中的分析器挂接** 现在，UltraLite 生成器在分析过程的前后都会激活存储过程。
- **执行计划信息** UltraLite 生成器现在可以输出访问计划，以便用于 UltraLite 应用程序中的查询。此外，您还可以从 Interactive SQL 查看将用于 UltraLite 的执行计划。
- **脚本版本控制** 您可以在 UltraLite 生成器命令行中指定用于同步的脚本版本。
- **更多的 SQL 和 API 功能** UltraLite 应用程序中现在提供了下列功能。

- **支持 @@identity** UltraLite 现在支持 @@identity 全局变量。该功能用于全局自动增量缺省列值的上下文中。在 C++ API 中, 请使用 `ULConnection::GetLastIdentity()` 方法。
- **表中的行数** 您可以使用 `ULTable::GetRowCount()` 方法从 C++ API 编程接口确定表中的行数。嵌入式 SQL 用户可继续使用 `SELECT COUNT(*) FROM table-name` 语句。
- **删除表中的所有行** 您可以使用 `ULTable::DeleteAllRows()` 方法从 C++ API 编程接口删除表中的所有行。嵌入式 SQL 用户可继续使用 `DELETE FROM table-name` 语句。
- **受影响的行数** 您可以使用 `SQLCOUNT` 宏从嵌入式 SQL 确定上一条 `INSERT`、`UPDATE` 或 `DELETE` 语句影响的行数。
- **要上载的行数** 您可以确定需要同步的行数。
有关详细信息, 请参见“[ULCountUploadRows 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
- **上次下载时间** 您可以从 UltraLite 应用程序获取发布的上次下载时间。
有关详细信息, 请参见“[ULGetLastDownloadTime 函数](#)”一节《[UltraLite - C 及 C++ 编程](#)》。
- **其它游标操作** C++ API 的 `ULTable` 类具有额外的方法 (`FindFirst`、`FindNext`、`FindPrevious`、`FindLast`), 可以用来在结果集中查找行。
- **从 DUMMY 系统表中进行查询** 现在支持 `SELECT ...FROM DUMMY` 形式的查询。
- **更新多个表** 多个表上的游标现在可以接受修改多个表的更新。
- **嵌入式 SQL 的 LONG 数据类型的处理得到改进** `DECL_LONGVARCHAR` 和 `DECL_LONGBINARY` 主机变量类型可用于在单个操作中发送或检索超过 32 KB 的数据。
有关详细信息, 请参见“[嵌入式 SQL 中的数据类型](#)”一节《[UltraLite - C 及 C++ 编程](#)》。

SQL Remote 的新功能

- **事件挂接过程** 添加了一组事件挂接过程以实现复制进程进行自定义。通过编写具有指定名称的存储过程, 您可以在消息代理于复制期间采取的操作中的多个位置添加自定义内容。
有关详细信息, 请参见“[SQL Remote 系统过程](#)”一节《[SQL Remote](#)》。

版本 8 中的行为更改

本节列出在 SQL Anywhere Studio 版本 8 的组件中引入的行为更改。

Adaptive Server Anywhere 的行为更改

下面是对该软件以前版本的行为更改。

有关最新的不建议使用和不受支持的功能的列表，请参见“[不再支持和不支持的功能](#)”一节第 318 页。

- **数据库中的 Java 单独授与许可** 其结果是，创建数据库时的缺省行为是排除对数据库中的 Java 的支持。

在 UltraLite 参考数据库中，不再需要数据库中的 Java，因为 UltraLite 生成器已经被更改为使用外部 Java 虚拟机。

有关详细信息，请参见“[SQL Anywhere 11 组件](#)”一节《[SQL Anywhere 11 - 简介](#)》。

- **集合函数和外部引用** Adaptive Server Anywhere 版本 8 遵循新的 SQL/99 标准，明确了集合函数出现在子查询中时的用法。这些更改将影响为该软件的以前版本编写的语句的行为：以前正确的查询现在可能产生错误消息，结果集也可能发生变化。

有关详细信息，请参见“[集合函数和外部引用](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

- **用户提供的选择性估计** Adaptive Server Anywhere 允许您指定显式选择性估计，以指导选择访问计划。在软件选择的访问计划较差时，这些估算作为性能问题的变通解决方法十分有用。新的 user_estimates 连接选项将控制优化程序是使用还是忽略用户提供的选择性估计。

如果您已经使用了这些估计作为性能问题的临时解决方案，则建议将 user_estimates 选项设置为 OFF，因为显式估计可能会变得不准确并可能会强制优化程序选择较差的计划。此版本包括了内部连接算法等查询处理增强，从而大大地提高了查询性能。

有关用户提供的选择性估计的详细信息，请参见“[user_estimates 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[显式选择性估计](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。

- **行排序** 对 8.0 版的查询处理的改进有一个副作用：行排序的确定性较低。当缺少 ORDER BY 子句时，Adaptive Server Anywhere 以最有效的顺序返回行。这意味着结果集的外观可能随您上一次访问行的时间和其它因素的不同而有所变化。确保以特定的顺序返回行的唯一方法是使用 ORDER BY。

LIST 函数是受此更改影响特别大的函数之一。

- **访问计划更改** Adaptive Server Anywhere 的这一版本选择的访问计划与该软件的先前版本相比使用索引的可能性较小。对表扫描效率的改进，以及在比较访问计划开销时使用的更具选择性的开销模型，使得对索引有用性的评估与该软件的先前版本相比更准确。

- **游标更改** 游标增强有一个副作用：该版本中的游标提供的行为与以前相比更接近于定义的标准。这可能会对某些游标产生游标敏感性变化，因为 Adaptive Server Anywhere 提供的行为可更好地符合 ODBC 和其它接口的期望。例如，嵌入式 SQL SCROLL 游标现在禁止预读，这样值更改就反映在游标中。

这种更改可能影响只检查 SQL_SUCCESS 返回代码而不检查 SQL_SUCCESS_WITH_INFO 返回代码的现有应用程序。如果游标行为不同于所要求的行为，检查 SQL_SUCCESS_WITH_INFO 的应用程序将收到警告。警告是 SQLCODE=121, SQLSTATE 01S02。

不敏感的游标是不可更新的。

有关详细信息，请参见“不敏感游标”一节《SQL Anywhere 服务器 - 编程》。

- **存储过程存储** 现在，存储过程在写入时存储。Adaptive Server Anywhere 将创建一个过程的内部表示，用于进行分析。
- **不支持在插入时打开游标** 使用 INSERT 语句打开游标的功能已经被删除。使用 SELECT 语句打开可更新的游标以业界标准的方式提供了相同的功能。
- **用户定义函数** 现在会高速缓存用户定义的函数参数和返回值。如果一个函数在一个 SQL 语句内使用多次，那么高速缓存的参数值可能会导致使用高速缓存的结果，而不是再对该函数进行求值。在以前的版本中，用户定义的函数在每次需要它们时都重新进行求值。新行为提供了更好的性能和更一致的结果，但与该软件的以前版本相比，结果可能会有所更改。
- **NUMBER(*) 函数更改** 限制了 NUMBER 函数的使用，以避免有问题的行为。NUMBER 函数通常的作用是在查询的选择列表中用来提供结果集的顺序行编号，而这种用法仍是允许的。

与之前不同，NUMBER 函数现在可能返回负数，例如，如果您执行值为 -1 的绝对读取，然后通过游标向后移动。新行为对应于 ISO/ANSI 读取偏移。

现在，在许多情况下（例如 WHERE 子句或 HAVING 子句）使用 NUMBER 函数都会出现错误。

有关详细信息，请参见“NUMBER 函数 [Miscellaneous]”一节《SQL Anywhere 服务器 - SQL 参考》。

- **自定义归类更改** 以前，归类实用程序中的 -d 选项接受三个参数；现在只接受两个参数。不再接受 *cust-map-file* 参数。

此外，脚本文件 *collsqmp.sql* 和 *custmap.sql* 不再存在，无法分别用于内置或自定义归类。

对于新创建的数据库，SYSCOLLATIONMAPPINGS 表只包含一个具有归类映射的行。对于用 Adaptive Server Anywhere 的先前版本创建的数据库，此表对每个内置归类都包含一行。

- **触发器名称更改** 触发器名称不再必须在数据库中唯一。它们只需要在其应用的表内唯一。DROP TRIGGER 和 COMMENT ON TRIGGER 的语法也因此改变，这样，如果您也指定了表，则只能指定所有者。这意味着，只用所有者限定触发器的旧脚本现在将产生 [未找到表] 错误。
- **示例数据库中的地址改变** Adaptive Server Anywhere 9.0 示例数据库中的地址也不同于以前的版本中的地址。
- **内部 JDBC 驱动程序的 JAR 文件名改变** 内部 JDBC 驱动程序类现在作为名为 ASAJRT（而不是 ASAJDBC）的 JAR 文件安装。
- **RESTORE DATABASE 语句权限** 执行 RESTORE DATABASE 语句时不再需要与实用程序数据库进行连接。执行 RESTORE DATABASE 语句需要的权限由 -gu 命令行选项进行控制。

有关详细信息，请参见“RESTORE DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

- **为 TDS 连接返回空的字符串时返回 NULL 字符串** `tds_empty_string_is_null` 选项控制服务器为 TDS 连接返回空字符串时返回包含一个空白字符的字符串还是返回一个 NULL 字符串。
有关详细信息，请参见“[tds_empty_string_is_null 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **COMMENT 语句改变** 以前，COMMENT ON INDEX 的语法包含了一个可选的索引所有者名称。现在索引名称还可以包含所有者和表。而现在 COMMENT ON INDEX 的语法为
COMMENT ON INDEX [[owner.]table.]index-name IS comment
有关详细信息，请参见“[COMMENT 语句](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **缺省情况下启用字符集转换** 在 Adaptive Server Anywhere 的先前版本中，字符集转换功能在缺省情况下是关闭的，您必须指定 `-ct` 命令行选项来启用字符集转换。现在，字符集转换功能在缺省情况下是启用的，但可以使用 `-ct-` 命令行选项加以禁用。
当服务器判断出连接的字符集与数据库的字符集不同时，服务器就会对该连接发送到服务器或从服务器发出的所有字符串应用字符集转换。
当服务器判断出数据库和连接具有等同的字符集时，它就会对连接禁用字符集转换。
在大多数情况下都应该启用字符集转换功能。当二进制数据被插入到数据库并作为字符数据读取或者执行相反操作时，可能会发生一个行为更改。在这种情况下，返回的数据可能不会与输入时完全相同，因为服务器只对字符数据应用字符集转换。为避免这一问题，应用程序不应该使用二进制类型发送或读取字符数据。
- **CONVERT、timestamp_format 和 date_format** 在使用 `timestamp_format` 或 `date_format` 选项时，如果您以大小写混合形式（如 `Mmm`）指定一个字符符号，Adaptive Server Anywhere 将选择适合正在使用的语言的大小写。此外，现在 `CONVERT` 函数还将字符日期转换成适合正在使用语言的大小写。例如，在英语中正确的大小写是 `May`，而在法语中是 `mai`。
有关详细信息，请参见“[date_format 选项 \[数据库\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》、“[timestamp_format 选项 \[兼容性\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》和“[CONVERT 函数 \[Data type conversion\]](#)”一节《[SQL Anywhere 服务器 - SQL 参考](#)》。
- **更改为三值的布尔逻辑** 两值布尔逻辑只适用于 `expr = NULL` 的情况，其中 `expr` 是指基列或基列上的一个表达式。否则，将应用三值逻辑。`ansinull` 选项现在只影响查询的 WHERE 子句中的这一特定情况。
- **Sybase Central 和 Interactive SQL 接受 COMMLINKS 连接参数** 在 Adaptive Server Anywhere 的以前版本中，Sybase Central 和 Interactive SQL (dbisql 实用程序) 忽略了 COMMLINKS 连接参数。Sybase Central 和 Interactive SQL 现在接受这一参数。
由于这一更改，某些连接字符串的行为可能与在 Adaptive Server Anywhere 的先前版本中不同。具体来说，如果您不提供 `COMMLINKS=tcpip`，Interactive SQL 和 Sybase Central 将不会查找网络上的服务器。
有关详细信息，请参见“[CommLinks 连接参数 \[LINKS\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **客户端忽略 SQLLOCALE 环境变量** 客户端可以使用 `CharSet` 连接参数指定在连接时使用的字符集。在 Adaptive Server Anywhere 的旧版本中，`SQLLOCALE` 环境变量的 `CHARSET` 参数用于

在未提供 CharSet 连接参数的情况下更改客户端的缺省字符集。客户端现在忽略 SQLLOCALE 环境变量。

- **不受支持的字符集导致连接失败** 客户端可以使用 CharSet 连接参数指定在连接时使用的字符集。然而，如果服务器不支持请求的字符集，连接就会失败。当客户端请求 Adaptive Server Anywhere 旧版本不支持的字符集时，连接会成功，但会出现一个警告。如果客户端未指定字符集，而且客户端的本地字符集不受服务器支持，那么连接也会成功，但会出现一个字符集不受支持的警告。

这种现象会在版本 8 的客户端连接到 6.x、7.x 和 8 版本数据库服务器时发生。

- **更改了缺省数据包大小** 客户/服务器通信的缺省数据包大小已从 1024 字节更改为 1460 字节。有关数据包大小的详细信息，请参见“CommBufferSize 连接参数 [CBSIZE]”一节《SQL Anywhere 服务器 - 数据库管理》和“-p 服务器选项”一节《SQL Anywhere 服务器 - 数据库管理》。

- **dbdsn 实用程序只管理 Adaptive Server Anywhere 数据源** 用于管理 Adaptive Server Anywhere ODBC 数据源的 dbdsn 实用程序现在被显式限制为只管理 Adaptive Server Anywhere 数据源。

- **login_procedure 选项要求 DBA 权限** login_procedure 选项只能由具有 DBA 特权的用户进行设置。在 Adaptive Server Anywhere 的旧版本中，设置此选项是不需要 DBA 权限的。具有 DBA 权限的用户可以更改其他用户的此选项的设置，但没有 DBA 权限的用户无法更改他们自己的此选项设置。由于这种更改，DBA 可以确保当用户进行连接时，如有必要，将执行公用过程。

有关详细信息，请参见“login_procedure 选项 [数据库]”一节《SQL Anywhere 服务器 - 数据库管理》。

- **ESTIMATE_SOURCE 返回新值** ESTIMATE_SOURCE 函数返回比以前更详细的值。

有关详细信息，请参见“ESTIMATE_SOURCE 函数 [杂类]”一节《SQL Anywhere 服务器 - SQL 参考》。

不再支持和不支持的功能

此列表包括不再支持和影响现有应用程序的功能。

- **不支持 NetWare 4.10** 仍支持 Novell NetWare 4.11 及更高版本。不支持 3.x 和 4.10 版。
- **不支持 NetBios** 不再支持 NetBios 端口。如果您使用 NetBios，则应切换到 TCP/IP 或 SPX。
- **不支持 IPX** 不再支持 IPX 端口。如果您使用 IPX，则应切换到 SPX 或 TCP/IP。
- **不再支持的归类** 不再支持以下归类。如下表所示，它们已经被其它归类所取代：

不再支持	取代归类
437	437LATIN1
850	850LATIN1
852	852LATIN2

不再支持	取代归类
860	860LATIN1
863	863LATIN1
865	865NOR
SJIS	932JPN
SJIS2	932JPN
WIN_LATIN1	1252LATIN1
WIN_LATIN5	1254TRK
Internal	850LATIN1
437EBCDIC	

- **不再支持 -e 选项** 不再支持用于对客户端/服务器通信进行加密的 -e 命令行选项及数据源实用程序中的 -e 选项。-ec 选项取代了它们。在服务器上，-ec **simple** 与 Adaptive Server Anywhere 先前版本中的 -e 使用相同的加密算法。
- **不建议使用 None 参数** 不再支持 isql_plan 选项的 None 参数。现在，查询优化计划出现在 [结果] 窗格中的 [计划] 选项卡上。单击 [计划] 选项卡时，总是出现一个计划。以前，计划出现在 [消息] 窗格中。
- **不再支持 WITH HASH SIZE n 子句** 不再支持 WITH HASH SIZE 子句。
- **不建议使用 max_work_table_hash_size 选项** 不再支持 max_work_table_hash_size 选项。
- **不建议使用 max_hash_size 选项** 不再支持 max_hash_size 选项。
- **不再支持 SATMP 环境变量** 不再支持 UNIX 版本的 Adaptive Server Anywhere 用来指示临时文件所在目录的 SATMP 环境变量。在 UNIX 上，ASTMP 环境变量可用于指示保存临时文件的位置。
有关详细信息，请参见“SATMP 环境变量”一节《SQL Anywhere 服务器 - 数据库管理》。
- **删除了 dbtran -id 选项** 此软件中没有提供 dbtran 实用程序中的 -id 命令行选项。

MobiLink 的行为更改

- **MobiLink Adaptive Server Anywhere 客户端安装** MobiLink 客户端现在使用发布和同步预订而不是使用同步定义进行配置。
有关详细信息，请参见“SQL Anywhere 客户端”《MobiLink - 客户端管理》。
- **上次下载时间戳参数更改了脚本** 在许多脚本中添加了一个新参数，使得基于时间戳的同步更容易实现。在现有脚本作为第一参数提供时，新参数会将这些现有脚本分成许多脚本。要继续使

用现有的脚本，请更改行为，通过提供 MobiLink 服务器命令行选项 `-zd` 将上次下载时间戳作为最后一个参数提供。

- **MobiLink 关机** 以前，发自远程连接的 `dbmlstop` 命令会导致 MobiLink 服务器关闭。现在，只有发自与 MobiLink 服务器相同的计算机的 `dbmlstop` 请求才会导致 MobiLink 服务器关闭。不再需要允许 `dbmlstop` 停止服务器的 `-zs` 选项。
- **TCP/IP 流中的活动检测的缺省设置已经改变** `keep_alive` 的缺省设置现在是 1（启用）。
- **MobiLink 可以隐藏 dbmluser 信息** 使用 `dbmluser` 命令行实用程序时显示的信息量，如时间戳、版权和其它 MobiLink 服务器消息在缺省情况下不再出现。
- **MobiLink 用户验证** 如果您不使用 MobiLink 用户验证，那么必须在 MobiLink 服务器命令上使用 `-zu+` 选项。
- **缺省日志扩展名现在是 .mls** 现在每个文件都命名为 DDMMYYNN.MLS，其中 DD 是日期，MM 是月份，YY 是本世纪年份。NN 是序列号，第一个文件的编号为 1。
- **不再支持 dbmlsync StreamCompression 扩展选项** 现在这个选项被忽略。

UltraLite 的行为更改

- **Palm 应用程序需要的代码改变** 您的代码必须指定对于 Palm Computing Platform 4.x 版是使用基于记录的标准数据库存储还是使用基于文件的扩展卡存储。在调用 `ULPalmLaunch`（嵌入式 SQL）或 `ULData.PalmLaunch` (C++ API) 之前必须添加一个函数调用。函数调用格式如下：

```
ULEnablePalmRecordDB( &sqlca );
```

或

```
ULEnableFileDB( & sqlca );
```

如果您使用基于记录的存储，则应提供 `ULEnablePalmRecordDB`，如果使用基于文件的存储，则应提供 `ULEnableFileDB`。如果设备不支持基于文件的存储，`ULPalmLaunch` 就会设置 `SQLCODE -82`。

UltraLite 不再支持以下环境和/或功能：

- **DOS 目标平台** DOS 不再是受支持的平台。
- **Metrowerks CodeWarrior 5 开发** CodeWarrior 6 现在是 UltraLite 开发所必需的。
- **不再支持 Palm 2.x** UltraLite 不再支持针对 Palm OS 2.x 设备（如 PalmPilot Professional）的开发。需要 3.0 或更高版本。
- **不再支持 ULPalmDBStream 和 ULConduitStream** Palm Computing Platform 上 HotSync 或 ScoutSync 同步的新同步流意味着 `ULPalmDBStream` 和 `ULConduitStream` 函数已过时。它们仍被接受，但不起作用。
- **UltraLite 生成器使用外部 Java 虚拟机** UltraLite 分析器现在在数据库引擎的外部运行，因此可以对照参考数据库使用，即使它们不具备 Java 功能。

- **UltraLite JDBC 包名称改变** UltraLite JDBC 函数的数据包名已由 **com.sybase.asa.ultralite.jdbc** 更改为 **ianywhere.ultralite.jdbc**。这要求对用于 UltraLite 应用程序的 `import` 语句进行更改。
 - **所有更改必须在下载同步之前提交** 只下载同步不再是异常，所有更改都必须在同步之前提交。
- 您还应检查 Adaptive Server Anywhere 的行为更改，因为某些更改可能会对应用程序产生影响。

升级到 SQL Anywhere 11

目录

升级 SQL Anywhere	324
升级 MobiLink	337
升级 QAnywhere	344
升级 UltraLite	345
升级 SQL Remote	353

升级 SQL Anywhere

请务必先查看行为更改列表以确定应用程序是否受到影响，然后再在此版本软件中使用现有应用程序。请参见 [SQL Anywhere 11 - 更改和升级第 1 页](#)。

注意

Adaptive Server Anywhere 已更名为 SQL Anywhere。在本章中，使用 SQL Anywhere 指代所有版本。

升级版本 10 及更高版本数据库

如果从 10 或更高版本升级，可以使用升级实用程序也可以重建数据库。升级或重建是可选步骤，因为版本 11 的软件可以和版本 10 的数据库一起使用。不过，如果要利用版本 11 中的所有新功能，就必须重建数据库。请参见“[升级版本 10 及更高版本数据库](#)”一节第 333 页和“[重建版本 10 及更高版本数据库](#)”一节第 327 页。

具有实例化视图的数据库

升级数据库服务器后，或使用已升级的数据库服务器重建或升级数据库后，建议您刷新数据库中的实例化视图。请参见“[刷新手动视图](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

升级版本 9 及更早版本的数据库

如果从版本 9 或更早版本升级到 SQL Anywhere 版本 11，必须重建数据库，即卸载旧的数据库然后将其重装到新的版本 11 数据库中。尝试启动版本 9 或更早版本的数据库将导致数据库启动时发生错误。重建现有数据库有四种方法：

- 使用版本 11 的卸载实用程序 (dbunload) 并使用 -an (新建数据库) 或 -ar (替换旧数据库) 选项。请参见“[使用卸载实用程序重建版本 9 或更早版本的数据库](#)”一节第 331 页。

注意

在所有版本的 SQL Anywhere 中，卸载实用程序 (dbunload) 的文件名均相同。必须确保所使用的版本正确。运行 dbunload -? 命令可确定正在使用哪个版本的卸载实用程序。请参见“[使用实用程序](#)”一节第 326 页。

此外，在将 dbunload 用于 10 或更高版本的数据库时，所使用的 dbunload 的版本必须与用于访问数据库的数据库服务器的版本相匹配。如果将较旧版本的 dbunload 和相对较新版本的数据库服务器一起使用，会报告出现错误，反之亦然。

- 使用 Sybase Central 中的 [[卸载数据库向导](#)]。可以选择创建新的数据库、用新数据库替换现有数据库或者将数据库卸载到文件中。请参见“[从 Sybase Central 重建版本 9 或更早版本的数据库](#)”一节第 330 页。
- 使用 11 版卸载实用程序卸载数据库，然后在 11 版的数据库服务器上使用 *reload.sql* 文件重装数据库。
如果需要更改模式，建议您使用这种升级方法。更改模式之后，可以初始化一个新数据库然后对其应用重装脚本。
- 使用较低版本的 dbunload 卸载数据库，然后使用 *reload.sql* 文件和 11 版的数据库服务器重装数据库。只有在其它方法失败后方可使用这种方法，因为会将不建议使用或不支持的数据库选项

设置、对象或 SQL 语法卸载到 *reload.sql* 文件中。如果在重装过程中发生问题，必须手动编辑该文件。版本 11 及更高版本的内部重装功能可以解决其中的许多问题。

重建 Mac OS X 数据库

PPC 上支持用于 Mac OS X 的 SQL Anywhere 9.0.2 版，而 Intel 上支持用于 Mac OS X 的 SQL Anywhere 10.0.0 及更高版本。如果在 Mac OS X 上有 9.0.2 版或更早版本的数据库，可以使用两种方法卸载数据库：

- 使用 9.0.2 版软件卸载数据库。
- 将数据库复制到安装了 SQL Anywhere 11 的其它平台上，然后使用版本 11 软件卸载数据库。

卸载数据库后，可以使用版本 11 软件在 Mac OS X 上执行重装。

如果要在卸载和重装过程中更改数据库的特性（例如，将区分大小写的数据库更改为不区分大小写的数据库），则此过程会更为复杂。有关详细信息，请参见“重建数据库”一节《SQL Anywhere 服务器 - SQL 的用法》。

与现有软件的兼容性

- SQL Anywhere 11 数据库服务器支持来自使用版本 6.0.0 或更高版本软件的客户端应用程序的连接。版本 5 及更低版本的客户端不能连接到版本 11 数据库服务器。若版本 9 及更低版本的客户端连接到版本 11 数据库服务器，则它们不能使用以下功能：
 - Kerberos 登录
 - 嵌入式 SQL NCHAR 类型
 - 对来自 ODBC、OLE DB 和 ADO.NET 的 Unicode 数据的改进支持（例如，将 NCHAR 描述为 WCHAR 列）
 - 针对使用 ODBC 的 Unicode 应用程序和使用 OLE DB 及 ADO.NET 的所有应用程序的 BLOB 性能增强
 - 主要提高 WAN 性能但同时也提高了 LAN 性能增强功能

为共享内存连接设置 SATMP

在版本 11 与版本 9（及更低版本）中，查找临时文件所用的搜索顺序不同。如果要通过共享内存将版本 9 或更低版本的客户端连接到版本 11 数据库服务器，则必须设置 SATMP（版本 11）和 ASTMP（版本 9 及更低版本）环境变量，以指定临时文件的位置。如果不设置这些环境变量，则共享内存连接尝试将会失败。

- 您可以按如下方式从当前版本的 Sybase Central 管理旧数据库和旧数据库服务器：
 - 可以连接并管理在版本 9 及更高版本数据库服务器上运行的版本 9 及更高版本的数据库。
 - 可以连接到在安装有 Sybase Central 的计算机上运行的版本 5 或更高版本的数据库，以使用 Sybase Central 中的 [卸载数据库向导] 来重建该数据库。数据库在卸载之前需停止运行。
 - 不支持在版本 7 及更低版本数据库服务器上运行的版本 7 及更低版本数据库。

使用实用程序

如果您的计算机上有多个版本的 SQL Anywhere，则在使用实用程序时必须注意系统路径。由于安装过程会将最近安装版本的可执行文件目录添加到系统路径的末尾，因此安装了该软件的新版本后，仍然有可能会出现在无意中运行以前安装的版本的情况。

例如，如果在路径中 Adaptive Server Anywhere 版本 8 可执行文件目录位于 SQL Anywhere 11 可执行文件目录之前，并且使用了 dbinit 命令，则将使用版本 8 实用程序，因此会创建版本 8 数据库。

有以下几种方法可确保您使用版本 11 实用程序：

- 修改系统路径，以使 SQL Anywhere 11 可执行文件目录位于任何旧版本的可执行文件目录之前。
- 在执行命令之前，将路径更改为 SQL Anywhere 11 可执行文件目录。
- 为实用程序名指定一个全限定路径名，该名称可指示要运行的实用程序的准确位置。
- 创建脚本更改环境，以使用正确版本的实用程序。
- 卸载旧软件。

重要的升级预防措施

在升级 SQL Anywhere 之前，应采取以下几项预防措施：

- **检查行为更改** 请确认记录的任何行为更改都不影响您的应用程序。如果影响，则必须更新应用程序。请参见“[11.0.1 版中的新功能](#)”第 1 页。
- **测试应用程序** 在升级任何用于生产环境的应用程序之前，请在 SQL Anywhere 11 环境中对您的应用程序进行全面测试。
- **使用正确版本的实用程序** 确保为新数据库使用正确版本的数据库实用程序。请参见“[使用实用程序](#)”一节第 326 页。
- **校验和备份数据库** 在开始升级前，请先校验数据库，然后对软件 and 数据库进行备份。为确保将来能够恢复，请在升级结束时完成对数据库的备份。
- **升级前同步** 对于同步时涉及到的数据库（如 MobiLink 安装中的 UltraLite 数据库或 SQL Anywhere 远程数据库），必须在升级之前成功执行一次同步。
- **测试升级过程** 先仔细测试升级过程，然后再在生产系统中执行升级过程。

SQL Anywhere 用于多种不同配置，无法保证升级准则适用于所有情况。

升级快速入门

针对使用旧版本软件的用户，以下步骤概括了将数据库升级到版本 11 的过程。

◆ 升级数据库（命令行）

1. 备份数据库。例如：

```
dbbackup -c "DBF=mydb.db;UID=DBA;PWD=sql" old-db-backup-dir
```

有关详细信息，请参见“备份快速入门”一节《SQL Anywhere 服务器 - 数据库管理》。

2. 如果可能，请对要存储新数据库的驱动器执行碎片整理，因为如果驱动器有太多碎片，会降低数据库性能。
3. 关闭所有 SQL Anywhere 和 Adaptive Server Anywhere 数据库服务器，因为版本 11 的 dbunload 实用程序不能用于在之前版本的数据库服务器上运行的数据库。例如：

```
dbstop -c "DBF=mydb.db;UID=DBA;PWD=sql"
```

4. 卸载旧的数据库并将其重装（重建）到新的版本 11 数据库。例如：

```
dbunload -c "DBF=mydb.db;UID=DBA;PWD=sql" -an mydb11.db
```

5. 使用前备份新数据库。例如：

```
dbbackup -c "DBF=mydb11.db;UID=DBA;PWD=sql" new-db-backup-dir
```

6. 使用前校验新数据库。例如：

```
dbvalid -c "DBF=mydb11.db;UID=DBA;PWD=sql"
```

另请参见

- “使用卸载实用程序重建版本 9 或更早版本的数据库”一节第 331 页
- “从 Sybase Central 重建版本 9 或更早版本的数据库”一节第 330 页

重建版本 10 及更高版本数据库

重建数据库包括卸载和重装数据库以升级其文件格式。升级文件格式时，会更改用于存储和访问磁盘上数据的文件格式，使您可以使用最新版本软件的所有新功能和增强性能。

小心

卸载并重装大数据库既耗时又需要大量磁盘空间。此过程大致需要相当于数据库两倍大小的磁盘空间，用于保存已卸载的数据和新的数据库文件。

如果要重建的数据库是 SQL Remote 复制中涉及的数据库或 MobiLink 安装中的远程数据库，并且使用 dbunload 实用程序，必须确保使用 -ar 或 -an 选项。此选项确保将新数据库的事务日志偏移设置为与旧数据库的事务日志偏移匹配。

在使用 dbunload 卸载版本 10 或更高版本的数据库时，所使用的 dbunload 的版本必须与用于访问数据库的数据库服务器的版本相匹配。如果将较旧版本的 dbunload 和相对较新版本的数据库服务器一起使用，会报告出现错误，反之亦然。

由于 SQL Anywhere 中索引的更改，当通过卸载和重装来重建数据库时，重建的数据库可能会比原始数据库小。数据库大小的减小并不表示出现了问题或有数据丢失。

注意

建议您先备份数据库，然后再重建数据库。

重装带有自动增量列的表

通过指定 `dbunload -l` 选项，可在重建的数据库中保留自动增量列的下一个可用值。此选项会针对包含自动增量值的每个表，将对 `sa_reset_identity` 系统过程的调用添加到生成的 `reload.sql` 脚本中，从而保留 `SYSTABCOL.max_identity` 的当前值。

重建数据库

◆ 重建数据库 (Sybase Central)

1. 执行升级软件的标准预防措施。请参见“重要的升级预防措施”一节第 326 页。
2. 选择 [开始] » [程序] » [SQL Anywhere 11] » [Sybase Central]。
3. 启动运行有要升级的数据库的版本 11 数据库服务器，然后通过 Sybase Central 连接到该数据库。
4. 选择 [工具] » [SQL Anywhere 11] » [卸载数据库]。
5. 阅读 [卸载数据库向导] 第一页的内容，然后单击 [下一步]。
6. 选择 [卸载在当前版本服务器上运行的数据库]，然后从列表中选择数据库。单击 [下一步]。
7. 选择卸载并重装到一个新的数据库。单击 [下一步]。
8. 为数据库指定一个新的文件名。
9. 还可以为新数据库指定页面大小，但指定的页面大小不能大于数据库服务器的页面大小。缺省的页面大小是 4096 个字节。如果愿意，可以加密数据库文件。如果选择高度加密，那么每次启动数据库时都需要加密密钥。单击 [下一步]。
有关数据库文件加密的详细信息，请参见“加密和解密数据库”一节《SQL Anywhere 服务器 - 数据库管理》。
10. 选择 [卸载结构和数据]。您也可以为数据库选择任何其它您需要的选项。单击 [下一步]。
11. 选择 [卸载所有数据库对象]。单击 [下一步]。
12. 指定是否希望在卸载/重装完成时连接到新数据库。
13. 单击 [完成] 开始此过程。应检查新数据库以确认重建已正确完成。

有关使用 [卸载数据库向导] 的详细信息，请参见“使用 [卸载数据库向导] 导出数据”一节《SQL Anywhere 服务器 - SQL 的用法》。

◆ 重建数据库 (命令行)

1. 执行升级软件的标准预防措施。请参见“重要的升级预防措施”一节第 326 页。
2. 请确保您对要升级的数据库有独占访问权，同时确保系统路径中版本 11 实用程序的路径位于其它实用程序的路径之前。请参见“使用实用程序”一节第 326 页。
3. 使用 `-an` 选项运行卸载实用程序 (`dbunload`)，以创建新数据库。

```
dbunload -c "connection-string" -an new-db-file
```

在 *connection-string* 中指定的数据库用户必须对要重建的数据库具有 DBA 权限。

此命令将创建新数据库。如果想用升级后的数据库替换现有的数据库，则使用 `-ar` 选项代替 `-an` 选项。要使用 `-ar` 选项，必须连接到个人数据库服务器或与卸载实用程序 (dbunload) 位于同一计算机上的网络数据库服务器。

有关其它卸载实用程序 (dbunload) 选项的信息，请参见“[卸载实用程序 \(dbunload\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

4. 先关闭数据库并将事务日志存档，然后再使用重装的数据库。

如果要在卸载和重装过程中更改数据库的特性（例如，将区分大小写的数据库更改为不区分大小写的数据库），则此过程会更为复杂。请参见“[重建数据库](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。

为版本 11 重建版本 9 及更早版本的数据库

本节介绍如何卸载数据库并将其重装到新的版本 11 数据库中。

有关升级 Windows Mobile 数据库的信息，请参见“[在 Windows Mobile 上重建数据库](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

重建 Mac OS X 数据库

PPC 上支持用于 Mac OS X 的 SQL Anywhere 9.0.2 版，而 Intel 上支持用于 Mac OS X 的 SQL Anywhere 10.0.0 及更高版本。如果在 Mac OS X 上有 9.0.2 或更早版本的数据库，可以使用两种方法卸载数据库：

- 使用 9.0.2 版软件卸载数据库。
- 将数据库复制到安装了 SQL Anywhere 11 的其它平台上，然后使用版本 11 软件卸载数据库。

卸载数据库后，可以使用版本 11 软件在 Mac OS X 上执行重装。

小心

卸载并重装大数据库既耗时又需要大量磁盘空间。此过程需要访问大小约为数据库两倍的磁盘空间，以容纳已卸载的数据和新的数据库文件。

升级限制

使用版本 11 工具重建版本 9 或更早版本的数据库时，需要注意以下限制：

- 必须断开此数据库与任何更低版本的数据库服务器之间的连接，同时必须关闭在计算机上运行的任何更低版本的数据库服务器。还必须关闭在计算机上运行的任何版本 11 数据库服务器。如果 dbunload 因检测到上述任一情况而无法继续运行，它会发出一条错误消息并失败。
- 对于旧数据库，不要在 dbunload 连接字符串中包含 ENG、START 或 LINKS 连接参数（在 `-c` 选项中指定）。如果指定这些参数，它们将被忽略并出现一条警告。在 Sybase Central 的 [连接] 窗口中，不要在 [服务器名] 和 [启动行] 字段中输入任何值。

- 必须在能够对旧数据库进行直接文件系统访问的计算机上运行 dbunload (dbunload 必须能够使用共享内存连接到数据库)。
- 不能在正在重建的计算机上运行名为 dbunload_support_engine 的数据库服务器。

特殊注意事项

- **口令区分大小写** 在新建的 SQL Anywhere 11 数据库中，所有口令均区分大小写，与数据库是否区分大小写无关。新数据库的缺省 DBA 口令是 sql。

重建现有数据库时，SQL Anywhere 按以下条件确定口令的大小写：

- 如果口令最初是在不区分大小写的数据库中输入的，则目前的口令依然不区分大小写。
 - 如果口令最初是在区分大小写的数据库中输入的，则大写和混合大小写的口令依然区分大小写。不过，如果口令全部以小写输入，则口令不区分大小写。
 - 对现有口令和新口令进行的更改均区分大小写。
- **页面大小** SQL Anywhere 11 数据库的缺省数据库页面大小为 4096 个字节。版本 11 中支持的页面大小为 2048 个字节、4096 个字节、8192 个字节、16384 个字节和 32768 个字节。如果旧数据库使用不受支持的页面大小，则缺省情况下新数据库的页面大小为 4096 字节。可以使用 dbinit -p 选项或 dbunload -ap 选项指定不同的页面大小。请参见“初始化实用程序 (dbinit)”一节《SQL Anywhere 服务器 - 数据库管理》和“卸载实用程序 (dbunload)”一节《SQL Anywhere 服务器 - 数据库管理》。
 - **归类** 在版本 9 及更早版本中，SQL Anywhere 支持一个与 CHAR 数据类型共同使用的归类。此归类使用 SQL Anywhere 归类算法 (SQL Anywhere Collation Algorithm, 简称 SACA)。版本 10 及更高版本的 SQL Anywhere 支持两种归类算法，即 SACA 和 UCA (Unicode Collation Algorithm, Unicode 归类算法)。除非为重建的数据库指定了一个新的或不同的归类，否则旧数据库中的 SACA 归类会被卸载并在重建的数据库中重新使用。

如果要重建具有自定义归类的数据库，则仅当执行单步重建 (内部卸载) 时，该归类才能够保留。如果选择卸载数据库，然后将模式和数据装载到您创建的数据库中，则必须使用所提供归类中的一种。请参见“支持的归类和替代归类”一节《SQL Anywhere 服务器 - 数据库管理》。

- **数据库文件大小** 由于 SQL Anywhere 中索引的更改，当通过卸载和重装来重建数据库时，重建的数据库可能会比原始数据库小。数据库大小的减小并不表示出现了问题或有数据丢失。

从 Sybase Central 重建版本 9 或更早版本的数据库

可以使用 [卸载数据库向导] 重建旧的数据库。使用此向导可将数据库卸载到重装文件和数据文件、卸载并重装到新的数据库或卸载并重装到现有数据库。强烈建议您先备份数据库，然后再重建数据库。

Sybase Central 升级注意事项

- 数据库文件必须位于安装 SQL Anywhere 11 的计算机上。
- 不能卸载数据库中表的子集。必须使用 dbunload 实用程序完成此操作。
- 如果 [卸载数据库向导] 确定数据库文件已运行，则在数据库停止之后才会继续进行卸载。

◆ 重建数据库 (Sybase Central)

1. 执行升级软件的标准预防措施。请参见“重要的升级预防措施”一节第 326 页。
2. 如果可能，请对要存储新数据库的驱动器执行碎片整理，因为如果驱动器有太多碎片，会降低数据库性能。
3. 确保对要卸载并重装的数据库具有独占访问权。禁止任何其他用户连接到数据库。
4. 选择 [开始] » [程序] » [SQL Anywhere 11] » [Sybase Central]。
5. 选择 [工具] » [SQL Anywhere 11] » [卸载数据库]。
6. 阅读 [卸载数据库向导] 介绍页的内容，然后单击 [下一步]。
7. 选择 [卸载在较早版本服务器上运行的数据库或当前未运行的数据库]。输入数据库的连接信息。单击 [下一步]。
8. 选择 [卸载并重装到一个新的数据库]。单击 [下一步]。
9. 为数据库指定一个新的文件名。单击 [下一步]。
可以为新数据库指定页面大小。在版本 11 中，缺省（推荐使用）的页面大小为 4096 个字节。
如果需要，可以加密数据库文件。如果选择高度加密，那么每次启动数据库时都需要加密密钥。请参见“加密和解密数据库”一节《SQL Anywhere 服务器 - 数据库管理》。
10. 选择卸载结构和数据。单击 [下一步]。
11. 指定重建完成后是否要连接到新建的数据库。
12. 单击 [完成]。检查新数据库以确认是否正确完成重建过程。

使用卸载实用程序重建版本 9 或更早版本的数据库

可以使用卸载实用程序 (dbunload) 并使用 -an 或 -ar 选项重建旧的数据库：

- 建议使用 -an 选项，因为它会创建一个新的数据库而保留原始数据库不变。
- 若使用 -ar 选项，则会用新的版本 11 数据库替换旧数据库。

建议重建数据库之前先对其进行备份。

注意

数据库的页面大小（以字节为单位）可以是 2048、4096、8192、16384 或 32768，缺省值为原数据库的页面大小。

◆ 重建数据库（命令行）

1. 执行升级软件的标准预防措施。请参见“重要的升级预防措施”一节第 326 页。
2. 确保系统路径中版本 11 实用程序在其它实用程序之前。请参见“使用实用程序”一节第 326 页。

3. 关闭所有 SQL Anywhere 和 Adaptive Server Anywhere 数据库服务器，因为版本 11 的 dbunload 实用程序不能用于在之前版本的数据库服务器上运行的数据库。例如：

```
dbstop -c "DBF=mydb.db;UID=DBA;PWD=sql"
```

4. 如果可能，请对要存储新数据库的驱动器执行碎片整理，因为如果驱动器有太多碎片，会降低数据库性能。
5. 备份数据库。例如：

```
dbbackup -c "DBF=mydb.db;UID=DBA;PWD=sql" old-db-backup-dir
```

请参见“备份快速入门”一节《SQL Anywhere 服务器 - 数据库管理》。

注意

确保使用正确版本的 dbbackup 来备份数据库。请参见“使用实用程序”一节第 326 页。

6. 使用 -an 或 -ar 选项运行卸载实用程序 (dbunload)，以创建新数据库。

```
dbunload -c "connection-string" -an database-filename
```

例如：

```
dbunload -c "DBF=mydb.db;UID=DBA;PWD=sql" -an mydb11.db
```

在 *connection-string* 中指定的数据库用户必须使用 DBA 特权连接到要卸载的数据库。此命令创建一个新的数据库（通过指定 -an）。如果指定 -ar 选项，则现有数据库会被重建后的数据库所替换。若要使用 -ar 选项，必须连接到个人数据库服务器，或卸载实用程序 (dbunload) 所在计算机上的网络数据库服务器。

有关卸载实用程序 (dbunload) 其它可用选项的信息，请参见“卸载实用程序 (dbunload)”一节《SQL Anywhere 服务器 - 数据库管理》。

已知问题

如果运行 dbunload 或 [卸载数据库向导] 时重建过程失败，则可以使用以下步骤来帮助诊断失败原因。

◆ 诊断重建失败

1. 在旧数据库上运行 dbunload -n。

```
dbunload -c "connection-string" -n directory-name
```

2. 创建一个新的版本 11 空数据库。

```
dbinit test.db
```

3. 将 *reload.sql* 文件应用到空数据库。

```
dbisql -c "DBF=test.db;UID=DBA;pwd=sql" reload.sql
```

4. 根据在将 *reload.sql* 文件应用到新数据库时所收到的消息，对 *reload.sql* 文件或原始数据库进行更改。

下表列出了导致重建失败的已知问题及其解决方案。

已知问题	解决方案
如果使用所有者的名称作为表名的前缀，则过程或触发器中的 DECLARE LOCAL TEMPORARY TABLE 语句会引起语法错误。	删除所有者名称。
如果 CREATE TRIGGER 语句不包含定义了触发器的表的所有者名称，并且在执行 <i>reload.sql</i> 文件的用户引用该表时必须使用所有者名称限定该表，则语句执行会失败，并返回错误，说明找不到该表。	将所有者名称作为表名的前缀。
如果对象名（比如表、列、变量或参数名）与 SQL Anywhere 高版本中所引入的保留字（比如 NCHAR）相同，则重装失败。例如： <pre>CREATE PROCEDURE p () BEGIN DECLARE NCHAR INT; SET NCHAR = 1; END;</pre>	将所有对保留字的引用改为使用不同的名称。对于变量名，将 @ 作为名称前缀是防止命名冲突的一个通用约定。 有关保留字的完整列表，请参见“保留字”一节《SQL Anywhere 服务器 - SQL 参考》。
如果使用 dbunload 的版本 9 或更低版本的副本卸载数据库，则 <i>reload.sql</i> 文件可以包含对 ml_add_property 系统过程的调用，但该过程在新的版本 11 数据库中不提供。	使用版本 11 dbunload 实用程序卸载数据库。 有关如何确保使用正确版本的数据库实用程序的信息，请参见“使用实用程序”一节第 326 页。
如果使用版本 9 或更低版本的 dbunload 卸载数据库，则可能不会在重装使用 Transact-SQL 外连接（通过指定 *= 或 =*）的视图时正确创建这些视图。	向重装脚本添加以下代码行： <pre>SET TEMPORARY OPTION tsql_outer_joins='on';</pre> 您应在稍后对使用 Transact-SQL 外连接的任何视图重新编写代码。

升级版本 10 及更高版本数据库

升级数据库将添加和修改系统表、系统过程和数据库选项，以便启用版本 11 功能。它不会更改用于存储和访问磁盘上数据的文件格式，因此不会授予最新版本软件的所有新功能和性能增强的访问权限。

有关升级数据库文件格式的信息，请参见“重建版本 10 及更高版本数据库”一节第 327 页。

[升级数据库向导] 不会将版本 9.0.2 或更早版本的数据库升级到版本 11。要将现有数据库升级到版本 11，必须使用 dbunload 或 [卸载数据库向导] 将数据库卸载，然后再重装数据库。请参见“升级版本 9 及更早版本的数据库”一节第 324 页。

小心

在升级前，应始终对数据库文件进行备份。如果在升级现有文件时升级失败，则这些文件将无法使用。有关备份数据库的信息，请参见“[备份和数据恢复](#)”《SQL Anywhere 服务器 - 数据库管理》。

◆ **升级数据库 (Sybase Central)**

1. 执行升级软件的标准预防措施。请参见“[重要的升级预防措施](#)”一节第 326 页。
2. 选择 [开始] » [程序] » [SQL Anywhere 11] » [Sybase Central]。
3. 从 SQL Anywhere 插件，连接要升级的数据库。数据库必须在版本 11 的数据库服务器上运行。
4. 选择 [工具] » [SQL Anywhere 11] » [升级数据库]。
5. 请按照 [升级数据库向导] 中的说明进行操作。
6. 如果在向导中未选择存档事务日志，则在使用升级后的数据库之前，应停止数据库并通过创建副本来存档事务日志。

提示

还可以通过以下方法访问 [升级数据库向导]：

- 右击数据库并选择 [升级数据库]。
- 选择数据库，然后选择 [文件] » [升级数据库]。

◆ **升级数据库 (命令行)**

1. 执行升级软件的标准预防措施。请参见“[重要的升级预防措施](#)”一节第 326 页。
2. 请确保您对要升级的数据库有独占访问权，同时确保系统路径中版本 11 实用程序位于其它实用程序之前。请参见“[使用实用程序](#)”一节第 326 页。
3. 对数据库运行升级实用程序 (dbupgrad):

```
dbupgrad -c "connection-string"
```

在 *connection-string* 中指定的数据库用户必须对要升级的数据库具有 DBA 权限。

有关详细信息，请参见“[升级实用程序 \(dbupgrad\)](#)”一节《SQL Anywhere 服务器 - 数据库管理》。

4. 先关闭数据库并存档事务日志，然后再使用已升级的数据库。

◆ **升级数据库 (SQL)**

1. 从 Interactive SQL 或可以执行 SQL 语句的另一个应用程序连接到数据库。
2. 执行 ALTER DATABASE 语句。

例如，使用以下语句可升级数据库：

```
ALTER DATABASE UPGRADE;
```

有关详细信息，请参见“ALTER DATABASE 语句”一节《SQL Anywhere 服务器 - SQL 参考》。

3. 先关闭数据库并存档事务日志，然后再使用已升级的数据库。

在数据库镜像系统中升级 SQL Anywhere 软件和数据库

使用数据库镜像时，需要采取额外的步骤来应用 SQL Anywhere 维护版本或 EBF 或者升级数据库文件：

- 有关应用维护版本的信息，请参见“在数据库镜像系统中安装 SQL Anywhere 维护版本”一节第 335 页。
- 有关应用 EBF 的信息，请参见“在数据库镜像系统中应用 SQL Anywhere EBF”一节第 335 页。
- 有关升级或重建数据库文件的信息，请参见“升级数据库镜像系统中的数据库”一节第 336 页。

在数据库镜像系统中安装 SQL Anywhere 维护版本

数据库镜像系统中的所有服务器必须使用相同的 SQL Anywhere 维护版本。如果使用以下过程应用 SQL Anywhere 维护版本，数据库仅在第 3 步和第 4 步期间不可用。

◆ 将 SQL Anywhere 维护版本应用到数据库镜像系统

1. 通过发出 dbstop 命令关闭镜像服务器。
2. 在镜像服务器上安装 SQL Anywhere 的新版本。
3. 通过对每个服务器发出 dbstop 命令关闭主服务器和仲裁服务器。
4. 在主服务器上安装 SQL Anywhere 的新版本。
5. 重新启动主服务器和镜像服务器。
6. 在仲裁服务器上安装此软件的新版本。
7. 重新启动仲裁服务器。

在数据库镜像系统中应用 SQL Anywhere EBF

要安装 EBF，您必须为镜像系统中的每台数据库服务器（主服务器、镜像服务器和仲裁服务器）执行以下步骤：

1. 发出 dbstop 命令停止数据库服务器。
2. 安装 EBF。
3. 重新启动数据库服务器。

只有在关闭主服务器导致故障转移期间才发生停机时间。

另请参见

- “停止镜像系统中的数据库服务器”一节 《SQL Anywhere 服务器 - 数据库管理》
- “启动主服务器上的故障转移”一节 《SQL Anywhere 服务器 - 数据库管理》

升级数据库镜像系统中的数据库

有两个过程可以用于升级或重建参与数据库镜像系统的数据库。第一个过程较简单，但是数据库停机时间比第二个过程长。

◆ 升级或重建数据库镜像系统中的数据库

1. 关闭镜像服务器。
2. 关闭主服务器。
3. 使用主服务器上的副本升级或重建数据库。请参见“升级版本 10 及更高版本数据库”一节第 333 页或“重建版本 10 及更高版本数据库”一节第 327 页。
4. 将已升级或重建的数据库和事务日志复制到镜像服务器中。
5. 重新启动主服务器。
6. 重新启动镜像服务器。

注意

应移走所有重命名的事务日志文件，因为它们与新的数据库不兼容。两台服务器都需要初始事务日志文件用于启动镜像。通过对数据库执行 `dbping` 命令可以创建事务日志文件。

◆ 升级或重建数据库镜像系统中的数据库时最小化停机时间

1. 备份数据库和重命名事务日志。
2. 升级或重建不同计算机上数据库的备份副本。请参见“升级版本 10 及更高版本数据库”一节第 333 页或“重建版本 10 及更高版本数据库”一节第 327 页。
3. 关闭主服务器和镜像服务器。
4. 在主数据库上保存事务日志的当前副本。
5. 使用 `dbtran` 实用程序转换步骤 4 中所保存的事务日志。
此事务日志包含自从在步骤 1 中进行了备份以来对数据库进行的所有更改。
6. 用本地数据库服务器启动重建后的数据库。
7. 使用 Interactive SQL 的 `READ` 语句来应用转换后的事务日志。
8. 停止重建后的数据库。
9. 将已升级或重建的数据库及其事务日志复制到主服务器和镜像服务器。
10. 启动主服务器。
11. 启动镜像服务器。

升级 MobiLink

与现有软件的兼容性

- 新的 MobiLink 客户端与早于版本 11.0.0 的 MobiLink 服务器不兼容。
- 版本 11 的 MobiLink 服务器可以与版本 8、9、10 或 11 的客户端一同使用。要使用版本 10 或 11 的客户端，请使用 -x 选项启动 MobiLink 服务器。要使用版本 8 或 9 的客户端，请使用 -xo 选项启动 MobiLink 服务器。如果需要支持早期版本的客户端，应保留早期版本的 MobiLink 服务器以支持这些客户端。
- 请确认记录的任何行为更改都不影响您的应用程序。如果影响，则必须更新应用程序。请参见 [SQL Anywhere 11 - 更改和升级第 1 页](#)。

升级顺序

如果要升级现有 MobiLink 安装，必须按照下面的顺序升级组件：

1. 关闭 MobiLink 服务器。
2. 升级统一数据库。
请参见“[升级统一数据库](#)”一节第 337 页。
3. 升级 MobiLink 服务器。
请参见“[升级 MobiLink 服务器](#)”一节第 342 页。
4. 启动 MobiLink 服务器。
5. 升级 MobiLink 客户端。

如果 MobiLink 客户端是版本 8 或更高的版本，客户端升级可以分段进行或逐步进行。

有关 SQL Anywhere 远程数据库的信息，请参见“[升级 SQL Anywhere MobiLink 客户端](#)”一节第 342 页。有关 UltraLite 应用程序的信息，请参见“[将版本 9 及更早版本的 UltraLite 应用程序代码移植到版本 11.0.0](#)”一节第 349 页。

升级之前，应检查是否存在可能影响升级的行为更改，并执行标准升级预防措施。

有关详细信息，请参见：

- “[行为更改和不建议使用的功能](#)”一节第 178 页
- “[重要的升级预防措施](#)”一节第 326 页

升级统一数据库

必须先运行安装新系统对象的升级脚本，然后才能将新的 MobiLink 服务器与现有的统一数据库共用。必须由当前安装了 MobiLink 系统表的所有者运行升级脚本。

注意

- 如果有版本 10.0.0 以前创建的 `authenticate_user_hashed` 脚本，则必须使用 RDBMS 的二进制等效类型将其更改为接受 `BINARY(32)` 而非 `BINARY(20)`。

升级 SQL Anywhere 10.0.0 及更高版本

◆ 升级统一数据库（SQL Anywhere 10.0.0 及更高版本）

1. 升级 SQL Anywhere 数据库。

请参见“[升级版本 10 及更高版本数据库](#)”一节第 324 页。

2. 对要进行升级的版本运行合适的升级脚本。

升级脚本名为 `upgrade_sa.sql`。它位于 SQL Anywhere 安装目录的 `MobiLink\upgrade\version` 子目录中，其中 `version` 是要升级的 SQL Anywhere 原版本。

例如，在 Interactive SQL 中连接到数据库，然后运行以下命令：

```
READ "c:\Program Files\SQL Anywhere 11\MobiLink\upgrade
\10.0.1\upgrade_sa.sql"
```

升级版本 10.0.0 以前的 SQL Anywhere

- 在版本 10.0.0 之前，MobiLink 系统表归 `dbo` 所有。要运行 SQL Anywhere 数据库的安装脚本，必须以 MobiLink 系统表所有者的身份登录至统一数据库。如果用户要运行这些脚本，仅具有更改表的权限是不够的。若要运行升级脚本，可以使用 `SETUSER SQL` 语句模拟 `dbo`。例如：

```
SETUSER "dbo";
```

若要在 Sybase Central 中升级统一数据库，应使用 `GRANT CONNECT` 语句创建 `dbo` 口令，然后以 `dbo` 身份连接。例如：

```
GRANT CONNECT TO dbo IDENTIFIED BY password;
```

在后一种情况中，升级后应使用 `ALTER USER` 删除 `dbo` 口令。例如：

```
ALTER USER TO dbo IDENTIFIED BY "";
```

- 如果已经建立了 SQL Anywhere 统一数据库，但一直未与其同步，则必须运行安装脚本（非升级脚本）。这一步仅适用于 SQL Anywhere 统一数据库。

◆ 升级统一数据库（版本 10.0.0 以前的 SQL Anywhere）

1. 如果要升级版本 10.0.0 以前的 SQL Anywhere 统一数据库，必须首先将数据库升级到版本 11：

- a. 关闭数据库服务器。
- b. 将数据库升级到版本 11。

有关说明，请参见：

- “[升级版本 10 及更高版本数据库](#)”一节第 333 页
- “[重建版本 10 及更高版本数据库](#)”一节第 327 页
- “[为版本 11 重建版本 9 及更早版本的数据库](#)”一节第 329 页

- c. 启动数据库服务器，以 `DBA` 身份登录。

必须以 DBA 身份登录才能升级。

2. 如果要从 6.0.x 版本升级，请运行 MobiLink 安装脚本，该脚本位于 SQL Anywhere 安装目录的 *MobiLink\setup* 子目录中。如果要从更高版本进行升级，则不要运行此安装脚本。

有关安装脚本的详细信息，请参见“[MobiLink 统一数据库](#)”《[MobiLink - 服务器管理](#)》。

3. 对要进行升级的版本运行合适的升级脚本。

升级脚本名为 *upgrade_asa.sql*。它位于 SQL Anywhere 安装目录的 *MobiLink\upgrade\version* 子目录中，其中 *version* 是要升级的 SQL Anywhere 原版本。

要运行升级脚本，必须模拟 *dbo* 用户。可使用 SETUSER SQL 语句完成此操作。

例如，要升级 SQL Anywhere 9.0.2 版统一数据库，可使用 Interactive SQL 连接到数据库然后运行以下命令：

```
SETUSER "dbo";
READ 'c:\Program Files\SQL Anywhere 11\MobiLink\upgrade
\9.0.2\upgrade_asa.sql'
```

4. 删除 *dbo* 口令。例如：

```
GRANT CONNECT TO "dbo";
```

5. 如果作为 DBA 以外的用户运行 MobiLink 服务器，则必须向该用户授予对新的 MobiLink 系统对象的执行权限。哪些系统对象是新的取决于从哪个版本升级。以下代码授予所有 MobiLink 系统对象必要的权限。执行代码前，必须将用户名 *my_user* 更改为正在运行 MobiLink 服务器的用户名。

```
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_column to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_connection_script to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_database to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_device to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_device_address to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_listening to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_property to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_clients to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_delivery to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_global_props to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_notifications to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository_props to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository_staging to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_status_history to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_status_staging to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_script to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_script_version to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_scripts_modified to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_server to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_active_remote_id to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_passthrough_script to
my_user;
```

```

GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_passthrough to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_passthrough_status to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_passthrough_repair to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository_archive to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_delivery_archive to
my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_status_history_archive
to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_qa_repository_props_archive
to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_sis_sync_state to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_subscription to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_table to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_table_script to my_user;
GRANT SELECT, INSERT, UPDATE, DELETE ON dbo.ml_user to my_user;
GRANT EXECUTE ON dbo.ml_qa_get_agent_network_property to my_user;
GRANT EXECUTE ON dbo.ml_qa_get_agent_object_property to my_user;
GRANT EXECUTE ON dbo.ml_qa_get_agent_property to my_user;
GRANT EXECUTE ON dbo.ml_qa_get_message_property to my_user;
GRANT EXECUTE ON dbo.ml_add_column to my_user;
GRANT EXECUTE ON dbo.ml_add_connection_script to my_user;
GRANT EXECUTE ON dbo.ml_add_dnet_connection_script to my_user;
GRANT EXECUTE ON dbo.ml_add_dnet_table_script to my_user;
GRANT EXECUTE ON dbo.ml_add_java_connection_script to my_user;
GRANT EXECUTE ON dbo.ml_add_java_table_script to my_user;
GRANT EXECUTE ON dbo.ml_add_lang_conn_script_chk to my_user;
GRANT EXECUTE ON dbo.ml_add_lang_connection_script to my_user;
GRANT EXECUTE ON dbo.ml_add_lang_table_script to my_user;
GRANT EXECUTE ON dbo.ml_add_lang_table_script_chk to my_user;
GRANT EXECUTE ON dbo.ml_add_property to my_user;
GRANT EXECUTE ON dbo.ml_add_table_script to my_user;
GRANT EXECUTE ON dbo.ml_add_user to my_user;
GRANT EXECUTE ON dbo.ml_delete_device to my_user;
GRANT EXECUTE ON dbo.ml_delete_device_address to my_user;
GRANT EXECUTE ON dbo.ml_delete_listening to my_user;
GRANT EXECUTE ON dbo.ml_delete_sync_state to my_user;
GRANT EXECUTE ON dbo.ml_delete_sync_state_before to my_user;
GRANT EXECUTE ON dbo.ml_delete_user to my_user;
GRANT EXECUTE ON dbo.ml_qa_add_delivery to my_user;
GRANT EXECUTE ON dbo.ml_qa_add_message to my_user;
GRANT EXECUTE ON dbo.ml_qa_handle_error to my_user;
GRANT EXECUTE ON dbo.ml_qa_stage_status_from_client to my_user;
GRANT EXECUTE ON dbo.ml_qa_staged_status_for_client to my_user;
GRANT EXECUTE ON dbo.ml_qa_upsert_global_prop to my_user;
GRANT EXECUTE ON dbo.ml_reset_sync_state to my_user;
GRANT EXECUTE ON dbo.ml_set_device to my_user;
GRANT EXECUTE ON dbo.ml_set_device_address to my_user;
GRANT EXECUTE ON dbo.ml_set_listening to my_user;
GRANT EXECUTE ON dbo.ml_set_sis_sync_state to my_user;
GRANT EXECUTE ON dbo.ml_upload_update_device_address to my_user;
GRANT EXECUTE ON dbo.ml_upload_update_listening to my_user;
GRANT EXECUTE ON dbo.ml_server_update to my_user;
GRANT EXECUTE ON dbo.ml_server_delete to my_user;
GRANT EXECUTE ON dbo.ml_add_passthrough_script to my_user;
GRANT EXECUTE ON dbo.ml_add_passthrough to my_user;
GRANT EXECUTE ON dbo.ml_add_passthrough_repair to my_user;
GRANT EXECUTE ON dbo.ml_delete_passthrough_script to my_user;
GRANT EXECUTE ON dbo.ml_delete_passthrough to my_user;
GRANT EXECUTE ON dbo.ml_delete_passthrough_repair to my_user;

```

升级 Adaptive Server Enterprise、Oracle 或 Microsoft SQL Server MobiLink 系统表

如果 MobiLink 服务器是版本 11.0.0 以前的版本，则仅需升级 Adaptive Server Enterprise、Oracle 或 Microsoft SQL Server 统一数据库中的 MobiLink 系统对象。

◆ 升级统一数据库（Adaptive Server Enterprise、Oracle 或 Microsoft SQL Server）

1. 如果要从版本 6.0.x 升级，请运行位于 SQL Anywhere 安装目录的 *MobiLink\setup* 子目录中的 MobiLink 安装脚本。如果要从更高版本进行升级，则不要运行此安装脚本。
有关安装脚本的详细信息，请参见“[MobiLink 统一数据库](#)”《[MobiLink - 服务器管理](#)》。
2. 对于 Adaptive Server Enterprise 数据库，必须设置 "SELECT INTO" 权限。在 Sybase Interactive SQL 中运行以下命令：

```
USE MASTER
go
sp_dboption your-database-name, "SELECT INTO", true
go
USE your-database-name
go
checkpoint
go
```

3. 对要进行升级的版本运行合适的升级脚本。

升级脚本名为 *upgrade_XXX.sql*，其中 *XXX* 表示统一数据库的 RDBMS。它们位于 SQL Anywhere 安装目录的 *MobiLink\upgrade\version* 子目录中，其中 *version* 是要升级的 MobiLink 原版本。

例如，要升级 Microsoft SQL Server 版本 9.0.2 统一数据库，请运行以下命令：

```
isql -S server_name -U user_name -P password -I
"c:\Program Files\SQL Anywhere 11\MobiLink\upgrade\9.0.2\upgrade_mss.sql"
```

升级 IBM DB2 LUW

如果 MobiLink 服务器是 11.0.0 之前的版本，则仅需升级 DB2 LUW 统一数据库。

◆ 升级 DB2 LUW 统一数据库

1. 如果从 MobiLink 版本 6 升级，应通过运行 SQL 安装脚本 *MobiLink\setup\syncdb2.sql* 来创建 MobiLink 系统表和存储过程。

有关如何运行 DB2 LUW 安装脚本的信息，请参见“[IBM DB2 LUW 统一数据库](#)”一节《[MobiLink - 服务器管理](#)》。

2. 找到 DB2 LUW 升级脚本。

升级脚本名为 *upgrade_db2.sql*，保存在 SQL Anywhere 安装目录的 *MobiLink/upgrade/version* 子目录中。*version* 目录指要升级的 MobiLink 的原版本。

3. 复制 *upgrade_db2.sql* 然后修改复制后的副本。更改脚本开头的 CONNECT 语句，以便该语句可以使用要连接的实例。将复制的 SQL 脚本应用于统一数据库。

升级 MobiLink 服务器

如果使用的是版本 11 的远程客户端，那么需要从版本 10.0.0 升级到版本 11.0.0。

在使用版本 11 MobiLink 服务器之前，应检查是否存在对您有影响的行为更改。请参见 [SQL Anywhere 11 - 更改和升级第 1 页](#)。

版本 11 的 MobiLink 服务器仅支持版本 8、9、10 和 11 的 SQL Anywhere 和 UltraLite 客户端。如果需要支持早期版本的客户端，应保留早期版本的 MobiLink 服务器以支持这些客户端。

升级 SQL Anywhere MobiLink 客户端

在生产环境中，只有在已经升级统一数据库和 MobiLink 服务器之后才能升级 SQL Anywhere 远程数据库。

在 10.0.0 版中，Adaptive Server Anywhere 已更名为 SQL Anywhere。

应考虑以下几种升级：

- 升级软件。
- 升级远程数据库本身。
- 升级整个应用程序。

小心

必须恰好在升级 MobiLink 同步所涉及的数据库之前成功完成一次同步。还应校验和备份数据库。

升级软件

建议您同时升级 dbmlsync 和 SQL Anywhere 数据库服务器。必须先升级远程数据库，然后才能运行新的 dbmlsync 实用程序。

版本 11 MobiLink 客户端需要使用版本 11 的 MobiLink 服务器来进行同步。版本 11 MobiLink 客户端不能与早于版本 11 的 MobiLink 服务器同步。

有关升级 MobiLink 的信息，请参见“[升级 MobiLink](#)”一节第 337 页。

升级远程数据库

可以通过升级 SQL Anywhere 数据库的过程来升级 MobiLink SQL Anywhere 远程数据库。有关说明，请参见“[升级 SQL Anywhere](#)”一节第 324 页。

在某些情况下（例如存在模式更改或其它重要数据库更改时），可能需要手工执行卸载和重装。

◆ 手工卸载/重装远程 SQL Anywhere 数据库

1. 停止所有数据库活动。
2. 成功执行一次同步然后校验并备份远程数据库。
3. 运行 dbtran 实用程序，以显示数据库事务日志的起始偏移和结束偏移。记下结束偏移。

请参见“[翻译日志文件实用程序 \(dbtran\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

4. 重命名事务日志。这可确保在卸载过程中不会修改该日志。将重命名的日志文件移动到一个安全位置，例如某个脱机目录中。
5. 卸载数据库。
请参见“[为版本 11 重建版本 9 及更早版本的数据库](#)”一节第 329 页。
6. 初始化新的数据库。
请参见“[初始化实用程序 \(dbinit\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
7. 将数据重装到新数据库中。
请参见“[为版本 11 重建版本 9 及更早版本的数据库](#)”一节第 329 页。
8. 关闭新数据库。
9. 消除新数据库的事务日志。
10. 在新数据库上运行 `dblog`，并使用以下选项：
 - 使用 `-z` 指定先前记下的结束偏移。
 - 使用 `-x` 将相对偏移设置为零。

例如：

```
dblog -x 0 -z 137829 database-name.db
```

请参见“[事务日志实用程序 \(dblog\)](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

11. 启动 `dbmlsync`，指定之前移动的原始日志文件的位置。
请参见“[dbmlsync 语法](#)”一节《[MobiLink - 客户端管理](#)》。
12. 当不再需要旧日志文件时，请设置数据库选项 `delete_old_logs`。
请参见“[delete_old_logs 选项 \[MobiLink 客户端\] \[SQL Remote\] \[复制代理\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。

升级应用程序

部署新版本的 MobiLink 应用程序时，建议您为同步脚本使用新版本的名称。例如，如果现有应用程序使用名为 `v1` 的脚本版本，则升级后的应用程序可以使用名为 `v2` 的脚本版本。可以同时使用这两个脚本版本。这样，以递增方式（而不是一次完成）升级远程数据库就会更加容易。

对于 9.0.0 及更高版本，MobiLink 服务器的 `-zd` 选项已被删除。如果您的部署使用 `-zd` 选项并希望进行升级，则必须更改下载脚本，以将上次下载时间戳作为第一个参数来接受。或者，可先升级客户端然后使用已命名参数启动，这样可按任意顺序放置脚本参数。

升级 QAnywhere

若要升级 QAnywhere 应用程序，可升级统一数据库、应用程序和客户端消息存储库。

要升级统一数据库，请参见“[升级统一数据库](#)”一节第 337 页。

若要升级应用程序，应查看此版本中的新功能和行为更改。

请参见 [SQL Anywhere 11 - 更改和升级第 1 页](#)。

◆ 升级 QAnywhere 消息存储库

1. 部署 QAnywhere 文件。

请参见“[部署 QAnywhere 应用程序](#)”一节《[MobiLink - 服务器管理](#)》。

2. 升级消息存储库：

使用 -su 选项或 -sur 选项运行 QAnywhere 代理。请参见：

- “[-su 选项](#)”一节《[QAnywhere](#)》
- “[-sur 选项](#)”一节《[QAnywhere](#)》

升级 UltraLite

请务必先查看新功能和行为更改列表以确定应用程序是否受到影响，然后再在此版本软件中使用现有应用程序。请参见：

- “UltraLite 的新功能” 一节第 67 页
- “UltraLite 的行为更改和不建议使用的功能” 一节第 71 页

使用 UltraLite 11.0.0 实用程序

如果您的计算机上有多个版本的 SQL Anywhere，在使用 UltraLite 实用程序时必须注意系统路径，以确保所使用的是版本 11 的实用程序。请参见“使用实用程序”一节第 326 页。

升级版本 10 及更高版本数据库

对于版本 10.0.0 及更高版本的 UltraLite 数据库，当您从版本 11 应用程序或任何版本为 11 的 UltraLite 工具连接到数据库时，数据库文件格式会自动升级到版本 11。数据库一旦升级，就不能连接版本 10 的应用程序、实用程序和软件。请参见“升级版本 10.0.0 或更高版本的 UltraLite 数据库”一节第 346 页。

升级版本 9 及更早版本的数据库

对于版本 9 及更早版本的 UltraLite 数据库，必须重建数据库，具体方法是卸载旧的数据库，然后将其重装到新的版本 11 数据库。尝试连接版本 9 或更早版本的 UltraLite 11 数据库将导致数据库启动时发生错误。请参见“升级版本 9 及更早版本的 UltraLite 数据库”一节第 346 页。

与现有软件的兼容性

- 版本 11 不支持 UltraLite for AppForge。
- UltraLite 11.0.0 数据库文件只支持来自版本 11.0.0 客户端应用程序或版本 11 UltraLite 引擎的连接。
- UltraLite 11.0.0 会自动升级版本 10 的数据库文件。
- 对于使用版本 9 及更早版本的 UltraLite 所创建的数据库文件和应用程序代码，UltraLite 版本 11 运行时和 UltraLite 版本 11 引擎不起作用。
- 从当前版本的 Sybase Central 对旧数据库和客户端应用程序进行的管理如下所示：
 - 对版本 11 的数据库进行完全管理。
 - 对版本 10 的数据库进行完全管理。版本 10 的数据库会自动升级到版本 11。
 - 只能连接到 8 或 9 版本的数据库来升级数据库文件格式。

Palm OS 的早期版本

Palm 设备的早期版本（例如，版本 4.x）只有大约 200 KB 的 RAM。由于 UltraLite 需要动态内存，此限制可能会引发问题。有关如何针对 UltraLite 优化这些早期版本的 Palm 设备的信息，请参见“UltraLite 优化策略”一节《UltraLite - 数据库管理和参考》。

升级版本 10.0.0 或更高版本的 UltraLite 数据库

从版本 11 UltraLite 应用程序或任一版本 11 的 UltraLite 工具（如 `ulsync`、`ulunload`、`ulinfo` 或 Sybase Central）连接到版本 10 UltraLite 数据库时，该数据库将自动升级到版本 11 数据库。数据库升级后，版本 10 的应用程序将无法再连接到该数据库。

升级版本 9 及更早版本的 UltraLite 数据库

升级以前版本的 UltraLite 数据库时，需要：

- 同步数据，如果要升级的数据库包含重要信息。
- 断开与所有应用程序和管理工具的连接。
- 将数据库文件复制到台式计算机。

注意

在升级旧版本数据库之前，无法使用除 `ulunloadold.exe` 以外的任何 UltraLite 11.0.0 管理工具连接到该数据库。

数据库升级的特殊注意事项

- 现在，UltraLite 模式是数据库的一部分，并不再存在于独立的 `.usm` 文件之中。应用程序必须部署一个初始数据库或使用新的 `CreateDatabase` 功能以编程方式创建数据库。
- 从 UltraLite 的版本 10 开始，文件格式已经统一。这表明现在多数平台可共享一个数据库，因此 Unicode 字符不再是必需的。

如果需要的字符并未包含在所选归类中，则应使用 UTF-8 对数据库进行编码。请参见“[字符集编码的 UltraLite 平台要求](#)”一节《UltraLite - 数据库管理和参考》和“[UltraLite utf8_encoding 创建参数](#)”一节《UltraLite - 数据库管理和参考》。

Windows Mobile 和桌面数据库

如果要升级以上平台中任一平台上的数据库，且不再需要 Unicode 字符，则不要使用 UTF-8 对数据库进行编码。因为 UTF-8 编码可能会不必要地增加数据库的大小。

- 从版本 10 开始，无论数据库是否区分大小写，所有数据库口令均区分大小写。因此，在升级数据库时，用户 ID、口令甚至是受信任的根证书可能都不会保留。必须新的 UltraLite 数据库中添加以前的用户 ID、口令和受信任的根证书。新数据库的缺省 DBA 口令是 `sql`。如果没有提供用户 ID 或口令，则假定用户 ID 和口令分别为 `DBA` 和 `sql`。

UltraLite 数据库升级途径

由于在 UltraLite 中创建数据库的方法多种多样，因此升级过程也因采取的方法而异。下表说明了根据需要升级的内容而应采取的方法。

如果需要升级先前版本的如下内容…	请使用此工具…
<ul style="list-style-type: none"> ● 模式文件 (.usm) ● 数据库文件 (.udb) ● Palm OS 数据库记录 (.pdb) 	[升级数据库向导] 或卸载旧数据库实用程序 (ulunloadold) 以及装载数据库实用程序 (ulload)
源自 SQL Anywhere 参考数据库的 UltraLite 数据库。 ¹	[抽取数据库向导] 或初始化数据库实用程序 (ulinit)

¹ 请确保事先升级了 SQL Anywhere 数据库。请参见“升级 SQL Anywhere”一节第 324 页。

有关使用本节所述的工具进行逐步升级的介绍，请参见“UltraLite 数据库升级工具”一节第 347 页。

UltraLite 数据库升级工具

若要升级现有 UltraLite 数据库或模式，请选择以下选项之一：

- **升级数据库向导** 如果希望经由指导来完成此过程并在选择可用属性/选项时获得帮助，可选择此向导。在完成 [升级数据库向导] 中的说明操作前，请确保已决定：
 - 数据源。可以选择数据库文件也可以选择模式文件。
 - 输出升级数据库的位置。可以从以下任一选项中选择：
 - **新 UltraLite 11.0.0 数据库** 选择此选项可创建数据库并与之相连。
 - **现有 UltraLite 11.0.0 数据库** 选择此选项可将旧数据库模式应用到新数据库。
- **ulunloadold 实用程序** 如果有以下任一需求，请选择实用程序：
 - 仅希望将已命名的表升级到新数据库。
 - 希望实现面向批处理的进程。

◆ 将现有 UltraLite 数据库升级到版本 11.0.0 (Sybase Central)

1. 完成执行升级软件的标准预防措施，包括备份数据库。请参见“重要的升级预防措施”一节第 326 页。
2. 选择 [开始] » [程序] » [SQL Anywhere 11] » [Sybase Central]。
3. 选择 [工具] » [UltraLite 11] » [升级数据库]。
4. 在 [要升级的数据库] 页面上，选择以下选项之一：
 - **旧数据库** 浏览以查找 UltraLite 数据库 (*.udb 或 *.pdb)。
 - **旧模式文件** 浏览以查找 UltraLite 模式文件 (*.usm)。
5. 指定连接参数，然后单击 [下一步]。
6. 在 [升级目标] 页面上，选择以下选项之一：

- **新数据库** 必须创建新数据库文件并设置所需的数据库属性。按照向导中的说明进行操作。
- **使用已经连接的数据库** 可以从提供的列表中选择一个已连接的数据库。
- **使用您未连接到的现有数据库** 单击 [数据库] 以打开 [连接] 窗口，然后连接到现有 UltraLite 11.0.0 数据库。

7. 请按照 [升级数据库向导] 中的说明进行操作。单击 [完成]。

如果在以前版本的 UltraLite 中包括受信任的根证书，请将它们添加到新的 UltraLite 数据库上。

如果以前版本的 UltraLite 数据库中的用户没有出现在新数据库中，请记得添加这些用户。请参见“使用 UltraLite 用户”一节《UltraLite - 数据库管理和参考》。

◆ 将现有 UltraLite 数据库升级到版本 11.0.0（命令行）

1. 执行升级软件的标准预防措施，包括备份数据库。请参见“重要的升级预防措施”一节第 326 页。
2. 请确保系统路径中版本 11.0.0 的 UltraLite 实用程序位于旧版本 UltraLite 实用程序之前。请参见“使用实用程序”一节第 326 页。
3. 请使用以下语法运行卸载旧数据库实用程序 (ulunloadold)，以创建 XML 中间文件：

```
ulunloadold -c "connection-string" [ options ] xml-file
```

请确保您已经：

- 命名了 ulunloadold 实用程序要创建的 XML 文件。
- 在 *connection-string* 中使用 DBF 或 *schema_file* 参数，这取决于是否升级旧 UltraLite 数据库 (*.udb 或 *.pdb) 还是要升级旧 UltraLite 模式文件 (*.usm)。

所有其它选项均为可选。

有关 ulunloadold 实用程序支持的选项的详细信息，请参见“UltraLite 卸载旧数据库实用程序 (ulunloadold)”一节《UltraLite - 数据库管理和参考》。

4. 执行装载 XML 到数据库实用程序 (ulload)，将 XML 重装到新的或现有 UltraLite 数据库。

如果要将 XML 装载到新数据库中，则 **-c connection-string** 选项会设置该数据库的连接参数（例如，验证 UltraLite 用户所需的 UID 和 PWD）。

所指定的 **-o [extended-options]** 取决于是否要更改数据库的特性/属性（例如，将区分大小写的数据库更改为不区分大小写的数据库）。

有关完整参照，请参见“UltraLite 装载 XML 到数据库实用程序 (ulload)”一节《UltraLite - 数据库管理和参考》。

例如，要使用名为 *dbschema.xml* 的中间 XML 文件将名为 *dbschema8.usm* 的 UltraLite 8.0.x 模式文件升级到名为 *db.udb* 的现有 UltraLite 版本 11.0.0 数据库中，需要运行以下两个命令：

```
ulunloadold -c schema_file=dbschema8.usm dbschema.xml  
  
ulload -c DBF=db.udb dbschema.xml
```

将版本 9 及更早版本的 UltraLite 应用程序代码移植到版本 11.0.0

必须使用新版本 11.0.0 API 重建先前的 UltraLite 应用程序。自引入 UltraLite 10 以来，这些 API 已得到明显增强，因而需要首先更改代码（如果适用），然后再重新构建应用程序。

在开始之前，请查看 UltraLite 的新功能和行为更改，以检查所使用的 API 有哪些更改。请参见“[UltraLite 的新功能](#)”一节第 224 页。

应用程序升级的特殊注意事项

- 必须更新连接代码。
 - UltraLite for C/C++: “[连接到数据库](#)”一节《[UltraLite - C 及 C++ 编程](#)》
 - UltraLite for embedded SQL: “[连接到数据库](#)”一节《[UltraLite - C 及 C++ 编程](#)》
 - UltraLite.NET: “[连接到数据库](#)”一节《[UltraLite - .NET 编程](#)》
 - UltraLite for M-Business Anywhere: “[连接到 UltraLite 数据库](#)”一节《[UltraLite - M-Business Anywhere 编程](#)》
- 从版本 10 开始，UltraLite 不再需要单独的 *.usm* 文件来创建数据库。可在桌面上配置数据库然后使用应用程序部署数据库，或者利用应用程序以编程方式创建数据库。
- 此版本的 UltraLite 始终启用验证功能，可支持多达四个用户 ID 和口令。不过，如果不想在数据库中运用验证功能，请不要创建或提供任何用户 ID 和口令。如果未提供任何用户 ID 和口令，则 UltraLite 将始终使用 **UID=DBA** 和 **PWD=sql** 这一缺省设置。请参见“[解释用户 ID 和口令组合](#)”一节《[UltraLite - 数据库管理和参考](#)》。
- 如果有多个嵌入式 SQL 文件，则必须仍然使用 SQL 预处理器 (sqlpp) 对每个文件进行预处理，以创建 C/C++ 源文件。不过，您不必再使用参考数据库。UltraLite 数据库现在直接支持嵌入式 SQL。
- 与先前版本一样不支持 Unicode 字符。相反，版本 11.0.0 的 UltraLite 数据库使用 UTF-8 为多字节字符编码。因此，不必再规划运行在非 Unicode 运行时系统中的 Unicode 数据库。

UltraLite 版本 9 及更早版本的应用程序代码升级路径

由于在 UltraLite 中开发 API 及方法不同，因此升级过程也因采取的方法而异。下表说明了根据需
要升级的内容而应采取的方法。

如果需要升级先前版本的如下内容...	请执行此操作...
ulgen 生成的 C/C++ 应用程序	<p>1. 升级 UltraLite 数据库。</p> <p>2. 为应用程序重写数据访问代码。</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>注意 您可以尝试在 SQL Anywhere 10 中使用 [迁移 C++ API 向导]，但有可能仍需为应用程序重写某些数据访问代码。</p> <p>该向导可帮助您识别可能不再与版本 10 的说明相符的嵌入式 SQL。如果无法完成迁移过程，则可以停止，然后将已经修改的 SQL 语句保存到 *.uag 文件中。</p> <p>在 SQL Anywhere 10 中提供了 [迁移 C++ API 向导]。您可以在 http://www.sybase.com/download_sqlany_10_developer_edition 下载 SQL Anywhere 10 Developer Edition。</p> </div>
嵌入式 SQL 应用程序	只进行了一些无关紧要的更改。不需要任何工具。
Java 应用程序	请参见 UltraLiteJ 。
UltraLite 组件	<p>主代码的重新编写包括以下内容：</p> <ul style="list-style-type: none"> ● 重新编写模式 在版本 10 及更高版本的数据库中集成了模式。重写 ULConnection 对象的 OpenWithCreate 函数所需的所有组件均已删除，所有模式升级代码也已删除。可以使用 ULDatabaseManager.CreateDatabase 在设备上创建数据库。若要减少在新数据库中定义属性所需的代码量，应先在桌面上创建数据库，待完成后将其部署到设备中。有关完整详细信息，请参见“在第一次连接时创建 UltraLite 数据库”一节《UltraLite - 数据库管理和参考》和“创建 UltraLite 数据库”一节《UltraLite - 数据库管理和参考》。 ● 重新编写连接 所有使用连接参数对象的接口均已删除：这包括创建、打开或删除数据库的函数或方法。可以使用字符串接口在这些连接参数中传递。

¹ 请确保事先升级了 SQL Anywhere 数据库。请参见“升级 SQL Anywhere”一节第 324 页。

从版本 9 及更早版本的 SQL Anywhere 数据库中抽取 UltraLite 数据库

可以使用 [抽取数据库向导] 或初始化数据库实用程序 (ulimit) 从版本 10.0.0 或更高版本的 SQL Anywhere 数据库中抽取 UltraLite 数据库。

- 如果希望经由指导来完成此过程并在选择可用属性/选项时获得帮助，可选择此向导。

- 如果希望实现面向批处理的进程，则选择此实用程序。

◆ 从版本 9 或更早版本的 SQL Anywhere 参考数据库初始化/抽取 UltraLite 数据库 (Sybase Central)

1. 执行升级软件的标准预防措施，包括备份数据库。请参见“重要的升级预防措施”一节第 326 页。
2. 升级 SQL Anywhere 数据库，并在准备该数据库时注意 UltraLite 的使用。如果需要更新发布，务必在重新创建 UltraLite 数据库之前进行。
有关升级版本 9 及更早版本的 SQL Anywhere 数据库的信息，请参见“为版本 11 重建版本 9 及更早版本的数据库”一节第 329 页。
3. 选择 [开始] » [程序] » [SQL Anywhere 11] » [Sybase Central]，启动 Sybase Central。
4. 选择 [工具] » [UltraLite 11] » [抽取 (ulinit) 数据库] 来抽取 UltraLite 版本的 SQL Anywhere 数据库。
5. 请按照 [抽取数据库向导] 中的说明进行操作。

◆ 从版本 9 或更早版本的 SQL Anywhere 参考数据库初始化/抽取 UltraLite 数据库 (命令行)

1. 执行升级软件的标准预防措施，包括备份数据库。请参见“重要的升级预防措施”一节第 326 页。
2. 请确保系统路径中版本 11.0.0 的 UltraLite 实用程序位于旧版本 UltraLite 实用程序之前。请参见“使用实用程序”一节第 326 页。
3. 升级现有 SQL Anywhere 数据库，并在准备该数据库时注意 UltraLite 的使用。如果需要更新发布，务必在重新创建 UltraLite 数据库之前进行。
有关升级版本 9 及更早版本的 SQL Anywhere 数据库的信息，请参见“为版本 11 重建版本 9 及更早版本的数据库”一节第 329 页。
4. 使用以下语法运行初始化数据库实用程序 (ulinit) 以抽取 UltraLite 数据库：

```
ulinit -a "SA-connection-string" -c "UL-connection-string"  
-n pubname [ options ]
```

请确保您已经：

- 为已升级的 SQL Anywhere 参考数据库和要使用此命令创建的新 UltraLite 数据库均提供了连接字符串。
- 为包含 UltraLite 数据库所需表的发布命名。要抽取所有表，请使用 **-n***。

所有其它选项均为可选。

注意

由于要创建新的数据库，因此，使用 **UID** 和 **PWD** 这样的参数创建初始用户 ID 和口令，以进行验证。在本例中没有参考 SQL Anywhere 数据库。不过，要替换其它 SQL Anywhere 参考数据库属性缺省值，可使用 **-o [extended-options]**。请参见“为 UltraLite 选择数据库创建参数”一节《UltraLite - 数据库管理和参考》。

有关完整参照，请参见“UltraLite 初始化数据库实用程序 (ulinit)”一节《UltraLite - 数据库管理和参考》。有关可配置的 UltraLite 数据库属性的详细信息，请参见“UltraLite 数据库属性”《UltraLite - 数据库管理和参考》。

升级 SQL Remote

SQL Remote 安装包括一个统一数据库和许多远程数据库，以及每个站点上的消息代理。

在每个站点上，消息代理处理消息的发送和接收。消息采用 SQL 语句的形式，数据库服务器处理这些 SQL 语句的实际执行。

SQL Remote 的升级要求如下：

- **可以逐个站点进行软件升级** 较低版本的消息代理 (dbremote) 可以同版本 11 的消息代理交换消息。只要将数据库选项 `compression` 的值设置为 -1，SQL Remote 版本 5 消息代理就可以与版本 11 消息代理交换消息。无需在安装期间同时升级软件。请参见“[compression 选项 \[SQL Remote\]](#)”一节《[SQL Anywhere 服务器 - 数据库管理](#)》。
- **升级数据库** 如果要升级使用 SQL Anywhere 版本 9 或更低版本的远程数据库或统一数据库，必须通过卸载然后再重装数据库的方式升级数据库的文件格式。无需同时升级所有数据库。
有关卸载和重装数据库的说明，请参见“[为版本 11 重建版本 9 及更早版本的数据库](#)”一节第 329 页。
- **升级 Adaptive Server Enterprise 统一数据库** SQL Remote 不再支持 Adaptive Server Enterprise 统一数据库。要同步 Adaptive Server Enterprise 数据库，应升级到 MobiLink。
有关从 SQL Remote 迁移到 MobiLink 的信息，请访问 <http://www.sybase.com/detail?id=1034174#335>。
- **升级 SQL Remote** 版本 11.0.0 中删除了 SQL Remote 对 VIM 和 MAPI 消息系统的支持。将使用 VIM 或 MAPI 的数据库升级到 SQL Anywhere 版本 11 时，必须将消息类型更改成 File、FTP 或 SMTP。如果消息类型是 MAPI 或 VIM，则 `dbremote.exe` 不会启动。

示例

以下是升级版本 5 SQL Remote 的一种方法：

1. 升级统一数据库服务器和消息代理，然后通过卸载并重装统一数据库的方式升级数据库文件。将 `compression` 数据库选项设置为 -1，这样所有消息都与远程站点的版本 5 软件兼容。要卸载并重装统一数据库，请参见“[重建参与同步或复制的数据库](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
2. 逐个升级远程数据库服务器和消息代理，然后通过卸载并重装远程数据库的方式升级数据库文件格式。可以将 `compression` 数据库选项设置为 -1 之外的其它值，以便对要发送到统一数据库服务器的消息进行压缩和编码。要卸载并重装远程数据库，请参见“[重建参与同步或复制的数据库](#)”一节《[SQL Anywhere 服务器 - SQL 的用法](#)》。
3. 升级所有远程数据库服务器和消息代理后，将统一站点处的 `compression` 数据库选项设置为 -1 之外的其它值。

索引

其它

- .cab 文件
 - 10.0.0 版中的增强, 133
- .cdb
 - 10.0.0 版不支持的文件扩展名, 162
- .NET
 - 11.0.0 版中的行为更改, 83
- .sasrv.ini
 - 10.0.0 版中的行为更改, 139
- .scRepository
 - 11.0.0 版中已重命名, 76
- .wrt
 - 10.0.0 版不支持的文件扩展名, 162
- [视图属性] 窗口
 - 10.0.1 版中的增强, 90
- @data 选项
 - 11.0.0 版中的增强, 30
- @filename 选项
 - 10.0.0 版中的增强, 115
- \$mml_connect
 - 10.0.0 版中的新功能, 178
- \$mml_password
 - 10.0.0 版中的新功能, 178
- \$mml_user
 - 10.0.0 版中的新功能, 178
- #hook_dict 表
 - 10.0.0 版中的 dbmsync 增强, 176
 - 10.0.0 版中的 SQL Remote 增强, 192
- % 运算符
 - 11.0.0 版中的行为更改, 56
- 10.0.0 版
 - 新功能, 107
 - 行为更改, 107
- 10.0.0 版中的新功能
 - 关于, 107
- 10.0.1 版
 - 新功能, 85
 - 行为更改, 85
- 11.0.0 版
 - 已知的升级问题, 333
 - 新功能, 23
 - 行为更改, 23
- 11.0.0 版中的新功能
 - 关于, 23
- 11.0.1 版
 - 新功能, 1
 - 行为更改, 1
- 11.0.1 版中的新功能
 - 关于, 1
- 1252NOR 归类
 - 10.0.0 版中的新功能, 136
- 1254TRKALT 归类
 - 9.0.1 版中的新功能, 242
- 256 位 AES 加密
 - 11.0.0 版中的新功能, 29
- 64 位
 - 11.0.0 版中的行为更改, 54
- 64 位平台, MobiLink 支持
 - 11.0.0 版中的新功能, 60
- 8.0.0 版
 - 行为更改, 298
- 8.0.1 版
 - 新功能, 290
 - 行为更改, 290, 295
- 8.0.2 版
 - 行为更改, 278, 286
- 9.0.0 版
 - 新功能, 254
 - 行为更改, 254
- 9.0.1 版
 - 新功能, 238
 - 行为更改, 238, 250
- 9.0.2 版
 - 新功能, 216
 - 行为更改, 216, 230
- 950TWN 归类
 - 9.0.2 版中不建议使用, 230
- bc 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 180
- bn 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 行为更改, 181
- cc 选项
 - 9.0.1 版中的新数据库服务器功能, 240
- ch 选项
 - 10.0.0 版中的数据库服务器行为更改, 144
- cl 选项
 - 10.0.0 版中的数据库服务器行为更改, 144
- cm 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 新功能, 171
 - 10.0.0 版中的新功能, 118

- dbunload, 11.0.1 版中的新功能, 3
- codepage 选项
 - 11.0.0 版中不建议使用, 79
- cp 选项
 - 11.0.0 版中的 dbunload 增强, 30
- cr 选项
 - 9.0.1 版中的新数据库服务器功能, 240
- cs 选项, MobiLink (mlsrv11)
 - 11.0.0 版中的新功能, 61
- ct 选项
 - 10.0.0 版中的数据库服务器行为更改, 139
- cv 选项
 - 9.0.1 版中的新数据库服务器功能, 240
- c 选项
 - 10.0.0 版中的数据库服务器行为更改, 144
- dc 选项
 - 9.0.1 版中的 dbmlsync 新功能, 245
- dd 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 180
- dh 选项
 - 10.0.0 版中的新功能, 118
- dsd 选项
 - 10.0.0 版中的 MobiLink 新功能, 172
- ds 选项
 - 10.0.1 版中的新功能, 91
- dt 选项
 - 10.0.0 版中的 MobiLink 新功能, 172
 - 10.0.0 版中的新功能, 118
- d 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 180
 - 10.0.0 版中删除的数据库服务器支持, 162
- ec 选项
 - 10.0.0 版中的行为更改, 146
- esu 选项
 - 10.0.0 版中的 MobiLink 新功能, 172
- es 选项
 - 11.0.0 版中的新功能, 31
- e 选项
 - 不建议在 10.0.1 版中使用初始化实用程序 [dbinit] 选项, 95
 - 在 11.0.0 版中不支持初始化实用程序 [dbinit] 选项, 56
- fd 选项
 - 10.0.0 版中的 QAnywhere 新功能, 188
- fr 选项
 - 10.0.0 版中的 QAnywhere 新功能, 188
- ftr 选项
 - 10.0.0 版中的 MobiLink [dbmlsrv10] 新功能, 171
- ga 选项, MobiLink [dblsn]
 - 11.0.0 版中不建议使用, 63
- gb 选项
 - 11.0.0 版中的增强, 31
- gi 选项, MobiLink [dblsn]
 - 11.0.0 版中不建议使用, 63
- gss 选项
 - 11.0.0 版中的增强, 44
 - 11.0.0 版中的行为更改, 50
- gtc 选项
 - 10.0.0 版中的新功能, 118
- gx 选项
 - 11.0.0 版不支持, 50
 - 不建议在 10.0.1 版中使用数据库服务器选项, 95
- g 选项
 - 10.0.0 版中取消的监听器 [dblsn] 选项, 177
 - 11.0.0 版中的 dbunload 增强, 30
- g 选项, MobiLink [dblsn]
 - 11.0.0 版中不建议使用, 63
- idl 选项
 - 10.0.1 版中的 QAnywhere 新功能, 98
- id 选项, QAnywhere (qastop)
 - 11.0.0 版中的新功能, 64
- im 选项
 - 11.0.0 版中的新功能, 31
- ja 选项
 - 10.0.0 版中不建议使用的 dbinit, 142
 - 10.0.0 版中不建议使用的 dbupgrad, 142
- jconnect 选项
 - 在 10.0.0 版中不受 Interactive SQL 支持, 209
- jdk 选项
 - 10.0.0 版中不建议使用的 dbinit, 142
 - 10.0.0 版中不建议使用的 dbupgrad, 142
- jr 选项
 - 10.0.0 版中不建议使用的 dbunload, 142
 - 10.0.0 版中不建议使用的 dbupgrad, 142
- j 选项
 - 10.0.0 版中不建议使用的 dbupgrad, 142
- ksc 选项
 - 11.0.0 版中的新功能, 32
- ksd 选项
 - 11.0.0 版中的新功能, 32
- ks 选项
 - 11.0.0 版中的新功能, 32
- k 选项

-
- 10.0.0 版中不建议使用的 MobiLink [dbmlsync] 选项, 185
 - la_port 选项
 - 10.0.0 版中不建议使用的 QAnywhere 选项, 190
 - lp 选项
 - 10.0.0 版中的 QAnywhere 新功能, 190
 - lsc 选项, MobiLink (mlsrv11)
 - 11.0.0 版中的新功能, 61
 - l 选项
 - dbunload, 11.0.1 版中的新功能, 3
 - mn 选项
 - 10.0.0 版中的 QAnywhere [qaagent] 新功能, 190
 - mp 选项
 - 10.0.0 版中的 QAnywhere [qaagent] 新功能, 190
 - mu 选项
 - 10.0.0 版中的 QAnywhere [qaagent] 新功能, 190
 - nc 选项
 - 10.0.0 版中的 MobiLink [dbmlsrv10] 新功能, 171
 - ni 选项
 - 10.0.0 版中的监听器 [dblsn] 新功能, 177
 - dblsn, 11.0.0 版中的新功能, 62
 - no 选项
 - 11.0.0 版中的 dbunload 增强, 30
 - ns 选项
 - 10.0.0 版中的监听器 [dblsn] 新功能, 177
 - nu 选项
 - 10.0.0 版中的监听器 [dblsn] 新功能, 177
 - dblsn, 11.0.0 版中的新功能, 62
 - odbc 选项
 - 在 10.0.0 版中不受 dbconsole 支持, 209
 - 在 10.0.0 版中不受 Interactive SQL 支持, 209
 - os 选项
 - 10.0.0 版中的行为更改, 146
 - ot 选项
 - 10.0.0 版中的新功能, 118
 - oy 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 181
 - pc 选项
 - 10.0.0 版中的 MobiLink [dbmlsync] 新功能, 176
 - 10.0.0 版中的 QAnywhere 新功能, 188
 - 10.0.0 版中的监听器 [dblsn] 新功能, 177
 - pc- 选项
 - dblsn, 11.0.0 版中的新功能, 62
 - policy 选项
 - 10.0.0 版中缺省值的更改, 190
 - port 选项
 - 10.0.0 版中已删除的 QAnywhere 选项, 190
 - ppv 选项
 - 11.0.1 版中的 MobiLink (mlsrv11) 增强, 12
 - push_notifications 选项 (见 -push 选项)
 - 10.0.0 版中重命名的 QAnywhere 选项, 190
 - push 选项
 - 10.0.0 版中的 QAnywhere 新功能, 188
 - p 选项
 - 10.0.0 版中删除的 dblic 支持, 162
 - qc 选项
 - 10.0.0 版中的 MobiLink [dbmlsync] 新功能, 185
 - qn 选项
 - 10.0.0 版中的新功能, 137
 - r 选项
 - 10.0.0 版中的监听器 [dblsn] 新功能, 177
 - sc 选项
 - 10.0.0 版中删除的数据库服务器选项, 162
 - sf 选项
 - 10.0.0 版中的新功能, 115
 - 11.0.0 版中的增强, 31, 43
 - sk 选项
 - 10.0.0 版中的新功能, 115
 - sm 选项
 - 10.0.0 版中的 MobiLink [dbmlsrv10] 新功能, 171
 - 11.0.0 版中的新功能, 27
 - 11.0.1 版中的 MobiLink 行为更改, 13
 - sn 选项
 - 10.0.0 版中的新功能, 109
 - sp 选项
 - dbmlsync, 11.0.0 版中的新功能, 62
 - sur 选项
 - 10.0.0 版中的 QAnywhere 新功能, 189
 - su 选项
 - 10.0.0 版中的新功能, 118
 - sv 选项
 - 11.0.1 版中的 MobiLink (dblsn) 增强, 12
 - s 选项
 - 9.0.1 版中的新的备份实用程序选项, 241
 - tc 选项, MobiLink (mlsrv11)
 - 11.0.0 版中的新功能, 61
 - tf 选项, MobiLink (mlsrv11)
 - 11.0.0 版中的新功能, 61
 - tu 选项
 - 9.0.1 版中的 dbmlsync 新功能, 245
 - ua 选项
 - 9.0.1 版中的新功能, 243
 - uc 服务器选项
 - 11.0.0 版中的行为更改, 55

- uf 选项
 - 10.0.0 版中的新功能, 133
 - ui 服务器选项
 - 11.0.0 版中的行为更改, 55
 - um 选项
 - 11.0.0 版中的新功能, 32
 - us 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 179
 - 9.0.1 版中的 dbmlsync 新功能, 245
 - ux 选项
 - 10.0.0 版中的 MobiLink [dbmlsync] 新功能, 173
 - 10.0.0 版中的 MobiLink [mlsrv10] 新功能, 173
 - 10.0.0 版中的 SQL Remote [dbremote] 新功能, 192
 - u 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 180
 - version
 - 11.0.0 版中的 dbisql 新功能, 76
 - ve 选项
 - 10.0.0 版中的 MobiLink [dbmlsrv10] 新功能, 171
 - vm 选项
 - 11.0.1 版中的 MobiLink (mlsrv11) 增强, 12
 - vr 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 行为更改, 181
 - vt 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 行为更改, 181
 - vu 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 行为更改, 181
 - wu 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 行为更改, 180
 - w 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 行为更改, 180
 - xd 选项
 - 10.0.1 版中的 QAnywhere 新功能, 98
 - 11.0.1 版中的数据库服务器新功能, 6
 - xf 选项
 - 10.0.0 版中的新功能, 109
 - xo 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 新功能, 171
 - xp 选项
 - 10.0.0 版中的新功能, 109
 - 11.0.0 版中的行为更改, 52
 - xs 选项
 - 10.0.0 版中的行为更改, 146
 - x 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 新语法, 170
 - y 选项
 - 10.0.0 版中删除的数据库服务器支持, 162
 - zac 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 181
 - za 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 181
 - zd 选项
 - 版本 9 中删除的 MobiLink [dbmlsrv] 功能, 272
 - zec 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 181
 - ze 选项
 - 10.0.0 版中删除的 MobiLink [mlsrv10] 选项, 181
 - zl 选项
 - 10.0.1 版中的行为更改, 93
 - zoc 选项
 - 11.0.0 版中的新功能, 43
 - zp 选项
 - 10.0.0 版中的新功能, 119
 - zus 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 新功能, 171
- ## A
- a_backup_db 结构
 - 10.0.0 版不支持的 backup_writefile 成员, 163
 - a_compress_db 结构
 - 10.0.0 版不支持, 163
 - a_db_collation 结构
 - 10.0.0 版不支持, 162
 - a_db_info 结构
 - 10.0.0 版不支持的 compressed 成员, 163
 - 10.0.0 版不支持的 wrtbufsize 成员, 163
 - 10.0.0 版不支持的 wrtnamebuffer 成员, 163
 - a_dblic_info 结构
 - 10.0.1 版中的行为更改, 104
 - a_stats_line 结构
 - 10.0.0 版不支持, 163
 - a_validate_type 枚举
 - 10.0.0 版中不建议使用的 VALIDATE_DATA 参数, 145
 - 10.0.0 版中不建议使用的 VALIDATE_FULL 参数, 145
 - 10.0.0 版中不建议使用的 VALIDATE_INDEX 参数, 145
 - 11.0.0 版中的增强, 46
 - a_writefile 结构
 - 10.0.0 版不支持, 163
 - AccentSensitive 属性
 - 10.0.0 版中的新功能, 123

ACCENT 子句
 CREATE DATABASE 语句, 不建议在 10.0.1 版
 中使用, 95

ActiveSync
 10.0.0 版中的 MobiLink 行为更改, 185

ActiveSync 提供程序活动日志
 9.0.1 版中的新功能, 245

Adaptive Server Anywhere (见 SQL Anywhere)
 10.0.0 版中重命名的 SQL Anywhere, 138
 升级到版本 11, 323

Adaptive Server Enterprise
 10.0.0 版中删除的 SQL Remote 支持, 192
 10.0.0 版中的 ODBC 驱动程序行为更改, 214

Adaptive Server Enterprise 兼容性
 10.0.0 版中的行为更改, 164

addBatch 方法
 10.0.0 版中的新功能, 131

ADO.NET 2.0 支持
 10.0.0 版中的新功能, 131

adsodbc 服务器类
 11.0.0 版中的新功能, 45

AES256_FIPS 加密算法
 11.0.0 版中的新功能, 29

AES256 加密算法
 11.0.0 版中的新功能, 29

AGENT 连接参数
 8.0.1 版中不再支持的功能, 295

allow_read_client_file 属性
 11.0.0 版中的新功能, 32

allow_read_client_file 选项
 11.0.0 版中的新功能, 31

allow_snapshot_isolation 属性
 10.0.0 版中的新功能, 120

allow_snapshot_isolation 选项
 10.0.0 版中的新功能, 110

allow_write_client_file 属性
 11.0.0 版中的新功能, 32

allow_write_client_file 选项
 11.0.0 版中的新功能, 31

ALTER DATABASE 语句
 10.0.0 版中的行为更改, 142, 160
 10.0.1 版中的增强, 90
 11.0.0 版中的增强, 36
 11.0.1 版中的增强, 4

ALTER DBSPACE 语句
 10.0.1 版中的行为更改, 92

ALTER EVENT 语句
 10.0.0 版中的行为更改, 160
 11.0.0 版中的增强, 38, 39

ALTER FUNCTION 语句
 11.0.0 版中的增强, 36

ALTER INDEX 语句
 10.0.0 版中的增强, 129

ALTER LOGIN POLICY 语句
 11.0.0 版中的新功能, 37

ALTER MATERIALIZED VIEW 语句
 10.0.0 版中的新功能, 127
 11.0.0 版中的增强, 37

AlternateMirrorServerName 属性
 11.0.0 版中的新功能, 34

AlternateServerName 属性
 10.0.0 版中的新功能, 123

ALTER PROCEDURE 语句
 11.0.0 版中的增强, 36

ALTER PUBLICATION 语句
 10.0.0 版中的 UltraLite 增强, 198

ALTER PUBLICATION 语句 [MobiLink] [SQL
 Remote]
 10.0.0 版中的行为更改, 160

ALTER SERVER 语句
 10.0.0 版中的行为更改, 160

ALTER SERVICE 语句
 10.0.0 版中的增强, 135

ALTER STATISTICS 语句
 10.0.0 版中的新功能, 128

ALTER SYNCHRONIZATION SUBSCRIPTION 语
 句 [MobiLink]
 10.0.0 版中的行为更改, 160

ALTER SYNCHRONIZATION USER 语句
 [MobiLink]
 10.0.0 版中的行为更改, 160

ALTER TABLE 语句
 10.0.0 版中的增强, 128
 10.0.0 版中的行为更改, 160
 9.0.1 版中的增强, 239

ALTER TEXT CONFIGURATION 语句
 11.0.0 版中的新功能, 38

ALTER TEXT INDEX 语句
 11.0.0 版中的新功能, 38

ALTER USER 语句
 11.0.0 版中的新功能, 37

ALTER VIEW 语句
 9.0.1 版中的增强, 239

ALTER WRITEFILE 语句

- 10.0.0 版不支持, 163
 - an_expand_db 结构
 - 10.0.0 版不支持, 163
 - ansi_blanks 选项
 - 10.0.0 版中的行为更改, 145
 - ansi_integer_overflow 选项
 - 10.0.0 版中的行为更改, 145
 - 11.0.0 版不支持, 56
 - ansi_nulls 选项
 - 9.0.1 版中的增强, 239
 - ansi_substring 属性
 - 10.0.0 版中的新功能, 120
 - ansi_substring 选项
 - 10.0.0 版中的新功能, 117
 - 11.0.0 版不支持, 56
 - 在 11.0.1 版中将受支持, 11
 - API 迁移向导
 - UltraLite 使用, 350
 - AppInfo 连接参数
 - 10.0.0 版中的增强, 113
 - 10.0.1 版中的增强, 94
 - 11.0.0 版中的增强, 28
 - apply 表达式
 - 11.0.0 版中的新功能, 39
 - ApproximateCPUTime 属性
 - 10.0.0 版中的新功能, 120
 - APP 连接参数
 - 11.0.0 版中的增强, 28
 - ArbiterState 属性
 - 10.0.0 版中的新功能, 123
 - asademo.db 文件
 - 在 10.0.0 版中重命名, 213
 - ASAJDBC
 - 10.0.0 版中已重命名, 160
 - ASANYSH 环境变量
 - 10.0.0 版中重命名, 166
 - ASANY 环境变量
 - 10.0.0 版中重命名, 166
 - ASAODBC
 - 10.0.0 版中已重命名, 160
 - ASAProv
 - 10.0.0 版中的行为更改, 140
 - ASCII
 - 10.0.0 版中的 UltraLite 增强, 201
 - ASP.NET 提供程序
 - 11.0.1 版中的新功能, 5
 - assume_distinct_servers 选项
 - 9.0.2 版中不建议使用, 230
 - ATTACH TRACING 语句
 - 10.0.0 版中的新功能, 128
 - AuditingTypes 属性
 - 10.0.0 版中的新功能, 123
 - authenticate_parameters
 - 10.0.0 版中的行为更改, 179
 - authenticate_user
 - 10.0.0 版中的行为更改, 179
 - authenticate_user_hashed
 - 10.0.0 版中的行为更改, 179
 - authenticate.sql
 - 10.0.1 版中的新功能, 94
 - Authenticated
 - 连接属性, 11.0.1 版中的新功能, 2
 - Authenticated 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - AuthType 属性
 - 11.0.0 版中的新功能, 32
 - AUTOINCREMENT
 - 在重建的数据库中保留下一个可用值, 328
 - 使用 reset.sql 脚本保留下一个可用值, 328
 - automatic_timestamp 选项
 - 11.0.0 版不支持, 56
 - AWE 高速缓存
 - 10.0.0 版中的增强, 118
 - 安全性
 - 10.0.0 版中的 MobiLink 增强, 176
 - 10.0.0 版中的 UltraLite 增强, 196
 - 11.0.0 版中的 UltraLite 增强, 69
 - 安装目录
 - 11.0.0 版中的行为更改, 83
- ## B
- background_priority 选项
 - 11.0.0 版中不建议使用, 58
 - backlog 选项
 - 10.0.0 版中已删除, 180
 - BACKUP 权限
 - 10.0.0 版中的新功能, 114
 - BACKUP 语句
 - 10.0.0 版中的增强, 114
 - 10.0.0 版中的行为更改, 160
 - 11.0.1 版中的增强, 6
 - 9.0.1 版中变量的增强支持, 239
 - 9.0.1 版中的增强, 241
 - BASE64_DECODE 函数

9.0.1 版中的新功能, 239
BASE64_ENCODE 函数
 9.0.1 版中的新功能, 239
begin_connection
 10.0.0 版中的行为更改, 179
BEGIN SNAPSHOT 语句
 11.0.0 版中的新功能, 38
big-endian UTF-16 编码
 11.0.0 版中的新功能, 44
BIGINT 数据类型
 10.0.0 版中的行为更改, 141
bin32 目录
 11.0.0 版中的行为更改, 83
bin64 目录
 11.0.0 版中的行为更改, 83
BINSEARCH 协议选项
 11.0.0 版不支持, 56
BIT_AND 函数
 10.0.0 版中的新功能, 125
BIT_LENGTH 函数
 10.0.0 版中的新功能, 125
BIT_OR 函数
 10.0.0 版中的新功能, 125
BIT_SUBSTR 函数
 10.0.0 版中的新功能, 125
BIT_XOR 函数
 10.0.0 版中的新功能, 125
BlackBerry
 11.0.0 版中的 UltraLite 增强, 71
 ULjDbT 实用程序, 17
 UltraLiteJ 数据库传输实用程序, 17
BLOB
 10.0.0 版中的 UltraLite 增强, 196
 10.0.0 版中的增强, 112
 查询, 11.0.0 版中的新功能, 27
BlobArenas 属性
 10.0.0 版中不建议使用的数据库属性, 167
BOM (字节顺序标记)
 11.0.0 版中的增强, 78
BREAK 语句
 10.0.0 版中的新功能, 129
buffer_size 选项
 10.0.0 中的新 MobiLink 客户端协议选项, 174
版本
 升级同步脚本, 343
版本 11.0.0
 升级到, 323
版本 5
 升级 SQL Remote 安装, 353
版本 8.0.0
Adaptive Server Anywhere 中的新功能, 298
MobiLink 中的新功能, 308
 新功能, 298
 行为更改, 315, 319, 320
版本 8.0.2
 新功能, 278
版本 9
 行为更改, 269
版本存储页数统计信息
 10.0.0 版中的新功能, 123
版本中的更改
 10.0.0, 107
 10.0.1, 85
 11.0.0, 23
 11.0.1, 1
 8.0.0, 298
 8.0.1, 290, 295
 8.0.2, 278, 286
 9.0.0, 254, 269
 9.0.1, 238, 250
 9.0.2, 216, 230
帮助
 技术支持, xii
保持活动状态的请求标头字段
 10.0.0 版中的新功能, 134
保留字
 版本 11 升级的疑难解答, 333
备份
 11.0.1 版中的增强, 6
备份和恢复
 11.0.0 版中的增强, 28
备份实用程序 [dbbackup]
 10.0.0 版中的增强, 114
 9.0.1 版中的增强, 241
备份数据库向导
 9.0.1 版中的增强, 243
标识符
 10.0.0 版中的行为更改, 140
 9.0.1 版中的行为更改, 251
标识文件
 11.0.0 版中的行为更改, 83
标准
 第 508 节遵从性, 301
标准和兼容性

- 第 508 节遵从性, 301
- 标准升级预防措施
 - 关于, 326
- 表加密
 - 10.0.0 版中的新功能, 112
 - 11.0.0 版中的增强, 29
- 表顺序
 - 10.0.0 版中的 UltraLite 新功能, 199
- 并发连接
 - 10.0.1 版中的 UltraLite 增强, 101
- 并行备份
 - 10.0.0 版中的新功能, 114
 - 11.0.1 版中的增强, 6
- 并行表扫描
 - 10.0.0 版中的新功能, 109
- 并行索引扫描
 - 10.0.0 版中得以增强, 109
 - 9.0.1 版中的新功能, 240
- 不建议使用的功能
 - 10.0.0 版, 107
 - 10.0.1 版, 85
 - 11.0.0 版, 23
 - 11.0.1 版, 1
 - 8.0.0 版, 298
 - 8.0.1 版, 290, 295
 - 8.0.2 版, 278
 - 9.0.1 版, 238, 250
 - 9.0.2 版, 216, 230
 - 版本 9.0.0, 254, 271
- 部署
 - 10.0.0 版不支持的写文件, 162
 - 10.0.0 版中的新向导, 169
- 部署同步模型向导
 - 10.0.0 版中的 MobiLink 新功能, 168
 - 11.0.0 版中的增强, 76
- 部署向导
 - 10.0.0 版中的新功能, 132
 - 11.0.0 版中的增强, 76
- C**
- C2 属性
 - 10.0.0 版中已删除, 162
- CacheHitsEng 属性
 - 11.0.0 版中已重命名, 54
- CacheHits 属性
 - 11.0.0 版中的新功能, 54
- CachePinned 属性
 - 10.0.0 版中的新功能, 122
- CacheReadEng 属性
 - 10.0.0 版中的新功能, 122
 - 11.0.0 版中已重命名, 54
- CacheReadWorkTable 属性
 - 11.0.0 版中的新功能, 32, 34
- CacheRead 属性
 - 11.0.0 版中的新功能, 54
- CacheSizingStatistics 属性
 - 10.0.0 版中的新功能, 126
- CALIBRATE PARALLEL READ 子句
 - 10.0.0 版中的增强, 128
- CarverHeapPages 属性
 - 10.0.0 版中的新功能, 122
- CASE 表达式
 - 11.0.0 版中的 UltraLite 增强, 70
 - 11.0.0 版中的增强, 38
- CASE 语句
 - 11.0.0 版中的 UltraLite 增强, 70
 - 11.0.0 版中的增强, 38
- CASE 子句
 - CREATE DATABASE 语句, 不建议在 10.0.1 版中使用, 95
- CAST 函数
 - 10.0.0 版中的行为更改, 146
- CatalogCollation 属性
 - 10.0.1 版中的新功能, 89
- CATALOGS 行集, OLE DB
 - 11.0.1 版中的新功能, 5
- Certicom
 - Security Builder GSE 版本, 10.0.1 版中的增强, 103
 - 在 2002 年 7 月后不再颁发证书, 287
- certificate_password 协议选项
 - 11.0.0 版不支持, 83
 - 11.0.0 版中的行为更改, 63
- certificate 协议选项
 - 11.0.0 版不支持, 83
 - 11.0.0 版中的行为更改, 63
- CharSet 属性
 - 10.0.0 版中的增强, 111
- CHECKPOINT 语句
 - 10.0.1 版中的 UltraLite 增强, 101
- Checksum 属性
 - 9.0.1 版中的新功能, 240
- 重定向器
 - 10.0.0 版中的增强, 173

- 11.0.0 版中不建议使用, 61
- 9.0.1 版中的 Apache Web 服务器支持, 246
- 重复文本索引
 - 11.0.1 版中的行为更改, 8
- 重建
 - 升级预防措施, 326
 - 版本 11 数据库, 326, 329, 333
 - 疑难解答, 332
 - 限制, 329
- 重建版本 9 及更早版本的数据库
 - 关于, 329
- 重建失败
 - 升级数据库到版本 11, 332
- 重建数据库
 - 10.0.0 版中的增强, 133
 - 10.0.0 版的必需条件, 138
 - 11.0.0 版中的增强, 27
 - 重建失败, 332
- CleanablePagesAdded 属性
 - 10.0.0 版中的新功能, 123
- CleanablePagesCleaned 属性
 - 10.0.0 版中的新功能, 123
- ClientNodeAddress 属性
 - 11.0.0 版中的新功能, 32
- ClientStmtCacheHits 属性
 - 10.0.1 版中的新功能, 87
- ClientStmtCacheMisses 属性
 - 10.0.1 版中的新功能, 87
- ClusteredIndexes 属性
 - 10.0.0 版中不建议使用的数据库属性, 167
- CodeWarrior
 - 支持的版本, 274
- CodeWarrior 模板
 - 9.0.1 版中的 UltraLite C++ 组件新功能, 247
- collect_statistics_on_dml_updates 属性
 - 10.0.0 版中的新功能, 120
- collect_statistics_on_dml_updates 选项
 - 10.0.0 版中的新功能, 117
- CollectStatistics 属性
 - 10.0.0 版中的新功能, 126
- CommBufferSize 连接参数
 - 10.0.0 版中的增强, 113
- COMMENT 语句
 - 10.0.0 版中的增强, 128
 - 10.0.0 版中的行为更改, 160
 - 11.0.0 版中的增强, 38, 39
 - 不建议使用的子句, 11
- COMMIT 语句
 - 9.0.1 版中的 UltraLite 动态 SQL 新功能, 246
- COMPARE 函数
 - 10.0.1 版中的增强, 89
- CompressedBTrees 属性
 - 10.0.0 版中不建议使用的数据库属性, 167
- CompressionThreshold 连接参数
 - 10.0.0 版中的增强, 113
- Compression 属性
 - 10.0.0 版不支持, 163
- COMPRESS 函数
 - 10.0.0 版中的增强, 127
 - 9.0.1 版中的新功能, 239
- confirmation_handler
 - 10.0.0 版中的新功能, 177
- conflicted_deletes
 - 10.0.0 版中的新行为, 182
- conflicted_inserts
 - 10.0.0 版中的新行为, 182
- conflicted_updates
 - 10.0.0 版中的新行为, 182
- conn_auditing 属性
 - 10.0.0 版中的新功能, 120
- conn_auditing 选项
 - 10.0.0 版中的新功能, 113
- CONNECTION_EXTENDED_PROPERTY 函数
 - 10.0.0 版中的新功能, 126
- CONNECTION_PROPERTY 函数
 - 10.0.0 版中的增强, 126
- ConnsDisabled 属性
 - 10.0.0 版中的行为更改, 139
- ConsoleLogFile 属性
 - 10.0.0 版中的新功能, 126
 - 9.0.1 版中的新功能, 243
- ConsoleLogMaxSize 属性
 - 10.0.0 版中的新功能, 126
- CONTAINS 搜索条件
 - 11.0.0 版中的新功能, 37
- contd_timeout 选项
 - 10.0.0 版中被替换的 MobiLink 客户端协议选项, 174
- CONTINUE 语句
 - 10.0.0 版中的增强, 129
- CORR 函数
 - 9.0.1 版中的新功能, 238
- COUNT_SET_BITS 函数
 - 10.0.0 版中的新功能, 125

- COVAR_POP 函数
 - 9.0.1 版中的新功能, 238
- COVAR_SAMP 函数
 - 9.0.1 版中的新功能, 238
- createcert 实用程序
 - 10.0.1 版中的新功能, 103
- CREATE COMPRESSED DATABASE 语句
 - 10.0.0 版不支持, 163
- CREATE DATABASE 语句
 - 10.0.0 版中的 BLANK PADDING 子句行为更改, 138
 - 10.0.0 版中的增强, 115, 128
 - 10.0.0 版中的行为更改, 142
 - 10.0.1 版中的增强, 87, 89
 - 10.0.1 版中的行为更改, 93
 - 9.0.1 版中的增强, 240
- CREATE DBSPACE 语句
 - 10.0.1 版中的行为更改, 92
- CREATE DECRYPTED DATABASE 语句
 - 11.0.1 版中的新功能, 4
- CREATE ENCRYPTED DATABASE 语句
 - 11.0.1 版中的新功能, 4
- CREATE ENCRYPTED FILE 语句
 - 10.0.0 版中的增强, 128
- CREATE EVENT 语句
 - 11.0.0 版中的增强, 39
- CREATE EXPANDED DATABASE 语句
 - 10.0.0 版不支持, 163
- CREATE FUNCTION 语句
 - 11.0.1 版中的增强, 4, 5
 - SET 子句, 10.0.1 版中的新功能, 90
- CREATE INDEX 语句
 - 10.0.0 版中的行为更改, 161
 - 9.0.1 版中的 UltraLite 动态 SQL 新功能, 246
- createkey 实用程序
 - 11.0.0 版中的新功能, 62
- CREATE LOCAL TEMPORARY TABLE 语句
 - 10.0.0 版中的增强, 130
- CREATE LOGIN POLICY 语句
 - 11.0.0 版中的新功能, 37
- CREATE MATERIALIZED VIEW 语句
 - 10.0.0 版中的新功能, 127
 - 11.0.0 版中的增强, 36, 38
- CREATE PROCEDURE 语句
 - 11.0.1 版中的增强, 4, 5
 - SET 子句, 10.0.1 版中的新功能, 90
- CREATE PUBLICATION 语句
 - 10.0.0 版中的 UltraLite 增强, 198
- CREATE SERVER 语句
 - 10.0.0 版中的行为更改, 160
- CREATE SERVICE 语句
 - 10.0.0 版中的增强, 135
- CREATE SYNCHRONIZATION DEFINITION 语句
 - 10.0.0 版中已删除, 184
- CREATE SYNCHRONIZATION SITE 语句
 - 10.0.0 版中已删除, 184
- CREATE SYNCHRONIZATION TEMPLATE 语句
 - 10.0.0 版中已删除, 184
- CREATE TABLE 语句
 - 10.0.0 版中的增强, 128
 - 11.0.1 版中的增强, 5
 - 9.0.1 版中的 UltraLite 动态 SQL 新功能, 246
- CREATE TEXT CONFIGURATION 语句
 - 11.0.0 版中的新功能, 38
- CREATE TEXT INDEX 语句
 - 11.0.0 版中的新功能, 38
- CREATE TRIGGER 语句
 - 11.0.1 版中的增强, 4
 - 版本 11 升级的疑难解答, 333
- CREATE USER 语句
 - 11.0.0 版中的新功能, 37
- CREATE VIEW 语句
 - 11.0.1 版中的增强, 4
- CREATE WRITEFILE 语句
 - 10.0.0 版不支持, 163
- CROSS APPLY 子句
 - 11.0.0 版中的新功能, 39
- CUBE 操作
 - 9.0.1 版中的新功能, 238
- CUME_DIST 函数
 - 9.0.1 版中的新功能, 238
- CURRENT_TIMESTAMP
 - 9.0.1 版中的行为更改, 250
- CURRENT_USER
 - 9.0.1 版中的行为更改, 250
- CurrentLineNumber 属性
 - 10.0.0 版中的新功能, 120
- CurrentProcedure 属性
 - 10.0.0 版中的新功能, 120
- CustDB
 - 11.0.0 版中的 UltraLite 行为更改, 72
- CustDB UltraLite 示例
 - 11.0.0 版中的行为更改, 72
- 参考数据库

- UltraLite 升级, 346
 - 参照完整性
 - 10.0.0 版中的 UltraLite 新功能, 199
 - 插件模块
 - 10.0.0 版中的 UltraLite 新功能, 196
 - 查询
 - BLOB, 11.0.0 版中的新功能, 27
 - 文件, 11.0.0 版中的新功能, 27
 - 查询编辑器
 - 9.0.1 版中的增强, 238
 - 查询内并行机制
 - 10.0.0 版中的新功能, 109
 - 10.0.1 版中的行为更改, 92
 - 查询性能
 - 11.0.1 版中的增强, 5
 - 查询优化
 - 11.0.1 版中的增强, 5
 - 查找详细信息并请求技术协助
 - 技术支持, xiii
 - 抽取实用程序
 - 11.0.0 版中的增强, 66
 - 11.0.0 版中的行为更改, 54, 55
 - 抽取数据库向导
 - 10.0.0 版中的行为更改, 144
 - 11.0.0 版中的 UltraLite 增强, 69
 - UltraLite 使用, 351
 - 初始化实用程序 [dbinit]
 - 10.0.0 版中的 -b 选项行为更改, 138
 - 10.0.0 版中的增强, 115
 - 10.0.0 版中的行为更改, 142
 - 10.0.1 版中的增强, 91
 - 9.0.1 版中的增强, 240
 - 触发器
 - 10.0.1 版中的增强, 91
 - 传送层安全性
 - 从何处获取证书, 287
 - 串行协议
 - 自 8.0.2 版起 MobiLink 不再支持, 287
 - 窗口函数
 - 9.0.1 版中的新功能, 238
 - 创建备份映像向导
 - 9.0.1 版中的增强, 243
 - 创建触发器向导
 - 10.0.1 版中的增强, 90
 - 创建服务向导
 - 11.0.1 版中的增强, 19
 - 9.0.1 版中的增强, 243
 - 创建数据库向导
 - 10.0.1 版中的增强, 91
 - 11.0.0 版中的增强, 76
 - 创建同步模型向导
 - 10.0.0 版中的 MobiLink 新功能, 168
 - 创建同步配置文件向导
 - 11.0.0 版中的新功能, 74
 - 创建维护计划向导
 - 11.0.1 版中的增强, 19
 - 创建写文件向导
 - 10.0.0 版不支持, 163
 - 创建用户向导
 - 11.0.0 版中的增强, 76
 - 创建远程服务器向导
 - 9.0.1 版中的增强, 243
 - 创建者 ID
 - 10.0.0 版中的 UltraLite 增强, 197
 - 创建直通脚本向导
 - 11.0.0 版中的新功能, 74
 - 创建直通下载向导
 - 11.0.0 版中的新功能, 74
 - 创建自定义归类向导
 - 10.0.0 版不支持, 162
 - 磁盘读取工作表统计信息
 - 11.0.0 版中的新功能, 34
 - 磁盘空间
 - 9.0.2 版中的新功能, 219
 - 磁盘已满
 - 9.0.2 版中的新功能, 219
 - 从结果集生成 sql 语句
 - 11.0.0 版中的增强, 77
 - 存储过程
 - 10.0.0 版中的增强, 130
 - 错误
 - 提供反馈, xii
 - 错误报告
 - 10.0.0 版中的新功能, 212
 - 10.0.1 版中的增强, 92
 - 11.0.0 版中的增强, 82
 - 错误处理
 - 9.0.1 版中的 UltraLite C/C++ 增强, 247
 - 错误字符串
 - 10.0.0 版中的 UltraLite 增强, 197
- ## D
- DataWindow .NET
 - 10.0.0 版中的新功能, 212

- 10.0.1 版中的 Vista 支持, 105
- date_format 选项
 - 10.0.0 版中的行为更改, 145
- DATEADD 函数
 - 10.0.0 版中的 QAnywhere 新功能, 189
- DATEPART 函数
 - 10.0.0 版中的 QAnywhere 新功能, 189
- DATETIME 函数
 - 10.0.0 版中的 QAnywhere 新功能, 189
- DB_BACKUP_WRITEFILE 参数
 - 10.0.0 版不支持, 163
- db_backup 函数
 - 10.0.0 版不支持的 DB_BACKUP_WRITEFILE 参数, 163
 - 10.0.0 版中的增强, 114
- DB_EXTENDED_PROPERTY 函数
 - 10.0.0 版中的增强, 126
 - 10.0.1 版中的增强, 89
- db_locate_servers_ex 函数
 - 10.0.0 版中的增强, 132
- DB_PROPERTY 函数
 - 10.0.0 版中的增强, 126
- db_register_a_callback 函数
 - 10.0.0 版中的行为更改, 139
- dbasdesk.dll
 - 在版本 10.0.0 中名称更改为 mlasdesk.dll, 185
- dbasdev.dll
 - 在版本 10.0.0 中名称更改为 mlasdev.dll, 185
- dBase 格式, INPUT 语句
 - 11.0.0 版中删除了支持, 79
- dBase 格式, OUTPUT 语句
 - 11.0.0 版中删除了支持, 79
- dbasinst 实用程序
 - 在版本 10.0.0 中名称更改为 mlasinst, 185
- dbbackup 实用程序
 - 10.0.0 版中的增强, 114
 - 9.0.1 版中的增强, 241
- DB_CALLBACK_FINISH 回调参数
 - 10.0.0 版中的行为更改, 139
- DB_CALLBACK_START 回调参数
 - 10.0.0 版中的行为更改, 139
- dbcapi.dll
 - 11.0.0 版中的新功能, 40
- DBChangeWriteFile 函数
 - 10.0.0 版不支持, 163
- DBCcollate 函数
 - 10.0.0 版不支持, 162
- dbcollat 实用程序
 - 10.0.0 版不支持, 162
- DBCompress 函数
 - 10.0.0 版不支持, 163
- dbconsole 实用程序
 - 10.0.0 版中的增强, 208
 - 11.0.0 版中的增强, 80
 - 9.0.1 版中的增强, 241
- DBCreatedVersion 函数
 - 10.0.1 版中的新功能, 91
- DBCreateWriteFile 函数
 - 10.0.0 版不支持, 163
- DBD::ASAny
 - 10.0.0 版中更改的名称, 132
 - 9.0.1 版中的新 Perl 驱动程序, 241
- dbdata10.dll
 - 10.0.1 版中的行为更改, 93
- dbdsn 实用程序
 - 10.0.0 版中的增强, 115
 - 10.0.0 版中的行为更改, 162
 - 10.0.1 版中的增强, 92
- dbelevate10.exe
 - 10.0.1 版中的 Vista 支持, 105
 - 10.0.1 版中的新功能, 104
- DBExpand 函数
 - 10.0.0 版不支持, 163
- dbexpand 实用程序
 - 10.0.0 版不支持, 163
- dbhist 实用程序
 - 10.0.0 版中的增强, 115
- dbinfo 实用程序
 - 10.0.0 版中的增强, 115
- dbinit 实用程序
 - 10.0.0 版中的 -b 选项行为更改, 138
 - 10.0.0 版中的增强, 115
 - 10.0.0 版中的行为更改, 142
 - 10.0.1 版中的增强, 91
 - 9.0.1 版中的增强, 240
- dbisql.exe
 - 11.0.0 版中的行为更改, 79
- dbisqlc 实用程序
 - 11.0.1 版中的行为更改, 10
- dbisqlg
 - 11.0.0 版中已重命名, 79
- dbisql 实用程序
 - 10.0.0 版中的增强, 208
 - 10.0.0 版中的行为更改, 208

- 11.0.0 版中的行为更改, 55
- dblang 实用程序
 - 10.0.1 版中的行为更改, 93
- DBLIB
 - 11.0.0 版中的新功能, 51
- dblibtb.dll
 - 11.0.0 版中的新功能, 51
- dblibtw.dll
 - 11.0.0 版中的新功能, 51
- dblic 实用程序
 - 10.0.0 版中的行为更改, 162
 - 10.0.1 版中的行为更改, 104
- dblocate 实用程序
 - 10.0.0 版中的新功能, 116
 - 10.0.0 版中的行为更改, 143
 - 9.0.1 版中的增强, 241
- DBLTM 服务类型
 - 10.0.0 版中的新功能, 116
- dbltm 实用程序
 - 10.0.0 版中的增强, 116
- dbmlctr9.dll
 - 在版本 10.0.0 中删除 MobiLink Windows 性能监
控器支持, 186
- dbmlmon 实用程序
 - 在版本 10.0.0 中名称更改为 mlmon, 185
- dbmlsrv.mle
 - 在版本 10.0.0 中名称更改为 mlsrv10.mle, 185
- dbmlsrv9
 - 在版本 10.0.0 中名称更改为 mlsrv10, 185
- dbmlstop 实用程序
 - 在版本 10.0.0 中名称更改为 mlstop, 185
- dbmlsync -sp 选项
 - 11.0.0 版中的新功能, 62
- dbmlsync StreamCompression 扩展选项
 - 11.0.0 版不支持, 63
- dbmlsync 集成组件
 - 11.0.0 版中不建议使用, 63
 - 9.0.1 版中的新功能, 245
- dbmlsync 消息日志扫描
 - 11.0.0 版中的增强, 62
- dbmluser 实用程序
 - 在版本 10.0.0 中名称更改为 mluser, 185
- dbmodenv
 - 11.0.1 版中的行为更改, 10
- dbns10 实用程序
 - 10.0.0 版中的新功能, 116
 - 11.0.0 版中的行为更改, 58
- dbping 实用程序
 - 10.0.0 版中的增强, 116
 - 10.0.0 版中的行为更改, 142
- dbrunsql 实用程序
 - 10.0.0 版中的新功能, 133
- dbshrink 实用程序
 - 10.0.0 版不支持, 163
- dbspace
 - 10.0.1 版中的增强, 91
 - 10.0.1 版中的行为更改, 92
 - 11.0.0 版中的行为更改, 51, 55
- dbsrv10.lic
 - 10.0.1 版中的新功能, 105
- DBStatusWriteFile 函数
 - 10.0.0 版不支持, 163
- dbsupport 实用程序
 - 10.0.0 版中的新功能, 116, 212
 - 10.0.1 版中的增强, 92
 - 11.0.0 版中的增强, 82
- dbsvc 实用程序
 - 10.0.0 版中的增强, 116, 133
 - 10.0.0 版中的行为更改, 143
 - 11.0.1 版中的增强, 3
- dbtltstb.dll
 - 11.0.0 版中的新功能, 51
- dbtltstw.dll
 - 11.0.0 版中的新功能, 51
- DBTools 接口
 - 9.0.2 版中不建议使用 libdbtool9.so, 230
- dbtran 实用程序
 - 11.0.0 版中的增强, 31
- dbuleng9
 - 9.0.1 版中的 UltraLite 的新功能, 247
- dbunload 实用程序
 - 10.0.0 版中的增强, 116
 - 10.0.0 版中的行为更改, 142, 143
 - 10.0.1 版中的行为更改, 94
 - 11.0.0 版中的增强, 30
 - 11.0.0 版中的行为更改, 54
 - 11.0.1 版中的增强, 3
 - 9.0.1 版中的增强, 241
 - 版本 11 升级的疑难解答, 333
- dbupgrad 实用程序
 - 10.0.0 版中的行为更改, 138, 142
 - 11.0.1 版中的增强, 3
- dbvalid 实用程序
 - 10.0.0 版中不建议使用的选项, 161

- 10.0.0 版中的增强, 117
- 11.0.0 版中的增强, 30
- 9.0.1 版中的增强, 240
- 9.0.1 版中的行为更改, 250
- dbwrite 实用程序
 - 10.0.0 版不支持, 163
- dbxtract 实用程序
 - 10.0.0 版中的更改, 192
 - 11.0.0 版中的增强, 66
 - 11.0.0 版中的行为更改, 54, 55
- DCX
 - 11.0.1 版中的新功能, 21
 - 关于, viii
- Deadlock 系统事件
 - 11.0.0 版中的新功能, 46
- debug_messages 选项
 - 9.0.1 版中的增强, 240
- DebuggingInformation 属性
 - 10.0.0 版中的新功能, 122, 126
- DEBUG 连接参数
 - 8.0.1 版中不再支持的功能, 295
- DECLARE LOCAL TEMPORARY TABLE 语句
 - 9.0.1 版中的行为更改, 251
 - 版本 11 升级的疑难解答, 333
- DECOMPRESS 函数
 - 10.0.0 版中的增强, 127
 - 9.0.1 版中的新功能, 239
- DECRYPT 函数
 - 11.0.0 版中的增强, 29
 - 9.0.1 版中的新功能, 239
- dedicated_task 选项
 - 9.0.1 版中的新功能, 242
- default_dbspace 属性
 - 10.0.0 版中的新功能, 120
- default_dbspace 选项
 - 10.0.0 版中的新功能, 117
- default_isql_encoding 选项
 - 9.0.1 版中的新功能, 242
- default_timestamp_increment 选项
 - 10.0.1 版中的行为更改, 93
- DefaultNcharCollation 属性
 - 10.0.0 版中的新功能, 122
- DEFAULT VALUES 子句
 - 11.0.1 版中的新功能, 4
- delete_old_logs 选项
 - 10.0.1 版中的增强, 97
- DELETE 语句
 - 10.0.0 版中的增强, 130
 - 10.0.1 版中的增强, 88, 91
 - 11.0.0 版中的增强, 40, 45
 - 11.0.0 版中的行为更改, 54
 - 11.0.1 版中的增强, 5
- demo.db 文件
 - 10.0.0 版中的示例数据库更改, 213
 - 11.0.0 版中的增强, 82
- DENSE_RANK 函数
 - 9.0.1 版中的新功能, 238
- describe_java_format 选项
 - 10.0.0 版不支持, 141
- DESCRIBE 语句
 - 10.0.0 版中的增强, 208
 - 10.0.1 版中的行为更改, 93
 - 11.0.0 版中的增强, 78
- DETACH TRACING 语句
 - 10.0.0 版中的新功能, 128
- 调度创建向导
 - 10.0.0 版中的新功能, 206
- DISH 服务
 - 10.0.1 版中的行为更改, 93
- DiskReadEng 属性
 - 10.0.0 版中的新功能, 122
 - 11.0.0 版中已重命名, 54
- DiskReadHintPages 属性
 - 11.0.0 版中的新功能, 54
- DiskReadHintScatterLimit 属性
 - 11.0.0 版中的新功能, 54
- DiskReadHint 属性
 - 11.0.0 版中的新功能, 54
- DiskReadWorkTable 属性
 - 11.0.0 版中的新功能, 32, 34
- DiskRead 属性
 - 11.0.0 版中的新功能, 54
- DiskRetryReadScatter
 - 11.0.0 版中的新功能, 33
- DiskRetryReadScatter 属性
 - 11.0.0 版中的新功能, 34
- DiskRetryRead 属性
 - 11.0.0 版中的新功能, 33
- DiskRetryWrite 属性
 - 11.0.0 版中的新功能, 33
- DiskSyncRead 属性
 - 11.0.0 版中的新功能, 32, 34
- DiskSyncWrite 属性
 - 11.0.0 版中的新功能, 32, 34

- DiskWaitRead 属性
 - 11.0.0 版中的新功能, 32
- DiskWaitWrite 属性
 - 11.0.0 版中的新功能, 32
- DiskWriteHintPages 属性
 - 11.0.0 版中的新功能, 32, 34
- DiskWriteHint 属性
 - 11.0.0 版中的新功能, 32, 34
- DISTINCT 子句
 - 10.0.0 版中的 UltraLite 增强, 198
- divide_by_zero_error 选项
 - 11.0.0 版不支持, 56
- DLL 协议选项
 - 10.0.0 版中的行为更改, 166
 - 11.0.0 版不支持, 58
- DocCommentXchange (DCX)
 - 11.0.1 版中的新功能, 21
 - 关于, viii
- Driver 连接参数
 - 11.0.0 版中的增强, 41
- DriveType 属性
 - 9.0.1 版中的新功能, 243
- DROP DBSPACE 语句
 - 10.0.1 版中的行为更改, 92
- DROP EVENT 语句
 - 11.0.0 版中的增强, 39
 - 11.0.1 版中的增强, 4
- DROP FUNCTION 语句
 - 11.0.1 版中的增强, 4
- DROP INDEX 语句
 - 9.0.1 版中的 UltraLite 动态 SQL 新功能, 246
- DROP LOGIN POLICY 语句
 - 11.0.0 版中的新功能, 37
- DROP MATERIALIZED VIEW 语句
 - 11.0.1 版中的增强, 4
- DROP PROCEDURE 语句
 - 11.0.1 版中的增强, 4
- DROP PUBLICATION 语句
 - 10.0.0 版中的 UltraLite 增强, 198
- DROP TABLE 语句
 - 11.0.1 版中的增强, 4
 - 9.0.1 版中的 UltraLite 动态 SQL 新功能, 246
- DROP TEXT CONFIGURATION 语句
 - 11.0.0 版中的新功能, 38
- DROP TEXT INDEX 语句
 - 11.0.0 版中的新功能, 38
- DROP TRIGGER 语句
 - 11.0.1 版中的增强, 4
- DROP USER 语句
 - 11.0.0 版中的新功能, 37
- DROP VARIABLE 语句
 - 11.0.1 版中的增强, 4
- DROP VIEW 语句
 - 11.0.1 版中的增强, 4
- DROP 语句
 - 10.0.0 版中的增强, 127
- DSN 连接参数
 - 10.0.0 版中的行为更改, 140
- 堆被锁可重定位统计信息
 - 10.0.0 版中的新功能, 123
- 堆可重定位统计信息
 - 10.0.0 版中的新功能, 123
- 打印
 - 11.0.0 版中的 Interactive SQL 增强, 77
- 代理表创建向导
 - 9.0.1 版中的增强, 243
- 代理端口
 - 10.0.0 版中的增强, 133
- 单步
 - 11.0.0 版中的新功能, 75
- 档案备份
 - 11.0.1 版中的增强, 6
 - 9.0.1 版中的增强, 241
- 档案消息存储库
 - 11.0.0 版中的增强, 64
- 导出向导
 - 11.0.0 版中的增强, 79
 - 9.0.1 版中的增强, 242
- 导入向导
 - 11.0.0 版中的增强, 79
 - 9.0.1 版中的增强, 242
- 登录策略
 - 11.0.0 版中的新功能, 25
- 第 508 节
 - 遵从性, 301
- 定制归类
 - 10.0.1 版中的新功能, 89
- 动态 SQL
 - 10.0.0 版中的 UltraLite 新功能, 195
- 动态陷阱
 - 10.0.1 版中的增强, 92
- 端对端加密
 - 11.0.0 版中的 UltraLite 增强, 69
 - 11.0.0 版中的新功能, 62

- 堆查询处理统计信息
 - 10.0.0 版中的新功能, 123
- 堆雕刻统计信息
 - 10.0.0 版中的新功能, 123
- 多表连接
 - UltraLite 动态 SQL, 267
- 多个版本
 - Adaptive Server Anywhere, 326
 - UltraLite, 345
- 多个结果集
 - 10.0.0 版中的 Interactive SQL 行为更改, 208
- 多数据库支持
 - 9.0.1 版中的 UltraLite 的新功能, 247
- E**
- EBF
 - 使用数据库镜像时应用, 335
- ECC
 - 10.0.0 版中的 UltraLite 新功能, 199
 - 10.0.0 版本中与 HTTPS 一起使用, 176
- ecc_tls
 - 10.0.0 版中重命名的 MobiLink [mlsrv10] 选项, 181
- Elevate 连接参数
 - 11.0.0 版中的新功能, 28
- ENAME 协议选项
 - 11.0.0 版不支持, 56
- encrypt_aes_random_iv 选项
 - 10.0.1 版中的新功能, 92
 - 11.0.0 版不支持, 58
- ENCRYPT_PASSWORD 连接属性
 - 11.0.0 版中的新功能, 29
- EncryptionScope 属性
 - 10.0.0 版中的新功能, 123
- Encryption 连接参数
 - 10.0.0 版中的行为更改, 146
- Encryption 属性
 - 10.0.0 版中的行为更改, 140
- ENCRYPT 函数
 - 11.0.0 版中的增强, 29
 - 9.0.1 版中的新功能, 239
- environment.plist
 - 11.0.1 版中的行为更改, 10
- error_handler
 - 10.0.0 版中的新功能, 177
- EVENT_PARAMETER 函数
 - 10.0.0 版中的增强, 109
 - 11.0.0 版中的增强, 36
- EventTypeDesc
 - 11.0.0 版中的新功能, 33
- EventTypeName
 - 11.0.0 版中的新功能, 33
- example_upload_cursor
 - 10.0.0 版中已删除, 180
- example_upload_delete
 - 10.0.0 版中已删除, 180
- example_upload_insert
 - 10.0.0 版中已删除, 180
- example_upload_update
 - 10.0.0 版中已删除, 180
- EXCEL 格式, INPUT 语句
 - 11.0.0 版中删除了支持, 79
- EXCEL 格式, OUTPUT 语句
 - 11.0.0 版中删除了支持, 79
- ExceptionHandler2 委派 [QA .NET API]
 - 10.0.1 版中的 QAnywhere 新功能, 98
- ExceptionHandler 委派 [QA .NET API]
 - 10.0.1 版中的 QAnywhere 新功能, 98
- EXCEPT 语句
 - 10.0.1 版中的增强, 88
 - 11.0.0 版中的增强, 40
 - 11.0.0 版中的行为更改, 54
- ExchangeTasksCompleted 属性
 - 10.0.1 版中的新功能, 92
- ExchangeTasks 属性
 - 10.0.0 版中的新功能, 122
- EXECUTE IMMEDIATE 语句
 - 9.0.1 版中的行为更改, 251
- EXIT 语句
 - 10.0.0 版中的行为更改, 209
- ExprCacheAbandons 属性
 - 10.0.0 版中的新功能, 120
- ExprCacheDropsToReadOnly 属性
 - 10.0.0 版中的新功能, 120
- ExprCacheEvicts 属性
 - 10.0.0 版中的新功能, 120
- ExprCacheHits 属性
 - 10.0.0 版中的新功能, 120
- ExprCacheInserts 属性
 - 10.0.0 版中的新功能, 120
- ExprCacheLookups 属性
 - 10.0.0 版中的新功能, 120
- ExprCacheResumesOfReadWrite 属性
 - 10.0.0 版中的新功能, 120

-
- ExprStarts 属性
 - 10.0.0 版中的新功能, 120
 - ExtendedName 协议选项
 - 11.0.0 版不支持, 56
 - F**
 - FileSize 属性
 - 10.0.0 版中删除的 writefile dbspace 支持, 163
 - FileVersion 属性
 - 10.0.0 版中不建议使用的数据库属性, 167
 - FIPS
 - 10.0.0 版中的 UltraLite 新同步功能, 196
 - 10.0.0 版中的增强, 114
 - 10.0.0 版中的行为更改, 140
 - 10.0.1 版中的 UltraLite 行为更改, 102
 - 11.0.0 版中的增强, 29
 - FIPS 选项
 - 10.0.0 版中的 MobiLink [mlsrv10] 新功能, 176
 - 10.0.0 版中的 MobiLink [mluser] 新功能, 176
 - FIRST_VALUE 函数
 - 10.0.1 版中的新功能, 91
 - 11.0.1 版中的新功能, 3
 - FirstOption 属性
 - 10.0.0 版中的新功能, 122
 - float_as_double 选项
 - 11.0.0 版不支持, 56
 - FORCE NO OPTIMIZATION 子句
 - 11.0.1 版中的新功能, 5
 - FORMAT 子句
 - FORMAT ASCII, 不建议在 11.0.0 版中使用, 55
 - FOR OLAP WORKLOAD 选项
 - 10.0.0 版中的新功能, 130
 - FOXPRO 格式, INPUT 语句
 - 11.0.0 版中删除了支持, 79
 - FOXPRO 格式, OUTPUT 语句
 - 11.0.0 版中删除了支持, 79
 - FreePageBitMaps 属性
 - 10.0.0 版中不建议使用的数据库属性, 167
 - FreePages 属性
 - 10.0.0 版中删除的 writefile dbspace 支持, 163
 - FROM 子句
 - 11.0.0 版中的增强, 25
 - 9.0.1 版中的增强, 240
 - FunctionMaxParms 属性
 - 10.0.0 版中的新功能, 122
 - FunctionMinParms 属性
 - 10.0.0 版中的新功能, 122
 - FunctionName 属性
 - 10.0.0 版中的新功能, 122
 - 发送列名
 - 10.0.0 版中的行为更改, 174
 - 翻译日志文件实用程序 [dbtran]
 - 11.0.0 版中的增强, 31
 - 反馈
 - 报告错误, xii
 - 提供, xii
 - 文档, xii
 - 请求更新, xii
 - 非阻塞下载确认, QAnywhere
 - 11.0.0 版中的行为更改, 65
 - 非阻塞下载确认脚本
 - 11.0.0 版中的增强, 61
 - 分散读取
 - 10.0.0 版中的行为更改, 140
 - 分析模式
 - 10.0.0 版中的新功能, 110
 - 分组提交刷新
 - 10.0.1 版中的 UltraLite 增强, 101
 - 服务
 - 10.0.0 版中的增强, 116
 - 9.0.1 版中的增强, 243
 - 服务器传输规则
 - 10.0.0 版中的增强, 189
 - 服务器错误代码
 - 9.0 版中的 MobiLink 的行为更改, 272
 - 服务器端备份
 - 11.0.1 版中的增强, 6
 - 服务器管理请求
 - 10.0.0 版中的 QAnywhere 新功能, 189
 - 11.0.0 版中的增强, 65
 - 服务器枚举实用程序 [dblocate]
 - 10.0.0 版中的新功能, 116
 - 10.0.0 版中的行为更改, 143
 - 9.0.1 版中的增强, 241
 - 服务器名称
 - 10.0.0 版中的行为更改, 140
 - 服务器启动的同步
 - 10.0.0 版中的新功能, 177
 - 服务器启动的同步监听器
 - 11.0.0 版中的增强, 62
 - 服务器群中服务器启动的同步
 - 11.0.0 版中的增强, 62
 - 服务器授权实用程序 [dblic]
 - 10.0.0 版中的行为更改, 162

- 10.0.1 版中的行为更改, 104
- 服务器属性
 - 10.0.0 版中的行为更改, 139
 - 11.0.0 版中的新功能, 33
- 服务器属性文件
 - 10.0.0 版中不建议使用的 QAnywhere 功能, 190
- 服务器消息
 - 11.0.0 版中的增强, 46
- 服务器组
 - 10.0.0 版中的 MobiLink 新功能, 173
- 服务实用程序 [dbsvc]
 - 10.0.0 版中的增强, 116, 133
 - 10.0.0 版中的行为更改, 143
 - 11.0.1 版中的增强, 3
- 服务属性窗口
 - 9.0.1 版中的增强, 243
- 复制代理
 - 10.0.0 版中的增强, 116
- 负载平衡, MobiLink 服务器群
 - 11.0.0 版中的新功能, 60
- G**
- 高速缓存大小的动态调整
 - 10.0.0 版中的增强, 132
- gencert 实用程序
 - 11.0.0 版中已删除, 82
 - 不建议在 10.0.1 版中使用, 104
- GET_BIT 函数
 - 10.0.0 版中的新功能, 125
- GetData 属性
 - 10.0.0 版中的新功能, 120
- getlastdownloadtime
 - 11.0.1 版中的增强, 12
- getPageReads() 方法
 - 11.0.1 版中的 UltraLiteJ 增强, 17
- getPageWrites 方法
 - 11.0.1 版中的 UltraLiteJ 增强, 17
- getPropertyNames
 - 10.0.0 版中已删除的 QAnywhere C++ 函数, 190
- getStreamErrorParameters 方法
 - 11.0.1 版中的 UltraLite 增强, 15
- Government Services Edition
 - Certicom Security Builder, 10.0.1 版中的增强, 103
- GROUPING SETS 操作
 - 9.0.1 版中的新功能, 238
- GUID 标识符
 - 11.0.0 版中的 UltraLite 增强, 69
- gzip 算法
 - 10.0.0 版中的新功能, 127
- 高度加密
 - 10.0.1 版中的增强, 87
- 高可用性
 - 10.0.0 版中的新功能, 109, 110
- 高速缓存
 - 10.0.0 版中的行为更改, 143
 - 9.0.1 版中的增强, 244
- 高速缓存大小
 - 10.0.0 版中的 UltraLite 行为更改, 199
 - 10.0.0 版中的行为更改, 143
 - 9.0.1 版中的增强, 244
- 高速缓存读取工作表统计信息
 - 11.0.0 版中的新功能, 34
- 高速缓存多页分配统计信息
 - 10.0.0 版中的新功能, 123
- 高速缓存混乱统计信息
 - 10.0.0 版中的新功能, 123
- 高速缓存搜寻统计信息
 - 10.0.0 版中的新功能, 123
- 高速缓存页空闲统计信息
 - 10.0.0 版中的新功能, 123
- 高速缓存页文件统计信息
 - 10.0.0 版中的新功能, 123
- 高速缓存页已分配结构统计信息
 - 10.0.0 版中的新功能, 123
- 高速缓存页脏文件统计信息
 - 10.0.0 版中的新功能, 123
- 高速缓存已访问搜寻统计信息
 - 10.0.0 版中的新功能, 123
- 高速缓存预热
 - 9.0.1 版中的新功能, 240
- 隔离级别
 - UltraLite ReadUncommitted, 11.0.0 版中的增强, 68
- 个人服务器
 - 10.0.0 版中更改的缺省 TCP/IP 地址, 143
- 工具栏
 - 11.0.0 版中的 Interactive SQL 增强, 77
- 工作线程
 - 10.0.0 版中的 MobiLink 行为更改, 180
- 功能统计信息集合
 - 10.0.0 版中的新功能, 212
- 故障转移, MobiLink 服务器群
 - 11.0.0 版中的新功能, 60

管理工具

- 10.0.0 版中的行为更改, 212
- 11.0.1 版中的增强, 7
- 11.0.1 版中的行为更改, 11

归类

- 10.0.0 版中的 UltraLite 增强, 196
- 10.0.0 版中的新功能, 136
- 8.0.0 版不支持, 318
- 9.0.1 版中的新功能, 242
- UltraLite 升级建议, 346

归类定制

- 10.0.1 版中的新功能, 89

归类实用程序 (dbcollat)

- 10.0.0 版不支持, 162

国际资源开发工具包

- 11.0.0 版中的行为更改, 55

过程

- 10.0.0 版中的增强, 124

H

handle_error

- 10.0.0 版中的行为更改, 179

handle_odbc_error

- 10.0.0 版中的行为更改, 179

HasCollationTailoring 属性

- 10.0.1 版中的新功能, 89

HasEndianSwapFix 属性

- 11.0.0 版中的新功能, 34

HASH 函数

- 10.0.0 版中的增强, 127
- 9.0.1 版中的新功能, 239

HEADER 子句

- 10.0.0 版中的新功能, 135
- 10.0.1 版中的增强, 90

HeapsCarver 属性

- 10.0.0 版中的新功能, 120, 122

HeapsLocked 属性

- 10.0.0 版中的新功能, 120, 122

HeapsQuery 属性

- 10.0.0 版中的新功能, 120, 122

HeapsRelocatable 属性

- 10.0.0 版中的新功能, 122

HistogramHashFix 属性

- 10.0.0 版中删除的数据库属性, 167

Histograms 属性

- 10.0.0 版中删除的数据库属性, 167

host 协议选项

- 10.0.0 版中的行为更改, 143

HP-UX

- 10.0.1 版中的增强, 94

HTML_DECODE 函数

- 10.0.1 版中的增强, 89
- 10.0.1 版中的行为更改, 89

HTTP

- 11.0.0 版中的行为更改, 52

http_session_timeout 属性

- 10.0.0 版中的新功能, 120, 123

http_session_timeout 选项

- 10.0.0 版中的新功能, 117

HttpAddresses 属性

- 11.0.0 版中的新功能, 33

HttpNumActiveReq 属性

- 11.0.0 版中的新功能, 33

HttpNumConnections 属性

- 11.0.0 版中的新功能, 33

HttpNumSessions 属性

- 11.0.0 版中的新功能, 33

HTTPS

- 10.0.1 版中的增强, 90
- 11.0.0 版中的增强, 44

https_fips

- 10.0.0 版中重命名的 MobiLink [mlsrv10] 选项, 181

HttpsAddresses 属性

- 11.0.0 版中的新功能, 33

HttpServiceName 属性

- 10.0.0 版中的新功能, 120

HttpsNumActiveReq 属性

- 11.0.0 版中的新功能, 33

HttpsNumConnections 属性

- 11.0.0 版中的新功能, 33

HTTP 客户端分块模式传输编码

- 11.0.1 版中的行为更改, 11

回调函数

- 9.0.2 版中的新功能, 219

合并模块

- 10.0.0 版中的行为更改, 132

后台同步

- 11.0.0 版中的 UltraLite 增强, 69

忽略脚本

- 11.0.1 版中的 MobiLink (mlsrv11) 新功能, 12

环境变量

- 10.0.0 版中的增强, 137
- 10.0.0 版中的行为更改, 142

- 命令 shell, xi
- 命令提示符, xi
- 恢复
 - 10.0.0 版中的增强, 114
- 恢复失败的下载
 - 9.0.1 版中的 dbmlsync 新功能, 245
- 恢复数据库向导
 - 9.0.1 版中的增强, 243
- 回退日志
 - 11.0.0 版中的行为更改, 54
- 混合 C++ 接口
 - 9.0.1 版中的 UltraLite 的新功能, 247
- 获取帮助
 - 技术支持, xii
- I**
- iAnywhere.UltraLite 命名空间
 - 10.0.0 版中删除的 UltraLite, 203
- iAnywhere JDBC 驱动程序
 - 10.0.0 版中的增强, 131
 - 10.0.1 版中的增强, 94
- iAnywhere Solutions Oracle 驱动程序
 - 10.0.1 版中的增强, 92
 - 11.0.0 版中的增强, 60
- iAnywhere 开发人员社区
 - 新闻组, xiii
- ias_CurrentDayOfMonth
 - 10.0.0 版中已删除的 QAnywhere 传输规则变量, 191
- ias_CurrentDayOfWeek
 - 10.0.0 版中已删除的 QAnywhere 传输规则变量, 191
- ias_CurrentMonth
 - 10.0.0 版中已删除的 QAnywhere 传输规则变量, 191
- ias_CurrentYear
 - 10.0.0 版中已删除的 QAnywhere 传输规则变量, 191
- ias_MaxDeliveryAttempts
 - 10.0.0 版中新增的 QAnywhere 属性, 188
- ias_MaxUploadSize
 - 10.0.0 版中新增的 QAnywhere 属性, 188
- ias_Status
 - 10.0.0 版中新增的 QAnywhere 属性, 188
- ias_StatusTime
 - 10.0.0 版中新增的 QAnywhere 属性, 188
- IBM DB2
 - 10.0.0 版中的 ODBC 驱动程序行为更改, 214
 - 9.0.1 版中安装脚本名称更改, 244
- IBM DB2 8.2 CLI 驱动程序
 - 在 10.0.0 版中受支持, 214
- ICU
 - 10.0.0 版中的新功能, 111
- IdleTimeout 属性
 - 10.0.0 版中的新功能, 126
- IF EXISTS 子句
 - 11.0.1 版中的新功能, 4
- IF NOT EXISTS 子句
 - 11.0.1 版中的新功能, 5
- IF 表达式
 - 11.0.0 版中的增强, 39
- IF 语句
 - 11.0.0 版中的 UltraLite 增强, 70
 - 11.0.0 版中的增强, 39
- ignored_deletes
 - 10.0.0 版中的新行为, 182
- ignored_inserts
 - 10.0.0 版中的新行为, 182
- ignored_updates
 - 10.0.0 版中的新行为, 182
- ignore 协议选项
 - 10.0.0 版中的 MobiLink 行为更改, 171
- INDEX ONLY 子句
 - 11.0.0 版中的新功能, 39
- IndexStatistics 属性
 - 10.0.0 版中删除的数据库属性, 167
- InfoMaker
 - 10.0.1 版中的 Vista 支持, 105
- InList 算法 (IN)
 - 11.0.0 版中的增强, 45
- INPUT INTO 语句
 - 11.0.0 版中的行为更改, 78
- INPUT 语句
 - 11.0.0 版中的增强, 78
 - 11.0.0 版中的行为更改, 79
 - 9.0.1 版中的增强, 242
- INSERT 语句
 - 10.0.0 版中的增强, 129
 - 10.0.0 版中的行为更改, 160
 - 10.0.1 版中的增强, 88, 91
 - 11.0.0 版中的增强, 40, 45
 - 11.0.0 版中的行为更改, 54
 - 11.0.1 版中的增强, 4, 5
- install-dir

文档用法, x

InstallShield
 10.0.0 版中的行为更改, 132

InstallShield 项目
 9.0.1 版中的新功能, 241

INSTEAD OF 触发器
 10.0.1 版中的新功能, 91

Intel x86 上的处理器
 SQL Anywhere 10.0.0 中支持的版本, 138

Interactive SQL
 10.0.0 版中的 UltraLite 新功能, 196
 10.0.0 版中的增强, 207, 208
 10.0.0 版中的行为更改, 208
 11.0.0 版中的增强, 77
 11.0.0 版中的新功能, 75
 9.0.1 版中的增强, 242

Interactive SQL 实用程序 [dbisql]
 10.0.0 版中的增强, 208

Interactive SQL 新功能
 10.0.0 版, 206
 11.0.0 版, 73
 11.0.1 版, 19

Interactive SQL 行为更改
 10.0.0 版, 208
 11.0.0 版, 76
 11.0.1 版, 20

INTERSECT 语句
 10.0.1 版中的增强, 88
 11.0.0 版中的增强, 40
 11.0.0 版中的行为更改, 54

INTO LOCAL TEMPORARY TABLE 子句
 11.0.1 版中的新功能, 4

invalid_extensions 选项
 10.0.0 版中的新功能, 192

IOParallelism 属性
 10.0.0 版中的新功能, 123

IPv6
 10.0.1 版中的增强, 91

IPv6 支持
 10.0.0 版中的 UltraLite 新功能, 199
 10.0.0 版中的新功能, 113

IPX 协议
 8.0.0 版不支持, 318

IRDK
 11.0.0 版中的行为更改, 55

IsDebugger
 连接属性, 11.0.1 版中的新功能, 2

IsEccAvailable 属性
 10.0.0 版中的新功能, 122

IsJavaAvailable 属性
 10.0.0 版不支持, 141

isolation_level 选项
 10.0.0 版中的增强, 110
 10.0.1 版中的增强, 88

isql_allow_client_file_read
 11.0.0 版中已重命名, 79

isql_allow_client_file_write
 11.0.0 版中已重命名, 79

isql_allow_read_client_file 选项
 11.0.0 版中的新名称, 79

isql_allow_write_client_file 选项
 11.0.0 版中的新名称, 79

isql_maximum_displayed_rows 选项
 10.0.0 版中的新功能, 207

isql_plan 选项
 10.0.0 版中的行为更改, 208
 11.0.0 版中已删除, 79

isql_show_multiple_result_sets 选项
 10.0.0 版中的新功能, 207

IsRsaAvailable 属性
 10.0.0 版中的新功能, 122

ISYSCAPABILITYNAME
 11.0.0 版中的行为更改, 48

ISYSDBFILE
 11.0.0 版中的行为更改, 48

ISYSDBSPACE
 11.0.0 版中的行为更改, 48

ISYSDBSPACEPERM
 11.0.0 版中的行为更改, 48

ISYSEVENTTYPE
 11.0.0 版中的行为更改, 48

ISYSFILE
 11.0.0 版中的行为更改, 48

ISYSFKEY
 系统表, 10.0.0 版中的新功能, 136

ISYSIDX
 11.0.0 版中的行为更改, 48

ISYSIDXCOL
 系统表, 10.0.0 版中的新功能, 136

ISYSLOGINMAP
 11.0.0 版中的新功能, 48

ISYSLOGINPOLICY
 11.0.0 版中的新功能, 48

ISYSLOGINPOLICYOPTION

- 11.0.0 版中的新功能, 48
- ISYSOBJECT
 - 11.0.0 版中的行为更改, 48
- ISYSPHYSIDX
 - 系统表, 10.0.0 版中的新功能, 136
- ISYSTAB
 - 11.0.0 版中的行为更改, 48
- ISYSTEXTCONFIG
 - 11.0.0 版中的新功能, 48
- ISYSTEXTIDX
 - 11.0.0 版中的新功能, 48
- ISYSTEXTIDXTAB
 - 11.0.0 版中的新功能, 48
- ISYSVIEW
 - 11.0.0 版中的行为更改, 48
- Itanium
 - 11.0.0 版中的行为更改, 54
- J**
- Java
 - 10.0.0 版中的行为更改, 141
- java_heap_size 选项
 - 10.0.0 版不支持, 141
- java_input_output 属性
 - 10.0.0 版不支持, 141
- java_input_output 选项
 - 10.0.0 版不支持, 141
- java_location 属性
 - 10.0.0 版中的新功能, 120
- java_location 选项
 - 10.0.0 版中的新功能, 118
- java_main_userid 属性
 - 10.0.0 版中的新功能, 120
- java_main_userid 选项
 - 10.0.0 版中的新功能, 118
- java_namespace_size 选项
 - 10.0.0 版不支持, 141
- java_page_buffer_size 选项
 - 10.0.0 版不支持, 141
- java_vm_options 选项
 - 10.0.0 版中的新功能, 118
- Java API
 - 10.0.0 版中的 QAnywhere 新功能, 187
- Java DownloadTableData 接口
 - 11.0.0 版中的增强, 60
- JavaGlobFix 属性
 - 10.0.0 版不支持, 141
- JavaHeapSize 属性
 - 10.0.0 版不支持, 141
- Java ME
 - 11.0.0 版中的 UltraLite 增强, 71
- JavaNSSize 属性
 - 10.0.0 版不支持, 141
- Java SE
 - 11.0.0 版中的 UltraLite 增强, 71
- JavaVM 属性
 - 10.0.0 版中的新功能, 123
- Java VM 属性
 - 10.0.0 版中的新功能, 141
- jConnect
 - 10.0.0 版不支持使用 jConnect 连接到 Interactive SQL, 209
 - 10.0.0 版不支持使用 jConnect 连接到 SQL Anywhere 控制台 (dbconsole), 209
 - 10.0.0 版不支持使用 jConnect 连接到 Sybase Central, 209
 - 10.0.0 版中删除的 4.5 版本支持, 161
 - 10.0.0 版中的增强, 137
 - 10.0.1 版中的行为更改, 94
 - 11.0.0 版中的增强, 29
- JDBC 2.0
 - 不再支持, 131
- JDBC 3.0
 - 10.0.0 版中的新功能, 131
- JDKVersion 属性
 - 10.0.0 版不支持, 141
- 校验
 - 10.0.0 版中的行为更改, 115
 - 11.0.0 版中的行为更改, 30
- 校验和
 - 10.0.0 版中的增强, 113
 - 11.0.0 版中的增强, 44
 - 11.0.0 版中的行为更改, 52
 - 9.0.1 版中的新功能, 240
- 校验实用程序 [dbvalid]
 - 9.0.1 版中的增强, 240
- 校验实用程序 [dbvalid]
 - 10.0.0 版中的增强, 117
 - 11.0.0 版中的增强, 30
 - 9.0.1 版中的行为更改, 250
- 校验数据库向导
 - 11.0.0 版中的新功能, 74
 - 9.0.1 版中的增强, 243
- 集成登录

- 9.0.1 版中的增强, 241
- 技术支持
 - 新闻组, xiii
- 计划查看器
 - 11.0.0 版中不建议使用的功能, 77
 - 11.0.0 版中的增强, 77
- 计划高速缓存
 - 10.0.1 版中的增强, 91
 - 11.0.0 版中的增强, 45
- 计算列
 - 11.0.0 版中的行为更改, 51
- 记录
 - 10.0.0 版中的增强, 130
- 记录数据库
 - 11.0.0 版中的新功能, 74
- 记忆页查询处理统计信息
 - 10.0.0 版中的新功能, 123
- 记忆页雕刻统计信息
 - 10.0.0 版中的新功能, 123
- 记忆页已固定游标统计信息
 - 10.0.0 版中的新功能, 123
- 加密
 - 10.0.1 版中的增强, 87
 - 10.0.1 版中的行为更改, 92
 - 11.0.0 版中的增强, 28, 29, 44
- 加密密钥
 - 10.0.0 版中的增强, 128
- 加引号的标识符
 - 11.0.0 版中的行为更改, 53
- 监控模式升级
 - 9.0.1 版中的 UltraLite 增强, 247
- 监控器
 - 11.0.1 版中的新功能, 21
- 监视项目列表
 - 11.0.0 版中的新功能, 36
- 兼容性
 - UltraLite 软件升级, 345
 - 客户端/服务器, 324
 - 数据库和数据库服务器, 324
 - 问题, 324
- 检查点
 - 10.0.0 版中的增强, 111
 - 带有自动增量列的表, 328
- 检查点操作
 - 10.0.1 版中的 UltraLite 增强, 101
- 键连接
 - 10.0.0 版中的行为更改, 160
- 键盘快捷方式
 - 11.0.0 版中的新功能, 73
- 健康和统计信息
 - 11.0.0 版中的新功能, 74
- 将连接字符串复制到剪贴板
 - 11.0.0 版中的新功能, 73
- 交换算法
 - 10.0.0 版中的新功能, 109
- 脚本
 - 升级, 343
- 脚本版本
 - 9.0.1 版中可配置, 244
- 脚本式上载
 - 10.0.0 版中的新功能, 175
- 脚本执行 [dbrunsql] 实用程序
 - 10.0.0 版中的新功能, 133
- 接口
 - 10.0.0 版中的 UltraLite 新功能, 196
- 结果
 - 9.0.1 版中的 Interactive SQL 增强, 242
 - 9.0.1 版中的 Sybase Central 增强, 242
 - 设置为只读, 11.0.0 版中的新功能, 75
- 结果窗格
 - 11.0.0 版中的增强, 77
- 结果集
 - 设置为只读, 11.0.0 版中的新功能, 75
- 结束脚本
 - 10.0.0 版中的 MobiLink 行为更改, 179
- 解压缩数据库向导
 - 10.0.0 版不支持, 163
- 仅下载发布
 - 10.0.0 版中的新功能, 175
- 进度偏移
 - 10.0.0 版中的行为更改, 184
- 禁用表编辑
 - 11.0.0 版中的新功能, 75
- 禁用结果集编辑
 - 11.0.0 版中的新功能, 75
- 静默安装
 - 11.0.0 版中的行为更改, 57
- 镜像
 - 11.0.0 版中的增强, 26
 - 升级, 335
 - 应用 EBF, 335
- 镜像数据库
 - 11.0.0 版中的增强, 27
- 聚簇散列 group by

9.0.1 版中的新功能, 240

K

可重新启动的下载

9.0.1 版中的 UltraLite 的新功能, 247

KeepaliveTimeout 协议选项

10.0.0 版中的新功能, 134

Kerberos

10.0.0 版中的新功能, 114

KTO 协议选项

10.0.0 版中的新功能, 134

Kyocera

10.0.0 版中 MobiLink Palm 监听器取消了支持, 186

开发人员社区

新闻组, xiii

开始脚本

10.0.0 版中的 MobiLink 行为更改, 179

可访问性

8.0.0 版中的新功能, 301

可配置提交刷新

10.0.1 版中的 UltraLite 增强, 101

客户端网络层

10.0.0 版中的 UltraLite 新功能, 199

客户端文件, 读取

11.0.0 版中的新功能, 26

客户端文件, 写入

11.0.0 版中的新功能, 26

客户端消息存储库

11.0.0 版中的增强, 64

在 10.0.0 版中 QAnywhere 不再使用事务日志, 190

客户端消息存储库 ID

10.0.0 版中的行为更改, 190

客户端语句高速缓存

10.0.1 版中的新功能, 87

10.0.1 版中的行为更改, 93

空格填充

10.0.0 版中的行为更改, 138

控制台实用程序 [dbconsole]

10.0.0 版中的增强, 208

9.0.1 版中的增强, 241

口令

10.0.0 版中的 UltraLite 行为更改, 196

10.0.0 版中的行为更改, 138

11.0.0 版中的增强, 28

9.0.1 版中的行为更改, 251

口令散列

10.0.0 版中的 UltraLite 行为更改, 196

10.0.0 版中的行为更改, 138

快捷方式

11.0.0 版中的新功能, 75

快速启动程序

11.0.0 版中的增强, 73

快速入门

将数据库升级到版本 11, 326

快速视图

11.0.0 版中的新功能, 27

快照隔离

10.0.0 版中的 MobiLink 支持, 172

10.0.0 版中的新功能, 110

宽字符

10.0.0 版中的 UltraLite 增强, 201

L

LargeProcedureIDs 属性

10.0.0 版中删除的数据库属性, 167

LAST_BACKUP 操作

10.0.0 版中的新功能, 114

LAST_VALUE 函数

10.0.1 版中的新功能, 91

11.0.1 版中的新功能, 3

LastConnectionProperty 属性

10.0.0 版中的新功能, 122

LastDatabaseProperty 属性

10.0.0 版中的新功能, 122

LastOption 属性

10.0.0 版中的新功能, 122

LastPlanText 属性

10.0.0 版中的新功能, 120

LastServerProperty 属性

10.0.0 版中的新功能, 122

LastStatement 属性

10.0.1 版中的行为更改, 93

LazyClose 连接参数

11.0.0 版中的行为更改, 51

LCLOSE 连接参数

11.0.0 版中的行为更改, 51

LDAP 验证

10.0.0 版中的增强, 113

9.0.1 版中的 MobiLink 增强, 244

LENGTH 函数

10.0.0 版中的 QAnywhere 新功能, 189

libdbtool9.so

- 9.0.2 版中不建议使用, 230
- LicensesInUse 属性
 - 10.0.0 版中已重命名, 140
- Linux
 - 10.0.0 版中的新功能, 133
 - 11.0.0 版中的增强, 44
 - 11.0.0 版中的新功能, 44
 - 11.0.0 版中的行为更改, 54
 - 9.0.1 版中的新功能, 243
- little endian UTF-16 编码
 - 11.0.0 版中的新功能, 44
- liveness_timeout 选项
 - 10.0.0 版中已删除, 180
 - 10.0.0 版中被替换的 MobiLink 客户端协议选项, 174
- LOAD TABLE 语句
 - 10.0.0 版中的增强, 129
 - 10.0.0 版中的行为更改, 161
 - 11.0.0 版中的增强, 26, 37, 39, 62, 66
 - 9.0.1 版中变量的增强支持, 239
 - 9.0.1 版中的增强, 239
- LockCount 属性
 - 10.0.0 版中的新功能, 120, 123
- LockedCursorPages 属性
 - 10.0.0 版中的新功能, 120
- LockIndexID 属性
 - 11.0.0 版中的新功能, 32
- LockRowID 属性
 - 11.0.0 版中的新功能, 32
- LockTableOID 属性
 - 10.0.0 版中的新功能, 120
- login_mode 选项
 - 10.0.0 版中的行为更改, 146
- login_procedure 选项
 - 11.0.0 版中的增强, 31
- LogMaxSize 协议选项
 - 10.0.0 版中的增强, 113
- LONG VARBIT 数据类型
 - 10.0.0 版中的新功能, 130
- LOTUS 格式, INPUT 语句
 - 11.0.0 版中删除了支持, 79
- LOTUS 格式, OUTPUT 语句
 - 11.0.0 版中删除了支持, 79
- LTM
 - 10.0.0 版中的增强, 116
- 联邦康复条例
 - 第 508 节, 301
- 联机手册
 - PDF, viii
- 连接
 - 10.0.1 版中的 UltraLite 增强, 101
 - 11.0.0 版中的行为更改, 52
 - 选择控制连接对象的参数, 346
- 连接 ID
 - 9.0.1 版中的增强, 243
- 连接参数
 - 10.0.0 版中的增强, 113
 - 10.0.0 版中的行为更改, 141
 - UltraLite API 控件, 346
- 连接窗口
 - 11.0.0 版中的增强, 73
 - 11.0.0 版中的新功能, 73
 - 9.0.1 版中的行为更改, 251
- 连接代码
 - UltraLite 升级, 349
- 连接配置文件
 - 10.0.0 版中的增强, 206
- 连接属性
 - 10.0.0 版中的行为更改, 139
 - 11.0.0 版中的新功能, 32
- 连接助手
 - 11.0.0 版中的新功能, 73
- 连接字符串
 - 10.0.0 版中的增强, 113
 - 10.0.0 版中的行为更改, 140
- 连字符
 - 11.0.1 版中的全文搜索行为更改, 8
- 列属性
 - 在重建的数据库中保留下一个可用值, 328
- 列压缩
 - 10.0.0 版中的新功能, 112
- 临时表
 - 10.0.0 版中的增强, 130
 - 10.0.1 版中的行为更改, 94
 - 11.0.0 版中的行为更改, 55
- 临时存储过程
 - 10.0.0 版中的新功能, 130
- 临时文件
 - 10.0.0 版中的增强, 118
 - 11.0.0 版中的 Unix 增强, 44
- 邻近搜索
 - 11.0.1 版中的行为更改, 8
- 另存为 ODBC 数据源
 - 11.0.0 版中的新功能, 73

逻辑索引

10.0.0 版中的新功能, 135

M

MachineName 属性

10.0.1 版中的增强, 94

Mac OS X

10.0.1 版中的增强, 87

11.0.0 版中的增强, 44

11.0.1 版中的增强, 7

11.0.1 版中的行为更改, 10, 11

重建 SQL Anywhere 11 数据库, 325

MAPI 消息类型

11.0.0 版不支持, 66

不建议在 10.0.1 版中使用 SQL Remote 支持,
100

MapPages 属性

11.0.0 版不支持, 56

MapPhysicalMemoryEng 属性

10.0.0 版中的新功能, 122

materialized_view_optimization 属性

10.0.0 版中的新功能, 120

materialized_view_optimization 选项

10.0.0 版中的新功能, 117

max_client_statements_cached 属性

10.0.1 版中的新功能, 87

max_client_statements_cached 选项

10.0.1 版中的新功能, 87

max_hash_size 选项

10.0.0 版不支持, 162

max_priority 选项

11.0.0 版中的新功能, 31, 32

max_query_tasks 属性

10.0.0 版中的新功能, 120

max_query_tasks 选项

10.0.1 版中的增强, 88

max_temp_space 选项

10.0.0 版中的新功能, 117

10.0.0 版中的新连接属性, 120

max_work_table_hash_size 选项

10.0.0 版不支持, 162

MaxConnections 属性

10.0.0 版中的新功能, 122

MaxEventType

11.0.0 版中的新功能, 33

MaxMessage 属性

11.0.0 版中不建议使用, 56

MaxRemoteCapability 属性

11.0.0 版中的新功能, 33

MaxRequestSize 协议选项

10.0.0 版中的增强, 113

MDSR 加密

9.0.1 版中停用, 250

MERGE 语句

11.0.0 版中的新功能, 25

MessageCategoryLimit 属性

11.0.0 版中的新功能, 33

MessageText 属性

11.0.0 版中不建议使用, 56

MessageTime 属性

11.0.0 版中不建议使用, 56

MessageWindowSize 属性

11.0.0 版中不建议使用, 56

Message 属性

11.0.0 版中不建议使用, 56

MESSAGE 语句

10.0.0 版中的增强, 130

9.0.1 版中的增强, 240

Microsoft Access

11.0.0 版中的远程数据访问增强, 45

Microsoft Excel

11.0.0 版中删除了 INPUT 语句支持, 79

11.0.0 版中删除了 OUTPUT 语句支持, 79

Microsoft SQL Server 2008 支持

11.0.1 版中的增强, 12

Microsoft 分布式事务处理协调器

11.0.0 版中的新功能, 60

Microsoft 卷影复制服务 (VSS)

11.0.0 版中的新功能, 28

min_table_size_for_histogram 选项

9.0.1 版中的行为更改, 251

9.0.2 版中不建议使用, 230

MIPS

10.0.0 版中删除的 SQL Anywhere 支持, 167

MirrorFailover 系统事件

10.0.0 版中的新功能, 109

MirrorMode 属性

11.0.0 版中的新功能, 34

MirrorServerDisconnect 系统事件

10.0.0 版中的新功能, 109

MirrorState 属性

10.0.0 版中的新功能, 123

ml_add_column 系统过程

10.0.0 版中的新功能, 170

ml_column
 10.0.0 版中的新功能, 170

ml_database
 10.0.0 版中的新功能, 170

ml_delete_sync_state_before 系统过程
 10.0.0 版中的 MobiLink 新功能, 169

ml_delete_sync_state system 系统过程
 10.0.0 版中的 MobiLink 新功能, 169

ML_GET_SERVER_NOTIFICATION 函数
 11.0.1 版中的 UltraLite 增强, 15

ml_global 脚本版本
 10.0.0 版中的 MobiLink 新功能, 172

ml_ignore
 11.0.1 版中的 MobiLink (mlsrv11) 新功能, 12

ml_listening
 10.0.0 版中的新模式, 170

ml_qa_clients
 10.0.0 版中的新功能, 170

ml_qa_delivery
 10.0.0 版中的新模式, 170

ml_qa_delivery_client
 10.0.0 版中的新模式, 170

ml_qa_global_props
 10.0.0 版中更改, 170

ml_qa_global_props_client
 10.0.0 版中更改, 170

ml_qa_repository
 10.0.0 版中的新模式, 170

ml_qa_repository_client
 10.0.0 版中更改, 170

ml_qa_repository_props
 10.0.0 版中更改, 170

ml_qa_repository_props_client
 10.0.0 版中更改, 170

ml_qa_repository_staging
 10.0.0 版中更改, 170

ml_remote_id
 11.0.1 版中的 UltraLite 增强, 15

ml_remote_id 选项
 10.0.0 版中的新功能, 174

ml_reset_sync_state 系统过程
 10.0.0 版中的 MobiLink 新功能, 169

ml_script
 10.0.0 版中的新模式, 170

ml_sis_sync_state
 10.0.0 版中的新功能, 170

ml_subscription
 10.0.0 版中的新模式, 170

ml_user
 10.0.0 版中的新模式, 170

mlsrv10.lic
 10.0.1 版中的新功能, 105

mlxtract
 10.0.0 版中已删除, 184

MobiLink
 升级, 337

MobiLink 插件
 10.0.0 版中的增强, 207

MobiLink 服务器
 升级, 342

MobiLink 服务器群
 11.0.0 版中的新功能, 60

MobiLink 监控器
 10.0.0 版中的增强, 172
 11.0.0 版中的行为更改, 80

MobiLink 客户端
 在版本 10.0.0 中删除了对 9.0 之前版本的支持, 185

MobiLink 日志文件
 11.0.0 版中的 UltraLite 增强, 69

MobiLink 日志文件查看器
 10.0.1 版中的新功能, 96

MobiLink 文件传输
 10.0.0 版中的 MobiLink 新功能, 174

MobiLink 系统表
 9.0.1 版中的新表, 244
 升级, 337

MobiLink 系统对象
 11.0.0 版中的增强, 59

MobiLink 系统过程
 11.0.0 版中的增强, 59

MobiLink 系统数据库
 11.0.0 版中的新功能, 59

MobiLink 新功能
 10.0.0 版, 168
 10.0.1 版, 96
 11.0.0 版, 59
 11.0.1 版, 12
 9.0.2 版, 221

MobiLink 行为更改
 10.0.0 版, 178
 10.0.1 版, 97
 11.0.0 版, 63
 11.0.1 版, 13

- 9.0.2 版, 231
 - MobiLink 用户名
 - 10.0.0 版中的行为更改, 184
 - MobiLink 中继服务器
 - 11.0.0 版中的新功能, 60
 - msaccessodbc 服务器类
 - 11.0.0 版中的新功能, 45
 - MultiPageAllocs 属性
 - 10.0.0 版中的新功能, 120
 - MultiProgrammingLevel 属性
 - 10.0.0 版中的新功能, 122
 - mysqlodbc 服务器类
 - 11.0.0 版中的新功能, 45
 - MySQL 统一数据库
 - 11.0.0 版中的新功能, 59
 - 密钥对生成器实用程序 [createkey]
 - 11.0.0 版中的新功能, 62
 - 命令 shell
 - 大括号, xi
 - 引号, xi
 - 括号, xi
 - 环境变量, xi
 - 约定, xi
 - 命令提示符
 - 大括号, xi
 - 引号, xi
 - 括号, xi
 - 环境变量, xi
 - 约定, xi
 - 命令行实用程序
 - UltraLite 升级, 345
 - UltraLite 多个版本, 345
 - 升级, 326
 - 多个版本, 326
 - 命名的脚本参数
 - 10.0.0 版中的 MobiLink 新功能, 171
 - 命名管道
 - 10.0.0 版不支持, 162
 - 模式
 - 10.0.0 版中的 UltraLite 新功能, 194
 - UltraLite 升级, 349
 - UltraLite 升级影响, 346
 - 模式管理器
 - 10.0.0 版中删除的 UltraLite, 204
 - 模式升级
 - 9.0.1 版中的 UltraLite 增强, 247
 - 目标别名
 - 10.0.0 版中的 QAnywhere 新功能, 189
 - 目标用户的日志详细程度
 - 11.0.1 版中的增强, 12
 - 目录
 - 10.0.0 版中的行为更改, 147
 - 11.0.0 版中的行为更改, 48
 - 目录访问服务器
 - 10.0.0 版中的新功能, 136
- ## N
- NamedConstraints 属性
 - 10.0.0 版中删除的数据库属性, 167
 - Native UltraLite for Java API 支持
 - 10.0.0 版中删除的 UltraLite, 203
 - NcharCharSet 属性
 - 10.0.0 版中的新功能, 123
 - NcharCollation 属性
 - 10.0.0 版中的新功能, 123
 - NCHAR 数据类型
 - 10.0.0 版中的新功能, 110
 - NetBios 协议
 - 8.0.0 版不支持, 318
 - network_connect_timeout 选项
 - 10.0.0 版中被替换的 MobiLink 客户端协议选项, 174
 - network_name 协议选项
 - 10.0.1 版中的增强, 96
 - Networking 选项卡
 - 11.0.0 版中的新功能, 73
 - new_row_cursor
 - 10.0.0 版中已删除, 179
 - 9.0.0 版中不建议使用, 272
 - NewPassword 连接参数
 - 11.0.0 版中的新功能, 28
 - NEWPWD 连接参数
 - 11.0.0 版中的新功能, 28
 - NEXT_SOAP_HEADER 函数
 - 10.0.0 版中的新功能, 135
 - NextScheduleTime 属性
 - 10.0.0 版中的新功能, 126
 - non_keywords 选项
 - 11.0.0 版中的行为更改, 52
 - NoSyncOnStarup dbmlsync 扩展选项
 - 10.0.0 版中的新功能, 175
 - NULL
 - 9.0.1 版中的行为更改, 251
 - NumLogicalProcessorsUsed 属性

- 10.0.0 版中的新功能, 122
- NumLogicalProcessors 属性
 - 10.0.0 版中的新功能, 122
- NumPhysicalProcessorsUsed 属性
 - 10.0.0 版中的新功能, 122
- NumPhysicalProcessors 属性
 - 10.0.0 版中的新功能, 122
- NumProcessorsAvail 属性
 - 10.0.0 版不支持, 161
- NumProcessorsMax 属性
 - 10.0.0 版不支持, 161
- NVFS
 - 10.0.0 版中的 UltraLite 增强, 197
- 内部卸载
 - 保留自定义归类, 330
- 内存模式
 - 11.0.0 版中的新功能, 31
- 挪威语
 - 10.0.0 版中的新功能, 136

O

- ODBC
 - 10.0.0 版中的行为更改, 140
 - 9.0.1 版中的 UltraLite 的新功能, 246
- odbc_distinguish_char_and_varchar 选项
 - 9.0.1 版中的新功能, 244
- ODBC 服务器类
 - 11.0.0 版中的增强, 45
- ODBC 配置窗口
 - 9.0.1 版中的行为更改, 251
- ODBC 驱动程序
 - 10.0.0 版中的新驱动程序, 214
 - 10.0.1 版中的新 Oracle 驱动程序, 103
- ODBC 驱动程序管理器
 - 10.0.0 版中新的 Unix 功能, 133
 - 11.0.0 版中的增强, 41
- oem_string 属性
 - 10.0.0 版中的新功能, 120
- oem_string 选项
 - 10.0.0 版中的新功能, 118
- OEM.ini
 - 10.0.0 版中的新功能, 212
 - 11.0.0 版中的增强, 75
- old_row_cursor
 - 10.0.0 版中已删除, 179
 - 9.0.0 版中不建议使用, 272
- OLE DB
 - 10.0.0 版中的行为更改, 140
- Open Client
 - 11.0.0 版中的增强, 29
- OpenString 算法
 - 11.0.0 版中的新功能, 27
- OPENSTRING 子句
 - 11.0.0 版中的新功能, 27
- OpenTableEx 函数 [UL C++]
 - 11.0.0 版中的 UltraLite 增强, 71
- openxml 系统过程
 - 10.0.0 版中的行为更改, 144
 - 9.0.1 版中的增强, 243
- OPEN 语句
 - 10.0.0 版中的增强, 110
- optimistic_wait_for_commit 选项
 - 11.0.0 版不支持, 56
- optimization_goal 选项
 - 10.0.1 版中的增强, 88
- optimization_level 选项
 - 10.0.1 版中的增强, 88
- optimization_workload 选项
 - 10.0.1 版中的增强, 88
 - 9.0.1 版中的新功能, 240
- OptionWatchAction 属性
 - 11.0.0 版中的新功能, 33
- OptionWatchList 属性
 - 11.0.0 版中的新功能, 33
- OPTION 子句
 - 10.0.0 版中的增强, 128
 - 10.0.1 版中的增强, 88
 - 11.0.0 版中的增强, 40
 - 11.0.0 版中的行为更改, 54
- Oracle
 - 11.0.0 版中的增强, 60
- Oracle varray
 - 11.0.1 版中的新功能, 12
- Oracle 驱动程序
 - 10.0.1 版中的增强, 92
 - 11.0.0 版中的增强, 60
- ORDER BY 和 UPDATE 语句
 - 9.0.1 版中的行为更改, 251
- ORDER BY 子句
 - 10.0.0 版中的 UltraLite 增强, 198
- OR REPLACE 子句
 - 11.0.1 版中的新功能, 4
- OSUser 属性
 - 11.0.0 版中的新功能, 32

- OUTER APPLY 子句
 - 11.0.0 版中的新功能, 39
- OUTPUT 语句
 - 11.0.0 版中的 dbisql 行为更改, 55
 - 11.0.0 版中的增强, 78
 - 11.0.0 版中的行为更改, 79
 - 11.0.1 版中的行为更改, 10
 - 9.0.1 版中的增强, 242
- P**
- Palm HotSync 管道安装实用程序
 - 10.0.0 版中的 UltraLite 增强, 197
- Palm OS
 - 10.0.0 版中的 UltraLite 增强, 196
 - 9.0.1 中受支持的最早版本, 252
 - Certicom Security Builder GSE 版本, 10.0.1 版中的增强, 103
- PartnerState 属性
 - 10.0.0 版中的新功能, 123
- PBUF 连接参数
 - 11.0.0 版中的增强, 28
- PDF
 - 文档, viii
- percent_as_comment 选项
 - 11.0.0 版中的行为更改, 56
- PERCENT_RANK 函数
 - 9.0.1 版中的新功能, 238
- Perl
 - 9.0.1 版中的新 DBD::ASAny 驱动程序, 241
- Perl DBD::ASAny
 - 10.0.0 版中更改的名称, 132
- PHP 模块
 - 10.0.0 版中的增强, 132
 - 10.0.0 版中的行为更改, 142
 - 函数名更改, 11.0.0 版中的行为更改, 50
- ping 实用程序 [dbping]
 - 10.0.0 版中的增强, 116
 - 10.0.0 版中的行为更改, 142
- PORT 协议选项
 - 10.0.0 版中的行为更改, 144
- post_login_procedure 属性
 - 10.0.0 版中的新功能, 120
- post_login_procedure 选项
 - 11.0.0 版中的行为更改, 52
- PowerDesigner
 - 10.0.1 版中的 Vista 支持, 105
- precision 选项
 - 11.0.0 版中的行为更改, 53
- PrefetchBuffer 连接参数
 - 10.0.0 版中的增强, 113
 - 10.0.0 版中的行为更改, 142
 - 11.0.0 版中的增强, 28
- PrefetchRows 连接参数
 - 10.0.1 版中的行为更改, 93
- PrefetchRows 属性 [SA .NET 2.0]
 - 10.0.1 版中的行为更改, 93
- PreparedStatement.addBatch 方法
 - 10.0.0 版中的新功能, 131
- Prepares 属性
 - 11.0.1 版中的新功能, 2
- PreserveSource 属性
 - 10.0.0 版中不建议使用的数据库属性, 166
 - 11.0.0 版不支持, 56
- priority 属性
 - 11.0.0 版中的新功能, 32
- priority 选项
 - 11.0.0 版中的新功能, 31
- procedure_profiling 服务器选项
 - 10.0.0 版中的行为更改, 144
- ProfileFilterConn 属性
 - 10.0.0 版中的新功能, 126
- PROFILE 特权
 - 11.0.0 版中的新功能, 30
- PROPERTY_NAME 函数
 - 11.0.0 版中的增强, 34
- PROPERTY 函数
 - 10.0.0 版中的增强, 126
- Python
 - Python 数据库 API, 11.0.0 版中的新功能, 40
- 排序块算法
 - 10.0.0 版已删除, 167
- 派生表
 - 10.0.0 版中的键连接行为更改, 160
 - 9.0.1 版中的 UltraLite 的新功能, 246
- 配置文件
 - 11.0.0 版中的增强, 30
- 批处理
 - 11.0.0 版中的增强, 45
- 平台
 - 10.0.0 版中删除的 UltraLite, 203
 - 10.0.0 版中的 UltraLite 增强, 195
 - 10.0.1 版中的 UltraLite 增强, 101
 - 11.0.0 版中的 UltraLite 增强, 69

Q

- QAMessageListener2 接口 [QA Java API]
 - 10.0.1 版中的 QAnywhere 新功能, 98
- QAMessageListener2 委派 [QA .NET API]
 - 10.0.1 版中的 QAnywhere 新功能, 98
- QAnywhere
 - 9.0.1 版中的 MobiLink 的新功能, 244
 - 升级数据库和应用程序, 344
- QAnywhere .NET API
 - 11.0.0 版中的增强, 64
- QAnywhere 11 Demo 数据源
 - 11.0.0 版中的行为更改, 83
- QAnywhere C# API
 - 11.0.0 版中的增强, 64
- QAnywhere Java API
 - 11.0.0 版中的增强, 64
- QAnywhere UltraLite 代理
 - 11.0.0 版中的新功能, 64
- QAnywhere Web 服务
 - 10.0.0 版中的新功能, 187
- QAnywhere 插件
 - 10.0.0 版中的新功能, 207
- QAnywhere 代理
 - 10.0.0 版中的行为更改, 188
- QAnywhere 的新功能
 - 9.0.2 版, 226
- QAnywhere 独立客户端
 - 11.0.1 版中的新功能, 14
- QAnywhere 服务器日志文件查看器
 - 10.0.1 版中的新功能, 98
- QAnywhere 新功能
 - 10.0.0 版, 187
 - 10.0.1 版, 98
 - 11.0.0 版, 64
 - 11.0.1 版, 14
- QAnywhere 行为更改
 - 10.0.0 版, 189
 - 10.0.1 版, 98
 - 11.0.0 版, 65
 - 9.0.2 版, 234
- qastop 实用程序
 - 11.0.0 版中的行为更改, 65
- qauagent 实用程序
 - 11.0.0 版中的新功能, 64
- query_mem_timeout 选项
 - 11.0.0 版中的新功能, 31, 32
- query_plan_on_open 选项
 - 11.0.0 版不支持, 56
- QueryBypassedCosted 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- QueryBypassedHeuristic 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- QueryBypassedOptimized 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- QueryDescribedBypass 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- QueryDescribedOptimizer 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- QueryHeapPages 属性
 - 10.0.0 版中的新功能, 120, 122
- QueryMemActiveCurr 属性
 - 11.0.0 版中的新功能, 32, 33
- QueryMemActiveEst 属性
 - 11.0.0 版中的新功能, 33
- QueryMemActiveMax 属性
 - 11.0.0 版中的新功能, 33
- QueryMemExtraAvail 属性
 - 11.0.0 版中的新功能, 32, 33
- QueryMemGrantBaseMI 属性
 - 11.0.0 版中的新功能, 33
- QueryMemGrantBase 属性
 - 11.0.0 版中的新功能, 33
- QueryMemGrantExtra 属性
 - 11.0.0 版中的新功能, 33
- QueryMemGrantFailed 属性
 - 11.0.0 版中的新功能, 32, 33
- QueryMemGrantGranted 属性
 - 11.0.0 版中的新功能, 32, 33
- QueryMemGrantRequested 属性
 - 11.0.0 版中的新功能, 32, 33
- QueryMemGrantWaiting 属性
 - 11.0.0 版中的新功能, 32, 33
- QueryMemPages 属性
 - 11.0.0 版中的新功能, 33
- QueryMemPercentOfCache 属性
 - 11.0.0 版中的新功能, 33
- QueryMemWaited 属性
 - 11.0.0 版中的新功能, 32, 33

- QueryOpened 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- QueryReused 属性
 - 11.0.0 版中的新功能, 34
- 区分重音
 - 10.0.1 版中的行为更改, 95
- quoted_identifier 选项
 - 10.0.0 版中的行为更改, 208
 - 11.0.0 版中的行为更改, 53
- 迁移
 - SQL Remote 到 MobiLink, 192
- 迁移 C++ API 向导
 - UltraLite 使用, 350
- 迁移 C++ 应用程序向导
 - 11.0.0 版不支持, 72
- 嵌入式 SQL
 - 在 UltraLite 9.0.2 中不建议使用, 233
- 嵌入式 SQL 导入库
 - 11.0.0 版中的新功能, 51
- 嵌套块连接算法
 - 10.0.0 版已删除, 167
- 轻量级轮询操作变量
 - 11.0.1 版中的增强, 12
- 轻量级轮询监听器关键字
 - 11.0.1 版中的增强, 12
- 请求级记录 (死请求记录)
- 请求记录
 - 10.0.0 版中的增强, 137
- 请求日志
 - 9.0.1 版中的增强, 241
- 区分大小写
 - 10.0.1 版中的行为更改, 94, 95
 - UltraLite 升级注意事项, 346
- 全局临时表
 - 10.0.0 版中的增强, 130
- 全文搜索
 - 11.0.0 版中的新功能, 25
 - 11.0.1 版中的行为更改, 8
- 缺省数据库服务器
 - 11.0.1 版中的增强, 6
- R**
- RAND 函数
 - 10.0.0 版中的行为更改, 139
- RANK 函数
 - 9.0.1 版中的新功能, 238
- read_authdn 参数
 - 10.0.0 版中的新功能, 113
- READ_CLIENT_FILE 函数
 - 11.0.0 版中的新功能, 34
- read_password 参数
 - 10.0.0 版中的新功能, 113
- readcert 实用程序
 - 11.0.0 版中已删除, 82
 - 不建议在 10.0.1 版中使用, 104
- READCLIENTFILE 特权
 - 11.0.0 版中的新功能, 30, 34
- READFILE 特权
 - 11.0.0 版中的新功能, 30
- ReadHintScatterLimit 属性
 - 11.0.0 版中的新功能, 33
 - 11.0.0 版中已重命名, 54
- ReadHintScatter 属性
 - 11.0.0 版中的新功能, 32
 - 11.0.0 版中已重命名, 54
- ReadHint 属性
 - 11.0.0 版中的新功能, 32
 - 11.0.0 版中已重命名, 54
- READPAST 表提示
 - 10.0.0 版中的新功能, 129
- READ 语句
 - 9.0.1 版中的增强, 242
- REBUILD 子句, ALTER INDEX 语句
 - 10.0.0 版中的增强, 128
- ReceiveBufferSize 协议选项
 - 10.0.0 版中的增强, 113
- ReceivingTracingFrom 属性
 - 10.0.0 版中的新功能, 123
- REFRESH MATERIALIZED VIEW 语句
 - 10.0.0 版中的新功能, 127
 - 11.0.0 版中的增强, 27
 - 11.0.0 版中的行为更改, 50
- REFRESH TEXT INDEX 语句
 - 11.0.0 版中的新功能, 38
- REFRESH TRACING LEVEL 语句
 - 10.0.0 版中的新功能, 128
- REFRESH 子句, ALTER MATERIALIZED VIEW 语句
 - 11.0.0 版中的新功能, 27
- REGBIN 协议选项
 - 11.0.0 版不支持, 56
- REGEXP_SUBSTR 函数
 - 11.0.0 版中的新功能, 25, 34

11.0.1 版中的行为更改, 9

REGEXP 搜索条件

11.0.0 版中的新功能, 25

11.0.1 版中的行为更改, 9

RegisterBindery 协议选项

11.0.0 版不支持, 56

REGR_AVGX 函数

9.0.1 版中的新功能, 238

REGR_AVGY 函数

9.0.1 版中的新功能, 238

REGR_COUNT 函数

9.0.1 版中的新功能, 238

REGR_INTERCEPT 函数

9.0.1 版中的新功能, 238

REGR_R2 函数

9.0.1 版中的新功能, 238

REGR_SLOPE 函数

9.0.1 版中的新功能, 238

REGR_SXX 函数

9.0.1 版中的新功能, 238

REGR_SXY 函数

9.0.1 版中的新功能, 238

REGR_SYY 函数

9.0.1 版中的新功能, 238

reload.sql

11.0.0 版中的增强, 46

RememberLastPlan 属性

10.0.0 版中的新功能, 122

RememberLastStatement 属性

10.0.0 版中的行为更改, 139

10.0.1 版中的行为更改, 93

RemoteCapability 属性

11.0.0 版中的新功能, 33

RemoteputWait 属性

10.0.0 版中的新功能, 122

REORGANIZE TABLE 语句

11.0.0 版中的行为更改, 50

ReqCountActive 属性

10.0.0 版中的新功能, 120

ReqCountBlockContention 属性

10.0.0 版中的新功能, 120

ReqCountBlockIO 属性

10.0.0 版中的新功能, 120

ReqCountBlockLock 属性

10.0.0 版中的新功能, 120

ReqCountUnscheduled 属性

10.0.0 版中的新功能, 120

ReqStatus 属性

10.0.0 版中的新功能, 120

ReqTimeActive 属性

10.0.0 版中的新功能, 120

ReqTimeBlockContention 属性

10.0.0 版中的新功能, 120

ReqTimeBlockIO 属性

10.0.0 版中的新功能, 120

ReqTimeBlockLock 属性

10.0.0 版中的新功能, 120

ReqTimeUnscheduled 属性

10.0.0 版中的新功能, 120

reqtool 实用程序

11.0.0 版中已删除, 82

不建议在 10.0.1 版中使用, 104

request_timeout 选项

10.0.0 版中的新功能, 118

RequestFilterConn 属性

10.0.0 版中的新功能, 122

RequestFilterDB 属性

10.0.0 版中的新功能, 122, 126

RequestLogging 属性

10.0.0 版中的增强, 127

RequestLogMaxSize 属性

10.0.0 版中的新功能, 122

RequestsReceived 属性

10.0.0 版中的新功能, 120

RequestTiming 属性

10.0.0 版中的新功能, 127

reset.sql

重装带有自动增量列的表, 328

RESTORE DATABASE 语句

9.0.1 版中变量的增强支持, 239

RetryConnectionTimeout 连接参数

10.0.0 版中的新功能, 113

RetryConnectionTimeout 属性

10.0.0 版中的新功能, 120

return_java_as_string 选项

10.0.0 版不支持, 141

REVERSE 函数

10.0.0 版中的新功能, 125

REVOKE CONNECT 语句

10.0.0 版中的行为更改, 160

ri_trigger_time 选项

11.0.0 版不支持, 56

RIM

11.0.0 版中的 UltraLite 增强, 71

- ROLLBACK 语句
 - 9.0.1 版中的 UltraLite 动态 SQL 新功能, 246
- ROLLUP 操作
 - 9.0.1 版中的新功能, 238
- ROW_NUMBER 函数
 - 9.0.1 版中的新功能, 238
- RSA
 - 10.0.0 版中的 UltraLite 新功能, 199
 - 10.0.1 版中的增强, 87
 - 随 10.0.0 版 SQL Anywhere 提供, 114
 - 随 10.0.0 版的 MobiLink 提供, 176
 - 随 10.0.0 版的 UltraLite 提供, 196
- rsa_tls
 - 10.0.0 版中重命名的 MobiLink [mlsrv10] 选项, 181
- rsa_tls_fips
 - 10.0.0 版中重命名的 MobiLink [mlsrv10] 选项, 181
- Ruby
 - 11.0.1 版中的新功能, 5
- 任务列表
 - 10.0.0 版中的新功能, 206
- 日期
 - 10.0.0 版中的增强, 136
- 日志传送管理器实用程序 [dbltm]
 - 10.0.0 版中的增强, 116
- 日志文件
 - 10.0.0 版中的增强, 118
- 软件兼容性
 - UltraLite 升级, 345
- S**
- sa_ansi_standard_packages 系统过程
 - 10.0.1 版中的新功能, 88
- sa_char_terms 系统过程
 - 11.0.0 版中的新功能, 34
- sa_check_commit 系统过程
 - 10.0.1 版中的行为更改, 93
- sa_clean_database 系统过程
 - 10.0.0 版中的新功能, 124
- sa_column_stats 系统过程
 - 10.0.0 版中的新功能, 124
- sa_conn_info 系统过程
 - 10.0.0 版中的增强, 127
 - 10.0.1 版中的行为更改, 92
- sa_conn_list 系统过程
 - 10.0.0 版中的新功能, 124
- sa_conn_options 系统过程
 - 10.0.0 版中的新功能, 124
- sa_conn_properties_by_conn 系统过程
 - 10.0.0 版不支持, 167
- sa_conn_properties_by_name 系统过程
 - 10.0.0 版不支持, 167
- sa_conn_properties 系统过程
 - 11.0.0 版中的增强, 36
- sa_convert_ml_progress_to_timestamp 系统过程
 - 10.0.0 版中的新功能, 175
- sa_convert_timestamp_to_ml_progress 系统过程
 - 10.0.0 版中的新功能, 175
- sa_db_list 系统过程
 - 10.0.0 版中的新功能, 124
- sa_db_properties 系统过程
 - 11.0.0 版中的增强, 36
- sa_dependent_views 系统过程
 - 10.0.1 版中的行为更改, 93
- sa_describe_query 系统过程
 - 10.0.0 版中的新功能, 124
- sa_disk_free_space 系统过程
 - 11.0.0 版中的增强, 35
- sa_external_library_unload 系统过程
 - 11.0.0 版中的新功能, 35
- sa_get_bits 系统过程
 - 10.0.0 版中的新功能, 124
- sa_get_dtt_groupreads 系统过程
 - 11.0.0 版中的新功能, 34
- sa_get_dtt 系统过程
 - 10.0.1 版中的行为更改, 93
- sa_get_request_profile 系统过程
 - 10.0.1 版中的行为更改, 93
- sa_get_request_times 系统过程
 - 10.0.1 版中的行为更改, 93
- sa_get_server_messages 系统过程
 - 11.0.0 版中不建议使用, 58
- sa_get_table_definition 系统过程
 - 11.0.1 版中的新功能, 3
- sa_get_user_status 系统过程
 - 11.0.0 版中的新功能, 35
- sa_index_density 系统过程
 - 11.0.0 版中的增强, 35
- sa_internal_text_index_postings 系统过程
 - (仅供内部使用) 11.0.0 版中的新功能, 35
- sa_load_cost_model 系统过程
 - 10.0.0 版中的新功能, 125
- sa_locks 系统过程

10.0.0 版中的行为更改, 139

sa_make_object 系统过程
10.0.0 版中的增强, 124

sa_materialized_view_can_be_immediate 系统过程
11.0.0 版中的新功能, 36

sa_materialized_view_info 系统过程
10.0.0 版中的新功能, 124
10.0.1 版中的行为更改, 93
11.0.0 版中的增强, 35

sa_nchar_terms 系统过程
11.0.0 版中的新功能, 35

sa_performance_diagnostics 系统过程
10.0.0 版中的增强, 127

sa_post_login_procedure 系统过程
11.0.0 版中的新功能, 36

sa_procedure_profile_summary 系统过程
10.0.0 版中的增强, 126

sa_procedure_profile 系统过程
10.0.0 版中的增强, 126

sa_refresh_materialized_views 系统过程
10.0.0 版中的新功能, 124

sa_refresh_text_indexes 系统过程
11.0.0 版中的新功能, 35

sa_remove_tracing_data 系统过程
10.0.0 版中的新功能, 124

sa_reset_identity 系统过程
10.0.0 版中的行为更改, 139
重装带有自动增量列的表, 328

sa_save_trace_data 系统过程
10.0.0 版中的新功能, 125

sa_send_udp 系统过程
9.0.2 版中的新功能, 220

sa_server_messages 系统过程
11.0.0 版中的新功能, 58

sa_server_option 系统过程
10.0.0 版中的增强, 126
11.0.0 版中的增强, 43
9.0.1 版中的增强, 241

sa_set_http_option 系统过程
10.0.0 版中的增强, 134
11.0.1 版中的增强, 3

sa_set_soap_header 系统过程
10.0.0 版中的新功能, 135

sa_set_tracing_level 系统过程
10.0.0 版中的新功能, 125

sa_sever_option 系统过程
11.0.0 版中的增强, 36

sa_snapshots 系统过程
10.0.0 版中的新功能, 125

sa_split_list 系统过程
10.0.0 版中的新功能, 125

sa_table_stats 系统过程
10.0.0 版中的新功能, 125

sa_text_index_handles 系统过程
仅供内部使用, 11.0.0 版中的新功能, 35

sa_text_index_postings 系统过程
仅供内部使用, 11.0.0 版中的新功能, 35

sa_text_index_stats 系统过程
11.0.0 版中的新功能, 35

sa_text_index_vocab 系统过程
11.0.0 版中的新功能, 35

sa_transactions 系统过程
10.0.0 版中的新功能, 110

sa_unload_cost_model 系统过程
10.0.0 版中的新功能, 125

sa_validate 系统过程
10.0.0 版中不建议使用的完整选项, 145
10.0.0 版中不建议使用的数据选项, 145
10.0.0 版中不建议使用的索引选项, 145
10.0.0 版中的行为更改, 139
11.0.0 版中的行为更改, 50
9.0.1 版中的新功能, 240

sa_verify_password 系统过程
9.0.2 版中的新功能, 220

SACHARSET 环境变量
10.0.0 版中的新功能, 142

SADatabase 代理
10.0.0 版中的新功能, 110

SADbType 枚举 [SA .NET 2.0]
10.0.1 版中的增强, 91
10.0.1 版中的行为更改, 95

SADIAGDIR 环境变量
10.0.0 版中的新功能, 212

SADIR 环境变量
10.0.0 版中的新功能, 142

SALANG 环境变量
10.0.0 版中的新功能, 142

SALOGDIR 环境变量
10.0.0 版中的新功能, 142

samples-dir
文档用法, x

SAOLEDB
10.0.0 版中的行为更改, 140

saopts.sql

- 10.0.1 版中的行为更改, 94
- SAServer 代理
 - 10.0.0 版中的新功能, 110
- SASpxOptionsBuilder 类 [SA .NET 2.0]
 - 11.0.0 版不支持, 56
- sasql_commit 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_connect 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_data_seek 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_disconnect 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_errorcode 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_error 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_execute 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_fetch_array 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_fetch_field 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_fetch_object 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_fetch_row 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_free_result 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_identity 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_num_fields 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_num_rows 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_pconnect 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_query 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_result_all 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_rollback 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasql_set_option 函数 (PHP)
 - 11.0.0 版中的新功能, 50
- sasrv.ini
 - 10.0.0 版中的行为更改, 139
- SATMP 环境变量
 - 10.0.0 版中的新功能, 142
 - 11.0.0 版中的增强, 44
- sbgse2.dll
 - 10.0.1 版中的 UltraLite 行为更改, 102
- scale 选项
 - 11.0.0 版中的行为更改, 53
- SCHEMATA 行集, OLE DB
 - 11.0.1 版中的新功能, 5
- ScoutSync
 - 不再支持, 274
- SearchBindery 协议选项
 - 11.0.0 版不支持, 56
- secure_feature_key 属性
 - 10.0.0 版中的新功能, 120
- secure_feature_key 选项
 - 10.0.0 版中的新功能, 115
- SecureFeatures 属性
 - 10.0.0 版中的新功能, 115
- Security Builder
 - 10.0.1 版中的增强, 103
- SELECT 语句
 - 10.0.0 版中的增强, 129
 - 10.0.1 版中的增强, 88
 - 11.0.0 版中的增强, 25, 39, 40
 - 11.0.0 版中的行为更改, 54
 - 11.0.1 版中的增强, 4, 5
- SELinux 策略
 - 11.0.0 版中的新功能, 44
- SendBufferSize 协议选项
 - 10.0.0 版中的增强, 113
- SendingTracingTo 属性
 - 10.0.0 版中的新功能, 123
- SeparateCheckpointLog 属性
 - 10.0.0 版中删除的数据库属性, 167
- SeparateForeignKeys 属性
 - 10.0.0 版中删除的数据库属性, 167
- ServerEdition 属性
 - 11.0.1 版中的新功能, 3
- ServerName 属性
 - 10.0.0 版中的新功能, 122
- ServerNodeAddress 属性
 - 11.0.0 版中的新功能, 32
- ServerPort 协议选项
 - 10.0.0 版中的行为更改, 144
- session_key
 - 10.0.0 版中的 MobiLink [mlsrv10] 新功能, 171

SessionCreateTime 属性
 10.0.0 版中的新功能, 120

SessionID 属性
 10.0.0 版中的新功能, 120

SessionLastTime 属性
 10.0.0 版中的新功能, 120

SET_BITS 函数
 10.0.0 版中的新功能, 126

SET_BIT 函数
 10.0.0 版中的新功能, 125

SET OPTION 语句
 10.0.0 版中的行为更改, 209
 11.0.0 版中已删除, 79

SET PARTNER FAILOVER 子句
 10.0.1 版中的新功能, 90

setup.exe
 10.0.1 版中的 UltraLite 行为更改, 102

SET 语句
 10.0.0 版中的增强, 110

SHA256 散列算法
 10.0.0 版中的新功能, 127

SHARED_DIR 环境变量
 11.0.0 版不支持, 57

Sierra Wireless Aircard
 11.0.0 版不支持, 63

SIMILAR TO 搜索条件
 11.0.0 版中的新功能, 25
 11.0.1 版中的行为更改, 9

SiteScriptName
 不再支持, 271

smartphone
 11.0.0 版中的 UltraLite 增强, 71

SnapshotCount 属性
 10.0.0 版中的新功能, 120, 123

SnapshotIsolationState 属性
 10.0.0 版中的新功能, 123

SNMP
 10.0.1 版中的增强, 92
 11.0.0 版中的增强, 46

SOAP_HEADER 函数
 10.0.0 版中的新功能, 135

SOAPHEADER 子句
 10.0.0 版中的新功能, 135

SOAP 服务
 10.0.0 版中的行为更改, 144

SORTKEY 函数
 10.0.1 版中的增强, 89

sp_hook_dbmsync_all_error
 10.0.0 版中的新功能, 175

sp_hook_dbmsync_communication_error
 10.0.0 版中的新功能, 175

sp_hook_dbmsync_download_com_error
 10.0.0 版中不建议使用, 184

sp_hook_dbmsync_fatal_sql_error
 10.0.0 版中不建议使用, 184

sp_hook_dbmsync_log_rescan
 10.0.0 版中的行为更改, 184

sp_hook_dbmsync_misc_error
 10.0.0 版中的新功能, 175

sp_hook_dbmsync_set_ml_connect_info
 10.0.1 版中的新功能, 97

sp_hook_dbmsync_sql_error
 10.0.0 版中不建议使用, 184
 10.0.0 版中的新功能, 175

Specification 属性
 10.0.1 版中的新功能, 89

SPX
 11.0.0 版不支持, 56

SpxOptionsBuilder 属性 [SA .NET 2.0]
 11.0.0 版不支持, 56

SpxOptionsString 属性 [SA .NET 2.0]
 11.0.0 版不支持, 56

SQL_BIGINT
 10.0.0 版中的行为更改, 141

sql_flagger_error_level 选项
 10.0.1 版中的增强, 88

sql_flagger_warning_level 选项
 10.0.1 版中的增强, 88

SQL/1992
 不建议在 10.0.1 版中使用 SQL Flagger 支持, 95

SQLANY10 环境变量
 11.0.0 版中已重命名, 58

SQLANY11 环境变量
 11.0.0 版中的新功能, 58

sqlanydb
 11.0.0 版中的新功能, 40

SQLANYSAMP10 环境变量
 10.0.0 版中的新功能, 137
 11.0.0 版中已重命名, 58

SQLANYSAMP11 环境变量
 11.0.0 版中的新功能, 58

SQLANYSH10 环境变量
 11.0.0 版不支持, 57

SQL Anywhere

- 11.0.1 版中的新功能, 19
- 文档, viii
- sqlanywhere_commit 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_connect 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_data_seek 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_disconnect 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_errorcode 函数
 - 10.0.0 版中的新功能, 132
- sqlanywhere_errorcode 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_error 函数
 - 10.0.0 版中的新功能, 132
- sqlanywhere_error 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_execute 函数
 - 10.0.0 版中的新功能, 132
- sqlanywhere_execute 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_fetch_array 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_fetch_field 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_fetch_object 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_fetch_row 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_free_result 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_identity 函数
 - 10.0.0 版中的新功能, 132
- sqlanywhere_identity 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_insert_id 函数
 - 10.0.0 版中的新功能, 132
- sqlanywhere_num_fields 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_num_rows 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_pconnect 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_query 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_result_all 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_rollback 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- sqlanywhere_set_option 函数
 - 10.0.0 版中的增强, 132
- sqlanywhere_set_option 函数 (不建议使用)
 - 11.0.0 版中不建议使用, 50
- SQL Anywhere 11
 - 升级到, 323
- SQL Anywhere 11 CustDB 数据源
 - 11.0.0 版中的行为更改, 83
- SQL Anywhere 11 Demo 数据源
 - 11.0.0 版中的行为更改, 83
- SQL Anywhere C API
 - 11.0.0 版中的新功能, 40
- SQL Anywhere Explorer
 - 10.0.0 版中的新功能, 131
- SQL Anywhere for MS Access 迁移实用程序
 - 11.0.0 版不支持, 46
- SQL Anywhere MIB
 - 11.0.0 版中的增强, 46
- SQL Anywhere OEM 版
 - 10.0.1 版中的文档行为更改, 94
 - 10.0.1 版中的行为更改, 94
- SQL Anywhere PHP 模块
 - 10.0.0 版中的增强, 132
 - 10.0.0 版中的行为更改, 142
- SQL Anywhere SNMP Extension Agent
 - 10.0.0 版中的增强, 109, 137
 - 10.0.1 版中的增强, 92
 - 11.0.0 版中的增强, 46
- SQL Anywhere 不建议使用的功能
 - 10.0.0 版, 161
 - 10.0.1 版, 95
 - 11.0.0 版, 55
 - 11.0.1 版, 11
- SQL Anywhere 插件
 - 10.0.0 版中的行为更改, 209
- SQL Anywhere 插件中的归类定制支持
 - 11.0.0 版中的新功能, 74
- SQL Anywhere 广播转发器实用程序 [dbns10]
 - 10.0.0 版中的新功能, 116
 - 11.0.0 版中的行为更改, 58
- SQL Anywhere 控制台实用程序 [dbconsole]
 - 10.0.0 版中的增强, 208
- SQL Anywhere 停用的功能
 - 10.0.0 版, 161

- 10.0.1 版, 95
- 11.0.0 版, 55
- 11.0.1 版, 11
- SQL Anywhere 新功能
 - 10.0.0 版, 109
 - 10.0.1 版, 87
 - 11.0.0 版, 25
- SQL Anywhere 行为更改
 - 10.0.0 版, 138
 - 10.0.1 版, 92
 - 11.0.0 版, 47
 - 11.0.1 版, 8
- SQL Anywhere 直通脚本
 - 11.0.0 版中的 UltraLite 增强, 67
- SQL API
 - 10.0.0 版中的 QAnywhere 新功能, 187
- SQLCA
 - 10.0.1 版中的 UltraLite 限制, 101
- SQL Flagger
 - 10.0.1 版中的增强, 88
- SQLFLAGGER 函数
 - 10.0.1 版中的新功能, 88
- SQLGetConnectAttr
 - 11.0.0 版中的行为更改, 51
- SQLLOCALE 环境变量
 - 10.0.0 版不支持, 161
- SQLPATH 环境变量
 - 10.0.0 版中的行为更改, 145
- sqlpp 实用程序
 - 10.0.1 版中的增强, 88
- SQL Remote
 - ASE 支持已删除, 192
 - MobiLink, 迁移到, 192
 - 不建议使用 ASE 支持, 232
 - 升级, 353
- SQL Remote for ASE
 - 10.0.0 版中已删除, 192
- SQL Remote 的新功能
 - 9.0.2 版, 223
- SQL Remote 新功能
 - 10.0.0 版, 192
 - 11.0.0 版, 66
- SQL Remote 行为更改
 - 10.0.0 版, 192
 - 10.0.1 版, 100
 - 11.0.0 版, 66
 - 9.0.2 版, 232
- SQL SECURITY 子句
 - 11.0.0 版中的新功能, 35
- SQL 预处理器
 - 10.0.1 版中的增强, 88
- SQL 支持
 - 11.0.1 版中的 UltraLiteJ 增强, 17
- SQL 直通
 - 11.0.0 版中的新功能, 60
- SSL
 - 11.0.0 版中的行为更改, 51
- ssqueue
 - 10.0.0 版中已删除, 192
 - 9.0.2 中不建议使用, 232
- ssremote
 - 10.0.0 版中已删除, 192
 - 9.0.2 中不建议使用, 232
- ssxtract
 - 10.0.0 版中已删除, 192
- START AT 子句
 - 10.0.0 版中的 UltraLite 增强, 198
- START DATABASE 语句
 - 10.0.0 版中的增强, 130
 - 10.0.1 版中的增强, 88
- StartDBPermission 属性
 - 10.0.0 版中的新功能, 122
- START SYNCHRONIZATION DELETE 语句
 - 10.0.0 版中的 UltraLite 增强, 198
- StatementDescribes 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- StatementPostAnnotatesSimple 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- StatementPostAnnotatesSkipped 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- StatementPostAnnotates 属性
 - 数据库属性, 11.0.1 版中的新功能, 2
 - 连接属性, 11.0.1 版中的新功能, 2
- Static Java API 支持
 - 10.0.0 版中删除的 UltraLite, 203
- STOP SYNCHRONIZATION DELETE 语句
 - 10.0.0 版中的 UltraLite 增强, 198
- StreamErrorParameters 属性
 - 11.0.1 版中的 UltraLite 增强, 15
- StreamsUsed 属性
 - 11.0.0 版中的新功能, 33

- string_truncation 选项
 - 10.0.0 版中的行为更改, 140, 146
- StringHistogramsFix 属性
 - 10.0.0 版中删除的数据库属性, 167
- STRIP ON 子句
 - 10.0.0 版中不建议使用, 161
- SUBSTR 函数
 - 10.0.0 版中的 QAnywhere 新功能, 189
- Sybase Central
 - 10.0.0 版中的增强, 206
 - 9.0.1 版中的新功能, 243
 - 升级数据库, 334
 - 管理旧数据库, 324
- Sybase Central 新功能
 - 10.0.0 版, 206
 - 10.0.1 版中 SQL Anywhere 插件的增强, 90
 - 11.0.0 版, 73
 - 11.0.1 版, 19
- Sybase Central 行为更改
 - 10.0.0 版, 208
 - 11.0.0 版, 76
 - 11.0.1 版, 20
- Sybase Central 中的模型模式
 - 10.0.0 版中的 MobiLink 新功能, 168
 - 10.0.1 版中的 MobiLink 新功能, 96
- Sybase 中继服务器托管服务
 - 11.0.0 版中的新功能, 60
- SYNC_PROFILE_OPTION_VALUE 函数 [System]
 - 11.0.1 版中的 UltraLite 增强, 15
- syncase125.sql
 - 10.0.0 版中已删除, 169
- synchronize_mirror_on_commit 属性
 - 10.0.0 版中的新功能, 120
- synchronize_mirror_on_commit 选项
 - 10.0.0 版中的新功能, 109, 118
- syncsa.sql
 - 10.0.0 版中的 MobiLink 新行为, 169
- SYNC 网关
 - 10.0.0 版中新服务器启动的同步功能, 177
- SYSATTRIBUTE
 - 系统表, 10.0.0 版中已删除, 158
- SYSATTRIBUTENAME
 - 系统表, 10.0.0 版中已删除, 158
- SYSCOLLATION
 - 系统表, 10.0.0 版中不建议使用, 156
- SYSCOLLATIONMAPPINGS
 - 系统表, 10.0.0 版中不建议使用, 156
- SYSCOLUMN
 - 兼容性视图, 10.0.0 版中的行为更改, 159
 - 系统表, 10.0.0 版中不建议使用, 156
- SYSCOLUMNS
 - 统一视图, 10.0.0 版中的行为更改, 159
- SYSCONSTRAINT
 - 系统视图, 10.0.0 版中的行为更改, 159
- SYSDEPENDENCY
 - 系统视图, 10.0.0 版中的新视图, 155
- SYSEXTENT
 - 系统表, 10.0.0 版中已删除, 158
- SYSEXTERNLOGINS
 - 系统表, 10.0.0 版中已重命名, 158
- SYSFILE
 - 系统视图, 10.0.0 版中的行为更改, 159
- SYSFILE 系统视图
 - 11.0.0 版中的行为更改, 55
- SYSFKCOL
 - 系统表, 10.0.0 版中不建议使用, 156
- SYSFKKEY
 - 系统视图, 10.0.0 版中的新视图, 155
- SYSFOREIGNKEY
 - 系统表, 10.0.0 版中不建议使用, 156
- SYSHISTORY
 - 10.0.0 版中的系统视图增强, 114
 - 系统表, 9.0.1 版中的新功能, 242
- SYSIDX
 - 系统视图, 10.0.0 版中的新视图, 155
- SYSIDXCOL
 - 系统视图, 10.0.0 版中的新视图, 155
- SYSINDEX
 - 10.0.0 版中的系统视图行为更改, 166
 - 系统表, 10.0.0 版中不建议使用, 156
- SYSINDEXES
 - 11.0.0 版中的行为更改, 48
- SYSINFO
 - 10.0.0 版中的兼容性视图行为更改, 142
- SYSIXCOL
 - 系统表, 10.0.0 版中不建议使用, 156
- SYSJAR
 - 系统视图, 10.0.0 版中的行为更改, 159
- SYSJARCOMPONENT
 - 系统视图, 10.0.0 版中的行为更改, 159
- SYSJAVACLASS
 - 10.0.0 版中的系统视图行为更改, 142
 - 系统视图, 10.0.0 版中的行为更改, 159
- SYSLOGIN

系统表, 10.0.0 版中已删除, 158

SYSLOGINMAP
 11.0.0 版中的新功能, 48
 系统视图, 10.0.0 版中的新视图, 155
 系统视图, 10.0.0 版中的行为更改, 159

SYSLOGINPOLICY
 11.0.0 版中的新功能, 48

SYSLOGINPOLICYOPTION
 11.0.0 版中的新功能, 48

SYSMVOPTION
 系统视图, 10.0.0 版中的新视图, 155

SYSMVOPTIONNAME
 系统视图, 10.0.0 版中的新视图, 155

sysncasa.sql
 在版本 10.0.0 中名称更改为 syncsa.sql, 185

SYSOBJECT
 11.0.0 版中的行为更改, 48
 系统视图, 10.0.0 版中的新视图, 155

SYSOPTBLOCK
 系统表, 10.0.0 版中已删除, 158

SYSOPTJOINSTRATEGIES
 系统视图, 10.0.0 版中已删除, 158

SYSOPTJOINSTRATEGY
 系统表, 10.0.0 版中已删除, 158

SYSOPTORDER
 系统表, 10.0.0 版中已删除, 158

SYSOPTORDERS
 系统视图, 10.0.0 版中已删除, 158

SYSOPTQUANTIFIER
 系统表, 10.0.0 版中已删除, 158

SYSOPTREQUEST
 系统表, 10.0.0 版中已删除, 158

SYSOPTREWRITE
 系统表, 10.0.0 版中已删除, 158

SYSPHYSIDX
 系统视图, 10.0.0 版中的新视图, 155

SYSPROCEDURES
 视图, 10.0.0 版中已删除, 158

SYSPROCPARM
 系统视图, 10.0.0 版中的行为更改, 159

SYSPROCPARMS
 统一视图, 10.0.0 版中的行为更改, 159

SYSPROCS
 系统视图, 10.0.0 版中的新视图, 155

SYSPROXYTAB
 系统视图, 10.0.0 版中的新视图, 155

SYSPUBLICATION
 10.0.0 版中的系统视图行为更改, 175

SYSREMOTEOPTION
 系统视图, 10.0.0 版中的行为更改, 159

SYSREMOTEEUSER
 系统视图, 10.0.0 版中的行为更改, 159

SYSSERVERS
 系统表, 10.0.0 版中已重命名, 158

SYSSOURCE
 系统视图, 10.0.0 版中的新视图, 155

SYSSUBSCRIPTION
 系统视图, 10.0.0 版中的行为更改, 159

SYSSYNC
 系统视图, 10.0.0 版中的行为更改, 159

SYSSYNCSRIPT
 系统视图, 10.0.0 版中的新视图, 155

SYSTAB
 10.0.0 版中的系统视图增强, 137

SYSTABCOL
 系统视图, 10.0.0 版中的新视图, 155

SYSTABLE
 系统表, 10.0.0 版中不建议使用, 156

SYSTEXTCONFIG
 11.0.0 版中的新功能, 48

SYSTEXTIDX
 11.0.0 版中的新功能, 48

SYSTEXTIDXTAB
 11.0.0 版中的新功能, 48

SYSUSER
 系统视图, 10.0.0 版中的新视图, 155

SYSUSERAUTH
 系统视图, 10.0.0 版中不建议使用, 156

SYSUSERAUTHORITY
 系统视图, 10.0.0 版中的新视图, 155

SYSUSERLIST
 系统视图, 10.0.0 版中不建议使用, 156

SYSUSERMESSAGES
 系统表, 10.0.0 版中已重命名, 158

SYSUSERPERM
 系统表, 10.0.0 版中不建议使用, 156

SYSUSERPERMS
 10.0.0 版中不建议使用的系统视图, 156

设备
 10.0.0 版中的 UltraLite 增强, 195
 10.0.1 版中的 UltraLite 增强, 101
 11.0.0 版中的 UltraLite 增强, 69

审计
 10.0.0 版中的增强, 113

- 10.0.0 版中的行为更改, 141
- 11.0.0 版中的增强, 28
- 11.0.0 版中的新功能, 74
- 生成数据库文档
 - 11.0.0 版中的新功能, 74
- 升级
 - 10.0.0 版中的行为更改, 138
 - MobiLink, 337
 - MobiLink 服务器, 342
 - MobiLink 系统表, 337
 - MobiLink 统一数据库, 337
 - QAnywhere, 344
 - SQL Anywhere, 324
 - SQL Remote, 353
 - UltraLite 应用程序代码, 349
 - UltraLite 应用程序途径, 349
 - UltraLite 数据库文件格式, 346
 - UltraLite 数据库途径, 346
 - 为版本 11 重建数据库, 329, 333
 - 关于升级到版本 11, 323
 - 具有实例化视图的数据库, 324
 - 同步脚本, 343
 - 数据库和应用程序的 UltraLite 概述, 345
 - 数据库镜像, 335
 - 远程数据库, 342
 - 镜像系统中的数据库, 335
 - 预防措施, 326
- 升级 SQL Anywhere
 - 关于, 323
- 升级到版本 11.0.0
 - 关于, 323
- 升级实用程序 [dbupgrad]
 - 10.0.0 版中的行为更改, 138, 142
 - 11.0.1 版中的增强, 3
- 升级数据库
 - 9.0.2 版中不建议使用的旧文件格式, 230
 - UltraLite 升级途径, 346
 - UltraLite 注意事项, 346
 - 从 Sybase Central, 334
 - 限制, 329
- 升级数据库向导
 - 10.0.0 版中的行为更改, 138
 - 9.0.1 版中的增强, 243
 - UltraLite 版本 11.0.0 示例, 347
 - 使用, 334
- 升级应用程序
 - UltraLite 升级途径, 349
 - UltraLite 进程, 349
 - 连接代码, 349
- 实例化视图
 - 10.0.0 版中的新功能, 110
 - 10.0.1 版中的增强, 91
 - 11.0.0 版中的增强, 27
 - 升级注意事项, 324
- 实体关系选项卡
 - 10.0.0 版中的新功能, 206
- 实用程序
 - 10.0.1 版中的 UltraLite 行为更改, 102
 - UltraLite 生成 [ulgen] 实用程序, 349
- 实用程序数据库
 - 10.0.0 版中的新功能, 118
 - 10.0.0 版中的行为更改, 167
 - 11.0.0 版中的行为更改, 51
- 实用程序选项卡
 - 10.0.0 版中的行为更改, 209
- 使用通用访问卡的客户端验证
 - 11.0.1 版中的新功能, 12
- 示例
 - 10.0.0 版中新的缺省安装位置, 213
- 示例数据库
 - 在 10.0.0 版中重命名, 213
 - 11.0.0 版中的 SQL Anywhere 增强, 82
- 事件
 - 11.0.1 版中的增强, 19
- 事件挂接
 - 9.0.1 版中的 MobiLink 的新功能, 252
- 事件日志
 - 10.0.0 版中的增强, 130
- 事务
 - 9.0.1 版中的 UltraLite 动态 SQL 新功能, 246
- 事务日志
 - 10.0.0 版中的增强, 114
 - 11.0.0 版中的增强, 25, 26
 - 9.0.2 版中的新功能, 219
- 事务日志偏移
 - 升级数据库文件, 329, 333
- 视图
 - 11.0.0 版中的增强, 44
- 视图匹配
 - 10.0.0 版中的新功能, 110
- 视图依赖性
 - 10.0.0 版中的新功能, 111
- 收藏夹列表
 - 11.0.0 版中的新功能, 75

- 授权
 - 10.0.1 版中的行为更改, 104
 - 受保护的功能
 - 10.0.0 版中的新功能, 115
 - 11.0.0 版中的行为更改, 51
 - 输入法编辑器
 - 9.0.1 版中的行为更改, 251
 - 属性
 - 10.0.0 版中的行为更改, 139
 - 属性窗口
 - 11.0.0 版中不建议使用的 UltraLite 功能, 77
 - 属性函数
 - 10.0.0 版中的增强, 126
 - 数据包大小
 - 11.0.0 版中增大了缺省值, 45
 - 数据库
 - 10.0.0 版不支持压缩, 162
 - 10.0.0 版不支持的写文件, 162
 - SQL Anywhere 10.0.0 支持的版本, 137
 - 从版本 10.0.0 和更高版本升级文件格式, 333
 - 从版本 9 和更早版本升级文件格式, 329
 - 升级, 324
 - 数据库大小
 - 11.0.0 版中的增强, 45
 - 数据库服务器
 - 10.0.0 版中的增强, 116
 - 数据库服务器消息窗口
 - 10.0.0 版中的增强, 136, 137
 - 数据库工具导入库
 - 11.0.0 版中的新功能, 51
 - 数据库加密
 - 11.0.0 版中的增强, 29
 - 数据库镜像
 - 10.0.0 版中的新功能, 109
 - 10.0.1 版中的增强, 90
 - 11.0.0 版中的增强, 27
 - 11.0.0 版中的行为更改, 52
 - 升级, 335
 - 应用 EBF, 335
 - 数据库属性
 - 10.0.0 版中的行为更改, 139
 - 11.0.0 版中的新功能, 33
 - 数据库属性 (UltraLite)
 - 10.0.0 版中的增强, 196
 - 9.0.1 版中的新功能, 248
 - 数据库文档向导
 - 11.0.0 版中的新功能, 74
 - 数据库文件格式
 - 9.0.2 版中不建议使用的旧格式, 230
 - UltraLite 升级, 346
 - 升级, 329, 333
 - 数据库选项
 - 10.0.0 版中的行为更改, 145
 - 11.0.0 版中的增强, 25
 - 数据库中的 Java
 - 10.0.0 版中的行为更改, 141
 - 数据类型转换
 - 10.0.1 版中的行为更改, 94
 - 数据源实用程序 [dbdsn]
 - 10.0.0 版中的增强, 115
 - 10.0.0 版中的行为更改, 162
 - 10.0.1 版中的增强, 92
 - 数值数据类型
 - 10.0.0 版中的行为更改, 144
 - 刷新
 - 实例化视图, 11.0.0 版中的行为更改, 52
 - 死锁选项卡
 - 10.0.0 版中的新功能, 206
 - 索引
 - 10.0.0 版中的增强, 135
 - 10.0.0 版中的行为更改, 140
 - 11.0.0 版中的增强, 25
 - 11.0.0 版中的性能增强, 45
 - 11.0.0 版中的行为更改, 55
 - 索引共享
 - 10.0.0 版中的新功能, 135
 - 索引顾问
 - 10.0.0 版中的增强, 136
 - 索引提示
 - 9.0.1 版中的新功能, 241
 - 锁
 - 10.0.0 版中的增强, 112
 - 11.0.0 版中的 Interactive SQL 新功能, 75
 - 11.0.1 版中的 Interactive SQL 行为更改, 20
 - 锁定
 - 10.0.0 版中的增强, 111
 - 11.0.0 版中的行为更改, 54
- ## T
- Table API
 - 11.0.0 版中的 UltraLite 增强, 71
 - TableBitMaps 属性
 - 10.0.0 版中删除的数据库属性, 167
 - TableOrderChecking dbmlsync 扩展选项

- 10.0.0 版中的新功能, 175
 - TablesQualTriggers
 - 10.0.0 版中删除的数据库属性, 167
 - TCP/IP
 - 10.0.0 版中的行为更改, 143, 144
 - TcpIpAddresses 属性
 - 11.0.0 版中的新功能, 33
 - TDS DATE
 - 10.0.1 版中的新功能, 91
 - TDS TIME
 - 10.0.1 版中的新功能, 91
 - temp_space_limit_check 选项
 - 10.0.0 版中的行为更改, 146
 - 调试模式
 - 11.0.0 版中的增强, 76
 - timeout 协议选项
 - 10.0.0 版中新 MobiLink 客户端协议选项, 174
 - timestamp_format 选项
 - 10.0.0 版中的更改行为, 146
 - TLS
 - 10.0.0 版中的 UltraLite 新加密类型, 196
 - 11.0.0 版中的行为更改, 51
 - TOP 子句
 - 9.0.1 版中变量的增强的支持, 239
 - TRACED_PLAN 函数
 - 10.0.0 版中的新功能, 126
 - TransactionsSpanLogs 属性
 - 10.0.0 版中删除的数据库属性, 167
 - TransactionStartTime 属性
 - 9.0.1 版中的新功能, 243
 - Transact-SQL 外连接
 - 版本 11 升级的疑难解答, 333
 - Treo 600
 - 10.0.0 版中添加的支持, 178
 - Treo 650
 - 10.0.0 版中添加的支持, 178
 - truncate_date_values 选项
 - 10.0.0 版中的行为更改, 146
 - 9.0.2 版中不建议使用, 230
 - truncate_with_auto_commit 选项
 - 11.0.0 版不支持, 56
 - TRUNCATE TEXT INDEX 语句
 - 11.0.0 版中的新功能, 38
 - tsql_hex_constant 选项
 - 11.0.0 版不支持, 56
 - tsql_outer_joins 属性
 - 10.0.0 版中的新功能, 120
 - 特权
 - 继承, 11.0.0 版中的新功能, 29
 - 提交
 - 10.0.1 版中的 UltraLite 增强, 101
 - 替代服务器名
 - 10.0.0 版中的新功能, 109
 - 添加表映射向导
 - 10.0.1 版中已删除, 97
 - 通信收到的请求统计信息
 - 10.0.0 版中的新功能, 123
 - 通用访问卡
 - 11.0.1 版中的新功能, 12
 - 同步
 - 10.0.0 版中的 UltraLite 行为更改, 199
 - 同步 ID
 - 10.0.0 版中的 MobiLink 新功能, 172
 - 同步观察器
 - 9.0.1 版中的 UltraLite 增强, 247
 - 同步流
 - 10.0.0 版中 UltraLite 的修订功能, 196
 - 同步流安全
 - 11.0.0 版中的新功能, 62
 - 同步配置文件
 - 11.0.0 版中的 UltraLite 增强, 67
 - 11.0.0 版中的新功能, 62
 - 统计信息
 - 10.0.0 版中的 MobiLink 行为更改, 181
 - 统一数据库
 - 升级, 337
 - 图标
 - 此帮助文档中使用的, xi
 - 图形式计划
 - 10.0.0 版中的行为更改, 208
 - 途径
 - UltraLite 应用程序升级, 349
 - UltraLite 数据库升级, 346
 - 托管的中继服务器
 - 11.0.0 版中的新功能, 60
- ## U
- UL_STORE_PARMS 宏
 - 9.0.0 版中 UltraLite 的行为更改, 275
 - ul_stream_error 结构
 - 11.0.0 版中的 UltraLite 增强, 69
 - ul_synch_info
 - 11.0.0 版中的 UltraLite 行为更改, 72
 - ulafreg 实用程序

10.0.1 版中的行为更改, 102

ulcond.log

10.0.0 版中的 UltraLite 行为更改, 205

ulcond10 实用程序

10.0.0 版中的 UltraLite 增强, 197

ulconv 实用程序

10.0.0 版中已删除, 204

9.0.1 版中的 UltraLite 的新功能, 247

ulcreate 实用程序

10.0.0 版中的 UltraLite 增强, 197

ULEnableGenericSchema 函数

9.0.1 版中不再支持的功能, 252

ulgen 实用程序

10.0.0 版中已删除, 204

UltraLite 数据库升级, 349

ulinfo 实用程序

10.0.0 版中的 UltraLite 新功能, 197

ulinit 实用程序

10.0.0 版中的 UltraLite 增强, 197

数据库升级介绍, 351

ulisql 实用程序

10.0.0 版中已删除, 204

ULjDbT

UltraLiteJ 实用程序, 17

ulload 实用程序

10.0.0 版中的 UltraLite 增强, 197

数据库升级介绍, 348

ulmbreg

名称更改为 ulafreg (10.0.0 版), 205

ulodbc 服务器类

11.0.0 版中的新功能, 45

ULPalmExit 函数

9.0.1 版中不再支持的功能, 252

ULPalmLaunch 函数

9.0.1 版中不再支持的功能, 252

ULRegisterErrorCallback 函数 [UL C/C++]

9.0.1 版中的 UltraLite C/C++ 新功能, 247

ULSQLCONNECT 环境变量

10.0.0 版中的 UltraLite 新功能, 198

ulsync 实用程序

10.0.0 版中的 UltraLite 增强, 198

UltraLite

升级数据库和应用程序, 345

UltraLite_Connection 对象

11.0.0 版中 UltraLite 的新行遍历功能, 71

UltraLite.NET

10.0.0 版中的 UltraLite 增强, 202

11.0.0 版中的 UltraLite 增强, 68, 71

UltraLite.NET 控件

9.0.1 版中的新功能, 246

UltraLite ALTER DATABASE SCHEMA FROM FILE 语句

11.0.0 版中的 UltraLite 新功能, 68

UltraLite ALTER SYNCHRONIZATION PROFILE 语句

UltraLite, 11.0.0 版中的新功能, 67

UltraLite C/C++

10.0.0 版中的 UltraLite 增强, 200

UltraLite C/C++ API

11.0.0 版中的 UltraLite 增强, 71

UltraLite CREATE SYNCHRONIZATION PROFILE 语句

11.0.0 版中的 UltraLite 新功能, 67

UltraLite DROP SYNCHRONIZATION PROFILE 语句

11.0.0 版中的 UltraLite 新功能, 67

UltraLite ESQL

11.0.0 版中的 UltraLite 增强, 71

UltraLite for M-Business Anywhere

10.0.0 版中的 UltraLite 增强, 203

11.0.0 版中的 UltraLite 增强, 71

9.0.1 版中的新功能, 246

UltraLite isolation_level 选项

11.0.0 版中的 UltraLite 新功能, 68

UltraLiteJ

11.0.0 版中的 UltraLite 增强, 71

ULjDbT 实用程序, 17

UltraLiteJ 数据库传输实用程序, 17

UltraLite 校验数据库实用程序 (ulvalid)

11.0.0 版中的 UltraLite 新功能, 68

UltraLiteJ 数据库传输实用程序

BlackBerry, 17

UltraLiteJ 新功能

11.0.1 版, 17

UltraLiteJ 行为更改

11.0.1 版, 16, 18

UltraLite LOAD TABLE 语句

11.0.0 版中的 UltraLite 新功能, 69

UltraLite SELECT 语句

11.0.0 版中的 UltraLite 新功能, 67

UltraLite SQL

10.0.0 版中的 UltraLite 增强, 201

多表连接方法, 267

UltraLite Static C++ API

- 9.0.2 中不建议使用, 233
- UltraLite Static Java API
 - 9.0.2 中不建议使用, 233
- UltraLite SYNCHRONIZE 语句
 - 11.0.0 版中的 UltraLite 新功能, 67
- UltraLite Table API
 - 11.0.0 版中的 UltraLite 增强, 71
- UltraLite 表
 - 11.0.0 版中 UltraLite 的新行遍历功能, 71
- UltraLite 插件
 - 10.0.0 版中的新功能, 207
- UltraLite 初始化数据库实用程序
 - 10.0.0 版中的 UltraLite 增强, 197
- UltraLite 初始化数据库实用程序 (ulinit)
 - 11.0.0 版中的 UltraLite 增强, 70
- UltraLite 计划选项卡
 - 已在 10.0.0 版中删除, 209
- UltraLite 将 XML 卸载到数据库实用程序
 - 10.0.0 版中的 UltraLite 增强, 197
- UltraLite 将 XML 装载到数据库实用程序
 - 10.0.0 版中的 UltraLite 增强, 197
- UltraLite 将数据库卸载到 XML 实用程序 (ulunload)
 - 11.0.0 版中的 UltraLite 增强, 70
- UltraLite 客户端
 - 11.0.0 版中的 UltraLite 增强, 69
- UltraLite 嵌入式 SQL
 - 10.0.0 版中的新功能, 195
 - 9.0.2 中不建议使用, 233
- UltraLite 数据库
 - 10.0.0 版中的新功能, 194
 - 10.0.1 版中的 UltraLite 连接增强, 101
 - 11.0.0 版中的增强, 67
 - 11.0.0 版中的新功能, 67
 - 升级文件格式, 346
 - 升级途径, 346
 - 连接方法, 346
- UltraLite 数据库创建实用程序
 - 10.0.0 版中的 UltraLite 增强, 197
- UltraLite 数据库转换程序
 - 9.0.1 版中的 UltraLite 的新功能, 247
- UltraLite 同步实用程序
 - 10.0.0 版中的 UltraLite 增强, 198
- UltraLite 卸载旧的数据库实用程序
 - 10.0.0 版中的 UltraLite 新功能, 197
- UltraLite 新功能
 - 10.0.0 版, 194
 - 10.0.1 版, 101
 - 11.0.0 版, 67
 - 11.0.1 版, 15
 - 9.0.2 版, 224
- UltraLite 信息实用程序
 - 10.0.0 版中的 UltraLite 新功能, 197
- UltraLite 行为更改
 - 10.0.0 版, 203
 - 10.0.1 版, 102
 - 11.0.0 版, 71
 - 9.0.2 版, 233
- UltraLite 引擎
 - 9.0.1 版中的 UltraLite 的新功能, 247
- UltraLite 应用程序
 - 11.0.0 版中 UltraLite 的新行遍历功能, 71
 - 升级途径, 349
- UltraLite 组件
 - 9.0.1 版中的 table API 行为更改, 252
- ulunloadold 实用程序
 - 10.0.0 版中的 UltraLite 新功能, 197
 - 数据库升级介绍, 348
- ulunload 实用程序
 - 10.0.0 版中的 UltraLite 增强, 197
- ULUtil
 - 名称更改为 ULDBUtil (10.0.0 版), 205
- ulvalid
 - 11.0.0 版中的 UltraLite 新功能, 68
- ulview 实用程序
 - 10.0.0 版中已删除, 204
- ulxml 实用程序
 - 10.0.0 版中已删除, 204
- Unicode
 - UltraLite 升级建议, 346
- UNION 语句
 - 10.0.1 版中的增强, 88
 - 11.0.0 版中的增强, 40
 - 11.0.0 版中的行为更改, 54
- UNION 运算符
 - 10.0.0 版中的 UltraLite 增强, 198
- UniqueClientAddresses 属性
 - 10.0.0 版中的新功能, 140
- UniqueIdentifier 属性
 - 10.0.0 版中删除的数据库属性, 167
 - 11.0.0 版不支持, 56
- UNIQUEIDENTIFIER 数据类型
 - 10.0.0 版中的增强, 131
- Unix
 - 10.0.0 版中的新功能, 133

- 10.0.0 版中的行为更改, 143
 - 10.0.1 版中的增强, 91
 - 11.0.0 版中的新功能, 44
 - unknown_timeout 选项
 - 10.0.0 版中被替换的 MobiLink 客户端协议选项, 174
 - UNLOAD TABLE 语句
 - 10.0.0 版中的增强, 129
 - 11.0.0 版中的增强, 37, 39
 - 9.0.1 版中变量的增强支持, 239
 - UNLOAD 语句
 - 11.0.0 版中的增强, 39
 - 9.0.1 版中变量的增强支持, 239
 - UpdatedRowSource 属性 [SA .NET 2.0]
 - 10.0.1 版中的行为更改, 93
 - UPDATE 语句
 - 10.0.1 版中的增强, 88, 91
 - 11.0.0 版中的增强, 39, 40, 45
 - 11.0.0 版中的行为更改, 54
 - 11.0.1 版中的增强, 5
 - UPDLOCK 表提示
 - 10.0.0 版中的新功能, 129
 - upload_cursor
 - 10.0.0 版中已删除, 179
 - 9.0.0 版中不建议使用, 272
 - upload_deleted_rows
 - 10.0.0 版中的新行为, 182
 - upload_fetch_column_conflict
 - 10.0.0 版中的 MobiLink 新功能, 172
 - upload_inserted_rows
 - 10.0.0 版中的新行为, 182
 - upload_updated_rows
 - 10.0.0 版中的新行为, 182
 - user_estimates 选项
 - 11.0.0 版中的增强, 40
 - UserAppInfo 属性
 - 9.0.1 版中的新功能, 243
 - UTF-16 编码
 - 11.0.0 版中的增强, 44
 - CSCONVERT 函数, 44
 - LOAD TABLE 语句, 44
 - UNLOAD 语句, 44
 - UTF-8
 - 10.0.0 版中的 UltraLite 增强, 196
 - UltraLite 升级建议, 346
 - 在 URL 中的支持, 11.0.0 版中的行为更改, 41
 - UTF8BIN 归类
 - 10.0.0 版中的新功能, 136
 - UTF8 归类
 - 10.0.0 版中不建议使用, 161
 - util_db.ini
 - 10.0.0 版中不建议使用, 167
 - UUID
 - 8.0.2 中的 MobiLink 的新功能, 282
 - uuid_has_hyphens 选项
 - 10.0.0 版中的新功能, 118
 - 11.0.0 版不支持, 56
- ## V
- V4T 模式
 - 9.0.1 版中的 UltraLite 的新功能, 247
 - VALIDATE CHECKSUM 语句
 - 9.0.1 版中的新功能, 240
 - VALIDATE DATABASE 语句
 - 10.0.0 版中的新功能, 128
 - VALIDATE INDEX 语句
 - 10.0.0 版中的增强, 129
 - VALIDATE MATERIALIZED VIEW 语句
 - 10.0.0 版中的新功能, 128
 - VALIDATE TABLE 语句
 - 10.0.0 版中不建议使用的选项, 161
 - VALIDATE 权限
 - 10.0.0 版中的新功能, 115
 - VALIDATE 语句
 - 10.0.0 版中的增强, 127
 - 10.0.0 版中的行为更改, 160
 - ValuePtr 参数
 - 10.0.0 版中的增强, 110
 - VARBIT 数据类型
 - 10.0.0 版中的新功能, 130
 - VariableHashSize 属性
 - 10.0.0 版中删除的数据库属性, 167
 - varray (Oracle)
 - 11.0.1 版中的新功能, 12
 - verify_password_function 属性
 - 10.0.0 版中的新功能, 120
 - verify_password_function 选项
 - 10.0.0 版中的新功能, 118
 - Veritas 聚簇服务器代理
 - 10.0.0 版中的新功能, 110
 - VersionStorePages 属性
 - 10.0.0 版中的新功能, 123
 - VFS
 - 10.0.0 版中的 UltraLite 增强, 197

- viewcert 实用程序
 - 10.0.1 版中的新功能, 103
- VIM 消息类型
 - 11.0.0 版不支持, 66
 - 不建议在 10.0.1 版中使用 SQL Remote 支持, 100
- Vista
 - 11.0.0 版中的增强, 28
 - 自 10.0.1 版起支持, 104
 - 运行 SQL Anywhere 10.0.0 的已知问题, 213
- Visual Studio
 - 11.0.0 版中的行为更改, 83
- Visual Studio .NET
 - 11.0.0 版中的行为更改, 83
- VSS
 - 11.0.0 版中的新功能, 28
- VxWork
 - 不再支持, 274
- W**
- WaitStartTime
 - 连接属性, 11.0.1 版中的新功能, 2
- WaitType
 - 连接属性, 11.0.1 版中的新功能, 2
- WebClientLogFile 属性
 - 11.0.0 版中的新功能, 43
- WebClientLogging 属性
 - 11.0.0 版中的新功能, 43
- Web 服务
 - 10.0.0 版中的 QAnywhere 新功能, 187
 - 10.0.0 版中的增强, 134
 - 10.0.1 版中的增强, 90
 - 11.0.0 版中的行为更改, 51
- Web 服务器
 - 10.0.0 版中的增强, 134
 - 11.0.0 版中的行为更改, 51
- win32 目录
 - 11.0.0 版中的行为更改, 83
- Windows
 - 10.0.0 版中的行为更改, 143
 - 11.0.0 版中的行为更改, 54
 - UltraLite 升级注意事项, 346
- Windows 95
 - 10.0.0 版不支持, 213
- Windows 98
 - 10.0.0 版不支持, 213
- Windows CE 现为 Windows Mobile
 - 11.0.0 版中的行为更改, 82
- Windows Me
 - 10.0.0 版不支持, 213
- Windows Mobile
 - 10.0.0 版中的增强, 132
 - 10.0.1 版中的 UltraLite FIPS 支持更改, 102
 - 11.0.0 版中的增强, 43, 82
 - Certicom Security Builder GSE 版本, 10.0.1 版中的增强, 103
 - UltraLite 升级注意事项针对, 346
 - UltraLite 直接支持, 194
- Windows NT
 - 10.0.0 版不支持, 213
- Windows Vista
 - 11.0.0 版中的增强, 28
 - 运行 SQL Anywhere 10.0.0 的已知问题, 213
- Windows 性能监控器
 - 11.0.0 版中的增强, 32
 - 版本 10.0.0 中删除了 MobiLink 支持, 186
- Winsock
 - 10.0.0 版中的增强, 166
- WITH (XLOCK) 功能
 - 9.0.1 版中的增强, 240
- WITH CONTENT LOGGING 子句
 - 11.0.0 版中的新功能, 26, 37
- WITH FILE NAME LOGGING 子句
 - 11.0.0 版中的新功能, 37
- WITH HASH SIZE 子句
 - 10.0.0 版中已删除, 161
- WITH ROW LOGGING 子句
 - 11.0.0 版中的新功能, 37
- WITH SHARE MODE 子句, REFRESH MATERIALIZED VIEW 语句
 - 11.0.0 版中的新功能, 27
- WRITE_CLIENT_FILE 函数
 - 11.0.0 版中的新功能, 34
- WRITECLIENTFILE 特权
 - 11.0.0 版中的新功能, 30, 34
- 外部卸载
 - 11.0.0 版中的增强, 46
- 外键属性窗口
 - 9.0.1 版中的增强, 243
- 外键约束
 - 10.0.0 版中的行为更改, 140
- 外连接
 - 10.0.0 版中的行为更改, 161
 - 版本 11 升级的疑难解答, 333

- 网络层
 - 10.0.0 版中 MobiLink 的改进功能, 172
 - 10.0.0 版中 UltraLite MobiLink 客户端的改进功能, 199
- 网络协议
 - 10.0.0 版中 MobiLink 修订的功能, 176
 - 10.0.0 版中 UltraLite 的修订功能, 196
- 唯一标识符
 - 10.0.0 版中的增强, 131
- 为 Windows Mobile 保留的堆栈大小
 - 11.0.0 版中的行为更改, 65
- 维护版本
 - 应用到数据库镜像系统, 335
- 维护计划
 - 10.0.0 版中的新功能, 207
- 位数组
 - 10.0.0 版中的新功能, 130
- 谓语
 - 10.0.0 版中的 UltraLite 新功能, 199
- 文本计划
 - 11.0.0 版中不建议使用的 Interactive SQL 功能, 77
- 文本配置对象
 - 11.0.0 版中的新功能, 25
- 文本索引
 - 11.0.0 版中的新功能, 25
 - 11.0.1 版中的行为更改, 8
- 文本完成
 - 10.0.0 版中的新功能, 207
- 文档
 - SQL Anywhere, viii
 - 约定, ix
- 文档增强
 - 10.0.0 版, 211
 - 11.0.0 版, 81
 - 11.0.1 版, 22
 - 9.0.2 版, 228
- 文件
 - 查询, 11.0.0 版中的新功能, 27
- 文件格式
 - 10.0.0 版中的 UltraLite 新功能, 194
 - UltraLite 升级, 346
 - 升级, 329, 333
- 文件名
 - 10.0.0 版不支持的 .cdb 扩展名, 162
 - 10.0.0 版不支持的 .wrt 扩展名, 162
- X**
- X64 目录
 - 11.0.0 版中的行为更改, 83
- xp_cmdshell 系统过程
 - 10.0.0 版中的行为更改, 139
- xp_read_file 系统过程
 - 10.0.0 版中的行为更改, 139
- xp_scanf 系统过程
 - 10.0.0 版中的行为更改, 139
- xp_sendmail 系统过程
 - 10.0.0 版中的增强, 127
- xp_sprintf 系统过程
 - 10.0.0 版中的行为更改, 139
- xp_startsmtp 系统过程
 - 10.0.0 版中的增强, 127
- xp_write_file 系统过程
 - 10.0.0 版中的行为更改, 139
- XPathCompiles 属性
 - 10.0.0 版中的新功能, 123
- X 窗口服务器
 - 10.0.0 版中的行为更改, 144
- 系统表
 - 10.0.0 版中的行为更改, 147
 - 11.0.0 版中的行为更改, 48
- 系统路径
 - UltraLite 实用程序, 345
 - 实用程序, 326
- 系统视图
 - 10.0.0 版中的行为更改, 147
 - 11.0.0 版中的行为更改, 48
- 下载确认
 - 11.0.0 版中的行为更改, 63
- 限制
 - 11.0.0 版中的提高, 46
- 向导
 - 10.0.0 版中的 UltraLite 新功能, 196
 - 11.0.0 版中的增强, 76
 - 11.0.0 版中的新向导, 74
- 消息传递
 - 9.0.1 版中的 MobiLink 的新功能, 244
- 消息存储库, UltraLite
 - 11.0.0 版中的新功能, 64
- 消息存储库, 档案
 - 11.0.0 版中的增强, 64
- 消息存储库, 服务器删除规则
 - 11.0.0 版中的增强, 65

- 消息选择程序
 - 10.0.0 版中的 QAnywhere 新功能, 188
 - 协议选项
 - 10.0.0 版中的增强, 113
 - 写文件
 - 10.0.0 版不支持, 162
 - 卸载
 - 10.0.0 版中的增强, 116
 - 到变量中, 11.0.0 版中的新功能, 26
 - 带有自动增量列的表中的数据, 328
 - 卸载实用程序 [dbunload]
 - 10.0.0 版中的增强, 116
 - 10.0.0 版中的行为更改, 142, 143
 - 10.0.1 版中的行为更改, 94
 - 11.0.0 版中的增强, 30
 - 11.0.0 版中的行为更改, 54
 - 11.0.1 版中的增强, 3
 - 9.0.1 版中不再支持的功能, 250
 - 9.0.1 版中的增强, 241
 - 升级数据库文件, 331
 - 版本 11 升级的疑难解答, 333
 - 卸载数据库
 - 10.0.0 版中的增强, 133
 - 卸载数据库向导
 - 10.0.0 版中的行为更改, 144
 - 重建版本 9 或更早版本的数据库, 330
 - 新功能
 - 10.0.0 版, 107
 - 10.0.1 版, 85
 - 11.0.0 版, 23
 - 11.0.1 版, 1
 - 8.0.1 版, 290
 - 9.0.0 版, 254
 - 9.0.1 版, 238
 - 9.0.2 版, 216
 - 版本 8.0.0, 298
 - 版本 8.0.2, 278
 - 新闻组
 - 技术支持, xiii
 - 信息实用程序 [dbinfo]
 - 10.0.0 版中的增强, 115
 - 星号
 - 11.0.1 版中的全文搜索行为更改, 8
 - 行遍历
 - 11.0.0 版中的 UltraLite 新功能, 71
 - 行号
 - 11.0.0 版中的 Interactive SQL 增强, 77
 - 行为更改
 - 10.0.0 版, 107
 - 10.0.1 版, 85
 - 11.0.0 版, 23
 - 11.0.1 版, 1
 - 8.0.0 版, 298
 - 8.0.1 版, 290, 295
 - 8.0.2 版, 278, 286
 - 9.0.1 版, 238, 250
 - 9.0.2 版, 216, 230
 - 版本 8.0.0, 315
 - 版本 9.0.0, 254, 269
 - 性能
 - 10.0.0 版中的 UltraLite 增强, 194
 - 10.0.0 版中的增强, 111
 - UltraLite 应用程序升级, 349
 - UltraLite 数据库升级, 346
 - 升级数据库文件, 329, 333
 - 性能监控器
 - 10.0.0 版中的增强, 123
 - 11.0.0 版中的增强, 32
 - 修改 HTTP 标头
 - 10.0.1 版中的增强, 90
 - 序列号
 - 10.0.0 版中的 MobiLink 系统表模式更改, 170
 - 选项监视项目列表
 - 11.0.0 版中的新功能, 36
- ## Y
- 压缩列
 - 10.0.0 版中的新功能, 112
 - 11.0.0 版中的增强, 45
 - 压缩数据库
 - 10.0.0 版不支持, 162
 - 压缩数据库向导
 - 10.0.0 版不支持, 163
 - 9.0.1 版中的增强, 243
 - 页面大小
 - 10.0.0 版中更改的缺省, 143
 - 一览表窗格
 - 11.0.0 版中的新功能, 74
 - 移动 Web 服务
 - 10.0.0 版中的新功能, 187
 - 移植
 - UltraLite 应用程序升级, 349
 - 疑难解答
 - UltraLite 命令行实用程序, 345

- 升级数据库到版本 11, 332
- 命令行实用程序, 326
- 新闻组, xiii
- 已提升的操作代理
 - 10.0.1 版中的新功能, 104
- 意图锁
 - 10.0.0 版中的新功能, 112
- 异步 I/O
 - 9.0.1 版中的新功能, 243
- 应用程序
 - UltraLite 升级, 349
- 应用程序分析
 - 10.0.0 版中的新功能, 110
- 映像备份
 - 9.0.1 版中的增强, 241
- 用户 ID
 - 10.0.0 版中的行为更改, 140
- 用户定义的函数
 - 10.0.0 版中的增强, 124
- 用于 .NET 和 C++ 的 dbmlsync API
 - 11.0.0 版中的新功能, 62
- 用于 MobiLink ASA 客户端的按行分区
 - 9.0.1 版中的 MobiLink 的新功能, 252
- 优化程序
 - 11.0.0 版中的增强, 25
 - 11.0.1 版中的增强, 5
- 游标
 - 10.0.0 版中的 UltraLite 增强, 198
- 语句高速缓存命中统计
 - 10.0.1 版中的新功能, 88
- 语句高速缓存未命中统计
 - 10.0.1 版中的新功能, 88
- 语言选择实用程序 [dblang]
 - 10.0.1 版中的行为更改, 93
 - 9.0.1 版中不建议使用的功能, 250
- 域
 - 10.0.0 版中的行为更改, 142
- 预防措施
 - 升级, 326
- 预取
 - 11.0.0 版中的增强, 28
- 远程 DBA 权限
 - 10.0.0 版中的行为更改, 160
- 远程 ID
 - 10.0.0 版中的 MobiLink 新功能, 174
- 远程 ID 文件
 - 10.0.0 版中的新功能, 178

- 远程数据访问
 - 10.0.0 版中的 ODBC 驱动程序更改, 214
 - 11.0.0 版中的行为更改, 55
- 远程数据库
 - 升级, 342
- 约定
 - 命令 shell, xi
 - 命令提示符, xi
 - 文档, ix
 - 文档中的文件名, x
- 约束
 - 10.0.0 版中的 UltraLite 增强, 198
 - 10.0.0 版中的增强, 136
- 运算符优先级, 全文搜索
 - 11.0.1 版中的行为更改, 8

Z

- 增加的连接
 - 10.0.1 版中的 UltraLite 增强, 101
- 增量上载和下载
 - 11.0.0 版中的增强, 64
- 诊断跟踪
 - 10.0.0 版中的新功能, 110
- 诊断目录
 - 10.0.0 版中的新功能, 212
- 正则表达式
 - 11.0.0 版中的新功能, 25
 - 11.0.1 版中的行为更改, 9
- 证书
 - 11.0.0 版中的行为更改, 83
 - 从何处获取, 287
- 支持
 - 新闻组, xiii
- 支持的平台
 - 11.0.0 版中的行为更改, 54
- 支持实用程序
 - 11.0.0 版中的增强, 82
- 支持实用程序 (dbsupport)
 - 10.0.0 版中的新功能, 116
- 支持实用程序 [dbsupport]
 - 10.0.0 版中的新功能, 212
 - 10.0.1 版中的增强, 92
- 直方图实用程序 [dbhist]
 - 10.0.0 版中的增强, 115
- 直接行处理
 - 10.0.0 版中的 MobiLink 新功能, 168
- 直接页扫描

- 11.0.0 版中的 UltraLite 新 Table API 功能, 71
- 执行 SQL 语句
 - 11.0.0 版中的 Interactive SQL 增强, 77
- 执行计划
 - 11.0.0 版中的增强, 46
 - 11.0.1 版中的增强, 5
- 只读
 - 10.0.0 版不支持的写文件, 162
 - 镜像数据库访问, 11.0.0 版中的新功能, 27
- 只读数据库
 - 10.0.0 版不支持的写文件, 162
 - 10.0.0 版中的行为更改, 141
 - 11.0.1 版中的行为更改, 20
- 致命错误
 - 10.0.0 版中的 Unix 增强, 133
- 智能手机
 - 11.0.0 版中的增强, 82
- 主键约束
 - 10.0.0 版中的行为更改, 140
- 主题
 - 图标, xi
- 转换
 - 10.0.0 版中的行为更改, 144
- 转换 NULL 常量
 - 10.0.0 版中的行为更改, 144
- 装载
 - 从事务日志, 11.0.0 版中的新功能, 26
 - 从某个值, 11.0.0 版中的新功能, 26
 - 从某列中, 11.0.0 版中的新功能, 26
- 装载数据
 - 10.0.1 版中的行为更改, 94
- 子查询
 - 9.0.1 版中的 UltraLite 的新功能, 246
- 自定义归类
 - 10.0.0 版不支持, 162
 - 仅在内部卸载时保存, 330
- 自动提交
 - 10.0.1 版中的 UltraLite 增强, 101
 - 11.0.0 版中的行为更改, 55
- 自动停止
 - 11.0.0 版中的行为更改, 52
- 字符长度语义
 - 10.0.0 版中的新功能, 130
- 字符集
 - 10.0.0 版中的 UltraLite 增强, 196
- 字符集转换
 - 10.0.0 版中的增强, 111
 - 10.0.0 版中的行为更改, 139
- 字节序
 - 11.0.0 版中的增强, 44
 - 11.0.1 版中的行为更改, 10
- 字体选择
 - 10.0.0 版中的新功能, 207
- 阻塞
 - 11.0.0 版中的行为更改, 54
- 作为服务登录的特权
 - 由 dbsvc 实用程序授予, 143