



新增功能摘要

Adaptive Server[®] Enterprise

15.7

文档 ID: DC01374-01-1570-01

最后修订日期: 2011 年 9 月

版权所有 © 2011 Sybase, Inc. 保留所有权利。

除非新版本或技术声明中另有说明, 否则本出版物适用于 Sybase 软件及所有后续版本。本文档中的信息如有更改, 恕不另行通知。本出版物中描述的软件按许可证协议提供, 其使用或复制必须符合协议条款。

仅在定期安排的软件发布日期提供升级。未经 Sybase, Inc. 事先书面许可, 本书的任何部分不得以任何形式、任何手段(电子的、机械的、手动、光学的或其它手段)进行复制、传播或翻译。

可在 <http://www.sybase.com/detail?id=1011207> 上的 Sybase 商标页中查看 Sybase 商标。Sybase 和列出的标记均是 Sybase, Inc. 的商标。® 表示已在美国注册。

SAP 和此处提及的其它 SAP 产品与服务及其各自的徽标是 SAP AG 在德国和世界各地其它几个国家/地区的商标或注册商标。

Java 和所有基于 Java 的标记都是 Oracle 和/或其在美国和其他国家/地区的附属机构的商标或注册商标。

Unicode 和 Unicode 徽标是 Unicode, Inc. 的注册商标。

IBM 和 Tivoli 是 International Business Machines Corporation 在美国和/或其它国家/地区的注册商标。

本书中提到的所有其它公司和产品名均可能是与之相关的相应公司的商标。

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

目录

Adaptive Server 15.7 版中的新增功能	1
应用程序功能配置组	1
新增 Adaptive Server 内核	1
在 Adaptive Server 中压缩数据	2
新增安全性功能	2
端到端 CIS Kerberos 鉴定	2
加密密钥和无人值守启动的双控制	3
保护登录名、角色和口令管理扩展	3
登录配置文件	3
员工生命周期管理	4
外部口令和隐藏文本	4
高速缓存语句中的抽象计划	4
缩减日志空间	5
用 sysoptions 显示当前设置的开关	5
大对象的更改	6
行内行外 LOB	6
在存储过程中使用大对象 text 、 unitext 和 image 数据类型	6
在 Transact-SQL 语句中使用 LOB 定位符	6
针对大对象对 where 子句进行了扩展	7
以 XML 格式显示高速缓存计划	7
使用 str 填充字符字段	7
对 select for update 的更改	7
创建非物化、非空列	8
共享行内缺省值	8
保留监控数据	8
分析动态参数	9
监控锁超时	9
截断尾随零	10
完全可恢复的 DDL	10
使用 merge 将行从源表传送到目标表	10

使用 <code>sp_showoptstats</code> 查看统计信息和直方图	10
对游标所做的更改	11
在游标关闭时释放游标锁	11
增强了对游标的事务支持	11
监控游标语句	11
嵌套 <code>select</code> 语句增强	12
对链式事务中的命令和系统过程所做的更改	12
扩展了可变长度的行	13
对 <code>like</code> 模式匹配所做的更改	13
对带引号的标识符所做的更改	14
允许 <code>Unicode</code> 非字符	14
降低查询处理延迟	15
<code>sybdiag</code> 实用程序	15
优化程序诊断实用程序	16
Adaptive Server 15.7 版中的系统更改	17
命令	17
函数	20
系统存储过程	21
配置参数	23
系统表	25
实用程序	26
监控表更改	26
新的监控表	27
对监控表的更改	27
全局变量	32
Adaptive Server 15.5 版 Cluster Edition 中的新增功能	33
Adaptive Server 15.5 Cluster Edition 功能和平台表	33
多个同时发生的故障切换	34
为归档数据库增加空间	35
共享磁盘集群中的分布式事务管理	35
Adaptive Server 15.5 Cluster Edition 的系统更改	37
已更改的命令	37
监控表	37
配置参数	38

函数	38
Adaptive Server® 15.5 版中的新增功能	41
Adaptive Server 15.5 功能和平台表	41
内存数据库和宽松持久性数据库	42
针对备份的更快压缩	43
针对 IBM® Tivoli® Storage Manager 的 Backup Server 支持	44
用户定义的存储过程的延迟名称解析	44
FIPS 140-2 登录口令加密	45
增量数据传输	45
bigdatetime 和 bigtime 数据类型	46
创建和管理 tempdb 组	46
Adaptive Server 15.5 中的系统更改	47
数据类型	47
函数	47
系统存储过程	48
命令	50
配置参数	52
监控表	52
系统表	52
实用程序	53
审计	54
Adaptive Server 15.0.3 中的新增功能	55
SQL 语句复制	55
安全性增强	55
LDAP 用户鉴定增强	55
自动 LDAP 用户鉴定和故障恢复	55
外部鉴定的登录映射	56
使用 SSL 指定公用名	56
并发 Kerberos 鉴定	56
虚拟散列表	57
巨内存页	57
高可用性配置期间的升级	58
重新安装系统存储过程	58

分布式事务管理 (DTM)	58
Adaptive Server 插件更新	59
Java 接口	59
Adaptive Server 15.0.3 中的系统更改	61
函数	61
系统存储过程	61
命令	62
配置参数	62
监控表	63
系统表	63
Adaptive Server 15.0.2 中的新增功能	65
加密列	65
归档数据库访问	65
查找慢速运行的查询	66
延迟编译	66
用于中文和日文字符集的不区分大小写的排序顺序	67
统计集合函数	67
标准偏差和方差	68
积极和消极集合	68
矢量和标量集合	69
数据插入的性能改进	69
在页面拆分期间使用异步写入	70
提高 tempdb 事务的吞吐量	70
提交后优化	71
对查询处理器的更改	71
延迟编译	71
非二进制字符集直方图插值	71
表达式直方图选择性估计	72
查看当前的优化程序设置	72
新增安全性功能	72
64 位 Adaptive Server 在 AIX 上支持 PAM	72
全局登录触发器自动设置	73
SSL 支持	73
改进的口令安全性	73

审计增强	73
高可用性考虑事项	74
安装和编辑监控表	74
语句高速缓存的监控表	75
系统表的行级锁定	75
xmltable() 函数	76
重新分配的连接	76
用户定义的 SQL 函数	77
instead of 触发器	77
Adaptive Server 15.0.2 中的系统更改	79
跟踪标志	79
命令	79
对 set 命令的更改	80
实用程序	80
系统存储过程	81
系统表	84
配置参数	86
函数	87
全局变量	87
Adaptive Server 15.0.1 中的新增功能	89
抽象计划更改	89
新的查询级设置	89
抽象计划和优化程序条件的运算符名称的一致性	90
扩展优化程序条件 set 语法	90
文字参数化	91
Adaptive Server 15.0.1 中的系统更改	93
函数	93
配置参数	93
命令	94
监控表	96
Adaptive Server 15.0 中的新增功能	97
分区支持	97
行锁定的系统目录	98

查询处理器	98
长标识符	99
计算列	99
计算列和基于函数的索引的区别	99
实现计算列与非实现计算列之间的区别	100
可滚动游标	100
unitext 数据类型支持	100
big int 数据类型支持	101
无符号整数数据类型支持	101
整数标识	102
对 XML 服务的增强	102
Adaptive Server 插件增强	103
Interactive SQL	103
用户定义的 Web 服务	103
超大容量存储支持	104
update statistics 的自动运行	104
SySAM 许可证管理	105
查询处理指标 (qp 指标)	105
对抽象计划的更新	106
showplan 更改	106
安全套接字层使用 FIPS 140-2	107
Adaptive Server 15.0 中的系统更改	109
实用程序	109
保留字	109
全局变量	110
配置参数	110
函数	112
命令	113
系统存储过程	114
系统表	115
监控表	119
获取帮助及其它信息	121
索引	123

Adaptive Server 15.7 版中的新增功能

Adaptive Server® 15.7 版引入了许多新的功能和增强。

应用程序功能配置组

Adaptive Server 15.7 版在配置文件添加了应用程序功能配置组。

下面是应用程序功能配置参数

- `enable functionality group`
- `select for update`
- `streamlined dynamic SQL`
- `enable inline default sharing`
- `enable permissive unicode`
- `quoted identifier enhancements`

使用 `enable functionality group` 可启用或禁用该组中的所有配置参数。启用或禁用各个配置参数可覆盖组值。

请参见《系统管理指南第一卷》中的“设置配置参数”。

新增 Adaptive Server 内核

Adaptive Server 15.7 版和更高版本包括两个内核：线程化内核和进程内核。

您为 Adaptive Server 配置的内核决定 Adaptive Server 的运行模式：

- 线程化模式 - Adaptive Server 作为单个多线程化操作系统进程运行，通过在线程池中的线程上运行的引擎来处理 SQL 查询。线程化模式利用没有引擎的线程来管理 I/O。管理员可以配置更多线程池来管理负载。
- 进程模式 - 以前运行 Adaptive Server 的遗留内核。在进程模式中，Adaptive Server 作为多个彼此合作、如同单个服务器一样工作的操作运行。进程模式使用引擎来管理 I/O，管理员配置引擎组来管理负载。

对于许多负载，线程化模式使用的 CPU 比进程模式少得多，但能提供同样的，甚至更好的性能。线程化模式不需要太多任务-引擎密切连接，因此，能在混有 I/O 密集和 CPU 密集的负载中提供更加一致的性能。

线程化内核能让 Adaptive Server 利用比较低版本内核具有更多处理器、处理器内核以及硬件线程的并行硬件和支持系统。虽然 15.7 版更改了内核，但查询处理器仍保持不变。要在线程化内核模式中运行，您无需更改大多数针对较低版本 Adaptive Server

编写的脚本，只是有几个命令和存储过程发生了更改。应用程序完全与线程化模式兼容。

线程化模式是 Adaptive Server 的缺省模式，也是 Sybase® 建议使用的模式。Windows 平台上的 Adaptive Server 仅在线程化模式中运行。

请参见《系统管理指南 第二卷》，以获取有关为 Adaptive Server 配置线程化模式的信息。

在 Adaptive Server 中压缩数据

Adaptive Server 15.7 版引入了数据压缩，能让您使用更少的存储空间存储等量数据，减少高速缓存内存耗用量，降低 I/O 需要而提高性能。

可以压缩大对象 (LOB) 和常规数据。

您创建压缩表或分区后，Adaptive Server 会压缩所有后来插入或更新的数据（也就是说，现有数据还未压缩）。如果 Adaptive Server 无法有效地压缩插入的数据，就会保留原来的行。如果新插入或更新的 LOB 数据占用小于或等于单个数据页的空间，Adaptive Server 便不会压缩该数据。

无需解压缩数据即可对其进行查询。可以插入、更新和删除压缩数据，对压缩列运行 **select** 或 **readtext** 语句会返回解压缩的行。由于需要 Adaptive Server 检索的数据较少，因此，I/O 较低，数据存储的效率较高。

数据压缩是单独许可的选项。请参见《压缩用户指南》。

新增安全性功能

Adaptive Server 15.7 版添加了以下安全性功能：端到端 CIS Kerberos 鉴定、加密密钥和无人值守启动的双控制、保护登录名、角色和口令管理扩展，以及登录配置文件。

端到端 CIS Kerberos 鉴定

Adaptive Server 15.7 版中包括通过组件集成系统 (CIS) 为远程 Adaptive Server 连接提供的端到端 CIS 鉴定支持。

端到端 (CIS) Kerberos 鉴定能让使用 Kerberos 登录到 Adaptive Server 的 Kerberos V5 用户在以下情况下使用 Kerberos 统一登录鉴定连接到远程 Adaptive Server：

- 请求对 Adaptive Server 进行 RPC
- 发出 CIS 直通连接
- 使用 CIS 对远程 Adaptive Server 发出常规分布式查询处理请求

Adaptive Server 支持使用 CIS 对与远程 ASE 服务器的 Kerberos 连接使用以下可选安全服务：

- 消息保密性
- 消息完整性
- 相互鉴定

请参见《安全性管理指南》。

加密密钥和无人值守启动的双控制

Adaptive Server 15.7 版添加了加密密钥和无人值守启动的双控制功能。

对双控制和分离知识以及无人值守启动所做的更改包括：

- `master` 和 `dual master` 系统密钥是数据库级密钥，由具有 `sso_role` 或 `keycustodian_role` 的用户创建，被用作用户创建的加密密钥的密钥加密密钥 (KEK)，可实现更高的安全性和更多数据加密密钥分离知识。`master` 密钥替换了当前系统加密口令，但 Adaptive Server 将继续支持它以保持向后兼容性。Sybase 建议用户不再使用系统加密口令来加密数据加密密钥。
- 能够使用 SQL 命令和通过专用文件来为 `master` 和 `dual master` 密钥提供支持。主密钥的口令是非持久的：它们不存储在数据库中。
- 能够通过双控制和分离知识来保护所有用户创建密钥。

请参见《加密列用户指南》。

保护登录名、角色和口令管理扩展

Adaptive Server 15.7 版中包括对保护登录名、角色和口令管理的扩展。

Adaptive Server 通过以下各项改进了登录名、角色及其口令：

- 磁盘上存储的角色口令的更强加密。
- 锁定的角色计数。
- 登录名、角色和全局口令策略选项的 ISO 8601 持续时间规范
- 口令管理扩展，如自定义 Rivest-Shamir-Adleman (RSA) 密钥对重新生成期间、将口令复杂性检查扩展到角色口令，以及扩展高可用性环境中的口令管理支持。
- 对角色定义的审计增强、激活、由于激活尝试失败而进行的角色锁定，以及口令管理扩展，如 RSA 密钥对重新生成。

请参见《安全性管理指南》。

登录配置文件

Adaptive Server 15.7 添加了登录配置文件，它们是 SQL 为登录属性及其值定义的容器。

登录帐户是用属性定义和管理的。例如，当您添加或修改登录帐户时，以下属性与登录帐户关联：

- 将使用的数据库

Adaptive Server 15.7 版中的新增功能

- 将自动激活的角色
- 将应用的语言
- 当您登录到 Adaptive Server 时将调用的登录脚本

可以将某些属性与所有登录帐户相关联（具体做法是在缺省登录配置文件中指定它们），将其它属性与一组特定的登录帐户相关联（具体做法是在登录配置文件中指定它们，并将该登录配置文件与登录帐户相关联）。

请参见《安全性管理指南》。

员工生命周期管理

系统安全员或数据所有者可以使用 **alter... modify owner** 来移交数据库对象的所有权。

此命令为数据库管理员提供一种方法来管理由于员工更改而带来的对象分配，以及分离数据库对象的创建所有权。

外部口令和隐藏文本

Adaptive Server 15.7 使用 AES-256 对称加密算法为外部登录口令和隐藏文本提供强加密。

外部口令的强加密适用于以下各项的口令：

- Replication Agent - 复制型数据库口令。
- CIS - 远程描述符和登录名的口令。
- Job Scheduler - Job Scheduler 代理的口令。
- RTMS - 实时消息传送服务的口令。
- 安全套接字层 (SSL) 和轻量目录访问协议 (LDAP) - SSL 和 LDAP 访问帐户的口令。口令是使用可以保护的存储过程 `sp_ldapadmin` 和 `sp_ssladmin` 管理的。

高速缓存语句中的抽象计划

Adaptive Server 15.7 版引入了在语句高速缓存中保存抽象计划信息的功能。

在这个包括抽象计划的示例中，散列表会保存 **select * from t1 plan '(use optgoal allrows_mix)'**（如 SQL TEXT 行中所示）：

```
1> select * from t1 plan '(use optgoal allrows_mix)'  
2> go  
1> dbcc prsqlcache  
2> go
```

```
Start of SSQL Hash Table at 0x0x1474c9050
```

```
Memory configured: 1000 2k pages          Memory used: 17 2k pages
```

```
Bucket# 243 address 0x0x1474c9f80
```

```

SSQL_DESC 0x0x1474cd070
ssql_name *ss0626156152_0290084701ss*
ssql_hashkey 0x0x114a575d          ssql_id 626156152
ssql_suid 1          ssql_uid 1          ssql_dbid 1          ssql_spid 0
ssql_status 0x0xa0          ssql_parallel_deg 1
ssql_isolate 1          ssql_tranmode 32
ssql_keep 0          ssql_usecnt 1          ssql_pgcount 6
SQL TEXT: select * from t1 plan '(use optgoal allrows_mix)'
End of SSQL Hash Table

```

在低于 15.7 版的 Adaptive Server 中，SQL TEXT 行仅包括 `select * from t1` 命令，没有 `plan` 子句。

请参见 Performance and Tuning Series: Monitoring Adaptive Server with `sp_sysmon`（《性能和调优系列：用 `sp_sysmon` 监控 Adaptive Server》）。

缩减日志空间

在 Adaptive Server 15.7 版和更高版本中，`alter database` 包括 `log off` 参数，用于删除数据库日志的不必要部分，能让您缩减日志空间以及无需重新创建数据库即可释放存储空间。

`log off` 可能在数据库结束时不再需要的额外分配的空间时，在运行数据库操作的完全记录选项（如 `select into`、`alter table` 或 `reorg rebuild`）后特别有用。

请参见本文档中的“完全可恢复的 DDL”以及《管理指南第二卷》中的“创建和管理用户数据库”。

用 sysoptions 显示当前设置的开关

Adaptive Server 15.7 版向 `sysoptions` 表中添加了 `number` 列，用于包含当前设置的开关的开关 ID。

`sysoptions` 显示以下开关：

- 在 `runserver` 文件中用 `-T` 标志设置的跟踪标志
- 用 `dbcc traceon(flag_number)` 或 `set switch serverwide on` 设置的跟踪标志
- 用 `set switch on` 为特定系统进程 ID (`spid`) 设置的跟踪标志和开关

`sysoptions` 仅显示用户可看到的开关。也就是说，用户看不到由其它 `spid` 私下设置的开关。对于所有非开关的选项类别，数字的值为空。

大对象的更改

Adaptive Server 15.7 版中包括大对象 (LOB) 的更改，如存储小的 `text`、`image` 和 `unitext` 数据类型的行内 LOB 列，存储包含 LOB 的已声明 SQL 语句，在 Transact-SQL 语句中间接引用 LOB，以及允许检查有无大对象的空值。

行内行外 LOB

Adaptive Server 15.7 支持存储 `text`、`image` 和 `unitext` 数据类型的行内 LOB 列（当它们较小而且符合页中的可用空间时）。

当 LOB 扩展大小或其空间用于其它行内列（如用于 `varchar` 和 `varbinary` 数据类型空间）时，Adaptive Server 可无缝地将行内 LOB 数据迁移到行外存储空间中，从而自动将该数据替换为行内文本指针。

在 Adaptive Server 15.7 中，可以使用：

- **create table** 来指定 LOB 列的行内存储
- **alter table** 来执行 LOB 列存储方式的修改
- **create database** 或 **alter database** 命令来管理 LOB 列的数据库范围行内长度

请参见《Transact-SQL 用户指南》中的“**In-Row, Off-Row LOB**”（行内、行外 LOB）。

在存储过程中使用大对象 `text`、`unitext` 和 `image` 数据类型

声明这些数据类型后，Adaptive Server 会存储包含 LOB 的 SQL 语句。

在 Adaptive Server 15.7 和更高版本中，您可以：

- 为本地变量声明大对象 (LOB) `text`、`image` 或 `unitext` 数据类型，并将该变量作为输入参数传递给存储过程。
- 准备包括 LOB 参数的 SQL 语句。

请参见《参考手册：构件块》。

在 Transact-SQL 语句中使用 LOB 定位符

大对象 (LOB) 定位符能让您在 Transact-SQL 语句中间接引用 LOB，而不是引用 LOB 自身。

由于 `text`、`unitext` 或 `image` LOB 的大小可能是许多兆字节，因此，在 Transact-SQL 语句中使用 LOB 定位符会减少客户端和 Adaptive Server 之间的网络通信量，减少客户端处理 LOB（如果不使用 LOB 定位符的话）所需的内存量。

Adaptive Server 15.7 能让客户端应用程序将定位符作为主机变量和参数标记来发送和接收。

当您创建 LOB 定位符时，Adaptive Server 会将 LOB 值高速缓存到其内存中，并生成 LOB 定位符以引用它。

创建 LOB 定位符后，它会在创建它的事务的存续期间内一直有效。当该事务提交或回退时，Adaptive Server 会使定位符无效。

LOB 定位符使用三种不同的数据类型

- `text_locator` - 用于 `text` LOB
- `unitext_locator` - 用于 `unitext` LOB
- `image_locator` - 用于 `image` LOB

请参见《Transact-SQL 用户指南》中的“使用和创建数据类型”。

针对大对象对 **where** 子句进行了扩展

`where` 子句得到了扩展，可允许检查有无大对象的空值。

请参见《参考手册：命令》。

以 XML 格式显示高速缓存计划

`show_cached_plan_in_xml` 函数针对高速缓存中的语句以 XML 格式返回 `showplan` 输出。

在使用 `show_cached_plan_in_xml` 之前，必须先启用语句高速缓存。

请参见 Performance and Tuning Series: Query Processing and Abstract Plans（《性能和调优系列：查询处理和抽象计划》）中的“显示查询优化策略和评估值”。

使用 **str** 填充字符字段

在 Adaptive Server 15.7 版中，`str` 函数的 `decimal` 参数得到了扩展，允许用指定字符或数字来填充字段。

请参见《参考手册：构件块》。

对 **select for update** 的更改

Adaptive Server 15.7 版支持 `select for update`，用以针对同一事务内的后续更新以及可更新的游标以排它方式锁定行。这可防止其它并发任务更新这些行或阻止后续更新。`select for update` 在隔离级别 1、2 和 3 受到支持。

可以在游标上下文之外发出 `select for update` 作为语言语句。如果既有语言语句，又有游标，您就必须在 `begin transaction` 命令中或在链式模式中执行 `select for update`。

如果在游标上下文内运行 **select for update**，游标 **open** 和 **fetch** 语句必须位于事务上下文中，否则 Adaptive Server 会恢复到 15.7 之前的功能。

请参见《Transact-SQL 用户指南》中的“查询：从表中选择数据”。

创建非物化、非空列

Adaptive Server 15.7 版能让您创建非物化、非空列。

非物化列是以虚拟方式存在的，并不以物理方式存储在行中。非物化列的使用和任何其它列的使用相同，可以在 SQL 查询中选择、更新和引用它们，或者将它们用作索引键。

Adaptive Server 对待非物化列和对待空列的方式相似：如果某个列不是以物理方式存在于行中，Adaptive Server 会提供缺省值。可空列的缺省值为空，但非物化列的缺省值是用户定义的非空值。

请参见《Transact-SQL 用户指南》中的“添加、更改、传输和删除数据”。

共享行内缺省值

Adaptive Server 15.7 支持在同一数据库内的不同表之间共享行内缺省值。

在创建新的行内缺省值之前，Adaptive Server 会在属于同一用户的数据库中寻找具有相同值的现有可共享行内缺省值。如果 Adaptive Server 找到现有可共享缺省值，它会将该对象绑定到列上，而不是创建新的缺省值。但如果 Adaptive Server 找不到现有可共享行内缺省值，它就会创建新的缺省值。

Adaptive Server 无法在 tempdb 中共享行内缺省值。

请参见《Transact-SQL 用户指南》中的“为数据定义缺省值和规则”。

保留监控数据

Adaptive Server 15.7 版不将某些对象的描述符存储在元数据高速缓存中。它会保留描述符中存储的监控数据，从而提高查询性能。

元数据高速缓存是有限的资源，可以保存有限数量的对象描述符。向高速缓存中添加描述符可能会导致其它描述符从高速缓存中被刷新。

当您运行以下函数时，Adaptive Server 会放弃尚未在高速缓存中的对象的描述符，而不是消耗资源：

- data_pages
- used_pages
- reserved_pages

- `object_id`
- `row_count`
- `datachange`
- `derived_stat`

请参见《参考手册：构件块》。

分析动态参数

Adaptive Server 15.7 版能让您在运行查询之前先分析动态参数（用问号指示），从而帮助您避免生成低效率的查询计划。

使用以下项分析动态参数：

- `@@lwpid` 全局变量 - 返回与动态 SQL prepare 语句对应的最近准备的轻量过程的对象 ID。
- `@@plwpid` 全局变量 - 返回与动态 SQL prepare 语句对应的下一个最近准备的轻量过程的对象 ID。
- `show_dynamic_params_in_xml` - 显示有关动态 SQL 语句中的参数的信息。

通过将 `@@plwpid` 提供的值用作 `show_dynamic_params_in_xml object_id` 参数的值，Adaptive Server 可显示有关查询中的动态参数的信息。继续细化查询计划，直到确定能提供最佳查询计划的参数为止。

请参见 Performance and Tuning Series: Query Processing and Abstract Plans（《性能和调优系列：查询处理和抽象计划》）中的“显示查询优化策略和评估值”。

监控锁超时

Adaptive Server 15.7 版能让您监控锁超时。

Adaptive Server 15.7 版添加了以下有关跟踪锁的信息：

- `monLockTimeouts` 监控表提供有关超时锁请求的信息，如锁类型、所有者、锁状态等。请参见《参考手册：表》
- 这些参数将 Adaptive Server 配置为收集锁等待超时信息并使其可供 `monLockTimeout` 表使用：
 - `lock timeout pipe active`
 - `lock timeout pipe max messages`

请参见《系统管理指南第一卷》中的“设置配置参数”。

截断尾随零

Adaptive Server 15.7 版中包括 **disable varbinary truncation** 配置参数，用于启用或禁用从 `varbinary` 和 `binary` 空数据中截断尾随零的功能。

缺省情况下，**disable varbinary truncation** 对于服务器为关闭。

请参见《Transact-SQL 用户指南》中的“添加、更改、传输和删除数据”。

完全可恢复的 DDL

Adaptive Server 15.7 版能让您使用 **dump transaction** 来完全恢复较低版本的 Adaptive Server 以最少日志记录模式记录的操作。

以下可恢复操作可通过 **dump transaction** 恢复：

- **select into**
- 需要移动数据的 **alter table** 命令
- **reorg rebuild**

在 `master` 数据库中运行 **sp_dboption** 可完全记录缺省情况下以最少日志记录模式记录的命令。

请参见《参考手册：命令》。

使用 **merge** 将行从源表传送到目标表

Adaptive Server 15.7 引入了 **merge** 命令，能让您将行从源表传送到目标表

请参见《参考手册：命令》。

使用 **sp_showoptstats** 查看统计信息和直方图

sp_showoptstats 能让您提取并显示（XML 文档中）系统表（如 `systabstats` 和 `sysstatistics`）中的多种类型数据对象的统计信息和直方图。

请参见《参考手册：过程》，以及 **Performance and Tuning Series: Improving Performance with Statistical Analysis**（《性能和调优系统：利用统计分析改进性能》）中的“Statistics Tables and Displaying Statistics with `optdiag`”（统计信息表和使用 `optdiag` 显示统计信息）。

对游标所做的更改

Adaptive Server 15.7 版中包括对游标锁、游标管理事务的方式以及它声明游标语句的方式所做的更改。

在游标关闭时释放游标锁

Adaptive Server 15.7 中包括 **declare cursor ... release_locks_on_close** 选项，用以在游标关闭时在隔离级别 2 和 3 释放游标锁，即使事务处于活动状态也是如此。

请参见《参考手册：命令》，以及《Transact-SQL 用户指南》中的“游标：访问数据”。

增强了对游标的事务支持

Adaptive Server 15.7 和更高版本更改了游标支持事务的方式：

Adaptive Server：

- 不会在您提交事务时自动关闭用 **for update** 声明的已打开游标。要在提交事务时关闭只读游标，可设置 **close on end tran** 选项
- 支持在提交事务后对打开的游标执行 **fetch** 操作

请参见《参考手册：命令》，以及《Transact-SQL 用户指南》中的“游标：访问数据”。

监控游标语句

Adaptive Server 15.7 版根据 `monCachedStatement` 监控表中的信息来监控游标语句。

例如，虽然您声明了 `new_cursor` 游标（由“`sq0267364184_1108036110ss`”指定），但 `sp_cursorinfo` 不显示其计划：

```
declare new_cursor cursor for select id from sysroles
go
sp_cursorinfo
go
Cursor name 'new_cursor' is declared on procedure
'*sq0267364184_1108036110ss*'
The cursor is declared as NON-SCROLLABLE cursor.
The cursor id is 983044.
The cursor has been successfully opened 0 times.
The cursor will remain open when a transaction is committed or rolled
back.
The number of rows returned for each FETCH is 1.
The cursor is updatable.
This cursor is declared on a stored procedure. It is presently using
'860'
bytes. However, the memory usage will increase when the cursor is
```

```
opened because  
the query plan will be associated with the cursor at that time.
```

Adaptive Server 会在您打开游标时编译它们。

使用 **enable functionality group** 配置参数可启用和禁用监控游标语句。

请参见《Transact-SQL 用户指南》中的“游标：访问数据”。

嵌套 **select** 语句增强

Adaptive Server 15.7 扩展了星号 (*) 的功能。

在 Adaptive Server 15.7 和更高版本中，可以在不是 **exists** 子查询的嵌套 **select** 语句中使用星号，条件是星号：

- 是 **select** 语句中的唯一一项
- 对于嵌套查询可解析成单个表列

此外，还可以：

- 通过使用 *qualifier.** 格式（其中 *qualifier* 是 **from** 子句中的一个表）将嵌套查询中的选定列限制为仅属于特定表的列。
- 在包括 **group by** 子句的嵌套查询中使用星号。

当星号针对嵌套查询解析为单个表列时，该查询等效于显式使用单个表列。

请参见《Transact-SQL 用户指南》中的“查询：从表中选择数据”。

对链式事务中的命令和系统过程所做的更改

Adaptive Server 15.7 版能让某些系统过程在使用链式事务模式的会话中运行。

- 以下系统过程可以在没有打开的事务时在使用链式事务模式的会话中运行：
 - **sp_configure**
 - **sp_engine**
 - **sp_rename**
- 以下系统过程可以在您使用 **sp_procxmode** 将事务模式更改为 **anymode** 后在使用链式事务的会话中运行：
 - **sp_addengine**
 - **sp_dropengine**
 - **sp_showplan**
 - **sp_sjobcontrol**
 - **sp_sjobcmd**
 - **sp_sjobcreate**

- **sp_sjbdrop** 可以在使用链式事务模式的会话中运行，但如果您在事务打开期间执行它，则会失败。

当您执行存储过程时，Adaptive Server 会在没有打开的事务时隐式地提交这些存储过程执行的更改，因此您无需发出 **commit** 或 **rollback**。

如果在您发出以下命令时存在打开的事务：

- **sp_rename**、**sp_configure**、**sp_engine**、**sp_addengine** 或 **sp_dropengine** - 过程会因错误 17260 而失败，因为它们无法在事务内运行。
- **sp_sjobcontrol**、**sp_sjobcmd**、**sp_sjobcreate**、**sp_sjbdrop** 或 **sp_showplan** - Adaptive Server 会让事务在过程执行后仍然打开。必须针对整个事务显式地发出 **commit** 或 **rollback**。如果过程在执行过程中收到错误，它们会仅回退过程内执行的操作，而不回退过程执行之前所执行的操作，即使操作是在同一事务内执行的也是如此。

使用 **set chained {on | off}** 可为会话设置链式模式。

请参见《参考手册：命令》和《参考手册：过程》。

扩展了可变长度的行

Adaptive Server 15.7 版重新定义了仅数据锁定 (DOL) 列，从而可使用多达 32767 字节的行偏移。必须为 Adaptive Server 配置 16K 的逻辑页大小，才能创建宽的、可变长度的 DOL 行。

缺省情况下，Adaptive Server 不使用宽的、可变长度的 DOL 行。可使用以下命令启用宽的、可变长度的 DOL 行：

```
sp_dboption database_name, 'allow wide dol rows', true
```

请参见 Performance and Tuning Series: Physical Database Tuning（《性能和调优系列：物理数据库调优》）中的“数据存储”。

对 like 模式匹配所做的更改

Adaptive Server 15.7 版能让您逐个处理 like 模式匹配算法中的中括号。

例如，要在较低版本的 Adaptive Server 中将某个行与 '[XX]' 匹配，需要使用：

```
select * from t1 where f1 like '[]XX[]'
```

但在 Adaptive Server 15.7 中，您还可以使用：

```
select * from t1 where f1 like '[][]XX'
```

对带引号的标识符所做的更改

在 Adaptive Server 15.7 和更高版本中，可以对表、视图、列名称、索引名称和系统过程参数使用带引号的标识符。

在低于 15.7 的版本中，Adaptive Server 将“ident”视为一个使用非字母数字字符且以双引号（带引号的标识符）或中括号分隔的标识符。这些标识符只能用于表、视图和列名称。

请参见《参考手册：块》。

允许 Unicode 非字符

在 Adaptive Server 15.7 版中，作为 **enable functionality group** 成员的 **enable permissive unicode** 配置参数能让您忽略 Unicode 非字符。

当您启用此功能时，在以下内容中将不会检测到 Unicode 非字符：

- 参数
 - 显示为 `univarchar` 和 `unitext (UTF-16)` 数据类型
 - 显示为 `varchar` 和 `text (UTF-8)` 数据类型
 - 作为动态 SQL 语句的参数
 - 作为参数化语言语句的输入
 - 作为参数化语言语句的输入
- 当服务器的字符集为 UTF-8 时的字符串文字
- 转义字符串文字（带有前缀 `U&`，无论服务器的字符集如何）
- `unichar (UTF-16)` 和 `varchar (UTF-8)` 之间的转换过程（任一方向）

此外，Unicode 非字符在比较等简单表达式中是可接受的，在这些表达式中，它们的排序比合法 Unicode 字符高。

在低于 15.7 的 Adaptive Server 版本中，缺省字符集为 `utf-8` 的 `unichar`、`univarchar`、`unitext`、`char`、`varchar` 和 `text` 数据类型不接受 Unicode 非字符（为内部使用而永久保留的码点）。

请参见《系统管理指南第一卷》中的“设置配置参数”和“配置客户端/服务器字符集转换”。

降低查询处理延迟

Adaptive Server 15.7 中的查询处理层能让多个客户端连接重复使用或共享动态 SQL 轻量过程 (LWP)。

Adaptive Server 使用语句高速缓存来存储转换为 LWP 的动态 SQL 语句。由于语句高速缓存存储在所有连接当中，因此，动态 SQL 语句可以在连接当中重复使用。以下语句不被高速缓存：

- **select into** 语句。
- 所有值均为实际值且无参数的插入值语句。
- 不引用任何表的查询。
- 包含多个 SQL 语句的各个 prepared 语句。例如：

```
statement.prepare( 'insert t1 values (1) insert  
t2 values (3) ' );
```

- 导致 instead-of 触发器引发的语句

使用 **streamlined dynamic SQL** 或 **enable functionality group** 配置参数可启用此功能。

请参见 Performance and Tuning Series: Basics (《性能和调优系列：基础知识》) 中的“内存使用和性能”以及《系统管理指南第一卷》中的“设置配置参数”。

sybdiag 实用程序

Adaptive Server 15.7 添加了 **sybdiag** 实用程序，这是一个基于 Java 的工具，用于收集综合的 Adaptive Server 配置和环境数据。Sybase 技术支持部门使用此信息来诊断服务器问题，从而加快客户案例处理。

sybdiag 连接到 Adaptive Server 并执行系统过程（如 **sp_configure**）和对表（如 **monLicense**）的查询。它通过执行 **ps**、**vmstat** 和 **netstat** 等命令来收集操作系统和平台诊断信息。

sybdiag 生成 .zip 输出文件，其中包含可以解压缩并在 Web 浏览器中查询的 HTML 和数据文件。收集的信息包括操作系统和环境数据、Adaptive Server 配置和监控数据，以及 Adaptive Server 文件和脚本。

sybdiag 不收集 Adaptive Server 或操作系统的登录、口令信息或用户列表，不收集应用程序数据库表中的信息。

请参见《实用程序指南》。

优化程序诊断实用程序

Adaptive Server 15.7 版中包括 **sp_opt_querystats** 系统过程，能让您分析 Adaptive Server 优化程序生成的查询计划以及影响其查询计划选择的因素。

此分析可帮助确定查询或执行环境中的元素是否影响 Adaptive Server 执行查询的方式以及它的性能。无需运行选定的查询即可执行分析。

sp_opt_querystats 提供下列信息：

- 由 **showplan** 生成的查询计划
- 启用的跟踪标志和开关
- 由 **set statistics io** 生成的查询的 I/O 活动
- 针对查询所涉及的任何表找到的缺少的统计信息
- 由优化程序计算的估计计划开销
- 由优化程序计算的最终计划和开销估计值
- 查询的抽象计划
- 执行结果集后的查询结果（例如，如果 **noexec** 不是 on）
- 由 **set option show** 生成的查询的逻辑运算符树
- 由 **set statistics time** 生成的查询执行时间
- 执行查询后，由 **set statistics time** 生成的查询执行时间

必须安装并配置 Job Schedule，才能运行 **sp_opt_querystats**。

请参见 [Performance and Tuning Series: Query Processing and Abstract Plans](#)（《性能和调优系列：查询处理和抽象计划》）中的“Controlling Optimization”（控制优化）。

Adaptive Server 15.7 版中的系统更改

Adaptive Server 15.7 中包括对命令、函数、系统过程、配置参数、系统表、监控表和全局变量所做的更改。

命令

Adaptive Server 15.7 中包含新的和更改的命令。

表 1. 新命令

命令	说明
alter login	更改登录帐户的属性
alter login profile	更改登录配置文件的属性
alter...modify owner	将数据库对象的所有权从一个所有者转交给另一个所有者
alter thread pool	改变线程池
create login	创建登录帐户；指定口令、帐户的登录配置文件，以及为帐户分配的用户提供的参数
create login profile	用指定的属性创建登录配置文件
create thread pool	创建用户定义的线程池
deallocate locator	删除内存中存储的大对象 (LOB) 并使其 LOB 定位符无效
drop login	删除登录帐户或帐户列表
drop login profile	删除登录配置文件或登录配置文件列表
drop thread pool	删除用户定义的池
merge	将源表中的行传送到目标表中
select for update	以排它方式锁定行以防执行同一事务内的后续更新
truncate lob	将 LOB 截断成指定长度

表 2. 已更改的命令

命令	更改
alter database 更改	<ul style="list-style-type: none"> • 能让您在数据库级更改压缩设置 • alter database .. inrow_LOB_length - 能让您更改数据库范围的行内 LOB 列的长度 • alter database ... log off - 删除不需要的数据库日志部分, 能让您缩减日志空间以及无需重新创建数据库即可释放存储空间
alter encryption key	<ul style="list-style-type: none"> • master 和 dual master - 指示您正在更改主加密密钥或双主加密密钥 • master key - 指示您正在更改具有主密钥的加密密钥 • [no] dual_control - 指示新密钥是否是使用双控制加密的 • for recovery - 指示是否在口令丢失的情况下使用密钥副本恢复主密钥 • for automatic_startup - 指示在服务器启动后是否使用密钥副本访问主密钥或双主密钥 • regenerate key - 将主密钥或双主密钥的原始密钥值替换为新的原始密钥, 并重新加密所有由主密钥或双主密钥加密的列加密密钥
alter table	<ul style="list-style-type: none"> • 能让您更改表、列和分区的压缩属性。 • alter table ... not materialized - 指示您正在创建非物化列 • alter table ... add lob-colname - 能让您将新添加的可空 LOB 列定义为在行内, 并指定其长度 • alter table ... modify lob-colname - 能让您将现有 LOB 列从在行外改为在行内
并置运算符	+ 和 Transact-SQL 运算符在并置运算中将 LOB 定位符接受为表达式。涉及一个或多个定位符的并置运算的结果是具有和输入定位符引用的数据类型相同类型的新 LOB 定位符。
create database	<ul style="list-style-type: none"> • compression = 指示要对新创建的表或分区应用的压缩级别。 • lob_compression = value - 确定新创建的表的压缩级别。选择 off 意味着表不使用 LOB 压缩。 • inrow_lob_length = value - 指定字节数。inrow_lob_length 的有效值范围是 0 到数据库的逻辑页大小。
create encryption key	<ul style="list-style-type: none"> • master 和 dual master - 指示您正在创建主加密密钥或双主加密密钥 • passwd system_encr_passwd master key - 指示您正在使用系统加密口令或口令的主密钥 • [no] dual_control - 指示新密钥是否是使用双控制加密的

命令	更改
create table ... [in row [[length]] off row]	能让您创建压缩表 create table 能让您指定 LOB 列中的数据保留在行内，而不是存储在行外。
declare cursor ... [release_ locks_on_ close]	能让您配置每个游标锁释放行为，以便在关闭游标时能释放共享锁，即使事务处于活动状态也是如此。
drop encryption key	[dual] master - 指示您正在删除主密钥或双主密钥
dump database ... with shrink_log	能让您删除数据库结尾的所有空洞，无论数据库是否处于转储序列中都是如此。
where 子句中的 like 子句	对于 <i>variables</i> 表达式和 <i>match_string</i> ， where 子句接受 text 和 unitext LOB 定位符，但不接受 image LOB 定位符。
select into ... [in row [[length]] off row]	设置或更改目标表中的文本列的行内特性。如果您不指定长度，Adaptive Server 会使用所配置的缺省行内长度。
set	set 添加了 <ul style="list-style-type: none"> • send_locator [on off] - 指定 Adaptive Server 在发送到客户端的结果集中是发送 LOB 还是发送引用 LOB 的定位符。 • cis_rpc_handling {on off} - 使 CIS 远程过程调用 (RPC) 处理机制成为共享磁盘集群 (SDC) 处理的缺省机制 • encryption passwd <char_literal> for key [dual] master - 设置主密钥或双主密钥的口令
where 子句经扩展可支持 LOB	select 、 insert 、 update 和 delete 语句中的 where 子句可以包括空大对象 (LOB) 的条件。

请参见《参考手册：命令》。

函数

Adaptive Server 15.7 中包含新的和更改的函数。

表 3. 新函数

函数	说明
dol_downgrade_check	返回指定数据库中含有宽于 8191 字节的可变长度列的仅数据锁定 (DOL) 表的数量
create_locator	显式创建指定大对象 (LOB) 的定位符
locator_literal	将二进制值标识为定位符文字
locator_valid	确定 LOB 定位符是否有效
lprofile_id	返回指定登录配置文件名称的登录配置文件 ID，或者与当前登录或指定登录名关联的登录配置文件的登录配置文件 ID
lprofile_name	返回指定登录配置文件 ID 的登录配置文件名称，或者与当前登录或指定登录 <code>suid</code> 关联的登录配置文件的登录配置文件名称
return_lob	取消引用定位符，并返回该定位符引用的 LOB
setdata	覆盖部分或全部 LOB
show_cached_plan_in_xml	以 XML 格式显示语句高速缓存中的查询的执行查询计划
show_dynamic_params_in_xml	以 XML 格式返回查询的文本

表 4. 已更改的函数

函数	说明
charindex	charindex 添加了对 <code>text_locator</code> 、 <code>unitext_locator</code> 和 <code>image_locator</code> LOB 定位符数据类型以及 <code>start</code> 选项的支持。
charlength	charlength 支持 <code>text_locator</code> 和 <code>unitext_locator</code> 数据类型。
datalength	datalength 接受 <code>text_locator</code> 、 <code>unitext_locator</code> 和 <code>image_locator</code> 数据类型。
patindex	patindex 接受 <code>text_locator</code> 和 <code>unitext_locator</code> 数据类型。

函数	说明
show_cached_plan_in_xml	show_cached_plan_in_xml 扩展了 statement_id 参数的范围，使其可接受引用任何轻量过程的对象 ID，而不仅仅是语句高速缓存中的对象 ID。
str	str 函数的小数参数得到了扩展，可以支持将具有字符或数值的输出填充到指定长度。
textptr	<p>由于行内/行外 LOB 功能可以拆分或缩减具有聚簇索引的所有页锁定表的数据页，从而导致数据行（包括位于这些数据行中的行内 LOB 列）移动到另外的页，因此，这种行内 LOB 列在拆分或缩减操作之前的 textptr 文本指针值不同于同样的列在这种操作之后的值。</p> <p>对于 Adaptive Server 15.7 和更高版本，针对位于仅数据锁定数据行中的行内 LOB 列返回的、由行转发的 textptr 值在转发后仍然有效。</p>
textvalid	可以在返回的文本指针上对行内和行外 LOB 使用 textvalid ，如果文本指针指向有效的 LOB 列，则返回 1，如果 LOB 列无效，则返回 0。

请参见《参考手册：构件块》。

系统存储过程

Adaptive Server 15.7 中包含新的和更改的系统过程。

表 5. 新增的系统存储过程

系统存储过程	说明
sp_merge_dup_inline_default	删除现有重复行内缺省对象，将唯一行内缺省对象转换为可共享行内缺省对象
sp_opt_querystats	返回选定查询的性能分析
sp_securityprofile	列出与登录配置文件关联的属性或绑定
sp_showoptstats	提取并显示系统表（如 systabstats 和 sysstatistics ）中多个数据对象的统计信息和直方图

表 6. 已更改的系统存储过程

系统存储过程	说明
sp_dboption	<ul style="list-style-type: none"> • enforce dump tran sequence - 防止会禁止后续转储事务的操作 • allow wide rows - 将数据库配置为允许宽的、可变长度的仅数据锁定 (DOL) 行 • full logging for all - 完全记录缺省情况以最少日志记录模式记录的命令 (select into、alter table 和 reorg rebuild)
sp_displaylogin	<p>显示</p> <ul style="list-style-type: none"> • 与登录帐户关联的登录配置文件名称。 • 缺省登录配置文件的名称 (如果没有直接与登录帐户关联的登录配置文件, 但有缺省登录配置文件) <p>登录名会覆盖 sp_addlogin、sp_modifylogin default database, default language, authenticate with 和 login script 参数。</p> <p>如果忽略登录配置文件, 或者没有直接或通过缺省登录配置文件与登录帐户关联的登录配置文件, sp_displaylogin 会按低于 15.7 的版本的格式显示信息。</p>
sp_displayroles	<ul style="list-style-type: none"> • 当在当前登录名下运行时, sp_displayroles 会显示向与其关联的登录配置文件授予的角色。sp_displayroles 需要有 sso_role 才能查看与其它登录配置文件关联的角色。 • 显示通过关联的登录配置文件向登录名授予的角色。输出中的 Grantee 列指示登录配置文件名称 (如果适用)。只有在登录名具有与其授予的角色关联的登录配置文件时, sp_displayroles 才会显示 Grantee 列。 • 显示锁定角色的日期、锁定原因, 以及锁定角色的登录 ID。对于受口令保护的角色, sp_displayroles 显示角色口令加密版本。
sp_encryption	<ul style="list-style-type: none"> • 当由 SSO、密钥管理者或 DBO 运行时, 会报告受双控制保护的密钥 • sp_encryption helpkey, master 和 sp_encryption helpkey, 'dual master' 报告有关主密钥和双主密钥的信息, 包括是否存在为自动启动而加密的副本以及是否存在恢复副本 • mkey_startup_file [, {<new_path> default_location null}] [, {sync_with_mem sync_with_qrm}] - 显示或设置主密钥启动文件名称和路径 • downgrade_kek_size [, { "true" "false" }] - 显示或设置服务器的 downgrade_kek_size 配置
sp_help	<p>显示列级、表级和分区级的压缩设置。显示列级和表级的行内 LOB 设置。</p>

系统存储过程	说明
sp_helpconstraint	进行了更新，显示有关可共享行内缺省对象的信息
sp_helprotect	<ul style="list-style-type: none"> 接受“master key”和“dual master key”作为对象名称 接受“Set Encryption Passwd”作为有效的权限名称 显示双主密钥权限和主密钥权限
sp_helpuser	<i>display_object</i> 列出当前表中的所有对象以及由 <i>name_in_db</i> 拥有的用户定义的数据类型
sp_locklogin	被排除的登录帐户不再由于不活动而被锁定。
sp_passwordpolicy	<ul style="list-style-type: none"> keypair regeneration period - 指定开始生成第一个密钥对的日期和时间以及后来重新生成密钥对的频率 keypair error retry wait/count - 指定可以针对尝试失败后重新生成密钥对而设置的多种配置
sp_serveroption	更改以下选项的定义： <ul style="list-style-type: none"> use message confidentiality - 为所有使用 Kerberos 鉴定与远程服务器建立的连接设置消息保密性 use message integrity - 为所有使用 Kerberos 鉴定与远程服务器建立的连接设置消息完整性

请参见《参考手册：过程》。

配置参数

Adaptive Server 15.7 引入了新的配置参数。

配置参数	说明
automatic master key access	确定 Adaptive Server 在无人值守启动模式下运行
capture compression statistics	启用 <i>monTableCompression</i> 监控表以开始捕获压缩统计信息
column default cache size	确定 Adaptive Server 为提供非物化列的缺省对象而必须在内存中保留的高速缓存大小
compression info pool size	确定用于压缩的内存池的大小

配置参数	说明
disable varbinary truncation	控制 Adaptive Server 是否在 varbinary 或 binary 空数据的结尾包括尾随零
enable console logging	启用后，Adaptive Server 会在启动后向主控台发送独立于错误日志的消息
enable functionality group	在 Adaptive Server 15.7 版和更高版本中启用或禁用以下功能： <ul style="list-style-type: none"> • 可共享行内缺省值 • 选择以进行更新 • Like 模式匹配 • 带引号的标识符 • Unicode 非字符 • 监控游标语句 • 降低查询处理延迟
enable hp posix async i/o	在 HP-UX 11.31 和更高版本上启用异步 I/O
kernel mode	确定 Adaptive Server 内核使用的模式（线程化或进程）
kernel resource memory	确定向所有线程池和其它内核资源分配内存的内核资源内存池的大小（以 2K 页为单位）
lock timeout pipe active	控制 Adaptive Server 是否收集锁超时消息
lock timeout pipe max messages	确定 Adaptive Server 存储的锁超时消息数，以及它为任务分配的内存量
number of disk tasks	控制专用于轮询和完成磁盘 I/O 的任务数
number of network tasks	控制专用于轮询和完成网络 I/O 的任务数

请参见《系统管理指南 第一卷》

系统表

Adaptive Server 15.7 中包含新的和更改的系统表。

表 7. 已更改的系统表

系统表	添加的列	说明
sysattributes	<ul style="list-style-type: none"> object_cinfo2 object_datetime 	<ul style="list-style-type: none"> 提供对象的字符说明 提供对象的日期和时间 sysattributes 添加了 SPObject_type (用于存储与 RSA 密钥对重新生成有关的选项) 和 LR object_type (用于存储与登录配置文件有关的选项)
sysdatabases		向 status4 列中添加了状态位, 用以指示数据库范围的压缩设置
sysoptions	number	以整数形式列出开关 ID
sysobjects	lobcomp_lvl	<ul style="list-style-type: none"> 向 status3 列中添加了状态位, 用以指示数据库范围的压缩设置 lobcomp_lvl - 为大对象定义的列的压缩级别
syscolumns	inrowlen	<ul style="list-style-type: none"> 向 status2 列中添加了状态位, 用以指示是否显式将列定义为压缩 inrowlen - 一个可空列, 用于存储创建为行内的 LOB 列的用户指定的或派生的行内长度 lobcomp_lvl - 为大对象定义的列的压缩级别
syslogins	crsuid	登录名或登录配置文件的创建者的服务器用户 ID
syslogins	lpid	登录配置文件 ID
sysssrvroles	<ul style="list-style-type: none"> lockdate lockreason locksuid 	<ul style="list-style-type: none"> 锁定角色的日期和时间 锁定角色的原因 锁定角色的用户的 ID

Adaptive Server 15.7 版中的系统更改

系统表	添加的列	说明
syssservers	srvprincipal	指定远程服务器 Kerberos 主管名
syscomments	<ul style="list-style-type: none">syb_syscommkey_dddddd	

Adaptive Server 15.7 版添加了以下系统表，它们是主数据库的视图，提供有关数据高速缓存和池的配置的信息。

系统表	说明
syscacheinfo	提供有关数据高速缓存的信息。
syspoolinfo	提供有关高速缓存池的信息。
syscachepoolinfo	为每个数据高速缓存池提供一行内容，用以包括数据高速缓存的信息。此视图是 <i>syscacheinfo</i> 和 <i>syspoolinfo</i> 视图之间的连接。

请参见《参考手册：表》

实用程序

Adaptive Server 15.7 中包含新的和更改的实用程序。

新增实用程序

命令	说明
sybdiag	sybdiag 是一个基于 Java 的工具，用于收集综合的 Adaptive Server 配置和环境数据。Sybase 技术支持部门使用此信息来诊断服务器问题，从而加快客户案例处理。

更改的实用程序

命令	说明
sybperf	15.7 版的 sybperf 公开一组 Adaptive Server 计数器，更加有助于监控 Adaptive Server 性能。

监控表更改

Adaptive Server 15.7 版中包含新的和更改的监控表。

新的监控表

Adaptive Server 15.7 版中包括新的监控表。

命令	说明
monDeviceSpaceUsage	提供有关在其上分配数据库设备的文件系统的信息。空间信息仅可用于文件系统设备。对于在裸设备上分配的数据库设备，文件系统大小和可用空间值为空。
monLockTimeout	提供有关锁超时请求的信息

请参见《参考手册：表》。

对监控表的更改

Adaptive Server 15.7 版中包括对一些监控表所做的更改。

对 *monCachePool* 所做的更改

监控表	说明
LogicalReads	从池中读取的缓冲区数
PhysicalWrites	在该池中针对数据执行的写入操作次数（一个写入操作可能包括多页）
APFReads	向该池中装载页的 APF 读取操作次数。
APFPercentage	为该池配置的异步预取限制
WashSize	内存池的清洗大小（以千字节为单位）

对 *monCachedProcedures* 所做的更改

监控表	说明
ExecutionCount	Adaptive Server 自高速缓存以来执行存储过程计划或树的次数
CPUTime	所用 CPU 时间总计（以毫秒为单位）
ExecutionTime	Adaptive Server 用来执行存储过程计划或树所花费的时间总计（以毫秒为单位）
PhysicalReads	所执行的物理读取次数
LogicalReads	读取的页数
PhysicalWrites	所执行的物理写入次数

监控表	说明
PagesWritten	写入的页数

对 *monCachedStatement* 所做的更改

监控表	说明
OptimizationGoal	语句高速缓存中存储的优化目标
OptimizerLevel	语句高速缓存中存储的优化程序级别

对 *monCachedProcedures* 所做的更改

监控表	说明
Status	高速缓存的状态
Type	高速缓存的类型
CacheSize	高速缓存的总大小（以千字节为单位）
ReplacementStrategy	高速缓存替换策略
APFReads	该数据高速缓存的异步预取 (APF) 读取次数
Overhead	高速缓存开销

添加到 *monDeadLock* 中的列

列	说明
HeldClientApplName	由持有锁的应用程序设置的 <i>clientapplname</i> 属性的值
HeldClientName	由持有锁的应用程序设置的 <i>clientname</i> 属性的值
HeldClientHostName	由持有锁的应用程序设置的 <i>clienthostname</i> 属性的值
HeldHostName	执行持有锁的查询的应用程序在其上运行的主机的名称
HeldNumLocks	持有 <i>spid</i> 当前持有的锁的数量
HeldProcDBName	在发生死锁时存储过程在其中执行阻塞进程的数据库的名称（如果适用）
HeldProcedureName	在发生死锁时阻塞进程执行的存储过程的名称（如果适用）
HeldProgramName	运行持有锁的进程的程序的名称
HeldStmtNumber	持有锁的 SQL 语句的 SQL 批处理中的语句数

列	说明
ObjectDBName	数据库的名称
ObjectID	对象的唯一标识符
WaitApplName	等待锁的应用程序的名称
WaitBatchID	发生锁超时，等待锁的进程所执行的 SQL 批处理的标识符
WaitClientApplName	由等待锁的应用程序设置的 <i>clientapplname</i> 属性的值
WaitClientHostName	由等待锁的应用程序设置的 <i>clienthostname</i> 属性的值
WaitClientName	由等待锁的应用程序设置的 <i>clientname</i> 属性的值
WaitCommand	进程被阻塞后超时时所执行的进程或命令的类别
WaitContextID	等待锁的进程在它被另一进程阻塞时的唯一上下文标识符
WaitHostName	运行等待锁的进程的主机的名称
WaitLineNumber	等待锁的 SQL 批处理或存储过程中的 SQL 语句行数
WaitProcDBID	等待锁的存储过程所在的数据库的唯一标识符（如果适用）
WaitProcDBName	等待锁的存储过程所在的数据库的名称（如果适用）
WaitProcedureID	等待锁的存储过程的 ID（如果适用）
WaitProcedureName	等待锁的存储过程的名称（如果适用）
WaitProgramName	运行进程的程序的名称
WaitStmtNumber	等待锁的 SQL 批处理中的行数
WaitTranName	在其中请求锁的事务的名称

对 *monErrorLog* 所做的更改

Adaptive Server 15.7 版和更高版本更改了 *monErrorLog.Severity* 列中的堆栈跟踪的值。较低版本使用 0 作为堆栈跟踪的值。在 Adaptive Server 15.7 版中，所有表示堆栈跟踪的行其 *Severity* 值都是 99。

对 *monLockTimeout* 所做的更改

监控表	说明
HeldProgramName	已从 <i>monLockTimeout</i> 中删除
WaitProgramName	已从 <i>monLockTimeout</i> 中删除

监控表	说明
HeldProcedureID	在发生超时时阻塞进程执行的存储过程的唯一对象标识符
WaitProcedureID	等待锁的存储过程的唯一对象标识符（如果适用）

对 monOpenObjectActivity 所做的更改

监控表	说明
SharedLockWaitTime	所有任务在等待共享锁时所花费的时间总计（以毫秒为单位）
ExclusiveLockWaitTime	所有任务在等待排它锁时所花费的时间总计（以毫秒为单位）
UpdateLockWaitTime	所有任务在等待更新锁时所花费的时间总计（以毫秒为单位）
ObjectCacheDate	指示对象添加到高速缓存中的日期和时间

对 monOpenPartitionActivity 所做的更改

监控表	说明
ObjectCacheDate	指示对象添加到高速缓存中的日期和时间

对 monProcess 所做的更改

监控表	说明
ProgramName	运行进程的程序的名称
HostName	启动进程的应用程序在其上运行的主机的名称
ClientName	由应用程序设置的 <i>clientname</i> 属性的值
ClientHostName	由应用程序设置的 <i>clienthostname</i> 属性的值
ClientApplName	由应用程序设置的 <i>clientapplname</i> 属性的值

对 monProcessActivity 所做的更改

监控表	说明
ProgramName	运行进程的程序的名称
HostName	执行查询的应用程序在其上运行的主机的名称

监控表	说明
Application	应用程序的名称
ClientName	由应用程序设置的 <i>clientname</i> 属性的值
ClientHostName	由应用程序设置的 <i>clienthostname</i> 属性的值
ClientApplName	由应用程序设置的 <i>clientapplname</i> 属性的值

对 *monProcessLookup* 所做的更改

监控表	说明
ProgramName	运行进程的程序的名称
ClientName	由应用程序设置的 <i>clientname</i> 属性的值
ClientHostName	由应用程序设置的 <i>clienthostname</i> 属性的值
ClientApplName	由应用程序设置的 <i>clientapplname</i> 属性的值

对 *monProcessProcedures* 所做的更改

监控表	说明
ExecutionCount	Adaptive Server 执行过程高速缓存中保存的此存储过程实例的次数
CPUTime	Adaptive Server 执行过程高速缓存中保存的此存储过程实例所花费的 CPU 时间（以毫秒为单位）
ExecutionTime	Adaptive Server 执行过程高速缓存中保存的此存储过程实例所花费的时间总计（以毫秒为单位）
PhysicalReads	过程高速缓存中保存的此存储过程实例所执行的物理读取次数
LogicalReads	过程高速缓存中保存的此存储过程实例所执行的逻辑读取次数
PhysicalWrites	过程高速缓存中保存的此存储过程实例所执行的物理写入次数
PagesWritten	过程高速缓存中保存的此存储过程实例所读取的页数

对 *monTableColumns* 的更改

monTableColumns 包括以下更改：

- Adaptive Server 15.7 版和更高版本中在 *monTableColumns* 的 Description 列中包括了列的测量单位。

Adaptive Server 15.7 版中的系统更改

- `monTableColumn` 添加了 `Label` 列（数据类型为 `varchar(50)`），其中包含列中提供的数据的简要说明。可以在应用程序用户界面中使用这些值，但不能在实际列名中使用它们。

对 `monTables`、`monTableColumns`、`monWaitEventInfo` 和 `monWaitClassInfo` 所做的更改

这些监控表添加了 `Language` 列（数据类型为 `varchar(30)`），能让您指定 Adaptive Server 返回 `Description` 列值和 `monTableColumns.Label` 列值时使用的语言。

请参见《参考手册：表》。

全局变量

Adaptive Server 15.7 版中包括新的全局变量。

命令	说明
<code>@@plwpid</code>	返回最近准备的轻量过程的对象 ID
<code>@@lwpid</code>	返回最近运行的轻量过程的对象 ID

请参见《参考手册：构件块》

Adaptive Server 15.5 版 Cluster Edition 中的新增功能

Adaptive Server® 15.5 Cluster Edition 引入了多个同时发生的故障切换、分布式事务管理、**mount** 和 **unmount** 命令，以及使用 **alter database** 来增加归档数据库空间的功能。

注意： Cluster Edition 当前不支持内存数据库、宽松持久性数据库、模板数据库或进行最少日志记录操作的 DML。

Adaptive Server 15.5 Cluster Edition 功能和平台表

此功能和平台表给出了 Adaptive Server 15.5 Cluster Edition 中针对所支持操作系统的功能可用性。“Y”表示该平台支持此功能。

Adaptive Server Cluster Edition 选项	HP-UX Itanium 64 位	IBM AIX 64 位	Linux Opteron 64 位	Solaris 64 位
安全和目录服务	Y	Y	Y	Y
CyberSafe Kerberos				Y
可插入鉴定模块	Y	Y	Y	Y
精细访问控制	Y	Y	Y	Y
LDAP 服务器目录	Y	Y	Y	Y
LDAP 用户鉴定	Y	Y	Y	Y
平台本机 Kerberos				Y
安全套接字层	Y	Y	Y	Y
MIT Kerberos	Y	Y	Y	Y
加密列，包括精细访问控制 (FGAC)	Y	Y	Y	Y
高可用性				

Adaptive Server Cluster Edition 选项	HP-UX Itanium 64 位	IBM AIX 64 位	Linux Opteron 64 位	Solaris 64 位
分区	Y	Y	Y	Y
内存数据库				
用于 Backup Server 的 Tivoli Storage Manager	Y	Y	Y	Y
Active Messaging	Y	Y	Y	Y
增强的全文本搜索 (EFTS)				
<i>基本 Adaptive Server 中包括的功能</i>				
跨平台转储和装载	Y	Y	Y	Y
Job Scheduler	Y	Y	Y	Y
本机 XML	Y	Y	Y	Y
IPv6	Y	Y	Y	Y
Java 选项	Y	Y	Y	Y
Web 服务	Y	Y	Y	Y
分布式事务管理	Y	Y	Y	Y
内容管理 (外部文件支持)	Y	Y	Y	Y
归档数据库访问	Y	Y	Y	Y

多个同时发生的故障切换

Adaptive Server Cluster Edition 15.5 版及更高版本支持处理多个同时发生的实例故障。

当单一集群视图内有多个实例出现故障时，可对多个同时发生的故障提供支持，但集群将保持联机状态，并提供与单一实例出现故障时同样的故障切换恢复功能。

故障的数量不能大于配置参数 **cluster redundancy level** 的值，该配置参数允许数据库管理员为集群设置可恢复的同时发生的实例故障的最大数量。

为归档数据库增加空间

Adaptive Server Cluster Edition 15.5 版及更高版本支持归档数据库。通常，在集群和非集群 Adaptive Server 中，对归档数据库的访问是相同的。在这两种情况下，当归档数据库空间不足时，都可使用 **alter database** 命令为归档数据库增加空间。

在集群 Adaptive Server 中，从更新归档数据库的节点运行 **alter database**。如果您从其它节点运行 **alter database**，Adaptive Server 就会输出错误消息，并标明实际更新归档数据库的节点的编号。

共享磁盘集群中的分布式事务管理

在 15.5 版及更高版本中，Adaptive Server 在其集群体系结构中支持分布式事务管理 (DTM)。

群集 Adaptive Server：

- 在用作资源管理器 (RM) 时完全符合 X/Open XA 协议，而不需要附加的服务，例如 XA-Server。
- 确保通过远程过程调用 (RPC) 和组件集成服务 (CIS) 更新 Adaptive Server 数据的所有事务的提交或回退保持一致。
- 可以由其它使用 Adaptive Server 事务协调 (ASTC) 机制的 Adaptive Server 安装所协调的分布式事务的一部分。
- 可以在多个使用 ASTC 机制的 Adaptive Server 安装之间协调分布式事务。

注意： Cluster Edition 不支持 Microsoft 分布式事务协调器 (MSDTC) 专有协议。

通常，分布式事务的用户界面在 Adaptive Server 的集群和非集群环境中是相同的。在非集群 Adaptive Server 上使用 DTM 的应用程序可以在集群 Adaptive Server 上使用同样的应用程序。请参见《使用 Adaptive Server 分布式事务管理功能》。

虽然 Cluster Edition 上的分布式事务的用户界面和非集群 Adaptive Server 的用户界面相同，但对集群中的分布式事务的支持必须考虑《集群用户指南》中讲述的特定于集群的问题。例如：

- 将集群用作资源管理器 (RM)
- 向非所有者实例发出的请求
- 处理实例故障
- 将事务协调与 ASTC 一起使用
- 连接迁移的影响
- 系统配置

Adaptive Server 15.5 Cluster Edition 的系统更改

Adaptive Server 15.5 Cluster Edition 支持对命令、监控表和配置参数进行更改。

已更改的命令

在 Adaptive Server Cluster Edition 15.5 版及更高版本中，您可以在共享磁盘集群中使用 **mount database** 和 **unmount database**。

如果在执行 **mount database** 或 **unmount database** 时某个实例出现故障，则此命令就可能中止。这种情况下，当实例故障切换恢复完成时，您必须重新发出 **mount database** 或 **unmount database**。

监控表

为 Adaptive Server 15.5 版添加的监控表

Adaptive Server Cluster Edition 15.5 版添加了以下监控表：

- **monTableTransfer** - 提供 Adaptive Server 活动内存中表的历史传输信息。
- **monInmemoryStorage** - 用于内存数据库。仅供内部使用。

Cluster Edition 15.0.1 版及更高版本包含用于收集表统计信息的监控表。以下是 15.5 版中包含的监控表：

- **monCIPC** - 提供从当前实例或所有实例查看的有关集群内部所有消息传送的摘要数字。
- **monCIPCEndpoints** - 提供详细摘要，为集群实例中的每个子系统提供通信量数据。
- **monCIPCLinks** - 监控集群中实例间链接的状态。
- **monCIPCMesh** - 按实例提供从当前实例到集群中所有其它实例的连接网络的摘要数字。
- **monCLMObjectActivity** - 收集集群锁信息。
- **monClusterCacheManager** - 存储有关每个实例上运行的集群高速缓存管理器守护程序的诊断信息。**monClusterCacheManager** 按实例报告集群范围的信息。
- **monCMSFailover** - 跟踪集群成员资格服务 (CMS) 检测到故障的时间、获取新的集群视图、重新同步心跳、发布失败事件以及完成失败事件。每个实例都有一行。

- **monDBRecovery** – 包含集群中所有实例的行，并包含有助于恢复的每个数据库的行。
- **monDBRecoveryLRTypes** – 跟踪恢复期间查看的日志记录。每个日志记录类型在该表中都有对应的一行，恢复进程至少查看了相应类型的一条日志记录。
- **monFailoverRecovery** – 包含集群锁管理程序 (CLM)、数据库恢复和 CMS 模块的聚合故障切换恢复诊断信息。
- **monLogicalCluster** – 显示有关系统上当前配置的逻辑集群的信息。
- **monLogicalClusterAction** – 显示从启动到释放这些操作的时间内对本地集群执行的所有管理操作。
- **monLogicalClusterInstance** – 显示有关实例与逻辑集群之间的多对多关系的信息。
- **monLogicalClusterRoute** – 显示有关已配置的路由（应用程序、登录名和别名绑定）的信息。您不需要具有 **mon_role** 角色即可查询该监控表。
- **monPCM** – 跟踪集群中的对等协调模块 (PCM) 客户端活动（例如，发送和接收的片段数），且每个 PCM 客户端在该表中都有相应的一行。
- **monProcessMigration** – 显示有关当前迁移的连接的信息。
- **monSysLoad** – 按引擎提供趋势统计信息。您不需要具有 **mon_role** 角色即可查询该监控表。
- **monTempdbActivity** – 提供有关所有打开的本地临时数据库的统计信息，包括在 **tempdb** 配置模式下启动实例时的全局系统 **tempdb**。
- **monWorkload** – 根据负载配置文件显示每个实例上每个逻辑集群的工作负载得分。
- **monWorkloadPreview** – 提供有关负载配置文件在不启用时对工作负载得分的影响的估计值。**monWorkload** 为每个逻辑集群以及运行该逻辑集群的实例都设置了一行。
- **monWorkloadProfile** – 显示当前配置的工作负载配置文件。您不需要具有 **mon_role** 角色即可查询该监控表。
- **monWorkloadRaw** – 提供每个实例的原始工作负载信息。您不需要具有 **mon_role** 角色即可查询该监控表。

配置参数

Adaptive Server Cluster Edition 15.5 版及更高版本的新增配置参数

cluster redundancy level – 可同时出现故障（同时允许恢复以并发方式继续执行其它活动）的实例的最大数量。如果失败的实例数超过最大值，集群将会关闭。

函数

Adaptive Server Cluster Edition 15.5 版及更高版本的新增函数

- **xact_owner_instance** – 返回运行外部事务的实例或 0。

- **xact_conmigrate_check** – 确定连接能否处理外部事务。

Adaptive Server® 15.5 版中的新增功能

Adaptive Server 15.5 引入了内存数据库和宽松持久性数据库、针对 IBM Tivoli Storage Manager 的 Backup Server 支持、针对备份的更快压缩、存储过程的延迟名称解析、增量数据传输、对 FIPS 140-2 口令加密的支持，以及新的数据类型。

Adaptive Server 15.5 功能和平台表

此功能和平台表给出了 Adaptive Server 15.5 中针对所支持操作系统的功能可用性。“Y”表示该平台支持此功能。

Adaptive Server 选项	HP-UX Itanium 64 位	HP-UX PA Risc 64 位	IBM AIX 64 位	Linux on Power 64 位	Linux Opteron 64 位	Linux x86 32 位	Solaris 32 位	Solaris 64 位	Solaris Opteron 64 位	Windows Opteron X64	Windows x86 32 位
安全和目录服务	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
CyberSafe Kerberos							Y	Y			Y
可插入鉴定模块	Y		Y	Y	Y	Y	Y	Y	Y		
精细访问控制	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
LDAP 服务器目录	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
LDAP 用户鉴定	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
平台本机 Kerberos							Y	Y			
安全套接字层	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
MIT Kerberos	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y
加密列，包括精细访问控制 (FGAC)	Y	Y	Y		Y	Y	Y	Y	Y	Y	Y
高可用性	Y	Y	Y		Y	Y	Y	Y			Y
分区	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
内存数据库	Y	Y	Y	Y	Y			Y	Y	Y	

Adaptive Server 选项	HP-UX Itanium 64 位	HP-UX PA Risc 64 位	IBM AIX 64 位	Linux on Power 64 位	Linux Opteron 64 位	Linux x86 32 位	Solaris 32 位	Solaris 64 位	Solaris Opteron 64 位	Windows Opteron X64	Windows x86 32 位
用于 Backup Server 的 Tivoli Storage Manager	Y		Y		Y			Y	Y	Y	
Active Messaging	Y	Y	Y		Y	Y	Y	Y	Y	Y	Y
增强的全文本搜索 (EFTS)		Y	Y		Y	Y	Y	Y			Y
<i>基本 Adaptive Server 中包括的功能</i>											
跨平台转储和装载	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Job Scheduler	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
本机 XML	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
IPv6	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Java 选项	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Web 服务	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
分布式事务管理	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
内容管理 (外部文件支持)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
归档数据库访问	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

内存数据库和宽松持久性数据库

内存数据库和宽松持久性数据库提供增强的性能。

内存数据库完全在命名高速缓存中（即，在 Adaptive Server 内存空间中）运行，而不使用磁盘存储空间来存储数据或日志。因为内存数据库不需要 I/O，所以它的性能可以比传统的磁盘驻留式数据库好得多。内存数据库不是为恢复而设计的：它们的事务日志被写入高速缓存而不是磁盘，因而如果服务器重新启动，任何数据更改都将丢失。内存数据库为运行时回退和其它操作（例如，引发触发器、延迟模式更新、复制等）执行事务性日志记录。

磁盘驻留式数据库可执行写入到磁盘操作，并可确保保持原子性、一致性、完整性和持久性等事务性属性（统称为 **ACID** 属性）。持久性是指事务在提交后继续存在。传统的 **Adaptive Server** 数据库在提交事务时将其事务日志写入磁盘，从而以完全持久性方式进行操作。此数据库与定期写入磁盘的数据页一起来确保所有已提交的事务是持久性的。

内存数据库不将数据或日志写入磁盘，以牺牲对事务持久性的保证为代价来换取性能上的改进。如果发生数据库故障，内存数据库将无法恢复。如果您的应用程序要求在服务器发生故障或正常关闭后保证数据可恢复性，请考虑使用传统的 **Adaptive Server** 数据库。

通过支持宽松持久性，**Sybase** 使磁盘驻留式数据库具备内存数据库的性能优点。传统的磁盘驻留式数据库可保证从服务器故障中进行事务性恢复。宽松持久性数据库以牺牲已提交事务的完全持久性为代价，来提高事务性负载的运行时性能。

内存数据库和宽松持久性数据库的性能优势包括：

- 内存数据库完全在内存中运行，因此它不等待 I/O。
- 改善了缓冲区和用户日志高速缓存管理，因此，当 **Adaptive Server** 对相同数据执行并发更新时，您不需要产生用户日志高速缓存刷新和缓冲区管理的开销。
- 当事务提交或中止时，运行时策略可以避免将在用户日志高速缓存中注册的任务更改刷新到事务日志中。这些策略减少了对内存日志页的争用。
- 支持使用内存日志记录技术的最少日志记录 **DML** 操作，从而提高大量 **DML** 操作的性能。

Adaptive Server 15.5 版允许您创建下列类型的数据库：

- 将持久性设置为 **full** 的磁盘驻留式数据库（这是缺省的 **Adaptive Server** 数据库）
- 用户定义的磁盘驻留式临时数据库
- 将持久性设置为 **no_recovery** 的内存用户数据库
- 将持久性设置为 **no_recovery** 的用户定义内存临时数据库
- 将持久性设置为 **no_recovery** 或 **at_shutdown** 的磁盘驻留式宽松持久性数据库

Adaptive Server 对内存数据库和宽松持久性数据库都支持 **dump database** 和 **load database**。此外，还可以转储内存数据库并将其装载到磁盘驻留式数据库中，反之亦可。**Adaptive Server** 支持在服务器重新启动时使用模板数据库来重新初始化不可恢复数据库的内容。

请参见《内存数据库用户指南》。

针对备份的更快压缩

针对转储数据库和事务的新增压缩级别提供更快、更完整且 CPU 占用率更低的压缩。

向 **dump database...compression=** 和 **dump transaction...compression=** 命令中添加了两个新的压缩级别：100 和 101。级别 100 可提供更快的压缩，而级别 101 可提供更好的压缩。这两种压缩级别对 CPU 的占用率比级别 0 - 9 都要低。

请参见《系统管理指南》和《参考手册：命令》。

针对 IBM® Tivoli® Storage Manager 的 Backup Server 支持

Adaptive Server 15.5 支持 IBM Tivoli Storage Manager (TSM) 与 Backup Server 一起来提供存储管理服务。对 TSM 的支持是一项获得许可的 Adaptive Server 功能。

TSM 是一种第三方客户端/服务器程序，可为获得许可的用户提供存储管理服务。Adaptive Server Backup Server 支持将磁带驱动器和磁盘文件用作备份介质。通过结合使用 TSM 和 Backup Server，可以扩大备份介质的可选范围。您可以将 Adaptive Server 备份存储在 TSM 支持的任何备份介质上。

Adaptive Server 负责向 TSM 转储数据库和事务及从 TSM 装载数据库和事务；TSM 则负责存储和检索存储介质。TSM 提供了存储管理服务，但您也可以从 Backup Server 中管理所有 Adaptive Server 备份和恢复操作，包括查询和删除备份对象。

当您执行 **dump** 或 **load** 命令时，Backup Server 会调用带 TSM API（可与 TSM 进行通信）的 Sybase 接口。在使用 **dump** 命令时，应指定一个只与备份对象关联的对象名。此对象名与 TSM 对象名相同，以后进行装载操作时也应使用此对象名来指定相同的数据库或事务转储。通常情况下，如果未配置 TSM，则在 TSM 中执行 **dump** 和 **load** 命令时，可使用在 Backup Server 中执行这些命令时所使用的选项。

Adaptive Server 引入以下存储过程来支持查询和删除 TSM 中的备份对象：

- **sp_deletesmobj** – 删除 TSM 中当前服务器的部分或全部备份对象。
- **sp_querysmobj** – 检索服务器的备份对象的列表。

请参见《将 Backup Server 与 IBM Tivoli Storage Manager 配合使用》。

用户定义的存储过程的延迟名称解析

延迟名称解析能让您在被引用对象存在以前创建存储过程。

在 Adaptive Server 15.5 之前的版本中，必须先存在被引用对象，然后才能创建使用它们的存储过程。现在，延迟名称解析功能允许在首次执行存储过程时解析对象，但用户定义的数据类型对象除外。

此功能使用在服务器级使用的 **deferred name resolution** 配置参数或在连接级使用的新 **set** 命令 **set deferred_name_resolution**。

请参见《Transact-SQL 指南》、《系统管理指南：卷 1》和《参考手册：命令》。

FIPS 140-2 登录口令加密

Adaptive Server 15.5 支持 FIPS 140-2 验证的加密模块。

如果使用了配置参数 **FIPS login password encryption**，FIPS 140-2 认证的 Certicom Security Builder GSE 会对传输的登录包中、内存中和磁盘上的登录口令进行加密。

```
sp_configure 'FIPS login password encryption', 1
```

注意： 启用该参数需要提供安全和目录服务许可证。如果未启用该参数，OpenSSL 安全提供程序会执行登录口令加密。

请参见《加密列用户指南》。

增量数据传输

增量数据传输能让您将数据传输到 Adaptive Server 或其它产品。

transfer table 命令能让您以增量方式传输数据，并且在需要时，可以将数据增量传输到不同产品。在 Adaptive Server 15.5 之前的版本中，只能将所有表从一个 Adaptive Server 传输到另一个 Adaptive Server。

注意： 在您购买、安装和注册内存数据库许可证后，Adaptive Server 即会启用数据传输功能。

增量数据传输：

- 允许从标记为增量传输的 Adaptive Server 表中导出数据，其中只包括自以前的传输之后更改过的数据。
- 允许读取表数据，而无需获取常规锁，无需保证任何行检索顺序，并且不会干扰其它正在进行的读取或更新操作。
- 允许将所选行写入输出文件（可以是命名管道），其格式为定义的接收方：IQ (Sybase IQ)、ASE (Adaptive Server Enterprise)、批量复制 (bcp) 或字符编码输出。将不加密传输所有选定行，并且缺省情况下，会在传输行中的任何加密列之前先对其解密。要写入的文件必须对运行 Adaptive Server 的计算机可见（文件可以是 Adaptive Server 可作为本地文件打开的 NFS 文件）。
- 保留合格表的传输历史记录，并且允许在不需要时删除传输历史记录。从已声明不符合增量传输的表导出数据时存在某些限制。
- 从指示的表传输所有行。目前无法选择某些列，可选择表中的分区或传输 SQL 查询的结果。

请参见《Transact-SQL 用户指南》中的第 8 章“添加、更改、传输和删除数据”。

bigdatetime 和 bigtime 数据类型

bigdatetime 和 *bigtime* 提供微秒级精度。

bigdatetime 和 *bigtime* 这两个新数据类型提供具有微秒级精度的时间戳，其中包括年份、月份、日期、小时、分钟、秒钟以及有 6 位小数的秒钟小数部分。*bigdatetime* 值需要 8 个字节来进行存储。*bigtime* 值包含一天中的时间，具体含有小时、分钟、秒钟和秒钟小数部分。小数部分存储成 6 个小数位。*bigtime* 值需要 8 个字节来进行存储。

请参见《Adaptive Server Transact-SQL 用户指南》、《参考手册：构件块》和《Adaptive Server Enterprise 中的 Java》。

创建和管理 tempdb 组

tempdb 组针对 15.5 版的更新

除了能让您管理缺省的 **tempdb** 组以外，Adaptive Server 还能让您创建和管理用户创建的 **tempdb** 组。用户创建的 **tempdb** 组可以包括其他用户创建的临时数据库，并支持应用程序和登录名绑定。

您不能从缺省临时数据库组中删除系统 **tempdb**。您不能将系统 **tempdb** 添加到用户创建的任何其它 **tempdb** 组中。

您可以指定和管理用户创建的 **tempdb** 组，以便仅包含磁盘驻留式数据库或仅包含内存临时数据库。Adaptive Server 没有显式施加这种限制，但通过控制成员资格，您可以将仅磁盘或仅内存 **tempdb** 组分配给特定的登录名或应用程序。

Adaptive Server 15.5 中的系统更改

Adaptive Server 15.5 支持新增和更改的数据类型、函数、系统存储过程、命令、配置参数、监控表、系统表和实用程序。另外，还添加了新的审计选项。

数据类型

新的 *bigtime* 和 *bigdatetime* 数据类型提供精确的时间戳信息。

数据类型	说明
<i>bigtime</i>	<i>bigtime</i> 值包含小时、分钟、秒钟和秒钟小数部分。小数部分存储成 6 个小数位。
<i>bigdatetime</i>	<i>bigdatetime</i> 值包含年份、月份、日期、小时、分钟、秒钟以及有 6 位小数的秒钟小数部分。

两个新函数返回 *bigtime* 和 *bigdatetime* 值：

- **current_bigtime**
- **current_bigdatetime**

bigtime 和 *bigdatetime* 可以用在以下现有函数中：

- **datepart**
- **datetime**
- **datediff**
- **dateadd**

请参见《参考手册：块》。

函数

Adaptive Server 15.5 中的新增和变更函数

表 8. 新函数

函数	说明
db_attr	返回指定数据库的 durability 、 dml_logging 和 template 设置。
object_attr	根据特定于会话的、表范围及数据库范围内的设置来报告表的当前记录模式。

函数	说明
cache_usage	返回指定对象所绑定到的高速缓存的使用量，即当前由所有绑定到该高速缓存的对象使用的高速缓存百分比。
current_bigdatetime	返回一个 <i>bigdatetime</i> 值，表示具有微秒级精度的当前日期和时间。
current_bigtime	返回一个 <i>bigtime</i> 值，表示具有微秒级精度的当前时间。

表 9. 已更改的函数

函数	说明
datepart	以整数形式生成指定日期的指定 <i>datepart</i> 参数。
datetime	以字符串形式生成指定的 <i>datepart</i> 。
datediff	计算两个指定的日期或时间之间的日期分量数。
dateadd	向指定日期或时间添加时间间隔。

请参见《参考手册：命令》。

系统存储过程

新增和变更的系统存储过程支持 Tivoli Storage Manager、内存数据库和宽松持久性数据库。

表 10. 新的系统存储过程

系统存储过程	说明
sp_deletesmobj	从 TSM 中删除备份对象。
sp_querysmobj	从 TSM 中检索备份对象的列表。

表 11. 已改变的系统存储过程

系统存储过程	说明
sp_addsegment	已更新，以便管理内存数据库中的空间。
sp_addthreshold	已更新，以便管理内存数据库中的空间。
sp_bindcache	您无法将对象或数据库绑定到内存存储高速缓存，并且无法将内存数据库或内存数据库中的对象绑定到任何高速缓存。
sp_cacheconfig	创建、删除内存存储高速缓存或扩大其大小。

系统存储过程	说明
sp_cachestrategy	prefetch 和 MRU 参数不适用于内存数据库中的表和索引。
sp_dbextend	对于内存数据库，当前不支持自动数据库扩展。
sp_deviceattr	directio 和 dsync device 属性不适用于内存设备。
sp_downgrade	支持将包含内存数据库或宽松持久性数据库，或者包含使用模板或最少日志记录的数据库的 Adaptive Server 降级。
sp_diskdefault	不能使用 sp_diskdefault 将内存设备指定为缺省设备。
sp_dropdevice	删除从内存存储高速缓存创建的内存设备。
sp_dropsegment	已更新，以便管理内存数据库中的空间。
sp_dropthreshold	已更新，以便管理内存数据库中的空间。
sp_extendsegment	已更新，以便管理内存数据库中的空间。
sp_help	报告表的属性，例如最少日志记录属性。
sp_helpcache	显示内存存储高速缓存的属性、在此高速缓存中创建的内存数据库以及此高速缓存中可用空间的详细信息。
sp_helpdb	报告数据库的属性，例如，持久性、DML 日志记录级别、是否是内存数据库、模板数据库（如果有）的使用或是否用作模板数据库。
sp_helpdevice	报告从内存存储高速缓存创建的内存设备属性。
sp_modifythreshold	已更新，以便管理内存数据库中的空间。
sp_plan_dbccdb	为内存数据库中的 checkstorage 执行设置 dbccdb 。
sp_poolconfig	内存数据库不支持大型 I/O 缓冲池。
sp_post_xpload	支持内存数据库的跨平台操作。
sp_tempdb	支持用户创建的临时数据库组、登录名或应用程序与临时数据库组的绑定以及用于内存数据库的登录或应用程序绑定。
sp_unbindcache, sp_unbindcache_all	不能从宿主内存存储高速缓存中解除对象或内存数据库本身的绑定。

请参见《参考手册：过程》。

命令

Adaptive Server 15.5 的新增和变更命令

表 12. 新命令

命令	说明
transfer table	<p>启动增量表传输。</p> <p>新的 grant with grant 选项支持 transfer table。它向指定用户授予传输指定表的权限。</p> <pre>grant transfer table on table_name to user with grant option</pre>

表 13. 已更改的命令

命令	更改
alter database	为支持更改数据库持久性、DML 日志记录级别、数据库模板而增加的语法。
alter table	<p>为支持更改表日志记录模式以执行插入、更新与删除 (DML) 操作而增加的语法。</p> <p>为支持 transfer table 而增加的语法：</p> <pre>set transfer table [on off]</pre>
create database	<p>为创建持久性设置为 full、no_recovery 或 at_shutdown 的内存数据库和宽松持久性数据库而增加的语法。</p> <p>为指定 DML 日志记录级别和模板数据库 (如果有) 而增加的语法。</p>
create table	<p>为针对内存数据库中的表指定 DML 日志记录级别而增加的语法。</p> <p>为支持 transfer table 而增加的语法：</p> <pre>with transfer table [on off]</pre>
disk init	为针对内存数据库创建内存数据设备而增加的语法。
dump database	<p>为支持 Tivoli Storage Manager (TSM) 而增加的语法。关键字 syb_tsm 用于调用带 TSM API 的 Sybase 接口 (libsyb_tsm)。</p> <pre>"syb_tsm::object_name" database_name to</pre>

命令	更改
dump database ... compression=	为支持更快的、CPU 占用率更低的压缩级别 100 和 101 而增加的语法。
dump transaction	为支持 TSM 而增加的语法。关键字 syb_tsm 用于调用带 TSM API 的 Sybase 接口 (libsyb_tsm)。 <pre> database_name to "syb_tsm::object_name" </pre>
dump transaction ... compression=	为支持更快的、CPU 占用率更低的压缩级别 100 和 101 而增加的语法。
load database	为支持 TSM 而增加的语法。关键字 syb_tsm 用于调用带 TSM API 的 Sybase 接口 (libsyb_tsm)。 <pre> database_name from "syb_tsm:: [[-S source_server_name] [-D source_database_name]::]object_name" </pre>
load transaction	为支持 TSM 而增加的语法。关键字 syb_tsm 用于调用带 TSM API 的 Sybase 接口 (libsyb_tsm)。 <pre> database_name from "syb_tsm:: [[-S source_server_name] [-D source_database_name]::]object_name" </pre>
select into	为针对通过选入内存数据库和宽松持久性数据库创建的表指定 DML 日志记录级别而增加的语法。
set	添加: <ul style="list-style-type: none"> • dml_logging 参数, 用于指定会话的记录量。 • deferred_name_resolution, 用于在连接级激活延迟名称解析。 • builtin_date_strings number 值包括: <ul style="list-style-type: none"> • 0 - 如果将字符串作为参数传给按时间顺序排列的系统函数, 则服务器会将其解释为 <i>datetime</i> 值 (而不考虑其外观精度)。这是缺省值。 • 1 - 服务器会将参数字符串解释为 <i>bigdatetime</i>。这会影响到按时间顺序排列的系统函数的结果。

请参见《参考手册: 命令》。

配置参数

Adaptive Server 15.5 引入了 **deferred name resolution** 配置参数。

配置参数	说明
deferred name resolution	能让您使用延迟名称解析来创建过程。值包括： <ul style="list-style-type: none"> • 0 - 禁用延迟名称解析。这是缺省值。 • 1 - 启用延迟名称解析。
builtin date strings	值包括： <ul style="list-style-type: none"> • 0 - 导致系统将作为参数传给按时间顺序排列的内置函数的字符串文字解释为 <code>datetime</code> 类型。 • 1 - 导致系统将作为参数传给按时间顺序排列的内置函数的字符串文字解释为 <code>bigdatetime</code> 类型。

请参见《系统管理指南：第一卷》和《Transact-SQL 用户指南》。

监控表

Adaptive Server 15.5 引入了 *monTableTransfer* 监控表。

监控表	说明
<i>monTableTransfer</i>	提供 Adaptive Server 活动内存中表的历史传输信息。

请参见《参考手册：表》。

系统表

Adaptive Server 15.5 中的新增和变更系统表

表 14. 新的系统表

系统表	说明
<i>spt_TableTransfer</i>	存储表传输中的结果。

表 15. 已更改的系统表

系统表	更改说明
<i>sysdevices</i>	在 <i>name</i> 和 <i>phyname</i> 列中列出内存存储高速缓存。内存设备不包括到磁盘设备的完整路径，而是存储在其上创建内存设备的高速缓存的名称。
<i>sysdatabases</i>	添加 <i>durability</i> 列，用以指明数据库的持久性级别。 <i>durability</i> 列的数据类型为 <i>int</i> 。其值为： <ul style="list-style-type: none"> • 1 - full • 5 - at_shutdown • 6 - no_recovery

请参见《参考手册：表》。

实用程序

Adaptive Server 15.5 中的新增和变更实用程序

表 16. 新的实用程序

实用程序	说明
openssl	执行由 <i>certreq</i> 、 <i>certauth</i> 和 <i>certpk12</i> 实现的所有认证管理任务。Sybase 提供此二进制程序是为了方便起见，对使用该二进制程序所引起的任何问题概不负责。有关详细信息，请参见 www.openssl.org 。

表 17. 更改的实用程序

实用程序	更改
backupserver	语法更改支持 <i>-V</i> 参数的另一个详细级别 (V4)。V4 显示所有 <i>-V0</i> 消息，但为每个连接事件输出的“Connection from Server”（与服务器的连接）消息除外。

请参见《实用程序指南》。

审计

增加审计选项是为了支持内存数据库和宽松持久性数据库、增量数据传输以及延迟名称解析。

表 18. 审计增强

审计选项	将被审计的命令或权限	事件	extrainfo 中的信息
all、 create	transfer table	136	关键字或选项: transfer_table
all、 create	alter table	3	如果 alter table 包含 set transfer table on , Adaptive Server 就会将此内容输出到 extrainfo: SET TRANSFER TABLE ON。 如果 alter table 包含 set transfer table off , Adaptive Server 就会将此内容输出到 extrainfo: SET TRANSFER TABLE OFF。
all、 create	create table	12	如果 create table 包含 with transfer table on , Adaptive Server 就会将此内容输出到 extrainfo: WITH TRANSFER TABLE ON。 如果 create table 包含 with transfer table off , Adaptive Server 就会将此内容输出到 extrainfo: WITH TRANSFER TABLE OFF。
all、 create	create database	9	关键字或选项: inmemory
all、 create	alter database	2	关键字或选项: inmemory
all、 create	create procedure	11	关键字或选项: deferred_name_resolution

Adaptive Server 15.0.3 中的新增功能

Adaptive Server 15.0.3 引入了分布式事务管理、对 Java 接口的增强、虚拟散列表、巨内存页、对 Adaptive Server 插件的更新、有关高可用性配置期间进行升级的指导，以及对 SQL 语句复制的支持。

SQL 语句复制

Replication Server 15.2 支持 Adaptive Server 数据库的 SQL 语句复制

Adaptive Server Enterprise 15.0.3 引入了由 Replication Server 15.2 及更高版本支持的 SQL 语句复制。

请参见 Replication Server 文档。

安全性增强

Adaptive Server 15.0.3 版引入了多种新的安全性增强。

LDAP 用户鉴定增强

修改 CA 受托根文件不再需要重新启动服务器。

在 Adaptive Server 的早期版本中，如果需要修改证书颁发机构 (CA) 受托根文件，您必须重新启动 Adaptive Server 以使修改生效。Adaptive Server 15.0.3 版及更高版本支持修改受托根文件，因此不必重新启动服务器。一个新的子命令 `reinit_descriptors` 可解除绑定 LDAP 服务器描述符，并重新初始化用户鉴定子系统。

自动 LDAP 用户鉴定和故障恢复

Adaptive Server 管家实用程序可以自动激活失败的 LDAP 服务器

Adaptive Server 15.0.3 提供了对辅助 LDAP 服务器的支持。在以前，在使出现故障的主 LDAP 服务器联机后，若要鉴定新的 LDAP 登录并将它们转移到主 LDAP 服务器，必须要手动激活 LDAP 服务器。

在 15.0.3 版及更高版本中，Adaptive Server 的管家实用程序中增加了一项新的杂事，用于自动激活 LDAP 服务器：'`set_failback_interval`'。

设置故障恢复间隔之后，管家任务将在每次执行其各项杂事时检查是否存在出现故障的 LDAP 服务器。如果找到出现故障的 LDAP 服务器，它将尝试在故障恢复时间间隔到期时激活 LDAP 服务器。

外部鉴定的登录映射

Adaptive Server 可以将外部用户的一个唯一映射映射到一个内部 Adaptive Server 登录。当您配置外部鉴定机制时，如果恰好有一个外部用户到内部 Adaptive Server 登录的映射，并且已成功对其进行鉴定，则 Adaptive Server 将更新内部登录的口令以匹配外部用户的口令。

例如，在以下情况下：

1. USER1 具有 Adaptive Server 登录名 “user_ase”，口令为 “user_password”。
2. 另一个用户具有 LDAP 登录名 “user_ase”，口令为 “user_password”。
3. Adaptive Server 具有 “user_ldap” 到 “user_ase” 的一对一映射。
4. 用户 “user_ldap” 使用口令 “user_ldappasswd” 登录到 Adaptive Server。Adaptive Server 将 “user_ase” 的口令更新为 “user_ldappasswd”。

使用 SSL 指定公用名

对 SSL 认证公用名使用完全限定域名

目录服务条目中指定的服务器名可与 SSL 服务器认证用于执行 SSL 握手的公用名不同。这样，您将能够为 SSL 认证公用名使用完全限定域名（例如，server1.bigcompany.com）。

要将公用名添加到 interfaces 文件中，请使用：

```
asel
  master tcp ether host_nameport_number ssl="CN='common_name'"
  query tcp ether host_nameport_number ssl="CN='common_name'"
```

当客户端使用 SSL 连接到同样使用 SSL 的 Adaptive Server 时，将在 interfaces 文件中的端口号后面放置 SSL 过滤器。目录服务包括您使用 **dsedit** 或文本编辑器添加的公用名。

sp_listener 包括 **CN=common_name** 参数，从而使您能够为 SSL 认证指定公用名。

并发 Kerberos 鉴定

Adaptive Server 可以建立多个 Kerberos 鉴定会话

Adaptive Server 15.0.3 版支持并发 Kerberos 鉴定，而早期版本则在 Kerberos 鉴定过程中使用锁定机制来保护内部数据结构。

如果存在使用 Kerberos 鉴定的并发登录，Adaptive Server 现在将建立多个 Kerberos 鉴定会话。

15.0.3 版还解决了并发登录会话在 Kerberos 鉴定过程中可能会被阻塞的问题。Adaptive Server 以前的版本与 MIT 1.3.x 版和 1.4.x Kerberos GSSAPI 库一起使用时，将发生此并发问题。

虚拟散列表

创建虚拟散列表来有效地组织表。

注意： 虚拟散列表只能在 Linux pSeries 中使用。

可以使用非聚簇索引或聚簇索引对仅数据锁定表执行基于散列的索引扫描。在此类扫描期间，各工作进程会浏览索引的较高级别并读取索引的叶级页。然后，每个工作进程在单独散列表中按数据页 ID 或键值散列，以确定要处理的数据页或数据行。

通过虚拟散列表可以更高效地组织表，因为它不需要单独的散列表。虚拟散列表可存储行，以便查询处理器可以使用散列键确定行 ID（基于行的序号）以及数据的位置。因为它不使用单独的散列表存储信息，因此称其为“虚拟”散列表。

对于必须更有效地使用其中央处理单元 (CPU) 的系统，虚拟散列表是一个不错的选择。

要创建虚拟散列表，请使用 **create table** 命令指定散列区域的最大值。

巨内存页

启用巨内存页可使用更少的页面来覆盖物理地址空间。

注意： 此功能仅在 Linux pSeries 上可用。

CPU 缓存转换后备缓冲器 (TLB) 存储有关从虚拟页地址到物理页地址的转换的信息，而且对物理内存的每个字节访问都需要一个转换（叫做“缓存未命中”）。虽然这些缓存未命中非常耗费成本，但您可以通过启用“巨内存页”来提高 TLB 命中率。

一旦启用，巨内存页会使用更少的页面来覆盖物理地址空间，因此，“簿记”（从虚拟地址到物理地址的映射）的大小减小，在 TLB 中需要的条目会更少，性能也会得到提高。

Adaptive Server 15.0.3 版及更高版本在缺省情况下使用巨内存页来分配共享内存。但如果系统没有足够多的巨内存页（或者未针对巨内存页进行配置），Adaptive Server 就会使用常规页面。

要启用巨内存页，请使用跟踪标志 1653 来启动 Adaptive Server。Adaptive Server 将共享内存上调至最接近 256MB 的倍数。

高可用性配置期间的升级

在启用高可用性 (HA) 后，请按照针对所用平台的 Adaptive Server 安装指南中的升级说明进行操作。

本节中的说明是对《安装指南》中的说明的补充。

重新安装系统存储过程

在启用高可用性后重新安装系统存储过程

1. 禁用主服务器中的 HA:

```
sp_companion secondary-server-name, 'drop'  
sp_configure 'enable HA', 0
```

2. 禁用辅助服务器中的 HA:

```
sp_configure 'enable HA', 0
```

3. 重新启动服务器。

4. 在两台服务器上都运行 installmaster 脚本。

5. 在两台服务器上都启用 HA 属性:

```
sp_configure 'enable HA', 1
```

6. 重新启动两台服务器。

7. 在两台服务器上都运行 installhasvss 脚本。该脚本位于 \$SYBASE/
\$SYBASE_ASE/scripts 中。

8. 重建协同:

```
sp_companion [companion_server_name], configure [,  
with_proxydb]
```

分布式事务管理 (DTM)

Adaptive Server 会自动阻止原打算在分布式事务内执行的 SQL 命令在该事务外执行。

分布式或外部事务由外部事务协调器（如 XA Transaction Manager (TM)）管理。

在 Adaptive Server 15.0.3 之前的版本中，用户应用程序决定在执行 DML 命令时外部事务是否回退。如果 Adaptive Server 在应用程序不知道的情况下隐式中中止外部事务，本来在该事务内部正常运行的 DML 命令就可能会改为在 Adaptive Server 启动的隐式事务内部执行。这种情况可能会导致业务数据的不一致。

在版本 15.0.3 及更高版本中，Adaptive Server 不允许在附加到外部事务的连接上执行任何 DML 命令，直到事务管理器发出分离请求。分离请求表示计划用于外部事务的一批命令的结束。

在版本 15.0.3 及更高版本中，Adaptive Server 会自动阻止原打算在分布式事务内执行的 SQL 命令在该事务外执行。用户应用程序不再必须在发出每个命令前检查全局变量；当某个事务被隐式中止时，将显示一条错误消息 (3953)，说明“由于外部事务已回退，无法执行该命令。”发出 **detach transaction** 命令时，此消息将消失。

Adaptive Server 插件更新

Adaptive Server 插件现在在 Sybase Central 6.00 上运行，而且包含多种新增功能。

用于 Sybase Central 的 Adaptive Server 插件可管理各种 Adaptive Server Enterprise 产品。在 15.0.3 之前的版本中，Adaptive Server 插件基于 Sybase Central 4.3 运行。在 15.0.3 中，Adaptive Server 插件基于 Sybase Central 6.00 运行。15.0.3 版 Sybase Central 6.00 新增了以下功能：

- 搜索工具可帮助您查找插件显示的对象。
- “连接配置文件说明” (Connection Profile Description)、 “导入” (Import) 和 “导出” (Export) 选项允许您向配置文件连接中添加文本说明。
- 更好地支持 Windows Vista。

15.0.3 版 Adaptive Server 插件新增了以下功能。

- 您可以通过从上下文相关工具栏中选择 “添加” (Add) 按钮来创建对象。
- 存储过程和 SQLJ 过程位于 Procedures 文件夹中。
- 现在支持标量函数或用户定义函数。
- 现在可从上下文相关工具栏上的菜单中访问实用程序项。

之前作为 Adaptive Server 插件的一部分提供的 DBISQL11 现在为独立产品（版本为 11.0），并包括以下增强功能：

- 多个结果集的数量不再限制为 10。
- Adaptive Server 的登录对话框现在保留并显示最后五个连接的服务器名称。
- DBISQL11（或 Interactive SQL）现在支持类似于连接配置文件的连接收藏夹。
- SQL 语句窗格现在包含行号。
- “结果” 窗格现在显示使用 **select all**、**insert/update/delete** SQL 语句的情况，以及选定行中的排序和生成情况。

Java 接口

Adaptive Server 中的 Java 现在支持第三方 JRE 和 JVM 组件（如 J2SE）。

Adaptive Server 15.0.3 版允许在 Adaptive Server 中插入现成的 Java 运行时环境 (JRE) 和 JVM 组件（如 J2SE）。这种 Adaptive Server Java 框架叫做可插入组件接口 (PCI)，其中包括可插入组件适配器 (PCA)。为 Adaptive Server 配置的任何 JVM 都叫做“插件”。

Adaptive Server 15.0.3 中的新增功能

Adaptive Server Java 框架能让您在失去重要功能的情况下，在 **Adaptive Server 15.0.2** 版及早期版本中的 **Java** 解决方案上进行构建。使用 **Adaptive Server 15.0.3** 之前的版本开发的任何 **Java** 应用程序都应该能与您使用 **15.0.3** 及之前的版本创建的 **Java** 应用程序一起无缝运行。

在将 **Adaptive Server** 配置为和 **PCI** 一起运行后，您可以包括任何支持 **Java 6** 或更高版本的标准 **JVM**。这会将您的 **Java** 应用程序和 **Adaptive Server** 分离开来，以便您在不受 **Adaptive Server** 影响的情况下更改或升级 **Java** 应用程序以及随时利用新增的 **Java** 功能。

有关新 **Adaptive Server Java** 接口的完整说明，请参见《**Adaptive Server Enterprise** 中的 **Java**》。

Adaptive Server 15.0.3 中的系统更改

Adaptive Server 15.0.3 引入了新增和更改的函数、系统存储过程、命令、配置参数和系统表。

函数

Adaptive Server 15.0.3 引入了 **password_random** 和 **pssinfo** 函数。

函数	说明
password_random	生成可满足在 Adaptive Server 上定义的全局口令复杂程度检查的伪随机口令。
pssinfo	返回进程状态结构中的信息，其中一个选项用于检索任何 spid 的事务隔离级别。

请参见《参考手册：命令》。

系统存储过程

Adaptive Server 15.0.3 中的新增和变更系统存储过程

表 19. 新的系统存储过程

存储过程	说明
sp_tabsuspectptn	基于字符的分区键的域分区表会在排序顺序更改后变为可疑，散列分区表会在跨平台转储装载后变为可疑。
sp_jreconfig	这是一个 Java 存储过程。

表 20. 已改变的系统存储过程

过程	更改
sp_idapadmin	支持新参数 set_failback_interval 和 reinit_descriptors 。
sp_addserver	支持 filter 参数，以便为远程过程调用添加远程服务器。
sp_passwordpolicy	支持 validate password options 参数。
sp_pciconfig	支持数据库中的 Java。

Adaptive Server 15.0.3 中的系统更改

过程	更改
sp_sysmon	支持附加计数器。

请参见《参考手册：过程》。

命令

Adaptive Server 15.0.3 引入了对 **create table** 和 **update statistics** 命令的修改。

命令	更改说明
create table	您使用此选项创建的表仅对 BCP IN 和 'alter table unpartition' 操作可用。
update statistics	重置全局非聚簇索引的数据更改计数器。

请参见《参考手册：命令》。

配置参数

列出了 Adaptive Server 15.0.3 的新增配置参数

过程	更改
enable pci	这是 Java 配置参数。
maximum nesting level	最大嵌套级别已经增大到 100。
mnc_full_index_filter	将防止 Adaptive Server 考虑在服务器级没有限制搜索参数的非覆盖索引： <ul style="list-style-type: none">索引中的列没有直方图的谓词
pci memory size	这是 Java 配置参数

请参见《系统管理指南：卷 1》。

监控表

Adaptive Server 15.0.3 的新增和变更监控表

表 21. 新的监控表

表	说明
<i>monSQLRepActivity</i>	提供使用 SQL 语句复制功能复制的 DML 语句上的所有打开对象的统计信息。
<i>monSQLRepMisses</i>	提供未使用 SQL 语句复制功能的复制操作的统计信息。 <i>threshold</i> 、 <i>querylimitation</i> 和 <i>configuration</i> 列指示这些因素之一阻止对象的 SQL 语句复制的次数。

表 22. 已更改的监控表

监控表	更改说明
<i>monSysStatement</i> 、 <i>monSysPlanText</i> 和 <i>monSysSQLText</i>	列 <i>BatchID</i> 、 <i>ContextID</i> 、 <i>ProcedureID</i> 和 <i>PlanID</i> 的值已被修改。
<i>monSysStatement</i>	支持两个新列： <i>ProcNestLevel</i> 和 <i>StatementNumber</i> 。

请参见《参考手册：表》。

系统表

Adaptive Server 15.0.3 在多个系统表中添加了新列。

表	更改
<i>sysqueryplans</i>	新列： <ul style="list-style-type: none"> • <i>dbid, int null</i> • <i>qptime, datetime null</i> • <i>sprocid, int null</i> • <i>hashkey2, int null</i> • <i>key1, int null</i> • <i>key2, int null</i> • <i>key3, int null</i> • <i>key4, int null</i> <hr/> 注意： 这些列留作将来使用。
<i>sysprocedures</i>	新列 <i>qp_setting varbinary(6) null</i> <hr/> 注意： 此列留作将来使用。
<i>sysprocesses</i>	新列 <i>clientport</i> <ul style="list-style-type: none"> • 对客户端进程显示客户端端口号 • 对系统进程显示 0 • 数据类型：无符号的 <i>smallint</i>
<i>syssservers</i>	列 <i>srvnetname</i> 已经从 <i>varchar(32)</i> 更改为 <i>varchar(255)</i> 。

请参见《参考手册：表》。

Adaptive Server 15.0.2 中的新增功能

Adaptive Server 15.0.2 引入了许多新的功能和增强。它们包括对安全性、加密列和性能的增强。新增功能包括归档数据库访问、延迟编译、积极和消极集合处理，以及用户定义的 SQL 函数。

加密列

Adaptive Server Enterprise 15.0.2 提供对加密列的增强

新增功能：

- 保护数据不被管理员使用。您可以使用自己的口令来保护密钥和加密列，以确保数据的隐私，使数据不被 **DBO** 或系统管理员使用。
- 使用由登录口令保护的密钥副本维护应用程序透明度。也就是说，您可以创建密钥副本，并将它们分配给单个用户。用户可以使用其登录口令来加密他们的密钥副本。密钥副本与登录口令关联之后，当用户访问使用密钥加密的数据时，不必再提供密钥加密口令。
- 提供密钥恢复。可以在丢失口令后恢复对密钥的访问。密钥所有者设置恢复密钥副本，该副本可在以后丢失口令后用于重新加密密钥。
- 为没有解密权限的用户返回缺省值。您可以创建或变更表，以便通过 **select** 语句为没有解密权限的用户返回指定的缺省值。这使您能够运行现有的应用程序和报告，而不会生成权限错误，同时可以保护私有数据的安全性，使该数据不被未经授权的用户使用。由未经授权的用户生成的报告不会泄露已加密的数据。
- 限制自动解密权限。当启用 **restricted decrypt permission** 配置参数时，系统安全委员会显式授予解密权限，以限制对数据的访问。当启用 **restricteddecrypt permission** 时：
 - 表所有者不会被隐式授予解密权限。即使在依赖系统加密口令以访问密钥的系统中，模式所有者也不具有对用户数据的自动和隐式访问权。
 - 只有具有 **sso_role** 的用户才能授予解密权限。解密权限支持 **with grant** 选项。
 - 通过所有权链跨视图和表或者过程和表的隐式访问受到限制。
- 添加数据类型。可以对以下附加数据类型进行加密：*date*、*time*、*datetime*、*smalldatetime*、*money*、*smallmoney*、*big int*、*unsigned big int*、*bit*、*unichar* 和 *univarchar*。

归档数据库访问

通过将数据库转储视为传统的只读数据库（“归档数据库”），验证或选择性地恢复数据库转储（归档）中的数据

与传统数据库不同，归档数据库使用实际的数据库转储作为其主磁盘存储设备，利用最小的传统存储量来表示数据库转储恢复过程中产生的新页或修改页。由于数据库

转储已包含许多（或大多数）数据库页的映像，因此不必使用 **Backup Server** 将页从归档转换为传统的数据库存储，就可以装载归档数据库。因此，装载速度明显快于传统数据库。

归档数据库访问使得可以在数据库转储上直接执行多种操作。

归档数据库不必是原始数据库的完全副本。根据使用 **sp_dumpoptimize** 转储数据库时所采用的优化方式，归档数据库可能被完全填充（数据库中的每一页都存储在数据库转储中），也可能被部分填充（仅分配的页存储在数据库转储中）。

由于数据库转储表现为只读数据库，因此数据库管理员可以使用以下常见工具和技术对它进行查询：

- 对从生产数据库生成的转储的最新副本运行数据库一致性检查。可以将这些检查卸载到其它服务器以避免生产环境中的资源争用。如果资源不成问题，则可以在创建归档所用的同一服务器中直接检查此归档。对归档进行检验能够提供执行恢复操作之前所必需的保证。
- 如果数据库转储的完整性存在问题，那么，通过将其装载到归档数据库中，可以快速测试出操作是否会成功，同时这也是一种很好的工具，可识别出恢复传统数据库时所应使用的正确数据库转储。
- 数据库转储的对象级恢复。通过使用 **select into** 从归档数据库的表中复制要恢复的行，可以恢复丢失的数据。**select into** 操作可以直接在承载归档数据库的服务器中执行，也可以通过使用“组件集成服务”代理表执行（如果归档数据库与需恢复对象不在同一台服务器上）。

另外，还可将事务日志装载到归档数据库中，从而确保执行恢复操作时可以应用相同的装载序列。

查找慢速运行的查询

Adaptive Server 15.0.2 引入了新的 **set** 命令，用于收集有关慢速运行查询的信息。

set 命令的以下参数能让您在不必事先启用 **showplan** 或其它检查参数的情形下，即可收集有关未正常运行查询的诊断信息：

- **tracefile** - 将诊断信息保存至跟踪文件
- **show_sqltext** - 显示 SQL 文本
- **export_options** - 保留会话设置

延迟编译

Adaptive Server 15.0.2 引入了延迟编译。使用延迟编译，优化程序可以编译引用实际运行时值的存储过程语句。

优化程序现在可以对引用本地变量和临时表的过程语句进行运行时编译，因此在优化查询时会使用实际运行时值，而不是幻数。

- 对于搜索子句中引用本地变量和参数的查询、在同一过程中创建的临时表使用了连接的查询以及子查询引用临时表的查询，**Adaptive Server** 都将使用延迟编译。
- 符合延迟编译条件的语句将在首次执行存储过程时被编译。那些在首次调用存储过程时未执行的语句（例如由于 **IF** 子句而被忽略的语句），只有在后续执行存储过程实际执行到这些语句时才会被编译。
- 一旦编译条件语句，无论是首次还是后续过程执行，它都会整合到查询计划中，而不会被重新编译。
- 引用存储参数的查询以前就被编译，而且是仅使用输入到存储过程中时的那些参数的值进行优化的。在 15.0.2 版及更高版本中，这些语句是使用首次执行查询时它们所具有的参数值进行优化的。如果在存储过程执行过程中参数值发生改变，就会使用当前值进行优化。
- 可以通过使用全局开关 **-T7730** 启动服务器，来关闭延迟编译。

用于中文和日文字符集的不区分大小写的排序顺序

本节介绍了用于以下中文和日文字符集的不区分大小写的排序顺序：

- EUC-GB
- GB-18030
- CP-936
- EUC-JIS
- SJIS
- DECKANJI

表 23. 可用于简体中文和日文的排序顺序

语言或脚本	字符集	排序顺序
简体中文	EUC-GB、GB-18030、CP936	不区分大小写的通用字典顺序
日语	EUCJIS、SJIS、DECKANJI	不区分大小写的通用字典顺序

统计集合函数

Adaptive Server 15.0.2 引入了统计集合函数，用于计算方差和标准偏差

集合函数可汇总数据库中一组行中的数据。这些组是使用 **select** 语句的 **group by** 子句构成的。

简单的集合函数（如 **sum**、**avg**、**max**、**min**、**count_big** 和 **count**）只允许用于 **select** 列表中、**having** 与 **order by** 子句中，以及 **select** 语句的 **compute** 子句。这些函数可汇总数据库中一组行中的数据。

Adaptive Server Enterprise 现在支持统计集合函数，从而允许对数值数据进行统计分析。这些函数包括 **stddev**、**stddev_samp**、**stddev_pop**、**variance**、**var_samp** 和 **var_pop**。

这些函数（包括 **stddev** 和 **variance**）都是真正的集合函数，因为它们可以计算由查询的 **group by** 子句所确定的一组行的值。如同其它基本集合函数（例如 **max** 或 **min**）一样，它们在计算时会忽略输入中的空值。此外，无论是否正在分析表达式的域，所有方差和标准偏差的计算都使用 IEEE 双精度浮点标准。

如果对任何方差或标准偏差函数的输入为空集，则每个函数都将返回空值作为其结果。如果对任何方差或标准偏差函数的输入为单个值，则每个函数都将返回 0 作为其结果。

标准偏差和方差

了解新的统计集合函数及其别名。

- **stddev_pop**（也称为 **stdevp**）- 总体的标准偏差。计算所提供的对组的各行进行求值的值表达式的总体标准偏差（如果指定了 **distinct**，则已删除在复制之后仍保留的各行），其定义为总体方差的平方根。
- **stddev_samp**（也称为 **stdev**、**stddev**）- 样本的标准偏差。计算所提供的对组的各行进行求值的值表达式的总体标准偏差（如果指定了 **distinct**，则已删除在复制之后仍保留的各行），其定义为样本方差的平方根。
- **var_pop**（也称为 **varp**）- 总体的方差。计算对组的各行进行求值的值表达式的总体方差（如果指定了 **distinct**，则已删除在复制之后仍保留的各行），总体方差定义为值表达式与值表达式均值之差的平方和，再除以组中的行数。
- **var_samp**（也称为 **var**、**variance**）- 样本的方差。计算对组的各行进行求值的值表达式的样本方差（如果指定了 **distinct**，则已删除在复制之后仍保留的各行），样本方差定义为与值表达式均值之差的平方和，然后再除以组中的行数减 1。

积极和消极集合

集合处理汇总具有集合值的大量数据。

集合处理是 DBMS 环境中最有用的操作之一。它汇总了以下值：

- 指定的一组行中某一列的最小值、最大值、总和或平均值
- 与条件匹配的行计数
- 其它统计函数

在 SQL 中，使用集合函数 **min()**、**max()**、**count()**、**sum()** 和 **avg()** 以及 **group by** 和 **having** 子句来执行集合处理。SQL 语言可实现两种集合处理类型，即 **矢量集合** 和 **标量集合**。**select-project-join** (SPJ) 查询说明了这两种类型的集合处理：

```
select r1, s1
from r, s
where r2 = s2
```

矢量和标量集合

Adaptive Server 15.0.2 支持矢量和标量集合

在矢量集合中，SPJ 结果集在 **group by** 子句表达式中进行分组，然后将 **select** 子句集合函数应用于每个组。查询会针对每组生成一个结果行：

```
select r1, sum (s1)
from r, s
where r2 = s2
group by r1
```

在标量集合中，没有 **group by** 子句，整个 SPJ 结果集由相同的 **select** 子句集合函数集成单个组。查询将生成一个结果行：

```
select sum (s1)
from r, s
where r2 = s2
```

数据插入的性能改进

Adaptive Server 15.0.2 优化了数据插入的性能

以下 15.0.2 功能可以增强数据插入的性能：

- 快速 **bcpc** 可以将数据复制到具有非聚簇索引或触发器的表中，从而提高 Adaptive Server 插入大量数据时的性能。
- Adaptive Server 15.0.2 版中包含用于会话临时数据库的单独 ULC，所以，当用户切换数据库或满足以下所有条件时，包含单个用户数据库和会话临时数据库的多数数据库事务不需要刷新 ULC：
 - Adaptive Server 当前正在提交事务。
 - 所有日志记录都在 ULC 中。
 - 没有提交后日志记录。

配置选项 **session tempdb log cache size** 能让您配置 ULC 的大小，从而有助于确定刷新频率。

- 当 Adaptive Server 拆分索引或数据页时，它会将原始页中的某些行移到新建页中。移动行的操作不会记录下来。Adaptive Server 版本 15.0.2 使用异步写入磁盘的功能，该功能在等待写入完成期间不需要服务器阻塞。Adaptive Server 版本 15.0.2 可自动使用这些异步写入，您无需进行任何配置。
- **tempdb** 事务的吞吐量提高
- Adaptive Server 15.0.2 版提供提交后优化。服务器执行两次日志扫描：第一次扫描查找数据页解除分配和未保留页，第二次扫描查找日志页解除分配。这些扫描是内部优化，对用户透明，并且自动执行；您无法打开或关闭扫描。借助提交后优化，Adaptive Server 可以记住包含这些日志记录的“下一”日志页（向后）。在提交后阶段，在处理完某一页的记录后，Adaptive Server 会移动到“下一”要求提交后处理的页。在并发环境中，许多用户同时将各自的事务记录到

syslogs 中，提交后优化通过避免读取或扫描不必要的日志页，可以提高提交后操作的性能。

在页面拆分期间使用异步写入

当 Adaptive Server 拆分索引或数据页时，它会将原始页中的某些行移到新建页中。移动行的操作不会记录下来。Adaptive Server 15.0.2 使用异步写入以确保一致性。

为确保一致性和持久性，Adaptive Server 必须满足以下条件：

- Adaptive Server 在将修改的页（含有删除的行）写入磁盘之前，先将新页写入磁盘。这样可以确保 Adaptive Server 能够在事务撤消时恢复页的以前版本。即使旧页中丢失了这些行并且未记录这些行的内容，Adaptive Server 也可以在新页中找到这些行，并将其移回旧页中。
- 新页在提交事务之前到达磁盘，这样可确保 Adaptive Server 不会使提交的数据变得松散。如果已提交事务，Adaptive Server 不需要为新页重做该事务，因为未记录行的移动，所以无法实现重做。在撤消的情况下，将收回新页分配；没有可恢复的前像。

以前的 Adaptive Server 版本通过将新页同步写入磁盘来确保满足上述两个条件。但是，由于在同步写入返回之前，服务器可能阻塞，因此这样会降低性能。

Adaptive Server 版本 15.0.2 使用异步写入磁盘的功能，该功能既满足上述条件，在等待写入完成期间也不需要服务器阻塞。

Adaptive Server 版本 15.0.2 可自动使用这些异步写入，您无需进行任何配置。

提高 tempdb 事务的吞吐量

Adaptive Server 的早期版本刷新数据页和单个日志记录 (SLR)，因为 *tempdb* 或不需要恢复的任何数据库都不支持崩溃恢复。

SLR 是在记录一条记录后会立即强制将用户日志高速缓存刷新到 *syslogs* 的日志记录。为 OAM 修改而创建 SLR，Adaptive Server 在混合日志和数据数据库中影响分配页的日志记录创建为 SLR。

- 对于常规数据库，会立即刷新包含 SLR 的 ULC，以避免在缓冲区“固定”期间引起任何未检测到的死锁。避免 SLR 的 ULC 刷新可减少日志信号争用，从而提高性能。
ULC 刷新可避免由缓冲区固定引起的死锁。由于 Adaptive Server 不为无需恢复的数据库固定缓冲区，因此可以避免此类死锁，并且不必为 SLR 刷新 ULC。
- 对于需要恢复的数据库，在检查点期间，Adaptive Server 会将脏页刷新到磁盘。如果 Adaptive Server 崩溃，这样可确保提交的所有数据都会保存到磁盘。但对于不需要恢复的数据库，Adaptive Server 支持运行时回退，而不支持崩溃恢复。这样可避免在检查点刷新脏数据页，并可提高性能。
- 在不需要恢复的数据库中，Adaptive Server 不支持前写式日志。前写式日志通过“重做”日志（重新执行日志中列出的事务）来确保已提交事务的数据可以恢复，通过“撤消”中止或回退事务所做的更改来保持数据库的一致性。前写式日志通过“缓冲区固定”机制来实现。由于 Adaptive Server 不能确保数据库上的前写式

日志不需要恢复，因此它不为这些数据库固定缓冲区，所以它在提交事务时可以跳过刷新日志。

提交后优化

Adaptive Server 15.0.2 版执行两次日志扫描：第一次扫描查找数据页解除分配和未保留页，第二次扫描查找日志页解除分配。这些扫描是内部优化，对用户透明，并且自动执行；您无法打开或关闭扫描。

以前的 Adaptive Server 版本：

- 提交事务后，使用日志记录的三次扫描，分别用于页解除分配、未保留页和日志解除分配。
- 使用页链接执行日志页的反向扫描。有些页没有需要提交后处理的日志记录。

借助提交后优化，Adaptive Server 可以记住包含这些日志记录的“下一”日志页（向后）。在提交后阶段，在处理完某一页的记录后，Adaptive Server 会移动到“下一”要求提交后处理的页。在并发环境中，许多用户同时将各自的事务记录到 *syslogs* 中，提交后优化通过避免读取或扫描不必要的日志页，可以提高提交后操作的性能。

该优化过程不会显示在任何诊断信息中。

对查询处理器的更改

这些对查询处理器的修改描述了可能影响查询计划选择的行为更改。

延迟编译

Adaptive Server 15.0.2 版的查询处理器将对存储过程中的语句的优化延迟到执行这些语句时。

延迟语句的优化有益于查询处理器，因为局部变量的值可用于优化其各自的语句。

早期版本的 Adaptive Server 使用缺省猜口令对使用局部变量的谓词进行选择估计。

非二进制字符集直方图插值

Adaptive Server 15.0.2 版允许选择性估计与二进制字符集具有相同的精确性。

在早期版本的 Adaptive Server 中，只有缺省二进制字符集利用直方图插值，该直方图插值用于估计域谓词的选择性。对于所有其它字符集，Adaptive Server 对直方图单元的选择性估计为 50%。这通常需要 Adaptive Server 使用字符列直方图的大量直方图单元，才能减少与此估计相关的错误。

Adaptive Server 15.0.2 版允许选择性估计与二进制字符集具有相同的精确性，而无需过多的直方图梯级数。这有益于诸如下面使用域谓词的查询：

```
select * from t1 where charcolumn > "LMC0021" and  
charcolumn <= "LMC0029"
```

Adaptive Server 15.0.2 中的新增功能

如果指定的域在同一直方图单元内，则 **Adaptive Server** 能够更精确地估计此选择性。

表达式直方图选择性估计

Adaptive Server 15.0.2 版将直方图估计应用于单列谓词（如果直方图在该列上存在）。

早期版本的 **Adaptive Server** 使用缺省“猜口令”进行选择估计。

Adaptive Server 15.0.2 版将直方图估计应用于单列谓词（如果直方图在该列上存在）。这样可使估计更精确，并改进了查询计划的连接次序选择。

在此示例中，如果表达式选择性非常大，最好还是将表 *t1* 放在连接次序的开始处：

```
select * from t1,t2 where substrng(t1.charcol, 1, 3)
= "LMC" and t1.a1 = t2.b
```

查看当前的优化程序设置

新的虚设表 *sysoptions* 和新的存储过程 **sp_options** 能让您查看优化程序设置。

为了使您能够轻松查看优化程序设置，**Adaptive Server 15.0.2** 版中引入了：

- *sysoptions* - 一个新的虚设表，用于存储有关每个 **set** 选项、其类别、其当前和缺省设置的信息。*sysoptions* 还包含一个位图，其中包含有关该选项的更多详细信息。
- **sp_options** - 一个新的存储过程，用于显示选项值。

新增安全性功能

Adaptive Server 15.0.2 中的新增安全性功能

64 位 Adaptive Server 在 AIX 上支持 PAM

Adaptive Server 15.0.1 在 AIX 5.2 上支持 PAMUA。

AIX 5.2 64 位上的 **Adaptive Server 15.0.2** 版支持基于可插入鉴定模块的用户鉴定 (PAMUA)。请与您的支持代表联系，以确定是否有助于您的 **IBM** 主机上的 **PAM** 的最新修补程序。

要在 AIX 5.2 上的 64 位 **Adaptive Server 15.0.2** 中启用 PAMUA，必须在 `/etc/pam.conf` 文件中提供 **PAM** 模块。例如：

```
ase auth required /usr/lib/security/pam_aix
```

全局登录触发器自动设置

对于 Adaptive Server 15.0.2 版及更高版本，当服务器启动时，在登录触发器中设置或取消设置的任何可导出选项将在登录过程中生效。

Adaptive Server 15.0.1、12.5.4 版及早期版本要求您使用跟踪标志 4073 启动 Adaptive Server 来启用登录触发器选项。

登录触发器中的任何可导出选项设置或取消设置现在都在服务器启动时在登录进程中生效

要禁用新的行为，请在登录触发器内执行 **set export_options off**。

SSL 支持

Adaptive Server 15.0.2 在 Windows 2003 x64 Enterprise Edition 上支持 SSL。

Adaptive Server 15.0.2 版在 Windows 2003 X64 Enterprise Edition 上支持安全套接字层 (SSL)。Windows 2003 X64 使用 OpenSSL 库实现 SSL 功能。

SSL 功能与其它平台上的 Adaptive Server 相同。此功能支持《系统管理指南》中所示的相同密码成套程序和预定义列表。用于 Windows 2003 X64 上的 Adaptive Server 的 OpenSSL 库使用 SSLPlus 中使用的相同密码成套程序名称与 SSLPlus 客户端和服务器通信。密码成套程序名称符合传送层安全性 (TLS)。TLS 是 SSL 3.0 的增强版本，且与 SSL 3.0 版密码成套程序兼容。

改进的口令安全性

Adaptive Server 15.0.2 改进了口令安全性。

Adaptive Server 15.0.2 通过以下方式提高了现有口令保护机制的安全性级别：

- 在网络上发送的口令的更强加密功能
- *syslogins* (磁盘上) 和内存中存储的口令的更强加密功能
- 使用新的时间戳数据和其它帐号报告功能管理登录名和口令的使用

审计增强

Adaptive Server 15.0.2 版引入了另外两种审计安全性功能。

隐藏系统存储过程和命令口令参数

系统存储过程和命令口令可以在审计记录中替换为星号。

如果配置并启用了审计，且设置了 **sp_audit** 选项 **'cmdtext'**，则会将系统存储过程和命令口令参数替换为审计日志中所含审计记录中固定长度的星号字符串。

例如，执行：

```
sp_password 'oldpassword', 'newpassword'
```

当启用审计并设置 **sp_audit cmdtext** 时，会产生类似以下内容的输出：

```
sp_password '*****', '*****'
```

这样做可以保护口令不被其他可以访问审计日志的人看到。

监控失败登录尝试次数

使用 **login_locked** 审计选项来监控失败登录尝试次数。

当登录帐户由于失败登录尝试次数超出配置次数而被锁定时，将记录新的审计选项 **login_locked** 和事件 **Locked Login (value 112)**。设置了审计选项 **login_locked** 即会启用此事件。要设置 **login_locked**，请输入：

```
sp_audit "login_locked", "all", "all", "ON"
```

如果审计表已满，不能记录事件，则会向错误日志发送包含相关信息的消息。

审计记录中包含主机名和网络 IP 地址。监控 **Locked Login** 事件 (112) 的审计日志有助于标识对登录帐户的攻击。

高可用性考虑事项

Adaptive Server 15.0.2 口令安全性更改影响高可用性 (HA)。

HA 配置

主服务器和协同服务器在进行 HA 配置之前，必须具有相同的 “**allow password downgrade**” 值。为了检查主服务器和协同服务器上的 “**allow password downgrade**” 值是否相同，添加了新的定额属性 “**allow password downgrade**”。如果定额属性的值相同，则此 HA 建议检查会成功；如果值不同，则 HA 建议检查会失败。

syslogins 中针对 HA 的口令行为更改

升级到 Adaptive Server 15.0.2 并成功配置 HA 后，首次连接到主服务器时，主服务器和协同服务器上的用户登录口令都会更新。这样会采用相同的磁盘上加密格式将主服务器和协同服务器上的登录口令同步。这样做是为了避免在使用 **sp_downgrade** 将口令降级到早期的 Adaptive Server 15.0 版后，在 “**allow password downgrade**” 阶段 (如 **sp_passwordpolicy** 中所述) 结束时，口令被重新设置或锁定。通过同步口令加密格式，无需被 **sp_passwordpolicy** 或 **sp_downgrade** 重新设置或锁定，就可以继续使用登录口令。

安装和编辑监控表

Adaptive Server 15.0.2 版包含新的用于监控表的安装和编辑功能。

- 安装监控表 - 您不需要再运行 **installmontables** 脚本即可安装监控表。
15.0.2 版以前的 Adaptive Server 版本需要运行 **installmontables** 脚本才能安装监控表。Adaptive Server 版本 15.0.2 将监控表安装包含在 **installmaster** 脚本中。
- 远程访问和编辑监控表 - Sybase 提供 **installmontables** 作为示例脚本，显示如何远程访问监控表。请运行 **installmontables** 以查看有关编辑的说明。

语句高速缓存的监控表

两个新增监控表能让您分析语句高速缓存的内容。

Adaptive Server 语句高速缓存可存储可能重用的特定 **update**、**delete** 和 **select** 语句以及其它语句的 SQL 文本。启用语句高速缓存后，会将这些语句换为轻量过程并保存它们的计划以供重用。发出新语句后，Adaptive Server 将搜索要重用的计划的语句高速缓存。如果 Adaptive Server 找到要重用的计划，则可避免重新编译语句，从而达到性能改善的目的。

Adaptive Server 版本 15.0.1 中引入的实际参数化允许它识别除实际值存在差异之外的所有相同查询，从而在使用语句高速缓存的同时节省了重新编译的开销。除性能优势外，实际参数化还可以在将指标和语句存储到高速缓存的同时减少大量空间的使用。

Adaptive Server 版本 15.0.2 引入两个新的监控表，使用这两个表可以轻松地分析语句高速缓存的内容。

- *monStatementCache* 可提供语句高速缓存的摘要快照。
- *monCachedStatement* 可显示有关高速缓存的每条语句的详细信息。

每个表中的列允许具有两个属性，如果列具有计数器值，则具有“counter”属性，如果可以使用诸如 **sp_sysmon** 的机制对列进行重新设置，则具有“reset”属性。

系统表的行级锁定

Adaptive Server 15.0.2 版使用系统表上的行级锁定来提高性能。

早于 15.0.2 的各版本的 Adaptive Server 在执行数据定义语言 (DDL) 和实用程序命令时，对系统表使用排它表锁。Adaptive Server 锁定的系统表集取决于您执行的 DDL 操作的类型。如果另一个同时运行的 DDL 尝试对同一系统表使用冲突的排它表锁，则该 DDL 必须等待获得任何系统目录的锁。依次执行这些 DDL 操作。

此方法妨碍临时数据库的性能，在这些数据库中，其 DDL 活动也非常频繁，因此它们的目录争用非常激烈。这限制了 Adaptive Server 对使用临时表的应用程序的吞吐量。

Adaptive Server 15.0.2 版使用行级锁定来解决以下问题：

- 系统表争用导致很多 DDL 和实用程序出现瓶颈。
- *tempdb* 争用。因为系统表在行级锁定，所以 Adaptive Server 15.0.2 避免了 *tempdb* 争用。
- 执行 DDL 和实用程序时的共享或排它表级锁。早期版本将大多数系统表转换为仅数据锁定 (DOL)，但在执行 DDL 和实用程序时仍创建共享或排它表级锁。使用系统表的行级锁避免了此争用。

Adaptive Server 仅对目录设置意向锁，这避免了潜在争用（意向锁表示当前对表使用了页级或行级锁）。

- DDL 和实用程序相互阻塞。Adaptive Server 15.0.2 允许 DDL 和实用程序并行运行。

早期版本的 Adaptive Server 使用表锁来实现系统目录同步。Adaptive Server 15.0.2 使用意向锁实现表级同步，使用行锁实现行级同步。早期版本的 Adaptive Server 在对对象执行操作时会锁定整个系统目录，因此会提出单个锁请求。然而，如果有多行对应于系统目录中的某个对象，则 Adaptive Server 15.0.2 版在对该对象执行操作时会为所有适用的行请求锁。

此更改意味着在执行相同操作时，Adaptive Server 15.0.2 比早期版本请求的锁多，并且增加了系统所需的锁资源的数目。因此，您可能需要在升级 Adaptive Server 之后更改 **number of locks** 配置选项。

xmltable() 函数

xmltable() 根据 XML 表中的元素创建 SQL 表。

xmltable() 用于从 XML 文档中提取一系列多值元素，并将这些元素组合成一个 SQL 表。调用一次 **xmltable()** 将替换在每个迭代上多次调用 **xmlextract()** 的 Transact-SQL 循环。**xmltable()** 被作为派生表调用（在不同 SQL 查询的 **from** 子句中指定的带括号的子查询）。调用 **xmltable()** 相当于对由 **xmltable()** 生成的表的每行执行单个 **xmlextract()** 表达式。

xmltable() 是 **xmlextract()** 的一般形式。这两个函数都返回从作为该函数中参数的 XML 文档中提取的数据。不同之处在于：

- **xmlextract()** 返回由单个 XPath 查询识别的数据。
- **xmltable()** 提取由 XPath 查询识别的数据的序列或行模式，并从该序列的每个元素中提取由一系列其它 XPath 查询识别的数据（列模式）。它将在一个 SQL 表中返回所有数据。

重新分配的连接

重新分配的连接允许将本地和远程表之间的连接重新分配到远程服务器。

Adaptive Server 15.0.2 版引入了重新分配的连接，允许将本地和远程表之间的连接重新分配到远程服务器。远程系统使用动态创建的引用回本地表的代理表执行连接。使用远程系统执行连接，可避免大量的网络通信量。

用户定义的 SQL 函数

Adaptive Server 15.0.2 引入了用户定义的 SQL 函数。

使用 **create function** 可创建用户定义的函数，使用 **drop function** 可删除用户定义的函数。

可以在标量函数中包括以下元素：

- **declare** 语句，用于定义该函数的局部数据变量和游标。
- 赋给此函数的局部对象的值（例如，使用 **select** 或 **set** 命令为表的局部标量和变量赋值）。
- 引用函数中已声明、打开、关闭和释放的局部游标的游标操作。
- 控制流语句。
- **set** 选项（仅在函数范围内有效）。

Adaptive Server 不允许在向客户端返回数据的标量函数中使用 **fetch** 语句。不能在其中包括：向客户端返回数据的

- **select** 或 **fetch** 语句。
- **insert**、**update** 或 **delete** 语句。
- 实用程序命令，如 **dbcc**、**dump** 和 **load** 命令。
- **print** 语句
- 引用 **rand**、**rand2**、**getdate** 或 **newid** 的语句。

可以包括仅为局部变量赋值的 **select** 或 **fetch** 语句。

instead of 触发器

使用 **instead of** 触发器可覆盖缺省触发操作。

视图通常用于将逻辑数据库模式与物理模式分开。可以在视图上定义 **instead of** 触发器来替换 **update**、**insert** 或 **delete** 语句的标准操作。使用 **instead of** 触发器，可使所有视图（包括那些通过其它方式不可更新的视图）得到更新。

instead of 触发器是一些特殊的存储过程，可覆盖触发语句（**insert**、**update** 和 **delete**）的缺省操作，并执行用户定义的操作。

就像 **for** 触发器一样，每次对特定视图执行数据修改语句时，就会执行 **instead of** 触发器。由于 **for** 触发器是在对表执行 **insert/update/delete** 语句后引发的，因此有时也称为 **after** 触发器。单个 **instead of** 触发器可应用于一个特定的触发操作：

```
instead of update
```

它也可以应用于多个操作，同一触发器可执行其中列出的所有操作：

```
instead of insert,update,delete
```

就像 **for** 触发器一样，**instead of** 触发器也可以在触发器处于活动状态时使用逻辑 **inserted** 和 **deleted** 表来存储修改的记录。这些表中的每一列都可以直接映射到触发器中所引用的基视图中的列。例如，如果名为 **V1** 的视图包含名为 **C1**、**C2**、**C3** 和 **C4** 的列，则 **inserted** 和 **deleted** 表将包含所有这四个列的值，即使触发器只修改列 **C1** 和 **C3** 也是如此。Adaptive Server 会以内存驻留对象的形式自动创建和管理 **inserted** 和 **deleted** 表。

instead of 触发器允许视图支持更新，并允许实现代码逻辑，该逻辑需要拒绝批处理的某些部分执行，而允许其它部分成功执行。

每个数据修改语句只引发一次 **instead of** 触发器。包含 **while** 循环的复杂查询可能多次重复执行 **update** 或 **insert** 语句，且每次都会引发 **instead of** 触发器。

Adaptive Server 15.0.2 中的系统更改

对 Adaptive Server 15.0.2 中的命令、函数、实用程序、系统过程、系统表、配置参数和全局变量进行的更改

跟踪标志

Adaptive Server 15.0.2 的新增跟踪标志

- 15340: 无论 `advanced_aggregation` 设置如何, 均在整个服务器范围内启用
- 15341: 无论 `advanced_aggregation` 设置如何, 均在整个服务器范围内禁用

命令

Adaptive Server 15.0.2 中的变更命令

表 24. 已更改的命令

命令	更改说明
disk init, disk reinit	如果尝试在 Sybase 建议不要在其上使用块设备的平台上创建块设备, 此命令将显示警告消息。 Sybase 建议您只在 HP-UX、Windows 和 Linux 平台上将块设备用作数据库设备。
create proxy table, create table at remote server, 和 alter table.	不支持 SQL UDF。
disk init, disk resize	当 <code>skip_alloc</code> 设置为 true 时, 允许用户不用零初始化页。 <code>skip_alloc</code> 的缺省值是 false。 在非 Windows 文件系统和 Windows 原始系统上创建的设备支持该参数。
dump transaction	dump transaction 现在可以包括 <code>string</code> 和 <code>char_variable</code> 参数。但是, 不能提供 <code>string</code> 和 <code>char_variable</code> 选项以执行下列命令: <code>begin transaction</code> 、 <code>commit</code> 、 <code>connect to</code> 、 <code>declare cursor</code> 、 <code>rollback</code> 、 <code>dbcc</code> 、 <code>use</code> 或嵌套的 <code>execute</code> 命令。

请参见《参考手册: 命令》。

对 **set** 命令的更改

Adaptive Server 15.0.2 引入了对 **set** 命令的多项修改。

set 命令更改
set advanced_aggregation 在会话级启用和禁用高级集合。
set switch 用于在本地和服务器范围内设置跟踪标志和开关名称。
在使用抽象计划来创建存储过程或在 Transact-SQL 批处理中运行这些抽象计划时，部分 set 参数的编译期行为发生了更改。
set 命令可以包含 string 和 char_variable 参数。

实用程序

Adaptive Server 15.0.2 支持对 **ddlgen** 实用程序的增强。

更改	说明
ddlgen 提示输入口令。	在 15.0.2 之前的版本中，如果不包含 -P 口令参数，则会返回错误。 ddlgen 现在会提示输入口令。
ddlgen 支持 15.0.2 之前的版本和 15.0.2 版本的加密。	示例语法： <pre>ddlgen -Usa -P -Sserver -TEK -NSampleKeysDB.dbo.ssn_key</pre>
不带 -XOD 标志的 ddlgen	可能会发生两种情况： <ul style="list-style-type: none"> 创建加密密钥时，如果用户未指定口令，ddlgen 将生成不带口令的 DDL。 创建加密密钥时，如果用户指定口令，ddlgen 将生成缺省口令。
带有 -XOD 标志的 ddlgen	生成系统加密口令。
密钥副本支持	ddlgen 可以为密钥副本以及基本密钥生成 DDL。

请参见《实用程序指南》。

系统存储过程

Adaptive Server 15.0.2 中的新增和变更系统过程

表 25. 新的系统存储过程

系统存储过程	说明
sp_downgrade	验证是否已准备好降级至早期的 15.0.x 版。同时降级 Adaptive Server 15.0.2 已修改的系统目录更改。
sp_spaceusage	报告表、索引或事务日志的空间使用并估计数据库中表和索引的分段数。估计值是使用数据和索引行的平均行长度以及表中的行数计算的。可将空间使用情况和碎片数据归档，以用于以后的报告和趋势分析。 sp_spaceusage 支持多种操作，包括 help 、 display 、 archive 和 report ，以指示当前的 Adaptive Server 空间使用情况。

表 26. 已改变的系统存储过程

系统存储过程	更改说明
sp_autoformat	现在接受数据类型为 <i>int</i> (<i>smallint</i> 、 <i>bigint</i> 、 <i>tinyint</i> 、 <i>unsigned int</i>)、 <i>numeric</i> 、 <i>money</i> 、 <i>date/time</i> 以及 <i>float</i> 、 <i>real</i> 和 <i>double</i> 精度的列。
sp_changedbowner	更改数据库的所有者。现在，可以使用 <i>sa_role</i> 或 <i>sso_role</i> 权限来执行它。该数据库阈值的所有者也更改为指定用户。
sp_checksourc	对用户定义的函数的文本进行加密。
sp_configure	现在用于显示非缺省值设置。
sp_depends	用于为用户定义的函数检查任何对象依赖性。
sp_deviceattr	如果为文件系统上的数据库设备禁用了 dsync 选项，此存储过程将显示一条警告消息。

系统存储过程	更改说明
sp_displaylogin	<p>包括以下更改：</p> <ul style="list-style-type: none"> 既支持通配符表达式，又支持服务器用户 ID，并可显示匹配的登录名： <pre>sp_displaylogin ['user_id' '[loginame wildcard]'</pre> <ul style="list-style-type: none"> user_id - 显示其登录名的用户的用户 ID (<i>suid</i>)。 wildcard - 用于搜索目的的通配符。 显示 <i>suid</i> 为 56 的用户的登录帐户： <pre>sp_displaylogin '56'</pre> 显示登录名以 “st” 开头的所有用户的登录帐户信息： <pre>sp_displaylogin 'st%'</pre>
sp_droplogin	如果由于任何数据库的 <i>sysusers</i> 中存在引用了登录 suid 的用户，造成 sp_droplogin 无法删除登录名，则错误消息中将显示在其中发现引用的数据库的名称。
sp_help	显示有关用户定义的函数的信息。
sp_helpdevice	sp_helpdevice 的 <i>description</i> 列显示有关设备类型的信息。设备类型为下列类型之一：raw device、block device 或 file system device。
sp_helprotect	sp_helprotect 中的新选项 permission_name 提供有关给定数据库中授予的任何特定权限的信息（授权者名称、被授者名称、表/列名称和可授权性）。
sp_hidetext	对用户定义的函数的文本进行加密。
sp_locklogin	sp_locklogin 的 lock 选项在与 number of inactive days 的值一起使用时，将锁定在该期间内未进行鉴定的不活动帐号。以下示例锁定在过去 60 天内未进行鉴定的所有登录帐号。
sp_modifylogin account	' max failed_logins ' 选择的新值指示：每当鉴定失败时 <i>syslogins</i> 列登录计数中的失败登录计数都会更新，但不会锁定帐户。
sp_modifystats	允许系统管理员或拥有权限的任何用户执行该过程和更新目标表的统计信息，以修改 <i>sysstatistics</i> 中的列的密度值。
sp_monitorconfig	经过增强，如果用户为 result_tabl_name 传递了尚不存在的表名，则将创建一个表来保存结果集。

系统存储过程	更改说明
sp_passwordpolicy	<ul style="list-style-type: none"> sp_passwordpolicy 中的 set 和 clear 列现在通过审计事件 115 “Password Administration” 进行审计。 附加语法： <pre>sp_passwordpolicy "enable last login updates" , "allow password downgrade" "regenerate keypair" , "expire login passwords" , "[login_name wildcard]" "expire role passwords" , "[role_name wildcard]" "expire stale login passwords" , "datetime" "expire stale role passwords" , "datetime" "maximum failed logins" , -1</pre>
sp_fixindex	现在用于处理一组索引而非单个索引。如果目标表具有位置索引或聚簇索引， sp_fixindex 将重建数据层（它在处理系统表的位置索引或聚簇索引时回收数据层中未使用的空间）。
sp_sendmsg	在以前的版本中，使用此系统过程发送的消息的最大长度为 255 个字符。对于 Adaptive Server 15.0.2 版， sp_sendmsg 消息的最大长度为 4096 个字符。
sp_who	新增列 <i>tempdbname</i> 用于显示所有活动会话的临时数据库名称。
sp_helptext	用于报告用户定义的函数的文本，并引入了 numlines 和 printops 参数。
sp_ldapadmin	引入了新的参数： set_max_ldapua_desc 、 set_num_retries 和 set_log_interval 。
sp_monitor	增强了 event 和 help 参数。
sp_tempdb	引入了 show 和 who 参数。

请参见《参考手册：过程》。

系统表

Adaptive Server 15.0.2 中的新增和变更系统表。

表 27. 新的系统表

系统表	说明
<i>sysoptions</i>	<i>sysoptions</i> 是一个由 sp_options 查询的虚设表。列名区分大小写。

表 28. 已更改的系统表

系统表	更改说明
<i>sysquerymetrics</i>	Adaptive Server 15.0.2 版增加了在用户 ID 之间共享的指标数。此更改减少了 <i>sysquerymetrics</i> (<i>sysqueryplans</i> 的一个视图) 中的条目数, 并可自动集合不同用户 ID 中相同查询的指标。
<i>syscolumns</i>	向 <i>status2</i> 列添加一个状态位, 以指明加密列中附加了 <i>decrypt_default</i> : 0x00001000 (4096) - 列具有解码缺省值
<i>sysobjects</i>	<i>sysobjects</i> 的 <i>type</i> 列对于具有解码缺省值的每个对象都包含一个 “DD” 条目。
<i>sysaudits</i>	更改包括: <ul style="list-style-type: none"> • <i>Alter Encryption Key</i> 审计事件名更改为 <i>AEK As/Not Default</i> • Adaptive Server 15.0.2 版支持以下审计事件和编号: <ul style="list-style-type: none"> • 118 - AEK 修改加密 • 119 - AEK 添加加密 • 120 - AEK 删除加密 • 121 - AEK 修改所有者 • 122 - AEK 密钥恢复

系统表	更改说明
<i>sysattributes</i>	<p><i>sysattributes</i> 包括以下更改：</p> <ul style="list-style-type: none"> • 增加了两个类： <ul style="list-style-type: none"> • 类 31 <i>allow password downgrade</i> - 当设置为 1 时，<i>allow password downgrade</i> 启用对登录口令的特殊处理以与 Adaptive Server 15.0 版及更早版本兼容。 • 类 32 <i>enable last login updates</i> - 当设置为 1 时，<i>enable last login updates</i> 使系统表存储上次登录的日期。 • <i>sysattributes</i> 包括有关缺省解码的信息。对列的更改如下： <ul style="list-style-type: none"> • <i>attribute</i> - 在值为 1 (DECRYPT-DEFAULT_ID) 的加密列中为类型为 <i>EC</i> 并且属于类 25 的对象指定缺省解码。 • <i>object</i> - 包括解码缺省 ID。 • <i>object_info_1</i> - 包括其加密列定义解密缺省值的表的表 ID。 • <i>object_info_2</i> - 指定包括解密缺省值的加密列的 colid。
<i>sysencryptkeys</i>	<p>对 <i>sysencryptkeys</i> 的更改包括</p> <ul style="list-style-type: none"> • 新增类型： <ul style="list-style-type: none"> • <i>EK_KEYCOPY</i> - 0x0010, • <i>EK_KEYBASE</i> - 0x0020 • <i>EK_RECOVERY</i> - 0x0040 • 新增状态位： <ul style="list-style-type: none"> • <i>EK_KEYRECOVERY</i>(0x00000004) - 为保护丢失的口令而加密的密钥。 • <i>EK_LOGINACCESS</i>(0x00000008) - 为登录访问而加密的密钥 • <i>EK_LOGINPASS</i>(0x00000010) - 用登录口令加密的密钥 • <i>EK_USERPWD</i>(0x00000100) - 用用户加密口令加密的密钥 • 对 <i>uid</i> 列说明的更改 - 用户访问或密钥恢复行。<i>uid</i> 包含标识与当前行关联的数据库用户的用户 ID (<i>uid</i>)。 以前的 Adaptive Server 版本不使用 <i>uid</i> 列。
<i>syslogins</i>	<p>在以前的版本中，<i>password</i> 列的最大长度为 30 个字节。在 Adaptive Server 15.0.2 版中，<i>password</i> 列的最大长度为 128 个字节。</p> <p>新增列为：<i>lastlogindate</i>、<i>crdate</i>、<i>locksuid</i>、<i>lockreason</i> 和 <i>lockdate</i>。</p>
<i>sysssrvroles</i>	<p>在以前的版本中，<i>password</i> 列的最大长度为 30 个字节。在 Adaptive Server 15.0.2 版中，<i>password</i> 列的最大长度为 128 个字节。</p>

请参见《参考手册：表》。

配置参数

Adaptive Server 15.0.2 中的新增和变更配置参数

表 29. 新的配置参数

配置参数	说明
enable merge join	在服务器级启用或禁用合并连接。
cost of a logical io	指定单个逻辑 IO 的成本。
cost of a physical io	指定单个物理 IO 的成本。
cost of a cpu unit	指定单个 CPU 运算的成本。
enable encrypted columns	在加密列获得许可时启用和禁用加密列功能。
max online q engines	max online q engines 是 MQ 系列必需的，用来指定允许联机的最大 Q 引擎数。
metrics elap max	如果查询经历的时间小于 metrics elap max 的值，则与此查询关联的查询指标将不写入系统表。 enable metrics capture 必须启用。
metrics exec max	如果查询的执行时间小于 metrics exec max 的值，则与此查询关联的查询指标将不写入系统表。 enable metrics capture 必须启用。
metrics liio max	如果查询的逻辑 IO 时间小于 metrics liio max 的值，则与此查询关联的查询指标将不写入系统表。 enable metrics capture 必须启用。
metrics pio max	如果查询的物理 IO 时间小于 metrics pio max 的值，则与此查询关联的查询指标将不写入系统表。 enable metrics capture 必须启用。
min pages for parallel scan	允许并行扫描较大的查询。
not password encryption read	要求使用 Adaptive Server 鉴定的所有传入登录鉴定请求在网络上传输时加密登录口令。
number of Q engines at startup	是 MQ 系列必需的，用来指定可以联机的最大 Q 引擎数。
prod-consumer overlap factor	会影响优化。Adaptive Server 更改了 group by 算法，因此不能将 set statistics IO 与并行计划一起使用。
send doneinproc tokens	替换 dbcc tune 选项 doneinproc 。

表 30. 更改的配置参数

配置参数	更改说明
max async i/os per engine	缺省值已从 2147483647 更改为 1024。
maximum failed logins	新值 -1 指示：每当鉴定失败时 <i>syslogins</i> 列 <i>logincount</i> 中的失败登录计数都会更新，但不会锁定帐户。
print deadlock information	添加了一个新参数值 2，这样您便可将死锁信息摘要输出至错误日志。

请参见《参考手册：表》。

函数

Adaptive Server 15.0.2 中的新增和变更函数

表 31. 新函数

函数	说明
authmec()	返回用于已登录服务器进程 ID 会话的鉴定方法。
index_name()	当您指定索引的 ID 和数据库以及定义了其上的索引的对象时，此函数将返回索引的名称。
hashbytes()	生成固定长度的散列值表达式。

表 32. 已更改的函数

函数	更改说明
used_pages()	在具有聚簇索引的所有页锁定表中，对于值 indid = 0 ， used_pages() 现在只传递数据层中的已用页。当传递 indid = 1 时，将返回数据层和聚簇索引层中的已用页。
创建函数时，Adaptive Server 将检查该函数是 SQL UDF 还是 SQLJ UDF。如果是 SQLJ UDF，Adaptive Server 将检查“sa”权限。如果是 SQL 函数，Adaptive Server 将检查 create function 权限。	

全局变量

Adaptive Server 15.0.2 引入了 **@@lastlogindate** 全局变量。

表 33. 新的全局变量

全局变量	说明
<code>@@lastlogindate</code>	全局 T-SQL 变量 <code>@@lastlogindate</code> 可供每个用户登录会话使用。 <i>datetime</i> 数据类型，其值为当前会话建立之前的登录帐号的 <i>lastlogindate</i> 列。此变量特定于每个登录会话，可供该会话用于确定帐户的上次登录。如果帐号之前未使用过，或 “ <code>sp_passwordpolicy 'set', enable last login updates</code> ” 为 0，则 <code>@@lastlogindate</code> 的值为 NULL。

表 34. 已更改的全局变量

全局变量	更改说明
<code>@@opttimeoutlimit</code>	以前版本的 Adaptive Server 用户文档将 <code>@@opttimeout</code> 列为服务器全局变量，用于显示查询优化的当前优化超时限制。 这是错误的。显示查询优化的当前优化超时限制的全局变量的实际名称为 <code>@@opttimeoutlimit</code> 。

Adaptive Server 15.0.1 中的新增功能

Adaptive Server 15.0.1 引入了对抽象计划的增强，能让您自动将 SQL 查询中的实际值转换为参数说明。

抽象计划更改

Adaptive Server 15.0.1 提供对抽象计划的增强。

这些增强包括：

- 抽象计划语法已得到扩展，以允许多个查询级设置，以前仅在会话级提供这些设置。
- Adaptive Server 在扩展的抽象计划语法中接受 `h_join` 和 `hash-join` 关键字。
- `set` 列支持 `opt criteria` 参数，用以打开和关闭当前优化目标设置。

新的查询级设置

已扩展了抽象计划语法（Adaptive Server Enterprise 用于强制实施优化程序选择的查询计划）以允许多个查询级设置，以前仅在会话级提供这些设置。

优化条件是在会话级由以下 `set` 语句处理的：

```
set
  nl_join|merge_join|hash_join|..
  0|1
```

`use ...` 抽象计划语法已得到扩展以便在抽象计划派生表之前接受任意数量的 `use` 格式。以前，`optgoal` 和 `opttimeout` 不能与派生表位于同一抽象计划中。例如，需要在查询中将以下语句与 `optgoal` 语句分隔开：

```
select ...
  plan
  "(use opttimeoutlimit 10) (i_scan r)"
```

但是，在 Adaptive Server 15.0.1 中，您可以通过两种方法在同一抽象计划中包含多个语句：

- 使用多个 `use` 语句，例如：

```
select ...
  plan
  "(use optgoal allrows_dss)
  (use nl_join off) (...)"
```

- 将多个项放在一个 `use` 格式中，例如：

```
select ...
  plan
  "(use (optgoal allrows_dss) (nl_join off)) (...)"
```

在查询级，通过 **use ...** 抽象计划语法使用优化目标 (**opt_goal**) 或超时 (**opttimeout**) 设置。在会话级，通过 **set plan ...** 语法使用以下设置：

- 优化目标
- 优化超时

抽象计划和优化程序条件的运算符名称的一致性

Adaptive Server 在抽象计划中接受 **h_join** 和 **hash_join** 关键字。

抽象计划中使用的算法名称与 **set** 命令中的名称不同。例如，散列连接在抽象计划中称为 **h_join**，但在 **set** 命令中称为 **hash_join**。Adaptive Server 在扩展的抽象计划语法中同时接受这两种关键字。例如：

```
select ...
  plan
    "(h_join (t_scan r) (t_scan s))"
```

等效于：

```
select ...
  plan
    "(hash_join (t_scan r) (t_scan s))"
```

以及：

```
select ...
  plan
    "(use h_join on)"
```

以及：

```
select ...
  plan
    "(use hash_join on)"
```

当存在表抽象计划时，表抽象计划会优先：

```
select
from r, s, t
...
  plan
    "(use hash_join off)(h_join (t_scan r)
      (t_scan s))"
```

查询使用 **hash_join** 来处理 **r** 和 **s** 扫描；但对于与 **t** 之间的连接，它不使用 **hash_join**（正如 **use** 抽象计划格式所指定的一样），因为在表抽象计划中没有指定它。

扩展优化程序条件 **set** 语法

set 列支持 *opt criteria* 参数，用以打开和关闭当前优化目标设置。

set <opt criteria> 语句（带有 **0** 或 **1** 语法）接受 **on/off/default**，其中，**default** 表示使用的是此优化条件的当前优化目标设置。

请参见《参考手册：命令》。

文字参数化

使用 Adaptive Server 15.0.1 版，您可以自动将 SQL 查询中的文字值转换为参数说明（与变量类似）。

在 Adaptive Server 的早期版本中，如果两个查询除一个或多个文字值不同外，其它部分完全相同，则语句高速缓存中将存储两个单独的查询计划，或者在 *sysqueryplans* 中额外生成两行。例如，单独存储以下查询的查询计划，即使它们几乎是完全相同的：

```
select count(*) from titles where total_sales > 100  
select count(*) from titles where total_sales > 200
```

使用 Adaptive Server 15.0.1 版，您可以自动将 SQL 查询中的文字值转换为参数说明（与变量类似）。新的 **sp_configure** 选项支持此功能，称为 **enable literal autoparam**。

要在全服务器范围内启用或禁用 **enable literal autoparam**，请使用：

```
sp_configure "enable literal autoparam", [0 | 1]
```

其中，1 会自动将文字转换为参数，0 则禁用此功能。缺省值为 1。

可以在会话级使用 **set** 命令来设置文字参数化：

```
set literal_autoparam [off | on]
```


Adaptive Server 15.0.1 中的系统更改

函数

Adaptive Server 15.0.1 引入了三个新函数。

函数	说明
isdate	确定输入表达式是否为有效的 <i>datetime</i> 值。
isnumeric	确定表达式是否为有效的 <i>numeric</i> 数据类型。
partition_object_id	显示指定分区 ID 和数据库 ID 的对象 ID。

请参见《参考手册：命令》。

配置参数

Adaptive Server 15.0.1 的新增和变更配置参数

表 35. 新的配置参数

配置参数	说明
startup delay	控制在服务器启动过程中何时启动 RepAgent。缺省情况下，RepAgent 与 Adaptive Server 同时启动。Adaptive Server 将在错误日志中写入一条消息以说明等待时间。
enable literal autoparam	在全服务器范围内启用和禁用文字参数化。
cis idle connection timeout.	<p>将 Adaptive Server 配置为检查未使用的时间超过指定秒数的任何远程服务器 CIS 连接。Adaptive Server 将删除未使用的连接并重新分配其资源。</p> <p>虽然指定的数字以秒为单位，管家任务至多每分钟唤醒一次，因此，空闲连接处于空闲状态的时间可能比配置的值长得多。如果空闲连接上的事务处于活动状态，则 Adaptive Server 不会删除这些连接；如果用户执行任何访问此连接的命令，则会自动重新建立连接。</p>
sproc optimize timeout limit	将 Adaptive Server 可在系统过程优化方面花费的时间指定为估计的执行时间的一部分。

表 36. 更改的配置参数

配置参数	更改说明
optimization timeout limit	optimization timeout limit 的可用值范围已发生更改。对于 15.0.1 版，它为 0-1000。值 0 表示没有优化超时。
max parallel degree	如果将 max parallel degree 设置为 1（已启用），Adaptive Server 将强制实施串行查询执行，优化程序可以选择并行度比禁用此参数时更高的计划。
number of worker processes	如果尚未为工作线程池中足够多数量的线程配置 number of worker processes ，Adaptive Server 将在运行时调整查询计划以使用较少的工作线程。如果 Adaptive Server 在运行时无法调整查询，则查询将依次重新编译。但是，如果 alter table 和 execute immediate 命令没有足够多的工作线程，将会中止这些命令。

请参见《系统管理指南：卷 1》。

命令

Adaptive Server 15.0.1 引入了语法以及对 **alter table**、**create index**、**create existing table**、**update statistics** 和 **set** 命令的其它更改。

表 37. 已更改的命令

表	更改说明
alter table	使用 alter table 命令删除一个或多个列表或域分区。不能使用 alter table 来删除散列或循环分区。 语法为： <pre>alter table table_name drop partition partition_name [, partition_name]...</pre>
create index	在域、列表和散列分区表上创建唯一本地索引时，索引键列表是分区键列表的超集。
create existing table	包含用于确定 RPC 是使用当前连接还是单独连接的语法： <pre>create existing table (<column_list>) EXTERNAL [non_transactional transactional] PROCEDURE at 'location'</pre> <ul style="list-style-type: none"> <code>non_transactional</code> - 用于执行 RPC 的单独连接。 <code>transactional</code> - 用于执行 RPC 的现有连接。 <p>缺省行为是事务型行为。</p>

表	更改说明
update statistics	<p>Adaptive Server 15.0.1 增加了在全局索引上运行 update statistics 的功能。</p> <pre>update table statistics table_name [partition data_partition_name] [index_name [partition index_partition_name]]</pre> <p>由于运行 update table statistics 时会产生运行 update statistics 的 I/O 开销，因此，应该使用 update statistics 来生成列和表统计信息。</p> <p>您可以创建全局索引来生成全局统计信息，然后删除全局索引。</p>

表 38. 新的 set 命令选项

新的 set 命令选项	说明
set literal-autoparam on off	在会话级启用和禁用文字参数化。
set opt-time-outlimit	opttimeoutlimit 的值范围已更改为 0 - 4000，其中 0 表示没有优化限制。
set index-union on off	<p>当启用时，set index_union 会使用 or 子句限制表扫描。</p> <p>索引联合（也称为 or 策略）用于包含 or 子句的查询。例如：</p> <pre>select * from titleauthor where au_id = "409-56-7008" or title_id = "PC8888"</pre> <p>如果启用了 index_union，则此示例使用 au_id 上的索引来查找所有具有 au_id = "409-56-7008" titleauthor 元组的行 ID (RID)；并使用 title_id 上的索引来查找所有具有 title_id = "PC8888" titleauthor 元组的 RID。Adaptive Server 然后对所有 RID 执行联合以删除重复项。结果 RID 将与 Rid 连接以访问数据元组。</p> <p>如果禁用 index_union，Adaptive Server 就不在查询中使用索引联合策略来限制表扫描，而是使用表其它访问路径（在上面的示例中，它对表 titleauthor 使用表扫描），并在扫描运算符中将 or 子句作为过滤器进行应用。</p>

请参见《参考手册：命令》。

监控表

Adaptive Server 15.0.1 引入两个新的监控表：*monProcedureCacheMemoryUsage* 和 *monProcedureCacheModuleUsage*。

监控表	说明
<i>monProcedureCacheMemoryUsage</i>	每个过程高速缓存分配器具有一行。分配器是由分配器 ID 标识的，此 ID 是由 Adaptive Server 内部提供的。
<i>monProcedureCacheModuleUsage</i>	每个从过程高速缓存中分配内存的模块具有一行。使用模块 ID 标识的模块是 Adaptive Server 过程高速缓存管理功能内部的功能范围分类。

有关锁定方案的详细信息，请参见《性能和调优指南：监控和分析》和《参考手册：表》。

Adaptive Server 15.0 中的新增功能

Adaptive Server 15.0 引入了对数据分区的支持、行锁定系统目录、增强的请求处理器、长标识符、计算列可滚动游标、新数据类型支持、XML 增强、对交互式 SQL 的支持、对 SySAM 和 Adaptive Server 插件的增强、用户定义的 Web 服务支持，以及安全性增强。

分区支持

通过将大表和索引划分成较小的、更易管理的片来对其进行管理时，分区很有用。分区像大规模的索引一样，利用它可以更快、更便捷地访问数据。

分区是数据库对象，可单独进行管理。例如，您可以装载数据，但无法在分区级执行 **create index**。然而分区对于最终用户是透明的，无论表是否已进行分区，最终用户都可以使用同一命令选择、插入和删除数据。

Adaptive Server 15.0 支持水平分区，可以在不同磁盘设备的分区之间分布表行的选择。各个表或索引行是根据语义或循环分区策略分配到分区的。

语义分区策略使用每一行中指定的键列中的数据值来确定该行的分区分配。循环分区策略随机分配各行，而不参考数据值。

分区策略有：

- *散列分区*（语义） - 一个由系统提供的散列函数，用于确定每一行的分区分配。
- *列表分区*（语义） - 将键列中的值与一组特定于每个分区的用户提供的值进行比较。精确匹配确定分区分配。
- *域分区*（语义） - 键列中的值与用户提供的一组同特定于各个分区的上限和下限进行比较。位于指定范围内的键列值决定分区分配。
- *循环分区* - 以循环方式将行随机分配到分区，使各个分区中包含的行数相差无几。这是缺省策略。

您可以

- 在使用 **create table** 和 **create index** 命令创建表或索引时创建分区。
- 使用 **alter table** 命令来改变表的分区策略。
- 使用 **add partition** 将分区添加到现有表。
- 可以使用分区来加速对大量表数据的装载，即使表将最终用作未分区的表也无妨。

行锁定的系统目录

许多系统目录现在可以使用数据行锁定方案。

Adaptive Server 15.0 版将大多数系统目录转换成了数据行锁定方案。这些系统目录继续使用所有页锁定方案：

- 已实现的表，如 *syslocks* 和 *sysprocesses*。这些表是运行时生成的，它们的锁定方案与并发无关。
- *sysmessages* 和 *sysusermessages*，它们都是只读表。
- *sybsecurity* 中的审计表，可以一次写入，多次读取。

Adaptive Server 的内部升级进程在安装、升级或装载升级期间转换系统表锁定方案。

因为 Adaptive Server 15.0 版中的 DDL 与 12.5.x 以及更早的版本使用相同的表级锁，所以运行 DDL 时无并发改进。

查询处理器

Adaptive Server 15.0 提供增强的查询处理器

Adaptive Server 15.0 版查询处理器可以自调优，需要的干预少于早期版本。此版本的 Adaptive Server 对步骤之间用于实现的工作表的依赖性较低，这是因为引擎支持步骤之间的数据流。但是，当 Adaptive Server 确定散列操作和合并操作是否有效时可能需要使用更多的工作表。

新增功能包括对以下各项的支持：

- 查询处理的垂直和水平并行度
- 改进的索引选择，特别是使用 **OR** 子句的连接和使用不匹配但兼容的数据类型的连接和搜索参数 (**SARG**)
- 更有效的算法
- 对在相连接的列中存在数据倾斜的连接使用连接直方图，改进了开销计算
- 改进的查询计划选择，它通过以下方式来提高性能：
 - 新的索引并集和索引交集策略，用于那些对不同索引进行的、使用了 **and/or** 谓词的查询
 - 使用内存中排序和散列的即时分组和排序运算符支持，用于进行带 **group by** 和 **order by** 子句的查询
 - 基于开销的清理和超时机制，为较大的多向连接和星型模式以及雪花模式连接使用排列搜索策略
- 使用以下两种输出改进了问题诊断和解决过程：
 - 可搜索的 **XML** 格式跟踪输出
 - 新的 **set** 命令的诊断输出

- 包含大量表的连接
- 数据和索引分区，尤其适用于大数据集

分区是并行的基本构件块。

Adaptive Server 15.0 版提供了循环分区。循环分区等同于 12.5 版的分区样式。在升级到 Adaptive Server 15.0 版的过程中，所有现有已分区的表都会取消分区并自动转换为单向循环分区表。

长标识符

Adaptive Server 15.0 引入了分隔标识符的扩展限制

对象名称或标识符的长度有了新的限制：常规标识符为 255 个字节，分隔标识符为 253 个字节。新限制适用于大多数用户定义的标识符，包括表名、列名、索引名等等。由于这些扩展限制，一些系统表（目录）和内置函数也进行了扩展。

对于变量，“@”计为 1 个字节，变量 *i* 的名称可以是 254 个字节。

计算列

计算列和基于函数的索引提供更简便的数据操纵和更快捷的数据访问。

计算列能让您创建计算列、计算列索引和基于函数的索引，从而可以更便捷地操作数据，更快速地访问数据。

- 计算列 - 由表达式定义，可以来自相同行、函数、算术运算符或路径名中的常规列。
- 计算列中的索引，或计算列索引 - 包含一个或多个用作索引键的计算列的索引。
- *基于函数的索引* - 包含一个或多个用作索引键的表达式索引。
- **Deterministic** 属性 - 一种可确保表达式总是从指定的一组输入返回相同结果的属性。

计算列与基于函数的索引的相似之处在于：都允许您将表达式或函数用作更复杂函数的基础。

计算列和基于函数的索引的区别

计算列和基于函数的索引在某些方面存在区别。

- 计算列向表达式和索引功能都提供速记，而基于函数的索引不提供任何速记，它允许直接为表达式创建索引。
- 计算列可以是确定性的，也可以是非确定性的，而基于函数的索引必须是确定性的。“确定性”是指如果表达式中的输入值相同，返回值也必须是相同的。

实现计算列与非实现计算列之间的区别

计算列可以是已实现的，也可以是未实现的。

- 当插入或更新基列时，会对实现的列进行预求值并将其存储在表中。与实现列相关的值同时存储在数据行和索引行中。任何对实现列的后续访问都不需要重新求值，因为可以访问它的预求值结果。一旦实现了列，每次访问该列都将返回相同的值。
- 未实现的列又称虚拟列；虚拟列会在接受访问时变为实现列。如果列是虚拟列（即未实现的列），每次访问该列时，必须求出它的结果值。这就意味着如果虚拟计算列是基于非确定性表达式或调用非确定性表达式的表达式，则它可能在您每次访问时都返回不同的值。当您访问虚拟计算列时，还可能会遇到运行时例外（如域错误）。

可滚动游标

Adaptive Server Enterprise 15.0 支持可滚动和不可滚动游标。

“可滚动”是指您可用一次获取所有行或多行（而不是一次一行）的方式来滚动游标结果集；您也可以重复扫描结果集。您必须使用 **Transact-SQL** 或 **JDBC** 来声明可滚动游标，并且您必须具有 Adaptive Server 15.0 或更高版本中提供的查询引擎。使用可滚动游标可以在游标打开的情况下，通过在 **fetch** 语句中指定 **first**、**last**、**absolute**、**next**、**prior** 或 **relative** 选项，将游标位置设置到游标结果集中的任何位置。

若要获取结果集中的最后一行，请输入：

```
fetch last [from] <cursor_name>
```

或者，要选择结果集中的特定行，如此例中的第 500 行，请输入：

```
fetch absolute 500 [from] <cursor_name>
```

“不敏感”或“半敏感”是指游标外部的数据更改对游标的可见度。游标可以是半敏感且不可滚动的。

所有可滚动游标都是只读的。所有 **update** 游标都是不可滚动的。

unitext 数据类型支持

Adaptive Server 15.0 支持可变长度的 *unitext* 数据类型。

可变长度的 *unitext* 数据类型最多可容纳 1,073,741,823 个 Unicode 字符（2,147,483,646 个字节）。您可以在使用 *text* 数据类型的任何位置，使用具有相同语义的 *unitext*。无论 Adaptive Server 缺省字符集是什么，*unitext* 列都采用 UTF-16 编码存储。

unitext 的优点包括：

- 大 Unicode 字符数据。除了 *unicar* 和 *univarchar* 数据类型，Adaptive Server 还提供了完整的 Unicode 数据类型支持，该支持最适用于增量多语种应用程序。
- *unitext* 将数据存储在 UTF-16 中，它是 Windows 和 Java 环境的本机编码数据类型。
请参见《系统管理指南》。

big int 数据类型支持

Adaptive Server 15.0 版支持精确数值数据类型 *bigint*。

以下是 *bigint* 数据类型允许的数字范围：

数据类型	有符号数据类型的范围
<i>bigint</i>	-2^{63} 到 $2^{63} - 1$ 之间的整数（从 -9,223,372,036,854,775,808 到 +9,223,372,036,854,775,807，包括这两个值）。

Adaptive Server *bigint* 支持还添加了 *hextobigint*、*biginttohex* 和 *count_big* 函数。

请参见《参考手册：块》。

无符号整数数据类型支持

Adaptive Server 15.0 支持无符号整数数据类型。

使用这些无符号整型数据类型，您可以扩展现有整数类型的正数范围，而不增加需要的存储大小。也就是说，这些数据类型的有符号版本既可以向负数方向扩展，也可以向正数方向扩展（例如，从 -32 到 +32）。但是，无符号版本只能向正数方向扩展。下面是有符号和无符号数据类型的范围：

数据类型	有符号数据类型的范围	无符号数据类型的范围
<i>bigint</i>	-2^{63} 到 $2^{63} - 1$ 之间的整数（从 -9,223,372,036,854,775,808 到 +9,223,372,036,854,775,807，包括这两个值）	0 到 18,446,744,073,709,551,615 之间的整数
<i>int</i>	-2^{31} 和 $2^{31} - 1$ （2,147,483,648 和 2,147,483,647）之间的整数，包括这两个值	0 到 4,294,967,295 之间的整数
<i>smallint</i>	-2^{15} 和 $2^{15} - 1$ （-32,768 和 32,767）之间的整数，包括这两个值	0 到 65535 之间的整数

整数标识

Adaptive Server 15.0 允许您将某些数据类型用作标识值。

将以下数据类型用作标识值：

- *bigint*
- *int*
- *numeric*
- *smallint*
- *tinyint*
- *unsigned bigint*
- *unsigned int*
- *unsigned smallint*

对 XML 服务的增强

15.0 版中的 XML 增强包括 XML 模式支持、**for xml** 子句增强和 Unicode (I18N) 支持。

XML 模式支持

您可以对 XML 文档进行 DTD 或 XML 模式验证。可以在 **xmlvalidate** 命令或文档本身中指定 DTD 或模式。

您可以使用 XML 模式声明分析、存储并查询 XML 文档。

for xml 增强

在 Transact-SQL 中，表达式子查询是带括号的子查询。它只包含一个列，该列的值是表达式子查询结果，而且必须返回一个行。您几乎可以在任何能够使用表达式的地方使用表达式子查询。

有关子查询的详细信息，请参见《Transact-SQL® 用户指南》。使用 **for xml** 子查询功能，您可以将任何包含 **for xml** 子句的子查询用作表达式子查询。有关 **for xml** 子查询的语法，请参见 XML Services Guide（《XML 服务指南》）。

unicode 国际化 (I18N) 支持

I18N 扩展分为三类：

- **for xml** 子句中的 I18N 支持。映射到 XML 的结果集的列可以包含非 ASCII 数据。此类数据可以在生成的 SQLX XML 文档中表示为纯字符或数字字符表示形式 (NCR)。
- **xmlparse** 和 **xmlvalidate** 中的 I18N，用于存储和验证包含非 ASCII 数据的文档。

- **xmlextract** 和 **xmltest** 中的 I18N，用于处理包含非 ASCII 数据的 XML 文档和查询。

Adaptive Server 插件增强

Adaptive Server 15.0 对 Adaptive Server 插件进行了增强。

对 Adaptive Server 插件的增强提高了效率和便捷性：

- 包括 **Server Discovery**（您可以使用它找到系统上的可用服务器）和自动服务器状态的企业视图。
- 更新服务器、管理远程服务器以及管理服务器日志的能力。
- **SQL Preview** 和 **Job Scheduler** 集成。
- 图形化查询计划查看器。
- 集成外部工具的能力。

Interactive SQL

使用 **Interactive SQL**，可以对服务器执行 SQL 语句、建立脚本和显示数据库数据。

可以独立运行 **Interactive SQL** 或从 **Adaptive Server** 插件运行。它已作为标准查询工具集成在了 **Adaptive Server** 插件中。使用 **Interactive SQL**，您可以：

- 浏览数据库中的信息。
- 对计划包含到应用程序中的 SQL 语句进行测试。
- 将数据装载到数据库并执行管理任务。

此外，**Interactive SQL** 还可运行命令文件或脚本文件。例如，可以建立要对数据库运行的可重复执行的脚本，然后使用 **Interactive SQL** 以批处理形式执行这些脚本。

用户定义的 Web 服务

Web 服务能让您在 **Adaptive Server** 中创建 Web 服务和执行 SQL。

除了由 **Adaptive Server Web 服务引擎** 提供的 Web 方法外，**Adaptive Server Web 服务** 还允许您使用 Web 浏览器或 SOAP 客户端在 **Adaptive Server Enterprise** 中创建 Web 服务并执行 SQL 命令。这些用户定义的 Web 服务使用现有安全性和 **Adaptive Server Enterprise** 中固有的审计控制。

可以用 **create service** 命令创建用户定义的 Web 服务；使用 Web 服务，您可以指定要执行的 SQL、为可以用 **grant** 命令控制的权限创建一级对象以及控制该服务是否可以用 Web 浏览器或 SOAP 客户端调用。**ASE Web Services Engine** 为用户定义的 Web 服务自动生成 WSDL。

请参见 **Adaptive Server Enterprise Web 服务** 《用户指南》。

超大容量存储支持

Adaptive Server 15.0 扩展了可用磁盘设备的数量和每个设备可用的 2K 块的数量。

在 Adaptive Server 15.0 之前的版本中，虚拟页是在内部以 32 位整数格式描述的：第一个字节存放设备号 (**vdevno**)，随后的三个字节描述设备中的页偏移（虚拟页数，以 2K 字节为单位）。这种体系结构将设备数限制为 256，将每个设备的大小限制为 32 千兆字节 — 整个服务器的最大存储限制为 8 TB。

在 Adaptive Server 15.0 中，设备号和页偏移以独立的 32 位整数存储。新的体系结构最多允许创建 2,147,483,647 个磁盘设备，每个磁盘设备大小最大为 2,147,483,648 2K 块或 4 TB。

注意： 由于 *sysdevices* 和 *sysusages* 系统表的模式更改，或许需要修改访问这些表的脚本和存储过程。现在，必须从 *sysdevices* 和 *sysusages* 的 *vdevno* 列中获取设备标识符。这些表的 *high*、*low* 和 *vstart* 列不再存储设备号和虚拟页号，而只存储虚拟页号。

update statistics 的自动运行

在适合您的站点的时间自动运行 **update statistics** 命令。

您不必在某一特定时间手动运行 **update statistics**，而是可以将 **update statistics** 设置为在站点最需要时自动运行，并避免其在妨碍系统的时间内运行。运行 **update statistics** 的最佳时间应根据 **datachange** 函数的反馈而定。**datachange** 还可帮助您确保避免不必要地运行 **update statistics**。可以使用这些模板来确定对象、日程表、优先级和触发 **update statistics** 的 **datachange** 阈值，这可以确保仅在查询处理器生成更有效的计划时使用关键资源。

因为它是资源密集型任务，所以运行 **update statistics** 的决定应当基于一组特定的标准。可以帮助您确定运行 **update statistics** 的合适时间的一些关键参数是：

- 自上次运行 **update statistics** 以来，数据特性发生了多少更改？这称为 “**datachange**” 参数。
- 是否有足够的可用资源来运行 **update statistics**？其中包括空闲 CPU 周期数等资源，并且还要确保在运行 **update statistics** 期间不会发生关键联机活动。

Datachange 是一个关键指标，它用于测量自上次运行 **update statistics** 以来更改的数据量。该指标由 **datachange** 函数跟踪。使用该指标和资源可用性标准，可以使运行 **update statistics** 的过程自动进行。Job Scheduler 提供了自动运行 **update statistics** 的机制。Job Scheduler 包括一组可自定义的模板，用于确定何时应运行 **update statistics**。这些输入包括所有的 **update statistics** 参数、**datachange** 阈值和应当运行 **update statistics** 的时间。Job Scheduler 在一个较低的优先级上运行 **update statistics**，因此不会影响同时运行的关键工作。

SySAM 许可证管理

Sybase 软件资产管理 (SySAM) 的实现已经发生了更改。

这些更改包括：

- 资产管理和报告工具随 SySAM 2.0 版一起提供。使用这些工具可以监控许可证使用和遵守情况。
- 单一安装方法支持所有的 Adaptive Server 版本。
- SySAM 配置不再是可选的。
- 提供灵活的 SySAM 配置选项。
- 不再随订单完成提供 SySAM 许可证。您必须从 Sybase 产品下载中心 (SPDC) 获取许可证书。
- SySAM 许可密钥包括有关购买的支持计划的信息。每当续订支持计划时，您必须更新这些许可证。
- 严格而一贯地实施许可政策。
- 如果不能获得许可证，Adaptive Server 可以在宽限期运行。这些宽限期可以使顾客有足够的时间对导致许可证失效的问题作出回应。Adaptive Server 在宽限期继续正常运行。如果在宽限期结束时还没有解决许可证问题，Adaptive Server 的功能或服务本身将关闭。
- 您会收到有关许可证事件的实时电子邮件通知。
- SPDC 颁发的许可证包括将配置许可证的主机的相关信息。除非 SPDC 重新颁发这些许可证，否则它们不能在另一台机器上使用。

这些更改影响 Adaptive Server 安装和配置进程。有关 SySAM 配置和配置选项的详细信息，请参见《配置指南》中的“SySAM 配置”一章。有关预安装计划和 SySAM 安装的相关信息，请参见针对所用平台的《Adaptive Server 安装指南》。

在安装 Adaptive Server 之前，制定您的 SySAM 配置计划。

警告! 当遇到许可证问题时，SySAM 会提供宽限期。当 Adaptive Server 进入宽限期时，此信息会添加到 Adaptive Server 的错误日志中以更新错误日志。或者，也可以为此类事件配置电子邮件通知。您必须解决导致 Adaptive Server 进入宽限期的问题。虽然 Adaptive Server 在宽限期内正常运行，但如果在宽限期内未得到解决导致许可证失效的问题，它就会关闭或者禁用许可的功能。

查询处理指标 (qp 指标)

查询处理 (QP) 指标用于标识和比较查询执行中的经验指标值。执行查询时，查询与一组作为 QP 指标比较基础的已定义指标关联。

捕捉的指标包括：

- CPU 运行时间 - 执行查询所用的时间（以毫秒为单位）。
- 占用的时间 - 按操作系统时钟上的时间计算出的开始执行命令与当前时间之间的时间差（以毫秒为单位）。
- 逻辑 IO (LIO) 读取数 - 逻辑 IO 读取的数量。
- 物理 IO (PIO) 读取数 - 物理 IO 的读取数量。
- 计数 - 一个查询的执行次数。
- 中断计数 - 因超过资源限制而导致资源管理器中断查询的次数。

每个指标都有三个值：最小值、最大值、平均值。计数和中断计数不包括在内。

对抽象计划的更新

Adaptive Server 15.0 支持对查询处理器的增强。

有关新增的和更改的抽象计划的说明，请参见《查询处理指南》。

showplan 更改

Adaptive Server 15.0 版的 **showplan** 可以更好地表示查询处理器的执行步骤。

Adaptive Server 更改了 **showplan** 消息的格式来表达查询计划的形式。**showplan** 消息不以纵向格式显示，而是显示为：

```
delete
from authors
where au_lname = "Willis"
and au_fname = "Max"

QUERY PLAN FOR STATEMENT 1 (at line 1).

The type of query is DELETE.
  The update mode is direct.

  FROM TABLE
    authors
  Nested iteration.
  Using Clustered Index.
  Index : au_names_ix
  Forward scan.
  Positioning by key.
  Keys are:
    au_lname  ASC
    au_fname  ASC
  Using I/O Size 2 Kbytes for index leaf pages.
  With LRU Buffer Replacement Strategy for index leaf pages.
  Using I/O Size 2 Kbytes for data pages.
  With LRU Buffer Replacement Strategy for data pages.
  TO TABLE
    authors
```

Adaptive Server 15.0 版的 **showplan** 显示了一系列“竖线”（“|”符号）来分隔运算符执行的各个步骤。

在下面的查询中，有三个运算符（EMIT、DELETE 和 SCAN），因此该查询包括三组竖线来显示这种结构：

```
delete
from authors
where au_lname = "Willis"
and au_fname = "Max"

QUERY PLAN FOR STATEMENT 1 (at line 1).
2 operator(s) under root

The type of query is DELETE.
ROOT:EMIT Operator
|DELETE Operator
|  The update mode is direct.
|
| |SCAN Operator
| |  FROM TABLE
| |  authors
| |  Index : aumind
| |  Forward Scan.
| |  Positioning by key.
| |  Keys are:
| |    au_lname ASC
| |    au_fname ASC
| |  Using I/O Size 8 Kbytes for index leaf pages.
| |  Using LRU Buffer Replacement Strategy for index leaf pages
| |  Using I/O Size 8 Kbytes for data pages.
| |  With LRU Buffer Replacement Strategy for data pages.
|
| TO TABLE
|  authors
|  Using I/O Size 8 Kbytes for data pages.
```

注意： 此版本 Adaptive Server 还具有以 XML 格式显示 **showplan** 消息的能力。

安全套接字层使用 FIPS 140-2

在 Adaptive Server 15.0 中，SSL 使用针对 FIPS 140-2 的 1 级验证的加密模块。

安全套接字层 (SSL) 是用于保护通过 Internet 传输的敏感信息（比如信用卡号、股票交易和银行交易）的标准。SSL 依赖于公开密钥和私钥密码术。

Adaptive Server 15.0 版中使用的 SSL 采用了为 FIPS 140-2 的 1 级验证的加密模块。这些加密模块是用于 Windows、Solaris、AIX 和 HP-UX 操作系统上运行的 Adaptive Server 产品的 Certicom Security Builder GSE。有关详细信息，请参见 NIST 网站 <http://csrc.nist.gov/cryptval/140-1/1401val.htm> 上的 542 号确认证书；该证书发行于 2005 年 6 月 2 日。

Adaptive Server 15.0 中的系统更改

Adaptive Server 15.0 引入了对实用程序、命令、系统表、函数、存储过程、保留字和监控表的更改。

实用程序

Adaptive Server 15.0 支持对 **bcp**、**dataserver**、**sqlsrvr**、**ddlgen** 和 **preupgrade** 实用程序的更改。

表 39. 对实用程序的更改

实用程序	更改
bcp	添加新参数 --sho-fi 和 --hide-vcc ，以支持计算列和函数索引。 添加新参数 --maxconn ，以支持并行装载到分区表。 bcp 界面已进行了更改，现在可以向特定分区运行 bcp in ，从特定分区运行 bcp out 。
dataserver	以 TB 为单位指定 -b master_database_size 参数。
sqlsrvr	以 TB 为单位指定 -b master_database_size 参数。
ddlgen	为 -T object_type 参数添加 WS 对象类型，以支持用户定义的 Web 服务。
preupgrade	preupgrade 包括用于执行各种升级检查的增量检查的选项，并且该命令还进行了增强，能够在使用 load database 进行升级的单个数据库上运行。

请参见《实用程序指南》。

保留字

Adaptive Server 15.0 现在在可滚动游标和 XML 服务的帮助下支持新的保留字。

- **insensitive** - 支持可滚动游标
- **xmlextract** - 支持 XML 服务
- **xmlparse** - 支持 XML 服务
- **xmltest** - 支持 XML 服务

在从早期版本的服务器升级前，必须先更改与新保留字相同的所有数据库名称。您可以更改表名、视图名和列名，或者使用分隔标识符。升级到 15.0 版后，如果过程、SQL 脚本和应用程序中包含名称与新保留字相同的数据库对象，则在修改这些名称前，不能使用这些数据库对象。

请参见 《参考手册：块》。

全局变量

Adaptive Server 15.0 引入了新的全局变量。

表 40. 新的全局变量

变量	显示内容	值
@@rowcount	进行了增强，以显示（可滚动或不可滚动）游标移动的行数。 如果游标是不可滚动游标，会将行从基础表移动到客户端。 如果游标是可滚动游标，会从当前结果集获取计数行，而不是从基础表中获取。	全局变量 @@rowcount 的值受所指定的游标类型的影响。 缺省情况下，不可滚动游标一次向前移动一行；最大值是结果集中的行数。 如果是可滚动游标，无论 fetch 命令的方向是什么，该值都会继续增加；没有最大值。
@@fetch_status	用于可滚动游标的 fetch 命令的状态。	0 - fetch 语句成功执行。 -1 - fetch 语句失败，或请求的行在结果集之外。 -2 - 保留的值。
@@cursor_rows	游标结果集中的总行数。	0 - 没有打开任何游标，或没有任何行符合上一次打开的游标。 -1 - 游标是半敏感的并且是可滚动的，但尚未填充滚动工作表。符合游标的行数未知。 <i>n</i> - 上一次打开的或获取的游标结果集是完全填充的；返回值是结果集中的总行数。

配置参数

Adaptive Server 15.0 引入了新的和变更的配置参数。

表 41. 新的配置参数

参数	说明
enable metrics capture	使 Adaptive Server 能够在服务器级捕获指标。

参数	说明
enable semantic partitioning	在许可的站点启用表和索引的语义 (hash、list、range) 分区。
enable web services	启用 Web 服务
enable xml	启用 XML 服务
max native threads per engine	定义服务器为每个引擎生成的最大本机线程数
max partition degree	配置 Adaptive Server 要求的动态重新分区数，这可使 Adaptive Server 能够使用水平并行
max resource granularity	设置一个查询可使用的系统资源的最大百分比
number of devices	指定 Adaptive Server 可使用的数据库设备数
number of dump threads	控制 Adaptive Server 为执行内存转储而产生的线程数
number of open partitions	指定 Adaptive Server 可同时访问的分区数。
optimization goal	允许您配置三个优化目标；您可在三个级别上指定这些目标：服务器级、会话级和查询级
optimization timeout limit	以处理查询总时间的百分比的形式指定 Adaptive Server 用于优化查询的时间
rtm thread idle wait period	定义当 Adaptive Server 空闲时，其使用的本机线程的等待时间
sysstatistics flush interval	确定 <i>sysstatistics</i> 前后两次刷新之间的时间间隔（以分钟为单位）
statement cache size	增加服务器分配的过程高速缓存，并限制过程高速缓存池中用于高速缓存的语句的内存数量。语句高速缓存功能是在服务器范围启用的。

表 42. 更改的配置参数

参数	更改
default network packet size	Adaptive Server 的以前版本使用的 default network packet size 为 512。在 Adaptive Server 15.0 版中， default network packet size 为 2048。

请参见《系统管理指南：第一卷》。

函数

Adaptive Server 15.0 的新增和变更函数

表 43. 新函数

函数	说明
biginttohex	返回与指定的整数等值且与平台无关的十六进制值
count_big	以 <i>bigint</i> 的形式返回 (不同) 非空值的数量或选定的行数
datachange	用于测量上次更新统计信息以来的数据分布更改量
data_pages	返回指定表、索引或特定分区所用的页数
hextobigint	返回与某十六进制字符串等值的 <i>bigint</i> 值
is_quiesced	如果数据库被停顿, 则返回 1; 否则返回 0。
partition_id	返回指定的数据或索引分区名称的分区 ID。
partition_name	新分区的显式名称, <i>partition_name</i> 返回分区名称
reserved_pages	报告为表、索引或特定分区保留的页数。
row_count	返回指定表中行数的估计值。
showplan_in_xml	返回 XML 形式的执行计划。
ssel_message	返回消息文本 (如果指定了消息 ID)。
tran_dumpable_status	返回一个指明是否允许 dump transaction 的真/假值。
used_pages	报告表、索引或特定分区所用的页数。
xmlvalidate	验证 XML 文档, 包括那些包含非 ASCII 字符 (I18N) 的文档。《XML 服务》中介绍了此函数。

已经将多个函数名称替换成可读性更强的名称。

表 44. 被替换成新函数名称的函数

被替换的函数名称	新函数名称
data_pgs	data_pages
used_pgs	used_pages
reserved_pgs	reserved_pages

被替换的函数名称	新函数名称
rowcnt	row_count
ptn_data_pgs	data_pages

命令

Adaptive Server 15.0 中的新增和变更命令

表 45. 新命令

命令	作用
create service	用于创建用户定义的 Web 服务。
drop service	用于创建用户定义的 Web 服务。
update table statistics	更新表或分区的 <i>systabstats</i> 统计信息。

表 46. 新的 set 命令选项

set 选项	说明
set delayed_commit	允许您确定日志记录写入磁盘的时间。当 delayed_commit 参数设置为 true 时，日志记录将以异步方式写入磁盘并且控制将返回客户端，而不等待 IO 完成。
set plan optgoal	在会话级设置优化目标。
set plan opttimeoutlimit	在会话级设置长期运行的复杂查询的时间限制。
set metrics_capture on/off	在会话级激活 QP 指标。

表 47. 已更改的命令

命令	更改
alter table	添加的语法，用于支持计算列、实现列或未实现列。添加了对分区的支持。
create index	已进行了增强，允许将计算列用作索引键（与常规列的方式相同），并创建基于函数的索引。添加了对分区的支持。

命令	更改
create table	添加的语法，用于支持计算列、实现列或未实现列。添加了对分区的支持。
dbcc	添加了对分区的支持。
declare cursor	为可滚动游标添加的语法。添加的语法用于支持 semi_sensitive 、 insensitive 和 scrollable 游标。
delete statistics	添加了对分区的支持。
disk init	可以 TB 为单位指定 size 参数。 添加了 directio 参数，允许配置 Adaptive Server 以将数据直接传送到磁盘，而绕过操作系统缓冲区高速缓存
disk reinit	添加了 directio 参数，允许配置 Adaptive Server 以将数据直接传送到磁盘，而绕过操作系统缓冲区高速缓存
fetch	添加了 fetch_orientation 选项以支持可滚动游标： next 、 prior 、 first 、 last 、 absolute 和 relative 。
reorg	添加了对分区的支持。
select	添加了 for xml 子句以支持 XML 服务。添加了对分区的支持。
truncate table	添加了对分区的支持。
update all statistics	添加了对分区的支持。
update statistics	添加了对分区的支持。
update partition statistics	使其过时。

系统存储过程

Adaptive Server 15.0 中的新增和变更系统存储过程

表 48. 新的存储过程

存储过程	作用
sp_helpcomputedcolumn	报告有关指定表中所有计算列的信息

存储过程	作用
sp_version	返回上次运行的安装脚本 (installmaster、installdbccdb 等) 的版本信息以及它是否成功运行。

表 49. 已改变的系统存储过程

存储过程	更改
sp_checksource	检查计算列源文本是否存在。
sp_help	报告有关计算列、基于函数的索引和分区的信息。
sp_helppartition	向其输出中添加了详细的分区信息。
sp_helpindex	报告有关计算列索引、基于函数的索引和分区的信息。
sp_helptext	显示计算列、基于函数的索引定义和分区的源文本。
sp_hidetext	隐藏计算列、基于函数的索引键和分区条件的文本。
sp_modifylogin	添加了选项 "enable logins during recovery"。
sp_webservices	添加了 addalias、deploy、dropalias、listudws、listalias 和 undeploy 选项，以支持用户定义的 Web 服务。
sp_monitorconfig	支持 number of open partitions 配置参数。
sp_countmetadata	支持 number of open partitions 配置参数。
sp_helpsegment	输出对象和分区的段绑定。
sp_objectsegment	显示一个对象所有分区的段信息。
sp_placeobject	使分区未来能够从新段进行分配。
sp_dbcc_faultreport	为特定的 OPID 或故障类型创建报告。
sp_sysmon	报告有关配置文件的打开分区“元数据高速缓存管理”部分的信息。

请参见《参考手册：过程》。

系统表

Adaptive Server 15.0 中的新增和变更系统表

Adaptive Server 15.0 版提供必要的行锁定目录基础结构，以支持增强的、多用户并发数据定义语言 (DDL) 操作。但是，此版本没有更改 DDL 操作的目录锁定行为。在此

版本中，对于执行大量多用户 DDL 操作（例如，在 *tempdb* 中创建或删除表）的应用程序来说，在目录锁定行为方面没有任何更改，也没有增加任何 DDL 并发。

表 50. 新的系统表

表	说明
<i>syspartitions</i>	<i>syspartitions</i> 与 15.0 之前版本的表完全不同。所有列都是新的。 <i>syspartitions</i> 同时支持表和索引的语义和循环分区。
<i>syspartitionkeys</i>	针对各个 hash 分区表、range 分区表和 list 分区表的分区键中的各个列，分别包含一行。

表 51. 已更改的系统表

表	更改
<i>syscolumns</i>	<p>新字段：</p> <ul style="list-style-type: none"> • computedcol • status3 <p>新列：</p> <ul style="list-style-type: none"> • <i>encrtype</i> - 加密类型 • <i>enclen</i> - 加密列的长度 • <i>encrkeydid</i> - 加密密钥 ID • <i>encrkeydb</i> - 包含加密密钥的数据库名称 • <i>encrdate</i> - 日期列进行了加密。 <p><i>status2</i> 字段中的新位：</p> <ul style="list-style-type: none"> • Hex: 0x00000010, Decimal 16 - 该列是计算列。 • Hex: 0x00000020, Decimal 32 - 该列是已实现的计算列。 • Hex: 0x00000040, Decimal 64 - 该列是视图中的计算列。
<i>sysconstraints</i>	状态字段中的新内部位： Hex 0x0100, decimal 265 - 表示一个计算列对象。
<i>sysdevices</i>	<p>新列：</p> <ul style="list-style-type: none"> • <i>vdevno</i> - 设备标识号 • <i>crdate</i> - 设备创建日期 • <i>resizedate</i> - 更改设备大小的日期 • <i>status2</i> - 附加的 <i>status2</i> 位。
<i>sysusages</i>	新列： <i>vdevno</i> - 设备标识号

表	更改
<i>sysstatistics</i>	<p>新列:</p> <ul style="list-style-type: none"> • <i>indid</i> - 数据分区的索引 ID。始终为 0。 • <i>partitionid</i> - 数据分区的 ID • <i>ststatus</i> - 内部状态位 <p><i>id</i>、<i>indid</i>、<i>partitionid</i>、<i>statid</i>、<i>colidarry</i>、<i>formatid</i>、<i>sequence</i> 上的唯一定位索引</p>
<i>systabstats</i>	<p>新列:</p> <ul style="list-style-type: none"> • <i>partitionid</i> - 数据分区或索引分区的 ID • <i>statmoddate</i> - 上一次在磁盘上修改统计信息的日期。 • <i>unusedpgcnt</i> - 未使用的页数。 • <i>oampagecnt</i> - 各个分区的 OAM 页数。
<i>syspartitions</i>	<i>syspartitions</i> 与 15.0 之前版本的同名表完全不同。所有列都是新的。 <i>syspartitions</i> 同时支持表和索引的语义和循环分区。
<i>syscomments</i>	<p>新列: <i>partitionid</i> - 数据分区或索引分区的 ID</p> <p>对表进行了增强, 以存储计算列的文本或基于函数的索引键表达式。</p>
<i>sysindexes</i>	<p>新列:</p> <ul style="list-style-type: none"> • <i>partitiontype</i> - 分区策略: 1 - 域, 2 - 散列, 3 - 循环, 4 - 列表 • <i>conditionid</i> - 分区条件的 ID <p>新行: 针对各个基于函数的索引或在计算列上创建的索引分别包含一行。</p> <p>向 <i>status2</i> 字段添加了一个新的内部状态位: Hex 0x8000, Decimal 32768 - 该索引是基于函数的索引。</p>
<i>syslocks</i>	<p>新列:</p> <ul style="list-style-type: none"> • <i>nodeid</i> - 留作将来使用。 • <i>partitionid</i> - 数据分区或索引分区的 ID。留作将来使用。始终为 0。

表	更改
<i>sysobjects</i>	<p><i>type</i> 列中的新对象：<i>N</i> - 分区条件</p> <p>新列：</p> <ul style="list-style-type: none"> <i>identburnmax</i> - IDENTITY 列的最大烧毁标识值 <i>spacestates</i> - 被跟踪的空间状态数。（仅适用于 DOL 表。） <i>erlchgts</i> - 上次更改期望行长度时的时间戳。（仅适用于 DOL 表。） <p>新行：一个针对每个计算列和基于函数的索引键对象的行</p> <ul style="list-style-type: none"> <i>type</i> 字段：如果对象是计算列，类型字段中会添加有类型“C” <i>status2</i> 字段：添加了新位，用以指示表包含一个或多个基于函数的索引。
<i>sysprocedures</i>	以二进制形式存储每个计算列或基于函数的索引定义的序列树

表 52. 系统表列中的数据类型更改

系统表	已更改的列	数据类型更改	标识符名称
<i>sysattributes</i>	<i>object_cinfo</i> <i>char_info</i>	<i>varchar(30) null</i> 更改为 <i>varchar(255) null</i> <i>varchar(255)</i> 更改为 <i>varchar(768)</i>	对象的标识符
<i>sysaudits01</i> - <i>sysaudits08</i>	<i>objname</i>	<i>varchar(30) not null</i> 更改为 <i>varchar(255) not null</i>	对象名
<i>syscolumnns</i>	<i>name</i>	<i>varchar(30) not null</i> 更改为 <i>varchar(255) not null</i>	列名
	<i>remote_name</i>	<i>varchar(30) null</i> 更改为 <i>varchar(255) null</i>	将本地名映射到远程名
<i>sysconfigures</i>	<i>name</i>	<i>varchar(80) null</i> 更改为 <i>varchar(255) null</i>	
<i>sysindexes</i>	<i>name</i>	<i>varchar(30) null</i> 更改为 <i>varchar(255) null</i>	表名称的索引
<i>sysjars</i>	<i>jname</i>	<i>varchar(30) null</i> 更改为 <i>varchar(255) null</i>	JAR 名称
<i>sysobjects</i>	<i>name</i>	<i>varchar(30) not null</i> 更改为 <i>varchar(255) not null</i>	对象名
<i>sysprocesses</i>	<i>hostname</i>	<i>char(10) not null</i> 更改为 <i>varchar(30) null</i>	主计算机名

系统表	已更改的列	数据类型更改	标识符名称
	<i>program_name</i>	<i>char(16) not null</i> 更改为 <i>varchar(30) null</i>	<i>front_end</i> 模块的名称
	<i>hostprocess</i>	<i>char(8) not null</i> 更改为 <i>varchar(30) null</i>	主机进程 ID 号
	<i>cmd</i>	<i>char(16) not null</i> 更改为 <i>varchar(30) null</i>	当前执行的命令或进程。对条件语句（例如， if 或 while 循环）求值返回。
<i>systimeranges</i>	<i>name</i>	<i>varchar(30) not null</i> 更改为 <i>varchar(255) not null</i>	时间范围的唯一名称
<i>systypes</i>	<i>name</i>	<i>varchar(30)</i> 更改为 <i>varchar(255)</i>	数据类型名称
<i>sysdatabases</i>	<i>def_remote_loc</i>	<i>varchar(255) null</i> 更改为 <i>varchar(349) null</i>	

请参见《参考手册：表》。

监控表

Adaptive Server 15.0 中的新增和增强监控表

表 53. 新的监控表

监控表	说明
<i>monOpenPartitionActivity</i>	提供分区的监控信息

表 54. 已更改的监控表

监控表	更改
<i>monEngine</i>	管家 GC 任务的新列
<i>monCachedObject</i>	分区的新列
<i>monProcessObject</i>	分区的新列

请参见《参考手册：表》。

获取帮助及其它信息

使用 Sybase 入门 CD、产品文档站点和联机帮助来了解关于此产品版本的更多信息。

- **Getting Started CD**（或下载） - 包含 PDF 格式的发行公告和安装指南，也可能包含其它文档或更新信息。
- 位于 <http://sybooks.sybase.com/> 上的产品文档 - 是 Sybase 文档的在线版本，您可以使用标准 Web 浏览器进行访问。您可以在线浏览文档，也可以采用 PDF 格式进行下载。除产品手册外，该网站还包含指向 EBF/维护、技术文档、案例管理、已解决的案例、社区论坛/新闻组 和其他资源的链接。
- 产品中的联机帮助（如果有）。

要阅读或打印 PDF 文档，您需要 Adobe Acrobat Reader，可以从 Adobe Web 站点免费下载。

注意： 产品文档网站可能会提供更新的发行公告，其中包含在产品发布后增加的重要产品或文档信息。

获取帮助及其它信息

索引

符号

? (问号) 动态参数 9
 @@lastlogindate 全局变量 87
 @@lwpid 全局变量 32
 @@plwpid, 全局变量 9
 @@plwpid 全局变量 32
 @@plwpid, 全局变量 9

A

Adaptive Server 插件 103
 Interactive SQL 103
 Adaptive Server 插件更新 59
 Adaptive Server 中的 Java 59
 Adaptive Server 中的并行度 98
 alter database 命令 17
 alter encryption key 命令 17
 alter login profile 命令 17
 alter login 命令 17
 alter object modify owner 命令 4
 alter table 6
 alter...modify owner 命令 17
 authmec() 函数 87
 automatic master key access 配置参数 23
 安全套接字层 (SSL) 73
 和 FIPS 140-2 107
 安全性功能
 保护登录名、角色和口令管理 3
 登录配置文件 3
 端到端 Kerberos 鉴定 2
 加密列 65
 加密密钥的双控制 3
 外部登录口令和隐藏文本 4
 无人值守启动 3
 移交对象所有权 4
 员工生命周期管理 4
 安装系统存储过程 58

B

Backup Server 43, 44
 backupserver 实用程序 53
 bcp 实用程序 109
 big int 数据类型 101

bigdatetime 数据类型 47
 biginttohex 函数 112
 bigtime 数据类型 47
 builtin date strings 配置参数 52
 保留字 109
 备份
 事务 43
 数据库 43
 变量, 使用 LOB 6
 标量集合 69
 标准偏差 68
 表
 传送行 10
 带引号的标识符 14
 压缩 2
 并置运算符 17

C

cache_usage 函数 47
 charindex 函数 20
 charlength 函数 20
 cis idle connection timeout 配置参数 93
 Cluster Edition
 存档数据库 35
 功能和平台表 33
 column default cache size 配置参数 23
 cost of a cpu unit 配置参数 86
 cost of a logical io 配置参数 86
 cost of a physical io 配置参数 86
 count_big 函数 112
 create database 6
 create database 命令
 命令 17
 create encryption key 命令 17
 create login profile 命令 17
 create login 命令 17
 create service 命令 113
 create table 6
 命令 57
 create table ... [in row [(length)] | off row] 命令
 17
 create_locator 函数 20
 参数, 动态 9
 查询, 慢速运行 66

索引

查询处理

- 延迟, 降低 15
- 指标 105

查询处理器 98

查询计划 71, 98

- 分析 16
- 格式 106

长标识符 99

超大容量存储支持 (VLSS) 104

抽象计划 90

- 保存 4
- 查询级设置 89
- 更新 106
- 增强 89

抽象计划中的 SQL 文本 4

触发器, instead of 77

传送

- 数据 45
- 行 10

磁盘设备, 超大容量存储支持 104

存储过程, 延迟编译 66

D

data_pages 函数 112

datachange 函数 112

datalength 函数 20

dataserver 实用程序 109

db_attr 函数 47

DDL, 完全可恢复 10

ddlgen 实用程序 80, 109

deallocate locator 命令 17

declare cursor ... [release_locks_on_close] command 17

deferred name resolution

- 配置参数 52

detach transaction 命令 58

disable varbinary truncation 配置参数 23

DOL

- 请参见 仅数据锁定 (DOL)

dol_downgrade_check 函数 20

drop encryption key 命令 17

drop login profile 命令 17

drop login 命令 17

drop service 命令 113

dsedit 实用程序 56

DTM

- 事务管理器 58
- 外部回退 58

dump database ... with shrink_log 命令 17

dump transaction 可恢复操作 10

大对象 (LOB)

- LOB 作为变量 6

- where 子句扩展 7

- 行内 LOB 6

- 行外 LOB 6

- 在 Transact-SQL 语句中使用 LOB 定位符 6

大对象 (LOB) 数据, 压缩 2

带引号的标识符 14

带引号的标识符, 更改 14

带有星号 (*) 的嵌套 select 语句 12

登录

- 配置文件 3

- 映射 56

登录触发器 73

登录名

- 保护 3

动态参数, 分析 9

端到端 Kerberos 鉴定

- 相互鉴定 2

- 消息保密性 2

- 消息完整性 2

对象描述符, 放弃 8

对象所有权, 移交 4

E

enable encrypted columns 配置参数 86

enable functionality group 配置参数 1, 15, 23

enable functionality group 配置参数 11

enable hp posix async i/o 配置参数 23

enable literal autoparm 配置参数 93

enable merge join 配置参数 86

enable metrics capture 配置参数 110

enable semantic partitioning 配置参数 110

enable web services 配置参数 110

enable xml 配置参数 110

exists 和 select 语句 12

F

FIPS 140-2 45

- 和 SSL 107

方差 68

非物化、非空列, 创建 8

非字符, Unicode 14

分布式事务管理 (DTM) 58

分区

- 范围 97
- 和查询处理器 98
- 散列 97
- 循环 97
- 压缩 2
- 支持 97

G

改变表

- log off 参数 5
- 缩减日志 5
- 高可用性 58
 - 和口令 74
- 跟踪标志, 查看当前设置 5
- 更新
 - 用 select for update 锁定 7
- 功能和平台表 33, 41
- 功能可用性, 对于所支持的平台 33, 41
- 共享内存 57
- 共享行内缺省值 8
- 关键字 109
- 管家任务 55
- 归档数据库 35
- 归档数据库访问 65

H

- hashbytes() 函数 87
- hextobigint 函数 112
- 函数
 - charindex 20
 - charlength 20
 - create_locator 20
 - datalength 20
 - dol_downgrade_check 20
 - locator_literal 20
 - locator_valid 20
 - lprofile_id 20
 - lprofile_name 20
 - patindex 20
 - return_lob 20
 - setdata 20
 - show_cached_plan_in_xml 20
 - show_dynamic_params_in_xml 9, 20
 - str 20
 - textptr 20
 - textvalid 20

I

- IBM Tivoli Storage Manager 44
- image 数据类型, 存储 6
- index_name() 函数 87
- Interactive SQL 103
- is_quiesced 函数 112
- isdate 函数 93
- isnumeric 函数 93
- ISO 8601 持续时间 3

J

- 基于函数的索引 99
 - 和计算列 99
- 集合处理 68
- 计算列 99
 - 和基于函数的索引 99
 - 实现与未实现 99
 - 未物化 100
 - 已物化 100
 - 已物化和未物化之间的区别 100
- 加密列, 增强 65
- 监控表 37, 74
 - monCachedProcedures 27
 - monCachedStatement 11, 27
 - monCachePool 27
 - monDeadLock 27
 - monDeviceSpaceUsage 27
 - monErrorLog 27
 - monLockTimeout 27
 - monLockTimeouts 9
 - monOpenObjectActivity 27
 - monOpenPartitionActivity 27
 - monProcess 27
 - monProcessActivity 27
 - monProcessLookup 27
 - monProcessProcedures 27
 - monTableColumns 27
 - monTables 27
 - monWaitClassInfo 27
 - monWaitEventInfo 27
 - 语句高速缓存 75
- 角色, 保护 3
- 仅数据锁定 (DOL) 列 13
- 进程模式 1
- 巨内存页 57

K**Kerberos**

- 端到端鉴定 2

- 鉴定 56

- 统一登录鉴定 2

- kernel mode 配置参数 23

- kernel resource memory 配置参数 23

- 开关, 查看设置 5

- 可变长度的行, 扩展 13

- 可插入鉴定模块 (PAM) 72

- 可滚动游标 100

- 口令 56

- 安全性 73

- 管理 3

- 宽松持久性数据库 42

- 系统过程 48

L

- LDAP 服务器鉴定 55

- LDAPS 55

- like 命令 17

- like 模式匹配 13

- Linux pSeries 功能 57

- LOB 定位符, 在 T-SQL 语句中使用 6

- locator_literal 函数 20

- locator_valid 函数 20

- lock timeout pipe active 配置参数 23

- lock timeout pipe max messages 配置参数 23

- lprofile_id 函数 20

- lprofile_name 函数 20

- 连接, 重新分配的 76

- 链式事务

- 对命令和系统过程所做的更改 12

- 列

- 创建非物化、非空列 8

- 仅数据锁定 (DOL) 13

- 行偏移 13

- 列名称

- 带引号的标识符 14

- 零, 截断 10

M

- max native threads per engine 配置参数 110

- max online q engines 配置参数 86

- max partition degree 配置参数 110

- max resource granularity 配置参数 110

- maximum nesting level 配置参数 62

- merge 命令 10, 17

- metrics elap max 配置参数 86

- metrics exec max 配置参数 86

- metrics lio max 配置参数 86

- metrics pio max 配置参数 86

- min pages for parallel scan 配置参数 86

- mnc_full_index_filter 配置参数 62

- monCachedProcedures 监控表 27

- monCachedStatement 监控表 11, 27

- monCachePool 监控表 27

- monDeadLock 监控表 27

- monDeviceSpaceUsage 监控表 27

- monErrorLog 监控表 27

- monLockTimeout 监控表 27

- monLockTimeouts 监控表 9

- monOpenObjectActivity 监控表 27

- monOpenPartitionActivity 监控表 27, 119

- monProcedureCacheMemoryUsage 监控表 96

- monProcedureCacheModuleUsage 监控表 96

- monProcess 监控表 27

- monProcessActivity 监控表 27

- monProcessLookup 监控表 27

- monProcessProcedures 监控表 27

- monSQLRepActivity 监控表 63

- monSQLRepMisses 监控表 63

- monTableColumns 监控表 27

- monTables 监控表 27

- monTableTransfer 监控表 52

- monWaitClassInfo 监控表 27

- monWaitEventInfo 监控表 27

- 慢速运行的查询 66

- 描述符, 放弃 8

- 命令

- alter database 17

- alter encryption key 17

- alter login 17

- alter login profile 17

- alter table 10

- alter...modify owner 17

- create database 17

- create encryption key 17

- create login 17

- create login profile 17

- create table ... [in row [(length)] | off row]

- 17

- deallocate locator 17

- declare cursor ... [release_locks_on_close]

- 17

declare cursor ... release_locks_on_close 11
 drop encryption key 17
 drop login 17
 drop login profile 17
 dump database ... with shrink_log 17
 dump transaction 10
 like 17
 merge 10, 17
 reorg rebuild 10
 select for update 17
 select into 10
 select into ... [in row [(length)] | off row] 17
 set 17
 truncate lob 17
 where 17
 where 命令 17
 链式事务的更改 12
 模式匹配, like 13

N

number of devices 配置参数 110
 number of disk tasks 配置参数 23
 number of dump threads 配置参数 110
 number of network tasks 配置参数 23
 number of open partitions 配置参数 110
 内存
 临时数据库 42
 数据库 42
 系统过程 48
 内核
 和 CPU 1
 进程模式 1
 线程池 1
 线程化模式 1

O

object_attr 函数 47
 openssl 实用程序 53
 optimization goal 配置参数 110
 optimization timeout limit 配置参数 110

P

partition_id 函数 112
 partition_name 函数 112
 partition_object_id 函数 93
 password_random 函数 61

patindex 函数 20
 pci memory size 配置参数 62
 preupgrade 实用程序 109
 pssinfo 函数 61
 排序顺序 67
 配置参数
 automatic master key access 23
 column default cache size 23
 disable varbinary truncation 10, 23
 enable functionality group 1, 11, 13–15, 23
 enable hp posix async i/o 23
 enable inline default sharing 1
 enable permissive unicode 1
 kernel mode 23
 kernel resource memory 23
 lock timeout pipe active 9, 23
 lock timeout pipe max messages 9, 23
 number of disk tasks 23
 number of network tasks 23
 quoted identifier enhancements 1
 select for update 1
 sp_opt_querystats. 16
 streamlined dynamic SQL 1, 15
 unicode noncharacters 14
 配置文件, 登录 3

Q

嵌套 select 语句中的星号 (*) 12
 全局变量 110
 @@lwpid 9, 32
 @@plwpid 9, 32
 缺省值, 共享 8

R

reinit_descriptors 子命令 55
 Replication Server 55
 reserved_pages 函数 112
 return_lob 函数 20
 Rivest-Shamir-Adleman (RSA) 密钥对重新生成期间 3
 row_count 函数 112
 rtm thread idle wait period 配置参数 110
 日志, 缩减 5

S

- select for update
 - command 17
 - 和游标 7
 - 锁定行 7
- select into ... [in row [(length)] | off row] 命令 17
- select 语句, 嵌套 12
- send doneinprov tokens 配置参数 86
- set 命令 17, 66, 80
- set...opt criteria 命令 90
- setdata 函数 20
- show_cached_plan_in_xml 函数 7, 20
- show_dynamic_params_in_xml 函数 9, 20
- showplan 更改 106
- showplan_in_xml 函数 112
- sp_dboption 系统过程 21
- sp_deletesmobj 存储过程 48
- sp_displaylogin 系统过程 21
- sp_displayroles 系统过程 21
- sp_downgrade 系统过程 81
- sp_encryption 系统过程 21
- sp_helpcomputedcolumn 存储过程 114
- sp_helpconstraint 系统过程 21
- sp_helpprotect 系统过程 21
- sp_helpuser 系统过程 21
- sp_jreconfig 系统过程 61
- sp_listener 存储过程 56
- sp_locklogin 系统过程 21
- sp_merge_dup_inline_default 系统过程 21
- sp_opt_querystats 配置参数 16
- sp_opt_querystats 实用程序 16
- sp_opt_querystats 系统过程 21
- sp_passwordpolicy 系统过程 21
- sp_querysmobj 存储过程 48
- sp_securityprofile 系统过程 21
- sp_serveroption 系统过程 21
- sp_showoptstats 系统过程 21
- sp_spaceusage 系统过程 81
- sp_tabsuspectptn 系统过程 61
- sp_version 存储过程 114
- sproc optimize timeout limit 配置参数 93
- spt_TableTransfer 系统表 52
- SQL
 - 轻量过程 (LWP) 15
 - 语句复制 55
- sqlsrvr 实用程序 109
- ssel_message 函数 112
- SSL 认证, 公用名 56
- SSL 支持 73
- startup delay 配置参数 93
- statement cache size 配置参数 110
- str 函数 20
 - 填充字符字段 7
- streamlined dynamic SQL 配置参数 15
- Sybase Central 103
- sybdiag 实用程序 15, 26
- sybperf 实用程序 26
- SySAM 更改 105
- sysattributes 系统表 25
- syscacheinfo 系统表 25
- syscachepoolinfo 系统表 25
- syscolumns 系统表 25
- syscomments 系统表 52
- sysdatabases 系统表 52
- sysdevices 系统表 52
- syslogins 系统表 25
- sysoptions 5
- sysoptions 系统表 25, 84
- syspartitionkeys 系统表 115
- syspartitions 系统表 115
- syspoolinfo 系统表 25
- syssservers 系统表 25
- sysssrvroles 系统表 25
- sysstatistics flush interval 配置参数 110
- 散列表 57
- 审计
 - 更改 54
 - 失败的登录名 74
 - 隐藏的口令 73
- 实用程序
 - sp_opt_querystats 16
 - sybdiag 15, 26
 - sybperf 26
- 矢量集合 69
- 事务
 - 链式 12
 - 模式 12
 - 用 select for update 锁定 7
- 视图
 - 带引号的标识符 14
- 数据库转储 65
- 数据类型
 - unitext 100
- 数据压缩
 - 常规数据 2
 - 大对象 (LOB) 2
- 双主系统密钥 3

缩减日志空间 5
索引名称
 带引号的标识符 14
锁
 在游标关闭时释放游标锁 11
锁超时, 监控 9

T

T-SQL 语句, 使用 LOB 定位符 6
tempdb 组 46
text 数据类型, 存储 6
textptr 函数 20
textvalid 函数 20
Tivoli Storage Manager 44
 系统过程 48
tran_dumpable_status 函数 112
transfer table 命令 50
truncate lob 命令 17
统计集合函数 67
统计信息
 收集 104
 用 sp_showoptstats 查看 10

U

Unicode
 非字符, 忽略 14
 增强 102
unitext 数据类型, 存储 6
Unitext 支持 100
update statistics 命令 104
update table statistics 命令 113
used_pages 函数 112

W

Web 服务 103
where 命令 17
where 子句, 扩展 7
外部登录口令 4
尾随零, 截断 10
文字参数化 91
问号, 动态参数 9
无符号整数数据类型 101

X

XML
 查看统计信息和直方图 10

国际化支持 102
模式支持 102
显示高速缓存计划 7
增强 102
xmltable() 函数 76
xmlvalidate 函数 112
系统表
 sysattributes 25
 syscacheinfo 25
 syscachepoolinfo 25
 syscolumns 25
 syscomments 25
 syslogins 25
 sysoptions 25
 syspoolinfo 25
 syssservers 25
 sysssrvroles 25
 行锁定 98
系统过程
 sp_dboption 10, 21
 sp_displaylogin 21
 sp_displayroles 21
 sp_encryption 21
 sp_helpconstraint 21
 sp_helpprotect 21
 sp_helpuser 21
 sp_locklogin 21
 sp_merge_dup_inline_default 21
 sp_opt_querystats 21
 sp_passwordpolicy 21
 sp_securityprofile 21
 sp_serveroption 21
 sp_showoptstats 10, 21
 带引号的标识符 14
 链式事务的更改 12
 系统加密口令, 已替换 3
 系统密钥
 双主 3
 主 3
 线程池 1
 线程化模式 1
 行
 传送 10
 可变长度 13
 行级锁定, 系统表 75
 行内 LOB 列, 存储 6
 行内缺省值, 共享 8
 行锁定的系统目录 98
 行外列, 存储 6

索引

虚拟散列表 57

Y

压缩级别 43

压缩数据 2

延迟, 降低 15

延迟编译 66

 存储过程 71

延迟名称解析 44

以 XML 格式显示高速缓存计划 7

隐藏文本 4

应用程序功能配置组 1

用户定义的函数 77

优化程序, 延迟编译 66

优化程序设置, 查看 72

游标

 可滚动 100

 用 `select for update` 锁定 7

 在游标关闭时释放游标锁 11

语句高速缓存 7

 保存抽象计划 4

元数据高速缓存, 放弃描述符 8

员工生命周期管理 4

Z

增量数据传输 45

整数标识, 数据类型作为标识值 102

直方图

 插值 71

 选择性估计 72

直方图 `sp_showoptstats` 查看

 用 `sp_showoptstats` 10

中括号 ([]) 和 `like` 模式匹配 13

主系统密钥 3

字符集, 简体中文和日文 67

字符字段, 使用 `str` 填充 7

组, `tempdb` 46

组件集成系统 (CIS)

 端到端 Kerberos 鉴定 2