

SYBASE®

Historical Server 用户指南

**Adaptive Server® Enterprise**

15.5

文档 ID: DC33435-01-1550-01

最后修订日期: 2009 年 10 月

版权所有 © 2010 Sybase, Inc. 保留所有权利。

除非在新版本或技术声明中另有说明, 否则本出版物适用于 Sybase 软件和任何后续版本。本文档中的信息如有更改, 恕不另行通知。此处说明的软件按许可协议提供, 其使用和复制必须符合该协议的条款。

若要订购附加文档, 美国和加拿大的客户请拨打客户服务部门电话 (800) 685-8225 或发传真至 (617) 229-9845。

持有美国许可协议的其他国家 / 地区的客户可通过上述传真号码与客户服务部门联系。所有其他国际客户请与 Sybase 子公司或当地分销商联系。仅在定期安排的软件发布日期提供升级。未经 Sybase, Inc. 的事先书面许可, 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。

Sybase 商标可在位于 <http://www.sybase.com/detail?id=1011207> 的“Sybase 商标页” (Sybase trademarks page) 处进行查看。Sybase 和列出的标记均是 Sybase, Inc. 的商标。® 表示已在美国注册。

Java 和所有基于 Java 的标记都是 Sun Microsystems, Inc. 在美国和其他国家 / 地区的商标或注册商标。

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。

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

关于本手册 .....	vii	
<b>第 1 章</b>	<b>简介</b> .....	<b>1</b>
	Adaptive Server Enterprise Monitor 说明 .....	1
	Adaptive Server Monitor 组件 .....	1
	Adaptive Server Enterprise Monitor 体系结构 .....	2
	Historical Server 概念 .....	3
<b>第 2 章</b>	<b>配置 Historical Server</b> .....	<b>7</b>
	Historical Server 配置概念 .....	7
	Historical Server 控制文件和主目录 .....	7
	操作系统启动帐户 .....	8
	Historical Server 超级用户帐户 .....	9
	Sybase Open Client/Server 连接 .....	9
	UNIX 平台上的初始配置 .....	9
	UNIX 平台的前提条件 .....	10
	UNIX 平台上的配置过程 .....	10
	Windows 平台上的初始配置 .....	12
	Windows 平台的前提条件 .....	13
	Windows 上的安装结果 .....	13
	Windows 上的配置过程 .....	13
	设置 Historical Server 启动参数 .....	17
	作用 .....	17
	语法 .....	17
	参数 .....	18
	配置多个 Historical Server 实例 .....	20
	何时创建 Historical Server 的多个实例 .....	20
	在 UNIX 平台上配置附加的 Historical Server .....	21
	在 Windows 上配置附加的 Historical Server .....	21

<b>第 3 章</b>	<b>启动和停止 Historical Server .....</b>	<b>29</b>
	启动和停止 UNIX 平台上的 Historical Server .....	29
	启动 UNIX 上的 Historical Server .....	29
	停止 UNIX 上的 Historical Server .....	30
	在 Windows 上启动和停止 Historical Server .....	32
	在 Windows 上启动 Historical Server .....	33
	确定注册表中的启动参数 .....	33
	检验 Historical Server 是否正在运行 .....	34
	在 Windows 上停止 Historical Server .....	34
<b>第 4 章</b>	<b>命令参考 .....</b>	<b>37</b>
	命令摘要 .....	37
	命令语法 .....	38
	命令状态和错误 .....	39
	将脚本文件输入到 Historical Server .....	39
	连接到 Historical Server .....	40
	连接前的假定条件 .....	40
	连接方法 .....	40
	Historical Server 活动所需的权限 .....	41
	互斥会话 .....	42
	Historical Server 命令 .....	43
	hs_create_alarm .....	43
	hs_create_filter .....	47
	hs_create_playback_session .....	50
	hs_create_playback_view .....	57
	hs_create_recording_session .....	58
	hs_create_view .....	61
	hs_delete_data .....	62
	hs_initiate_playback .....	63
	hs_initiate_recording .....	63
	hs_list .....	64
	hs_playback_sample .....	68
	hs_shutdown .....	71
	hs_status .....	72
	hs_terminate_playback .....	73
	hs_terminate_recording .....	74
<b>第 5 章</b>	<b>数据文件和输出选项 .....</b>	<b>75</b>
	Historical Server 数据文件概述 .....	75
	对 Historical Server 文件的描述 .....	75
	文件权限 .....	76
	一般文件格式 .....	76

控制文件 .....	76
标头记录 .....	77
会话控制记录 .....	78
视图控制记录 .....	79
数据项控制记录 .....	79
报警控制记录 .....	80
过滤器控制记录 .....	80
数据文件 .....	81
错误消息文件 .....	81
脚本文件 .....	82
脚本文件表名 .....	82
脚本文件表列名 .....	83
传递脚本文件命令 .....	83
脚本应用示例 .....	83
批量复制示例 .....	84
示例 .....	85
Cut 实用程序示例 .....	86
允许 Historical Server 将监控数据输出到数据库 .....	87
设置接收 Adaptive Server .....	87
启动 Historical Server .....	87
查看数据 .....	88
删除 Historical Server 会话 .....	91
报告错误 .....	91
<b>附录 A</b>	<b>数据项 .....</b>
	<b>93</b>
	有关数据项及其定义的表 .....
	93
<b>附录 B</b>	<b>定义记录会话视图的规范 .....</b>
	<b>109</b>
	键和结果的定义 .....
	109
	设计记录会话视图 .....
	110
	使用进程 ID .....
	110
	使用应用程序名 .....
	111
	视图中空行与没有行 .....
	111
	有效键和结果数据项组合表 .....
	112
	有效组合示例 .....
	134
	无效组合示例 .....
	135
	数据项有效统计类型表 .....
	135
<b>附录 C</b>	<b>定义回放视图的规范 .....</b>
	<b>143</b>
	汇总级别详细信息 .....
	143
	原始回放 .....
	143
	实际回放 .....
	144
	完整回放 .....
	145

使用用户定义间隔的回放 .....	145
汇总间隔摘要 .....	146
设计回放视图 .....	147
指定输入会话的规则 .....	147
回放视图与输入视图间的关系 .....	147
定义视图的规则 .....	148
有关回放视图的数据项要求的表 .....	150
关于某些数据项的其它信息 .....	155
使用“时间戳”、“时间戳日期时间”和“经历时间” .....	155
<b>附录 D</b>	
<b>记录会话视图的示例 .....</b>	<b>157</b>
高速缓存性能摘要 .....	158
数据库对象锁状态 .....	159
数据库对象页 I/O .....	159
各个高速缓存的数据高速缓存活动 .....	160
记录会话的数据高速缓存统计信息 .....	160
采样间隔期间的数据高速缓存统计信息 .....	161
记录会话的设备 I/O .....	161
采样间隔期间的设备 I/O .....	162
设备 I/O 性能摘要 .....	162
引擎活动 .....	163
锁性能摘要 .....	163
记录会话的网络活动 .....	163
采样间隔期间的网络活动 .....	164
网络性能摘要 .....	164
页 I/O .....	165
记录会话的过程高速缓存统计信息 .....	165
采样间隔期间的过程高速缓存统计信息 .....	166
过程页 I/O .....	166
进程活动 .....	167
进程数据库对象页 I/O .....	167
进程的锁明细 .....	168
进程的页 I/O 明细 .....	169
进程锁 .....	169
进程页 I/O .....	170
进程状态摘要 .....	170
进程存储过程页 I/O .....	171
服务器性能摘要 .....	171
存储过程活动 .....	172
事务活动 .....	173
<b>索引 .....</b>	<b>175</b>

# 关于本手册

## 读者

本书面向负责以下工作的人员：

- 配置和管理 Historical Server
- 使用 Historical Server 监控 Adaptive Server 的性能

## 如何使用本手册

本书包含以下几章：

- [第 1 章 “简介”](#) 概要介绍了 Adaptive Server<sup>®</sup> Enterprise Monitor 并阐述了 Historical Server 的一些基本概念。
- [第 2 章 “配置 Historical Server”](#) 说明了如何为 UNIX 和 Windows 平台完成 Historical Server 初始配置或更改配置。
- [第 3 章 “启动和停止 Historical Server”](#) 说明了如何在 UNIX 和 Windows 平台上启动和停止 Historical Server。
- [第 4 章 “命令参考”](#) 说明了用于操作 Historical Server 的 isql 命令接口。
- [第 5 章 “数据文件和输出选项”](#) 描述了 Historical Server 创建的文件以及如何访问其中的数据，包括如何使用 bcp 实用程序将数据装载到 Adaptive Server 表中。
- [附录 A “数据项”](#) 提供了对可通过 Historical Server 获得的数据项的描述。
- [附录 B “定义记录会话视图的规范”](#) 描述了用于定义记录会话视图的数据项名称和统计类型的有效组合。
- [附录 C “定义回放视图的规范”](#) 描述了用于定义回放会话视图的数据项名称和统计类型的有效组合。
- [附录 D “记录会话视图的示例”](#) 提供了记录会话视图的示例。

## 相关文档

Adaptive Server<sup>®</sup> 文档集包括：

- 针对所用平台的发行公告 — 包含未能及时写入手册的最新信息。  
最新版本的发行公告可能已经推出。要了解本产品 CD 发行以后增加的重要产品或文档信息，请使用 Sybase Product Manuals 网站。

- 
- 针对所用平台的安装指南 — 介绍所有 Adaptive Server 产品及相关 Sybase 产品的安装、升级和某些配置过程。
  - New Feature Summary (《新增功能摘要》) — 介绍 Adaptive Server 中的新增功能, 为支持这些功能所做的系统更改, 以及可能会影响现有应用程序的更改。
  - Active Messaging Users Guide (《Active Messaging 用户指南》) — 介绍如何使用 Active Messaging 功能捕获 Adaptive Server Enterprise 数据库中的事务 (数据更改), 并将它们作为事件实时发送给外部应用程序。
  - 《组件集成服务用户指南》— 说明如何使用组件集成服务功能来连接远程 Sybase 和非 Sybase 数据库。
  - 针对所用平台的《配置指南》— 提供执行特定配置任务的操作说明。
  - 《词汇表》— 定义 Adaptive Server 文档中使用的技术术语。
  - 《Historical Server 用户指南》— 介绍如何使用 Historical Server 从 Adaptive Server 获取性能信息。
  - 《Adaptive Server Enterprise 中的 Java》— 介绍在 Adaptive Server 数据库中如何安装 Java 类, 如何将它们用作数据类型、函数及存储过程。
  - 《Job Scheduler 用户指南》— 提供有关如何使用命令行或图形用户界面 (GUI) 在本地或远程 Adaptive Server 上进行安装和配置以及创建和调度作业的操作说明。
  - 《迁移技术指南》— 介绍迁移到其它 Adaptive Server 版本所需的策略和工具。
  - 《Monitor Client Library 程序员指南》— 介绍如何编写访问 Adaptive Server 性能数据的 Monitor Client Library 应用程序。
  - 《Monitor Server 用户指南》— 介绍如何使用 Monitor Server 从 Adaptive Server 获取性能统计信息。
  - Monitoring Tables Diagram (《监控表框图》) — 以张贴画的形式阐明监控表及其实体关系。大图只提供印刷品; 采用 PDF 格式时提供缩略图。



- Performance and Tuning Series (《性能和调优系列》) — 是一系列丛书, 介绍如何调节 Adaptive Server 以获得最佳性能:
  - Basics (《基础知识》) — 包含通晓和研究 Adaptive Server 中的性能问题需具备的基础知识。
  - Improving Performance with Statistical Analysis (《利用统计分析改进性能》) — 介绍 Adaptive Server 如何存储和显示统计信息, 及如何使用 `set statistics` 命令分析服务器统计信息。
  - Locking and Concurrency Control (《锁定和并发控制》) — 介绍如何使用锁定方案来改进性能, 以及如何选择索引以最大限度地减少并发。
  - Monitoring Adaptive Server with `sp_sysmon` (《使用 `sp_sysmon` 监控 Adaptive Server》) — 讨论如何使用 `sp_sysmon` 监控性能。
  - Monitoring Tables (《监控表》) — 介绍如何从 Adaptive Server 的监控表中查询统计信息和诊断信息。
  - Physical Database Tuning (《物理数据库调优》) — 介绍如何管理物理数据放置、为数据分配的空间及临时数据库。
  - Query Processing and Abstract Plans (《查询处理和抽象计划》) — 介绍优化程序如何处理查询以及如何使用抽象计划更改某些优化程序计划。
- 《快速参考指南》— 这是一本袖珍手册, 完整地列出了各种命令、函数、系统过程、扩展系统过程、数据类型和实用程序的名称和语法 (该手册在用 PDF 格式阅读时采用正常大小)。
- 《参考手册》— 是一系列丛书, 包含详细的 Transact-SQL<sup>®</sup> 信息:
  - 《构件块》— 讨论数据类型、函数、全局变量、表达式、标识符、通配符和保留字。
  - 《命令》— 提供了命令的文档资料。
  - 《过程》— 介绍系统过程、目录存储过程、系统扩展存储过程及 `dbcc` 存储过程。
  - 《表》— 讨论系统表、监控表及 `dbcc` 表。

- 
- 《系统管理指南》—
    - 《卷 1》— 介绍系统管理的基础知识，包括对配置参数、资源问题、字符集、排序顺序的说明以及有关诊断系统问题的操作说明。《卷 1》的第二部分深入讨论了安全性管理。
    - 《卷 2》— 包括管理物理资源、镜像设备、配置内存和数据高速缓存、管理多处理器服务器和用户数据库、装入和卸下数据库、创建和使用段、使用 `reorg` 命令及检查数据库一致性的操作说明和指导。《卷 2》的后半部分介绍了如何备份和恢复系统数据库和用户数据库。
  - System Tables Diagram（《系统表框图》）— 以张贴画的形式阐明系统表及其实体关系。大图只提供印刷品；采用 PDF 格式时提供缩略图。
  - 《Transact-SQL 用户指南》— 提供有关 Transact-SQL 这一 Sybase 关系数据库语言增强版的文档资料。此指南可用作数据库管理系统初级用户的教科书，其中还包含有关 `pubs2` 和 `pubs3` 示例数据库的详细说明。
  - Troubleshooting Series（《故障排除系列》）—
    - Troubleshooting: Error Messages Advanced Resolutions（《故障排除：错误消息高级解析》）— 包含针对您可能遇到的问题的故障排除过程。此处讨论的问题是 Sybase 技术支持部门的员工最常听到的问题。
    - Troubleshooting and Error Messages Guide（《故障排除和错误消息指南》）— 包含有关如何解除最常出现的 Adaptive Server 错误消息的详细说明。
  - 《加密列用户指南》— 介绍了如何利用 Adaptive Server 配置和使用加密列。
  - 《内存数据库用户指南》— 介绍如何配置和使用内存数据库。
  - Using Adaptive Server Distributed Transaction Management Features（《使用 Adaptive Server 分布式事务管理功能》）— 介绍如何在分布式事务处理环境中配置、使用 Adaptive Server DTM 功能以及如何排除其中的故障。
  - 《将 Backup Server 与 IBM® Tivoli® Storage Manager 配合使用》— 说明如何设置和使用 IBM Tivoli Storage Manager 以创建 Adaptive Server 备份。

- 《在高可用性系统中使用 Sybase 故障切换》— 提供有关使用 Sybase 的故障切换功能将 Adaptive Server 配置为高可用性系统中的协同服务器的操作说明。
- Unified Agent and Agent Management Console (《Unified Agent 和 Agent Management Console》) — 介绍用于提供管理、监控和控制分布式 Sybase 资源的运行时服务的 Unified Agent。
- 《实用程序指南》— 提供有关在操作系统级别执行的 Adaptive Server 实用程序 (如 isql 和 bcp) 的文档资料。
- 《Web 服务用户指南》— 介绍如何配置、使用 Adaptive Server Web 服务以及如何排除其中的故障。
- 《适用于 CICS、Encina 和 TUXEDO 的 XA 接口集成指南》— 提供有关将 Sybase DTM XA 接口与 X/Open XA 事务管理器配合使用的说明。
- 《Adaptive Server Enterprise 中的 XML 服务》— 介绍了 Sybase 本机 XML 处理器、Sybase 基于 Java 的 XML 支持及数据库中的 XML, 并提供了 XML 服务中可用的查询和映射函数的文档资料。

#### 其它信息来源

使用 Sybase Getting Started CD、SyBooks™ CD 和 Sybase® Product Manuals 网站可以了解有关产品的更多信息:

- Getting Started CD 包含 PDF 格式的发行公告和安装指南, 还可能包含 SyBooks CD 中未收纳的其它文档或更新信息。它随软件一起提供。若要阅读或打印 Getting Started CD 上的文档, 需要使用 Adobe Acrobat Reader, 该软件可以使用 CD 上提供的链接从 Adobe 网站免费下载。
- SyBooks CD 含有产品手册, 它随软件一起提供。基于 Eclipse 的 SyBooks 浏览器使您能够以易于使用的、基于 HTML 的格式阅读手册。

有些文档可能是以 PDF 格式提供的, 您可以通过 SyBooks CD 上的 PDF 目录访问这些文档。若要阅读或打印 PDF 文件, 您需要使用 Adobe Acrobat Reader。

有关安装和启动 SyBooks 的说明, 请参见 Getting Started CD 上的《SyBooks 安装指南》或 SyBooks CD 上的 *README.txt* 文件。

- Sybase Product Manuals 网站是 SyBooks CD 的联机版本, 您可以使用一种标准 Web 浏览器来访问它。除了产品手册之外, 还可以找到有关 EBFs/Maintenance (EBF/ 维护)、Technical Documents (技术文档)、Case Management (案例管理)、Solved Cases (解决的案例)、Newsgroups (新闻组) 和 Sybase Developer Network (Sybase 开发人员网络) 的链接。

---

若要访问 Sybase Product Manuals 网站，请转到位于 <http://www.sybase.com/support/manuals/> 的“产品手册” (Product Manuals)。

**Web 上的 Sybase 认证** Sybase 网站上的技术文档不断在更新。

❖ **查找有关产品认证的最新信息**

- 1 将 Web 浏览器定位到位于 <http://www.sybase.com/support/techdocs/> 的“技术文档” (Technical Documents)。
- 2 单击“认证报告” (Certification Report)。
- 3 在“认证报告” (Certification Report) 过滤器中选择相应的产品、平台和时间范围，然后单击“查找” (Go)。
- 4 单击“认证报告” (Certification Report) 标题显示此报告。

❖ **查找有关组件认证的最新信息**

- 1 将 Web 浏览器定位到位于 <http://certification.sybase.com/> 的“可用性和认证报告” (Availability and Certification Reports)。
- 2 在“按基本产品搜索” (Search by Base Product) 下选择产品系列和产品，或在“按平台搜索” (Search by Platform) 下选择平台和产品。
- 3 选择“搜索” (Search) 以显示所选项目的可用性和认证报告。

❖ **创建个性化的 Sybase 网站（包括支持页）视图**

建立 MySybase 配置文件。MySybase 是一项免费服务，用于创建个性化的 Sybase 网页视图。

- 1 将 Web 浏览器定位到位于 <http://www.sybase.com/support/techdocs/> 的“技术文档” (Technical Documents)。
- 2 单击“我的 Sybase” (MySybase) 并创建 MySybase 配置文件。

❖ **查找有关产品认证的最新信息**

- 1 将 Web 浏览器定位到位于 <http://www.sybase.com/support/techdocs/> 的“技术文档” (Technical Documents)。
- 2 单击“认证报告” (Certification Report)。
- 3 在“认证报告” (Certification Report) 过滤器中选择相应的产品、平台和时间范围，然后单击“查找” (Go)。
- 4 单击“认证报告” (Certification Report) 标题显示此报告。

#### ❖ 查找有关组件认证的最新信息

- 1 将 Web 浏览器定位到位于 <http://certification.sybase.com/> 的“可用性和认证报告” (Availability and Certification Reports)。
- 2 在“按基本产品搜索” (Search by Base Product) 下选择产品系列和产品，或在“按平台搜索” (Search by Platform) 下选择平台和产品。
- 3 选择“搜索” (Search) 以显示所选项目的可用性和认证报告。

#### ❖ 创建个性化的 Sybase 网站（包括支持页）视图

建立 MySybase 配置文件。MySybase 是一项免费服务，用于创建个性化的 Sybase 网页视图。

- 1 将 Web 浏览器定位到位于 <http://www.sybase.com/support/techdocs/> 的“技术文档” (Technical Documents)。
- 2 单击“我的 Sybase” (MySybase) 并创建 MySybase 配置文件。

## Sybase EBF 和软件维护

#### ❖ 查找有关 EBF 和软件维护的最新信息

- 1 将 Web 浏览器定位到位于 <http://www.sybase.com/support> 的“Sybase 支持页” (Sybase Support Page)。
- 2 选择“EBF/ 维护” (EBFs/Maintenance)。如果出现提示信息，请输入您的 MySybase 用户名和口令。
- 3 选择一个产品。
- 4 指定时间范围并单击“查找” (Go)。即会显示 EBF/ 维护版本的列表。

锁形图标表示因为您没有注册为“技术支持联系人” (Technical Support Contact)，因此您没有某些 EBF/ 维护版本的下载授权。如果您尚未注册，但拥有 Sybase 代表提供的或通过支持合同获得的有效信息，请单击“编辑角色” (Edit Roles) 将“技术支持联系人” (Technical Support Contact) 角色添加到 MySybase 配置文件中。

- 5 单击信息图标可显示 EBF/ 维护报告，单击产品说明可下载软件。

## 约定

以下各部分将说明在本手册中使用的约定。

SQL 是一种形式自由的语言。没有规定每一行中的单词数量或者必须折行的地方。然而，为便于阅读，本手册中所有示例和大多数语法语句都经过了格式设置，以便语句的每个子句都在一个新行上开始。有多个成分的子句会扩展到其它行，这些行会有缩进。复杂命令使用已修改的 Backus Naur Form (BNF) 表示法进行了格式处理。

表 1 说明本手册中出现的语法语句的约定：

**表 1：本手册的字体和语法约定**

元素	示例
命令名、过程名、实用程序名和其它关键字用 sans serif 字体显示。	<code>select</code> <code>sp_configure</code>
数据库名和数据库类型用 sans serif 字体显示。	<code>master</code> 数据库
书名采用正常字体并加书名号；文件名、变量和路径名用斜体显示。	《系统管理指南》 <i>sql.ini</i> 文件 <i>column_name</i> <i>SYBASE/ASE</i> 目录
变量（即代表您要填充的值的词语）作为查询或语句的一部分出现时用斜体的 Courier 字体显示。	<code>select column_name</code> <code>from table_name</code> <code>where search_conditions</code>
将小括号作为命令的一部分键入。	<code>compute row_aggregate (column_name)</code>
双冒号加等号表示语法是用 BNF 表示法编写的。请勿输入此符号。表示“被定义为”。	<code>::=</code>
大括号表示必须至少选择括号中的一个选项。不要输入大括号。	<code>{cash, check, credit}</code>
中括号表示可以选择括号中的一个或多个选项，也可不选。不要输入中括号。	<code>[cash   check   credit]</code>
逗号表示可以选择任意多个显示的选项。可用逗号作为命令的一部分来分隔选项。	<code>cash, check, credit</code>
竖线 ( ) 表示只可选择所显示的选项中的一个。	<code>cash   check   credit</code>
省略号 (...) 表示可以将最后一个单元重复任意次。	<code>buy thing = price [cash   check   credit]</code> <code>[, thing = price [cash   check   credit]]...</code> 您必须至少购买一件产品，并给出其价格。可以选择一种付款方式：方括号中的选项之一。还可选择购买其它物品：可根据需要购买任意数量的物品。对于要买的每种产品，给出其名称、价格和付款方式（可选）。

- 语法语句（显示命令的语法和所有选项）显示如下：

`sp_dropdevice [device_name]`

对于具有多个选项的命令：

```
select column_name
from table_name
where search_conditions
```

在语法语句中，关键字（命令）采用常规字体，而标识符为小写。斜体表示用户提供的內容。

- Transact-SQL 命令用法的示例显示如下：

```
select * from publishers
```

- 计算机输出的示例显示如下：

pub_id	pub_name	city	state
0736	New Age Books	Boston	MA
0877	Binnet & Hardley	Washington	DC
1389	Algodata Infosystems	Berkeley	CA

(3 rows affected)

本手册中的大多数示例都用小写显示。不过，输入 Transact-SQL 关键字时可以忽略大小写。例如，SELECT、Select 和 select 之间没有区别。

Adaptive Server 是否区分数据库对象（如表名）的大小写，取决于安装在 Adaptive Server 上的排序顺序。通过重新配置 Adaptive Server 的排序顺序，可改变单字节字符集的区别大小写设置。有关详细信息，请参见《系统管理指南》。

## 辅助功能特性

本文档具有为提供辅助功能而进行了专门设计的 HTML 版本。可以利用适应性技术（如屏幕阅读器）浏览 HTML 文档，也可以用屏幕放大器进行查看。

Adaptive Server HTML 文档已经过测试，符合美国政府“第 508 节辅助功能”的要求。符合“第 508 节”的文档一般也符合美国之外的辅助功能原则，如 World Wide Web 协会 (W3C) 针对网站的原则。

---

**注释** 可能需要配置辅助功能工具，以便获得最佳使用效果。某些屏幕阅读器按照大小写来辨别文本，例如将“ALL UPPERCASE TEXT”看作首字母缩写，而将“MixedCase Text”看作单词。对工具进行配置，规定语约定，您可能会感觉更方便。有关工具的信息，请查阅文档。

---

有关 Sybase 如何支持辅助功能的信息，请参见位于 <http://www.sybase.com/accessibility> 的“Sybase 辅助功能” (Sybase Accessibility)。“Sybase 辅助功能” (Sybase Accessibility) 站点包括指向“第 508 节”和 W3C 标准相关信息的链接。

## 如果需要帮助

对于购买了支持合同的客户安装的每一个 Sybase 产品，都会有一位或多位指定人员获得与 Sybase 技术支持部门联系的授权。如果使用手册或联机帮助不能解决问题，可让指定人员与 Sybase 技术支持部门联系或与所在区域的 Sybase 子公司联系。





主题	页码
<a href="#">Adaptive Server Enterprise Monitor 说明</a>	1
<a href="#">Adaptive Server Enterprise Monitor 体系结构</a>	3

## Adaptive Server Enterprise Monitor 说明

使用 Adaptive Server Enterprise Monitor™（简称 Adaptive Server Monitor），可以实时地或以历史数据收集模式监控 Adaptive Server 性能。系统管理员可使用此信息找出潜在的资源瓶颈、研究当前问题以及调优以获得更好的性能。Adaptive Server Monitor 提供反馈来进行几个层次上的性能调优：

- Adaptive Server 配置
- 数据库设计
- 应用程序和存储过程中的 SQL 语句

## Adaptive Server Monitor 组件

Adaptive Server Monitor 由四个收集或显示 Adaptive Server 性能数据的组件组成：

- Adaptive Server Enterprise Monitor Server（简称 Monitor Server）— 该服务器实时收集 Adaptive Server 性能数据，并将数据提供给其它 Adaptive Server Monitor 组件。Monitor Server 是一个 Sybase Open Server 应用程序。
- Adaptive Server Enterprise Monitor Historical Server（简称 Historical Server）— 该服务器从 Monitor Server 获得 Adaptive Server 性能数据，并且将这些数据保存在文件中供以后分析。Historical Server 是一个 Sybase Open Server 应用程序。

- 用于 Sybase Central™ 的 Adaptive Server Enterprise 插件中的监控器 (Monitor Viewer) — 这些监控器从 Monitor Server 获得 Adaptive Server 的性能数据，并以表和图的形式实时显示这些数据。
- Adaptive Server Enterprise Monitor Client Library (简称 Monitor Client Library) — 用户可使用 Monitor Server 和 Historical Server 的该应用程序编程接口 (API) 开发监控应用程序。Historical Server，以及用于 Sybase Central 的 Adaptive Server Enterprise 插件中的监控器都是 Monitor Client Library 应用程序。

## Adaptive Server Enterprise Monitor 体系结构

Adaptive Server 将性能数据保存在 Monitor Server 可以读取的一个共享内存区域中。因为采用了这种共享内存的技术，所以必须在受监控的 Adaptive Server 所在的同一台计算机上安装和运行 Monitor Server。Adaptive Server 和 Monitor Server 之间存在一对一的关系。有关 Monitor Server 的详细信息，请参见《Sybase Adaptive Server Enterprise Monitor Server 用户指南》。

Monitor Client Library 应用程序从 Monitor Server 获取 Adaptive Server 性能统计信息。这些应用程序是 Monitor Server 的客户端。出于性能方面的考虑，Sybase 建议不要在运行 Adaptive Server 及对应的 Monitor Server 的计算机上运行 Monitor Client Library 应用程序，而应在其它的计算机上运行。有关详细信息，请参见《Sybase Adaptive Server Enterprise Monitor Client Library 程序员指南》。

用于 Sybase Central 的 Adaptive Server Enterprise 插件包括一组监控器，这些监控器从不同的角度以多种详细级别显示 Adaptive Server 的资源使用情况。每个打开的监控器都是一个独立的应用程序，与 Monitor Server 都有唯一的客户端连接。Sybase Central 中，每个 Adaptive Server 安装都有它自己的 Monitors 文件夹，里面包含一系列监控器对象。

Historical Server 从 Monitor Server 收集性能信息，并将数据保存到文件以进行随后的分析。Historical Server 界面让用户指定要收集的数据和期望的时间段。它们也包括历史数据回放功能。这些接口是：

- isql 中的命令接口。有关详细信息，请参见《Sybase Adaptive Server Enterprise Monitor Historical Server 用户指南》。
- 使用 Monitor Client Library 的编程接口。有关详细信息，请参见《Sybase Adaptive Server Enterprise Monitor Client Library 程序员指南》。

## Historical Server 概念

本节介绍以下 Historical Server 概念：

- [记录会话](#)
- [回放会话](#)
- [视图](#)
- [数据项和统计类型](#)

### 记录会话

记录会话收集 Adaptive Server 性能数据，并将该数据存储在文件中供以后分析。以下是记录会话的一些属性：

- **Monitor Server 名** — 它通过关联来定义正在监控其性能的 Adaptive Server。
- **采样间隔** — 该属性定义收集性能数据的频率。
- **视图、报警和过滤器** — 视图和过滤器定义要收集的数据。报警定义指定数据项命中预定义阈值时可进行的操作。
- **开始时间和结束时间** — 指定的这两个值定义要收集哪个时间段的数据。

若要创建一个记录会话，请使用 Historical Server `isql` 命令接口中的一系列命令：

- [hs\\_create\\_recording\\_session](#)
- [hs\\_create\\_view](#)
- [hs\\_create\\_filter](#)（可选）
- [hs\\_create\\_alarm](#)（可选）
- [hs\\_initiate\\_recording](#)

创建记录会话时，Historical Server 给它分配一个会话 ID。可以使用 `hs_list` 命令列出已定义的记录会话的会话 ID。`hs_list` 还可以显示包括视图名以及视图中的数据项、报警和过滤器的完整记录会话定义。

Historical Server 在它的控制文件中存储这些记录会话定义，它的控制文件位于 Historical Server 主目录中。因此，`hs_list` 只能看到 Historical Server 实例用当前 Historical Server 正在使用的相同主目录创建的记录会话定义。有关配置 Historical Server 主目录的详细信息，请参见第 20 页的“配置多个 Historical Server 实例”。

若要检查由记录会话收集到的数据，您可以：

- 通过使用 Sybase 批量复制 (bcp) 实用程序，用记录会话数据填充 Adaptive Server 表。有关详细信息，请参见第 84 页的“批量复制示例”。
- 启动 Historical Server 回放会话。

## 回放会话

用回放会话可以检索一个或多个记录会话期间收集的数据。可以按两种方式回放数据：

- 回放到客户端 — 将回放的结果发送给用户，用户可以在终端上查看结果，或将它们重定向到文件。
- 回放到文件 — 将回放的结果存储在文件中。所产生的文件实质上是一个新记录会话。可以将此文件用作另一个回放会话的输入，也可以用作填充 Adaptive Server 表的 bcp 实用程序的输入，或以使用记录会话文件的其它方式使用这些文件。

下面的属性定义了回放会话：

- 输入记录会话 — 一个回放会话的输入是一个或多个记录会话。
- 视图、开始时间和结束时间 — 这些属性定义输入记录会话中要包含在回放会话中的数据。
- 汇总级别 — 可以指定原始回放，这种回放将记录的内容原样显示；也可以指定不同的汇总级别。

若要创建一个回放会话，请从 Historical Server 命令接口使用以下一系列命令：

- [hs\\_create\\_playback\\_session](#)
- [hs\\_create\\_playback\\_view](#)
- [hs\\_initiate\\_playback](#)
- [hs\\_playback\\_sample](#)（只用于回放到客户端）
- [hs\\_terminate\\_playback](#)

## 视图

记录会话视图定义要让 Historical Server 收集的性能数据。回放会话视图定义的性能数据，来自于要让 Historical Server 回放的记录会话视图。

视图由视图名、一个或多个数据项组成。每个数据都有一个与它关联的统计类型。有关详细信息，请参见第 5 页的“数据项和统计类型”。

定义一个记录会话时，应定义要包括在该记录会话中的一个或多个视图。一个记录会话必须至少有一个视图。有关创建记录会话视图的详细信息，请参见第 61 页的 [hs\\_create\\_view](#)。

附录 D “记录会话视图的示例” 包含许多记录会话视图示例。

定义一个回放会话时，应定义在该回放会话中包括以前定义的记录会话中的哪些视图。回放会话视图的名称必须与用于记录会话视图的名称相同。在相应的回放视图中，可以包括所有数据项，或来自记录会话视图的数据项子集。有关创建回放视图的详细信息，请参见第 57 页的 [hs\\_create\\_playback\\_view](#)。

## 数据项和统计类型

**数据项**标识要包括在视图中的特定信息。如果数据项包括嵌入的空格，使用名称时，必须用引号将它括起来。数据项示例：页 I/O、登录名和 CPU 时间。

第 93 页的表 A-1 列出了所有可用数据项，并且对每一项都进行了说明。

每个数据项都有一个与它关联的**统计类型**。统计类型定义数据项（采样或会话）的持续时间，以及 Historical Server 是否对数据项进行计算。

统计类型含有嵌入的空格。在 Historical Server 命令中使用这些统计类型时，必须用引号将它们括起来。

不是所有的统计类型对所有数据项都有效。第 136 页的表 B-3 显示了对每个数据项有效的统计类型。

统计类型有以下六种：

- “采样值” — 该统计类型返回适用于最近一次采样间隔的活动计数或某种类型的信息。它不隐含任何计算。
- 活动计数 — 对于表示活动计数的数据项，该统计类型返回最近一次采样间隔期间某项活动发生的次数。例如，“页 I/O”的“采样值”是在最近一次采样间隔期间页 I/O 发生的次数。

- 其它信息 — 对于表示字符串的数据项，它是唯一有效的统计类型。例如，“对象名”的“采样值”返回一个表的名称。如果数据项表示诸如 ID 之类的值以及诸如“进程 ID”和“代码内存大小”之类的配置参数值，则该统计类型也是对此类数据项唯一有效的统计类型。
- “会话值” — 该统计类型返回自开始收集数据以来（自打开连接以来）的活动累计计数。不执行计算。例如，“页 I/O”的“会话值”是自记录会话开始以来发生的页 I/O 数。
- “采样速率” — 该统计类型计算每秒速率。它返回最近一次采样间隔期间某项活动平均每秒钟发生的次数。例如，“页 I/O”的“采样速率”是在最近一次采样间隔期间平均每秒发生的页 I/O 次数。

计算公式为：

$$\frac{\text{最近一次采样间隔期间的计数}}{\text{采样间隔持续的秒数}}$$

- “会话速率” — 该统计类型计算每秒速率。它返回当前记录会话期间某项活动平均每秒发生的次数。例如，“页 I/O”的“会话速率”是自记录会话开始以来平均每秒发生的页 I/O 次数。

计算公式为：

$$\frac{\text{会话期间的计数}}{\text{会话持续的秒数}}$$

- “采样平均数” — 该统计类型计算某项活动在最近一次采样间隔期间每次发生时的平均值。只有几个数据项可使用此统计类型。返回值的含义取决于数据项。例如，“过程占用时间”的“采样平均数”是最近一次采样间隔期间，一个存储过程每执行一次所花的平均执行时间。
- “会话平均数” — 此统计类型计算某项活动在会话期间每次发生时的平均值。只有几个数据项可使用此统计类型。返回值的含义取决于数据项。例如，“过程占用时间”的“会话平均数”是记录会话期间，一个存储过程每执行一次所花的平均执行时间。

第 136 页的表 B-3 显示数据项和统计类型的有效组合。

# 配置 Historical Server

主题	页码
<a href="#">Historical Server 配置概念</a>	7
<a href="#">UNIX 平台上的初始配置</a>	9
<a href="#">Windows 平台上的初始配置</a>	12
<a href="#">设置 Historical Server 启动参数</a>	17
<a href="#">配置多个 Historical Server 实例</a>	20

## Historical Server 配置概念

本节介绍在配置 Historical Server 之前应该理解的概念，包括：

- [Historical Server 控制文件和主目录](#)
- [操作系统启动帐户](#)
- [Historical Server 超级用户帐户](#)

## Historical Server 控制文件和主目录

Historical Server 控制文件维护有关用户创建的记录会话的信息。该信息在启动过程中仍保留，这样用户可以访问先前执行 Historical Server 期间创建的记录会话。控制文件限制用户访问私人记录会话文件。（记录会话文件可公共访问或私人访问。）

Historical Server 主目录之所以重要的原因有以下两点：

- 它包含 Historical Server 控制文件。Historical Server 在启动时会主目录内查找控制文件。如果控制文件不存在，Historical Server 就创建一个。
- 对于 Historical Server 在记录对话中写入的数据文件，这是缺省目录位置。创建该记录会话的用户可以替换这个缺省位置。

Historical Server 启动命令中的 `-D` 参数指定主目录的位置。这是必需的参数。

只有当前执行和先前执行在相同主目录下使用相同的控制文件时，Historical Server 的当前执行才可以访问先前执行中的数据文件。因此在多数情况下，两次启动之间不应改变 Historical Server 主目录。有关使用不同主目录的讨论，请参见第 20 页的“配置多个 Historical Server 实例”。

有关由 Historical Server 创建的文件的信息，请参见第 5 章“数据文件和输出选项”。

## 访问控制文件信息

使用 Historical Server `hs_list` 命令访问 Historical Server 控制文件中的信息。

## 不要编辑控制文件

不要编辑控制文件。您可能会无意中破坏它。不管您用什么编辑器，都不要打开并保存此文件。

特别是当 Historical Server 是在 Windows 上运行时更是如此。与 Historical Server 创建的其他文件不同，控制文件不是一个标准格式的 Windows 文本文件。控制文件的文本行只用换行符结束，而不是通常的回车符 / 换行符对。编辑程序可能会将不需要的回车符 / 换行符对嵌入到文本中，从而损坏文件。

## 操作系统启动帐户

该帐户要启动 Historical Server，必须满足以下条件：

- 每次启动 Historical Server 必须使用相同帐户。

当 Historical Server 创建控制文件时，首次启动 Historical Server 的操作系统帐户是以后可以访问该控制文件的唯一帐户。要访问该控制文件，以后每次启动 Historical Server 都要使用相同的主目录用相同帐户来执行。该限制可防止未经授权阅读和修改 Historical Server 文件。

如果启动命令指定了不同的主目录，就可以用不同的帐户启动 Historical Server。如果在新位置不存在控制文件，Historical Server 就创建一个。一个主目录中的记录会话数据文件对于使用另一个不同主目录的 Historical Server 是不可见的。



- 帐户必须对在启动命令中指定的 Historical Server 主目录具有搜索（执行）和写入的权限。
- 帐户必须对记录会话数据文件的位置具有搜索（执行）和写入的权限，该位置由创建记录会话的用户指定。缺省位置是 Historical Server 主目录，但是用户可以在创建记录会话时逐个替换缺省值。

## Historical Server 超级用户帐户

可以选择使用 Historical Server 启动命令中的 `-U` 参数指定 Historical Server 的超级用户。如果指定该参数，则超级用户可以：

- 关闭 Historical Server
- 查看或删除任何历史数据文件

如果启动命令不包括 `-U` 参数，任何用户都可停止 Historical Server，但是没有用户可以无限制地访问历史数据文件。

## Sybase Open Client/Server 连接

Historical Server 使用 *interfaces* 文件（在 UNIX 上为 *interfaces* 文件；在 Windows 上为 *sql.ini* 文件）或 Sybase Open Client/Server 11.1.x 版支持的目录服务来建立客户端 / 服务器连接。

使用目录服务的好处在于不必更新所有客户端计算机上的 *interfaces* 或 *sql.ini* 文件。单个目录服务条目就可替代这些文件。对于管理员而言，像将服务器移动到新地址或改变服务器名称这样的变动会更容易。

有关详细信息，请参见 Open Client/Server Configuration Guide for UNIX（《用于 UNIX 的 Open Client/Server 配置指南》）或 Open Client/Server Configuration Guide for Desktop Platforms（《用于桌面平台的 Open Client/Server 配置指南》）。

## UNIX 平台上的初始配置

本节介绍如何在 UNIX 平台上配置 Historical Server。

## UNIX 平台的前提条件

本节中的说明基于以下假设：

- 按照随分发介质提供的操作说明，从分发介质卸载了 Historical Server 软件。
- 在网络上配置了 Adaptive Server/Monitor Server 对。

## UNIX 平台上的配置过程

在 UNIX 平台上配置 Historical Server：

- 1 将 \$SYBASE 环境变量设置为卸载 Historical Server 所在的 Sybase 安装目录的值。
- 2 使用“sybase”帐户或另一个对 \$SYBASE 目录有读、写和搜索（执行）权限的帐户登录。
- 3 设置 \$PATH 环境变量。

Historical Server 可执行文件位于 *\$SYBASE/bin* 中。将该路径名添加到将启动 Historical Server 的帐户的 \$PATH 环境变量中。

- 4 创建用于启动 Historical Server 的脚本文件。

脚本文件确保每次启动 Historical Server 时都使用正确的参数。脚本文件包含 Historical Server 启动命令 `histserver` 及其参数。

若要创建 Historical Server 脚本文件，请执行以下操作：

- a 使用任意编辑器创建一个新文件。新文件的推荐名称和位置是：

```
install_dir/install/run_histServerName,
```

其中 *histServerName* 是 Historical Server 的名称。

- b 编辑新文件，插入 `histserver` 命令并提供适合安装的参数和值。不要在命令中使用回车符；使用 UNIX 延续符 (\) 在多行中延续命令。在参数和值之间留不留空格均可。

第 18 页的表 2-1 介绍了此命令及其参数。表中标记了“必需”的参数必须出现在脚本文件中。对于那些带有缺省值的参数，如果缺省值可接受，则可以省略。

下面是一个启动 Historical Server 的脚本文件示例：

```
histserver -Dserver1HistDir -Sserver1Hist \
-Usa -PsaPasswd \
-lserver1HistLog -n15 &
```

- c 使用 `chmod` 命令向将启动 Historical Server 的帐户授予执行此脚本文件的权限。
- 5 为 Historical Server 添加连接信息。

该任务为 Historical Server 分配端口或网络地址。它也确保了 Historical Server 可以连接到一个或多个 Adaptive Server/Monitor Server 对。

将 Historical Server 连接信息添加到 *interfaces* 文件或目录服务中。有关 `dsedit`、`dscp`、`dsedit_dce` 和 `dscp_dce` 的信息，请参见 Open Client/Server Configuration Guide for UNIX（《用于 UNIX 的 Open Client/Server 配置指南》）。

如果依靠 *interfaces* 文件建立客户端 / 服务器连接：

- a 检查位于 Historical Server 所使用的 *interfaces* 文件中的服务器列表。要让 Historical Server 运行，该文件必须包含下列所有服务器的条目：
  - 任何您想要监控的 Adaptive Server
  - 与每个 Adaptive Server 配对使用的 Monitor Server
  - Historical Server
- b 使用 `dsedit`（如果系统正在运行 X-Windows）或 `dscp`（一个命令行实用程序）将条目添加到 *interfaces* 文件中。按照 Open Client/Server Configuration Guide for UNIX《用于 UNIX 的 Open Client/Server 配置指南》中的说明进行操作。要添加这些条目，您必须知道：
  - 想要让 Historical Server 连接到的 Monitor Server 和 Adaptive Server 的名称。
  - 配置这些服务器时分配给它们的端口号或网络地址。若要研究此信息，请在配置服务器的计算机上使用 `dsedit` 或 `dscp` 来检查适当的 *interfaces* 文件。

如果依靠目录服务建立客户端 / 服务器连接：

- 应确保安装 Historical Server 的计算机上的 *libtcl.cfg* 文件指向适当的目录服务。使用编辑器检查和更新 *libtcl.cfg* 文件。
- 使用 `dsedit_dce`（如果系统正在运行 X-Windows）或 `dscp_dce`（一个命令行实用程序）将 Historical Server 添加到目录服务中。要完成此步骤您需要知道 Historical Server 的名称。

## 6 配置 Historical Server 客户端计算机。

此任务使得客户端能够连接到 Historical Server。Historical Server 客户端是创建记录会话或回放会话的用户。每个客户端计算机必须正确配置。

有关使用 `dsedit` 和 `dscp` 的说明，请参见 *Open Client/Server Configuration Guide for UNIX* 《用于 UNIX 的 Open Client/Server 配置指南》。

如果依靠 *interfaces* 文件建立客户端 / 服务器连接：

- a 更新所有 Historical Server 客户端使用的 *interfaces* 文件。客户端 *interfaces* 文件必须包含以下条目：
  - Historical Server
  - 任何想要通过 Historical Server 监控的 Adaptive Server
  - 与每个列出的 Adaptive Server 相关联的 Monitor Server
- b 使用 `dsedit`（如果系统运行的是 X-Windows）或 `dscp`（一个命令行实用程序）向 *interfaces* 文件中添加条目。要添加这些条目，您必须知道下列信息：
  - Historical Server 名。
  - 想要让 Historical Server 连接到的 Monitor Server 和 Adaptive Server 的名称。
  - 配置这些服务器时分配给它们的端口号或网络地址。若要研究此信息，请在配置服务器的计算机上使用 `dsedit` 或 `dscp` 来检查适当的 *interfaces* 文件。

如果依靠目录服务建立客户端 / 服务器连接，应确保所有 Historical Server 客户端计算机上的 *libtcl.cfg* 文件都指向适当的目录服务。使用文本编辑器检查和更新 *libtcl.cfg* 文件。

## Windows 平台上的初始配置

本节介绍如何在 Windows 计算机上配置 Historical Server。包括以下主题：

- [Windows 平台的前提条件](#)
- [Windows 上的安装结果](#)
- [Windows 上的配置过程](#)

## Windows 平台的前提条件

这些过程假定：

- 按照随分发介质提供的操作说明，从分发介质装载了 Historical Server 软件。
- 在网络上配置了 Adaptive Server/Monitor Server 对。

## Windows 上的安装结果

在 Windows 平台上，Sybase 安装过程要执行几乎所有的 Historical Server 安装。安装进程：

- 将 Historical Server 文件复制到 Sybase 安装目录。
- 将 Historical Server 启动命令的参数值添加到注册表中。它使用这些参数的缺省值。这些条目位于下面的目录之下：

```
\\HKEY_LOCAL_MACHINE\SOFTWARE\SYBASE\SERVER\  
servername\Parameters
```

- 将新的 Historical Server 添加到注册表中的服务列表。这些条目位于下面的目录之下：

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\  
Services
```

- 将 Historical Server 连接信息添加到 Historical Server 计算机上 Sybase 安装目录中的 *sql.ini* 文件。

## Windows 上的配置过程

完成 Historical Server 配置：

- 1 为 Historical Server 添加连接信息。

该任务为 Historical Server 分配端口或网络地址。它也确保了 Historical Server 可以连接到一个或多个 Adaptive Server/Monitor Server 对。向 *sql.ini* 文件或目录服务添加连接信息。有关以下任何过程和 *ocscfg* 或 *dsedit* 的更多信息，请参见 *Open Client/Server Configuration Guide for Desktop Platforms* 《用于桌面平台的 Open Client/Server 配置指南》。

如果依靠 *sql.ini* 文件建立客户端 / 服务器连接, 请检查 *sql.ini* 文件中的服务器列表。要让 Historical Server 运行, 该文件必须包含下列所有服务器的条目:

- 任何您想要监控的 Adaptive Server。
- 与每个 Adaptive Server 配对使用的 Monitor Server。
- Historical Server。Historical Server 条目应该存在, 因为 Sybase 安装过程会添加它们。不过, 因为运行 Adaptive Server/Monitor Server 对的计算机通常不是安装 Historical Server 的计算机, 所以 Historical Server 计算机中可能不存在用于这些对的条目。在这种情况下, 必须为 Historical Server 要连接的每个 Adaptive Server/Monitor Server 对添加条目。

使用 *dsedit* 向 *sql.ini* 文件添加条目。要添加这些条目, 您必须知道下列信息:

- 想要让 Historical Server 连接到的 Monitor Server 和 Adaptive Server 的名称。
- 配置这些服务器时分配给它们的端口号或网络地址。若要研究此信息, 请在配置服务器的计算机上使用 *dsedit* 来检查适当的 *sql.ini* 或 *interfaces* 文件。

---

**注释** Adaptive Server 名称不允许用别名。它必须是 Monitor Server 用来识别它的名称。例如, 使用在 Monitor Server 启动命令的 *-s* 参数中使用的值。

如果使用文本编辑器而非 *dsedit* 来编辑 *sql.ini* 文件, 应确保文件中最后一行的末尾是回车。

---

如果依靠目录服务建立客户端 / 服务器连接:

- 应确保安装了 Historical Server 的计算机上的 *libtcl.cfg* 文件指向合适的目录服务。使用 *ocscfg* 检查和更新 *libtcl.cfg* 文件。
- 使用 *dsedit* 将 Historical Server 添加到适当的目录服务中。要完成此步骤, 您必须知道 Historical Server 的名称。安装进程所创建的缺省名称的格式为 *machineName\_hs*。例如 *smith\_hs*。

## 2 配置 Historical Server 客户端计算机。

此任务使得客户端能够连接到 Historical Server。Historical Server 客户端是创建记录会话或回放会话的用户。每个客户端计算机必须正确配置。

如果依靠 *sql.ini* 文件建立客户端 / 服务器连接，请更新所有由 Historical Server 客户端使用的 *sql.ini* 文件。客户端 *sql.ini* 文件必须包含以下条目：

- Historical Server。
- 与每个列出的 Adaptive Server 相关联的 Monitor Server。
- 任何需要 Historical Server 收集数据的 Adaptive Server。

使用 *dsedit* 向 *sql.ini* 文件添加条目。要添加这些条目，您必须知道下列信息：

- Historical Server 名。
- 想要让 Historical Server 连接到的 Monitor Server 和 Adaptive Server 的名称。
- 配置这些服务器时分配给它们的端口号或网络地址。若要研究此信息，请在配置服务器的计算机上使用 *dsedit* 来检查适当的 *sql.ini* 文件。

如果依靠目录服务建立客户端 / 服务器连接，应确保所有 Historical Server 客户端计算机上的 *libtcl.cfg* 文件都指向适当的目录服务。

使用 *ocscfg* 检查和更新 *libtcl.cfg* 文件。

### 3 检查注册表中的启动参数。

这项任务确保安装进程插入到注册表中的缺省启动参数值适合您的节点。

当使用“控制面板”的“服务”窗口启动 Historical Server 时，服务器从此注册表条目中读取其启动参数。如果从命令行或通过批处理文件启动服务器，Historical Server 就会同时使用注册表条目和命令中的启动参数。如果两个位置出现相同的参数，命令中指定的值优先于注册表条目中的值。如果没有在命令中指定任何启动参数，则缺省使用所有注册表条目参数。

## 更改启动参数

在注册表中更改启动参数或更改服务器名：

- 1 启动注册表编辑器 (*regedt32.exe*，通常在 *winnt\system32* 中)。
- 2 对于 Windows，请选择具有如下名称的窗口：

```
\\HKEY_LOCAL_MACHINE
```

或者，对于 Windows 3.5.1，请选择具有如下名称的窗口：

```
\\HKEY_USER
```

并选择正确的用户条目。

- 3 先保存或打印现有设置，再继续。从注册表菜单中，选择“将子目录树另存为”命令或“打印”命令。
- 4 在树视图中突出显示：

```
\\SOFTWARE\\SYBASE\\Server\  
  srvrName\\Parameters
```

其中 *srvrName* 是您想更改其启动参数的服务器的名称。在窗口的右侧，查看以如下格式显示的现有启动参数列表：

```
Argx, dataType, parameter
```

其中：

- *x* 是一个按连续顺序出现的整数。
- *dataType* 定义了参数值中数据的类型。
- *parameter* 是一个启动选项，它前面是破折号，后面跟着参数值。

下面是一个包含 Historical Server -D、-U 和 -P 启动参数的例子：

```
Arg0:REG_SZ:-Dc:\sybase\data\hs_data  
Arg1:REG_SZ:-Uhssuper  
Arg2:REG_SZ:-Pxwdf
```

- 5 若要添加新启动参数，请执行以下操作：
  - 选择“编辑” | “添加值”。
  - 在随即出现的对话框的“值名”框中，输入 *Argx*，其中 *x* 是当前未指定的下一个整数。
  - 从“数据类型”下拉列表中选择 REG\_SZ。
  - 在随即出现的“字符串”对话框中，输入参数和值。
- 6 若要修改现有参数，请执行以下操作：
  - 双击您要更改的参数行。
  - 在随即出现的“字符串”对话框中更改条目。
- 7 从“注册表”菜单中，选择“退出”。



## 设置 Historical Server 启动参数

本节讲述 Historical Server 的启动命令和参数。本节适用于运行在 UNIX 和 Windows 这两个平台上的 Historical Server。

### 作用

启动 Historical Server。

### 语法

```
histserver -U<user name> -P<password> -D<output dir> -l<log file> -I<interfaces file> [-d<delimiter>] [-O<ASE name>] [-o<DATABASE name>] [-f] -u<output ASE user name> -p<outputASEpassword> [-m]
```

其中：

- -O 指定用于输出 Historical Server 监控数据的 Adaptive Server 的 interfaces 文件名
- -o 指定要在其中存储监控数据的数据库名
- -u 输入用于连接到输出 Adaptive Server 的登录名
- -p 输入与输出 Adaptive Server 相应的口令
- -f 指定需要创建输出文件，它是必需的，即使您已经指定了一个输出 Adaptive Server 也是如此
- -m 表示应该使用并行输出模式

在 UNIX 平台上可执行文件名是 `histserver`，在 Windows 平台上是 `histsrvr`。

## 参数

表 2-1 介绍了 Historical Server 启动命令的参数。

**表 2-1: Historical Server 启动参数**

参数	说明
-DhomeDir	<p><b>必需的。</b> 该参数指定此 Historical Server 实例的主目录。启动 Historical Server 的帐户必须有 Historical Server 主目录的读、写和搜索（执行）权限。</p> <p>缺省主目录是启动 Historical Server 帐户所处的当前工作目录。该 Historical Server 实例使用的控制文件位于主目录中。控制文件包含有关过去记录会话的信息，建立新记录会话时，该信息在当前运行期间进行更新。</p> <p>该参数指定的目录也是记录会话期间写入历史数据文件的缺省目录。创建记录会话的用户可以替换该缺省位置，却不会影响控制文件的定义。</p> <p>多个 Historical Server 实例可以共享相同的主目录。有关详细信息，请参见第 20 页的“配置多个 Historical Server 实例”。</p> <p><b>注释</b> 如果任何特定于 UNIX shell 的字符（如 C shell 的否定符号 (~)）出现在目录名的开始处，则必须要用一个或多个空格字符将 -D 参数和目录名隔开。否则，shell 将不能识别出文件名扩展是必需的。</p>
-d delim	<p>为 Historical Server 所生成的视图文件指定缺省文件分隔符。Delim 可以是：</p> <ul style="list-style-type: none"> <li>• 字符。例如：./histserver -d ' ' 或 ./histserver -d/</li> <li>• 转义符。例如：./histserver -d'\t'</li> <li>• 表示分隔符的 ASCII 代码的四位十六进制数字。例如：./histserver -d0x09</li> </ul>
-iinterfacesFile (破折号后的大写或小写“i”是有效的。)	<p>要使用的 <i>interfaces</i> 文件的路径名。该文件必须包含 Historical Server 的连接信息（包括正确名称）以及希望该 Historical Server 连接到的所有 Adaptive Server 和 Monitor Server 对。如果省略该参数，缺省的 <i>interfaces</i> 名称和位置是：</p> <p>UNIX: \$SYBASE/interfaces</p> <p>Windows: SYBASE\sql.ini（其中 SYBASE 是 SYBASE 环境变量的值）</p>
-llogFile	<p>记录 Historical Server 信息和错误消息的日志文件的路径名。收集的数据包括启动信息、错误消息，还可能包括已触发报警的记录。当前目录下的缺省路径名是 <i>hs.log</i>。</p>

参数	说明
<code>-UserName</code>	<p>Historical Server 超级用户的名称。当超级用户登录到 Historical Server 时，该用户有权发出 <code>shutdown</code> 命令以终止 Historical Server，并可查看或删除由 Historical Server 创建的任何数据文件，而不管是哪个用户启动记录会话。</p> <p>该用户名不必与 Adaptive Server 登录帐户或操作系统的注册相对应。有关详细信息，请参见第 9 页的“<a href="#">Historical Server 超级用户帐户</a>”。</p> <p>如果不指定该参数，任何用户都可停止 Historical Server，但是没有用户可以无限制地访问 Historical Server 数据文件。</p>
<code>-Password</code>	<p>由 <code>-U</code> 参数指定的用户的口令。登录到 Historical Server 的用户必须提供此 <code>username</code> 和 <code>password</code> 才能行使 Historical Server 超级用户权限。有关详细信息，请参见第 9 页的“<a href="#">Historical Server 超级用户帐户</a>”。</p> <p><b>注释</b> 如果在 UNIX 平台上指定 <code>-U</code> 参数却不指定该参数，Historical Server 就在启动期间提示输入口令。不过，要收到提示，Historical Server 必须在前台模式下启动。如要系统提示用户输入口令，请对启动脚本作如下更改：</p> <ul style="list-style-type: none"> <li>• 从启动脚本文件中删除 <code>-P</code> 参数和“与”符号 (<code>&amp;</code>)。</li> <li>• 执行脚本文件。</li> <li>• 提示输入口令时，输入在 <code>-U</code> 参数中为 <code>username</code> 指定的口令。</li> <li>• 将 Historical Server 进程放在后台。</li> </ul>
<code>-nmaxConnections</code>	<p>指定 Historical Server 允许的 Open Client 并发连接的最大数量。有效值为 1 到 20。缺省值为 20。</p> <p>Historical Server 的 Open Client 连接是 <code>isql</code> 连接。</p>
<code>-ShsName</code>	<p>出现在 <code>interfaces</code> 文件中的 Historical Server 的名称。如果省略，Historical Server 名缺省为 <code>DSLISNEN</code> 环境变量的值。如果 <code>DSLISNEN</code> 未设置，则使用名称“<code>histserver</code>”。</p>
<code>-v</code>	<p>显示 Historical Server 版本信息并退出，忽略所有其它参数。</p>

## 配置多个 Historical Server 实例

有时可以适当地运行多个 Historical Server 实例。不过，必须正确配置多个实例以允许或限制用户访问多个实例的记录会话。

本节包括以下内容：

- [何时创建 Historical Server 的多个实例](#)
- [在 UNIX 平台上配置附加的 Historical Server](#)
- [在 Windows 上配置附加的 Historical Server](#)

### 何时创建 Historical Server 的多个实例

如果 Historical Server 的单个实例正在管理采样间隔短的多个并发记录会话，则处理和记录样本所需的时间量可能会影响新样本的及时获得。如果记录数据文件中的时间戳指出未能在所需样品间隔的合理近似时间收集样本，则可以选择两个或多个 Historical Server 实例来分配装载。

当 Historical Server 的多个实例使用 *相同* 的主目录时：

- 每个实例都必须由相同的操作系统帐户启动。
- 所有使用相同主目录的实例都使用相同的控制文件。
- 一个实例记录的会话也可被其它的实例看到。
- 文件锁定机制确保控制文件不会被并发的访问破坏。

当多个 Historical Server 实例使用 *不同* 的主目录时：

- Historical Server 实例可以由不同的用户帐户启动。
- Historical Server 实例使用不同的控制文件。
- 一个服务器的记录会话对其它服务器不可见。例如，通过 `isql` 发出的 `hs_list` 命令不能访问由另一 Historical Server 使用不同的主目录所定义的记录会话。

在决定多个实例是应该使用相同还是不同的主目录时，请考虑以下因素：

- 数据可见性 — 如果多个实例的用户需要访问彼此的记录会话定义，则使用相同的主目录。如果不希望共享，则使用不同的主目录。所有输入到回放会话中的会话必须在相同的主目录中。
- 状态信息的可见性 — `hs_status` 命令不能区分使用相同主目录的多个 Historical Server 实例。如果此区分对于 Historical Server 管理很重要，则使用不同的主目录。

- 对控制文件的争用 — 如果多个用户争相访问一个大型控制文件，则采用不同的主目录可能会提供较快的用户响应时间。诸如创建记录会话和列出记录会话的有关信息这类操作要使用控制文件；记录会话本身在运行时不会阻止其它用户使用控制文件。

## 在 UNIX 平台上配置附加的 Historical Server

若要在 UNIX 平台上配置附加的 Historical Server，请执行以下操作：

- 1 复制为原始 Historical Server 创建的启动脚本。命名新文件，以反映要启动的新 Historical Server 的名称。
- 2 编辑启动文件，更改启动命令的参数以便适合新的 Historical Server 实例。请参见第 17 页的“设置 Historical Server 启动参数”。特别要注意 -D 参数，它指定 Historical Server 的主目录。必须决定是需要多个 Historical Server 实例共享相同的主目录，还是保持相互独立。
- 3 为 Historical Server 建立连接信息，为新的 Historical Server 添加条目。新的 Historical Server 在本机上使用唯一的端口号。
- 4 在客户端计算机上编辑连接信息，为新的 Historical Server 添加条目。

## 在 Windows 上配置附加的 Historical Server

服务器配置实用程序不能配置新的 Historical Server。若要在初始安装后配置附加的 Historical Server，请执行以下操作：

- 1 将启动参数添加到 Windows 注册表
- 2 更新 Windows 注册表中的服务列表
- 3 为 Historical Server 添加连接信息
- 4 配置 Historical Server 客户端计算机
- 5 创建 .bat 文件（可选）

### 将启动参数添加到 Windows 注册表

若要将 Historical Server 启动参数添加到注册表，请执行以下操作：

- 1 启动注册表编辑器（*regedt32.exe*，通常在 *winnt\system32* 中）。
- 2 对于 Windows 4.0，请选择具有如下名称的窗口：

```
\\HKEY_LOCAL_MACHINE
```

或者，对于 Windows 3.5.1，请选择具有如下名称的窗口：

```
\\HKEY_USER
```

并选择合适的用户。

- 3 先保存或打印现有设置，再继续。从注册表菜单中，选择“将子目录树另存为”命令或“打印”命令。

- 4 在树视图中突出显示：

```
\SOFTWARE\SYBASE\Server
```

- 5 从“编辑”菜单中，选择“添加项”。

- 6 在随即出现的对话框的“项名”框中，输入要添加的 Historical Server 的名称。

将“类”框保留为空白。

- 7 在树视图中突出显示：

```
\SOFTWARE\SYBASE\Server\hsName
```

其中 *hsName* 是新的 Historical Server 名。

- 8 选择“编辑”|“添加值”。

- 9 在随即出现的对话框的“值名”框中，输入：

```
ServerType
```

从“数据类型”下拉列表中，选择：

```
REG_SZ
```

在随即出现的“字符串”对话框中输入：

```
HISServer
```

- 10 应确保仍旧突出显示新的 Historical Server 名对应的条目。

- 11 选择“编辑”|“添加项”。

- 12 在“项名称”框中，输入：

```
Parameters
```

将“类”框保留为空白。

- 13 在树视图中突出显示：

```
\SOFTWARE\SYBASE\Server\hsName\Parameters
```

其中 *hsName* 是新的 Historical Server 名。

- 14 选择“编辑”|“添加值”。

15 在“值名”框中，输入：

Arg0

从“数据类型”下拉列表中，选择：

REG\_SZ

在随即出现的“字符串”对话框中输入：

*-ShsName*

其中 *hsName* 是 Historical Server 名。

16 重复第 13 步到 15 步，直到输入完所有的 Historical Server 启动参数为止。

至少要输入下列所需参数。顺序无关紧要；例如，-D 不必是 Arg1。

值名	数据类型	字符串
Arg0	REG_SZ	<i>-ShsName</i> 其中 <i>hsName</i> 是在第 5 步中使用的 Historical Server 名。 示例： <i>-SHS_SERVER1</i>
Arg1	REG_SZ	<i>-DdataDirectoryName</i> 其中 <i>dataDirectoryName</i> 是要将 Historical Server 数据存储到的 <i>现有</i> 目录的完整路径名。所有要创建 Historical Server 记录会话的用户都必须有权写入该目录。 示例： <i>-Dc:\SYBASE\data\hs.data</i>
Arg2	REG_SZ	<i>-InstallationRootDir\ini\sql.ini</i> 示例： <i>-Ic:\SYBASE\ini\sql.ini</i>
Arg3	REG_SZ	<i>-InstallationRootDir\install\logName</i> 其中 <i>logName</i> 是要将 Historical Server 错误消息存储到的文件名。如果该文件不存在，服务器将在启动时创建它。 示例： <i>-lc:\SYBASE\install\hs.log</i>

## 更新 Windows 注册表中的服务列表

若要将 Historical Server 添加到注册表中的服务列表，请执行以下操作：

1 启动注册表编辑器 (*regedt32.exe*，通常在 *winnt\system32* 中)。

2 对于 Windows 4.0，请选择具有如下名称的窗口：

```
\\HKEY_LOCAL_MACHINE
```

或者，对于 Windows 3.5.1，请选择具有如下名称的窗口：

```
\\HKEY_CURRENT_USER
```

并选择正确的用户。

3 先保存或打印现有设置，再继续。从注册表菜单中，选择“将子目录树另存为”命令或“打印”命令。

4 在树视图中突出显示：

```
\\SYSTEM\\CurrentControlSet\\Services
```

5 选择“编辑”|“添加项”。

6 在“项名称”框中，输入：

```
SYBHIS_< i>hsName
```

其中 *hsName* 是刚刚配置的 Historical Server 的名称。

将“类”框保留为空白。

7 在树视图中突出显示：

```
\\SYSTEM\\CurrentControlSet\\Services\\SYBHIS_< i>hsName
```

8 根据需要多次选择“编辑”|“添加值”，将以下所有值、数据类型和字符串都输入到 *SYBHIS\_< i>hsName* 项条目中。

值名	数据类型	字符串
DisplayName	REG_SZ	<i>Sybase HISServer_&lt; i&gt;hsName</i>
ErrorControl	REG_DWORD	01 (选中十六进制单选按钮)
Group	REG_SZ	(留空)
ImagePath	REG_EXPAND_SZ	<i>rootInstlDir\\bin\\histsrvr.exe -ShsName -C</i>
ObjectName	REG_SZ	LocalSystem
Start	REG_DWORD	03 (选中十六进制单选按钮)
Type	REG_DWORD	010 (选中十六进制单选按钮)

9 确保 *SYBHIS\_< i>hsName* 项条目保持突出显示。



10 选择“编辑” | “添加项”。

11 在“项名称”框中，输入：

Security

将“类”框保留为空白。

12 在树视图中，突出显示下面的 Security 项条目：

\SOFTWARE\SYBASE\Server\hsName\Security

13 选择“编辑” | “添加值”。

14 输入下列值作为值名、数据类型和二进制编辑器：

值名	数据类型	二进制编辑器
安全性	REG_BINARY	01001480

15 选择“注册表” | “退出”。

16 重新启动计算机。

## 为 Historical Server 添加连接信息

该任务为 Historical Server 分配端口或网络地址。它也确保了 Historical Server 可以连接到一个或多个 Adaptive Server/Monitor Server 对。

该任务因是依靠 *sql.ini* 文件还是目录服务来建立客户端 / 服务器连接而不同：

- 如果依靠 *sql.ini* 文件建立客户端 / 服务器连接，请检查 *sql.ini* 文件中的服务器列表以查找 Historical Server 计算机。要让 Historical Server 运行，该文件必须包含下列所有服务器的条目：
  - 任何您想要监控的 Adaptive Server
  - 与每个 Adaptive Server 配对使用的 Monitor Server
  - Historical Server

使用 *dsedit* 向 *sql.ini* 文件添加条目。要添加这些条目，您必须知道下列信息：

- 想要让 Historical Server 连接到的 Monitor Server 和 Adaptive Server 的名称。
- 配置这些服务器时分配给它们的端口号或网络地址。

若要研究此信息，请在配置服务器的计算机上使用 `dsedit` 来检查适当的 `sql.ini` 文件。

---

**注释** Adaptive Server 名称不允许用别名。它必须是 Monitor Server 用来识别它的名称。例如，使用在 Monitor Server 启动命令的 `-S` 参数中使用的值。

---

- 如果依靠目录服务建立客户端 / 服务器连接：
  - 应确保安装 Historical Server 的计算机上的 `libtcl.cfg` 文件指向适当的目录服务。使用 `ocscfg` 检查和更新 `libtcl.cfg` 文件。
  - 使用 `dsedit` 将 Historical Server 添加到适当的目录服务中。要完成此步骤，您必须知道 Historical Server 的名称。安装进程所创建的缺省名称的格式为 `machineName_hs`。例如 `smith_hs`。

有关使用 `ocscfg` 和 `dsedit` 的说明，请参见 *Open Client/Server Configuration Guide for Desktop Platforms* 《用于桌面平台的 Open Client/Server 配置指南》。

## 配置 Historical Server 客户端计算机

此任务使得客户端能够连接到 Historical Server。Historical Server 客户端是创建记录会话或回放会话的用户。每个客户端计算机必须正确配置。

该配置任务因是使用 `sql.ini` 文件还是目录服务建立客户端 / 服务器连接而不同。

- 如果您使用 `sql.ini` 文件建立客户端 / 服务器连接，则更新所有由 Historical Server 客户端使用的 `sql.ini` 文件。客户端 `sql.ini` 文件必须包含以下条目：
  - Historical Server
  - 与每个列出的 Adaptive Server 相关联的 Monitor Server
  - 任何需要 Historical Server 收集数据的 Adaptive Server

使用 `dsedit` 向 `sql.ini` 文件添加条目。要添加这些条目，您必须知道下列信息：

- Historical Server 名
- 想要让 Historical Server 连接到的 Monitor Server 和 Adaptive Server 的名称
- 配置这些服务器时分配给它们的端口号或网络地址

若要研究此信息，请在配置服务器的计算机上使用 `dsedit` 来检查适当的 `sql.ini` 文件。

- 如果依靠目录服务建立客户端 / 服务器连接，应确保所有 Historical Server 客户端计算机上的 `libtcl.cfg` 文件都指向适当的目录服务。  
使用 `ocscfg` 检查和更新 `libtcl.cfg` 文件。

## 创建 .bat 文件（可选）

通过此任务，您可以通过执行批处理文件启动 Historical Server，而不必使用服务图标启动。

若要用批处理文件启动 Historical Server，请执行以下操作：

- 1 启动任何文本编辑器并打开一个新文本文件。
- 2 输入 Historical Server 启动命令 `histsrvr` 和所有需要的参数。使用启动命令 `histsrvr.exe` 的完整路径名。有关 Historical Server 获取启动参数所使用的优先规则，请参见第 33 页的“确定注册表中的启动参数”。有关启动参数的解释，请参见第 17 页的“设置 Historical Server 启动参数”。

以下为 `.bat` 文件中的示例行。这个示例在命令中插入了回车。不要在 `.bat` 文件中放入回车。整个文件应该是一行。

```
c:\sybase\bin\histsrvr.exe -Shs_server1  
-Dc:\sybase\data\hs_data  
-Ic:\sybase\ini\sql.ini  
-lc:\sybase\data\hs.log
```

- 3 用扩展名 `.bat` 保存该文件。Sybase 建议您使用 `RUN_hsName.bat`。例如：

```
RUN_hs_server1.bat
```



主题	页码
启动和停止 UNIX 平台上的 Historical Server	29
在 Windows 上启动和停止 Historical Server	32

## 启动和停止 UNIX 平台上的 Historical Server

本节介绍如何启动和停止在 UNIX 平台上运行的 Historical Server。

### 启动 UNIX 上的 Historical Server

在 UNIX 上，您可以通过两种方式启动配置好的 Historical Server：

- 根据 UNIX shell 提示执行 `histserver` 命令。如果使用此方法，您必须每次键入所有适当的参数。
- 执行包含 `histserver` 命令以及所有适当的参数的脚本文件。如果您已按照第 2 章“配置 Historical Server”中的说明进行了配置，您可以使用下列方法启动 Historical Server：

```
install_dir/install/RUN_histServerName
```

其中 `install_dir` 是 Sybase 根目录，`histServerName` 是要启动的 Historical Server 的名称。

无论您使用的是哪种方法：

- 检验要监控的 Adaptive Server 及其相应的 Monitor Server 是否正在运行。
- 如果希望新的实例有权访问先前的记录会话，则每次启动 Historical Server 都应使用相同的帐户。有关详细信息，请参见第 8 页的“操作系统启动帐户”。

- 将 \$SYBASE 环境变量设置为 Sybase 安装的根目录。

\$SYBASE 环境变量必须包含一个目录名，该目录有适用于 Historical Server 的适当 *locales* 和 *charsets* 子目录。这些子目录由安装过程自动创建和填充。

\$SYBASE 环境变量还标识 *interfaces* 文件的缺省位置。使用 `histserver` 命令的参数覆盖这些文件的缺省位置。

Historical Server 显示以下消息来指示启动成功：

```
Initialization is over. Ready to accept connections.
```

Historical Server 在启动期间将消息写入其日志文件中。如果启动成功，可以忽略这些消息。如果启动没有成功，检查日志文件研究问题所在。

启动时，Historical Server 日志文件的缺省路径名是在当前目录下的 *hs.log*。您可以使用 `histserver` 命令的 `-l` 参数（字母 l）覆盖此缺省路径名。

## 停止 UNIX 上的 Historical Server

本节介绍：

- [谁可以关闭 Historical Server](#)
- [确定 Historical Server 上的当前活动](#)
- [延迟关闭与立即关闭](#)
- [详细的关闭过程](#)

### 谁可以关闭 Historical Server

如果启动命令指定了超级用户帐户，则只有超级用户帐户才可以停止 Historical Server。如果未在启动命令中指定超级用户，则任何用户都可以停止 Historical Server。

超级用户是在 Historical Server 启动命令的 `-U` 和 `-P` 参数中指定其帐户的用户。

## 确定 Historical Server 上的当前活动

在关闭 Historical Server 之前，检查 Historical Server 上的当前活动，以便决定是延迟关闭还是立即关闭。延迟关闭允许在终止 Historical Server 之前完成所有当前活动。

Historical Server 活动可能包括完全定义和启动的记录会话、仍在定义过程中的未启动记录会话、进行中的回放会话和定义的回放会话。客户端连接可以来自于多台计算机，并可能监控着不同的 Adaptive Server。

若要确定 Historical Server 中的当前活动，请使用 `isql` 实用程序连接到 Historical Server 并发出 `hs_status activity` 命令。

## 延迟关闭与立即关闭

若要关闭 Historical Server，请使用 `isql` 实用程序连接到它，并发出以下命令之一：

- `hs_shutdown` — 延迟关闭，直至完成所有活动的记录会话并关闭所有其它活动连接。在此时间内不接受任何新连接。  
延迟关闭可能会阻塞很长时间，并且在 `hs_shutdown` 命令阻塞时键入 `Ctrl+C` 不起作用。
- `hs_shutdown no_wait` — 立即关闭 Historical Server 并终止所有连接和活动的记录会话。

在任一情况下，都以可控的方式关闭活动的记录会话。Historical Server 重新启动后，仍可使用 Historical Server 控制文件和历史数据文件。

## 详细的关闭过程

按以下顺序关闭服务器：

- Historical Server
- Monitor Server
- Adaptive Server

在 UNIX 平台上停止 Historical Server:

- 1 连接到 Historical Server。如果您使用的是 `isql`，则该命令为:

```
isql [ -Uhs_superuser_name -Phs_superuser_password ]  
-Shistorical_server
```

其中:

- *superuser\_name* 是与 Historical Server 启动命令的 `-U` 参数一起使用的名称。如果在启动命令中没有使用 `-U`，则任何用户都可以停止 Historical Server，并且此 `isql` 命令中的 `-U` 参数是可选项。
  - *superuser\_password* 是与 Historical Server 启动命令的 `-P` 参数一起使用的口令。如果启动命令中没有使用 `-U`，则任何用户都可以停止 Historical Server，并且该参数是可选项。
  - *historical\_server* 是想要停止的 Historical Server 的名称。
- 2 若要确定 Historical Server 上的当前活动，请在出现 `isql` 提示时发出以下命令:

```
1> hs_status activity  
2> go
```

- 3 若要关闭 Historical Server，请发出以下命令之一:

```
1> hs_shutdown  
2> go
```

或:

```
1> hs_shutdown no_wait  
2> go
```

## 在 Windows 上启动和停止 Historical Server

本节介绍如何启动和停止在 Windows 平台上运行的 Historical Server。主题有:

- [在 Windows 上启动 Historical Server](#)
- [确定注册表中的启动参数](#)
- [检验 Historical Server 是否正在运行](#)
- [在 Windows 上停止 Historical Server](#)



## 在 Windows 上启动 Historical Server

可以通过以下方式启动 Historical Server:

- 使用 Windows “控制面板”的“服务”窗口。
- 使用包含启动命令和参数的一个批处理 (.bat) 文件。文件名是 *run\_hsName.bat*，其中 *hsName* 是 Historical Server 实例的名称。Sybase 建议从命令行 shell 中调用批处理文件，而不要在“文件管理器”中双击它。如果发生启动错误，命令行 shell 就会捕捉启动错误消息，而“文件管理器”却不能。
- 直接从命令行 shell 中键入启动命令。用于 Windows 的 Historical Server 可执行文件的名称是 *histsrvr.exe*。

在使用 .bat 文件或命令行命令启动 Historical Server 时，服务器进程将与当前的登录帐户关联起来。当您注销时服务器关闭。

对于生产系统，Sybase 建议使用 Windows “控制面板”中的“服务管理器”来启动 Historical Server。将服务器作为一项服务启动时，它在切换登录名后不受影响。

## 确定注册表中的启动参数

Historical Server 的启动参数位于以下注册表条目中:

```
\SOFTWARE\SYBASE\SERVER\servername\Parameters
```

当使用“控制面板”启动 Historical Server 时，服务器从该注册表条目中读取它的启动参数。如果从命令行或通过批处理文件来启动服务器，则要从注册表条目和命令二者中获取启动参数。如果两个位置出现相同的参数，命令中指定的值优先于注册表条目中的值。如果没有在命令中指定任何启动参数，则缺省使用所有注册表条目参数。

有关编辑注册表条目的信息，请参见第 21 页的“将启动参数添加到 Windows 注册表”。

## 检验 Historical Server 是否正在运行

在 Windows 上，检查 Historical Server 在 Windows “控制面板”中的“服务”窗口中的状态。

## 在 Windows 上停止 Historical Server

本节介绍：

- [何种用户可以关闭 Windows 上的 Historical Server](#)
- [确定 Windows 上的当前活动](#)
- [Windows 上的延迟关闭与立即关闭](#)
- [Windows 上的详细关闭过程](#)
- [在 Windows 上关闭时避免使用的方法](#)

### 何种用户可以关闭 Windows 上的 Historical Server

如果在启动时指定了超级用户帐户，则只有超级用户帐户可以停止 Historical Server。如果未在启动命令中指定超级用户，则任何用户都可以停止 Historical Server。有权使用“控制面板”的“服务”窗口的管理员也可以关闭 Historical Server。

超级用户帐户是在 Historical Server 启动命令的 -U 和 -P 参数中指定的。

### 确定 Windows 上的当前活动

在关闭 Historical Server 之前，检查 Historical Server 上的当前活动，以便决定是延迟关闭还是立即关闭。延迟关闭允许在终止 Historical Server 之前完成所有当前活动。

Historical Server 活动可能包括完全定义和启动的记录会话、仍在定义过程中的未启动记录会话、进行中的回放会话和定义的回放会话。客户端连接可以来自于多台计算机，并可能监控着不同的 Adaptive Server。

若要确定 Historical Server 中的当前活动，请使用 isql 实用程序连接到 Historical Server 并发出 `hs_status activity` 命令。

## Windows 上的延迟关闭与立即关闭

若要关闭 Historical Server，请使用 isql 实用程序连接到它，并发出以下命令之一：

- `hs_shutdown` — 延迟关闭，直至完成所有活动的记录会话并关闭所有其它活动连接。在此时间内不接受任何新连接。  
延迟关闭可能会阻塞很长时间，并且 `hs_shutdown` 命令在 isql 内阻塞时键入 Ctrl+C 不起作用。
- `hs_shutdown no_wait` — 立即关闭 Historical Server 并终止所有连接和活动的记录会话。

在任一情况下，都以可控的方式关闭活动的记录会话。Historical Server 重新启动后，仍可使用 Historical Server 控制文件和历史数据文件。

## Windows 上的详细关闭过程

按以下顺序关闭服务器：

- Historical Server
- Monitor Server
- Adaptive Server

若要停止 Windows 上的 Historical Server，请使用以下任一方法：

- 单击 Windows “控制面板”的“服务”窗口中的“停止”按钮。
- 从 isql 中发出 `hs_shutdown` 命令。
- 如果从命令行或通过批处理文件启动，那么 Historical Server 就与启动它的登录相关联。在这些情况下，如果关闭启动 Historical Server 的窗口或注销系统，Historical Server 将会关闭。

如果受控的 Historical Server 关闭花费的时间比预定的时间间隔长（关闭窗口 5 秒钟，注销 20 秒钟），系统会在正常的时间间隔显示一个弹出式对话框，询问是否要终止进程。

---

**警告！** 除非在每次出现对话框时都回答“等待”，否则 Historical Server 就会以非控方式关闭，这样会有数据丢失和文件破坏的危险。

---

若要使用 isql 停止 Historical Server，请执行以下操作：

- 1 使用 isql 连接到 Historical Server：

```
isql -Uhs_superuser_name -Phs_superuser_password  
-Shistorical_server
```

其中：

- *superuser\_name* 是与 Historical Server 启动命令的 -U 参数一起使用的名称。如果启动命令中没有使用 -U，则任何用户都可以停止 Historical Server，并且该参数是可选项。
  - *superuser\_password* 是与 Historical Server 启动命令的 -P 参数一起使用的口令。如果启动命令中没有使用 -U，则任何用户都可以停止 Historical Server，并且该参数是可选项。
  - *historical\_server* 是想要停止的 Historical Server 的名称。
- 2 若要确定 Historical Server 上的当前活动，请在出现 isql 提示时发出以下命令：

```
1> hs_status activity  
2> go
```

- 3 当出现 isql 提示时，发出以下命令之一：

```
1> hs_shutdown  
2> go
```

或：

```
1> hs_shutdown no_wait  
2> go
```

## 在 Windows 上关闭时避免使用的方法

在一些情况下，未首先关闭 Historical Server 就关闭系统会导致非控关闭。除非 Historical Server 正在经历级别非常高的记录活动，否则其受控关闭时间应少于 20 秒。但如果受控关闭时间用了 20 秒或更长时间，系统可能以非控方式干扰和中断 Historical Server。

---

**警告！** 为保险起见，Sybase 建议在关闭系统之前手动停止 Historical Server。

使用“注销进程”按钮会导致非控关闭。

---

如果系统中安装有 Microsoft Visual C++ 进程查看程序工具，Sybase 建议不要在“进程查看程序”窗口使用“注销进程”按钮。

## 命令参考

本章介绍 Historical Server 命令接口。

主题	页码
<a href="#">命令摘要</a>	37
<a href="#">命令语法</a>	38
<a href="#">命令状态和错误</a>	39
<a href="#">将脚本文件输入到 Historical Server</a>	39
<a href="#">连接到 Historical Server</a>	40
<a href="#">Historical Server 命令</a>	43

## 命令摘要

表 4-1 简述了各 Historical Server 命令。

**表 4-1: Historical Server 命令**

活动	命令
创建记录会话	使用下面这些命令创建记录会话： <ul style="list-style-type: none"> <li>• <code>hs_create_recording_session</code> — 定义记录会话的特性。</li> <li>• <code>hs_create_view</code> — 定义一个表示要记录的一组数据的视图。</li> <li>• <code>hs_create_filter</code> — 指定数据项的过滤条件。</li> <li>• <code>hs_create_alarm</code> — 为一个数据项指定触发报警操作的阈值。</li> <li>• <code>hs_initiate_recording</code> — 结束记录会话定义，使记录从其预定义的开始时间开始。</li> <li>• <code>hs_terminate_recording</code> — 停止记录会话。</li> </ul>
查看记录会话定义	<ul style="list-style-type: none"> <li>• <code>hs_list</code> — 列出为记录会话及其数据项定义的所有视图。</li> </ul>

活动	命令
创建回放会话	<p>使用下面这些命令创建回放会话：</p> <ul style="list-style-type: none"> <li>• <code>hs_create_playback_session</code> — 定义回放会话的特性。</li> <li>• <code>hs_create_playback_view</code> — 从输入记录会话中指定一个或多个视图，放入回放会话。</li> <li>• <code>hs_initiate_playback</code> — 结束回放会话定义。另外也开始回放到文件中。</li> <li>• <code>hs_playback_sample</code> — 将一个示例回放到客户端。</li> <li>• <code>hs_terminate_playback</code> — 停止回放会话。</li> </ul>
管理 Historical Server	<p>使用下面这些命令管理 Historical Server：</p> <ul style="list-style-type: none"> <li>• <code>hs_delete_data</code> — 删除包含指定记录会话的记录数据的文件。</li> <li>• <code>hs_status</code> — 显示 Historical Server 的当前状态。</li> <li>• <code>hs_shutdown</code> — 停止 Historical Server。</li> </ul>

## 命令语法

Historical Server 命令的语法是：

```
hs_XXX arg1, arg2, arg3, ..., argn
```

其中 XXX 是命令名。例如，`hs_create_recording_session` 就是一个典型的命令。

参数之间必须用逗号分隔。可以忽略可选的或有缺省值的参数，但是，如果这些参数后面有其它参数值，则必须对它们使用 NULL 值作为占位符。例如，以下命令对大多数参数使用缺省值：

```
hs_create_playback_session null, null, raw, null,
null, client, null, null, null, null, 7
```

命令名、命令关键字、单词 `null`、数据项以及统计类型都不区分大小写。文件名、视图名和其它由用户提供的名称都区分大小写。

如果某参数值中包含内嵌空格（如数据项、统计类型和日期时间规范中的空格），则必须用引号括住该值。相匹配的单引号对或双引号对都是有效的分隔符。

如果参数值包含与用于分隔整个值的字符相同的嵌入引号，请在参数值内应用一个引号对。Historical Server 将引号对压缩为单个字符。

引号内的单词 `null` 不是关键字。

可以在多行中输入 Historical Server 命令。

## 命令状态和错误

所有 Historical Server 命令都返回状态值 0 或 1。

- 0 表示成功执行。
- 1 表示错误情况。如果返回的状态值为 1，则会将错误消息及其相关的错误代码、严重级和状态返回到客户端。

Historical Server 是一个 Open Server 应用程序，它使用 Client-Library™ 与一对或多对 Adaptive Server 和 Monitor Server 进行通信。任何这些组件都会检测和报告错误情况。

Historical Server 也能检测和报告错误情况，并将错误情况记入客户端和 / 或报告给客户端。Historical Server 错误代码都是五位数字的整数，范围是从 30000 到 30999。Open Server 错误代码的范围是从 16000 到 20000。对于在这两个范围内的错误代码，Historical Server 将分配下列严重性级别之一：

**表 4-2: 命令错误中的严重性级别**

严重级	说明
1	信息性消息
2	警告消息
3	致命服务器错误
4	致命进程错误
5	操作系统错误

由 Client-Library、Adaptive Server 和 Monitor Server 检测到的错误使用其自身的错误编号和严重性级别方案。

## 将脚本文件输入到 Historical Server

为 Historical Server 提供输入的一种便捷方法是在文本文件中输入命令，然后使用该文件作为输入。Historical Server 输入文件不是 isql 文件，并且不必遵循 isql 约定。例如，Historical Server 不具有注释功能。

附录 D “记录会话视图的示例” 提供了可用作初始输入文件的视图示例。安装在安装目录的 *sample/histserver* 子目录下的 *views* 文件中也会出现这些视图。但是，该文件中包含一些解释性文本，您必须先删除这些文本，然后才能将该文件输入到 Historical Server。

## 连接到 Historical Server

本节包括以下主题：

- [连接前的假定条件](#)
- [连接方法](#)
- [Historical Server 活动所需的权限](#)
- [互斥会话](#)

### 连接前的假定条件

在连接到 Historical Server 之前：

- 要从其中收集性能数据的 Adaptive Server 必须正在运行。
- 必须有一个监控 Adaptive Server 的 Monitor Server 正在运行中。有关检验 Adaptive Server/Monitor Server 对是否正在运行的说明，请参见《Sybase Adaptive Server Enterprise Monitor Server 用户指南》。
- Historical Server 必须正在运行。有关验证 Historical Server 是否正在运行的说明，请参见第 3 章 “启动和停止 Historical Server”。
- 必须在 Historical Server 计算机及其客户端计算机上正确配置了连接信息。有关详细信息，请参见第 2 章 “配置 Historical Server”。

### 连接方法

使用可为 Sybase 服务器提供交互式界面的实用程序（例如 SQL Advantage 或 isql）连接到 Historical Server。按照实用程序的标准连接步骤进行操作。例如，在 SQL Advantage 中，选择“连接”菜单项。在 isql 中，使用 Historical Server 名称作为 -S 参数的值来启动 isql。



## Historical Server 活动所需的权限

使用拥有所要执行的活动权限的用户登录名和口令连接到 Historical Server，如表 4-3 中所述。根据所需的权限，您可能需要断开连接，然后用不同的登录帐户重新连接，以执行各种不同的活动。

**表 4-3: Historical Server 活动所需的权限**

活动	执行该活动所需的权限
超级用户活动： <ul style="list-style-type: none"> <li>• 停止 Historical Server。</li> <li>• 删除 Historical Server 文件，包括记录会话数据文件。</li> <li>• 访问专有记录会话的文件。</li> </ul>	超级用户权限。要获得超级用户权限，请使用启动时定义的超级用户帐户连接到 Historical Server。 超级用户帐户不必是任何 Adaptive Server 的有效帐户。
创建记录会话。	所监控的 Adaptive Server 上的有效登录名和口令。 对所监控的 Adaptive Server 中的 <code>mon_rpc_connect</code> 存储过程的执行权限。该权限是连接到 Monitor Server 所必需的。
使用公共记录会话文件创建回放会话（从 <code>protection_level</code> 参数为 <b>public</b> 的记录会话中）。 对公共记录会话文件使用 <code>hs_list</code> 命令。	所监控的 Adaptive Server 上的有效登录名和口令。
使用私有记录会话文件创建回放会话（从 <code>protection_level</code> 参数为 <b>private</b> 的记录会话中）。 对私有记录会话文件使用 <code>hs_list</code> 命令。	用于创建私有会话的同一登录名或超级用户。在后一种情形下，超级用户帐户必须是所监控的 Adaptive Server 上的有效登录名。

单个 Historical Server 可以连接到多个 Adaptive Server/Monitor Server 对并从中收集数据。例如，在同一个 Historical Server 连接中，可以创建两个记录会话，一个使用名为 `server1_MON` 的 Monitor Server 收集名为 `server1` 的 Adaptive Server 上的性能数据，另一个使用名为 `server2_MON` 的 Monitor Server 收集名为 `server2` 的 Adaptive Server 上的性能数据。如果您不具有在两个 Adaptive Server 实例中都有有效的登录帐户，就需要使用不同的登录帐户在与 Historical Server 的不同连接中创建这些记录会话。但是，来自这两个记录会话的数据文件可以位于相同的目录下。

`hs_create_recording_session` 命令指定您要连接到的 Monitor Server，这也就间接地隐含了要从中收集性能信息的 Adaptive Server。

## 互斥会话

一个 Historical Server 连接一次只能涉及一个会话。您可以定义一个记录会话或者定义并执行一个回放会话，但不能对多个会话一起混用命令。也就是说，如果用 `hs_create_recording_session` 命令开始一个记录会话定义，则必须先完成定义该会话（或者取消该会话）的命令序列，才能再开始定义一个回放会话或新的记录会话。

定义一个记录会话与进行中的记录会话并不是一回事。在定义一个记录会话期间，定义记录会话的用户连接到 Historical Server 并主动与之通信。在实际记录过程中，用户并不主动与 Historical Server 通信。在记录期间，任何用户连接都是不相关的。例如，某用户可能在下午 3 点连接到 Historical Server，并定义一个在晚上 10 点开始的记录会话，然后断开与 Historical Server 的连接。到了晚上 10 点，如果 Historical Server 正在运行，则记录会话就会发生，这与该用户是否连接到 Historical Server 并不相关。

但对于回放会话来说，在定义回放和执行回放期间都需要有用户连接。要取消回放执行并允许用户连接执行一些其它通信，用户必须发出 `hs_terminate_playback` 命令。

## 开始和结束记录会话定义

`hs_create_recording_session` 命令开始一个记录会话定义。在发出此命令后，可以发出其它命令来为该记录会话创建视图、过滤器和报警，以进一步定义记录该会话。

若要结束记录会话定义，请使用以下命令之一：

- `hs_initiate_recording` — 指示定义已完成，并使记录会话在其所指定的开始时间启动。
- `hs_terminate_recording` — 取消正在进行中的记录会话定义。

## 开始和结束回放会话

`hs_create_playback_session` 命令开始一个回放定义。发出此命令后，您可使用其它命令定义回放会话并执行回放。

在成功发出 `hs_create_playback_session` 命令后，必须成功执行 `hs_terminate_playback` 命令才能结束回放会话。该命令将取消定义并结束回放。

## Historical Server 命令

下面的篇幅描述 Historical Server 命令：

- [hs\\_create\\_alarm](#)
- [hs\\_create\\_filter](#)
- [hs\\_create\\_playback\\_session](#)
- [hs\\_create\\_playback\\_view](#)
- [hs\\_create\\_recording\\_session](#)
- [hs\\_create\\_view](#)
- [hs\\_delete\\_data](#)
- [hs\\_initiate\\_playback](#)
- [hs\\_initiate\\_recording](#)
- [hs\\_list](#)
- [hs\\_playback\\_sample](#)
- [hs\\_shutdown](#)
- [hs\\_status](#)
- [hs\\_terminate\\_playback](#)
- [hs\\_terminate\\_recording](#)

### hs\_create\_alarm

**说明** 创建报警。当某个数据项的值达到指定阈值时，将触发一个报警。

**语法** `hs_create_alarm view_name, data_item_name, data_item_stat, alarm_action, alarm_action_data, alarm_value, occurrence_threshold.`

**参数** *view\_name*  
包含报警所应用的数据项的视图名。

*data\_item\_name*  
报警所应用的数据项。如果数据项包含嵌入空格，则应将其用引号括住。

*data\_item\_stat*

*data\_item\_name* 的统计类型。*data\_item\_name* 和 *data\_item\_stat* 组合必须存在于视图定义中。用引号将统计类型引起来。

*alarm\_action*

当满足报警条件时所采取的操作。值包括：

- **log** — 当满足报警条件时，将消息记录到日志中。日志文件名是在 *alarm\_action\_data* 参数中指定的。
- **execute** — 当满足报警条件时，执行一个程序或脚本文件。要执行的文件是在 *alarm\_action\_data* 参数中指定的。

*alarm\_action\_data*

指定执行 *alarm\_action* 所需的信息：

如果 <i>alarm_action</i> 是	则 <i>alarm_action_data</i> 是
log	Historical Server 将在其中记录报警消息的文件名。缺省文件名是 Historical Server 日志文件。如果指定的文件不存在，Historical Server 将创建该文件。如果指定了一个现有文件，那么启动 Historical Server 的用户必须拥有对此文件的写权限。
execute	要执行的文件名，后面可有选择地加上用空格分隔的参数列表。此文件必须已存在，并且启动 Historical Server 的用户必须拥有对其的执行权限。

**警告！** 当满足报警条件且 *alarm\_action* 是 **execute** 时，将由启动 Historical Server 的帐户执行指定的文件，而不是由创建该报警的帐户执行。这意味着定义报警的用户可拥有启动 Historical Server 的用户所具有的访问权限。通常情况下无法执行某个文件的用户也许可以通过 Historical Server 报警执行该文件。

*alarm\_value*

报警的触发值。

*occurance\_threshold*

在 Historical Server 执行阈值操作之前阈值条件发生的次数。

## 示例

- 1 此示例为名为“Page I/O”的视图创建了一个报警。该视图包含统计类型为“会话值”(Value for Session)的页 I/O 数据项。当达到触发值 50 或 50 以上时，将数据记录到 *page\_io\_alarm\_file* 文件中。

```
hs_create_alarm PageIO,"Page I/O",
    "Value for Session",log,page_io_alarm_file,50
```

- 2 此示例为名为“Page I/O”的视图创建了一个报警。当页 I/O 数据项经过连续 6 次刷新后达到触发值 100 或 100 以上时，此报警将导致执行 */user/script1* 脚本，并传递值 100 作为其第一个参数。

```
hs_create_alarm PageIO,"Page I/O",
    "Value for Session",execute,
    "/user/script1 100",100,6
```

## 用法

- 如果 *alarm\_action* 是 **execute**，当满足报警条件时，将在后台执行脚本或程序。如果适用的话，脚本或程序内的命令在前台执行和显示。  
例如，如果想在某个报警条件下显示时钟，应将 *clock.exe* 放到一个脚本文件中，并在创建报警时指定该脚本文件名，而不是在创建报警时指定 *clock.exe*。
- 当满足报警条件且 *alarm\_action* 是 **log** 时，Historical Server 将在日志文件中写入三行。第一行包括：
  - 触发报警的样本的时间戳
  - 为其设置报警的数据项名
  - 该数据项的统计类型
  - 触发报警的数据项值
  - 报警的阈值
- 日志文件中的第二行包含：
  - 会话 ID
  - 视图名
  - 报警 ID
- 日志文件中的第三行包含：
  - 所监控的 Adaptive Server 的名称
  - 触发报警的样本数据行视图中的所有关键数据项的名称和值

- 当满足报警条件且 *alarm\_action* 是 **execute** 时，Historical Server 将运行 *alarm\_action\_data* 参数中指定的脚本或程序。Historical Server 使用 *alarm\_action\_data* 中指定的参数调用程序或文件，后面附加上可在程序内访问的其它位置参数。由 Historical Server 附加到程序调用后的位置参数表示如下信息：
  - 会话 ID
  - 报警 ID
  - 样本时间戳
  - 在其上设置报警的数据项
  - 该数据项的统计类型
  - 触发报警的数据项值
  - 报警的阈值
  - 所监控的 Adaptive Server 的名称
  - 视图中的关键数据项数目
  - 对于每个关键数据项：
    - 数据项名
    - 触发报警的样本数据行中的数据项值
- 当满足报警条件且 *alarm\_action* 是 **execute** 时，将由启动 Historical Server 的帐户执行指定的文件，而不是由创建该报警的帐户执行。这意味着定义报警的用户可拥有启动 Historical Server 用户所具有的访问权限。

---

**警告！** 通常情况下无法执行某个文件的用户也许可以通过 Historical Server 报警执行该文件。

---

## hs\_create\_filter

说明	指定一个数据项的过滤条件。过滤器限制视图中单个数据项的数据收集范围。过滤器是可选的。
语法	<code>hs_create_filter view_name, data_item_name, data_item_stat, filter_type, value_spec</code>
参数	<p><i>view_name</i> 包含要过滤的数据项的视图名。</p> <p><i>data_item_name</i> 过滤器所应用的数据项。如果数据项包含嵌入空格，则应将其用引号括住。</p> <p><i>data_item_stat</i> <i>data_item_name</i> 的统计类型。<i>data_item_name</i> 和 <i>data_item_stat</i> 组合必须存在于视图定义中。用引号将统计类型引起来。</p> <p><i>filter_type</i> 要使用的过滤器类型。有效值为：</p> <ul style="list-style-type: none"><li>• <b>eq</b> — 让与值列表中的任何值匹配的值通过。也就是说，如果某个值等于第一个过滤器值，或者等于第二个过滤器值，或者等于列表中的任何其它值，则该值会通过过滤器。请在 <i>value_spec</i> 参数中指定值列表。</li><li>• <b>neq</b> — 让不等于值列表中的任何值的值通过。也就是说，如果某个值不等于第一个过滤器值，不等于第二个过滤器值，也不等于列表中的任何其它值，则将该值传递给过滤器。请在 <i>value_spec</i> 参数中指定值列表。</li><li>• <b>range</b> — 让某个值范围内的值通过。请在 <i>value_spec</i> 参数中指定范围。</li><li>• <b>top</b> — 让在采样间隔内为数据项接收到的最大的一些值（指定了值数量）通过。例如，“前 10 个”过滤器传递样本中 10 个具有该数据项的最大值的行。</li></ul>

*value\_spec*

指定过滤器测试的值。语法取决于 *filter\_type* 的值：

如果 <i>filter_type</i> 为	则 <i>value_spec</i> 语法为
eq 或 neq	<i>value1</i> [, <i>value2</i> ].. 有关当 <i>valuen</i> 是一个字符串时使用通配符 (%) 的信息, 请参见第 49 页的“使用通配符”。有关为对象名和过程名数据项指定值的信息, 请参见第 49 页的“为对象名和过程名指定过滤器”。
range	指定大于或等于下限的值: <i>low,value</i> 指定小于或等于上限的值: <i>high,value</i> 指定大于或等于下限且小于或等于上限的值: <i>low,value1,high,value2</i> 有关为对象名和过程名数据项指定值的信息, 请参见第 49 页的“为对象名和过程名指定过滤器”。如果为其中一个数据项指定了上限和下限, 则每个限值必须具有相同数目的子组件。
top	<i>value</i> 其中 <i>value</i> 是要传递的项数。该数必须大于 0。

示例

此示例为名为“Page I/O”的视图创建了一个过滤器。该视图包含统计类型为“会话值”(Value for Session)的页 I/O 数据项。该示例将为采样记录的行数限制为自记录会话开始到目前为止页 I/O 总数最高的前 20 行。

```
hs_create_filter PageIO,"Page I/O","Value for Session",top,20
```

用法

在一个视图中使用多个过滤器

过滤器限制 Historical Server 记录的数据量。每个过滤器应用于视图中的一个数据项。每个数据项只能应用一个过滤器。

如果一个视图中的多个数据项拥有过滤器, Historical Server 将记录满足所有过滤器的数据。可有效地对各种过滤器的值应用逻辑 AND (与) 运算。

如果 top 过滤器对某个数据项有效, 而另一个过滤器对第二个数据项有效, 则由 top 过滤器传递的一些值可能会被另一个过滤器消除掉。其结果是, 返回少于指定的 top *n* 的值。



### 使用通配符

通配符是百分比符号 (%)。当 *filter\_type* 是 *eq* 或 *neq* 时，可以在 *value\_spec* 中对返回字符串的数据项使用通配符。通配符与零个或多个字符的字符串匹配。通配符可以出现在过滤器值内的任何位置（开头、末尾或其间任何位置）。在 *value\_spec* 中只能使用单一通配符。

单一通配符规则对于对象名和过程名数据项存在例外。在这些多组件数据项中，可以在每个组件中使用单一通配符。有关详细信息，请参见第 49 页的“为对象名和过程名指定过滤器”。

当 *filter\_type* 是 *range* 时，不能在 *value\_spec* 中使用通配符。

### 为对象名和过程名指定过滤器

当 *filter\_type* 是 *eq*、*neq* 和 *range* 时，对象名和过程名数据项可在 *value\_spec* 中接受多组件值。这些复合过滤器值可以使用如下形式的名称，其中 *object\_name* 是一个表名或存储过程名：

- *object\_name*
- *owner\_name.object\_name*
- *database\_name.owner\_name.object\_name*

如果 *filter\_type* 是 *eq* 或 *neq*，则每个组件都可包含单一通配符 (%)。通配符可用于代替整个组件。例如，为对象名 *data\_item* 设置的一个值为 *%.\*.authors* 的 *value\_spec* 过滤器，将返回所有数据库中的 *authors* 表，而与表的所有者无关。有关通配符的详细信息，请参见第 49 页的“使用通配符”。

如果 *filter\_type* 是带有上、下限的 *range*，则上、下限必须拥有相同数目的子组件。

下面是用这些组件数据项设计过滤器的一些具体说明：

- 如果 *database\_name* 是某个“对象名”数据项复合过滤器值的一个组件，则视图中也必须存在一个“数据库名”数据项。
- 如果 *owner\_name* 是某个“对象名”数据项复合过滤器值的一个子组件，则视图中也必须存在一个“所有者名”数据项。
- 如果 *database\_name* 是某个“过程名”数据项复合过滤器值的一个子组件，则视图中也必须存在一个“过程名”数据项。
- 如果 *owner\_name* 是某个“过程名”数据项复合过滤器值的一个组件，则视图中也必须存在一个“过程所有者名”数据项。

## hs\_create\_playback\_session

**说明** 定义回放会话的特性。此命令是创建回放会话的第一步。

**语法** `hs_create_playback_session start_time, end_time, summarization_interval, allow_estimation, missing_data_option, target, directory_name, protection_level, sample_interval, script_type, delete_option, session_id [, session_id...]`

**参数**

*start\_time*

指定要回放的记录数据的开始日期和时间。缺省情况是从 *session\_id* 指定的第一个会话的开头开始回放。使用值 NULL 即表示接受缺省值。

*start\_time* 的格式为：

“year/month/day hour:minute[:second] [time zone]”

根据 *summarization\_interval* 和 *missing\_data\_option* 参数，同时根据在指定时间是否有可用数据，回放时可以使用晚于指定时间的数据，但不能使用早于 *start\_time* 的数据。

*start\_time*（如果指定）必须早于 *session\_id* 中指定的至少一个会话的结束时间。可以采用下列时区选项：

参数值	解释
EST	美国东部时区，标准时间。
EDT	美国东部时区，夏令时。
CST	美国中部时区，标准时间。
CDT	美国中部时区，夏令时。
MST	美国山地时区，标准时间。
MDT	美国山地时区，夏令时。
PST	美国太平洋时区，标准时间。
PDT	美国太平洋时区，夏令时。
MET	中欧时区，标准时间。
MET DST	中欧时区，夏令时。
WET DST	西欧（格林威治）时区，夏令时。
GMT	格林威治标准时间。它等同于西欧（格林威治）时区，与夏令时无关。上面的所有时区规范（例如 EST 或 EDT）都只能在标准时间或夏令时有效时，以日期和时间的组合形式提供。GMT 可以以任何日期和时间规范进行组合。
GMT{+-} <i>hours_offset</i>	用于指定任何其它时区，其中 <i>hours_offset</i> 表示一个小时数，只有在格林威治标准时间的基础上加上这个小时数才能得出当地的时间。可接受的时差值介于 +24 与 -24 之间（包括这两个值）。小数时差值（如 +5.5）也有效。

缺省 *time\_zone* 是 Historical Server 的当地时区。

#### *end\_time*

指定要回放的记录数据的结束日期和时间。缺省值是在所指定的最后一个会话终止时结束回放。使用值 NULL 即表示接受缺省值。

此参数的格式与 *start\_time* 格式相同。如果指定了 *end\_time*，则其必须迟于至少一个输入会话的 *start\_time*。

#### *summarization\_interval*

必需的参数（无缺省值），指定回放的详细程度。有效值为：

- **raw** — 按照数据的收集方式回放数据，二者使用相同的采样间隔。选择该选项可查看到与记录时一样的原始数据。此外，这是可用于回放快照数据（如当前的 SQL 语句数据和锁或进程的状态信息等）的唯一选项。有关快照数据项的确切列表，请参见第 150 页的表 C-2。（快照数据项是那些在“非原始允许”列中值为“否”的数据项。）

只有当 *target* 是 **client** 时，该选项才有效。

- **actual** — 使用与输入记录会话相同的采样间隔来回放数据。

该选项使您可以在回放视图中指定一些数据项变化。另外，当记录会话时间与请求的回放会话时间不一致时，Historical Server 将对第一个和最后一个样本进行适当调整。

不需要汇总时，选择该选项可增加或改变某些数据项。此外，当 *target* 被设置为文件时，该选项提供了在多个记录会话中并置非快照数据的方法。

只有当 *target* 是 **client** 时，该选项才有效。

- **entire** — 回放被汇总成单个采样的每个输入记录会话的数据。采样间隔是所请求的回放 *start\_time* 和 *end\_time* 之间的时间跨度。

该选项使您可以在回放视图中指定一些数据项变化。另外，Historical Server 将对数据值进行适当调整，以准确反映所请求的回放会话的开始和结束时间。

选择该选项可合并记录的数据，将长时间积累的详细内容合并到活动概述中。

*sample\_interval*

回放概括到具有指定长度的采样间隔内的数据。参数值是采样间隔长度，按如下方式指定：

```
"S"  
"M: S"  
"H: M: S"  
"D H: M: S"
```

其中：

- *S* — 是秒数。
- *M* — 是分钟数。
- *H* — 是小时数。
- *D* — 是天数。

每一部分都是一个一位或两位数字。以下是一些示例：

```
"30" (指定 30 秒作为采样间隔)  
"10:0" (指定 10 分钟作为采样间隔)  
"8:30:0" (指定 8.5 小时作为采样间隔)  
"5 0:0:0" (指定 5 天作为采样间隔)
```

第一个采样间隔从 *start\_time* 开始，每个采样都有指定的长度（最后一个可能有例外）。

该选项使您可以在回放视图中指定一些数据项变化。另外，**Historical Server** 将对数据值进行适当调整，以准确反映所请求的回放会话长度和回放采样间隔。

选择该选项可将数据按任意需要的间隔进行汇总。这种汇总可以调节活动中的偏差，对于观察随时间变化的趋势很有用。

**actual**、**entire** 和用户定义的间隔选项有下列共同特性：

- 回放视图中的数据项变化 — 对于这三种选项，回放视图可以使用与输入视图不同的统计类型，并且可以包含输入视图中没有的一些估计和计算出来的数据项。它不能包含快照数据，例如当前 SQL 语句数据。有关选择了这些选项后回放视图中可包含的数据项和统计类型的确切列表，请参见 [第 150 页的表 C-2](#)。

- 数据调整 — 对于所有这三种选项，Historical Server 执行适当的数学计算以补偿下列各项之间存在的差异：
  - 输入记录会话采样间隔和请求回放采样间隔
  - 输入记录会话的会话长度和请求回放的开始与结束时间
 要处理这些差异，对累计的计数进行按比例调整。百分比和比率都进行加权平均，加权因子为每个输入采样贡献的秒数。有关详细信息，请参见附录 C “定义回放视图的规范”。

#### *allow\_estimation*

指定回放时是否估计不能精确计算的数据项的值。有效值为：

- **disallow**（缺省值）— 如果它遇到需要估计的数据项，则会使 `hs_create_playback_view` 命令返回一个错误。
- **allow** — 必要时让回放估计某些数据项的值。有些数据项只有当允许在回放会话中估计后才能包含于回放视图中。

当 *summarization\_interval* 是 **raw** 时，将忽略此参数。

请参见第 150 页的表 C-2 了解哪些数据项需要估计。

#### *missing\_data\_option*

当 *target* 是 **client** 时，此参数指定当输入会话中没有数据时，`hs_playback_sample` 命令如何处理各时间段。有效值为：

- **skip**（缺省值）— 当某个时间段内不含数据时，`hs_playback_sample` 命令直接转到包含数据的下一个时间段，而不是返回不含数据的样本。
- **show** — 即使对没有数据的时间段，`hs_playback_sample` 命令也返回一个样本。在客户端回放期间，将返回含零行的列标头。

当 *target* 是 **file** 时，不允许有间隔。

#### *target*

指定回放会话的目标结果。有效值为：

- **client**（缺省值）— 对客户端启用回放。
- **file** — 允许创建包含由此 `hs_create_playback_session` 命令和随后的 `hs_create_playback_view` 命令指定的所有数据的新会话。使用 `hs_initiate_playback` 命令创建新会话。如果 *summarization\_interval* 参数是 **raw** 或 **actual**，则不允许将 *target* 设置为 **file**。

*directory\_name*

当 *target* 是 **file** 时，此参数指定创建新文件的目录。当 *target* 是 **client** 时，将忽略该参数。缺省目录是 Historical Server 主目录。

启动 Historical Server 的操作系统帐户必须对所指定的目录拥有执行（搜索）和写权限。

*protection\_level*

当 *target* 是 **file** 时，此参数为新创建的会话指定 **hs\_initiate\_playback** 命令分配给数据文件的权限级别。保护级别控制如何查看控制文件中的元数据（使用 **hs\_list** 命令），并控制如何创建带有数据的回放会话（使用 **hs\_create\_playback\_session** 命令）。有效值为：

- **private**（缺省值）— 指定新文件是带口令保护的。只有创建这些文件的帐户或 Historical Server 超级用户才可访问它们。
- **public** — 允许不受限制地访问新会话的文件中的数据。
- **null** — 采用缺省值，即 **private**。

当 *target* 是 **client** 时，将忽略该参数。

*script\_type*

当 *target* 是 **file** 时，此参数指定是否要 **hs\_initiate\_playback** 命令为新建的会话创建脚本文件。该脚本文件包含为在新会话中定义的每个回放视图创建 Adaptive Server 表的 SQL 命令。有效值为：

- **no\_script**（缺省值）— 不为新会话创建脚本文件。
- **sybase\_script** — 创建脚本文件。该文件位于 Historical Server 主目录中。其名称为 *sSessionId*，其中：
  - *s* 是一个常量。
  - *SessionId* 由 Historical Server 分配。
- **null** — 采用缺省值，即 **no\_script**。

当 *target* 是 **client** 时，将忽略该参数。

*delete\_option*

当 *target* 是 *file* 时，此参数指定是否让 `hs_initiate_playback` 命令在成功创建新会话后删除输入会话文件。有效值为：

- **retain**（缺省值）— 不删除包含输入会话数据的文件。
- **delete** — 如果创建新会话成功，则删除所有输入会话文件。用户必须是 Historical Server 超级用户或所有输入会话的所有者。
- **null** — 采用缺省值，即 **retain**。

当 *target* 是 *client* 时，将忽略该参数。

*session\_id[,session\_id...]*

为要回放的输入会话指定唯一的标识符。至少需要一个 *session\_id*。如果使用了多个 ID，则必须按照其开始时间以正确的顺序指定，开始时间最早的最先指定。回放将跳过顺序混乱的会话。如果 *summarization\_interval* 是 *raw*，那么只有一个 *session\_id* 是有效的。

在定义一个记录会话时，Historical Server 为该记录会话分配一个会话 ID。使用 `hs_list` 命令可查找要回放的会话 ID。

如果 *target* 是 *file*，则在一个输入会话的结束时间与下一个输入会话的开始时间之间不能有间隙。另外，所指定的每个会话必须已完成其记录过程。

如果 *target* 是 *client*，那么每个会话必须已开始记录，但它在提交 `hs_create_playback_session` 命令后可继续记录。

## 示例

- 1 此示例以单个输入会话（会话 7）为基础创建一个回放会话。回放整个会话，不提取摘要或进行规范化，也不创建新会话。

```
hs_create_playback_session null, null, raw,
null, null, client, null, null, null, 7
```

- 2 此示例以三个输入会话（会话 4、6 和 9）为基础创建一个回放会话。只回放从 9:00 到 5:00 的时间段内采集的数据。回放的数据每半小时提取一次摘要。将回放保存为一个新会话。

```
hs_create_playback_session "1996/5/3 9:00",
"1996/5/3 17:00", "0 00:30:00", disallow, skip,
file, null, public, no_script, retain, 4, 6, 9
```

### 用法

- 一个 Historical Server 连接可定义一个记录会话，或者定义和执行一个回放会话，但不能两者兼施。在成功执行 `hs_create_playback_session` 命令后，必须成功执行一次 `hs_terminate_playback` 命令才能开始定义一个记录会话或开始定义另一个回放会话。
- 回放会话的行为因 *target* 参数值的不同而有所不同：

- 如果 *target* 是 `client`，则该客户端使用 `hs_playback_sample` 命令检索数据。回放可以发生在仍然处于记录过程的记录会话之中。
- 如果 *target* 是 `file`，那么客户端不为该回放会话检索任何数据；而是由 Historical Server 创建一个新会话，并用该回放会话的数据填写此新会话的数据文件。新会话的可访问性由 *protection\_level* 参数确定。调用此命令的用户即该新会话的所有者。`hs_initiate_playback` 命令创建新会话。

回放只限于提交 `hs_create_playback_session` 命令时已完成记录的那些会话。

- 如果 *target* 是 `file`，并且使用了多个回放会话作为输入，则不允许有时间间隙。例如，如果从星期一到星期五，每天从上午 9 点到下午 5 点收集数据，则不能回放这五个记录会话来创建一个新的、汇总的一周会话。但是，如果从星期一到星期五，每天从上午 9 点到第二天上午 9 点，不间断地收集数据，则可以使用回放功能创建一个新的、汇总的一周会话。另一个不间断的方法是保留上午 9 点到下午 5 点的记录会话，但另增加一组计划从下午 5 点到第二天上午 9 点的记录会话。对非高峰时间采用较长采样间隔，从而减少收集的数据量。
- `hs_create_playback_session` 命令中所用的 *summarization\_interval* 参数的值会影响 `hs_create_playback_view` 命令用于定义视图的规则。



## hs\_create\_playback\_view

**说明** 从要包括在回放会话内的输入记录会话中，指定一个视图。还需指定要回放的每个视图中的数据项。

**语法** `hs_create_playback_view view_name, [data_item_name_1, data_item_stat_1 [, data_item_name_2, data_item_stat_2]...]`

**参数** *view\_name*  
要回放的视图名。回放视图名必须与输入会话中的相应视图名相匹配。如果有两个或更多输入会话，那么该视图必须存在于所有的输入会话中，并且在所有输入会话中必须包含完全相同的数据项和过滤器。

对于一个给定的视图名，只能定义一个回放视图。如试图用相同名称创建第二个回放视图，则会失败。

使用 `hs_list` 命令列出输入会话中的视图以及每个视图中的数据项。

*data\_item\_name n*  
希望在回放视图中查看的输入视图中现有的数据项。（在某些情况下，可以引入不在输入视图中的新数据项。）如果数据项包含嵌入空格，则应将其用引号括住。

如果没有指定任何数据项，那么使用输入会话中相应视图内的所有数据项定义该视图。但是，如果相应视图中的某个数据项在回放视图中无效，则出现错误。

有关设计回放视图的详细信息，请参见第 150 页的表 C-2。

*data\_item\_stat n*  
各数据项的统计类型。统计类型不必与输入视图中所用的统计类型相同。请参见第 150 页的表 C-2 以确定回放视图中的数据项的有效统计类型。

用引号括住统计类型。

**示例** 1 此示例基于输入会话中的 *device\_view* 视图创建一个回放视图。回放视图中包括输入视图中的所有数据项，并具有相同的数据项和统计类型组合：

```
hs_create_playback_view device_view
```

2 此示例从输入会话创建一个回放视图，该视图返回一个时间戳和两个数据项：

```
hs_create_playback_view
device_view,"Timestamp","Value for
Sample","Device Name", "Value for Sample",
"Device Reads","Rate for Sample"
```

- 用法
- `hs_create_playback_view` 命令仅在定义回放会话期间有效；也就是说，它必须在成功执行一次 `hs_create_playback_session` 命令之后发出，但必须是在成功执行 `hs_initiate_playback` 命令之前。
  - 回放会话必须至少包含一个视图。
  - [附录 C “定义回放视图的规范”](#) 解释设计有效回放视图的要求，包括回放视图的有效的数据项和统计类型组合。

## hs\_create\_recording\_session

说明 建立新记录会话。

语法 `hs_create_recording_session monServerName, sample_interval`  
`[,dir_name][,start_time][,end_time]`  
`[,protection_level][,error_option][,script_type], [tab_delimited]`

参数 *monServerName*

Monitor Server 的名称，用于从要为其收集历史数据的 Adaptive Server 收集数据。用于连接到 Historical Server 的用户名和口令，必须与由此处指定的 Monitor Server 所监控的 Adaptive Server 上的有效登录帐户相匹配。

*sample\_interval*

确定各次采样之间的时间（以秒计）。最短的有效采样间隔是 1 秒，最长的有效采样间隔是 86400 秒（即一天）。

*dir\_name*

目录的路径名，将在此处存放从该记录会话获取的历史数据文件。启动 Historical Server 的操作系统帐户必须对该目录拥有搜索（执行）和写权限。

缺省目录是 Historical Server 主目录，它在 Historical Server 启动命令的 `-D` 参数中指定。

*start\_time*

开始记录的日期和时间，使用的格式为：

`year/month/day hour:minute[:second] [time zone]`

缺省值是立即开始。可以采用下列时区选项：

参数值	解释
EST	美国东部时区，标准时间。
EDT	美国东部时区，夏令时。
CST	美国中部时区，标准时间。
CDT	美国中部时区，夏令时。
MST	美国山地时区，标准时间。
MDT	美国山地时区，夏令时。
PST	美国太平洋时区，标准时间。
PDT	美国太平洋时区，夏令时。
-T	中欧时区，标准时间。
MET DST	中欧时区，夏令时。
WET DST	西欧（格林威治）时区，夏令时
GMT	格林威治标准时间。它等同于西欧（格林威治）时区，与夏令时无关。上面的所有时区规范（例如 EST 或 EDT）都只能在标准时间或夏令时有效时，以日期和时间的组合形式提供。GMT 可以以任何日期和时间规范进行组合。
GMT{+-} <i>hours_offset</i>	用于指定任何其它时区，其中 <i>hours_offset</i> 表示一个小时数，只有在格林威治标准时间的基础上加上这个小时数才能得出当地的时间。可接受的时差值介于 +24 与 -24 之间（包括这两个值）。小数时差值（如 +5.5）也有效。

如果不指定时区，则使用 Historical Server 的当地时区。

*start\_time* 不能大于从当前时间算起的 31 天。

#### *end\_time*

停止记录的时间。缺省设置是在 *start\_time* 后 24 小时停止记录会话。如果指定了 *end\_time* 但未指定 *start\_time*，则 *start\_time* 缺省为当前时间。

#### *protection\_level*

指定在此记录会话中创建的数据文件是带口令保护的还是所有用户都可访问。保护级别控制如何查看控制文件中的元数据（使用 `hs_list` 命令），并控制如何创建带有数据的回放会话（使用 `hs_create_playback_session` 命令）。

- `private`（缺省值）— 指定记录数据是带口令保护的。只有创建这些文件的帐户或 Historical Server 超级用户才能访问它们。
- `public` — 指定对所记录数据的访问将不受限制。

*error\_option*

指定在记录会话期间希望 Historical Server 如何处理非致命错误。有效值为：

- **continue**（缺省值）— 指定当检测到非致命错误时继续记录。例如，如果 Monitor Server 在提取监控信息摘要时填充其所有已配置的缓冲区，将向 Historical Server 发送一个或多个非致命错误，但 Historical Server 仍可以从当前采样和以后的采样中收集可用数据。
- **halt**— 指定当检测到非致命错误后终止记录会话。

*script\_type*

指定是否希望 Historical Server 用记录会话视图定义创建用于创建表的脚本文件。

- **sybase\_script**— 创建包含 SQL 命令的脚本文件，这些命令为记录会话中的每个视图创建一个 Adaptive Server 表。
- **no\_script**（缺省值）— 不创建脚本文件。

*tab\_delimited*

在将此项添加到 `hs_create_recording_session` 后，将忽略 HS 当前使用的缺省输出文件分隔符。将改用制表符。

示例

此示例创建一个记录会话，以捕获来自名为 `SERVER1_MON` 的 Monitor Server 的数据。数据每 30 秒钟捕获一次，并写入位于 `/user/hist_dir` 目录下的数据文件中，从现在开始，到 1997 年 8 月 8 日上午 10:30EDT（东部时区，夏令时）结束。所产生的文件具有专用限制。如果检测到非致命错误，记录会话仍继续。将创建一个脚本文件，用于为每个已记录的视图创建 Adaptive Server 表。

```
hs_create_recording_session SERVER1_MON,30,  
/user/hist_dir,NULL,"97/08/08 10:30 EDT",  
private,continue, sybase_script
```

用法

- 致命错误包括崩溃和与 Adaptive Server 或 Monitor Server 连接时发生的其它故障。致命错误强迫终止记录会话，而不管 *error\_option* 的值是什么。
- 必须首先发出此命令，然后才能为记录会话定义任何视图、报警或过滤器。在发出此命令并定义了至少一个视图之后，可以用 `hs_initiate_recording` 命令开始记录。

## hs\_create\_view

说明	定义会话期间要记录的一组数据项。
语法	<pre>hs_create_view view_name,                data_item_name_1, data_item_stat_1                [,data_item_name_2, data_item_stat_2]...</pre>
参数	<p><i>view_name</i> 用户定义的视图名。视图名必须由 a - z、A - Z 和 0 - 9 的字符以及下划线字符 ( _ ) 构成。</p> <p><i>data_item_name_n</i> 参与组成视图的数据项。如果数据项包含嵌入空格，应将其用引号括住。</p> <p>有关可用的数据项的列表，请参见第 93 页的表 A-1。有关视图内数据项的有效组合，请参见第 112 页的表 B-2。</p> <p><i>data_item_stat_n</i> 各数据项的统计类型。统计类型必须用引号引起来。有效的统计类型是：</p> <ul style="list-style-type: none"> <li>• “采样值”</li> <li>• “会话值”</li> <li>• “采样速率”</li> <li>• “会话速率”</li> <li>• “采样平均值”</li> <li>• “会话平均值”</li> </ul> <p>并不是所有统计类型对所有数据项都有效。请参见第 136 页的表 B-3。</p>
示例	<p>此示例创建一个名为“Page I/O”的视图，该视图包含统计类型为“会话值” (Value for Session) 的页 I/O 数据项，以及统计类型为“采样值” (Value for Sample) 的内核进程 ID 数据项。</p> <pre>hs_create_view PageIO, "Page I/O", "Value for Session", "Kernel Process ID", "Value for Sample"</pre>
用法	<ul style="list-style-type: none"> <li>• 选项的指定顺序决定了数据项存储在视图的记录数据文件中的顺序。</li> <li>• 必须为每个记录会话至少定义一个视图。</li> </ul>

- 如果想使用 Historical Server 回放功能来检验记录会话中的数据，那么在设计记录会话视图时也许要考虑回放视图。

对于很多数据项，Historical Server 可使用与记录期间不同的统计类型来回放数据项。但是，对于有些数据项，Historical Server 通过估计值（相对于精确值）来总结数据，除非已在记录数据中包含了某些其它数据项。请参见第 150 页的表 C-2 以确定哪些数据项是精确计算时必需的。

## hs\_delete\_data

说明	删除与一个或多个不活动的记录会话关联的历史监控文件。
语法	<code>hs_delete_data low_session_id [,high_session_id]</code>
参数	<p><i>low_session_id</i> 要删除其数据的第一个记录会话的唯一标识符。</p> <p><i>high_session_id</i> 要删除其数据的会话范围中的最后一个会话的唯一标识符。如果省略此参数，则只有由 <i>low_session_id</i> 标识的会话能删除它的数据。</p>
示例	此示例删除会话 ID 为 1 至 15 的所有不活动记录会话的历史监控文件。
用法	<pre>hs_delete_data 1,15</pre> <ul style="list-style-type: none"><li>• 能删除历史数据文件的用户只能是：<ul style="list-style-type: none"><li>• Historical Server 超级用户</li><li>• 使用与定义记录会话时相同的用户名和口令连接到 Historical Server 的用户</li></ul></li><li>• 只能删除不活动的记录会话文件。与活动记录会话关联的文件不能被删除，不管其会话 ID 是否在 <code>hs_delete</code> 命令指定的范围内。</li><li>• 不活动记录会话是指因达到 <i>end_time</i> 时间或因发出了 <code>hs_terminate_recording</code> 命令而终止的会话。</li></ul>

## hs\_initiate\_playback

说明	指定已完成回放会话定义和准备执行回放。如果回放目标是到文件，则此命令开始回放。如果回放目标是到客户端，则此命令初始化回放，以使带有缺省 <i>step</i> 的 <code>hs_playback_sample</code> 命令返回回放会话中的第一个样本。
语法	<code>hs_initiate_playback</code>
用法	<ul style="list-style-type: none"><li>• <code>hs_create_playback_session</code> 命令和一个或多个成功的 <code>hs_create_playback_view</code> 命令必须在 <code>hs_initiate_playback</code> 命令之前执行。</li><li>• 如果在执行 <code>hs_initiate_playback</code> 命令之前未创建任何视图，那么 <code>hs_initiate_playback</code> 返回一个错误。用户仍然可以尝试使用 <code>hs_create_playback_view</code> 命令，也可以用 <code>hs_terminate_playback</code> 命令取消回放会话。</li></ul>

## hs\_initiate\_recording

说明	指定记录会话的定义已完成，并请求于记录会话的 <i>start_time</i> 时间开始记录或安排在此时间开始记录。
语法	<code>hs_initiate_recording</code>
用法	<ul style="list-style-type: none"><li>• <code>hs_create_recording_session</code> 命令和一个或多个 <code>hs_create_view</code> 命令必须在 <code>hs_initiate_recording</code> 命令之前执行。</li><li>• 一个或多个 <code>hs_create_alarm</code> 和 <code>hs_create_filter</code> 命令也可以在 <code>hs_initiate_recording</code> 命令之前执行。</li><li>• 在执行 <code>hs_initiate_recording</code> 命令后，稍后可以取消会话，但不能再将更多其它视图、过滤器或报警添加到此记录会话的说明中。</li></ul>

## hs\_list

说明

列出有关过去和当前记录会话的信息。

语法

`hs_list level [,restriction]`

参数

*level*

指定为每个记录会话返回的细节等级。它的值可以是 `sessions`、`views`、`data_items`、`alarms`、`filters` 或 `summarization_data_items`。

*restriction*

为给定 *level* 选择关于记录会话的可用数据的子集。

---

### 如果 level 是 则 restriction 是

---

`sessions`

- 下列值之一：
  - `active` — 将返回的数据限制为此 Historical Server 实例的当前活动记录会话。
  - `inactive` — 仅选择已完成的记录会话。
  - `latest` — 仅选择在当前客户端连接中最新启动的记录会话（如果存在）。
- 如果省略了 *restriction*，则将列出用户有权访问的所有活动和非活动记录会话。

---

`views`

- 要列出其视图的会话 ID。
- 如果省略了 *restriction*，则将列出用户有权访问的所有记录会话的视图。

---

`data_items`、  
`alarms`、`filters`  
或  
`summarization`  
\_ `data_items`

- 一个会话 ID 和（可选）一个要列出其数据项的视图名。语法为：

`session_id [,view_name]`

*session\_id* 将列表限制为只列出参与单个会话的数据项。如果指定了 *view\_name*，则将列表限制为列出在为会话定义的单个视图中出现的数据项。

- 如果省略了 *restriction*，则将列出属于为您有权访问的所有活动和非活动记录会话定义的所有视图的数据项。
- 

示例

1 此示例列出该 Historical Server 实例的所有活动会话：

```
hs_list sessions,active
```

2 对于会话 ID 为 10 的会话的页 I/O 视图，此示例列出为其中的数据项定义的所有报警：

```
hs_list alarms,10,PageIO
```



## 用法

- `hs_list` 命令将下列字段作为整数数据类型返回：
  - 会话 ID
  - 采样间隔
  - 报警计数
  - 过滤器计数
- `hs_list` 命令将以下字段返回为浮点数据类型：
  - 报警值
- 所有其它字段都作为字符串返回。
- 如果一个会话是以 `private` 保护级别记录的，并且当前用户不是 `Historical Server` 超级用户，则会对照在使该会话可见之前记录该会话的用户的名称和口令来检验当前用户的用户名和口令。

## 返回数据说明

由 `hs_list` 返回的数据取决于 `level` 的值：

- `sessions` 使每个会话返回单独一行，其中含有以下的列：
  - 会话 ID（会话的唯一标识符）
  - 会话状态：
    - *活动*
    - *不活动的*
    - *远程活动或不活动*— 不能明确确定会话的状态。对于与当前实例使用相同主目录的其它 `Historical Server` 实例，其当前活动会话也归入此类别。（有关运行 `Historical Server` 的多个实例和共享控制文件的详细信息，请参见第 20 页的“配置多个 `Historical Server` 实例”。）因正在运行的 `Historical Server` 实例异常终止而结束的会话以及还未达到其预定结束时间的会话也归入此类别。
  - 启动记录会话的用户的名称
  - 监控的 `Adaptive Server` 的名称
  - 使用的 `Monitor Server` 名
  - 记录会话的开始日期和时间
  - 记录会话的结束日期和时间
  - 包含记录数据的目录

- 使用的采样间隔  
如果该值为 0，这意味着会话由 *summarization\_level* 为 *entire* 的回放创建。（整个会话由一个样本表示，并且没有采样间隔。）
- 所用的错误选项（发生非致命错误时继续或暂停）
- *views* 为每个视图生成一行，其中含如下的列：
  - 会话 ID
  - 视图名
- *data\_items* 为视图中定义的每个数据项生成一行，其中含如下的列：
  - 会话 ID
  - 视图名
  - 数据项
  - 数据项统计类型
  - 为数据项定义的报警数
  - 为数据项定义的过滤器数
  - 下列关键字之一：
    - *recorded* — 表示数据来自原记录会话或在原会话中保留。
    - *summarized* — 表示在与原会话不同的采样间隔内汇总。
    - *estimated* — 表示数据是在汇总期间估计出来的，而非准确计算出来的。
- *alarms* 为视图中的某数据项定义的每个报警生成一行，其中含如下的列：
  - 会话 ID
  - 视图名
  - 数据项
  - 数据项统计类型
  - 报警操作（*log* 或 *execute*）
  - 日志文件名或要执行的文件名
  - 报警值

- **filters** 为视图中的某数据项的每个过滤器生成一行，其中含如下的列：
  - 会话 ID
  - 视图名
  - 数据项
  - 统计类型
  - 过滤器类型（eq、neq、range 或 top）
  - 指定值（返回为单个字符串）
- **summarization\_data\_items** 从视图中为非原始回放请求的每个数据项和统计类型组合生成一行，其中含如下的列：
  - 会话 ID
  - 视图名
  - 数据项
  - 统计类型
  - 下列关键字之一：
    - **recorded** — 表示如果回放该数据项，则该数据项的数据将是来自原记录会话的数据或在原会话中保留的数据。
    - **summarized** — 表示如果回放该数据项，则该数据项的数据将以汇总形式提供。
    - **estimated** — 表示如果回放该数据项，则该数据项的数据将只能用估计值形式提供。

## hs\_playback\_sample

说明	当回放 <i>target</i> 是 <i>client</i> 时，回放一个采样。 <i>target</i> 在 <code>hs_create_playback_session</code> 命令中定义。
语法	<code>hs_playback_sample [step [, retry_count] ]</code>
参数	<p><i>step</i></p> <p>指定要发送到客户端的采样，相对于在最近为当前回放会话执行的 <code>hs_playback_sample</code> 命令中发送的采样而言。有效值是 0 或正数。</p> <p>缺省值是 +1，这时将发送随后的下一个采样。在开始处， <i>step</i> 值为 +1 返回第一个采样。 0 值则重新发送最近发送的采样。 <i>step</i> 值为 +2 则发送 <i>step</i> 值为 +1 时所发送采样的下一个采样，依此类推。如果试图显示时间迟于最后一个采样的采样，将返回一个错误消息。</p> <p><code>hs_initiate_playback</code> 命令初始化回放，以使缺省值 (+1) 发送回放会话中的第一个采样。</p> <p>构成采样的数据的定义受 <code>hs_create_playback_session</code> 命令的 <i>summarization_interval</i> 和 <i>missing_data_option</i> 参数的影响。</p> <p>如果创建回放会话时使用的 <i>summarization_interval</i> 参数值为 <i>actual</i>、<i>entire</i> 或用户指定的间隔，并且回放会话内的任何视图中含有统计类型为 “Value for Session” 或 “Rate for Session” 的数据项，那么 <i>step</i> 的唯一可允许值是 +1，即缺省值。</p> <p><i>retry_count</i></p> <p>对于回放时仍在进行记录的记录会话，指定 Historical Server 将重试从其历史数据文件读取数据的次数（间隔为 1 秒）。</p> <p>缺省值是 0，即不进行重试，如果在读取历史数据文件时回放到达文件结尾状态，将返回一个错误。</p>
示例	<ol style="list-style-type: none"> <li>1 此示例回放下一个采样： <pre>hs_playback_sample</pre> </li> <li>2 此示例跳过一个采样，回放下一个采样。如果正在进行记录，Historical Server 在到达文件结尾时最多重试 30 次以检索另一个采样。 <pre>hs_playback_sample 2, 30</pre> </li> </ol>

## 用法

## 所返回数据的格式

`hs_playback_sample` 命令以表形式返回数据。它为每个回放视图返回一个表。这些表的排列顺序与提交创建视图的

`hs_create_playback_view` 命令的顺序相同。

每个表中的列与相应的回放视图中的数据项相对应。这些列按定义回放视图时列出数据项的顺序排列。请参见第 136 页的表 B-3 以确定每一列的数据类型。

每一行代表视图中的关键数据项的一个不同组合。如果视图不含关键数据项，则该表返回反映服务器级数据的单独一行。

当没有要报告的活动时，有些视图返回含 0 值的行，其它视图则省略这一行。控制是否在视图中显示含 0 值的行的规则是：

- 含有服务器级数据项的视图总是返回一行，而不管是否有要报告的活动。
- 当没有要报告的活动时，包含关键数据项进程 ID、对象 ID 或过程 ID 的视图省略这一行。有关何时返回进程 ID 的详细说明，请参见第 69 页的“含有进程 ID 的视图”。
- 包括未在前一项中列出的键的视图即使在没有活动时也返回行。

如果回放正在汇总数据，则当任何输入会话中的任何采样包含相同的键组合时，将为这些键组合返回一行。

如果某个整数数据项出现上溢，Historical Server 将返回该数据项中的最大有效数值，并返回一个信息性消息给客户端。

## 含有进程 ID 的视图

当服务器进程终止时，Adaptive Server 会将它的进程 ID 重新用于一个新的进程。因此，“进程 ID”数据项不能保证唯一地标识进程。但“内核进程 ID”数据项可以唯一地标识进程。

包含“进程 ID”的视图按如下方式返回行：

- 记录会话视图（因此也包括原始回放视图）只为代表采样间隔结束时存在的进程的进程 ID 返回一行。如果某个服务器进程在一个采样间隔中间终止，则不为其进程 ID 返回行。
- 摘要回放视图不要求服务器进程在回放采样期间存在。对于包含在任何输入视图中的所有服务器进程，摘要视图都将返回行。但是，由于进程 ID 不能保证唯一性，因此必须在包含进程 ID 的任何摘要视图中包括内核进程 ID 以确保键的唯一性。否则，摘要中可能错误地组合来自两个不同进程的数据。

### 回放会话内的数据间隙

`hs_create_playback_session` 命令的 `start_time` 和 `end_time` 参数定义回放会话包括的时间段。如果 `hs_create_playback_session` 命令的 `target` 参数是 `client`，则在任何输入会话中都没有数据时，对应回放会话的开始时间和结束时间之间可以有时间间隔。

---

**注释** 当 `target` 是 `file` 时，在指定的输入会话中不允许有间隙，因此回放会话中将没有间隙。

---

在下列情况下，可用的数据中可能出现间隙：

- 为回放会话指定的输入会话不是连续的。例如，如果一个输入会话系列包括星期一至星期五的上午 9 点到下午 5 点的时间，则每天下午 5 点到第二天上午 9 点之间可能存在数据间隙。在星期五晚上到星期一上午之间可能存在更大的间隙。
- 回放会话进行汇总，其开始或结束时间超出能够从输入会话获取数据的时间。

当回放目标是客户端时，Historical Server 根据

`hs_create_playback_session` 命令的 `missing_data_option` 参数的值来处理间隙：

- `skip`（缺省值）— 当某个时间段内不含数据时，`hs_playback_sample` 命令直接转到包含数据的下一个时间段，而不是返回不含数据的样本。
- `show` — 即使对没有数据的时间段，`hs_playback_sample` 命令也返回一个样本。不返回数据（所返回的每个数据表的长度是 0）。

在这种情况下，为使客户端接收到时间戳，必须有一个仅包含 `Timestamp` 或 `Timestamp Datim` 数据项的回放视图。包含任何其它数据项的视图将返回 0 个行。

回放会话的汇总级别对回放期间返回的空采样数有影响：

- 如果 `summarization_level` 是 `raw` 或 `actual`，则整个间隙用一个采样表示。例如，下午 5 点到上午 9 点之间的数据间隙将用一个采样表示。
- 如果 `summarization_level` 是一个时间间隔，例如 1/2 小时（表示为 "0 00:30:00"），则间隙用每个时间间隔的一个采样表示。例如，如果时间间隔指定为 1/2 小时，下午 5 点到上午 9 点之间的数据间隙将用 32 个空采样表示。
- 如果 `summarization_level` 是 `entire`，则忽略间隙。

## hs\_shutdown

说明	关闭 Historical Server。
语法	<code>hs_shutdown [wait_option]</code>
参数	<p><i>wait_option</i></p> <p>指定是立即终止还是等待活动连接先关闭。有效值为：</p> <ul style="list-style-type: none"><li>• <code>no_wait</code> — 让 Historical Server 立即关闭并终止所有连接（包括那些执行回放的连接）和活动的记录会话。正在创建新会话的活动记录会话和回放会话以受控方式关闭，这样可以防止破坏控制文件或任何历史数据文件。</li><li>• <code>wait</code>（缺省值）— 延迟关闭操作，直到所有活动记录会话（包括执行回放的那些会话）完成且所有其它活动连接关闭为止。在此时间内不接受任何新连接。</li></ul>
示例	<p>此示例立即关闭 Historical Server 并终止所有连接和活动记录会话。</p> <pre>hs_shutdown no_wait</pre>
用法	<ul style="list-style-type: none"><li>• 如果使用不带 <code>no_wait</code> 选项的 <code>hs_shutdown</code> 命令，然后想取消此请求，可以向 Historical Server 发送一个中断指令来取消关闭请求。DB-Library 应用程序调用 <code>dbcancel()</code> 函数，CT-Library 应用程序调用 <code>ct_cancel()</code> 函数，以产生这种中断指令。向 <code>isql</code> 命令发送一个中断信号（通常通过键入 <code>Ctrl+C</code> 来实现）后，该命令将生成一个上述调用。</li><li>• 您可以随时发送下列四种信号之一来请求立即关闭 Historical Server:<ul style="list-style-type: none"><li>• <code>SIGQUIT</code>（退出）</li><li>• <code>SIGINT</code>（中断）</li><li>• <code>SIGABORT</code>（中止）</li><li>• <code>SIGTERM</code>（终止）</li></ul>任何时候接收到这些信号之一（包括在等待 <code>hs_shutdown</code> 命令完成时）都与发出 <code>hs_shutdown no_wait</code> 命令等效。</li></ul> <hr/> <p><b>警告！</b> Sybase 强烈建议不要使用注销信号 (<code>SIGKILL</code>) 来关闭 Historical Server。注销信号不允许完成任何受控数据清理，可能导致数据丢失。</p> <hr/>

- 如果在关闭期间终止了任何活动会话，Historical Server 会将有关这些会话的信息写入其日志文件中。活动会话是指正在创建新记录会话过程中的记录会话或回放会话。
- 如果在 Historical Server 启动命令中指定了超级用户，则该超级用户是唯一能关闭 Historical Server 的用户。如果未指定超级用户，那么任何用户都能关闭 Historical Server。
- 有关停止 Historical Server 的详细信息，请参见第 3 章“启动和停止 Historical Server”。

## hs\_status

说明  
语法  
参数

获取状态信息。

`hs_status option`

*option*

下列指定值之一：

- `directory` — 显示 Historical Server 主目录名，该目录名由 Historical Server 启动命令的 `-D` 参数指定。
- `superuser` — 显示 Historical Server 超级用户名称，该名称由 Historical Server 启动命令的 `-U` 参数指定（如果省略此参数，则为 NULL）。
- `interfaces` — 显示 Historical Server 正在使用的 *interfaces* 或 *sql.ini* 文件的路径名。
- `max_connections` — 显示允许的并行客户端连接的最大数目。
- `logfile` — 显示 Historical Server 日志文件的名称。
- `version` — 显示 Monitor Historical Server 的版本字符串和版权信息。



- `activity` — 显示关于 Historical Server 上的当前活动级别的下列信息：

连接数	与 Historical Server 的连接，包括当前连接
活动记录会话数	使用 <code>hs_create_recording_session</code> 创建并使用 <code>hs_initiate_recording</code> 启动的会话。包括还未到达其开始时间的会话。
未启动的记录会话	使用 <code>hs_create_recording_session</code> 创建但未执行匹配的 <code>hs_initiate_recording</code> 的会话。
活动回放会话数	使用 <code>hs_create_playback_session</code> 创建并使用 <code>hs_initiate_playback</code> 启动的会话。
未启动的回放会话数	使用 <code>hs_create_playback_session</code> 创建但未执行匹配的 <code>hs_initiate_playback</code> 会话。

活动信息与停止 Historical Server 的时间相关，因为当存在其它连接或有任何活动会话时，Historical Server 不会关闭，除非为 `hs_shutdown` 命令明确地请求了 `no_wait` 选项。

示例

此示例显示 Historical Server 主目录名：

```
hs_status directory
```

用法

`hs_status` 命令将 `max_connections` 字段和所有 `activity` 字段作为整数数据类型返回。所有其它字段都作为字符串返回。

## hs\_terminate\_playback

说明

终止一个回放会话的定义和实际回放。

语法

```
hs_terminate_playback
```

用法

- 此命令只有在成功执行 `hs_create_playback_session` 命令后才有效。
- 成功发出 `hs_create_playback_session` 命令之后，必须先发出 `hs_terminate_playback`，然后才能开始定义任何其它回放会话或记录会话。
- 与记录会话定义不同的是，回放会话定义不存储在 Historical Server 控制文件中。

## hs\_terminate\_recording

说明	终止记录会话的定义，取消记录会话的预定启动，或终止正在进行的记录会话。
语法	<code>hs_terminate_recording session_id [,delete_option]</code>
参数	<p><i>session_id</i></p> <p>指定要终止的会话。在使用 <code>hs_create_recording_session</code> 定义一个会话时，Historical Server 为该会话分配一个会话 ID。使用 <code>hs_list</code> 命令可以查看会话 ID。</p> <p><i>delete_option</i></p> <p>指定正在记录时是否删除与记录会话关联的文件。</p> <ul style="list-style-type: none"><li>• <code>delete</code> — 删除文件。</li><li>• <code>retain</code>（缺省值）— 不删除文件。如果要取消的是未启动的记录会话，将忽略该参数。</li></ul>
示例	此示例终止标识符为 5 的活动记录会话，然后删除与会话 5 关联的所有文件。
用法	<pre>hs_terminate_recording 5,delete</pre> <ul style="list-style-type: none"><li>• 启动记录的用户即会话的所有者。</li><li>• 要终止一个记录会话，您必须是该会话的所有者或 Historical Server 超级用户。</li><li>• <code>hs_terminate_recording_session</code> 命令可以取消在当前连接中创建但从未启动的记录会话的定义。这样就允许使用当前连接定义其它记录会话或定义回放会话。</li><li>• 如果尝试终止一个不活动的记录会话，该命令将返回一个错误。</li></ul>

主题	页码
<a href="#">Historical Server 数据文件概述</a>	75
<a href="#">控制文件</a>	76
<a href="#">数据文件</a>	81
<a href="#">错误消息文件</a>	81
<a href="#">脚本文件</a>	82
<a href="#">批量复制示例</a>	84
<a href="#">Cut 实用程序示例</a>	86
<a href="#">允许 Historical Server 将监控数据输出到数据库</a>	87

## Historical Server 数据文件概述

本节的主题有：

- [对 Historical Server 文件的描述](#)
- [文件权限](#)
- [一般文件格式](#)

### 对 Historical Server 文件的描述

Historical Server 文件有：

- **控制文件** — 同一 Historical Server 的所有记录会话都使用相同的控制文件。有关多个 Historical Server 实例和控制文件的信息，请参见第 20 页的“配置多个 Historical Server 实例”。
- **错误消息文件** — 每个记录会话都有它自己的错误消息文件。如果没有会话的错误记录，则该文件是空的。
- **数据文件** — 记录会话中的每个视图都由一个数据文件来表示。数据文件保留在会话期间为视图收集的所有样本。

- **脚本文件** — 当您使用 `hs_create_recording_session` 命令定义会话记录时，`script_type` 参数可指定创建一个 SQL 脚本文件。脚本文件在记录会话中为每一个视图定义一个 Adaptive Server 表。

您可以使用批量复制 (bcp) 实用程序来填充现有的 Adaptive Server 表。如果您是 UNIX 用户，且第一次调用 `cut` 实用程序从文件中删除所需的列，则可以将字段的子集放入单独的 Adaptive Server 表中。

## 文件权限

给定记录会话的所有历史监控数据文件都存储于同一目录中。缺省情况下，此目录是由 Historical Server 启动命令的 `-D` 参数指定的。创建会话时，可以为会话替换此缺省值。

启动 Historical Server 的用户是 Historical Server 创建的文件的所有者。所有者拥有对历史监控文件的读写权限。如果包含在一个历史数据文件或错误消息文件中的数据是公用而不是专用的，则所有的用户和组都具有读权限。

## 一般文件格式

所有文件都用换行符作为记录结尾标记。每个文件中的字段长度可变并且由逗号分隔。所有数据都以 ASCII 格式存储，这与 Sybase 批量复制 (bcp) 实用程序兼容。

## 控制文件

`hs.ctl` 控制文件位于 Historical Server 主目录中。控制文件维护关于为 Historical Server 所知的所有记录会话（过去和现在的）的信息。有关 Historical Server 如何使用控制文件的详细信息，请参见第 7 页的“[Historical Server 控制文件和主目录](#)”。

使用 Historical Server `hs_list` 命令访问 Historical Server 控制文件中的信息。不要编辑控制文件。您可能不小心损坏文件。

不管您用什么编辑器，都不要打开并保存此文件。特别是当 Historical Server 是在 Windows 上运行时更是如此。与 Historical Server 创建的其它文件不同，控制文件不是一个标准格式的 Windows 文本文件。控制文件的文本行只用换行符结束，而不是通常的回车符 / 换行符对。编辑程序可能会将不需要的回车符 / 换行符对嵌入到文本中，从而损坏文件。

控制文件包含下列类型的记录：

- 标头记录
- 会话控制记录
- 数据项控制记录
- 报警控制记录
- 视图控制记录
- 过滤器控制记录

## 标头记录

控制文件中的第一个记录是包含下列六个整数字段的标头记录：

- “更新进行中”标志
- 被更新的信息在文件中的字节偏移量
- 要由 Historical Server 分配给下一个记录会话的唯一会话 ID
- 用于文件损坏检测和恢复的唯一会话 ID 的副本
- 其控制信息最近被写入控制文件的会话 ID
- 控制文件中的会话条目号不再有效，因为它们引用的一个或多个数据文件或错误消息文件已被删除

在此标头记录之后有许多不同类型的记录，描述 Historical Server 所知的过去和现在的所有记录会话。这些记录按层次顺序出现在控制文件中，其中记录类型的层次自上而下的顺序是：

- 会话控制
- 视图控制
- 数据项控制
- 报警控制和过滤器控制

例如，由两个视图组成的会话（每个视图有两个数据项，每个数据项分别有一个报警和一个过滤器）将由下面的控制记录层次表示，各个控制记录的存储顺序如下：

```
session 1
  view 1
    data item 1
      alarm 1
      filter 1
    data item 2
      alarm 2
      filter 2
  view 2
    data item 3
      alarm 3
      filter 3
    data item 4
      alarm 4
      filter 4
```

各种控制记录类型的内容在下一节中进行说明。

## 会话控制记录

会话控制记录包括：

- 记录标识符（单词“session”）。
- 会话状态。状态可以是活动的、不活动的或无效的。如果属于会话的任何数据文件或错误消息文件被删除，或者会话在到达开始时间前被终止，则无效状态适用。
- 在控制文件中，从该记录开始到下一个会话记录的字节偏移。
- 会话 ID：在此控制文件的所有会话中唯一标识某个会话的号码。会话 ID 是按递增顺序分配的整数。
- 请求进行记录的客户端的用户名。
- 记录会话的开始日期和时间。

控制文件开始和结束时间尽可能接近地反映记录会话实际开始和结束的时间。这些实际时间可能与命令中指定的参数值 *start\_time* 和 *end\_time* 稍有出入。例如，*start\_time* 可能预定为下午 1 点，但是由于系统忙碌，会话可能实际开始于下午 1:02。

- 记录会话的结束日期和时间。

- 监控的 Adaptive Server 的名称。
- 参与记录会话的 Monitor Server 的名称。
- 存储会话的监控文件的目录路径名。
- 会话错误消息文件名。
- 记录时使用的采样间隔（以秒计）。

如果该值为 0，这意味着会话由 *summarization\_level* 为 *entire* 的回放创建。（整个会话由一个样本表示，并且没有采样间隔。）

- 有效保护级别 (*public/private*)。
- 错误选项（如果在记录过程中出现非致命错误，*continue/halt*）。
- 创建的脚本文件类型 (*sybase\_script/no\_script*)。
- 创建记录会话的客户端加密口令。
- 如果创建会话是回放的结果，则关键字为 *summary*。

在文件中每个会话记录有一个会话控制记录。其后立即跟有会话的第一个视图控制记录。

## 视图控制记录

视图控制记录包括：

- 记录标识符（单词“*view*”）
- 视图名
- 视图数据文件的名称

在记录会话中定义的每个视图都有一个视图控制记录。在视图中标识数据项的一个或多个数据项控制记录跟在每个控制记录之后。报警控制和过滤器控制记录相应地散布在数据项控制记录之间。

## 数据项控制记录

数据项控制记录包括：

- 记录标识符（单词“*dataitem*”）。
- 数据项名称。
- 统计类型。

- 在从回放创建的会话中，关键字 **estimated** 表示这样的数据项：它的值是根据其它会话中的数据项估计得出的。

在当前视图的定义中出现的每个数据项都有一个数据项控制记录。如果报警和 / 或过滤器应用于数据项，则它们的控制记录跟在该数据项的控制记录之后。该数据项的所有报警控制记录都在同一数据项的过滤器控制记录之前。

## 报警控制记录

报警控制记录包括：

- 记录标识符（单词 “alarm”）
- 报警操作
- 报警操作数据（文件名，其后可能跟有一列参数）
- 报警数据类型
- 报警值

为当前数据项定义的每个报警都有一个报警控制记录。数据项所有报警控制记录在第一个数据项过滤器控制记录之前被分组。

## 过滤器控制记录

过滤器控制记录包括：

- 记录标识符（单词 “filter”）
- 过滤器类型
- 过滤器数据类型
- 值的说明（以逗号分隔的一个或多个字段，与 `hs_create_filter` 命令使用的格式相同）

应用于当前数据项的每个过滤器都有一个过滤器控制记录。



## 数据文件

数据文件包含在会话过程中记录的单个视图数据。文件名是 *dsessionid\_viewnumber\_viewname*，其中：

- *sessionid* 是会话的唯一标识符。
- *viewnumber* 是一个数字，该数字大于或等于反映为会话定义视图的顺序的数字。
- *viewname* 是用户提供给视图的名称，如有必要，会截断该名称以符合当前平台的文件命名约定。在 Windows 平台上，文件名限制在八个字符以内。

出现在每个采样中的每个数据项键字段值唯一组合在此文件中都有一条对应的记录。

每个这样的记录保留：

- 采样的日期和时间  
此时间戳基于在被监控的 Adaptive Server 上有效的当地时间
- 被采样的所有数据项的值

如果在给定的采样间隔内没有为此视图返回任何数据，将在历史数据文件中写入占位符记录。此记录包含样本的时间戳，其后跟有一系列用逗号分隔的空值。也就是说，时间戳字段后跟有所需数量的逗号，这些逗号用来分隔视图的剩余数据项，但是没有为这些数据项存储值。如果数据项被复制到的列接受空值，则批量复制实用程序可接受此格式。

## 错误消息文件

在记录会话的样本时收到的任何错误消息都存储在单独的错误消息文件中。此文件的名称是 *eSessionId*，其中：

- *e* 是一个常量。
- *SessionId* 是会话的唯一标识符。

此文件中的每个记录都包含下列字段：

- 样本出现错误消息的日期和时间。此时间戳基于在被监控的 Adaptive Server 上有效的当地时间。如果错误阻止 Historical Server 检索样本时间戳，则时间戳字段可能为空。
- 错误号

- 错误严重性
- 错误状态
- 错误源：
  - 1 — 未知
  - 2 — Historical Server
  - 3 — CT-Library
  - 4 — Adaptive Server
  - 5 — Monitor Server
- 错误消息文本

有关错误消息字段的详细信息，请参见《参考手册》。

## 脚本文件

如果 `hs_create_recording_session` 命令的 `script_type` 参数的值是 `sybase_script`，则为会话创建脚本文件。脚本文件名为 `ssessionId`，其中 `sessionId` 是会话的唯一标识符。

## 脚本文件表名

脚本文件包含创建表命令的文本，每个为会话定义的视图都对应一个命令。每个表名为 `sSessionId_viewnumber_viewname`，其中：

- `s` 是一个常量。
- `SessionId` 是会话的唯一标识符。
- `viewnumber` 是一个数字，该数字大于或等于反映为会话定义视图的顺序的数字。
- `viewname` 是用户提供给视图的名称，为了遵守 Adaptive Server 标识符长度的限制，可根据需要将该名称相应地截断。

## 脚本文件表列名

为每个表定义的第一列包含样本时间戳字段。它始终被命名为“Timestamp”并且其数据类型为 `datetime`。

剩余的表列的名称是从它们代表的数据库项派生的。每个列名为 `dataitemname_stattype`，其中：

- `dataitemname` 是数据库项名的压缩形式。
- `stattype` 是数据库项统计类型的缩写形式。

会为每一列分配一个适用于该数据库项表示的信息类型的数据类型。除了样本时间戳列以外，所有的列都可以包含空值。

## 传递脚本文件命令

通过使用 `isql` 实用程序的 `:r` 命令，可以将脚本文件中的创建表命令传递到 Adaptive Server。在通过 `:r` 命令定义表后，您可以使用批量复制实用程序将视图数据文件导入表中。有关批量复制示例，请参见第 84 页的“批量复制示例”。

## 脚本应用示例

使用下面的 `isql` 命令可连接到要存储记录的监控数据的 Adaptive Server:

```
isql -Username -Ppassword -Sserver
```

其中，`username` 是该 Adaptive Server 中的登录帐户的名称，`password` 是该登录帐户的口令，`server` 是 Adaptive Server 的名称。

执行下列 `isql` 命令：

```
use database  
go
```

其中，`database` 是将要包含新表的数据库的名称。

```
:r scriptfile  
go
```

其中，`scriptfile` 是 Historical Server 创建的脚本文件的路径名。

在 Adaptive Server 中，为记录会话定义的每个视图现在都有一个表。

## 批量复制示例

本节描述如何将历史数据文件传递到批量复制 (bcp) 实用程序来填充 Adaptive Server 表。

若要使用 bcp 存储 Adaptive Server 表中记录的数据，对于为会话定义的每个视图，必须首先为其创建一个表。在表中，为构成视图的每个数据项创建单独的列。每个表的第一列用于存储样本的日期和时间，必须是 `datetime` 数据类型。

在 Historical Server 记录时，在采样间隔内，如果没有为某一视图返回数据，则将一条占位符记录写入该视图的数据文件中。此记录包含样本的时间戳，其后跟有逗号分隔的空值列表。因此，建议将除第一列以外的所有列都定义为允许空值。

可以通过让 Historical Server 为您创建脚本文件来简化表的创建过程。使用 `hs_create_recording_session` 命令的 `script_type` 参数。

在 Adaptive Server 上创建表之后，便可以使用 bcp 实用程序将数据从视图数据文件中导入到它们相应的表中。对于要将其数据导入 Adaptive Server 表中的每个数据文件，运行一次以下命令：

```
bcp [[database_name.]owner.]table_name in
view_data_file -c [ -e errfile] [-U username]
[-P password] [-S sqlserver] -t,
```

其中：

- `database_name` 是保存导入数据的表所在的数据库的名称。
- `owner` 是该表的所有者的名称（大概是包含该表的数据库中您自己的用户名）。
- `table_name` 是要将来自视图数据文件的数据导入到的数据库表的名称。
- `view_data_file` 是视图数据文件的名称。
- `errfile` 是错误文件名，不能传送到数据库表的任何行都将存储在该错误文件中。
- `username` 是 Adaptive Server 的登录名。
- `password` 是 `username` 的口令。
- `sqlserver` 是 Adaptive Server 的名称。
- `-t` 指定列终结符为逗号。

使用 `-c` 选项时，缺省的行终结符是数据文件使用的 `\n`（换行）字符。因此，您不必指定 `-r` 选项。如果您指定记录终结符选项 (`-r`)，请务必再使用一个反斜杠 (`\`) 字符，以避免一个反斜杠字符所产生的特殊含义，例如，`-r\\n`。

有关 `bcp` 的更多信息，请参见《实用程序指南》。

## 示例

假设您要每秒记录一次由 SMS1100 Monitor Server 监控的 Adaptive Server 上每个设备发生的读取数、写入数和 I/O 总数。可以按照下列步骤来记录数据，创建表来接受数据，然后使用 `bcp` 实用程序来将数据导入到表中。在本例中，由 Historical Server 为视图创建的视图数据文件是 `dl_1_device_io`。

- 1 使用 `isql` 登录到 Historical Server。
- 2 用下列命令记录数据：

```
hs_create_recording_session SMS1100, 1
go
hs_create_view device_io,
"device name", "value for sample",
"device reads", "value for sample",
"device writes", "value for sample",
"device i/o", "value for sample"
go
hs_initiate_recording
go
```

- 3 使用 `isql` 登录到 Adaptive Server。因为记录会话只包含一个视图（“`device_io`”），所以通过使用下列 Transact-SQL 命令在 `pubs2` 数据库中仅创建一个表，用以为该视图存储数据：

```
use pubs2
go
create table device_io
(
sample_time          datetime          not null,
device_name          varchar (255)      null,
device_reads_val     int                null,
device_writes_val    int                null,
device_io_val        int                null,
)
go
```

- 4 在记录会话完成后，可以用 `bcp` 将数据从视图的数据文件导入到数据库表。使用下列命令：

```
bcp pubs2..device_io in d1_1_device_io -c
-e d1_1_device_io.err -Uusername
-Ppassword -Sserver -t
```

## Cut 实用程序示例

如果您使用的是 UNIX 计算机，并且是第一次调用 `cut` 实用程序从文件中删去并保存所需列，则可以将数据文件中字段的子集放入单独的 Adaptive Server 表中。请使用下面的命令：

```
cut -flist -d, view_data_file > output_file_name
```

其中：

- `list` 是您要从初始视图数据文件中剪切并保存的字段编号的逗号分隔列表。第一个字段的字段编号为 1。
- `view_data_file` 是文件名，所有记录数据都存储在该文件中。
- `output_file_name` 是要存储删去的数据的文件的名称。

使用 `cut` 之后，可以使用 `bcp` 实用程序将 `output_file_name` 中的数据导入到 Adaptive Server 表中。Historical Server 可能将类似于以下内容的行返回到 `view_data_file`：

```
1995/3/25 10:32:39,master,0,0,0
1995/3/25 10:32:39,sysprocsdev,0,0,0
1995/3/25 10:32:39,ANewDBDevice,0,0,0
1995/3/25 10:32:39,ATestDevice,0,0,0
1995/3/25 10:32:40,master,0,0,0
1995/3/25 10:32:40,sysprocsdev,0,0,0
1995/3/25 10:32:40,ANewDBDevic,0,0,0
1995/3/25 10:32:40,ATestDevice,0,0,0
```

假设您仅对每个设备在采样期间的 I/O 总数感兴趣。可以通过执行以下命令来剪切出您想要的数：

```
cut -f1,2,5 -d, d1_1_device_io > d1_1_device.io.new
```

其中字段 1 是采样的日期和时间，2 是第二个字段（即设备名），5 是第五个字段（即设备 I/O 值）。

现在您可以像前一节第 84 页的“批量复制示例”中所描述的那样，将 `d1_1_device.new` 文件导入到 Adaptive Server 表中。

有关 `cut` 实用程序的更多信息，请参见您系统上的 UNIX 参考页。

## 允许 Historical Server 将监控数据输出到数据库

除了将 Historical Server 输出保存到文件外，您还可以将 Historical Server 输出直接发送到数据库。将数据保存到数据库更易于对数据进行分析。

如果数据保存在数据库中，则每个 Historical Server 视图的输出存储在一个单独的表中。对 Historical Server 视图定义的更改会自动反映在表定义中。

### 设置接收 Adaptive Server

在允许 Historical Server 向某个 Adaptive Server 发送其监控数据之前，您必须选择该 Adaptive Server。选择接收 Historical Server 监控数据的 Adaptive Server 之后，您必须：

- 1 为 Historical Server 创建一个数据库，用来存储监控数据。该数据库的缺省名称为 `hs_monitoring`。如果您选择为该数据库指定其它名称，可创建该指定数据库，然后在 `hs_directload.sql` 脚本中更改该数据库的名称。
- 2 对该数据库运行安装脚本 `hs_directload.sql`。安装脚本会创建 `sessions` 和 `views` 这两个目录表以及存储过程 `sp_hs_dboutput`。

### 启动 Historical Server

要允许 Historical Server 向 Adaptive Server 数据库发送其监控数据，您必须指定以下项：

- 来自 Historical Server 的监控数据的目标位置，该目标位置应为特定 Adaptive Server 和数据库，而不是一个平面文件。
- 连接到输出服务器的用户名和口令（如果它们与 Historical Server 连接到监控服务器使用的用户名和口令不同）。如果没有指定用户名和口令，则它们缺省为命令行的 `-U` 和 `-P` 参数中指定的用户名和口令。

Historical Server 启动时，历史数据的目标 Adaptive Server 和数据库必须可用。

命令行语法为：

```
histserver -U<user name>  
- P<password> -D<output dir>  
-l<log file> -l<interfaces file>  
[-d<delimiter>] [-O<ASE name>] [-o<DATABASE name>] [-f ]  
-u<outputASE user name> -p<outputASEpassword>
```

- -O *ASE name* – 目标 Adaptive Server 的名称。
- -o *DATABASE name* – 将监控数据发送到的数据库的名称，如果该数据库不是 `hs_monitoring`，请在此选项中指定该数据库的名称。
- -u *outputASE user name* – 用以连接到保存输出内容的 Adaptive Server 的登录名。
- -p *outputASEpassword* – 输出 Adaptive Server 登录名的口令。
- -f – 在指定 -O 选项时，如果您希望 Historical Server 将数据发送到数据库和输出目录中的文件，则必须使用该参数。

您必须能够访问 Historical Server 并拥有对目标数据库的更新权限。

在下面的示例中，指定了 -O 和 -o 参数，并且输出数据库名为 `hs_monitoring`：

```
$SYBASE/ASE-12_5/bin/histserver -D$SYBASE -Sajax_hs  
-l$SYBASE/ajax_hs.log -Usa -Pnorthstar -Oajax  
-ohs_monitoring
```

## 查看数据

### 数据存储

来自 Historical Server 的数据的格式与在该服务器上创建相应视图时定义的格式相同。为了存储数据，Historical Server 创建两个系统表：

- `Sessions` 表记录使用输出数据库的每个记录会话。
- `Views` 表列出每个记录会话使用的视图。
- 除了每个视图输出到数据库的一个表外，Historical Server 输出数据库中还创建了两个系统表。这些表是：
  - `sessions` 表记录使用输出数据库的每个记录会话。
  - `views` 表列出了每个记录会话使用的视图。



下面的示例显示 `sessions` 表可能包含的内容：

```
1> select * from sessions
2> go
```

DbSessionId	HsSessionId	HsName	HsAseName	StartDate	EndDate
0	0	ajax_hs	ajax	Jul 22 2005 11:26AM	Jul 22 2005 11:45AM
1	14	ajax_hs	ajax	Jul 23 2005 12:25PM	Jul 23 2005 6:10AM
2	15	ajax_hs	ajax	Jul 24 2005 11:41AM	Jul 24 2005 11:59AM
3	16	ajax_hs	ajax	Jul 25 2005 11:56AM	Jul 25 2005 12:09PM
4	19	ajax_hs	ajax	Jul 26 2005 10:59AM	Jul 27 2005 11:25AM
5	23	ajax_hs	ajax	Jul 27 2005 11:26AM	Jul 27 2005 11:36AM

下面的示例显示了 `views` 表可能包含的内容：

```
1> select * from views
2> go
```

DbSessionId	ViewName
1	stored_procs
1	connections
2	stored_procs_shutdown_test
2	connections
3	stored_procs_shutdown_test
3	connections
4	connections
5	stored_procedure_activity
5	connections
5	process_activity

视图数据输出表的结构类似于输出数据文件的结构，其中：

- 第一列是监控会话 ID。
- 第二列是监控的服务器名称。
- 第三列是日期 / 时间戳。
- 每个数据项的后续列在视图定义中指定。

对于日期 / 时间戳和数据项来说，这一结构与 `Historical Server` 提供的 DDL 脚本（这些脚本在 `Historical Server` 数据文件被批量复制到其它 `Adaptive Server` 时提供）中定义的结构相同。

**注释** viewname 是视图的名称，如有必要，会截断该名称以符合当前平台的文件命名约定。在 Windows 平台上，文件名限于 30 个字符。

---

在下面的示例中，查询报告 2005 年 7 月 27 日处于活动状态的会话和这些会话使用的视图：

```
1> select HsAseName, HsName, StartDate, EndDate, ViewName
2> from sessions, views
3> where sessions.DbSessionId = views.DbSessionId
4> and StartDate >= 'July 27, 2005'
5> and EndDate < 'July 28, 2005'
6> go
```

HsAseName	HsName	StartDate	EndDate	ViewName
tribble	tribble_hs	Jul 27 2005 10:59AM	Jul 27 2005 11:00AM	connections
tribble	tribble_hs	Jul 27 2005 11:00AM	Jul 27 2005 11:05AM	sproc_activity
tribble	tribble_hs	Jul 27 2005 11:00AM	Jul 27 2005 11:05AM	connections
tribble	tribble_hs	Jul 27 2005 11:17AM	Jul 27 2005 11:24AM	sproc_activity
tribble	tribble_hs	Jul 27 2005 11:17AM	Jul 27 2005 11:24AM	connections
tribble	tribble_hs	Jul 27 2005 11:25AM	Jul 27 2005 11:25AM	sproc_activity
tribble	tribble_hs	Jul 27 2005 11:25AM	Jul 27 2005 11:25AM	connections
tribble	tribble_hs	Jul 27 2005 11:26AM	Jul 27 2005 11:36AM	sproc_activity
tribble	tribble_hs	Jul 27 2005 11:26AM	Jul 27 2005 11:36AM	connections
tribble	tribble_hs	Jul 27 2005 11:26AM	Jul 27 2005 11:36AM	process_activity

下面是 Historical Server 视图的示例：

```
hs_create_view process_activity,
"Login Name", "Value for Sample",
"Process ID", "Value for Sample",
"Kernel Process ID", "Value for Sample",
"Connect Time", "Value for Session",
"Page I/O", "Value for Session",
"CPU Time", "Value for Session",
"Current Process State", "Value for Sample"
```

下面的查询显示由上述 `process_activity` 视图生成的数据库表的内容。

```
1> select * from process_activity
3> go
```

DbId	CPUTime	ProcessID	KernelProcessID	ConnectTime	PageIO	ProcessState	LogName
5	0.128447	17	3211313	61	28531	6	jsmith
5	0.128447	17	3211313	121	28531	6	jsmith
5	0.032169	19	3276850	60	32072	6	byoung
5	0.128447	17	3211313	182	28531	6	jsmith
5	0.099188	18	3342387	61	28759	6	dcharles
5	0.032169	19	3276850	121	33283	6	byoung
5	0.128447	17	3211313	242	28531	6	jsmith
5	0.105076	18	3342387	121	24373	6	dcharles
5	0.032169	19	3276850	181	33283	6	byoung
5	0.128447	17	3211313	303	28531	6	jsmith

## 删除 Historical Server 会话

当您使用 `hs_delete_data` 命令删除记录会话时，缺省行为是只删除与该会话对应的输出文件而完整地保留输出表中的条目。

如果您要删除输出表中的会话数据，应输入：

```
hs_deletedata database_also, sessionId
```

此命令会删除 `Session` 和 `ViewPerSession` 表中的所有对应条目。

## 报告错误

`Historical Server` 在监控数据输出保存过程中报告错误的方式与输出到平面文件时报告错误的方式相同。您会得到通过会话错误文件和 `HS` 日志文件遇到错误的通知。与数据库输出模式相关的错误可能包括：

- 如果目标 `Adaptive Server` 不存在，`Historical Server` 将报告失败。
- 如果命令行上指定的目标数据库不存在，`Historical Server` 将报告失败。
- 如果由于命令行上指定的数据库用尽了磁盘空间而导致 `Historical Server` 不能写入该数据库，`Historical Server` 将报告失败。
- 如果 `Historical Server` 必须向现有表添加列才能将该表用于其监控输出，它将记录一条警告消息。



# 数据项

## 有关数据项及其定义的表

表 A-1 描述了可包含在记录会话视图中的数据项。该表按字母顺序列出各数据项并提供有关每项的以下信息：

- 定义
- Adaptive Server 版本依赖性指定
- 结果或键指定

**表 A-1: 数据项和定义**

数据项	说明
应用程序执行类 版本：11.5 和更高版本	<p>一个给定的应用程序名的已配置执行类（如果存在的话）。由于覆盖特性，配置的执行类不一定反映运行时的优先级和引擎组。在监控器中将使用以下表示法：</p> <ul style="list-style-type: none"> <li>• 空白 — 没有为此应用程序配置执行类。</li> <li>• 执行类名 — 是按一般情况为应用程序而配置的执行类，没有考虑特定登录名。（也就是说，给应用程序配置的执行类的作用域为空。）</li> <li>• 执行类名后跟星号 (*) — 除了按一般情况为应用程序配置的执行类，还为特定登录名配置了附加的执行类。（也就是说，为应用程序配置的执行类的作用域为空，另外至少还为应用程序配置了一个具有特定作用域的附加执行类。）</li> <li>• 星号 (*) 而无执行类名 — 没有按一般情况给应用程序配置执行类，但为使用该应用程序的特定登录名配置了执行类。（也就是说，至少为应用程序配置了一个具有特定作用域的执行类。）</li> </ul> <p>类型：结果</p>
应用程序名 版本：11.0 和更高版本	<p>正在为其累积其它统计信息的应用程序的名称。包含应用程序名的视图仅报告有关采样周期结束后处于活动状态的进程的信息。</p> <p>在视图中，应用程序名与进程 ID 互斥。</p> <p>类型：键</p>
阻塞进程 ID 版本：11.0 和更高版本	<p>拥有某个锁的进程 ID，另一个进程（如果存在的话）正在等待解开此锁。返回值为 0 表示其它进程未被阻塞。</p> <p>类型：结果</p>

数据项	说明
高速缓存效率 版本: 11.0 和更高版本	一个特定数据高速缓存的每兆字节的每秒高速缓存命中次数。 类型: 结果
高速缓存命中百分比 版本: 11.0 和更高版本	按照下列公式计算出特定数据高速缓存, 对绑定到该数据高速缓存中的对象的页读取数的百分数: $cache\_hits / (cache\_hits + cache\_misses) * 100$ 类型: 结果
高速缓存命中数 版本: 11.0 和更高版本	读取某个特定数据高速缓存的页读取次数。 类型: 结果
高速缓存 ID 版本: 11.0 和更高版本	Adaptive Server 版本 11.0 或更高版本中数据高速缓存的 ID。特定的数据库表和索引可以绑定到一个特定数据高速缓存上, 或者一个数据库中的所有对象都可以绑定到同一个数据高速缓存上。任何对象都不能绑定到多个数据高速缓存上。 类型: 键
高速缓存未命中数 版本: 11.0 和更高版本	从磁盘而不是从某个特定数据高速缓存中读取的页读取次数。此数据项包括在页分配期间查找数据高速缓存中的页失败的次数。因此, 所报告的物理页读取数可能比实际值大。在此情况下, 由高速缓存命中百分比报告的数据高速缓存未命中数就比实际值小。 类型: 结果
高速缓存名 版本: 11.0 和更高版本	数据高速缓存的名称。特定的数据库表和索引可以绑定到一个特定数据高速缓存上, 或者一个数据库中的所有对象都可以绑定到同一个数据高速缓存上。任何对象都不能绑定到多个高速缓存上。 类型: 键
高速缓存预取效率 版本: 11.0 和更高版本	大 I/O 缓冲区中重用的页数占 Adaptive Server 曾引用过的总页数的百分比。当重用某缓冲区时, 会认为其中的所有页都被重用了。当缓冲池中没有空闲的缓冲区来从数据库设备接受新的物理读取时, 缓冲区被重用。由 Adaptive Server 引用的重用页数, 再除以每个缓冲区的页数与重用缓冲区数目相乘后的值, 所得结果表示大 I/O 预取的效率。 不管在命名数据高速缓存中配置了多少个缓冲池, Adaptive Server 只使用其中两个。它使用 2K 的缓冲池和配置有最大缓冲区的缓冲池。预取的效率对 2K 缓冲池没有意义, 因为 2K 的读取不算大 I/O。因此, 预取的效率是针对高速缓存中最大的缓冲池而言的。例如, 若数据高速缓存中缓冲池的缓冲区大小分别为 2K、8K 和 16K, 则不使用 8K 的缓冲池, 此值仅反映 16K 缓冲区的池中大 I/O 的效率。 如果该比值较大, 则预取是有效的, 否则预取就没有什么优点。这可能表示应该删除某个缓冲池 (或者意味着某表的聚簇索引是零散的, 因此应删除相应的索引并重新创建)。 类型: 结果

数据项	说明
高速缓存引用和重用 版本: 11.0 和更高版本	缓冲区中既被引用又被重用的页数。当确定预取缓冲区的效率时, 将使用该计数 (参见“高速缓存预取效率”)。与“高速缓存预取效率”不同的是, 该数据项包括了缺省的 2K 缓冲池中的活动。 关于缓冲区重用的定义, 请参见“高速缓存预取效率”。 类型: 结果
高速缓存重用 版本: 11.0 和更高版本	缓冲区中重用的页数。该值大则表示 (此内存池中缓冲区的) 周转率高, 并说明缓冲池可能太小了。零值则表示此内存池可能太大了。与“高速缓存预取效率”不同的是, 该数据项包括了缺省的 2K 缓冲池中的活动。 关于缓冲区重用的定义, 请参见“高速缓存预取效率”。 类型: 结果
高速缓存重新使用脏页 版本: 11.0 和更高版本	重用的缓冲区发生需要写入的更改的次数。非零值表示清洗空间太小。 关于缓冲区重用的定义, 请参见“高速缓存预取效率”。 类型: 结果
高速缓存大小 版本: 11.0 和更高版本	数据高速缓存的大小 (单位为兆字节)。 类型: 结果
高速缓存螺旋锁争用 版本: 11.0 和更高版本	对一个被强迫等待的数据高速缓存螺旋锁的请求比率。 <i>spinlock_waits / spinlock_requests</i> 类型: 结果
代码内存大小 版本: 11.0 和更高版本	为 Adaptive Server 分配的内存字节数。 类型: 结果
连接时间 版本: 11.0 和更高版本	从启动进程以来经历的秒数, 或者是启动会话以来经历的秒数 (两者中的较小值)。 类型: 结果
CPU 忙时百分比 版本: 11.0 和更高版本	Adaptive Server CPU 忙的时间占总的服务器 CPU 时钟周期数的百分比。 类型: 结果
CPU 百分比 版本: 11.0 和更高版本	在具有进程 ID 的视图中使用时, 该数据项表示单个进程处于运行状态的时间占处于所有进程运行状态的时间的百分比。 在具有应用程序名的视图中使用时, 该数据项表示运行给定应用程序的一组进程处于运行状态的时间占所有进程处于运行状态的时间的百分比。 类型: 结果
CPU 时间 版本: 11.0 和更高版本	在不含任何键的视图中使用时, 该数据项表示服务器上的总 CPU 忙时间 (单位为秒)。在含有键的视图中使用中, 该数据项表示每个进程、应用程序或引擎所用的忙时间 (单位为秒)。进程 ID 和应用程序名是互斥的。 类型: 结果

数据项	说明
CPU 放弃 版本: 11.0 或更高版本	Adaptive Server 将控制权交给操作系统的次数。 类型: 结果
当前应用程序名 版本: 11.0 和更高版本	当前正在特定进程中执行的应用程序的名称。 类型: 结果
当前引擎 版本: 11.0 和更高版本	该进程最近在其上运行的 Adaptive Server 引擎的编号。 类型: 结果
当前执行类 版本: 11.5 和更高版本	进程当前运行时所使用的执行类的名称。 类型: 结果
当前进程状态 版本: 11.0 和更高版本	进程的当前状态。有关可能状态的定义, 请参见“进程状态”。 类型: 结果
当前语句批处理 ID 版本: 11.5 和更高版本	正在一个特定进程中执行的特定查询批处理的 ID。 类型: 结果
当前语句批处理文本 版本: 11.5 和更高版本	正为某个特定进程执行的特定查询批处理的文本。该文本可以仅仅是某个查询批处理的完整文本的一个初始子字符串。该字段中存储的文本数量由 Adaptive Server <code>max SQL text monitored</code> 配置参数确定。 类型: 结果
当前语句批处理文本字节偏移 版本: 11.5 和更高版本	是指在为特定进程执行的查询批处理或存储过程内, 相对于某一条语句的开始位置的字节偏移。如果下列条件都满足: 当前语句过程数据库 ID 等于 0, 并且当前语句过程 ID 等于 0, 那么该语句是此查询批处理中当前正在执行的 SQL 语句。否则, 该语句是在存储过程中由这两个 ID 唯一确定的当前正在执行的 SQL 语句。 类型: 结果
当前语句批处理文本是否启用 版本: 11.5 和更高版本	报告 Adaptive Server (版本 11.5 和更高版本) 是否保存当前正在执行的查询批处理的 SQL 文本, 如果是, 则报告保存的数量。 0 值 = 禁止保存 SQL 文本 值为 1 或以上 = 每个服务器进程可保存的批处理文本的最大字节数。 类型: 结果
当前语句环境 ID 版本: 11.5 和更高版本	该 ID 在为特定进程执行的特定查询批处理内唯一确定一个存储过程调用。 类型: 结果
当前语句 CPU 时间 版本: 11.5 和更高版本	在运行状态下, 当前正在执行的 SQL 语句所用的时间量 (单位为秒)。 类型: 结果



数据项	说明
当前语句经历时间 版本：11.5 和更高版本	当前正在执行的 SQL 语句所用的时间量（单位为秒）。 类型：结果
当前语句行号 版本：11.5 和更高版本	（在查询批处理或存储过程中）包括当前正在为特定进程执行的 SQL 语句的开头的行号。如果当前语句过程数据库 ID 等于 0，并且当前语句过程 ID 等于 0，则当前正在执行的 SQL 语句在查询批处理中。 否则，当前正在执行的 SQL 语句在由这两个 ID 唯一确定的存储过程中。 类型：结果
当前语句等待后被授予的锁数 版本：11.5 和更高版本	在等待后授予的当前正在执行的 SQL 语句的锁请求数。 类型：结果
当前语句被立即授予的锁数 版本：11.5 和更高版本	由当前正在执行的 SQL 语句请求的、应立即授予或不再需要的（因请求者已拥有足够的锁）锁数目。 类型：结果
当前语句未被授予的锁数 版本：11.5 和更高版本	当前正在执行的 SQL 语句的被拒绝的锁请求数。 类型：结果
当前语句逻辑读取数 版本：11.5 和更高版本	当前正在执行的 SQL 语句从设备读取或从高速缓存读取的数据页读取数。 类型：结果
当前语句号 版本：11.5 和更高版本	（在查询批处理或存储过程中）当前正在为特定进程执行的 SQL 语句的语句编号。如果当前语句过程数据库 ID 等于 0，并且当前语句过程 ID 等于 0，则当前正在执行的 SQL 语句在查询批处理中。 否则，当前正在执行的 SQL 语句在由这两个 ID 唯一确定的存储过程中。 0 值表示当前正在执行的 SQL 语句的部分结果数据。换言之，该 SQL 语句在监控开始前已开始执行。可以获得有关的性能信息，但数字只能反映监控开始以后时间段内的性能。 类型：结果
当前语句页 I/O 版本：11.5 和更高版本	由当前正在执行的 SQL 语句完成的组合逻辑页读取和页写入数目。 类型：结果
当前语句物理读取数 版本：11.5 和更高版本	由当前正在执行的 SQL 语句请求的、不能从数据高速缓存读取的数据页读取数。 类型：结果

数据项	说明
当前语句过程数据库 ID 版本: 11.5 和更高版本	包含当前正在为特定进程执行的 SQL 语句的存储过程（包括触发器，它是一种特殊类型的存储过程）的数据库 ID。如果当前正在执行的 SQL 语句未包含在存储过程内，则此 ID 为 0。 类型: 结果
当前语句过程数据库名 版本: 11.5 和更高版本	包含当前正在为特定进程执行的 SQL 语句的存储过程（包括触发器，它是一种特殊类型的存储过程）的数据库名。如果当前正在执行的 SQL 语句未包含在存储过程内，则此名称为 “**NoDatabase**”。 类型: 结果
当前语句过程 ID 版本: 11.5 和更高版本	包含当前正在为特定进程执行的 SQL 语句的存储过程（包括触发器，它是一种特殊类型的存储过程）的 ID。如果当前正在执行的 SQL 语句未包含在存储过程内，则此 ID 为 0。 类型: 结果
当前语句过程名 版本: 11.5 和更高版本	包含当前正在为特定进程执行的 SQL 语句的存储过程（包括触发器，它是一种特殊类型的存储过程）的名称。如果当前正在执行的 SQL 语句未包含在存储过程内，则此名称为 “**NoObject**”。 类型: 结果
当前语句过程所有者名 版本: 11.5 和更高版本	包含当前正在为特定进程执行的 SQL 语句的存储过程（包括触发器，它是一种特殊类型的存储过程）的所有者名称。如果当前正在执行的 SQL 语句未包含在存储过程内，则此名称为 “**NoOwner**”。 类型: 结果
当前语句过程文本 版本: 11.5 和更高版本	为特定进程执行的特定存储过程（包括触发器，它是一种特殊类型的存储过程）的文本。如果当前语句过程数据库 ID 为 0，并且当前语句过程 ID 等于 0，则当前没有执行存储过程，且该文本是以空值终止的空字符串 (“”)。 如果不能获得文本（因为该存储过程已被编译，且其文本已被丢弃，或者因为文本是以加密格式存储的），那么该文本是以空值终止的空字符串 (“”)。 类型: 结果
当前语句查询计划文本 版本: 11.5 和更高版本	为特定进程执行的特定查询的查询计划的文本。 如果不能获得该文本（因为 Adaptive Server 已从其查询计划目录中删除了此计划），那么该文本是以空值终止的空字符串 (“”)。 类型: 结果
当前语句开始时间 版本: 11.5 和更高版本	当目前执行的 SQL 语句开始执行时， Adaptive Server 所在时区的日期和时间。 如果此 SQL 语句在开始监控前已开始运行，则此结果是该语句首次遇到这种活动时的日期和时间。 类型: 结果

数据项	说明
当前语句文本字节偏移	是指在为特定进程执行的查询批处理或存储过程内，相对于某一条语句的开始位置的字节偏移。
版本：11.5 和更高版本	如果当前语句过程数据库 ID 等于 0，并且当前语句过程 ID 等于 0，那么该语句是此查询批处理中当前正在执行的 SQL 语句。否则，该语句是在存储过程中由这两个 ID 唯一确定的当前正在执行的 SQL 语句。 类型：结果
数据库 ID	一个数据库的唯一标识。
版本：11.0 和更高版本	类型：键
数据库名	数据库的名称。
版本：11.0 和更高版本	类型：结果
死锁计数	死锁的数目。
版本：11.0 和更高版本	类型：结果
请求锁	一个字符串（Y 或 N），表示某个锁是否已升级到请求锁状态。
版本：11.0 和更高版本	类型：结果
设备命中百分比	设备请求百分比计算方法为：设备命中数除以设备未命中数加设备命中数之和所得的商再乘以 100。
版本：11.0 和更高版本	类型：结果
设备命中数	获得设备访问授权的次数。
版本：11.0 和更高版本	类型：结果
设备 I/O	设备读取数和设备写入数的组合。
版本：11.0 和更高版本	类型：结果
设备未命中数	需等待获得设备访问授权的次数。
版本：11.0 和更高版本	类型：结果
设备名	Adaptive Server 中定义的数据库设备的名称。
版本：11.0 和更高版本	类型：键
设备读取数	从某个设备读取的次数。
版本：11.0 和更高版本	类型：结果
设备写入数	写入某个设备的次数。
版本：11.0 和更高版本	类型：结果

数据项	说明
经历的时间 版本: 11.0 和更高版本	以秒为单位的时间增量, 可能是从一次数据刷新到下一次 (采样) 的时间, 或者是从创建视图到当前 (会话) 的时间。 类型: 结果
引擎号 版本: 11.0 和更高版本	Adaptive Server 引擎的编号。 类型: 键
主机名 版本: 11.0 和更高版本	建立特定连接的主机计算机的名称。 类型: 结果
索引逻辑读取数 版本: 11.0 和更高版本	从高速缓存和从数据库设备读取的索引页读取数。 类型: 结果
索引物理读取数 版本: 11.0 和更高版本	不能从数据高速缓存读取的索引页读取数。 类型: 结果
内核进程 ID 版本: 11.0 和更高版本	一个 Adaptive Server 进程标识符, 它在长时间内保持唯一。 类型: 键
内核结构内存大小 版本: 11.0 和更高版本	为内核结构分配的内存字节数。 类型: 结果
拒绝的大 I/O 版本: 11.0 和更高版本	此值表示当一次从磁盘中获取多个连续页并载入此数据高速缓存的缓冲区中时, 缓冲区管理器不能满足 (优化器的) 此类请求的次数。 类型: 结果
执行的大 I/O 版本: 11.0 和更高版本	此值表示当一次从磁盘中获取多个连续页将数据载入到此数据高速缓存的缓冲区中时, 缓冲区管理器可以满足 (优化器的) 此类请求的次数。 类型: 结果
请求的大 I/O 版本: 11.0 和更高版本	此值表示当一次从磁盘中获取多个连续页将数据载入到此数据高速缓存的缓冲区中时, 优化器发出的 (高速缓存管理器的) 此类请求的次数。 类型: 结果
锁计数 版本: 11.0 和更高版本	锁的数目。 类型: 结果
锁命中百分比 版本: 11.0 和更高版本	锁的成功请求的百分比。 类型: 结果

数据项	说明
锁结果 版本：11.0 和更高版本	<p>逻辑锁请求的结果。锁结果值有：</p> <ul style="list-style-type: none"> <li>• 1 — 立即授予。</li> <li>• 2 — 不需要；请求者已拥有一个足够的锁。</li> <li>• 3 — 已等待；请求者已等待。</li> <li>• 4 — 不等待；不能立即获得锁，且请求者不想将该锁请求放入队列中。</li> <li>• 5 — 死锁；请求者已被选为死锁的牺牲品。</li> <li>• 6 — 中断；锁请求被关注情况所中断。</li> </ul> <p>类型：键</p>
汇总的锁结果 版本：11.0 和更高版本	<p>已授予或未授予级别上的锁结果摘要。</p> <ul style="list-style-type: none"> <li>• 1 — 授予的锁结果汇总由以下锁结果组成：已授予、不需要和已等待。</li> <li>• 2 — 未授予的锁结果汇总由以下锁结果组成：不等待、死锁和已中断。</li> </ul> <p>类型：键</p>
锁状态 版本：11.0 和更高版本	<p>锁的当前状态，包括如下锁状态值：</p> <ul style="list-style-type: none"> <li>• 1 — 已拥有和阻塞。</li> <li>• 2 — 已拥有和未阻塞。</li> <li>• 3 — 已请求和阻塞。</li> <li>• 4 — 已请求和未阻塞。</li> </ul> <p>类型：键</p>
锁状态计数 版本：11.0 和更高版本	<p>每个锁状态下的锁数目。这是一个快照值。</p> <p>类型：结果</p>
锁类型 版本：11.0 和更高版本	<p>Adaptive Server 通过锁定活动事务当前使用的表或数据页来保护它们。Adaptive Server 使用下列锁类型：</p> <ul style="list-style-type: none"> <li>• 1 — 排它表锁。</li> <li>• 2 — 共享表锁。</li> <li>• 3 — 排它意向锁。</li> <li>• 4 — 共享意向锁。</li> <li>• 5 — 排它页锁。</li> <li>• 6 — 共享页锁。</li> <li>• 7 — 更新页锁。</li> <li>• 8 — 排它行锁。</li> <li>• 9 — 共享行锁。</li> <li>• 10 — 更新行锁。</li> </ul> <p>类型：键</p>

数据项	说明
锁阻塞数 版本: 11.0 和更高版本	被拥有此类 “held_and_blocking” (持有和阻塞) 锁的这一进程所阻塞的锁数目。 类型: 结果
立即授予的锁 版本: 11.5 和更高版本	无需等待释放其它锁就可立即授予的锁数目。 类型: 结果
等待后授予的锁 版本: 11.5 和更高版本	需等待释放其它锁后才授予的锁数目。 类型: 结果
未授予的锁 版本: 11.5 和更高版本	已请求但未授予的锁数目。 类型: 结果
日志争用百分比 版本: 11.0 和更高版本	指当将用户日志高速缓存刷新到事务日志中时, 必须等待日志信号的次数占总次数的百分比。 此值过高可能表示应增加用户日志高速缓存的大小。 类型: 结果
逻辑页读取数 版本: 11.0 和更高版本	单位时间内的数据页读取数, 包括从高速缓存读取或从数据库设备读取。 类型: 结果
登录名 版本: 11.0 和更高版本	与 Adaptive Server 进程相关联的登录名。 类型: 结果
活动最频繁设备 I/O 版本: 11.0 和更高版本	在给定的时间段内, 最频繁地读取和写入设备的组合数目。 类型: 结果
活动最频繁设备名 版本: 11.0 和更高版本	在给定的时间段内, 具有最大的读取和写入组合数目的设备名称。 类型: 结果
接收的网络字节数 版本: 11.0 和更高版本	接收到的网络字节数。 类型: 结果
发送的网络字节数 版本: 11.0 和更高版本	发送的网络字节数。 类型: 结果
网络缺省包大小 版本: 11.0 和更高版本	缺省的网络包大小。 类型: 结果
网络 I/O 字节数 版本: 11.0 和更高版本	发送和接收的总网络字节数。 类型: 结果

数据项	说明
最大网络包大小	已配置的最大网络包大小。 类型：结果
版本：11.0 和更高版本	
接收的网络包大小	接收的网络包平均大小。 类型：结果
版本：11.0 和更高版本	
发送的网络包大小	发送的网络包平均大小。 类型：结果
版本：11.0 和更高版本	
接收的网络包数	接收的网络包数。 类型：结果
版本：11.0 和更高版本	
发送的网络包数	发送的网络包数。 类型：结果
版本：11.0 和更高版本	
引擎数	为 Adaptive Server 配置的引擎数。 类型：结果
版本：11.0 和更高版本	
进程数	当前正在 Adaptive Server 上运行的进程数；如果与键应用程序名一起使用，则是当前正在运行一个给定应用程序的进程数。
版本：11.0 和更高版本	类型：结果
对象 ID	数据库对象的 ID。返回的对象可以是一个数据库表、存储过程或临时表。 对象 ID 可以是负数。Adaptive Server 分配给临时表的对象 ID 可为正，也可为负。 类型：键
版本：11.0 和更高版本	
对象名	数据库对象的名称。对于临时表，报告结果为字符串 <b>**TempObject**</b> 。 类型：结果
版本：11.0 和更高版本	
对象类型	数据库对象的类型： • 0—无。 • 1—存储过程（包括触发器）。 • 2—表。 类型：结果
版本：11.0 和更高版本	
所有者名称	对象所有者的名称。 类型：结果
版本：11.0 和更高版本	
页高速缓存大小	为页高速缓存分配的内存字节数。 类型：结果
版本：11.0 和更高版本	

数据项	说明
页命中百分比 版本：11.0 和更高版本	能够从高速缓存读取（而不必进行物理页读取）的数据页读取次数百分比。 类型：结果
页 I/O 版本：11.0 和更高版本	总计组合逻辑页读取数和页写入数。 类型：结果
页号 版本：11.0 和更高版本	一个特定的锁或锁请求的数据页编号。 类型：键
页写入数 版本：11.0 和更高版本	写入到数据库设备中的页数。 类型：结果
物理页读取数 版本：11.0 和更高版本	不能从数据高速缓存读取的数据页读取数。 类型：结果
过程缓冲区大小 版本：11.0 和更高版本	为过程缓冲区分配的内存字节数。 类型：结果
过程 CPU 时间 版本：11.0 和更高版本	执行一个存储过程所耗用的 CPU 秒数。 类型：结果
过程数据库 ID 版本：11.0 和更高版本	活动存储过程的数据库 ID。 类型：键
过程数据库名 版本：11.0 和更高版本	活动存储过程的数据库名。 类型：键
过程占用时间 版本：11.0 和更高版本	执行一个存储过程期间所占用的秒数。对此数据项有效的所有统计类型都以秒为单位报告时间。例如，“过程占用时间”、“会话平均时间”报告执行每个过程时平均经历的秒数。 类型：结果
过程执行类 版本：11.5 和更高版本	一个给定存储过程的已配置执行类（如果存在的话）。 类型：结果
过程执行计数 版本：11.0 和更高版本	执行一个存储过程或执行其中某一行的次数。 类型：结果
过程头大小 版本：11.0 和更高版本	为过程头分配的内存字节数。 类型：结果



数据项	说明
过程命中百分比	一个过程执行时，在过程高速缓存中找到该过程的查询计划且可用的次数百分比。
版本：11.0 和更高版本	类型：结果
过程 ID	活动存储过程。“活动”表示调用了最顶层的存储过程。
版本：11.0 和更高版本	类型：键
过程行号	存储过程行号。
版本：11.0 和更高版本	类型：键
过程行文本	存储过程的全部文本。
版本：11.0 和更高版本	类型：结果
过程逻辑读取数	执行一个存储过程的请求数目，包括从过程高速缓存或从 <i>sysprocedures</i> 读取。
版本：11.0 和更高版本	类型：结果
过程名称	活动存储过程的名称。
版本：11.0 和更高版本	类型：结果。
过程所有者名	活动存储过程所有者的名称。
版本：11.0 和更高版本	类型：结果
过程物理读取数	执行一个存储过程的请求数目，并且必须是从 <i>sysprocedures</i> 请求的读取。
版本：11.0 和更高版本	类型：结果
过程语句号	在一个存储过程内的语句编号。单个存储过程行可以包含一个或多个语句。
版本：11.0 和更高版本	类型：键
进程 ID	Adaptive Server 进程标识号。包含进程 ID 的视图仅报告采样周期结束后活动的进程。在一个视图中，进程 ID 与应用程序名是互斥的。
版本：11.0 和更高版本	类型：键

数据项	说明
进程状态 版本: 11.0 和更高版本	<p>进程状态:</p> <ul style="list-style-type: none"> <li>• 0 — 无。</li> <li>• 1 — 报警休眠。等待报警。</li> <li>• 2 — 后台。Adaptive Server 进程执行。</li> <li>• 3 — 无效状态。未确定的错误状况。</li> <li>• 4 — 感染。由 Adaptive Server 标记为不可处理。</li> <li>• 5 — 锁休眠。等待获取锁。</li> <li>• 6 — 接收休眠。等待网络读取。</li> <li>• 7 — 可运行。按照优先级和 CPU 可用性等待运行。</li> <li>• 8 — 运行中。正在执行。</li> <li>• 9 — 发送休眠。等待网络发送。</li> <li>• 10 — 休眠。由于这里未列出的任何其它原因而暂停，例如：等待设备 I/O（物理读取）或等待客户端活动。</li> <li>• 11 — 已停止。进程已终止。</li> <li>• 12 — 终止中。进程正在终止。</li> <li>• 13 — 未知。无法确定进程状态。</li> <li>• 14 — 远程 I/O。等待远程 (OMNI) 服务器完成一项操作。</li> <li>• 15 — 同步休眠。等待与为执行给定查询而正在并行工作的其它服务器进程同步。</li> </ul> <p>类型: 键</p>
进程状态计数 版本: 11.0 和更高版本	<p>处于一个特定状态下的进程数。</p> <p>类型: 结果</p>
删除行 版本: 11.0 和更高版本	<p>从数据库表中删除的行数。</p> <p>类型: 结果</p>
延迟删除的行 版本: 11.0 和更高版本	<p>以延迟模式从数据库表中删除的行数。</p> <p>类型: 结果</p>
直接删除的行 版本: 11.0 和更高版本	<p>以直接模式从数据库表中删除的行数。</p> <p>类型: 结果</p>
插入行 版本: 11.0 和更高版本	<p>数据库表中插入的行数。</p> <p>类型: 结果</p>
集群插入的行 版本: 11.0 和更高版本	<p>插入带聚簇索引的数据库表中的行数。</p> <p>类型: 结果</p>

数据项	说明
堆表插入的行	插入无聚簇索引的数据库表中的行数。 类型：结果
版本：11.0 和更高版本	
更新的行	对一个数据库表的更新行数。 类型：结果
版本：11.0 和更高版本	
延迟更新的行	需要用两个步骤来完成的更新行数。首先，将删除现有条目和插入新条目的记录写入日志中，但在数据页上实际只执行删除。在第二步，再次扫描日志，并对数据页执行插入操作。 类型：结果
版本：11.0 和更高版本	
直接更新的行	昂贵的、现场的和非现场的更新数目的总和（除延迟更新数外的一切更新）。 类型：结果
版本：11.0 和更高版本	
昂贵更新的行	这种更新是一种直接更新，从原位置删除一行，然后将它插入到新位置。 类型：结果
版本：11.0 和更高版本	
现场更新的行	一种直接更新，不需要在数据页上移动行。 类型：结果
版本：11.0 和更高版本	
非现场更新的行	一种直接更新，不需要移动更新行，但因更新行的长度发生变化，将会移动数据页上的其它行。也称为廉价更新。 类型：结果
版本：11.0 和更高版本	
选择语句	SELECT 或 OPEN CURSOR 语句的数目。 类型：结果
版本：11.0 和更高版本	
服务器结构大小	为 Adaptive Server 结构分配的内存字节数。 类型：结果
版本：11.0 和更高版本	
SQL Server 名称	被监控的 Adaptive Server 的名称，该名称由所用的 Monitor Server 的启动命令中的 -S 参数指定。 类型：结果
版本：11.0 和更高版本	
SQL Server 版本	被监控的 Adaptive Server 的版本。有关详细信息，请参见《Transact-SQL 用户指南》中的全局变量 @@version。 类型：结果
版本：11.0 和更高版本	
超过最大值的线程	是指因为试图超过 Adaptive Server 版本 11.5 和更高版本中配置的服务器端工作线程池的线程数限制值，从而在运行时调整查询计划的次数。 类型：结果
版本：11.5 和更高版本	
超过最大百分比的线程	是指因为试图超过 Adaptive Server 版本 11.5 和更高版本中配置的服务器端工作线程池的线程数限制值，从而在运行时调整查询计划次数的百分比。 类型：结果
版本：11.5 和更高版本	

数据项	说明
使用最多的线程 版本: 11.5 和更高版本	是指服务器端工作线程池中的线程, 在服务器上并发使用的最大数目。 类型: 结果
锁等待时间 版本: 11.0 和更高版本	一个锁请求所等待的时间 (单位为秒)。 类型: 结果
时间戳 版本: 11.0 和更高版本	是指在记录会话视图和回放视图中, 当 <i>summarization_level</i> 是 <i>raw</i> 时, 在收集记录会话数据时, 在 Adaptive Server 上的日期和时间。 在回放视图中, 当 <i>summarization_level</i> 是 <i>actual</i> 、 <i>entire</i> 或一个用户定义的时间间隔时, 该时间将转换为 Historical Server 的时区。 有关详细信息, 请参见 《Transact-SQL 用户指南》中的函数 <i>getdate()</i> 。 类型: 结果
时间戳 Datim 版本: 11.0 和更高版本	是指在记录会话视图和回放视图中, 当 <i>summarization_level</i> 是 <i>raw</i> 时, 收集记录会话数据时 Adaptive Server 的日期和时间 (以 CS_DATETIME 结构返回)。 有关详细信息, 请参见 《Transact-SQL 用户指南》中的函数 <i>getdate()</i> 。 在回放视图中, 当 <i>summarization_level</i> 是 <i>actual</i> 、 <i>entire</i> 或一个用户定义的时间间隔时, 该时间将转换为 Historical Server 的时区。 类型: 结果
事务 版本: 11.0 和更高版本	由 <i>begin transaction</i> 和 <i>commit transaction</i> 语句分隔的已提交的 Transact-SQL 语句块总数。 类型: 结果

# 定义记录会话视图的规范

本附录描述了设计记录会话视图的规则和需考虑的问题。

主题	页码
<a href="#">键和结果的定义</a>	109
<a href="#">设计记录会话视图</a>	110
<a href="#">有效键和结果数据项组合表</a>	112
<a href="#">数据项有效统计类型表</a>	135

## 键和结果的定义

数据项可以是键或结果。

- 键数据项唯一标识视图中返回的行。例如，在定义每个进程数据的视图中，进程 ID 数据项标识每一个进程。
- 结果数据项标识涉及每个键值要返回的信息。

通过混合使用键和结果来定义记录会话视图，您可以控制 Historical Server 返回的数据。

您可以在一个视图中组合键数据项来缩小返回数据的范围。包含每个后续键时，可以看作是在视图定义中添加了词语“每个”。例如，每个进程的每个数据库表的页 I/O。（只有某些组合键对每个数据项都有效。请参见第 112 页的表 B-2。）

要返回全服务器范围数据，不用键来定义视图。在没有键的视图中，结果通常正好包含一行。（只有某些数据项在服务器级是有效的。请参见第 112 页的表 B-2。）

表 B-1 用不同的组合键提供视图的示例。

**表 B-1: 用不同的组合键查看示例**

视图定义	结果
页 I/O	由于视图中没有键，因此结果是整个服务器的页 I/O。例如： <pre> Page I/O ----- 145                     </pre>
进程 ID (键)，登录名，页 I/O	结果是每个进程的页 I/O。例如： <pre> Process ID      Login Name      Page I/O ----- 1              sa              45 5              joe             100                     </pre>
进程 ID (键)，数据库 ID (键)，对象 ID (键)，数据库名，对象名，页 I/O	结果是每个进程的每个数据库表的页 I/O。例如： <pre> Process  Database  Object  Databse  Object  Page ID       ID           ID      Name     Name    I/O ----- 1        5           208003772  pubs2  authors  10 1        5           336004228  pubs2  titles   35 5        5           208003772  pubs2  authors  100                     </pre>

## 设计记录会话视图

定义视图的步骤如下：

- 用第 112 页的表 B-2 选择键数据项和结果数据项的有效组合。
- 用第 136 页的表 B-3 为每个数据项选择有效的统计类型。

## 使用进程 ID

当服务器进程终止时，Adaptive Server 会将它的进程 ID 重新用于一个新的进程。因此，“进程 ID”数据项不能保证唯一地标识进程。但“内核进程 ID”数据项可以唯一地标识进程。

要创建一个显示每个进程数据的非原始回放视图，记录会话视图和回放视图必须包括进程 ID 和内核进程 ID 数据项。如果回放是 raw，只允许使用“进程 ID”。

包含“进程 ID”的视图按如下方式返回行：

- 记录会话视图和原始回放视图。  
这些视图只为存在于采样间隔末尾的进程 ID 返回一行。如果某个服务器进程在一个采样间隔中间终止，则不为其进程 ID 返回行。
- 当会话的 *summarization\_level* 参数是 *actual*、*entire* 或用户定义的间隔时的回放视图。

这些视图为包含在任何输入样本中的所有服务器进程返回行。然而，由于进程 ID 不确保是唯一的，所以内核进程 ID 数据项也必须包含在内，以确保键的唯一性。否则，视图可能错误地汇总两种不同的进程。

## 使用应用程序名

Historical Server 通过汇总所有具有相同应用程序名的进程的数据来累计每个应用程序的性能数据。它只为采样间隔结束时存在的进程收集性能数据。有关详细信息，请参见第 110 页的“使用进程 ID”。

## 视图中空行与没有行

当没有要报告的活动时，某些数据项会导致视图中出现一个空行（也就是结果数据项为零值的行），而其它数据项导致行被省略。控制空行是否在视图中显示的规则是：

- 服务器级数据项总是返回一行，即使没有要报告的活动也如此。
- 包含键数据项进程 ID 或应用程序名的视图仅报告采样周期结束后激活的进程。
- 当在采样周期没有活动报告时，包含键数据项对象 ID 或存储过程 ID 的视图省略行。
- 包括未在前一项中列出的键的视图，即使在没有活动时也都返回行。

## 有效键和结果数据项组合表

可以在视图内被组合的数据项有限制。限制基于数据项之间的关系。例如，请求每个“页码”的“CPU 忙时百分比”是没有意义的，因此不被允许。

**表 B-2** 为视图显示结果和键数据项的有效组合。此表按字母顺序列出了结果数据项。确定一个结果数据项是否可以被包含在记录会话视图中的基本规则是：

- 视图必须包括为结果数据项所列出的*所有*必需的键。
- 视图*可以*包括为结果数据项所列出的可选键。然而，不能确保所有为数据项列出的可选键都可以协调地工作。
- 除了作为必需的键或可选键列出的那些键，视图*不能*包括其它任何键。（注释栏中描述的个别除外。）
- 数据在服务器级是否可用？— 指示是否可以在没有任何键的视图中使用此数据项来获得关于 Adaptive Server 的摘要或静态信息。
- 没有必需的键的结果数据项可以用于服务器级的视图。服务器级的视图没有关于 Adaptive Server 的键和报告摘要或静态信息。
- 脚注中包含关于在视图中使用结果数据项的其它信息。

**表 B-2: 有效键和结果组合**

数据项	必需的键和可选键
应用程序执行类 <sup>2</sup>	在服务器级有效吗？ 否 必需的键： 应用程序名
应用程序名（键） <sup>3</sup>	
阻塞进程 ID	在服务器级有效吗？ 否 必需的键： • 进程 ID • 数据库 ID • 对象 ID • 锁状态 • 页号 可选键： 锁类型



<b>数据项</b>	<b>必需的键和可选键</b>
高速缓存效率	在服务器级有效吗? 否 必需的键: 高速缓存 ID
高速缓存命中百分比	在服务器级有效吗? 否 必需的键: 高速缓存 ID
高速缓存命中数	在服务器级有效吗? 否 必需的键: 高速缓存 ID
高速缓存 ID (键) <sup>4</sup>	
高速缓存未命中数	在服务器级有效吗? 否 必需的键: 高速缓存 ID
高速缓存名 <sup>4</sup>	在服务器级有效吗? 否 必需的键: 高速缓存 ID
高速缓存预取效率	在服务器级有效吗? 否 必需的键: 高速缓存 ID
高速缓存引用和重用	在服务器级有效吗? 否 必需的键: 高速缓存 ID
高速缓存重用	在服务器级有效吗? 否 必需的键: 高速缓存 ID

数据项	必需的键和可选键
高速缓存重新使用脏页	在服务器级有效吗? 是 必需的键: 无 可选键: 高速缓存 ID
高速缓存大小	在服务器级有效吗? 否 必需的键: 高速缓存 ID
高速缓存螺旋锁争用	在服务器级有效吗? 否 必需的键: 高速缓存 ID
代码内存大小	在服务器级有效吗? 是 必需的键: 无
连接时间	在服务器级有效吗? 否 必需的键: 进程 ID
CPU 忙时百分比	在服务器级有效吗? 是 必需的键: 无 可选键: 引擎号
CPU 百分比 <sup>3</sup>	在服务器级有效吗? 否 必需的键: 进程 ID 或 应用程序名 可选键: 引擎号

<b>数据项</b>	<b>必需的键和可选键</b>
CPU 时间 <sup>3</sup>	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <ul style="list-style-type: none"> <li>• 进程 ID 或</li> <li>• 应用程序名</li> <li>• 引擎号</li> </ul>
CPU 放弃	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <p>引擎号</p>
当前应用程序名	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>进程 ID</p>
当前引擎	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>进程 ID</p>
当前执行类 <sup>2</sup>	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>进程 ID</p>
当前进程状态	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>进程 ID</p>
当前语句批处理 ID <sup>2</sup>	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>进程 ID</p>

<b>数据项</b>	<b>必需的键和可选键</b>
当前语句批处理文本 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句批处理文本字节偏移 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句批处理文本是否启用 <sup>2</sup>	在服务器级有效吗? 是 必需的键: 无
当前语句环境 ID <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句 CPU 时间 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句经历时间 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句行号 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
等待后当前授予的语句锁 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前立即授予的语句锁 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID

<b>数据项</b>	<b>必需的键和可选键</b>
当前未授予的语句锁 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句逻辑读取数	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句号 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句页 I/O <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句页写入数 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句物理读取数 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句过程数据库 ID <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句过程数据库名 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句过程 ID <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID

<b>数据项</b>	<b>必需的键和可选键</b>
当前语句过程名 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句过程所有者名 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句过程文本 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句查询计划文本 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句开始时间 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
当前语句文本字节偏移 <sup>2</sup>	在服务器级有效吗? 否 必需的键: 进程 ID
<b>数据库 ID (键)</b>	
数据库名 <sup>1</sup>	在服务器级有效吗? 否 必需的键: 数据库 ID
死锁计数	在服务器级有效吗? 是 必需的键: 无

数据项	必需的键和可选键
请求锁	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 数据库 ID</li> <li>• 对象 ID</li> <li>• 锁状态</li> <li>• 页号</li> </ul> <p>可选键:</p> <p>锁类型</p>
设备命中百分比	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <p>设备名</p>
设备命中数	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <p>设备名</p>
设备 I/O	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <p>设备名</p>
设备未命中数	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <p>设备名</p>
设备名 (键)	

数据项	必需的键和可选键
设备读取数	在服务器级有效吗? 是 必需的键: 无 可选键: 设备名
设备写入数	在服务器级有效吗? 是 必需的键: 无 可选键: 设备名
经历的时间	在服务器级有效吗? 是 必需的键: 无
引擎号 (键)	
主机名	在服务器级有效吗? 否 必需的键: 进程 ID
索引逻辑读取数	在服务器级有效吗? 是 必需的键: 无 可选键: • 进程 ID • 数据库 ID • 对象 ID • 引擎号 • 组合键: • 过程数据库 ID • 过程 ID



<b>数据项</b>	<b>必需的键和可选键</b>
索引物理读取数	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 数据库 ID</li> <li>• 对象 ID</li> <li>• 引擎号</li> <li>• 组合键: <ul style="list-style-type: none"> <li>• 过程数据库 ID</li> <li>• 过程 ID</li> </ul> </li> </ul>
内核进程 ID <sup>1</sup>	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>进程 ID</p>
内核结构内存大小	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p>
拒绝的大 I/O	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>高速缓存 ID</p>
执行的大 I/O	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>高速缓存 ID</p>
请求的大 I/O	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>高速缓存 ID</p>

数据项	必需的键和可选键
锁计数	在服务器级有效吗? 是 必需的键: 无 可选键: <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 锁类型</li> <li>• 锁结果</li> <li>• 汇总的锁结果</li> </ul>
锁命中百分比	在服务器级有效吗? 是 必需的键: 无
锁结果 (键)	
汇总的锁结果 (键)	
锁状态 (键)	
锁状态计数	在服务器级有效吗? 否 必需的键: 锁状态
锁类型 (键)	
锁阻塞数	在服务器级有效吗? 否 必需的键: <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 数据库 ID</li> <li>• 对象 ID</li> <li>• 锁状态</li> <li>• 锁类型</li> <li>• 页号</li> </ul>

数据项	必需的键和可选键
等待后授予的锁 <sup>3</sup>	<p data-bbox="763 230 969 256"><i>在服务器级有效吗?</i></p> <p data-bbox="763 265 783 291">是</p> <p data-bbox="763 300 864 326"><i>必需的键:</i></p> <p data-bbox="763 335 783 361">无</p> <p data-bbox="763 369 844 395"><i>可选键:</i></p> <ul data-bbox="763 404 969 708" style="list-style-type: none"> <li data-bbox="763 404 901 430">• 进程 ID 或</li> <li data-bbox="763 439 901 465">• 应用程序名</li> <li data-bbox="763 473 864 499">• 组合键: <ul data-bbox="763 508 921 595" style="list-style-type: none"> <li data-bbox="763 508 921 534">• 数据库 ID</li> <li data-bbox="763 543 895 569">• 对象 ID</li> </ul> </li> <li data-bbox="763 595 864 621">• 组合键: <ul data-bbox="763 630 969 708" style="list-style-type: none"> <li data-bbox="763 630 969 656">• 过程数据库 ID</li> <li data-bbox="763 664 895 690">• 过程 ID</li> </ul> </li> </ul>
立即授予的锁 <sup>3</sup>	<p data-bbox="763 716 969 743"><i>在服务器级有效吗?</i></p> <p data-bbox="763 751 783 777">是</p> <p data-bbox="763 786 864 812"><i>必需的键:</i></p> <p data-bbox="763 821 783 847">无</p> <p data-bbox="763 855 844 881"><i>可选键:</i></p> <ul data-bbox="763 890 969 1192" style="list-style-type: none"> <li data-bbox="763 890 901 916">• 进程 ID 或</li> <li data-bbox="763 925 901 951">• 应用程序名</li> <li data-bbox="763 960 864 986">• 组合键: <ul data-bbox="763 994 921 1081" style="list-style-type: none"> <li data-bbox="763 994 921 1020">• 数据库 ID</li> <li data-bbox="763 1012 895 1038">• 对象 ID</li> </ul> </li> <li data-bbox="763 1081 864 1107">• 组合键: <ul data-bbox="763 1116 969 1192" style="list-style-type: none"> <li data-bbox="763 1116 969 1142">• 过程数据库 ID</li> <li data-bbox="763 1150 895 1177">• 过程 ID</li> </ul> </li> </ul>

数据项	必需的键和可选键
未授予的锁 <sup>3</sup>	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <ul style="list-style-type: none"> <li>• 进程 ID 或</li> <li>• 应用程序名</li> <li>• 组合键:                             <ul style="list-style-type: none"> <li>• 数据库 ID</li> <li>• 对象 ID</li> </ul> </li> <li>• 组合键:                             <ul style="list-style-type: none"> <li>• 过程数据库 ID</li> <li>• 过程 ID</li> </ul> </li> </ul>
日志争用百分比	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p>
逻辑页读取数	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 引擎号</li> <li>• 组合键:                             <ul style="list-style-type: none"> <li>• 数据库 ID</li> <li>• 对象 ID</li> </ul> </li> <li>• 组合键:                             <ul style="list-style-type: none"> <li>• 过程数据库 ID</li> <li>• 过程 ID</li> </ul> </li> </ul>
登录名 <sup>1</sup>	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <p>进程 ID</p>

<b>数据项</b>	<b>必需的键和可选键</b>
活动最频繁设备 I/O	在服务器级有效吗? 是 必需的键: 无
活动最频繁设备名	在服务器级有效吗? 是 必需的键: 无
接收的网络字节数	在服务器级有效吗? 是 必需的键: 无
发送的网络字节数	在服务器级有效吗? 是 必需的键: 无
网络缺省包大小	在服务器级有效吗? 是 必需的键: 无
网络 I/O 字节数	在服务器级有效吗? 是 必需的键: 无
最大网络包大小	在服务器级有效吗? 是 必需的键: 无
接收的网络包大小	在服务器级有效吗? 是 必需的键: 无
发送的网络包大小	在服务器级有效吗? 是 必需的键: 无

<b>数据项</b>	<b>必需的键和可选键</b>
接收的网络包数	在服务器级有效吗? 是 必需的键: 无
发送的网络包数	在服务器级有效吗? 是 必需的键: 无
引擎数	在服务器级有效吗? 是 必需的键: 无
进程数 <sup>2</sup>	在服务器级有效吗? 是 必需的键: 无 可选键: 应用程序名
对象 ID (键)	
对象名 <sup>1</sup>	在服务器级有效吗? 否 必需的键: • 数据库 ID • 对象 ID
对象类型 <sup>1</sup>	在服务器级有效吗? 否 必需的键: • 数据库 ID • 对象 ID
所有者名称 <sup>1</sup>	在服务器级有效吗? 否 必需的键: • 数据库 ID • 对象 ID

<b>数据项</b>	<b>必需的键和可选键</b>
页高速缓存大小	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p>
页命中百分比	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 引擎号</li> <li>• 组合键: <ul style="list-style-type: none"> <li>• 数据库 ID</li> <li>• 对象 ID</li> </ul> </li> <li>• 组合键: <ul style="list-style-type: none"> <li>• 过程数据库 ID</li> <li>• 过程 ID</li> </ul> </li> </ul>
页 I/O	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 引擎号</li> <li>• 组合键: <ul style="list-style-type: none"> <li>• 数据库 ID</li> <li>• 对象 ID</li> </ul> </li> <li>• 组合键: <ul style="list-style-type: none"> <li>• 过程数据库 ID</li> <li>• 过程 ID</li> </ul> </li> </ul>
页号 (键)	

<b>数据项</b>	<b>必需的键和可选键</b>
页写入数	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <ul style="list-style-type: none"><li>• 进程 ID</li><li>• 引擎号</li><li>• 组合键:<ul style="list-style-type: none"><li>• 数据库 ID</li><li>• 对象 ID</li></ul></li><li>• 组合键:<ul style="list-style-type: none"><li>• 过程数据库 ID</li><li>• 过程 ID</li></ul></li></ul>
物理页读取数	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p> <p>可选键:</p> <ul style="list-style-type: none"><li>• 进程 ID</li><li>• 引擎号</li><li>• 组合键:<ul style="list-style-type: none"><li>• 数据库 ID</li><li>• 对象 ID</li></ul></li><li>• 组合键:<ul style="list-style-type: none"><li>• 过程数据库 ID</li><li>• 过程 ID</li></ul></li></ul>
过程缓冲区大小	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p>



数据项	必需的键和可选键
过程 CPU 时间	<p>在服务器级有效吗？</p> <p>否</p> <p>必需的键：</p> <ul style="list-style-type: none"> <li>• 过程数据库 ID</li> <li>• 过程 ID</li> </ul> <p>可选键：</p> <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 过程语句号</li> <li>• 过程行号</li> </ul>
过程数据库 ID (键)	
过程数据库名 <sup>1</sup>	<p>在服务器级有效吗？</p> <p>否</p> <p>必需的键：</p> <p>过程数据库 ID</p>
过程占用时间	<p>在服务器级有效吗？</p> <p>否</p> <p>必需的键：</p> <ul style="list-style-type: none"> <li>• 过程数据库 ID</li> <li>• 过程 ID</li> </ul> <p>可选键：</p> <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 过程语句号</li> <li>• 过程行号</li> </ul>
过程执行类 <sup>2</sup>	<p>在服务器级有效吗？</p> <p>否</p> <p>必需的键：</p> <ul style="list-style-type: none"> <li>• 过程数据库 ID</li> <li>• 过程 ID</li> </ul>

数据项	必需的键和可选键
过程执行计数	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <ul style="list-style-type: none"> <li>过程数据库 ID</li> <li>过程 ID</li> </ul> <p>可选键:</p> <ul style="list-style-type: none"> <li>进程 ID</li> <li>过程语句号</li> <li>过程行号</li> </ul>
过程头大小	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p>
过程命中百分比	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p>
过程 ID (键)	
过程行号 (键)	
过程行文本	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <ul style="list-style-type: none"> <li>过程数据库 ID</li> <li>过程 ID</li> </ul>
过程逻辑读取数	<p>在服务器级有效吗?</p> <p>是</p> <p>必需的键:</p> <p>无</p>
过程名称 <sup>1</sup>	<p>在服务器级有效吗?</p> <p>否</p> <p>必需的键:</p> <ul style="list-style-type: none"> <li>过程数据库 ID</li> <li>过程 ID</li> </ul>

数据项	必需的键和可选键
过程所有者名 <sup>1</sup>	在服务器级有效吗? 否 必需的键: • 过程数据库 ID • 过程 ID
过程物理读取数	在服务器级有效吗? 是 必需的键: 无
过程语句号 (键)	
进程 ID (键) <sup>3</sup>	
进程状态 (键)	
进程状态计数	在服务器级有效吗? 否 必需的键: 进程状态
删除行	在服务器级有效吗? 是 必需的键: 无
延迟删除的行	在服务器级有效吗? 是 必需的键: 无
直接删除的行	在服务器级有效吗? 是 必需的键: 无
插入行	在服务器级有效吗? 是 必需的键: 无

<b>数据项</b>	<b>必需的键和可选键</b>
集群插入的行	在服务器级有效吗? 是 必需的键: 无
堆表插入的行	在服务器级有效吗? 是 必需的键: 无
更新的行	在服务器级有效吗? 是 必需的键: 无
延迟更新的行	在服务器级有效吗? 是 必需的键: 无
直接更新的行	在服务器级有效吗? 是 必需的键: 无
昂贵更新的行	在服务器级有效吗? 是 必需的键: 无
现场更新的行	在服务器级有效吗? 是 必需的键: 无
非现场更新的行	在服务器级有效吗? 是 必需的键: 无
选择语句	在服务器级有效吗? 是 必需的键: 无

<b>数据项</b>	<b>必需的键和可选键</b>
服务器结构大小	在服务器级有效吗? 是 必需的键: 无
SQL Server 名称	在服务器级有效吗? 是 必需的键: 无
SQL Server 版本	在服务器级有效吗? 是 必需的键: 无
超过最大值的线程 <sup>2</sup>	在服务器级有效吗? 是 必需的键: 无
超过最大百分比的线程 <sup>2</sup>	在服务器级有效吗? 是 必需的键: 无
使用最多的线程 <sup>2</sup>	在服务器级有效吗? 是 必需的键: 无
锁等待时间	在服务器级有效吗? 否 必需的键: <ul style="list-style-type: none"> <li>• 进程 ID</li> <li>• 数据库 ID</li> <li>• 对象 ID</li> <li>• 锁状态</li> <li>• 页号</li> </ul> 可选键: 锁类型

数据项	必需的键和可选键
时间戳	在服务器级有效吗? 是 必需的键: 无
时间戳 Datim	在服务器级有效吗? 是 必需的键: 无
事务	在服务器级有效吗? 是 必需的键: 无

1. 这些数据项根据 ID 或状态类型推断名称。它们在任何包含必需的键的视图中都是有效的，不考虑也可能出现在视图中的其它键。（例如，数据库名称在任何包含数据库 ID 的键的视图中都是有效的；因此，它在键为数据库 ID 和进程 ID 的视图也将有效。）

2. 这些数据项仅在您监控 Adaptive Server 版本 11.5 和更高版本时可用。

3. 进程 ID 和应用程序名不能用在同一个视图中。

4. 在 11.5 以前的版本中，“高速缓存名”是一个键。现在，“高速缓存名”是一个结果。新键“高速缓存 ID”代替了“高速缓存名”键。为保持兼容，“高速缓存名”仍然是有效的键。然而，为了效率和向前兼容，在新视图中应将高速缓存 ID 作为键使用。

## 有效组合示例

- Login Name, Process ID, Page I/O
- Process ID, CPU Percent
- Login Name, Process ID
- Deadlock Count, CPU Busy Percent
- Transactions, Rows Deleted, Rows Inserted, Rows Updated
- Current Engine, Login Name, Process ID
- Current Engine, Login Name, Process ID, CPU Percent
- Login Name, Process ID, Page I/O, Connect Time

## 无效组合示例

- Current Engine, Process ID, Lock Status Count  
锁状态计数在每个进程不可用。
- Login Name, Blocking Process ID  
需要添加其它必需的键数据项。
- Net I/O Bytes, Demand Lock  
请求锁要求键且网络 I/O 字节数对任何键数据项都无效。
- Current Engine, Login Name, Deadlock Count  
死锁计数与任何键数据项都不兼容，但是对当前引擎和登录名要求键。

## 数据项有效统计类型表

有六种有效的统计类型：

- “采样值”
- “会话值”
- “采样速率”
- “会话速率”
- “采样平均值”
- “会话平均值”

然而，不是所有六种类型对所有的数据项都有效。表 B-3 显示了对每个数据项都有效的统计类型。在该表中：

- 空白单元格表示此统计类型对那个数据项无效。
- 当指定了统计类型时，通过显示结果数据的数据类型来表示有效统计类型：
  - long 和 ENUMS 表示整数
  - double 表示浮点数
  - charp 表示字符串

表 B-3: 数据项和它们的有效统计类型

统计类型 数据项	采样值	会话值	采样速率	会话速率	采样平 均值	会话平 均值
应用程序执行类	charp					
应用程序名	charp					
阻塞进程 ID	long					
高速缓存效率	double	double				
高速缓存命中百分比	double	double				
高速缓存命中数	long	long	double	double		
高速缓存 ID	long					
高速缓存未命中数	long	long	double	double		
高速缓存名	charp					
高速缓存预取效率	double	double				
高速缓存引用和重用	long	long	double	double		
高速缓存重用	long	long	double	double		
高速缓存重新使用脏页	long	long	double	double		
高速缓存大小	double					
高速缓存螺旋锁争用	double	double				
代码内存大小	long					
连接时间	long	long				
CPU 忙时百分比	double	double				
CPU 百分比	double	double				
CPU 时间	double	double				
CPU 放弃	long	long	double	double		
当前应用程序名	charp					
当前引擎	long					
当前执行类	charp					
当前进程状态	ENUMS					
当前语句批处理 ID	long					



统计类型 数据项	采样值	会话值	采样速率	会话速率	采样平 均值	会话平 均值
当前语句批处理文本	charp					
当前语句批处理文本字节偏移	long					
当前语句批处理文本是否启用	long					
当前语句环境 ID	long					
当前语句 CPU 时间	double	double				
当前语句经历时间	double	double				
当前语句行号	long					
当前语句被立即授予的锁数	long	long	double	double		
当前语句等待后被授予的锁数	long	long	double	double		
当前语句未被授予的锁数	long	long	double	double		
当前语句逻辑读取数	long	long	double	double		
当前语句号	long					
当前语句页 I/O	long	long	double	double		
当前语句页写入数	long	long	double	double		
当前语句物理读取数	long	long	double	double		
当前语句过程数据库 ID	long					
当前语句过程数据库名	charp					
当前语句过程 ID	long					
当前语句过程名	charp					

统计类型 数据项	采样值	会话值	采样速率	会话速率	采样平 均值	会话平 均值
当前语句过程所有者名	charp					
当前语句过程文本	charp					
当前语句查询计划文本	charp					
当前语句开始时间	charp					
当前语句文本字节偏移	long					
数据库 ID	long					
数据库名	charp					
死锁计数	long	long				
请求锁	charp					
设备命中百分比	double	double				
设备命中数	long	long	double	double		
设备 I/O	long	long	double	double		
设备未命中数	long	long	double	double		
设备名	charp					
设备读取数	long	long	double	double		
设备写入数	long	long	double	double		
经历的时间	long	long				
引擎号	long					
主机名	charp					
索引逻辑读取数	long	long	double	double		
索引物理读取数	long	long	double	double		
内核进程 ID	long					
内核结构内存大小	long					
拒绝的大 I/O	long	long	double	double		
执行的大 I/O	long	long	double	double		
请求的大 I/O	long	long	double	double		
锁计数	long	long	double	double		
锁命中百分比	double	double				

统计类型 数据项	采样值	会话值	采样速率	会话速率	采样平 均值	会话平 均值
锁结果	ENUMS					
汇总的锁结果	ENUMS					
锁状态	ENUMS					
锁状态计数	long	long	double	double		
锁类型	ENUMS					
锁阻塞数	long					
等待后授予 的锁	long	long	double	double		
立即授予的锁	long	long	double	double		
未授予的锁	long	long	double	double		
日志争用百 分比	double	double				
逻辑页读取数	long	long	double	double		
登录名	charp					
活动最频繁设 备 I/O	long	long	double	double		
活动最频繁 设备名	charp					
接收的网络字 节数	long	long	double	double		
发送的网络字 节数	long	long	double	double		
网络缺省包 大小	long					
网络 I/O 字 节数	long	long	double	double		
最大网络包 大小	long					
接收的网络包 大小	double	double	double	double		
发送的网络包 大小	double	double	double	double		
接收的网络 包数	long	long	double	double		
发送的网络 包数	long	long	double	double		
引擎数	long					
进程数	long					

统计类型 数据项	采样值	会话值	采样速率	会话速率	采样平 均值	会话平 均值
对象 ID	long					
对象名	charp					
对象类型	ENUMS					
所有者名称	charp					
页高速缓存 大小	long					
页命中百分比	double	double				
页 I/O	long	long	double	double		
页号	long					
页写入数	long	long	double	double		
物理页读取数	long	long	double	double		
过程缓冲区 大小	long					
过程 CPU 时间	double	double			double	double
过程数据库 ID	long					
过程数据库名	charp					
过程占用时间	double	double			double	double
过程执行类	charp					
过程执行计数	long	long	double	double		
过程头大小	long					
过程命中百 分比	double	double				
过程 ID	long					
过程行号	long					
过程行文本	charp					
过程逻辑读 取数	long	long	double	double		
过程名称	charp					
过程所有者名	charp					
过程物理读 取数	long	long	double	double		
过程语句号	long					
进程 ID	long					
进程状态	ENUMS					
进程状态计数	long					
删除行	long	long	double	double		
延迟删除的行	long	long	double	double		

统计类型 数据项	采样值	会话值	采样速率	会话速率	采样平 均值	会话平 均值
直接删除的行	long	long	double	double		
插入行	long	long	double	double		
集群插入的行	long	long	double	double		
堆表插入的行	long	long	double	double		
更新的行	long	long	double	double		
延迟更新的行	long	long	double	double		
直接更新的行	long	long	double	double		
昂贵更新的行	long	long	double	double		
现场更新的行	long	long	double	double		
非现场更新 的行	long	long	double	double		
选择语句	long	long	double	double		
服务器结构 大小	long					
SQL Server 名称	charp					
SQL Server 版本	charp					
超过最大值的 线程	long	long	double	double		
超过最大百分 比的线程	double	double				
使用最多的 线程	long					
锁等待时间	long					
时间戳	charp					
时间戳 Datim	datim					
事务	long	long	double	double		



# 定义回放视图的规范

本附录描述设计回放视图的规则和需考虑的问题。

主题	页码
<a href="#">汇总级别详细信息</a>	143
<a href="#">设计回放视图</a>	147
<a href="#">有关回放视图的数据项要求的表</a>	150
<a href="#">关于某些数据项的其它信息</a>	155

## 汇总级别详细信息

一个会话中的所有回放视图均按相同的汇总级别回放。使用 `hs_create_playback_session` 命令的 `summarization_level` 参数，可定义回放汇总级别。汇总级别是：

- raw
- actual
- entire
- 用户定义间隔

## 原始回放

该选项用相同的采样间隔回放数据，如同收集数据时一样。选择该选项可查看到与记录时一样的原始数据。这是回放快照数据（如当前的 SQL 语句数据和每个进程状态信息）的唯一可用选项。有关快照数据项的确切列表，请参见第 150 页的表 C-2。（快照数据项是那些在“非原始允许”列中值为“否”的数据项。）

只有当 `target` 是 `client` 时，该选项才有效。

Historical Server 不处理数据。每个回放采样的时间正好与输入会话的采样时间一致。数据与输入会话中的数据正好一致。

原始回放只包括那些完全在回放 *start\_time* 和 *end\_time* 之间的记录会话采样。例如，如果回放会话 *start\_time* 是下午 3 点，并且输入记录会话下午 2:55 开始采样，10 分钟采样一次，那么回放中的第一个采样开始于下午 3:05。

如果回放 *start\_time* 比输入记录会话开始时间晚，那么累计会话值将不作调整，按收集时的状态回放。例如，如果输入记录会话从下午 1 点开始，但回放 *start\_time* 是下午 3 点，那么数据项“设备 I/O”和“会话总计”反映从下午 1 点开始的总 I/O。

## 实际回放

该选项采用与输入记录会话相同的采样间隔回放数据。回放视图可以使用与记录数据不同的统计类型，而且可以包括原视图中没有的估计数据项和计算数据项。它不能包括快照数据，如进程或锁状态数据和当前 SQL 语句数据。有关可以转换或添加到回放视图的数据项的确切列表，请参见第 150 页的表 C-2。

不需要汇总时，选择该选项可增加或改变某些数据项。

只有当 *target* 是 *client* 时，该选项才有效。

**Historical Server** 返回采样的时间间隔与输入会话中采样的时间间隔相对应。但是，为了与指定的 *start\_time* 或 *end\_time* 相符合，或出现了输入会话重叠情况，采样有可能会被截断。

如果请求的回放开始或结束时间落在采样间隔中间，那么 **Historical Server** 按比例分配计数值，并按百分比和比率应用加权平均量。权值是每次输入一个采样所花的秒数。

例如，如果回放会话 *start\_time* 是下午 3 点，并且输入记录会话在下午 2:55 开始采样，10 分钟采样一次，那么回放中的第一个采样开始于下午 2:55，所有的值按 50% 的比例分配，反映一半采样。

如果回放 *start\_time* 比输入记录会话开始时间晚，那么累计会话值按比例回放，反映部分会话。例如，如果输入记录会话从下午 1 点开始，但回放 *start\_time* 为下午 3 点，那么数据项“设备 I/O”和“会话总计”将是下午 3 点开始的 I/O 按比例分配的值。



## 完整回放

该选项将所有输入记录会话的数据汇总在一个采样中回放。例如，七个输入记录会话可以产生一个回放采样。采样间隔是所请求的回放 *start\_time* 和 *end\_time* 之间的时间跨度。

回放视图可以使用与记录数据不同的统计类型，而且可以包括原视图中没有的估计数据项和计算数据项。它不能包括快照数据，如进程或锁状态数据和当前 SQL 语句数据。有关可以转换或添加到回放视图的数据项的确切列表，请参见第 150 页的表 C-2。

选择该选项可合并记录的数据，将长时间积累的详细内容合并到活动概述中。

如果请求的回放开始或结束时间落在输入记录会话中间，那么 Historical Server 按比例分配计数值，并按百分比和比率应用加权平均量。权值是每次输入一个采样所花的秒数。

例如，如果回放会话 *start\_time* 是下午 3 点，*end\_time* 是下午 6 点，而输入记录会话在下午 1 点开始并在晚上 9 点结束，那么 Historical Server 将计算从下午 3 点到下午 6 点之间的这段时间的数据项值。如果为回放指定的开始和结束时间与输入会话的开始和结束时间不一致，它就按比例分配累计（会话）计数，并对累计百分比和比率使用加权平均量。

## 使用用户定义间隔的回放

该选项回放已汇总到指定长度的采样间隔内的数据。参数值是采样间隔，指定为：

“S”  
“M:S”  
“H:M:S”  
“D H:M:S”

其中：

- S 是秒数。
- M 是分钟数。
- H 是小时数。
- D 是天数。

每一部分都是一个一位或两位数字。以下是一些示例：

- "30"（指定 30 秒作为采样间隔）
- "10:0"（指定 10 分钟作为采样间隔）
- "8:30:0"（指定 8.5 小时作为采样间隔）
- "5 0:0:0"（指定 5 天作为采样间隔）

第一个采样间隔从 *start\_time* 开始，每个采样都有指定的长度（最后一个可能有例外）。如果请求的回放开始或结束时间落在初始记录会话采样间隔的中间，那么 **Historical Server** 对回放采样按比例分配记录的数据值。最后一个采样可能较短，如果需要，可结束于指定的 *end\_time*。

选择该选项可将数据按任意需要的间隔进行汇总。这种汇总可以调节活动中的偏差，对于观察随时间变化的趋势很有用。

回放视图可以使用与记录数据不同的统计类型，而且可以包括原视图中没有的估计数据项和计算数据项。有关可以转换或添加到回放视图的数据项的确切列表，请参见第 150 页的表 C-2。

**Historical Server** 对每个采样间隔按比例分配计数值，并按百分比和比率应用加权平均量，从而将数据与请求的回放采样间隔对应起来。权值是每次输入一个采样所花的秒数。

## 汇总间隔摘要

表 C-1 汇总每个 *summarization\_interval* 值提供的回放功能。

**表 C-1: 汇总间隔功能**

功能	原始	实际	完整	定义的间隔
所有记录的数据项都可用于回放。	是	否	否	否
回放视图中允许时间戳。	是	是	是	是
回放视图中允许已改变的统计类型。	否	是	是	是
允许回放到一个文件（创建汇总的记录会话）。	否	否	是	是
回放视图中允许输入视图中没有的计算或估计的数据项。	否	是	是	是
当回放开始于输入会话中间时，会话（累计）数据按比例分配。	否	是	是	是
当回放开始于输入采样中间时，采样数据按比例分配。	否 — 采样被省略	是	是	是
汇总。	否	否	是	是
带适当数据调整的标准采样间隔。	否	否	否	是

## 设计回放视图

本节讲述以下主题：

- 指定输入会话的规则
- 回放视图与输入视图间的关系
- 定义原始回放视图的规则
- 定义非原始回放视图的规则

### 指定输入会话的规则

使用多个输入会话给一个文件创建回放会话时，输入会话中不允许有时间间隔。也就是说，*target* 是 *file* 时，输入会话的结束时间和下一个输入会话的开始时间之间不能有间隔。

例如，如果从星期一到星期五，每天从上午 9 点到下午 5 点收集数据，则不能回放这五个记录会话来创建一个新的、汇总的一周会话。但是，如果从星期一到星期五，每天从上午 9 点到第二天上午 9 点，不间断地收集数据，则可以使用回放功能创建一个新的、汇总的一周会话。另一个不间断的方法是保留上午 9 点到下午 5 点的记录会话，但另增加一组计划从下午 5 点到第二天上午 9 点的记录会话。对非高峰时间采用较长采样间隔，从而减少收集的数据量。

### 回放视图与输入视图间的关系

使用 `hs_create_playback_session` 命令创建一个回放会话。然后用一个或多个 `hs_create_playback_view` 命令为回放会话添加视图。回放会话必须至少含有一个视图。它可以包括多个视图。

在 `hs_create_playback_session` 命令中，可以指定想包括在回放中的输入会话 ID。输入视图是为输入记录会话定义的视图。回放视图定义相应输入视图中哪些数据项要回放。

回放视图来自输入视图。回放视图的名称必须与输入会话中的视图名称一致。有多个输入会话的情况下，视图必须存在于所有输入会话中，而且所有输入会话中的视图定义都必须含有完全相同的数据项列表和相同的过滤器列表。

## 定义视图的规则

定义回放视图的规则根据为回放会话定义的 *summarization\_interval* 是 raw 还是非 raw 而有所不同。

### 定义原始回放视图的规则

当回放会话的 *summarization\_interval* 是 raw 时：

- 输入视图中的所有键数据项 **必须** 包括在回放视图中。表 C-2 中标记为“键”的列指示哪些数据项是键。对键唯一有效的统计类型是“采样值”。
- 输入视图中的结果数据项在回放视图中是可选的。
- 回放视图中所有数据项的统计类型必须与输入视图中使用的相同。
- 任意回放视图都可以包括以下这些数据项：
  - “时间戳” “采样值”
  - “时间戳日期时间” “采样值”
  - “经历时间” “采样值”
  - “经历时间” “会话值”

如果想在输入视图中回放所有数据项，而且不需要增加时间戳，请使用不带 *data\_item\_name\_n* 和 *data\_item\_stat\_n* 参数的 *hs\_create\_playback\_view* 命令。忽略这些参数时，缺省设置是来自相同名称的输入视图的所有数据项定义回放视图。

### 定义非原始回放视图的规则

当 *summarization\_interval* 是 *entire*、*actual* 或摘要间隔时：

- 输入视图中的所有键数据项 **必须** 包括在回放视图中。第 150 页的表 C-2 中标记为“键”的列指示哪些数据项是键。对键唯一有效的统计类型是“采样值”。
- 任意回放视图都可以包括以下这些数据项：
  - “时间戳” “采样值”
  - “时间戳日期时间” “采样值”
  - “经历时间” “采样值”
  - “经历时间” “会话值”
- 表 C-2 中标记为“是否对非原始有效？”的列指示在非原始回放视图中哪些结果数据项是允许的。标记为“包括条件”的列表示允许数据项的条件。

- 如果“包括条件”列指示“完全”，那么不管输入视图中使用的是何种统计类型，对该数据项有效的全部统计类型都可用于回放视图。通过第 136 页的表 B-3 可确定每个数据项的有效统计类型。
- 如果“包括条件”列指示的是“估计”，要想将数据项包括在回放视图中，下面的两个条件必须都成立：
  - 数据项的统计类型在输入视图中必须是“采样值”，并且
  - 回放会话必须定义为允许估计。（在 `hs_create_playback_session` 命令中，`allow_estimation` 参数必须为 `allow`。）

无论输入视图中使用的是何种统计类型，对该数据项有效的所有统计类型都可用于回放视图。

- 如果“包括条件”列指示的是“已计算”，要想将数据项包括在回放视图中，下面的条件之一必须成立：
  - 在标记为“计算所需要的其它数据项”的列中列出的数据项存在于输入视图中。如果它成立，则 Historical Server 在回放过程中会计算值；或者
  - 如果计算所需的数据项不在输入视图中，则 Historical Server 在回放过程中可以估计值。要得到这种结果，如前所述的估计条件必须为真。

无论输入视图中使用的是何种统计类型，对该数据项有效的所有统计类型都可用于回放视图。

## 有关回放视图的数据项要求的表

表 C-2 描述在回放视图中包括数据项所需要的条件。请阅读上一节关于列的解释。

**表 C-2: 回放视图的数据项要求**

数据项（回放视图中要求的输入视图中的键）	是否对非原始有效？	包括条件
应用程序执行类	有效	采样值
应用程序名（键）		
阻塞进程 ID	否	
高速缓存效率	有效	估计
高速缓存命中百分比	有效	计算依据： <ul style="list-style-type: none"> <li>• 高速缓存命中数</li> <li>• 高速缓存未命中数</li> </ul>
高速缓存命中数	有效	完全
高速缓存 ID（键） <sup>1</sup>		
高速缓存未命中数	有效	完全
高速缓存名 <sup>1</sup>	有效	采样值
高速缓存预取效率	有效	估计
高速缓存引用和重用	有效	完全
高速缓存重用	有效	完全
高速缓存重新使用脏页	有效	完全
高速缓存大小	有效	采样值
高速缓存螺旋锁争用	有效	估计
代码内存大小	有效	采样值
连接时间	有效	完全
CPU 忙时百分比	有效	估计
CPU 百分比	有效	估计
CPU 时间	有效	完全
CPU 放弃		
当前应用程序名		
当前引擎	否	
当前执行类	否	
当前进程状态	否	
当前语句批处理 ID	否	
当前语句批处理文本	否	
当前语句批处理文本字节偏移	否	
当前语句批处理文本是否启用	否	

数据项（回放视图中要求的输入视图中的键）	是否对非原始有效？	包括条件
当前语句高速缓存读取数	否	
当前语句环境 ID	否	
当前语句 CPU 时间	否	
当前语句经历时间	否	
当前语句行号	否	
当前语句被立即授予的锁数	否	
当前语句等待后被授予的锁数	否	
当前语句未被授予的锁数	否	
当前语句最大文本配置大小	否	
当前语句最大文本缺省大小	否	
当前语句最大文本运行大小	否	
当前语句号	否	
当前语句页 I/O	否	
当前语句页写入数	否	
当前语句物理读取数	否	
当前语句过程数据库 ID	否	
当前语句过程数据库名	否	
当前语句过程 ID	否	
当前语句过程名	否	
当前语句过程所有者名	否	
当前语句过程文本	否	
当前语句查询计划文本	否	
当前语句开始时间	否	
当前语句文本字节偏移	否	
数据库 ID（键）	有效	
数据库名	有效	采样值
死锁计数	有效	完全
请求锁	否	
设备命中百分比	有效	计算依据： • 设备命中数 • 设备未命中数
设备命中数	有效	完全
设备 I/O	有效	完全
设备未命中数	有效	完全
设备名（键）	有效	采样值
设备读取数	有效	完全

数据项（回放视图中要求的输入视图中的键）	是否对非原始有效？	包括条件
设备写入数	有效	完全
经历时间 <sup>2</sup>	有效	完全
引擎号（键）	有效	采样值
主机名	否	
索引逻辑读取数	有效	完全
索引物理读取数	有效	完全
内核进程 ID（键）	有效	采样值
内核结构内存大小	有效	采样值
拒绝的大 I/O	有效	完全
执行的大 I/O	有效	完全
请求的大 I/O	有效	完全
锁计数	有效	完全
锁命中百分比	有效	估计
锁结果（键）	有效	采样值
汇总的锁结果（键）	有效	采样值
锁状态（键）	有效	采样值
锁状态计数	否	
锁类型（键）	有效	采样值
锁阻塞数	否	
立即授予的锁	有效	完全
等待后授予的锁	有效	完全
未授予的锁	有效	完全
日志争用百分比	有效	估计
逻辑页读取数	有效	完全
登录名	有效	采样值
活动最频繁设备 I/O	否	
活动最频繁设备名	否	
接收的网络字节数	有效	完全
发送的网络字节数	有效	完全
网络缺省包大小	有效	采样值
网络 I/O 字节数	有效	完全
最大网络包大小	有效	采样值
接收的网络包大小	有效	计算依据： <ul style="list-style-type: none"> <li>• 接收的网络包数</li> <li>• 接收的网络字节数</li> </ul>



数据项（回放视图中要求的输入视图中的键）	是否对非原始有效？	包括条件
发送的网络包大小	有效	计算依据： <ul style="list-style-type: none"> <li>• 发送的网络包数</li> <li>• 发送的网络字节数</li> </ul>
接收的网络包数	有效	完全
发送的网络包数	有效	完全
引擎数	有效	采样值
进程数	有效	估计
对象 ID（键）	有效	采样值
对象名	有效	采样值
对象类型	有效	采样值
所有者名称	有效	采样值
页高速缓存大小	有效	采样值
页命中百分比	有效	计算依据： <ul style="list-style-type: none"> <li>• 逻辑读取数</li> <li>• 物理读取数</li> </ul>
页 I/O	有效	完全
页号（键）	有效	采样值
页写入数	有效	完全
物理页读取数	有效	完全
过程缓冲区大小	有效	采样值
过程 CPU 时间		
过程数据库 ID（键）	有效	采样值
过程数据库名	有效	完全
过程占用时间	有效	需要：过程执行计数 有关详细信息，请参见本文中的“使用‘过程占用时间’和‘过程 CPU 时间’”
过程执行类	否	
过程执行计数	有效	完全
过程头大小	有效	采样值
过程命中百分比	有效	计算依据： <ul style="list-style-type: none"> <li>• 过程逻辑读取数</li> <li>• 过程物理读取数</li> </ul>
过程 ID（键）	有效	采样值
过程行号（键）	有效	采样值

数据项（回放视图中要求的输入视图中的键）	是否对非原始有效？	包括条件
过程行文本	有效	采样值
过程逻辑读取数	有效	完全
过程名称	有效	采样值
过程所有者名	有效	采样值
过程物理读取数	有效	完全
过程语句号（键）	有效	采样值
进程 ID（键）	有效	采样值
进程状态（键）	有效	采样值
进程状态计数	否	
删除行	有效	完全
延迟删除的行	有效	完全
直接删除的行	有效	完全
插入行	有效	完全
集群插入的行	有效	完全
堆表插入的行	有效	完全
更新的行	有效	完全
延迟更新的行	有效	完全
直接更新的行	有效	完全
昂贵更新的行	有效	完全
现场更新的行	有效	完全
非现场更新的行	有效	完全
选择语句	有效	完全
服务器结构大小	有效	采样值
SQL Server 名称	有效	采样值
SQL Server 版本	有效	采样值
超过最大值的线程	有效	完全
超过最大百分比的线程	有效	完全
使用最多的线程	否	
锁等待时间	否	
时间戳 <sup>2</sup>	有效	采样值
时间戳 Datim <sup>2</sup>	有效	采样值
事务	有效	完全

数据项（回放视图中要求的输入视图中的键）	是否对非原始有效？	包括条件
1. 在 11.5 以前的版本中，“高速缓存名”是一个键。现在，“高速缓存名”是一个结果。新键“高速缓存 ID”代替了“高速缓存名”键。为保持兼容，“高速缓存名”仍然是有效的键。然而，为了提高效率和向前兼容，在新视图将“高速缓存 ID”作为键使用。		
2. 这些数据项即使不存在于输入视图中，也可以包括在任意回放视图中。		

## 关于某些数据项的其它信息

本节提供有关在视图中使用某些数据项的其它信息。

### 使用“时间戳”、“时间戳日期时间”和“经历时间”

下面的数据项即使不在输入视图中，也始终可以包括在回放视图中：

“时间戳”“采样值”  
“时间戳日期时间”“采样值”  
“经历时间”“采样值”  
“经历时间”“会话值”

可以创建只含有一个或多个与时间相关的这些数据项的回放视图。为此，您需有来自输入会话的有效视图名，即输入会话中必须有某个您不想回放的视图。

当 *summarization\_level* 是 *raw* 时，“时间戳”和“时间戳日期时间”值与原始数据文件中的值相同，即最初进行记录时 Adaptive Server 报告的时间。当 *summarization\_level* 是 *actual*、*entire* 或用户定义的间隔时，“时间戳”和“时间戳日期时间”值反映进行汇总的 Historical Server 系统上的时区。

### 使用进程 ID

当服务器进程终止时，Adaptive Server 会将它的进程 ID 重新用于一个新的进程。因此，“进程 ID”数据项不能保证唯一地标识进程。但“内核进程 ID”数据项可以唯一地标识进程。

要创建显示每个进程数据的非原始回放视图，输入视图和回放视图必须包括“进程 ID”和“内核进程 ID”数据项。如果回放是原始类型，则只允许使用“进程 ID”。

包含“进程 ID”的视图按如下方式返回行：

- 记录会话视图和原始回放视图。

这些视图只为存在于采样间隔末尾的进程 ID 返回一行。如果某个服务器进程在一个采样间隔中间终止，则不为其进程 ID 返回行。

- 当会话的 *summarization\_level* 参数是 *actual*、*entire* 或用户定义的间隔时的回放视图。

这些视图为包括在任意输入视图中的所有服务器进程返回行。但是，因为不能保证“进程 ID”是唯一的，所以还必须包括“内核进程 ID”以保证该键的唯一性。否则，视图可能错误地汇总两种不同的进程。

## 使用“过程占用时间”和“过程 CPU 时间”

本节讨论如何使用可带有“采样平均值”和“会话平均值”统计类型的数据项。这些数据项是：

- 过程占用时间
- 过程 CPU 时间

如果回放视图使用带有“采样值”或“会话值”统计类型的数据项名，那么输入视图必须有相同的统计类型。

如果回放视图使用带“采样平均值”或“会话平均值”统计类型的数据项名：

- 如果输入视图包括附加的数据项“过程执行计数”，那么返回的值是*计算得出的*。
- 如果下面所有条件都成立，则返回的值是*估计出来的*：
  - “过程执行计数”不在输入视图中，并且
  - 带有“采样平均值”统计类型的“过程占用时间”存在于输入视图中，而且
  - 回放会话允许估计。即，在 *hs\_create\_playback\_session* 命令中，*allow\_estimation* 参数设置为 *allow*。

## 记录会话视图的示例

本附录中包含了 Historical Server 有效视图的示例。在安装目录的 *sample/histserver* 子目录下的 *views* 文件中，也会出现这些视图。

这里有些视图收集了您需要的信息，其它视图可以作建立视图的模板使用。

有些样本视图与其它视图的区别只是数据累计的时间间隔（记录会话持续时间，或最近一次采样间隔）不同。其它视图含有顺序不同的相似数据项。数据项在视图中出现的顺序很重要，因为数据是按键字段排序的。出现在视图定义中的第一个键字段充当主排序键，第二个键字段是第二排序键，以此类推。

不要将 *views* 文件直接输入给 *isql*，原因如下：

- 为单个记录会话定义该文件中包括的所有视图，这种情况不太可能出现。
- 该文件中有些注释行，与含有视图定义的实际文本的行相混合。Historical Server 不理解注释行。剪切和粘贴视图定义时，放弃注释行。

主题	页码
<a href="#">高速缓存性能摘要</a>	158
<a href="#">数据库对象锁状态</a>	159
<a href="#">数据库对象页 I/O</a>	159
<a href="#">各个高速缓存的数据高速缓存活动</a>	160
<a href="#">记录会话的数据高速缓存统计信息</a>	160
<a href="#">采样间隔期间的数据高速缓存统计信息</a>	161
<a href="#">记录会话的设备 I/O</a>	161
<a href="#">采样间隔期间的设备 I/O</a>	162
<a href="#">设备 I/O 性能摘要</a>	162
<a href="#">引擎活动</a>	163
<a href="#">锁性能摘要</a>	163
<a href="#">记录会话的网络活动</a>	163
<a href="#">采样间隔期间的网络活动</a>	164
<a href="#">网络性能摘要</a>	164

主题	页码
页 I/O	165
记录会话的过程高速缓存统计信息	165
采样间隔期间的过程高速缓存统计信息	166
过程页 I/O	166
进程活动	167
进程数据库对象页 I/O	167
进程的锁明细	168
进程的页 I/O 明细	169
进程锁	169
进程页 I/O	170
进程状态摘要	170
进程存储过程页 I/O	171
服务器性能摘要	171
存储过程活动	172
事务活动	173

## 高速缓存性能摘要

该视图表示 Adaptive Server 高速缓存在最近一次采样间隔期间的总体效率。它显示由 Adaptive Server 数据高速缓存满足的数据页读取数所占的百分比，以及由 Adaptive Server 过程高速缓存满足的过程执行请求数所占的百分比。

```

hs_create_view cache_perf_sum,
  "Page Hit Percent", "Value for Sample",
  "Procedure Hit Percent", "Value for Sample"

```

## 数据库对象锁状态

该视图显示到最近一次采样间隔结束时 Adaptive Server 进程所持有或请求的数据库对象锁的状态。每个锁用被锁定对象的名称和 ID、含有该对象的数据库名称和 ID、锁应用的页号（如果它是页锁）来标识。

也可以通过它的登录名、进程 ID 和内核进程 ID 来标识每个与锁关联的 Adaptive Server 进程。将同时显示锁的类型、锁的当前状态以及这是否是一个请求锁的指示。如果锁是进程请求的，则显示该进程等待以获得锁的时间量和已持有锁的进程的“进程 ID”。如果该进程已持有锁，则显示等待获得该锁的其它进程的计数。

```
hs_create_view object_lock_status,
"Database ID", "Value for Sample",
"Database Name", "Value for Sample",
"Object ID", "Value for Sample",
"Object Name", "Value for Sample",
"Page Number", "Value for Sample",
"Login Name", "Value for Sample",
"Process ID", "Value for Sample",
"Kernel Process ID", "Value for Sample",
"Lock Type", "Value for Sample",
"Lock Status", "Value for Sample",
"Demand Lock", "Value for Sample",
"Time Waited on Lock", "Value for Sample",
"Blocking Process ID", "Value for Sample",
"Locks Being Blocked Count", "Value for Sample"
```

## 数据库对象页 I/O

该视图显示 Adaptive Server 数据库中的对象以及与其关联的页 I/O。它显示 Adaptive Server 数据库名称和 ID，以及每个数据库中的对象名和 ID。对于每个对象，该视图显示最近一次采样间隔期间以及记录会话期间发生的与其关联的逻辑页读取次数、物理页读取次数和页写入次数。

```
hs_create_view object_page_io,
"Database ID", "Value for Sample",
"Database Name", "Value for Sample",
"Object ID", "Value for Sample",
"Object Name", "Value for Sample",
"Logical Page Reads", "Value for Sample",
"Physical Page Reads", "Value for Sample",
"Page Writes", "Value for Sample",
```

```
"Logical Page Reads", "Value for Session",  
"Physical Page Reads", "Value for Session",  
"Page Writes", "Value for Session"
```

## 各个高速缓存的数据高速缓存活动

该视图只能用于 SQL Server 版本 11.0.x 以及 Adaptive Server 版本 11.5 和更高版本。它包含了关于单个数据高速缓存性能的信息。

对于 Adaptive Server 中配置的每个命名高速缓存，以及缺省数据高速缓存，该视图收集该高速缓存的名称以及绑定到该高速缓存的对象自记录会话开始以来由该高速缓存满足的页读取数所占的百分比。

通过该视图还可以测量高速缓存的空间利用效率，以及自会话开始以来为了获得高速缓存的螺旋锁而被迫等待的时间所占的百分比。它还收集会话实际的高速缓存命中数和未命中数。

```
hs_create_view data_cache_sum,  
"Cache Name", "Value for Sample",  
"Cache Hit Pct", "Value for Session",  
"Cache Efficiency", "Value for Session",  
"Cache Spinlock Contention", "Value for Session",  
"Cache Hits", "Value for Session",  
"Cache Misses", "Value for Session"
```

## 记录会话的数据高速缓存统计信息

该视图表示从记录会话开始以来 Adaptive Server 的组合数据高速缓存的总体效率。它显示记录会话从高速缓存实现的页读取数的请求百分比。

该视图还显示记录会话期间发生的逻辑页读取次数、物理页读取次数、页写入次数以及它们的速率。

```
hs_create_view session_page_cache_stats,  
"Page Hit Percent", "Value for Session",  
"Logical Page Reads", "Value for Session",  
"Logical Page Reads", "Rate for Session",  
"Physical Page Reads", "Value for Session",  
"Physical Page Reads", "Rate for Session",  
"Page Writes", "Value for Session",  
"Page Writes", "Rate for Session"
```



## 采样间隔期间的数据高速缓存统计信息

该视图表示最近一次采样间隔期间 Adaptive Server 组合数据高速缓存的总体效率。它显示最近一次采样间隔从高速缓存实现的页读取数的请求百分比。

该视图还显示最近一次采样间隔期间发生的逻辑页读取次数、物理页读取次数、页写入次数以及它们的速率。

```
hs_create_view sample_page_cache_stats,
"Page Hit Percent", "Value for Sample",
"Logical Page Reads", "Value for Sample",
"Logical Page Reads", "Rate for Sample",
"Physical Page Reads", "Value for Sample",
"Physical Page Reads", "Rate for Sample",
"Page Writes", "Value for Sample",
"Page Writes", "Rate for Sample"
```

## 记录会话的设备 I/O

该视图表示从记录会话开始以来，在 Adaptive Server 数据库设备上发生的 I/O 活动。它用名称标识每个设备。设备 I/O 级别有两种表示方法：一种是以自会话开始以来发生的设备 I/O、读取和写入操作总计数表示；另一种是以总速率表示，即自会话开始以来每秒发生的 I/O、读取和写入操作总数。

```
hs_create_view session_device_io,
"Device Name", "Value for Sample",
"Device Reads", "Value for Session",
"Device Writes", "Value for Session",
"Device I/O", "Value for Session",
"Device Reads", "Rate for Session",
"Device Writes", "Rate for Session",
"Device I/O", "Rate for Session"
```

## 采样间隔期间的设备 I/O

该视图表示在最近一次采样间隔期间，在 Adaptive Server 数据库设备上发生的 I/O 活动。它用名称标识每个设备。设备 I/O 级别有两种表示方法：一种是以最近一次采样间隔期间发生的设备 I/O、读取和写入操作总计数表示，另一种是以速率表示，即在此采样间隔期间每秒发生的 I/O、读取和写入操作总数。

```
hs_create_view sample_device_io,  
  "Device Name", "Value for Sample",  
  "Device I/O", "Value for Sample",  
  "Device Reads", "Value for Sample",  
  "Device Writes", "Value for Sample",  
  "Device I/O", "Rate for Sample",  
  "Device Reads", "Rate for Sample",  
  "Device Writes", "Rate for Sample"
```

## 设备 I/O 性能摘要

该视图表示从记录会话开始累计的、由 Adaptive Server 对数据库设备进行的读取数和写入数。它显示从会话开始以来数据库设备读取和写入的总体速率，以及这段时间内活动最频繁的数据库设备和活动最频繁设备的读取、写入速率。

```
hs_create_view device_perf_sum,  
  "Device I/O", "Rate for Session",  
  "Most Active Device Name", "Value for Session",  
  "Most Active Device I/O", "Rate for Session"
```

## 引擎活动

该视图显示最近一次采样间隔期间，每个活动 Adaptive Server 引擎的活动级别。对于每个引擎，当该引擎使用 CPU 时，显示采样间隔的百分比。

它还显示采样间隔期间该引擎产生的逻辑页读取数、物理页读取数和页写入数。

```
hs_create_view engine_activity,
"Engine Number", "Value for Sample",
"CPU Busy Percent", "Value for Sample",
"Logical Page Reads", "Value for Sample",
"Physical Page Reads", "Value for Sample",
"Page Writes", "Value for Sample"
```

## 锁性能摘要

该视图表示最近一次采样间隔期间请求和授予的每种锁的总数。

```
hs_create_view lock_perf_sum,
"Lock Type", "Value for Sample",
"Lock Results Summarized", "Value for Sample",
"Lock Count", "Value for Sample"
```

## 记录会话的网络活动

该视图表示从记录会话开始所有 Adaptive Server 网络连接上的网络活动。它显示缺省包大小、最大包大小，以及从会话开始以来发送和接收的平均包大小。该视图显示发送包的数目、接收包的数目，以及发送和接收包的速率。它还显示发送的字节数、接收的字节数，以及发送和接收字节的速率。

```
hs_create_view session_network_activity,
"Net Default Packet Size", "Value for Sample",
"Net Max Packet Size", "Value for Sample",
"Net Packet Size Sent", "Value for Session",
"Net Packet Size Received", "Value for Session",
"Net Packets Sent", "Value for Session",
"Net Packets Received", "Value for Session",
"Net Packets Sent", "Rate for Session",
```

```
"Net Packets Received", "Rate for Session",  
"Net Bytes Sent", "Value for Session",  
"Net Bytes Received", "Value for Session",  
"Net Bytes Sent", "Rate for Session",  
"Net Bytes Received", "Rate for Session"
```

## 采样间隔期间的网络活动

该视图表示在最近一次采样间隔期间，所有 Adaptive Server 网络连接上的网络活动。它显示缺省包大小、最大包大小，以及该采样间隔期间发送和接收的平均包大小。该视图显示发送包的数目、接收包的数目，以及发送和接收包的速率。它还显示发送的字节数、接收的字节数，以及发送和接收字节的速率。

```
hs_create_view sample_network_activity,  
"Net Default Packet Size", "Value for Sample",  
"Net Max Packet Size", "Value for Sample",  
"Net Packet Size Sent", "Value for Sample",  
"Net Packet Size Received", "Value for Sample",  
"Net Packets Sent", "Value for Sample",  
"Net Packets Received", "Value for Sample",  
"Net Packets Sent", "Rate for Sample",  
"Net Packets Received", "Rate for Sample",  
"Net Bytes Sent", "Value for Sample",  
"Net Bytes Received", "Value for Sample",  
"Net Bytes Sent", "Rate for Sample",  
"Net Bytes Received", "Rate for Sample"
```

## 网络性能摘要

该视图表示在最近一次采样间隔期间，所有 Adaptive Server 网络连接上 Adaptive Server 活动的速率。它显示该间隔期间 Adaptive Server 每秒接收和发送的字节数。

```
hs_create_view network_perf_sum,  
"Net Bytes Received", "Rate for Sample",  
"Net Bytes Sent", "Rate for Sample"
```

## 页 I/O

用该视图可以确定在记录会话持续期间，对哪些表进行的页读取数最多。因为页读取数也按每个样本累计，所以用它还可以跟踪一天的不同时间内表的活动。可以任意设置记录会话的采样间隔，使该间隔在一天的不同时间内最有利于建立活动趋势。

对于整个记录会话，您可能只想累计总的页读取数。这种情况下，可以将采样间隔设置成接近记录会话的长度，并从视图中删除“逻辑页读取数”、“采样值”。

```
hs_create_view page_ios,
  "Database ID", "Value for Sample",
  "Database Name", "Value for Sample",
  "Object ID", "Rate for Sample",
  "Logical Page Reads", "Value for Sample",
  "Logical Page Reads", "Rate for Session"
```

---

**注释** 该视图为给定数据库表收集的页 I/O 计数，除了对该表本身的数据页执行的 I/O 外，还包括对该表的索引执行的 I/O。

---

## 记录会话的过程高速缓存统计信息

该视图表示从记录会话开始 Adaptive Server 的过程高速缓存的效率。它为执行存储过程显示过程高速缓存实现的请求百分比。

该视图还显示自会话开始以来存储过程的逻辑读取数和物理读取数，以及自会话开始以来存储过程的逻辑读取和物理读取总速率。

```
hs_create_view session_procedure_cache_stats,
  "Procedure Hit Percent", "Value for Session",
  "Procedure Logical Reads", "Value for Session",
  "Procedure Logical Reads", "Rate for Session",
  "Procedure Physical Reads", "Value forSession",
  "Procedure Physical Reads", "Rate for Session"
```

## 采样间隔期间的过程高速缓存统计信息

该视图表示最近一次采样间隔期间 Adaptive Server 的过程高速缓存的效率。它为执行存储过程显示最近一次采样间隔过程高速缓存实现的请求百分比。

该视图还显示最近一次采样间隔期间存储过程的逻辑读取数和物理读取数，以及最近一次采样间隔期间存储过程的逻辑读取和物理读取速率。

```
hs_create_view sample_procedure_cache_stats,
"Procedure Hit Percent", "Value for Sample",
"Procedure Logical Reads", "Value for Sample",
"Procedure Logical Reads", "Rate for Sample",
"Procedure Physical Reads", "Value for Sample",
"Procedure Physical Reads", "Rate for Sample"
```

## 过程页 I/O

该视图表示在最近一次采样间隔期间，运行存储过程时发生的页 I/O。对于在该采样间隔期间生成页 I/O 的每个存储过程，该视图都会显示存储过程名和 ID 以及包含该过程的数据库名和 ID。如果在没有存储过程是活动时产生了页 I/O，则这些 I/O 与过程 ID 和数据库 ID 的零值关联。

在每个存储过程级别，该视图还显示以下项：页 I/O 总数；可以由 Adaptive Server 数据高速缓存满足的页 I/O 请求数所占的百分比；最近一次采样间隔期间执行存储过程时产生的逻辑页读取数、物理页读取数和页写入数。

```
hs_create_view procedure_page_cache_io,
"Procedure Database Name", "Value for
Sample","Procedure Database ID", "Value for
Sample","Procedure Name", "Value for Sample","Procedure
ID", "Value for Sample","Page I/O", "Value for
Sample","Page Hit Percent", "Value for Sample","Logical
Page Reads", "Value for Sample","Physical Page Reads",
"Value for Sample","Page Writes", "Value for Sample"
```

## 进程活动

该视图显示 Adaptive Server 中所有进程的 CPU 利用率、页 I/O 数和当前进程状态。

对于最近一次采样间隔，给出了每个进程的登录名、进程 ID 和内核进程 ID 及它当前的进程状态。

该视图还显示自记录会话开始以来累计的每个进程的连接时间、页 I/O 总数和 CPU 使用时间。

```
hs_create_view process_activity,
  "Login Name", "Value for Sample",
  "Process ID", "Value for Sample",
  "Kernel Process ID", "Value for Sample",
  "Connect Time", "Value for Session",
  "Page I/O", "Value for Session",
  "CPU Time", "Value for Session",
  "Current Process State", "Value for Sample"
```

## 进程数据库对象页 I/O

该视图按数据库对象表示每个 Adaptive Server 进程的页 I/O。它显示最近一次采样间隔期间，每个有页 I/O 的进程的登录名、进程 ID 和内核进程 ID。

对于每个这样的进程和它所访问的每个数据库对象，该视图还显示对象名和对象 ID、对象的数据库名和数据库 ID 以及与该对象关联的页 I/O。

该视图还显示以下项：页 I/O 总数；可以由 Adaptive Server 高速缓存满足的页 I/O 请求数所占的百分比；最近一次采样间隔期间发生的逻辑页读取数、物理页读取数和页写入数。

```
hs_create_view process_object_page_io,
  "Login Name", "Value for Sample",
  "Process ID", "Value for Sample",
  "Kernel Process ID", "Value for Sample",
  "Database Name", "Value for Sample",
  "Database ID", "Value for Sample",
  "Object Name", "Value for Sample",
  "Object ID", "Value for Sample",
  "Object Type", "Value for Sample",
  "Page I/O", "Value for Sample",
  "Page Hit Percent", "Value for Sample",
```

```
"Logical Page Reads", "Value for Sample",  
"Physical Page Reads", "Value for Sample",  
"Page Writes", "Value for Sample"
```

## 进程的锁明细

该视图显示到最近一次采样间隔结束时， Adaptive Server 进程持有或请求的锁的状态。每个锁由以下项标识：与该锁关联的 Adaptive Server 进程的登录名、进程 ID 和内核进程 ID，被锁定对象的名称和 ID，含有该对象的数据库名和 ID，以及该锁应用的页号（如果它是页锁）。系统显示每个锁的当前状态，作出该锁是否是请求锁的指示。如果锁是进程请求的，则显示该进程等待以获得锁的时间量和已持有锁的进程的“进程 ID”。如果该进程已持有该锁，则显示等待获得该锁的其它进程的计数。

```
hs_create_view process_detail_locks,  
"Login Name", "Value for Sample",  
"Process ID", "Value for Sample",  
"Kernel Process ID", "Value for Sample",  
"Database Name", "Value for Sample",  
"Database ID", "Value for Sample",  
"Object Name", "Value for Sample",  
"Object ID", "Value for Sample",  
"Page Number", "Value for Sample",  
"Lock Status", "Value for Sample",  
"Demand Lock", "Value for Sample",  
"Time Waited on Lock", "Value for Sample",  
"Blocking Process ID", "Value for Sample",  
"Locks Being Blocked Count", "Value for Sample"
```



## 进程的页 I/O 明细

该视图表示每个 Adaptive Server 进程的页 I/O 的详细情况。显示在最近一次采样间隔末尾时，每个 Adaptive Server 进程的登录名、进程 ID、内核进程 ID、当前进程状态和当前引擎。该视图显示采样间隔期间以及会话开始之后，可由 Adaptive Server 数据高速缓存实现的页 I/O 请求的百分比。

它还显示自记录会话开始以来累计的逻辑页读取数、物理页读取数和页写入数。

```
hs_create_view process_detail_io,
"Login Name", "Value for Sample",
"Process ID", "Value for Sample",
"Kernel Process ID", "Value for Sample",
"Current Process State", "Value for Sample",
"Current Engine", "Value for Sample",
"Connect Time", "Value for Session",
"CPU Time", "Value for Session",
"Page Hit Percent", "Value for Sample",
"Page Hit Percent", "Value for Session",
"Logical Page Reads", "Value for Session",
"Physical Page Reads", "Value for Session",
"Page Writes", "Value for Session"
```

## 进程锁

该视图显示最近一次采样间隔期间，Adaptive Server 中产生锁请求的每个进程的锁请求计数。

```
hs_create_view process_lock,
"Login Name", "Value for Sample",
"Process ID", "Value for Sample",
"Kernel Process ID", "Value for Sample",
"Lock Count", "Value for Sample"
```

## 进程页 I/O

该视图表示针对最近一次采样在每个 Adaptive Server 进程级别汇总的页 I/O。它显示间隔期间 Adaptive Server 中生成页 I/O 的每个进程的登录名、进程 ID 和内核进程 ID。

在每个进程级别，该视图还显示以下项：页 I/O 总数；可以由 Adaptive Server 数据高速缓存满足的页 I/O 请求数所占的百分比；最近一次采样间隔期间发生的逻辑页读取数、物理页读取数和写入数。

```
hs_create_view process_page_io,  
"Login Name", "Value for Sample",  
"Process ID", "Value for Sample",  
"Kernel Process ID", "Value for Sample",  
"Page I/O", "Value for Sample",  
"Page Hit Percent", "Value for Sample",  
"Logical Page Reads", "Value for Sample",  
"Physical Page Reads", "Value for Sample",  
"Page Writes", "Value for Sample"
```

## 进程状态摘要

该视图显示最近一次采样间隔结束时，处于每种进程状态的进程数目。

```
hs_create_view process_perf_sum,  
"Process State", "Value for Sample",  
"Process State Count", "Value for Sample"
```

## 进程存储过程页 I/O

该视图表示与 Adaptive Server 进程执行的存储过程关联的页 I/O。它显示采样间隔期间，生成页 I/O 的每个进程的登录名、进程 ID 和内核进程 ID。它为生成页 I/O 的每个进程和存储过程，显示含有存储过程的数据库名和 ID 以及该过程本身的名称和 ID。

该视图还显示以下项：页 I/O 总数；可由数据高速缓存满足的页 I/O 请求数所占的百分比；最近一次采样间隔期间发生的逻辑页读取数、物理页读取数和页写入数。

```
hs_create_view process_procedure_page_io,  
"Login Name", "Value for Sample",  
"Process ID", "Value for Sample",  
"Kernel Process ID", "Value for Sample",  
"Procedure Database Name", "Value for Sample",  
"Procedure Database ID", "Value for Sample",  
"Procedure Name", "Value for Sample",  
"Procedure ID", "Value for Sample",  
"Page I/O", "Value for Sample",  
"Page Hit Percent", "Value for Sample",  
"Logical Page Reads", "Value for Sample",  
"Physical Page Reads", "Value for Sample",  
"Page Writes", "Value for Sample"
```

## 服务器性能摘要

该视图表示 Adaptive Server 的总体性能。它显示每秒提出的锁请求数目、Adaptive Server 处于繁忙状态的时间占采样间隔的百分比、每秒处理的事务数目，以及最近一次采样间隔期间 Adaptive Server 检测到死锁的次数。

```
hs_create_view server_perf_sum,  
"Lock Count", "Rate for Sample",  
"CPU Busy Percent", "Value for Sample",  
"Transactions", "Rate for Sample",  
"Deadlock Count", "Value for Sample"
```

## 存储过程活动

该视图显示过程语句级别的存储过程活动。对于最近一次采样间隔期间执行的任意存储过程的每条语句，都将由含有该过程的数据库的名称和 ID、该过程的名称和 ID、该语句在该过程中的相对编号以及作为该语句起始位置的过程文本行来标识。

该视图包括在最近一次采样间隔期间内以及自记录会话开始以来，每条语句执行的次数。它还包括在采样间隔期间以及该记录会话自开始到目前为止的这段时间，执行该语句平均需要的时间。

```
hs_create_view procedure_activity,  
"Procedure Database ID", "Value for Sample",  
"Procedure Database Name", "Value for Sample",  
"Procedure ID", "Value for Sample",  
"Procedure Name", "Value for Sample",  
"Procedure Line Number", "Value for Sample",  
"Procedure Statement Number", "Value for Sample",  
"Procedure Execution Count", "Value for Sample",  
"Procedure Execution Count", "Value for Session",  
"Procedure Elapsed Time", "Avg for Sample",  
"Procedure Elapsed Time", "Avg for Session"
```

## 事务活动

该视图详述了在采样间隔期间和记录会话持续期间，Adaptive Server 中发生的事务活动。

```
hs_create_view transaction_activity,
"Transactions", "Value for Sample",
"Rows Deleted", "Value for Sample",
"Rows Inserted", "Value for Sample",
"Rows Updated", "Value for Sample",
"Rows Updated Directly", "Value for Sample",
"Transactions", "Value for Session",
"Rows Deleted", "Value for Session",
"Rows Inserted", "Value for Session",
"Rows Updated", "Value for Session",
"Rows Updated Directly", "Value for Session",
"Transactions", "Rate for Sample",
"Rows Deleted", "Rate for Sample",
"Rows Inserted", "Rate for Sample",
"Rows Updated", "Rate for Sample",
"Rows Updated Directly", "Rate for Sample",
"Transactions", "Rate for Session",
"Rows Deleted", "Rate for Session",
"Rows Inserted", "Rate for Session",
"Rows Updated", "Rate for Session",
"Rows Updated Directly", "Rate for Session"
```



# 索引

## 符号

::= (BNF 表示法)  
    SQL 语句中的 xiv  
{ } (大括号)  
    SQL 语句中的 xiv  
, (逗号)  
    SQL 语句中的 xiv  
( ) (小括号)  
    SQL 语句中的 xiv  
[ ] (中括号)  
    SQL 语句中的 xiv  
\$DSDLISTEN 19  
\$SYBASE 10, 18, 30  
.bat 文件 27

## 英文

Adaptive Server 3, 29, 58  
Backus Naur Form (BNF) 表示法 xiii, xiv  
bcp 实用程序 4, 76  
    示例 84, 86  
charsets 目录 30  
cut 实用程序 86  
-D 参数 8, 18, 58, 76  
dscp 实用程序 11  
dsedit 实用程序 11, 14  
DSDLISTEN 环境变量 19  
Historical Server  
    Monitor Client Library 2  
    回放 2  
histserver 命令 10, 17, 29  
histsrvr 命令 17, 27, 33  
hs.ctl 文件。参见控制文件  
hs.log 文件。参见日志文件  
hs\_create\_alarm 命令 3, 43, 63  
hs\_create\_filter 命令 3, 47, 63

hs\_create\_playback\_session 命令 4, 50, 147, 149, 156  
hs\_create\_playback\_view 命令 4, 57, 147, 148  
hs\_create\_recording\_session 命令 3, 58, 63, 82  
hs\_create\_view 命令 3, 61, 63  
hs\_delete\_data 命令 62  
hs\_initiate\_playback 命令 4, 54, 55, 63  
hs\_initiate\_recording 命令 3, 42, 63  
hs\_list 命令 3, 8, 20, 55, 57, 64, 76  
hs\_playback\_sample 命令 4, 53, 68  
hs\_recording 命令 60  
hs\_shutdown 命令 31, 35, 71  
hs\_status 命令 20, 72  
hs\_terminate\_playback 命令 4, 42, 56, 73  
hs\_terminate\_recording 命令 42, 62, 74  
-i 参数 18  
interfaces 文件 9, 11, 12, 18  
-l 参数 18, 30  
libctl.cfg 文件 14  
locales 目录 30  
Monitor Client Library 2  
    Historical Server 2  
Monitor Server 1, 29  
    Adaptive Server 2  
    连接到 3  
Monitor Server 名称 58  
Monitor Viewer 1, 2, 3  
monserver 命令 10  
-n 参数 19  
no\_wait  
    请参见 hs\_shutdown 命令  
no\_wait。请参见 hs\_shutdown 命令  
ocsefg 实用程序 14  
Open Server 1  
-P 参数 19, 30, 34  
regedt32.exe 21, 24  
-S 参数 19, 40  
shutdown 71  
SIGKILL 信号 71  
SQL 语句中的 BNF 表示法 xiii, xiv  
SQL 语句中的大括号 ({} ) xiv

- sql.ini 文件 9, 13
- Sybase Central 2, 3
- SYBASE 环境变量 10, 18, 30
- U 参数 9, 19, 30, 34
- UNIX
  - 配置 Historical Server 9
  - 配置第二台服务器 21
  - 启动 Historical Server 29
  - 停止 30
- v 参数 19
- Windows
  - 检验 34
  - 配置 Historical Server 12
  - 配置第二台服务器 21
  - 确定启动参数 33
  - 停止 34
  - 注册表 21
- Windows 注册表 24
  - 启动参数 33

## A

- 安装
  - 目录 10
  - 说明 10, 13

## B

- 版本
  - Historical Server 19, 72
- 报警
  - 定义 43
  - 列表定义 64
  - 日志文件条目 45
- 报警控制记录, 控制文件中 80
- 编辑控制文件 8, 77
- 标头记录 77
- 表, 创建 60, 82
- 表列名, 在脚本文件中 83
- 不活动的记录会话 65

## C

- 采样间隔 3, 20, 58
  - 回放中的用户定义 145
- 参数
  - D 8, 18, 58, 76
  - i 18
  - l 18, 30
  - n 19
  - P 19, 30, 34
  - S 19, 40
  - U 9, 19, 30, 34
  - v 19
  - 启动 17, 19
- 超级用户 9, 19, 30, 34, 41, 72
- 初始化回放会话 63
- 创建
  - 报警 43
  - 表 54, 60, 82
  - 过滤器 47
  - 回放会话 42
  - 回放视图 57, 143
  - 记录会话 3, 42
  - 记录会话视图 61, 109
- 从回放输出 56
- 错误
  - 文件 81
  - 消息 39
  - 用于记录会话的选项 60
  - 状态码 39
- 错误。参见日志文件

## D

- 定义。参见创建
- 逗号 (,)
  - SQL 语句中的 xiv
- 对控制文件的争用 20
- 多个实例 20
  - Windows 上 21
  - 在 UNIX 上 21



**F**

符号

SQL 语句中的 xiv

服务列表 24

服务器级视图 112

服务器配置实用程序 21

**G**

共享内存 2

估计, 在回放视图中 53, 149

关闭

权限 9

管理 20

过程 CPU 时间 156

过程占用时间 156

过滤器 47

列表定义 64

过滤器控制记录 80

过滤器值说明 47, 48

**H**

互斥会话 42, 56

环境变量

DSLISTEN 19

SYBASE 10, 18, 30

回放 2

间隔 52, 146

数据间隙 56

回放会话

采样间隔 145

创建 42

到客户端 68

互斥 42

简介 4

结果 56

结束时间 70

开始时间 70

列表汇总 64

启动 63

数据间隙 70

文件权限 54

终止 73

回放会话的间隔 52, 146

回放会话目标 53, 56

回放结果 56

回放视图 5

创建 57, 143

估计 149

汇总级别 143

计算 149

名称 57

要求 150

有效的统计类型 150

回放数据间隙 56, 70

会话 ID 3

会话, 互斥 56

会话控制记录 78

汇总级别 64, 143

活动

无 111

显示当前 73

活动记录会话 65

**J**

计算, 在回放视图中 149

记录功能 3

记录会话

并发 19

创建 3, 42

过去的 64

互斥 42

检查数据于 4

简介 3

结束时间 59

开始时间 58

列表定义 64

期间错误 60

启动 63

终止 62, 74

状态 65

- 记录会话视图
  - 创建 61
  - 名称 61
  - 示例 157
  - 要求 112
- 检验启动 34
- 键数据项
  - 定义 109
- 脚本文件 7
  - 报警与 44
  - 启动 10
  - 用于创建表 54, 60
  - 在 Adaptive Server 中创建表 82
- 结果数据项, 定义 109
- 结束会话 42, 73, 74
- 结束时间
  - 回放会话 51, 70
  - 记录会话的 59
- 进程 ID 155

## K

- 开始时间
  - 回放会话 50, 70, 144, 145
  - 记录会话的 58
- 客户端回放 53, 68
- 客户端连接。参见连接
- 空行 111
- 控制文件 3, 8, 18, 20
  - 编辑 8, 77
  - 标头记录 77
  - 格式 76
  - 过滤器控制记录 80
  - 会话控制记录 78
  - 启动帐户 8
  - 权限 8
  - 视图控制记录 79
  - 数据项控制记录 79
- 争用 20
- 口令
  - 提示 19
  - 在启动命令中 19

## L

- 连接
  - 到 Historical Server 40
  - 到 Monitor Server 3, 40, 58
  - 定义 42
  - 和 interfaces 文件 9
  - 和 sql.ini 文件 9
  - 和目录服务 9
  - 互斥会话 42
  - 权限 41
  - 显示当前 73
  - 最大数目 19, 72
- 连接的最大数目 19, 72
- 列表
  - 报警 64
  - 过滤器 64
  - 汇总级别 64
  - 记录会话定义 64
  - 视图 64

## M

- 命令
  - 另请参见实用程序
  - histserver 10, 17, 29
  - histserver 17, 27, 33
  - hs\_create\_alarm 3, 43, 63
  - hs\_create\_filter 3, 47, 63
  - hs\_create\_playback\_session 4, 50, 147, 149, 156
  - hs\_create\_playback\_view 4, 57, 147, 148
  - hs\_create\_recording\_session 3, 58, 63, 82
  - hs\_create\_view 3, 61, 63
  - hs\_delete\_data 62
  - hs\_initiate\_playback 4, 54, 55, 63
  - hs\_initiate\_recording 3, 42, 60, 63
  - hs\_list 3, 8, 20, 55, 57, 76
  - hs\_playback\_sample 4, 53, 68
  - hs\_shutdown 31, 35, 71
  - hs\_status 20, 72
  - hs\_terminate\_playback 4, 42, 56, 73
  - hs\_terminate\_recording 42, 62, 74
  - monserver 10
  - 语法 38
  - 摘要 37

## 目录

- 另请参见主目录
- locales 和 charsets 30
- 安装 10
- 数据文件 58, 76

目录服务 9

## P

排它会话 56

## 配置

- Historical Server 启动参数 17, 19
- UNIX 上的 Historical Server 9
- Windows 上的 Historical Server 12
- Windows 上的第二个 Historical Server 21
- 服务列表 24
- 在 UNIX 上的第二台 Historical Server 21
- 批量复制实用程序。参见 bcp 实用程序
- 平均, 统计类型
  - 定义 6

## Q

## 启动

- .bat 文件 27
- Windows 上 33
- 参数 17, 19
- 回放会话 63
- 脚本文件 10
- 在 UNIX 上 29
- 帐户 8, 20, 29

## 启动参数

- Windows 注册表中 33

嵌入的空格 38

## 区分大小写

- 在 SQL 中 xv

## 权限

- 超级用户 41
- 对于安装目录 10
- 控制文件 8
- 数据文件 8, 9, 20, 76
- 用于关闭 9

在回放文件上 54

在记录会话文件中 59

摘要 41

确定启动参数 33

## R

日志文件 7, 18, 30

报警与 44, 45

名称 72

位置 30

## S

删除数据文件 55, 62

时间戳 20, 155

时区 50, 58

实际回放 51, 144

实例。参见多个实例

## 实用程序

- bcp 76
- cut 86
- dscp 11
- dsedit 11, 14
- ocscfg 14
- 服务器配置 21

## 视图

另请参见记录会话视图

服务器级 112

回放 5

回放的要求 150

简介 5, 109

列表定义 64

视图控制记录 79

视图示例 157

输入会话。请参见记录会话

输入视图 147

数据文件 7, 75

格式 81

检查内容 4, 38, 75

权限 8, 9, 20, 59, 76

删除 55, 62

## 索引

- 所有者 76
- 位置 58, 76
- 数据文件的所有者 76
- 数据项
  - 定义 93
  - 过程 CPU 时间 156
  - 过程占用时间 156
  - 回放的要求 150
  - 记录会话的要求 112
  - 进程 ID 155
  - 时间戳 155
  - 有效统计类型 136
- 数据项控制记录 79

## T

- 添加
  - Windows 上的第二个 Historical Server 21
  - 到 .bat 文件 27
  - 到 Windows 注册表 21
  - 到服务列表 24
  - 在 UNIX 上的第二台 Historical Server 21
- 停止 71
  - Windows 上 34
  - 回放会话 73
  - 在 UNIX 上 30
- 统计类型 5
  - 回放视图中 150
  - 每个数据项 135
- 图形用户界面 2

## W

- 完整回放 51, 145
- 文件
  - 另请参见控制文件
  - .bat 27
  - interfaces 9, 11, 12, 18
  - libctl.cfg 14
  - sql.ini 9, 13
  - 数据 75
  - 作为回放目标 53

## X

- 行, 空 111
- 小括号 ()
  - SQL 语句中的 xiv

## Y

- 引号字符 38
- 应用程序编程接口 2
- 用户定义的采样间隔 145
- 用户名 19
- 语法定义, Transact-SQL xiv
- 原始回放 51, 143, 148
- 约定
  - 另请参见语法
  - Transact-SQL 语法 xiv
  - 在参考手册中使用 xiii

## Z

- 帐户
  - 报警与 44
  - 超级用户 9, 19
  - 启动 8, 20, 29
- 中括号 []
  - SQL 语句中的 xiv
- 中括号。请参见中括号 []
- 主目录 3, 8, 18
- 启动帐户 8
- 状态
  - Historical Server 20, 72
  - 会话的 65
  - 记录会话的 65
- 状态码 39