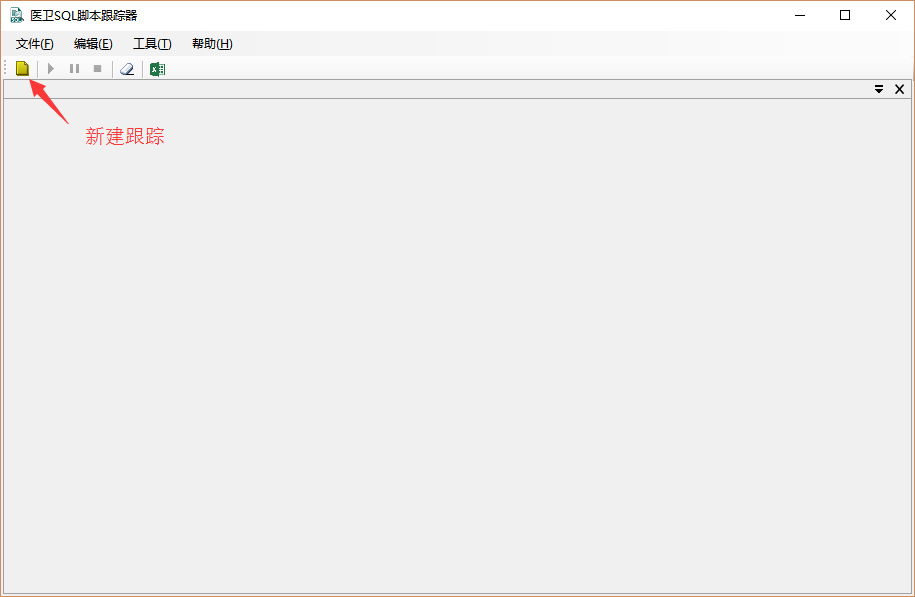
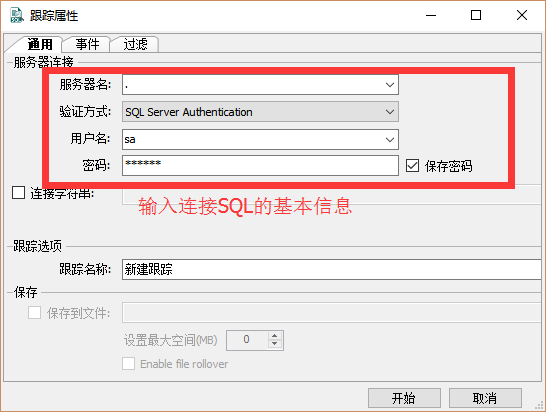
**医卫SQL跟踪器使用说明**

## 操作步骤

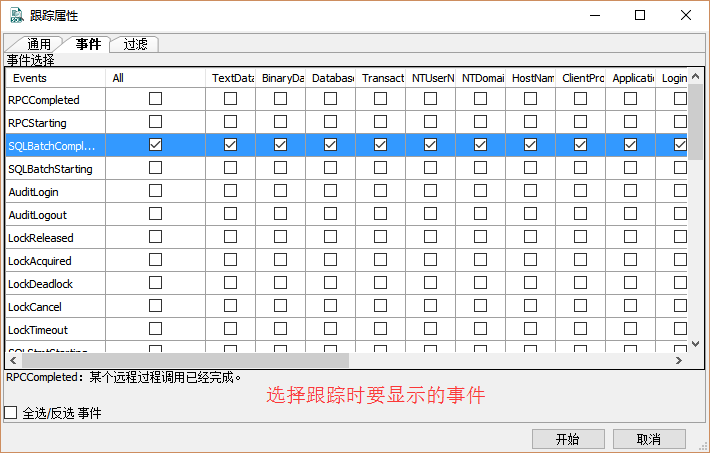
1. 新建跟踪



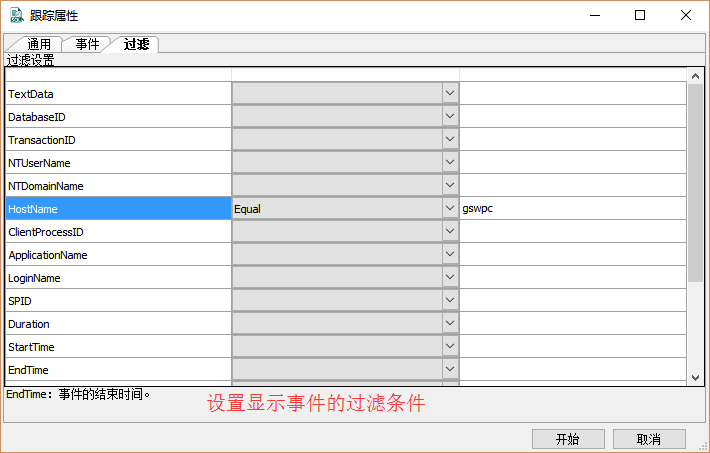
1. 连接SQL Server



1. 选择事件



1. 选择过滤条件



1. 开始执行即可。

## 注意事项

1. SQL跟踪事件

|  |  |
| --- | --- |
| RPCCompleted | 某个远程过程调用已经完成。 |
| RPCStarting | 远程过程调用已启动。 |
| SQLBatchCompleted | Transact-SQL 批处理已完成。 |
| SQLBatchStarting | 正在启动 Transact-SQL 批处理。 |
| AuditLogin | 用户已成功登录到 MicrosoftSQL Server。 此类中的事件由新连接或从连接池中重用的连接触发。 |
| AuditLogout | 用户已注销 MicrosoftSQL Server。 此类中的事件由新连接或从连接池中重用的连接触发。 |
| LockReleased | 已释放某个资源（例如页）的锁。  Lock:Acquired 和 Lock:Released 事件类可以用于监视锁定对象的时间、使用的锁类型以及持有锁的时间。 保留较长时间的锁可能导致争用问题，应进行调查。 例如，应用程序可以为表中的行获取锁，然后等待用户输入。 由于用户输入可能需要较长时间，所以锁可能会阻塞其他用户。 在这种情况下，应重新设计应用程序，以便只在需要时才请求锁，并且在获取锁后不需要用户输入。 |
| LockAcquired | 已获取某个资源（如数据页）的锁。  Lock:Acquired 和 Lock:Released 事件类可以用于监视锁定对象的时间、使用的锁类型以及持有锁的时间。 保留较长时间的锁可能导致争用问题，应进行调查。 例如，应用程序可以为表中的行获取锁，然后等待用户输入。 由于用户输入可能需要较长时间，所以锁可能会阻塞其他用户。 在这种情况下，应重新设计应用程序，以便只在需要时才请求锁，并且在获取锁后不需要用户输入。 |
| LockDeadlock | 对于死锁中的每个参与者都会产生 Lock:Deadlock Chain 事件类。  使用 Lock:Deadlock Chain 事件类可以监视何时出现死锁情况。 此信息有助于确定死锁是否会对应用程序的性能造成重大影响，以及会涉及哪些对象。 可以检查用于修改这些对象的应用程序代码，以便确定是否可以做出更改以便将死锁的情况减到最少。 |
| LockCancel | 已取消获取某个资源的锁；例如，由于查询被取消。 |
| LockTimeout | Lock:Timeout 事件类指示由于其他事务持有所需资源的阻塞锁而使对资源（例如页）锁的请求超时。  超时由 @@LOCK\_TIMEOUT 系统函数决定，可用 SET LOCK\_TIMEOUT 语句设置。 超时情况出现时，使用 Lock:Timeout 事件类进行监视。 此信息有助于确定超时是否对应用程序的性能造成重大影响，以及涉及哪些对象。 您可以检查修改这些对象的应用程序代码，以确定是否可以进行更改以将超时减到最小。持续时间为 0 的 Lock:Timeout 事件通常是内部锁探测的结果，并不表示存在问题。 可以使用 Lock:Timeout (timeout > 0) 事件以忽略持续时间为 0 的超时。 |
| SQLStmtStarting | Transact-SQL 语句已开始执行。 |
| SQLStmtCompleted | Transact-SQL 语句已完成。 |
| SPStarting | 存储过程将要开始执行。 |
| SPCompleted | 存储过程已执行完毕。 |
| SPStmtStarting | 已开始执行存储过程中的 Transact-SQL 语句。 |
| SPStmtCompleted | 存储过程中的 Transact-SQL 语句已执行完毕。 |
| SQLTransaction | 可以监视事务开始和完成的时间，尤其是当您测试应用程序、触发器或存储过程时。 |
| ScanStarted | 开始扫描表或索引时，会发生 Scan:Started 事件类。 |
| ScanStopped | 停止扫描表或索引时，会发生 Scan:Stopped 事件类。 |
| CursorOpen | 说明在应用程序编程接口 (API) 游标中发生的游标打开事件。  SQL Server 数据库引擎定义与游标和游标选项相关联的 SQL 语句时发生游标打开事件，然后填充游标。   在记录游标性能的跟踪中包括 CursorOpen 事件类。 在跟踪中包括 CursorOpen 事件类时，产生的开销取决于跟踪期间数据库中使用游标的频率。 如果广泛使用游标，则跟踪可能会显著地降低性能。 |
| TransactionLog | 可以监视 SQL Server 数据库引擎实例的事务日志中的活动。 |
| LockDeadlockChain | 对于死锁中的每个参与者都会产生 Lock:Deadlock Chain 事件类。   使用 Lock:Deadlock Chain 事件类可以监视何时出现死锁情况。 此信息有助于确定死锁是否会对应用程序的性能造成重大影响，以及会涉及哪些对象。 可以检查用于修改这些对象的应用程序代码，以便确定是否可以做出更改以便将死锁的情况减到最少。 |
| LockEscalation | 较细粒度的锁已转换为较粗粒度的锁；例如，行锁已转换为对象锁。 升级事件类是事件 ID 为 60 的事件类。 |
| ExecutionWarnings | 在执行 SQL Server 语句或存储过程期间出现的内存授予警告。   监视此事件类可确定查询在继续进行之前是否必须等待一秒或几秒再获取内存，或确定获取内存的初始尝试是否失败。 查询等待时间的信息有助于揭示影响性能的系统争用问题。 |
| SQLFullTextQuery | 当 SQL Server 执行全文查询时，会发生 SQL:FullTextQuery 事件类。  请将此事件类包括在监视与全文目录相关的问题的跟踪中。  当包含 SQL:FullTextQuery 事件类时，开销量将很高。 如果此类事件频繁发生，则跟踪可能会显著地降低性能。 为了尽量减小这种影响，此事件类的使用应限于短期监视特定问题的跟踪。 |
| DeadlockGraph | 有关死锁的 XML 描述。 该类与 Lock:Deadlock 事件类同时发生。 |
| SQLStmtRecompile | 由下列所有类型的批处理引起的语句级重新编译：存储过程、触发器、即席批查询和查询。  可以通过使用 sp\_executesql、动态 SQL、“准备”方法、“执行”方法或类似接口来提交查询。 应该用 SP:Recompile 事件类来代替 SQL:StmtRecompile 事件类。 |
| TMBeginTranStarting | 正在启动 BEGIN TRANSACTION 请求。 将通过事务管理接口从客户端发送请求。 |
| TMBeginTranCompleted | 已完成 BEGIN TRANSACTION 请求。 该请求是通过事务管理界面从客户端发送的。 |
| TMPromoteTranStarting | 正在启动 PROMOTE TRANSACTION 请求。 将通过事务管理接口从客户端发送请求。 |
| TMPromoteTranCompleted | PROMOTE TRANSACTION 请求已完成。 将通过事务管理接口从客户端发送请求。 |
| TMCommitTranStarting | 正在启动 COMMIT TRANSACTION 请求。 将通过事务管理接口从客户端发送请求。EventSubClass 列指示在提交当前事务之后是否启动新事务。 |
| TMCommitTranCompleted | COMMIT TRANSACTION 请求已完成。 该请求是通过事务管理界面从客户端发送的。EventSubClass 列指示在提交当前事务之后是否启动新事务。 |
| TMRollbackTranStarting | 正在启动 ROLLBACK TRANSACTION 请求。 客户端通过事务管理界面发送请求。EventSubClass 列指示在当前事务回滚后是否启动新事务。 |
| TMRollbackTranCompleted | ROLLBACK TRANSACTION 请求已完成。 该请求是通过事务管理界面从客户端发送的。EventSubClass 列指示在当前事务回滚后是否启动新事务。 |
| TMSaveTranstarting | 正在启动 SAVE TRANSACTION 请求。 该请求是从客户端通过事务管理界面发送的。 |
| TMSaveTrancompleted | SAVE TRANSACTION 请求已完成。 该请求是通过事务管理界面从客户端发送的。 |

1. 过滤字段说明

|  |  |
| --- | --- |
| TextData | 依赖于跟踪中捕获的事件类的文本值。 但是，如果跟踪参数化查询，则不以 **TextData** 列中的数据值显示变量。 |
| BinaryData | 依赖于跟踪中捕获的事件类的二进制值 |
| DatabaseID | 由 USE database\_name 语句指定的数据库的 ID；如果未对给定实例发出 USE database\_name语句，则为默认数据库的 ID。 如果在跟踪中捕获 **ServerName** 数据列而且服务器可用，则 SQL Server Profiler 将显示数据库名。 若要确定数据库的值，请使用 DB\_ID 函数。 |
| TransactionID | 系统为事务分配的 ID。 |
| NTUserName | Windows 用户名。 |
| NTDomainName | 用户所属的 Microsoft Windows 域。 |
| HostName | 正在运行客户端程序的计算机的名称。 如果客户端提供了主机名，则填充此数据列。 若要确定主机名，请使用 HOST\_NAME 函数。 |
| ClientProcessID | 由主机分配给正在运行客户端应用程序的进程的 ID。 如果客户端提供了客户端进程 ID，则填充此数据列。 |
| ApplicationName | 客户端应用程序的名称，该客户端应用程序创建了指向 SQL Server 实例的连接。 此列由该应用程序传递的值填充，而不是由程序名填充的。 |
| LoginName | 用户的登录名（SQL Server 安全登录名或 Windows 登录凭据，格式为“域/用户名”）。 |
| SPID | SQL Server 为客户端的相关进程分配的服务器进程 ID (SPID)。 |
| Duration | 事件的持续时间（微秒）。 |
| StartTime | 事件（如果有）的开始时间。 |
| EndTime | 事件的结束时间。 对指示事件开始的事件类（例如 **SQL:BatchStarting** 或**SP:Starting**）将不填充此列。 |
| Reads | 由服务器代表事件读取逻辑磁盘的次数。 这些读取操作数包含在语句执行期间读取表和缓冲区的次数。 |
| Writes | 由服务器代表事件写入物理磁盘的次数。 |
| CPU | 事件使用的 CPU 时间（毫秒）。 |
| ObjectID | 系统分配的对象 ID。 |
| ServerName | 正在跟踪的 SQL Server 实例的名称。 |
| EventClass | 捕获的事件类的类型。 |
| ProviderName | OLEDB 访问接口的名称。 |
| LoginSID | 已登录的用户的安全标识符 (SID)。 您可以在 **master** 数据库的 **sys.server\_principals** 视图中找到此信息。 服务器中的每个登录名都具有唯一的 ID。 |
| DataBaseName | 正在运行用户语句的数据库的名称。 |

|  |  |
| --- | --- |
| RPCCompleted | 某个远程过程调用已经完成。 |
| RPCStarting | 远程过程调用已启动。 |
| SQLBatchCompleted | Transact-SQL 批处理已完成。 |
| SQLBatchStarting | 正在启动 Transact-SQL 批处理。 |
| AuditLogin | 用户已成功登录到 MicrosoftSQL Server。 此类中的事件由新连接或从连接池中重用的连接触发。 |
| AuditLogout | 用户已注销 MicrosoftSQL Server。 此类中的事件由新连接或从连接池中重用的连接触发。 |
| LockReleased | 已释放某个资源（例如页）的锁。 |
| LockAcquired | 已获取某个资源（如数据页）的锁。 |
| LockDeadlock | 对于死锁中的每个参与者都会产生 Lock:Deadlock Chain 事件类。 |
| LockCancel | 已取消获取某个资源的锁；例如，由于查询被取消。 |
| LockTimeout | Lock:Timeout 事件类指示由于其他事务持有所需资源的阻塞锁而使对资源（例如页）锁的请求超时。 |
| SQLStmtStarting | Transact-SQL 语句已开始执行。 |
| SQLStmtCompleted | Transact-SQL 语句已完成。 |
| SPStarting | 存储过程将要开始执行。 |
| SPCompleted | 存储过程已执行完毕。 |
| SPStmtStarting | 已开始执行存储过程中的 Transact-SQL 语句。 |
| SPStmtCompleted | 存储过程中的 Transact-SQL 语句已执行完毕。 |
| SQLTransaction | 可以监视事务开始和完成的时间，尤其是当您测试应用程序、触发器或存储过程时。 |
| ScanStarted | 开始扫描表或索引时，会发生 Scan:Started 事件类。 |
| ScanStopped | 停止扫描表或索引时，会发生 Scan:Stopped 事件类。 |
| CursorOpen | 说明在应用程序编程接口 (API) 游标中发生的游标打开事件。 |
| TransactionLog | 可以监视 SQL Server 数据库引擎实例的事务日志中的活动。 |
| LockDeadlockChain | 对于死锁中的每个参与者都会产生 Lock:Deadlock Chain 事件类。 |
| LockEscalation | 较细粒度的锁已转换为较粗粒度的锁；例如，行锁已转换为对象锁。 |
| ExecutionWarnings | 在执行 SQL Server 语句或存储过程期间出现的内存授予警告。 |
| SQLFullTextQuery | 当 SQL Server 执行全文查询时，会发生 SQL:FullTextQuery 事件类。 |
| DeadlockGraph | 有关死锁的 XML 描述。 该类与 Lock:Deadlock 事件类同时发生。 |
| SQLStmtRecompile | 由下列所有类型的批处理引起的语句级重新编译：存储过程、触发器、即席批查询和查询。 |
| TMBeginTranStarting | 正在启动 BEGIN TRANSACTION 请求。 将通过事务管理接口从客户端发送请求。 |
| TMBeginTranCompleted | 已完成 BEGIN TRANSACTION 请求。 该请求是通过事务管理界面从客户端发送的。 |
| TMPromoteTranStarting | 正在启动 PROMOTE TRANSACTION 请求。 将通过事务管理接口从客户端发送请求。 |
| TMPromoteTranCompleted | PROMOTE TRANSACTION 请求已完成。 将通过事务管理接口从客户端发送请求。 |
| TMCommitTranStarting | 正在启动 COMMIT TRANSACTION 请求。 将通过事务管理接口从客户端发送请求。EventSubClass 列指示在提交当前事务之后是否启动新事务。 |
| TMCommitTranCompleted | COMMIT TRANSACTION 请求已完成。 该请求是通过事务管理界面从客户端发送的。EventSubClass 列指示在提交当前事务之后是否启动新事务。 |
| TMRollbackTranStarting | 正在启动 ROLLBACK TRANSACTION 请求。 客户端通过事务管理界面发送请求。EventSubClass 列指示在当前事务回滚后是否启动新事务。 |
| TMRollbackTranCompleted | ROLLBACK TRANSACTION 请求已完成。 该请求是通过事务管理界面从客户端发送的。EventSubClass 列指示在当前事务回滚后是否启动新事务。 |
| TMSaveTranstarting | 正在启动 SAVE TRANSACTION 请求。 该请求是从客户端通过事务管理界面发送的。 |
| TMSaveTrancompleted | SAVE TRANSACTION 请求已完成。 该请求是通过事务管理界面从客户端发送的。 |