

12.6 SQL Server 2005 所提供的主动通知 Microsoft 认证考试 PDF 转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/640/2021\\_2022\\_126SQLSer\\_c100\\_640530.htm](https://www.100test.com/kao_ti2020/640/2021_2022_126SQLSer_c100_640530.htm)

在数据库内存放的数据的使用模式有许多是变动少而查询多的。据统计，数据库中一般有 20% 以上的数据属于此类型，例如公司的产品、人事数据，以及用作参照的数据表（lookup table），例如提供地址输入的可选择的国家、城市，人事部门、职级等等。这些数据可能都是在数据输入时，供人点选而以代号填入某个数据表字段中，其内容自身较少更改。在多人同时访问的环境中，我们一般会设计高速缓存的机制，将上述不常更改的数据临时存放在应用程序服务器或是用户的机器上，避免频繁地往返访问数据库。但为了要提供数据库内他人更新后的状况，仍必须轮询（Poll）数据库，或是在数据表上编写触发器（trigger），以及可能通过消息队列（Message queue）来通知等等。不管方式为何，大多需要设计者花些巧思来同步应用程序高速缓存与数据库中更新过的记录。SQL Server 2005 搭配 ADO.NET 2.0 的 SqlClient 后，内置由 SQL Server 通过 Service Broker 服务通知前端应用程序它有兴趣的相关数据已经更改，该机制称为 Query Notification。根据使用到的功能不同，完成通知的整体架构参与组件可能有 SQL Server 2005 Query Engine、Broker Service、系统存储过程 sp\_DispatcherProc、ADO.NET 的 SqlNotification 类（System.Data.Sql.SqlNotificationRequest）、SqlDependency 类（System.Data.SqlClient.SqlDependency）以及 ASP.NET Cache 类（System.Web.Caching.Cache）等等。而运行的逻辑简单说明如下：SqlCommand 有 Notification 属性

，用来存放 Notification 相关的设置。当 SqlCommand 执行时，会让传递该执行需求的 TDS 协议附加上 Notification 的信息。SQL Server 收到该需求后，为这个需求注册 Notification，并执行该需求自身的 T-SQL 语法。SQL Server 会监控后续执行的 DML 语法，看看是否会影响前一步返回给前端的数据集合（rowset），一旦有影响到，则送一个消息到 Service Broker。Service Broker 的队列中有消息后，可能发生的状况如下：

- Notification 在前端应用程序侦听的队列中放入消息
- ADO.NET 的下层自动读取消息并触发事件。在 Service Broker 内的消息持续保留着，较高级的前端应用程序自己来处理这个消息。

上述步骤的示意图如图12-8所示：图12-8 SQL Server 2005 提供的 Notification 机制以往与 SQL Server 沟通时，都是传递 T-SQL 的批处理语法。而在 SQL Server 2005 客户端应用程序除了依旧传递这些 T-SQL 语法外，还可以加上三种消息：用来发送通知的 Service Broker Services 的名称、以字符串表示的 notification identifier、通知的超时间隔。一旦带有这三种消息，SQL Server 2005 则会注意该语法批次所创建并返回的数据集合（rowset），若有一个以上的数据集合，SQL Server 也会全部都注意，当有其他语法会影响到这些数据集合时就发出通知。在 SQL Server 2000 的 Indexed View 中其实就已经实现了监控视图表（View）所涵盖的数据表内相关的记录改变，并自动更新 Indexed View 的实例数据。而 SQL Server 2005 也采用相同的引擎来监控数据集合。但并不是所有视图表所定义的查询语法都可以建索引，同样地，若你传递到 SQL Server 的查询语法无法创建通知，将会得到如 Invalid Query 一类的报错信息。因为 SQL Server 2005 的通知机制实际

是搭配该版所新增的 Services Broker，所以要发出通知的数据库必须让 Services Broker 启动（enable）。Services Broker 利用 SQL Server 2005 所提供的队列（Queue）创建异步通知。而通知机制就是一组 Services Broker 内置好的服务（Service），也就是有标准的消息（Message）、发送的队列、发送消息的规则（Contract）等等。你可以通过“SQL Server Management Studio”的“对象资源管理器”窗口查看每个数据库节点底下“Services Broker”节点内相关的设置，如图12-9所示：

图12-9 SQL Server 2005 在 Service Broker已经准备好查询通知的相关设置由于我们是通过 SQL Server 2005 的范例数据库 AdventureWorks 来测试，因此若要让应用程序可以收到通知，须先启动该数据库的 Service Broker Services，同时要允许登录的账号订阅查询通知。而需要执行的 T-SQL 设置如程序代码列表12-8所示：程序代码列表12-8 设置 AdventureWorks 数据库启动 Service Broker，并允许账号 Sandy 订阅查询通知--启动 Service Broker，停止则是设置 DISABLE\_BROKERALTER DATABASE AdventureWorks SET ENABLE\_BROKER--并无法在 sp\_dboption 看出某个 DB 是否启动了 BROKER--EXEC sp\_dboption AdventureWorks--需要查看sys.databases 的 is\_broker\_enabled 字段才知道是否启动SELECT \* FROM sys.databases--允许某个账号订阅查询GRANT SUBSCRIBE QUERY NOTIFICATIONS TO Sandy我们分别提供 Windows Forms 以及 Web Forms 两种应用程序范例。首先是 Windows Forms，执行画面如图12-10所示：图12-10 修改数据后，通过 SQL Server 的通知来更新已经查询的现有数据 100Test 下载频道开通，各类考试题目直接下载。详细请访问

