

SQLServer分布式分区视图分解数据表 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/460/2021_2022_SQLServer_E5_c98_460947.htm 分布式分区视图可以将来自一个或多个SQL Server数据库中的数据连接起来。当开发一个水平分区数据库环境时，你可以使用分布式分区视图将来自不同服务器的分区表连接起来，使得这些数据看起来像来自同一个服务器。你可以设计这些视图，因此，如果你的潜在数据表结构设计合理的话，查询优化器就可以知道从那个数据表得到查询需要的数据，从而加速运行。一个设计合理的分布式分区视图还可以实现更新、插入和删除。我们将在本文的下一部分深入探讨它是如何实现这样操作的。示例 本例中，我们假设SalesHistory表非常大，如果水平分割表中的各行记录到不同的服务器上，这将对我们很有利。每个服务器上的SalesHistory表的表结构是一样的，不过，一台服务器上存放该国东部地区的销售信息，而另外一台存放该国西部地区的销售信息。我们根据Region(地区)字段和SaleID 来区分表中的各条记录。其中SaleID字段是整型数据域，我们为该国不同的地区设定了不同的SaleID。这个字段对于设计概念来说非常重要，因为这是我们用来作为分区键值字段。(注意：要在缩小场景中进行表的设计，这一点极其重要，因为这样表中的各行是唯一的，从而可区别于其它服务器上的表。)这个字段集合是分区键。设计很多SaleHistory表，根据所在的表SaleID始终是唯一可区别的。我们可以通过CHECK约束来实现这一点。我们将使用两个独立的SQL Server实例，对于本例，这两个实例在同一台机器上。服务器的名字叫Chapman

，实例分别称为实例A和实例B。这两个实例都是SQL Server 2005开发版，允许远程连接以及Windows和SQL Server认证。使用脚本创建SalesDB数据库，设置每台服务器的lazy schema validation选项，使用该选项在SQL Server中通过确保在确实需要服务器上的数据时才进行服务器链接请求来提高性能。列表A中的脚本需要在两个数据库实例上运行。列表B用来创建SalesDB数据库中的读者登录及用户，该脚本也需要在两个数据库实例上运行。列表A: CREATE DATABASE SalesDB. EXECUTE sp_serveroption @server = CHAPMANServerA, @optname = lazy schema validation, @optvalue = true 列表B: CREATE LOGIN reader WITH Password = 654asod3e**!! USE SalesDB CREATE USER reader FROM LOGIN reader 列表C: ServerA: EXECUTE sp_addlinkedserver CHAPMANServerB, SQL Server EXEC sp_addlinkedsrvlogin CHAPMANServerB, false, NULL, reader, 654asod3e**!! ServerB: EXEC sp_addlinkedserver CHAPMANServerA, SQL Server EXEC sp_addlinkedsrvlogin CHAPMANServerA, false, NULL, reader, 654asod3e**!! 列表C将独自在每台数据库实例上运行。该脚本在每台服务器上创建一个链接到对方的链接服务器。连接服务器允许SQL Server依靠某个OLEDB数据源来执行命令，就像其它的SQL Server。我们使用上面的脚本中创建的登录作为连接服务器进行连接的安全上下文。这些链接服务器允许我们从一台服务器查询另一台服务器。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com