# C:\Users\stefano.maggi.CONBIPELSPA\Desktop\prom38.png

**Date**: 16/05/2012

**Procedure:** SQL Server – Setting Up an Oracle Linked Server

**Source:** [**LINK**](http://www.sqlmag.com/article/sql-server/setting-up-an-oracle-linked-server49687)

**Permalink:** [**LINK**](http://heelpbook.altervista.org/2012/sql-server-setting-up-an-oracle-linked-server/)

**Created by:** HeelpBook Staff

**Document Version:** 1.0

# [**SQL Server – Setting Up an Oracle Linked Server**](http://heelpbook.altervista.org/2012/sql-server-setting-up-an-oracle-linked-server/)

SQL Server Linked Servers feature lets you access **Oracle** data and data from other **OLE DB/ODBC** compatible data sources from **SQL Server**. Here are the basic steps for setting up an **Oracle** linked server.

**1. Install and Configure the Oracle Client Software**

Oracle client software provides the network libraries required to establish connectivity to an Oracle database system.Download the software from <http://www.oracle.com/technology/software/products/database/oracle10g/index.html>.

Install the software on your **SQL Server** system and configure it by using **Oracle Net Configuration Assistant**.

**2. Create the Linked Server**

Create a linked server by using the **T-SQL** command:

EXEC sp\_addlinkedserver

 'OracleLinkedServer', 'Oracle',

 'MSDAORA', 'OracleServer'

The name of the linked server is **Oracle-LinkedServer**.The second parameter, product name (**Oracle**),is optional.The third parameter specifies the **OLE DB** provider. **MSDAORA** is the name of the *Microsoft OLE DB Provider for Oracle*.

The final required parameter is the data source name, **Oracle Server**.

**3. Add Logins for the Linked Server**

Next, provide the SQL Server system with an Oracle login to access the **Oracle** database by using the sp\_addlinkedsrvlogin command:

EXEC sp\_addlinkedsrvlogin '

 OracleLinkedServer ', false,

 'SQLuser', 'OracleUser',

 'OraclePwd'

The first parameter, **Oracle Linked Server**, specifies the name of the linked server system that you created.The second parameter determines the name of the login to be used on the remote system. A value of **True** indicates that the current **SQL Server** login will be used to connect to the linked server.

This requires that the logins on the two database servers match, which is typically not the case. A value of **False** means you'll supply the remote login.

The third parameter specifiesthe name of a **SQL Server** login that this remote login will map to. A value of **NULL** indicates that this remote login will be used for all connections to the linked Oracle server.

If the **Oracle** system uses Windows authentication, you can use the keyword *domain\* to specify a **Windows** login.

The fourth and fifth parameters supply login and password values for the **Oracle** system.

**4. Query the Linked Server**

To test the connection, run a sample query using the linked server name. Linked servers support updates as well as queries. To access the tables on a linked server, use a four-part naming syntax: *linked\_server\_name.catalog\_ name.schema\_name.table\_name.*

For example, to query the sample **Oracle Scott** database, you'd enter the statement:

SELECT \* FROM

 OracleLinkedServer..SCOTT.EMP

**5. List the Linked Servers**

To list your linked servers and show the **OLE DB** provider that they employ, use the **sp\_linkedserver** stored procedure.