Running Head: SQL STANDARDS

SQL Standards: the “Real” Language

Jered McClure

Walden University

SQL Standards: the “Real” Language

SQL standards across platforms is a hot subject for some database administrators and a moot point for others. The primary reason is in environments where more than one architecture is being used. For instance, an Oracle database for one application and a Microsoft database for another. If queries from one platform are to be used in the other, then there is a high probability that the queries will not work on one or the other of the DBMSs due to proprietary incompatibility.

Let’s compare MS TSQL (transact SQL) with Oracle SQL:

TSQL

**CREATE** **TABLE** WORDS\_1 (

SOME\_WORDS **NVARCHAR**(50) **PRIMARY** **KEY**);

Oracle SQL

**CREATE** **TABLE** WORDS\_1(

SOME\_WORDS **VARCHAR2**(50),

**PRIMARY** **KEY**(SOME\_WORDS));

These are two very different statements. However, they both do pretty much the same thing. They both create a table named WORDS\_1, with an attribute called SOME\_WORDS set as the primary key. However, if you were to run the TSQL query on an Oracle box, the system would throw an error. The same goes with the Oracle query run on a Microsoft box.

The biggest difference between the two is the data types of the attribute. NVARCHAR is a non-ANSI standard data type that is much more flexible and useful in a Microsoft database. It is a slightly heavier data type than VARCHAR, but is much easier to manipulate in terms of casting and encoding than VARCHAR.

VARCHAR2 is in the same boat, in that it is a much more useful data type than VARCHAR. It stores more data, and takes up no room when a NULL value exists. However, there is no such data type as VARCHAR2 in MSSQL. As such, when the query is ported from Oracle to MSSQL it must be rewritten or the query plan will fail.

As such, if the two databases will not be sharing queries, and there will be no overlap between the data, then there is no reason to not use the proprietary query language for each DBMS. However, in such situations where the data will be shared between architectures, it is best to stay as close to ANSI standard as possible. The counter to this is that in most cases you will end up having to rewrite the query anyway, as the data types will be incorrect, or the fields will not match up as intended. That is, the argument could be made that it does not matter in either case as there is no such thing as a “standard” language in the real world of SQL.

Reference

Coronel, C., Morris, S., & Rob, P. (2012). *Database Systems: Design, Implementation, and Management* (10th ed.). Boston, MA, USA: Cengage Learning.

Microsoft Corporation. (2012, November). *nchar and nvarchar (Transact-SQL)*. Retrieved December 18, 2012, from MSDN: http://msdn.microsoft.com/en-us/library/ms186939.aspx

Oracle FAQs. (2005, November 26). *What is the difference between VARCHAR, VARCHAR2 and CHAR data types?* Retrieved December 18, 2012, from Oracle FAQs: Underground: http://www.orafaq.com/faq/what\_is\_the\_difference\_between\_varchar\_varchar2\_and\_char\_data\_types