



Best Practices for Migrating SQL queries to Denodo VQL

Revision 20200616

NOTE

This document is confidential and proprietary of **Denodo Technologies**. No part of this document may be reproduced in any form by any means without prior written authorization of **Denodo Technologies**.

Copyright © 2022
Denodo Technologies Proprietary and Confidential

Goal

This document lists some recommendations and good practices to migrate legacy SQL queries to Denodo VQL.

Content

- Use single quotes to surround literals replacing “ by ‘ as indicated by the SQL standard.
- The Knowledge Base article [VDP Conformance with Standard SQL](#) contains a reference of the Virtual DataPort conformance with the SQL 92 standard. The document is focused on query capabilities and contains information about Data Types, SQL Predicates support and SQL Functions support. The Query Expressions section lists the expressions defined by the standard SQL and their equivalent in Virtual DataPort, explaining the differences with the standard when appropriate.
- Identify dialect differences in the transformation functions. There are some functions that need to be slightly translated to Denodo VQL. In many of these cases, the effort can be automated. For example, changing from Oracle’s LENGTH to Denodo VQL’s LEN only requires a “search and replace”. The same occurs if we need to modify from Teradata’s SUBSTRACT to Denodo VQL’s SUBTRACT. For a complete reference of VQL functions equivalent to functions in other database systems see:
 - [Oracle SQL to Denodo VQL Quick Reference](#)
 - [Microsoft SQL Server SQL to Denodo VQL Quick Reference](#)
 - [Teradata SQL to Denodo VQL Quick Reference](#)
 - [DB2 SQL to Denodo VQL Quick Reference](#)
- Identify other differences in format, for example, the use of hard-coded dates. In this particular case, two options are available:
 - Modify the default date representation format in Denodo to match the original
 - Modify the queries to match Denodo’s default representation.The configuration of type date is detailed in the section “[Managing Internationalization Configurations](#), subsection Configuration of dates” of the Advanced VQL Guide.
- Remove references to databases in the FROM clause. For example:

```
SELECT * FROM acme_crm.client;
```

should be rewritten as

```
SELECT * FROM client;
```

since in Virtual DataPort the base view is going to have a name without ‘.’
- Use the “Create Base View From Query” option, if needed, to leverage the existing query syntax. Denodo can reuse those queries as subqueries to maximize query delegation. This is a very powerful feature that allows reusing

complex queries already built, allows using database-proprietary functions or constructions (e.g. Teradata recursive queries) and it also allows invoking stored procedures for those databases where graphical introspection is not supported. The Knowledge Base article [Using the Create Base View From Query Option](#) explains when and how to use this option. The limitations of this functionality are also explained.

- Denodo's library of transformations is extensible via plugins. This is useful, for instance, in cases where customers have their own custom functions in the underlying sources. Using the custom functions API, those can also be registered and used in Denodo. Further information can be found in section "[Developing Custom Functions](#)" of the Virtual DataPort Developer Guide.

Note: The [Denodo4Eclipse](#) plugin used to develop custom functions is deprecated in Denodo 8.0 and it may be removed in future major versions of the Denodo Platform.

- Test the queries to verify their correct behavior. Denodo provides a powerful Testing Tool called [Denodo Testing Tool](#) that can be downloaded from the Denodo Support Site, specially designed to handle migration scenarios. This tool will be extremely useful to verify scenarios with large amounts of queries in an automated fashion. The tool also generates HTML reports to easily identify erroneous queries.