Installation and Setup of UnityJDBC with Rapid SQL®

Overview

UnityJDBC allows Rapid SQL® users to create a virtual data source that may consist of multiple data sources on different servers and platforms. The user can enter one SQL query to combine and join information from multiple sources. Any database that has a JDBC driver is supported including NoSQL databases such as MongoDB.

Benefits

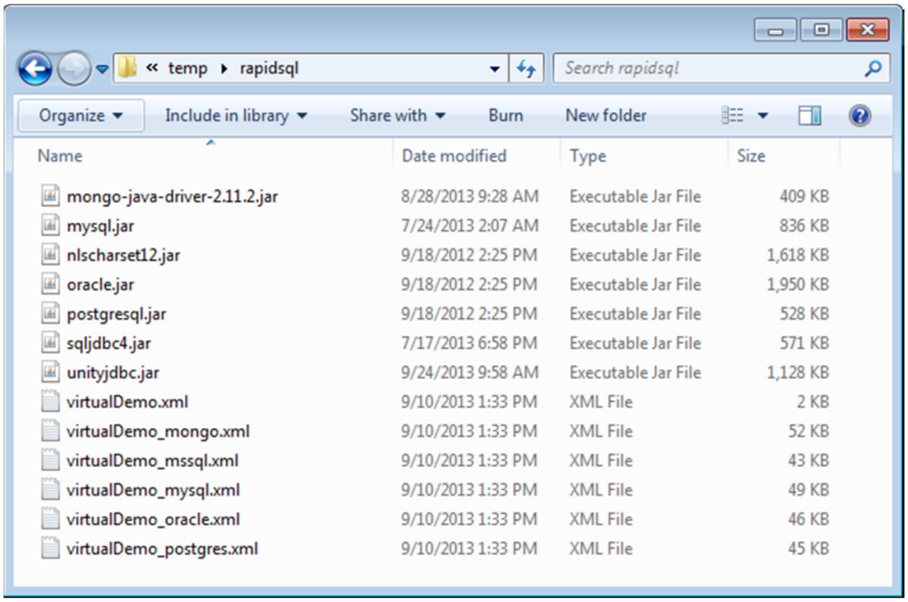
- No data source or server changes are required.
- Support for standard SQL including joins, group by, aggregation, LIMIT, and ordering where tables may come from one or more sources.
- UnityJDBC will perform function translation when a user requests a function or SQL feature/syntax that is not supported by a certain source.
- Users can use the virtualization information and driver in other Java programs and reporting software.

Installation

1. Download and install UnityJDBC. The UnityJDBC installation consists of the main `unityjdbc.jar`, a sample database and programs, and JDBC drivers for popular databases. Using UnityJDBC Source Builder, define the data virtualization. See: https://www.unityjdbc.com/support/doc/html/ch03.html.

2. After building your sources, you should have a `sources.xml` file and one or more XML files describing database schema information. You should create a directory on your system to copy these files into. Also, copy the `unityjdbc.jar` and other JDBC drivers that may be needed for your sources. In the example below, there are five databases: MySQL, Oracle, Microsoft SQL Server, PostgreSQL, and MongoDB. Each database has the JDBC driver in the directory. The main sources file is called `virtualDemo.xml`.

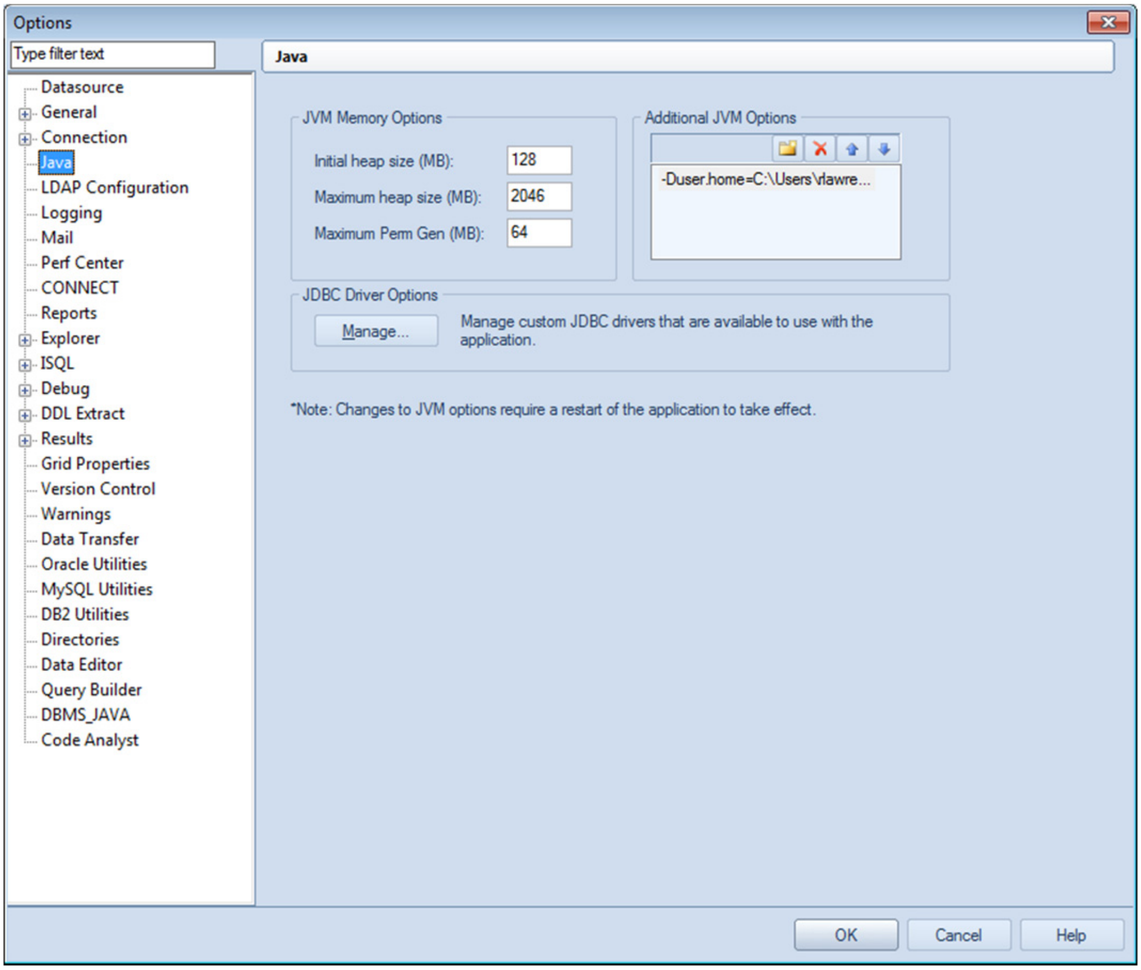
Sample Directory Storing JDBC Drivers and Source Information



Name	Date modified	Type	Size
mongo-java-driver-2.11.2.jar	8/28/2013 9:28 AM	Executable Jar File	409 KB
mysql.jar	7/24/2013 2:07 AM	Executable Jar File	836 KB
nlscharset12.jar	9/18/2012 2:25 PM	Executable Jar File	1,618 KB
oracle.jar	9/18/2012 2:25 PM	Executable Jar File	1,950 KB
postgresql.jar	9/18/2012 2:25 PM	Executable Jar File	528 KB
sqljdbc4.jar	7/17/2013 6:58 PM	Executable Jar File	571 KB
unityjdbc.jar	9/24/2013 9:58 AM	Executable Jar File	1,128 KB
virtualDemo.xml	9/10/2013 1:33 PM	XML File	2 KB
virtualDemo_mongo.xml	9/10/2013 1:33 PM	XML File	52 KB
virtualDemo_mssql.xml	9/10/2013 1:33 PM	XML File	43 KB
virtualDemo_mysql.xml	9/10/2013 1:33 PM	XML File	49 KB
virtualDemo_oracle.xml	9/10/2013 1:33 PM	XML File	46 KB
virtualDemo_postgres.xml	9/10/2013 1:33 PM	XML File	45 KB

3. Register the UnityJDBC driver in Rapid SQL under the `File` Menu, select `Options`…, then under `Java` click on the `Manage`… button for JDBC Driver Options.

Selecting Manage JDBC Driver Options



Initial heap size (MB): 128

Maximum heap size (MB): 2046

Maximum Perm Gen (MB): 64

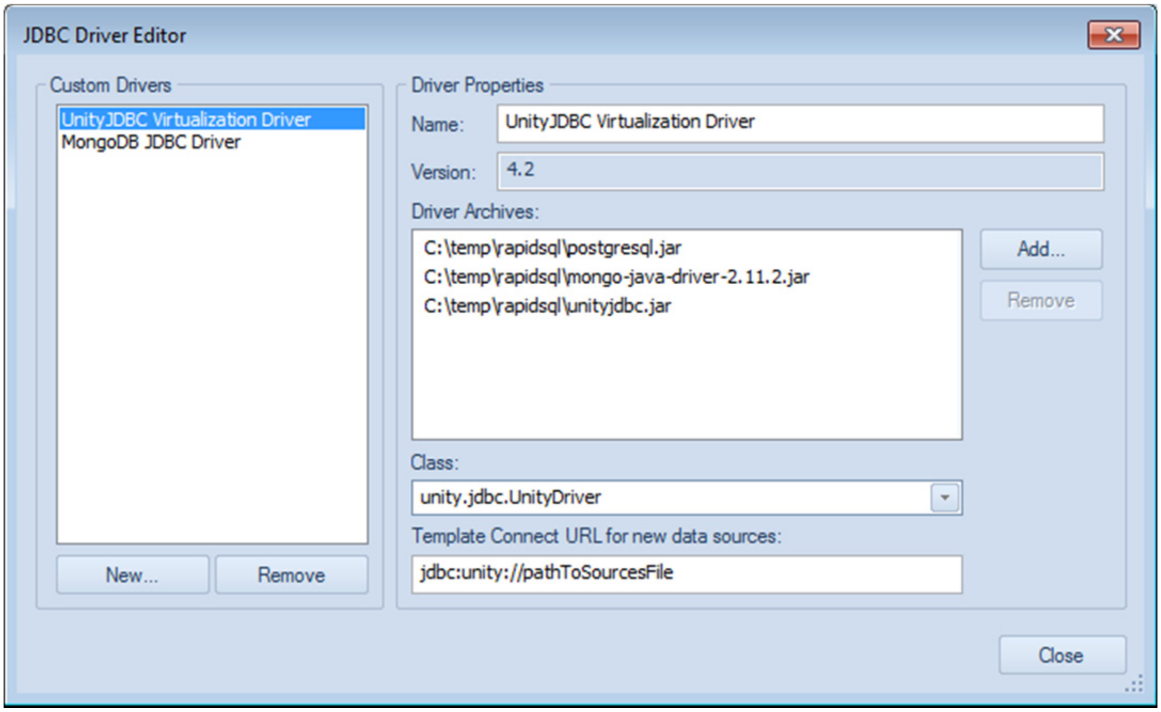
-Duser.home=C:\Users\rlawre...

Manage custom JDBC drivers that are available to use with the application.

*Note: Changes to JVM options require a restart of the application to take effect.

4. Then create a new JDBC driver. Steps:
    1. Name is: **UnityJDBC Virtualization Driver**
    2. Add the path to the `unityjdbc.jar` and other needed JDBC drivers under `Driver Archives`.
    3. The class is: **unity.jdbc.UnityDriver**
    4. The template URL is: **jdbc:unity://pathToSourcesFile**

Register UnityJDBC Driver



Name: UnityJDBC Virtualization Driver

Version: 4.2

Driver Archives:

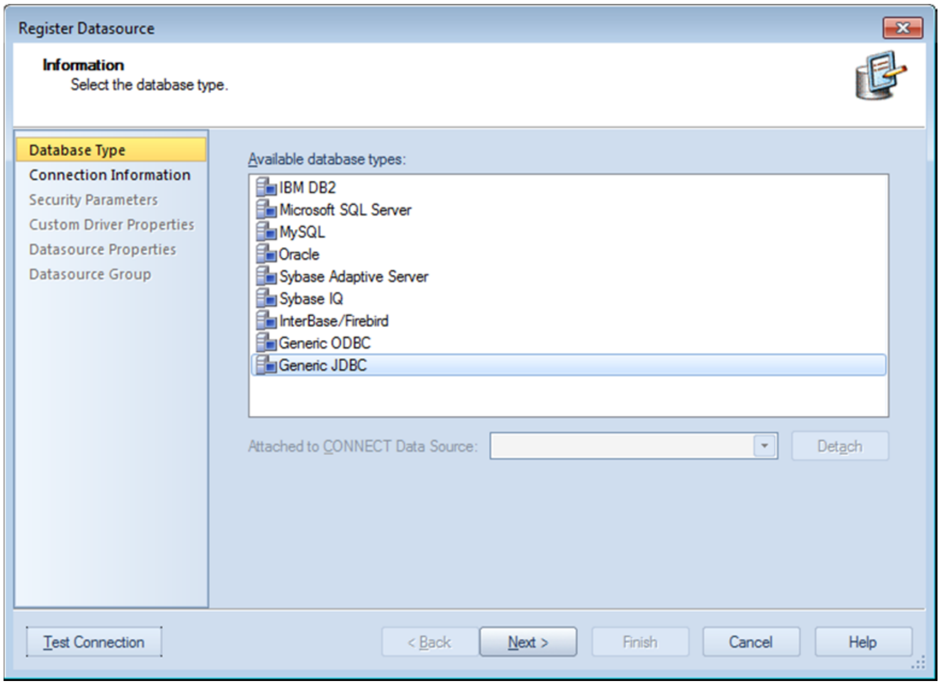
C:\temp\rapidsql\postgresql.jar
C:\temp\rapidsql\mongo-java-driver-2.11.2.jar
C:\temp\rapidsql\unityjdbc.jar

Class: unity.jdbc.UnityDriver

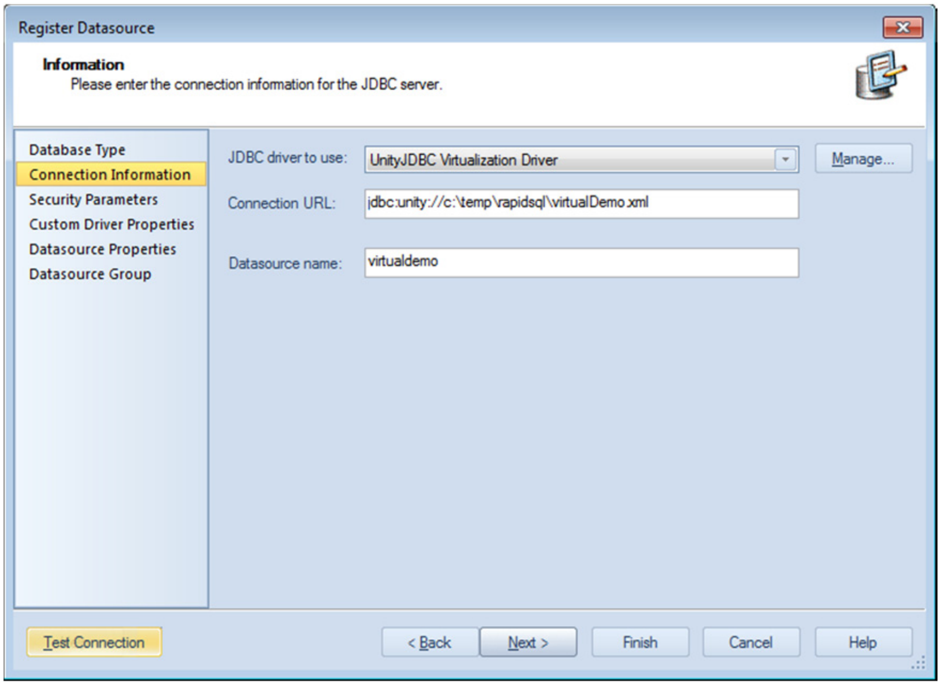
Template Connect URL for new data sources: jdbc:unity://pathToSourcesFile

5. Create a new source under the `Datasource` menu selecting `Register DataSource…` The database type is **Generic JDBC**.

Register Datasource: Select Generic JDBC



Select UnityJDBC Virtualization Driver, Provide Path to XML Configuration File, and Give the Data Source a Name



Register Datasource

Information

Please enter the connection information for the JDBC server.

JDBC driver to use: UnityJDBC Virtualization Driver

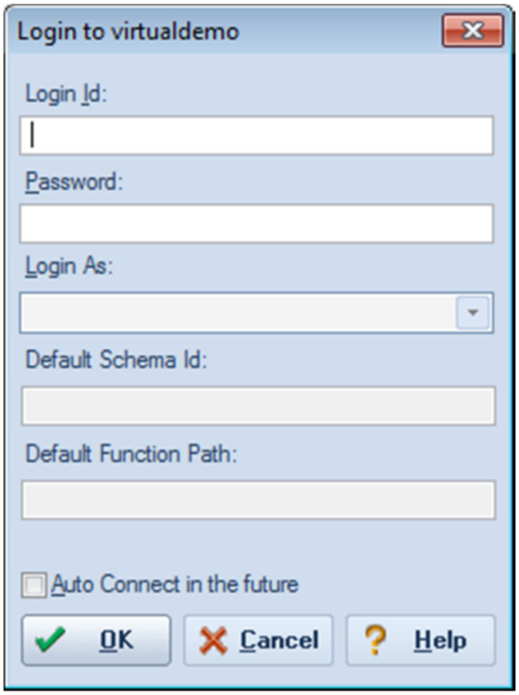
Connection URL: jdbc:unity://c:\temp\rapidsql\virtualDemo.xml

Datasource name: virtualdemo

6. There is no configuration needed under the **Security Parameters** tab if the source files are not encrypted. There is no configuration required under the other tabs: Custom Driver Properties, Datasource Properties, Datasource Group. Click `Finish`.

7. The data source will now be visible for querying. In the connection window, no login id or password needs to be specified (assuming encryption is not used). Click `OK` to connect.

Connection Window – User id/Password not Required

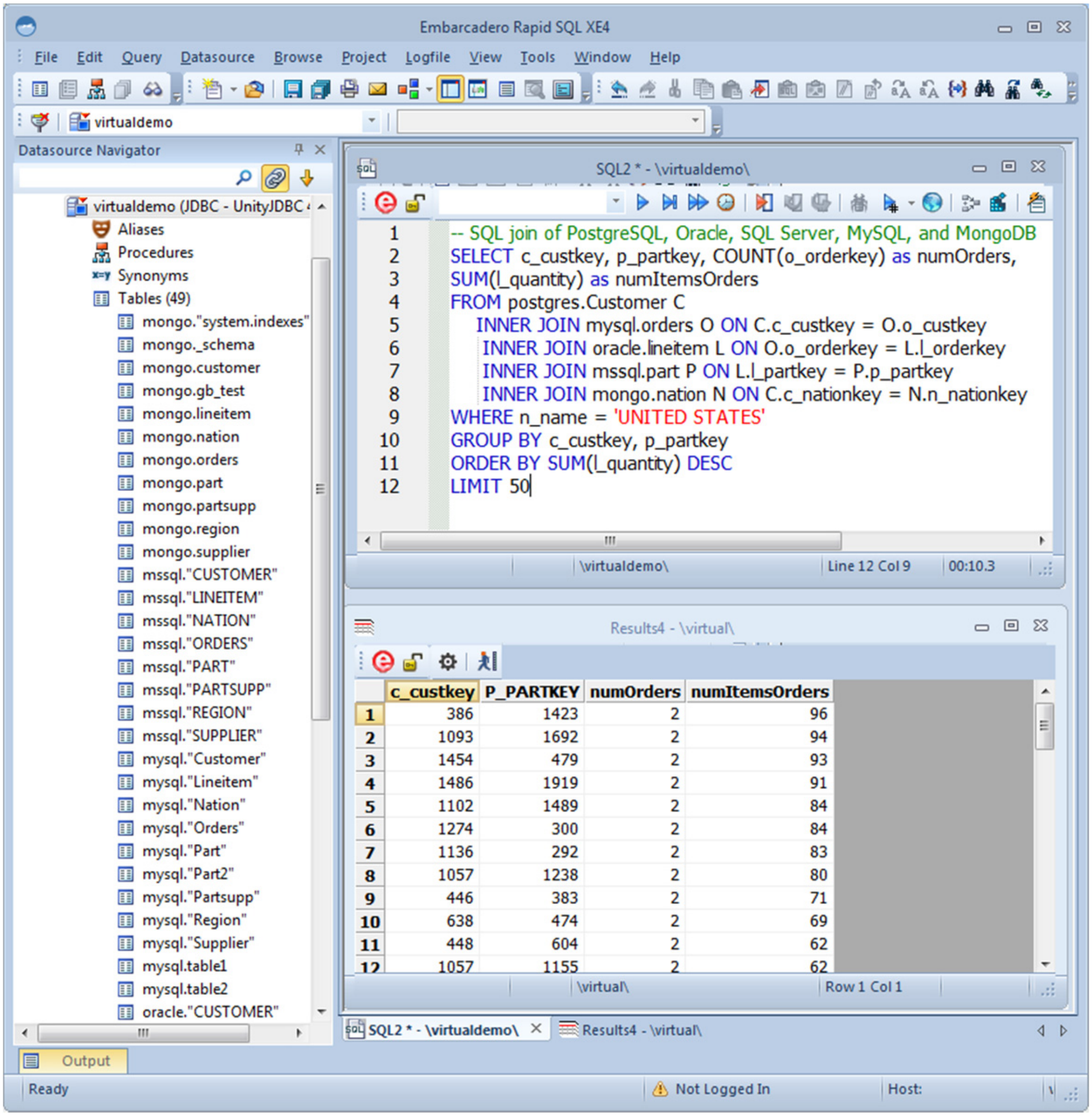


□ Auto Connect in the future

8. Tables appear as usual although they are prefixed by the source name. SQL queries function the same as usual although table names may need to be prefixed by a source name. All other query features in Rapid SQL are unchanged.

9. Below is a sample query that joins five databases together.

SQL Query joining tables in MySQL, Microsoft SQL Server, Oracle, PostgreSQL, and MongoDB



```sql
-- SQL join of PostgreSQL, Oracle, SQL Server, MySQL, and MongoDB
SELECT c_custkey, p_partkey, COUNT(o_orderkey) as numOrders,
SUM(l_quantity) as numItemsOrders
FROM postgres.Customer C
    INNER JOIN mysql.orders O ON C.c_custkey = O.o_custkey
     INNER JOIN oracle.lineitem L ON O.o_orderkey = L.l_orderkey
     INNER JOIN mssql.part P ON L.l_partkey = P.p_partkey
     INNER JOIN mongo.nation N ON C.c_nationkey = N.n_nationkey
WHERE n_name = 'UNITED STATES'
GROUP BY c_custkey, p_partkey
ORDER BY SUM(l_quantity) DESC
LIMIT 50
```

	c_custkey	P_PARTKEY	numOrders	numItemsOrders
1	386	1423	2	96
2	1093	1692	2	94
3	1454	479	2	93
4	1486	1919	2	91
5	1102	1489	2	84
6	1274	300	2	84
7	1136	292	2	83
8	1057	1238	2	80
9	446	383	2	71
10	638	474	2	69
11	448	604	2	62
12	1057	1155	2	62

Virtualization Export for use with Other Programs and Reporting Software

Once a virtualization has been created using SourceBuilder, all the information necessary is available in the XML files created. These files can be copied and moved to another location and used with any Java/JDBC program or reporting software.

Existing XML configuration files can be used by specifying an absolute or relative path in the JDBC URL. For example, if the user has saved the `sources.xml` file (and the associated schema files for sources in the directory `C:\tmp`, then a JDBC connection URL is: `jdbc:unity://c:\tmp\sources.xml`.

Trial Version

The UnityJDBC virtualization driver is released under a commercial license. The trial UnityJDBC driver is fully functioning with no time limits allowing an unlimited number of sources and queries. The only limitation is the size of the result set is limited to the first 100 rows. (Note there is no limit on the number of rows extracted from each source. `SELECT COUNT(*) FROM` *`table`* with a 1 million row table works as it only returns one result row.) Use `LIMIT 100` to get the first 100 results of a query.

For more information and technical support for the UnityJDBC driver contact:

UnityJDBC Support, support@unityjdbc.com, 250-807-9390

UnityJDBC driver information: www.unityjdbc.com