



Rdb JDBC Update

3

ORACLE

Agenda

- Driver Overview
- Versioning
- Persona Support
- Timeout Features
- Comments
- Executor Name Prefix
- Executors for RdbNative
- New Features in V7.2.5.1

4

ORACLE

Driver Overview

- **Oracle Rdb Native Driver:**

The Oracle Rdb native driver is a Type II driver for use with client-server Java applications.

This driver runs only on OpenVMS and is not suitable for applets.

- **Oracle Rdb Thin Driver:**

The Oracle Rdb thin driver is a 100 percent pure Java, Type IV driver.

Because it is written entirely in Java, this driver is platform-independent. It does not require any additional Oracle software on the client side.

Driver Overview

Oracle Rdb Thin Driver:

The Rdb Thin Driver needs a server-side component that services JDBC requests issued by applications using the Oracle Rdb thin driver.

- **Oracle JDBC for Rdb Thin Server:**

The server communicates with the Oracle Rdb thin driver using Java sockets over TCP/IP with the default Port ID 1701.

- **Oracle JDBC for Rdb Multi-Process Thin Server:**

The Oracle JDBC for Rdb multi-process server is using small memory footprint subprocesses to carry out the requested operations on the Oracle Rdb database.

The server communicates with the thin driver using Java sockets over TCP/IP with the default Port ID 1701.

Driver Overview

Oracle Rdb Thin Driver:

- **Oracle JDBC for Rdb Pool Server:**

The Oracle JDBC for Rdb pool server is a server-side component that accepts connection requests from the thin driver and redirects the requests to the next available Oracle JDBC for Rdb server for processing.

The pool server communicates with the thin driver using Java sockets over TCP/IP with the default Port ID 1702.

- **Oracle JDBC for Rdb Controller:**

The Oracle JDBC for Rdb controller allows basic management of Oracle JDBC for Rdb servers.

The controller may be used to start and stop servers as well as other operations pertaining to servers and connected clients. In addition the controller may be used to show the current status of Oracle JDBC for Rdb servers running throughout your network.

7

ORACLE

Versioning

- V7.1.1 – May 2003
- V7.1.2 – December 2003
- V7.1.2.1 – March 2004
- V7.1.3 – April 2005
- V7.1.3.2 – September 2005
- V7.1.3.3 – November 2005
- **V7.1.4 – January 2006: Alpha**
- **V7.2.4 – January 2006: Itanium**
- **V7.2.4.1 – March 2006: Consolidation of 7.1.4 and 7.2.4 into one version, for both platforms**
- **V7.2.5 – July 2006**
- **V7.2.5.1 – April/May 2007 (?)**

8

ORACLE

Persona Support

- V7.1.3.2 introduced the persona support.
- PERSONA support in the database and the PERSONA option for the Rdb JDBC driver are two different things:
 - In Rdb, it is part of security checking:
SECURITY CHECKING IS EXTERNAL (PERSONA SUPPORT IS ENABLED)
Enables the full impersonation of an OpenVMS user. This means the UIC and the granted right identifiers are used to check access control list permissions.
 - In Rdb JDBC, the thin server process will inherit its privileges and identifiers from the named persona.

9

ORACLE

Persona Support

- Server defaults to current process privileges and identifiers
- With Persona can start server with different user characteristics to that of starting process
- ThinServer, MPSTServer, PoolServer
- Useful when starting from SQL/Services

10

ORACLE

Persona Support

- OpenVMS Persona used
- To use Persona:
 - Requires read access to system authorization database
 - Requires IMPERSONATE privilege
 - Must be valid OpenVMS persona on the system

11

ORACLE

Persona Support

- The persona option can either be used in an XML formatted configuration file, or on the command line.
- E.g.:

```
$ java -jar rdb$jdbc_home:rdbthinsrv.jar -persona jdbc_user  
2007-04-03 05:10:02.549 JNI_Init changing user to jdbc_user
```

12

ORACLE

Timeout Features

- `srv.bindTimeout` (since V7.1.3)
- `srv.idleTimeout` (since V7.1.3)
- `cli.idleTimeout` (since V7.2.4.1)

- Values are in milliseconds.
- Can be specified in an XML formatted configuration file, or on the command line.
- The option `cli.idleTimeout` can also be specified as a qualifier on the connection string on the client-side application.

13

ORACLE

Timeout Features

- `srv.bindTimeout`
- Sets the timeout on waiting for a database connection to complete.
- If the database fails to connect within this time an exception will be raised.
- The default is 0 that means unlimited timeout.

14

ORACLE

Timeout Features

- `srv.idleTimeout`
- Determined by the lack of activity on the server socket.
- The timeout period is started after each new connection.
- If the timeout expires and there are current connections still using the server, the timeout period will be reset to start again.
- Log message:
 - `“Server terminated due to inactivity”`
- The default is 0 that means unlimited timeout.

15

ORACLE

Timeout Features

- `cli.idleTimeout`
- Determined by the lack of activity on the socket for that connection.
- When specified as a server switch, the timeout will apply to all clients connected using that server.
- You may also specify the client timeout as a qualifier on the connection string on the client-side application.
- Log message:
 - `“Client terminated due to inactivity”`
- The default is 0 that means unlimited timeout.

16

ORACLE

Comments

- Comments now supported in SQL text (since V7.2.4.1).
- Previous version problem in determining correct statement type if leading comment.

```
SQL = " /* comment */ select * from jobs"
```

```
SQLException: in <rdbjdbcsrv:execute_immediate>
```

```
%SQL-F-EXESELSTA, Attempted to EXECUTE a SELECT  
statement:RR000
```

17

ORACLE

Comments

- Layered software may expect comments to be supported, e.g. HIBERNATE.
- Several styles now supported

```
/* comment */
```

```
! comment <newline>
```

```
// comment <newline>
```

```
-- comment <newline>
```

18

ORACLE

Executor Name Prefix

- Executor sub-process name default
 - First seven (7) characters of server name + eight (8) character hexadecimal id eg.
RDBTHIN00000220
- This fails if there are two or more MP thin servers which are identical in the first seven characters, e.g.:
name="RDBTHINMP1"
name="RDBTHINMP2"

19

ORACLE

Executor Name Prefix

- Can now specify name prefix
 - srv.execPrefix (since V7.2.4.1)
 - E.g.:

```
<server
  name="RDBTHINMP1"
  type="RdbThinSrvMP"
  url="//localhost:1799/"
  srv.execPrefix="MP1"
/>
```
- OpenVMS process name 15 character limit
- Executor name :
PREFIX + upto 8 hex digits id

20

ORACLE

Executors for RdbNative

- By default all database operations within a standard Rdb Native driver instance are carried out synchronously, within a single OpenVMS process.
- Connection option multiprocess (since V7.1.3.2)
- Connection conn = DriverManager.getConnection("jdbc:rdbNative:my_db_dir:pers@multiprocess=true", user, pass);

21

ORACLE

Executors for RdbNative

- This switch is only available when you use the Rdb Native driver.
- To improve concurrency in a multi-threaded environment.
- Comparable to the MP thin server.

22

ORACLE

New Features in V7.2.5.1

- Setting Maximum Handshake Tries and Wait Duration.
- Using OpenVMS Failsafe IP.
- SQLDA Dumping.
- Blob restriction released.
- Server password.

23

ORACLE

New Features in V7.2.5.1

- **Setting Maximum Handshake Tries and Wait Duration.**
- Only for RdbNative with executors.
- A handshake protocol is established between the two processes to allow them to carry out subsequent inter-process communication.
- The main process will attempt 100 times in quick succession to establish the handshake, and then, by default, will try 500 more times with a delay of 10 ms between each try.

24

ORACLE

New Features in V7.2.5.1

- The defaults may be insufficient in certain circumstances.
- Two new connection options.
- *handshakeTries* (default 500)
- Sets the maximum number of times the main process will attempt to establish handshake with its associated executor sub-process.
- *handshakeWait* (default 10)
- Sets the time (in milliseconds) between handshake tries attempted between the main process and its associated executor sub-process.

25

ORACLE

New Features in V7.2.5.1

- **Using OpenVMS FailSafe IP**
- The thin driver should receive a socket exception on the failed TCP/IP port as the original service is no longer available.
- But if the driver is currently idle and not carrying out a read or write on the socket to the server, no exception will be raised.

26

ORACLE

New Features in V7.2.5.1

- Two new connection options.
- *networkKeepalive* (default false)
- If true the socket used to connect the client to the server will have **SoKeepAlive** enabled
- *networkTimeout* (default 0)
- Sets the maximum time (in milliseconds) this client connection will wait on the completion of a read or write on the network. If the read or write does not complete within the designated time an exception will be raised.

27

ORACLE

New Features in V7.2.5.1

- **SQLDA Dumping**
- Setting the tracelevel to 0x00004000 (decimal 16384) will provide information about the SQLDA information passed to and from SQL.

28

ORACLE

New Features in V7.2.5.1

- **Blob restriction released.**

Blobs will only be returned correctly from a SQL join statements for the first table mentioned in the join set. For example, given the SQL statement

```
Select ta.blob, tb.blob from table1 ta, table2 tb where
ta.name = tb.name
```

ta.blob will be returned correctly as it is from the first table referenced in the join set. Trying to access tb.blob may result in the following SQL error:
%SQL-F-BADPREPARE, Cannot use DESCRIBE or EXECUTE on a statement that is not prepared

- This is working now correctly without any error.

29

ORACLE

New Features in V7.2.5.1

- **Server password**

- For example:

```
<server
  name="rdbthnsrv9"
  type="RdbThinSrv"
  url="//localhost:1709/" >
  srv.password="0x811B15F866179583EB3C96751585B843"
/>
```

- This obfuscated pw is for "jdbc_user"
- Connection conn = DriverManager.getConnection("jdbc:rdbThin://bravo:1709/db_dir:pers@srv.password=jdbc_user", user, pass);

30

ORACLE

