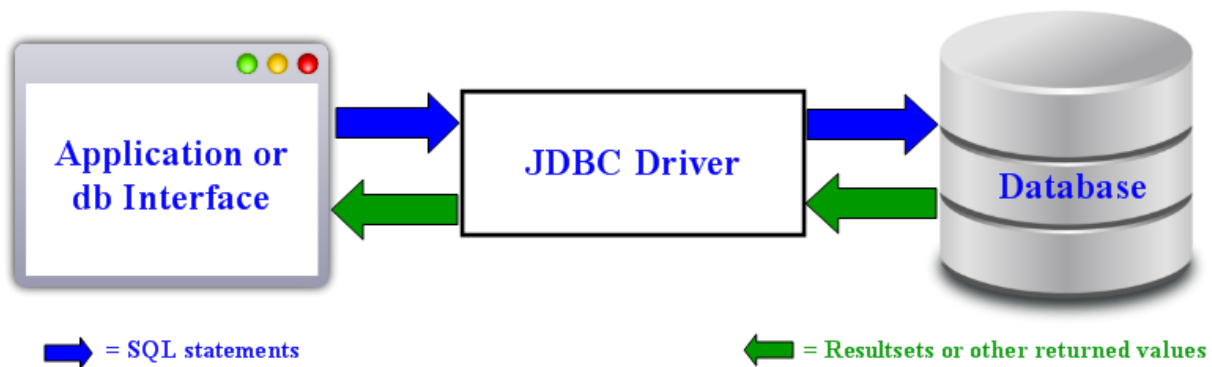


LISA 6 – JDBC Simulation Driver - Solutions/Issues

General JDBC

A JDBC (Java DataBase Connectivity) driver is a software component enabling a Java application to interact with a database. The JDBC driver provides a standard connection and API to the Java application while implementing database specific protocols to the database.

Normal JDBC Connection



There are two ways for java to connect to a database using JDBC:

Driver Manager:

To connect using a data manager the application uses two pieces of information, a Driver and a url.

For example, using MySQL:

Class/Driver Name: com.mysql.jdbc.Driver

url: jdbc:mysql://<server>:<port>/<database>

or: jdbc:mysql://localhost:3306/myDatabase

note: The url may contain other information such as user id and password.

Data Source:

A Data Source connection is first registered with a JNDI (Java Naming and Directory Interface) with the application querying the JNDI for the datasource by name.

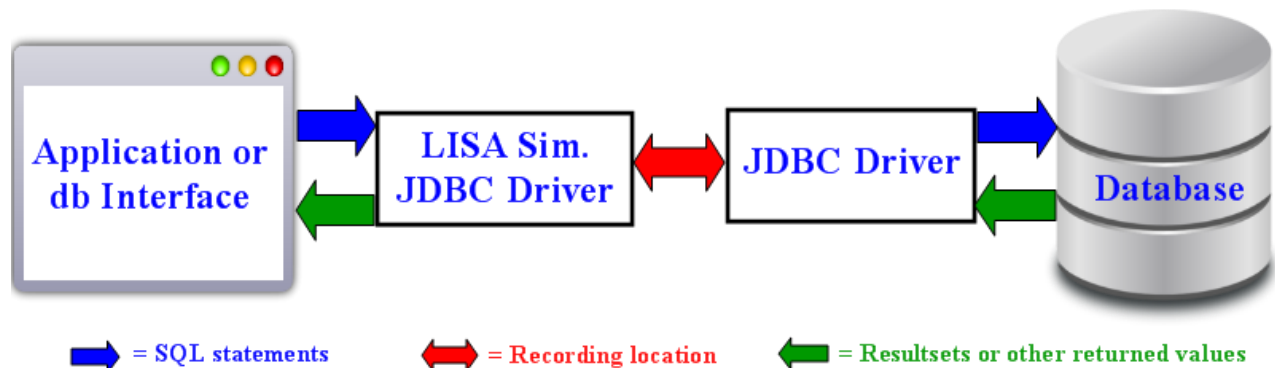
Thus, with a database registered with the name of “jdbc/myDatabase” code similar to the following will establish a connection with user id “user” and password “password”

```
Context ctx = new InitialContext();
DataSource ds = (DataSource)ctx.lookup("jdbc/myDatabase");
Connection con = ds.getConnection("user", "password");
```

Driver Based JDBC Recording with LISA

The LISA simulation driver (i.e. lisasim) is contained in the lisajdbcsim.jar and is a “pass through” JDBC driver. Information from the application will be passed to the lisasim, possibly recorded and then passed to the normal JDBC driver. The returning traffic also passes through the lisasim before returning to the application.

LISA Simulation/JDBC Connection



To use the lisasim, the connection information must be changed at the **application level**.
See [Application Setup](#):

What Does LISA Record?

The lisasim driver records the data conversation between the application and the database. It does not virtualize the database itself. Since conversations (i.e. 2-way communications) are recorded, SQL statements that do not return a resultset or any other type of data are not recorded.

SQL statements recorded: SELECT.

SQL statements normally not recorded: INSERT, UPDATE and DELETE.

Note: If your application is a data loader and only performs insert, update and/or delete; LISA may record little to no transactions.

<Need to add more commands, like CREATE TABLE, DESC TABLE>

General setup of the LISA JDBC Simulation Driver

Driver Manager:

Both the Driver and url must be changed at the application level.

Also, both the lisajdbcsim.jar and the original driver must be in the application's classpath.

New values are:

Class/Driver Name: com.itko.lisa.vse.jdbc.driver.Driver

url: jdbc:lisasim:driver=<original class/driver>;jdbcSimPort=<open port>;url=<original url>

Note: There is another parameter "state=<initial state>;" listed in our documentation. The connection appears to be more reliable if this parameter is omitted.

Where: Original class/driver = Original/Normal class/driver used by the application.

Open port = Port used by lisasim to communicate with the application.

Note: Do not use port 2999 as this port is used by the lisasim driver.

Original url = Original/Normal url used by the application.

MySQL url Example:

jdbc:lisasim:driver=com.mysql.jdbc.Driver;jdbcSimPort=7161;url=jdbc:mysql://localhost:3306/myDatabase

Data Source:

In contrast to a Driver Manager based connection, a Data Source connection is not specified by url. Instead, the connection is first registered with a JNDI (Java Naming and Directory Interface) with the application querying the JNDI for the datasource by name.

Thus, if a database had been registered with the name of "jdbc/myDatabase" code similar to the following will establish a connection with user name "user" and password "password"

```
Context ctx = new InitialContext();
```

```
DataSource ds = (DataSource)ctx.lookup("jdbc/ myDatabase");
```

```
Connection con = ds.getConnection("user", "password");
```

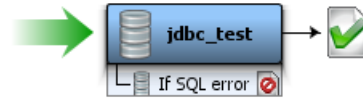
Application Setup

Quick/Temporary Testing:

Some customers only need to record a couple of transactions or want to create a quick recording solely for testing or training purposes. There are several tools available but they generally break down into two categories, LISA and DB interfaces.

LISA 6:

1. Create a new test case.
2. Add an “SQL Database Execution (JDBC)” step.
3. A configuration window will open to the left; configure it as follows



JDBC Driver: com.itko.lisa.vse.jdbc.driver.Driver

- **Connect String:**

jdbc:lisasim:driver=<original class/driver>;jdbcSimPort=<open port>;url=<original url>

(MySQL example):

jdbc:lisasim:driver=com.mysql.jdbc.Driver;jdbcSimPort=7161;url=jdbc:mysql://localhost:3306/myDatabase

- **User ID:** User ID used for database connection.
 - **Password:** Password for this user.
4. Click on the “Test Connection” button. You should receive a “success” message.
 5. **SQL Statement:** Enter a valid SELECT statement.
 6. Click the “Test/Execute SQL” button. A resultset should be returned.
 7. Go to “[LISA JDBC Recording](#)”.

DB Interfaces:

DB interfaces are applications that provide an easy to use user interface allowing the user to interact with a database.

There are several DB interfaces available such as Squirrel, DbVisualizer, Toad, MS Query, etc...

Currently only the following have been tested:

Worked: Squirrel v3.4.0

Did Not Work: DbVisualizer v8.0.11

It is not feasible to list the steps for each DB interface, so only general instructions will be given.

1. Copy the lisajdbcsim.jar file to a directory accessible by the DB interface. Usually this can be in the same directory as the interface’s existing jdbc drivers. However, sometimes the interface has a specific directory for adding new jdbc drivers.
2. Start the DB interface.
3. Add and configure the new driver. At a minimum you will need to provide:

- A name for the new driver.
- **Class/Driver Name:** com.mysql.jdbc.Driver
- **url:** jdbc:licasim:driver=<original class/driver>;jdbcSimPort=<open port>;url=<original url>

MySQL url Example:

jdbc:licasim:driver=com.mysql.jdbc.Driver;jdbcSimPort=7161;url=jdbc:mysql://localhost:3306/myDatabase

4. Verify new driver is working by running a SELECT statement.
5. Go to “*LISA JDBC Recording*”.

Applications:

Applications need to be re-configured to use the lisasim Driver.

This is usually done by changing a configuration file with the new values and restarting the application. Ensure the lisajdbcsim.jar has been added to the applications classpath.


Only one connection can be recorded at a time. Therefore if an application has multiple database connections, each one must be recorded individually.

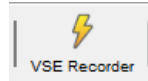
Also, if the application has the connection information hard coded (very rare), then the application must be re-compiled.

Change the application’s jdbc connection information as shown in “*General setup of the LISA JDBC Simulation Driver*”. Remember to comment out the existing jdbc connection information.

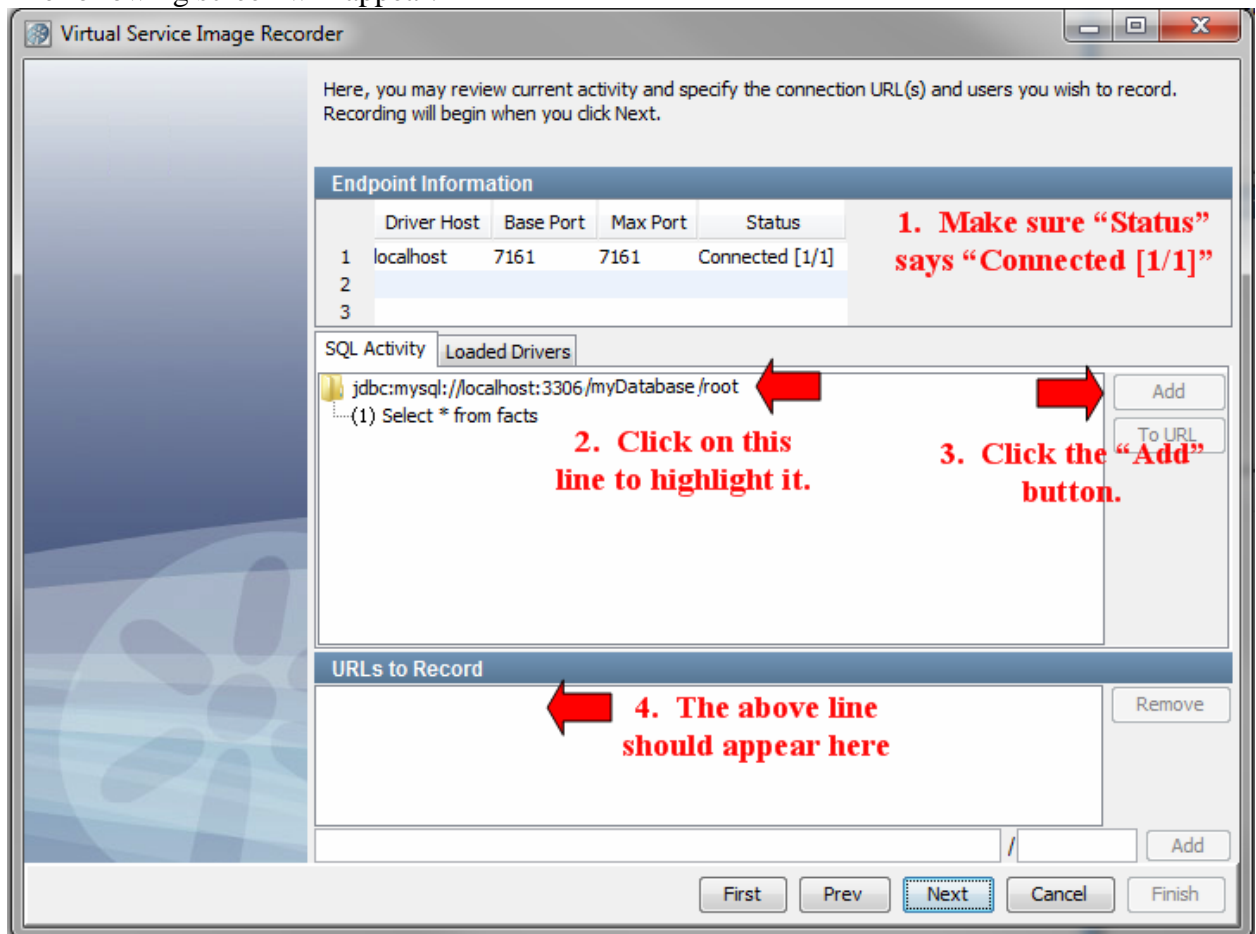
Go to “*LISA JDBC Recording*”.

LISA JDBC Recording:

1. Start the LISA Workstation.
2. Click on the “VSE Recorder” button in the toolbar.
3. Configure the recorder as follows:
 - **Write Image to:** Enter a name for the vsi file.
 - **Transport Protocol:** Select “JDBC (Driver based)”.
Note: “JDBC (Agent based)” does not work in LISA 6.0.x
 - **Model file:** Enter a name for the vsm file.
 - Click “Next”
 - Click the  button to add the recording endpoint.
 - Enter the correct endpoint information (server and port). The Base port MUST match the jdbcSimPort number entered in the connection information earlier.
 - Click “Next”
 - The following screen will appear:

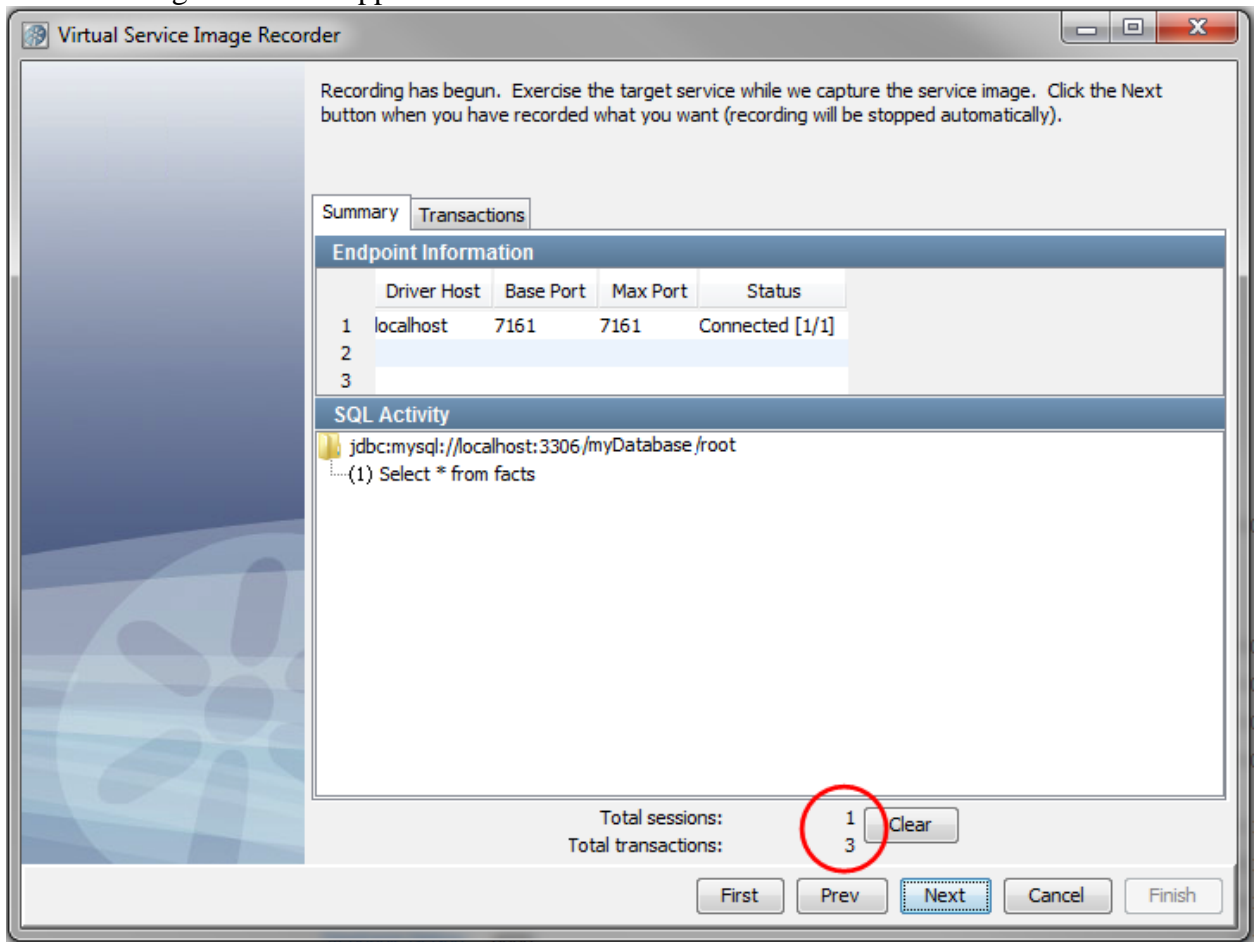


Endpoint Information			
	Driver Host	Base Port	Max Port
1	localhost	7161	7161



1. Ensure the LISA Recorder sees the connection.
If the status says “Connected [0/1]”, establish the connection by exercising the source (i.e. running an SQL statement in LISA, DB interface or the application).
2. Highlight the actual jdbc connection.
Note: Multiple SQL statements may appear below this line.
Note: If this line does not appear, see the “Issues” section of this document.
3. Click the “Add” button to add the jdbc connection to list of “URL’s to Record”.
4. Ensure the jdbc connection is listed in the bottom panel.
5. Click “Next”.

- The following screen will appear:



1. Run the application, DB interface, etc. you wish to record.
 2. Ensure the transaction count is increasing.
 3. When the recording is complete, click “Next”.
- If you wish to open the newly created vsm and vsi, check the two checkboxes.
 - Click “Finish”.

Issues:

Recording the LISA Demoserver:

It is impossible to record the LISA demoserver as described above. Since the Demoserver comes preconfigured with a lisasim driver already installed, attaching a second lisasim driver will cause an error, usually a null pointer exception.

To record the Demoserver, record the kiosk as described in the LISA documentation.

JDBC URL never appears in the setup window:

This occurs for unknown reasons but it is always after at least one mis-configured recording, i.e. wrong port entered.

To correct this issue, restart the LISA Workstation and Registry.