Creating SQL Responders from a Database Recording

This topic explains how to convert your database recordings from a Parasoft JDBC Driver into virtual assets with SQL Responders.

Sections include:

Overview

Virtualize provides a wizard to create and deploy SQL Responders for database recordings that were created using a Parasoft JDBC Driver (as described in Using the Parasoft JDBC Driver). These database recordings capture database queries along with their corresponding results.

Virtualize will create one SQL Responder to handle the SQL queries for a particular JDBC URL. When a SQL Responder receives a query, it attempts to match it against the set of available query templates. If a match is found, the correlated result set is used as a response. The data used to match queries and provide correlated responses can be saved as either a data repository or a .csv file.

Repository-driven Versus CSV-driven SQL Responders

Virtualize provides two options for generating SQL Responders from database recordings:

- **Repository-driven** SQL Responders are automatically parameterized with query templates and result sets defined in a Parasoft Data Repository (which is also automatically created when you use this option). This option allows you to rapidly add and modify query templates and responses using a graphical interface designed for manipulating large, complex data sources. Data values are stored and manipulated independently of the SQL Responder tool and .pva.
- **.CSV driven** SQL Responders specify query templates directly in the SQL Responder tool interface; the correlated result sets are stored in a .csv file.

Creating Repository-driven SQL Responders

Prerequisites

Before you can start creating repository-driven SQL Responders:

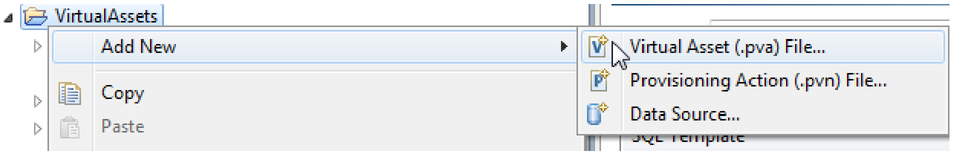
- Your team must have a Data Repository Server installed and running. For details, see Installing a Remote Data Repository Server.
- You must have access to a running Virtualize Server that hosts the database recording files for which you want to create SQL Responders.

Monitoring the Console View

It's helpful to keep the Console view visible as you are creating SQL Responders. This view will display any warnings, errors, and informational messages that are generated while processing the database recording file.

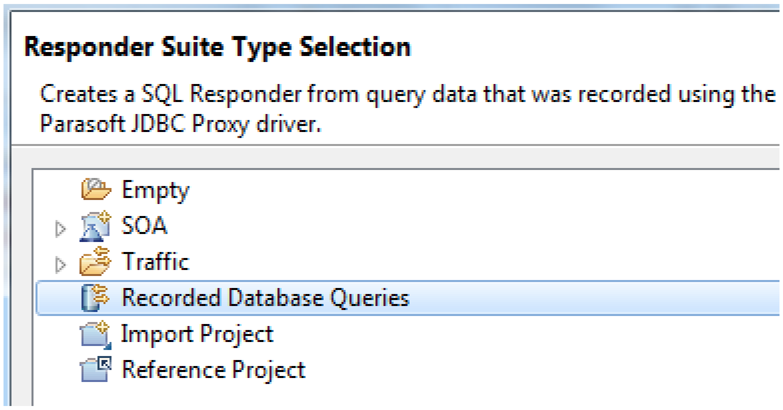
To automatically create and deploy repository-driven SQL Responders:

1. Right-click the **VirtualAssets** project and choose **Add New> Virtual Asset (.pva)** file.



VirtualAssets
Add New
Copy
Paste
Virtual Asset (.pva) File...
Provisioning Action (.pvn) File...
Data Source...
SQL Template

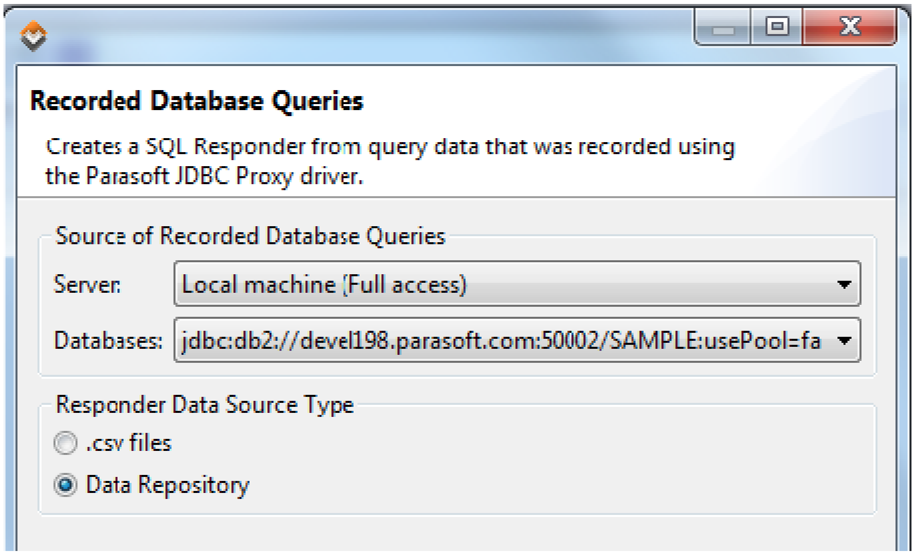
2. Specify a file name, then click **Next**.
3. Select **Recorded Database Queries**, then click **Next**.



Responder Suite Type Selection

Creates a SQL Responder from query data that was recorded using the Parasoft JDBC Proxy driver.

4. Complete the Recorded Database Queries wizard page as follows:
   a. For **Server**, select the Virtualize Server where the database recording file is located.
   b. For **Databases**, select the appropriate data source from the list of JDBC URLs. This list is populated based on the available recordings; the available URLs will direct Virtualize to the appropriate set of files that captured the recording.
   c. For **Responder Data Source Type**, select **Data Repository**.



Recorded Database Queries

Creates a SQL Responder from query data that was recorded using the Parasoft JDBC Proxy driver.

Source of Recorded Database Queries

Server: Local machine (Full access)

Databases: jdbc:db2://devel198.parasoft.com:50002/SAMPLE:usePool=fa

Responder Data Source Type

○ .csv files

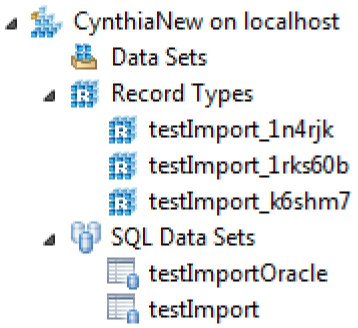
● Data Repository

   d. (Optional) If you want date fields automatically replaced with wildcards, enable **Wildcard Date Fields**. If this option is enabled and a query takes a date as a parameter, Virtualize won't require the dates to match exactly. We recommend enabling this option when a query takes the current time or a changing data as input, but you want a consistent response from Virtualize (ignoring changes in the date field).
   e. Click **Next**.
5. In the Parasoft Data Repository Settings page, specify which data repository should store the data used to parameterize the SQL Responder, then click **Next**.
   - From the **Server** box, specify which server you want to connect to (either the embedded server or a remote server). If you select the embedded server, the **Port**, **User**, and **Password** fields will be grayed out. If you select a remote server, the **Port**, **User**, and **Password** fields will be automatically populated (these can be adjusted if needed).
   - Under **Repository name**, select or enter the name of the repository you want to use. If you enter the name of a new repository, that repository will be created.
   - When defining a repository connection, you can check the connection by clicking **Validate**.

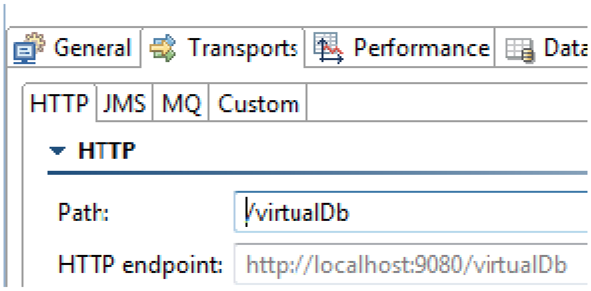
6. Click **Finish**.

The following items will be created and configured:

- A SQL Responder with parameterized query templates, result sets, and parameters will be added.
- A SQL data set will be added to the **SQL Data Sets** area of the selected repository. If this repository did not already exist, it will be created. Applicable record types will be added to the repository.
- A repository data source will be created for each added data set and the SQL Responder will be configured to use this data source.



- A .pva will be deployed to the local Virtualize server at /virtualDb.



Path: /virtualDb

HTTP endpoint: http://localhost:9080/virtualDb

Configuring Deployment Settings

For details on deployment settings, see Configuring Individual Virtual Asset Deployment Settings.

Creating CSV-driven SQL Responders

Prerequisites

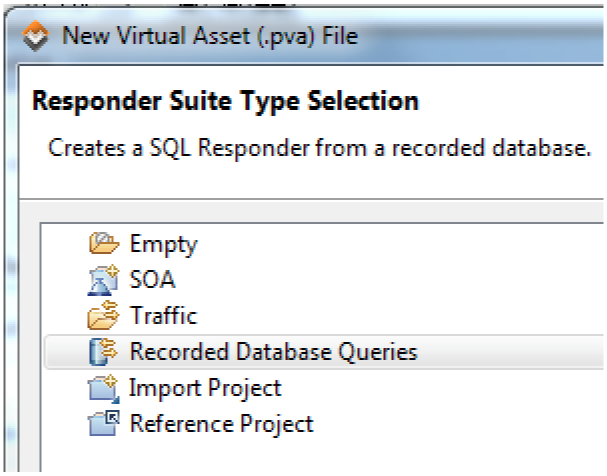
Before you can start creating .CSV-driven SQL Responders, you must have access to a running Virtualize Server that hosts the database recording files for which you want to create SQL Responders.

Monitoring the Console View

It's helpful to keep the Console view visible as you are creating SQL Responders. This view will display any warnings, errors, and informational messages that are generated while processing the database recording file.

To automatically create and deploy .CSV-driven SQL Responders:

1. Choose the **Recorded Database Queries** option in one of the available creation wizards.

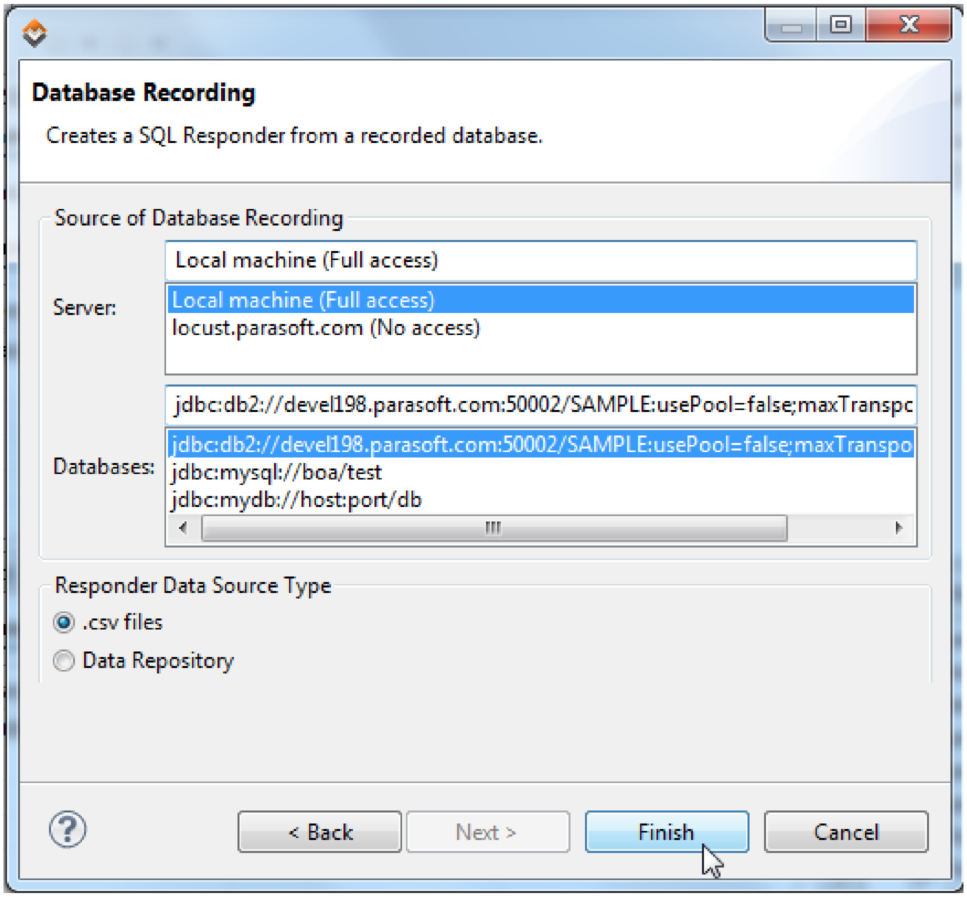


Responder Suite Type Selection

Creates a SQL Responder from a recorded database.

For details on accessing the wizards, see Adding Projects, Virtual Assets, and Responder Suites.

2. Complete the Recorded Database Queries wizard page as follows:
   a. For **Server**, select the Virtualize Server where the database recording file is located.
   b. For **Databases**, select the appropriate data source from the list of JDBC URLs. This list is populated based on the available recordings; the available URLs will direct Virtualize to the appropriate set of files that captured the recording.
   c. For **Responder Data Source Type**, select **.csv files**.



Database Recording

Creates a SQL Responder from a recorded database.

Server: Local machine (Full access)
- Local machine (Full access)
- locust.parasoft.com (No access)

Databases: jdbc:db2://devel198.parasoft.com:50002/SAMPLE:usePool=false;maxTranspc
- jdbc:db2://devel198.parasoft.com:50002/SAMPLE:usePool=false;maxTranspo
- jdbc:mysql://boa/test
- jdbc:mydb://host:port/db

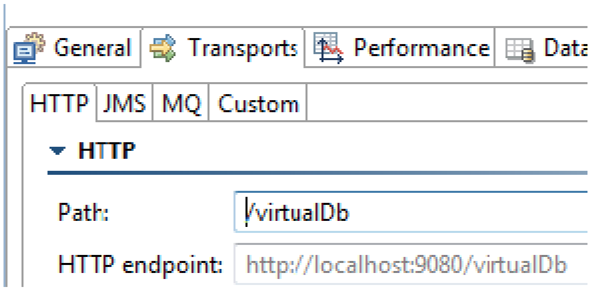
Responder Data Source Type
- ● .csv files
- ○ Data Repository

   d. (Optional) If you want date fields automatically replaced with wildcards, enable **Wildcard Date Fields**. If this option is enabled and a query takes a date as a parameter, Virtualize won't require the dates to match exactly. We recommend enabling this option when a query takes the current time or a changing data as input, but you want a consistent response from Virtualize (ignoring changes in the date field).
   e. Click **Finish**.

The following items will be created and configured:

- A SQL Responder that is configured with the appropriate JDBC URL, SQL Template, and Parameter Criteria.
- One or more ResultSet files stored within the database_recorded_data folder inside the VirtualAssets Project.

- A .pva will be deployed to the local Virtualize server at /`virtualDb`.



Path: /virtualDb

HTTP endpoint: http://localhost:9080/virtualDb

Configuring Deployment Settings

For details on deployment settings, see Configuring Individual Virtual Asset Deployment Settings.

Customizing the Virtual Asset

You can modify or extend the database behavior by editing the SQL Responder tool configuration. This is described in SQL Responder.