

Diagnostics

Diagnostic Tips

Here are the list of actions that can be taken to debug issues encountered.

Ask yourselves the following questions:

1. Can the report be generated via Elixir Report Professional?
 - If **No**, validate the template and datasource
 - If **Yes**, proceed to the next question
2. Can the report be generated via Elixir Report Server's sample Java application? (Located at *\$ERS_HOME/clients/demo/JavaClientApp*)(Ensure that you are able to launch the sample Java application, otherwise, state that you are not able to launch it)
 - If **Yes**, verify our calling application
 - If **No**, verify
 1. if the necessary files, such as JDBC drivers are available (JDBC drivers should be place at folder of Elixir Report Server)
 2. if the job request is sent to Elixir Report Server (Job request message will be sent to Elixir Report Server, visible in **server.log** of directory: *\$ELIXIR_REPORT_SERVER_HOME/logs/*)
 3. if error message is logged in **server.log** of directory:- *\$ELIXIR_REPORT_SERVER_HOME/logs/*
 - If **Yes**, verify if
 1. you do encounter any error message when generating through your application and debug accordingly
 2. there is some activity after clicking the report generation button
 - If **No**, verify if the web page is able to reference the necessary files
 3. there is a request sent to the report server?
Check **server.log** of Elixir Report Server.
If no job message, verify the report server configuration
 4. is the report generated in cache folder?
 - If report generated with correct file size, verify your cache configuration and cache path within your application
 - If report generated with incorrect file size, verify if the necessary parameters are passed in correctly for report generation

- If report is not generated, verify if the report is generated to another directory
- If report can be found in another directory, modify our cache configuration and cache path with your your application.
- If **No**, send us the feedback to the above questions and report template, datasource file, sample data and calling application (if necessary) to replicate the issue.

List of things to take note to avoid any issues:

1. Practice good design
2. Standardize on naming convention for files
3. Organize files
4. Keep files, JVM and products used, always in sync
5. Use repository path for reference to files (if possible)

* Example, Image URL: *repository:/ReportSamples/images/logo.jpg*

* Instead of, Image URL: *C:/ReportSamples/images/logo.jpg*