How to Run SQL Profiler Trace to Capture Information (Internal)

07/16/2025 6:00 pm EDT

Issue:

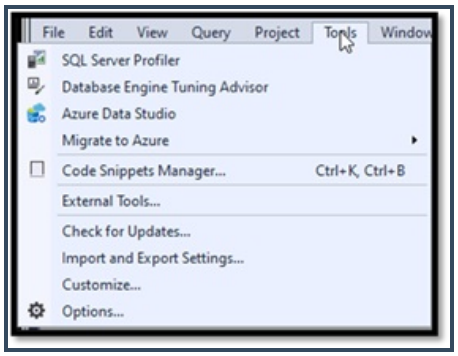
The customer is receiving errors in their SedonaOffice company. Is there a way to get more information on why the errors are occurring?

Resolution:

SQL Server Profiler can be used to capture information from processes accessing the SQL Server databases. Using SQL Server Profiler, you can create a trace that records and displays the commands sent to and the responses from the SQL server.

Open SQL Server Management Studio.

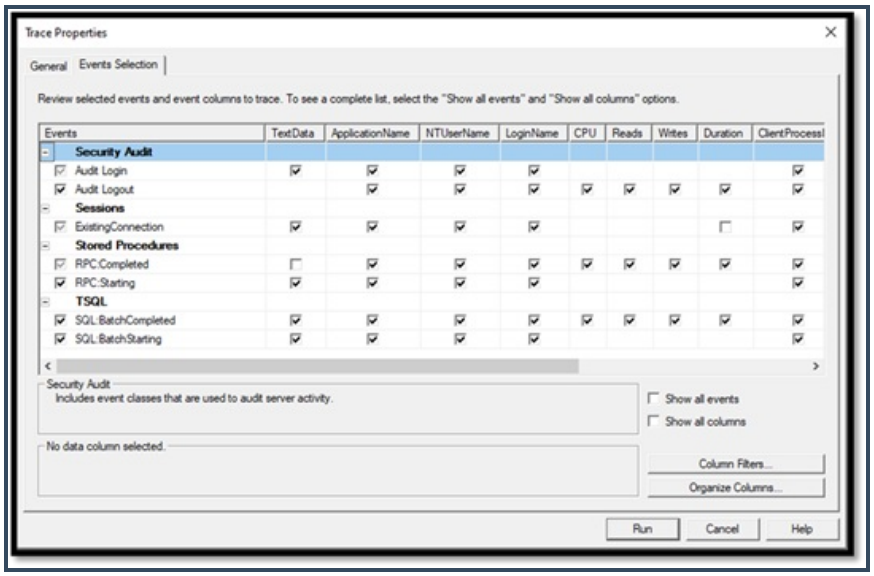
Connect to the correct server, select Tools, then SQL Server Profiler.



It can also be launched from the Windows Menu.

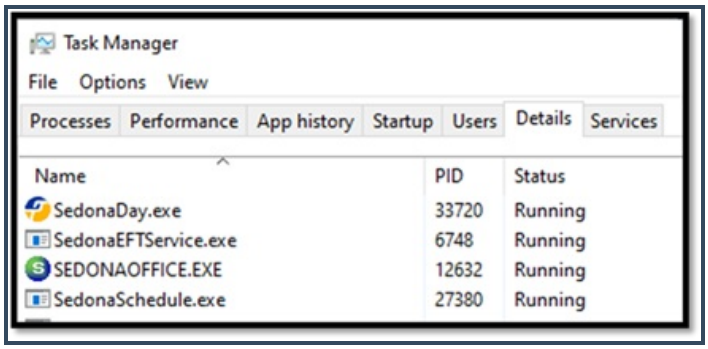
In the Events Selection Tab, you'll see the events it will monitor.

Make sure the options selected are the same as in the screenshot below.

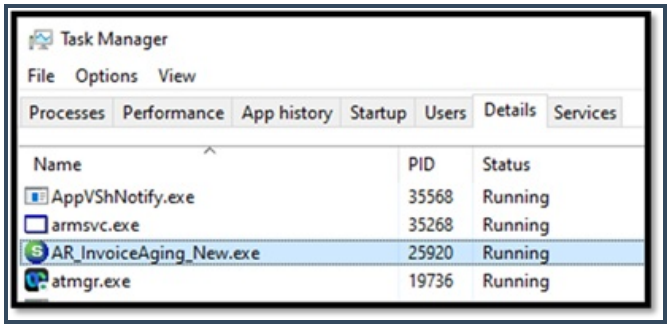


To limit the results to the machine having the issue, we use the ProcessID from the application. This can be obtained from the Windows Task Manager on the Details tab.

If the problem is with a report or Service Ticket, you will need to look for the .exe for the particular report or the SedonaScheduler.



If the problem is a report or any other application that opens outside of the SedonaOffice window, look for a separate application in the Task Manager. You may need to use this instead of SedonaOffice.exe.



Enter the PID from the application in the Column Filter for ClientProcessID.

Click OK.

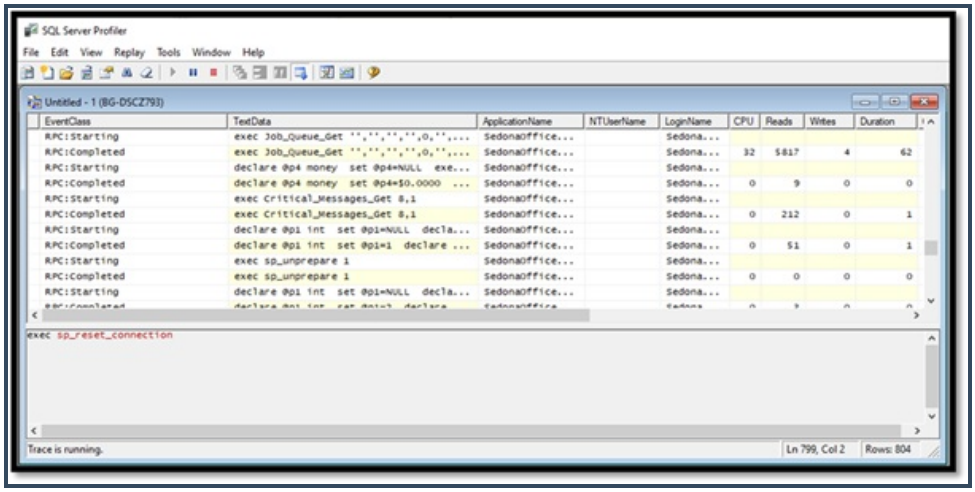
Once the events and filters are correct, click Run on the Trace Properties.

Start the trace right before performing the action.

Once running, it will capture the activity between the application and the server.

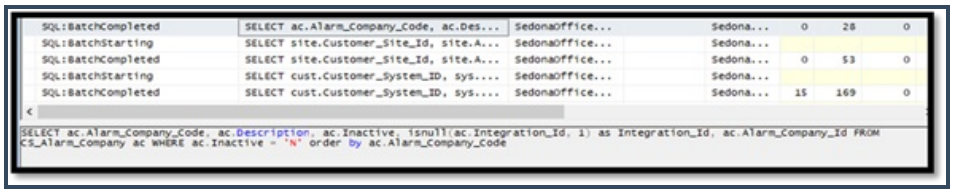
Perform the action in the application that is causing the error. Once the error is received, stop the Trace and save the file.

The trace will show every transaction between the application and server and can be used to see which event caused the errors. The saved trace file can be attached to the support case or emailed to Support for analysis.

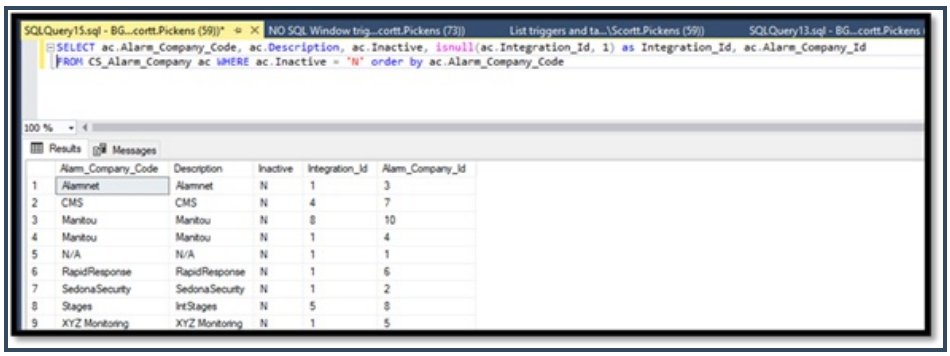


If the trace is stopped as soon as the error occurs, you can see the last transactions that caused the error.

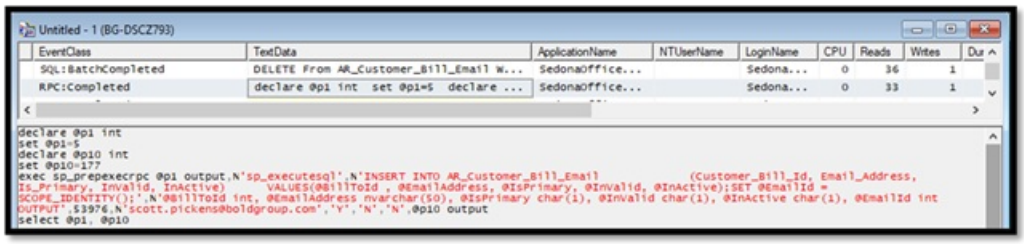
You can select the line in the trace to see what transaction took place.



You can copy and paste this in an SSMS Query window to execute the transaction to see if there is a problem.

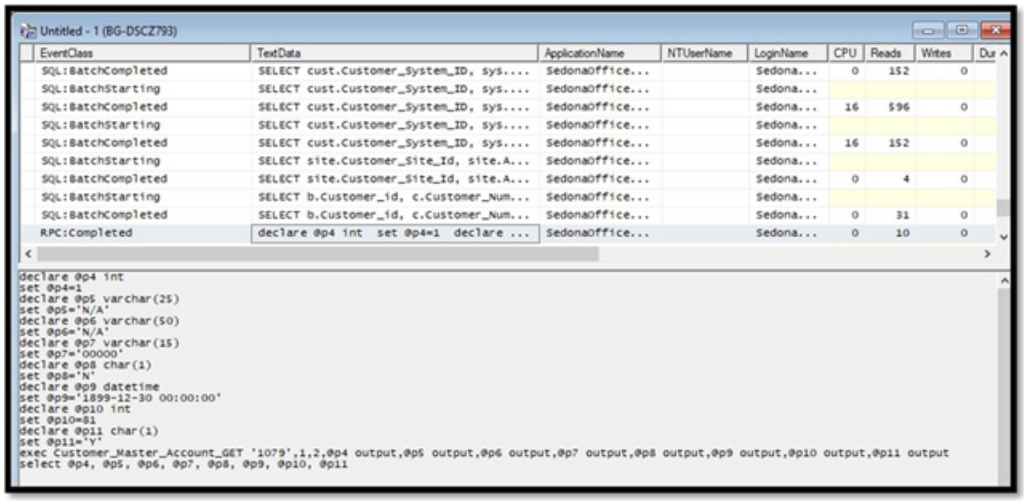


If it is a stored procedure, you can also copy and paste the statement. Be careful of procedures that are an Update, Add, Delete, or an Insert. If executed, these may add or change records that should not be changed.



If you need to test something like this, always use the Sandbox company.

If it is a Select or a stored procedure to GET information, it is OK to run in the production company. This is OK to run since it is a stored procedure to get the Master Account information.



This will help narrow down the location of the problem.