



Qure Optimizer™ for SQL Server

Quick Start Guide

January 2012



Qure Optimizer's optimization process should never be performed on production databases directly. The optimization process automatically applies changes to the optimized databases. Always use a copy of production on a non-production server.



- > Website: www.dbsophtic.com
- > Sales: sales@dbsophtic.com
- > Support: support@dbsophtic.com

Quick Start Checklist

Download and install Qure Optimizer for SQL Server

- Download and install Microsoft .Net framework 3.5 SP1 from the Microsoft web site.
- Install SQL Server 2005, 2008 or 2008R2 client tools from the setup program on the licensed SQL Server installation media.
- Download and install Qure Optimizer from the DBSophic web site on www.dbsophic.com.
- Prepare the Copy-of-Production Database and Production Traces.
- Use the SQL Server Profiler template provided with Qure installation (typically located in “C:\Program Files\DBSophic\Qure\TraceTemplates\”) to perform a trace on the production database.
- Restore a recent backup of the production database on the server to be used for optimization analysis.



Note: You must restore the copy-of-production database using the same name that is used in production. This is important for correct context correlations between the trace events and the database. See the section “Preparing an Effective Workload Trace” below, for additional information and guidelines.

Start the Optimization Session

- Launch Qure Optimizer for SQL Server.
- Click **Start New Analysis** and follow the wizard’s on screen instructions. Four simple steps are required to set up the analysis:
 - Provide connection details to the copy-of-production database.
 - Select the prepared workload traces for the optimization.
 - Configure a few simple optional optimization settings.
 - Start the optimization session.



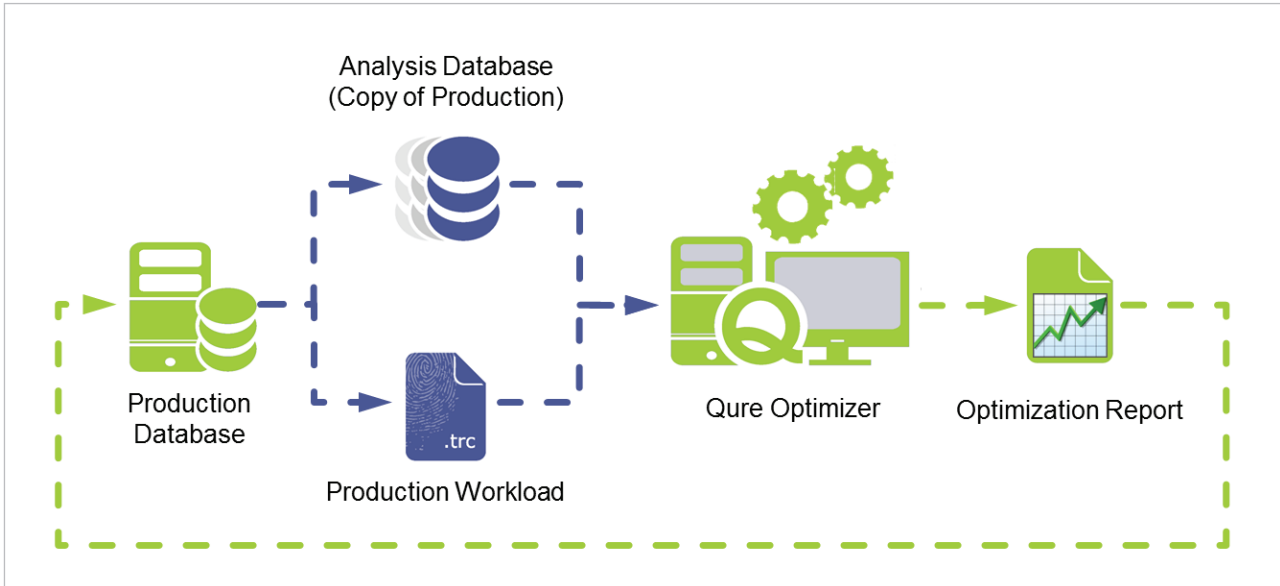
Note: The optimization session is fully unattended and may require from a few hours to a few days to complete, depending on the size and complexity of the database and workload, the hardware resources, etc.

Review Batches and Qure Optimizer’s Recommendations

Once the optimization session completes, use Qure Optimizer’s ribbon to navigate between the two main pages: **Batches** and **Recommendations**. These pages let you review the analyzed batches and their corresponding recommendations. Further...

- The **Batches page** lists all batches, their aggregated resource consumption and their actual performance improvements after optimization. For each batch, Qure Optimizer provides high level summary information regarding its operations and interactions with other database objects. It provides its script, its original and optimized execution plans, and categorized links to all associated recommendations.

- The **Recommendations page** provides full details about the recommendation. It includes short textual explanations, links to affected objects and batches, and correctional scripts. The filter pane on the left is automatically set when you navigate to this page using the links from the Batches page. Use the **Clear Filter** button to view all recommendations.



Qure Optimization Session Workflow



Tip: Use a standalone dedicated server and a “sterile” environment for the optimization session. Any concurrent activity on the server or client may affect the accuracy of the benchmark’s duration measurements. You may install the Qure Optimizer client on the server that hosts the analysis database.

Preparing an Effective Workload Trace

The workload trace is the primary source of information for Qure Optimizer’s optimization of the database workload. Qure assumes that the provided traces represent the full range of activities that the application and databases perform.

To get the most effective recommendations from Qure Optimizer, it is crucial that you adhere to the following guidelines.

Use the Provided Trace Templates to Capture the Workload

It is highly recommended that you use the provided SQL Profiler templates for capturing the trace. Although Qure will accept any trace that complies with its minimal requirements in terms of events and data columns, traces that don’t use the provided templates have an increased probability of errors, missed events, and unnecessary data.



Note: Qure automatically adds the required templates to SQL Profiler's list of available templates during installation. However, this list is user specific. If you need to run Profiler in a context of a user different than the one who installed Qure, start at step 1 below. If not, start at step 3.

To use the Profiler template:

1. In Windows Explorer, navigate to the Qure installation folder and then to the **templates** sub-folder.
 2. Double-click the **Qure_For_SQL_2005.tdf**, **Qure_For_SQL_2008.tdf** or **Qure_For_SQL_2008R2.tdf** file according to the version of production instance. This imports the Qure template into Profiler's available templates list.
 3. Launch Profiler, and then click **New Trace** or select **File > New Trace**.
 4. Connect to the production database server.
 5. From the **Use the template** drop-down list, select the Qure template.
-



IMPORTANT: Do not select the *Save to file* or *Save to table* options before starting the trace as these options may significantly increase the load overhead that Profiler generates on the production server. Always save the trace to a file or a table only after you have completed and stopped the trace.

6. Before starting the trace, read the rest of these guidelines thoroughly.

To perform a server-side trace using the provided templates:



Warning: On highly loaded systems, using Profiler may incur significant overhead. Server-side traces, although slightly more complex to set up, generate considerably less overhead.

1. Import the provided template into Profiler, as described in the previous procedure.
 2. Add the appropriate filters according to "Filter the Trace" below.
In Profiler, **Select File > Export > Script trace definition > For SQL Server 2005–2008**. Save the trace script, edit it manually if required, and use it to perform the server-side trace.
-



Note: For more information on performing a server-side trace, see <http://msdn.microsoft.com/en-us/library/ms191006.aspx>.

Filter the Trace

The trace template provided by Qure contains a few basic default filters.

You can add additional filters to the trace template to achieve the following goals:

- Reduce trace file size.
- Reduce the overhead that the tracing may incur to the production environment.
- Reduce the time required for the optimization session.
- Increase the efficiency of Qure's trace search algorithms.

The most important filter is on the IDs of the analyzed databases. If the trace contains events that are in the context of any databases other than the ones selected for analysis, these events are simply ignored. The databases' ID filter should be manually added to the trace definition.

Additional filters can be added at your discretion.

To add a filter on the databases' IDs to the trace:

1. Find the databases' IDs using the following query: **SELECT DB_ID (N'<Database_Name>')**
2. In SQL Profiler's trace properties window, click the **Events** tab.
3. Click **Column Filters**. The **Edit Filter** window appears.
4. In the left pane, select the **DatabaselD** column.
5. In the right pane, expand **Equals** by clicking on the [+] icon and enter the database ID. Repeat for the IDs of all optimized databases.

Qure does not analyze or provide recommendations for system activities (such as executing SQL Server system procedures), replication activities, maintenance activities (such as backups and statistics updates), security activities (such as logins, permission changes), and so on.

You can safely filter these activities out according to your application-specific behavior. We recommend that you perform a short sample trace and manually analyze it to see which events can be safely filtered out.

For example, here are a few filters you may want to consider.

- Filter out SQL Agent activities using a filter on the application name, unless you use SQL Agent jobs to execute application procedures.
- Filter out ADO internal operation activities using filters on text data, such as "EXEC sp_reset_connection", "IF @@trancount > 0 COMMIT TRANSACTION", and so on.
- Filter out Management Studio activities using a filter on the application name unless you want to deliberately execute specific activities manually to be included in the trace file.

Additional Trace File Recommendations

- **Do not use filters based on resource consumption such as duration, reads, writes, or CPU.** Filtering out any "real" application activities during the trace will have an effect on Qure Optimizer's final recommendations. For example, Qure Optimizer considers the table activity when evaluating index recommendations. Filtering out "INSERT... VALUES..." statements, which usually consume very little resources and which cannot be explicitly optimized, may

lead Qure Optimizer to assume that the table has fewer modifications than in reality, leading to less-than-optimal index recommendations.

- **When using backups of the production databases as the target for analysis, perform the trace soon after the backup is made.** This enables Qure Optimizer to correctly replay and analyze the queries from the trace against the data in the database during the analysis benchmark process. For example, if you use an old backup with a recent trace file, or vice versa, the queries may return different results than they would in production due to data inconsistencies; this in turn may result in less accurate analysis results.



Note: A trial analysis is limited to 100,000 workload events. Therefore, there would be no need to collect more than 150,000 events in the trace, assuming some will be filtered out.
