|  |
| --- |
|  |

## Creating Charts with Azure Log Analytics

The instructor will work demonstrate how we can use Log Analytics to identify query issues and audit query executions.

### 20 longest queries by execution time chart

1. Navigate to the Log Analytics workspace that was configured to receive Synapse diagnostic data as part of the pre-requisites of this engagement
2. Click on ‘logs’ from the log analytics workspace homepage

Graphical user interface, text, application, chat or text message

Description automatically generated

1. Close out the ‘example queries’ popup
2. In the query editor window – past in the text from ‘LongestRunningQueriesByExecutionTime’
3. Replace the database name variable with your database name

Text

Description automatically generated

1. Save this query for future use. The customer can now recall the query using the ‘saved queries’ button.

Graphical user interface, text, application, email

Description automatically generated

1. Press ctrl-A to select ALL of the data in the window – including the commented out portions on the bottom. Then Press Run to Execute Query.

Graphical user interface, text, application, email

Description automatically generated

1. Select ‘Pin To Dashboard’ then in the popup choose the dashboard we are working with

Graphical user interface, text

Description automatically generated

1. Navigate to the dashboard and find the new chart. Click the pencil icon to rename the chart to “20 Longest Queries by Execution Time (Completed Queries excluding batches)

Table

Description automatically generated

1. In the popup at the top of the window click ‘Save’ – you will want to do this each time you make a change.



### Most Rows Moved by Query charts

1. Navigate back to the log analytics workspace
2. In the query window, insert the query text for ‘MostRowsMovedByQueryStep’
3. Replace databaseName with the actual database name in the query. Note this is case-sensitive.
4. Save the query if desired
5. Select ALL data in the query window including commented sections – press run
6. Pin to dashboard
7. Without leaving the query window, press the ‘Chart’ button. Set the 3 parameters as seen in the screenshot below. Note that you will need to set ‘OperationType\_s’ first to be able to set the first parameter to RequestID\_s

A picture containing graphical user interface

Description automatically generated

1. Navigate back to the dashboard. Rename the first chart to “Rows Processed By Query step”
2. Rename the second chart to “Rows Processed by Request\_id”

Graphical user interface, application

Description automatically generated

1. Click save at the top of the dashboard



### Rows processed per hour chart

1. Navigate back to the log analytics workspace
2. In the query window, insert the query text for ‘RowsProcessedPerHour’
3. Replace databaseName with the actual database name in the query. Note this is case-sensitive.
4. Save the query if desired
5. Select ALL data in the query window – press run
6. In the bottom half of the window, press the ‘Chart’ button. Set the 3 parameters as seen in the screenshot below.

Chart, line chart

Description automatically generated

1. Pin to dashboard

### Longest BuildReplicatedTableCache Chart and table

1. Navigate back to the log analytics workspace
2. In the query window, insert the query text for ‘longestBuildReplicatedTableCache’
3. Replace databaseName with the actual database name in the query. Note this is case-sensitive.
4. Save the query if desired
5. Select ALL data in the query window including commented sections – press run
6. Pin to dashboard
7. Without leaving the query window, press the ‘Chart’ button. Set the 3 parameters as seen in the screenshot below. Note you want to select the third parameter and set it to “don’t split”

Chart, bar chart

Description automatically generated

8. Pin to dashboard

### Most Replicated Table Rebuilds – 7 days

1. Navigate back to the log analytics workspace
2. In the query window, insert the query text for ‘MostReplicatedTableRebuilds’
3. Replace databaseName with the actual database name in the query. Note this is case-sensitive.
4. Save the query if desired
5. Select ALL data in the query window including commented sections – press run
6. Pin to dashboard
7. Navigate to the dashboard and find the chart you just pinned
8. Click the three dots on the top right corner of the chart and select ‘Customize Tile Data’



1. ON the customize screen set the settings as seen in the screenshot below. This will make it so that this chart is not affected by the time range set on the dashboard. This makes sense because the chart explicitly looks at the last 7 days of rebuilds.

Graphical user interface, text, application, email

Description automatically generated