KAFTA Hourly Execution Plan deletes field data when run for older date ranges

3024291

Question / Problem:

Why would the KAFTA Hourly Execution Plan delete field data when run for older date ranges?

Answer / Solution:

The KAFTA Hourly Execution Plan runs by default at five minutes after the hour. If a need arises to manually run the KAFTA Hourly Execution Plan with a specific date range, it is important to understand how Insight uses the data load date range and how that interacts with retention of the source KTA reporting data.

When a data load is performed with an execution plan, Insight is using the specified date range to get data from the source, which in this case is the TotalAgility_Reporting database. By default, KTA field data only persists in the reporting database for five days, thus for any older date ranges there is no data in the source database. This concept also applies to document data, however the default retention time is ten years, so this only becomes relevant if the retention period is greatly reduced from default.

In specific versions of Insight, 6.1.0.0 – 6.1.0.3, and 6.2.0.0 – 6.2.1.0, doing a data load with a date range extending beyond the bounds of what is available in the source database will cause older data to be lost.

All other versions: Data loads that go beyond the range of existing data will not cause older data to be lost, with one exception. If the data loads starts from “Beginning of times” to or beyond the most recent data in KAFTA (example: “Beginning of times” to “Current time”). In this situation, older data is still lost and replaced with only what is currently available in the reporting database.

To learn more about retention and removal of reporting data, see the following articles:

- KTA Reporting Field Retention Period
- KTA Reporting Document Data Retention

Data ranges of data in KAFTA and Reporting databases
Field Data

To see the amount and time range of field data in KAFTA vs the KTA Reporting database, use the following query. Note that ETL times are used for the date range of a execution plan data load, however a field’s last session time is used to determine when it is subject to be removed by data retention settings.

```
SELECT MIN(etl_end_time) AS KAFTAMinEtl, MAX(etl_end_time) AS KAFTAMaxEtl, COUNT(*) FieldFactCount
FROM [KAFTA_Data].[dbo].[Field Fact] AS ff

SELECT MIN([a1].[etl_end_time])+(GETDATE()-GETUTCDATE()) AS ReportingMinEtl, MAX([a1].[etl_end_time])+(GETDATE()-GETUTCDATE()) AS ReportingMaxEtl,
    MIN(faf.dt_last_sess) AS MinFieldSessionTime,  MAX(faf.dt_last_sess) AS MaxFieldSessionTime,
    COUNT(*) AS FieldAccumFactCount
FROM [TotalAgility_Reporting].[dbo].[field_accum_fact] AS faf
INNER JOIN [TotalAgility_Reporting].[dbo].[etl_job] [a1] ON faf.etl_job_key = [a1].etl_job_key
```

In the following example query results, the data in KAFTA extends back to March, however the data in the reporting database only goes back to September. If a data load was run on this system from "Beginning of times" to "Current time", then the data from March until that date in September would be lost, and KAFTA would only have the 35 records currently in the reporting database, rather than the 36684 that are currently stored.

Document Data

To see the amount and time range of document data in KAFTA vs the KTA Reporting database, use the following query. Note that ETL times are used for the date range of a execution plan data load, however a document’s created date is used to determine when it is subject to be removed by data retention settings, in addition to requiring that the document be marked completed.

```
SELECT MIN(dt_last_proc_datetime) AS KAFTAMinEtl, MAX(dt_last_proc_datetime) AS KAFTAMaxEtl,
    COUNT(*) DocCount
FROM [KAFTA_Data].[dbo].[Document] AS d
INNER JOIN [KAFTA_Data].[dbo].[Document Completed First] AS dcf ON dcf.doc_key=d.doc_key

SELECT MIN([a1].[etl_end_time])+(GETDATE()-GETUTCDATE()) AS ReportingMinEtl, MAX([a1].[etl_end_time])+(GETDATE()-GETUTCDATE()) AS ReportingMaxEtl,
    MIN(dd.dt_create_datetime) AS MinDocCreateTime, MAX(dd.dt_create_datetime) AS MaxDocCreateTime,
    COUNT(*) AS DocDimCount
FROM [TotalAgility_Reporting].[dbo].[doc_dim] AS dd
INNER JOIN [TotalAgility_Reporting].[dbo].[etl_job] [a1] ON dd.etl_job_key = [a1].etl_job_key
WHERE dd.is_processing_completed>0
```

In the following example results, the document data in the reporting database goes back as far as the data in the KAFTA database, so no document data would be lost regardless of the date range of the data load.
Summary

Do not ever do a “Beginning of times” to “Current time” data load or KAFTA will lose any previously accumulated data that is not still retained in the reporting database.

On specific versions of Insight (6.1.0.0 – 6.1.0.4, 6.2.0.0 – 6.2.1.1) do not do data loads beyond the data currently available in the reporting database (for example, the default 5 day retention of field data). Ideally update Insight.

Applies to:

<table>
<thead>
<tr>
<th>Product</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>KAFTA</td>
<td>ALL</td>
</tr>
<tr>
<td>KTA</td>
<td>ALL</td>
</tr>
</tbody>
</table>