

Performance Considerations for Lاسernet

- 2023-11-30 - Comments (0) - Lاسernet FO Connector FAQs

Lasernet for Dynamics 365

There are several considerations in relation to performance, particularly with reference to Data source. Any one or all of the following steps can be taken to ensure strong and fast application performance and a stable structure:

- Delete Tables and Fields unused by the **Query wizard**.
- Test your report with real and sufficient data.
- Check your table relations, that you do not get unnecessary data/structure.
- Empty ranges or no ranges often cause slow performance as all the records can get fetched.
- Add a field to an index. Having fields that do not belong to any index can cause slow performance because a full table scan is used for finding the related record(s). To avoid a potential full table scan as well as to ensure performance, it is recommended to add a field to an index.
- If a Data source has multiple child data sources at same level with Fetch mode 1:1 where an inner join is required consider using a subquery. It is possible to avoid performance issues and optimize the structure by adding subqueries to the existing query (main query)

Available starting with version 6.8.0

A new option for disabling as well as re-enabling the data sources has been introduced within the Query Wizard. It may be useful when optimizing performance and locating data sources (tables) causing no output.

To solve the performance issues using performance indicators, follow the steps listed below:

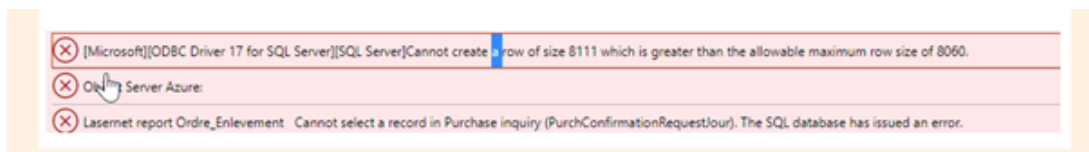
1. Activate **Performance indicators** when optimizing performance for a report.
2. Run the report to a destination, but not **Lasernet screen** to avoid the following response from Lاسernet:

'A dummy destination like fax like "Fax - faxnumber: 1234" or an Email.'

3. Locate the part causing slow performance and disable data source(s) within this part of the Query (e.g.: disabling a half of data sources).
4. Re-run the report and re-check the infolog in relation to the part of the structure which was causing slow performance.
5. Check the table used for the relations, indexes for the tables causing a slow performance.

Data source and size limits

SQL Server has a limit row size of 8060; if the size exceeds the limit, the following error is displayed:



The error is caused because you cannot have a row in SQL server which is larger than 8KB (the size of one page) because rows are not allowed to span pages – it is a basic limit of SQL Server: For more information, please see the following:

docs.microsoft.com/.../pages-and-extents-architecture-guide.

Data contained for all fields within an entity are contained in an SQL table with a maximum storage of 8060 bytes.

See documentation from MS:

<https://docs.microsoft.com/en-us/sql/sql-server/maximum-capacity-specifications-for-sql-server?redirectedfrom=MSDN&view=sql-server-ver15>

For more information, see the links below

[Error message: 1:N Fetch Mode is not allowed on Datasource having join condition with node other than its immediate parent](#)

Solution

[Error message: "Customer#, Invoice# was not delivered because of a print management setup issue" and "SQL server cannot create a row...size of 8060"](#).

How does the data get selected?

You can see the infolog with the SQL query in the performance log (if enabled). Selecting the **Full** value in the **Performance indicators** drop-down combo box allows you to add a query statement (SQL statement) to the Infolog once a report is executed.

Grouping, Fetch mode (1:1/1:N) affects how the SQL query is executed.

Fetch Mode / Fetch **1:1** are creating the following SQL-statement like the following

```
SELECT ... FROM CUSTCOLLECTIONLETTERJOUR T1  
CROSS JOIN CUSTCOLLECTIONLETTERTRANS T2  
CROSS JOIN CUSTTABLE T3  
CROSS JOIN CUSTTRANS T4  
CROSS JOIN DIRPARTYTABLE T5 ...
```

Fetch Mode / Fetch **1:N** are creating the following multiple and separate SQL-statements

```
SELECT ...  
FROM CUSTTABLE T1 ...  
  
SELECT ... FROM CUSTTRANSOPEN T1  
WHERE ... ACCOUNTNUM=@P3 ...
```

Related Content

- [Error Message: '1:N Fetch Mode is not allowed on Datasource having join condition with node other than its immediate parent.'](#)
- [Error Message: "Customer#, Invoice# was not delivered because of a print management setup issue" and "SQL server cannot create a row...size of 8060".](#)