Sage X3 Performance basics: Part 1

Mike Shaw

30/03/2022

Contents

- Introduction

- Types of performance problem

- Causes of performance issues

- How to approach a performance issue

- Case Studies

- Appendices

Introduction

Introduction

What is "good" performance?

- When an activity can be completed in an acceptable time

- "Acceptable time"

  - Generally based on user perception and/or experience

    - What are they comparing against

  - Likely to be different for different users

  - May change depending on workload/time pressure!

Types of performance problem

Types of performance problem

- Intermittent

    - Often the trickiest to understand and diagnose

    - Generally need to take a wider look at the overall system

- Consistent / Reproducible

    - Often the easiest to describe and diagnose, as can focus on one area and enable relevant debug/trace logs

- "Is my system giving best performance?"

    - Looking for validation of specific installation. i.e. Performance Review

Causes of performance issues

Causes of performance issues

sage

- Resource exhaustion

  - For example: CPU at 100%, Memory exhausted so paging to disk, process has hit configured memory limit so is thrashing (e.g. java), process needs lots of data from Database and is taking long time to return the data set

  - Root cause could be that more hardware is needed for the given workload, poorly written application code or poor application/database tuning

    - Check published sizing recommendations (Architecture and system requirements) and tuning guides (Node.js web server recommendations)

    - Test with representative workloads!

Causes of performance issues

- Resource contention

  - Can be due to Database locks, Application logical locks, hot blocks in database, sequence number generation, etc.

  - Root cause could be data model design, application code, poor application/database tuning, unusual data set, customisation or configuration choices

    - e.g. could consider to change Sequence numbers to Database, check your Anti-Virus exclusions

  - Both causes can be tricky to track down, but could consider using the Sage Support "Investigation Scripts" to collect data for review, if no other tools are available

    - See presentation from "Index page: Sage X3 Technical Support Tips and Tricks (September 2021)" for more information

How to approach a performance issue

Understand and describe the problem fully

- Troubleshooting and data gathering techniques

For ALL processes and screens considered slow, try to gather as much detail as possible, for example:

- When did the issue start, what changed since it was acceptable performance

- Navigation path to screen or process and function code being executed, also any parameters being used

- Does it happen in all instances (e.g. SEED, LIVE, TEST)

- Is it consistently slow or intermittent

  - If intermittently slow, does it only go slow dependant on amount of data, number of users on system, time of day, only after a system restart or some other factor

- Does it take the same time to complete in Edge, Firefox, Chrome browsers (for screens)

- How long the process takes when performing slowly, what is the normal amount of time it takes and what is the time expected

- Step by step test case to demonstrate the issue

- Do you see the same performance issue when running via a browser session on the server itself

- Business impact the problem is introducing, to allow prioritisation where there are multiple issues

Understand and describe the problem fully

- Troubleshooting and data gathering techniques

Consider what additional information can sensibly be gathered (and relevant to the reported issue) to provide evidence:

- Sage X3

  - Additional Syracuse logging

    - See "Syracuse logging" presentation from "Index page: Sage X3 Technical Support Tips and Tricks (March 2021)"

  - Performance Index ("IO Bench")

    - See "What is Sage X3 performance measurement tool" (Article 75064)

  - Timing Trace or Engine trace logs

  - SQL Server trace

  - For Web Service issues, obtain the "Technical information" from the Web Service responses

- Outside of Sage X3

  - Operating System statistics

  - SQL Server statistics

  - Virtual Server statistics

Understand and describe the problem fully

- User perspective

  - Iterate through with the user what they do and when the performance is poor, until a clear picture emerges of the exact nature and scope of the reported issue

    - e.g. is it a particular account number that causes an account enquiry to be slow, a particular date range, do other users have same issue, etc.

- Application perspective

  - e.g. Timing trace shows big wait in one procedure.   Engine trace highlights some SQL that seems to take long time to return a few rows, SQL trace also backs this up. Can see logical locks in X3

- System perspective

  - e.g. When the issue happens, one adonix process peaks CPU on the X3 Application server and lot of activity in the database

Research for known issues

sage

- Use all the resources available to look for known issues

  - Blog article [Sage X3 Support: Where can I go to find out stuff?](#)

    - Points to Knowledge Base, Blogs, Forums, etc

  - Also check "Patch finder"

Display columns: **Key - Type - Summary - Solution details - Product Area - Objects - Release - Behavior Change**

`Copy` `CSV` `Excel` `PDF` `Print`

Showing 1 to 20 of 20 results

🔍 Filter and highlight results...

Show `50` ▾

`Previous` `1` `Next`

Key ⬍	Type	Summary	Solution details	Product area	Objects	Release ⬍	Behavior Change ⬍
X3-238463	Bug	Unit price and amount are incorrect when running stock valuation in detail (STOCK12D) and warehouse subtotals selected.	When running the stock valuation report in detail with warehouse subtotals selected, the unit price and amount per warehouse are correct. Companies with a large number of stock transactions may experience slow **performance** when using the warehouse subtotals and should reduce the selection criteria as much as possible.	Distribution	• ARP/STOCK12D • ARP/STOCK12G • REPORT/STOCK12D • REPORT/STOCK12G • TRT/FUNSTVA	2021 R2 (12.0.26)	No
X3-232700	Bug	Stock valuation report with warehouse subtotal: Incorrect unit price and amount	When running the stock valuation report with the warehouse subtotal selected, the 'Unit price' and 'Amount' per warehouse are correct. Companies with a high number of stock transactions may experience slow **performance** when using the warehouse subtotal and should reduce the selection criteria as much as possible.	Distribution	• TRT/FUNSTVA	2021 R2 (12.0.26)	No
X3-	Bug	Slow	The WO production tracking	Distribution	• TRT/STKMAJ	2021 R2	No

Case Studies

Case Studies

sage

Based on real cases raised with Sage Support

1. ZMIKEREAD process slow

2. Invoice query is slow

3. Stock Prep list running slowly

4. Slow to render Sales order screen

5. "Everything slow"

6. Performance poor for all functions

Case Study 1: ZMIKEREAD process slow

- ZMIKEREAD process runs slowly but only when executed through X3 batch server

  - Normally takes 15 seconds to complete when run interactively

    ```
    22-05-18 10:00:05 Request activation (51000)

    Running

    Finished

    22-05-18 10:00:20 Request terminated (00000)
    ```

  - Takes 1 min 40 seconds when run through batch server

    Log reading C:\Sage\X3ERPV11\Folders\X3\SRV\TRA\RQT00000752

		Batch request log ZMIKEREAD
1		22-05-18 10:02:47 Request activation (51000)
2		Running
3		Finished
4		22-05-18 10:04:27 Request terminated (00000)
5		

Case Study 1: ZMIKEREAD process slow

sage

- Approach:

  - Check batch server job list to see timings and confirm if multiple jobs are running at the same time (i.e. is it a batch server throughput issue)

      ..\InvestigationScripts\SQL\mzBatchJobs.sql

  - Is it consistently slow when run through batch server

  - Is some other process interfering (e.g. SQL Server backup or some other operating system process)

    - Check SQL Server ERRORLOG

    - Collecting Windows PerfMon output may also be useful

Case Study 1: ZMIKEREAD process slow

- Findings:

  - Found that the run times through batch server did vary (Some good some bad)

  - Identified a scheduled job running outside of Sage X3 (via SQL Server Agent) which was hitting the database hard

- Solution:

  - After disabling the SQL Server Agent job, we now see consistent run times

  - NOTE: the user perception that ZMIKEREAD job was only running slowly through the X3 Batch Server turned out to be an incorrect assumption

Case Study 2: Invoice query is slow

sage

- Querying back certain invoices takes long time to show on screen

    - Invoices with large number of lines seemed to be the trigger

Case Study 2: Invoice query is slow

sage

- Approach:

    - Is it just down to the number of lines on the invoice (i.e. invoices with lots of lines are consistently slow and invoices with few lines are always quick)

        - If so, timing trace and SQL traces may identify long running functions/SQL

    - Is there anything at Sage X3 level that could influence the speed of the screen rendering

        - Remember that on a grid layout, increasing (columns * rows) will get exponentially slower

Case Study 2: Invoice query is slow

sage

- Findings:

    - The speed of loading looks to be related to the number of invoice lines

    - Found activity code SIH "Number of sales invoice lines" is set to 9999.

- Solution:

    - Changing SIH maximum number to a lower value resolves the issue

- Other thoughts:

    - Could also consider if any fields can be removed to reduce the amount of data that needs to be rendered

Case Study 3: Stock Prep list running slowly

sage

- Customer advises:

    - We have been experiencing long delays on running the Stock Prep List. Usually it takes 2 minutes for both small and large orders, but over the last couple of days the quickest runtime has been 25 minutes, with the slowest being 42 minutes. This job runs through the X3 Batch Server

Case Study 3: Stock Prep list running slowly

sage

- Approach:

  - Review the log files for the Stock Prep List runs, to ascertain exactly how long each took to run and how much data was being processed (number of invoices and how many lines on each invoice)

  - Also review the batch server jobs to see if any other processing was happening at the same time, which may have caused contention for the data, or potentially caused heavy usage of the database

Case Study 3: Stock Prep list running slowly

sage

- Findings:

  - There were 17 runs in total with data to analyse. The analysis proved that the number of lines/sec being processed was fairly consistent across most of the runs at around 6 lines/sec. There was one specific run which had exceptionally bad run time, however it is highly likely this run was effected by a "Stock Update" process which was running at the same time.

- Conclusion

  - In general the performance of Stock Prep List was agreed to be within reasonable boundaries, aside the one incident where there was contention caused by the "Stock Update" process

  - Customer perception of performance was based on the number of invoices, so did not take into account how many lines of data was being processed in each Stock Prep List run.

  - The customer accepted 6 lines/sec was adequate performance and fitted within their processing schedule.

- Additional notes

  - Additional work could be done to look at improving the process runtime if required.

  - Can consider to configure the Stock Prep and Stock Update jobs to be mutually exclusive, e.g. using Hourly Constraints to prevent contention.

Case Study 4: Slow to render Sales order screen

sage

- After upgrading from Sage X3 Version 6 to Version 12, find that it takes a long time to render the Sales Order screen

  - Currently takes 60-70 seconds to render the page in Version 12, but used to take <10 seconds on Version 6

  - No spikes being seen in server CPU, Memory or Disk usage

  - Problem is consistent for all Sales Orders, regardless of number of lines

  - Problem happens even when only 1 user logged in

Case Study 4: Slow to render Sales order screen

sage

- Approach:

  - As the problem is consistent, it doesn't seem to be hardware related and happens when only 1 user on system; gathering SQL Server trace is likely to provide the most useful information

Case Study 4: Slow to render Sales order screen

sage

- Findings:

    - Could see large amounts of data being accessed, which turned out to be related to the display of the data in the left list

- Solution:

    - Changing the parameters to restrict data in the left list resolved the issue

        - Parameters, General Parameters, Parameter Values

            - Module: SUP   Group: SEL

            - LFTBOX changed from 5000 to 50
                - NOTE: arbitrarily set to 50, additional testing is required to confirm if this is suitable value for the users requirments

Case Study 5: "Everything slow"

sage

- Customer reports that "Everything is slow"

  - Have just started using the system after migrating from a different software vendors system

Case Study 5: "Everything slow"

- Approach:

  - Need to identify the key business processes which are causing the biggest business problem and focus on these areas first

  - Review overall system statistics to understand where the bottleneck is

    - e.g. any CPU, Memory, Disk spikes, SQL Server performance statistics

- Also work with customer to understand if they are comparing "like for like" for their performance expectation, as in this case Sage X3 is proving a lot more functionality than the previous, simpler system

Case Study 5: "Everything slow"

- Findings:

  - Purchase invoice data entry screen was identified as key business process to review first

    - Was a heavily customised screen.   Using the standard screen was much quicker, but still relatively slow against expectation

    - Operating system and SQL Server were not showing as being heavily loaded

- Solution:

  - When reviewing VMWare level performance statistics, found that VMWare itself was "ballooning" – Once this was resolved, performance improved to acceptable levels in all areas

    - http://www.vmwarearena.com/vmware-memory-management-part-3-memory-ballooning/

Case Study 6: Performance poor for all functions

sage

- Customer reports that performance is poor across all functions

  - Have recently gone live

Case Study 6: Performance poor for all functions

- Approach:

  - Review general system performance statistics

  - Ascertain how many users are using the system at peak times and the kind of workload going through

  - Confirm if performance is OK when few users are on the system

Case Study 6: Performance poor for all functions

sage

- Findings:

    - Getting frequent "read ECONNRESET" messages in Syracuse logs. These errors seem to be happening only during user work hours

    - Have average of 40 concurrent users during these hours

    - Syracuse Web Server has been configured with only 1 child process, which is not enough for this level of user activity

- Solution:

    - Syracuse web server needed to be tuned as per sizing guidelines to resolve this issue

        - Node.js web server recommendations

Appendices

Appendix A: SQL Server tuning suggestions

sage

Whilst Sage Support cannot provide specific guidance on Microsoft SQL Server tuning, there are two settings which have been noted by our customers as having potential for significant performance impact

These suggestions are relevant to all Sage X3 versions, running all SQL Server versions

1. Cap SQL server maximum memory

2. Check "Max Degree OP" is set to the default of 0

Appendix A: SQL Server tuning suggestions

sage

1. Cap SQL server maximum memory

By default, SQL Server can change its memory requirements dynamically based on available system resources. Unless SQL Server is running on a Windows server dedicated to SQL Server, this can potentially cause memory starvation for other processes running on the same server, such as the Sage runtime

Refer to your Microsoft SQL Server support team to ascertain the best setting for your specific circumstances. The following links provide more information.

https://technet.microsoft.com/en-us/library/ms191144%28v=sql.105%29.aspx

- How to: Set a Fixed Amount of Memory (SQL Server Management Studio)

https://technet.microsoft.com/en-us/library/ms180797%28v=sql.105%29.aspx

- Effects of min and max server memory

Appendix A: SQL Server tuning suggestions

sage

2. Check "Max Degree OP" is set to the default of 0

SQL Server provides parallel queries to optimize query execution and index operations for computers that have more than one microprocessor (CPU)

The default setting of 0 would likely be the best setting, unless you have identified a specific reason to change this value

https://technet.microsoft.com/en-us/library/ms189094%28v=sql.105%29.aspx
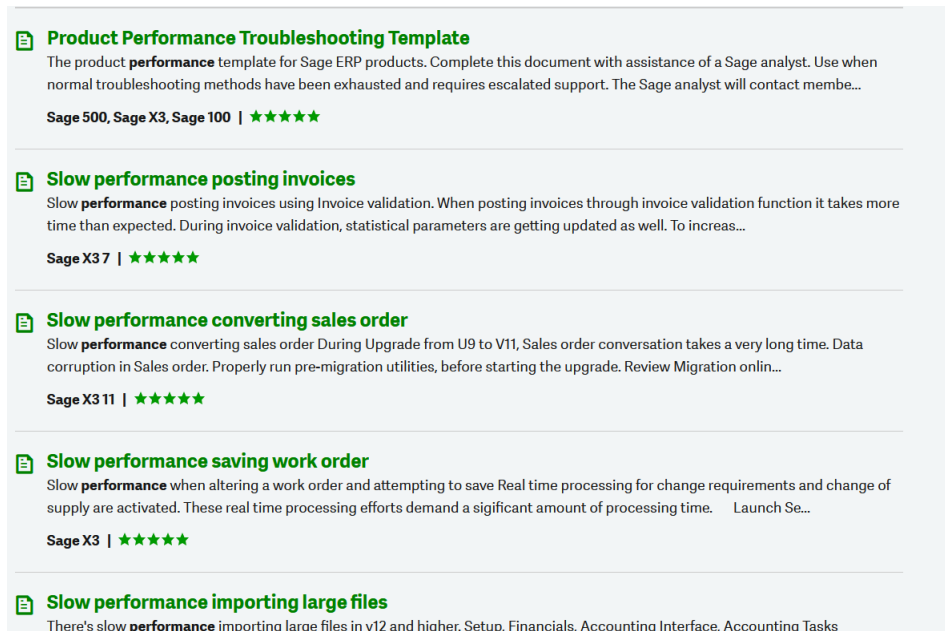
- How to: Configure the Number of Processors Available for Parallel Queries (SQL Server Management Studio)

https://technet.microsoft.com/en-us/library/ms178065%28v=sql.105%29.aspx

- Parallel Query Processing

Appendix B: Additional reading

- Blog articles

    - How do I determine the code that is performing poorly (slow)?

- KB articles

    - Search for "Performance" gives a load of hits!

Thank you