
Optimizing and Maintaining Performance for Microsoft Dynamics GP[®]

White Paper
Date: September 13, 2017



Contents

Introduction.....	5
Optimizing and Maintaining Microsoft SQL Server Performance.....	6
<i>Server Operating System Considerations.....</i>	6
1. Microsoft Dynamics GP System Requirements.....	6
2. Windows Updates.....	6
3. Windows Server 2008 R2 Service Pack 1 or later.....	6
4. Hyper-threading.....	7
5. x64 vs. 32bit.....	7
6. Anti-Virus Software.....	8
7. BIOS.....	9
<i>Disk Configuration Considerations.....</i>	9
1. Recommended Redundant Array of Independent disks (RAID) configuration.....	9
2. Isolation of the database files.....	9
3. Latency.....	10
4. Storage Area Networks (SAN).....	11
5. Disk Controller.....	11
<i>Database Considerations.....</i>	11
1. Microsoft SQL Server Updates.....	11
2. Microsoft SQL Server 2008 R2 Service Pack 1 or later.....	11
3. Backup Strategy.....	12
4. Database Maintenance Plans.....	12
5. Priority Boost.....	12
6. Parallelism.....	12
7. Update Statistics.....	14
8. Blocking.....	14
9. Deadlocking.....	14
10. Database Triggers.....	15
11. Indexes.....	15
12. PJOURNAL jobs.....	16
13. Offload Reporting.....	16
<i>Network Considerations.....</i>	17
1. TCP Chimney.....	17
2. Network Adapter.....	17
3. Local Area Network (LAN)/Wide Area Network (WAN) Considerations.....	18
Optimizing and Maintaining Client and Terminal Server Performance.....	18
<i>Client and Terminal Server Operating System Considerations.....</i>	18
1. Microsoft Dynamics GP System Requirements.....	18
2. Windows Updates.....	18
3. Anti-Virus Software.....	19
4. RemoteApp.....	19

5. Power Save Options.....	19
6. Terminal Server.....	19
7. BIOS.....	19
Network Considerations.....	20
1. TCP Chimney.....	20
2. Local Area Network (LAN)/Wide Area Network (WAN) Considerations.....	21
3. User Profile Home Path Configuration.....	21
Microsoft Dynamics GP Client Considerations.....	21
1. Service Packs.....	21
2. Modified Reports and Forms.....	22
3. ODBC Configuration.....	22
4. Dex.ini Paths and Settings.....	23
Optimizing Performance for Common Microsoft Dynamics GP Processes.....	23
Login Performance.....	23
1. Reminders.....	23
2. Internet Access.....	23
3. User Profile Home Path Configuration.....	24
4. Shortcuts.....	24
5. Metrics.....	24
Reporting Performance.....	24
1. SQL Server Reporting Services.....	24
2. Report Writer Reports.....	24
3. Offloading Reporting.....	24
Posting Performance.....	25
1. PJOURNAL jobs.....	25
2. Analytical Accounting Impact.....	25
3. Inventory Average Costing Impact.....	25
SmartList Performance.....	25
1. SmartList Usage.....	25
2. SmartList Search Criteria.....	25
3. Default SmartList Objects vs SmartList Builder Objects.....	26
4. SmartLists with Extender Data Added.....	26
5. Export of SmartList Data to Microsoft Excel.....	26
Audit Trails Performance.....	26
Receivables Management Performance.....	26
Fixed Assets General Ledger Posting Routine Performance.....	27
Other Performance.....	27
1. Process Servers.....	27
Distribution focused Performance issues.....	27
1. Project Accounting Performance.....	27
2. Purchase Order Performance.....	27
3. Inventory Performance.....	28
4. Inventory Average Costing Impact.....	28

Analysis Tools.....	29
<i>Analysis Tools for Troubleshooting Performance.....</i>	29
1. SQL Server Profile Trace.....	29
2. Performance Monitor.....	29
3. Performance Analysis of Logs (PAL).....	29
4. SQL Diag.....	30
5. Process Monitor.....	30
6. Performance Analyzer Version 1.01 for Microsoft Dynamics.....	30
7. Microsoft Dynamics GP Client Logging Tools.....	30
8. Network Monitor.....	31
<i>Analysis Tools Used for Specific Performance Issues.....</i>	31
1. Blocking.....	31
2. Deadlocking.....	31
3. Long Running Queries.....	31
4. Hardware Bottlenecks.....	31
5. Networking.....	32
6. External Influences.....	32
Common Bottlenecks.....	32
Advanced Troubleshooting Techniques for Performance.....	36
Resources.....	39

Introduction

The purpose of this white paper is to complement, rather than replace, existing resources that are specific to optimizing and maintaining the components of a Microsoft Dynamics GP implementation. This white paper offers relevant information on maintaining and optimizing a Microsoft Dynamics GP environment as well as links to related resources that may offer additional guidance. This white paper also offers a starting point for troubleshooting performance issues with Microsoft Dynamics GP. If you are experiencing performance issues with Microsoft Dynamics GP, please review this white paper to ensure your environment meets the recommendations provided.

Important: The optimization techniques and performance improvements offered in this paper are based on a Microsoft Dynamics GP implementation with a standard configuration. When considering the applicability of the information offered in this paper to a specific implementation, be sure to keep in mind the following points:

- Depending on the level of customization in a specific Microsoft Dynamics GP implementation, these techniques may perform differently or yield varying results.
- Verify the functionality and performance impact of any of these optimization techniques before implementing them in a production environment.

Important: Create backup copies of all databases before performing any of the optimization techniques described in this white paper.

If you have basic questions on the content in the white paper, please contact Microsoft using the links below:

Customers:

<https://mbs.microsoft.com/customersource/northamerica/GP/support>

Partners:

<https://mbs.microsoft.com/partnersource/northamerica/support>

If you would like assistance applying the recommendations to your system or would like assistance with continued performance issues following the review of the white paper, please contact your Microsoft Dynamics Partner, or create a support case asking for assistance and it will be reviewed for a billable service.

If you have questions specifically what performance related issue can be supported in a support case or what requires an advisory service please refer to the following article:

Customers:

Support Boundaries for Microsoft Dynamics GP – Americas

<https://mbs.microsoft.com/customersource/northamerica/GP/support/support-news/MSDgpSupboundariesAmericas>

Partners:

Support Boundaries for Microsoft Dynamics GP – Americas

<https://mbs.microsoft.com/partnersource/northamerica/support/support-news/MSDgpSupboundariesAmericas>

Optimizing and Maintaining Microsoft SQL Server Performance

Server Operating System Considerations

1. Microsoft Dynamics GP System Requirements

Verify the server operating system meets the system requirements defined based on transaction volume.

GP 2010

[System Requirements for Microsoft Dynamics GP 2010](#)

[System Requirements for Microsoft Dynamics GP 2010 Web Applications](#)

GP 2013

[System Requirements for Microsoft Dynamics GP 2013](#)

[System Requirements for Microsoft Dynamics GP 2013 Web Client](#)

GP 2015

[System Requirements for Microsoft Dynamics GP 2015:](#)

[System Requirements for Microsoft Dynamics GP 2015 Web Components:](#)

GP 2016

[System Requirements for Microsoft Dynamics GP 2016:](#)

[System Requirements for Microsoft Dynamics GP 2016 Web Components:](#)

GP 2018

[System Requirements for Microsoft Dynamics GP 2018](#)

[System Requirements for Microsoft Dynamics GP 2018 Web Components](#)

2. Windows Updates

Install all required and recommended Windows Updates on the server operating system.

How to keep your Windows computer up-to-date

<https://support.microsoft.com/en-us/help/311047/how-to-keep-your-windows-computer-up-to-date>

3. Windows Server 2008 R2 Service Pack 1 or later

Service Pack 1 or later for Windows Server 2008 R2 is required if experiencing any type of performance issue. Service Pack 1 includes the following important performance hotfixes:

- An application stops responding, experiences low performance, or experiences high privileged CPU usage if many large I/O operations are performed in Windows 7 or Windows Server 2008 R2:
<http://support.microsoft.com/kb/976700>
- Poor performance occurs on a computer that has NUMA-based processors and that is running Windows Server 2008 R2 or Windows 7 if a thread requests lots of memory that is within the first 4 GB of memory:
<http://support.microsoft.com/kb/2155311>
- A computer that is running Windows 7 or Windows Server 2008 R2 becomes unresponsive when you run a large application:
<http://support.microsoft.com/kb/979149>

The article below explains one of the performance issues addressed with a hotfix that is included in Service Pack 1.

Unexplained slowness in SQL 2008 on Windows 2008 R2:

<http://blogs.msdn.com/b/repltalk/archive/2010/10/27/unexplained-slowness-in-sql-2008-on-windows-2008-r2.aspx>

4. Hyper-threading

Hyper-threading is a technology used to improve parallelization of computations (doing multiple tasks at once) performed on machines processors. It is recommended to turn hyper-threading off in the BIOS of the operating system. While there may be times when enabling hyper-threading can aid in performance there have been cases where it has significantly decreased performance. Contact your Hardware Vendor for assistance with turning off hyper-threading.

SQL Server support in a hyper-threaded environment:

<http://support.microsoft.com/kb/322385>

"The performance of hyper-threaded environments varies. For example, applications that cause elevated levels of contention can cause **decreased performance in a hyper-threaded environment.**"

SQL Server Optimizing SQL Server CPU Performance:

<http://technet.microsoft.com/en-us/magazine/2007.10.sqlcpu.aspx>

"On SQL Server systems, the DBMS handles its own extremely efficient queuing and threading to the OS, so hyper-threading only serves to overload the physical CPUs on systems with already high CPU utilization. When SQL Server queues multiple requests to perform work on multiple schedulers, the OS must switch the context of the threads back and forth on the physical processors to satisfy the requests that are being made even if the two logical processors are sitting on top of the same physical processor. **If you are seeing Context Switches/sec higher than 5000 per physical processor you should strongly consider turning off hyper-threading on your system and retesting performance.**"

5. x64 vs. x86 (32-bit)

Implementing an x64 operating system as well as SQL Server will achieve greater performance and scalability. Dramatic improvements are available with memory utilization and availability with x64.

Refer to the article below for more information about the benefits of an x64 environment.
Will 64-bit increase the performance of my SQL Server application?
<http://blogs.msdn.com/b/mssqlisv/archive/2007/04/30/will-64-bit-increase-the-performance-of-my-sql-server-application.aspx>

If you are using a 32bit operating system with SQL Server you can optimize the amount of memory available to SQL Server via the Physical Address Extension (PAE), /3GB switch, or the Addressing Windowing Extensions (AWE) options. Refer to the chart below for recommendations based on the amount of memory available on a 32bit server. The information below does not apply to a x64 operating system.

RAM	/3GB switch	/PAE switch	AWE option
4 GB	Yes	No	No
>4 - 12 GB	Yes	Yes	Yes
>12+ GB	No	Yes	Yes

The optimal cutoff for the /3GB switch can vary between 8 GB and 16 GB thus the 12 GB is a general good rule of thumb. Refer to the following article for more information:
PAE and /3GB and AWE oh my...
<http://blogs.msdn.com/b/chadboyd/archive/2007/03/24/pae-and-3gb-and-awe-oh-my.aspx>

Refer to the articles below for more information about enabling the options above:

Memory Limits for Windows and Windows Server Releases
[http://msdn.microsoft.com/en-us/library/aa366778\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/aa366778(VS.85).aspx)

How to Enable Physical Address Extension (PAE)
[http://msdn.microsoft.com/en-us/library/windows/desktop/aa366796\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa366796(v=vs.85).aspx)

How to Enable Addressing Windowing Extensions (AWE)
<http://technet.microsoft.com/en-us/library/ms190961.aspx>

How to Enable /3GB Switch
[http://technet.microsoft.com/en-us/library/bb124810\(EXCHG.65\).aspx](http://technet.microsoft.com/en-us/library/bb124810(EXCHG.65).aspx)

6. Anti-Virus Software

The following exclusions must be configured if anti-virus software is running on the SQL Server:

- Exclude the location where the database files and log files are stored from the anti-virus software scan.
- Configure exceptions to exclude *.ldf, *.mdf, *.ndf and SQL Backup files from the anti-virus software scan.

Some anti-virus software may have scanning within the tcp/ip stack; if enabled performance may degrade. Contact your Anti-Virus Software Vendor for more information about this possibility.

7. BIOS

Install the latest BIOS update available for the server operating system from your Hardware Vendor.

Disk Configuration Considerations

1. Recommended Redundant Array of Independent disks (RAID) configuration

Refer to the System Requirements below for the recommended RAID configuration based on transaction volume.

GP 2010

[System Requirements for Microsoft Dynamics GP 2010](#)

[System Requirements for Microsoft Dynamics GP 2010 Web Applications](#)

GP 2013

[System Requirements for Microsoft Dynamics GP 2013](#)

[System Requirements for Microsoft Dynamics GP 2013 Web Client](#)

GP 2015

[System Requirements for Microsoft Dynamics GP 2015:](#)

[System Requirements for Microsoft Dynamics GP 2015 Web Components:](#)

GP 2016

[System Requirements for Microsoft Dynamics GP 2016:](#)

[System Requirements for Microsoft Dynamics GP 2016 Web Components:](#)

GP 2018

[System Requirements for Microsoft Dynamics GP 2018](#)

[System Requirements for Microsoft Dynamics GP 2018 Web Components](#)

2. Isolation of the database files

- **Log Data Files (LDF)**

It is recommended to always isolate the Microsoft Dynamics GP database log files on a dedicated RAID 1 or RAID 10 volume. This is to ensure that SQL Server can always write to the log file as fast as possible with no contention from the MDF Input/Output (I/O) activity. This will have a positive impact on the performance of SQL statements (inserts, updates, deletes) in the database because SQL Server can continue to the next statement as soon as the log is written.

- **Master Data Files (MDF)**

It is recommended to always isolate the Microsoft Dynamics GP database data files on a dedicated RAID 5 or RAID 10 volume. Ensure you have adequate disks to handle your I/O load, otherwise latency will increase once the disks are saturated.

- **TEMPDB (MDF and LDF)**

It is recommended to always isolate the TEMPDB database files (MDF and LDF) on a dedicated RAID 1 or RAID 10 volume. Performance may also benefit by creating 1 data file per CPU for TEMPDB as recommended in the article below.

Refer to the article below for the SQL Server Storage Top 10 Best Practices

<http://msdn.microsoft.com/en-us/library/cc966534.aspx>

3. Latency

Latency from the disk subsystem is one of the most common causes for performance issues. Proper placement of the database files across correctly sized RAID volumes will ensure a solid base to troubleshoot against. The following disk **latency** counters offer the best evidence of how well the disk subsystem is handling the requests from SQL Server.

Performance Monitor Counter	Description
Average Disk Sec/Read & Average Disk Sec/Write	Measure of disk latency. Lower values are better however this can vary and is dependent on the size and nature of the I/Os being issued. Numbers also vary across different storage configurations (cache size/utilization can impact this).

SQL Server Best Practices Article:

<http://technet.microsoft.com/en-us/library/cc966412.aspx>

“On well-tuned I/O subsystems, **ideal** values would be:

1–5 ms for Log (ideally 1 ms on arrays with cache)

4–20 ms for Data on OLTP systems (ideally 10 ms or less)”

Latency values and level of concern:

*.ldf (Avg Disk Sec/Write)		*.mdf (Avg Disk Sec/Read & Avg Disk Sec/Write)	
<10 ms	Low	<20 ms	Low
10-25 ms	Medium	20-50 ms	Medium
25-75 ms	High	50-150 ms	High
>75 ms	Severe	>150 ms	Severe

While the amount of memory can mask the disk I/O latency it cannot completely conceal it, as there will always be a point in time when SQL Server will need to save/access data from the disks. When the Average Disk Sec/Write and Average Disk Sec/Read values are high the focus should be on the following:

- Reducing the I/O load through tuning queries (assuming it is high and thus causing the latency).
- Make the appropriate changes to the disk subsystem to handle the I/O load.
- Verify the latency makes sense for the **volume** of I/Os being requested. When the I/O volume is much lower than what the disks can handle, and high latency is occurring, then further investigation of hardware, firmware, or potentially even operating system patches is warranted since the physical disks themselves cannot be the bottleneck, rather something between SQL Server and the physical disk.

Performance Counter	Description
Disk Reads/sec & Disk Writes/sec	Number of I/Os per second (IOPs) being issued against a particular disk or volume. This number varies based on the size of I/Os issued. Consult the Hardware Vendor for an estimation of the number of I/Os per second support per disk on their hardware.

4. Storage Area Networks (SAN)

Storage Area Networks (SAN) are becoming increasingly popular and configuration varies. It is important to ensure the SAN is handling the request with the same latency expectations as above. Be aware that one SAN can share the same physical disks to separate servers via creating separate Logical Unit Numbers (LUN) from the same RAID group. Thus, it is important to consider the I/O load from other servers if they are indeed sharing the same RAID group. This type of situation may present itself as high latency / low volume at the SQL Server at random times due to spikes in I/O requests coming from other servers. For this reason, it is recommended to have dedicated RAID groups for the Microsoft Dynamics GP SQL Server to isolate outside influences as much as possible. Be aware your Hardware Vendor may have their own best practices in relation to setting up their SAN for use with SQL Server. If I/O latency is constantly low, there may be no need to make any changes to the SAN.

5. Disk Controller

Install the latest firmware/driver for the disk/host bus adapters (HBA) controllers. Contact your Hardware Vendor for help if needed.

Database Considerations

1. Microsoft SQL Server Updates

It is recommended to install the latest service pack, hotfix, or cumulative updates for SQL Server.

2. Microsoft SQL Server 2008 R2 Service Pack 1 or later

If running SQL Server 2008 R2, Service Pack 1 or later is required if experiencing any type of performance issue.

3. Backup Strategy

Implementing a backup strategy is essential for disaster recovery and to protect critical data in SQL Server. Database backups can be scheduled and configured through a database maintenance plan. Plan your backup strategy based on the transaction volume and how much down time you can afford if a disaster should occur. Refer to the article below for disaster recovery options:

Description of disaster recovery options for Microsoft SQL Server
<http://support.microsoft.com/kb/822400>

4. Database Maintenance Plans

Once Microsoft Dynamics GP is installed and all company databases are created, a database maintenance plan should be configured for each database. Database maintenance plans create a workflow of the tasks required to make sure that your database is optimized, is regularly backed up, and is free of inconsistencies. The Maintenance Plan Wizard allows you to easily create an Integration Services package, which is run by a SQL Server Agent job. These maintenance tasks can be run manually or automatically at scheduled intervals. Refer to the article below for recommended maintenance plans for Microsoft Dynamics GP.

CustomerSource:

Recommended Maintenance with Microsoft SQL Server for Microsoft Dynamics GP Databases:
https://mbs.microsoft.com/customersource/northamerica/GP/learning/documentation/how-to-articles/MSD_GPRecommendedMaintenanceSQLDatabases

PartnerSource:

Recommended Maintenance with Microsoft SQL Server for Microsoft Dynamics GP Databases:
https://mbs.microsoft.com/partnersource/northamerica/deployment/documentation/how-to-articles/MSD_GPRecommendedMaintenanceSQLDatabases

5. Priority Boost

The "Boost SQL Server Priority" option in the SQL Server Management Studio SQL Server Properties window should **not** be enabled. If enabled, this option can starve other processes including basic kernel I/O activity, thus defeating the purpose of trying to enhance SQL Server performance.

How to determine proper SQL Server configuration settings:
<http://support.microsoft.com/kb/319942>

"Based on actual support experience, you do not need to use priority boost for reliable performance. If you do **use priority boost, it can interfere with smooth server functioning under some conditions and you should not use it except under very unusual circumstances.** For example, Microsoft Product Support Services might use priority boost when they investigate a performance issue."

6. Parallelism

It is recommended to set the "Max Degree of Parallelism" option in the SQL Server Management Studio SQL Server Properties Advanced window to equal 1 in a SQL Server environment when used with Microsoft Dynamics GP. This setting will still use all available CPUs; however, each SQL Statement will be isolated to 1 CPU for its execution plan.

SQL Max Degree of Parallelism (General Recommendation for Microsoft Dynamics GP)

Summary of reasons behind the recommendation to set Max Degree of Parallelism = 1:

- Microsoft Dynamics GP is an online transaction processing (OLTP) application as such we want SQL Server to balance the concurrency workload.
- All users are less affected by sporadic high cost queries.
- In rare cases, a parallel query can run very slow compared to the execution plan with 1 CPU.
- Deadlock situations may occur for a single SPID across multiple ECIDs (CPUs).
- Execution plan analysis is simplified.

Top SQL Server 2005 Performances Issues for OLTP Applications

<http://technet.microsoft.com/en-us/library/cc966401.aspx>

"Given the high volumes of OLTP, **parallel queries usually reduce OLTP throughput** and should be avoided."

Troubleshooting Performance Problems in SQL Server 2005

<http://technet.microsoft.com/en-us/library/cc966540.aspx>

"Running with a parallel plan is not inherently bad and should provide the fastest response time for that query. However, the response time for a given query must be weighed against the overall throughput and responsiveness of the rest of the queries on the system. Parallel queries are generally best suited to batch processing and decision support workloads and **might not be desirable in a transaction processing environment.**"

SQL Max Degree of Parallelism (Alternate Recommendation based on general SQL)

Even though with Microsoft Dynamics GP the general recommendation is to set Max Degree of Parallelism to 1, some environments may differ based on hardware / load. The following article offers alternate recommendations from a general SQL perspective.

Summary of reasons behind alternate recommendations:

- Certain queries may benefit from a parallelized execution plan, especially complex ones that are not already bottlenecked by disk latency / CPU
- Complex reporting benefits

Recommendations and guidelines for the "max degree of parallelism" configuration option in SQL Server

<https://support.microsoft.com/en-us/help/2806535/recommendations-and-guidelines-for-the-max-degree-of-parallelism-confi>

"Note the MAXDOP option does not limit the number of processors that SQL Server uses. Use the affinity mask configuration option to configure the number of processors that SQL Server uses.

Use the following guidelines when you configure the MAXDOP value:

- For servers that use more than eight processors, use the following configuration: MAXDOP=8.

- For servers that have eight or less processors, use the following configuration where N equals the number of processors: MAXDOP=0 to N.
- For servers that have NUMA configured, MAXDOP should not exceed the number of CPUs that are assigned to each NUMA node.
- For servers that have hyper-threading enabled, the MAXDOP value should not exceed the number of physical processors.”

7. Update Statistics

Update statistics on a table or indexed view is crucial to ensure the query optimizer for SQL Server uses the most up to date information for the query optimization process.

To ensure the statistics are up to date, it is recommended to keep the Auto Create Statistics and Auto Update Statistics database setting enabled for all Microsoft Dynamics GP databases in the SQL Server Management Studio Database Properties Options window. These options by default are enabled.

8. Blocking

Blocking occurs when one connection to SQL Server locks one or more records, and a second connection to SQL Server requires a conflicting lock type on the record or records locked by the first connection. This causes the second connection to wait until the first connection releases its locks. By default, a connection will wait an unlimited amount of time for the blocking lock to go away. Short duration blocking is normal for database consistency. However, if one process is holding long duration locks, it has the potential to block several other processes/users for extended periods of time. Long duration blocking should be investigated and resolved if possible. Refer to the article below for more information on blocking:

Understanding and resolving SQL Server blocking problems
<http://support.microsoft.com/kb/224453/EN-US>

9. Deadlocking

A deadlock occurs when two or more tasks permanently block each other by each task having a lock on a resource which the other tasks are trying to lock. At which point, SQL Server will choose to stop one of the blocking processes. Deadlocking is more prevalent in a poorly performing environment due to longer transaction completion time. Refer to the articles below for more information on deadlocking:

Deadlock Explanation
<http://msdn.microsoft.com/en-us/library/ms177433.aspx>

Detecting and Ending Deadlocks
<http://msdn.microsoft.com/en-us/library/ms178104.aspx>

eConnect Deadlocks

In some situations customers who are using eConnect to import transactions into Microsoft Dynamics GP will run into deadlock errors. Unfortunately, there is no way to completely eliminate deadlocks with eConnect, since we cannot get around the fact that both Microsoft Dynamics GP and eConnect will require exclusive locks on tables.

Any time you are posting documents in Microsoft Dynamics GP and have eConnect importing transactions from multiple users, you could definitely run into deadlocks depending on the timing of those events. We have a few recommendations that we typically make to try to minimize the frequency of the deadlocks. Please review the below items and determine what applies to your environment.

1. Make sure posting is not going on in Microsoft Dynamics GP at the same time you are running the eConnect integration. The posting process in GP locks several tables that eConnect also hits when creating orders. You should not be importing eConnect transactions when posting is occurring in the back office, mainly because we can't get around the fact that both processes require an exclusive lock on several tables for the batches.

2. Reduce the XML files to contain only 1 transaction per XML file.
When eConnect processes an XML file it creates a SQL transaction for the entire XML document. So if there are multiple documents in one XML file, this transaction will run longer and potentially hold up tables longer, dramatically increasing your chances of hitting a deadlock situation. We've always recommended that eConnect be called with only 1 Microsoft Dynamics GP document per API call.

3. Reduce the number of machines running this eConnect service. For example, load the service on one machine (or the server) and have the users all send their XML files to that location. In that manner, the process will become "single-threaded" and we'll only have 1 active transaction at a time happening in SQL. You mentioned you are running a web app, so I would assume this is already running on only 1 server.

4. Along the same line as #2, if you are using message queuing, load the eConnect Incoming Service on 1 box (or the server) and then have the client apps simply sending the XML file to an MSMQ message queue on that server. This would also take the process down to running "single-threaded" the same as above.

5. Ensure that the line items in these orders are all sorted by ITEMNMBR (if applicable). In the past where we've seen deadlocks on the IV00102 table when updating quantities. Our developers have determined that it's possible for 2 SOP orders with the same 2 items on them to deadlock if the order of the items is reversed from one SOP document to another. Ensuring that the sorting of the line items is consistent across all SOP documents coming in, should help with any deadlocks that you might be getting on the IV00102 table.

10. Database Triggers

The use of table triggers may impact performance by increasing I/O or increased blocking times since the trigger is part of the original statement's transaction. If implementing table triggers, careful planning and testing is required to ensure performance is not degraded.

11. Indexes

Default indexes on Microsoft Dynamics GP tables should not be changed or removed. Careful planning is necessary when implementing new indexes to avoid performance decreases. Adding indexes can adversely affect performance due to the overhead of maintaining the index as well as

negatively affecting other execution plans.

12. PJOURNAL jobs

The PJOURNAL table captures records as posting occurs in Microsoft Dynamics GP. This table can grow to be very large depending on posting volume. The Microsoft Dynamics GP installation creates a SQL Server Agent Job to truncate the PJOURNAL table in each company. Verify the SQL Server Agent service is running in the SQL Server Management Studio. Expand SQL Server Agent in the SQL Server Management Studio and verify the "Remove Posted PJOURNALS From All Companies" job exists and is enabled under the Jobs folder. By default, the table is cleared every 30 minutes. If the "Remove Posted PJOURNALS From All Companies" job does not exist, execute the PJJOB.SQL script in the SQL Server Management Studio to create the jobs. The PJJOB.SQL script can be found in the Microsoft Dynamics\GP\SQL\Util folder.

13. Offload Reporting

For complex reporting needs, running reports on a separate SQL Server is a good practice. This eliminates the reporting load on the production database server. Several options are available to configure the SQL Server databases on a reporting server. Refer to the article below for requirements for Microsoft Dynamics GP:

Description of the requirements to run replication, clustering, log shipping, and database mirroring together with Microsoft Dynamics GP

<http://support.microsoft.com/kb/926490>

14. Maximum Server Memory

It is suggested to configure the Maximum Server Memory option on the SQL Server instance to ensure there is an appropriate amount of memory available for the operating system. Below is one calculation that is available to help with figuring out the appropriate settings. Even if your SQL Server is not experiencing >80% memory usage setting the Maximum Server Memory setting correctly can positively impact performance.

"Max Server Memory"

MAX_SERVER_MEMORY = TOTAL_SERVER_MEMORY - [OS/Apps Memory] - [Threads * Thread Size]

TOTAL_SERVER_MEMORY = 4GB

OS/Apps Memory = 2 GB to 4 GB (As a rule use 2GB on system with <= 16GB of RAM and 4GB on systems with > 16GB of RAM)

Threads = `SELECT max_workers_count FROM sys.dm_os_sys_info`

(This allows us to factor in the activity level of the server, this example uses 255)

Thread Size =

Platform	Size
x86	512k
x64	2MB
IA64	4MB

Example: x64 SQL Server with 4GB of RAM

MAX_SERVER_MEMORY = (4GB - (2GB - (255 * 2MB)))

MAX_SERVER_MEMORY = (4GB - (2GB - (510MB)))
MAX_SERVER_MEMORY = (4GB - (1.5GB))
MAX_SERVER_MEMORY = (2.5GB)

To set Maximum Server Memory right-click on the SQL Server instance name in SQL Server Management Studio and select **Properties**, then the **Memory** tab. It will be set to 2147483647MB by default, which means that SQL Server can use all the memory resources if necessary.

Network Considerations

1. TCP Chimney

It is recommended to disable TCP Chimney at all servers and workstations for best network performance. TCP Chimney must be disabled in the following areas:

- Disable at the operating system level
- Disable various Offload options at the network interface card (NIC) level

TCP Chimney is designed to offload certain tasks that the CPU typically would handle to the NIC card. Having these settings enabled may cause database connections to be dropped in turn causing various communications errors, such as "DBMS 12" or "## Object Does Not Exist" in Microsoft Dynamics GP. It can also show up as performance issues due to network delays such as noted in the following blog article:

TCP Chimney Offload – Possible Performance and Concurrency Impacts to SQL Server Workloads
<http://blogs.msdn.com/b/psssql/archive/2008/10/01/windows-scalable-networking-pack-possible-performance-and-concurrency-impacts-to-sql-server-workloads.aspx>

"We've also identified situations where TCP Chimney has impacted transaction throughput and **caused delays between when a statement has been completed by the SQL engine and the time to receive the begin event of the next statement.** This impact can be significant especially in application workloads that have throughput requirements to execute a series of statements within a certain time boundary."

Refer to the articles below to disable TCP Chimney:

How to Disable TCP Chimney in Windows Server 2008
<http://support.microsoft.com/kb/951037>

How to Disable TCP Chimney in Windows 2003 Server
<http://support.microsoft.com/kb/942861>

2. Network Adapter

It is recommended to configure a single dedicated network adapter card for the SQL Server. Ensure the network adapter firmware/driver is up-to-date.

3. Local Area Network (LAN)/Wide Area Network (WAN) Considerations

In a Wide Area Network environment, a Terminal Server is required for Microsoft Dynamics GP. The Terminal Server must be connected to the SQL Server with a high-speed LAN connection.

Optimizing and Maintaining Client and Terminal Server Performance

Client and Terminal Server Operating System Considerations

1. Microsoft Dynamics GP System Requirements

Verify the client and Terminal Server operating system meets the system requirements defined below:

GP 2010

[System Requirements for Microsoft Dynamics GP 2010](#)

[System Requirements for Microsoft Dynamics GP 2010 Web Applications](#)

GP 2013

[System Requirements for Microsoft Dynamics GP 2013](#)

[System Requirements for Microsoft Dynamics GP 2013 Web Client](#)

GP 2015

[System Requirements for Microsoft Dynamics GP 2015:](#)

[System Requirements for Microsoft Dynamics GP 2015 Web Components:](#)

GP 2016

[System Requirements for Microsoft Dynamics GP 2016:](#)

[System Requirements for Microsoft Dynamics GP 2016 Web Components:](#)

GP 2018

[System Requirements for Microsoft Dynamics GP 2018](#)

[System Requirements for Microsoft Dynamics GP 2018 Web Components](#)

2. Windows Updates

Install all required and recommended Windows Updates on the operating system.

3. Anti-Virus Software

The following exclusions must be configured if anti-virus software is running on the client workstations or the Terminal Server:

- Configure exceptions to exclude *.tmp, *.idx, *.dat, TNT*. * files in the user's profile temp folder from the anti-virus scan.
- In some anti-virus programs, an exception for the Dynamics.exe process may be necessary.

4. RemoteApp

If using the RemoteApp feature with Windows Server 2008, the latest operating system updates must be installed to ensure Microsoft Dynamics GP will function properly with RemoteApp.

5. Power Save Options

Ensure power save options, such as hibernate or sleep, are not enabled at each client workstation where the Microsoft Dynamics GP client is running. This is to prevent SQL Server from auto-closing inactive connections which would remove TEMPDB objects Microsoft Dynamics GP is using at the time.

6. Terminal Server

Do not limit the amount of time that active, disconnected, and idle (without user input) sessions allowed on the server. It is important to leave any active Microsoft Dynamics GP clients running remotely intact. Data corruption can occur if Microsoft Dynamics GP is abruptly shut down as several windows have code on the window close event to complete data processing. Refer to the article below for more information regarding Terminal Server timeout and reconnection settings:

Configure Timeout and Reconnection Settings for Terminal Services Sessions
[http://technet.microsoft.com/en-us/library/cc754272\(Ws.10\).aspx](http://technet.microsoft.com/en-us/library/cc754272(Ws.10).aspx)

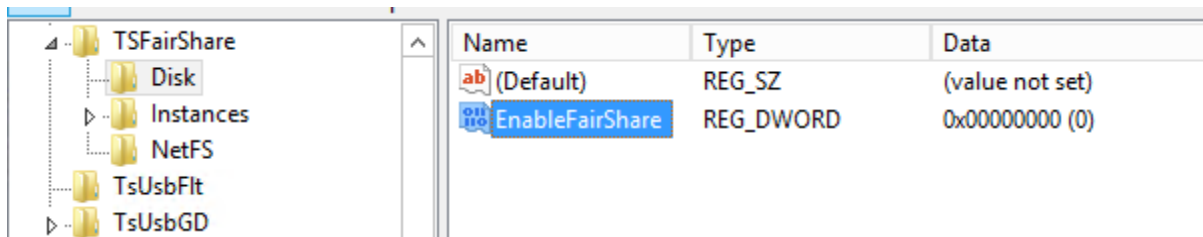
7. BIOS

Install the latest BIOS update available for the operating system from your Hardware Vendor.

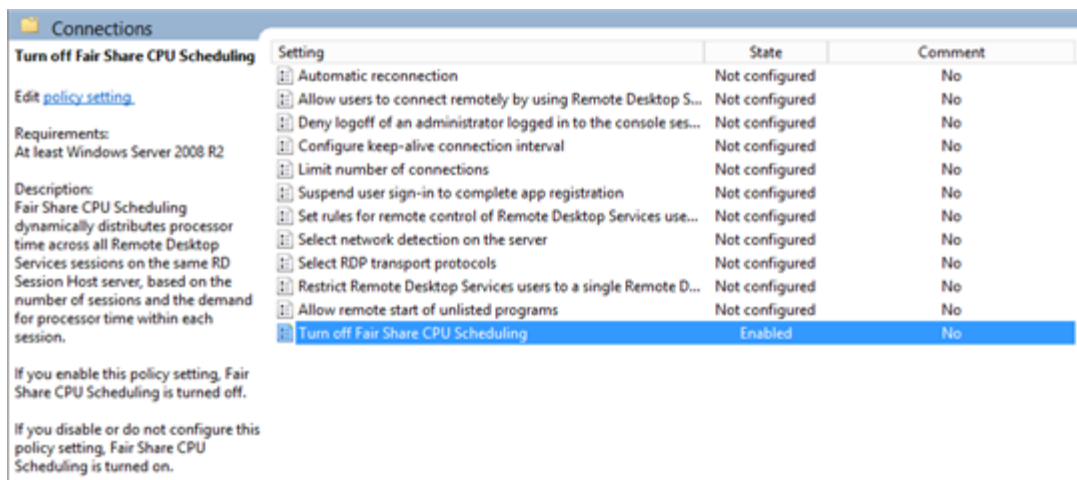
8. Tuning Remote Desktop Services 2012 and higher

The Remote Desktop Services feature was rewritten in Windows Server 2012, specifically around how the available resources on the server are pooled to allow multiple users to log into the machine at once. As part of this pooling a functionality called TSFairShare was introduced to manage how resources are divided up. This setting allows you to tune the pooling behavior and has resolved performance issues for a couple of our Dynamics GP customers. It is just that tuning, which can be disabled using the following steps, that you can try to address performance issues you are seeing over remote connections.

First, in the registry (Start > Run > regedit) you will want to go to **HKLM\System\CurrentControlSet\Services\TSFairShare** and set **EnableFairShare** to 0 for both **Disk** and **NetFS** options. As with all registry changes you will want to be sure that you made a backup of the affected keys before you do this.



Second, open the Group Policy Editor on the RDS server (Start > Run > gpedit.msc) and navigate to **Computer Configuration/Administrative Templates/Windows Components/Remote Desktop Services/Remote Desktop Session Host/Connections**. There we enabled the '**Turn off Fair Share CPU Scheduling**' setting to disable this feature.



Reboot the machine after this change and test your issue again to see if performance has improved.

Network Considerations

1. TCP Chimney

It is recommended to disable TCP Chimney at all servers and workstations for best network performance. TCP Chimney must be disabled in the following areas:

- Disable at the operating system level
- Disable various Offload options at the NIC level

TCP Chimney is designed to offload certain tasks that the CPU typically would handle to the NIC card. Having these settings enabled may cause database connections to be dropped in turn causing various communications errors, such as "DBMS 12" or "## Object Does Not Exist" in Microsoft Dynamics GP. It can also show up as performance issues due to network delays such as noted in the following blog article:

TCP Chimney Offload – Possible Performance and Concurrency Impacts to SQL Server Workloads
<http://blogs.msdn.com/b/psssql/archive/2008/10/01/windows-scalable-networking-pack-possible-performance-and-concurrency-impacts-to-sql-server-workloads.aspx>

"We've also identified situations where TCP Chimney has impacted transaction throughput and **caused delays between when a statement has been completed by the SQL engine and the time to receive the begin event of the next statement**. This impact can be significant especially in application workloads that have throughput requirements to execute a series of statements within a certain time boundary."

Refer to the following articles to disable TCP Chimney:

- How to Disable TCP Chimney in Windows Server 2008 and Windows 7:
<http://support.microsoft.com/kb/951037>
- How to Disable TCP Chimney in Windows 2003 Server and Windows Vista
<http://support.microsoft.com/kb/942861>

2. Local Area Network (LAN)/Wide Area Network (WAN) Considerations

In a Wide Area Network environment, a Terminal Server is required for Microsoft Dynamics GP. The Terminal Server must be connected to the SQL Server with a high-speed LAN connection.

3. User Profile Home Path Configuration

Microsoft Dynamics GP performance can be adversely affected if the user's profile is setup to use a Home Path pointed to a network share, especially on network shares with slow connection speeds between the client workstation and the shared folder.

Microsoft Dynamics GP Client Considerations

1. Service Packs

It is recommended to install the latest service pack, hotfix, or compliance update for Microsoft Dynamics GP to ensure you have the most recent performance fixes. Refer to the links below to download the latest patch releases for Microsoft Dynamics GP:

GP 2010

[Service Pack, Hotfix, and Compliance Update Patch Releases for Microsoft Dynamics GP 2010](#)

GP 2013

[Service Pack, Hotfix, and Compliance Update Patch Releases for Microsoft Dynamics GP 2013](#)

GP 2015

[Service Pack, Hotfix, and Compliance Update Patch Releases for Microsoft Dynamics GP 2015](#)

GP 2016

[Service Pack, Hotfix, and Compliance Update Patch Releases for Microsoft Dynamics GP 2016](#)

GP 2018

[Service Pack, Hotfix, and Compliance Update Patch Releases for Microsoft Dynamics GP 2018](#)

2. Modified Reports and Forms

Placement of the modified reports and forms may affect performance when printing reports or accessing modified forms in Microsoft Dynamics GP. If you experience performance issues printing modified reports or accessing modified forms, consider the following:

- If the modified reports and forms are placed in a network share, copy the files locally. Change the path in the Dynamics.set file to point to the local files and test out the modified reports and forms again. The Dynamics.set file is found in the Microsoft Dynamics\GP folder and can be opened in a text editor. Having the modified reports and forms placed in a network share may increase latency and file contention thus causing performance issues in Microsoft Dynamics GP.
- If the client workstations or Terminal Server operating system is x64 and the modified reports and forms are placed in a network share, verify the operating system where the modified reports and forms are found is also x64.
- If the modified reports and forms are placed locally, verify a good administration/management strategy is implemented to ensure all local modified reports and forms are consistent when new modifications/changes are made.

3. ODBC Configuration

It is recommended to use the ODBC driver that correlates to the SQL Server version used with the Microsoft Dynamics GP databases. Using the same version will take advantage of potential enhancements and ensure stability. The SQL Native Client driver correlates to Microsoft SQL Server 2005 and the SQL Server Native Client 10.0 driver correlates to Microsoft SQL Server 2008 or Microsoft SQL Server 2008 R2. Refer to the links below to download the right ODBC driver:

SQL Server Native Client (SQL Server 2005)

<http://www.microsoft.com/download/en/details.aspx?DisplayLang=en&id=24793>

SQL Server Native Client 10.0 (SQL Server 2008 or SQL Server 2008 R2)

<https://www.microsoft.com/en-us/download/details.aspx?id=27596>

Note: It is the following file that you want to download: ENU\IA64\sqlncli.msi

Refer to the article below for the steps to configure an ODBC DSN for use with Microsoft Dynamics GP:

Customers:

How to Setup an ODBC Data Source on Microsoft SQL Server for Microsoft Dynamics GP

https://mbs.microsoft.com/customersource/northamerica/GP/learning/documentation/how-to-articles/MDGP_HOWTO_SETUP_ODBC

Partners

How to Setup an ODBC Data Source on Microsoft SQL Server for Microsoft Dynamics GP
https://mbs.microsoft.com/partnersource/northamerica/deployment/documentation/how-to-articles/MDGP_HOWTO_SETUP_ODBC

4. Dex.ini Paths and Settings

Dexsql.log

When troubleshooting issues in Microsoft Dynamics GP, it is common to use the Dexsql.log. However, once troubleshooting is complete ensure the following entries are set back to FALSE to disable the Dexsql.log. If the Dexsql.log continues to run, performance issues may occur as the user works in Microsoft Dynamics GP.

SQLLogSQLStmt=FALSE

SQLLogODBCMessages=FALSE

SQLLogAllODBCMessages=FALSE

OLE Path

If the OLE Path is placed on a shared location and the network share is on a slow connection or the network share path is non-existent, users may encounter performance issues opening windows and retrieving records if an OLE note exists for that window or record. This is due to Microsoft Dynamics GP trying to retrieve the OLE note information from the OLEPath= path specified in the Dex.ini.

Optimizing Performance for Common Microsoft Dynamics GP Processes

NOTE: Please verify the recommendations in the earlier section titled "Microsoft Dynamics GP Client Considerations" are in place before reviewing the recommendations below.

Login Performance

There are many areas to take into consideration when the login to Microsoft Dynamics GP is performing slowly.

1. Reminders

All custom reminders in Microsoft Dynamics GP are based on SmartList queries. If the SmartList query that the reminder is based on runs slow in SmartList, the query will also run slow upon login into Microsoft Dynamics GP. If the user would like to improve login performance, they can remove unnecessary or slow running custom Reminders. In Microsoft Dynamics GP, click Microsoft Dynamics GP, click User Preferences, and then click Reminders. Remove the reminders from the Custom Reminders area.

2. Internet Access

In certain environments, revoking access to the Internet may be a security policy requirement. When launching Microsoft Dynamics GP, there are calls to view certain sites on the Internet. If the Internet is unavailable, the login into Microsoft Dynamics GP may appear to be slow.

If the Internet access must be revoked, follow the steps below to reduce the performance issue at login:

- a. Go to the Microsoft Dynamics GP folder and make a backup copy of the Dynamics.exe.config file.
- b. Open the Dynamics.exe.config file into Notepad.
- c. Following the `</shell>` tag and before the `</configuration>` tag, copy in the following:

```
<runtime>  
<generatePublisherEvidence enabled="false"/>  
</runtime>
```
- d. Close and save the Dynamics.exe.config file.
- e. Repeat Steps a-c at each client workstation or Terminal Server.

3. User Profile Home Path Configuration

Microsoft Dynamics GP login performance can be adversely affected if the User's Profile is setup to use a Home Path pointed to a network share, especially on network shares with slow connection speeds between the client workstation and the shared folder.

4. Shortcuts

Verify all external shortcuts on the Shortcut bar in Microsoft Dynamics GP are valid links. At login, Microsoft Dynamics GP will validate the external links and if the links are not valid the login may slow down.

5. Metrics

Metrics that have slow performing queries may increase login times. If the user would like to improve login performance, they can remove unnecessary or slow running Metrics. In Microsoft Dynamics GP, click the Change Details icon on the Metrics Title Bar. Remove the Metrics from the Metrics to Display area.

Reporting Performance

1. SQL Server Reporting Services

Utilize SQL Server Reporting Services reports if possible. The logic is SQL based and reports are faster than Dexterity based Report Writer reports.

2. Report Writer Reports

Modified reports may drastically change reporting performance. For example, if additional tables are added to a report, depending on how they are linked it could cause performance issues. A good test would be to compare performance of the modified report against the default report.

3. Offloading Reporting

For complex reporting needs, running reports from a separate SQL server is a good practice. This eliminates the reporting load on the production database server. Several options are available to configure the SQL Server databases on a reporting server. Refer to the article below for requirements for Microsoft Dynamics GP:

Description of the requirements to run replication, clustering, log shipping, and database mirroring together with Microsoft Dynamics GP
<http://support.microsoft.com/kb/926490>

Posting Performance

1. PJOURNAL jobs

The PJOURNAL table captures records as posting occurs in Microsoft Dynamics GP. This table can grow to be very large depending on posting volume. The Microsoft Dynamics GP installation creates a SQL Server Agent Job to truncate the PJOURNAL table in each company. Verify the SQL Server Agent service is running in the SQL Server Management Studio. Expand SQL Server Agent in the SQL Server Management Studio and verify the "Remove Posted PJOURNALS From All Companies" job exists and is enabled under the Jobs folder. By default, the table is cleared every 30 minutes. If the "Remove Posted PJOURNALS From All Companies" job does not exist, execute the PJOB.SQL script in the SQL Server Management Studio to create the jobs. The PJOB.SQL script can be found in the Microsoft Dynamics\GP\SQL\Util folder.

2. Analytical Accounting Impact

Be aware if Analytical Accounting is installed and activated, the SQL Server workload will increase per distribution during posting routines. This is because Analytical Accounting is managing distribution breakdown information for every distribution even if Analytical Accounting codes are not assigned.

3. Inventory Average Costing Impact

Be aware if items are setup with the average cost valuation method, the SQL Server workload can increase during document posting. If a document causes an inventory cost change ripple across many inventory layers, significant differences in posting times can occur. For more information on average cost valuation, refer to the article below:

Enhancements made to the calculation of average cost in Microsoft Dynamics GP
<http://support.microsoft.com/kb/923960>

SmartList Performance

1. SmartList Usage

SmartList and SmartList Builder are intended to be used as a query tool only. If you decide to increase the maximum record count, be aware that query processing will be slow, and you may experience errors. Therefore, we recommend that you use the default record count limit of 1,000 when you execute queries.

2. SmartList Search Criteria

SmartList allows vast amounts of searching options that users can apply to the SmartList query data. With this flexibility comes the potential for very high I/O costs if the query / database design

was not optimized for a specific search criterion. Review the suggestions below to reduce I/O cost when designing SmartList search criteria:

- Avoid using the "is not equal to" filter
- Use the "equal to" instead of "contains" filter
- If the SmartList includes WORK, OPEN and HISTORY tables and the user is only interested in WORK data, use the "Document Status" or similar search criteria to skip searching OPEN and HISTORY tables.

3. Default SmartList Objects vs SmartList Builder Objects

Default SmartList may not always be the best choice as the query is generic. Given the amount of flexibility and amount of data to search it may be better to use a custom SmartList Builder that is perfected for the necessary results.

4. SmartList with Extender Data Added

Adding Extender fields from an Extender Window, Form, or Detail Form can cause a SmartList to return data slower. The linking needed to include the Extender data adds complexity to the SmartList. It may be beneficial to look at using SmartList Builder to build a custom SmartList that includes the Extender data to optimize the SmartList.

5. Export of SmartList Data to Microsoft Excel

The export of SmartList data to Excel can take some time depending on the amount of data you are trying to export. The number of columns included in the SmartList as well as the number of rows can influence the export performance. A SmartList exports to Excel field by field to ensure the formatting is also brought across with the data. Therefore, the export can take longer for all the data to appear in the Excel spreadsheet. If you are running SmartList to exporting the data to an Excel spreadsheet, you may want to consider the default Excel Reports that can be deployed in Microsoft Dynamics GP. They are setup like the default SmartList, however the data is directly pulled into Excel so that you do not have to export from SmartList. The added benefit to this is that the data is refreshable in Excel, so you do not have to continually export the data from SmartList.

Audit Trails Performance

Audit Trails increases the I/O workload on the SQL Server. Careful consideration is recommended when deciding on which tables to audit.

Receivables Management Performance

Paid Transaction Removal should be run periodically to transfer records from the OPEN to HISTORY table. This will help performance in various windows and reports. This is typically a monthly business process. Refer to the article below for recommended month-end procedures for receivables management:

Recommended month-end procedures for Open Item customers in Receivables Management in Microsoft Dynamics GP

<http://support.microsoft.com/kb/865910>

Fixed Assets General Ledger Posting Routine Performance

Avoid running the fixed assets general ledger posting routine with the period range current period to current period. The general recommendation is to go from "0000-000" to a current period. For more information, refer to the article below:

GL Interface (Posting) Takes a Long Time in Fixed Assets

<http://support.microsoft.com/kb/860140>

Other Performance

1. Process Servers

If the process server configuration is defined with a non-existent process server machine, sporadic delays may occur every 5-10 minutes due to the handshake with process server. Click Microsoft Dynamics GP, click Tools, click Setup, click System, and then click Process Servers. Click Server and verify all process servers are valid. Click Services and verify all process servers are valid.

Distribution focused Performance issues

1. Project Accounting Performance

You may experience a delay when you attempt to access Budget Maintenance or Budget Detail Entry within Project Accounting. Typically, this delay is caused because the Begin and End Date of your budget, project or contract is a wide span of time. There are periodic records created for each period within each year that the budget covers. These periodic records exist in 4 tables (PA01304, PA01221, PA01121, PA00511). These tables need to be read to produce the information in these windows so if there are a lot of records to read; opening the window will be delayed.

If you are not sure of the End Date on your project, we would recommend entering something realistic rather than a date way out into the future, (e.g. 2030) so that extra, unnecessary fiscal periods are not created. If you find you have Begin or End Dates that are not correct, change the date on the Budget. Once all budget dates are reset, the Project Dates will conform. Once dates on all the projects are reset, then the Contract Dates will conform.

Once changes have been made, run PA Recreate Periodic with the Option: Delete Periodic Records Outside Fiscal Years to remove those stranded periodic records that you no longer need.

2. Purchase Order Performance

- a. In **Print Purchasing Documents**, avoid running your POs by a Range of anything other than PO Number. Running the POs by Buyer ID, or vendor ID for example will cause significant delays in producing the POs since there is not an index on the table for these fields.

This has been written up as a Bug and is addressed in GP 2013.

- b. If you experience delays when creating Purchase Orders through **PO Generator**, and you are using a 3rd party product from Trinity Myridas called Advanced Inventory Replenishment, verify that the version of that code is compatible with your version of Microsoft Dynamics GP.
- c. If you experience a delay when you print **the Received/Not Invoiced** report, consider moving Closed and Cancelled POs from Work to History. This will reduce the number of records to be reviewed in the POP10500 table when the report is generated.

To move POs to history, first make sure you are keeping history by marking the Maintain History: Purchase Order checkbox in **Purchase Order Processing Setup**. Then use the **Remove Completed Purchase Orders** routine to move as many POs as possible to history.

To view this historical POs, make sure to mark "Historical Purchase Orders" in the Purchase Order Processing Document Inquiry and Purchase Order Processing Item Inquiry windows.

3. Inventory Performance

Be aware that if you override the quantities of your items in inventory that you will experience performance degradation when you run Reconcile Quantities in Inventory. This is because each override has at least 2 records (one in the IV10200 and another in the IV10201) that need to be verified, instead of just the one.

The Field Service reconcile has been added to the core Inventory reconcile starting with GP 10 SP2. If you are using Field Service, you can expect the IV Reconcile to take as long as running the separate IV Reconcile and the Field Service Reconcile did previously.

Also, unless you have recently made changes to the decimal places of your items, there is no benefit in marking the Include Item History checkbox in the Reconcile Inventory Quantities window. Leaving this box marked can slow down the reconcile process.

4. Inventory Average Costing Impact

Be aware if items are setup with the average cost valuation method, the SQL Server workload can increase during document posting. If a document causes an inventory cost change ripple across many inventory layers, significant differences in posting times can occur. For example, cost changes will occur when transactions are backdated. Since the average cost is date sensitive, backdating your document date will cause the cost to change. As a result, any outflows of that item dated after your document date will be revalued thus slowing the process. Even the backdate of one day can cause significant delays depending on the number of transactions involved.

Another thing that can cause performance issues when posting inventory related transactions is overriding quantities. Since when you override a quantity selling is done at an assumed cost, the

item will need to be revalued once the goods come in to fulfill the override at the actual cost.

Also, be aware that if you post a batch that has multiple different document types (e.g. Sales Invoices and Sales Returns) for the same item, that items in the batch post by Document Number order, and not Document Date. As a result, you could see an average cost change and subsequent ripple, due to the order of transactions in the batch.

For more information regarding average cost valuation, refer to the article below:

Enhancements made to the calculation of average cost in Microsoft Dynamics GP
<http://support.microsoft.com/kb/923960>

Analysis Tools

Analysis Tools for Troubleshooting Performance

1. SQL Server Profile Trace

Microsoft SQL Server Profiler is a graphical user interface to SQL Trace for monitoring an instance of the Database Engine.

How to use SQL Profiler to create an SQL trace in Microsoft SQL Server

<https://support.microsoft.com/en-us/help/912281/how-to-use-sql-profiler-to-create-an-sql-trace-in-microsoft-sql-server>

Additional Performance Overhead When Running the SQL Server Profile Trace Using a Client-Side Trace

<http://support.microsoft.com/kb/929728>

How to troubleshoot SQL Server performance issues

<http://support.microsoft.com/kb/298475>

2. Performance Monitor

Use Performance Monitor to monitor the utilization of system resources. Collect and view real-time performance data in the form of counters, for server resources such as processor and memory use, and for many Microsoft SQL Server resources such as locks and transactions.

Performance and Reliability Monitoring Step-by-Step Guide for Windows Server 2008

[http://technet.microsoft.com/en-us/library/cc771692\(WWS.10\).aspx](http://technet.microsoft.com/en-us/library/cc771692(WWS.10).aspx)

3. Performance Analysis of Logs (PAL)

The PAL tool reads in a performance monitor counter log (any known format) and analyzes it using complex, but known thresholds (provided). The tool generates an HTML based report that graphically charts important performance counters and throws alerts when thresholds are exceeded.

Using the Performance Analysis of Logs (PAL) Tool
[http://msdn.microsoft.com/en-us/library/cc296652\(v=BTS.10\).aspx](http://msdn.microsoft.com/en-us/library/cc296652(v=BTS.10).aspx)

4. SQL Diag

The SQLdiag utility is a general-purpose diagnostics collection utility that can be run as a console application or as a service. You can use SQLdiag to collect logs and data files from SQL Server and other types of servers, and use it to monitor your servers over time or troubleshoot specific problems with your servers. SQLdiag is intended to expedite and simplify diagnostic information gathering for Microsoft Customer Support Services.

SQLdiag Utility
<http://msdn.microsoft.com/en-us/library/ms162833.aspx>

5. Process Monitor

Process Monitor is an advanced monitoring tool for Windows that shows real-time file system, Registry, and process/thread activity. It combines the features of two legacy Sysinternals utilities, Filemon and Regmon, and adds an extensive list of enhancements including rich and non-destructive filtering, comprehensive event properties such as session IDs and user names, reliable process information, full thread stacks with integrated symbol support for each operation, simultaneous logging to a file, and much more.

Process Monitor v2.96
<http://technet.microsoft.com/en-us/sysinternals/bb896645>

6. Performance Analyzer Version 2.0 for Microsoft Dynamics

Performance Analyzer for Microsoft Dynamics is a set of scripts to collect and analyze performance information from the database and application tiers of Microsoft Dynamics products.

Performance Analyzer Version for Microsoft Dynamics
<http://dynamicsperf.codeplex.com/>

7. Microsoft Dynamics GP Client Logging Tools

Several Microsoft Dynamics GP client logging tools exist to capture detailed information on the specific processes and tasks taking place in Microsoft Dynamics GP.

Dexsql.log

The Dexsql.log is a trace log that captures all calls the Microsoft Dynamics GP client performs as a user works in Microsoft Dynamics GP.

How to create a Dexsql.log file to troubleshoot error messages in Microsoft Dynamics GP
<http://support.microsoft.com/kb/850996>

Script.log

The Script.log captures all the Dexterity code calls the Microsoft Dynamics GP client performs as a user works in Microsoft Dynamics GP.

How to gather a Dexterity Scriptlog in Microsoft Dynamics GP
<https://support.microsoft.com/en-us/help/3077859/how-to-gather-a-dexterity-scriptlog-in-microsoft-dynamics-gp>

Profile.txt

The Profile.txt file captures reads, inserts and updates for all SQL Server tables and offers statistical information for each statement.

How to use the Dexterity Script Debugger to trace bugs and performance issues in Microsoft Dynamics GP

<http://support.microsoft.com/kb/910982>

8. Network Monitor

Network Monitor is a protocol analyzer that allows the capture of network traffic and the ability to view and analyze it.

Microsoft Network Monitor 3.4

<http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=4865>

Analysis Tools Used for Specific Performance Issues

1. Blocking

- Performance Analyzer Version 1.01 for Microsoft Dynamics
- SQL Server Profile Trace with Blocked Process Report

2. Deadlocking

- SQL Server Profile Trace including the following events:
 - Lock: Deadlock Graph
 - Lock: Deadlock
 - Lock: Deadlock Chain
- Analyzing Deadlocks in SQL Server 2008 R2 with SQL Server Profiler
<http://msdn.microsoft.com/en-us/library/ms188246.aspx>

3. Long Running Queries

- SQL Server Profile Trace including the following events:
 - Performance: Showplan All
 - Performance: Showplan Statistics Profile
 - Performance: Showplan XML
- Performance Analyzer Version 1.01 for Microsoft Dynamics

4. Hardware Bottlenecks

- Performance Monitor
- PAL

5. Networking

- Netmon
- Microsoft Dynamics GP Client Logging Tools

6. External Influences

- Process Monitor

Common Bottlenecks

The following table lists the common bottlenecks to investigate using the Performance Monitor tool. Monitor the objects below and ensure your environment is at the recommended thresholds as shown below.

If you have questions regarding the bottleneck information below or would like assistance in reviewing Performance Monitor, please contact your Partner or e-mail askpts@microsoft.com to arrange for an advisory services engagement

Object	Counter	Threshold	Description
Disk Subsystem			
Physical Disk	Avg. Disk Queue Length	> 2 * number of spindles	Avg. Disk Read Queue Length is the average number of requests that were queued for the selected disk during the sample interval. More Info: < (2+ no of spindles) Excellent < (2*no of spindles) Good < (3* no of spindles) Fair Note: If the disk has say 20 disks and it is RAID 10 then no. of spindles = 20/2 = 10. If it is RAID 5 then the no. of spindles = no of disks = 20.
Physical Disk	Current Disk Queue Length	> 2 * number of spindles	Current Disk Read Queue Length is the current number of requests that were queued for the selected disk during the sample interval. More Info: < (2+ no of spindles) Excellent < (2*no of spindles) Good < (3* no of spindles) Fair Note: If the disk has say 20 disks and it is RAID 10 then no. of spindles = 20/2 = 10. If it is RAID 5 then the no. of spindles = no of disks = 20.
Physical Disk	Avg. Disk Sec/Transfer	> 10-20ms	Avg. Disk sec/Transfer is the average time, in seconds, of a read or write of data from the disk. (Latency)
Physical Disk 32	Avg. Disk Sec/Read	> 10-20ms	Avg. Disk sec/Read is the average time, in seconds, of a read of data from the disk. (Latency)
Physical Disk	Avg. Disk Sec/Write	> 10-20ms	Avg. Disk sec/Write is the average time, in seconds, of a write of data to the disk. (Latency)

Object	Counter	Threshold	Description
Physical Disk	Disk Reads/sec	Verify against 80% drive capacity for random I/Os per second	<p>Number of Reads per second requested to the Disk. (Volume)</p> <p>More Info:</p> <p>General rule of thumb 80% capacity per drive is around 120 random I/Os per second. Use the following formulas depending on the RAID configuration.</p> <p>Raid 0 -- I/Os per disk = (reads + writes) / number of disks</p> <p>Raid 1 -- I/Os per disk = [reads + (2 * writes)] / 2</p> <p>Raid 5 -- I/Os per disk = [reads + (4 * writes)] / number of disks</p> <p>Raid 10 -- I/Os per disk = [reads + (2 * writes)] / number of disks</p>
Physical Disk	Disk Writes/sec	Verify against 80% drive capacity for random I/Os per second	<p>Number of Writes per second requested to the Disk. (Volume)</p> <p>More Info:</p> <p>General rule of thumb 80% capacity per drive is around 120 random I/Os per second. Use the following formulas depending on the RAID configuration.</p> <p>Raid 0 -- I/Os per disk = (reads + writes) / number of disks</p> <p>Raid 1 -- I/Os per disk = [reads + (2 * writes)] / 2</p> <p>Raid 5 -- I/Os per disk = [reads + (4 * writes)] / number of disks</p> <p>Raid 10 -- I/Os per disk = [reads + (2 * writes)] / number of disks</p>
Physical Disk	Disk Bytes/sec	> 10MB/sec	Rate that bytes are transferred to or from the disk

Object	Counter	Threshold	Description
CPU			
Processor	% Processor Time	>80%	Percentage of time CPU spends executing a non-Idle thread
Processor	% Privileged Time	>10%	Percentage of time CPU spends executing code in Privileged mode. (i.e. performing actions such as I/O requests)
System	Processor Queue Length	>2	Number of threads waiting to be processed by any available CPU
System	Context Switches/sec	>300~1000 per processor	Rate at which all processors are switching between threads
SQL Server: SQL Statistics	SQL Re-Compilations /sec		Recompiles per second Higher values put more load on the CPU
SQL Server: SQL Statistics	SQL Compilations /sec		Compiles per second Higher values put more load on the CPU
Memory			
Memory	Available Mbytes	<100-300 MB	Amount of Free Memory (less than 10 MB would be reaching critical stage where crash of OS may occur)
Memory	Pages/Sec	>100	Rate at which pages are written to disk to resolve hard page faults
MSSQL: Buffer Manager	Buffer cache hit ratio	<90%	Percentage of pages found in the buffer pool without having to read from physical disk
MSSQL: Buffer Manager	Free Pages	<640	Total number of Free pages FYI - 640 pages ~ 5 Meg
MSSQL: Memory Manager	Memory Grants Pending	=0	Number of processes waiting for a memory grant
MSSQL: Memory Manager	Target Server Memory	~Physical Memory	Total amount of dynamic memory the server is willing to consume
MSSQL: Memory Manager	Total Server Memory	~Target Server Memory	Total amount of dynamic memory the server is currently consuming

Object	Counter	Threshold	Description
Network			
Network Interface	Bytes Total/sec	> 50% capacity	Rate at which bytes are sent and received over network adaptor
Network Interface	Current Bandwidth		Estimated current bandwidth in bits per second
Network Segment	% Net Utilization	> 80% Bandwidth	Percent usage of total bandwidth (Need to install the Network Monitor Driver from the Local Network protocol list of network components for the network adaptor to see this counter)
Miscellaneous			
SQLServer: Access Methods	Full Scans/sec	> 1	If we see high CPU then we need to investigate this counter, otherwise if the full scans are on small tables we can ignore this counter. Values greater than 1 or 2 indicate that we are having table / Index page scans. We need to analyze how this can be avoided.
SQLServer: General Statistics	User Connections		The number of users currently connected to the SQL Server. Note - Gradual increase might indicate connections are not being closed properly
SQL Server: SQL Statistics	Batch Requests/sec		Rough estimate on number of Batch requests / sec. Good for trend data
SQL Server: Locks	Number of Deadlocks/sec	= 0	The number of lock requests that resulted in a deadlock.
SQL Server: Locks	Average Wait Time (ms)	> 200-500	This is the average wait time in milliseconds to acquire a lock. Lower the value the better it is. If the value goes higher than 500, there may be blocking going on; we need to run blocker script to identify blocking.
SQL Server: Locks	Lock Timeouts/sec		Number of locks requests that timed out
SQL Server: Latches	Total Latch Wait Time(ms)		Total latch wait time for latch requests that had to wait in the last second

Advanced Troubleshooting Techniques for Performance

Every performance issue is different; however, each will have similar characteristics. Below are a few common scenarios requiring specific focus areas while troubleshooting. While this list will assist in identifying the issue, it is not meant to provide detailed information regarding the resolution as each issue may require a different approach.

If you have questions regarding the scenarios below, please contact your Partner or Microsoft Dynamics GP Support to arrange for an advisory services engagement.

Issue 1 - Random performance issues that affect multiple users/processes simultaneously

Possible Cause: This issue is most likely a blocking issue, however also recommend to review potential hardware issues especially disk latency.

Troubleshooting Techniques:

- Capture and review long duration blocking information using the SQL Server Profile Trace or the Performance Analyzer for Microsoft Dynamics to determine lead blocker.
 - Identify the application/code calling the lead blocker statement.
 - From a development perspective, the following are ways to reduce long duration blocking:
 - Attempt to use shorter TRAN logic if possible
 - Ensure statements inside the TRAN are tuned appropriately
 - Potentially use (nowait) hint on reporting to minimize report blocking other processes
- Capture and review Performance Monitor counters to validate sufficient hardware.
 - Focus on disk latency
 - Validate CPUs are not saturated
 - Check for network issues
- Capture and review long running queries using the SQL Server Profile Trace.
 - If Disk latency is high and I/O is high attempt to tune high cost queries

Issue 2 - Random performance issues that affect various users/processes at various times

Possible Cause: This issue is most likely related to hardware infrastructure issue, however, also recommend looking into a potential blocking issue.

Troubleshooting Techniques:

- Verify the latest hardware, operating system, SQL Server, and Microsoft Dynamics GP patches are installed.
- Verify anti-virus software has the required exclusions configured as recommended in this document.
- Capture and review Performance Monitor counters for potential bottlenecks and to validate sufficient hardware
 - Focus on Disk latency
 - Validate CPUs are not saturated
 - Check for network issues
 - Check for antivirus interference

- Capture and review long duration blocking information using the Performance Analyzer for Microsoft Dynamics to determine lead blocker.
 - Identify the application/code calling the lead blocker statement.
 - From a development perspective, the following are ways to reduce long duration blocking:
 - Attempt to use shorter TRAN logic if possible
 - Ensure statements inside the TRAN are tuned appropriately
 - Potentially use (nowait) hint on reporting to minimize report blocking other processes
- Capture long running queries
 - If disk latency is high and I/O is high attempt to tune high cost queries.

Issue 3 - SQL Server Profile Trace shows single insert statements taking 50 - 500ms

Possible Cause: This issue can usually be attributed to high latency on the disks where the *.LDF files are located. This is seen when the *.LDF files are not on a dedicated RAID volume, rather on the same volume as the *.MDF file. Thus when *.MDF files have heavy read I/O and the drives are already saturated the *.LDF files are not written to as quickly as they should.

Troubleshooting Techniques:

- Capture and review Performance Monitor counters to validate sufficient hardware
 - Focus on Disk latency
 - If high latency and low IO activity
 - Check if shared RAID group on SAN
 - Ensure disk controllers have the latest firmware / drivers
- Verify anti-virus software has the required exclusions configured as recommended in this document.
- Isolate the *.LDF files to a dedicated RAID group.

Issue 4 - SQL Server Profile Trace shows simple statements taking 500ms or higher

Possible Cause: This issue can usually be attributed to long duration blocking.

Troubleshooting Techniques:

- Verify the latest hardware, operating system and SQL Server patches are installed.
- Verify anti-virus software has the required exclusions configured as recommended in this document.
- Verify TCP Chimney is disabled, and other network considerations as recommended in this document.
- Capture long duration blocking information using SQL Server Profile Trace or the Performance Analyzer for Microsoft Dynamics to figure out the lead blocker.
 - Identify the application/code calling the lead blocker statement.
 - From a development perspective, the following are ways to reduce long duration blocking:
 - Attempt to use shorter TRAN logic if possible
 - Ensure statements inside the TRAN are tuned appropriately
 - Potentially use (nowait) hint on reporting to minimize report blocking other processes

- Review the ShowPlan Statistics in the SQL Server Profile Trace to validate if table statistics are up-to-date
 - If the actual and the estimated values of Showplan Statistics are extremely different, statistics are off, and it is a promising idea to update stats with full scan for all affected tables.
 - If stats are off, this can lead to poor execution plan / wrong index might be utilized

Issue 5 - Specific process is slow

Possible Cause: Various causes could be possible.

Troubleshooting Techniques: When a specific process is consistently slow, recreate the issue while capturing performance logs. Review the logs and tune if possible.

- Capture detailed SQL Server Profile Trace
- Capture Dexterity Script.log and Profile.txt
- Screenshots and steps detailing the issue recreation process.

Issue 6 - Windows is stating Microsoft Dynamics GP is "Not Responding"

Possible Cause: This issue usually is caused by a foreground Microsoft Dynamics GP process that made a call to SQL Server and is waiting for a response back. This could take a long time if blocked by another long running process, or a complex request was sent, or a large dataset is being returned. The best approach is to rule out blocking, verify any long running queries are optimized and then review the health of the SQL Server and client workstations.

Troubleshooting Techniques:

- Capture long duration blocking information using SQL Server Profile Trace or the Performance Analyzer for Microsoft Dynamics to figure out the lead blocker.
 - Identify the application/code calling the lead blocker statement.
 - From a development perspective, the following are ways to reduce long duration blocking:
 - Attempt to use shorter TRAN logic if possible
 - Ensure statements inside the TRAN are tuned appropriately
 - Potentially use (nowait) hint on reporting to minimize report blocking other processes
- Capture long running queries using the SQL Server Profile Trace.
 - If disk latency is high and I/O is high attempt to tune high cost queries
- Capture and review Performance Monitor counters for bottlenecks

Resources

1. Architecture White Paper for Microsoft Dynamics GP 2010 (valid for later versions as well)
Customers:
https://mbs.microsoft.com/customersource/northamerica/GP/learning/documentation/white-papers/gp2010_architecturewhitepaper

Partners:
Architecture White Paper for Microsoft Dynamics GP 2010 (valid for later versions as well)
https://mbs.microsoft.com/partnersource/northamerica/deployment/documentation/white-papers/gp2010_architecturewhitepaper
2. Performance Tuning Guidelines for Windows Server 2008
<http://msdn.microsoft.com/en-us/windows/hardware/gg463394.aspx>
3. Performance Tuning Guidelines for Windows Server 2008 R2
<http://msdn.microsoft.com/en-us/windows/hardware/gg463392.aspx>
4. Refer to the article below for the SQL Server Storage Top 10 Best Practices
<http://msdn.microsoft.com/en-us/library/cc966534.aspx>
5. Trace flag 4199 is added to control multiple query optimizer changes previously made under multiple trace flags
<http://support.microsoft.com/kb/974006/en-US>
6. Troubleshooting Performance Problems in SQL Server 2008
[http://msdn.microsoft.com/en-us/library/dd672789\(v=SQL.100\).aspx](http://msdn.microsoft.com/en-us/library/dd672789(v=SQL.100).aspx)
7. How to troubleshoot SQL Server performance issues
<https://support.microsoft.com/en-us/help/298475/how-to-troubleshoot-sql-server-performance-issues>

(c)2011 Microsoft Corporation. All rights reserved. This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes. This document is confidential and proprietary to Microsoft. It is disclosed and can be used only pursuant to a non-disclosure agreement.