Opportunistic Locking and Read Caching on Microsoft Windows Networks

A White Paper by Dennis Piccioni

Revised: November 2016

Summary

Most DataFlex applications today run on one of the leading SQL databases with Microsoft SQL Server being the most popular choice. SQL databases do not suffer from the OpLock risks or recent changes to the SMB protocols. Data Access Worldwide highly recommends that customers move DataFlex web, mobile and Windows desktop applications still using the file system "embedded database" to an SQL database as soon as possible to avoid the issues discussed below, and to enjoy better performance, reliability and scalability.

Without special configuration, any file system database, including DataFlex's, can experience data corruption on Windows networks from opportunistic locking on servers and read caching on clients.

- For developers using file system databases other than DataFlex's, visit Microsoft for more information about this issue, or consider consulting with our Professional Services experts for application development assistance with resolving this issue.

- For Data Access Worldwide customers still using the DataFlex embedded file system database, continue reading this paper for more about these Windows network behaviors, their effects, and what can be done to reduce the chances of data corruption until a transition to SQL is accomplished.

- Learn more about the features of DataFlex SQL Connectivity Kits.

- Our Professional Services group's services include assistance with embedded to SQL database migrations if needed.

The information in this paper is compiled from the latest available information regarding these issues from Microsoft, our own in-house testing and customer reports. We are attempting to combine the limited information provided my Microsoft on these topics in one place. Please revisit this white paper from time to time to check for updated information. The Last Edited date at the top of the paper will reflect when the latest edits were made. The information in this white paper only deals with operating systems that we currently support. You can view information about supported products & environments in the Data Access Worldwide Current Products List.

Contents

What is Opportunistic Locking?

Opportunistic locking (oplocks) is a Windows-specific mechanism for client/server databases to allow multiple processes to lock the same file while allowing for local (client) data caching to improve performance over Windows networks. Unfortunately, the default setting of the oplocks mechanism that enhances the performance of one type of database (client/server) also introduces data integrity issues for other database types (file system/ISAM).

Microsoft's documentation states "An *opportunistic lock* (also called an oplock) is a lock placed by a client on a file residing on a server. In most cases, a client requests an oplock so it can cache data locally, thus reducing network traffic and improving apparent response time. Oplocks are used by network redirectors on clients with remote servers, as well as by client applications on local servers" and "Oplocks are requests from the client to the server. From the point of view of the client, they are opportunistic. In other words, the server grants such locks whenever other factors make the locks possible."

You can read more about oplocks in Microsoft's documentation. Please see the Resources section below for more information.

What is Read Caching?

Read caching, sometimes referred to as read-ahead caching, is a feature of oplocks. It is a technique used to speed network access to data files. It involves caching data on clients rather than on servers when possible.

The effect of local caching is that it allows multiple write operations on the same region of a file to be combined into one write operation across the network. Local caching reduces network traffic because the data is written once. Such caching improves the apparent response time of applications because the applications do not wait for the data to be sent across the network to the server.

Problems with read caching usually occur if something unforeseen happens, such as a workstation crash, where data is not properly flushed from the workstation, which can lead to data corruption.

Microsoft's documentation states that 'Under extreme conditions, some multiuser database applications that use a common data store over a network connection on a file server may experience transactional integrity issues or corruption of the database files and/or indexes stored on the server. This typically applies to some so-called "ISAMstyle", or "record oriented" multiuser database applications, not to a client/server relational system like SQL Server' and 'A hazard of local caching is that written data only has as much integrity as the client itself for as long as the data is cached on the client. In general, locally cached data should be flushed to the server as soon as possible.'

You can read more about read caching in Microsoft's documentation.

What Are SMB2 and SMB3?

SMB2 and SMB3 are the second and third generations, respectively, of server message block (SMB) communication on Windows networks. SMB2 was introduced in Windows Vista, 7 and Windows Server 2008 to enable faster communication between computers that are running Windows Vista, 7 and Windows Server 2008. SMB3 was introduced in Windows 8 and Windows Server 2012. SMB1, also called "traditional SMB", is still supported in current Windows versions for backward compatibility.

Data Access Worldwide Recommendations

The embedded (DataFlex) database is an ISAM database and thus susceptible to the effects of the default Windows oplocks settings. **Using the embedded database on Windows networks without disabling oplocks is not recommended or supported** and has a high likelihood of data corruption.

The best data integrity, security and performance is available by using a client/server database, such as IBM DB2, Microsoft SQL Server or Pervasive.SQL with your Visual DataFlex and DataFlex applications. Data Access Worldwide has direct drivers (Connectivity Kits) available for IBM DB2, Microsoft SQL Server and Pervasive.SQL, as well as an ODBC Connectivity Kit for access to any ODBC-compliant databases. All of these drivers are loaded at runtime and require no coding changes to be used with existing VDF, DataFlex or WebApp Server applications.

Reliable database operation on Windows Networks can be achieved using the embedded database, provided that the network is properly configured. You can use the information in this paper to set up your Windows network's oplocks parameters. One downside to using this method are maintenance issues: you must continually ensure that each and every server and client using an application accessing the embedded database has oplocks disabled and are always maintained in that state.

One method to ensure that oplocks are disabled on client PCs is to add code to applications that checks those settings on application startup. The folks at VDF-Guidance.com have created an open source project named RegCheck for this purpose.

Disabling oplocks may have a performance impact on Windows networks.

Oplocks do not apply to client-server databases. DAW makes no specific recommendation on oplocks if you use a client server database and no embedded database tables.

This paper will tell you how to disable oplocks, but due to the reasons stated above, **Data Access Worldwide recommends using a client-server database for multi-user DataFlex applications on Windows networks.**

What Operating Systems are Affected?

All computers running Windows operating systems that host or access embedded database tables accessed by other Windows PCs need to have oplocks disabled in order to minimize the chances of database corruption.

Oplocks can be disabled on either (or both) of these:

- the client side (a Windows PC that accesses an embedded database table hosted on another PC)
- the server side (a Windows PC that hosts an embedded database table accessed from another PC)

The Windows operating system list that we currently support for our products includes Windows 7, Windows 8, Windows 10, Windows Server 2008 and Windows Server 2012.

What Environments Are Not Affected?

There are some environments and scenarios that we support that may not be affected by oplocks, even if using the embedded database:

- **Local Database Access**

In general, whenever an embedded database table is accessed on the same PC where that table is located, oplocks do not apply.

- **Windows Terminal Services and Citrix**

Under normal use for these environments, users log onto a Windows server and run applications locally on that server. If, however, an embedded database is located on another server than the one running WTS/Citrix, oplocks between the WTS/Citrix server and the database server must be disabled.

- **Web Application Server Applications/Web Services**

In web applications users access a web browser which requests data from a web application and/or data is requested via a web service. In both cases, the web application on a web server accesses the database, the client does not. If the data is located on the same server, oplocks do not come into play. If, however, an embedded database is located on another server than the one running the web application, oplocks between the web server and the database server must be disabled.

Making Windows Registry Changes

The topics below discuss changing editing the Windows Registry.

Caution: The following warning appears in every Microsoft article that discusses editing the Windows Registry:

WARNING : You can edit the registry by using Registry Editor (Regedit.exe or Regedt32.exe). If you use Registry Editor incorrectly, you can cause serious problems that may require you to reinstall your operating system. Microsoft does not guarantee that problems that you cause by using Registry Editor incorrectly can be resolved. Use Registry Editor at your own risk.

If you change any of the Registry values discussed below, you will have to reboot the PC on which the value was changed to ensure that the new setting goes into effect.

If any subkeys or values described do not exist in your Registry, you will have to add them. Please check carefully before doing so.

Disabling Oplocks on Windows Client PCs

To disable oplocks on a Windows client PC (a Windows PC that accesses an embedded database table hosted on another PC), change or add the following DWord Registry value:

- HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\MRXSmb\Parameters**OplocksDisabled = 1**

Disabling Oplocks on Windows Servers

To disable oplocks on a Windows server (a Windows PC that hosts an embedded database table accessed from another PC), change or add the following DWord Registry value:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters**EnableOplocks = 0**

Disabling Oplocks on SMB2 and SMB3

Oplocks **cannot** be turned off for SMB2 and SMB3. You can disable SMB2 and SMB3 themselves, how to do so is documented by Microsoft in Knowledge Base article 2696547.

According to that article, SMB2 and SMB3 can be disabled on Windows operating systems that support these.

To disable SMB2 and SMB3 on a Windows Vista, 7, 8, Server 2008 or Server 2012 PC hosting embedded database tables, change or add the following DWord Registry value:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters **SMB2 = 0**

Once SMB2 and SMB3 are disabled, SMB1 should be re-enabled to be used again and the methods described above applied to disable oplocks for SMB1.

To re-enable SMB1 on a Windows Vista, 7, 8, Server 2008 or Server 2012 PC hosting embedded database tables, change or add the following DWord Registry value:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters **SMB1 = 1**

Do Coding Practices Affect These Issues?

- If your application code uses DataDictionaries and/or Data_Sets, there should be no data integrity problems after oplocks have been disabled.

  Customers have reported that with application code that does not use Data Dictionaries and/or Data Sets (for example, in a Find loop using the record buffer for finding), data in records that is new or edited since the data was first accessed will still not be retrieved properly, even with oplocks disabled. Workarounds for this condition are to do the Find in a locked state or issuing a Reread command after each record is found (remember to issue an unlock command after the reread as a reread performs a lock as part of its functionality). We will publish any additional information we obtain about how to get around this Microsoft operating system problem when it becomes available.

- We have tried using the Win32 FlushFileBuffers Windows API function that Microsoft recommends in their documentation in the Visual DataFlex/DataFlex runtime when the DF_HIGH_DATA_INTEGRITY attribute was turned on. However, application performance degraded to the point that it was virtually unusable when doing so, because this Windows API function is a very generic call that flushes all buffers on a client PC instead of just those used by one application.

Persistent Data Corruption

If you have applied all of the settings discussed in this paper but data corruption problems and other symptoms persist, here is some additional information:

- We have credible reports from developers that faulty network hardware, such as a single faulty network card, can cause symptoms similar to data corruption.
- If you see persistent data corruption even after repeated reindexing, you may have to rebuild the tables in question. This involves creating a new table with the same definition as the table to be rebuilt and transferring the data from the old table to the new one. There are several known methods for doing this that can be found in our Knowledge Base.

Terms

- **ISAM**
  Indexed Sequential Access Method is a file management system developed at IBM that allows records to be accessed either sequentially (in the order they were entered) or randomly (with an index).

- **SMB**
  The Server Message Block (SMB) Protocol is a network file sharing protocol, and as implemented in Microsoft Windows is known as Microsoft SMB Protocol. If you wish to learn more about SMB, consult Microsoft's documentation.

You may want to check for an updated version of this white paper from time to time. Many of our white papers are updated as information changes. For those papers, the **Last Edited** date is always at the top of the paper.

Resources

- Opportunistic Locks, Microsoft Developer Network (MSDN)

- Microsoft Knowledge Base Article 2696547 How to enable and disable SMBv1, SMBv2, and SMBv3 in Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, and Windows Server 2012

- Microsoft Knowledge Base Article Q296264 Configuring Opportunistic Locking in Windows

- Microsoft Knowledge Base Article Q224992 Maintaining Transactional Integrity with OPLOCKS

- Microsoft Knowledge Base Article Q129202 PC Ext: Explanation of Opportunistic Locking on Windows NT

- Microsoft Knowledge Base Article Windows registry information for advanced users.

- **RegCheck**
  One of the best ways to ensure that oplocks are disabled on client PCs is to add code to applications that checks those settings on startup. The folks at VDF-Guidance.com have created an open source project named RegCheck for this purpose. Note that the code in this project is not verified or maintained by DAW.

- **Data Access Worldwide Support**
  Visit the DAW Support Home page for information about all of our support offerings, including the list of supported products, bug report forms and free support offerings, such as the Knowledge Base, White Papers and Peer Support Forums.

- **DAW Knowledge Base**
  Visit the Data Access Worldwide Knowledge Base, a great resource for the latest technical information about all Data Access Worldwide products.

- **Data Access Worldwide Forums**
  Visit the DAW Forums for sharing information about Data Access Worldwide products with other developers and users.

Contacting Data Access Worldwide

Data Access Worldwide
14000 SW 119 Ave
Miami, FL 33186
305-238-0012
Domestic Sales: 800-451-3539
Fax: 305-238-0017
email: sales@dataaccess.com
Internet: http://www.dataaccess.com

Data Access Worldwide - Brasil
Av.Paulista, 1776 - 21st.Floor
São Paulo -SP - Brazil
CEP 01310-921
Phone: 5511-3262-2000
Fax 5511-3284-1579
Sales: info@dataaccess.com.br
Support: suporte@dataaccess.com.br
Internet: http://www.dataaccess.com.br

Data Access Worldwide - Europe
Lansinkesweg 4
7553 AE Hengelo
The Netherlands
Telephone: +31 (0)74 - 255 56 09
Fax: +31 (0)74 - 250 34 66
Sales: info@dataaccess.nl
Support: support@dataaccess.nl
Internet: http://www.dataaccess.nl

DISCLAIMER - NO LIABILITY FOR CONSEQUENTIAL DAMAGES
To the maximum extent permitted by applicable law, in no event shall Data Access Corporation be liable for any special, incidental, indirect, or consequential damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use any information provided in this document, even if Data Access Corporation has been advised of the possibility of such damages. Because some states and jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.