

Pervasive.SQL - Configuration & Performance Settings

Neil Hughes
Support4Omega Ltd.

19th June 2005

1 Introduction

Although one of Pervasive.SQL's advantages is its ability to be installed and run straight out of the box when used with many applications, its default settings are not suitable for running Omega (in the case of Pervasive.SQL 2000) or are not set for optimum performance. This document describes briefly the settings that should be configured when Omega is initially installed and those settings that should be reviewed from time-to-time to check whether they are still suitable based on external factors (e.g. number of users, upgrades to the workstation/server running the database engine).

This document concentrates on Pervasive.SQL versions 2000 and 2000i (those with an internal version number of 7.50 to 7.94), though much of the advice also applies to earlier versions (7) and those still supported by Pervasive (8.x and 9). As there is a shift to later versions this document will be adjusted accordingly. For the time being any differences will be described in footnotes as they are discovered¹.

2 Pervasive.SQL 2000 vs. 2000i

Since the configuration settings available depend on the version of the database engine that you are running, it is worth reviewing what versions are available and how they can be upgraded. Pervasive.SQL 2000 exists in four common versions:

- Version 7.5 - Pervasive.SQL 2000 with no service pack
- Version 7.8 - Pervasive.SQL 2000 + service pack 2a
- Version 7.9 - Pervasive.SQL 2000 + service pack 3
(shipped as Pervasive.SQL 2000i)

¹The term 8.x refers to the fact that there are three versions of Pervasive.SQL 8 (8, 8.1 and 8.5) and any notes in this document apply to all three unless otherwise noted.

- Version 7.94 - Pervasive.SQL 2000 + service pack 3 + service pack 4 (or Pervasive.SQL 2000i + service pack 4)

Pervasive.SQL 2000i is effectively Pervasive.SQL 2000 with the addition of service pack 3. Service pack 3 contains all of the changes in service pack 2a and can be installed on a machine running no service pack or service pack 2a.

Service pack 4 can only be applied to a machine running service pack 3, or a machine with Pervasive.SQL 2000i since it is the same in all but name.

Although Pervasive no longer support Pervasive.SQL 2000, they still recommend that those sites using it upgrade to the latest service pack. As well as containing various bug fixes and performance enhancements, the last service pack included support for running the database engine on Windows XP, Windows Server 2003 and Novell NetWare 6. Service pack 3 also removed the need to set the maximum number of files and file handles that the engine needs to be able to cope with (see later). Additionally, the infamous “phantom” status 2 issue has been reduced in frequency with each service pack, though never totally resolved².

From Prestim/Frontline/Support4Omega experience I would add the following comments:

1. If you are upgrading the database engine, upgrade all the way to service pack 4. Do not stop at service pack 3 - this introduced a serious locking issue.
2. If you are upgrading the client machines (it is recommended that the client requester version matches the engine version if possible) you will find that there is no service pack 4 for the clients. The latest version is service pack 3. This is OK - the locking issue mentioned previously does not apply to the clients, just to the database engine.
3. If you are not experiencing any serious issues currently with the database engine and are using an earlier service pack (or none at all) then I would advise using the “if it ain’t broke, don’t fix it” principle.

3 Configuration

The following sections describe the settings relevant to running Omega at optimum performance. The advice is based on the performance tuning tips given by Pervasive in their documentation and on their web site, along with observations recorded by myself and other Prestim/Frontline staff over the

²There are not enough sites running versions 8.x or 9 to know yet whether phantom status 2 errors have finally been resolved. This problem also appears to be more common with the NetWare version of the database engine.

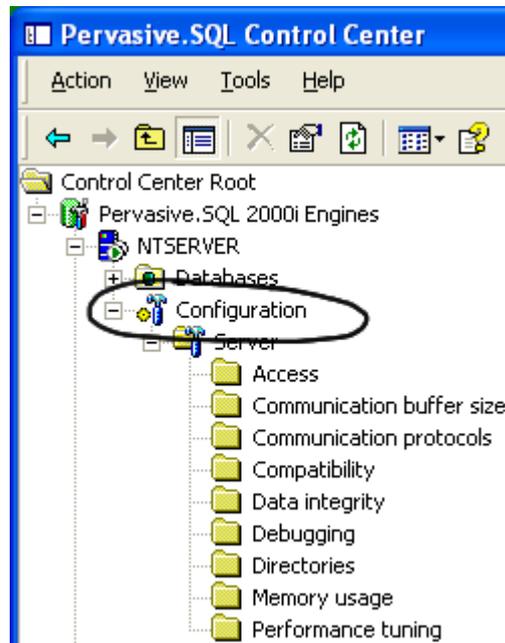


Figure 1: Pervasive Control Center, showing Configuration module.

years. Any differences due to engine version (server or workgroup) and service pack version are included where necessary.

The **Location** description indicates where in the “Configuration” module in the Pervasive Control Center the particular setting can be found³.

3.1 Number of Sessions (not required in v8.x and 9 - handled automatically)

Location: Server ► Access ► Number of Sessions

Default Value: 100 or 1000

Recommended Value: Maximum number of connections to the database, not including those running on the same machine as the engine

Resources required: approx. 32KB per session

The maximum number of remote sessions (multiple Omega modules running on a workstation count as one session) accessing the database engine. This should include all Omega workstations and dedicated print server machines, plus any machines being used to run Crystal Reports stand-alone.

This is not a performance tuning setting - if you do not have this value high enough then you will see a Pervasive error (usually 91 - server error or

³The version 9 Pervasive Control Center has been completely rewritten and differs quite a lot in terms of look and functionality

96 - communications error). Likewise, having this setting higher than the recommended value does not gain any performance - it just wastes memory.

3.2 Communication Protocols

Location: Server ► Communication protocols ► Supported protocols

Client ► Communication protocols ► Supported protocols

Default Value: Microsoft SPXII + Microsoft TCP/IP + Microsoft Net-Bios⁴

Recommended Value: Just specify the protocol required

Resources required: n/a

The database engine can be accessed from other workstations using a number of different network protocols. It is recommended that only one protocol is specified because this will reduce the initial connection time between the remote machine and the engine.

Pervasive does not have to use the same protocol as that used for network login and general file sharing, but it is usually recommended to help in troubleshooting and optimising performance.

It is also important to remember that both the machine hosting the database (the server engine in the case of client/server, or the gateway machine in the case of peer-to-peer workgroup setups) and the remote machines have to be set up with matching protocols. The former uses the setting under “Server”, the latter under “Client”.

3.3 Trace Debugging

Location: Server ► Debugging ► Trace Operation

Default Value: Off

Recommended Value: Off

Resources required: n/a

Make sure that Btrieve API operations are not being logged to the trace file - this can impact performance a great deal and should only be used when trying to troubleshoot a specific problem.

3.4 Database Cache Size

Location: Server ► Performance tuning ► Cache Allocation Size

Default Value: Depends on RAM present when Pervasive.SQL is installed

Recommended Value: Equal to total size of Omega database, but not

⁴The default values will vary depending on the engine type (Microsoft Windows or Novell NetWare) and the network client installed on the client machine. The server engine does not support NetBios as a communication protocol.

greater than 40% of the physical RAM on the server

Resources required: Contiguous block of memory to allocate the cache from⁵

The Pervasive.SQL database engine tends to be limited by I/O rather than being processor bound - you can see this when a complex query is in operation, since the processor utilization tends to stay low. The biggest contributors to overall performance are:

1. The ability to retrieve the data from a RAM-based cache, since it is faster than retrieving from disk;
2. The speed of the disk sub-system, for retrieving any data not cached;
3. The speed of the network sub-system, for returning query results to the remote workstation.

3) is less important because ideally the queries in Omega have been optimized to retrieve only that data which is necessary. Therefore speeds beyond 100Mb/s may not result in a noticeable improvement. Crystal Reports queries that employ many selection criteria that are not being performed on the server may be another story.

The biggest improvements come with increasing the cache size and using the fastest hard disk technology available. Ideally you should aim to fit the whole database in the cache, but this may not be feasible for a number of reasons:

1. Pervasive advise keeping the cache to 40% of the physical RAM present, to leave enough memory for the operating system and other services or applications;
2. The database is so big that the host machine cannot accomodate the required amount of physical RAM, particularly when you take into account 1), or the cost of the required RAM makes upgrading impractical;
3. Pervasive.SQL 2000 requires a single block of memory for the cache, and it may not be able to find this due to memory fragmentation⁶. Research indicates that cache sizes beyond 700MB are difficult to achieve.

⁵Versions 8.x and 9 use a two level caching mechanism, partly to get around the limitation in cache size with 32 bit Windows. The main cache is usually set to 20-70 percent of the total database size and the dynamic second level cache used as required. Please contact Omega support for further information.

⁶As mentioned already, Pervasive have had to workaroud this issue with Pervasive.SQL 8.x and later by allowing the cache to be allocated from multiple blocks, and employing a 2-level cache mechanism.

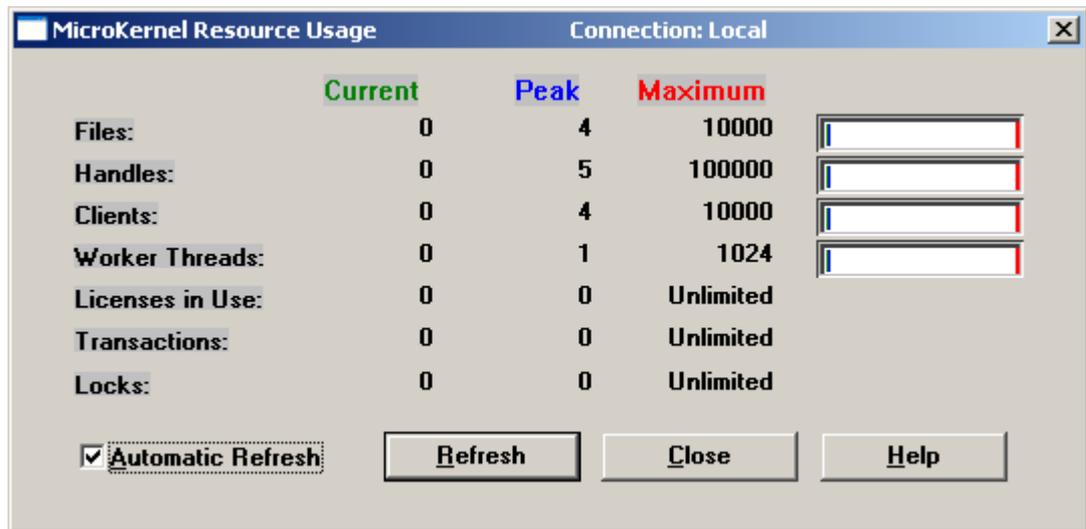


Figure 2: Pervasive Monitor utility showing resources currently in use (none in this case) and the peak usage since the database engine was started.

3.5 Communication Threads

Location: Server ► Performance tuning ► Communication Threads

Default Value: 3⁷

Recommended Value: Equal to the maximum number of files open at any one time

Resources required: Unknown

The communication threads are used to handle the database engine's file access on behalf of the remote clients requests. Ideally you should aim for one thread per open file. It is unlikely that every Omega database file is going to be open at the same time, so instead you should use the Pervasive Monitor utility to find out how many are open (MicroKernel menu ► Resource Usage).

3.6 Input/Output Threads

Location: Server ► Performance tuning ► Number of Input/Output Threads

Default Value: 3

Recommended Value: Equal to (the maximum number of files open at any one time)/8

Resources required: Unknown

The database engine has a fixed number of threads used for handling in-

⁷Version 9 defaults to 16 and handles the setting automatically.

put/output requests. Pervasive recommend no more than one-eighth the maximum number of files open at once (see the communication threads setting above for instructions on finding this value).

3.7 Other Settings

The **Worker Threads** setting, found in the Windows server engine, does not need to be adjusted because it automatically increases when required. The worker threads are responsible for handling database file access from local applications on the database engine machine.

Pervasive.SQL 2000 without service pack 3 applied needs to have the **Maximum Files**⁸ and **Maximum Handles**⁹ settings set - these are handled automatically in Pervasive.SQL 2000i and later. The **Maximum Files** setting must match the maximum number of open files that the database engine will need to cope with at any one time. The **Maximum Handles** setting needs to equal or exceed the calculated total of all the files being open multiplied by the number of times these files are open on the same workstation, i.e. somebody running integrated engineering and despatches would have two handles created for files such as sales orders and delivery schedules.

The maximum file setting should have been configured correctly during Omega installation and is unlikely to need changing unless a future update adds or removes a considerable number of tables. There are currently around 250 files in the database. Not having a large enough value will result in a status 86 (file table is full).

The maximum handles setting may need adjusting as the number of users changes and their use of Omega changes, or as more print servers and dedicated Crystal Reports machines run reports concurrently. A useful starting point is 1000, which assumes a theoretical situation where every file in the Omega database is in use by 4 different users or applications. Not having a large enough value will result in a status 87 (handle table is full).

4 Other Applications

When deciding on suitable configuration values, do not forget that applications such as the print server and Crystal Reports also need to be taken into account. Their use of files, handles and threads is likely to be shorter in duration than a user running an Omega module and therefore harder to monitor. If you are getting sporadic database errors complaining about a lack of files, handles or sessions it is likely that a report or print server job is

⁸Server ► Access ► Maximum Open Files

⁹Server ► Access ► Logical File Handles

occasionally hitting the limit or preventing an Omega user from obtaining the resources.

5 Summary

The file, handle and remote session values are not performance-related - you must have these set correctly otherwise Omega will not run. Pervasive.SQL 2000i administrators need only be concerned with the number of remote sessions.

Try to get all of the clients or peer-to-peer workgroup machines communicating with the same network transport protocol and remove the others from the list available to all of the machines.

With regard to getting the server engine running as fast as the hardware will allow, first make sure that the cache is as large as possible. Then analyse with the Pervasive Monitor utility the peak number of connections and files in use and adjust the settings accordingly.

6 Document History

23-12-2004: Original Document

19-06-2005: Amended to include differences in Pervasive.SQL 8.x and 9