Performance Comparison of PSQL v9 and PSQL v10

Performance and Memory

Database performance is strongly affected by the amount of memory available for caching data. Retrieving data from DRAM is orders of magnitude faster than retrieving data from disk. Simply put, the more cache an application can use, the better its performance. The best case for performance is when all of your accessed data can be kept in cache.

As your business expands, and you begin to track more information about customers, vendors and partners, the key to avoiding performance degradation is to take advantage of additional system memory. Additional memory allows you to grow cache as much as needed and thereby limit disk input/output (I/O).

In some cases, the easiest way to increase memory is simply to add more. Unfortunately, 32-bit hardware and operating systems have limits to the amount of memory they can access. For instance, Windows 2003 Server Standard Edition (x86) limits all applications to 2GB of addressable (virtual) memory, which includes the executable image. Therefore, adding more memory to a machine running that operating system will not necessarily translate into a direct performance boost to applications.

PSQL v9 runs only on 32-bit operating systems, and thus is constrained by the operating system to 2 GB of program memory, not all of which can be used for cache. Pervasive PSQL v10, however, provides two important features that significantly improve application performance.

This paper shows you how Pervasive PSQL v10 allows applications to take advantage of more memory in both 32-bit and 64-bit systems, providing big performance improvements over PSQL v9.

Performance Features in PSQL v10

The PSQL v10 release includes a 64-bit server engine and a component called XIO (eXtreme IO) with the 32-bit server engine. XIO expands memory capacity of 32-bit operating systems. The net result of these performance features is that Pervasive PSQL v10 provides access to much more cache than PSQL v9, reduces disk I/O, and significantly improves application performance, in some tests by a factor of 30 to 1.

Increasing Performance on 32-bit Systems

PSQL v10 includes a component called XIO (eXtreme I/O). This component uses memory beyond the 2 GB limit on 32-bit Windows operating system. XIO caches data files in that extra memory so that PSQL v10 does not have to read from disk as often. Less disk I/O means increased performance.

XIO can run on machines that meet a certain minimum configuration. Refer to the chapter "Performance" in the *Advanced Operations Guide* for details.

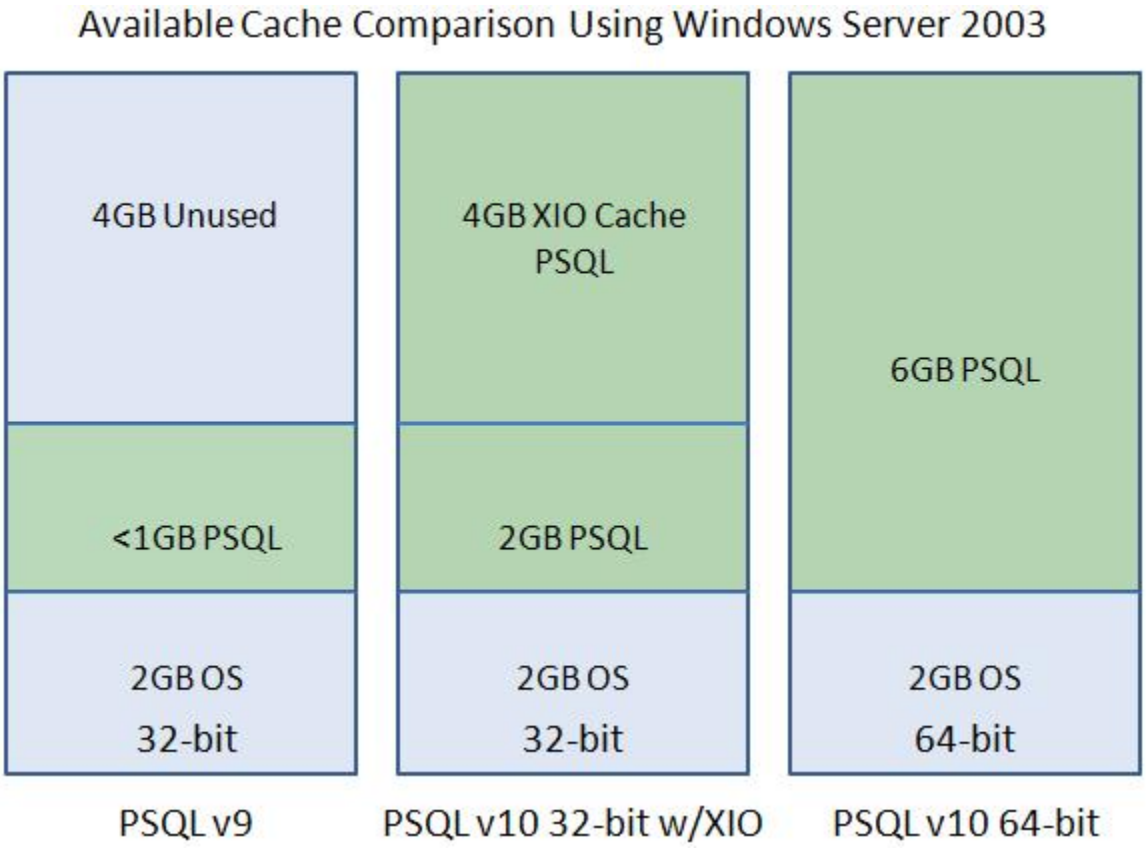
Increasing Performance on 64-bit Systems

PSQL v10 also supports 64-bit operating systems and can run as a 64-bit application (i.e. not just a 32-bit application running on a 64-bit OS). Because 64-bit operating systems support a huge amount of

physical memory (up to 4TB), running as a native 64-bit application means that PSQL v10 applications can use a cache orders of magnitude larger than possible on a 32-bit system.

Comparisons of 8 GB Memory Usage

The following diagram compares memory allocation for PSQL versions on 32-bit and 64-bit operating systems. For the purposes of the test (and to show the advantages of increasing cache size), we configured the database server with 8 GB of RAM, a single 2.7 GHz CPU and 32- or 64-bit Windows 2003 SP2. PSQL v9 and v10 were both installed with default settings.



Performance Testing

To demonstrate the effect of different cache sizes, Pervasive Software ran a test harness with a standard installation of PSQL on each of the above configurations. The test harness simulates a read/write load similar to the industry standard TPCB benchmark. A random record is read and updated in each of three files, and a log record is written to a fourth file. Sixteen harnesses run simultaneously on a single machine and measure throughput in transactions-per-second.

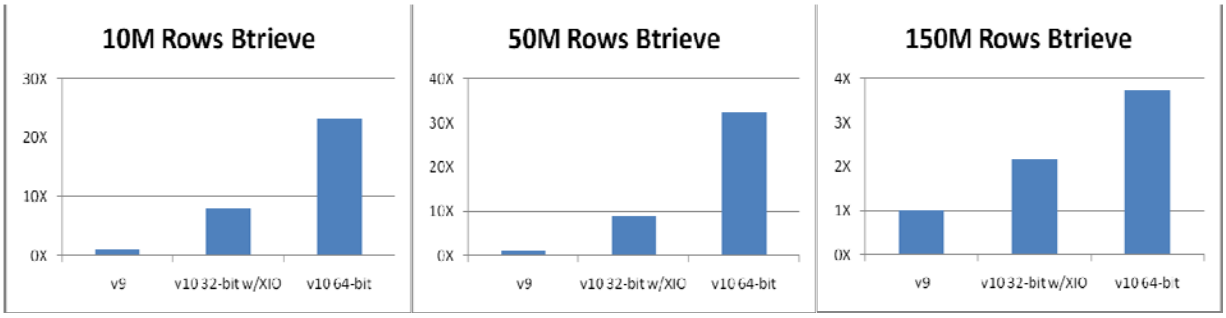
Data Sets

The data sets for this harness come in three sizes: 10 million rows, 50 million rows, and 150 million rows. The following table compares PSQL v9 and PSQL v10 behavior for each data set.

Data Set; Size	PSQL v9	PSQL v10 32-bit w/ XIO and 64-bit
10 million rows; about 1 GB	Somewhat larger than standard cache size. Test will be I/O bound.	Easily fits in cache. Test will only write new data.
50 million rows; about 6 GB	Exceeds standard cache size. Test will be I/O bound.	Slightly larger than standard cache size. Test will have some extra I/O to re-read data.
150 million rows; about 20 GB	Exceeds standard cache size. Test will be I/O bound.	Exceeds standard cache size. Test will be I/O bound.

Results for Btrieve Transactional Interface

The results for the harness that uses the Btrieve transactional interface are shown below. The PSQL v9 results are normalized to "1X" and the PSQL v10 results are shown relative to that. For example, we can see from the diagram below that in the Btrieve interface test with a 10 million row data set, PSQL v10 with XIO executed 7 times more transactions-per-second than PSQL v9.71.



In all cases, PSQL v9 is I/O bound and has the lowest throughput. For the 10M row (1GB) data set, PSQL v10 32-bit with XIO is able to use the extra memory of the XIO cache and thus achieves and 7 to 1 throughput improvement over PSQL v9. PSQL v10 64-bit is able to cache the entire data set directly and achieves a 23 to 1 throughput advantage over PSQL v9.

For the 50M row (6GB) data set, PSQL v9 and PSQL v10 32-bit with XIO hold proportionally less of the data set in main memory and must either read from disk or the XIO cache. XIO is still much faster than disk and PSQL v10 32-bit with XIO achieves 9 to 1 better throughput over PSQL v9. PSQL v10 64-bit can hold most of the dataset in memory and achieves a 32 to 1 throughput improvement over PSQL v9.

With the 150M row (20GB) data set, all of the cases are I/O bound. However PSQL v10 can still cache more of the dataset than PSQL v9 and achieve better throughput. PSQL v10 32-bit with XIO is about 2 times faster than PSQL v9, and PSQL v10 64-bit is almost 4 times faster than PSQL v9.

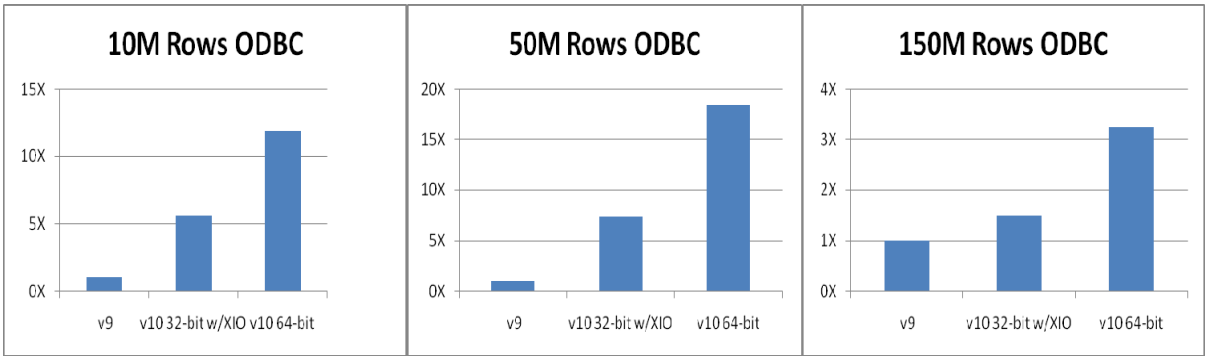
Results for ODBC Relational Interface

The results for the harness that uses the ODBC relational interface show a similar story with the biggest performance boost coming where PSQL can put the largest percentage of the data set into cache (and reduce disk I/O).

For the 10M row data set, PSQL v10 32-bit with XIO achieves a 5 to 1 throughput improvement over PSQL v9. PSQL v10 64-bit is able to cache the entire data set directly and achieves a 12 to 1 throughput advantage over PSQL v9.

For the 50M row data set, PSQL v10 32-bit with XIO achieves 7 to 1 better throughput over PSQL v9. PSQL v10 64-bit can hold most of the dataset in memory and achieves an 18 to 1 throughput improvement over PSQL v9.

With the 150M row data set PSQL v10 32-bit with XIO is about 1.5 times faster than PSQL v9, and PSQL v10 64-bit is over 3 times faster than PSQL v9.



Summary

PSQL v10 64-bit makes good use of the large memory available on today's computers, and PSQL v10 32-bit with XIO can take advantage of extra memory on a 32-bit operating system. Add to that support for Windows Vista, Windows 7 and Server 2008 and you have a formula for fast, modern access to your data by upgrading to PSQL v10.