Tech Tip #10: Viewing Btrieve File Statistics

Overview: How to view important file statistics from WBEXEC, BUTIL and ListStat

The PSQL database engine is extremely flexible and has a high level of backward compatibility, which is a great boon to developers using this product. However, it can also lead to end users seeing lower-than-expected performance from an environment because they forget to upgrade their environment to take advantage of new features.

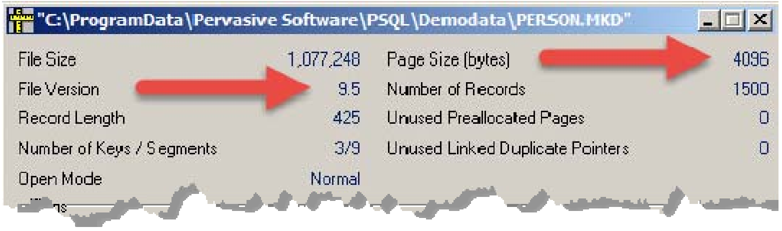
One such feature is the database file version. PSQLv11 supports reading of files in version 5.x and older, and both reading and writing of files in the formats 6.x, 7.x, 8.x, 9.0, and 9.5. If you have very old files in your system, so may not be able to take advantage of some key new features. One such feature is the maximum file size. A file in the 6.x format can grow to be only 4GB. However, 7.x/8.x files can grow to be 64GB, split into multiple 2GB extents. A 9.0 file can grow to 128GB, and a 9.5 file can get to 256GB – all in one segment. Another feature is write speed – files in the 8.x or newer format can leverage the *Turbo Write Accelerator*, a feature of the newer engines that speeds up disk writes.

Another feature is the page size. Some older systems may use "odd" page sizes (i.e. non-power-of-2 multiples of 512, like 1536, 3072, etc.) which don't efficiently use the database cache blocks. Many older applications also use very small page sizes. Small page sizes are OK for smaller environments, but in very large environments with huge data sets, these small accesses can mean that the server has to work harder to read data. Moving to a 16K page size can improve this read performance, with only a minor impairment to the writes.

In this Tech Tip, we're going to see three different ways to get the statistics for your data files. If you are not sure where your data files are located, use the **PSQL Monitor** as discussed in a previous Tech Tip.

The first way to locate the file statistics is with the **Function Executor**, which we discussed in our last Tech Tip. Simply open up a database file and click on the File Statistics button (🗎) to get the *Statistics* dialog, as shown in Figure 1.

You will see the statistics for the file, including the *File Version* and *Page Size* fields, in addition to other useful information such as the actual *File Size* and *Number of Records*.

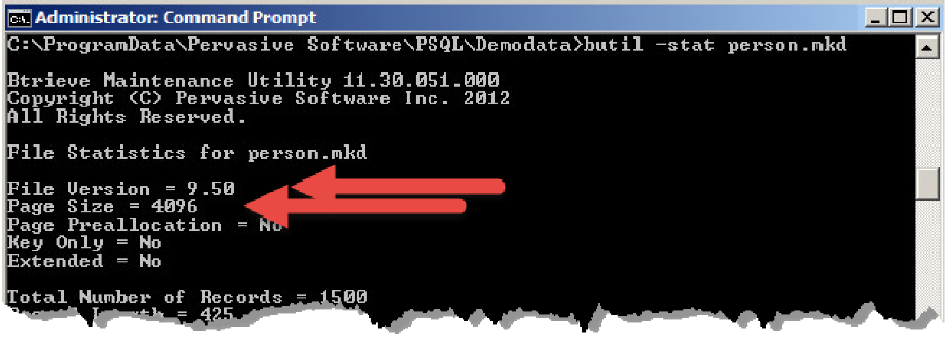


"C:\ProgramData\Pervasive Software\PSQL\Demodata\PERSON.MKD"

File Size	1,077,248	Page Size (bytes)	4096
File Version	9.5	Number of Records	1500
Record Length	425	Unused Preallocated Pages	0
Number of Keys / Segments	3/9	Unused Linked Duplicate Pointers	0
Open Mode	Normal		

Figure 1: File Statistics Dialog Box

For people who are familiar with the Windows command prompt, the **Maintenance Utility** may be substantially easier for you. First, open a **Command Prompt** window and navigate it to the directory that contains your data files.



Administrator: Command Prompt

```
C:\ProgramData\Pervasive Software\PSQL\Demodata>butil -stat person.mkd

Btrieve Maintenance Utility 11.30.051.000
Copyright (C) Pervasive Software Inc. 2012
All Rights Reserved.

File Statistics for person.mkd

File Version = 9.50
Page Size = 4096
Page Preallocation = No
Key Only = No
Extended = No

Total Number of Records = 1500
Record Length = 425
```

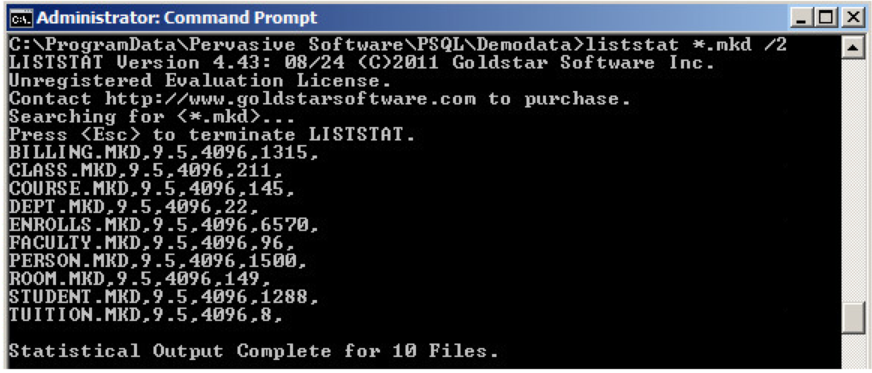
Figure 2: Result of BUTIL –STAT Command

Then, issue the command `BUTIL –STAT <filename>` to see the statistics for the indicated file. An example of this from the same table is shown in Figure 2. There is also a GUI version of the **Maintenance Utility**, but its dialog box (*Options*/*Create State Report*) is a bit confusing, so we'll skip it here.

If you have a large number of files, however, this process can get a bit tedious, since you must do each table separately. Again, Goldstar Software's large number of administrative tools can come to your rescue. One such tool, called **ListStat**, is designed to get a statistics report for a large number of files (from either one folder or an entire directory structure) and report on exactly what you want to see. Of course, it can also provide a report similar to **BUTIL**, but in a more compact format and without the large *Legend* section at the end. Download your 30-day trial of **ListStat** from here:

http://www.goldstarsoftware.com/ListStat.asp

Like **BUTIL**, we run **ListStat** from a *Windows Command Prompt*, but unlike **BUTIL**, we can supply wildcards and options to see exactly the data we care about. An example is shown in Figure 3 that is obtaining statistics about all of the MKD files in the current directory. However, we are using a special shortcut (/2) to obtain just the basic information about each file that might be important when we are upgrading a system. Therefore, in the output screen, instead of getting *all* of the information for each file, we only get a list that shows us the file name, the file version, the page size, and the number of records.



```
C:\ProgramData\Pervasive Software\PSQL\Demodata>liststat *.mkd /2
LISTSTAT Version 4.43: 08/24 (C)2011 Goldstar Software Inc.
Unregistered Evaluation License.
Contact http://www.goldstarsoftware.com to purchase.
Searching for <*.mkd>...
Press <Esc> to terminate LISTSTAT.
BILLING.MKD,9.5,4096,1315,
CLASS.MKD,9.5,4096,211,
COURSE.MKD,9.5,4096,145,
DEPT.MKD,9.5,4096,22,
ENROLLS.MKD,9.5,4096,6570,
FACULTY.MKD,9.5,4096,96,
PERSON.MKD,9.5,4096,1500,
ROOM.MKD,9.5,4096,149,
STUDENT.MKD,9.5,4096,1288,
TUITION.MKD,9.5,4096,8,

Statistical Output Complete for 10 Files.
```

Figure 3: Result of ListStat *.MKD /2 Command

Another Goldstar Software tool might be of interest to those who like to write batch files. The **BtrvInfo** tool is very similar to **ListStat**, but it reads ONE attribute from ONE file at a time and returns the data in %ERRORLEVEL%. This can be used to write batch files that conditionally do things based on the file version, page size, existence of keys, record counts, and more. Other tools like this can be found here:

http://www.goldstarsoftware.com/tools.asp

Of course, if you need a feature that one of these tools cannot handle yet, let us know!