actian®

Actian PSQL Defragmenter
Defragment to boost database performance

Key Benefits

Boost database performance

No engine down time

Use during normal operations

Identify heavily used files

Fix fragmented files

See before and after statistics

Does not alter data

Command line scripting

Zero DBA

Supported PSQL Platforms

v12 Server

v12 Vx Server

v12 Workgroup

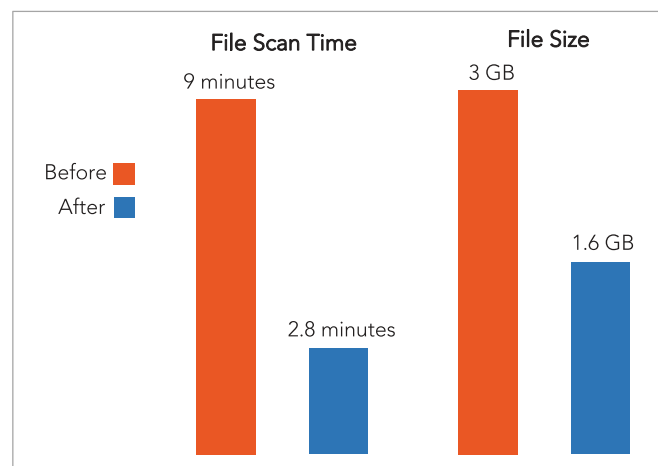## A new feature in PSQL v12

Over time, a busy database becomes fragmented as records are created, updated, or deleted. File access time grows and transaction response slows. This fragmentation is unlike that of hard disks because it occurs within the data files themselves. As a developer or DBA, you may know the files most likely to fragment from use, but in some cases, you may be guessing. Either way, manually rebuilding files means down time for your database and its applications. But doing nothing can result in declining application performance.

The PSQL Defragmenter utility solves these problems by finding data fragmentation and letting you correct it while the engine is running. It rebuilds indexes and removes unused space so that clients can access files efficiently again. Defragmenter does not alter data. It verifies successful defragmentation of every file before putting it into operation. Records can be created, read, updated, or deleted during defragmentation, so there's no need to disrupt normal database operations.

Opened from PSQL Control Center, Defragmenter lists files in use by read and write counts to alert you to the ones under heaviest traffic. Drag these files to the Watch List to analyze their degree of fragmentation. The analysis results display file size and percentage of fragmentation, unused space, and loss of record proximity. Pick the files where defragmenting will help the most and fix them – without stopping your PSQL engine.

Database transactions can run more quickly against newly compacted, reindexed files. Regular use of Defragmenter helps maintain efficiency, capacity, and performance.

Before and after defragmentation

File Scan Time — Before: 9 minutes, After: 2.8 minutes

File Size — Before: 3 GB, After: 1.6 GB

Before (orange), After (blue)

Here are the results of a test of a heavily used file of 2.5M records, before and after defragmenting.

Defragmenting provides faster reports and benefits for storage, backup, and data capacity licensing.

Defragmentation considerations

Every database is different. Good defragmenting depends on knowledge and experience with your database. Here are some general tips:

- Defragmenter requires free disk space equal to twice the size of any files to be defragmented. Files undergoing intensive write activity during defragmenting may require still more disk space.
- In databases where many writes occur, analysis of watched files may show rising fragmentation. If you notice changes in performance, such as slower queries and reports, you may need to defragment.
- Bulk delete actions greatly increase unused file space. Defragment to remove space and compact files.
- The dbdefrag command line utility offers scripting options to automate a defragmentation schedule.
- You can fix fragmented data files only on the system where Defragmenter runs.
- Defragmenter makes no changes to system data and the key used by DataExchange. After defragmenting, you do not need to run the table synchronization and check utility (dxsynctables).

If defragmenting does not improve performance, then your problem likely lies elsewhere and will need another diagnosis and solution.

Performance while defragmenting
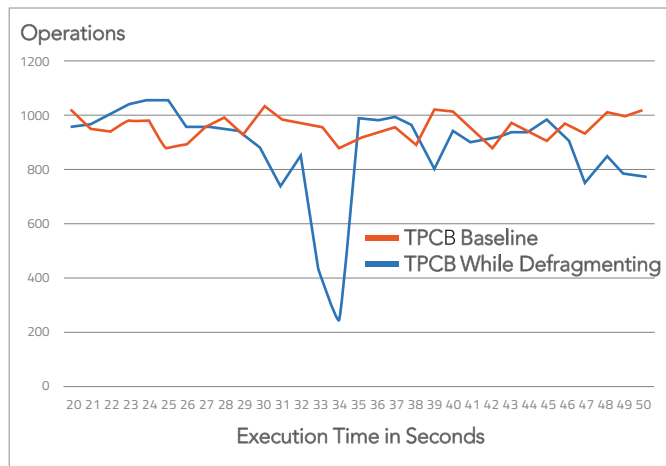
Reads and writes continue uninterrupted during defragmenting, which runs at lower priority than normal operations. The following chart compares a standard TPCB benchmark test with and without Defragmenter execution.

During high traffic periods you may see brief effects on performance. If this is a concern, then consider low traffic times for your defragmentation strategy.