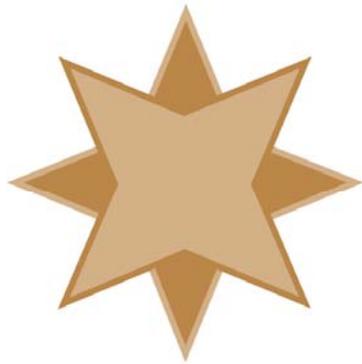


Using the Online Database Defragmenter

A White Paper From



**GOLDSTAR
SOFTWARE**

www.GoldstarSoftware.com

For more information, see our web site at
<http://www.goldstarsoftware.com>

Using the Online Database Defragmenter

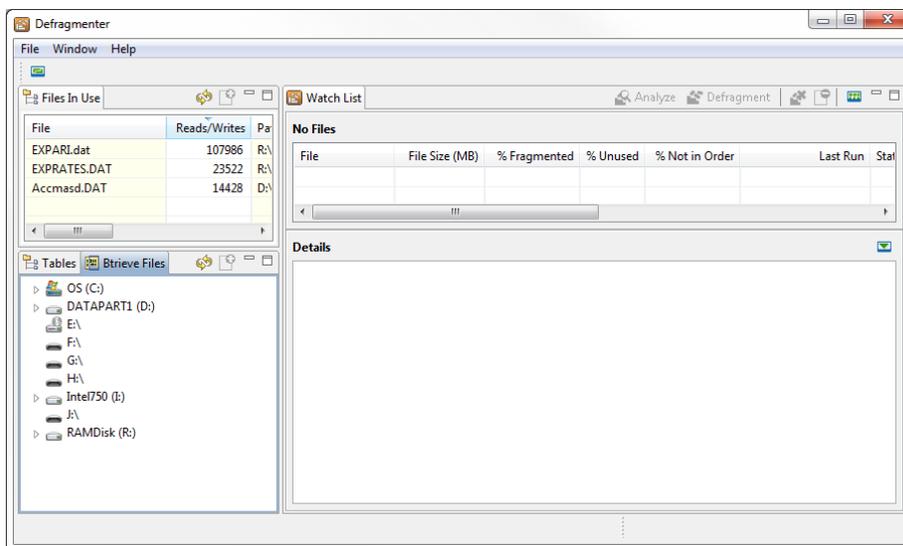
Last Updated: 12/18/2015

Action included a brand new tool as part of the PSQL v12 database engine environment – the Online Database Defragmenter (ODD). This tool makes it possible for you to analyze your database files as they are being used, and to then defragment them, reducing the file size and often increasing the engine’s efficiency at the same time.

Using the Defragmenter can also be a great help when you are using PSQL Vx Server, because you can maintain your files in the smallest possible file size and thus save on licensing costs.

Starting the Defragmenter

Launching the Defragmenter is easy – just go to the PSQL Control Center, select Tools, then Defragmenter, and you’ll get to the home screen that looks like this:



Selecting Files for the Watch List

The first step in a defragmentation job is to locate the files you want to consider and add them to the **Watch List**. Once a set of files has been added to the watch list, you can further analyze and defragment those files.

The upper-left quadrant of the Defragmenter dialog box (**Files In Use**) will show you the database files recently opened by the database engine, as well as the number of reads and writes that have taken place against those files. Sorting the data by the **Reads/Writes** column can make it easy to determine which files are likely to make the biggest difference in your system when they are optimized.

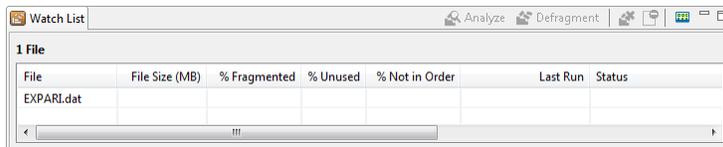
Information Provided By **Goldstar Software Inc.**

<http://www.goldstarsoftware.com>

To add a file to the watch list from the **Files In Use** window, you can either right-click the file and select **Add to Watch List**, or you can drag & drop the file onto the **Watch List** window to the right. Of course, you can also select multiple files with *Shift-Click* and *Ctrl-Click* to add many files at once.

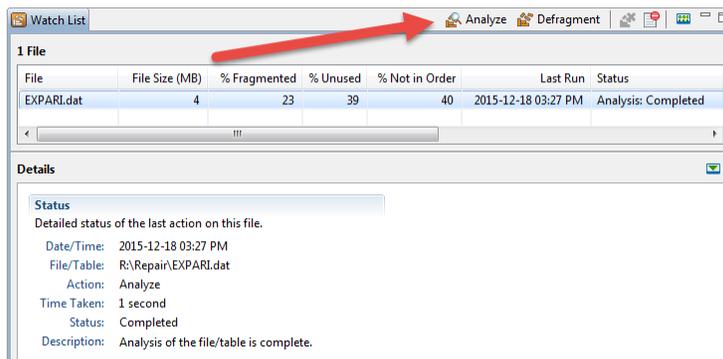
If you want to add a file to the watch list that is not currently open, you can use the **Btrieve Tables** view in the lower left quadrant and navigate to the file directly. If your file is defined inside a SQL database, you can change that view to the **Tables** view, and drill down to the database name and table name that you want to add. Again, either right-click and select **Add to Watch List**, or drag & drop the file to the **Watch List** window.

When adding a new file to the watch list, the Defragmenter will check that file to make sure that it can be defragmented in the first place. Assuming that the file is OK, you will now see your file(s) in the **Watch List** window:



Analyzing Your Files

The next step is to analyze your database files and see which ones (if any) warrant a defragmentation. In the **Watch List** window, select the file or files that you want to analyze and click the **Analyze** button in the toolbar.



When the analysis is complete, you will see the metrics for the file completed in the watch list. In this case, the file is currently 4MB in size, it is 23% fragmented, has 39% unused space, and 40% of the records are not in order. If these numbers are high, then defragmentation will likely be beneficial.

If you would like additional information about your database file, you can use Goldstar Software's KeyData tool to obtain more detailed metrics about the file. Below is a sample screenshot from this utility with the most important metrics highlighted with red arrows. If these values are high, then defragmentation is definitely recommended.

Information Provided By **Goldstar Software Inc.**

<http://www.goldstarsoftware.com>

```

R:\Repair>c:\KeyData EXPARI.DAT
KeyData Version 1.34: 11/09 (C)2015 Goldstar Software Inc.
Registered to Goldstar Software Inc. (Site License GS)
Opening file EXPARI.DAT at 15:34:36 on 2015-12-18..

File Version          : 950
Record Count          : 14616 Records
Total Pages In File   : 1204 Pages
Page Size             : 4096 Bytes
Computed File Size    : 4931584 Bytes

Page Size             : 4096 Bytes
Overhead Per Data Page : 10 Bytes
Logical Record Length : 128 Bytes
Internal Record Length : 130 Bytes
Records Per Page      : 31
Waste Per Data Page   : 56 Bytes ( 1.367% of Page, 43.077% of Record)
Lowest Waste Page Size : 4096 Bytes (43.077% of Record, 31 Records/Page)

Overhead Page Count   : 12 ( 0.997%)
Shadow Page Count     : 484 (40.199%)
Page Error Count      : 0
Logical Page Count    : 708
  Data Pages          : 507 (42.110%)
  Index Pages         : 188 (15.615%)
  Extra Pages         : 13 ( 1.080%)
  Variable Pages      : 0 ( 0.000%)
  Unknown Pages       : 0 ( 0.000%)

Number of Data Pages  : 507 Pages
Records Per Page      : 31 Records
Total Record Space    : 15717 Records
Actual Record Count   : 14616 Records
Unused Record Space   : 1101 Records ( 7.005%)

Key Number Selected   : 0
Logical Key Length    : 4
Physical Key Length   : 12
Total Key Pages Scanned : 85
Maximum Key Page Depth : 2
Total Key Entries Found : 14616
Total Key Page Bytes   : 348160
Total Overhead Bytes  : 1360
Total Key Space Used   : 175392
Total Key Space Wasted : 171408
Percentage Space Wasted : 49.23%

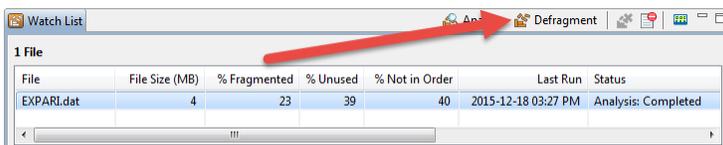
# Keys at BTree Level 1: 83 (1 page)
# Keys at BTree Level 2: 14533 (84 pages)
=====
Key Number Selected   : 1
Logical Key Length    : 8
Physical Key Length   : 16
Total Key Pages Scanned : 103

```

For additional information on the other metrics reported by KeyData, see the documentation that came with the download.

Defragmenting Your Files

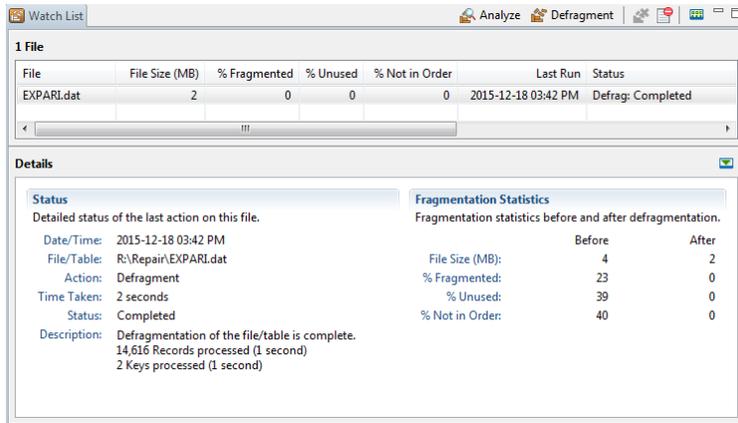
Once you have analyzed your data files and identified the ones that you want to defragment, you can continue with the defragmentation process. To do this, select the files in the Watch List window that you want to defragment, and click on the Defragment button, then watch the Defragmenter do its thing!



Note that the defragmentation process is considered a background process controlled by the engine itself. It will throttle itself back during periods of heavy load, and it will only run a limited number of files at a time. During the defragmentation of any given file, you cannot use Continuous Operations to back up this file, and you cannot take a snapshot of

it with the VSS Writer. As the file will be open for the duration of the process, any attempt to delete or rename the file by the application will also fail. Needless to say, you should be very particular about the files you defragment, as well as the times at which you start up a defragmentation.

When the process is completed, the screen will be updated with the new file metrics, and the **Details** window will also show you the statistics for that file:



As you can see, the file has shrunk from 4MB to 2MB, the unused space is gone, and the records are now in order. The process took only 2 seconds, and the file has been substantially optimized.

We can confirm these changes with the KeyData utility as well. Note that the shadow pages and unused record space have both diminished by quite a bit (but will never quite be eliminated completely). Further, the metrics for Key 0 also show that the index has been optimized, and a substantial amount of index pages have been reclaimed as well. All in all, this was a good use of the Online Database Defragmenter.

```

R:\Repair>cd:KeyData EXPARI.DAT
KeyData Version 1.34: 11/09 (C)2015 Goldstar Software Inc.
Registered to Goldstar Software Inc. (Site License GS)
Opening file EXPARI.DAT at 15:48:06 on 2015-12-18..

File Version          :          950
Record Count          :        14616 Records
Total Pages In File   :          588 Pages
Page Size             :        4096 Bytes
Computed File Size    :    2408448 Bytes

Page Size             :        4096 Bytes
Overhead Per Data Page :         10 Bytes
Logical Record Length :         128 Bytes
Internal Record Length :         130 Bytes
Records Per Page      :          31
Waste Per Data Page   :          56 Bytes ( 1.367% of Page, 43.077% of Record)
Lowest Waste Page Size :        4096 Bytes (43.077% of Record, 31 Records/Page)

Overhead Page Count   :          12 ( 2.041%)
Shadow Page Count     :           1 ( 0.170%)
Page Error Count      :           0
Logical Page Count    :          575
  Data Pages          :        472 (80.272%)
  Index Pages         :        103 (17.517%)
  Extra Pages         :           0 ( 0.000%)
  Variable Pages      :           0 ( 0.000%)
  Unknown Pages       :           0 ( 0.000%)

Number of Data Pages  :          472 Pages
Records Per Page      :          31 Records
Total Record Space    :        14632 Records
Actual Record Count   :        14616 Records
Unused Record Space   :          16 Records ( 0.109%)

Key Number Selected   :           0
Logical Key Length    :           4
Physical Key Length   :          12
Total Key Pages Scanned :          44
Maximum Key Page Depth :           2
Total Key Entries Found :        14616
Total Key Page Bytes   :       180224
Total Overhead Bytes   :          704
Total Key Space Used   :       175392
Total Key Space Wasted :          4128
Percentage Space Wasted :          2.29%

# Keys at BTree Level 1:         42 (1 page)
# Keys at BTree Level 2:       14574 (43 pages)
=====
Key Number Selected   :           1
Logical Key Length    :           8
Physical Key Length   :          16
Total Key Pages Scanned :          59

```

Using the Command Line Utility

In addition to the GUI Defragmenter utility, you can also use the command line utility called *dbdefrag*. To analyze a file, use this command:

```

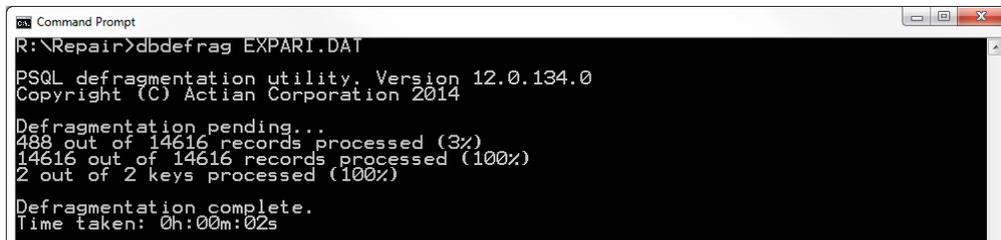
R:\Repair>dbdefrag -analyze EXPARI.DAT

PSQL defragmentation utility. Version 12.0.134.0
Copyright (C) Actian Corporation 2014

Fragmented: 0%
Unused: 0%
Not in Order: 0%
File Size: 2 MB

```

Note that you will get the same metrics, just on a simple text screen. You can then launch the defragmentation process by dropping the *-analyze* option:



```
Command Prompt
R:\Repair>dbdefrag EXPARI.DAT
PSQL defragmentation utility. Version 12.0.134.0
Copyright (C) Actian Corporation 2014
Defragmentation pending...
488 out of 14616 records processed (3%)
14616 out of 14616 records processed (100%)
2 out of 2 keys processed (100%)
Defragmentation complete.
Time taken: 0h:00m:02s
```

If you want to run the defragmentation in the background so that you can keep working, then just add the `-background` switch before the file name.

Automating the Defragmentation Process

The command line *dbdefrag* tool can be used to automate the defragmentation process (by calling a batch file from the Windows Task Scheduler), if you know that you want to defragment a file or set of files every so often. However, you should note that running excessive defragmentation processes is not recommended, as it can generate extra load on the server, slowing down other operations. To that end, you should intelligently run the defragmenter **ONLY** on the files on which it will make a big difference.

At this time, however, Actian has not yet released any way to automate or schedule the defragmentation process intelligently, and if you run a defrag job every week, you may be defragmenting files that really don't need it.

If automation is important to you, you might want to check out Goldstar Software's SmartODD utility, released in February 2015. This tool reads a configuration file which contains the files you want to analyze, runs the analyzer to determine the metrics for each data file, compares the metrics with a limiter value that you specify, and if the file exceeds the limit value, the defragmentation job is scheduled to run. It can then send you a report of what transpired via Email, so that you stay in charge of your environment. You can get more information about this tool from www.goldstarsoftware.com/tools.asp.

If you still can't get it to work, [contact Goldstar Software](#) and let us work with you to help! Please note that this may be a billable support call if you have already used up your free support time.