

User Manual for HandyFileMonitor v0.9

Program activity monitoring application

Program activity monitoring is extremely important for many users. Maintenance utilities, many server and client applications, or real time calculations require continuous operation of a running process. In this case, no matter how reliable the program is, errors or unexpected emergency shutdowns may still occur- for this reason there are many monitor applications allowing in the case of crashes executing scripts, restarting program or different ways of notifying the user about the occurred error.

Different applications of program activity monitoring use different methods of control. Most of them check in pre-set intervals the availability of the basic process of the program in memory or the use of computer system resources by them - it is a quite reliable and convenient method. But there are times when this method is not applicable. For example, if a program freezes or has issued a critical error, the process in the memory remains, but the operations are no longer performed. In this case, other program testing criteria are necessary.

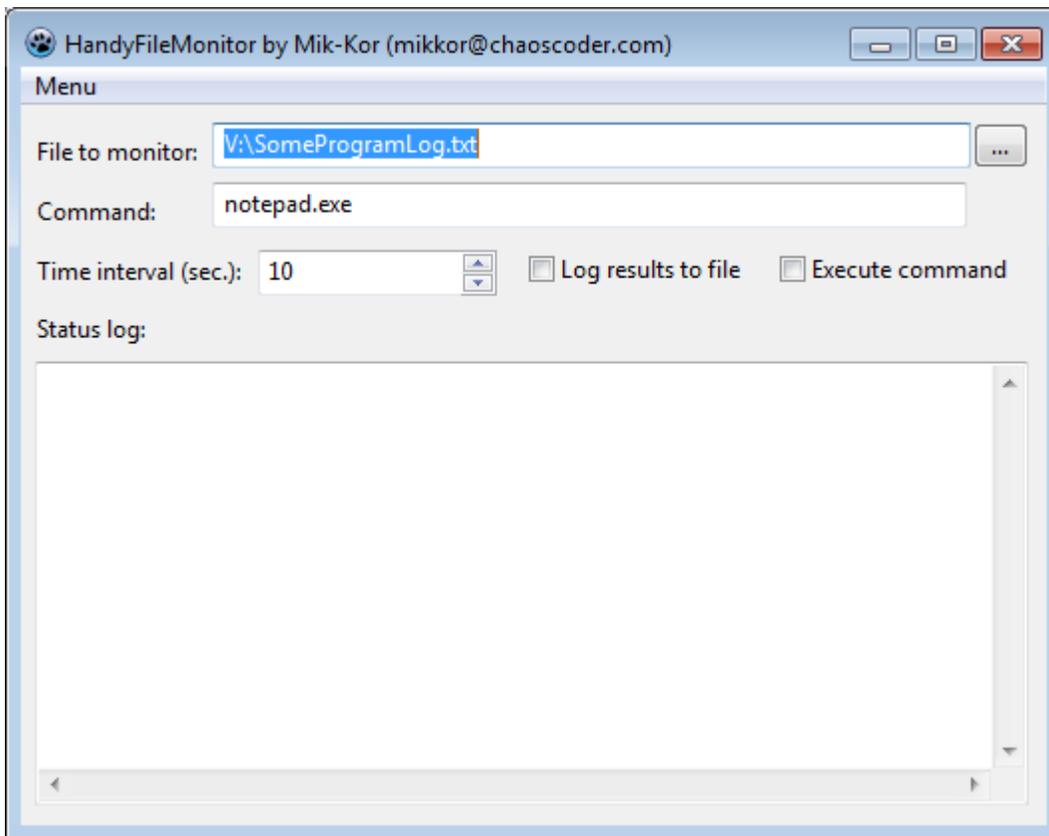
For example - the program HandyFileMonitor, can monitor the activity of any application that has a log file. This application is based on a simple rule: if the size of the log file increases, the program works, and if the size of the log file is unchanged - the program either accomplished all current tasks, or stopped working because of an error or hang. For the cases where the program should run until the user stops it (various modules of data stream-processing, distribution, network scanning utilities and other software) using HandyFileMonitor is more convenient.



(Fig. 1. Informative window)

Application Interface and Functionality

HandyFileMonitor has a simple and intuitive interface (see Fig. 2). The main program window contains the fields to select the program log file to be controlled, specify the command to be performed when the program is stopped, select the check frequency and display the checked log file statistics.



(Fig. 2. HandyFileMonitor Interface)

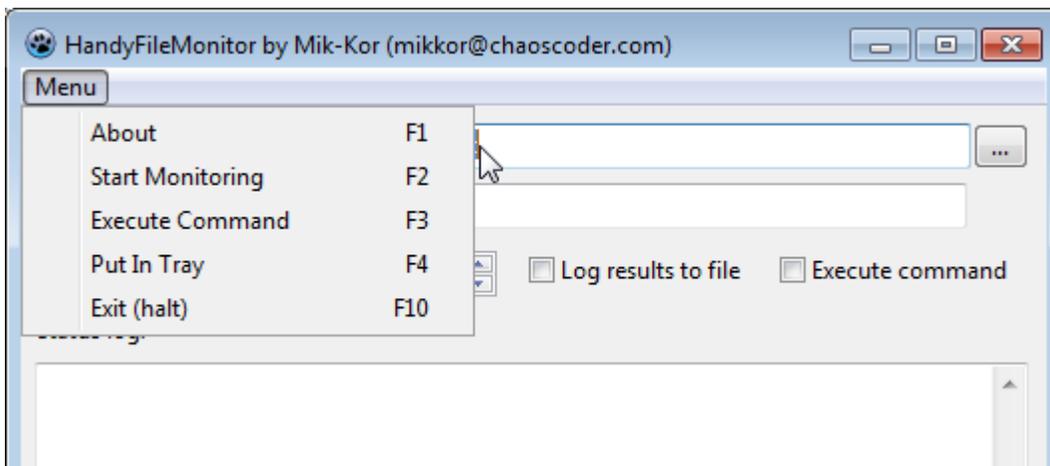
Program Interface Elements (top to bottom):

- **File to monitor (monitoring file)** – a field to select a log file of the controlled program. It can be filled in manually or selected with the standard file manager window of the operating system by pressing the "..."
- **Command** – field for specifying the command or the path to the executable file or script. In the event the checked log remains unchanged and the option «Execute command is checked» (see item 5), HandyFileMonitor automatically runs the specified program or script. This option is highly demanded to reboot program automatically restored with the data automatically resume working with the data.
- **Time interval (sec.) (time interval in seconds)** – in this field you can set the interval for checking the log file. It should be taken noted that in case when the running program saves its log in long intervals of time, making checking intervals shorter does not make sense.

Also, the checking interval should not be short when «Executecommand» is checked (see item 5), since executing scripts or restarting the program also takes time.

- **Logresultstofile (save the results to a file)** – this option allows the program to keep its own log file, with the information about the analyzed file: the date and time of analyze, the total size of the file in bytes, the amount of added data in bytes since the last request.
- **Executecommand (execute a command)** – using this option, you can execute any external command or script in the case if since the last inspection of the protocol of the controlled program no new information was added and an error or a crash is suspected.
- **Filestats (file statistics)** – this field will host in real-time the information that HandyFileMonitor keeps in the protocol: file scanning date and time, total and accumulated data volume since the last test.

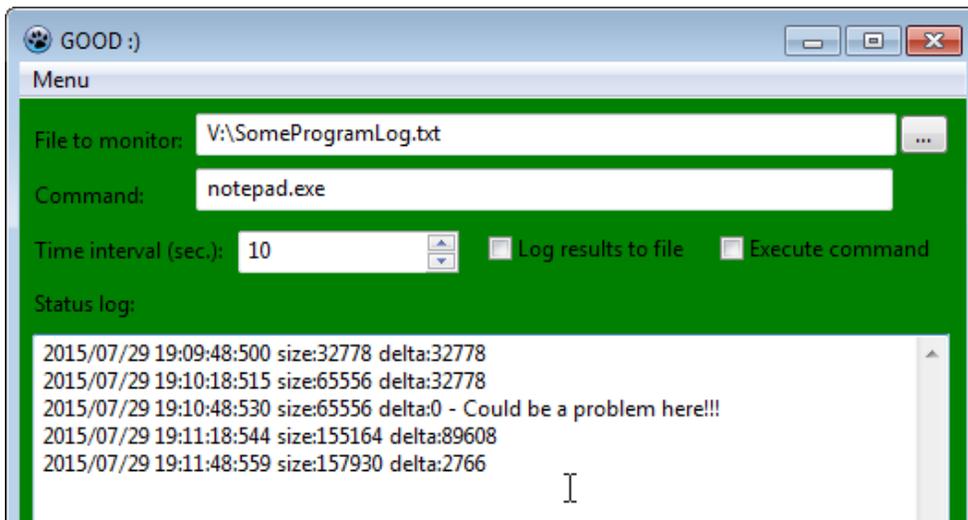
Program menu contains control commands required for application management (see Fig. 3).



(Fig.3. Menu and controls)

- **About - F1.** Displays the reference window (see Fig. 1).
- **StartMonitoring / StopMonitoring) - F2.** Starts the monitoring process or stops it if it is already running.
- **ExecuteCommand - F3.** Manual launch of command or script specified in the «Command» field.
- **Put in Tray – F4.** Put application in tray and hide the main window.
- **Exit (halt) (exit (stop)) - F10.** Instant shutdown of HandyFileMonitor.

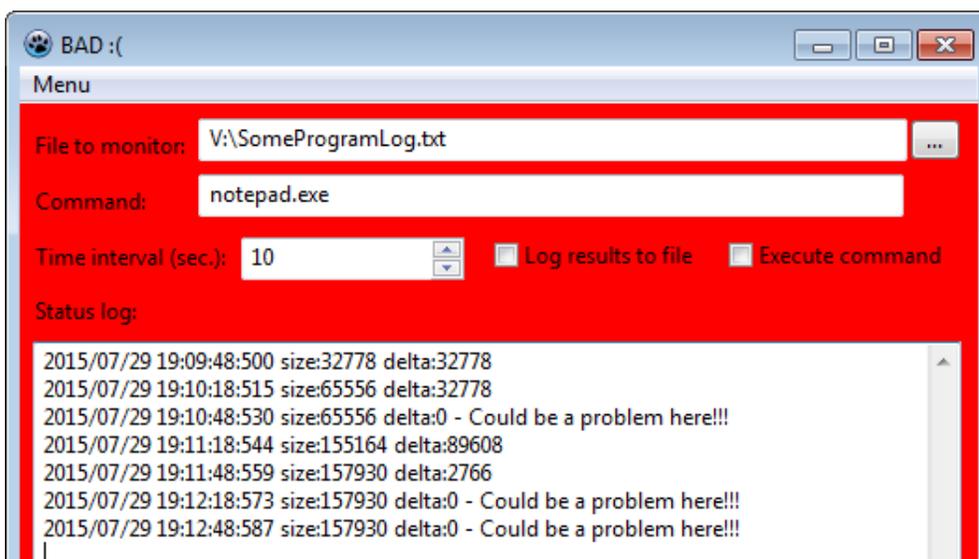
Start using the application is extremely easy - just select a file to control and specify the checking interval in seconds. It is possible to optionally specify a script or a program to execute in case there is no change in the log of the monitored program and also indicate the program to run its own log. After all the necessary settings are done, the monitoring can be started by selecting the appropriate menu item or by pressing a hot key. The application starts running (see Fig. 4).



(Fig. 4. Application operation. The monitored program is active)

FileStats field will be filled with information about the state of the monitored lines of the log at specified intervals. The order of priority are: date, time (up to a thousandth of a second), the size of the file in bytes (size), changing the size in bytes (delta). When monitored file keeps growing in size (delta remains bigger than 0) window title will change to "GOOD" and the form color will be green.

In the case the size of the monitored file stops growing (assuming that some log generating application stopped working), the HandyFileMonitor window background will become red and the window title - «BAD»: (. If the option «Execute command» is checked then a program or a script specified in the field «Command» will be launched. The value of delta will be less or equal to zero in the string displayed in the «Status logs» field, (since there is no increase in the monitored file) and the text «Could be a problem here!!!» will be added. The program window in this mode is shown in Fig. 5.



(Fig. 5. Operation of the application. The monitored application is not active)

HandyFileMonitor possible applications

The application HandyFileMonitor, despite its apparent simplicity, is a convenient activity program monitor. For example, you can create a script to restart the program as shown in the example:

```
S:\somefolder\pskill someprocess  
S:\somefolder\someprocess.exe
```

In the case, where the volume of the monitored log file stops increasing, the program ends and then restarts. This helps restarting it in case of errors or freezes without the direct involvement of the user, eliminating the need to continuously monitor a running program. This can be useful in some cases, for example, in the absence of physical access to the computer that is running the program and the inability to connect to it through a remote access.

Moreover, the user can create a website and a guestbook, and use HandyFileMonitor to set a script in order to automatically add notifications to them when a controlled stop of the program is detected. Thus, a user at any time and from any mobile device, by going to his website can check the exact time of program stops and restarts. Here the script line will look like the following:

```
S:\somefolder\wget.exe http://www.somesite.com/guestbook.php --post-data="message=Process I-  
Log_file_stopped_growing"
```

Using a variety of scripts allows setting up an automatic notification for the program shut-down to an e-mail or a phone through SMS.

HandyFileMonitor also has a number of advantages compared to other applications:

- Easy to set up. No redundant parameters, and at the same time all the necessary functionality. At shutdown all the settings are stored in a file - no need to configure the application each time.
- Portability. This application does not require installation, does not register information to the computer registry and can be launched from any mobile carrier. Start-up file size is less than two megabytes.
- The possibility to run simultaneously multiple HandyFileMonitor copies, allowing the monitoring of several programs simultaneously, and setting different check intervals.