IISPools

IIS Application Pools Monitoring - IISPools

- This program allows monitoring of IIS 6.0 application pools.
- All the monitoring data is stored in log files.
- It is possible to analyze a pool to find a website (or a virtual application) using too much CPU/RAM resources.

Note: Current version of the program allows monitoring and logging of both CPU and RAM usage. But there is an auto-analysis of CPU usage only.

Important: IISPools must be run with administrator rights as it needs to manage IIS.

Logging

Logged data:
- Date and time
- Application pool name
- Number of applications in the pool
- RAM usage, kB
- CPU usage, %
- Threads number in the pool
- Number of open files being used at the moment
- Number of open TCP/IP connections being used at the moment
- Number of open system descriptors being used at the moment
- PID (process ID)

You can see the log in the program interface and it is also saved to a file.

Monitoring

In addition to information mentioned in "Logging" paragraph above, the following data is shown during the monitoring process: website name and ASP.NET version of the application.

Important: It takes quite a long time to retrieve information about Microsoft IIS 6.0 applications programmatically. Therefore it is not recommended to start monitoring of a pool containing lots of applications. IISPools application may be working very slowly (and even hung) in that case. Unfortunately it is not possible to determine how many pools is "lots of" exactly, but on our test server it is about 300 applications.

Pool analysis

WARNING! Sometimes a pool analysis may cause stopping of some virtual applications of the pool. Of course, it means downtime. Now you are aware of. And we recommend analyzing the pools when downtime is acceptable. You can restart the stopped applications, but IISPools won't do it itself automatically. IISPools is a tool for Windows server administrators and you should use it at your own risk and only if you understand what you are doing.

All the operations during the pool analysis are always logged.

Important: The more virtual applications the pool contains the more free RAM is needed to perform the analysis. It takes 1MB RAM per virtual application **at least**.

Pool analysis algorithm (Please spend a minute to read it and your analysis will be efficient.):
- An administrator selects a pool to be analyzed. (The purpose is to find out those virtual applications using too much CPU time).

- IISPools creates temporary application pools, one pool per website and virtual app. (So if there are 40 websites and apps in the pool then 40 new temporary pools are created.)
- IISPools distribute websites and virtual apps by moving them to the temporary pools one by one.
- The initial pool (analyzing one) is stopped by the program forcedly. (It is the way Microsoft IIS works.)
- IISPools starts monitoring and analyzing of the temporary pools.
- If, at some moment, one of the pools has more than specified CPU load, IISPools makes a record in the log.
- It continues until stopped by the administrator.
- Once the analysis is stopped, IISPools moves all websites and virtual apps from temporary pools to the old one and then remove the temp pools.
- The administrator looks at the results, making a thoughtful face. ☺

Important: Do not forget to stop the analysis, otherwise the applications will stay in the temp pools.

All the errors are logged, including the pool creation and applications transfer problems. So it is recommended to look through the log once the analysis is finished. Moreover it is useful to look at the log during the analysis starting and periodically during the process.

An example of the analysis:
Let's say you need to analyze a pool named DefaultAppPool containing 5 virtual applications.
- IISPools will create 5 temporary pools and distribute the virtual applications among them. The pools are named DotnetparkAnalyse*[SomeBigRandomNumberHere]* (e,g, *DotnetparkAnalyse7256).*
- The source pool (DefaultAppPool) will be disabled.
- IISPools will start monitoring of those *DotnetparkAnalyse????* pools.
- If, at some moment, one of *DotnetparkAnalyse????* pools has more than specified CPU load, IISPools makes a record in the log.
- Once the analysis is stopped by you, all 5 applications will be moved back to DefaultAppPool and all *DotnetparkAnalyse????* pools deleted.
- Then you should look through the log and make your decision.

Description of the Pool Monitoring interface
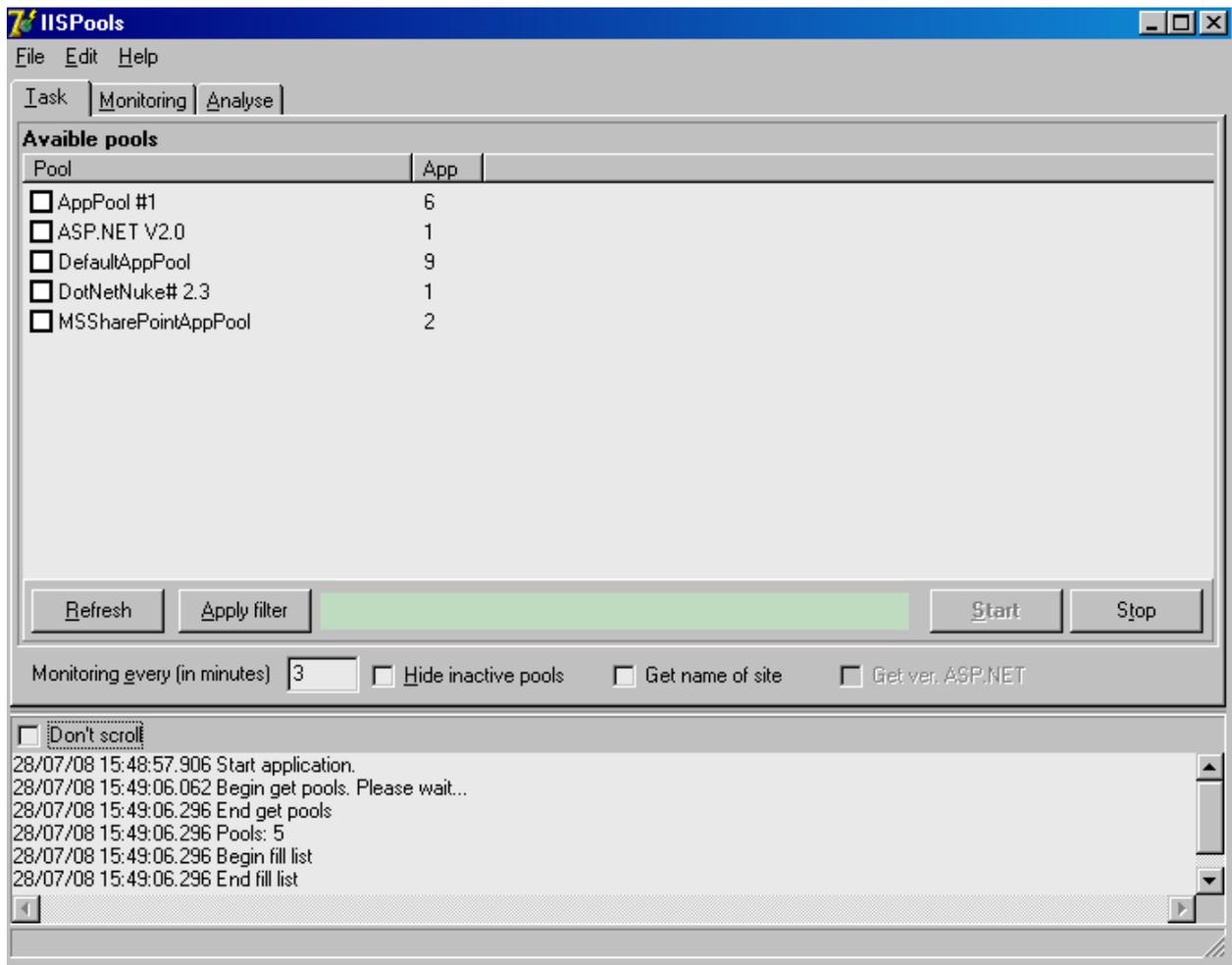
Press [Refresh] button to see the list of application pools of your server.

At the bottom of this window you can see the log showing the monitoring and analysis related information.

The "Apps" column shows the number of virtual applications in corresponding pool.

If there are many pools and you wish to see only some of them, you can set the filter. Just type **an expression** like *Default* in the green field and press [Apply filter] button, and you'll see only the pools containing *Default* word in their name. When we say "an expression", it means "any regexp expression" actually. But usually it is enough to use * which means "any symbols here". Clear the field and press [Apply filter] button to see all your pools again.

To start monitoring of a pool:
- Set an interval in "Monitoring every…" field
- Check the pool in the list
- Press [Start] button.

The monitoring log and results will be shown on "Monitoring" tab and saved in iisp-mon-XXXX.log file in LOG folder. (XXXX will be replaced with numbers)

"Hide inactive pools" check box allows to hide all stopped pools.

"Get name of site" check box allows to see website names. (By default there is ADSI path only.)

"Get ver. ASP.NET" check box allows to see ASP.NET versions of all applications. **Important:** It takes quite a long time to retrieve ASP.NET version of Microsoft IIS 6.0 applications programmatically. Therefore it is not recommended to enable this check box if there are lots of applications in the pool and the monitoring interval is less than 5 minutes.
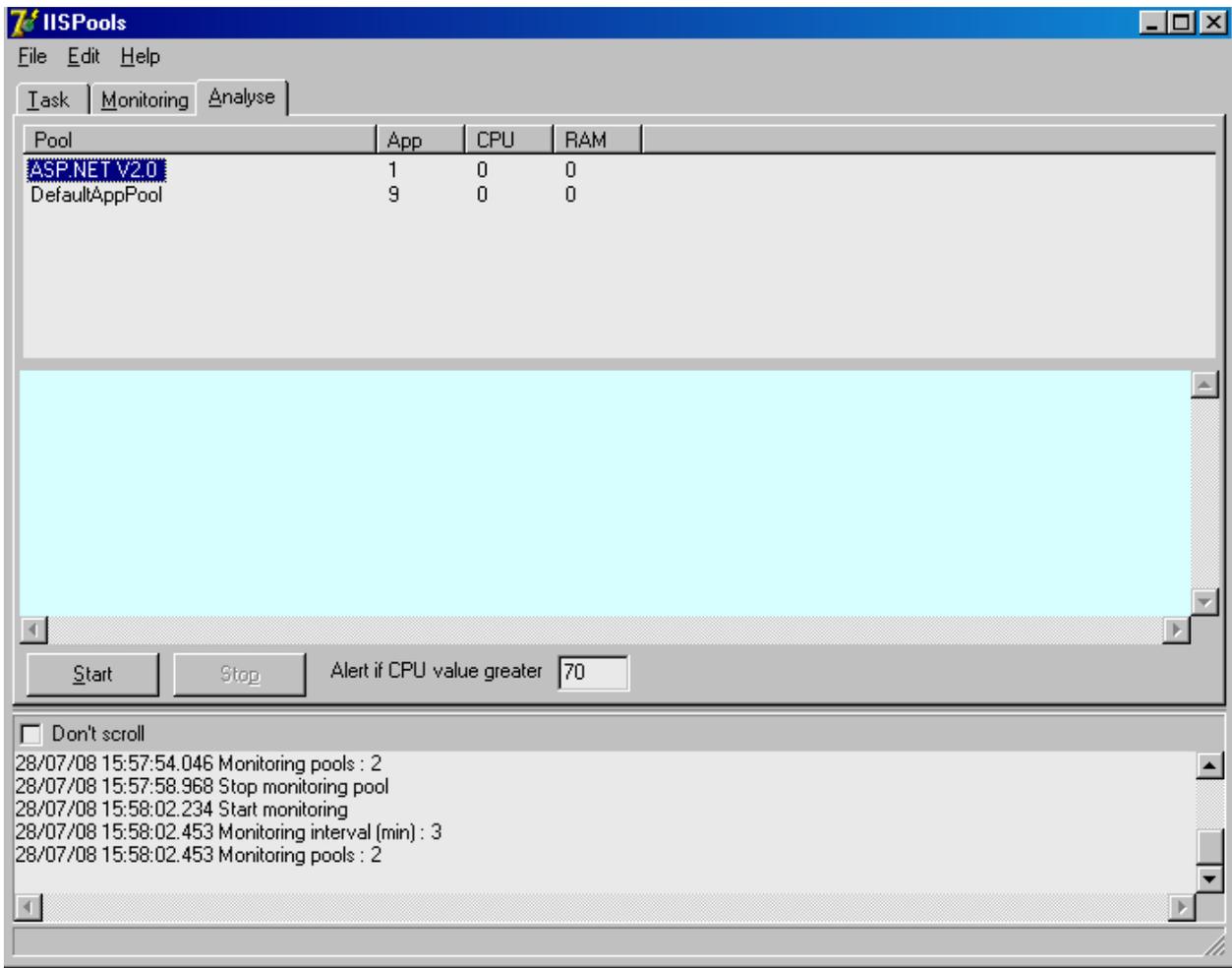
[Stop] is for the monitoring stopping.

You can see a sample of the monitoring results in the fig. below:
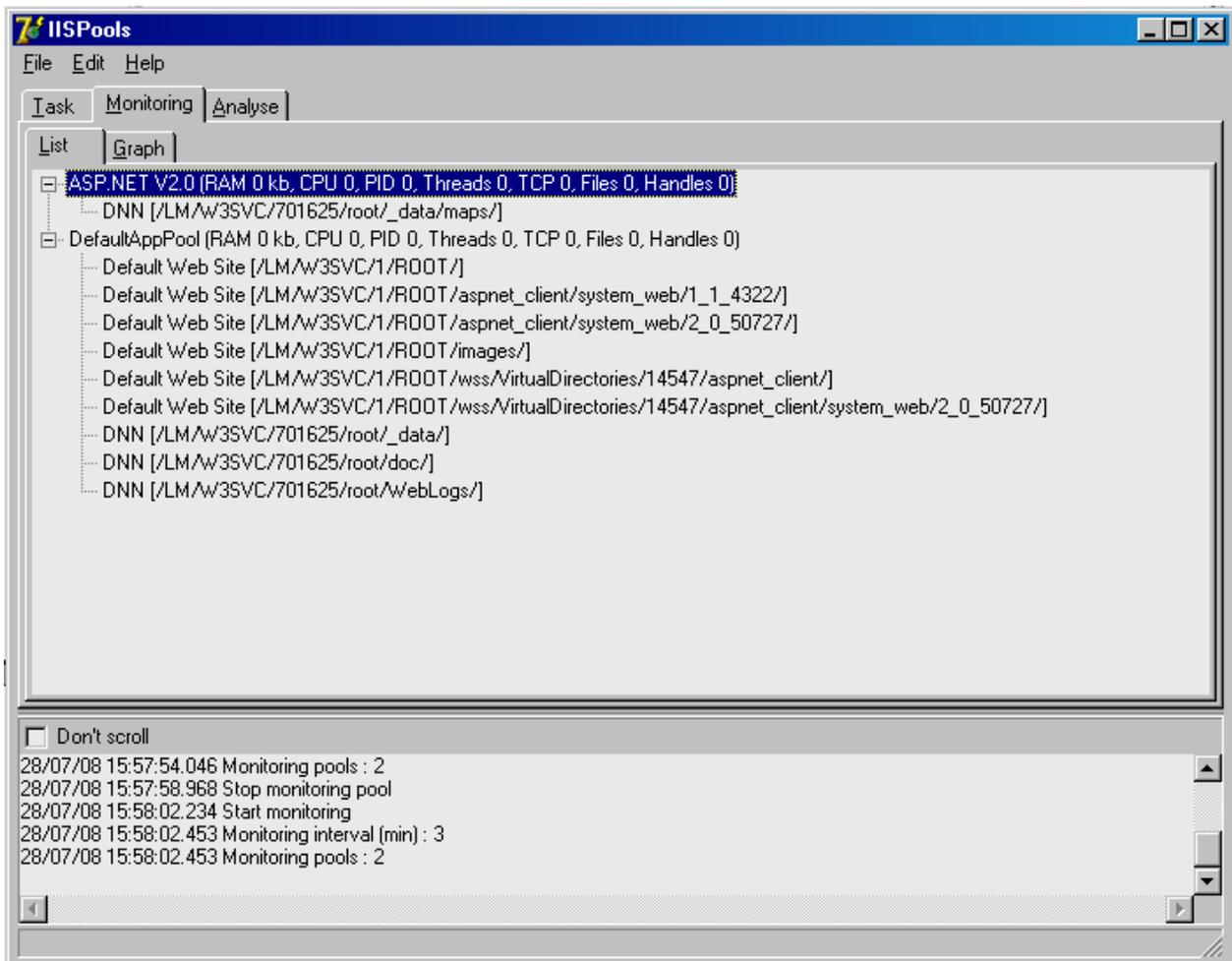
Description of the Pool Analysis interface

Pool analysis is active only if the monitoring is active too.

- First you should select a pool to be analyzed. It is possible to choose one pool at a time only.
- Specify some value in "Alert if …" field and the program will make a record every time the CPU usage (%) by the pool is greater than specified.
- Press [Start] button to start the analysis.
- 20 minutes is needed for the analysis at least. But you may decide to run it for longer time (even for a day or two) to get more precise results. (As you understand, the more requests are being served the more CPU time is used. And the load is less at night usually.)
- **Important:** You have to stop the analysis by pressing [Stop] button on "Analyse" tab at the end of the analysis. Remember that [Stop] button on the "Task" tab does not stop the analysis!

The analysis log and results will be shown on "Monitoring" tab and saved in iisp-ans-XXXXXX.log file in LOG folder. (XXXXXX will be replaced with numbers)