

# ReSharper Profiling Instructions

- Performance Issues
  - ReSharper embedded profiler
  - Standalone dotTrace profiler
- Memory Issues
  - ReSharper embedded memory profiler
  - Standalone dotMemory profiler
  - .Net Memory Profiler

## Performance Issues

### ReSharper embedded profiler

In order to get a performance snapshot for ReSharper using embedded profiler follow these steps (full manual is [here](#)):

1. Go to 'ReSharper | Help | Profile Visual Studio'.
2. Note: on a 1st run self-profiling tool will be downloaded from JetBrains server, so please be patient.
3. This action will start the profiling of the VS immediately - after you finish doing the repro actions (basically, reproduce the slowdown), please press the 'Get Snapshot' button in the dialog.
4. After profiling is complete, you'll be able to upload the snapshot to our FTP server by clicking on Submit button

### Standalone dotTrace profiler

In order to get a performance snapshot for ReSharper using dotTrace follow these steps:

1. Download and install dotTrace Performance profiler from <https://www.jetbrains.com/> | Select 64-bit version
2. Start dotTrace and select Profile Local App
3. In the Standalone Application section, specify a path to the Visual Studio executable (devenv.exe)
4. Make sure the following options are set in Profiler Options:
  - a. Profiling type: Sampling (or Tracing or Timeline if you has been asked to collect exactly Tracing/Timeline snapshot)
  - b. Collect profiling data from start: on (if you're experiencing performance problems on Visual Studio startup or during solution loading) or off otherwise
5. Click Run
6. If there's a performance problem during startup
  - a. Wait while Visual Studio starts, open your solution
  - b. Wait until the solution is loaded and ReSharper finishes loading caches
  - c. Click Get Snapshot and Wait button in the profiling controller window
7. Otherwise
  - a. Wait while Visual Studio starts, open your solution
  - b. Click Start button in the profiling controller window
  - c. Perform actions which are slow with ReSharper (typing, switching between documents, building etc)
  - d. Click Get Snapshot and Wait button in the profiling controller window
8. In dotTrace Performance Viewer, save the resulting snapshot using the File | Export Snapshot... menu
9. Compress the snapshot file using Zip
10. Upload the file to <ftp://ftp.intelij.net/.uploads/>. Don't worry if you don't see the upload progress - it is hidden according to the FTP folder privileges.
11. [Create a new issue](#) in ReSharper issue tracker. In the issue, provide a short description of the performance problems you're experiencing, and specify the name of your snapshot.

## Memory Issues

### ReSharper embedded memory profiler

In order to get a memory snapshot for ReSharper using embedded profiler follow these steps (full manual is [here](#)):

1. Go to 'ReSharper | Help | Report bug or submit feedback'.
2. Choose 'Problem Report | Problem category: Performance problem'.
3. Hit "green cross" icon next to "Attachments" text | select "Memory Snapshot" | It will start self-profiling tool.
4. Note: on a 1st run self-profiling tool will be downloaded from JetBrains server, download times may vary so please be patient.
5. After profiling is complete, you'll be able to upload the snapshot to our FTP server by clicking on Submit button

## Standalone dotMemory profiler

In case of memory issues with ReSharper, please follow the instructions below.

1. Download and install the trial version of [dotMemory](#).
2. Run dotMemory.
3. On the dotMemory Home page, select Local | Standalone.
4. In the right panel, in Standalone Application:
  - a. In Application, specify the path to the Visual Studio executable. For example: C:\Program Files (x86)\Microsoft Visual Studio 12.0\Common7\IDE\devenv.exe
  - b. Click Run. This will run Visual Studio.
5. Open your solution and wait until it is completely loaded.
6. Collect a memory snapshot by clicking Get Snapshot in dotMemory.
7. In Visual Studio, perform actions that are suspected to cause memory issues (high memory traffic, exceptions, etc.).
8. Collect a snapshot one more time by clicking Get Snapshot in dotMemory.
9. Close Visual Studio.
10. In dotMemory, save collected snapshots to a file using the menu File | Export Workspace.
11. Upload the file to <ftp://ftp.intelij.net/uploads/>.  
Don't worry if you don't see the upload progress – it is hidden according to the FTP folder privileges.
12. [Create a new issue](#) in our issue tracker.  
In the issue, please provide:
  - a. A short description of the issue.
  - b. Steps required to reproduce the issue.
  - c. The name of the uploaded file.

## .Net Memory Profiler

Prepare environment:

1. Before starting Visual Studio:
  - Open devenv.exe.config in "C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE"
  - In the <runtime> section add <gcConcurrent enabled="false"/>
2. Download [NmpCore.exe](#) (or you can try newer version from the [vendor site](#))
3. Enable Managed Memory Indicator in ReSharper | Options / Environment / General

Get snapshot:

1. Get process ID for devenv.exe process in question
2. Double-click managed memory indicator in Visual Studio status bar (to the right) to perform Garbage Collection
3. Execute the following command, substituting <pid> and path to snapshot:

```
NmpCore.exe /a <pid> /cs1 /sf "full-path-to-snapshot-file.prfsession"
```

Advanced snapshots:

1. Download evaluation version of [.Net Memory Profiler](#)
2. Install and start it
3. Open Tools / Options, select "Call Stacks" page and uncheck "Enable stack reducer" on General tab
4. Click "Profile application" on the main screen
5. Enter path to Visual Studio executable "C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE\devenv.exe" and click Next
6. Select "Custom" profiling level, and click Next
7. Select "Full" on instance tracking page, and click Next
8. Select "Yes" for the collect allocation call stacks choice, and click Next
9. Select "No" for dispose tracker, real-time data collection, heap utilization, AppDomain tracker, unmanaged resources tracker and instance data trackers.
10. Click Start to start process and begin profiling
11. Open your solution and after everything settles down, click Collect Snapshot in profiler to create base-line snapshot
12. Perform actions that are producing memory traffic, or cause excess memory allocations
13. Collect snapshot
14. Hit "Stop profiling" button and save snapshots in a .prfsession file

Upload snapshot and create an issue:

1. Compress the resulting snapshot using Zip
2. Upload it to <ftp://ftp.intelij.net/uploads/>. If upon uploading you're unable to see the snapshot in the remote folder, this is normal: this is the way privileges are set up for the FTP folder.
3. [Create a new issue](#) in ReSharper issue tracker. In the issue, provide a short description of the memory consumption

problems you're experiencing, and specify the name of your snapshot.