

Interpreting unexpected results

When profiling, you may see some results that do not look accurate. This article explains the most common causes of unexpected profiling results.

Negative timings

You may find that some very simple methods are reported as taking a negative amount of time, especially if you have chosen to display line-level timings.

This is caused because an estimate of the profiler's overhead is subtracted from results by default, and, for a very simple method, too much may be subtracted.

To disable this option, on the **Tools** menu, click **Advanced Options**, and clear the **Adjust timings to compensate for overhead added by the profiler** checkbox. CHECK THIS

In the call graph, timings do not add up to 100% of the parent node

If you are profiling in *Sample method-level timings* mode, sampling adds the least overhead but is also the least accurate profiling mode. As a result, timings may not add up to 100% in sampling mode.

If you are not profiling in *Sample method-level timings* mode, the percentages shown in the call graph are as a percentage of the selected node (colored black), which may not be the immediate parent of the nodes you are looking at.

In sampling mode, a method seems unexpectedly slow

This is an artifact of how *Sample method-level timings* mode works. In order to assess timings, ANTS Performance Profiler checks for each method twice. If a child method is significantly faster than its parent method, the probability that ANTS Performance Profiler will detect the child method twice is low, and so the time taken is attributed to the parent method.

Data shown in the call tree is not the same as the source code view

This generally happens because the call tree displays a specific stack trace, so a method's hit counts and timings are only for the stack trace being displayed. The source code view shows the hit counts and timings for every time the method is called, in any stack trace.

The other reason this can happen, especially if you are profiling line-level timings, is that an estimate of the profiler's overhead is subtracted from results by default, and too much may be subtracted in one of the views.

To stop the profiler from subtracting its own overhead estimate, on the **Tools** menu, click **Advanced Options**, and clear the **Adjust timings to compensate for overhead added by the profiler** checkbox. CHECK THIS

Timings for SQL results seem too long

The initial call to a database may be substantially slower than subsequent calls. The timings shown include the time taken to generate execution plans and to index the database, if required. This is the most common reason for SQL results appearing to take too long.

Hit counts seem too high for some methods

If the number of calls to a method seems too high, remember to consider any delegates or lambda expressions in the code, which may call the method.

CPU usage exceeds 100%

ANTS Performance Profiler analyzes CPU usage per core. So, on a multi-core machine, CPU usage may appear to exceed 100%. For example, an application running at 100% CPU usage on all four cores on a quad-core machine would be shown as taking 400% of CPU time.

Call stacks appear to overlap

If your methods do not take long to complete, the time taken by methods in separate call stacks (that should execute serially) may appear to overlap in the timeline, even with overhead correction disabled. This is because the resolution on the start and end of the method in the timeline is only 250ms.