

Features	Developer Edition	Developer Edition <i>Enterprise</i>
<b>Tracing Engine</b>		
Seamless integration into Microsoft Visual Studio	Visual Studio 2005, 2008, 2010	Visual Studio 2005, 2008, 2010
Minimal performance overhead. Analyze even multithreaded applications		
"Single-click" tracing without rebuild		
Control flow tracing. Reveals the nested structure of function calls	up to 30.000 function calls	up to 1.000.000 function calls
Recording program states with DataProbes, i.e., selected variables at code locations	1 DataProbe	unlimited DataProbes
Recording program states with ParamProbes, i.e., parameter values being passed into functions		
Recording program states with MemAccessProbes, i.e., record when specific memory is read or written to quickly find difficult bugs due to memory overwriting side-effects		
Start the application with the tracing engine attached		
Trace from within a Visual Studio debugging session		
Attaching the tracing engine to a running process		
<b>MemoryLeak Tracing Engine</b>		
Take snapshots of a process' allocated memory at any time during the process' runtime		
Compare memory snapshots to identify memory leaks		
Reveal where in the code memory is allocated but not deallocated		
See the complete call stack to memory allocating code		
Navigate to the memory leak in the code with a single click		
<b>Analysis Engine</b>		
<b>Code View</b>		
In Microsoft Visual Studio's source code view, markers are placed that indicate in which call code lines were executed		
Additionally, the markers reveal the real function that was called. This valuable information solves the problem arising from polymorphism and function pointers		
<b>Timeline View</b>		
Overview on application behavior during complete tracing period		
Permits you to identify execution phases and patterns		
Selecting points in time or time periods for more detailed analysis in the Call Stack view		

Developer Edition

Developer Edition  
Enterprise

**Call Stack View**

Temporal representation of tracing data		
Visual depiction of the call stack over time		
You have the call stack for a complete tracing period		
You can travel back and forth in time		
Function calls contain information on parameter values if Probes were active during tracing		

**Single Call View**

Representation of all subcalls that are triggered for a selected call		
View is an intermediate view between graphical views and source code views. It essentially reduces the source code to code lines with executed call statements.		
Used to quickly grasp what happens in the respective function during execution and to find a subcall of interest.		

**Call Graph View**

Structural representation of trace data		
Caller and callee relation between selected functions		
Interactively unfolding and hiding subcalls		
Annotation of call relations with performance statistics: "call count" and "call costs".		
Color encodes performance statistics		

**Search View**

Complete trace data is searchable		
Quickly identify and navigate to functions of interest		
Quickly find variable values of interest captured in Probes		

<b>Supported Platforms:</b>	Windows XP, Windows Vista, Windows 7, Microsoft Visual Studio 2005, 2008, 2010
<b>Minimal System Requirements:</b>	1 GHz CPU, 1 GB of RAM, 150MB hard disc capacity, Graphic card with OpenGL 1.4 support

Subjects for change without notice.  
16 February 2011

Software Diagnostics, Software Diagnostics Developer Edition, Software Diagnostics Developer Edition Enterprise and Software Diagnostics Studio are trademarks or registered trademarks of Software Diagnostics GmbH, Germany. All other company and product names are the property of their respective owners.

For further information:

Verifysoft Technology GmbH, In der Spöck 10-12, 77656 Offenburg, Germany,  
Phone +49 781 6392-027, [www.verifysoft.com](http://www.verifysoft.com)