int₌l ®

Intel® Software Development Products for Intel Platforms and Technologies
Intel® Trace Collector[1]

Overview

Intel® Cluster Tools assist developers and managers of distributed systems in getting the best application performance.

Intel® Trace Collector brings the advantages of event-based tracing to applications in a low-overhead tracing library. It supports MPI, Java*, and multi-threaded processes with and without MPI. The tool is completely thread safe, allowing tracing of multi-threaded MPI applications. Automatic function profiling is also now supported on all platforms when the GNU Compiler Collection is used to compile C or Fortran source code. Intel® Trace Analyzer[1] and Intel Trace Collector support IA-32 and Intel® Itanium® architectures with Linux* (MPICH). Intel Trace Collector supports the LAM MPI implementation on IA-32 architectures and HP-UX* (HP-MPI) and SGI Altix* (SGI-MPI) systems on Itanium 2 microarchitectures. Intel Trace Analyzer displays the data produced by Intel Trace Collector.

Graphically Analyze Runtime Event Traces

- Event-based tracing tool — Successful use of Intel Trace Collector for MPI performance analysis demonstrates the advantages of event-based tracing compared to exclusively statistical approaches. The timeline display helps developers visualize the concurrent behavior of parallel applications. Statistics can be calculated on demand for given time intervals and specific processes. The tool also brings the advantages of event-based tracing to non-MPI applications such as Java processes and multi-threaded processes without MPI. Java function profiling is possible without recompiling the Java classes. In addition, the traditional Virtual Terminal Application Programming Interface (VT API) is also available to instrument Java source code.
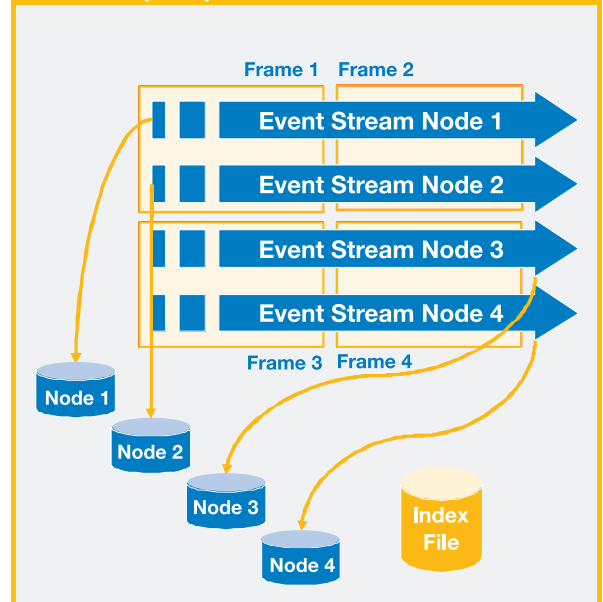
Features and Benefits

- **Profiling library** records distributed, event-based trace data
- **Low-overhead** and compact data representation provides structured trace file format (STF)
- **Completely thread safe** to allow tracing of multi-threaded Message Passing Interface (MPI) applications
- **Easy-to-use** Application Program Interface (API)
- **Low intrusion instrumentation** for MPI, Java*, or multi-threaded applications
- **Complements Intel's offerings** designed for developing high-performance software including compilers, libraries, and Intel® VTune™ Performance Analyzers
- **Automatic function profiling** supported on all platforms when the GNU Compiler Collection is used to compile C or Fortran source code
- **Global Array programming model** allows tracing and performance analysis of applications

Graphical View of Structured Trace Format (STF) Data Collection

- Ease of use — Intel Trace Collector works as an add-on to existing MPI implementations. In most cases, using the tool requires relinking with the Intel Trace Collector library. If application-defined events must be recorded, a recompile may be necessary. The low-overhead instrumentation limits the perturbation of the application execution, assuring that the runtime behavior remains fundamentally unaltered. Experience shows that lessons learned from studying the traces of a run with Intel Trace Collector invariably lead to improved performance of non-instrumented runs.

Intel®
software development
products

- Structured trace file format — Intel Trace Collector introduces a new file format for trace data: the *Structured Trace Format* (STF). This file format was designed from the ground up for scalability and compact data representation. Files can be written in parallel, thus generating trace files faster, and allows random access to portions of a trace, making it suitable for analysis of traces too large for Intel Trace Analyzer.

- Scalability and memory handling — Reduction of large memory consumption, often associated with event-based tracing, is addressed by the filtering and memory-handling capabilities of Intel Trace Collector. For deep function call stacks, folding avoids superfluous details by logging the first call to a system function and then hiding all internal functions below this call. Trace data is cached in memory to reduce runtime overhead, but can be written to a cache file in the background without blocking the application. Counters can be used to monitor Intel Trace Collector's memory handling. If that still interferes with the application or simply produces unneeded data, Intel Trace Collector can be used in a purely statistical mode. In this mode, statistics regarding function calls, messages, and collective operations are calculated by Intel Trace Collector at runtime and stored without event data. Intel Trace Analyzer can display these statistics instantaneously without having to load any trace data.

- Completely thread safe — Intel Trace Collector allows tracing of multi-threaded MPI applications. The tool brings the advantages of event-based tracing to non-MPI applications by supporting Java processes and multi-threaded processes without MPI. Java function

profiling is possible without recompiling the Java classes. In addition, the traditional VT API is available to instrument Java source code. Automatic function profiling is supported on all platforms when the GNU Compiler Collection is used to compile C or Fortran source code.

PERFORMANCE
Processor and Data Scalability
Intel Trace Collector easily scales up to several hundred processors due to a hierarchical approach to address data scalability.

COMPATIBILITY
Diverse Hardware and Software Support
Intel Trace Collector supports IA-32 and Itanium architectures with Linux. MPI implementations such as MPICH, LAM MPI, ScaMPI, and Myrinet* MPI are supported on IA-32 architectures. On Itanium architectures, MPICH, HP-UX (HP-MPI), and SGI Altix (SGI-MPI) systems are supported. Automatic function profiling is supported on all platforms when the GNU Compiler Collection is used to compile C or Fortran source code.

SUPPORT
Intel® Premier Support
Every purchase of an Intel® Software Development Product includes a year of support services, which provides access to Intel® Premier Support and all product updates during that time. Intel Premier Support gives you online access to technical notes, application notes, and documentation. Install the product, and then register to get support and product update information.

Hardware Requirements
- Intel® Pentium® 4, Intel® Xeon™, or newer processor
- 512 MB RAM
- 1 GB disk space
   or
- Intel® Itanium® 2 processor
- 1 GB RAM
- 1 GB disk space

Software Requirements
Supports IA-32 and Itanium architecture-based systems with Linux*. MPI implementations such as MPICH, LAM MPI, ScaMPI, and Myrinet* MPI are supported on IA-32 architectures. On Itanium architecture-based systems, MPICH, HP-UX*, and SGI Altix* systems are supported.

Note: Intel Trace Collector Parallel Performance Analysis Library is a machine (cluster) bound license. This means for every machine (cluster), you need a separate license. Licensing of Intel Trace Collector[1] Parallel Performance Analysis Libraries restricts the maximum number of MPI processes used in a parallel program to no more than the total number of processors used, regardless of the number of MPI processes executed per processor.

[1] Intel® Trace Analyzer, formerly marketed as Vampir. Intel® Trace Collector, formerly marketed as Vampirtrace.

Intel provides both the tools and support to enhance the performance, functionality and efficiency of software applications.

Compatible with leading Windows* and Linux* development environments, Intel® Software Development Products are the fastest and easiest way to take advantage of the latest features of Intel processors. Intel Software Development Products are designed for use in the full development cycle, and include Intel® Performance Libraries, Intel Compilers (C++, Fortran for Windows and Linux), Intel® VTune™ analyzers, Intel® Threading Tools and Intel® Cluster Tools.

The Intel® Premier Customer Support Web site provides expert technical support for all Intel software products, product updates and related downloads. **For additional product information visit: www.intel.com/software/products**