

TimeTrac

Event Recording and Analyzer Tool

Features

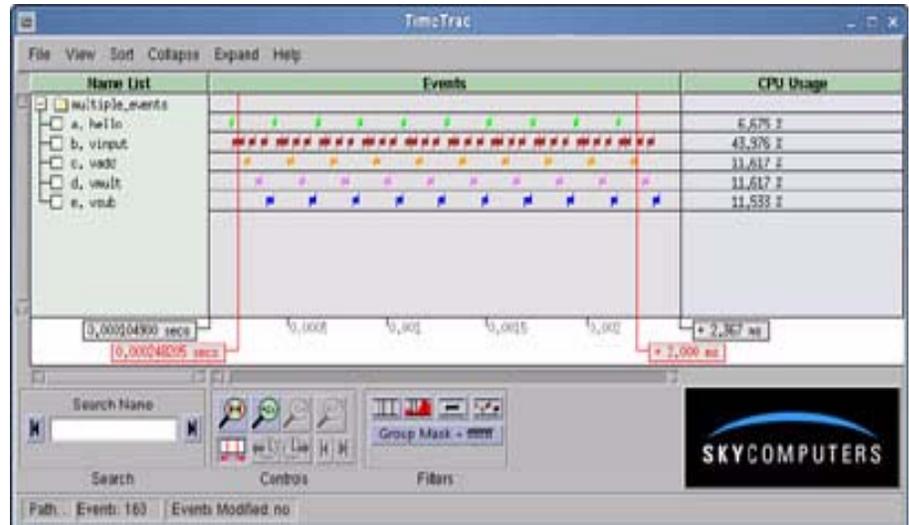
- Evaluates multi-thread applications running on multi core or multi-processor systems
- Non-intrusive - minimal data is logged directly to memory and is saved to a disk file when the application is at a non-critical point
- Easy to instrument - requires only a small amount of application code that can be easily compiled out of the final application.
- Intuitive viewing controls - independent of your current integrated development environment; no need to learn another IDE

Application Instrumentation

- Easy-to-use, minimally intrusive code linked with a SKY-provided library
- Record single events or range events (two single events that provide a start and stop timestamp)
- Define contexts to organize large lists of recorded events for easier viewing and analysis

TimeTrac Viewer

- Full-featured, easy-to-use graphical interface
- Customizable start up options that are saved from session to session
- Results viewing and analysis are done off-line at the developer's convenience
- Histograms display minimum, maximum, average compute times, and clustering of compute times



TimeTrac is an event recording and analysis tool originally developed by SKY Computers for real-time, high performance multiprocessor systems. It is used to debug, measure, and optimize application software throughout the development cycle, from initial algorithm development through algorithm modifications to system deployment.

Effective debugging of software applications developed for multi core or multiple processor systems can be very challenging. Using a debugger to step through the source code or inserting printf statements may resolve some of the simpler program issues. However, using these methods can affect the proper execution of a multi-threaded application or cause it to behave in a different manner than expected. Also, these methods will provide little information about the performance of the processors or how they are communicating.

Using TimeTrac, a software developer can quickly uncover the complex program issues that prevent project schedules from being met. TimeTrac allows the developer to instrument the application in order to easily collect performance data and the easy-to-use TimeTrac Viewer displays the collected data for analysis. Using this data, the developer can view computation times and how they are clustered. By analyzing event relationships and eliminating race conditions, correct program operation can be verified. Also, by using TimeTrac in overwrite mode, problems that occur infrequently can be isolated in a fraction of the time that would normally be required to detect these hard-to-diagnose problems.

With its easy-to-use viewer and minimal intrusion into the application code, TimeTrac is a superior tool for evaluating multi-threaded applications running on high performance, multi-core and multi-processor systems.

Performance Improvement

- **Optimize performance by determining where the application is spending its time**
- **Identify speed improvement opportunities in algorithms**
- **Isolate infrequently occurring problems**
- **Determine which processors are overloaded or underutilized**
- **Display causal relationships**
- **Synchronize communications and semaphores**
- **Observe operating system context switches**
- **Identify memory contention**

Application Instrumentation

TimeTrac includes an application programming interface (API) that can be used to record when events occur and to accurately display the execution time for a section of an algorithm. The API includes functions for opening an event recording session; stopping, starting, and recording events; and grouping events into common sets.

With TimeTrac, a developer can record an event with a single time stamp indicating when the event occurred. Or, a range event can be programmed so that two single events provide a stop and start time stamp. The recorded data includes when the range event occurred, its duration, when the event completed, as well as a user-specified value associated with the recorded event (such as variable value or buffer size). In addition, events can be grouped into a common set called a context. Contexts are very useful in organizing a large list of recorded events so that they are more easily viewed for analysis.

Events are written in a format that is independent of the architecture on which the events are recorded. In this way, an algorithm that is split between a single host and multiple processors can be analyzed to determine how the processors are performing and/or synchronizing with each other.

The API functions add a small amount of code to the application and can easily be compiled out of the final application. TimeTrac records a minimal amount of data directly to memory and then saves the data to a disk file at a non-critical point in the application.

TimeTrac Viewer

The TimeTrac viewer is a graphical interface that displays the data recorded using the API functions. It is designed to provide a visual presentation of the operation of the application. By analyzing the recorded events contained in previously created trace files, the developer can identify opportunities for real time performance improvements. The viewer can display a large number of recorded events, making it easy to analyze complex applications.

The viewer can be customized according to the preferences of the user and the preferences can be saved for subsequent analysis sessions. Multiple trace files can be displayed and their content can be manipulated to display events in a number of ways. The viewer's zoom capabilities can focus on specific events in a trace file.

TimeTrac is a trademark of SKY Computers, Inc.
Doc #350-081 Revision 2 01/98

Specifications subject to change



27 Industrial Ave. Chelmsford, MA 01824
Phone: (978) 250-2420
FAX: (978) 250-3220
<http://www.skycomputers.com>