White Paper

Real-time Capabilities for Linux
SGI REACT™ Real-Time for Linux®

Abstract

This white paper describes the real-time capabilities provided by SGI REACT™ Real-Time for Linux. software. REACT enables SGI Linux-based system platforms to deliver hard real-time (HRT) capabilities including an interrupt response time of 30us or less. A brief review of current real-time Linux capabilities, a description of typical real-time applications, and a comparison of SGI real-time capabilities for Linux versus IRIX are also included.

Table of Contents

1.0 Introduction

The ability to perform computations in real time is critical to a variety of applications. In casual use, people often refer to any computing system capable of returning results quickly as "real-time", but such systems do not guarantee the latency of each computation. True real-time processing requires that computations return the correct result within a set period of time.

An important distinction in real-time processing is the differentiation between hard real-time (HRT) and soft real-time (SRT). In hard real-time systems, the computation is successful only when the answer is accurate and it is delivered within the required time constraint. This time constraint is often referred to as the "guaranteed time." In soft real-time systems, the computation is successful if the answer is accurate and it is delivered within a threshold of time. HRT systems require specific capabilities within the operating system that have guaranteed latencies and determinism; making HRT much more challenging to achieve than SRT.

The SGI approach to real-time delivers desirable latency properties in a standard Linux user process without requiring a special kernel or the compiling of user code inside the kernel. SGI has been producing systems capable of HRT for over ten years. SGI REACT™ Real-Time for Linux® is the SGI real-time extension for SGI Linux platforms. With REACT, SGI platforms deliver hard real-time interrupt response times—the time that passes between the instant a hardware device raises an interrupt signal and the instant the system returns control to a user process. REACT for Linux provides significant new capabilities that allow SGI Linux-based systems to sustain interrupt response times of 30µs or less. Such guarantees are important to real-time programs because they put an upper bound on the overhead of servicing interrupts and thus make computations more predictable. REACT is supported on systems ranging from 2 to 64 processors in size. Competing HRT systems are typically limited to no more than 8 processors. In addition, REACT for Linux running on SGI platforms provides:

- Large memory support
- Bandwidth and latency advantages of NUMAlink™ technology
- Independent scaling of CPU, memory and I/O
- Hardware-dependent external interrupt capability
- User-level interrupt handling
- Frame Rate Scheduler
- High resolution timers/clock
- Kernel barrier feature
- sgi-shield
- SGI Linux Trace
- Dynamic reconfiguration capability

This paper provides an overview of real-time capabilities on Linux, looks at typical real-time applications, examines the unique features of SGI REACT Real-Time for Linux, and provides a comparison between the real-time capabilities of SGI REACT for Linux versus IRIX.

2.0 Real-Time Systems and Linux

As is often the case, initial real-time system development began with proprietary operating systems. These were often extensions of other proprietary operating systems used for more generalized applications. This was the case with the real-time extensions that SGI created for its SGI IRIX operating system. More recently, the open source advantages as well as the "lighter" initial weight of Linux has favored its adoption and use in real-time systems. The widespread use of Linux for Internet-based applications and the need for embedded systems in Internet and telecommunications hardware has provided additional impetus.

SGI has leveraged its ten plus years of experience with real-time development on the SGI IRIX operating system in the development of SGI REACT for Linux. REACT for Linux combines the base capabilities of the standard Linux kernel with the unique HRT capabilities necessary to go beyond the capabilities of previous Linux-based real-time systems to create a complete solution capable of tackling the most difficult real-time applications.

3.0 Real-Time Applications

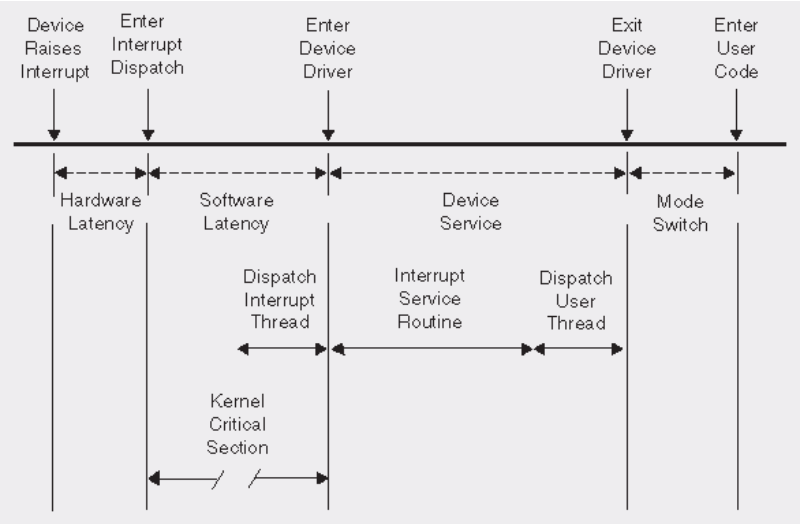
Applications that require real-time computation are many and varied. SGI has significant experience with real-time applications including:

- Man-in-the-loop simulators. Closed-loop simulators used for operator training (aircraft, ship, tank), engineering simulators, and power plant simulators.
- Hardware-in-the-loop simulators. Closed-loop simulators with high frame rates (thousands of Hz) often requiring high-speed, low-level I/O.
- Telemetry, radar, and data acquisition. Open-loop systems for acquiring, processing and storing (or re-transmitting) data in real-time.
- C4I applications (Command, Control, Communications, and Intelligence). Systems that combine a lot of the previous application needs and often include displays that need to be updated for the latest information.
- Video applications. Systems that handle live video such as virtual sets, real-time broadcast, and video-on-demand (VOD)

4.0 REACT for LINUX Features

The SGI Linux-based platform runs on standard Linux distributions including SUSE® Linux Enterprise. The REACT software consists of modules that run on top of the standard SUSE Linux kernel (but do not replace the kernel), providing a variety of additional features that enhance real-time execution.

Figure 1. Components of interrupt response.



4.1 Altix Real-Time Foundation

The key characteristic of REACT for Linux is its ability to guarantee an interrupt response time of 30µs on systems ranging from 2-64 processors. The total interrupt response time includes:

- Hardware latency. The time required for a CPU respond to an interrupt signal.
- Software latency. The time required to dispatch an interrupt thread.
- Device service time. The time the device driver spends processing the interrupt and dispatching a user thread.
- Mode switch. The time it takes to exit kernel mode and resume execution of the user thread.

REACT ensures that the combined time for all these activities will not exceed 30µs on a properly configured system.

A variety of factors contribute to this response time including:

- At system startup, REACT performs a number of configuration operations such that each CPU designated as a real-time CPU provides minimal thread latency. This includes appropriate redirection of all I/O interrupts to non real-time CPUs.
- REACT for Linux includes features that provide programmatic and command-line access to features that allow process pinning, dynamic real-time CPU configuration, scheduler prioritization, memory locking, and dynamic I/O interrupt redirection.

- I/O interrupts to be serviced by real-time threads can be redirected to real-time CPUs. In general, real-time CPUs should only service interrupts from devices critical to the execution of the associated application(s). The interrupt service routine (ISR) that handles the interrupt should do very little processing and should avoid time-consuming operations such as allocating buffers.
- By default, CPUs are not isolated from system timer interrupts since the system timer typically produces no major latency issues. (CPU 0 always performs system-wide clock processing in ia64 Linux.) If desired, sgi-shield can be used to isolate CPUs from system timer interrupts. This eliminates jitter and gives the user application complete control of the CPU such that a CPU can be dedicated to run only the desired user code.

A number of additional REACT features ensure correct and predictable real-time operation and facilitate the creation and execution of real-time applications.

4.2 Hardware-specific Features

REACT for Linux includes the ability to take advantage of a number of capabilities unique to SGI hardware.
High Resolution Clocks and Timers
SGI Altix provides a system-wide clock called a real-time clock (RTC) that is accessible to every CPU. The RTC provides a raw time source that is incremented in 50-ns intervals. High resolution POSIX timers on Altix utilize the RTC as a time source.

The RTC is 55 bits wide, which ensures that it will return a unique value for more than half a century. RTC values are mapped into the local memory of each processing node. Multiple nodes accessing the RTC value do not reduce the performance of the clock functions. The RTC is synchronized among all of the nodes in an SGI system using a special pin on the NUMAlink cable providing superior scalability. This feature is not available on Altix XE.

External Interrupts

Real-time processes often require the ability to respond to external events. The external interrupt capability of SGI Linux-based systems allows a real-time process to receive an external signal. An external interrupt is generated when an electrical signal is applied to the external interrupt socket on the SGI PCI-RT-Z card. This card has an external jack into which a 0-5V signal can be fed. External equipment can assert this line,

causing the card to generate an interrupt. The I/O card also has the ability to generate interrupts for signaling external equipment using the same external jack. This capability enables SGI Linux systems to coordinate activities with external devices or other, independent, SGI systems.

VME Support

Many real-time system designs require the ability to interface to external VME equipment. VME solutions are available from third party vendors through SGI Professional Services.

4.3 Unique REACT Software Capabilities

SGI systems running REACT are able to utilize all of the real-time capabilities of the standard Linux distribution. SGI continues to develop and contribute to the real-time capabilities of Linux. In addition, REACT provides some unique capabilities.

User Level Interrupts

The user-level interrupt (ULI) facility allows a hardware interrupt to be handled by a user process. A user process may register a function with the kernel, linked into the process in the normal fashion, to be called when a particular interrupt is received. The interrupt handler is called asynchronously and has access only to the process's address space. A significant advantage of the ULI facility is that it can be very fast. Applications that need to handle specific interrupts in less than the 30 microsecond maximum guaranteed by REACT utilize this feature. The best case is a response in 3-4µs. The worst case is no slower than the system's worst case for scheduling a process.

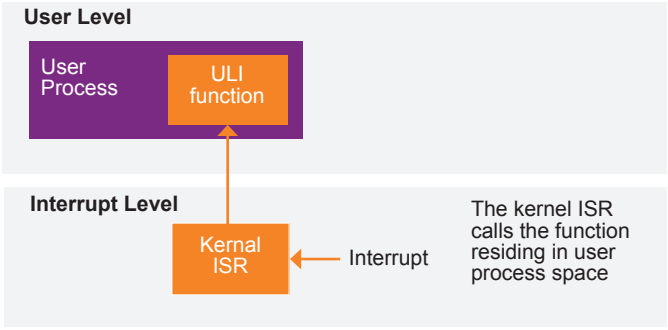


Figure 2. User-level interrupt (ULI) handling.

The ULI facility is also intended to simplify the creation of device drivers for unsupported devices. ULIs can be written to respond to interrupts initiated from external interrupt ports. A programming error in the driver will result in nothing more serious than the termination of a process rather than crashing the entire system, and the developer need not know anything about interfacing a driver into the kernel.

The ULI feature may also be used for high-performance I/O applications when combined with memory-mapped device I/O. Applications can make all device accesses in user space. This is useful for high-performance I/O applications such as hardware-in-the-loop simulators.

Frame Rate Scheduler

The Frame Rate Scheduler makes it easy to structure a real-time program as a family of independent, cooperating activities that are running on multiple CPUs and are scheduled in sequence at the frame rate of the application. The frame rate scheduler can synchronize the activities of many CPUs, while providing flexible notification and schedule changes during over and under runs.

Instead of scheduling threads according to priorities, the frame scheduler dispatches them according to a strict, cyclic rotation governed by a repetitive time base. The time base determines the fundamental frame rate. Some examples of the time base include:

- A specific clocked interval in microseconds
- An external interrupt
- A device interrupt from a specially modified device driver

The interrupts from the time base define minor frames. Together, a fixed number of minor frames make up a major frame. The length of a major frame defines the application's true frame rate. The minor frames allow you to divide a major frame into subframes.

The frame scheduler maintains a queue of threads for each minor frame. Each activity thread of the program is queued to a specific minor frame. The order of cyclic execution within a minor frame is determined by the order in which threads are queued.

Kernel Barrier Facility

The kbar kernel barrier facility provides for the fast wake-up of many blocked user threads. When the barrier is signaled, the operating system will use a configurable number of CPUs to quickly wake all blocked threads. A maximum of 64 barriers are supported system-wide. Kbar barriers do not behave precisely the same as traditional barriers. A specific number of threads do not have to be blocked on the barrier for them to be woken.

SGI-shield

The sgi-shield feature allows CPUs to be isolated from system timer interrupts. This eliminates jitter (short interruptions in process execution) which gives a user application complete control of the CPU. Shielding can be turned on dynamically from within the application and should only be used for short periods where jitter-free program execution is required.

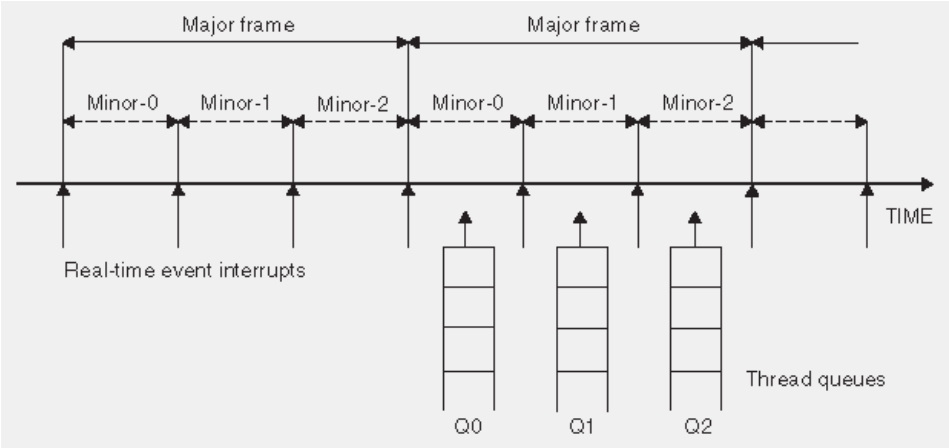


Figure 3. Relationship between major and minor frames and thread queues.

SGI Real-time Library

REACT for Linux includes a userspace library called libsgirt that contains IRIX compatibility routines and other user space routines unique to REACT. This library provides a set of interfaces similar to those found in previous-generation SGI IRIX real-time offerings to ease the migration path for customers moving real-time applications from IRIX to Linux. Using the library can also insulate users from some future changes in the real-time operability and configuration of Altix systems.

Configuration Tools

REACT provides a script that is used to configure systems for real-time processing. This script performs CPU isolation and allows dynamic reconfiguration of real-time CPU assigments.

SGI LINUX Trace (SLT)

SLT is a debug kernel that can generate traces of kernel events such as interrupt handling, scheduling and system calls. Recorded trace events are viewable with the graphical tracevisualizer tool, allowing users to analyze how kernel behavior impacts the execution of real-time applications.

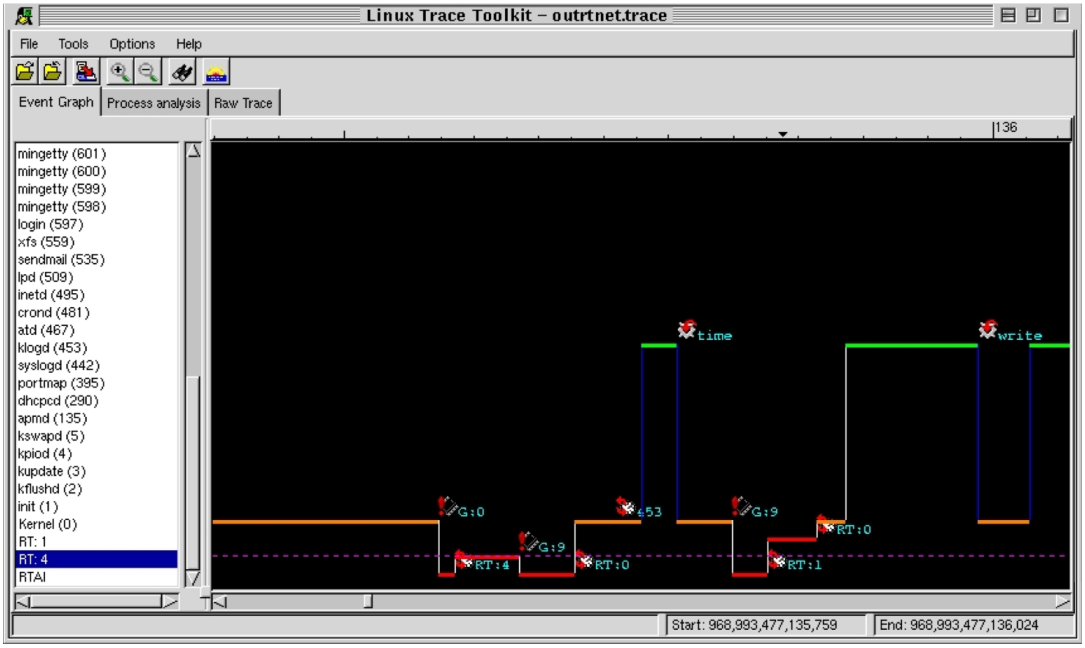
5.0 Migrating Real-Time Applications from IRIX to Linux

SGI has made every effort to provide comparable capabilities between its real-time extensions for IRIX and Linux to simplify migration. Both real-time implementations provide such features as:

- User level interrupts
- Frame rate scheduler (API is nearly identical)
- Debug tools
- Dynamic reconfiguration
- External interrupts
- Programmatic and command line access to features that allow memory locking, process pinning, and CPU restriction and isolation

In addition, REACT for Linux adds unique capabilities including sgi-shield and kbar that were not available on IRIX. The Linux-based platform delivers better real-time performance than SGI Origin running IRIX with realtime extensions: 30µs guaranteed interrupt response time versus 50µs for Origin.

Figure 4. SLT output.



6.0 Conclusion

SGI has been an industry leader in the real-time arena for over a decade. SGI is actively participating in the Linux community, identifying hold-offs and submitting modifications that result in reduced latencies. SGI delivers expertise and support and setup information that can't be found anywhere else.

Our long term strategy is to continue to work with the Linux community to enhance, fix and test the real-time capabilities of the standard Linux kernel. SGI tests the Linux kernel on a weekly basis to ensure that no regressions end up in the distribution. In addition, SGI will continue to add specific capabilities to REACT for Linux to address specific customer requirements.

7.0 References

1. http://www.sgi.com/products/software/react/linux_overview.html
2. http://www.sgi.com/products/software/react/
3. http://www.sgi.com/support/custeducation/courses/sw_dev/real-time_programming.html
4. SGI® REACT™ Real-Time for Linux® Programmer's Guide, (document number 007-4746-004) available at http://techpubs.sgi.com

Corporate Office
SGI
1140 East Arques Avenue
Sunnyvale, CA 94085-4602
650.960.1980

North America +1 800.800.7441
Latin America +55 11.5185.2860
Europe +44 118.912.7500
Japan +81 3.5488.1811
Asia Pacific +1 650.933.3000

3993 [2.2007] J15254