ORACLE

SPARC SERVERS

An Oracle White Paper
March 2013

SPARC M5-32 Server Architecture

ORACLE

Introduction

Organizations now rely on technology more than ever before. Today, compute systems play a critical role in every function from product design to customer order fulfillment. In many cases, business success is dependent on the continuous availability of IT services. Once only required in pockets of the data center, mainframe-class reliability and serviceability are now essential for systems throughout the enterprise. In addition, powering data center servers and keeping services running through a power outage are significant concerns.

While availability is a top priority, costs must also remain within budget and operational familiarity must be maintained. To deliver networked services as efficiently and economically as possible, organizations look to maximize the use of every IT asset through consolidation and virtualization strategies. As a result, modern IT system requirements reach far beyond simple measures of compute capacity. Highly flexible servers are required with built-in virtualization capabilities and associated tools, technologies, and processes that work to optimize sever utilization. New computing infrastructures must also help protect current investments in technology and training.

Oracle's SPARC M5-32 server is a highly reliable, easy-to-manage, vertically scalable system with many of the benefits of traditional mainframes—without the associated cost, complexity, or vendor lock-in. In fact, this server delivers a mainframe-class system architecture at open-systems prices. With symmetric multiprocessing (SMP) scalability from one to 32 processors, memory subsystems as large as 32 TB, and high-throughput I/O architectures, the SPARC M5-32 server easily performs the heavy lifting required by consolidated workloads. Furthermore, the server runs the powerful Oracle Solaris 10 and Oracle Solaris 11 operating systems that include leading virtualization technologies. By offering Dynamic Domains, Oracle VM Server for SPARC, dynamic reconfiguration, and Oracle Solaris Zones technology, the SPARC M5-32 server brings sophisticated mainframe-class resource control to an open-systems compute platform.

Overview of SPARC M5-32 Server Capabilities

The SPARC M5-32 server features a balanced, highly scalable SMP design that utilizes the latest generation of SPARC processors connected to memory and I/O by a high-speed, low-latency system interconnect that delivers exceptional throughput to applications. Also architected to reduce planned and unplanned downtime, this server includes stellar reliability, availability, and serviceability (RAS) capabilities to avoid outages and reduce recovery time. Design features such as advanced CPU integration and data path integrity, memory Extended-ECC (error correction code), end-to-end data protection, hot-swappable components, fault-resilient power options, and hardware redundancy boost the reliability of this server.

The SPARC M5-32 server also provides unmatched configuration flexibility. As in other high-end servers from Oracle, administrators can use Dynamic Domains (also called Physical Domains, or PDoms) to physically divide a single SPARC M5-32 server into multiple electrically isolated partitions, each running independent instances of Oracle Solaris. Hardware or software failures in one PDom do not affect applications running in other PDoms. Each PDom can run its own copy of Oracle VM Server for SPARC, a hypervisor-based virtualization technology. In this environment, logical domains are created, allowing multiple instances of Oracle Solaris to run in a PDom. This ability to mix physical and logical domains—with Oracle Solaris Zones—allows for broad and complex mixing of virtualization technologies.

Dynamic reconfiguration can then reallocate hardware resources between logical domains or reallocate virtual resources between logical domains within a physical domain—without interrupting critical systems. Table 1 shows the characteristics of the SPARC M5-32 server.

TABLE 1. CHARACTERISTICS OF SPARC M5-32 SERVER

	SPARC M5-32 SERVER
ENCLOSURE	One cabinet
SPARC M5 PROCESSORS	• 3.6 GHz with 48-MB level 3 (L3) cache • Up to 32 six-core SPARC M5 processors • Eight threads per core • Two SPARC M5 processors per CPU Memory Unit (CMU)
MEMORY	• Up to 32 TB • 64 DIMM slots per CMU • 16-GB and 32-GB DIMMs supported
INTERNAL I/O SLOTS	• Up to 64 x8 PCIe Generation 3 cards • Low-profile cards on a hot-pluggable carrier
INTERNAL STORAGE	• Up to 32 Serial Attached SCSI (SAS) drives
DYNAMIC DOMAINS	• Up to four physical domains
LOGICAL DOMAINS	• Up to 512 guests

SPARC M5-32 Server Components

The SPARC M5-32 server is mounted in an enterprise system cabinet and supports up to sixteen CPU memory units (CMU) and four I/O units (IOU). Fully configured, the SPARC M5-32 server houses 32 processor chips, 32 TB of memory, 64 short internal PCI Express (PCIe) slots, and it can be divided into four physical domains. Each physical domain can support up to 128 logical domains. In addition, the SPARC M5-32 server supports up to 32 disk drives. Twelve power supplies and 18 fan units power and cool the SPARC M5-32 server. Front and rear views of the SPARC M5-32 server are shown in Figure 1 and Figure 2.

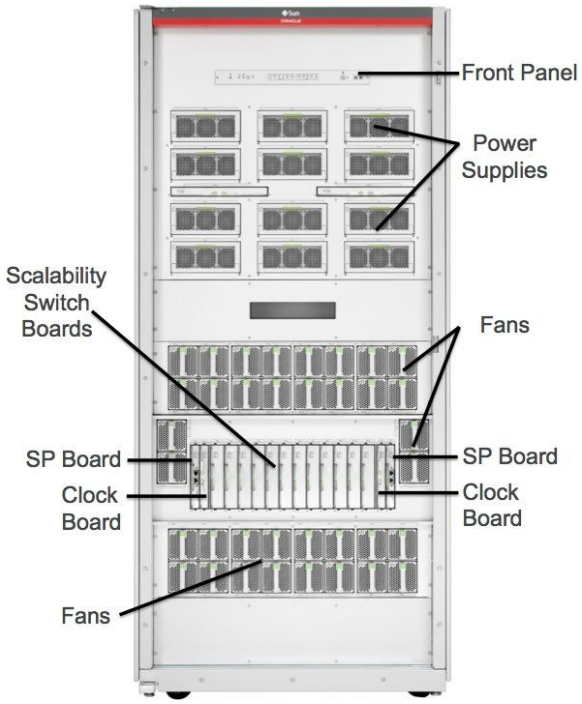

Figure 1. SPARC M5-32 server enclosure (front).

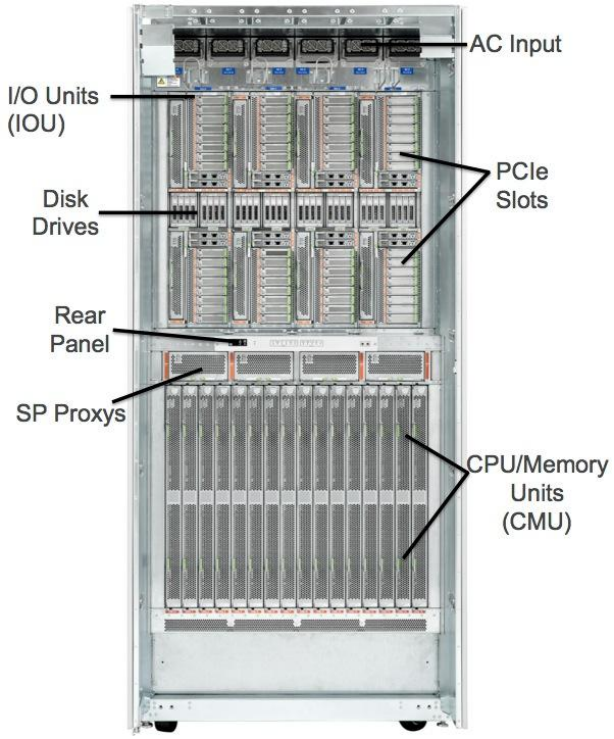

Figure 2. SPARC M5-32 server enclosure (rear).

System Architecture

Continually challenged by growing workloads and demands to do more with less, IT organizations realize that meeting processing requirements with fewer, more powerful systems can provide economic advantages. In the SPARC M5-32 server, the interconnect, processors, memory subsystem, and I/O subsystem work together to create a scalable, high-performance platform ready to address server consolidation needs. By taking advantage of this server, organizations can load multiple projects onto a single platform and accelerate application execution at lower costs.

System Component Overview

The design of the SPARC M5-32 server specifically focuses on delivering high reliability, outstanding performance, and true SMP scalability. The characteristics and capabilities of every subsystem within this server work toward this goal. A high-bandwidth system bus, powerful SPARC M5 processors, dense memory options, and fast PCIe expansion slots combine within this server to deliver high levels of uptime and throughput, as well as dependable scaling for enterprise applications.

System Interconnect

The system interconnect fosters high levels of performance, scalability, and reliability for the SPARC M5-32 server. The scalability switches within the SPARC M5-32 server provide point-to-point connections between the CPU, memory, and I/O subsystems. Offering more than one bus route between components enhances performance and allows system operation to continue in the event of a faulty switch. Indeed, the system interconnect used in these servers delivers as much as 3072 GB/sec of peak bandwidth, offering substantially more system throughput than Oracle's previous generation of high-end servers.

Memory

The memory subsystem of the SPARC M5-32 server increases scalability and throughput. The SPARC M5-32 server uses DDR3 DIMM technology. While multiple DIMM sizes are not supported within a single system board, DIMM capacities can vary across system boards. Available DIMM sizes include 16 GB and 32 GB. Further details about the memory subsystems are included in Table 2.

TABLE 2. MEMORY SUBSYSTEM SPECIFICATIONS

	SPARC M5-32 SERVER
MAXIMUM MEMORY CAPACITY	32 TB
DIMM SLOTS	32 per SPARC M5 processor
INCREMENTS PER CMU	16, 32, or 64 DIMMs per CMU

Beyond performance, the memory subsystem of the SPARC M5-32 server is built with reliability in mind. ECC protection is implemented for data stored in main memory and the following advanced features foster early diagnosis and fault isolation, which preserves system integrity and raises application availability.

- **Memory patrol**—Memory patrol periodically scans memory for errors. This proactive function prevents the use of faulty areas of memory before they can cause system or application errors, improving system reliability.
- **Memory Extended-ECC**—The memory Extended-ECC function of these servers provides single-bit error correction, supporting continuous processing despite events, such as burst read errors, that are sometimes caused by memory device failures. This feature is similar to IBM's Chipkill technology.
- **Memory lane sparing**—The memory lanes between the SPARC M5 processor memory controllers and the DIMMs continue to function even with the loss of a single lane in the data path.

System Clock

The SPARC M5-32 server is engineered with reliability in mind. In particular, the clock board is built with redundant internal components. Every SPARC M5-32 server comes with redundant clock boards. Further enhancing availability and easing maintenance, in the event that one board fails, the system can be restarted via the other board.

PCIe Technology

The SPARC M5-32 server uses a PCIe bus to provide high-speed data transfer within the I/O subsystem. PCIe technology doubles the peak data transfer rates of original PCI technology and reaches 8 GT/sec of throughput. In fact, PCIe was developed to accommodate high-speed interconnects such as Fibre Channel, InfiniBand, and Gigabit Ethernet. The SPARC M5-32 server provides a low-profile (LP) carrier that allows individual PCIe cards to be hot-plugged in or out of the IOU. There are 64 LP PCIe Gen3 slots in the SPARC M5-32 server.

Power and Cooling

SPARC M5-32 server uses separate modules for power and cooling. Sensors placed throughout the system measure temperatures on processors and key ASICs as well as the ambient temperature at several locations. Hardware redundancy in the power and cooling subsystems combined with environmental monitoring keep servers operating even under power or fan fault conditions.

Fan Unit

Fully redundant, hot-swappable fans function as the primary cooling system for the SPARC M5-32 server. If a single fan fails, the SP detects the failure and switches the remaining fans to high-speed operation to compensate for the reduced airflow. The SPARC M5-32 server can operate normally under these conditions, allowing ample time to service the failed unit. Replacement of fan units can occur without interrupting application processing.

Power Supply

The use of redundant power supplies and power cords adds to the fault resilience of the SPARC M5-32 server, as shown in Table 3. Power is supplied by redundant hot-swappable power supplies, helping to support continued server operation even if a power supply fails. Since the power units are hot-swappable, removal and replacement can occur while the system continues to operate.

The SPARC M5-32 server uses three-phase power. The three-phase power supply permits dual power feed. All six power cords come with every SPARC M5-32 server.

TABLE 3. POWER AND COOLING SPECIFICATIONS FOR THE SPARC M5-32 SERVER

	SPARC M5-32 SERVER
FAN UNITS	• 36 fan units • N+1 redundant
POWER SUPPLIES	• 7,000 watts • 12 units • N+1 redundant • Three-phase
POWER CORDS	• Three power cables (single feed) • Six power cables (dual feed)

Operator Panel

The SPARC M5-32 server features an operator panel to display server status, store server identification and user setting information, change between operational and maintenance modes, and turn on power supplies for domains (Figure 3). During server startup, the front panel LED status indicators verify SP and server operation.

Figure 3. The SPARC M5-32 server operator panel.

System Interconnect Reliability Features

Built-in redundancy and reliability features of the system interconnect enhance the stability of the SPARC M5-32 server. The interconnect protects against loss or corruption of data with full ECC protection on system buses. When a single-bit data error is detected in a CPU or I/O controller, hardware corrects the data and performs the transfer. The SPARC M5-32 server features degradable scalability switches and bus routes. In the rare event of a hardware failure within the interconnect, the system uses the remaining scalability switches upon restart, isolating the faulty switch and facilitating the resumption of operations. Each coherency link supports lane sparing for protection against loss of a lane in a particular link.

SPARC M5 Processor

The SPARC M5 processor is a highly integrated chip that eliminates the need for expensive custom hardware and software development by integrating computing, security, and I/O onto a single chip. Achieving binary compatibility with earlier SPARC processors, no other processor delivers so much performance in so little space and with such small power requirements. It enables organizations to rapidly scale the delivery of new network services with maximum efficiency and predictability. The SPARC M5 processor is shown in Figure 4.

Figure 4. The SPARC M5 processor allows organizations to rapidly scale the delivery of new network services and compute-intensive workloads with maximum efficiency and predictability.

When designing the next-generation of Oracle's multicore/multithreaded processors, the in-house design team started with the following key goals in mind:

- Radically increase the throughput computational capabilities over that of Oracle's SPARC64 VII+ processor for workloads that require this level of performance.
- Provide networking performance to serve network-intensive workloads.
- Provide end-to-end data center encryption with significantly higher performance as well as adding new ciphers implemented within hardware.
- Increase service levels and reduce planned and unplanned downtime.
- Improve data center capacities while reducing costs.

Oracle's multicore/multithreaded architecture is ultimately very flexible, allowing different modular combinations of processors, cores, and integrated components. The SPARC M5 processor utilizes the same S3 core architecture introduced in Oracle's SPARC T4 processor and used in Oracle's SPARC T5 processor.

The SPARC M5 processor design recognizes that memory latency is truly the bottleneck to improving performance. By redesigning the cores within each processor, designing a new floating-point pipeline, and further increasing network bandwidth, this processor is able to provide approximately 6x the throughput of the SPARC64 VII+ processor.

Each SPARC M5 processor provides six cores, with each core able to switch between up to eight threads (48 threads per processor) using a modified LRU (Least Recently Used) algorithm for thread choice. In addition, each core provides two integer execution pipelines, so that a single SPARC core is capable of executing two threads at a time. Unlike the SPARC64 VII+ processor, the SPARC M5 processor fetches one of eight threads for instruction propagation through stages of the pipeline to present to the select stage by the fetch3 stage. Thread instructions are grouped into two-instruction decode groups and proceed through decode, rename, and pick stages before proceeding to the issue

stage, after which they are sent to one of four subsequent execution pipelines, depending upon the type of instruction to be performed.

Up to this point, each instruction from any thread has proceeded through the pipeline independent of the type of instruction. Two instructions are issued for execution per cycle by the issue stage per cycle.

Figure 5 provides a simplified high-level illustration of the thread model supported by a 6-core SPARC M5 processor.

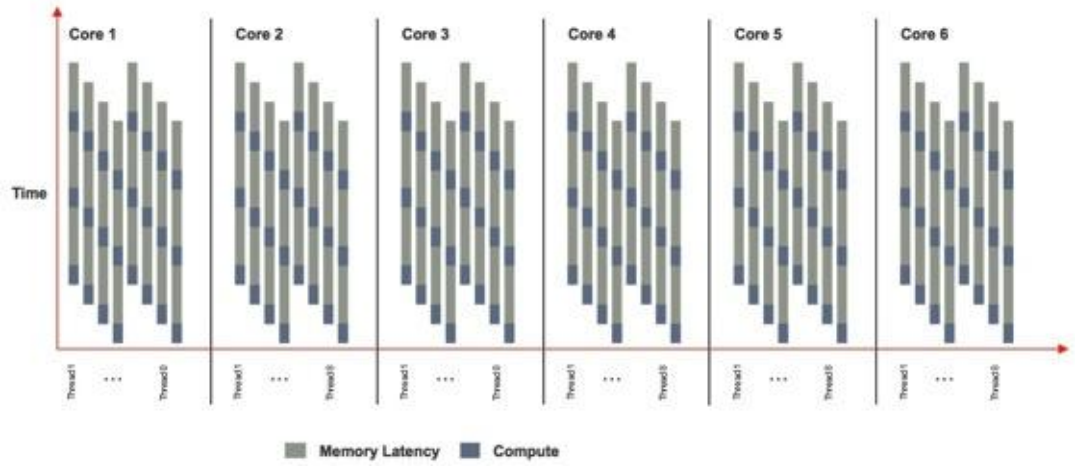

Figure 5. A single 6-core SPARC M5 processor supports up to 48 threads, with up to two threads running in each core simultaneously.

Oracle Solaris for Multicore Scalability

Oracle Solaris 10 and 11 are specifically designed to take full advantage of the considerable resources of the SPARC M5-32 system. In fact, Oracle Solaris provides key functionality for virtualization, optimal use high availability, unparalleled security, and extreme performance for both vertically and horizontally scaled environments. Oracle Solaris runs on a broad range of SPARC- and x86-based systems, and compatibility with existing applications is guaranteed. One of the most attractive features of systems based on the SPARC M5 processors is that they appear as a familiar SMP system to Oracle Solaris and the applications it supports. In addition, Oracle Solaris 10 and 11 have incorporated many features to improve application performance on Oracle's multicore/multithreaded architectures.

- **Accelerated cryptography.** Accelerated cryptography is supported through the cryptographic framework in Oracle Solaris (initial release) as well as the SPARC M5 processor. The SPARC M5 processor permits access to cryptographic cypher hardware implementations. For the first time, through user-level instructions, the cyphers are implemented within the appropriate pipeline itself rather than as a co-processor. This means a more efficient implementation of the hardware-based cyphers as well as no privilege-level changes, resulting in large increase in efficiency in cryptographic algorithm calculations. In addition, database operations can make much more efficient use of the various cryptographic cyphers that are implemented within the instruction pipeline itself.

- **Critical thread optimization.** Oracle Solaris 10 and 11 permit either a user or a programmer to allow the Oracle Solaris Scheduler to recognize a critical thread by means of raising its priority to 60 or above through the use of either the CLI or through system calls to a function. If this is done, that thread will run by itself on a single core, garnering all the resources of that core for itself. The one condition that would prevent this single thread from executing on a single core is when there are more runnable threads than available CPUs. This limit was put into place to prevent resource starvation to other threads. Further enhancements to critical thread optimization are planned for Oracle Solaris.
- **Multicore/multithreaded awareness.** Oracle Solaris 10 and 11 are aware of the SPARC M5 processor hierarchy, so the scheduler can effectively balance the load across all available pipelines. Even though it exposes each of these processors as 48 logical processors, Oracle Solaris understands the correlation between cores and the threads they support, and it provides a fast and efficient thread implementation.
- **Fine-granularity manageability.** For the SPARC M5 processor, Oracle Solaris 10 and 11 have the ability to enable or disable individual cores and threads (logical processors). In addition, standard Oracle Solaris features, such as processor sets, provide the ability to define a group of logical processors and schedule processes or threads on them.
- **Binding interfaces.** Oracle Solaris allows considerable flexibility in that processes and individual threads can be bound to either a processor or a processor set, as required or desired.
- **Support for virtualized networking and I/O.** Oracle Solaris contains technology to support and virtualize components and subsystems on the SPARC M5 processor, including support for the on-chip PCIe interfaces. As part of a high-performance network architecture, Oracle multicore/multithreaded–aware device drivers are provided so that applications running within virtualization frameworks can effectively share I/O and network devices.
- **Non-uniform memory access optimization in Oracle Solaris.** With memory managed by each SPARC M5 processor, these implementations represent a non-uniform memory access (NUMA) architecture. In NUMA architectures, the time needed for a processor to access its own memory is slightly shorter than that required to access memory managed by another processor. Oracle Solaris provides the following technology, which can specifically help to decrease the impact of NUMA on applications and improve performance on NUMA architectures:
  - **Memory placement optimization (MPO).** Oracle Solaris uses MPO to improve the placement of memory across the physical memory of a server, resulting in increased performance. Through MPO, Oracle Solaris helps ensure that memory is as close as possible to the processors that access it, while still maintaining enough balance within the system. As a result, many database applications are able to run considerably faster with MPO.

- **Hierarchical Lgroup Support (HLS).** HLS improves the MPO feature in Oracle Solaris by optimizing performance for systems with more-complex memory latency hierarchies. HLS lets Oracle Solaris distinguish between the degrees of memory remoteness, allocating resources with the lowest-possible latency for applications. If local resources are not available by default for a given application, HLS helps Oracle Solaris allocate the nearest remote resources.

- **Oracle Solaris ZFS.** Oracle Solaris ZFS offers a dramatic advance in data management, automating and consolidating complicated storage administration concepts and providing unlimited scalability with the world's only 128-bit file system. Oracle Solaris ZFS is based on a transactional object model that removes most of the traditional constraints on I/O issue order, resulting in dramatic performance gains. Oracle Solaris ZFS also provides data integrity, protecting all data with 64-bit checksums that detect and correct silent data corruption.

- **A secure and robust enterprise-class environment.** Best of all, Oracle Solaris does not require arbitrary sacrifices. Existing SPARC applications continue to run unchanged on SPARC M5 platforms, protecting software investments. Certified multilevel security protects Oracle Solaris environments from intrusion. The fault management architecture in Oracle Solaris means that elements such as Oracle Solaris Predictive Self Healing can communicate directly with the hardware to help reduce both planned and unplanned downtime. Effective tools, such as Oracle Solaris DTrace, help organizations tune their applications to get the most out of the system's resources.

I/O Subsystem

Powerful I/O subsystems are crucial to effectively moving and manipulating today's large data sets. The SPARC M5-32 server delivers exceptional I/O expansion and performance, helping organizations readily scale systems and accommodate evolving data storage needs.

The use of PCIe technology is key to the performance of the I/O subsystem within the SPARC M5-32 server. A PCIe bridge supplies the connection between the main system and components of the I/O unit, such as PCIe slots and internal drives.

In order to facilitate hot-plugging of PCIe adapter cards, the server utilizes PCIe cassettes. PCIe cards, which support PCIe hot-plugging, can be mounted by administrators into a PCIe cassette and inserted into an internal PCIe slot of a running server.

On the SPARC M5-32 server, there are four I/O units (IOUs). Each IOU has 16 PCIe Gen3 x8 slots, up to eight disk drives, and up to four base-I/O cards. Each IOU is dedicated to a specific domain configuration unit (DCU), which is described later when discussing domains. See Figure 6 for the components in a single IOU.

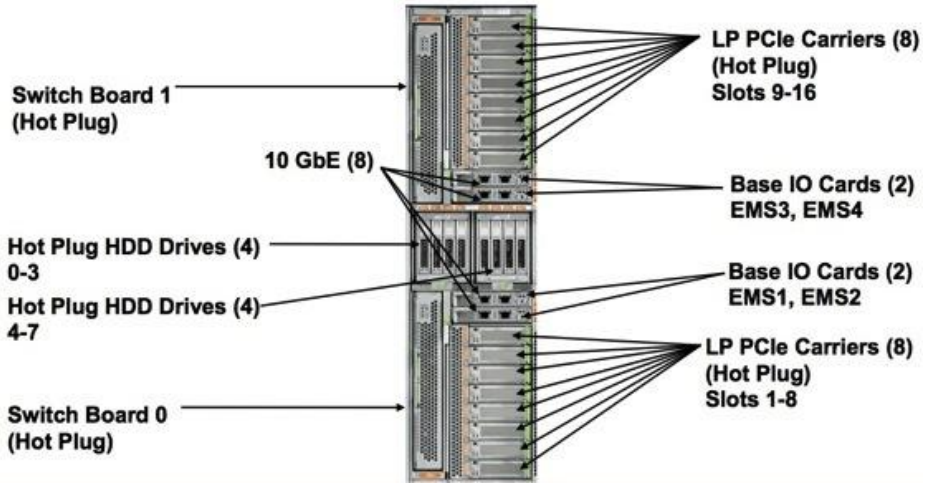

Figure 6. Components in an IOU.

Each PCIe slot has a carrier that allows for the hot-plugging of individual PCIe cards. Switch board 0 controls PCIe slots 1–8 and base I/O cards 1 and 2. Switch board 1 controls PCIe slots 9–16 and base I/O cards 3 and 4. Because the disk drives are dual ported, a disk is controlled by one base I/O card from pair 1 and 2 and a second base I/O card from pair 3 and 4.

When a physical domain boots, a SPARC M5 processor controls the PCIe slots. Figure 7 shows the mapping of SPARC M5 processors to PCIe slots in a fully loaded DCU.

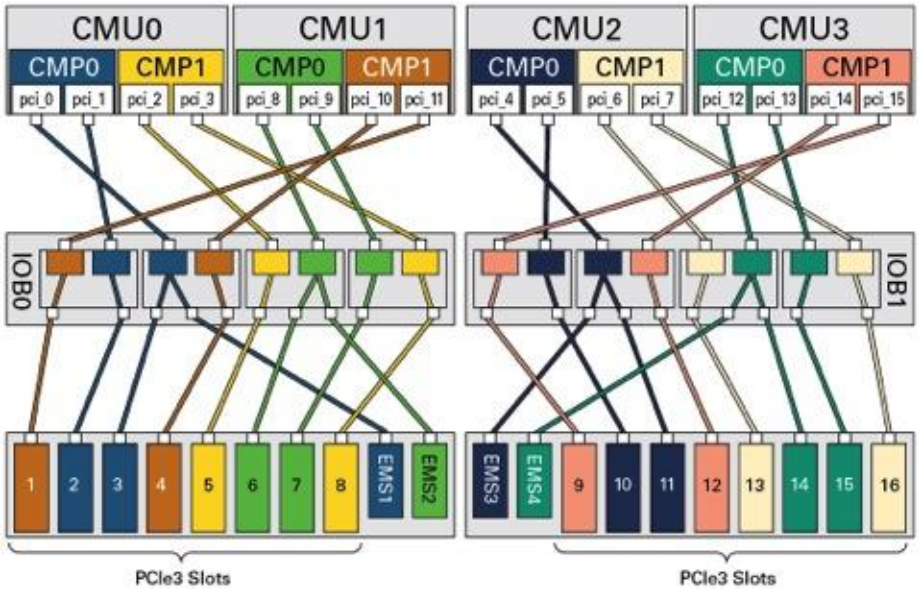

Figure 7. SPARC M5-32 server I/O subsystem per DCU.

Complete Virtualization Spectrum

Virtualization technology is increasingly popular as organizations strive to consolidate disparate workloads onto fewer, more-powerful systems, while increasing use. The SPARC M5-32 server is designed specifically for virtualization, providing all three levels of virtualization technologies: Dynamic Domains (i.e. physical domains), logical domains, and Oracle Solaris Zones (OS virtualization).

- Dynamic Domains are used to divide a single large system into multiple, fault-isolated servers.
- Logical domains created using Oracle VM Server for SPARC are used to virtualize a dynamic domain so that it can host multiple virtual machines (VMs), each running its own instance of Oracle Solaris.
- Oracle Solaris Zones enable OS virtualization so that a single instance of Oracle Solaris can securely isolate applications from each other and also allocate specific system resources to each zone.

Most important, Oracle's virtualization technology is provided as a part of the system, not an expensive add-on. See Figure 8 for an example of all the virtualization technologies in the SPARC M5-32 server.

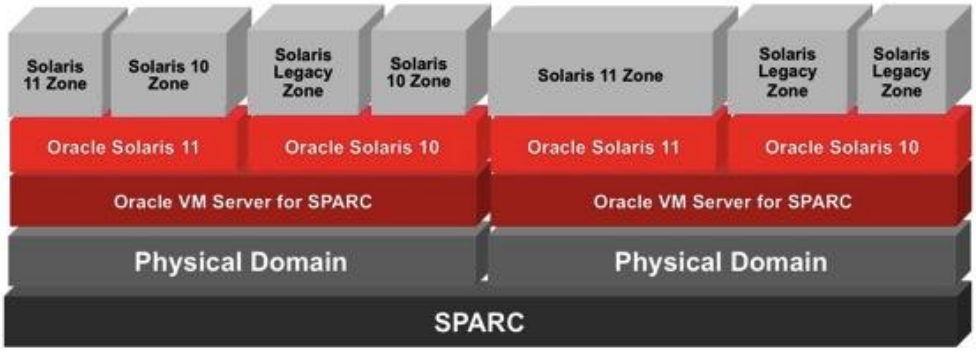

Figure 8. Virtualization technology stack on the SPARC M5-32 server.

Domain Configuration Units

Each domain configuration unit (DCU) is a grouping of four CPU memory units (CMUs), one IO unit (IOU), and a service processor proxy (SPP) board. The DCUs are building blocks that are used to configure the physical domains. All the SPARC M5 processors within a DCU communicate directly with each other. SPARC M5 processors in different DCUs must communicate with each other by using the scalability switch boards (SSB). Figure 9 shows the layout for DCU0.

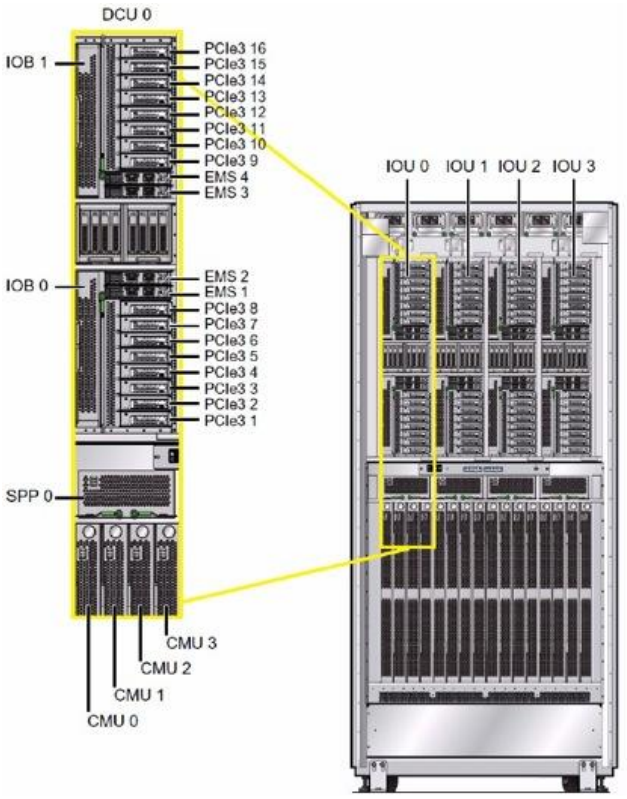

Figure 9. Layout of DCU0.

Figure 10 shows the layout of the four possible DCUs in the SPARC M5-32 server.

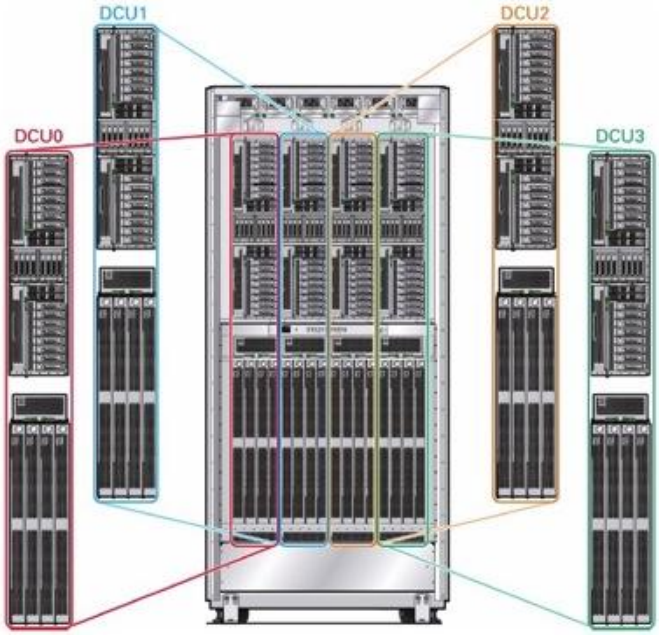

Figure 10. All four DCUs in the SPARC M5-32 server.

Dynamic Domains

In order to reduce costs and administrative burden, many enterprises look to server consolidation. However, organizations require tools that increase the security and effectiveness of hosting multiple applications on a single server. Dynamic Domains (i.e. PDoms) on the SPARC M5-32 server provide IT organizations with the ability to divide a single large system into multiple, fault-isolated servers each running independent instances of the Oracle Solaris operating system. With proper configuration, hardware or software faults in one domain remain isolated and unable to impact the operation of other domains. Each domain within a single server platform can even run a different version of Oracle Solaris, making this technology extremely useful for pre-production testing of new or modified applications. The maximum number of PDoms is 4.

There are two different types of PDoms: regular PDoms and bounded PDoms. Regular PDoms can grow from 4 to 32 processors and are made from combining whole DCUs. A bounded PDom is restricted to a single DCU. As such, it can have only four or eight SPARC M5 processors. The advantage of a regular PDom is that it can grow to include all CMU and IOU boards. The advantage of a bounded PDom is that it has lower latency and is not affected by the failure of an SSB.

A regular PDom can be made by combining any of the DCUs. There can be a single domain, made by combining DCU0–DCU3. There can be two PDoms by combining DCU0 and DCU1 in one domain and DCU2 and DCU3 in the other domain. The DCUs do not have to be next to each other. A PDom could also be made from DCU0 and DCU3. Three PDoms can be made from similar combinations.

Figure 11 shows two PDoms with a different number of DCUs assigned to them. PDom 0 has a single DCU assigned, while PDom 1 has two DCUs assigned to it.

Figure 11. Two PDoms

Logical Domains Supported by a Multithreaded Hypervisor

Like prior generations of SPARC processors, the SPARC M5 processor offers a multithreaded hypervisor that enables the creation of logical domains within a single PDom. The hypervisor is a small firmware layer that provides a stable virtual machine architecture that is tightly integrated with the processor. Multithreading is crucial, because the hypervisor interacts directly with the underlying multicore/multithreading processor. This architecture is able to context switch between multiple threads in a single core, a task that would require additional software and considerable overhead in competing architectures.

Corresponding layers of virtualization technology are built on top of the hypervisor, as shown in Figure 12. The strength of Oracle's approach is that all the layers of the architecture are fully multithreaded, from the processor up through applications that use the fully threaded Java application model. Far from being new technology, Oracle Solaris has provided multithreading support since 1992. This capability has been woven into all Oracle Solaris services at every level. In addition to the processor and hypervisor, Oracle provides fully multithreaded networking and the fully multithreaded

Oracle Solaris ZFS file system. Oracle VM Server for SPARC (previously called Sun Logical Domains or LDOMs), Oracle Solaris Zones, and multithreaded applications are able to receive exactly the resources they need.

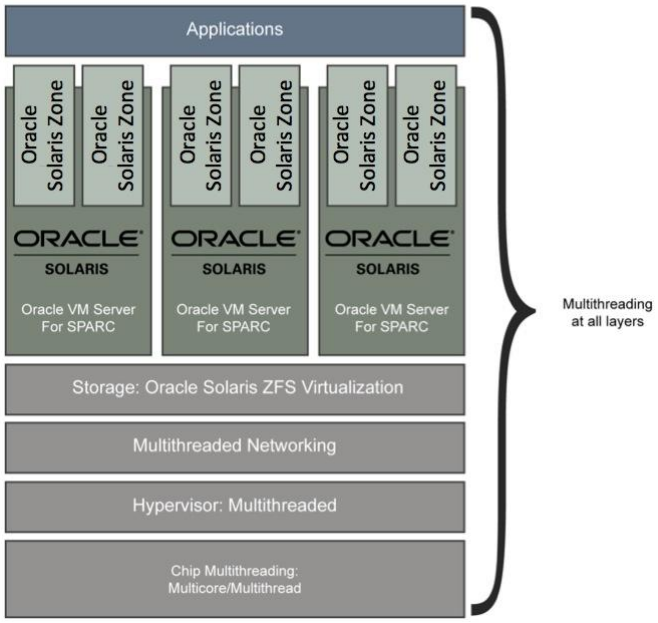

Figure 12. Oracle provides parallelization and virtualization at every level of the technology stack.

Oracle VM Server for SPARC

Supported in all servers from Oracle using Oracle's multicore/multithreaded technology, Oracle VM Server for SPARC provides full virtual machines that run an independent operating system instance. Each operating system instance contains virtualized CPU, memory, storage, console, and cryptographic devices. Within the Oracle VM Server for SPARC architecture, operating systems such as Oracle Solaris 10 are written to the hypervisor, which provides a stable, idealized, and virtualizable representation of the underlying server hardware to the operating system in each domain. Each domain is completely isolated, and the maximum number of virtual machines created on a single platform relies upon the capabilities of the hypervisor, rather than on the number of physical hardware devices installed in the system. For example, the SPARC M5-32 server with four SPARC M5 processors in a single DCU supports up to 128 domains, and each individual domain can run a unique OS instance.

Oracle VM Server for SPARC 3.0 has the ability to perform a live migration from one domain to another. As the term *live migration* implies, the source domain and application no longer need to be halted or stopped. Migration of a running application from one domain to another is now possible with Oracle VM Server for SPARC 3.0. This allows a logical domain on the SPARC M5-32 server to be live migrated to another PDom on the same server, to a different SPARC M5-32 server, or to a SPARC T3- or SPARC T5–based server from Oracle.

By taking advantage of domains, organizations gain the flexibility to deploy multiple operating systems simultaneously on a single platform. In addition, administrators can leverage virtual device capabilities

to transport an entire software stack hosted on a domain from one physical machine to another. Domains can also host Oracle Solaris Zones to capture the isolation, flexibility, and manageability features of both technologies. Deeply integrating Oracle VM Server for SPARC with the SPARC M5 processor, Oracle Solaris increases flexibility, isolates workload processing, and improves the potential for maximum server utilization.

Figure 13 shows logical domains created within each PDom.

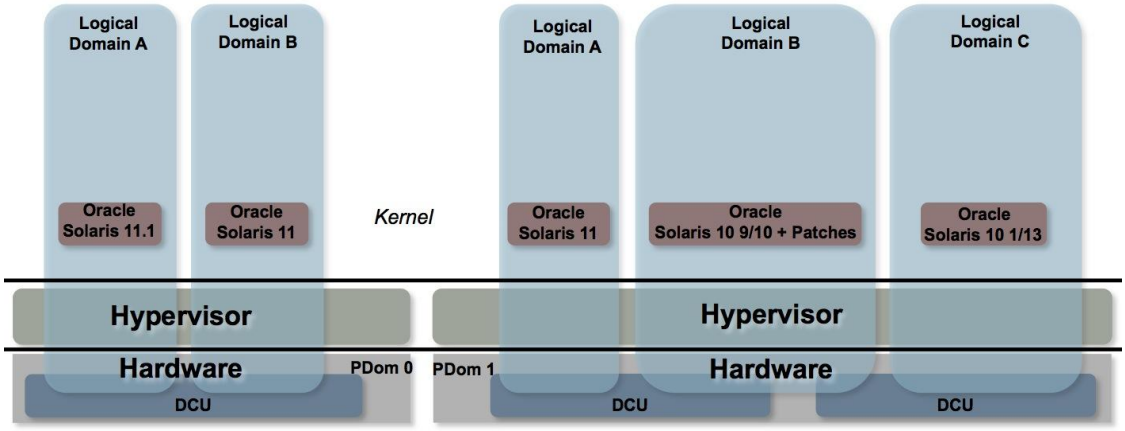

Figure 13. Logical domains in each PDom.

Oracle Solaris Zones

Oracle Solaris 11 provides a unique partitioning technology called Oracle Solaris Zones, which was called Oracle Solaris Containers in Oracle Solaris 10. This technology can be used to create an isolated and secure environment for running applications. An Oracle Solaris Zone is a virtualized operating system environment created within a single instance of Oracle Solaris. Oracle Solaris Zones can be used to isolate applications and processes from the rest of the system. This isolation helps enhance security and reliability since processes in one Oracle Solaris Zone are prevented from interfering with processes running in another Oracle Solaris Zone.

CPUs in a multiprocessor system (or threads in the SPARC M5 processor) can be logically partitioned into processor sets and bound to a resource pool, which in turn can be assigned to an Oracle Solaris Zone. Resource pools provide the capability to separate workloads so that the consumption of CPU resources does not overlap. They also provide a persistent configuration mechanism for processor sets and scheduling class assignment. In addition, the dynamic features of resource pools enable administrators to adjust system resources in response to changing workload demands.

Oracle Solaris Zones technology is an excellent tool for consolidating older environments on to newer platforms. This allows the applications to benefit from the increased performance of the latest CPU, memory, and I/O technology, as well as enabling applications to be deployed on a system with higher levels of RAS. Figure 14 shows a typical consolidation scenario.

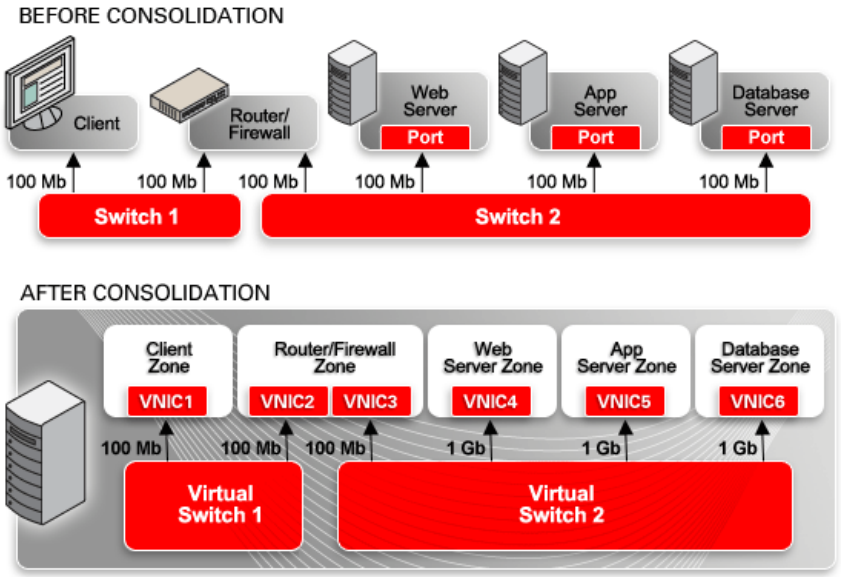

Figure 14. Advantages of server consolation with Oracle Solaris Zones.

Reduced TCO and business goals are achieved when multiple virtualization technologies are combined. For example, multiple Oracle Solaris Zones can run inside each logical domain. In Figure 15, each PDom has its own hypervisor and supports logical domains. Each logical domain separates different Oracle Solaris releases. This is usually done when there are different costs for running the latest OS. The next step is to isolate applications inside their own Oracle Solaris Zone. This allows for fine-grained resource allocation and process isolation. The use of Oracle Solaris Legacy Containers is for applications certified only for Oracle Solaris 8 or 9 and allows the administrator to take advantage of the new SPARC M5-32 performance and features, while not being forced to upgrade all the software until the business dictates the change.

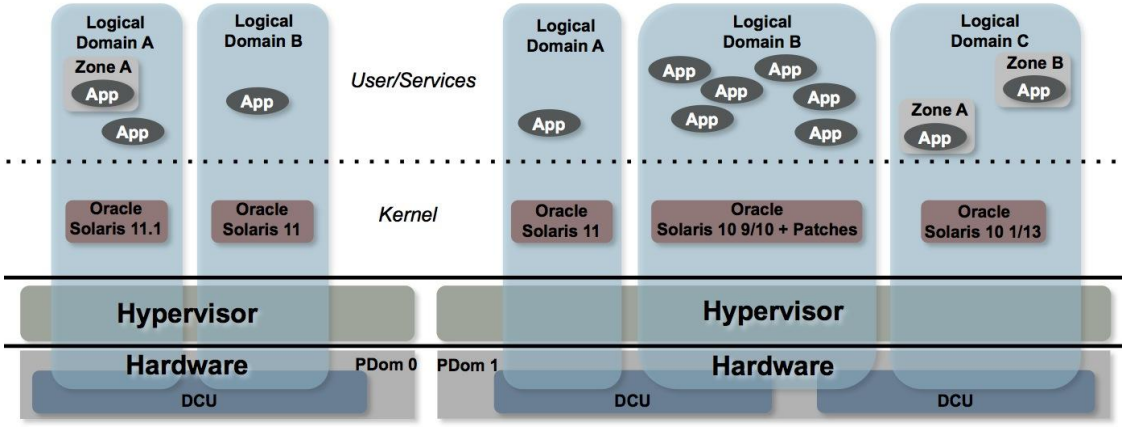

Figure 15. Multiple virtualization technologies on the SPARC M5-32 server.

The SPARC M5-32 server is configured with two PDoms. One PDom consists of a single DCU, while the second PDom consists of two DCUs. Any hardware issues in the first PDom will have no effect

on the second PDom, because there is complete hardware and electrical fault isolation between all PDoms.

Reliability, Availability, and Serviceability

Reducing downtime—both planned and unplanned—is critical for IT services. System designs must include mechanisms that foster fault resilience, quick repair, and even rapid expansion without impacting the availability of key services. Specifically designed to support complex network computing solutions and stringent high availability (HA) requirements, the SPARC M5-32 server includes redundant and hot-swappable system components, diagnostic and error recovery features throughout the design, and built-in remote management features. The advanced architecture of this reliable server fosters high levels of application availability and rapid recovery from many types of hardware faults, simplifying system operation and lowering costs for enterprises.

Advanced Reliability Features

Advanced reliability features included within the components of the SPARC M5-32 server increase the overall stability of the platform. For example, the SPARC M5-32 server includes multiple system controllers and degradable crossbar switches to provide redundancy within the system bus. Reduced component count and complexity within the server architecture contributes to reliability. In addition, advanced CPU integration and guaranteed data path integrity provide for autonomous error recovery by the SPARC M5 processors, reducing the time to initiate corrective action and subsequently increasing uptime.

Oracle Solaris Predictive Self Healing software further enhances the reliability of SPARC servers. The implementation of Oracle Solaris Predictive Self Healing software for the SPARC M5-32 server provides constant monitoring of CPUs and memory. Depending upon the nature of the error, persistent CPU soft errors can be resolved by automatically offlining a thread, a core, or an entire CPU. In addition, the memory page retirement function supports the ability to take memory pages offline proactively in response to multiple corrections to data for a specific memory DIMM.

Error Detection, Diagnosis, and Recovery

The SPARC M5-32 server features important technologies that correct failures early and keep marginal components from causing repeated downtime. Architectural advances that inherently increase reliability are augmented by error detection and recovery capabilities within the server hardware subsystems. Ultimately, the following features work together to raise application availability:

- End-to-end data protection detects and corrects errors throughout the system, ensuring complete data integrity.
- State-of-the-art fault isolation helps the server isolate errors within component boundaries and offline only the relevant chips instead of the entire component. Isolating errors down to the chip improves stability and provides continued availability of maximum compute power. This feature applies to CPUs, memory access controllers, crossbar ASICs, system controllers, and I/O ASICs.

- Constant environmental monitoring provides a historical log of pertinent environmental and error conditions.
- The host watchdog feature periodically checks for the operation of software, including the domain operating system. This feature also uses the SP firmware to trigger error notification and recovery functions.
- Dynamic CPU resource degradation provides processor fault detection, isolation, and recovery. This feature dynamically reallocates CPU resources to an operational system using dynamic reconfiguration without interrupting the applications that are running. The SPARC M5-32 server supports dynamic CPU degradation.
- Periodic component status checks are performed to determine the status of many system devices to detect signs of an impending fault. Recovery mechanisms are triggered to prevent system and application failure.
- Error logging, multistage alerts, electronic field replaceable unit (FRU) identification information, and system fault LED indicators contribute to rapid problem resolution.

Redundant and Hot-Swappable Components

Today's IT organizations are challenged by the pace of non-stop business operations. In a networked global economy, revenue opportunities remain available around the clock, forcing planned downtime windows to shrink and, in some cases, disappear entirely. To meet these demands, the SPARC M5-32 server employs built-in redundant, hot-pluggable, and hot-swappable hardware to help mitigate the disruptions caused by individual component failures or changes to system configurations. In fact, these systems are able to recover from hardware failures—often with no impact to users or system functionality.

The SPARC M5-32 server features redundant, hot-swappable power supply and fan units, as well as the option to configure multiple CPUs, memory DIMMs, and I/O cards. Administrators can create redundant internal storage by combining hot-pluggable disk drives with disk mirroring software. The SPARC M5-32 server also supports redundant, hot-swappable service processors and degradable scalability switch boards and also includes redundant clock boards. If a fault occurs, these duplicated components can support continued operation. Depending upon the component and type of error, the system might continue to operate in a degraded mode or might reboot—with the failure automatically diagnosed and the relevant component automatically configured out of the system. In addition, hot-swappable hardware within these servers speeds service and allows for simplified replacement or addition of components, without a need to stop the system.

With hot-plugging, devices must be prepared to leave Oracle Solaris, or Oracle Solaris must be explicitly told to add the device. For hot-swapping, only the SP must be notified of the removal or addition of a component. In both cases, it's not necessary to stop and start Oracle Solaris just to add or remove a device. Table 4 lists the different components on the M5-32 servers.

TABLE 4. HOT-SWAPPABLE AND HOT-PLUGGABLE COMPONENTS

COMPONENT	HOT-SWAPPABLE	HOT-PLUGGABLE

POWER SUPPLY AND FAN	Yes	No
PCIE CARD	No	Yes
DISK DRIVES	No	Yes
SERVICE PROCESSOR	Yes	No
SERVICE PROCESSOR PROXY	Yes	No

System Management Technology

Providing hands-on, local system administration for server systems is no longer realistic for most organizations. Around the clock system operation, disaster recovery hot sites, and geographically dispersed organizations lead to requirements for remote management of systems. One of the many benefits of Oracle servers is the support for lights-out data centers, enabling expensive support staff to work in any location with network access. The design of the SPARC M5-32 server combines with a powerful Service Processor (SP) running the Oracle Integrated Lights-Out Management (Oracle ILOM) software; this along with Oracle Enterprise Manager Ops Center software helps administrators remotely execute and control nearly any task that does not involve physical access to hardware. These management tools and remote functions lower administrative burden, saving organizations time and reducing operational expenses.

Oracle ILOM and Service Processor

The Oracle ILOM software on the SP provides the heart of remote monitoring and management capabilities for the SPARC M5-32 server. The SP consists of a dedicated processor that is independent of the server system and runs the Oracle ILOM software package. There is an internal 100BaseT network used for communication between the SP and the domains. While input power is supplied to the server, the SP constantly monitors the system even if all domains are inactive.

The SP regularly monitors the environmental sensors, provides advance warning of potential error conditions, and executes proactive system maintenance procedures as necessary. For example, the SP can initiate a server shutdown in response to temperature conditions that might induce physical system damage. The Oracle ILOM software package running on the SP helps administrators to remotely control and monitor domains, as well as the platform itself.

Using a network or serial connection to the SP, operators can effectively administer the server from anywhere on the network. Remote connections to the SP run separately from the operating system and provide the full control and authority of a system console.

On the SPARC M5-32 server, one SP is configured as active and the other is configured as a standby. The SP network between the two SPs facilitates the exchange of system management information. In case of failover, the SPs are already synchronized and ready to change roles.

Provided across all of Oracle's servers, the Oracle ILOM SP acts as a system controller, facilitating remote management and administration of the SPARC M5-32 server. The SP is full-featured and is similar in implementation to that used in Oracle's other modular and rackmount servers. As a result,

the server integrates easily with existing management infrastructure. Critical to effective system management, the Oracle ILOM SP does the following:

- Implements an IPMI 2.0–compliant SP, providing IPMI management functions to the server's firmware, OS, and applications and to IPMI-based management tools accessing the SP via the Oracle ILOM 3.2 Ethernet management interface. The SP also provides visibility to the environmental sensors on the server module and elsewhere in the chassis.
- Manages inventory and environmental controls for the server, including CPUs, DIMMs, and power supplies, and provides HTTPS, CLI, and SNMP access to this data.
- Supplies remote textual console interfaces.
- Provides a means to download upgrades to all system firmware.

The Oracle ILOM SP also allows the administrator to remotely manage the server, independent of the operating system running on the platform and without interfering with any system activity. Oracle ILOM can send e-mail alerts about hardware failures, warnings, and other events related to the server. Its circuitry runs independently from the server, using the server's standby power. As a result, Oracle ILOM 3.2 firmware and software continue to function when the server operating system goes offline or when the server is powered off. Oracle ILOM monitors the following server conditions:

- CPU temperature conditions
- Hard drive presence
- Enclosure thermal conditions
- Fan speed and status
- Power supply status
- Voltage conditions
- Oracle Solaris Predictive Self Healing, boot time-outs, and automatic server restart events

All PDom configuration, including the hot-plugging of CMU boards, is performed on the SP using Oracle ILOM commands via the CLI or in a Web browser.

Power Management

The power and cooling costs for servers are becoming significant, and lowering these costs is a top challenge in the corporate data center. Limitations in the availability of power and space to expand data centers force customers to look closely at the power efficiency of servers. Contracts with power providers, which specify penalties for exceeding stated power consumption, require servers to be able to cap their power consumption under customer control. Power efficiency and carbon footprint have become factors when customers evaluate servers.

Beyond the inherent efficiencies of Oracle's multicore/multithreaded design, the SPARC M5 and T5 processor incorporates unique power management features at both the core and memory levels of the

processor. These features include reduced instruction rates, parking of idle threads and cores, and the ability to turn off clocks in both cores and memory to reduce power consumption.

In addition to the power management support in Oracle ILOM, Oracle Solaris 11.1 provides a power manager. This way, Oracle Solaris determines which power savings features to enable based on the `poweradm` settings, which are set by the platform based on the system (Oracle ILOM) policy but can be overridden by the Oracle Solaris administrator.

Substantial innovation is present in the areas of

- Limiting speculation, such as conditional branches not taken
- Extensive clock gating in the data path, control blocks, and arrays
- Power throttling, which allows extra stall cycles to be injected into the decode stage

In a virtualized environment using Oracle VM Server for SPARC, the power management manager performs the following tasks when managing logical domain guests:

- Determining which power savings features to enable based on the power management policy
- Calling the Power Management engine to initiate power state changes on its resources to achieve a power adjustment or utilization level (for resources not owned by Oracle Solaris 11.1 guests), or telling the hypervisor to enable or disable hypervisor/hardware–managed power states. Only Oracle Solaris 11.1 guests have a power management peer.

On systems that support only one physical domain, a power policy is set for the entire system via the existing interface: `/SP/powermgmt policy`.

On systems with multiple physical domains, the power policy can be set for each physical domain, but not for the chassis as a whole. Control and monitoring can be done on the golden service processor proxy (SPP) of the PDom:

- `/SP/powermgmt/budget` controls the physical domain budget.
- `/SYS/PWRBS` monitors the status of the physical domain budget.

Alternatively, use `/Servers/PDomains/PDomain_<#>/SP/powermgmt/budget` on the primary SP, where <#> is the specific physical domain number. Configuring the physical domain budgets requires the platform administrator privilege/role.

An important element of controlling power utilization is the ability to cap power consumption. Both "hard caps" (power limit with grace period of 0) and "soft caps" (power limit with grace period > 0) are supported. A power cap enabled at the physical domain is measured against the power consumption of boards fully owned by the physical domain. For soft caps, the logical domain Power Manager adjusts the power states of the physical domain resources to converge to the cap. For hard caps, Oracle ILOM enforces the cap; it won't permit the system to boot if it would exceed the cap and a violation action of `HardPowerOff` is enabled. On SPARC M5-based servers, Oracle ILOM is able to meet a hard power cap with the help of the per-board field-programmable gate arrays (FPGAs). On

the SPARC M5-32 server, the only mechanism that Oracle ILOM has to meet the power cap is to prevent boards from being added to the PDom.

Managing the SPARC M5-32 Server Using Oracle Enterprise Manager Ops Center

Oracle Enterprise Manager Ops Center delivers a converged hardware management solution for the SPARC M5-32 server that integrates management across the infrastructure stack. With advanced virtualization management and reporting capabilities, application-to-disk management, intelligent configuration management and more, Oracle Enterprise Manager Ops Center helps IT managers reduce complexity and streamline and simplify infrastructure management. The inclusion of Oracle Enterprise Manager Ops Center with every SPARC M5-32 server enables data center administrators to monitor and manage the storage, network, servers, Oracle Solaris, and virtualized environments from a single interface. This improves operational efficiency and lowers operational costs.

Figure 16. Management of the Oracle stack.

Oracle Enterprise Manager Ops Center is the most comprehensive management solution for Oracle servers and engineered systems infrastructure. Offering a single console to manage multiple server architectures and myriad operating systems, Oracle Enterprise Manager Ops Center can manage the components in the SPARC M5-32 server using asset discovery, provisioning of firmware and operating systems, automated patch management, patch and configuration management, virtualization management, and comprehensive compliance reporting.

Oracle Enterprise Manager Ops Center automates workflow and enforces compliance via policy-based management—all through a single, intuitive interface. With Oracle Enterprise Manager Ops Center, IT staff can implement and enforce data center standardization and best practices, regulatory compliance, and security policies while efficiently deploying infrastructure to meet business requirements. Figure 17 shows the intuitive browser-based user interface for Oracle Enterprise Manager Ops Center.

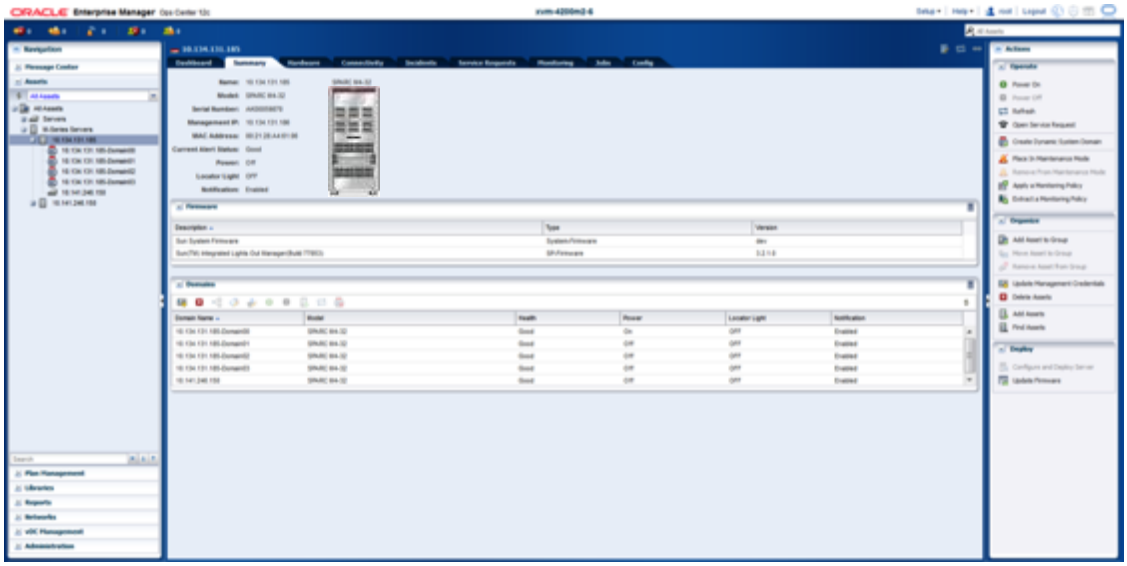

Figure 17. The Oracle Enterprise Manager Ops Center interface.

Oracle Solaris 11 Operating System

The SPARC M5-32 server supports both Oracle Solaris 10 and Oracle Solaris 11. Oracle Solaris 11 can be used for all domains. Oracle Solaris 10 can be used only for logical domain guests, not in the control logical domain.

Oracle Solaris includes the following features:

- **Advanced reliability**—Uptime is enhanced through comprehensive testing across an integrated solution stack and features such as Predictive Self Healing for hardware and software faults, data integrity with Oracle Solaris ZFS, and live observability with Oracle Solaris DTrace.
- **Superior performance**—Oracle Solaris is optimized for throughput and scalability for the latest SPARC processor technologies and has achieved outstanding performance for the Transaction Processing Performance Council (TPC) TPC-H and TPC-C benchmarks, Oracle's PeopleSoft, Oracle Business Intelligence Enterprise Edition, and many others.
- **Built-in virtualization**—Oracle Solaris Zones and Oracle VM Server for SPARC (previously known as Sun Logical Domains), along with other OS and network virtualization capabilities, enables efficient consolidation for flexibility and performance without significant overhead.
- **Pervasive security infrastructure**—Oracle Solaris provides the compartmentalization and control needed for multitenancy environments and enables governments and financial institutions to meet their strict requirements.
- **Committed support**—Oracle offers sustaining support for Oracle Solaris releases for as long as customers operate their systems, making it possible to keep software infrastructures in place for as long as it makes business sense.

Oracle Solaris Fault Management, Service Management Facility, and Predictive Self Healing

Oracle Solaris provides an architecture for building and deploying systems and services capable of fault management and predictive self-healing.

- The Predictive Self Healing feature in Oracle Solaris automatically diagnoses, isolates, and recovers from many hardware and application faults. As a result, business-critical applications and essential system services can continue uninterrupted in the event of software failures, major hardware component failures, and even software misconfiguration problems.
- The Oracle Solaris Fault Manager Architecture in Oracle Solaris collects data relating to hardware and software errors. This facility automatically and silently detects and diagnoses the underlying problem, with an extensible set of agents that automatically respond by taking the faulty component offline.
- The Oracle Solaris Service Manager Facility feature in Oracle Solaris creates a standardized control mechanism for application services by turning them into first-class objects that administrators can observe and manage in a uniform way. These services can then be automatically restarted if an administrator accidentally terminates them, if they are aborted as the result of a software programming error, or if they are interrupted by an underlying hardware problem.

Predictive Self Healing and the Fault Management Architecture can offline processor threads or cores in faults, retire suspect pages of memory, log errors or faults from I/O or any other issue detected by the system.

Conclusion

To support demands for greater levels of scalability, reliability, and manageability in the data center, infrastructures need to provide ever-increasing performance and capacity while becoming simpler to deploy, configure, and manage. The SPARC M5-32 server outfitted with the SPARC M5 processor, large memory capacity, and an inherently reliable architecture delivers new levels of performance, availability, and ease-of-use to enterprises. The sophisticated resource control provided by Dynamic Domains, Oracle VM Server for SPARC, and Oracle Solaris Zones further increases the value of this server by helping enterprises to optimize the use hardware assets. By deploying fast, scalable SPARC M5-32 servers from Oracle, organizations gain extraordinary performance and flexibility—a strategic asset in the quest to gain a competitive business advantage.

For More Information

For more information on Oracle's SPARC M5-32 sever and related software and services from Oracle, please see the references listed in Table 5.

TABLE 5. REFERENCES

SPARC systems	http://www.oracle.com/us/products/servers-storage/servers/sparc-enterprise
Oracle Solaris	http://www.oracle.com/us/products/servers-storage/solaris
Oracle Solaris Cluster	http://www.oracle.com/us/products/servers-

	storage/solaris/cluster/overview/index.html
Oracle Enterprise Manager Ops Center software	http://www.oracle.com/technetwork/oem/ops-center
Oracle Support	http://www.oracle.com/us/support/index.html

Appendix A: SPARC M5 Processor Architecture

The SPARC M5 processor extends Oracle's multicore/multithreaded initiative with an elegant and robust architecture that delivers real performance to applications. Figure 18 provides a block-level diagram of the SPARC M5 processor.

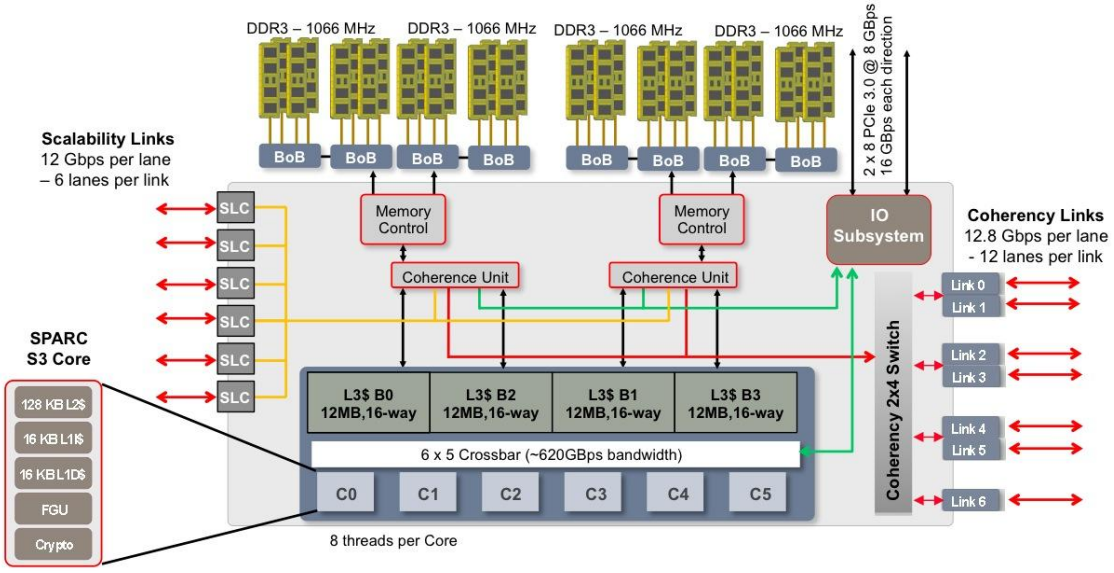

Figure 18. The SPARC M5 processor.

The SPARC M5 processor is a single chip multiprocessor (CMP) and contains six physical processor cores. Each physical processor core has full hardware support for eight strands, two integer execution pipelines, one floating-point execution pipeline, and one memory pipeline.

The SPARC M5 processor provides a robust out-of-order, dual-issue processor core that is heavily threaded among eight strands. It has a 16-stage integer pipeline to achieve high operating frequencies, advanced branch prediction to mitigate the effect of a deep pipeline, and dynamic allocation of processor resources to threads. This allows the SPARC M5 processor to achieve very high throughput performance (about 6x that of previous SPARC64 processors), while still scaling to very high levels of single-thread execution.

Each physical core has a 16-KB, four-way associative instruction cache (32B lines), a 16-KB, four-way associative data cache (32B lines), a 64-entry fully associative instruction translation lookaside buffer (TLB), and a 128-entry fully associative data TLB that are shared by the eight strands. It also includes a private, 128-KB, eight-way inclusive writeback L2 cache with 32B lines. Each physical core also includes cryptographic acceleration hardware, accessible via user-level instructions.

The SPARC M5 processor has coherence link interfaces to allow communication between up to eight SPARC M5 chips in a physical domain without requiring any external hub chip. There are seven coherence links, each with 12 lanes in each direction running at 153.6 Gb/sec. The SPARC M5 processor has seven coherence link units, two coherence units, and a cross bar (CLX) between coherence units and CLUs.

The SPARC M5 processor has scalability link interfaces to allow communication with the Scalability Switch Boards (SSB). This allows a SPARC M5 processor in one physical domain to communicate with SPARC M5 processors in a different physical domain. There are six scalability links, each with six lanes in each direction running at 72 Gb/sec.

The SPARC M5 processor interfaces to external DDR3 DIMMs via an external buffer-on-board (BoB) chip using proprietary unidirectional high-speed links. There are two memory links on the SPARC M5 processor. Each memory link is 12 lanes southbound and 12 lanes northbound and operates at 12.8 Gb/sec. Each memory communicates with two BoBs in a cascaded configuration. Each BoB chip has two DDR3 channels for total of up to 16 DDR3 channels per SPARC M5 processor. Each DDR3 channel has two DIMMs providing up to 32 DDR3 DIMMs per SPARC M5 processor (four DIMMs per BoB).

SPARC M5 Processor Cache Architecture

The SPARC M5 processor has a three-level cache architecture. Level 1 (L1) and Level 2 (L2) are specific to each core, that is, these two levels of cache are not shared with other cores. Level 3 (L3) is shared across all cores of a given processor. Cache sharing does not occur across another processor even though that processor may be in the same physical system. The SPARC M5 processor has L1 caches that consist of separate data and instruction caches. Both are 16 KB and are per core. A single L2 cache, again per core, is 128 KB. The L3 cache is *shared* across all six cores of the SPARC M5 processor and is 48 MB, has four banks, and is 16-way set associative. Figure 19 illustrates the relationship between L2 and L3 caches and shows them connected by a 4x5 crossbar:

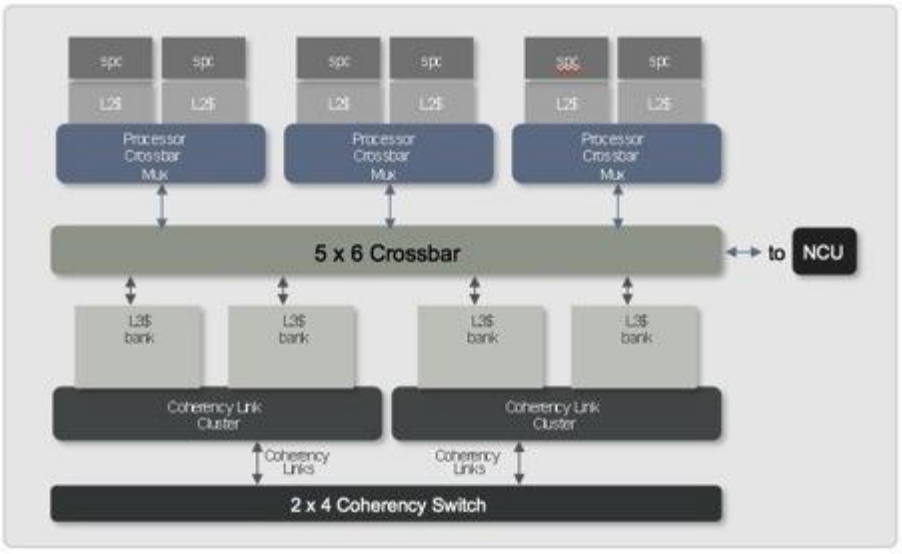

Figure 19. The relationship between Level 2 and Level 3 caches.

SPARC M5 Core Architecture

The SPARC M5 processor represents a fundamental redesign of the core from the previous SPARC64 multicore architecture and a continuation of the previous SPARC T4 processor from Oracle. Now included within the core are the following aspects that are more conventionally associated with superscalar designs:

- Out-of-Order (OoO) instruction execution
- Sophisticated branch prediction
- Prefetching of both instructions and data
- Much deeper pipelines (relative to previous versions of multicore processors from Sun/Oracle)
- Three levels of cache
- Support for a much larger memory management unit (MMU) page size (2 GB)
- Multiple instruction issue

All these characteristics in the SPARC M5 processor have yielded improvements in throughput performance by 6x.

There are many functional units, pipelines, and associated details that are present within the SPARC M5 core but are beyond the scope of this paper. However, due to the significantly new characteristics and features of the SPARC M5 core, this paper does attempt to touch upon the major exposed features or characteristics (that is, those that are visible to either programmers or users of a SPARC M5-32 system).

One aspect by which the designers of the SPARC M5 architecture were able to achieve a physical space savings of chip real estate was to reuse many physical pieces of a given core for widely varying functionality. For example, for each of the four major pipelines present within each core, the first 14 stages of each pipeline are actually shared. This represents a major space utilization efficiency by making each of the first 14 stages identical. Thus, they can be used by one of two integer instructions, a floating-point graphics instruction, or a load-store instruction. In Figure 20, the first six blocks represent the 14 identical stages, which are specifically defined in Figure 22.

Dynamic Threading

The SPARC M5 processor is dynamically threaded. While software can activate up to eight strands on each core at a time, hardware dynamically and seamlessly allocates core resources such as instruction, data, and L2 caches and TLBs, as well as out-of-order execution resources such as the 128-entry reorder buffer in the core. These resources are allocated among the active strands. Software activates strands by sending an interrupt to a halted strand. Software deactivates strands by executing a HALT instruction on each strand that is to be deactivated. No strand has special hardware characteristics. All strands have identical hardware capabilities.

Since the core dynamically allocates resources among the active strands, there is no explicit single-thread mode or multithread mode for software to activate or deactivate. If software effectively halts all strands except one on a core via critical thread optimization (described earlier in this document), the core devotes all its resources to the sole running strand. Thus, that strand will run as quickly as possible. Similarly, if software declares six out of eight strands as noncritical, the two active strands share the core execution resources.

The extent to which strands compete for core resources depends upon their execution characteristics. These characteristics include cache and TLB footprints, inter-instruction dependencies in their execution streams, branch prediction effectiveness, and others. Consider one process that has a small cache footprint and a high correct branch prediction rate such that when running alone on a core, it achieves two instructions per cycle (the SPARC M5 processor's peak rate of instruction execution). This is termed a high IPC process. If another process with similar characteristics is activated on a different strand on the same core, each of the strands will likely operate at approximately one instruction per cycle. In other words, the single-thread performance of each process has been cut in half. As a rule of thumb, activating N high-IPC strands will result in each strand executing at 1/N of its peak rate, assuming each strand is capable of executing close to two instructions per cycle.

Now consider a process that is largely memory-bound. Its native IPC will be small, possibly 0.2. If this process runs on one strand on a core with another clone process running on a different strand, there is a good chance that both strands will suffer no noticeable performance loss, and the core throughput will improve to 0.4 IPC. If a low-IPC process runs on one strand with a high-IPC process running on another strand, it's likely that the IPC of either strand will not be greatly perturbed. The high-IPC strand might suffer a slight performance degradation (as long as the low-IPC strand does not cause a substantial increase in cache or TLB miss rates for the high-IPC strand).

The guidelines above are only general rules-of-thumb. The extent to which one strand affects another strand's performance depends upon many factors. Processes that run fine on their own, but suffer from destructive cache or TLB interference when run with other strands, might suffer unacceptable performance losses. Similarly, it is also possible for strands to cooperatively improve performance when run together. This might occur when the strands running on one core share code or data. In this case, one strand may prefetch instructions or data that other strands will use in the near future.

The same discussion can apply between cores running in the chip. Since the L3 cache and memory controllers are shared between the cores, activity on one core can influence the performance of strands on another core.

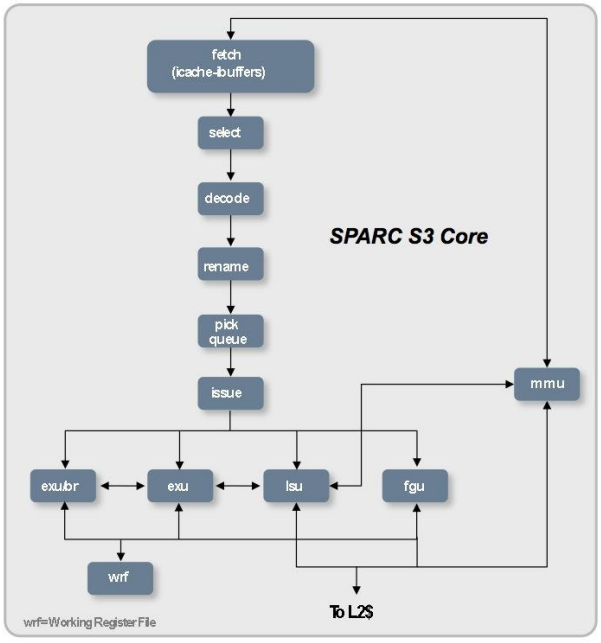

Figure 20. Block-level diagram of a single core of the SPARC M5 processor.

Components implemented in each core include the following:

- **Trap logic unit (not shown).** The trap logic unit (TLU) updates the machine state as well as handling exceptions and interrupts.
- **Instruction fetch unit.** The instruction fetch function is responsible for selecting the thread, fetching instructions from instruction cache (icache) for the selected thread, and providing up to four instructions to the select stage every cycle. It performs the following major functions:
  - Select the thread to be fetched.
  - Fetch instructions from icache for the selected thread, and place them in the Instruction Buffers for the select unit.
  - Predict direction and target of delayed control transfer instructions (DCTI) on the thread being fetched.
  - On icache miss, fetch data from the L2 cache (L2$), pre-decode it, and store it in icache.
- **Select unit.** The primary responsibility of the select unit is to schedule a thread for execution on processor's pipeline for each cycle. For each cycle up to one thread out of eight threads total can be selected for execution. A thread is in one of two states: Ready or Wait. Threads can be in a Wait state due to postsync conditions, mispredicted branches, lack of valid instructions, or other instruction related wait conditions. For each cycle, the select unit selects one thread for execution from among the ready threads using a least-recently-used (LRU) algorithm for fairness. For the selected thread, up to two instructions are sent to the decode unit per cycle.

- **Decode unit.** The decode unit on the SPARC M5 processor is responsible for the following:
  - Identifying illegal instructions
  - Decoding integer and FP sources and sinks for up to two instructions per cycle as well as detecting source/sink dependencies
  - Generating flat mapping of integer and FP registers
  - Decoding condition-code sources and destinations
  - Generating micro-ops for complex instructions
  - Generating instruction slot assignments
  - Detecting DCTI (delayed control transfer instruction) couples
  - Creating NOOPs when exceptions or annulling are detected
  - Maintaining speculative copies of window registers and executing certain window register instructions
  - Decoding up to two instructions every cycle
  - Preparing the data and the addressing for the Logical Map Tables (LMTs), which are part of the rename unit (RU)
- **Rename unit.** The rename unit is responsible for renaming the destinations of instructions and resolving destination-source dependencies between instructions within a thread as well as creating age vector dependency based on issue-slot. Renaming takes three cycles: R1, R2, and R3. For each cycle, the rename unit gets up to two instructions from the decode unit at the end of the D2 cycle. Each group of instructions is called a decode group. The rename unit does not break the decode group of instructions received from decode.
- **Pick unit**. The pick unit schedules up to three instructions per cycle out of a 40-entry pick queue (PQ). Up to three instructions (two instructions plus one store data acquisition op) are written into the PQ during the second phase of the R3. The PQ is read during the first phase of the pick cycle.
- **Issue unit.** The primary responsibility of the issue unit is to provide instruction sources and data to the execution units. The SPARC M5 processor has six execution units corresponding to the three issue slots as shown in Figure 21.

Issue Slot	*Unit*
0	Load/Store Unit Integer Execution Unit 0
1	Integer Execution Unit 1 Branch Unit FGU SPU
2	Store data operation

Figure 21. Relationship between issue slots and execution units.

- **Floating point/graphics unit.** A floating point/graphics unit (FGU) is provided within each core and it is shared by all eight threads assigned to the core. Thirty-two floating-point register file entries are provided per thread. A fused floating point Mul/Add instruction is implemented. In addition, the integer Fused Mul/Add instruction from the SPARC64 VII instruction set has been added. This also performs part of the cryptographic calculations based upon the algorithm being executed.

  The S3 core for the SPARC M5 processor implements a 16-stage integer pipeline, a 20-stage load-store pipeline, and a 27-stage floating-point graphics pipeline. All are present in each of the six cores of a SPARC M5 processor (Figure 22).

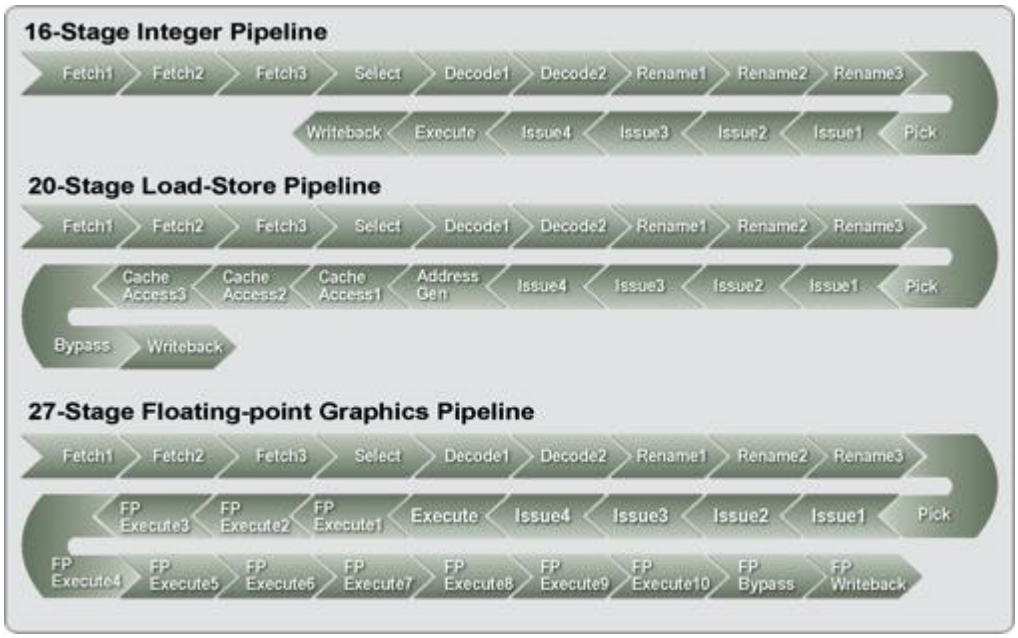

Figure 22. A 16-stage integer pipeline, a 20-stage load-store pipeline, and a 27-stage floating-point graphics pipeline are provided by each processor core.

- **Stream processing unit.** Each core contains a stream processing unit (SPU) that provides cryptographic processing. This functionality has been implemented within the core pipelines in the SPARC M5 processor and is accessible by 29 new user-level instructions.
- **Load-store unit.** The load-store unit (LSU) is responsible for processing all memory reference instructions and properly ordering all memory references. The LSU will receive load and store instructions out of order as they are picked by the pick unit. Loads might be issued out of order with respect to other loads, and stores might be issued out of order with the respect to other loads and stores. However, loads will not be issued ahead of previous stores. In addition to the memory references required by the instruction set, the LSU also contains a hardware prefetcher, which prefetches data into the L1 cache based upon detected access patterns.
- **Memory management unit.** The memory management unit (MMU) provides a hardware table walk (HWTW) and supports 8-KB, 64-KB, 4-MB, 256-MB and 2-GB pages.
- **Integer execution unit.** The integer execution unit (EXU) is capable of executing up to two instructions per cycle. Single-cycle integer instructions are executed in either the EXU0 (slot0) or EXU1 (slot1) pipeline. Load and store address operations go to EXU0 (slot0). Branch instructions are executed in EXU1 (slot1). Floating point, multicycle integer, and SPU instructions go through the EXU1 (slot1) pipeline. Store data operations go to EXU0 (slot2), but are not considered separate instructions by the EXU since the store address operation must also occur for the same instruction.

  To illustrate how the dual integer pipelines function, Figure 23 depicts the dual EXUs with the working register files (WRFs), floating-point register files (FRFs), and integer register files (IRFs) shown along with the various data paths.

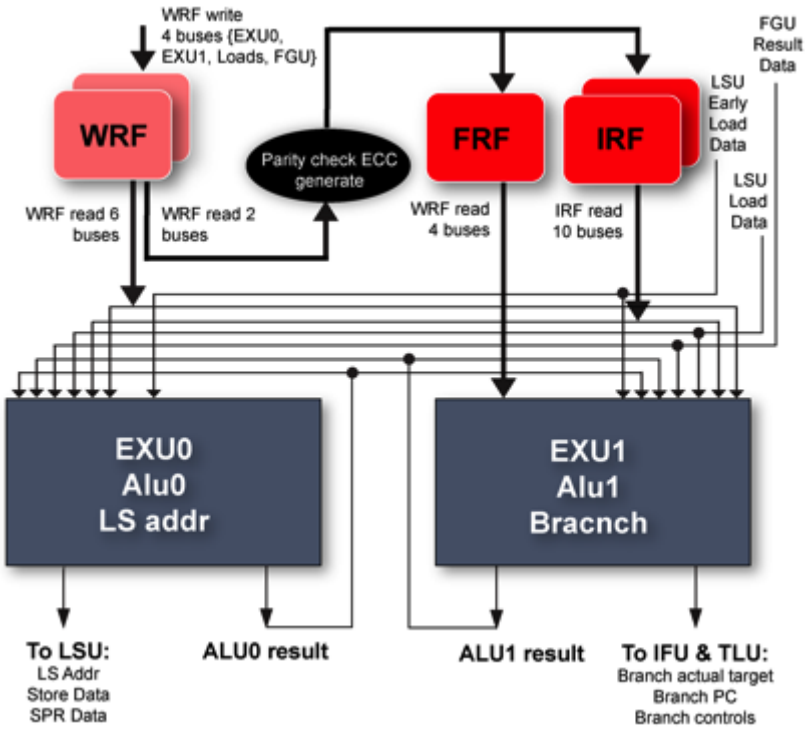

Figure 23. Threads are interleaved between the two integer pipelines and are restricted to EXU0 or EXU1 according to which type of integer operation is to be executed.

Stream Processing Unit

The SPU on each core is implemented within the core as part of the pipelines themselves and operates at the same clock speed as the core. The SPARC M5 processor supports the following cryptographic algorithms:

- DH, DES/3DES
- AES-128/192/256
- Kasumi, Camellia
- CRC32c, MD5
- SHA-1, SHA-224, SHA-256, SHA-384, SHA-512
- RSA via MPMUL/MONTMUL/MONTSQR instructions

A cryptographic algorithm (that is supported in hardware from the group previously listed) actually uses parts of the FGU and the integer pipelines. Figure 24 illustrates the basic logical pipeline of the SPU.

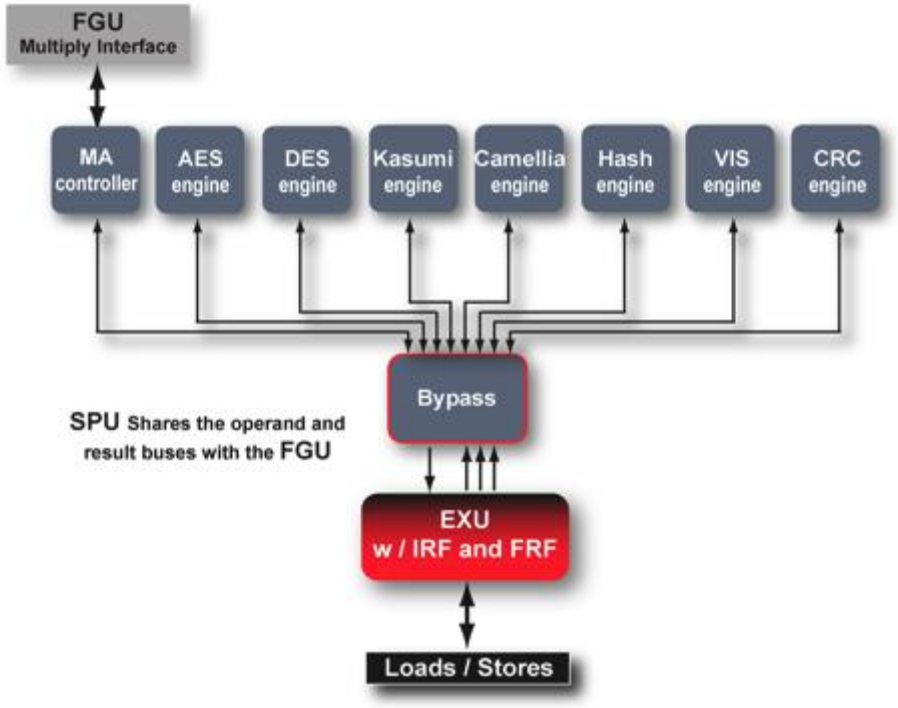

Figure 24. A logical depiction of the SPU pipeline that is in each core.

Integral PCIe Generation 3 Support

The SPARC M5 processor provides dual on-chip PCIe Generation 3 interfaces. Each operates at 8 Gb/sec per x1 lane bidirectionally through a point-to-point dual-simplex chip interconnect, meaning that each x1 lane consists of two unidirectional bit-wide connections, one for northbound traffic and the other for southbound traffic. An integral IOMMU supports I/O virtualization and process device isolation by using the PCIe BUS/Device/Function (BDF) number. The total theoretical I/O bandwidth (for an x8 lane) is 16 GB/sec, with a maximum payload size of 256 bytes per PCIe Gen3 interface. The actual realizable bandwidth is more likely to be approximately 14.8 GB/sec. An x8 SerDes interface is provided for integration with off-chip PCIe switches.

SPARC M5-32 Server Architecture
March 2013, Version 1.1
Author: Gary Combs

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Hardware and Software, Engineered to Work Together