CONSOLIDATING THE SUN STORE ONTO SUN FIRE™ T2000 SERVERS

HOW SUN IT USES THE ULTRASPARC® T1 PROCESSOR WITH COOLTHREADS™ TECHNOLOGY AND SOLARIS™ CONTAINERS

Casey Costley, Sun IT Chief Technology Office
Srinivasa Bodicharla, Sun Services IT Operations
Brad Coates, Sun Services IT Operations
Yunas Nadiadi, Sun IT Strategy and Architecture
Ragu Venkatesan, Sun IT Business Engagement and Applications

Sun BluePrints™ OnLine — December 2005

Part No 819-5148-10
Revision 1.0, 12/1/05
Edition: December 2005

Table of Contents

Consolidating the Sun Store onto Sun Fire™ T2000 Servers 1
How this Article is Organized 1
Background 2
The Latest Sun Technologies 2
Upgrading the Business Tier 3
Consolidation Proof-of-Concept 3
Current Sun Store Architecture 3
The Role of i2 Technologies 4
Current Sun Store Software Stack 5
Current Sun Store Physical Architecture 7
Motivation to Upgrade and Consolidate 10
Configuring the Sun Store Business Tier 12
Configuring Solaris Containers 12
Securing Containers and the Global Context 15
Installing the Sun Store Application 17
Reducing Datacenter Space, Power, and Cooling Requirements 18
Summary 18
About the Authors 19
Acknowledgments 19
References 19
Ordering Sun Documents 20
Accessing Sun Documentation Online 20

Consolidating the Sun Store onto Sun Fire™ T2000 Servers

Many datacenters today are at or near capacity in terms of space, power, and cooling, even as they are compelled to provide secure and available services that will scale into the future. Faced with hard limits on real estate, power, and thermal capacity, datacenter managers are increasingly changing the ways they evaluate infrastructure. Performance in particular must be viewed in an envelope of space, power, and dissipated heat — with performance per watt, performance per square foot, and performance per rack unit of paramount importance.

Sun faces these same demands and constraints in its own internal Information Technology (IT) organization, and is actively seeking effective solutions. In particular, Sun is deploying architectures and strategies to consolidate its own mission-critical Sun Store (*http://store.sun.com*) application using commercially available technology from both Sun and third-party vendors. Based on the UltraSPARC® T1 processor with CoolThreads™ technology, the new Sun Fire™ T2000 server offers an effective consolidation platform for these efforts, complemented by the flexibility of Solaris™ Containers partitioning technology introduced in the Solaris 10 Operating System.

Providing a unique insight into Sun's own operations and adoption of new products and technologies, this article discusses the existing Sun Store architecture and describes a timely real-world consolidation effort. In addition to architecture and configuration information, an analysis of anticipated savings in power, cooling, and space is also provided.

How this Article is Organized

This Sun BluePrints™ article is organized into the following sections:

- "Background" on page 2 provides the context in which this project was undertaken, including the constraints faced by Sun IT, and the technologies incorporated into the new version of the Sun Store.
- "Current Sun Store Architecture" on page 3 discusses the software architecture of the Sun Store and how the software is deployed onto the current physical architecture using Sun Enterprise™ 4500, 6500, and 10000 servers.
- "Motivation to Upgrade and Consolidate" on page 10 presents the value that both the Sun Fire T2000 server and the Solaris 10 Operating System offered when upgrading the entire software stack. The business logic tier is identified as the most important hardware to upgrade, as it is the greatest consumer of CPU resources.
- "Configuring the Sun Store Business Tier" on page 12 discusses how the software was configured on the Sun Fire T2000 server. The section outlines the commands used to create Solaris Containers and allocate resources to them. It summarizes the application software installation process and the security hardening process that followed.
- "Reducing Datacenter Space, Power, and Cooling Requirements" on page 18 provides an analysis that estimates a 90 percent savings in both power consumption and heat generation in the move to Sun Fire T2000 servers.

- “Summary” on page 18, briefly summarizes the tasks accomplished by the project, its current status at Sun, and concludes with a look to the future of how Sun IT plans to leverage the lessons learned from this initial proof-of-concept project.

Background

Sun IT, Sun’s internal information technology organization, provides information technology services worldwide for more than 40,000 employees, contractors, partners and vendors. Sun IT currently operates six datacenters across the globe supporting more than 600 business applications, processing more than five million e-mail messages per day, managing over 120 terabytes of data, and administrating more than four million internal web pages — and all with a goal of 99.995% availability.

As with most companies, Sun finds itself facing ever-increasing demands to deploy and upgrade more business applications within its internal infrastructure. Sun IT is challenged with accomplishing this within the limitations of its current datacenter space, power, and Heating, Ventilation, and Cooling (HVAC) capacity. With each datacenter operating with limited headroom, server consolidation has become one of the primary drivers in datacenter planning. Today’s servers must not only run cooler and in a smaller space: they must run at higher utilization rates through the hosting of multiple applications. In response to these demands and constraints, Sun IT decided to deploy a number of applications utilizing two of Sun's latest products: the Sun Fire T2000 server with CoolThreads Technology, and Solaris Containers implemented in the Solaris 10 Operating System.

The Latest Sun Technologies

The Sun Fire T2000 server is based on the UltraSPARC T1 processor with CoolThreads Technology. The processor uses Chip Multithreading (CMT) that is optimized for highly concurrent transactional processing. The UltraSPARC T1 processor improves throughput while using less power and dissipating less heat than conventional processor designs. Depending on the model, each Sun Fire T2000 server features an UltraSPARC T1 processor configured with four, six, or eight processor cores. Each core employs a 64-bit execution pipeline with four *logical processors* capable of processing four active execution contexts, sometimes referred to as *threads*. The result is that the 32 logical processors in an 8-core processor can process up to 32 active execution contexts.

Solaris Containers provide the ability to isolate software applications and services using flexible, software-defined boundaries. Each container can be customized with its own resources including secure, dedicated IP addresses, file systems, users, and assigned resources — including logical processors. This system resource partitioning and virtualization allows multiple applications or application stacks to be hosted on a single physical platform, each application with its own dedicated CPU resources.

The Sun Fire T2000 server, combined with the Solaris 10 OS, have provided the foundation for Sun IT to consolidate its applications onto a power and space-efficient platform. This allows old and new business applications to take advantage of the latest technologies while at the same time reducing power, HVAC and space demands.

Sun IT has identified a number of business application candidates for consolidation. These applications are resident in multiple IT service domains and include mission-critical, business-critical, and business-operational applications. Among these, the Sun Store is an online service that allows customers to research, price, and purchase Sun's product and service offerings from entry-level servers to spare parts. Sun IT considers the Sun Store as a mission-critical application. According to Sun policies, it must meet stringent scalability, availability, and security requirements.

Upgrading the Business Tier

The Sun Store uses a traditional three-tier Web-services architecture. The store is based on i2 Technologies' Intelligent Selling Solution (ISS) product, customized to meet Sun's online e-commerce requirements. In order to take advantage of the extended capabilities offered in i2 Technologies' latest ISS release (version 6.1), and meet the company's supportability requirements, Sun IT chose to upgrade the majority of the technology stack supporting the Sun Store. This included upgrading from the Solaris 8 OS and Sun Java™ Enterprise System 3 components to the Solaris 10 OS and Java Enterprise System 4, including Java System Application Server 8.1, Standard Edition, and Java System Web Server 6.1.

The first phase of the upgrade focused on the middle, or business-logic tier of the Sun Store. This tier supports the i2 software, requires the most computing horsepower, and currently consumes the most space, power, and cooling resources. The current Sun Store (version 1.0) production environment is replicated across four instances of the i2 software hosted in multiple Dynamic System Domains on Sun Enterprise 6500 and 10000 servers. Incoming requests from Web browsers are load balanced across these four domains. The Sun Fire T2000 configured with multiple Solaris Containers provides the capability to consolidate these domains onto a single two rack-unit server while still maintaining multiple instances of the i2 software as required for both availability and scalability. High availability in the business tier is supported by a redundant Sun Fire T2000 server providing fail-over capabilities.

Consolidation Proof-of-Concept

This project has served not only as a proof-of-concept for upgrading and consolidating the middle tier's hardware and software — it served as a demonstration for using Solaris Containers to support multiple software instances on a single server, hardening each container to meet security requirements, installing Java System Application Server into each container, and allocating hardware resources to each software instance. Preliminary performance and functionality testing has provided initial positive feedback, and the new Sun Store (version 2.0) is targeted to move into production in 2006.

Current Sun Store Architecture

The Sun Store is an online store where customers can browse the Sun catalog; view applicable list and discounted pricing; configure solutions that meet their needs; obtain quotations; place orders for hardware, software, and services; view order history, and status. The Sun Store is a high-availability, high-performance, internationalized platform built using software from i2 Technologies and running on the Sun Java Enterprise System integrated software stack.

The Sun Store is sized to handle thousands of concurrent users and tens of thousands of site visits per week. The Sun Store catalog contains more than 2,500 separate part numbers, and the application's goal is to accomplish complex page rendering in less than five seconds. The store's business and IT requirements include:

- Using Sun's global platform to support multiple Sun Stores (Retail, Custom, Partners, Specialty, Education, B2B)
- Internationalization and localization
- Integration with Sun common services
- Customer specific views and configurations
- Rule-based pricing
- Catalog export and syndication to customer and partner electronic procurement systems
- Customer profiling – portals, catalog & promotions

The Role of i2 Technologies

The Sun Store uses i2 Technologies software because of its functional characteristics, including its pricing solution with business-rules capability, and its ability to integrate with Sun's supply chain. Sun also chose i2 Technologies because its software can leverage the benefits of Sun hardware and software platforms, including the Sun Fire T2000 server, the Solaris 10 OS, and the Java Enterprise System.

The Sun Store has utilized the i2 Intelligent Selling System (ISS) Catalog, ISS Pricer, and ISS Quoter components. ISS multi-portal capability supports Sun's multiple stores by leveraging a single catalog, with country-specific and custom catalogs possible through ISS multi-catalog capability.

Integration with Sun Common Services

The middle, or business tier of the Sun Store's three-tier software stack integrates with Sun common services available in its local datacenter and available over Sun's wide-area corporate network. ISS integrates with existing Sun common services including login and session transfer, tax, Denied Restricted Parties List (DRPL), credit card, address validation, search configurator, leasing, order routing, history, and status, licensing, downloads, Sun systems of record: Content Management System (CMS), product catalog, pricing, lead time, customer agreements, and Customer Relationship Management (CRM).

Integrating Multiple Sun Applications

Using a set of common services allows the use of common features across multiple Sun applications. Common login and session transfer using Java System Access Manager provides virtually seamless user access across multiple Sun online services including the Sun Store, MySun Portal, the *www.sun.com* Web site, Order Status, and the Online Support Center.

Integrating ISS components with Sun common services enforces sharing of data, service administration and support, and re-uses the same business rules and content across all Sun applications. Integrating with core Sun systems of record leverages normalized data and avoids out-of-synch content issues that can arise if catalog and other data is simply duplicated.

Current Sun Store Software Stack

Figure 1 illustrates the three-tier architecture of the Sun Store, with the concentration of i2 software in the middle, or business tier. The figure illustrates the store's dependency on Sun common services elsewhere in Sun's IT infrastructure, and it illustrates the ability of the architecture to scale independently by tier — a capability used when consolidating to the Sun Fire T2000 server.

Presentation Tier

The presentation tier includes the use of JavaServer Pages™ (JSP™) and Java servlet technology; and i2's Rhythm eBusiness Framework (ReF) services running on the Java Enterprise System software stack. The Sun Store's presentation tier leverages i2 Technologies' Workflow Engine for managing user flow states, events and transitions across store functionality.

Java technology resource bundles are used to internationalize text, image, and error message components rendered by the JavaServer Pages and Java servlet software. JSP technology layout templates are also integrated for flexible, localized, layout control and to reduce the impact of user interface code or layout design changes.

The presentation tier also leverages i2's Workflow Content Engine for rapid design and development of workflow-based user interfaces, affording easier maintenance for future changes.

Business Tier

The business tier's core components include i2 ReF Application Services, Business Objects (BO), Database Engine (IO), and an Inktomi search engine. The ReF and BO components form the core business services platform that also manages the persistent business objects for user transactions. The Sun Store directly leverages i2 business components to implement catalog, pricing, cart/quote and checkout functions.

The Sun Store relies on additional common services including order status, configurator, leasing, licensing, payment and service token entitlement services.

Database Tier

Several database tables are used to store catalog, pricing, customer, and configuration data. Several external repositories are also critical to the Sun Store's operation, including user authentication, user profile synchronization, customer agreements, licensing, and order status, all of which are accessed via specific service Application Programming Interfaces (APIs).

The core database instance hosts the primary data for the catalog, pricer, and user profile services.

Horizontal Scaling Considerations

The Sun Store deployment is designed with a multi-string architecture for better user experience at high loads and resilient session handling in the event of a string failure. A *string* represents a stack of presentation and business logic tier servers that provide the Internet-facing Web servers and the business tier. Multiple strings can be configured to handle average workloads, and they can be increased rapidly to handle peak loads typical following new product announcements. All strings share the same database instance.

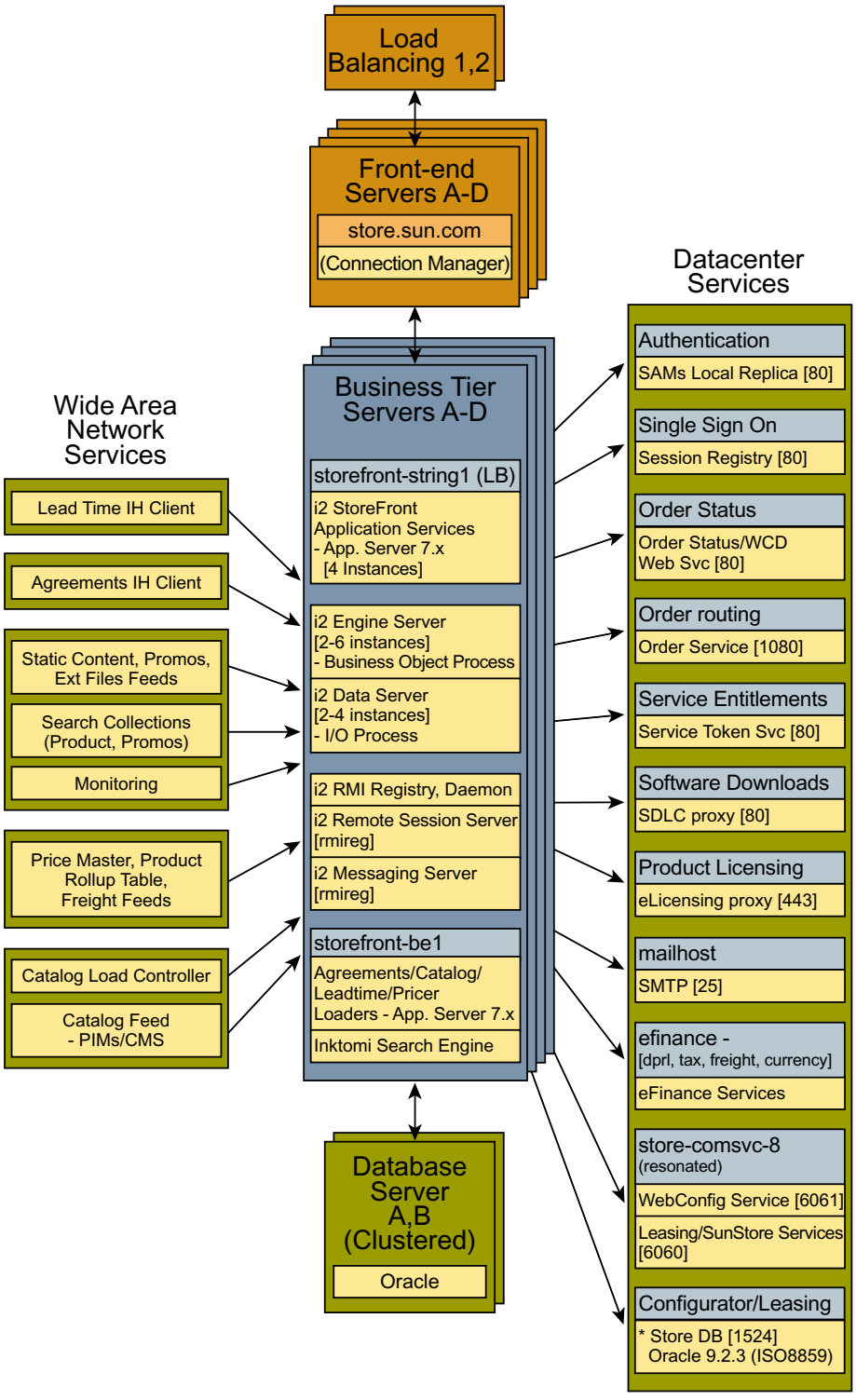

Figure 1. The Sun Store is implemented using a traditional three-tier model, with a high degree of dependency on common services used to support other Sun IT and customer systems

The load-balancing mechanism deployed for Sun Store is sensitive to the number of active sessions as well as server resource usage so that new user requests are always targeted to the least-loaded string to facilitate the best-possible user experience.

The use of multiple strings also provides failover capabilities so that a failure of one presentation tier server will redirect a user to another string's presentation tier servers for uninterrupted access. The architecture supports session replication across strings as well as within strings so that the failure of a single string component or an entire string will not interrupt a user session or require a new login. In the remote case that a string hosting a replicated session also fails, affected users would be required to login again. In the even more remote case that all string pipelines are offline, the Sun Store architecture provides for static site failover to a maintenance page that can point users to a site allowing only product browsing.

Figures 2 and 3 are configuration diagrams that illustrate horizontal scaling with multiple strings shown by separate presentation and business tier servers.

Vertical Scaling Considerations

In addition to horizontal scaling, the Sun Store architecture also supports vertical scaling within each string. The Java Enterprise System software and i2 ISS instances can be configured with defaults for average traffic load and increased rapidly to handle peak loads for product announcement traffic. This enables server resources to be more highly utilized in a shared datacenter environment.

The physical configuration diagrams that follow illustrate vertical scaling of software components in brackets, where the default scaling is shown. For example, “[4 instances]” indicates that four instances of a component are configured by default. The ability of the business tier to scale both horizontally and vertically lends itself well to consolidation on the Sun Fire T2000 server.

Current Sun Store Physical Architecture

The Sun Store is deployed on shared Sun Enterprise servers to leverage common infrastructure, security, release and support processes across multiple applications for economies of scale. Sun's datacenter also provides secure firewalls to protect Internet-facing applications in its De-Militarized Zone (DMZ) networks. The presentation tier is hosted in a DMZ, where servers mostly act as Web servers and proxies for HTTP and HTTPS traffic. The back-end servers include the business and database server tiers, and the common services hosted within Sun’s internal wide-area network. This network integrates to the back-end subnet to supply system-of-record feeds.

The current Sun Store is implemented using four strings, each using its own dedicated hardware. Within each string, software instances are scaled vertically for optimum resource utilization (Figure 2).

Presentation Tier

The presentation tier includes four Java System Web Server 6.0 instances that pass intermediate connections between user Web browsers and the business tier application logic. A load-balancing mechanism allocates the workload across four front-end servers. The Connection Manager component supports multi-string targets and handles failover. The current deployment provisions the presentation tier across four Sun Enterprise 4500 servers.

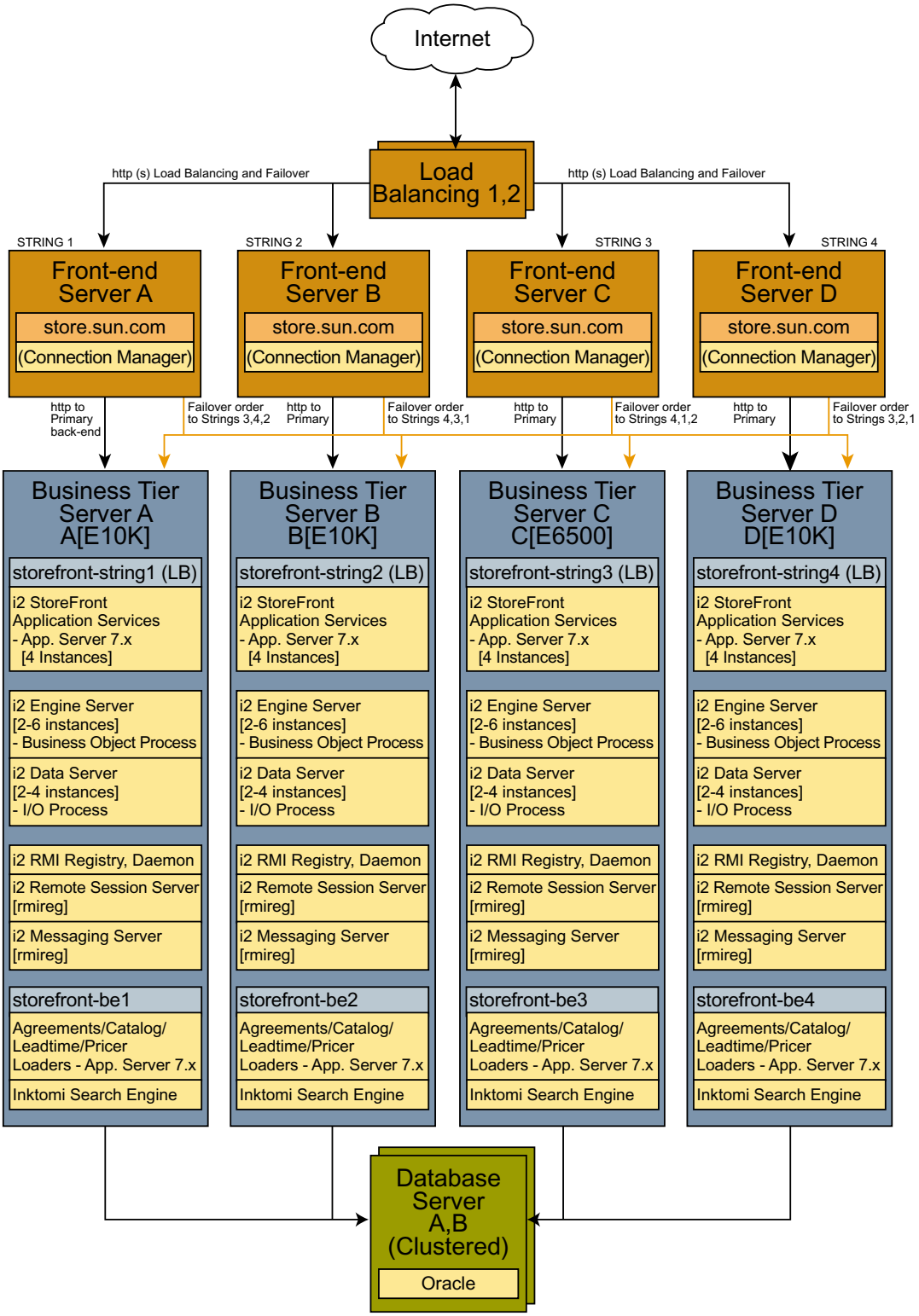

Figure 2. The current Sun Store implementation uses four strings, each using dedicated hardware

Business Logic Tier

The core i2 ISS application engines are distributed across four Dynamic System Domains configured on both Sun Enterprise 6500 and 10000 servers. Four Java System Application Server instances host i2 ReF application services which in turn call on 2-6 Application Server instances that host the i2 Business Object processes. The BO processes call 2-4 instances of the i2 Data Server (IO) processes that cache data and connect to the back-end database. Additional core components include i2 Remote Session Server for user session replication, i2 Messaging Server for e-mail services, and i2 DNA Bridge for protocol mapping between integration components.

The i2 components use TCP socket connections and Java Remote Method Invocation (“Java RMI”) protocols to service client requests. The Java RMI components bind to the Java RMI registry and broker via its daemon for client requests. The business tier also hosts an Inktomi search engine and data loaders for uploading catalog, lead-time, agreements and pricer feeds into the database tier.

Administration components (not shown) are hosted on two of the strings to provide sufficient availability in the event that one string with administration components is unavailable. The pricing service is similarly hosted to provide internal client applications and interface to the store's pricing functionality.

Session replication across the multi-string environment is configured to replicate user session data between string pairs. The string pairing is arranged so that each string in a pair resides on separate hardware platforms so that if one string goes down, user sessions can continue uninterrupted on the secondary string.

The Sun Store’s business tier also relies on common Sun services including single sign-on, order status, customer agreements, order routing; and existing services for Web configurator and leasing engines. The current deployment provisions the business tier across three domains on Sun Enterprise 10000 servers and one domain on a Sun Enterprise 6500 server. The CPU and memory resources allocated to each domain is summarized in Table 1.

Table 1. Business tier CPU and memory resources

String	Server Type	CPU Series	CPU Speed	Number of CPUs	Memory
A	Sun Enterprise 10000	UltraSPARC II	400 MHz	8	8 GB
B	Sun Enterprise 10000	UltraSPARC II	400 MHz	8	8 GB
C	Sun Enterprise 6500	UltraSPARC II	400 MHz	10	10 GB
D	Sun Enterprise 10000	UltraSPARC II	400 MHz	12	12 GB

Database Tier

The database tier consists of an Oracle database running in a clustered configuration for high availability and ease of maintenance. The database server leverages technologies including Sun StorEdge™ Network Data Replicator (SNDR) software, Oracle replication, and Shareplex. The current deployment architecture provisions both database instances onto separate Dynamic System Domains on two separate Sun Enterprise 10000 servers.

Motivation to Upgrade and Consolidate

The Sun Store was a primary candidate for upgrade because continued support of the i2 software required upgrading to the current version. This software upgrade prompted both an upgrade of the entire software stack to the versions listed in Table 2, and the hardware platform itself.

Table 2. Software upgrade opportunities

Tier	Functionality	Current Software and OS	Upgraded Software and OS
Presentation	Static content and connection management	Java System Web Server 6.0 SP8/Solaris 8	Java System Web Server 6.1, Standard Edition/Solaris 10
Business	All business modules	Java System Application Server 7.0, Standard Edition/Solaris 8	Java System Application Server 8.1, Standard Edition/Solaris 10
Business	Application core engines	i2 ISS 5.x/Solaris 8	i2 ISS 6.1/Solaris 10
Database	Back-end database	Oracle 8.1/Solaris 8	Oracle 9.2/Solaris 10

The existing need to reduce power, space, and HVAC resources provide an opportunity to consolidate onto the energy-saving Sun Fire T2000 server. With the business tier requiring the most CPU and memory resources today, it is the best candidate for consolidating onto the chip-multithreaded UltraSPARC T1 processor with CoolThreads technology. The modular architecture of the Sun Store allows the business tier to be replaced while maintaining the same connectivity to Sun common services and to the back-end database.

The processing power of the 8-core UltraSPARC T1 processor is such that a single Sun Fire T2000 server should be able to replace the entire set of 38 400 MHz UltraSPARC II processors that currently support the business tier. The current production environment uses four active strings deployed onto four physically separate domains to meet the needs for load and hardware redundancy. The processing power and reliability of the Sun Fire T2000 server, combined with the latest Java System Application Server and Solaris Containers technology allows a reduction in the number of active strings from four to two, with an increase in the number of application instances configured in each string (Figure 3).

The Sun Fire T2000 server meets many of Sun's RAS requirements through features including hot-swappable disk drives; redundant and hot-swappable power supplies and fans; environmental monitoring; and error detection and correction. For even greater availability, the number of strings can be doubled, and two servers used, with two strings active and two in standby mode. These four strings can be deployed in an active/standby configuration on each server, or with two active strings on one server and two standby strings on the second server. For testing purposes, both active strings were configured onto one Sun Fire T2000 server to validate whether it can indeed support the entire business-tier workload.

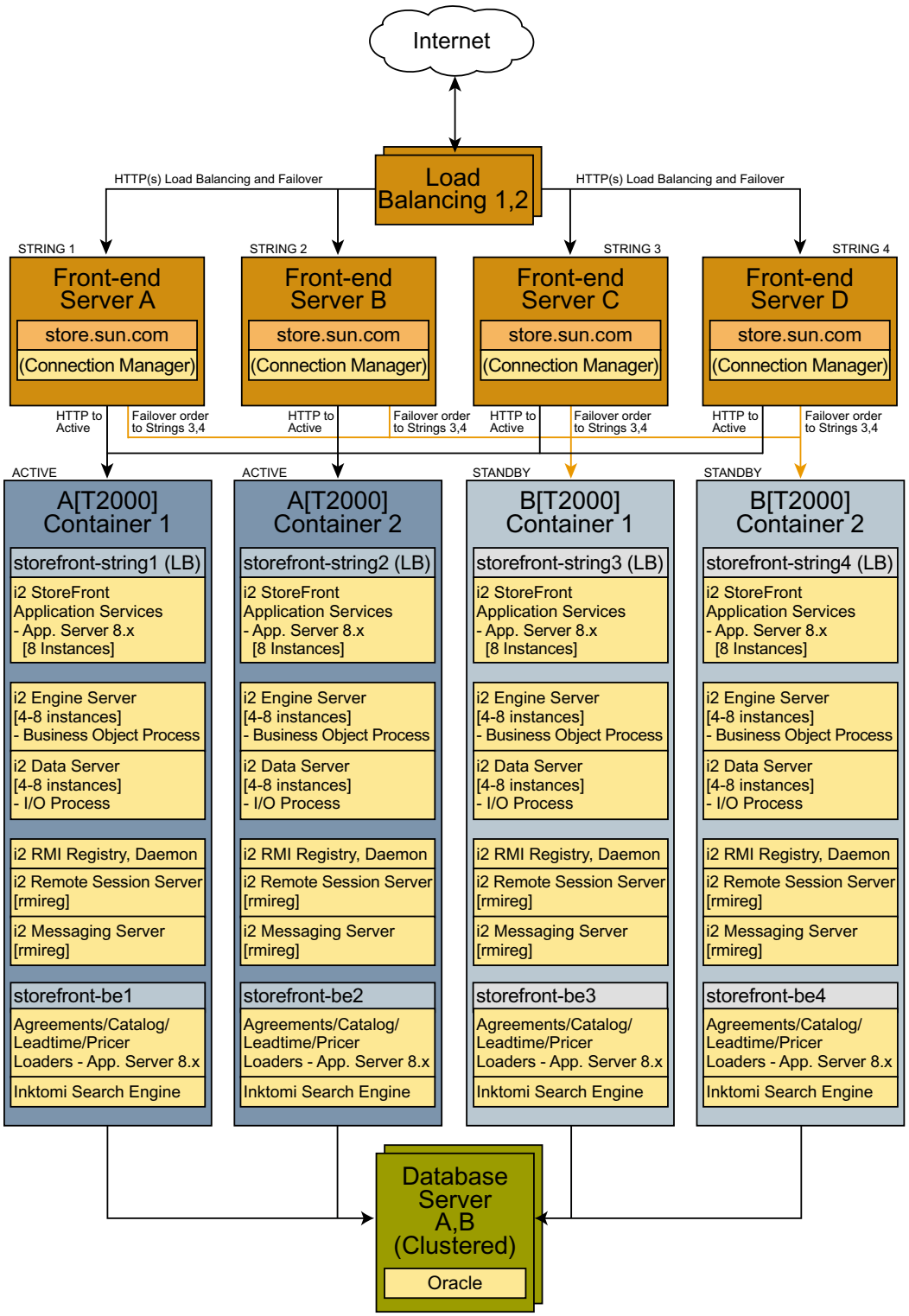

Figure 3. The proposed Sun Store architecture uses two strings hosted in Solaris Containers on each of two Sun Fire T2000 servers, with the second server used for failover

Solaris Containers, introduced in the Solaris 10 OS, is a mechanism for application isolation from both a security and a resource-management standpoint. Solaris Zones describe the secure, lightweight, virtualized OS services that appear as independent OS instances. Solaris Resource Manager can allocate resources to zones, and the combination of these two features make up what is known as Solaris Containers. Solaris Containers are used in the Sun Store to securely support multiple independent strings on a single server. They also guard against a rogue process in one string consuming more than its share of resources, affecting the ability of the other string to process user requests. Using Solaris resource pools, logical processors on the UltraSPARC T1 processor can be assigned to specific zones.

The Sun Fire T2000 server used in this evaluation featured an UltraSPARC T1 processor with 8 cores, each core supporting four hardware contexts. The Solaris OS presents these resources as 32 logical processors. Using Solaris resource pools, these logical processors can be assigned (using processor sets) to individual zones. For this implementation, two zones are configured to run on each of two Sun Fire T2000 servers, with each zone hosting a single string of the Sun Store application. Each zone was assigned eight logical processors, equivalent to the resources of two UltraSPARC T1 processor cores. As a result, this configuration uses only half the 32 logical processors available on each server, allowing ample headroom to add more logical processors to each zone, and leaving more than sufficient resources for the global zone. Sizing will be re-evaluated once load and performance testing has been completed. The CPU and memory allocation for the new business tier is summarized in Table 3.

Table 3. Proposed resource allocation for two primary and two failover Sun Fire T2000 servers

String	Server Type	Purpose	CPU Series	CPU Speed	CPU Allocation	Memory
A, B	Sun Fire T2000	Primary containers 1 & 2	8-core UltraSPARC T1	1 GHz	2 cores, or 8 logical processors per string	16 GB
C, D	Sun Fire T2000	Secondary containers 1 & 2	8-core UltraSPARC T1	1 GHz	2 cores, or 8 logical processors per string	16 GB

Configuring the Sun Store Business Tier

Having made the choice to deploy the Sun Store's business tier onto two containers on each of two Sun Fire T2000 servers, the next step was to implement that decision. The operating system was installed, containers created, each container hardened for security, and software installed. This section focuses on the process of creating the containers, allocating resources to them, and hardening each container. The result of this process is that:

- Security hardening using the Solaris Security Toolkit worked within the container environment just as it does in a global zone.
- Installing the required Java Enterprise System software into whole-root zones was straightforward, and the process generated no errors.

Configuring Solaris Containers

Each server was installed with the Solaris 10 3/05 Hardware 2 OS release. Each server was set up with two zones, each with two processor cores (eight logical processors) allocated to each zone. Two whole-

root containers were set up on each of the two servers. Whole-root containers include a container-specific copy of the entire root filesystem, so that software installation and hardening tools cannot distinguish between their container environment and the global zone. This allowed software with complex installation requirements — like Sun Java Enterprise System 4 — to work within containers.

The following commands illustrate using `zonecfg` to create a new whole-root zone named `ibl-sun4v-z1`:

```
#zonecfg -z ibl-sun4v-z1
create -bibl-sun4v-z10: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:ibl-sun4v-z1> create
```

The zone's path is specified, and autoboot is set to true. The `zonepath` variable describes where to locate the zone's root filesystem:

```
zonecfg:ibl-sun4v-z1> set zonepath=/export/zone/ibl-sun4v-z1
zonecfg:ibl-sun4v-z1> set autoboot=true
```

Next, the `/export` filesystem is added to the zone, with its mount point specified:

```
zonecfg:ibl-sun4v-z1> add fs
zonecfg:ibl-sun4v-z1> set dir=/export/home
zonecfg:ibl-sun4v-z1> set special=/export/home
zonecfg:ibl-sun4v-z1> set type=lofs
zonecfg:ibl-sun4v-z1> add options nosuid
zonecfg:ibl-sun4v-z1> end
```

Similarly, the `/share` filesystem is added:

```
zonecfg:ibl-sun4v-z1> add fs
zonecfg:ibl-sun4v-z1> set dir=/share
zonecfg:ibl-sun4v-z1> set special=/export/share
zonecfg:ibl-sun4v-z1> set type=lofs
zonecfg:ibl-sun4v-z1> add options ro
zonecfg:ibl-sun4v-z1> end
```

A virtual network interface is added to the zone, along with its IP address and the physical device to be used:

```
zonecfg:ibl-sun4v-z1> add net
zonecfg:ibl-sun4v-z1> set address=10.1.224.72
zonecfg:ibl-sun4v-z1> set physical=ipge0
zonecfg:ibl-sun4v-z1> end
```

Finally, the zone's configuration is verified and stored, and the `zonecfg` command is exited:

```
zonecfg:ibl-sun4v-z1> verify
zonecfg:ibl-sun4v-z1> commit
zonecfg:ibl-sun4v-z1> exit
```

Output from the `zoneadm` command confirms that the zone has been created. Every Solaris 10 OS installation using Solaris Containers has a global context. The output from `zoneadm` shows the global zone with ID 0 and the new zone `ibl-sun4v-z1` as context 1:

```
#zoneadm list -vc
  ID NAME             STATUS         PATH
   0 global           running        /
   1 ibl-sun4v-z1     configured     /export/zone/ibl-sun4v-z1
```

Now the operating system can be installed into the zone and the zone booted:

```
#zoneadm -z ibl-sun4v-z1  install
#zoneadm -z ibl-sun4v-z1  boot
```

The `zoneadm` command output shows that the zone has booted and is running:

```
#zoneadm list -vc
  ID NAME             STATUS         PATH
   0 global           running        /
   1 ibl-sun4v-z1     running       /export/zone/ibl-sun4v-z1
```

The following sequence of commands creates a pool of eight logical processors on the UltraSPARC T1 processor:

```
pooladm -e
poolcfg -c discover
poolcfg -c "create pool store-string1-pool"
poolcfg -c info
poolcfg -c "create pset store-string1-pset (uint pset.min=8; uint
pset.max=8)"
poolcfg -c info
poolcfg -c "associate pool store-string1-pool (pset store-string1-pset)"
poolcfg -c info
pooladm -c
poolbind -p store-string1-pool -i zoneid  ibl-sun4v-z1
```

The following command binds the resource pool to the active zone `ibl-sun4v-z1`:

```
poolbind -p store-string1-pool -i zoneid  ibl-sun4v-z1
```

When logged into a zone, the operating system can see only the eight logical processors assigned to the zone:

```
[Connected to zone 'ibl-sun4v-z1' pts/6]
# psrinfo
0       on-line   since 01/02/2000 18:03:28
1       on-line   since 01/02/2000 18:03:28
2       on-line   since 01/02/2000 18:03:28
3       on-line   since 01/02/2000 18:03:28
4       on-line   since 01/02/2000 18:03:28
5       on-line   since 01/02/2000 18:03:28
6       on-line   since 01/02/2000 18:03:28
7       on-line   since 01/02/2000 18:03:28
```

When the above sequence of commands is used to create, install, and allocate processors to two zones on each server, the result is that two of the processor's eight cores are bound to each zone. Logical processors 0, 1, 2, and 3 bind to core 0, and logical processors 4, 5, 6, and 7 bind to core 1, and so on.

Securing Containers and the Global Context

Sun IT's security standards dictate that every server must meet security requirements that are satisfied in part by running the Solaris Security Toolkit. The toolkit minimizes the operating system installation by removing non-essential software, and it hardens the operating system by restricting user rights, locking down file permissions, and reducing the number of ways by which the server can be accessed. Once the server is locked down, it can be accessed via the secure shell (`ssh`), and files can be copied to and from the server using `scp` and `sftp`.

The instructions provided in the Solaris Security Toolkit 4.2 Administration Guide were followed to harden the global zone and each of the containers. No special instructions were necessary for downloading or installing the latest version of the tool in the container environment. The operating system was hardened on the server itself, without the use of Solaris JumpStart™ software as a staging environment.

Installing Patches

Sun's recommended IT best practices include keeping operating system patch levels up to date. The Solaris Security Toolkit provides the capability to install the Recommended and Security patch cluster available from *http://www.sunsolve.sun.com*. This patch cluster includes both security patches and patches to correct problems discovered after a given release.

In the consolidation effort, the latest patch cluster available was already included with the installed OS. It was not necessary to use the Solaris Security Toolkit to install the latest patch cluster from Sun.

The Hardening Process

From a security standpoint, one of the objectives was to determine whether out-of-the-box scripts could be used for hardening the Sun Fire T2000 server. While it is possible to customize the hardening procedure implemented by the Solaris Security Toolkit, a straightforward approach was used in hardening the server. In this scenario, the `secure.driver` script was used to secure and harden the OS.

Similarly, getting a baseline understanding of out-of-the-box security provides a context for actions taken by the hardening scripts for both the global zone and the local zones. The baseline also sheds light on whether there are differences between using the scripts on a standard operating system installation and one in which zones are used.

The following command were used to audit and harden the global and local zones on the T2000 server:

```
# /opt/SUNWjass/bin/jass-execute -a secure.driver -o /var/tmp/jass.before
# /opt/SUNWjass/bin/jass-execute -d secure.driver -o /var/tmp/jass.during
```

The `-o` option was used to track and log the hardening activities, which helped in understanding whether the process on the Sun Fire T2000 server introduced any new problems or issues.

When running the Solaris Security Toolkit in the local zone (sparse-root or whole-root), the Solaris Security Toolkit was indeed zone-aware. When running the tool in each zone, checks that require direct access to either the kernel or the hardware itself were automatically bypassed. This is because Solaris Containers prohibit direct access to the kernel and hardware platform in a zoned environment. The following audit scripts were bypassed automatically by the script:

- Power Management Service
- IP Filtering
- NFS Privileged Ports
- RFC 1948 Sequence Number Generation
- Stack Execution
- nddconfig
- EEPROM Security Mode

After hardening activities were completed, testing confirmed the hardening process ran as expected in both the global and local zones. The server and all zones were rebooted to ensure all hardening changes went into effect.

Verifying the Hardening

Once the Sun Fire T2000 server's global and local zones were hardened, each zone was verified to ensure that the global and all local zones were hardened with the appropriate security profiles

Most of the issues encountered were around system-level access (which are locked down significantly by the `secure.driver` script.) After applying the script, remote access was available only through the console device. This problem was handled by updating the following template files provided as part of the Solaris Security Toolkit to allow access via the network:

- `/opt/SUNWjass/Files/etc/hosts.allow`
- `/opt/SUNWjass/Files/etc/hosts.deny`

Whenever possible, it's best to remedy issues like these by customizing the hardening scripts rather than manually working around the problem. This prevents the knowledge of the manual process from

disappearing when and if the process needs to be repeated for another installation. The Solaris Security Toolkit 4.2 Administration Guide provides more detail on the process of customizing hardening scripts and files to capture such events.

One important note is that, if a local zone is hardened before hardening the global zone, auditing within the local zone cannot be enabled until auditing is enabled within the global zone.

As a last step, to confirm that the server was hardened properly, the following command was executed:

```
# /opt/SUNWjass/bin/jass-execute -a secure.driver -o /var/tmp/jass.after
```

A noticeable improvement in the execution time of the Solaris Security Toolkit hardening scripts was observed, likely an improvement due to the performance of the UltraSPARC T1 processor. Hardening and audits were completed on average within five minutes.

Installing the Sun Store Application

With containers created and hardened, appropriate Java Enterprise System applications were installed followed by the Sun Store application.

Sun IT uses a variety of scripts to install the Sun Store software. The application uses Java servlet software to link the Web server presentation-tier components to the i2 business-tier components. These components are deployed in the application layer of the application server. The back-end components, namely BO Server, IO Server, i2 RMI registry, i2 Remote Session Server, and i2 Messaging Server, are all daemon process started during application initialization.

Property files capture the virtual IP addresses, port numbers, user names, and database connection information used by the business tier. The property files were customized for each of the containers. The Sun Store application depends on a number of external systems and data feeds to function correctly. The catalog loader is used to load the Extended Markup Language (XML) catalog file from the system of record into production database. The agreement loader, pricer loader and lead-time loaders perform functions implied by their names: the store interfaces with the authentication service for authentication, tax service for tax calculation, credit card service for credit card authorization, for example.

Once all of the components were deployed, and the database scripts run, the different components were executed using the start up process. Finally, automated tools tested the store's functionality. As of the date of this document these tests are still underway. The majority of the code changes are in the deployment scripts, not the store software, so the expectation is that modifications will be confined to the automated deployment process rather than any software that is specific to the Sun Fire T2000 server, the Solaris 10 operating system, or Solaris Containers.

Reducing Datacenter Space, Power, and Cooling Requirements

As datacenter temperatures continue to rise and drive energy costs ever higher, the combination of the Sun Fire T2000 server and the Solaris 10 Operating System are factors that can help to reverse the trend. Consolidating the Sun Store onto Sun Fire T2000 servers and the Solaris 10 OS is one of Sun IT's first steps in leveraging these technologies to reduce Sun's own datacenter space, power, and cooling expenses. This Sun BluePrints article has presented the current state of this consolidation effort. All indications are that the Sun Store will move from its current staging environment to production in 2006.

Some relatively straightforward calculations can give a rough idea for the power and cooling savings expected in the transition from the distributed Sun Enterprise 6500 and 10000 server environment to the Sun Fire T2000 server. The current distributed environment uses approximately the same resources of half of a fully loaded Sun Enterprise 10000 server. (38 400 MHz UltraSPARC II processors, or 10 boards.) The input power and heat output for a fully loaded server is 13,456 W and 52,000 BTU/hr, respectively (see "References" on page 19). This suggests that the current Sun Store business-tier environment consumes approximately 6728 W and produces approximately 26,000 BTU/hr of heat.

Consolidating onto Sun Fire T2000 servers can result in a significant reduction in both power consumption and heat generation. The Sun Fire T2000 server configuration used for the business tier consumes 344 W and produces 1174 BTU/hr at maximum power. Accounting for the fact that two servers were deployed to increase availability, a fair comparison requires doubling these figures to 688 W and 2358 BTU/hr. This results in a bottom-line reduction of 6040 W and 23642 BTU/hr over the current configuration. This reflects *an overall reduction of approximately 90 percent* in both input power and heat output.

A corresponding benefit in datacenter space requirements accompanies the power and cooling savings. A Sun Enterprise 10000 server occupies approximately 13 square feet of floor space and an equivalent height of 40 rack units. Each Sun Fire T2000 sever uses merely three square feet of floor space and only two rack units, resulting in substantial space savings.

Summary

The Sun Store uses a traditional three-tier Web services architecture, where the core components are based on i2 Technologies' Intelligent Selling System. Although the architecture depends on a large number of common services from Sun's IT infrastructure, we upgraded the entire software stack, from the Solaris 10 operating system to the i2 ISS solution, without disrupting the basic store architecture.

The business tier, containing most of the application software, is the largest consumer of CPU resources, and is thus the best candidate for consolidation onto the Sun Fire T2000 server. We designed an architecture to deploy two i2 ISS strings into two Solaris Containers on each of two servers, with two strings actively processing requests and two strings acting in standby mode. We compensated for the reduced number of strings by increasing the number of software instances running in each string.

Our use of Solaris Containers to host the Sun Store business-tier components proved to be straightforward. The Solaris OS resource management facilities allowed us to dedicate two UltraSPARC T1 processor cores to each container. The transition from physical servers to containers was easier than

expected: no Sun Fire T2000 server-specific changes were required to successfully implement and harden the Solaris 10 OS on the new platform. Correspondingly, the Solaris Security Toolkit proved to operate independently of the underlying hardware, and it worked in Solaris Containers, allowing us to use the toolkit out-of-the-box in this new hardware and software environment.

As the consolidation effort moves from the current phase into production, Sun IT plans to use the Sun Fire T2000 server and Solaris 10 Operating System consolidation as a model for other internal and external-facing IT systems. As an increasing number of IT operations are re-hosted to exploit the efficiencies of the UltraSPARC T1 processor with CoolThreads technology, we intend not only to keep our datacenter space, power, and cooling expenses in check, but to actually begin reducing them from today's levels.

About the Authors

The authors of this paper represent a cross section of the organizations responsible for the deployment and management of business solutions within Sun's internal infrastructure. Casey Costley is a member of the Sun IT Chief Technology Office and is responsible for the internal adoption of Sun's early access server technologies providing reference solutions and discovering and resolving issues before products are released to customers. Brad Coates and Srinivasa Bodicharla are both members of the Sun Services IT Operations organization, and are responsible for the configuration and maintenance of data centers. Yunas Nadiadi is a member of the Sun IT Strategy and Architecture group, and is responsible for defining and driving the IT enterprise architecture under which internal applications are run. Ragu Venkatesan is a member of the Sun Business Engagement and Applications group which is responsible for business application development and integration.

Acknowledgments

The authors would like to express their thanks to Steve Gaede and Eric Liefeld for their contributions to this Sun BluePrints article.

References

Information on the Sun Fire T2000 server is available at:

- `http://www.sun.com/products-n-solutions/hardware/docs/Servers/coolthreads/t2000/index.html`

Information on the Solaris Security Toolkit, formerly known as the JumpStart Architecture and Security Scripts (JASS) toolkit, can be found at:

- `http://www.sun.com/software/security/jass`

Current Recommended and Security patches can be downloaded from:

- `http://sunsolve.sun.com`

Power-consumption specifications used for the comparison presented in “Reducing Datacenter Space, Power, and Cooling Requirements” on page 18 can be found at the following locations:

- `http://sunsolve.sun.com/handbook_pub/Systems/E10000/spec.html`
  (Information on Sun Enterprise 10000 server power consumption)
- `http://ww.sun.com/servers/coolthreads/t2000/calc`
  (Calculator to estimate power consumption for specific Sun Fire T2000 server configurations)

Ordering Sun Documents

The SunDocs℠ program provides more than 250 manuals from Sun Microsystems, Inc. If you live in the United States, Canada, Europe, or Japan, you can purchase documentation sets or individual manuals through this program.

Accessing Sun Documentation Online

The `docs.sun.com` web site enables you to access Sun technical documentation online. You can browse the `docs.sun.com` archive or search for a specific book title or subject. The URL is `http://docs.sun.com/`

To reference Sun BluePrints OnLine articles, visit the Sun BluePrints OnLine Web site at: `http://www.sun.com/blueprints/online.html`