

Sun™ Cluster Software — Quality by Design for Advanced Availability

A White Paper
August 2004



© 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054 USA

All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California.

Sun, Sun Microsystems, the Sun logo, Java, Solaris, SunUP, Sun BluePrints, SunPlex, and Sun StorEdge are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a). DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS HELD TO BE LEGALLY INVALID.



Please
Recycle



Adobe PostScript

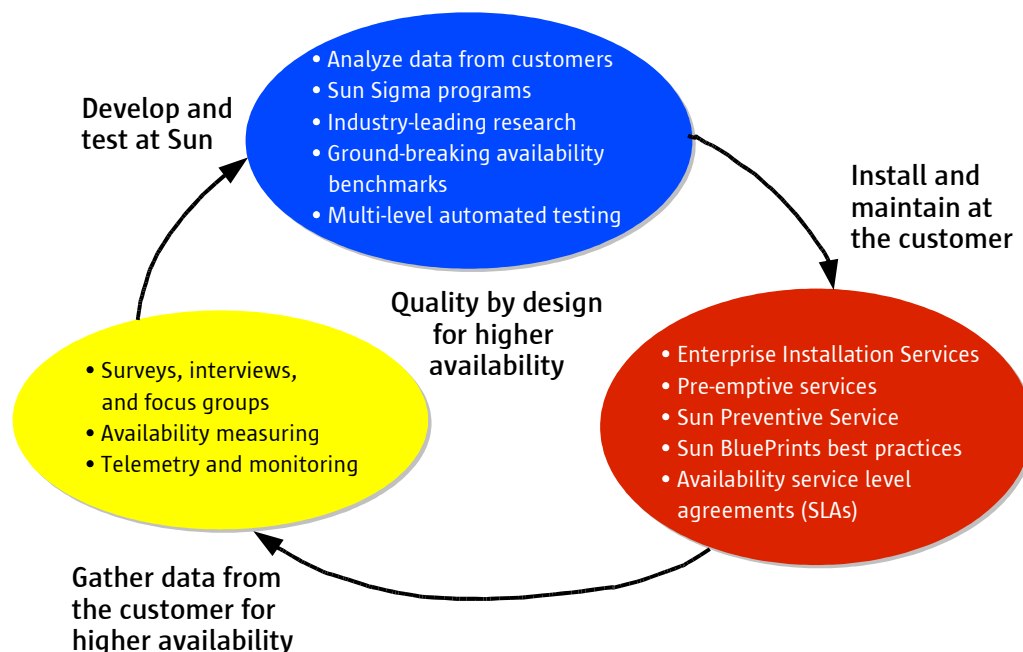
Introduction

Driven by the demands of a 24x7 global economy, optimizing the levels of availability for all types of application services is becoming a primary focus for many IT managers. In order to meet this increasing demand for continuous availability organizations need lower cost solutions with the predictability and high availability features of expensive, fault tolerant systems. It is well known in the industry that only 20 percent of system failures are actually due to product failures (this includes both hardware and software — people, product, and process are the three contributors to downtime), with the other 80 percent of failures resulting from process or human errors. Faced with the demand for continuously available application services that minimize the impact of these failures caused by product, people, and processes, customers are turning to clustering solutions. By tying together multiple servers, storage devices, and software, cluster solutions are able to tolerate the failure of any component in the environment. In order to help ensure the highest levels of availability, the cluster software must be designed, tested, and deployed with the highest level of quality and dependability in mind.

Quality, as defined in the realm of software and application availability, is the ability of a component to consistently perform in accordance to its functional specification, without deviation or unexpected failure. The challenge involved in evaluating a clustering software's quality comes with creating metrics to measure, manage, and monitor quality, especially in today's increasingly complex environments. As systems and environments become more complex and interdependent, the challenge of providing consistent high availability becomes more daunting. While the quality of individual systems is still important, it must now be viewed in the overall context of delivering consistent, high levels of application services. Sun understands that quality and its business implications are imperative to driving higher customer satisfaction. Sun is constantly striving to deliver high quality as an integral feature of the cost-effective, dependable Sun™ Cluster software by which customers can deploy their mission-critical applications with a high degree of confidence.

This paper details the investments Sun makes towards designing quality into Sun Cluster software from its initial development to when it is deployed at a customer site. It is intended for IT managers and decision makers who are evaluating high availability options for their data centers. The paper describes the various initiatives involved in delivering the high levels of quality necessary in a clustering product. These include company-wide Sun Sigma programs, industry-leading research, ground-breaking availability benchmarks, multi-level automated testing, aggressive availability service level agreements (SLAs), best practices for deploying and maintaining a clustering solution, and programs to help customers resolve issues and provide feedback to Sun. From start to finish, Sun executes on an evolutionary loop back cycle, illustrated in Figure 1, internalizing customer needs through the design process, together with innovative quality advances, to consistently provide higher quality availability solutions, giving Sun Cluster software a distinct competitive advantage over other clustering solutions. The paper also demonstrates how customers gain competitive advantage with higher levels of availability and reliability by deploying Sun Cluster software.

Figure 1. Sun's evolutionary design feedback loop, quality by design for higher availability



Sun Cluster 3 Software

Sun Cluster 3 software is the industry's premier availability platform for improving the predictability and resilience of business critical applications. Sun Cluster 3 software is designed to provide a highly available *service level management* platform that enables organizations to deploy applications as services and provide high availability, performance, and scalability of these services for the Solaris™ Operating System (Solaris OS) on both SPARC® and x86 platforms.

Sun Cluster 3 software is unique in that it is integrated with the Sun Java™ Enterprise System, which delivers Web and Application, Network Identity, Portal, Communications and Collaboration, Availability, and Security services. Sun Cluster software helps increase the availability of these network services by providing dependable application service level management. The Java Enterprise System offers a single, integrated, comprehensive software system with all of the critical enterprise network services organizations need to deliver service level agreements to their customers. All of the components in the Java Enterprise System are successfully tested together before each release, enhancing quality and helping to ensure that each of the components behaves as it should while interacting with the others.

Why the Improved Quality of Sun Cluster Software Delivers Higher Availability

High quality in each of the components of a service oriented architecture, including servers, operating system, storage, interconnects, and applications helps improve availability because the components themselves are less likely to fail or cause a failure. Higher quality leads to an increase in uptime, easier manageability, and decreased TCO. End-to-end solutions deployed on Sun Cluster software are tightly integrated and tested together with the tools discussed later in this paper for a simpler, more predictable solution.

Since Sun Cluster software is responsible for maintaining the availability of an application it must be of high quality itself in order to perform as expected. Continuously improving the quality of Sun Cluster software helps maximize the probability that the cluster performs the correct function when a failure occurs. It also helps ensure that the software works as it is designed to work without producing false alarms, which can potentially increase

system administrator tasks. In addition, higher quality software is more likely to have simpler manageability and lower cost of acquisition and deployment due to fewer problems in the deployment phase.

The quality initiatives, academic research, and innovative test tools discussed in this paper enable Sun to perform fine grained performance tuning, quality tuning, and testing. This allows Sun to offer very high quality Sun Cluster software deployments and SLAs that exhibit a high degree of predictability. The end result is that IT managers can experience an elevated degree of confidence that the solution they implement is highly available, enabling them to concentrate on the quality of their own processes and products and worry less about the risk of downtime.

Sun's Quality Initiatives

From the executive level down, Sun constantly endeavors to provide better products and services by executing a number of initiatives to design, test, track, and improve quality. This includes an on-going series of customer surveys, interviews, and focus groups to determine customer needs that are then assigned to business metrics, linked to critical quality factors, and analyzed for gaps that are addressed via engineering projects or analysis tools.

Recognizing that higher quality leads to higher availability, and therefore better service level attainment, Sun launched its own version of Six Sigma, called the *SunSM Sigma program*, in calendar year 2000. Six Sigma is an established quality-improvement program that is implemented at many major corporations, including General Electric and Motorola, to measure defects in a specific process and systematically eliminate them. In addition, Six Sigma emphasizes the ability to measure quality as a means to managing a specific service level. The Sun Sigma program is the foundation for Sun's methodology to achieve industry-leading availability and quality.

One of Sun's major quality initiatives focuses on availability with Sun Sigma program activities centered around the concepts of *specify and design it right, build it right, test it right, and deploy it right*. The goal of the initiative is to drive product teams to think about quality during each stage of development and deployment. Some of the activities include extensive pre-release testing and interoperability testing, reducing installation times, and translating customer availability requirements into engineering and service requirements. As part of this program, it is imperative to understand what availability *really* is so that it can be realistically measured, tested, and improved.

Understanding and Measuring Availability

As availability becomes an increasingly important metric for evaluating the quality of application services, there is a natural demand for independent, qualitative ways to measure availability, hence availability benchmarks. Benchmarks are critical in guiding research and development because they provide techniques to measure progress in both features and quality, to help perform trade-off analysis between designs, and to provide a mechanism for comparing the availability of one system to another. However, in order for a benchmark to be credible, it must accurately represent real world environments and the results must be reproducible.

Unfortunately, industry-standard availability benchmarks do not exist today primarily because it is extremely difficult to measure system availability in a repeatable fashion. This is due to the complexity of the overall system and of human error in the diagnosis and repair process. Many availability studies have been performed by academia and the research divisions of technology companies addressing various system robustness to fault events, but have thus far — to Sun's knowledge — failed to produce real-world fault rates, measure the duration of outages, or measure the effect of maintenance events on system availability.

After extensive research in the field of availability benchmarking, Sun engineers determined that a more appropriate and feasible alternative to measuring overall system availability is to measure the components that *comprise* availability. Demonstrating industry thought leadership, they developed a hierarchical approach, called

the R³ (R-cubed) framework, that decomposes the availability of a system into sub-metrics that can be measured with less difficulty than overall system availability. Taking availability measurements to the next level, the R³ framework stipulates a suite of single-attribute benchmarks, each of which measures one aspect of system availability. Combined, they provide a realistic, complete picture of system availability that is applicable to *any* system (total or subsystem), not just clusters.

A Complete Picture of the Main Components of Availability

The R³ framework divides system availability into three attributes — rate, robustness, and recovery — the three R's for which the framework is named. These metrics are measurable and portable, the results are repeatable, and thus they can be benchmarked. Each attribute can be broken down for more in-depth analysis and comparison.

- **Rate** — Fault and maintenance events are potential sources of system outage. Rate defines the *number of fault and maintenance events* that occur on a system over a given period of time. Without knowing the fault rate, system availability cannot be determined. Maintenance rate is the number of maintenance events that are essential to system operation. Maintenance events can be categorized as either fault induced or non-fault induced:
 - Fault induced events, such as replacing faulty hardware or installing a software patch that requires a subsequent reboot, are the direct result of a fault.
 - Non-fault induced events do not cause a fault but are essential to system uptime, such as a hardware configuration change or software version upgrade.

An internal study of Sun customers revealed that maintenance events accounted for approximately 79 percent of total outage events and 84 percent of total outage duration. The study clearly indicates that maintenance events are a critical factor in determining rate. The R³ framework includes an Outage Source Index (OSI) for measuring rate. There are currently no repeatable benchmarks for rate today. However, the OSI enables data to be gathered via controlled field data benchmarking.

- **Robustness** — Robustness is a system's *ability to detect and handle fault, maintenance, and system-external events*, and the resulting degree to which it continues to provide service in the face of these faults. There are three types of events a system must protect against. The first one is hardware and software faults. The second type is maintenance events, such as a hardware repair, software patch installation, upgrade, etc. Maintenance events can also result in system outage, and a system's robustness in the face of these events is a key differentiator among vendors. The third type is system-external events, such as human error and environmental events.
- **Recovery** — Recovery follows an outage and measures *how quickly a system is able to return to service*. At its highest level, recovery can be divided into automated recovery and manual recovery, which can be further decomposed into lower level metrics such as reboot time, etc. Automated recovery procedures are easily repeatable and measurable. On the other hand, manual recovery interjects a human factor, making the measurement less repeatable. At this time, manual recovery is measured through data collected in the field.

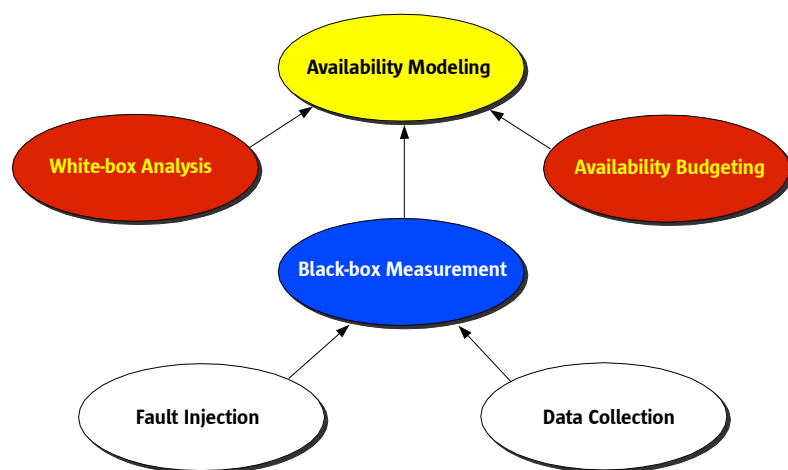
Sun has developed benchmarks for the R³ framework, shown in Figure 2, including the Maintenance Robustness Benchmark -A (MRB-A) for measuring the maintenance robustness of hardware, the System Recovery Benchmark -A (SRB-A) for measuring the recovery of single systems, and the System Recovery Benchmark -X (SRB-X) for measuring the recovery of clusters. To Sun's knowledge, the availability benchmarks developed with the R³ framework are the first of their kind to be published in the industry and used for evaluating commercial computer systems. The acceptance of these benchmarks by the Institution of Electrical and Electronic Engineers (IEEE) conferences represents a breakthrough in availability benchmarking work — they offer the first availability benchmarks that are repeatable and portable across systems manufactured by different vendors. Therefore, they have significant potential to become the first industry-standard availability benchmarks.

Figure 2. The R³ framework — availability innovation

In addition, the presence of third-party vendor layers in the stack can add further complexity. One way to approach this issue is to measure the availability characteristics of the third-party component, determine the typical outage imposed by that layer, and then assign an *availability budget* to it. An availability budget is the amount of outage time contributed by that layer and the number of times it can fail.

Using a combined approach of black-box measurement, white-box analysis, and availability budgeting, as illustrated in Figure 3, the process of modeling the availability of a specific cluster stack includes selecting a set of representative faults, analyzing the outages, investigating how the component times affect the outage, deciding which components need to be assigned budgets, gathering and analyzing customer data, deriving models for each of the parameters that contribute to system availability, and plugging these values into the RAScad model.

Figure 3. Availability modeling approach



Analyzing the availability model parameters reveals the main areas of focus for developing new functionality and quality testing for failures. In addition, this availability model is used by other software groups at Sun for building similar models for their stacks. From the availability model, Sun has developed general best practices rules for designing and maintaining a highly available Sun Cluster software environment.

In addition, Sun uses the R³ framework and the availability model to establish very stringent quality guidelines for Sun Cluster software releases. In other words, Sun not only uses academic work to improve availability, it also utilizes it in the process of testing the software for quality before releasing it. As a result, Sun improved failover time of certain types of loads and applications by 90 percent in a recent release of Sun Cluster software.

Perhaps most importantly, from a customer's perspective, the model clearly pinpoints outages that need to be avoided in order to maintain high availability in a system. The same best practice rules are applied to derive a set of best practices for minimizing the impact of people and process errors on availability.

Automated Testing for Higher Quality

Sun not only employs its ground-breaking academic theory and work to improve the quality of Sun Cluster software, but also leverages them to enhance the process of testing the software for quality on different configurations before releasing it. With a better understanding of the elements of availability gained by analyzing RAScad models, Sun is uniquely positioned to create and continuously improve an innovative automated test environment that is also undergoing several patent reviews. The Sun Cluster Automated Test Environment enables Sun to automate 95 percent of the testing for Sun Cluster software and permits the same level of extensive testing

for each subsequent release. This level of predictability leads to higher quality for Sun Cluster software environments, which in turn translates to higher availability and more sustainable service levels.

Sun Cluster Automated Test Environment

Sun Cluster Automated Test Environment (SCATE) is a suite of automated tools designed to verify Sun Cluster software with new hardware and software products from Sun and other vendors. SCATE includes an innovative state-of-the-art, automated, distributed test environment with comprehensive functional, load, and fault capabilities to verify end-to-end functionality of the entire cluster.

Fault injection technology provides total programmed control of granularity, with the ability to inject faults deterministically at precise locations in the Sun Cluster software source code. The preciseness and repeatability of the software generated faults helps validate the reliability and availability of Sun Cluster software. SCATE also includes an automated system fault generator for injecting system faults and provides an extensible framework for defining custom faults via fault plug-ins. The fault injection technology within SCATE is pivotal in advancing the overall quality of Sun Cluster software to make it truly robust, helping customers to deliver on their service level guarantees.

SCATE's automated test suites and tools are employed to qualify data services, server platforms, storage devices (including arrays, host bus adapters, switches, and hubs) and network interface cards (NICs) for public network and private interconnects. It is currently used by Sun Cluster software engineering to perform internal tests before releasing new versions and updates of Sun Cluster software. It is also used in the Open Storage program discussed later in this paper.

Sun also distributes the suite to a select group of IHVs (independent hardware vendors) to help them test and qualify their products in a Sun Cluster software environment. In addition, Sun Services has the ability to utilize SCATE at customer sites to identify problems with specific components or configurations.

SCATE is a client/server architecture with the ability to validate features of Sun Cluster software using sophisticated component level fault injection technology. It also includes system level load tests and tools to inject faults while the cluster is under load. Additionally, SCATE features tools to validate the integration of Sun Cluster software with other hardware and software components (both Sun and third party), providing customers with higher quality and more choice for their clustered environments.

Two types of fault injection — software faults (black box and white box) and hardware faults — are included to simulate real-life customer environments. Black box faults are essentially external faults, or faults that can occur in any real-world cluster. For example, power failures, software problems, cluster node reboots, etc. These faults are injected on clusters running real application workloads and the tests verify that the cluster recovers successfully from the fault and the services continue to stay up and run.

White box fault injection testing provides finer granularity with the ability to inject faults at precise locations in the Sun Cluster software source code. Fault points are injected in the source code of various Sun Cluster software components after carefully understanding the implementation and the component's interaction with other components in a clustered system. White box fault injection tests trigger these fault points and help ensure that the cluster software recovers successfully from the faults and the services are highly available. These tests are comprehensive and exercise code paths in the Sun Cluster software that cannot be exercised with black box fault injection alone. Black box fault injection, combined with white box fault injection, is very instrumental in improving the robustness of Sun Cluster software. Finally, hardware faults are primarily component failures, such as CPU, memory, cache, drives, network cards, etc. SCATE uses all three levels of fault injection to simulate and test real-world environments for more accurate results, and therefore, better quality.

Manual versus Automated Testing

Before Sun developed SCATE, testing of Sun Cluster software was performed manually. This form of testing only allowed a very limited number of faults to be tested per release, and due to its nature, was prone to faults, making

it less predictable (even with a low number of tests). With SCATE, Sun is now able to automate 95 percent of the tests performed on Sun Cluster software, enabling 45,000 to 50,000 faults to be tested with every release, and approximately 350,000 tests per release on different configurations, a feat that is obviously impossible in a manual testing environment. Now, due to SCATE, quality testing cycles are reduced from months to several weeks. This gives Sun the capability, as well as the confidence to schedule regular releases of high quality Sun Cluster software that is stringently tested with other hardware and software components, addressing customer needs for newer features and functionalities.

Automated tests are developed using the SCATE framework as new features are developed, not after. Tests for new functionality are developed along with the new code for each release and the new code is not allowed to be

implementations even before starting the Sun Cluster software qualification process. Without this tool, it is extremely difficult to troubleshoot and isolate problems with storage running in a Sun Cluster environment. This is a key quality enabler/tool for successfully testing the interoperability of storage products and Sun Cluster software.

SCATE is the de facto tool leveraged in Sun's Open Storage Program — a business program launched by the Sun Cluster group to build relationships with other hardware vendors. It is employed by third-party vendors to pre-qualify products before starting the Sun Cluster software qualification process, and because of its automated, repeatable functionality, even helps find problems that the vendors may not have been aware of before. As a result of the Open Storage Program, customers can benefit from end-to-end high availability for mission-critical applications on a diverse choice of storage platforms.

After successfully running the tool, the storage device is configured in a clustered environment and subjected to the complete SCATE test suites applicable for storage. For example, one vendor is using SCATE and the tool in its labs to help increase quality and to integrate well with Sun Cluster software. This is just another example of how SCATE is helping to successfully integrate a variety of storage products, thus providing customers with higher quality, more flexibility, and investment protection. In addition, these certified and tested configurations are cooperatively supported by Sun and the associated storage vendor.

High Quality Sun Cluster Software Agents

SCATE can also validate any data service, both scalable and failover. This helps certify HA and scalable agents that are developed at Sun, by third parties, and in the future by customers. Time to release new agents is dramatically reduced because the developer does not have to develop a suite of tests to test the functionality and quality of the agent. SCATE has helped to quickly qualify a wide range of HA applications with very little test development effort. It can also be used to test agents developed using the developer toolkit and SunPlex™ Agent Builder, helping to ensure that agents developed outside of Sun adhere to the rigorous quality practices followed by Sun Cluster software engineering.

The Future of SCATE

The drive to automate quality testing logically started with Sun Cluster software because it must be a high quality product in order to provide high availability. Inevitably, it is now trickling down through the rest of Sun. Other development areas are starting to leverage the automated test infrastructure to build test suites to raise the bar of the quality of additional Sun products.

Multiple Levels of Pre-Release Testing

As previously discussed, Sun Cluster software is released as part of the Java Enterprise System. One of Sun's quality initiatives expands testing for interoperability between the components in the Java Enterprise System, including Sun Cluster software. Pre-release product testing is enhanced with scenario-based end-to-end testing and adds another level of testing to the exhaustive testing performed with SCATE by the Sun Cluster software quality team. Hardware teams at Sun also qualify their new products with Sun Cluster software using SCATE to eliminate any interoperability issues with new hardware and Sun Cluster software, thus providing a third level of testing. In the end, the whole solution from Sun is of higher quality because it undergoes multiple levels of testing.

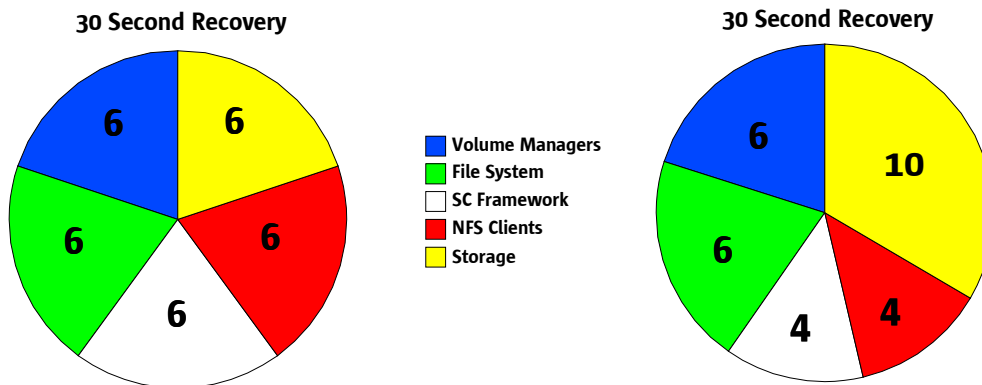
As a result of these extensive pre-release testing processes, Java Enterprise System components, including Sun Cluster software, are refined in numerous areas prior to release, translating into higher availability and reliability for customers. Sun is currently observing a trend that indicates that rigorous pre-release testing processes leads to fewer uncovered problems after the product is released, providing better performance and quality, and reducing the need for patches and related planned downtime. In addition, a Java Enterprise System tool called the Common Installer aids in identifying pre-release configuration issues in components.

Availability Budgeting for Accurate Service Level Agreements

Once a version of Sun Cluster software is available for release, the next challenge is solving a customer's requirement for high availability by budgeting the correct configuration of hardware and software for a particular application service level. Sun has the capability to use the same availability budgeting process from the design team to determine the right components to deliver a prescribed level of service by taking that requirement and allocating it among the significant sub-components of an implementation. For example, a customer might require that the system not be down for more than 20 seconds. The 20 seconds is apportioned equally among the components of the system and can be further divided between a component's sub-components. Each of the component teams then performs an analysis to determine whether the component can meet the budget requirement. Some components may be able to meet the requirement, while others may not because it is simply too aggressive. The goal in the end is for everyone to budget their components at the appropriate level, allowing the components of the solution to be joined to meet the objectives for the solutions.

For example, a customer seeking an SLA might require that their HA NFS service not be down for more than 30 seconds in the event of a failure. For this solution the main components are the Sun Cluster software framework, volume managers, file system, NFS clients, and storage. In this case the initial allocation is 6 seconds per component. A particular component, perhaps storage, comes back and reports that 6 seconds is too aggressive given the limitations of SCSI technology, and that it requires 10 seconds. In order to allocate 10 seconds to storage, 4 seconds must be appropriated from another component. As a result, the customer still sees the failover occur within 30 seconds, even though one of the components takes longer than the others to recover, as in Figure 5.

Figure 5. Availability budgeting for 30 second recovery



The budgeting exercise and this level of analyzing, understanding, verifying, and actually committing to a certain budget that a component can deliver enables Sun to offer service level agreements to its customers. Customers generally have different budget requirements, and depending on that budget, different components may need to be combined. The only way to successfully put components together is to know what budgets they are designed to support. Without availability budgeting, Sun would not have the confidence to state that a given component can recover in a certain amount of time.

For instance, one Sun customer requires a five nines (99.999) percent SLA, for a total downtime not to exceed five minutes per year. The SLA states that if the system is down for more than 25 seconds per month, Sun makes restitution to the customer. This then becomes the goal for the Sun Cluster software team. They know the exact stack of the configuration so they understand that the components consist of two Sun Cluster software components and third-party storage. Due to past availability budgeting exercises, they know that this type of storage typically requires 9 to 10 seconds and the Sun Cluster software components require approximately 9 to 10 seconds. Sun then contacts the third-party storage vendor and asks them to verify the 10 second recovery time by

running a series of SCATE tests and measuring the results, which subsequently verifies or substantiates the 10 second budget. This enables Sun to offer the customer an SLA with the understanding that it is based on the assumption that the third-party storage can meet this budget. Since Sun has the capability to measure each component, if a failure should go over budget, the offending party can be held responsible. In other words, if the storage goes over budget and forces the entire failover to exceed its monthly downtime, the third-party can be held responsible to the customer.

With the ability to budget the availability of a solution's components, Sun can offer SLAs that make the risk of downtime truly Sun's responsibility. Sun's continuous focus on quality enables it to write SLAs that allow customers to deploy Sun Cluster software and applications with a high degree of confidence and predictability, translating into high quality application services.

Best Practices for Deploying High Quality Sun Cluster Environments

Availability budgeting enables Sun to fine-tune configurations to best meet a customer's demanding availability requirements, but human error when installing systems can still impact the quality of the clustering solution. Using real-world customer data, Sun routinely notices an enormous rate of lack of availability when customers poorly install Sun Cluster software. These environments are referred to as vulnerable configurations. For example, one large financial customer with over 50 clusters was running down-revision patches on every cluster node, making every node vulnerable in that respect. Fortunately, this is something that can be easily detected and rectified with a rules database derived from Sun's extensive knowledge base that can be run immediately after installation and periodically after that to check for vulnerable configurations.

To successfully manage and leverage all of Sun's knowledge from across the company and around the globe to help customers deploy high quality Sun Cluster environments — including the best practice rules derived from the availability modeling discussed earlier — Sun developed and executes a knowledge management and delivery strategy based on a comprehensive knowledge architecture and the collective strength of highly skilled individuals, proven processes, and state-of-the-art technology. Sun Services aptly delivers a significant part of this strategy in the form of Enterprise Installation Services, pre-emptive service, and Sun Preventive Services.

- **Enterprise Installation Services** — One of the primary causes of system failure during the early stages of the product lifecycle is improperly installing and configuring the system. Using proven processes, methodologies, and best practices incorporated in the Enterprise Installation Services, Sun Service engineers or its authorized partners can correctly and quickly install Sun Cluster software environments to greatly increase the dependability of new deployments.
- **Pre-emptive service** — Pre-emptive service focuses on anticipating problems in installed environments and solving them before they impact service levels. Sun developed a set of eRAS applications that are designed to leverage existing knowledge to minimize the number of critical issues, thus maximizing availability and uptime. The eRAS program is constantly evolving in Sun's field organization from astute trend analysis that enables Sun to develop a system of checks that scan installed systems, searching for potential failures. For Sun Cluster software deployments, this includes the `SCcheck` command, which allows customers to check their configurations for cluster vulnerabilities.

In addition, using enhanced instrumentation within Sun Cluster software, system data is monitored and automatically collected using Sun's Configuration Service Tracker (CST) and sent to Sun via Sun Remote Services Net Connect (with the customers permission) where it is evaluated against a series of rules to identify potential problems. Once identified, they are matched against possible solutions, and a failure report summarizing the problems and potential solutions is generated. The account team can then work with the customer to resolve the issue. Internal studies suggest that customers that use the eRAS program for Sun Cluster environments, as well as other applications, are experiencing reduced high-priority calls and escalations.

- **Sun Preventive Services** — Sun Services also offers Sun Preventive Services. This service exposes and manages leading risk indicators with a blend of consulting, training, and support services that are delivered via an

iterative methodology of assessing, mitigating, measuring, and supporting a given environment. Sun and the customer work together to implement this methodology with a focus on preventing first and restoring second to help increase availability and decrease long-term support costs.

- **Sun BluePrints™ documents** — Finally, Sun recently published a set of Sun BluePrints documents that focus best practices on 10 of the most common mis-configurations of Sun Cluster software that typically lead to availability problems. The best practices detail processes for carefully planning the installation, installing the latest patches and firmware versions, performing an acceptance test before putting the system into production, periodically auditing the cluster components, promptly repairing failed components, periodically checking quorum devices and third party licenses, and recommendations on RAID and volume manager settings for cluster configurations. Following the proven processes in these Sun BluePrints documents can help customers deploy high quality Sun Cluster environments.

Working with Customers to Develop High Quality Availability Solutions

An important aspect of the Sun Sigma program is actually feeding measurements and the voice of the customer back into the loop to help ensure that customer requirements are met and to constantly develop high quality availability solutions. To meet this goal for Sun Cluster software, Sun works with customers in a program called the SunUP™ Network program. The SunUP Network program is a cooperative agreement with approximately 400 customers that work closely with Sun in order to provide feedback and concentrate on their cluster issues and requirements. It is a no-cost, sign-up program, where customers agree to share availability best practices amongst groups. Customers within similar industries, such as financial services, are separated into different groups, enabling them to share information while maintaining their competitive edge as it relates to highly available services. The SunUP Network program provides customers the opportunity to work closely with Sun to provide more data to enhance the quality and features of Sun Cluster software. Customers also benefit by exchanging their experiences successfully deploying data centers with high availability.

In addition, Sun Cluster software engineering utilizes an availability monitoring system to obtain data directly from hundreds of customers via the Configuration Service Tracker. Customers can also insert extra data to further explain a particular failure. Sun engineers post-process the data to examine the availability level for a system during a given time period. They can also view the data in groups to search for trends. This helps Sun understand how Sun Cluster software is performing, which applications behave well or mis-behave, and provides a direct feedback loop from the customer. For customers this can be extremely valuable because a snapshot of their system not only helps to pinpoint failures, but also illustrates exactly how many times Sun Cluster software behaves as expected to save the system from unexpected downtime. This can give customers confidence that they have a high quality, predictable availability solution that functions as it should.

Conclusion

It is clear that Sun's investment in the quality of Sun Cluster software can be a competitive differentiator for customers deploying high availability environments to support demanding service levels. Sun rigorously and thoroughly tests Sun Cluster software configurations before they are declared supported, resulting in high quality deployments for customers with a reduced possibility of errors. Sun builds this high level of quality by executing on its Sun Sigma program-based availability initiatives with a constantly evolving design loop that features:

- An on-going series of customer surveys, interviews, and focus groups — Listening to customers to determine their needs and requirements.
- Forward-thinking academic research — Decomposing availability into sub-components that can be measured with less difficulty than overall system availability, thus providing the R³ framework and a set of benchmarks that can be applied to any system and have the potential to become the first industry-standard availability benchmarks.

- Extensive modeling, analysis, and availability budgeting — Enabling Sun Cluster software engineers to minutely focus on areas to improve functionality and quality and implement customer requirements.
- Multiple levels of exhaustive testing — Providing outstanding quality and predictability first with Sun Cluster Automated Test Environment in the Sun Cluster software quality group, secondly with integration testing with Java Enterprise System components, and finally with other software and hardware groups within Sun, as well as third-parties. SCATE truly differentiates Sun Cluster software from other cluster products in the market, offering an added assurance to customers that Sun Cluster software is subjected to rigorous testing before it is released.
- Building appropriate configurations and SLAs — Using availability budgeting to assemble the correct components to meet stringent service levels.
- Best practices for higher quality installations — Leveraging company and customer-wide data, experience, and tools to make sure new deployments are installed quickly and correctly, eliminating common problems and human error for higher quality and availability from the start.
- Pre-emptive and preventive services — Working with the customer to maintain high availability environments with a goal of preventing problems before they happen with eRAS and Sun Preventive Services.
- A feedback loop from the customer — Gathering data to understand how Sun Cluster software works in real-world environments and working closely with customers in the SunUP Network program to gather data, facilitate knowledge exchange, solve problems, and improve quality.

This level of confidence in both the quality and predictability of its products allows Sun to offer aggressive Sun Sigma program SLAs for Sun Managed Services to its customers. Sun Sigma program SLAs enable customers to work with Sun to design, implement, and measure customized managed service programs with defined levels of quality and availability. And, using tools and methodologies such as eRAS and the SunUP Network program helps increase the consistency and dependability of measuring and managing these service levels.

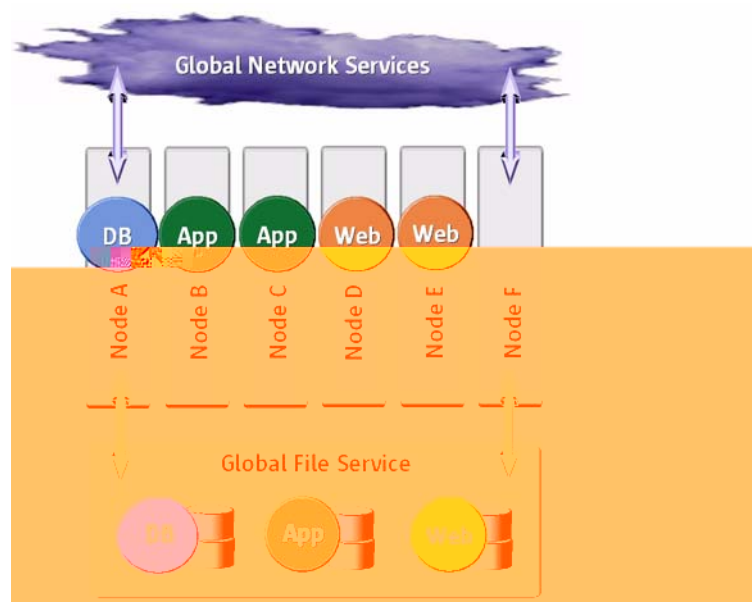
Because Sun focuses so extensively on the quality of Sun Cluster software in order to improve availability, the company believes that it can truly offer customers something other companies cannot — the confidence and ability to deliver higher service levels to their customers with a lower cost of ownership and cost of availability.

Appendix — Sun Cluster Software

Sun Cluster 3 software is designed to deliver improved service levels by providing continuous network, data, and service availability, making recovery from failure transparent to clients. Sun Cluster 3 software automatically restarts a failed resource — on the same node if it is healthy or on a backup node if it is not — without manual intervention, enabling more predictable application service levels, while simultaneously reducing risk and cost.

Sun Cluster 3 software also allows applications to be distributed across nodes in a cluster, providing easy manageability, application scalability, and automatic recovery of service levels, as illustrated in Figure 6. For example, using the Scalable Java System Web Server agent, system administrators can load-balance Web services across multiple nodes to achieve greater performance without sacrificing availability. With Sun Cluster Global Network Services, this load balancing is transparent to clients because the cluster appears to them as a single IP address. And, by adding more systems to the cluster, capacity and continuity can easily be increased.

Figure 6. Sun Cluster software environment



The Sun Cluster 3 software framework extends the operating system, enabling core operating system services — devices, file systems, and networks — to operate seamlessly with existing applications. In addition to extending services, key fault detection and recovery functions of the Sun Cluster software are integrated with the operating system, leading to faster failure detection and recovery, and therefore higher quality and availability.

Sun Cluster software environments can be purchased as part of Sun's Customer Ready Systems (CRS) program. The Sun CRS program delivers customer-specific, factory-integrated, and quality tested Sun Cluster environments, consisting of Sun servers, Sun StorEdge™ devices, Solaris OS, and Sun Cluster 3 software. Factory integrated and tested solutions can reduce early-life system issues by as much as 80 percent, and they help lower overall total cost of ownership. They also help to reduce the time and resources required for deployment and unexpected on-site assembly and integration issues.

References

For more information on topics discussed in this paper, please refer to the following sources.

Sun Web sites

- Sun Cluster software: www.sun.com/software/cluster/index.html
- Enterprise Installation Services for Sun Cluster software: www.sun.com/service/support/install/suncluster-software.html

Related Sun white papers

- *Robustness Benchmarking for Hardware Maintenance Events*, Ji Zhu, James Mauro, Ira Pramanick
- *A System Recovery Benchmark for Clusters*, Ira Pramanick, James Mauro, Ji Zhu
- *R³: Rate, Robustness, and Recovery — A Framework for Benchmarking System Availability*, Ji Zhu, James Mauro, Ira Pramanick
- *Modeling Sun Cluster Availability*, Ira Pramanick
- *Sun Microsystems, An Emerging Force in Knowledge Management*, Technical White Paper, Rita Gupta, Melissa Risteff

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 USA Phone 1-650-960-1300 or 1-800-555-9SUN Web sun.com



Sun Worldwide Sales Offices: Argentina +5411-4317-5600, Australia +61-2-9844-5000, Austria +43-1-60563-0, Belgium +32-2-704-8000, Brazil +55-11-5187-2100, Canada +905-477-6745, Chile +56-2-3724500, Colombia +571-629-2323, Commonwealth of Independent States +7-502-935-8411, Czech Republic +420-2-3300-9311, Denmark +45 4556 5000, Egypt +202-570-9442, Estonia +372-6-308-900, Finland +358-9-525-561, France +33-134-03-00-00, Germany +49-89-46008-0, Greece +30-1-618-8111, Hungary +36-1-489-8900, Iceland +354-563-3010, India-Bangalore +91-80-2298989/2295454; New Delhi +91-11-6106000; Mumbai +91-22-697-8111, Ireland +353-1-8055-666, Israel +972-9-9710500, Italy +39-02-641511, Japan +81-3-5717-5000, Kazakhstan +7-3272-466774, Korea +822-2193-5114, Latvia +371-750-3700, Lithuania +370-729-8468, Luxembourg +352-49 11 33 1, Malaysia +603-21161888, Mexico +52-5-258-6100, The Netherlands +00-31-33-45-15-000, New Zealand-Auckland +64-9-976-6800; Wellington +64-4-462-0780, Norway +47 23 36 96 00, People's Republic of China-Beijing +86-10-6803-5588; Chengdu +86-28-619-9333, Guangzhou +86-20-8755-5900; Shanghai +86-21-6466-1228; Hong Kong +852-2202-6688, Poland +48-22-8747800, Portugal +351-21-4134000, Russia +7-502-935-8411, Saudi Arabia +9661 273 4567, Singapore +65-6438-1888, Slovak Republic +421-2-4342-94-85, South Africa +27 11 256-6300, Spain +34-91-596-9900, Sweden +46-8-631-10-00, Switzerland-German 41-1-908-90-00; French 41-22-999-0444, Taiwan +886-2-8732-9933, Thailand +662-344-6888, Turkey +90-212-335-22-00, United Arab Emirates +9714-3366333, United Kingdom +44-1-276-20444, United States +1-800-555-9SUN or +1-650-960-1300, Venezuela +58-2-905-3800, or online at sun.com/store

SUN™ THE NETWORK IS THE COMPUTER © 2004 Sun Microsystems, Inc. All rights reserved. Sun, Sun Microsystems, the Sun logo, and The Network Is The Computer are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc. UNIX is a registered trademark in the United States and other countries, exclusively licensed through X/Open Company, Ltd. Other brand and product names are trademarks of their respective companies. Information subject to change without notice. Printed in USA 00/00 XX0000-0/#K