

Predictive Self-Healing in the Solaris™ 10 Operating System

A Technical Introduction
September 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, BigAdmin, Solaris, Sun Fire, and The Network is the Computer are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries.

UNIX is a registered trademark in the United States and other countries, exclusively licensed through X/Open Company, Ltd.

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.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

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

Abstract

Sun has developed a new architecture for building and deploying systems and services capable of Predictive Self-Healing. Self-healing technology enables Sun systems and services to maximize availability in the face of software and hardware faults. It facilitates a simpler and more effective end-to-end experience for system administrators, reducing cost of ownership.

The first self-healing features are available as part of the Solaris™ 10 Operating System (OS). This paper describes Sun's approach to building self-healing systems as well as the new technologies available in the Solaris 10 OS. Beta versions of the Solaris 10 OS can be downloaded through the Software Express program beginning with the July 2004 release. For more information, visit sun.com/solaris/10.

Solaris 10 Predictive Self-Healing Features

- Automatic monitoring and diagnosis of CPU, memory, and I/O subsystems
- Automatic offlining of faulty resources while the Solaris OS is running
- Administrator tools to view self-healing logs and results
- Standardized messaging for all self-healing diagnosis results
- Knowledge article Web site linking to online diagnosis messages

Solaris 10 Predictive Self-Healing Benefits

- Improved system and service availability through predictive diagnosis and isolation of faulty components
- Diagnosis of faulty components performed automatically, in some cases reducing analysis time from days to seconds
- Simplified administration model for managing self-healing activities, reducing cost of ownership
- Links to knowledge articles for learning more about problem impacts and repairs, updated in Internet time
- Scalable architecture that can be rapidly adapted to new problems and updated without requiring system downtime

Introduction

From their very inception, computers have been assigned tasks of critical real-world importance, and the desire to improve their ability to avoid or recover efficiently from faults has grown in proportion to user needs. In the intervening time, little has been done to change the intrinsic model for error handling and fault management in UNIX® software. The original UNIX design was not concerned with mechanisms for hardware or software fault handling; complex error handling routines and fault diagnosis capabilities would have compromised its simple elegance and portability. Even a basic concept like error logging, provided in UNIX by the syslog service developed in the 1980s, is little more than interprocess or networked *printf()* that has barely evolved since then.

In recent years, more advanced mechanisms and policies for fault diagnosis, response, and management have typically been implemented outside of the operating system in hardware or firmware, or implemented as a set of closed vendor and platform-specific extensions that do not generalize well. As a result, system administrators have to spend time reviewing log files filled with system-specific implementation artifacts in order to manually diagnose and repair problems. In addition, heterogeneous management software must either offer a lowest common denominator feature set or act as a cross-platform repository for system-specific implementation artifacts such as non-standard error messages.

Sun has designed a new lightweight, flexible architecture for building and deploying self-healing technology for hardware and software, and has implemented this architecture as well as a set of initial components in the Solaris 10 OS. The extensible software architecture and management tools provide a new model for how to build a self-healing system; yield simple, stable interactions for human administrators and management software; and cleanly separate implementation details from messages intended for human administrators and events passed along to higher-level management software.

The first set of technologies implement Predictive Self-Healing for CPU, memory, and I/O bus nexus components for UltraSPARC® systems, and are available as part of the July 2004 Software Express download. Future Software Express releases will include analogous capabilities for AMD Opteron processor-based x86 systems as well as self-healing features for other system components.

Sun has designed the self-healing architecture to permit rapid conversion of other hardware and software components in the future as well as adaptation of the technology to other parts of Sun's product line. Predictive Self-Healing will:

1. Simplify the task of composing, configuring, and deploying high-availability solutions and continuously measuring their availability
2. Maximize the availability of the system and services once deployed by automatically diagnosing, isolating, and recovering from faults and being predictive and proactive wherever possible
3. Guide system administrators through any tasks that require human intervention, including repairs, and explain problems detected or predicted in the system using clear, concise language and links to continuously updated repair procedures and documentation
4. Enhance the data-driven feedback loop between Sun and customers to ensure continuous improvement in product quality for both deployed and future products

With the ability to consolidate many services onto even a single blade, and the advent of technologies such as multiple CPU cores per die and chip-level multithreading, Sun believes that it is essential to build self-healing technology into systems from the lowest levels of the hardware/software stack upward. This design strategy facilitates fine-grained responses to failures, such as disabling an individual CPU core or device node or restarting an individual service, and ensures that the system can present a stable, understandable model of self-healing activities either to a human administrator or a layered management agent.

In the remainder of this document, you will learn about the attributes of the Predictive Self-Healing architecture for the Solaris OS, and the initial collection of features available through the Software Express program.

Telemetry and Diagnosis

The first major attribute of a self-healing system is that it is self-diagnosing; the system itself provides technology to automatically diagnose problems from observed symptoms, and the results of the diagnosis can then be used to trigger automated reactions. A fault or defect in software or hardware can be associated with a set of possible observed symptoms called errors; the data generated by the system as the result of observing an error is referred to as an *error report*. Historically, systems have exported error reports directly to human administrators and management software as a set of syslog messages, or have embedded mechanisms and policies for the diagnosis of the underlying problem directly into the code that is responsible for handling the error and generating the error report.

In a system capable of Predictive Self-Healing, error reports captured by the system are instead encoded as a set of name-value pairs described by an extensible protocol, forming an *error event*. Error events and other data that can be gathered to facilitate self-healing are dispatched to software components called *diagnosis engines*, which are designed to diagnose the underlying problems corresponding to these symptoms. Diagnosis engines run in the background, silently consuming telemetry until a diagnosis can be completed or a fault can be predicted. Once system components have been converted to properly handle errors and produce telemetric events, diagnosis software can be developed, improved, and deployed in parallel without inducing further system downtime by requiring operating system kernel patches.

After processing sufficient telemetry to reach a conclusion, a diagnosis engine produces another event, called a *fault event*, that is broadcast to any agents deployed on the system that know how to respond. A software component known as Solaris Fault Manager, `fmd(1M)`, manages the diagnosis engines and agents, provides a simplified programming model for these clients as well as common facilities such as event logging, and manages the multiplexing of events between producers and consumers. Tools are provided to view persistent logs of both error and fault telemetry and to permit developers and service technicians to correlate fault events back to the error report data that led to the diagnosis. This aids root-cause analysis and improvement of the diagnosis.

Reconfiguration and Messaging Agents

Once a diagnosis is complete or a fault is predicted, the self-healing architecture permits any number of subscribed agents to react to a particular type of problem. The Solaris 10 OS includes self-healing agents that can dynamically offline processors, regions of physical memory, and I/O devices. Through these reconfiguration agents, a self-healing Solaris system can take proactive and immediate action to isolate and disable a faulty component and continue providing service — even before human administrators know there is a problem. Solaris reconfiguration agents are integrated with other Solaris features such as Solaris Zones and Resource Management. They provide a consistent administrative experience and are transparent to application programmers.

Any number of additional agents may be deployed on a self-healing system to act in parallel with reconfiguration, including agents to provide local or remote messaging or other connections to higher-level management software. The Solaris 10 OS includes a syslog messaging agent for problems detected by self-healing diagnosis engines that utilizes a new standardized messaging format, described in the next section of this document. Sun also plans to deploy agents to permit system administrators to create customized responses to detected faults and defects, such as sending e-mail to a particular account, alias, or pager, and executing customized scripts.

Messaging and Knowledge Articles

The self-healing architecture also provides system administrators and service personnel with a more informative and effective end-to-end user experience, simplifying tasks that require human involvement and reducing the chances for human error. Every diagnosis result is associated with a stable, concise identifier code that can be read over the phone to Sun technical support or provided through remote services. The Sun message identifier (SUNW-MSG-ID) is also used to uniquely identify a corresponding knowledge article that provides more detailed information about the nature of the problem, the impact to the system and services, and the appropriate corrective action. Knowledge articles can be retrieved by appending the Sun message identifier to the Web link sun.com/msg.

Diagnosis results are also associated with a Universal Unique Identifier (UUID), permitting customers as well as Sun to build cross-indexed databases of actual problems experienced in production, the corresponding telemetry streams and diagnosis results, responses, and root-cause analyses. A standard agent is provided in the Solaris implementation to message faults using the existing syslog service. The messages briefly describe the impact of the fault, the automated response taken by the system, and the recommended repair procedure.

```
SUNW-MSG-ID: SUN4U-8000-AC, TYPE: Fault, VER: 1, SEVERITY: Major
EVENT-TIME: Tue Dec 28 17:17:42 PST 2004
PLATFORM: SUNW,Sun-Fire-V240, CSN: -, HOSTNAME: johnniac
SOURCE: cpumem-diagnosis, REV: 1.0
EVENT-ID: 16c1f33a-fa77-c0b6-9eff-938f0c25a02c
DESC: The number of errors associated with this CPU has exceeded
acceptable levels. Refer to http://sun.com/msg/SUN4U-8000-AC for
more information.
AUTO-RESPONSE: An attempt will be made to remove the affected CPU
from service.
IMPACT: Performance of this system may be affected.
REC-ACTION: Schedule a repair procedure to replace the affected CPU.
Use fmdump-v -u <EVENT_ID> to identify the CPU.
```

Figure 1. Example of a Fault Diagnosis Message

Figure 1 shows an example fault diagnosis message for a processor that has experienced a failure or for a processor experiencing a series of correctable errors that suggest a more serious failure may be imminent. The message is sent to the Solaris `syslogd(1M)` service, which can be configured for both local logging to a file as well as remote messaging to other hosts on the network, and is also printed to the system console. The SUNW-MSG-ID appears in the upper left-hand corner and can be used to access the corresponding knowledge article. The EVENT-ID field displays the UUID that globally and uniquely identifies this particular problem diagnosis.

By the time a human sees this message, the CPU will have been offlined automatically by another agent.

```
# psrinfo
0 faulted since 12/28/2004 17:17:50
1 on-line since 12/28/2004 16:33:00
```

Figure 2. Example of a Reconfiguration Result

Event Logging

Solaris Fault Manager provides a collection of common services for the self-healing diagnosis engines and agents that are deployed on the system, including persistent logging of all error event telemetry and diagnosis results. Separate logs are kept for error telemetry and diagnosis results so that different log rotation and persistence policies can be set for each using the Solaris `logadm(1M)` facility. The logs are maintained in a structured binary format, but are interchangeable and readable between Solaris SPARC® and x86-based platforms.

The self-healing infrastructure includes a two-phase commit algorithm for transferring telemetric events between the operating system kernel and Solaris Fault Manager. This ensures that events are never lost, even if a fault on the system affects Solaris Fault Manager itself. In addition, the Solaris kernel saves and replays any in-transit events across operating system failure or reboot. These design features ensure highly reliable delivery and storage for the self-healing telemetry flow and results.

The details of a particular diagnosis can be retrieved by applying the new `fmdump(1M)` utility and specifying the EVENT-ID (UUID) of a particular diagnosis. For example, to view additional detail for the diagnosis from Figure 1, an administrator might use the command shown in Figure 3.

```
# fmdump -v -u 16c1f33a-fa77-c0b6-9eff-938f0c25a02c
TIME                UUID                SUNW-MSG-ID
Dec 28 17:17:42.3075 16c1f33a-fa77-c0b6-9eff-938f0c25a02c SUN4U-8000-AC
100% fault.cpu.ultraSPARC-III.l2cachedata
FRU: legacy-hc:///component=MB
rsrc: cpu:///cpuid=0/serial=122CD203105
```

Figure 3. Examining the Fault Log

According to the log detail, the diagnosis system has predicted or determined that a particular processor has failed. The Field Replaceable Unit (FRU) corresponding to the faulty resource is shown, indicating that on this small system the “MB” (motherboard) component must be removed in order to repair the affected component. The FRU names shown are designed to match the labels used on the physical machine hardware to guide repair. On platforms that support them, LEDs on the physical machine are also used to indicate faulty components.

The `fmdump(1M)` utility can also be used by Sun service and repair technicians to view the raw telemetry that led to the diagnosis. Remote serviceability agents can also leverage the infrastructure to retrieve information vital to diagnosing the underlying root-cause of the failure, as well as permitting Sun to learn and improve on the diagnosis technology itself.

Service and Repair

Solaris Fault Manager associates diagnosis state with persistent identifiers corresponding to the system resources, such as hardware serial numbers. As a result, Solaris Fault Manager automatically updates this state after most repair actions without requiring any manual intervention.

Similarly, if a system with one or more faulty resources is reset, the reconfiguration agents for those faults will automatically run when the system is booting and automatically disable any faulty resources that are still configured into the system. On platforms that support Component Health Status (CHS), such as Sun Fire™ servers, faulty resources will be blacklisted and isolated out of use by the system before reboot.

Self-Healing Ecosystem

Sun has designed the Predictive Self-Healing architecture to be deployed across different types of systems, including service processors, and to scale and evolve rapidly as new diagnosis and availability technologies are added to the system. Over time, the Predictive Self-Healing initiative envisions a model wherein new systems and software components are delivered coincident with corresponding modules that are added to the ever-growing ecosystem of self-healing components.

Self-healing modules such as those shown in Figure 4 can be dynamically loaded and unloaded from the system while it is running, and can be upgraded on-the-fly while the system is running without requiring downtime or losing any of the active diagnosis state. Sun plans to use these features to facilitate continuous delivery of improved availability technology to self-healing systems.

\$ fmadm config	
MODULE	VERSION DESCRIPTION
cpumem-diagnosis	1.0 UltraSPARC-III CPU/Memory Diagnosis
cpumem-retire	1.0 CPU/Memory Retire Agent
fmd-self-diagnosis	1.0 Fault Manager Self-Diagnosis
syslog-msgs	1.0 Syslog Messaging Agent

Figure 4. Viewing the Self-Healing Modules

Summary

Predictive Self-Healing delivers the next generation of availability technology today, including features that keep systems and services running and simplify life for system administrators. Through the Software Express program, customers can download and take advantage of the new Solaris 10 technology immediately. Over time, a rapidly evolving ecosystem of self-healing components will ensure consistent, easy-to-use, and always-available Sun systems.

References

- Solaris Software Express — sun.com/software/solaris/10
- Predictive Self-Healing Knowledge Article Web — sun.com/msg
- Predictive Self-Healing BigAdminSM Forum — sun.com/bigadmin/content/selfheal

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 0 1252 420000, United States +1-800-555-9SUN or +1-650-960-1300, Venezuela +58-2-905-3800, or online at sun.com/store

SUN™ © 2004 Sun Microsystems, Inc. All rights reserved. Sun, Sun Microsystems, the Sun logo, BigAdmin, Solaris, Sun Fire, 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. Information subject to change without notice. 09/04 R1.0