



---

## Netra Proxy Cache Array Product Notes

---

A Sun Microsystems, Inc.  
Business  
901 San Antonio Road  
Palo Alto, , CA 94303-4900

Part No: 805-3513-10  
Revision A, January 1998

USA 650 960-1300 fax 650 969-9131



---

# Netra Proxy Cache Array Product Notes

Part No: 805-3513-10  
Revision A, January 1998

Copyright 1997 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California 94303-4900 U.S.A. All rights reserved.

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.

Portions of this product may be derived from the UNIX<sup>®</sup> system, licensed from Novell, Inc., and from the Berkeley 4.3 BSD system, licensed from the University of California. UNIX is a registered trademark in the United States and in other countries and is exclusively licensed by X/Open Company Ltd. Third-party software, including font technology in this product, is protected by copyright and licensed from Sun's suppliers. 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).

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

The OPEN LOOK<sup>®</sup> and Sun<sup>™</sup> Graphical User Interfaces were developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox Corporation in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a nonexclusive 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.

THIS PUBLICATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

Copyright 1997 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303-4900 U.S.A. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie et la décompilation. Aucune partie de ce produit ou de sa documentation associée ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a.

Des parties de ce produit pourront être dérivées du système UNIX<sup>®</sup> licencié par Novell, Inc. et du système Berkeley 4.3 BSD licencié par l'Université de Californie. UNIX est une marque enregistrée aux Etats-Unis et dans d'autres pays, et licenciée exclusivement par X/Open Company Ltd. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Sun, Sun Microsystems, le logo Sun, et Solaris sont des marques déposées ou enregistrées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC, utilisées sous licence, sont des marques déposées ou enregistrées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

Les utilisateurs d'interfaces graphiques OPEN LOOK<sup>®</sup> et Sun<sup>™</sup> ont été développés de Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox Corporation pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique, cette licence couvrant aussi les licenciés de Sun qui mettent en place les utilisateurs d'interfaces graphiques OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" SANS GARANTIE D'AUCUNE SORTE, NI EXPRESSE NI IMPLICITE, Y COMPRIS, ET SANS QUE CETTE LISTE NE SOIT LIMITATIVE, DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DES PRODUITS A REPENDRE A UNE UTILISATION PARTICULIERE OU LE FAIT QU'ILS NE SOIENT PAS CONTREFAISANTS DE PRODUITS DE TIERS.



# Contents

---

- 1. Netra Proxy Cache Array 1.0 Product Notes 1**
  - Product Components 1
  - Known Problems 2
  - Controlling Electromagnetic Interference (EMI) 3
  - Test and Load Objects 3
    - Test Objects 4
    - Load Objects 5
    - Relationships Among Objects 5



# Netra Proxy Cache Array 1.0 Product Notes

---

This document contains information not available at the time the manuals for the Netra<sup>™</sup> Proxy Cache Array were published. It contains a list of known problems with the Netra Proxy Cache Array and other useful information.

---

## Product Components

Version 1.0 of the Netra Proxy Cache Array product consists of the following:

- One or more Netra Proxy Cache 30 Servers, each with two internal 4.2 GB SCSI disks, a CD-ROM drive, and a diskette drive
- A recovery CD containing the Solaris<sup>™</sup> operating environment and the Netra Proxy Cache product packages
- An installation diskette, labeled “Netra Proxy Cache Utilities 1.0,” that enables you to perform initial host configuration without connecting a terminal to the Netra Proxy Cache Server
- A set of product documents:
  - *Netra Proxy Cache Quick Install Guide*
  - *Netra Proxy Cache Array Configuration Guide*
  - *Netra Proxy Cache Array User’s Manual*
  - *Netra Proxy Cache 30 Hardware Setup Instructions*

- *Netra Proxy Cache 30 Hardware Installation Guide*
- *Netra Proxy Cache 30 Hardware Reference Manual*
- *Solaris Handbook for SMCC Peripherals*

Note that the Netra Proxy Cache product does not have a video display or keyboard.

---

## Known Problems

We are aware of the following problems with the Netra Proxy Cache product at the time of publication:

- If you have a serial (TTY) connection to an unconfigured Netra Proxy Cache Server and monitor console output upon initial boot you see a message such as:

```
<date><time> NewNetra snmpXdmid: Registration with DMI failed. err = 831.
```

You might also see a message such as:

```
<date><time> NewNetra snmpdx: SNMP error (genErr(5), 1) sent back to localhost.<portnum>
```

Following configuration of your machine, you should not see these messages. The messages can safely be ignored as they do not reflect a problem with the server or the SNMP agent.

If you do see a message from `snmpd` or `snmpdx` following configuration, enter the following command to restart the SNMP master agent (`snmpdx`):

```
# /opt/SUNWoam/bin/snmpreset
```

- If the name of the last host in the table of parent and sibling proxy caches cannot be resolved at the time a Netra Proxy Cache Array host starts, that host will be unable to use other hosts in the array as siblings. Messages such as the following are generated every minute in the cache log file:

```
[<date><time>] neighbors.c:1617: membership_update: discovered new neighbor: SCALR:0x80827257  
[<date><time>] neighbors.c:1661: add_neighbor_scalr_ip_list: too many neighbors
```

**Workaround:** Rearrange the table of parents and siblings so that the host with the unresolvable name is not last. As with any configuration modification, you must use the Install Configuration link the Proxy Cache Administration page after you make the change.



- In the Proxy Cache Monitoring for Host page (described on page 15-10 of the *Netra Proxy Cache Array User's Manual*), the values displayed for SSL Connections, both total and current, are not reliable. The SSL Connections row is in the Proxy Cache Connection Statistics table in the Proxy Cache Monitoring for Host page.

The problem referred to in the preceding paragraph manifests itself in the SNMP variables `currentSSLConnections` and `totalSSLConnections`, in the group `proxyClientStatGroup`. If you use an SNMP-conformant management platform, such as Solstice Domain Manager, to retrieve these variables, you might obtain an inaccurate value.

---

## Controlling Electromagnetic Interference (EMI)

If you install or replace a CD-ROM drive or diskette drive in a Netra Proxy Cache 30 system, ensure that all peripheral power and data cables are properly routed through the clips adjacent to the hard disk drive bay provided for that purpose.

Route all data cables (SCSI and diskette drive) through both plastic spring clips installed adjacent to the hard disk drive bay.

Refer to drive installation documentation for proper cable routing.

If there are no drives installed in the upper drive bays, you should route only the SCSI cable into the upper drive bay. Attach the SCSI cable into the clip affixed on the rear wall of the upper drive's bay.

Unconnected peripheral power cables should remain clipped inside the main chassis.

---

## Test and Load Objects

Chapter 15 of the *Netra Proxy Cache Array User's Manual* omits the following information.

Test and load objects are pieces of software that run in the context of the Netra Proxy Cache array daemon, communicating the health of a service/host instantiation to the monitor object (`cache_monitor` or `dns_monitor`) in that daemon. The monitor object is responsible for monitoring a service on a given array host.

The format of the values returned by test and load objects are:

- From a test object, a monitor object expects a boolean value, indicating, for example, whether an interface is up or whether a service is available.

- From a load object, a monitor object expects two integers, one for current load, the other for current capacity.

The return values for test and load objects can be applied to a wide variety of resources. For example, a memory-intensive service might call for a load object to measure the availability of swap space.

In the current release of the Netra Proxy Cache product, all array members have the same set of test and load objects. These objects are selected for their appropriateness for a proxy cache service and an array DNS.

The array daemon configuration file, `scalrd.conf`, contains parameter settings for each test and load object. The file `scalrd.conf` is stored in `/etc/opt/SUNWscalr`. If you have a serial connection to an array host, you can use the `scalrcontrol (1)` utility, stored in `/opt/SUNWscalr/bin`, to obtain the output from the test and load objects.

There is a man page for each test object type, in `/opt/SUNWscalr/man/man5`. These man pages describe the parameters for each test object instance below. There is also a man page for `scalrcontrol`, in `/opt/SUNWscalr/man/man1`.

## Test Objects

The test objects listed below are shipped with the Netra Proxy Cache product. Their output is displayed in the Host Status page that you invoke from the Array Status page.

`cache_connect_test`

An object of type `ConnectTest (5)`. Tests the TCP port used by the proxy cache service (8080). Also tests the service address(es) and control address used by the proxy cache service. The test object instance is configured to test persistent TCP connections.

`cache_process_test`

An object of type `ProcessTest (5)`. Tests for the presence of the process associated with the proxy cache service.

`cache_test`

An object of type `AndTest (5)`. Combines the outputs from `cache_connect_test`, `cache_process_test`, and `service_interface_test`. Reports failure to the monitor object (`cache_monitor`) if any of these “child” test objects returns failure.

`control_interface_test`

An object of type `PingTest (5)`. Tests the integrity of the control interface.

`dns_connect_test`

An object of type `ConnectTest (5)`. Tests the TCP port used by the array DNS (53). Also tests the service address(es) and control address used by the DNS.

#### dns\_process\_test

An object of type `ProcessTest` (5). Tests for the presence of the process associated with the array DNS.

#### dns\_test

An object of type `AndTest` (5). Combines the outputs from `dns_connect_test`, `dns_udp_test`, `dns_process_test`, and `service_interface_test`. Reports failure to the monitor object (`dns_monitor`) if any of these “child” test objects returns failure.

#### dns\_udp\_test

An object of type `DNSTest` (5). Tests the ability of the array DNS to resolve the name of a domain. By default the name `localhost` is used.

#### service\_interface\_test

An object of type `PingTest` (5). Tests the integrity of the service interface used by a monitor object.

## Load Objects

The load objects listed below are shipped with the Netra Proxy Cache product. Their output is displayed in the Host Status page that you invoke from the Array Status page.

There is a man page for each load object type, in `/opt/SUNWscalr/man/man5`. These man pages describe the parameters for each load object instance below.

#### cache\_adjust\_load

An object of type `AdjustLoad` (5). Adjusts the output from the `cpu_load` object to account for special conditions, such as startup and shutdown.

#### cpu\_load

An object of type `CPULoad` (5). Returns the CPU utilization on a host.

## Relationships Among Objects

The relationships among monitor, test, and load objects are illustrated in Figure 1-1.

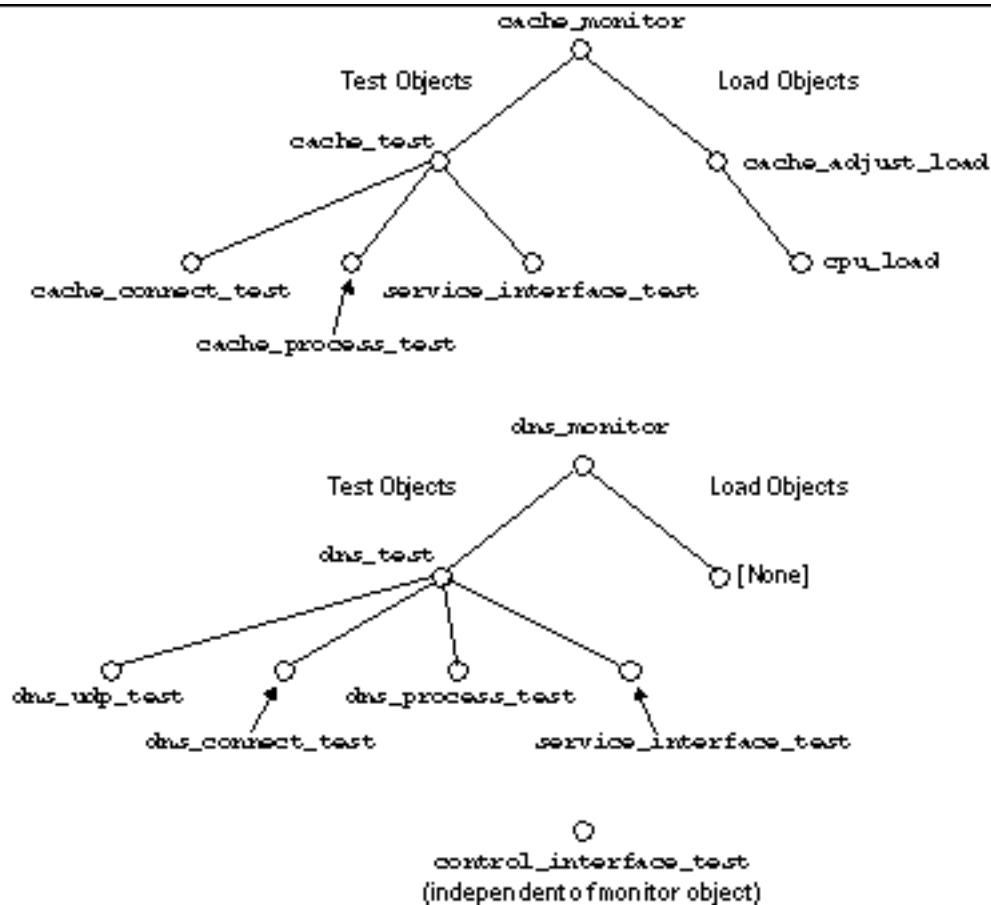


Figure 1-1 Relationships Among Objects

The significance of the relationships illustrated in Figure 1-1 is as follows:

- For test objects, a failure of a lowest-level object (indicated by a not-OK status in the Host Status page) causes the parent object (`cache_test` and `dns_test`, both of type `AndTest`) to fail. The failure of such a parent object, in turn, causes the monitor object return failure status. This failure is also reflected in the Host Status page. When a service on a host fails, the monitor object removes the service address associated with that service from the array's DNS zone and moves the service address to the least loaded host in the array.
- For load objects, the lowest-level object (`cpu_load`) returns its load and capacity figures to its parent (`cache_adjust_load`, of type `AdjustLoad`). Using our example, the `cache_adjust_load` object performs any adjustments required and returns "final" load and capacity figures to the monitor object, `cache_monitor`. The monitor object compares figures obtained from `cache_adjust_load` to

high- and low-water marks that it maintains for the service and takes action if one of these thresholds is crossed.

If a monitor object determines that a service is overloaded, it removes its service address from the array's DNS zone. If the monitor object determines that a formerly overloaded service is now in its normal range, it reintroduces the service address for that service in the DNS zone.