
Mac OS X Server Failover Messaging Architecture Guide

[Mac OS X Server](#) > [Networking](#)



2005-04-29



Apple Inc.
© 2005 Apple Computer, Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, Bonjour, FireWire, Mac, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

Xserve is a trademark of Apple Inc.

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

Introduction **About This Manual** 7

Conventions Used in This Manual 7
See Also 7

Chapter 1 **Concepts** 9

Failover Architecture 9
Failover Messages 11
 Configuration Command 12
 Configuration Reply 12
Notifications 13
 Configuration Changed Notification 14
 Service Status Changed Notification 14
Definitions 15

Document Revision History 17

Figures and Tables

Chapter 1

Concepts 9

Figure 1-1	Failover hardware scenario	9
Figure 1-2	Interaction of failover components	10
Table 1-1	Message commands	11
Table 1-2	Message attributes	12

About This Manual


This manual describes new failover procedures for AFP, NFS, and SMB in Mac OS X v10.4. The failover model consists of two nodes – a master node and a backup node – each running a daemon that monitors, announces, and synchronizes changes to file service settings. Third-party software can use the NSDistributedNotificationCenter to issue messages or to receive notifications of changes to configuration settings or of cluster events, such as refresh, failover, and terminate.

Conventions Used in This Manual

The Letter Gothic font is used to indicate text that you type or see displayed. This manual includes special text elements to highlight important or supplemental information:

Note: Text set off in this manner presents sidelights or interesting points of information.

Important: Text set off in this manner—with the word Important—presents important information or instructions.

 **Warning:** Text set off in this manner—with the word Warning—indicates potentially serious problems.

See Also

For information about NSDistributedNotificationCenter, see NSDistributedNotificationCenter.

INTRODUCTION

About This Manual

Concepts

Mac OS X v10.4 provides a new failover mechanism composed of two nodes – a master node and a backup node – forming a failover pair.

The following rules govern whether a node can be configured as a master or as a backup:

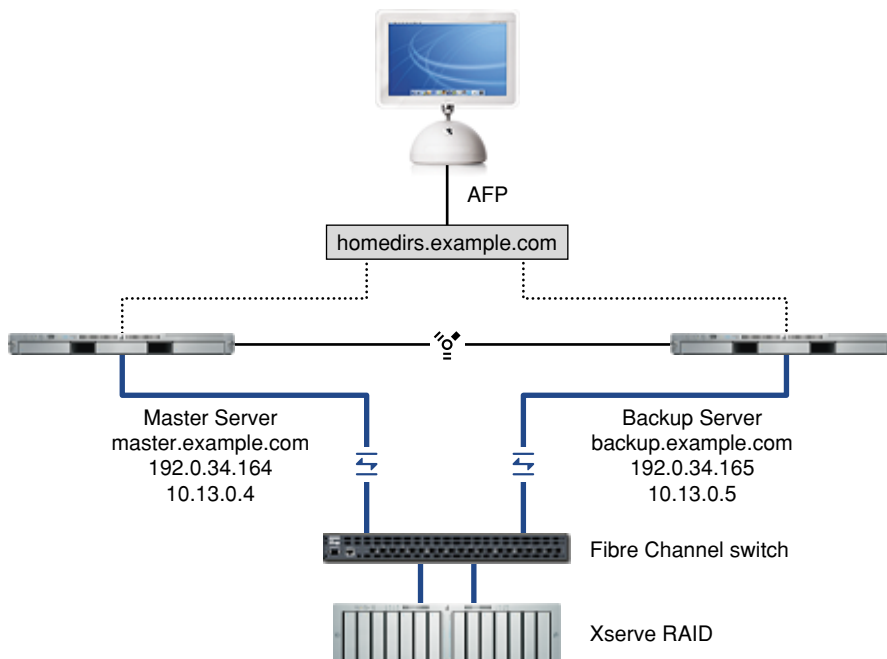
- The master and backup nodes must reside on the same private FireWire network.
- The master and backup nodes cannot run a conflicting service, such as being an Open Directory master or replica.
- Both the master and the backup node must be listed in an accessible DNS server.

Intra-host communication between processes running on the master and backup nodes is accomplished through the NSDistributedNotificationCenter. Third-party software can use the NSDistributedNotificationCenter to receive notification of changes such as a configuration change or a failover event. Inter-host communication is accomplished through a custom, proprietary protocol managed by the cluster daemon (`clusterd`).

Failover Architecture

Figure 1-1 (page 9) shows the typical hardware scenario in which the new failover mechanism is used.

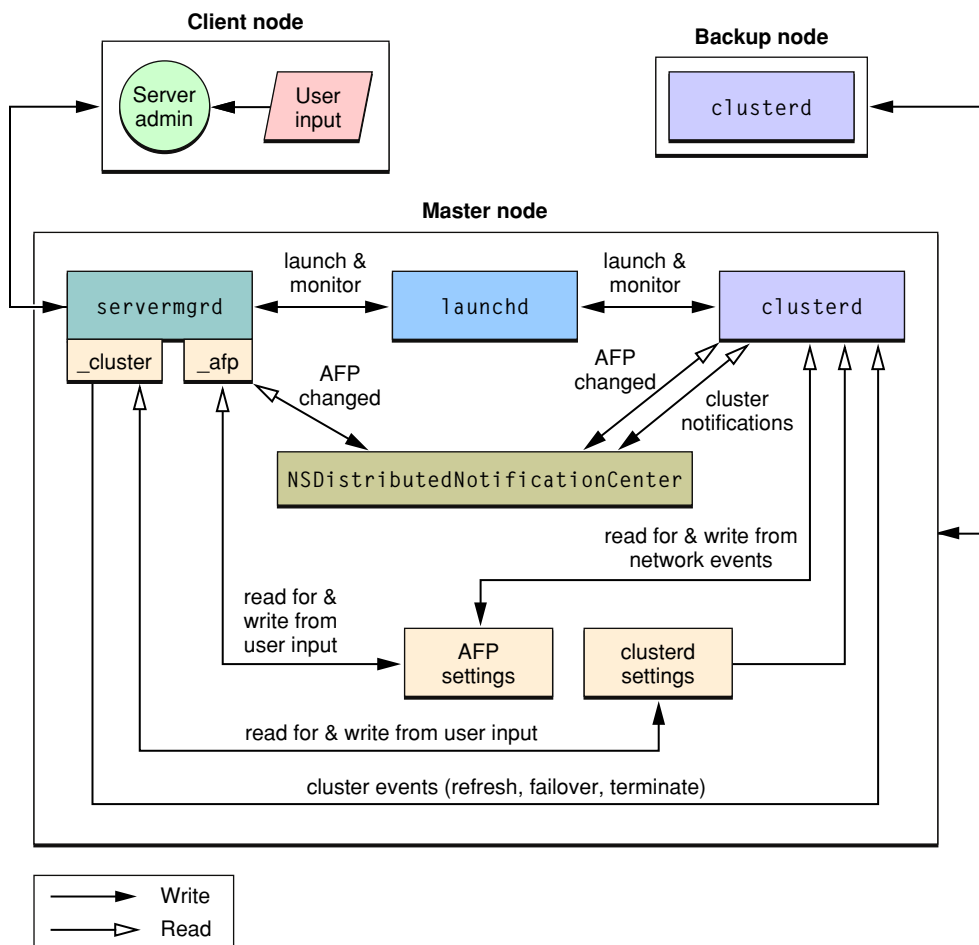
Figure 1-1 Failover hardware scenario



In Figure 1-1, FireWire connects two Xserve G5 cluster nodes and forms a private, FireWire network. Both Xserve G5 cluster nodes are connected to an Xserve RAID device through a Fibre Channel switch. Each node has a public name and IP address defined in DNS. In Figure 1-1, the public address 192.0.34.164 is assigned in DNS to master.example.com, and the public address 192.0.34.165 is assigned to backup.example.com. In addition to public addresses, private addresses (10.13.0.4 and 10.13.0.5, respectively), are also assigned to each node. The public address 192.0.34.166 is also assigned in DNS to homedirs.example.com but is not permanently assigned to a particular computer. This type of assignment is sometimes called a **virtual IP address**.

Figure 1-2 (page 10) shows the interaction of the failover components installed on the master and the backup node.

Figure 1-2 Interaction of failover components



On the master node, the `launchd` daemon starts and monitors the server manager daemon (`servermgrd`) and the cluster daemon (`clusterd`). Administrators provide configuration input to `servermgrd` through the Server Admin application.

The `servermgrd_clusterd` module of `servermgrd` reads and writes the `clusterd` settings and also sends cluster events, such as refresh, failover, and terminate, to the `clusterd` daemon.

The `servermgr_afp` module of `servermgrd` reads and writes the AFP settings. A change to the AFP settings causes a notification to be sent to the `NSDistributedNotificationCenter`. The `clusterd` daemon has registered with `NSDistributedNotificationCenter` to receive notification of AFP settings changes, so it gets the notification when an AFP setting changes.

The `clusterd` daemon on the master node communicates with the `clusterd` daemon on the backup node using IP over the dedicated, secure FireWire link.

For this release, the `servermgrd` daemon also has a `servermgr_nfs` module and a `servermgr_smb` module that perform the same functions as the `servermgr_afp` module.

The backup node's interactions are similar to the master node's. However, certain server administration operations, such as modifying file service settings, are not allowed.

Failover Messages

The master and backup nodes exchange messages that are actually plists. The messages involve the use of two node record types:

- **Public node record**— A public node record represents publicly accessible IPv4 or IPv6 address information obtained from a DNS server. A public node's IPv4 or IPv6 address is assigned to the master and assumed by the backup when the master fails. This node record or its IP addresses are virtual IP addresses.
- **Cluster node record** —A cluster node record represents one of the computers in the failover pair. One or more public node records are associated with the cluster node currently hosting it.

For this release, a single peer can be identified by querying the public node.

Table 1-1 Message commands

Message	Function
<code>configuration</code>	This message requests the list of public nodes the target is monitoring, as well as the target's list of private addresses. A response is expected. For more information on this command, see " Configuration Command " (page 12).
<code>failover</code>	This message is sent by a node that is giving up a public node to a backup node. In the case of a manual failover on the network shown in Figure 1-1 (page 9), the master (<code>master.example.com</code>) releases the monitored public node <code>homedirs.example.com</code> to the backup (<code>backup.example.com</code>). When the transition is complete, the master still has its own public node (<code>master.example.com</code>), but the other public node (<code>homedirs.example.com</code>) is hosted on the backup (<code>backup.example.com</code>).
<code>notification</code>	This message notifies the <code>clusterd</code> daemon of a significant event. No response is expected but this message may trigger other messages. The <code>clusterd</code> daemon processes but does not forward certain internal notifications, such as <code>heartbeat</code> notifications, which provide status and performance data to nodes that are listening.

Any of the commands listed in Table 1-1 can be accompanied by any of the standard attributes listed in Table 1-2.

Table 1-2 Message attributes

Message	Function
data (data or dictionary)	Packet content describing notification or requested data. This attribute appears in notifications and may appear in replies.
errorCodes (array of integer)	List of errors generated as a result of a corresponding request. This attribute only appears in replies.
version (integer)	Cluster plist format version number. This may appear in any message.

Configuration Command

The `configuration` command is the first command sent by the backup node. It queries the public nodes of interest to determine the current owner. Here is an example of the request `backup.example.com` would send to `homedirs.example.com`:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>command</key>
  <string>configuration</string>
  <key>version</key>
  <integer>1</integer>
</dict>
</plist>
```

Configuration Reply

The reply to a `configuration` command is a dictionary containing identifying information, including a UUID (valid for the life of the process), a list of private names and addresses, and the list of hosted public nodes. Here is an example of the response from `homedirs.example.com` (currently hosted on `master.example.com`) to the request from `backup.example.com`:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>command</key>
  <string>configuration</string>
  <key>data</key>
  <dict>
    <key>_id_</key>
    <string>E773FBDD-5CFE-4CF0-9F4C-10B6604064D7</string>
    <key>addresses</key>
    <array>
      <string>10.13.0.5</string>
    </array>
    <key>names</key>
  </dict>
</dict>
```

```

    <array>
      <string>master</string>
    </array>
  <key>publicNodes</key>
  <dict>
    <key>homedirs</key>
    <array>
      <string>192.0.34.166</string>
    </array>
    <key>master.example.com</key>
    <array>
      <string>192.0.34.164</string>
    </array>
  </dict>
</dict>
<key>errorCodes</key>
</array>
<key>version</key>
<integer>1</integer>
</dict>
</plist>

```

In the example above,

- E773FBDD-5CFE-4CF0-9F4C-10B6604064D7 is a UUID valid for the life of the `clusterd` process. It is used to quickly identify a node.
- The `addresses` key always specifies at least one valid address, which is the private IP address of the target that is being monitored, which, in this case is 10.13.0.5.
- The `names` key is guaranteed to contain at least one value, the node's Bonjour name. In this example, `master.local` is the Bonjour name that corresponds to 10.13.0.5.
- The `publicNodes` key is a dictionary instead of an array because each node can have just one DNS name and multiple IP addresses. In this example, `homedirs.example.com` is the DNS name of a node the target is monitoring.
- 192.0.34.166 is the IP address for `homedirs.example.com`.

Notifications

The server manager daemon (`servermgrd`) posts messages using the `NSDistributedNotificationCenter` mechanism. The cluster daemon (`clusterd`) registers for notifications posted by `servermgrd` and uses them to trigger synchronization actions. Third-party software can register to receive notification of messages. For additional information, see `NSDistributedNotificationCenter`.

Failover messaging uses two notification types:

- `com.apple.ServiceConfigurationChangedNotification`
- `com.apple.ServiceStatusChangedNotification`

Configuration Changed Notification

This notification is posted when a service's configuration changes. The related dictionary indicates the name of the service. For example, here is a configuration changed notification for AFP:

```
<dict>
  <key>ServiceName</key>
  <string>afp</string>
</dict>
```

For this release, the possible values for `ServiceName` are:

- afp
- nfs
- smb
- sharepoints

The `sharepoints` value is not actually a service but is used to notify interested parties of a change in the set of sharepoints.

Service Status Changed Notification

This notification is posted when a service is stopped or started. The related dictionary indicates the name of the service and its new state. For example, here is a service status changed notification for AFP:

```
<dict>
  <key>ServiceName</key>
  <string>afp</string>
  <key>State</key>
  <string>RUNNING</string>
</dict>
```

For this release, the possible values for `ServiceName` are:

- afp
- nfs
- smb

The possible values for `state` are:

- RUNNING
- STOPPED
- STARTED
- STOPPING
- UNKNOWN

Definitions

The header file `/usr/include/NSFailoverEvents.h` contains the following failover messaging definitions for use by third-party applications that want to receive notifications. It contains definitions for symbolic names for the notifications and keys and values for the dictionaries included in the notifications.

```
#define NSFailoverServiceStatusChanged
@"com.apple.ServiceStatusChangedNotification"
#define NSFailoverServiceConfigurationChanged
@"com.apple.ServiceConfigurationChangedNotification"
#define NSFailoverServiceNameKey      @"ServiceName"
#define NSFailoverServiceStateKey     @"State"

#define NSFailoverServiceStateRunning @"RUNNING"
// Any value other than NSFailoverServiceStateRunning means "not running".
#define NSFailoverServiceStateStopped @"STOPPED"
#define NSFailoverServiceStateStarting @"STARTING"
#define NSFailoverServiceStateStopping @"STOPPING"
#define NSFailoverServiceStateUnkonwn @"UNKNOWN"

#define NSFailoverServiceNameAFP      @"afp"
#define NSFailoverServiceNameSMB     @"smb"
#define NSFailoverServiceNameNFS     @"nfs"
#define NSFailoverServiceNameWeb     @"web"
#define NSFailoverServiceNameMail    @"mail"

// NSFailoverServiceNameSharepoints is a virtual service that reflects the
// set of file service sharepoints
#define NSFailoverServiceNameSharepoints@"sharepoints"

#define NSFailoverPathToScriptsDir    @"/Library/Failover"
#define NSFailoverPathToDefaultScriptsDir@"/Library/Failover/Default Scripts"
```


Document Revision History

This table describes the changes to *Mac OS X Server Failover Messaging Architecture Guide*.

Date	Notes
2005-04-29	New document that describes new failover procedures for AFP, NFS, and SMB in Mac OS X v10.4.

REVISION HISTORY

Document Revision History