

# NetCache<sup>®</sup> 6.0

## Deployment Guide

Network Appliance, Inc.  
495 East Java Drive  
Sunnyvale, CA 94089 USA  
Telephone: +1 (408) 822-6000  
Fax: +1 (408) 822-4501  
Support telephone: +1 (888) 4-NETAPP  
Information email: [doccomments@netapp.com](mailto:doccomments@netapp.com)  
Information Web: <http://www.netapp.com>

Part number 210-01009  
November 2004

# Copyright and trademark information

---

## Copyright information

Copyright © 1994–2004 Network Appliance, Inc. All rights reserved. Printed in the U.S.A.

Portions copyright © 1998–2001 The OpenSSL Project. All rights reserved.

No part of this book covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Portions of this product are derived from the Berkeley Net2 release and the 4.4-Lite-2 release, which are copyrighted and publicly distributed by The Regents of the University of California.

Copyright © 1980–1995 The Regents of the University of California. All rights reserved.

Portions of this product are derived from NetBSD, which is copyrighted by Carnegie Mellon University.

Copyright © 1994, 1995 Carnegie Mellon University. All rights reserved. Author Chris G. Demetriou.

Permission to use, copy, modify, and distribute this software and its documentation is hereby granted, provided that both the copyright notice and its permission notice appear in all copies of the software, derivative works or modified versions, and any portions thereof, and that both notices appear in supporting documentation.

CARNEGIE MELLON ALLOWS FREE USE OF THIS SOFTWARE IN ITS “AS IS” CONDITION. CARNEGIE MELLON DISCLAIMS ANY LIABILITY OF ANY KIND FOR ANY DAMAGES WHATSOEVER RESULTING FROM THE USE OF THIS SOFTWARE.

Software derived from copyrighted material of The Regents of the University of California and Carnegie Mellon University is subject to the following license and disclaimer:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notices, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions of the software were created by Netscape Communications Corp.

The contents of those portions are subject to the Netscape Public License Version 1.0 (the "License"); you may not use those portions except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/NPL/>.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Mozilla Communicator client code, released March 31, 1998.

The Initial Developer of the Original Code is Netscape Communications Corp. Portions created by Netscape are Copyright © 1998 Netscape Communications Corp. All rights reserved.

Software derived from copyrighted material of Network Appliance, Inc. is subject to the following license and disclaimer:

Network Appliance reserves the right to change any products described herein at any time, and without notice. Network Appliance assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by Network Appliance. The use and purchase of this product do not convey a license under any patent rights, trademark rights, or any other intellectual property rights of Network Appliance.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

## Trademark information

NetApp and the Network Appliance design are registered trademarks of Network Appliance, Inc. in the United States, Canada, and the European Union. Network Appliance is a registered trademark of Network Appliance, Inc. in Monaco and a trademark of Network Appliance, Inc. in the United States and Canada. FAServer is a registered trademark of Network Appliance, Inc. in the United States and the European Union. NetCache is a registered trademark of Network Appliance, Inc. in the European Union and Japan, and a trademark of Network Appliance, Inc. in the United States. SnapCopy is a registered trademark of Network Appliance, Inc. in the European Union and a trademark of Network Appliance, Inc. in the United States. WAFL is a registered trademark of Network Appliance, Inc. in the United States, the European Union, and Canada. FilerView, SecureShare, SnapManager, SnapMirror and SnapRestore are registered trademarks of Network Appliance, Inc. in the United States. Data ONTAP is a trademark of Network Appliance, Inc. in the United States and Canada. Snapshot is a trademark of Network Appliance, Inc. in the United States and the European Union. NetApp—the Network Appliance Company is a registered trademark of Network Appliance, Inc. in the United States and other countries. ApplianceWatch, BareMetal, Center-to-Edge, DataFabric, gFiler, MultiStore, NearStore, SecureAdmin, Serving Data by Design, Smart SAN, SnapCache, SnapDrive, SnapVault, vFiler, and Web Filer are trademarks of Network Appliance, Inc. in the United States.

Apple is a registered trademark and QuickTime is a trademark of Apple Computer, Inc. in the United States and/or other countries.

Microsoft is a registered trademark and Windows Media is a trademark of Microsoft Corporation in the United States and/or other countries.

RealAudio, RealNetworks, RealPlayer, RealSystem, RealText, and RealVideo are registered trademarks and RealMedia, RealProxy, and SureStream are trademarks of RealNetworks, Inc. in the United States and/or other countries.

Network Appliance is a licensee of the CompactFlash and CF Logo trademarks.

All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

Network Appliance NetCache is certified RealSystem compatible.

# Table of Contents

---

	<b>Preface</b> . . . . .	.xi
<b>Chapter 1</b>	<b>Getting Started with NetCache Deployment</b> . . . . .	1
	Overview of NetCache . . . . .	2
	Caching service described . . . . .	3
	Modes in which you can run a NetCache appliance . . . . .	6
	Additional NetCache features . . . . .	9
	Preplanning activities: define your goals . . . . .	12
	Preplanning activities: describe your environment . . . . .	13
	Questions to ask yourself about deployment . . . . .	14
<b>Chapter 2</b>	<b>Strategies for Client Access to NetCache</b> . . . . .	17
	<b>Section A: Summary of available strategies</b> . . . . .	18
	<b>Section B: Client access through transparent proxying</b> . . . . .	20
	Overview of transparent proxying . . . . .	21
	About transparent proxying with an L4 or L7 switch . . . . .	28
	Request distribution with an L4 or L7 switch . . . . .	30
	About transparent proxying with a WCCP router . . . . .	33
	Request distribution with a WCCP router . . . . .	39
	Transparent proxy deployment examples . . . . .	43
	<b>Section C: Direct (nontransparent) client access methods</b> . . . . .	51
	Summary of nontransparent client access methods. . . . .	52
	Pointing Web browsers to an automatic proxy configuration file . . . . .	54
	Pointing Web browsers to a single NetCache appliance . . . . .	58
	Methods for nontransparent access to a streaming media cache . . . . .	60
	<b>Section D: Request distribution for nontransparent client access methods</b> . . . . .	62
	Using DNS round robin for request distribution . . . . .	63
	Using a Server Load Balancer for request distribution. . . . .	65

	<b>Section E: Client access through global request routing . . . . .</b>	<b>68</b>
<b>Chapter 3</b>	<b>Multiple Cache Deployments . . . . .</b>	<b>69</b>
	<b>Section A: Request resolution hierarchies . . . . .</b>	<b>70</b>
	What is a hierarchy? . . . . .	71
	Hierarchy deployment examples . . . . .	75
	<b>Section B: Failover by using NetCache appliance takeover pairs . . . . .</b>	<b>78</b>
<b>Chapter 4</b>	<b>Deploying NetCache as a Web Cache . . . . .</b>	<b>83</b>
	About NetCache as a Web cache . . . . .	84
	Deployment considerations . . . . .	86
	Optimizing a Web site for caching . . . . .	88
	Scenario: NetCache deployed in an enterprise environment . . . . .	89
	Scenario: NetCache deployed at a global carrier and an ISP. . . . .	91
	Scenario: NetCache deployed with high-latency, high-bandwidth links . . . . .	95
<b>Chapter 5</b>	<b>Deploying NetCache as a Streaming Media Cache . . . . .</b>	<b>99</b>
	<b>Section A: Streaming media basics. . . . .</b>	<b>100</b>
	Overview of streaming media . . . . .	101
	Streaming media and bandwidth . . . . .	104
	Transmission methods for streaming media delivery . . . . .	106
	<b>Section B: Streaming media service with NetCache . . . . .</b>	<b>110</b>
	Overview of NetCache as a streaming media cache . . . . .	111
	Overview of NetCache support for live streams . . . . .	114
	NetCache support for live streams over unicast . . . . .	115
	NetCache support for live streams over multicast . . . . .	120
	NetCache support for on-demand streams . . . . .	123
	<b>Section C: Deployment considerations. . . . .</b>	<b>126</b>
	Considerations for bandwidth . . . . .	127
	Considerations for deploying NetCache multicast support. . . . .	130

Planning the number of streaming media caches needed. . . . .	132
Planning for client access to streaming media caches . . . . .	133
Planning for failover for streaming media service . . . . .	135
Considerations for firewalls and streaming media service . . . . .	137
Prefilling streaming media caches . . . . .	140
<b>Section D: Deployment scenarios. . . . .</b>	<b>141</b>
Scenario: ISP adding streaming media caches . . . . .	142
Scenario: Distributed streaming media caches . . . . .	146
Scenario: Deployment with a Windows Media server . . . . .	150
Scenario: Prefilling content of corporate NetCache appliances . . . . .	155
Scenario: Deploying multicast in an enterprise. . . . .	157
Scenario: Multicast support for transmission over a satellite link . . . . .	160
Scenario: Multicast support for a CDN. . . . .	163

## Chapter 6

<b>Deploying NetCache as an Accelerator . . . . .</b>	<b>167</b>
What is a NetCache accelerator? . . . . .	168
Strategies for client access to an accelerator . . . . .	172
Scenario: an accelerator outside the firewall . . . . .	174
Scenario: NetCache as a distributed Web site accelerator . . . . .	176
Scenario: multiple accelerators accelerating a single server . . . . .	179
Scenario: single accelerator accelerating multiple servers . . . . .	181
Scenario: allowing limited access from another company . . . . .	183
Scenario: accelerator for an historical stock performance Web site . . . . .	185

## Chapter 7

<b>Deploying NetCache as a News Cache . . . . .</b>	<b>187</b>
Introduction to NetCache news caching . . . . .	188
Interaction between the news cache and news server . . . . .	191
Software, clients, and features that NetCache supports . . . . .	193
About news data that is cached . . . . .	194
Deployment considerations . . . . .	195

	Scenario: news caches at an ISP Data Center and POPs . . . . .	199
<b>Chapter 8</b>	<b>Content Adaptation Services . . . . .</b>	<b>203</b>
	<b>Section A: NetCache support for ICAP . . . . .</b>	<b>204</b>
	Learning about ICAP . . . . .	205
	When ICAP services are invoked. . . . .	208
	<b>Section B: ICAP deployment considerations . . . . .</b>	<b>212</b>
	Deployment overview . . . . .	213
	Feature summary . . . . .	214
	Planning for ICAP services and ICAP servers . . . . .	216
	Security and ICAP . . . . .	219
	<b>Section C: ICAP service scenario . . . . .</b>	<b>221</b>
	Scenario: virus checking in an enterprise. . . . .	222
<b>Chapter 9</b>	<b>NetCache Deployment with Firewalls . . . . .</b>	<b>225</b>
	Deploying NetCache parallel to a firewall . . . . .	226
	Deploying NetCache inside a firewall . . . . .	227
	Deploying NetCache inside multiple firewalls . . . . .	230
	Relationship between the firewall and NetCache authentication. . . . .	232
	Scenario: access to a company Web server outside a firewall . . . . .	234
<b>Chapter 10</b>	<b>NetCache Routing Deployment Examples . . . . .</b>	<b>237</b>
	Distributing outgoing traffic over multiple links . . . . .	238
	Distributing incoming traffic over multiple links . . . . .	240
<b>Chapter 11</b>	<b>Global Request Manager . . . . .</b>	<b>243</b>
	About Global Request Manager . . . . .	244
	Request redirection with GRM . . . . .	248
	Scenario: telco movie delivery to home subscribers . . . . .	253
	Scenario: enterprise CDN spanning continents. . . . .	256

	Additional GRM features . . . . .	.258
<b>Chapter 12</b>	<b>NetCache Deployments in IPv6 Networks . . . . .</b>	<b>.259</b>
	Proxy cache deployments in IPv6 networks . . . . .	.260
	NetCache as an accelerator in v4/v6 client and v4 server networks . . . . .	.261
<b>Appendix A</b>	<b>Automatic Proxy Configuration File . . . . .</b>	<b>.263</b>
	Introduction to automatic proxy configuration . . . . .	.264
	Examples of automatic proxy configuration files . . . . .	.267
<b>Appendix B</b>	<b>Requirements for Transparent Proxying . . . . .</b>	<b>.271</b>
<b>Appendix C</b>	<b>OSI Model Relationship with Switches . . . . .</b>	<b>.275</b>
<b>Appendix D</b>	<b>Considerations When Pushing Content . . . . .</b>	<b>.277</b>
	Reasons for pushing content to NetCache appliances . . . . .	.278
	Content distribution and management using DataFabric Manager. . . . .	.280
	<b>Glossary . . . . .</b>	<b>.281</b>
	<b>Index . . . . .</b>	<b>.295</b>



# Preface

---

## **Purpose of this guide**

This guide provides information to help managers and evaluators understand the options available for deploying NetCache<sup>®</sup> appliances running NetCache 6.0 software in an existing network environment. The information in this guide pertains to all of the supported NetCache platforms: C1200, C2100, C3100, C6100, and C6200.

This guide is not a “cookbook” that tells you exactly how to deploy NetCache appliances. Networks are different and different system administrators have different goals for what they want to achieve. Instead, this guide provides many deployment examples to help you understand the available deployment options and clearly define what you want to achieve by adding NetCache appliances to your network.

## **Obtaining additional deployment assistance**

Network Appliance sales engineers provide assistance by helping you evaluate your requirements, determine how many and what type of NetCache appliances you need, and how to deploy your NetCache appliances.

You can also find information related to NetCache on the NOW Web site (<http://now.netapp.com/>).



**About this chapter** This chapter provides an overview of the NetCache product, including the protocols and clients NetCache supports and how you can use a NetCache appliance. This chapter also provides worksheets to help you describe your environment and specify your goals for deploying NetCache appliances.

**Chapter contents** This chapter contains the following sections:

- ◆ “[Overview of NetCache](#)” on page 2
- ◆ “[Caching service described](#)” on page 3
- ◆ “[Modes in which you can run a NetCache appliance](#)” on page 6
- ◆ “[Additional NetCache features](#)” on page 9
- ◆ “[Preplanning activities: define your goals](#)” on page 12
- ◆ “[Preplanning activities: describe your environment](#)” on page 13
- ◆ “[Questions to ask yourself about deployment](#)” on page 14

# Overview of NetCache

---

## Clients for which NetCache provides services

NetCache provides proxy and caching service to clients that support the use of proxy agents (such as Netscape Navigator and Microsoft® Internet Explorer). NetCache provides service to network clients, such as UNIX systems, Windows PCs, and Macintosh computers, that run Web browsers that support the following protocols:

- ◆ HTTP (Hypertext Transfer Protocol)
- ◆ Gopher
- ◆ Tunnel (any protocol)
- ◆ SSL (Secure Sockets Layer)
- ◆ NNTP (Network News Transport Protocol)
- ◆ FTP (File Transfer Protocol)

NetCache provides service to network clients that run media players that support the following protocols:

- ◆ MMS (Microsoft Media Streaming)
- ◆ RTSP (Real Time Streaming Protocol) for RealNetworks® RealSystem® and Apple® QuickTime™

## Protocols that NetCache supports

NetCache supports the following network protocols:

- ◆ HTTP, HTTPS, FTP over HTTP, FTP over TCP, Gopher, and Tunnel for Web requests
- ◆ NNTP for news requests
- ◆ MMS and RTSP for streaming media requests
- ◆ SNMP for network management
- ◆ Domain Name Service (DNS) for name resolution and DNS caching
- ◆ Routing Information Protocol (RIP) for determining which candidate default routers are alive
- ◆ Telnet for remote administration logins (recommended only during disaster recovery)
- ◆ Internet Content Adaptation Protocol (ICAP) for content adaptation services

# Caching service described

---

## Basic caching objective

A proxy-cache server (a NetCache appliance or third-party device) is deployed between clients and origin servers on the network and intercepts requests for content being sent from clients to origin servers. The basic objective of caching is to store content close to users. Benefits of caching include the following:

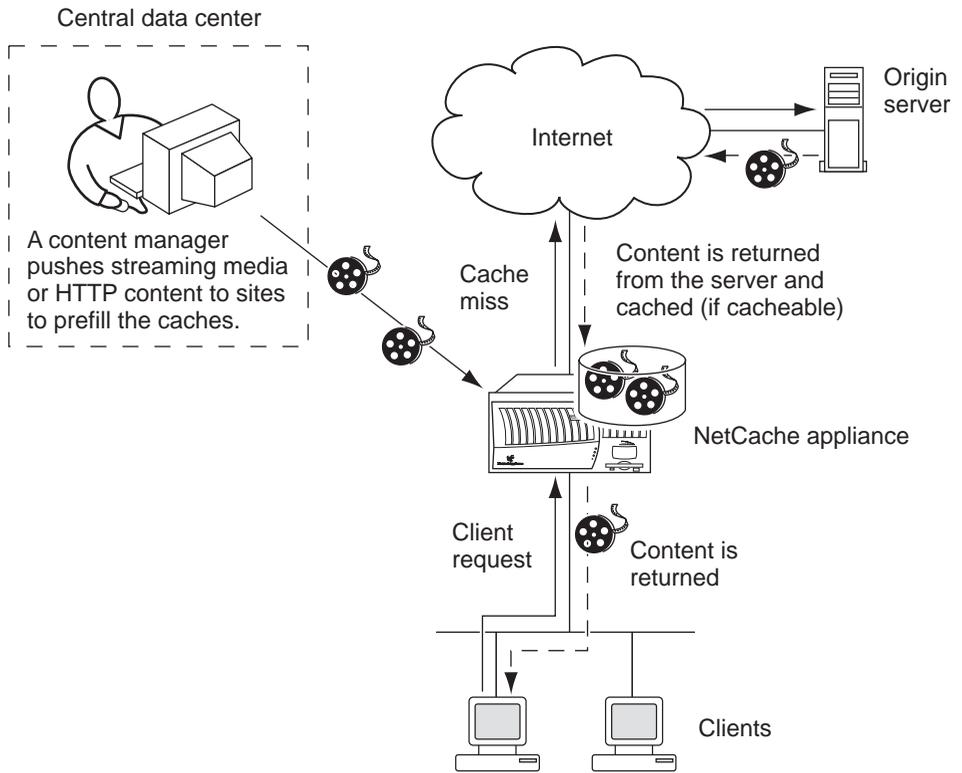
- ◆ You can provide content more quickly to your users than you can without caching service.
- ◆ You can save money because your Internet connection is used efficiently.
  - ❖ The physical distance that information must travel to reach end users is reduced.
  - ❖ Duplicate requests for the same content are satisfied from the proxy-cache server instead of being sent over the WAN, which decreases bandwidth costs and reduces the amount of traffic on your network.
- ◆ You can improve the quality of content delivered.
  - ❖ For streaming media, the more congested the network, the poorer the quality of the presentation. Caching streaming media content reduces the possibility of network congestion affecting streaming media quality.

## How a cache is populated

A cache, the area on the disk of a proxy-cache server that is used to store objects, can be populated in the following ways:

- ◆ As a result of user requests. This is the traditional caching service model.
- ◆ By prefilling caches, for example, with training videos, videos of corporate meetings, and Web pages.

The following illustration shows content retrieved from origin servers as a result of cache misses and content being pushed to a cache from a central management console.



The request flow, as shown in the previous illustration, is listed in the following table.

Stage	Description
1	A user enters a URL or requests streaming media through a media player.
2	NetCache checks its cache for the requested content.
3	<p>If the requested content <i>is not</i> in the cache, NetCache</p> <ul style="list-style-type: none"> <li>◆ Attempts to fetch the requested content from the origin server.</li> <li>◆ Caches the content (if the object is cacheable) while returning it to the client. This content is then available for the next client that requests it.</li> </ul>

Stage	Description
4	If the requested content <i>is</i> in the cache (because NetCache fetched it as a result of a previous request or the content was pushed to the cache), NetCache returns the content to the client immediately.

Organizations that prefill caches might also populate the cache as a result of user requests.

### About prefilling content in the cache

Providers of content are becoming increasingly interested in controlling content so they can provide the best possible experience for their end users. They want to *push* content from the center of the network to remote offices, closer to end users.

Pushing content to NetCache appliances in remote offices has quite a few advantages. For example:

- ◆ You can proactively distribute content at off-peak hours, which enables you to optimize bandwidth use and ensure that users enjoy fast response times.
- ◆ Multimedia, in particular, requires great amounts of network bandwidth, and network congestion greatly affects the quality of the content that is delivered. By pushing streaming media content close to end users, that content is less susceptible to quality degradation due to network congestion.

See Appendix D, “[Considerations When Pushing Content](#),” on page 277 for more information about pushing content.

# Modes in which you can run a NetCache appliance

---

## About this section

NetCache documentation describes NetCache appliances in terms of the modes in which they can run, as determined by the protocols that they are configured to run. This section relates terminology sometimes used in the industry regarding proxy-cache servers to NetCache operation modes and describes those modes.

## Forward and reverse proxies and NetCache modes

In the industry, proxy-cache servers are often identified as forward proxies and reverse proxies. Forward and reverse proxies are defined as follows:

- ◆ *Forward proxy (server)*

From the client's perspective, a forward proxy operates on behalf of the client Web browser or media player. A forward proxy can handle requests for a virtually unlimited number of origin servers. Forward proxies are located close to the client.

- ◆ *Reverse proxy (server)*

A reverse proxy (also referred to as an accelerator) handles requests on behalf of the origin server, acting as an extension of the origin server. A reverse proxy, unlike a forward proxy, services one or a few origin servers. Random servers cannot be accessed through a reverse proxy server. Clients use the reverse proxy to access all origin servers that the reverse proxy is servicing.

A reverse proxy is usually operated by the same organization that operates the origin servers that the reverse proxy services. A reverse proxy is located close to the origin server.

**Proxy-cache server type related to NetCache modes:** The following table shows the modes in which NetCache appliances can run and how those modes relate to forward and reverse proxies as they are described in the industry.

NetCache mode	Type of proxy-cache server
<ul style="list-style-type: none"><li>◆ Web cache</li><li>◆ Streaming media cache</li><li>◆ News cache</li></ul>	Forward proxy

NetCache mode	Type of proxy-cache server
<ul style="list-style-type: none"> <li>◆ Web server accelerator</li> <li>◆ Streaming media server accelerator</li> </ul>	Reverse proxy

**NetCache as a Web cache**

A NetCache appliance that handles any or all of HTTP, FTP, Gopher, and Tunnel (for example, HTTPS and SSL) requests is referred to as a *Web cache*. Requests of these protocol types that would ordinarily have been sent directly to a Web server are sent to the Web cache instead.

See Chapter 4, “[Deploying NetCache as a Web Cache](#),” on page 83 for more information.

**NetCache as a streaming media cache**

Network Appliance uses the term *streaming media cache* to describe a NetCache appliance that is configured to handle RTSP, MMS, or both types of streaming media. When NetCache is configured as a *streaming media cache*, streaming media requests (for example, audio and video) that would otherwise have been sent directly to a streaming media server are sent to the NetCache streaming media cache. A streaming media cache caches on-demand content (referred to video-on demand or VOD) and splits live media streams.

For live streaming media presentations, bandwidth savings are realized when multiple clients request the same unique stream. In this case, the streaming media cache makes a copy of a stream it is already delivering for each additional client that requests the same unique live media stream.

See Chapter 5, “[Deploying NetCache as a Streaming Media Cache](#),” on page 99 for more information.

**NetCache as an accelerator**

When NetCache is configured as a *Web accelerator* (a reverse proxy for Web requests), it caches content from one or more *Web servers* that you identify, and provides that content to clients that request it. The Web accelerator is, therefore, an extension of the Web server.

Likewise, when NetCache is configured as a *streaming accelerator*, it caches streaming media on-demand content and proxies live streams from one or more streaming media servers that you identify, and provides that content to clients that request it. The streaming accelerator is, therefore, an extension of the streaming server.

See Chapter 6, “[Deploying NetCache as an Accelerator](#),” on page 167 for more information.

### **NetCache as a news cache**

A NetCache appliance can be configured as a *news cache* to provide news caching service for news servers that support NNTP. A news cache requests news from a news server, on behalf of a client, delivers news content to clients, and caches news objects.

See Chapter 7, “[Deploying NetCache as a News Cache](#),” on page 187 for more information.

### **NetCache as a global request router**

The NetCache Global Request Manager (GRM) feature uses DNS routing to direct content requests from clients in a Content Delivery Network (CDN) to the NetCache appliances that are closest to the clients.

See Chapter 11, “[Global Request Manager](#),” on page 243 for more information.

### **Running more than one mode on a NetCache appliance**

For deployments with heavy traffic loads, Network Appliance recommends that you configure the NetCache appliance for a single service, such as streaming media or Web service. Deploying dedicated NetCache appliances improves hit rates and bandwidth savings. Smaller deployments with lighter traffic loads, such as branch offices in an enterprise, might be able to run multiple services on a single NetCache appliance without affecting performance.

## Additional NetCache features

---

### Features available with your NetCache appliance

NetCache offers a variety of features, which include those shown in the following table. The features are discussed in detail in the *Administration Guide*, the *Security Guide*, the *ICAP Services Guide*, and *Guide to Client Monitoring and Control for Streaming*.

Feature	Description
Access controls	A robust and flexible access control system is available, which enables you to create access control rules (ACLs) to allow or deny requests based on a number of variables, such as client IP address. ACLs are associated with specific users, groups of users, and types of requests, such as HTTP.
Filtering for inappropriate requests	NetCache includes two choices of content filter software for restricting, by category, access to Web sites that contain objectionable content. You can use SmartFilter software or WebWasher DynaBLocator software.
Authentication	NetCache provides support for authentication of users and groups through an internal database, LDAP, NTLM, Kerberos, or RADIUS. NetCache also supports interoperation with custom authentication systems that support cookies, for example, Netegrity SiteMinder®.
Logs	<p>A variety of logs are available for monitoring the activity of your NetCache appliance. Log features include customization of headers to be included in a specific log and the ability to push logs to an FTP server or Web server according to a schedule.</p> <p><b>Note</b> _____ The WebWasher ContentReporter software application fetches NetCache logs and prepares that data for use with third-party reporting applications.</p> _____

Feature	Description
Security	<p>Security features include the following:</p> <ul style="list-style-type: none"> <li>◆ Through the SecureAdmin™ software, you can administer NetCache in a nontrusted environment using an encrypted exchange of information between NetCache and a client.</li> <li>◆ You can restrict the locations from which NetCache accepts administrative traffic.</li> <li>◆ Vulnerability of a NetCache appliance to attacks by intruders is lower than with general operating systems, such as Solaris, because a NetCache appliance does not contain any other software on it (for example, Telnet or RSH) that would enable an intruder to access the network.</li> </ul>
Cache hierarchies	<p>You can logically define a hierarchy (pyramid) of NetCache appliances and third-party proxy-cache servers to enable an appliance to forward a request it cannot resolve to another hierarchy member for request resolution.</p>
ICAP	<p>A NetCache appliance can interact with one or more ICAP servers to adapt content to local policies. Deploying ICAP services enables you to off-load expensive services, for example, virus checking, content filtering, and advertisement insertion to another machine.</p> <p>See Chapter 8, “<a href="#">Content Adaptation Services</a>,” on page 203 for more information.</p>
Bandwidth allocation	<p>You can create rules that allocate various-size bandwidth “pipes” on your NetCache appliance and assign specific categories of connections to use these bandwidth pipes. Doing so enables you to restrict the share of your appliance’s total bandwidth capacity that can be used by any one category of connections.</p>

<b>Feature</b>	<b>Description</b>
e-Commerce	<p>NetCache e-Commerce technologies are based on a set of protocols on the appliance that enable the appliance to work in conjunction with application servers to monitor user streaming activity.</p> <p>Application server developers program their servers to use the NetCache APIs, which enables the servers to communicate with the appliance. Configuration on the appliance enables the appliance to communicate with application servers. See the <i>Guide to Client Monitoring and Control for Streaming</i> for more information.</p>

## Preplanning activities: define your goals

---

### List and prioritize your goals

When you are planning how to deploy a NetCache appliance, you need to understand what you want to achieve by adding a Web cache, news cache, streaming media cache, or accelerator to your network. The following table lists some of the reasons that organizations use a NetCache appliance. You can use the table to record your goals, if they are different, and prioritize them.

Goal	Priority
Reduce bandwidth usage.	
Improve the speed at which Web requests, news requests, and streaming media requests are resolved.	
Log requests.	
Block access to particular Web sites.	

## Preplanning activities: describe your environment

---

### Record information about your environment

Use the following table to record information about your environment. This information helps you when planning for NetCache deployment. The table includes some factors that you need to consider when deploying a NetCache appliance. Add other information in the blank cells, as needed.

Factors	Your information	How this information helps you plan
Number of users		You can determine how many NetCache appliances you need.
Location of users		If your users are geographically remote from each other, you might want to install NetCache appliances at both remote sites and local sites.
Type of work or activity		<ul style="list-style-type: none"><li>◆ If your users' work requires uninterrupted access to the Internet, you must ensure that you include a failover strategy in your deployment.</li><li>◆ If users have similar interests, the cache hit rate for your NetCache appliance is likely to be higher.</li></ul>
Expected demand, including peak hours		When determining the number of NetCache appliances you need, always consider the demand at peak load.

## Questions to ask yourself about deployment

---

### Deployment considerations

The following table includes some questions you should ask yourself when planning for NetCache deployment, and shows where you can find information to help you answer those questions.

Deployment consideration	Where to find information
How many NetCache appliances do you need?	Work with your NetCache system engineer.
What NetCache appliance model do you need?	Work with your NetCache system engineer.
Where do you locate the NetCache appliances?	Examples throughout this guide suggest where to locate NetCache appliances in relation to other network devices.
How do you want user requests to be directed to your NetCache appliance?	See Chapter 2, “ <a href="#">Strategies for Client Access to NetCache</a> ,” on page 17 for information about transparent proxying and nontransparent strategies for directing requests to your NetCache appliances.
Can you afford to have Internet access interrupted if a NetCache appliance goes down?	If your answer is no, see Chapter 2, “ <a href="#">Strategies for Client Access to NetCache</a> ,” on page 17, Chapter 3, “ <a href="#">Multiple Cache Deployments</a> ,” on page 69, and deployment considerations sections in chapters specific to NetCache operation mode.

<b>Deployment consideration</b>	<b>Where to find information</b>
How does NetCache work with a firewall?	If you have a firewall, see Chapter 9, “ <a href="#">NetCache Deployment with Firewalls</a> ,” on page 225. For a streaming media cache, also see “ <a href="#">Considerations for firewalls and streaming media service</a> ” on page 137.
Do you want to use NetCache as a Web or streaming media accelerator?	If your answer is yes, see Chapter 6, “ <a href="#">Deploying NetCache as an Accelerator</a> ,” on page 167.
Do you want NetCache to authenticate users before servicing their Web, news, and streaming media requests?	If your answer is yes, see the <i>Security Guide</i> for details about the controls that NetCache provides.
Do you want to restrict access to certain servers on the Internet?	If your answer is yes, see the <i>Security Guide</i> for details about the access controls that NetCache provides for authentication.
Do you want to log access to Web sites, streaming media sites, or news sites?	If your answer is yes, see the <i>Administration Guide</i> for details about the log files that NetCache provides.



**About this chapter** This chapter describes the different strategies for setting up how client requests reach NetCache appliances.

**Chapter contents** This chapter contains the following sections:

- ◆ Section A, “[Summary of available strategies](#),” on page 18
- ◆ Section B, “[Client access through transparent proxying](#),” on page 20
- ◆ Section C, “[Direct \(nontransparent\) client access methods](#),” on page 51
- ◆ Section D, “[Request distribution for nontransparent client access methods](#),” on page 62
- ◆ Section E, “[Client access through global request routing](#),” on page 68

## Section A: Summary of available strategies

### Possible strategies for client access to NetCache appliance

The following table summarizes the strategies discussed in this chapter. Each strategy in the table is discussed in detail later in this chapter. An important consideration when determining your strategy for client access to an appliance is whether you want users to have to configure their Web browsers and media players to access the appliance.

Strategy	Summary
<b>Transparent client access method</b>	
Transparent proxying	<ul style="list-style-type: none"> <li>◆ This strategy is available for DNS, HTTP, HTTPS, FTP, NNTP, MMS, and RTSP. (DNS caching can also be deployed nontransparently.)</li> <li>◆ Transparent proxying is the only strategy you can use that does not require users to configure their Web browsers or media players for client access to a NetCache appliance.</li> <li>◆ You add a Layer 4 (L4) switch, a Layer 7 (L7) switch, a policy-based router, or a WCCP 2.0-based router and configure your NetCache appliance for transparency.</li> <li>◆ Automatic failover to other NetCache appliances, third-party proxy-cache servers, and the Internet is standard.</li> <li>◆ This strategy has good request distribution capabilities. Request distribution is based on IP address or URL (L7 switch only), which is desirable because this method results in a higher hit rate than methods such as DNS round robin.</li> </ul>
<b>Nontransparent (direct) client access methods</b>	
Pointing client Web browsers to an automatic proxy configuration file	<ul style="list-style-type: none"> <li>◆ This strategy can be used for access to a Web cache only.</li> <li>◆ Users must configure their Web browsers to point to this file.</li> <li>◆ You do not need to add hardware to the network.</li> <li>◆ Automatic failover to other NetCache appliances, third-party proxy-cache servers, and the Internet is available. However, this strategy relies on the browser to detect failover. Not all Web browsers detect failover well.</li> <li>◆ This strategy has good request distribution capabilities. This file lists proxy cache-servers over which requests are to be distributed. Request distribution is based on IP address. Distribution is static.</li> </ul>

Strategy	Summary
Pointing client Web browsers to the NetCache appliance	<ul style="list-style-type: none"> <li>◆ Users must configure their Web browsers to point to a single NetCache appliance.</li> <li>◆ You do not need to add hardware to the network.</li> <li>◆ No failover is available. If the NetCache appliance goes down, news and Web access is not available because the NetCache appliance cannot fail over to another appliance.</li> <li>◆ No request distribution is available.</li> </ul>
Pointing a client media player to the NetCache appliance	<ul style="list-style-type: none"> <li>◆ Users of a RealNetworks® RTSP-based media player, QuickTime™ media player from Apple® Corporation, and Windows Media™ 7 or later player from Microsoft® can point the media player to a NetCache appliance.</li> <li>◆ For Windows Media players prior to WMP 7, you must use NetCache Windows metafile rewriting.</li> <li>◆ No failover is available.</li> <li>◆ No request distribution is available.</li> </ul>
Pointing client Web browsers to the NetCache appliance that is functioning as a DNS proxy cache	<ul style="list-style-type: none"> <li>◆ If DNS caching will be nontransparent, some client setup is required.</li> <li>◆ You do not need to add hardware to the network.</li> <li>◆ No failover is available.</li> <li>◆ No request distribution is available.</li> </ul>
<b>Request routing for CDNs</b>	
NetCache Global Request Manager (GRM)	<p>GRM directs content requests from clients in a Content Delivery Network (CDN) to the NetCache appliances that are closest to the clients. GRM supports two different redirection methods:</p> <ul style="list-style-type: none"> <li>◆ DNS-based redirection services</li> <li>◆ L7 redirection services</li> </ul>

## Section B: Client access through transparent proxying

---

**About this section** This section describes options for how you can provide client access to NetCache appliances by using a switch or router to transparently redirect specific types of traffic to the appliances.

**Contents of this section** This section contains the following topics:

- ◆ [“Overview of transparent proxying”](#) on page 21
- ◆ [“About transparent proxying with an L4 or L7 switch”](#) on page 28
- ◆ [“Request distribution with an L4 or L7 switch”](#) on page 30
- ◆ [“About transparent proxying with a WCCP router”](#) on page 33
- ◆ [“Request distribution with a WCCP router”](#) on page 39
- ◆ [“Transparent proxy deployment examples”](#) on page 43

## Overview of transparent proxying

---

### Advantages of transparent proxying for client access

When *transparent proxying* is deployed, the client software is unaware of the existence of a proxy-cache server. The addition of the caching service is transparent to end users because you perform all the configuration necessary to deploy transparent proxying—configuration of your NetCache appliances and the network devices with which you are deploying transparent proxying. Users do not have to configure client applications, such as Web browsers or media players, to use a particular NetCache appliance, as is the case with a nontransparent client access method.

Transparent proxying eliminates the possibility of users reconfiguring their Web browsers or media players (RealNetworks RTSP and Microsoft WMP 7 only) to bypass a NetCache appliance without the knowledge of NetCache administrators.

---

#### Note

The RTSP for RealNetworks media player, Apple QuickTime player, and Microsoft Windows Media player 7 and later can be used with transparent proxying.

---

Additional benefits of transparent proxying include the following:

- ◆ Efficient distribution of requests across available NetCache appliances
- ◆ Failover to another NetCache appliance or, if all NetCache appliances fail, to the origin server

### Protocols that NetCache can handle transparently

The following traffic can be transparently redirected to NetCache appliances:

- ◆ HTTP
- ◆ HTTPS
- ◆ FTP
- ◆ NNTP
- ◆ MMS
- ◆ RTSP
- ◆ DNS

You can implement DNS caching transparently or nontransparently. Users do not need to configure their browsers for nontransparent DNS caching. See the *Administration Guide* for details about setting up transparent and nontransparent DNS caching.

FTP over HTTP, Gopher, and tunnel traffic (for example, HTTPS and SNEWS) cannot be serviced transparently. For these protocols, you can set up proxy service by having users configure their browsers to point to the NetCache appliance or use the NetCache protocol tunneling feature.

### **Devices with which you can deploy transparent proxying**

You can use any of the following network devices to deploy transparent proxying for DNS, HTTP, FTP, NNTP, MMS, and RTSP traffic:

- ◆ L4 or L7 switch
  - Switches typically include multiple methods for handling traffic. When this guide refers to L4 or L7 switches, it refers to a feature on a switch that can support transparent proxying; that is, a feature that operates on the L4 or L7 layer of the Open Systems Interconnection (OSI) Reference model. Switch names vary by vendor.
- ◆ WCCP 2.0-based router (hereafter referred to as a WCCP router)
  - NetCache supports the use of Web Cache Communication Protocol (WCCP) 2.0 for redirecting traffic to NetCache appliances. WCCP defines the protocol that is used between proxy-cache servers (such as NetCache appliances) and routers to exchange information. WCCP 2.0 is supported on Cisco routers running IOS 12.0(4) T images and later versions. See the Cisco Web site for more information about IOS versions.
- ◆ Policy-based router (Cisco or non-Cisco)
  - Few organizations use this method for transparent proxying because of significant limitations, as follows:
    - ❖ Complex filters can have a dramatic negative impact on the performance of the router.
    - ❖ The system administrator must manually set up how requests will be distributed, which results in less efficient partitioning of requests than if a switch or WCCP router were used, and might impact NetCache performance.
    - ❖ No failover is available if a policy-based router goes down.
    - ❖ Routers might not be able to perform policy routing at high speeds.

Subsequent sections about transparent proxying discuss deploying transparent proxying with an L4 switch, an L7 switch, and a WCCP router. If you need additional information about deploying transparency with a policy-based router, contact your NetCache sales engineer.

## Drawbacks of a transparent proxy deployment

When determining your client access strategy, you must consider the drawbacks of implementing a transparent proxy deployment as well as the benefits. The following list shows the drawbacks of transparent proxy deployment. These drawbacks apply to transparency provided by any vendor.

- ◆ Network *routing flaps* might prevent the NetCache appliance from servicing requests.

A stable route is essential between the clients and the NetCache appliance. If a routing flap occurs, a client route might not continue to go through the NetCache appliance. If routing flaps occur frequently, you cannot take full advantage of your NetCache appliance.

Network Appliance recommends that you deploy transparent proxying only in locations where network routing cannot become asymmetric through the NetCache appliance.

- ◆ Some Web browsers might not refresh Web pages correctly when the user clicks the Reload button.

With transparent proxying, the browser is not aware that it is using a proxy-cache server. Some Web browsers cannot request the proxy-cache server to fetch a new copy from the Internet when the user clicks the Reload button. Instead the Web browser delivers the object that is in the cache again. Therefore, with some browsers, users might not receive refreshed versions of Web pages when they click the Reload button.

- ◆ With HTTP transparent proxying, cache activity might not be completely transparent to users.

Users can receive error messages generated by NetCache for conditions such as the Web server being unavailable. If you do not want users to receive error messages generated by NetCache, you can configure Stealth Mode in NetCache to prevent these messages from appearing.

---

### Note

Stealth mode is available for nontransparent HTTP requests also.

---

- ◆ Problems with URLs can occur for sites using IP-based authentication, as described in [“Caution for Web sites using IP-based authentication”](#) on page 24.

### **Caution for Web sites using IP-based authentication**

Sites using IP-based authentication, for example, electronic commerce Web sites, expect to receive the client's IP address with a request. However, when a request is redirected through a NetCache appliance, the NetCache appliance replaces the client IP address with the appliance IP address. (A proxy-cache server belonging to any vendor routinely replaces the client IP address with the proxy-cache server IP address.)

If you expect your users to send requests to sites that require the client IP address, you need to plan how to handle those requests. Some switches and WCCP routers provide features that enable you to send requests for sites that require the client IP address directly to those sites; that is, requests for those sites bypass the NetCache appliance. Alternatively, NetCache provides the transparency configuration options in the following list that you can use to ensure that the client IP address arrives at the destination server intact.

- ◆ IP spoofing

You can use the NetCache appliance IP spoofing feature to configure NetCache to use the client IP address as the source address when communicating with servers. Requests originating from a client retain the client's source address even if requests are passed through a chain of proxy-cache servers. The requests, therefore, appear to originate from the client rather than from the NetCache appliance.

On the server side, NetCache intercepts the responses from origin servers and caches objects fetched on behalf of the spoofed client. The symmetric routing allows you to gather more accurate client statistics in origin server logs than is possible if the NetCache appliance IP address was used to initiate the connection to the server.

- ◆ Request forwarding

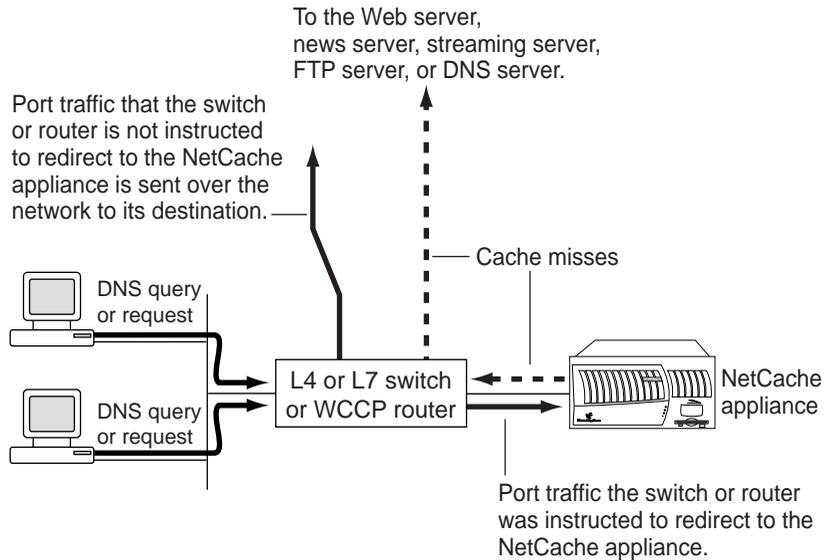
If you do not want NetCache to service some client requests, you can configure NetCache to forward these requests directly to origin servers. Therefore, requests arrive at the origin server with the client's IP address intact. This method of request bypass is called *request forwarding*. Request forwarding is particularly useful for listing URLs that you know are not cacheable or for sites that use IP authentication.

See the transparent proxying chapter in the *Administration Guide* for more details about NetCache IP spoofing and request forwarding features, their use, and their limitations.

### How a switch or WCCP router redirects traffic to a NetCache appliance

An L4 or L7 switch or a WCCP router redirects traffic to a NetCache appliance based on your specifications for the port traffic to be redirected, which you provide in your switch or router configuration and in NetCache configuration.

The following illustration shows the traffic flow from the clients to the NetCache appliance when an L4 or L7 switch or a WCCP router is deployed.



If the NetCache appliance cannot resolve a request, it redirects the request back through the switch or the WCCP router to the origin server—a Web server, news server, or streaming server—as applicable. The origin server returns the requested data to the NetCache appliance. The NetCache appliance returns the requested data to the client while caching the data, if it is cacheable.

### About port redirection

For DNS, FTP, and MMS, required ports for traffic redirection are as follows:

- ◆ UDP port 53 (DNS)
- ◆ TCP port 21 (FTP)
- ◆ TCP port 1755 (MMS)

For the other protocols, common ports for traffic redirection are as follows:

- ◆ TCP port 80 (HTTP)
- ◆ TCP port 443 (HTTPS)
- ◆ TCP port 119 (NNTP)

- ◆ TCP port 554 (RTSP)

If you are using QuickTime for streaming, you must also redirect RTCP port traffic, typically TCP port 2001.

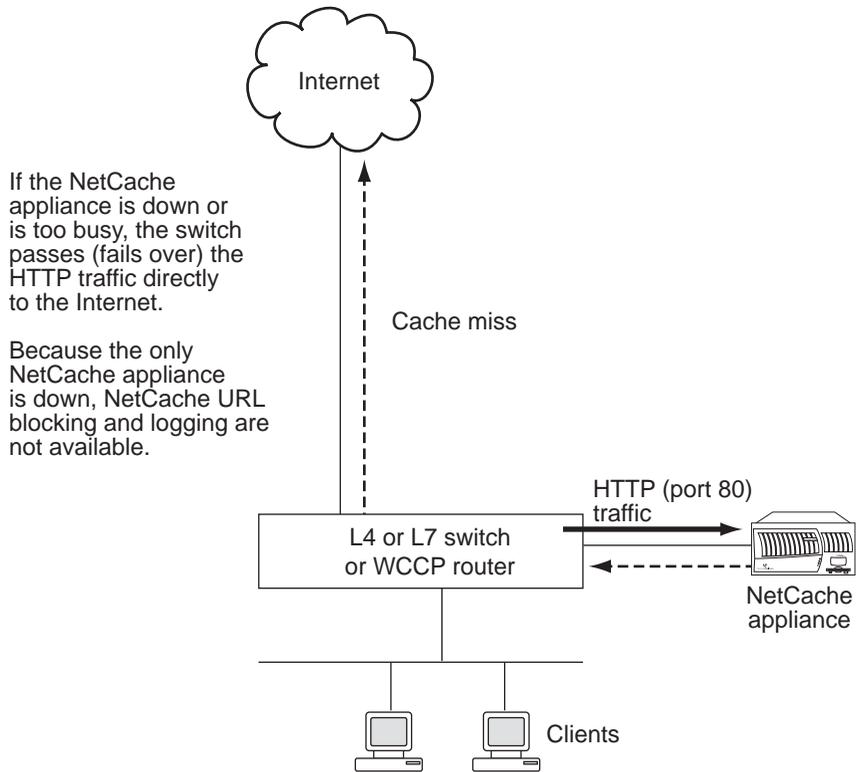
Additional port information for streaming media service is as follows:

- ◆ Redirect TCP port 80 (HTTP) in addition to other protocols.
- ◆ If you are using WCCP and a firewall is between the WCCP router and the appliance, you must allow UDP port 2048. WCCP runs over UDP using port 2048.

### **About failover with transparent proxying**

Failover to another appliance and to the Internet are standard features of transparent proxying with an L4 or L7 switch or a WCCP router. Only standard setup is required to ensure that transparent proxying has the failover functionality. Failover with transparent proxying is very reliable.

The following illustration shows one switch or WCCP router and one NetCache appliance added to the network to handle HTTP requests transparently. If no NetCache appliances are available, the switch or WCCP router directs traffic to the Internet. Although this illustration shows failover for HTTP traffic only, switches and WCCP routers fail over to the Internet by default for all types of traffic.



**Consideration for failover to the Internet:** If you set up your L4 or L7 switch or WCCP router for failover to the Internet, be sure that you are not relying on NetCache features to provide controls over your requests. For example, if the switch fails over to the Internet directly, no URL blocking or logging of client access to URLs is available. If you need to monitor or restrict access to the Internet all the time, you do not want automatic failover to the Internet. You can configure a switch or WCCP router so that it does not fail over to the Internet automatically.

**Failover of streaming media:** You cannot set up a failover mechanism to handle media streams that are in the process of being delivered to clients. The reason is that streaming connections are dropped if a problem occurs between the streaming media cache and the streaming server, or between the streaming media cache and clients. Ensuring that your network is sound should reduce the number of dropped connections.

## About transparent proxying with an L4 or L7 switch

---

### Comparing L4 and L7 switch features

Switches typically include multiple methods for handling traffic. When this guide refers to L4 or L7 switches, it refers to a feature on a switch that can support transparent proxying, that is, operate on the L4 or L7 layer of the OSI model. The layer in the OSI model at which a switch operates determines the capabilities of the switch. Switch names vary by vendor.

**About an L4 switch:** An L4 switch operates at Layer 4 in the OSI model—the Transport layer. L4 switches base their switching decisions on information in the TCP header, and TCP is a protocol that resides at Layer 4 in the OSI seven-layer model. These switches examine only the port number and determine, based on the port number, where to redirect the traffic.

**About an L7 switch:** An L7 switch operates at Layer 7 of the OSI model—the Application layer. Because these switches operate at Layer 7, they can understand URLs and can understand much more about the traffic than an L4 switch can.

An L7 switch has a more sophisticated partitioning capability than an L4 switch. Therefore, an L7 switch can understand much more of the traffic than an L4 switch can. It can, for example, partition HTTP client traffic based on the requested URL, which can be useful in influencing the traffic that the switch redirects to the NetCache appliances. See [“Request distribution with an L4 or L7 switch”](#) on page 30 for more information.

**Features common to L4 and L7 switches:** Both L4 and L7 switches provide the following features:

- ◆ Some L4 and L7 switches can switch more than a gigabyte of data.
- ◆ By default, they partition traffic based on the destination IP address.
- ◆ They can be configured to send traffic directly to the origin server if a NetCache appliance fails.

### Performance comparison between L4 and L7 switches

The performance of L4 and L7 switches is similar. However, the L7 switch examines TCP/IP packets more closely than an L4 switch does. Therefore, the response time of the L7 switch is slightly slower than that of an L4 switch.

**Number of L4 or L7 switches needed**

The number of L4 or L7 switches that you need depends on factors such as the following:

- ◆ The number of NetCache appliances and routers that must be connected to a switch
- ◆ The type of switch
- ◆ Whether you want to deploy a switch failover pair so that one switch can take over for the other if the other switch is unavailable

**Switch location**

The following table discusses issues related to switch location.

Switch location in relation to...	Comments
NetCache appliances	L4 and L7 switches use MAC address rewriting as the method for redirecting packets. Therefore, your L4 and L7 switches must be on the same subnet as the NetCache appliances to which the switch is to redirect packets.  For maximum efficiency, Network Appliance recommends directly connecting the switch to each NetCache appliance.
Clients	The switch must be located so that it can view all network traffic for the clients that it is expected to serve.

**Desirable switch features**

For information about sizing specific switches, see the switch vendor’s documentation. Appendix B, “[Requirements for Transparent Proxying](#),” on page 271 provides a list of key features to look for when buying switches.

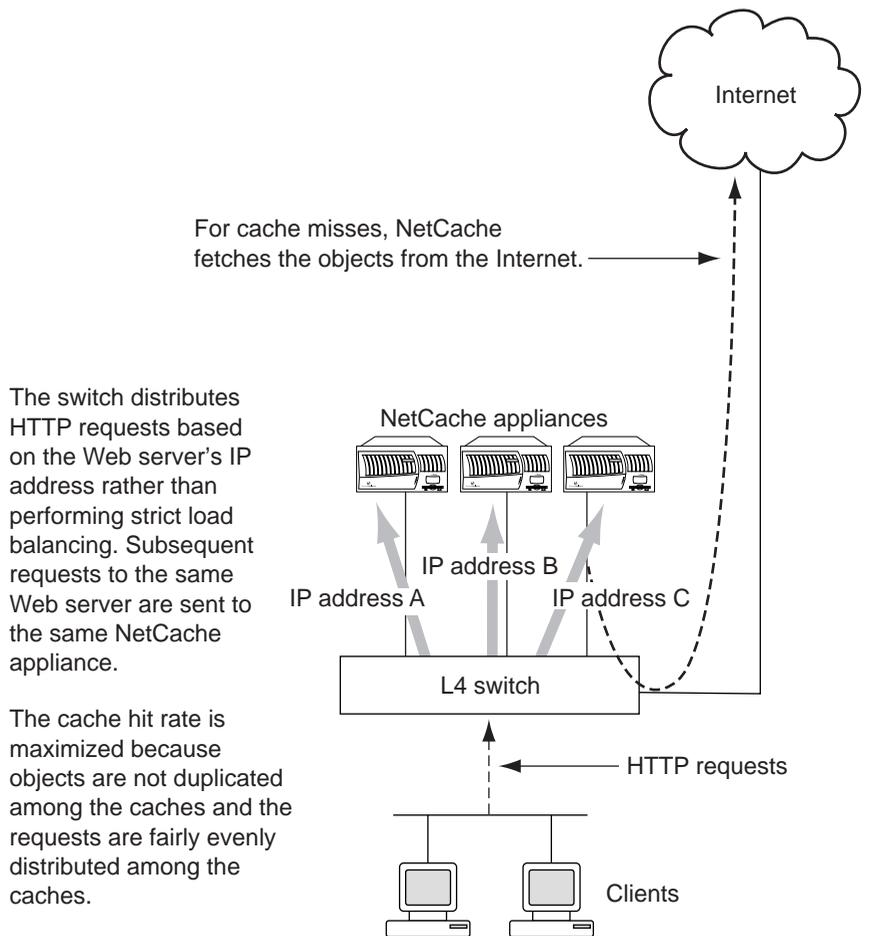
**Avoiding traffic loops when using an L4 or L7 switch**

If you have more than one network interface and you are deploying transparent proxying with an L4 or L7 switch, make sure that the port through which outgoing NetCache traffic is sent is not configured as a transparent port. If outgoing NetCache traffic is sent back through an incoming cache port on the switch, the switch diverts the traffic back to NetCache, creating a traffic loop. If this occurs, no DNS, HTTP, HTTPS, FTP, NNTP, MMS, or RTSP requests are serviced.

## Request distribution with an L4 or L7 switch

### How a switch determines available proxy-cache servers

Setting up transparent proxying with an L4 or L7 switch includes specifying in the switch configuration each NetCache appliance to which the switch can redirect traffic. Based on your configuration of the switch, the switch automatically distributes requests across the NetCache appliances. The following illustration shows a typical forward proxy deployment with a L4 switch distributing HTTP requests over three NetCache appliances.



If a NetCache appliance that is identified in the switch configuration is unavailable, the switch distributes the requests over the available appliances.

**Switch reconfiguration requirement**

Each time you add a proxy-cache server to your network, you must reconfigure the switch with information about the new proxy-cache server. This reconfiguration is not necessary with a WCCP router.

**Influencing how requests are distributed over proxy-cache servers**

The hashing function on a switch can be set up to distribute requests over NetCache appliances based on either of the following:

- ◆ IP address (the most commonly used method)
- ◆ URL (L7 switches only)

Typically, you will want to achieve an even distribution of requests across your NetCache appliances.

**Distribution based on IP address**

When determining how distribution based on IP address will be set up on a switch, consider the most efficient method for the mode in which you are running your NetCache appliance, as the following table shows.

<b>For a NetCache appliance running as a...</b>	<b>Applicable distribution methods for IP address hashing</b>
Forward proxy	Destination IP address (default)
Reverse proxy	Source IP address

The details about the two distribution methods for IP address hashing are described in the following paragraphs.

**Forward proxy (Web cache, streaming media cache, or news cache):**

When users send requests to arbitrary Internet servers (that is, requests are sent to many destinations), distribution on *destination IP address* provides the most equal distribution over multiple NetCache appliances that are running as forward proxies.

Requests for objects from the same origin server are sent to the same proxy-cache server, thereby minimizing duplication of objects among proxy-cache servers. If a proxy-cache server becomes unavailable, another proxy-cache server temporarily handles requests for the unavailable proxy-cache server, thereby

caching objects for origin servers for which it is not ordinarily responsible. When the unavailable proxy-cache server is again available, it takes over responsibility for handling requests for the same origin servers that it handled previously.

**Reverse proxy (Web accelerator or streaming media accelerator):** If you are deploying an accelerator, configuring the switch to partition requests based on *source IP address* provides the best distribution over your NetCache appliances. With request distribution based on source IP address, each appliance receives a portion of the load of client requests. The problem with partitioning requests based on destination IP address is this: if the NetCache appliance is accelerating only one origin server, all requests will be sent to one NetCache appliance because the destination address would be the same for all requests. (The hashing function sends requests for the same destination IP address to the same NetCache appliance.) Even if your NetCache appliance is accelerating a few origin servers, the variation in destination IP addresses would be limited.

When partitioning is based on source IP address, objects might be duplicated among your accelerators because multiple NetCache appliances would be fetching the same objects. However, the hit rate with an accelerator is higher than with a forward proxy. Therefore, duplication of objects is not a concern, as it is with a forward proxy. The reason is that the Web server or streaming server that NetCache accelerates has a limited amount of data, as compared to the World Wide Web, which has nearly an infinite amount of data. With an accelerator, it is likely that many users will send requests for the same data from the Web server or streaming server.

See “[Strategies for client access to an accelerator](#)” on page 172 for an example of an accelerator deployment with a switch or WCCP router.

## **Distribution based on URL**

Some L7 switches allow for request distribution based on URL. The L7 switch examines the request and determines whether the object is cacheable. This capability enables you to set up your switch so that requests for obviously uncacheable objects, such as URLs for CGI, bypass the NetCache appliance. Noncacheable objects are then obtained directly from an origin server.

## About transparent proxying with a WCCP router

---

### About WCCP

WCCP provides a way for routers to redirect traffic to proxy-cache servers. WCCP has built-in mechanisms for request distribution and redundancy.

### Router location

The following table discusses issues related to WCCP router location.

Router location in relation to...	Comments
NetCache appliances	<p>When you have multiple subnets, the router and NetCache appliances can be on different subnets. However, Network Appliance recommends that, for efficiency, the router and NetCache appliances be on the same subnet.</p> <p>You can directly connect a WCCP router to a NetCache appliance. However, a direct connection is not a requirement or a best practice recommendation.</p>
Clients	<p>The router must be located so that it can view all network traffic for the clients that it is expected to serve.</p>
Firewalls	<ul style="list-style-type: none"><li>◆ Do not place the WCCP router in a location where a firewall that uses Network Address Translation (NAT) is located between it and the appliance.</li><li>◆ Firewall configuration for WCCP traffic can be difficult. Consult your Network Appliance service engineer for configuration help and deployment recommendations.</li></ul>

**Traffic redirection methods available**

The following table shows the methods that NetCache supports for WCCP routers to redirect traffic to NetCache appliances.

<b>Traffic redirection method</b>	<b>Description</b>
MAC address rewriting (sometimes referred to as L2 rewriting)	Rewriting of the destination MAC address. The router replaces the packet’s destination address with the address of the target proxy-cache server. The router and the NetCache appliances must be on the same subnet to use MAC address rewriting. A router can redirect traffic more quickly when MAC address rewriting instead of IP-GRE encapsulation is used.
IP-GRE encapsulation (default)	IP packets are encapsulated in another IP packet. IP-GRE encapsulation must be used when the router and NetCache appliances are on separate subnets, but can be used when the router and NetCache appliances are on the same subnet.

**Note**

Not all routers are capable of L2 rewriting. See the Cisco documentation for information about additional limitations with MAC address rewriting.

**WCCP service groups control communication**

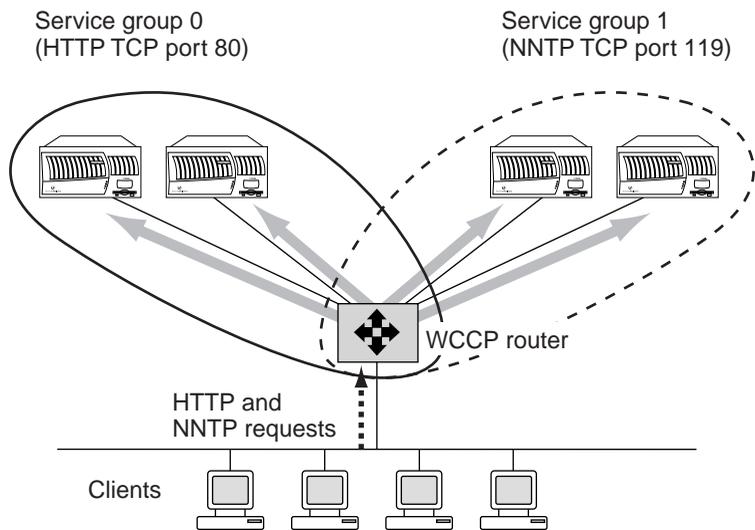
In WCCP, traffic redirection and distribution are based on logical *WCCP service groups*. A service group is a group of one or more routers and one or more proxy-cache servers that can work together in traffic redirection and distribution because they have been defined, through a *service group definition*, with the same settings.

Information you provide in a service group definition on a NetCache appliance includes the following:

- ◆ An ID
- ◆ The traffic to be redirected to the appliance (for example, client-side HTTP TCP port 80 traffic)
- ◆ A specification for how the traffic will be distributed over the NetCache appliances (for example, by destination IP address)
- ◆ Transmission method (unicast or multicast)

The WCCP routers in the same service group have the corresponding configuration, which ensures that they can redirect client-side HTTP TCP port 80 traffic to those NetCache appliances. You can also set passwords by service group to ensure that a WCCP router and proxy-cache server authenticate with each other.

The following illustration shows several NetCache appliances, some that are configured as Web caches handling client-side HTTP traffic and others that are configured as news caches handling client-side NNTP traffic. The router is configured to redirect both HTTP and NNTP traffic. The service groups to which the router and specific appliances “belong” determine the appliances to which the router can redirect a particular type of traffic.



In the preceding illustration, the two NetCache appliances on the left handle only one type of traffic, HTTP (TCP port 80). Service group 0 has been defined in NetCache as the service group for HTTP (TCP port 80) traffic, and the router has been configured to listen for requests from service group 0.

The NetCache appliances on the right handle NNTP (TCP port 119) requests only. Service group 1 has been defined in NetCache as the service group for NNTP traffic, and the router has been configured to listen for requests from service group 1.

Typically, administrators configure a separate service group for each protocol. A single NetCache appliance that handles multiple protocols could, therefore, be configured with multiple service groups. Likewise, a WCCP router can belong to multiple service groups.

---

**Note**

Third-party proxy-cache servers must support transparency and WCCP 2.0 to be included in a service group.

---

**Number of routers with which a NetCache appliance can interact**

Each service group can consist of 64 WCCP routers and up to 32 NetCache appliances (or third-party proxy-cache servers). The routers that a specific NetCache appliance can communicate with depends on the service groups to which the NetCache appliance and the WCCP routers belong.

**Transmission methods between WCCP routers and NetCache appliances**

NetCache supports both unicast and multicast transmission between WCCP routers and NetCache appliances. If your WCCP routers are configured for WCCP multicast, you can configure WCCP service groups for multicast transmission between WCCP routers and NetCache appliances. Unicast and multicast are defined as follows:

- ◆ With unicast transmission, communication occurs between one host and another, in this case, between a WCCP router and a NetCache appliance.
- ◆ With multicast transmission, multiple hosts can listen for transmission from one host. For example, all transmissions from a NetCache appliance for a service group configured for multicast will be detected by all devices listening to the multicast address for that service group.

The advantages of using multicast transmission between members of a service group are as follows:

- ◆ Use of multicast significantly reduces network traffic.
- ◆ Less configuration is required in NetCache.

With unicast transmission, you must explicitly identify WCCP routers with which the NetCache appliance interacts. If a router is later removed from the network, you must remove the router's IP address from the NetCache configuration.

Network Appliance recommends that you select one transmission scheme for your WCCP transparency deployment—either unicast or multicast. The reason is that your configuration and management will be simpler.

## Plan for your service groups

Setting up transparent proxying with WCCP routers and NetCache appliances involves planning for service groups and completing the configuration supporting service groups on both the router and NetCache appliance.

Consider the following requirements:

- ◆ A service group must be defined in the same way on each NetCache appliance that offers the service for which the service group is being set up. That is, if you define a service group for client-side TCP port 119 traffic on one NetCache appliance and assign an ID of 4 to that service group, all service groups that you define on your NetCache appliances for TCP port 119 traffic must be identified as service group 4. Additionally, all other settings for that service group must be the same on all the appliances.
- ◆ You must coordinate your NetCache appliance settings with the WCCP router configurations.

See your WCCP router documentation and the *Administration Guide* for help with planning your service groups.

## Advantages of using a WCCP router instead of a switch

Some advantages of using a WCCP router instead of a switch are as follows:

- ◆ The router can dynamically add to and delete from its list of known caches the proxy-cache servers in service groups the router listens for. No explicit configuration is necessary to make the router aware of NetCache appliances that are in service groups that the router is listening for, as is the case with L4 or L7 switches. You can add and remove NetCache appliances to and from your network without having to reconfigure your WCCP routers.
- ◆ Because WCCP configuration on the NetCache appliance provides information to WCCP routers, configuring a WCCP router is less complex than configuring a switch.
- ◆ NetCache support for the WCCP service group scheme provides you with the flexibility to set up service groups to handle specialized needs, for example, for traffic using nonstandard ports and for IP spoofing.
- ◆ You do not have to buy an extra router. WCCP is typically installed on an existing router.
- ◆ You do not have to configure the router with information about a new proxy-cache server that you have added to the network.

## **WCCP and IP spoofing**

IP spoofing deployments require symmetric routing so that server responses are redirected to the same appliance that received the client request. To implement IP spoofing for WCCP, you must do the following:

- ◆ Assign separate router interfaces to appliances, clients, and servers.
- ◆ Assign appropriate inbound or outbound redirection filters on the router. If you use outbound redirection filters, you must also assign a *redirect exclude in* rule on the port receiving appliance traffic.

See the transparency chapter of the *Administration Guide* for more information.

## Request distribution with a WCCP router

---

### How a WCCP router determines available proxy-cache servers

A WCCP router automatically distributes requests over all NetCache appliances that are in the *same WCCP service group* as the router.

Configuration of service group information on a NetCache appliance enables the appliance to send `HERE_I_AM` messages identifying the services it supports, for example, client-side TCP port 80 traffic (HTTP). Corresponding configuration of service group information on the router enables it to redirect and distribute traffic for services it has been configured to support. A WCCP router listens for `HERE_I_AM` messages to determine the existence of proxy-cache servers and the services each supports.

For example, assume that a WCCP router is configured to listen for HTTP and NNTP requests. When it detects a NetCache appliance in WCCP service group 0 sending `HERE_I_AM` messages, it determines that it can send TCP port 80 (HTTP) requests to that appliance. The router uses its hashing function to determine which appliance in service group 0 to send the request to.

If a particular NetCache appliance is not sending `HERE_I_AM` messages because it is unavailable, the appliance is no longer in the router's list of available appliances over which the router can distribute traffic at that time.

### Influencing the number of requests redirected to each proxy-cache server

Part of WCCP service group definition in NetCache is identifying the hashing function the router will use to distribute requests across proxy-cache servers in the same service group as the router. Typically, you will want to achieve an even distribution of requests across your NetCache appliances. You can, however, add weighting factors to the basic request distribution criteria to achieve an uneven distribution of requests over your NetCache appliances, as described in [“Adjusting request distribution based on weighting”](#) on page 41.

The following table shows the basic criteria available for request distribution and how they apply to the modes in which you can run a NetCache appliance.

For a NetCache appliance running as a...	Applicable distribution methods (hashing on...)
Forward proxy	◆ Destination IP address
Reverse proxy	One or both of the following: ◆ Source IP address ◆ Source port

The details about the two distribution methods for IP address hashing are described in the following paragraphs.

**Forward proxy (Web cache, streaming media cache, or news cache):**

When users are sending requests to arbitrary Internet servers (that is, requests are sent to many destinations), distribution on *destination IP address* provides the most equal distribution over multiple NetCache appliances that are running as forward proxies.

Requests for objects from the same origin server are sent to the same proxy-cache server, thereby minimizing duplication of objects among proxy-cache servers. When a proxy-cache server that was unavailable becomes available again, it takes over responsibility for handling requests for the same origin servers that it handled previously.

**Reverse proxy (Web accelerator or streaming media accelerator):**

If you are deploying an accelerator, configuring the router to partition requests based on *source IP address* provides the best distribution over your NetCache appliances. Each appliance receives a portion of the load of client requests. The problem with partitioning requests based on destination IP address is this: if the NetCache appliance is accelerating only one origin server, all requests will be sent to one NetCache appliance because the destination address would be the same for all requests. (The hashing function sends requests for the same destination IP address to the same NetCache appliance.) Even if your NetCache appliance is accelerating a few origin servers, the variation in destination IP addresses would be limited.

For an accelerator, adding the additional criteria of hashing on source port as well as source IP address results in requests from the same client being distributed over multiple NetCache appliances. The reason is that the source port numbers are dynamically assigned by the operating system on the client and are in effect only for the life of the connection.

When partitioning is based on source IP address, objects might be duplicated among your accelerators because multiple NetCache appliances would be fetching the same objects. However, the hit rate with an accelerator is higher than with a forward proxy. Therefore, the duplication of objects is not a concern, as it is with a forward proxy. The reason is that the Web server or streaming server that NetCache accelerates has a limited amount of data, as compared to the World Wide Web, which has nearly an infinite amount of data. With an accelerator, it is likely that many users will send requests for the same data from the Web server or streaming server.

See “[Strategies for client access to an accelerator](#)” on page 172 for an example of an accelerator deployment with a switch or WCCP router.

### **Adjusting request distribution based on weighting**

NetCache provides default weighting criteria to enable the WCCP router to factor in weighting with the basic request distribution criteria you provide. More requests would, therefore, be sent to a C6200 (a large model NetCache appliance) than to a C1100 (a small model). Typically, you would not change NetCache defaults for weighting. However, it might be beneficial to change NetCache defaults if your traffic patterns indicate that the default weighting does not achieve the most efficient request distribution for your organization.

---

#### **Note**

The weighting feature cannot be used with appliances running NetCache versions prior to 5.1 or with third-party proxy-cache servers.

---

### **Redirecting load when overload occurs**

If a NetCache appliance is overloaded, NetCache can refuse to accept packets for new connections, automatically returning them to the router. The router then directs those packets to the origin server.

NetCache maintains all connections initiated prior to overloading and refuses packets only for new connections. When the NetCache appliance begins to accept new connections again, a client connection timeout is initiated so that any new client connections (that might hang because they have been interrupted) are not left open indefinitely.

**Load balancing data to clients over NetCache appliance network interfaces**

Typically, in a transparent deployment with a WCCP router, the NetCache appliance chooses one network interface over which to receive requests from clients. NetCache always delivers data to clients over the same interface over which it received the requests. However, delivery of data over a single network interface might not be optimal for some types of deployments.

For example, a streaming media cache can deliver streaming data to clients very quickly over multiple network interfaces. However, if the streaming media cache can deliver streaming data over only a single network interface, it is likely to become limited in its network throughput by the bandwidth of this single network interface.

Through NetCache configuration, you can identify the NetCache network interfaces over which you want the NetCache appliance to receive requests from clients, thereby enabling the appliance to deliver data over multiple network interfaces. In this case, the NetCache appliance appears to the WCCP router to be multiple proxy-cache servers.

## Transparent proxy deployment examples

---

### Types of examples

This section provides several examples of transparency deployments. The deployments are essentially the same whether you are using an L4 or L7 switch or a WCCP router. The examples discuss the strategy for each deployment and the effects of the deployment on traffic. This section includes the following examples:

- ◆ “[Example 1: transparent deployment of a DNS proxy cache](#)” on page 43
- ◆ “[Example 2: deployment of a streaming media cache](#)” on page 45
- ◆ “[Example 3: transparent proxying site with NNTP and HTTP service](#)” on page 46
- ◆ “[Example 4: POP and data center using HTTP transparent proxying](#)” on page 48

Some scenarios in other chapters in this guide show deployments that use transparent proxying.

### Example 1: transparent deployment of a DNS proxy cache

In this example, an enterprise company wants to improve the speed of DNS lookups and decrease the amount of bandwidth that is consumed by sending DNS requests to remote servers. Additionally, the managers want to reduce the load on the DNS nameservers. They do so by configuring the NetCache appliance as a transparent DNS proxy cache.

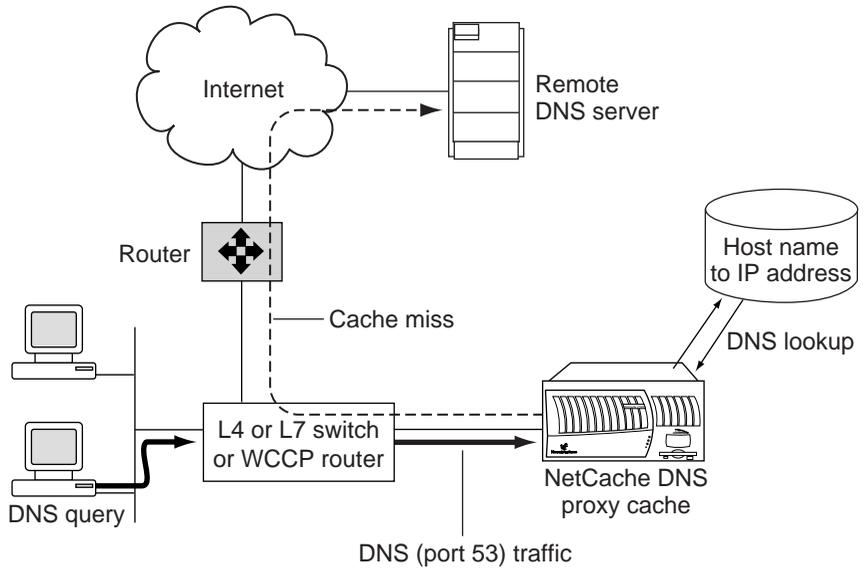
---

#### Note

Alternatively, DNS caching can be set up for nontransparent access to the NetCache appliance.

---

The following illustration shows a transparent proxying topology consisting of one NetCache appliance and one L4 or L7 switch or one WCCP router.



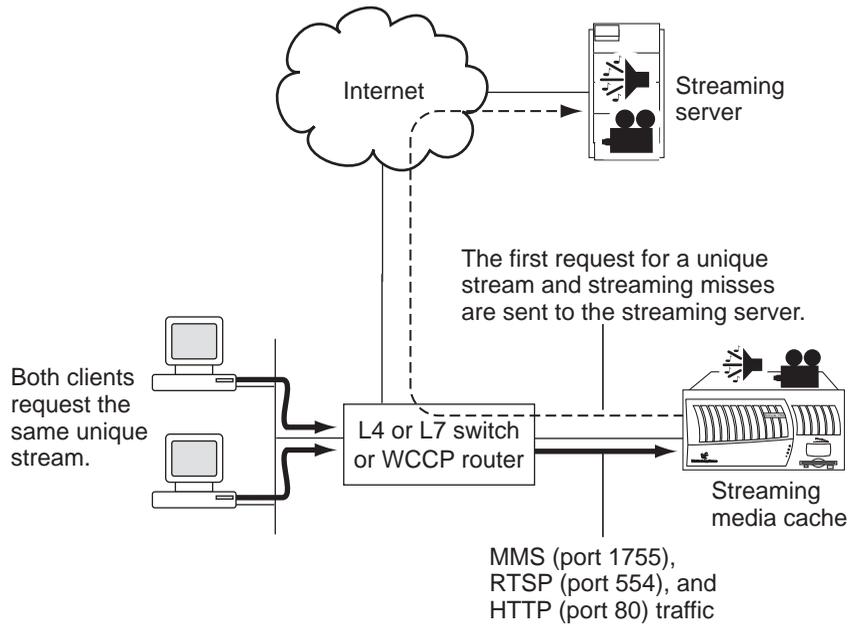
In this deployment

- ◆ The switch or router is located where it can intercept streaming requests from clients.
- ◆ The NetCache appliance was configured as a transparent DNS proxy cache.
- ◆ For deployment with an L4 or L7 switch, the DNS proxy cache was explicitly identified in the switch configuration and the switch was configured to redirect DNS (UDP port 53) traffic to the DNS proxy cache.
- ◆ For deployment with a WCCP router, a WCCP service group was configured on both the router and the DNS proxy cache to enable the router to redirect DNS (UDP port 53) traffic to the DNS proxy cache.

If the DNS proxy cache has the host name to IP addresses mapping in its cache, NetCache can resolve the DNS query without contacting the remote DNS nameserver. If the DNS proxy cache cannot resolve a DNS query, it contacts a remote DNS nameserver to resolve the query, caches the host name and IP addresses resulting from the query, and returns the host's IP addresses to the client.

**Example 2:  
deployment of a  
streaming media  
cache**

In this example, an enterprise company has deployed streaming media service by configuring a NetCache appliance to run as a streaming media cache and by configuring an L4 or L7 switch or a WCCP router to be aware of the streaming media cache.



---

**Note**

Although this illustration shows a single streaming server, MMS and RTSP traffic might be handled by different streaming servers.

---

In this deployment

- ◆ The switch or router is located where it can intercept streaming requests from clients.
- ◆ For deployment with an L4 or L7 switch, the streaming media cache was explicitly identified on the switch and the switch was configured to redirect MMS (port 1755) and RTSP (port 554) traffic to the streaming media cache. HTTP (port 80) traffic was also redirected so that streaming traffic from clients whose media players are set for HTTP only can be handled.
- ◆ For deployment with a WCCP router, WCCP service groups were configured on both the router and the streaming media cache to enable the router to redirect MMS (port 1755) and RTSP (port 554) traffic to the streaming media cache. HTTP (port 80) traffic was also redirected so that

streaming traffic from clients whose media players are set for HTTP only can be handled.

- ◆ Requests for live streams

In this example, both clients are requesting the same unique live (real-time) stream and the stream is being transmitted over unicast. When the streaming media cache receives the first of the two requests for the same stream, the proxy connects to the streaming server to obtain the stream.

When the proxy receives a second request for the same unique stream, the proxy splits the stream that it is already delivering to the first client and delivers the same unique stream to the second client also.

NetCache also supports streaming over multicast transmission. See “[NetCache support for live streams over multicast](#)” on page 120 for more information.

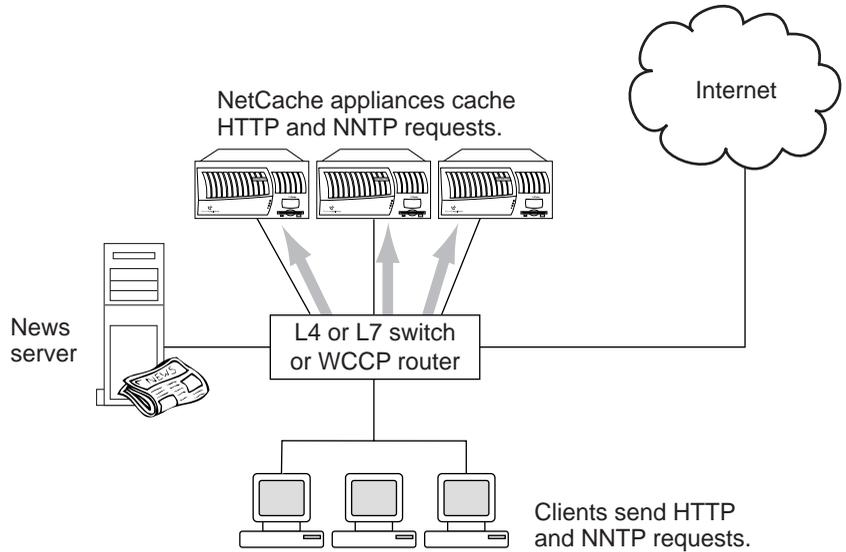
- ◆ Requests for on-demand streams

When an on-demand stream is requested by a client, NetCache determines whether the stream is already in the cache. If the stream is in the cache, NetCache serves it directly to the client from the cache. If the stream is not in the cache, NetCache fetches the on-demand stream from the streaming server and delivers the stream to the client while caching it for future clients.

See Chapter 5, “[Deploying NetCache as a Streaming Media Cache](#),” on page 99 for more information about the NetCache streaming media feature.

**Example 3:  
transparent  
proxying site with  
NNTP and HTTP  
service**

The following illustration shows a transparent proxying deployment consisting of three NetCache appliances and one switch or one WCCP router.



For the organization in this example, three NetCache appliances are required to handle the number of client HTTP and NNTP requests expected. All appliances handle requests of both protocols. The traffic load was not considered to be heavy enough that the cache hit rate or bandwidth savings would be adversely affected by multiple protocols being handled on the same NetCache appliance.

---

**Note**

If your NetCache appliance must handle a heavy load, Network Appliance recommends that you do not set up the same appliance to handle more than one type of traffic.

---

**In this deployment**

- ◆ The switch or router is located so that it can intercept HTTP and NNTP client traffic that is to be redirected to the NetCache appliances.
- ◆ Transparent proxying for HTTP and NNTP was enabled on each NetCache appliance.
- ◆ For deployment with an L4 or L7 switch, the NetCache appliance was explicitly identified in the switch configuration and the switch was configured to redirect HTTP (TCP port 80) and NNTP (TCP port 119) traffic to the NetCache appliances.
- ◆ For deployment with a WCCP router, a WCCP service group was configured on both the router and the NetCache appliance to enable the router to redirect

HTTP (TCP port 80) and NNTP (TCP port 119) traffic to the NetCache appliances.

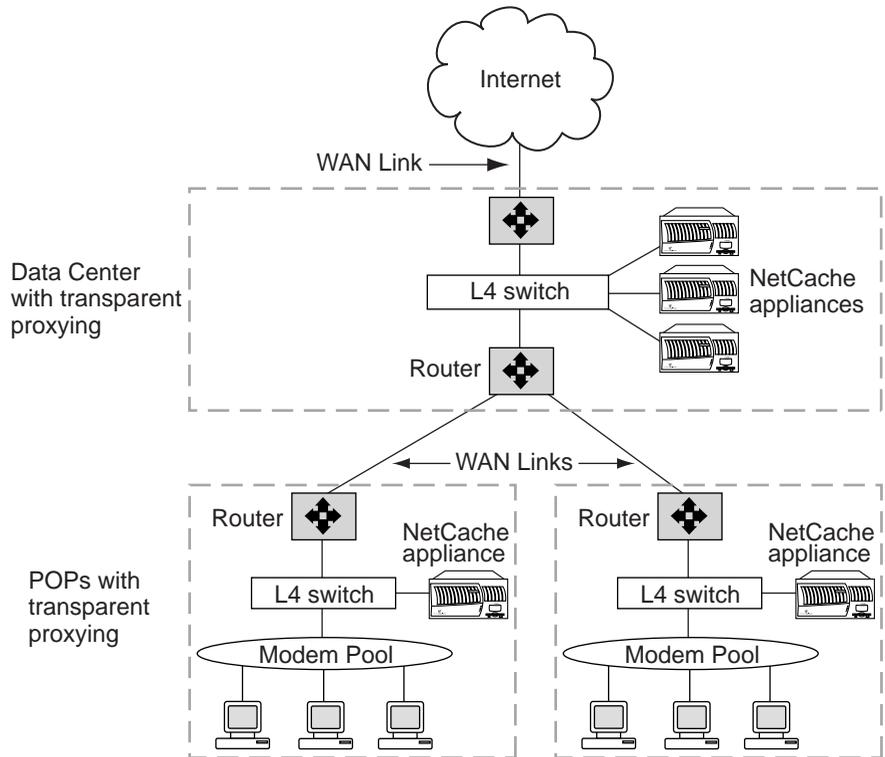
- ◆ The switch or router bypasses the NetCache appliance for all traffic other than HTTP and NNTP.
- ◆ The switch or router distributes HTTP and NNTP requests over the NetCache appliances to ensure the maximum possible hit rate.
- ◆ If the NetCache appliance to which the switch or router redirects a request does not have the requested object, NetCache fetches objects from the Web server or news server, as applicable.

#### **Example 4: POP and data center using HTTP transparent proxying**

In this example, an ISP has two POPs (points of presence) and a data center. Three NetCache appliances are required at the data center to handle the expected load. One NetCache appliance is required at each POP to handle the expected load.

This ISP wants cache misses at the POPs redirected to the data center, instead of having the NetCache appliances at the POPs fetch the objects directly from the Internet. One advantage of this deployment is that the two POPs can share the data cached in the data center NetCache appliances. Additionally, if a NetCache appliance at a POP goes down, the ISP wants the data center switch to intercept HTTP traffic.

The following illustration shows deployment for the ISP using L4 switches to redirect HTTP traffic to the NetCache appliances. Deployment with a WCCP router is essentially the same, with the WCCP software installed on the router on the same subnet as the NetCache appliances. Although the WCCP router and NetCache appliances do not have to be on the same subnet, the most efficient approach is to have the routers at the POPs redirect traffic to the NetCache appliances at the POPs.



**At each POP:** Configuration is as follows:

- ◆ An L4 switch is placed in front of the NetCache appliance at the POP.
- ◆ Transparency is enabled on the NetCache appliance for HTTP.
- ◆ The switch at the POP is configured to be aware of only the NetCache appliance at the POP. The NetCache appliances at the data center are not on the same subnet, so the switch at the POP cannot be made aware of them.

Traffic is handled as follows:

- ◆ The switch redirects all HTTP traffic from the modem pool to the NetCache appliance at the POP.
- ◆ If a NetCache appliance has the object in its cache, it returns the object to the client. Otherwise, the NetCache appliance redirects the request to the data center.

**At the data center:** Configuration is as follows:

- ◆ In this case, one switch is sufficient for the number of routers and NetCache appliances.
- ◆ Transparency is enabled on all the NetCache appliances at the data center.
- ◆ The switch at the data center is configured to be aware of all the NetCache appliances at the data center.

Traffic is handled as follows:

- ◆ The switch distributes requests over all the NetCache appliances at the data center.
- ◆ If none of the NetCache appliances at the data center has the requested object, NetCache sends the request directly to the Internet.

**Disadvantage of the topology in Example 3:** Noncacheable objects requested through the POP's NetCache appliance are also transparently processed by the proxying set up at the data center. Therefore, noncacheable objects are processed by two proxying sites before the object is requested from the Internet, resulting in an increased load on the NetCache appliances at the data center, an increased response time, and lower hit rates.

Because all clients connect to the POP, enabling transparent proxying at the data center is not necessary, unless the NetCache appliance at the POP fails.

**How to avoid the disadvantage of the topology in Example 3:** Ways to avoid the disadvantage noted for Example 3 are as follows:

- ◆ Use an L7 switch instead of an L4 switch, configuring it so that requests for noncacheable objects are sent directly to the origin server, thus bypassing the NetCache appliance completely. For this approach to be successful, your firewall must be configured to allow direct access to origin servers from clients on your network.
- ◆ Use a NetCache logical hierarchy to query caches and distribute requests. See Section A, “[Request resolution hierarchies](#),” on page 70 and “[Example 2: some clients connect directly to the ISP, network hub, or data center](#)” on page 93 for variations of the deployment in Example 3.

## Section C: Direct (nontransparent) client access methods

---

**About this section** This section provides information about the methods available for accessing a NetCache appliance if you do not want to use transparent proxying. “[Summary of nontransparent client access methods](#)” on page 52 contains information about the NetCache mode to which each nontransparent method applies and the request distribution and failover capabilities of each method.

**Contents of this section** This section contains the following topics:

- ◆ “[Summary of nontransparent client access methods](#)” on page 52
- ◆ “[Pointing Web browsers to an automatic proxy configuration file](#)” on page 54
- ◆ “[Pointing Web browsers to a single NetCache appliance](#)” on page 58
- ◆ “[Methods for nontransparent access to a streaming media cache](#)” on page 60

## Summary of nontransparent client access methods

### Nontransparent client access methods

Use the following table to determine the nontransparent client access methods that are applicable to the mode in which you want to run your NetCache appliance.

Nontransparent client access method	Applicable to NetCache appliances running as...	Failover and request distribution
Pointing Web browsers to an automatic proxy configuration file	Web cache (for HTTP, HTTPS, FTP over HTTP, Gopher, and Tunnel protocols)	<p><b>Failover:</b> Includes failover capabilities.</p> <p><b>Request distribution:</b> Includes request distribution capabilities. The browser checks the automatic proxy configuration file at startup, however, so distribution of requests is not dynamic. If you need dynamic distribution of requests, you can use a Server Load Balancer. See <a href="#">“Using a Server Load Balancer for request distribution”</a> on page 65.</p>
Pointing Web browsers to a single NetCache appliance	Any forward proxy mode	<p><b>Failover:</b> Does not provide failover. If you have multiple NetCache appliances and you want failover, see Section B, <a href="#">“Failover by using NetCache appliance takeover pairs,”</a> on page 78.</p> <p><b>Request distribution:</b> If you have multiple NetCache appliances over which you want requests distributed, you must use DNS round robin or a Server Load Balancer. See <a href="#">“Using DNS round robin for request distribution”</a> on page 63 and <a href="#">“Using a Server Load Balancer for request distribution”</a> on page 65.</p>

<b>Nontransparent client access method</b>	<b>Applicable to NetCache appliances running as...</b>	<b>Failover and request distribution</b>
<p>For more recent media players, pointing the media players to the NetCache appliance</p> <p>For Windows WMP media players prior to WMP 7, using NetCache Windows Media metafile rewriting</p>	<p>Streaming media cache (for MMS and RTSP)</p>	<p><b>Failover:</b> Does not provide failover. If you have multiple NetCache appliances and you want failover, see Section B, “<a href="#">Failover by using NetCache appliance takeover pairs</a>,” on page 78.</p> <p><b>Request distribution:</b> If you have multiple NetCache appliances over which you want requests distributed, you must use DNS round robin or a Server Load Balancer. See “<a href="#">Using DNS round robin for request distribution</a>” on page 63 and “<a href="#">Using a Server Load Balancer for request distribution</a>” on page 65.</p>

## Pointing Web browsers to an automatic proxy configuration file

---

### Applicable NetCache mode

An automatic proxy configuration file can only be used to direct requests nontransparently to NetCache appliances running as Web caches.

### About the automatic proxy configuration file

If you deploy transparent proxying, proxy service is available without any configuration of client Web browsers. However, if you are not using transparent proxying, you must configure client browsers so that client requests can be sent through the NetCache appliance. One way you can do this is to create an automatic proxy configuration file. An automatic proxy configuration file enables you to specify, in one place, the Web proxy configurations for your entire organization. Clients in your network must be configured to point to the file instead of to a specific Web cache. Thereafter, Web traffic is directed to the NetCache appliances identified in the file.

With an automatic proxy configuration file, you specify, in one place, the proxy configurations for your entire organization.

### When to use this method

Using an automatic proxy configuration file is the better of the two nontransparent client access methods that are available to you for Web caches. The other method, configuring Web browsers to point to a single NetCache appliance, has a number of disadvantages. “[Pointing Web browsers to a single NetCache appliance](#)” on page 58 describes the limitations of that method.

### Failover possible with the automatic proxy configuration file

If you are using the automatic proxy configuration file for directing browsers to the NetCache appliance, you can include functions in the file for failover to another NetCache appliance or a third-party Web cache, the Internet, or both. Failover through an automatic proxy configuration file has the following limitations, however:

- ◆ Failover through automatic proxy configuration is not very reliable. Not all browsers detect failover well. For example, a browser might not fail over at all or it might take several minutes.
- ◆ You can sometimes swap out a NetCache appliance (for example, for maintenance) without affecting client requests. However, because the Web browser is failing over, problems sometimes occur, such as the browser becoming hung.

- ◆ Failover does not occur if a cache is overloaded.

---

**Note**

---

If your organization requires URL blocking or logging to be in place at all times, do not include functions for failover to the Internet in the automatic proxy configuration file. Because NetCache provides URL blocking and logging, these functions are not invoked if the request is not sent through the NetCache appliance.

---

**Setting up the file for request distribution**

If you are using the automatic proxy configuration file for directing Web browsers to a NetCache appliance, you can include multiple NetCache appliances in the file so that requests can be distributed among the appliances. The JavaScript function then uses a hashing function to distribute requests over the NetCache appliances and any third-party Web caches you specify. Request distribution through an automatic proxy configuration file has the following limitations, however:

- ◆ An automatic proxy configuration file provides browser-based traffic partitioning; therefore, how well an automatic proxy configuration file works for distributing requests varies, depending on the browser and browser version. This method usually does not present a problem with properly sized individual NetCache appliances. However, if you want more predictable results, you can combine the use of an automatic proxy configuration file with a Server Load Balancer (SLB), which provides predictable request distribution.
- ◆ Request distribution is coarser than distributing requests with an L4 or L7 switch. Overload protection is not available.
- ◆ The browser checks the automatic proxy configuration file at startup. Therefore, distribution of requests is not dynamic, unlike an SLB.

**Example of an automatic proxy configuration file**

The automatic proxy configuration file is described in detail in Appendix A, “[Automatic Proxy Configuration File](#),” on page 263. An example of a complete file is provided, as well as incomplete examples that highlight parts of the file. You can use the examples as a basis for creating your own file.

### **Advantages of using an automatic proxy configuration file**

This method has the following advantages in providing client access to the NetCache appliance:

- ◆ It works with all NetCache products and versions, and with third-party proxy-cache servers.
- ◆ The file can be configured easily. You can copy the file shown in “[Examples of automatic proxy configuration files](#)” on page 267, and adapt it for your environment.
- ◆ You can set up the file so that browsers can directly access Web servers inside your local domain instead of sending requests through the NetCache appliance.
- ◆ Multiple NetCache appliances can be included in the automatic proxy configuration file, unless you are using a Server Load Balancer (SLB).
- ◆ Initially, there is no difference in the amount of work required to configure client browsers to point directly to a NetCache appliance or to an automatic proxy configuration file. However, if you use the file, you can save reconfiguration time later because you can make changes to the file without having to reconfigure client Web browsers to match the changes.
- ◆ You can set up your file so that requests are distributed over multiple NetCache appliances. Setting up the file to distribute requests based on IP address maximizes the cache hit rate.
- ◆ Failover can occur to another Web cache or to the Internet, if you set up your file for this feature.

### **Disadvantages of using an automatic proxy configuration file**

This method has the following disadvantages in providing client access to the NetCache appliance:

- ◆ If users do not configure their Web browsers correctly, you can receive many calls for support.
- ◆ Sophisticated users can bypass the NetCache appliance by changing the configuration in the Web browser, unless the firewall configuration prevents this reconfiguration.
- ◆ With this method, Web browsers interact with the automatic proxy configuration file. Because clients use different Web browsers and different versions of Web browsers, results can be unpredictable. This disadvantage is a particular problem if you have included functions for failover, request distribution over Web caches, or both, in the file.
- ◆ You cannot use the automatic proxy configuration file when deploying NetCache appliances as news caches and streaming media caches.

**When this method is not recommended**

Network Appliance does not recommend that dial-up ISPs use an automatic proxy configuration file because of startup problems with some Web browsers. With some versions of Netscape Navigator and Microsoft Internet Explorer, the browser times out before a PPP session is established. The user then receives an error message that might be confusing.

Network Appliance's recommendation for dial-up ISPs is either to use transparent proxying or to start with an SLB and then move to the next generation of SLBs that help with partitioning. If you use an SLB instead of an automatic proxy configuration file, configure the Web browsers to point directly to the SLB.

If you are using an SLB to load balance across NetCache appliances, make sure that no routing loops exist.

## Pointing Web browsers to a single NetCache appliance

---

### Applicable NetCache mode

This method for nontransparently directing client requests to NetCache appliances is applicable to NetCache running in the following modes:

- ◆ Web cache  
For Web caches, pointing browsers directly to a NetCache appliance is usually the least desirable of the nontransparent proxying methods for client access to the NetCache appliance. Using an automatic proxy configuration file is typically a better solution. (See [“Pointing Web browsers to an automatic proxy configuration file”](#) on page 54.)
- ◆ News cache (NNTP)  
For news caches, this is the only nontransparent method available for directing client requests to NetCache appliances.

### When to use this method

In general, using an automatic proxy configuration file for access to Web caches is a better solution. However, this method is advantageous if your organization already has a single NetCache appliance. The reason is that you can avoid having to reconfigure client Web browsers if you add a switch or Server Load Balancer (SLB) and assign it the IP address that the browser expects, that is, the IP address for the NetCache appliance. Then configure the switch or SLB to be aware of the NetCache appliance as required by that switch or SLB.

### Disadvantages of this method

The disadvantages of pointing browsers to a NetCache appliance are as follows:

- ◆ Client browsers must be individually configured and can point only to a single NetCache appliance.
- ◆ If you need to swap out the NetCache appliance, all client browsers must be reconfigured to point to the new appliance, unless you assign the same IP address to the new appliance.
- ◆ If the NetCache appliance goes down, clients cannot access the Internet. No method for failover exists, unlike transparent proxying or an automatic proxy configuration file.
- ◆ A client’s request can be sent only to the specified NetCache appliance. Therefore, load balancing is not possible.
- ◆ Sophisticated users can bypass the NetCache appliance by changing the configuration in the browser.

**If you want a failover method**

If you have multiple NetCache appliances and you to set up a method for failover to another cache, see Section B, “[Failover by using NetCache appliance takeover pairs](#),” on page 78.

**If you want to distribute requests over multiple NetCache appliances**

If you have multiple NetCache appliances over which you want requests distributed, you must use DNS round robin or a Server Load Balancer. See “[Using DNS round robin for request distribution](#)” on page 63 and “[Using a Server Load Balancer for request distribution](#)” on page 65.

## Methods for nontransparent access to a streaming media cache

---

### **When to use nontransparent access to streaming media caches**

Network Appliance recommends that you enable transparent proxying for client access to a streaming media cache. However, if you do not want to set up transparent proxying for streaming media service, you can set up nontransparent access to a streaming media cache.

### **For RealNetworks, WMP 7, and QuickTime media players**

Users of the following media players must configure them to point to a streaming media cache if transparent proxying is not configured for streaming media service:

- ◆ RealPlayer media player
- ◆ Windows WMP 7 or later media player
- ◆ QuickTime media player

Users who request streaming media typically do so by clicking links on Web pages or by entering the same streaming URL in the media player that they would if a streaming media cache were not deployed.

### **For WMP media players prior to WMP 7**

Users of WMP media players prior to WMP 7 can enter URLs into the media player; however, the URL they need to enter to send the request to a streaming media cache is different from the URL they need to enter to send the request to the streaming server.

The NetCache solution for supporting pre-WMP 7 media players is to deploy a Web cache and a streaming media cache and configure the Web cache for Windows Media metafile rewriting. Windows Media metafile rewriting enables MMS requests to be sent to the streaming media cache instead of to the WMT streaming server. The Web cache determines the location of the streaming media cache from the Windows Media metafile rewriting you have configured, and rewrites the destination address to be that of the streaming media cache instead of the WMT streaming server. When a pre-WMP 7 media player user clicks a link on a Web page, streaming requests are sent to the Web cache first, and then to the streaming media cache.

The same NetCache appliance can be configured to run as both a Web cache and streaming media cache, in which case the Windows Media metafile rewriting directs requests to the streaming media module on the same NetCache appliance. See “[Scenario: Deployment with a Windows Media server](#)” on page 150 for more information.

**Request distribution**

If you are not using transparent caching for streaming media and you want requests distributed over multiple NetCache appliances, you must use DNS round robin or a Server Load Balancer. See “[Using DNS round robin for request distribution](#)” on page 63 and “[Using a Server Load Balancer for request distribution](#)” on page 65.

**If you want a failover method**

If you have multiple NetCache appliances and you want to set up a method for failover to another cache, see Section B, “[Failover by using NetCache appliance takeover pairs](#),” on page 78.

## Section D: Request distribution for nontransparent client access methods

---

### About this section

The only nontransparent client access method that provides request distribution is an automatic proxy configuration file. Depending on the nontransparent client access method you select, or if you want different functionality than an automatic proxy configuration file provides, you must find a method for distributing requests over multiple NetCache appliances. “[Summary of nontransparent client access methods](#)” on page 52 contains information about the request distribution features of each client access method discussed in the previous section.

### Contents of this section

This section contains the following topics:

- ◆ “[Using DNS round robin for request distribution](#)” on page 63
- ◆ “[Using a Server Load Balancer for request distribution](#)” on page 65

## Using DNS round robin for request distribution

---

### When you would use this method

If you are using either of the following client access methods, you can use DNS round robin to distribute requests over NetCache appliances:

- ◆ Pointing clients to a single NetCache appliance
- ◆ Using a nontransparent access method for streaming media

Compare this discussion of using DNS round robin for request distribution to [“Using a Server Load Balancer for request distribution”](#) on page 65.

### Type of DNS round robin described

This section applies to browsers that use DNS round robin to choose between NetCache appliances. It does not apply to NetCache appliances using DNS round robin to choose between Web servers.

### DNS round robin described

DNS round robin distributes traffic to NetCache appliances in a predetermined cyclical pattern. For example, the first request is sent to NetCache A, the second to NetCache B, the third to NetCache C, and so on. When the end of the list of NetCache appliances is reached, the cycle starts again at NetCache A. This method, therefore, gives each NetCache appliance an equal number of requests. There is no possibility, for example, to have more requests sent to a high-end NetCache appliance than to a low-end appliance.

### Setting up DNS round robin

You set up DNS round robin so that the IP addresses of the different NetCache appliances correspond to a single DNS host name. You should set up the DNS nameserver so that it periodically changes the primary IP address that it gives out for the requested host name, for example, for every request or every other request. The result is that different clients are pointed to the NetCache appliance that is next in the round-robin cycle.

Additionally, you configure the client’s Web browser proxy settings with the host name, not the IP address, of the NetCache appliance.

### **Advantages of this method for request distribution**

This method has the following advantages:

- ◆ It works with all NetCache products and with third-party Web caches.
- ◆ It is an inexpensive method of request distribution because you do not have to purchase hardware.

### **Disadvantages of this method for request distribution**

Network Appliance does not recommend this method because you can experience the following problems when using DNS round robin for caching service:

- ◆ DNS cannot determine the availability or performance of a given server. Therefore, it continues to direct client requests to failed or overloaded servers. If one NetCache appliance goes down, DNS continues to send requests to it. Clients receive error messages; for Web caching, the messages indicate that those URLs are unreachable.

In contrast, if an L4 or L7 switch, a WCCP router, a Server Load Balancer (SLB), or an automatic proxy configuration file is used and one appliance goes down, requests are sent to another appliance.

- ◆ Because all requests can be sent through any NetCache appliance, the same objects can be cached in more than one appliance. To achieve the maximum cache hit rate, requests for the same objects should be sent to the same NetCache appliance.

- ◆ For Web caching, problems with cookies can occur if the proxy route changes.

Cookies are often used to store authentication credentials. When cookies are used for this purpose, it is not always necessary to reauthenticate future requests. The cookies commonly encode the IP address from which the request originates and the user name. If the IP address in the request changes because a different Web cache is used to send a request from a user, a cookie might be rendered invalid.

This problem is unlikely to occur unless you are using multiple levels of caches to handle requests, as described in Section A, “[Request resolution hierarchies](#),” on page 70.

### **Where to get more information about DNS round robin**

You can obtain more details about DNS round robin in technical report TR3007, “Scaling Web sites for the Expanding Internet Universe.” You can find this report at the Network Appliance Web site, <http://www.netapp.com/>. To find the report quickly, enter TR3007 as the search keyword in the Web site’s search engine.

## Using a Server Load Balancer for request distribution

---

### When to use this method

If you are using any of the following client access methods, you can use a Server Load Balancer (SLB) to distribute requests over NetCache appliances:

- ◆ Pointing clients to a single NetCache appliance
- ◆ Using an automatic proxy configuration file
- ◆ Using a nontransparent access method for access to a streaming media cache

Compare this discussion of using an SLB to [“Using DNS round robin for request distribution”](#) on page 63.

### Methods SLBs use to distribute client traffic

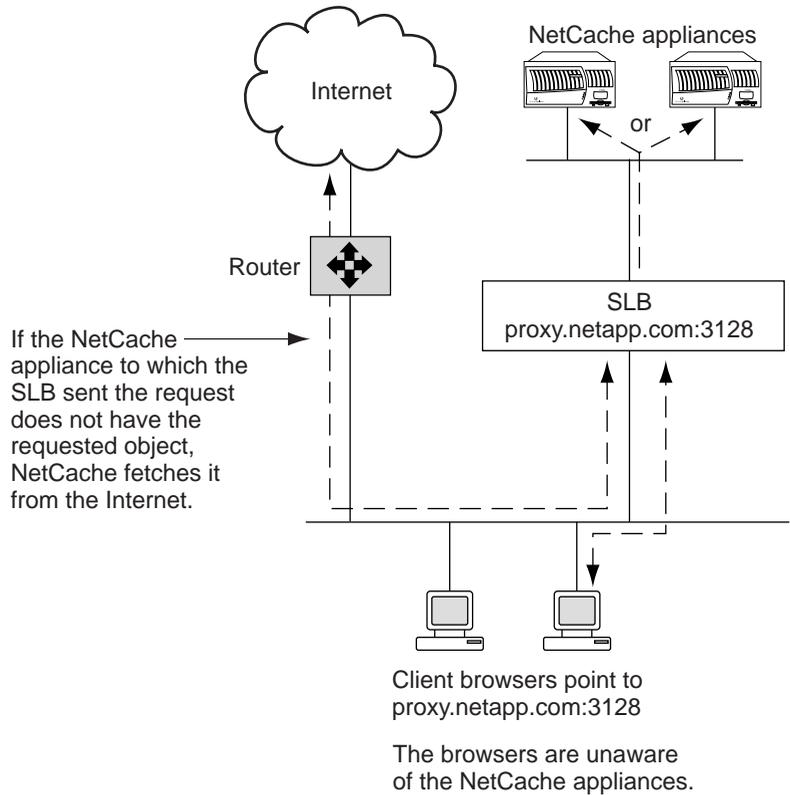
SLBs are likely to provide strict load-balancing algorithms, for example:

- ◆ Maximum connections
- ◆ Weighted percentage
- ◆ Fastest connection
- ◆ Round robin

SLBs do not distribute requests based on IP address, which is desirable for caching service because distribution based on IP address results in a higher hit rate.

### Example: using an SLB for Web caching

The following illustration shows a Web caching deployment that uses an SLB to balance Web requests over the NetCache appliances.



Although the previous illustration shows the SLB interacting with NetCache appliances functioning as Web caches, you can also deploy an SLB to balance requests over NetCache appliances configured as news caches and streaming media caches. The news caches would fetch cache misses from a news server rather than from the Internet.

To deploy an SLB with NetCache appliances, do the following:

- ◆ Ensure that the SLB and NetCache appliances are on the same subnet.
- ◆ On the SLB, specify the IP addresses of the NetCache appliances so that the SLB is aware of them.
- ◆ Set up client browsers to point to the SLB rather than to the NetCache appliances. You can do this by having users configure their browsers to point to the SLB's IP address or, for Web caching, by entering the SLB's IP address in the automatic proxy configuration file.

**Advantages of this method for request distribution**

Using an SLB has the following advantages when compared to other methods:

- ◆ Because the client browsers are unaware of the NetCache appliances, you can swap out NetCache appliances (for example, for maintenance) without affecting client requests.
- ◆ If one NetCache appliance fails, the SLB shifts requests to the remaining NetCache appliances.
- ◆ Requests are dynamically distributed over NetCache appliances

**Disadvantages of this method for request distribution**

Using an SLB has the following disadvantages when compared to other methods:

- ◆ The cache hit rate is not as high as with methods that distribute requests based on IP address.
- ◆ No failover to the Internet is available if all the NetCache appliances fail.

## Section E: Client access through global request routing

---

### **An alternative request routing solution for CDNs**

Another method for directing requests to NetCache appliances is the NetCache Global Request Manager (GRM) feature. GRM directs content requests from clients in a telco or an enterprise Content Delivery Network (CDN) to the NetCache appliances that are closest to the clients. GRM supports two different redirection methods:

- ◆ DNS-based redirection services
- ◆ L7 redirection services

For more information about using this feature for routing requests to NetCache appliances, see Chapter 11, “[Global Request Manager](#),” on page 243 and the *Guide to Global Request Manager*.

**About this chapter** This chapter describes some of the features that you can use to optimize caching services when you are deploying multiple NetCache appliances.

**Chapter contents** This chapter contains the following sections:

- ◆ Section A, “[Request resolution hierarchies](#),” on page 70
- ◆ Section B, “[Failover by using NetCache appliance takeover pairs](#),” on page 78

## Section A: Request resolution hierarchies

---

**About this section** This section describes how you can use the NetCache hierarchy feature to build logical relationships between NetCache appliances and third-party proxy-cache servers that enable them to work together to resolve requests. Examples in this chapter and scenarios throughout this guide show how hierarchies can be used to optimize request resolution.

**Contents of this section** This section contains the following topics:

- ◆ [“What is a hierarchy?”](#) on page 71
- ◆ [“Hierarchy deployment examples”](#) on page 75

## What is a hierarchy?

---

### About a request resolution hierarchy

Through NetCache configuration, you can logically define different levels of NetCache appliances to create a request resolution hierarchy. This structure enables a NetCache appliance to forward a request it cannot resolve to specific NetCache appliances, third-party proxy-cache servers, and nontransparent firewalls for request resolution. The purpose of a hierarchy, therefore, is to enable you to define what is to happen when the NetCache appliance you are configuring cannot resolve a request—that is, what that appliance is to do when a *cache miss* occurs. Hierarchy configuration affects traffic *outbound* from the NetCache appliance on which you configure the hierarchy, not the request traffic from clients.

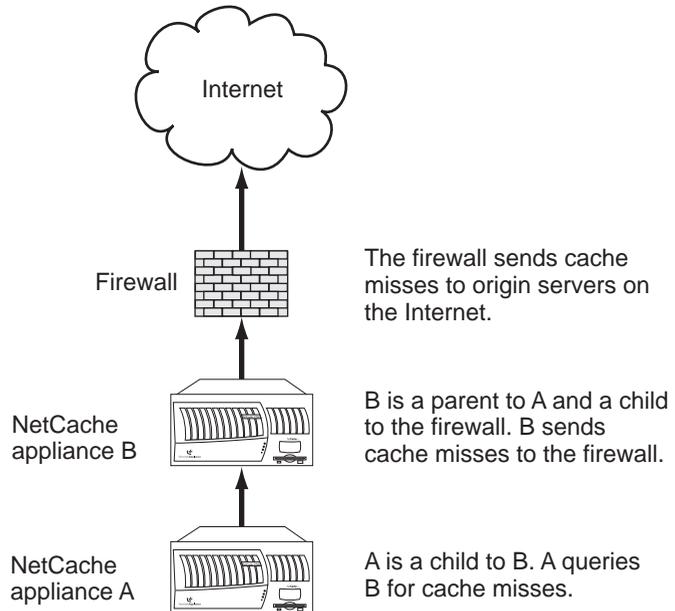
---

#### Note

A hierarchy cannot be used for resolving NNTP and DNS requests.

---

The following illustration shows an example of a simple hierarchy.



You can include in a hierarchy multiple levels of parents and multiple parents for a particular child. If desired, you can define parents to be in a logical cluster to take advantage of distribution through a hashing function (see “[Request distribution in a cluster](#)” on page 72). You can also include NetCache appliances at the same level as siblings. However, because ICP is used for communication between caches at the same level, Network Appliance does not recommend including appliances at the same level in your hierarchy. The reason is that ICP is not as efficient as TCP.

### Functions of hierarchy members

Consider the following when planning a hierarchy:

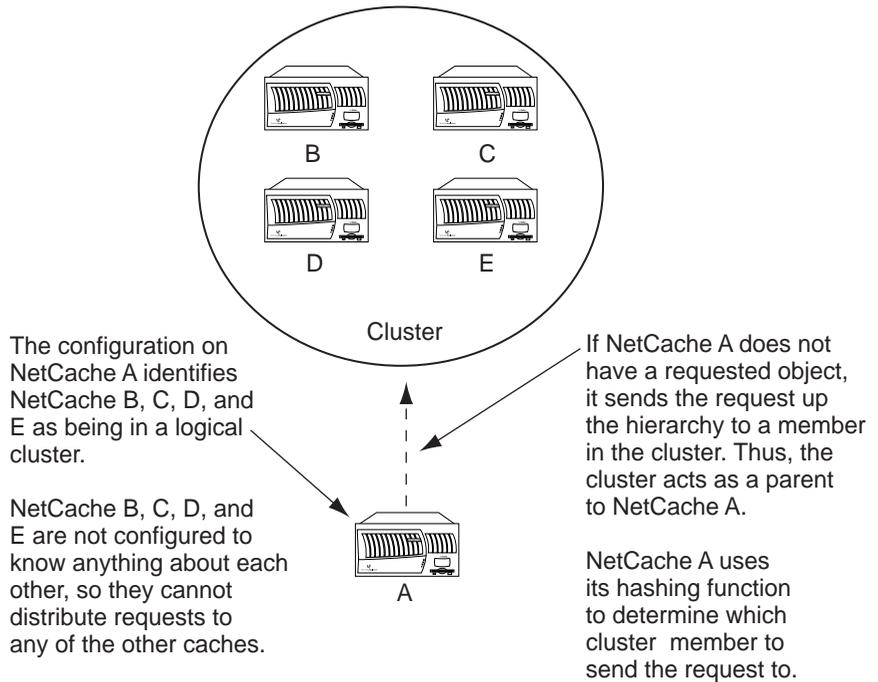
- ◆ A parent cache can fetch objects directly from the origin server. If you define multiple levels of parents, however, the request must be passed up through the hierarchy, with the top-level parent being the one that fetches the request directly. If the top-level parent is inside a nontransparent firewall running as a proxy, the request must be sent through the firewall to the origin server.
- ◆ A child cannot fetch objects directly from the origin server, only through a parent. In the preceding illustration, NetCache appliance A sends requests that it cannot resolve to NetCache appliance B, its parent in the hierarchy.
- ◆ Child caches fetch noncacheable objects directly from the origin server, unless network security policies prevent the child from accessing the origin server directly, or unless you explicitly configure the child not to fetch noncacheable objects directly from the origin server.

You should devise an overall plan for the interaction among all of your NetCache appliances for resolving cache misses. You will, however, configure each NetCache appliance individually, with its individual view of the hierarchy. For example, NetCache appliance A in the previous illustration is configured to be aware only of NetCache appliance B as its parent and, therefore, can pass its cache misses only to NetCache appliance B. If other NetCache appliances were in the hierarchy above its parent NetCache appliance B, NetCache appliance A would not be configured with any information about them.

### Request distribution in a cluster

The following example shows a hierarchy from the viewpoint of NetCache A, that is, the hierarchy as it is configured on NetCache A. In this case, four NetCache appliances are identified as a parent cluster so that a hashing function on the child (NetCache A), by looking at the URL, can determine which of the four appliances in the cluster should receive the request. If the parents were not identified as part of a cluster, NetCache A would query each one individually for cache misses, which is not as efficient.

The hashing function that NetCache uses enables NetCache to distribute requests optimally among the members of the cluster, thereby minimizing duplication of objects among cluster members.



The NetCache appliances in the cluster are not configured to be aware of the cluster. The reason is that it is not efficient for caches at the same level to query each other for objects; it is faster for them to send requests directly to the next level or to the Internet to obtain the objects.

“[Example 1: cluster used with transparent Web caching](#)” on page 75 provides an example of a deployment using a cluster.

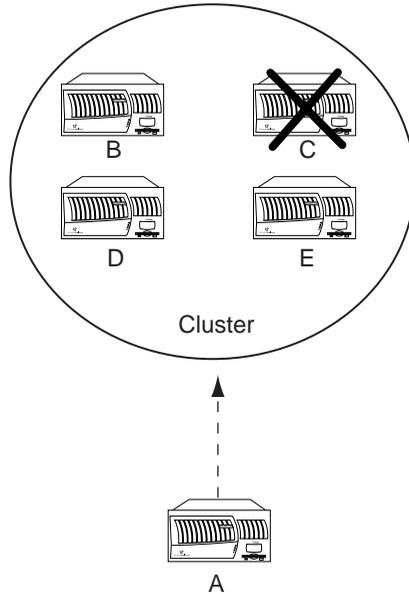
---

**Note** NetCache clusters are similar to other Web caches that use Microsoft’s Cache Array Protocol (CARP).

---

**Failover to another cluster member**

A NetCache appliance distributes requests over any cluster members that are available.



With the preceding illustration, assume that NetCache appliance A had been sending cache misses for Web server *www.abc.com* to cluster member C (because the hashing function determined that C could handle the URL in the request most efficiently). If C goes down, A sends a cache miss for *www.abc.com* to another cluster member.

When C comes back up, NetCache resumes sending cache misses for *www.abc.com* to C. NetCache returns to its “preferred algorithm” as soon as possible.

If all the members in the cluster go down, the child cache to the cluster (A in this example) sends requests directly to the Internet if security policies allow this failover approach.

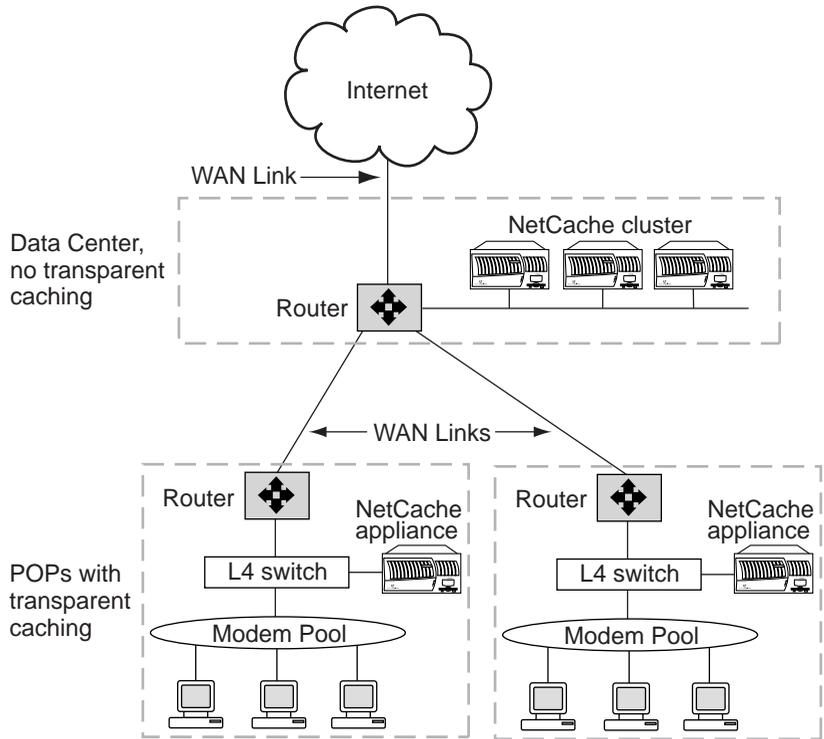
## Hierarchy deployment examples

### Example 1: cluster used with transparent Web caching

The ISP in this example has two goals:

- ◆ Implement transparent Web caching so that users do not have to configure their Web browsers to support caching.
- ◆ If the NetCache appliance at the POP does not have a requested object, a cache miss will be sent to the data center for processing instead of having the NetCache appliance at the POP fetch objects from the Internet directly.

This following illustration shows the ISP's deployment, in which transparent proxying was deployed only at the POPs.



#### Configuration:

- ◆ The NetCache appliance at each POP was configured to view the NetCache appliances at the data center as a cluster. This enables the NetCache appliance at each POP to distribute requests over the data center NetCache

appliances, using the NetCache hashing function. Because the cluster acts as a parent to the NetCache appliance at each POP, the NetCache appliances in the cluster can fetch objects from the Internet, but the NetCache appliances at the POPs cannot.

- ◆ The NetCache appliances at the data center were not configured to be aware of each other; that is, they are not aware that the NetCache appliance at each POP considers them to be in a cluster, and they do not know that the other NetCache appliances in the cluster exist. The reason is that after a NetCache appliance at the data center receives a request from a POP's NetCache appliance, it is more efficient for a data center appliance to send a request directly to the Internet if it cannot resolve the request rather than try to find the object through intercache communication.

### **Why transparent proxying was deployed only at the POPs:**

- ◆ The ISP's goal of making caching transparent to the user is fulfilled by implementing transparent proxying only at the POPs because all the ISP's customers dial in to one of the POPs.
- ◆ If a POP NetCache appliance does not have an object, it requests the object from the data center cluster. Even if a NetCache appliance in the cluster must obtain the object from the Internet, the object is then available to any POP that requests it. After the data center's NetCache appliance returns a requested object to a POP's NetCache appliance, the next request a client makes for that object can be resolved by the NetCache appliance at the POP. This process is more efficient and faster than if a POP's NetCache appliance had to send requests directly to the Internet to obtain objects it does not have, and it saves bandwidth.
- ◆ If a POP has multiple routes, the routes can flap or be asymmetric, which breaks transparent proxying. If multiple routes exist, it is better to use hierarchical caching to avoid these problems. The reason is that it is more difficult to constrain routing at a POP if you have transparent proxying at the data center.
- ◆ If transparent proxying with an L4 switch was deployed at both the POPs and the data center, noncacheable objects would be processed by two caching sites before the object is requested from the Internet, which is not efficient.

Alternatively, you can use an L7 switch instead of an L4 switch. You can configure an L7 switch so that it obtains noncacheable objects from the Web server directly, bypassing the NetCache appliance completely. However, this approach does not fulfill the goal of preventing direct access to the Internet from the POPs.

---

**Note**

---

You can compare this example to “[Example 4: POP and data center using HTTP transparent proxying](#)” on page 48, which shows a switch at the data center and transparent proxying enabled on all the NetCache appliances at the data center.

---

**Where to get more information**

You can use hierarchies and clusters to achieve many different results. This section provides a brief introduction to the subject. The chapter about hierarchies in the *Administration Guide* provides detailed information about using and configuring clusters and hierarchies. It describes ways to customize request distribution to handle various situations, such as forcing certain NetCache appliances to handle specific types of requests and bypassing parent caches for requests to Web servers that are close to a NetCache appliance.

## Section B: Failover by using NetCache appliance takeover pairs

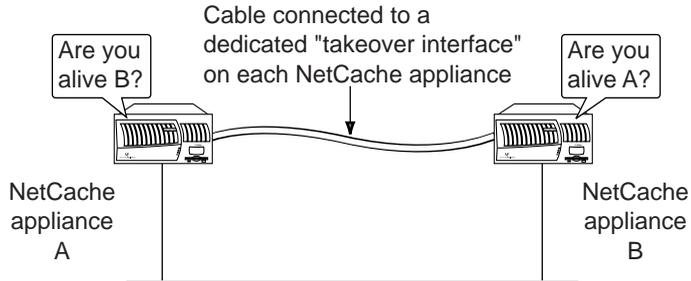
---

### About NetCache appliance takeover

You can use NetCache configuration to specify one or more NetCache takeover pairs. Then, if one NetCache appliance in the pair goes down, the other takes over servicing HTTP, HTTPS, FTP, Gopher, MMS, and RTSP requests for the other appliance. This feature does not take over servicing NNTP requests.

For MMS and RTSP requests, existing streaming connections are lost when a NetCache appliance in the takeover pair fails. However, the takeover partner handles new streaming media requests for the appliance that failed.

The following illustration shows a NetCache appliance takeover pair.



On each appliance in the takeover pair, the other appliance is specified as its takeover partner. Every 10 seconds, each appliance then communicates with its partner by using a communication “heartbeat,” similar to a ping, to determine whether its partner is alive. If a NetCache appliance does not respond to its partner’s heartbeat query, the partner takes over and starts servicing all requests for the IP addresses and IP address aliases of its partner.

This takeover process enables clients of the down partner to continue to use the same IP address to contact a cache. Therefore, to most users and other NetCache appliances, it appears as if no failure occurred.

Each appliance in the takeover pair is equal from a failover perspective. There is no concept of a primary NetCache appliance and a secondary NetCache appliance, as there is with using the automatic proxy configuration file for Web caching.

**When to use this feature with transparent proxying**

If you are deploying transparent proxying with an L4 or L7 switch or a WCCP router, you would not use this failover mechanism because the switch or router provides better failover.

If you are deploying transparent proxying with a policy-based router, using this failover feature in combination with transparent proxying is a good strategy because currently available routers that use policy routing cannot provide failover.

**Networks that this feature supports**

This feature is supported for Ethernet and FDDI. It does not work with ATM.

**Reliable connection is needed between the appliances**

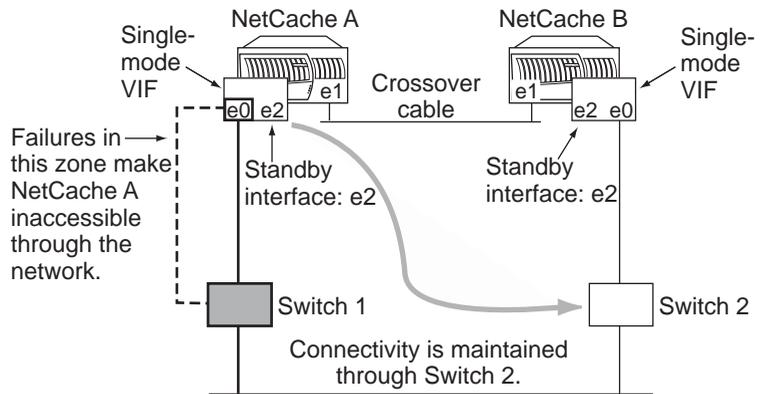
Choose the simplest and most reliable connection possible between the two NetCache appliances. Network Appliance recommends that you use a crossover cable to connect the interfaces on the appliances that are the dedicated “takeover interfaces.”

If you use another device, such as a switch, between the appliances, the risk exists that the device will fail, making this failover mechanism unavailable. If the connection between the two NetCache appliances fails, each NetCache appliance appears to the other to have failed and both appliances attempt to take over for each other. The result on your production networks is the same as if you had two appliances on the same network with the same IP address. “[Example of adding network fault-tolerance](#)” on page 79 describes how you can add network fault-tolerance to your NetCache takeover scheme.

**Example of adding network fault-tolerance**

Although a NetCache appliance takeover pair provides higher system availability in case of a system failure, it does not ensure higher network fault-tolerance. If a network-related problem occurs, a NetCache appliance in a takeover pair might still become inaccessible.

The following illustration shows a deployment in which a single mode virtual interface (VIF) is combined with NetCache takeover to achieve a higher fault tolerance than can be achieved with NetCache takeover alone. The following discussion shows a single mode VIF used with a non-takeover interface. Alternatively, a single mode VIF can be used with the takeover interface.



The previous illustration shows the failure zone for NetCache A (between Switch 1 and e0 on NetCache A, or the network between Switch 1 and NetCache A) that NetCache B cannot detect even though NetCache A might be inaccessible through the network.

To provide a fully redundant network architecture for your NetCache appliance takeover pair, you can set up a single-mode virtual interface on both takeover partners. Subsequently, if a failure occurs in the failure zone for NetCache A, the standby interface (e2) on NetCache A maintains connectivity through Switch 2. NetCache B can then take over caching services for NetCache A if NetCache A fails. Likewise, the standby interface on NetCache B would maintain connectivity to Switch 1 if a failure occurs in the failure zone for NetCache B. With this approach, if a network-related failure occurs, both NetCache appliances remain accessible through the network.

**Note**

In a single-mode virtual interface, all interfaces that constitute the virtual interface share an IP address. However, only one interface in the virtual interface is active. The other interfaces are on standby, ready to take over if the active interface fails.

**Interface requirements**

The interface requirements for NetCache appliance takeover are as follows:

- ◆ You must dedicate one interface on each NetCache appliance in the pair for takeover. Do not send proxy traffic to that interface.
- ◆ Each NetCache appliance in the pair must have an interface on the same LAN.

**Duplicate configuration requirement**

You must set up access controls the same way on both appliances in the designated pair. Otherwise, when one of the appliances fails and the other appliance takes over, the takeover partner does not apply the same controls. Likewise, if you are using user authentication, you must set up users the same way on both appliances. Alternatively, you can configure NetCache to use an LDAP or RADIUS authentication server or the NTLM authentication protocol. Otherwise, users who are not configured on the takeover partner are denied access if protocol authentication is turned on.

**Relationship with automatic proxy configuration**

If you are using an automatic proxy configuration file for Web caching, you might have included entries to enable failover to another appliance. If so, the automatic proxy configuration file failover is not invoked unless both NetCache appliances in the pair go down. NetCache takeover takes precedence over other mechanisms.

**Note**

---

Do not enter the IP address of the dedicated takeover interface in an automatic proxy configuration file.

---



**About this chapter**      This chapter describes deploying NetCache as a Web cache.

**Chapter contents**      This chapter contains the following sections:

- ◆ “[About NetCache as a Web cache](#)” on page 84
- ◆ “[Deployment considerations](#)” on page 86
- ◆ “[Optimizing a Web site for caching](#)” on page 88
- ◆ “[Scenario: NetCache deployed in an enterprise environment](#)” on page 89
- ◆ “[Scenario: NetCache deployed at a global carrier and an ISP](#)” on page 91
- ◆ “[Scenario: NetCache deployed with high-latency, high-bandwidth links](#)” on page 95



---

**Note**

---

The term *Web request* in the previous illustration refers to an HTTP, FTP, Gopher, or Tunnel request. DNS requests for DNS caching would also be sent to a Web cache.

---

NetCache caches only cacheable objects. In response to requests for noncacheable objects (for example, CGI or private pages), NetCache fetches those objects from the Web server and passes them to the client without caching them.

---

**Note**

---

A Layer 7 (L7) switch can be configured to obtain noncacheable objects directly from the Web server, rather than having the Web cache fetch them and pass them to the client.

---

**Advantages of  
using a Web cache**

The advantages of using a NetCache appliance as a Web cache include the following:

- ◆ Enables Web pages to be provided more quickly to client users because Web objects are cached close to users
- ◆ Reduces network latency to enhance the overall Web experience for e-commerce customers, intranet users, and external suppliers and partners
- ◆ Reduces cost because the Internet connection is used efficiently  
Duplicate requests for the same content are satisfied from the cache instead of being sent over the WAN, thus saving bandwidth costs.
- ◆ Reduces bandwidth demands during network traffic surges, during which a large group of users wants access to the same small number of pages
- ◆ Enables access to certain Web sites to be blocked  
Some organizations block access to protect themselves from possible legal actions if employees are exposed to content that the organization considers inappropriate.
- ◆ Gathers and caches network-management information that allows for more efficient network management
- ◆ Off-loads work from the Web server onto the Web cache so that fewer Web servers are needed

## Deployment considerations

---

### Strategies for client access to your NetCache appliances

You can set up client access to a Web cache in the following ways:

- ◆ Through transparent proxying—for HTTP and FTP requests only
- ◆ Through a nontransparent client access

You would need to manually configure client browsers to point directly to one of the following:

- ❖ A NetCache appliance
- ❖ A device such as a Server Load Balancer (SLB) that is in front of the NetCache appliances and is configured to be aware of the NetCache appliances
- ❖ An automatic proxy configuration file

See Chapter 2, “[Strategies for Client Access to NetCache](#),” on page 17.

### Determining how many Web caches you need

No one formula exists to determine the number of Web caches that you need. Your Network Appliance sales engineer can help you determine the number of caches that are appropriate for your organization, weighing factors such as the following:

- ◆ Size of your organization, the number of branch offices the organization, and the distance between branch offices
- ◆ The number of Web requests that you expect would be sent to a particular NetCache appliance
- ◆ Whether you want the same NetCache appliance to handle traffic of multiple protocols

You can configure a NetCache appliance to operate in more than one mode—for example, as a Web cache and a news cache. However, if you expect your Web cache to handle a substantial load, for example, the volume of an ISP data center, Network Appliance recommends that you dedicate a NetCache appliance to Web caching. By doing so, you avoid conflicts between different software programs, and performance and reliability are better.

- ◆ Whether you can afford to have Web service unavailable, that is, whether you need a failover strategy

You might, for example, want to plan for one Web cache in addition to the number you think you need to handle the Web traffic. That way, if one Web

cache goes down, the remaining Web caches can still handle the volume of traffic. If you are deploying transparent proxying with an L4 switch, you might consider deploying a switch failover pair in case a switch fails.

**If your NetCache appliance is behind a firewall**

If your Web cache is behind a firewall, you might have to configure NetCache to enable it to send requests through the firewall, depending on the type of firewall you have.

Type of firewall	On the streaming media cache
Transparent	No explicit configuration is necessary to pass requests through the firewall
Nontransparent, running as a proxy	You must include the firewall in the cache hierarchy of the streaming media cache.

For more information about NetCache operation with firewalls, see Chapter 9, “[NetCache Deployment with Firewalls](#),” on page 225.

# Optimizing a Web site for caching

---

**Headers in HTTP 1.1** HTTP 1.1 allows for explicit control of content through HTTP headers (which is not possible with HTTP 1.0). By configuring HTTP headers on your Web server, you can influence how content is handled by proxy-cache servers. In general, current proxy-cache servers do not cache content if the content does not include appropriate headers. (NetCache includes controls you can use to prevent content from being cached even if an HTTP 1.1 header indicates it is cacheable; for example, you can prevent caching of a specific content type.)

## Static and dynamic content and cacheability

Because content cacheability is decided entirely by the HTTP headers that are returned in the response from the origin server, whether the content is static or dynamic is irrelevant for determining cacheability.

**Static content:** The majority of bytes (typically, 60 percent or greater) from Web sites are static (for example, images and Java applets). Although the HTML might be generated again for every request, many items within the page might be cached.

**Dynamic content:** Much content that is generated dynamically by a server (for example, database queries) is valid for some period of time and, therefore, should be cacheable. Typically, it is straightforward to design the server so that dynamic content can be cached. For example, a script that queries a database can send an explicit expiration header (usually after a *Content-Type* header) with the response. Some content must always be retrieved from the origin server, for example, bid prices for an online auction. (In the case of bid prices for an online auction, an appropriate *Cache-Control* header should indicate this fact.)

Making a URL cacheable even for only a short period can have a dramatic effect on the network. For example, if a particular URL is requested 100 times per minute, making this content cacheable for one minute could remove 99 percent of the load from the origin server if an accelerator is being used.

## For additional information

For detailed information about optimizing your Web site for Web caching services, see “Caching Tutorial for Web Authors and Webmasters” at [http://www.mnot.net/cache\\_docs](http://www.mnot.net/cache_docs). For example, this information includes recommendations for using cookies only when necessary, minimizing the use of SSL, and using consistent references to the same objects.

## Scenario: NetCache deployed in an enterprise environment

---

**About this scenario** This scenario shows a NetCache appliance deployed in a corporation that includes a remote office. This corporation handles Web requests only.

**Corporation's goals** The corporation's goals for using Web caching are as follows:

- ◆ Improve the quality of service because the corporation is interested in its employees being as productive as possible
- ◆ Reduce bandwidth requirements

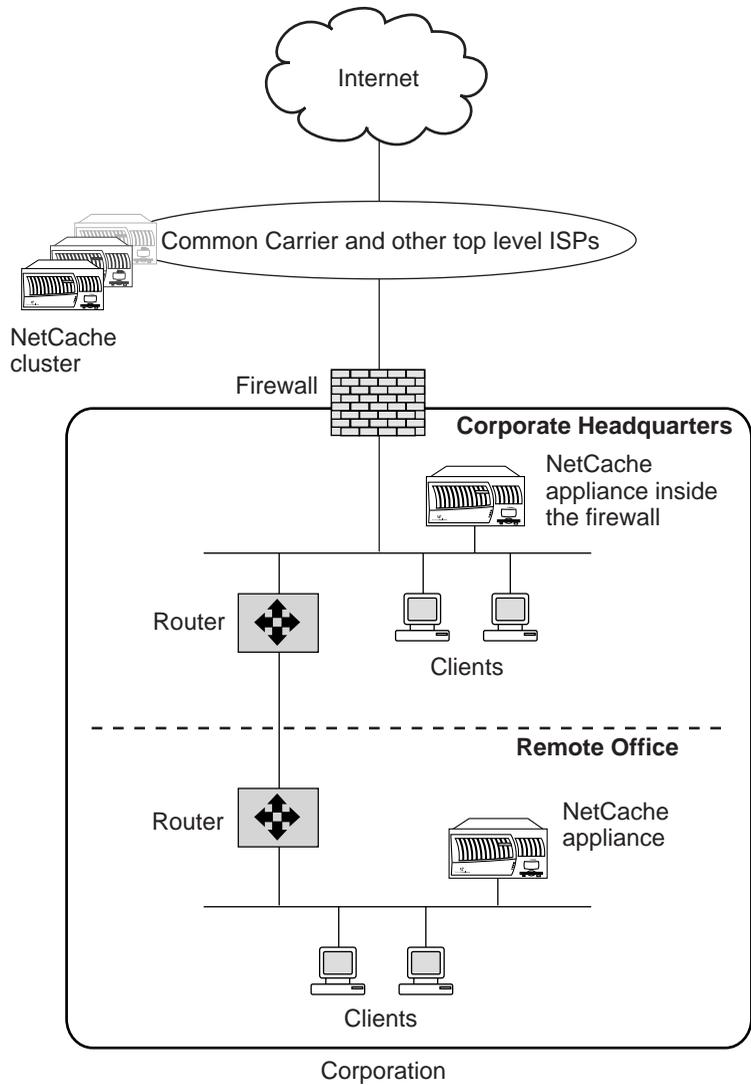
### Where NetCache appliances are deployed and why

**Deployment at the corporate headquarters:** A NetCache appliance was deployed at the corporate headquarters to improve the quality of service and to obtain bandwidth savings. Service is improved by deploying a NetCache appliance because after NetCache fetches a requested object from a Web server the first time, subsequent requests for the same object can be serviced from the cache. This results in a faster response rate than if every request were sent over the slower lines to the Internet. By not having every request sent to the Internet, the goal of reducing bandwidth usage was achieved.

**Deployment at the remote office:** The corporation added a NetCache appliance to the remote office to improve quality of service without having to incur the cost of improving the external network. If the corporation did not deploy a NetCache appliance at the remote office, the corporation would have to pay more for bandwidth to improve the quality of service at the remote office or accept slower responses to requests.

### Deployment illustrated

The following illustration shows the enterprise's environment with NetCache appliances deployed at both the corporate headquarters and the remote office.



# Scenario: NetCache deployed at a global carrier and an ISP

---

**About this scenario** This section contains examples of a total Web caching solution for a global carrier and an ISP.

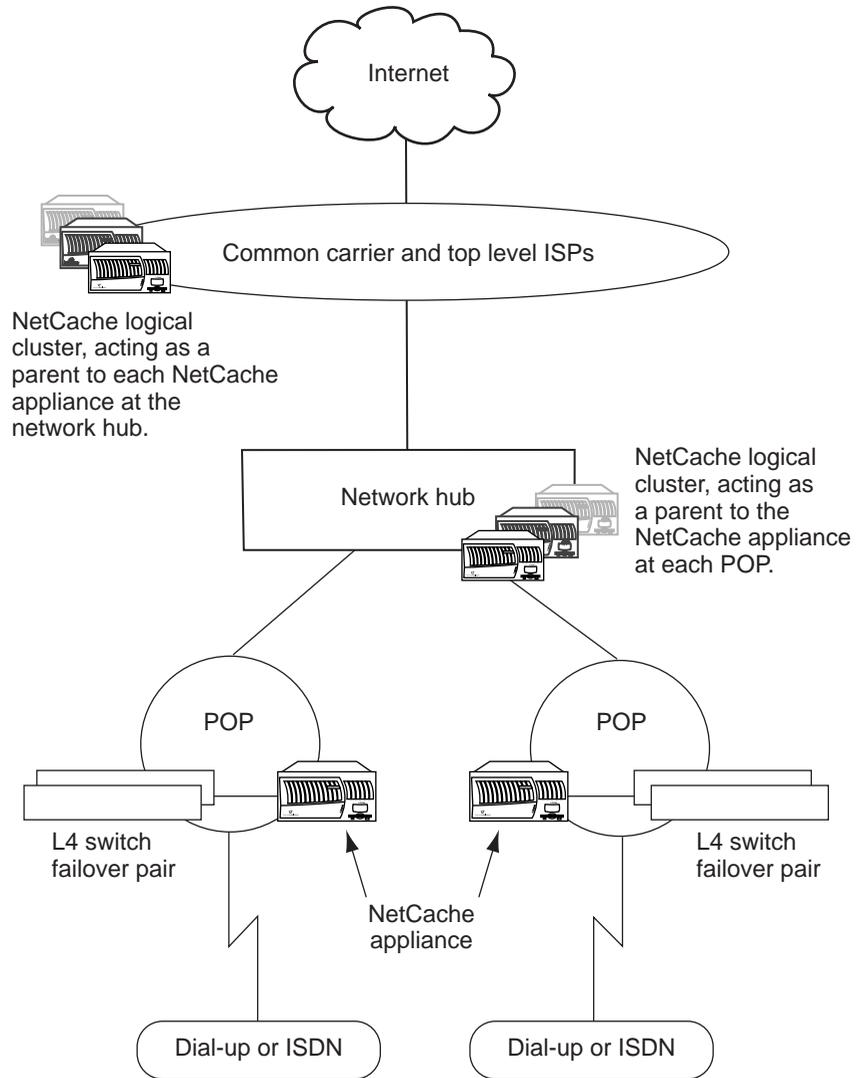
**Global carrier's and ISP's goals** The carrier is interested in caching for the following reasons:

- ◆ The carrier and the ISP want to reduce the use of bandwidth whenever possible.
- ◆ Fast response of Web pages to the customer provides a competitive advantage.

**Global carrier's and ISP's requirements** The global carrier's and ISP's requirements for caching are as follows:

- ◆ Caching must be invisible to the users; it is not practical to expect the users to configure their Web browsers to send traffic to the Web cache at the ISP.
- ◆ The caching system must be reliable. Service outages not only inconvenience customers, but the company reputation suffers because the outages are given a good deal of negative attention from the media.

**Example 1: all users handled by POPs** The following illustration shows a deployment in which all users are handled by points of presence (POPs).



Details of the preceding illustration are as follows:

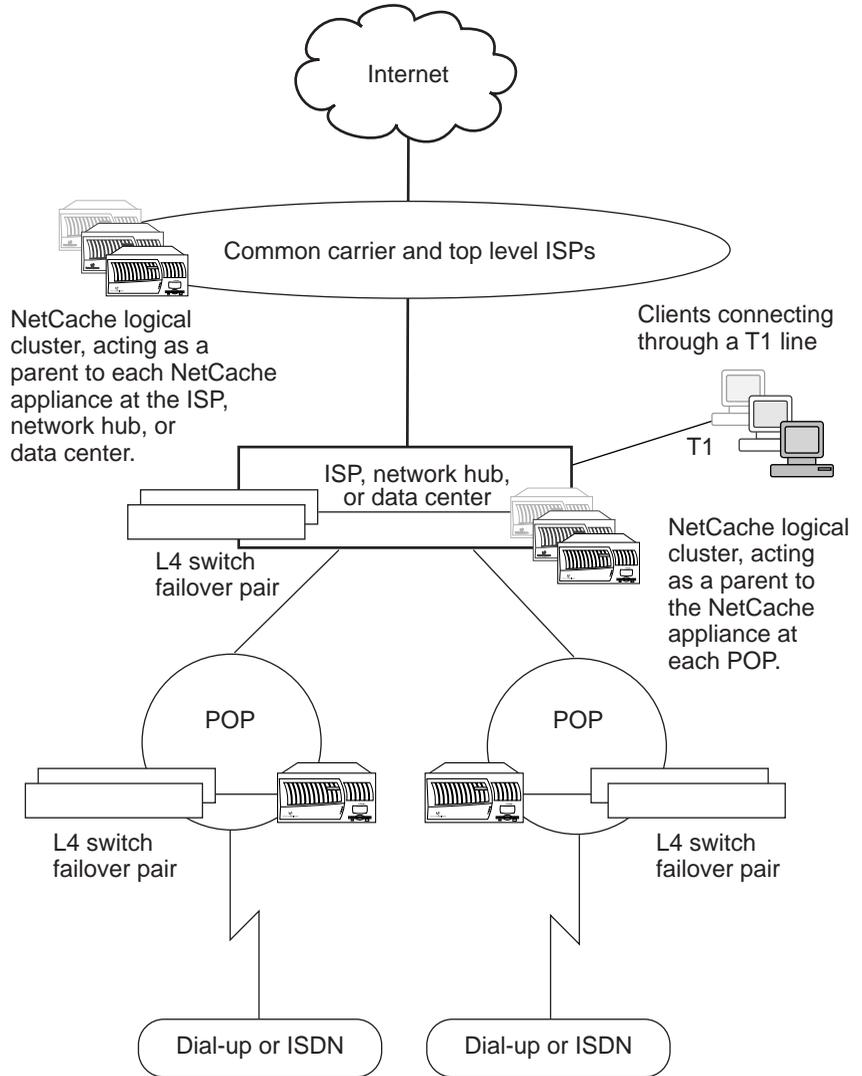
- ◆ By adding NetCache appliances at each level (POP, network hub, carrier), copies of frequently used Web objects are stored locally, thereby reducing network latency and bandwidth consumption. If the object is not in the cache at one level, a NetCache appliance can try to obtain the object from a cache at the next level up.

- ◆ To fulfill the requirement that caching must be invisible to the user, transparent proxying is deployed at each POP.
- ◆ A pair of L4 switches, at the same level, were added to the network at the POP to provide redundancy in case one switch fails. This helps to fulfill the requirement for reliability. (A WCCP router could be used instead of L4 switches.)
- ◆ It is not necessary to deploy transparent proxying at the network hub because all users connect to the POP, where transparent proxying is implemented.
- ◆ Because the network hub is to handle cache misses at the POP, the NetCache appliance at each POP is configured to view the NetCache appliances at the network hub as a cluster in a logical hierarchy. This configuration enables the NetCache appliances at the POPs to send the requests they cannot resolve to the NetCache appliances at the network hub and distribute the requests among them. See Section A, “[Request resolution hierarchies](#),” on page 70 for more information.
- ◆ There is no advantage to deploying transparent proxying at the carrier level because transparent proxying is deployed at the POP level. Instead, the NetCache appliances at the network hub are configured to view the NetCache appliances at the carrier level as a logical cluster. The result is that the network hub’s NetCache appliances can distribute requests they cannot resolve across the carrier’s appliances.

See Section A, “[Request resolution hierarchies](#),” on page 70 and the *Administration Guide* for more information about clusters in hierarchies.

**Example 2: some clients connect directly to the ISP, network hub, or data center**

In this example, some clients connect to the POPs. Other clients connect to the ISP, network hub, or data center through a T1 line.

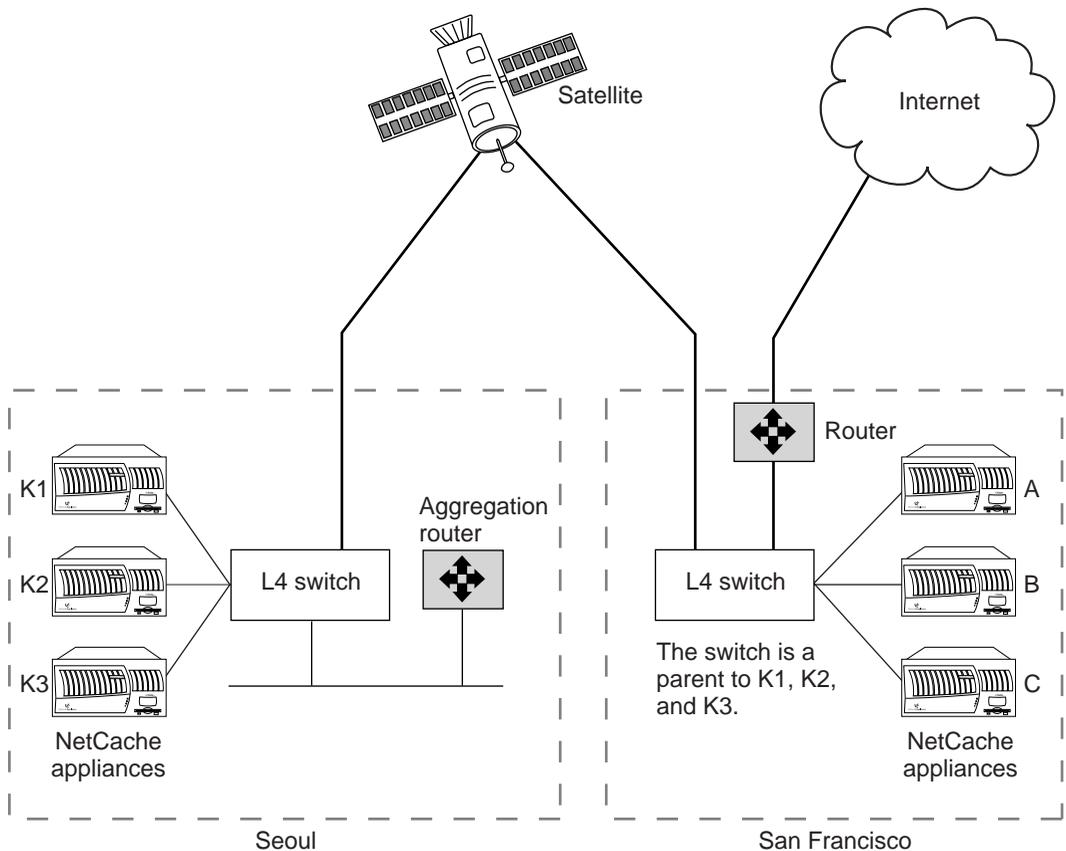


The difference between this deployment and the deployment in Example 1 is that transparent proxying is deployed at the ISP, network hub, or data center level to provide transparent services to the clients connecting over the T1 line. The L4 switches are deployed in a pair also, for redundancy, but this strategy is not required. A WCCP router could be used instead of L4 switches.

# Scenario: NetCache deployed with high-latency, high-bandwidth links

---

- About this scenario** This company has a site in Seoul, South Korea, and another in San Francisco, California. These two sites are connected by a satellite (high-latency) link. The Seoul site accesses the Internet through the San Francisco site.
- This company provides Web caching service only.
- Goal for deploying NetCache appliances** NetCache appliances are deployed on each side of the satellite link. The goal is to optimize TCP transmission over the satellite link to achieve persistent connections and higher throughput.
- Deployment Illustrated** The following illustration shows the deployment of NetCache appliances in this multisite company.



### Optimization of TCP connections over the satellite link

TCP connections over the satellite link were optimized as follows:

- ◆ NetCache appliances were deployed at each end of the satellite link.
- ◆ Cache-to-cache communication over the link was set up by specifying in the configuration of each NetCache appliance in Seoul that the L4 switch in San Francisco is a parent. Cache-to-cache communication over the satellite link keeps the TCP connection open. The benefits of this persistent connection are as follows:
  - ❖ TCP connection setup overhead is reduced.
  - ❖ TCP connection setup is reduced, eliminating the needless use of slow start. The result is that response time is significantly increased.

Because NetCache implements selective acknowledgments, NetCache can better handle dropped segments, resulting in increased throughput.

### **How higher throughput was achieved**

Higher throughput was also achieved by configuring TCP large windows on each NetCache appliance at each site. There are two ways you can achieve higher throughput when configuring TCP large windows in the Appliance Manager:

- ◆ On all the NetCache appliances at one site, when configuring TCP large windows, specify the subnet on which the appliances at the other site are located and the window size. Repeat this process on the NetCache appliances at the other site.
- ◆ On all the NetCache appliances at one site, when configuring TCP large windows, specify each NetCache appliance at the other site and the window size.

The advantage of this method is that you enable TCP large windows only for the traffic being sent over the satellite link between the caches. If you specify a subnet, you enable TCP large windows for any other devices on the subnet that use TCP large windows.

### **Limitations of this type of deployment**

The combination of persistent connections and TCP large windows works well when the load over the satellite link remains fairly low. The use of TCP large windows is problematic when a large number of connections between NetCache appliances is required. The reason is that each connection can use a large amount of memory. NetCache safeguards against this problem by establishing an upper limit on the number of TCP large window connections.



**About this chapter** This chapter describes deploying a NetCache appliance as a streaming media cache.

**Chapter contents** This chapter contains the following sections:

- ◆ Section A, “[Streaming media basics](#),” on page 100
- ◆ Section B, “[Streaming media service with NetCache](#),” on page 110
- ◆ Section C, “[Deployment considerations](#),” on page 126
- ◆ Section D, “[Deployment scenarios](#),” on page 141

## Section A: Streaming media basics

---

**About this section** Information in this section provides the background to understand information in the rest of the chapter about NetCache support for streaming media and considerations for deploying streaming media.

**Contents of this section** This section contains the following topics:

- ◆ [“Overview of streaming media”](#) on page 101
- ◆ [“Streaming media and bandwidth”](#) on page 104
- ◆ [“Transmission methods for streaming media delivery”](#) on page 106

## Overview of streaming media

---

### What is streaming media?

*Streaming media* is a term used to describe media files that are served in discrete paced individual packets rather than in bulk, playing while they are being transmitted over the network to the media player on the client computer. In contrast, conventional Web files, which are downloaded through a file transfer, must be downloaded entirely before the user can view them. Commonly requested types of streaming media are video and audio. Streaming media also includes interactive media, cartoon-like animations, panoramic data, and more.

### Live versus on-demand streaming media

Streaming media is delivered in the following ways:

- ◆ Live media streams  
Live media streams occur in real time, like the news program that you watch on your television set. Some organizations record a live media stream and then broadcast the media stream to their employees or customers at a specified time. All users who have requested the media stream see the same media stream at the same time. Users are not able to rewind or fast-forward the media stream.
- ◆ On-demand, or previously recorded, media streams  
Users can request these on-demand media streams at a time most convenient to them. Users can rewind the media and fast-forward on-demand media streams. On-demand streaming content is commonly referred to as VOD (video on demand).

NetCache supports both of these types of streaming media.

### Streaming media presentation versus a unique stream

Discussions in this chapter refer to *streaming media presentations* and *unique streams*.

**Streaming media presentation:** *Streaming media presentation* is a general term used to describe the delivery of live or on-demand streaming media. Multiple *unique streams* can make up a streaming media presentation.

**Unique stream:** Each stream has the following characteristics:

- ◆ Bandwidth speed
- ◆ Media type, for example, audio or video
- ◆ Thinning parameters

The streaming server drops a consistent number of frames to try to make the stream that is being delivered to the client one that the client can handle.

A stream is *unique* if one or more of the characteristics in the previous list are different from another stream that is part of the same streaming media presentation.

For example, even though two clients request the same streaming media presentation, if one client notifies the server that it requires a 28.8-kbps video stream and another client notifies the server that it requires a 56-kbps video stream, the two clients are requesting *different* streams. Similarly, a 56-kbps video stream is different from a 56-kbps audio stream.

---

**Note**

When a content creator creates a streaming media presentation, the creator can encode multiple bit rates. One advantage of providing multiple bit rates for a streaming media presentation is that clients can negotiate with the streaming server for a stream at a bit rate that the client's network connection can handle. Another advantage of a multi-bit rate file is that only a single URL needs to be published. Conversely, if multiple single-bit rate files are published, a separate URL must be published for each file.

---

**Encoded bit rate can differ from delivery bit rate**

Content authors normally encode streaming media content into different bit rates to meet the needs of the different speeds of Internet access—modem, ISDN, DSL, and LAN. In contrast, the delivery bit rate is the actual speed at which the content is delivered to the client. For example, a stream encoded for playback at 56 kbps must be delivered to clients at a bit rate of 56 kbps or higher. A client with enough bandwidth might ask the streaming server to send the 56-kbps encoded stream at 220 kbps; the data is buffered locally and played back at 56 kbps. The playback experience of 56-kbps stream delivered at 220 kbps would be better at 220 kbps than at 56 kbps. The reason is that more time is available for the client to request packets to be retransmitted if packets are dropped.

**Clients and servers actively respond to changing network conditions**

During streaming media presentations, the client and server communicate quality of service (QOS) information to maintain an acceptable level of playback quality. A variety of QOS mechanisms are used by streaming media vendors:

- ◆ If data is arriving too slowly, clients might request a higher delivery bit rate.
- ◆ If data is arriving too quickly, clients might request an entirely different bit rate to match the current network conditions. This process is called *stream switching*.

- ◆ If clients perceive network congestion, they might request thinned streams.

**Note**

---

Streaming media vendors support varying combinations of the preceding QOS mechanisms.

---

Thinning and stream switching can occur several times during playback.

## Streaming media and bandwidth

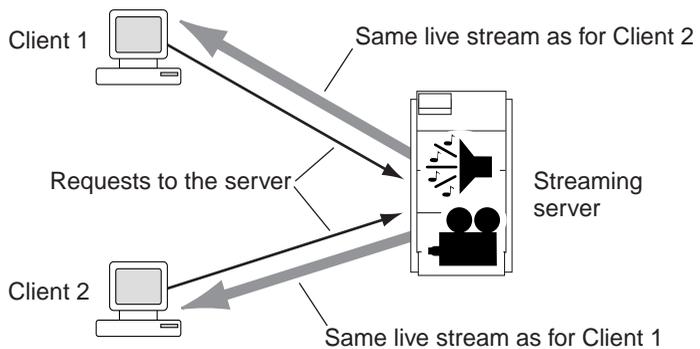
---

### High bandwidth use with streaming media

Video, audio, and other streaming media use a considerable amount of bandwidth—*much* more than the amount of bandwidth needed for Web and news traffic. For example, a media stream could require 10 KB each second, whereas a Web page that the user views for 10 seconds could require 10 KB.

The following illustration shows the flow of requests and media data that a streaming server sends back in a typical server-client streaming model for live streaming media presentations.

Typical server-client streaming model for live broadcasts



As the previous illustration shows, in the typical streaming server-client model, the streaming server sends a separate copy of the media stream to each client that requested the *same* unique stream.

Because streaming media uses a considerable amount of bandwidth, delivering multiple copies of the same media data between the streaming server and the clients can cause significant network and server congestion. The more clients that request the same media stream, the more bandwidth is used.

### Effect of high bandwidth use on the quality of streaming media

Planning for efficient bandwidth use is a particularly important issue for streaming media. The reason is that bandwidth use has a direct correspondence to the quality of the media streams that are delivered to the clients. If your network is congested, your users are likely to experience problems such as jagged video,

patchy audio, and unsynchronized video and audio as packets are dropped or arrive late. Conversely, the more bandwidth that is available, the better the quality of media streams. See “[Considerations for bandwidth](#)” on page 127 for more information.

## Transmission methods for streaming media delivery

---

### About this section

This section describes the two transmission methods for streaming media—unicast and multicast. One of your planning tasks for streaming media is to determine whether unicast, multicast, or both transmission methods are best suited for your organization. This section provides a high-level comparison of the two transmission methods.

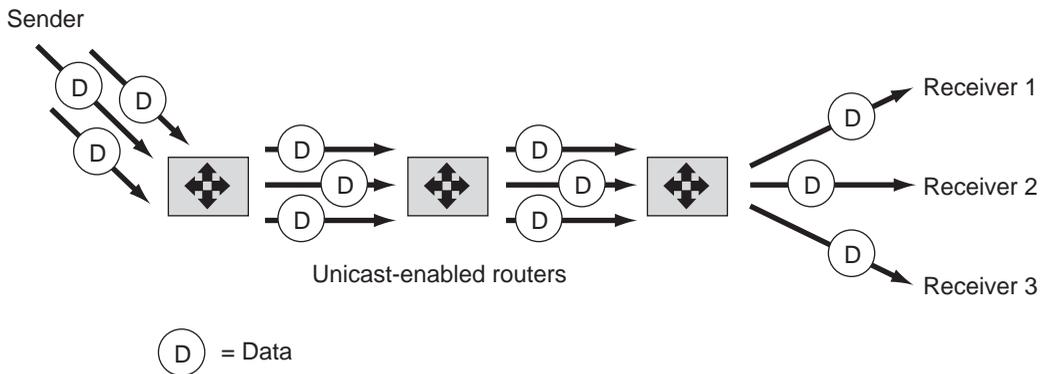
### Feature summary for transmission methods

The following table provides a high-level comparison of unicast and multicast transmission.

Element	Unicast	Multicast
Connections	One-to-one transmission	One-to-many transmission
Transport	Any of TCP, UDP, HTTP	IP multicast channel
Efficiency	Not as efficient as multicast for delivering streaming media to a large number of clients	More efficient than unicast for delivering streaming media to a large number of clients
Type of stream	VOD or live streams	Live streams only
Devices	The network devices use unicast	The network devices through which multicast transmissions are to be routed must support multicast.

### Unicast described

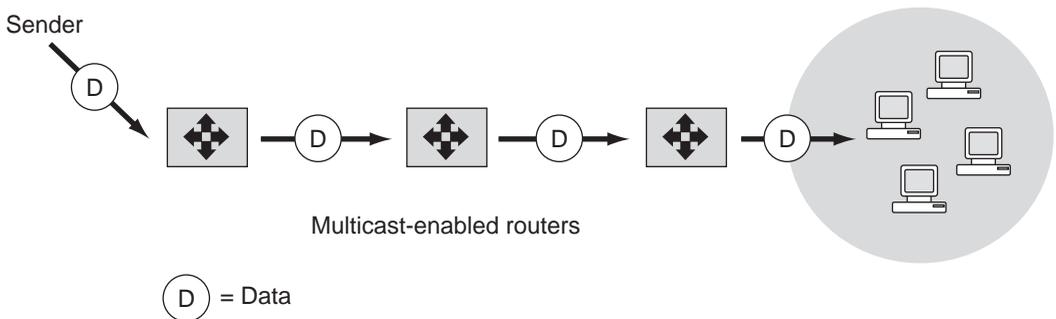
As the following illustration shows, unicast transmission is point-to-point; that is, a separate copy of content is sent to each recipient that has explicitly requested the content.



“[NetCache support for live streams over unicast](#)” on page 115 describes how NetCache optimizes transport of requests over unicast.

### Multicast described

Multicast transmission is analogous to a radio frequency on which any device can listen. Any device that supports multicast can transmit on the multicast channel. One copy of the data is sent to a group address. Devices in the group listen for traffic at the group address and join the stream if clients in the routing tree are requesting the stream. Only the group participants receive the traffic at the address associated with the group. Broadcasts differ from multicast because broadcast traffic is sent to the entire network.



For multicast transmission to occur, the network devices through which the content is to be sent must support multicast. For example:

- ◆ Content creators must explicitly set up their streaming servers to support multicast.

For example, for MMS, content creators can set up multicast-enabled stations, stations that are not multicast-enabled, or both. For RealNetworks,

the configuration of the server includes specifying whether the server supports multicast and, if so, which clients (subnets) can use multicast.

- ◆ Routers on the path must support multicast.
- ◆ Clients must request a multicast transmission. Media players that are set for multicast transmission simply join the multicast channel to receive the streaming data, sometimes without establishing an explicit one-to-one connection to the device sending the transmission.

**Benefits of multicast:** The benefits of using multicast for streaming media include the following:

- ◆ It alleviates network congestion.
- ◆ For live streaming events that have a large audience, multicast significantly reduces network traffic compared to the traffic that would result from transmitting the same live event over unicast. If unicast transport is used, the same content must be sent across the network multiple times or it must be broadcast to all devices on the network.
- ◆ It scales well as the number of participants expand.
- ◆ It is well suited for efficient transmission over satellite links.

A company might, for example want to reserve WAN connections for business-critical traffic, such as stock trades, but it needs a way to deliver corporate broadcasts. The company could efficiently transmit corporate broadcasts over satellite by using multicast transmission and reserve the WAN for business-critical traffic.

- ◆ It enables network planners to proactively manage network growth and control cost because deploying multicast is more cost-effective than alternatives for increasing LAN and WAN capabilities.

**Limitations of multicast:** The limitations of multicast include the following:

- ◆ In general, multicast support is not yet available on the Internet. Therefore, using multicast to deliver content is limited to intranet-style deployments. However, "[Scenario: Multicast support for transmission over a satellite link](#)" on page 160 describes how you can use NetCache multicast support to convert unicast traffic for live streams to multicast.
- ◆ Not all networking equipment supports multicasting. In addition, not all network administrators enable the multicast functionality of their networking equipment.

**Multicast transmission and switches:** Switches do not understand multicast. When a multicast stream reaches a switch, the switch sends the multicast stream to all of its ports. A switch treats a multicast address as an Ethernet broadcast.

**Details about  
NetCache support  
for unicast and  
multicast**

Details about NetCache support for live streams over unicast and multicast and on-demand streams over unicast are provided in the following sections:

- ◆ [“NetCache support for live streams over unicast”](#) on page 115
- ◆ [“NetCache support for live streams over multicast”](#) on page 120
- ◆ [“NetCache support for on-demand streams”](#) on page 123
- ◆ Section D, [“Deployment scenarios,”](#) on page 141

## Section B: Streaming media service with NetCache

---

**About this section** This section describes NetCache support for streaming media and the benefits of configuring a NetCache appliance as a streaming media cache.

**Contents of this section** This section contains the following topics:

- ◆ [“Overview of NetCache as a streaming media cache”](#) on page 111
- ◆ [“Overview of NetCache support for live streams”](#) on page 114
- ◆ [“NetCache support for live streams over unicast”](#) on page 115
- ◆ [“NetCache support for live streams over multicast”](#) on page 120
- ◆ [“NetCache support for on-demand streams”](#) on page 123

## Overview of NetCache as a streaming media cache

---

### What is a NetCache streaming media cache?

Network Appliance uses the term *streaming media cache* to describe a NetCache appliance that is configured to handle any or all of the following streaming media protocols:

- ◆ Microsoft Media Streaming (MMS) for Windows Media, from Microsoft Corporation
- ◆ Real Time Streaming Protocol (RTSP) for streaming, for example, QuickTime from Apple Corporation, RealSystem from RealNetworks

You can configure a streaming media cache for DNS caching also.

When a streaming media cache is deployed between clients and a streaming media server, streaming requests that would otherwise have been sent directly to the streaming server are sent to the cache.

You must obtain a license from Network Appliance for each streaming media service you want to run—RTSP for RealNetworks, RTSP for QuickTime, and MMS.

#### Note

---

NetCache can also be configured as a streaming accelerator to cache and split content from one or more *streaming servers* that you identify and provide that content to clients that request it. In contrast, when NetCache runs as a streaming media cache, it acts as an agent for the browser, requesting streaming media content for clients and caching and splitting that content so that it can be delivered directly to remote offices. See Chapter 6, “[Deploying NetCache as an Accelerator](#),” on page 167 for more information.

---

### Support for live and on-demand streaming media presentations

A streaming media cache handles both live streams and on-demand streams. See “[Overview of NetCache support for live streams](#)” on page 114 and “[NetCache support for on-demand streams](#)” on page 123 for information about the differences in how NetCache handles these two types of streaming media presentations.

## Benefits of adding a streaming media cache to the network

Adding a streaming media cache to the network can benefit you in the following ways:

- ◆ Bandwidth savings

NetCache can help you save bandwidth in the following ways:

- ❖ By adding a streaming media cache to the network, you can significantly reduce the number of round trips to the streaming server for live streaming media presentations.
- ❖ By deploying streaming media caches close to users, you can reduce the number of requests that are sent across the network to the streaming server.

For example, if your organization is large or spans multiple regions, you can save considerable internal bandwidth by deploying your streaming media caches in separate locations, close to users. The result is that fewer requests are sent across the network to the streaming server.

---

### Note

You can use the NetCache bandwidth allocation feature to allocate a specific amount of bandwidth to a particular protocol.

---

- ◆ Improved quality of streaming media delivered to clients

Streaming media is time dependent and sensitive to changing network conditions. The closer the streaming media content is to clients, the better the playback experience will be because packets are more likely to be delivered at the constant rate required for smooth playback.

- ◆ Might eliminate the need to purchase additional streaming servers

If you own the streaming server, deploying a streaming media cache might eliminate the need to purchase additional streaming servers as the volume of streaming traffic increases. The reason is that clients connect to the streaming media cache instead of to the streaming server.

A streaming server can handle only a finite number of connections. By deploying a streaming media cache, you can off-load connections that your streaming servers would have had to handle.

## Supported media player versions

NetCache supports the following media player versions:

- ◆ RealNetworks RealPlayer and later
- ◆ Microsoft Windows Media Player 6 and later
- ◆ Apple QuickTime 4 and later

**Client logging information provided by NetCache**

Many streaming media content providers require that clients send logging information back to the streaming server so that information can be collected for billing, advertisement revenue, performance analysis, and other purposes. When configured to do so, a streaming media cache forwards client logging information to the streaming server for every client that received a stream.

For MMS, NetCache tracks logging information and sends logging packets to the Windows Media server even if Windows Media media players crash. This type of functionality is not available for RealNetworks.

**If a streaming server requires client authentication**

NetCache does not perform authentication for streaming clients. If a streaming server requires client authentication, NetCache establishes a connection to the streaming server and forwards the authentication information given by the client.

NetCache forwards the authentication information for *every* client—even if NetCache can resolve a request.

**Note**

---

NetCache does not split RealNetworks streams that require user authentication. Every RealNetworks authenticated stream must be treated as a unique stream because of pay-per-view, pay-per-time, and other commerce restrictions.

---

## Overview of NetCache support for live streams

---

### Single connection to the streaming server per unique live stream

Rather than fetching streaming data from the streaming server for each client request (as discussed for the typical streaming client-server model), the streaming media cache connects to the streaming server only once for each *unique* live stream. Therefore, the number of connections that the streaming media cache makes to the streaming server for live streaming media presentations is significantly reduced.

### Delivery of streams to clients

How NetCache returns live streams to clients depends on whether the delivery method is over unicast transport or multicast transport. NetCache supports unicast by default. When you plan your streaming media deployment, you need to determine whether you want your NetCache appliance to support multicast. However, even if you have configured a NetCache appliance for multicast, NetCache uses unicast transmission if a device or network to which it is to transmit cannot handle multicast.

Read the following sections to help determine whether setting up your NetCache appliance for unicast, multicast, or both is appropriate for your organization.

- ◆ [“NetCache support for live streams over unicast”](#) on page 115
- ◆ [“NetCache support for live streams over multicast”](#) on page 120
- ◆ [“Considerations for deploying NetCache multicast support”](#) on page 130

## NetCache support for live streams over unicast

---

### About this section

The previous section describes how NetCache makes a single connection to the streaming server for each unique live stream. This section describes the interaction between a NetCache appliance and clients when NetCache uses unicast transmission (TCP, UDP, or HTTP).

The behavior described in this section is the NetCache default behavior. NetCache support for multicast, which is described in “[NetCache support for live streams over multicast](#)” on page 120, must be explicitly configured on a NetCache appliance.

### When NetCache uses unicast transmission

Unicast transmission is the NetCache default behavior. NetCache uses unicast transmission to send content to the client network under the following circumstances:

- ◆ You have not configured NetCache for multicast.
- ◆ Clients request unicast transmission.
- ◆ A device or network does not support multicast.

Even if you configure a NetCache appliance for multicast, if a device or network to which NetCache is to transmit the content does not support multicast, NetCache uses unicast to transmit the data to the client network. A single NetCache appliance might, therefore, send some content to the client network over unicast and send some content over multicast, depending on the devices on the network and the transmission requested by clients.

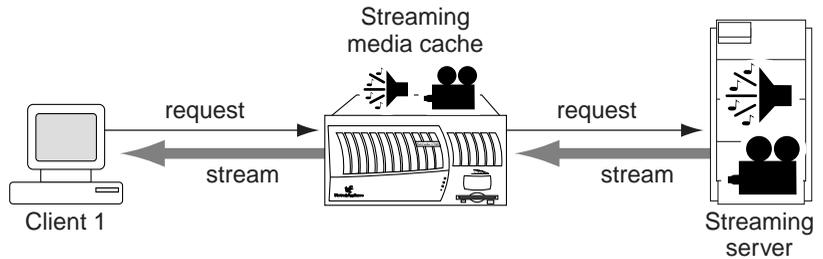
Even if NetCache sends content to clients by using unicast transmission, NetCache helps you conserve bandwidth for live streams. The reason is that NetCache conserves upstream bandwidth because it makes only a single connection to the origin server on behalf of multiple clients that request the same stream.

### Request flow with unicast transmission

In the unicast streaming model, NetCache receives a single copy of the stream from the server and splits it when sending it out to clients. Therefore, NetCache conserves downstream bandwidth by replicating the stream to multiple downstream clients.

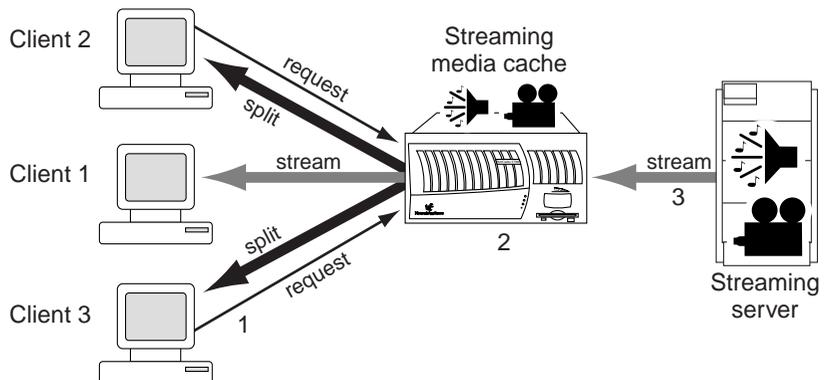
A streaming media cache handles the *first* request for a live media stream differently than it handles subsequent requests for the same media stream.

**Path of the first request for a live stream:** The following illustration shows what happens when the *first* client makes a request for a particular stream.



The streaming media cache intercepts the streaming request sent by the client. Because the streaming media cache is not already delivering the stream to another client, the cache connects to the streaming server to obtain the requested stream.

**Path of subsequent requests for the same unique stream:** The following illustration shows the path for additional requests for the *same unique* stream that Client 1 requested.



-  Request to the cache
-  The stream that was returned from Client 1's request
-  The stream split for Client 2 and Client 3

The flow of requests and streams in the preceding illustration is as follows:

1. Assume that Client 1 already requested a stream from the streaming server. Client 2 and Client 3 now request the same unique stream that Client 1 requested (as shown in the illustration of the path of the first request).
2. Because the streaming media cache is already delivering the requested stream to Client 1, the cache *splits* the stream and delivers the same unique stream to Client 2 and Client 3. The stream is split without impacting the quality of the stream.

If Client 2 or 3 had requested a different unique stream than Client 1 requested, the streaming media cache would have sent that unique request to the streaming server, as it did for Client 1.

3. The streaming media flow that is shown between the streaming server and the streaming media cache was initiated by the request from Client 1. No connection to the streaming server is necessary after the streaming media cache connects to the server for the first request for a unique stream.

At first glance, the streaming media cache-client model resembles the server-client model illustrated in “[High bandwidth use with streaming media](#)” on page 104. However, in the streaming media cache-client model in the previous illustration, most of the streaming activity takes place between the client and the cache, which is closer to the clients than the streaming server. The cache connects to the streaming server only once for each unique stream, thereby conserving bandwidth. Additionally, when the stream is served from a source close to the client, quality is higher because less packet loss occurs, and packet retransmissions occur quickly because client-to-cache latency is low.

## Transports for unicast

NetCache supports TCP, UDP and HTTP for unicast transmission.

**NetCache support for TCP and UDP:** NetCache supports both TCP and UDP from the client to the streaming media cache. Many streaming media administrators prefer to use UDP because it is the most efficient and provides the best stream quality. The use of UDP depends on whether the client media player is configured to use UDP and whether the firewall is configured to allow UDP ports. See “[Considerations for firewalls and streaming media service](#)” on page 137 for more information.

**NetCache support for HTTP:** Although HTTP is not usually associated with streaming media, HTTP can be used as a streaming media transport. HTTP use for streaming media is necessary in circumstances such as the following:

- ◆ When the streaming protocols are blocked, for example, if the firewall administrator does not open ports for streaming media on the firewall (see [“Considerations for firewalls and streaming media service”](#) on page 137 for information about NetCache HTTP support for streaming media and firewall setup)
- ◆ If a client media player is set for HTTP rather than TCP, UDP, or multicast

In these cases, NetCache uses HTTP encapsulation; that is, NetCache tunnels a streaming media protocol over HTTP. NetCache can use HTTP encapsulation in the following ways:

- ◆ **Client-side HTTP encapsulation**  
When the client tries to contact the streaming server using HTTP, NetCache can fetch the content from the server over either TCP or HTTP.
- ◆ **Server-side HTTP encapsulation**  
NetCache can use HTTP to retrieve streaming content from the server under the following circumstances:
  - ❖ The NetCache appliance is behind a firewall that blocks TCP or UDP traffic.
  - ❖ The streaming server is listening only on an HTTP port.NetCache performs server-side HTTP encapsulation for Windows Media, RealNetworks, and QuickTime clients.

---

**Note**

If you have not installed a streaming license or if streaming is disabled, NetCache will proxy streaming traffic over HTTP. NetCache does not cache or split HTTP-proxied streaming traffic.

---

**About setting up an L4 or L7 switch to support HTTP encapsulation:**

The HTTP encapsulation protocols for Windows Media, RealSystem, and QuickTime require that multiple HTTP connections from the client are directed to the same server (or proxy-cache server). Therefore, you must configure your L4 or L7 switch’s load balancing algorithm accordingly. A hash based on client IP address, server IP address, or both works but round-robin or least connections does not.

**About stream  
upgrade requests**

NetCache supports clients that request a change from one unique stream to another during a specific streaming media presentation, for example, a request for an upgrade from 28 kbps to 56 kbps. NetCache intercepts a request for a stream upgrade. If the streaming media cache is already delivering that upgrade stream to a client, the cache splits the stream to resolve the upgrade request. The streaming media cache does not need to open a new connection to the streaming server if the streaming media cache is already delivering the stream.

## NetCache support for live streams over multicast

---

### About this section

This section describes NetCache support for multicast. You must explicitly configure a NetCache appliance for multicast. Unicast support, as described in [“NetCache support for live streams over unicast”](#) on page 115, is the NetCache default behavior.

### NetCache support for multicast

NetCache supports both multicast in and multicast out. You can configure the same NetCache appliance for both multicast in and multicast out, or for only one of these settings. The following table describes multicast in and multicast out.

Support for...	Description
multicast in (or input)	NetCache receives a live stream through the multicast transport from a streaming server that is configured to support multicast streaming.
multicast out (or output)	NetCache sends the live stream to clients through the multicast transport, regardless of how the stream was received from the streaming server.  NetCache uses multicast when the next device on the hop to the client supports multicast and the client requesting multicast transmission supports multicast. A single NetCache appliance might, therefore, send some content to a client network over unicast and some content over multicast, depending on the devices on the network and the transmission requested by clients.

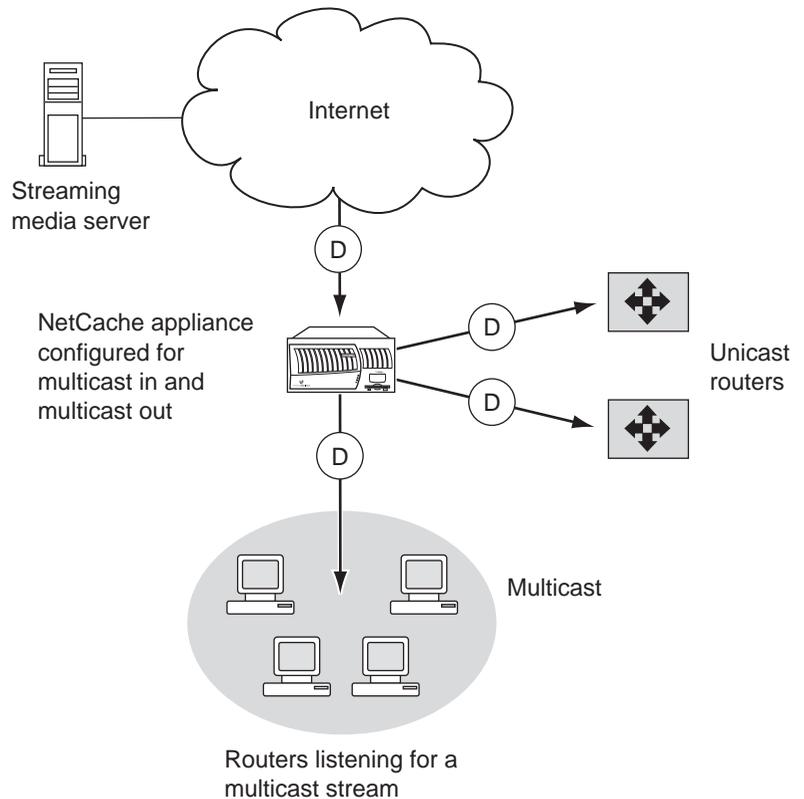
A stream does not have to travel over multicast the entire distance from the origin server to the client. NetCache can

- ◆ Receive a stream from the server over unicast (default behavior) and send the stream to the client network over multicast  
In contrast, multicast-enabled routers cannot rout unicast transmissions.
- ◆ Receive a stream from the server over multicast and send the stream to the client network over unicast

- ◆ For RealPlayer connections, receive a stream from the server over multicast and send it to the client network over multicast

NetCache behavior depends on how you have configured the appliance and whether origin servers, client networks, and clients support multicast.

The following illustration shows a simplified concept of stream transmission with multicast in a deployment with a streaming media cache.



(D) = One copy of the stream

**Transmission from the media server to the NetCache appliance:** If the streaming media server sends the content over multicast, a NetCache appliance configured for multicast input joins the multicast channel to receive the stream. By default, a NetCache appliance can receive unicast transmissions.

**Transmission from the NetCache appliance to clients:** If the NetCache appliance is configured for multicast output, NetCache sends one copy of the stream to each multicast-enabled network on which the clients reside. Each router on the network that listens for multicast transmissions determines whether to split the copy or to pass the single copy further on the network. Therefore, multicast-enabled routers are responsible for splitting at appropriate points in the routing tree, not the NetCache appliance. Clients configured for multicast also listen on the multicast channel.

As the previous illustration shows, the multicast router sends only one copy of the data to the network of multicast-enabled clients. However, the multicast router must send a separate copy of the data to each client whose media player is not set for multicast and to each network that is not multicast-enabled.

If the network on which the clients reside does not support multicast, NetCache splits the stream, as described in [“NetCache support for live streams over unicast”](#) on page 115.

**Type of multicast that NetCache supports**

NetCache supports the types of multicast shown in the following table.

For..	NetCache supports...
RealNetworks multicast	<p>Back-channel multicast.</p> <p>Back-channel multicast uses a control data connection between the server and the client in order to track information, such as client setup and client logging.</p> <p>NetCache does not support scalable multicast for RealNetworks.</p>
QuickTime multicast	<p>Back-channel multicast.</p> <p>NetCache does not support scalable multicast for QuickTime.</p>
MMS multicast	<p>Scalable multicast.</p> <p>Scalable multicast does not use a control connection. Media player clients receive streaming parameters through a nonstreaming protocol, such as HTTP.</p>

## NetCache support for on-demand streams

---

### Transport

Transmission for on-demand streams is over unicast. Multicast transmission applies only to live streams.

### Caching on-demand streams

A streaming media cache caches on-demand streaming media. When streaming media caches are deployed near clients, the number of connections to the streaming media server is significantly reduced because NetCache can deliver on-demand streams directly to clients.

### Types of on-demand content that NetCache caches

A streaming media cache caches on-demand streams from the following vendors:

- ◆ Microsoft (Windows Media)
- ◆ RealNetworks (RealMedia)
- ◆ Apple (QuickTime)

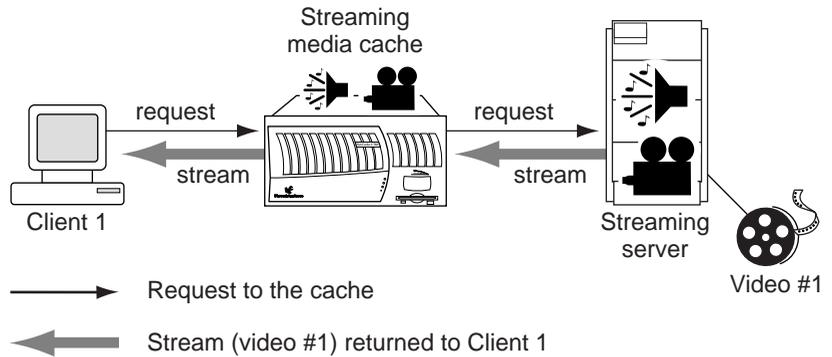
### Overview of on-demand streaming media request flow

When an on-demand stream is requested by a client, NetCache checks to see if the stream is already in the cache. If the stream is in the cache, NetCache serves it directly to the client from the cache. If the stream is not in the cache, NetCache fetches the on-demand stream from the streaming server and delivers the stream to the client while caching it for future clients.

### Path taken by a request for on-demand streaming media

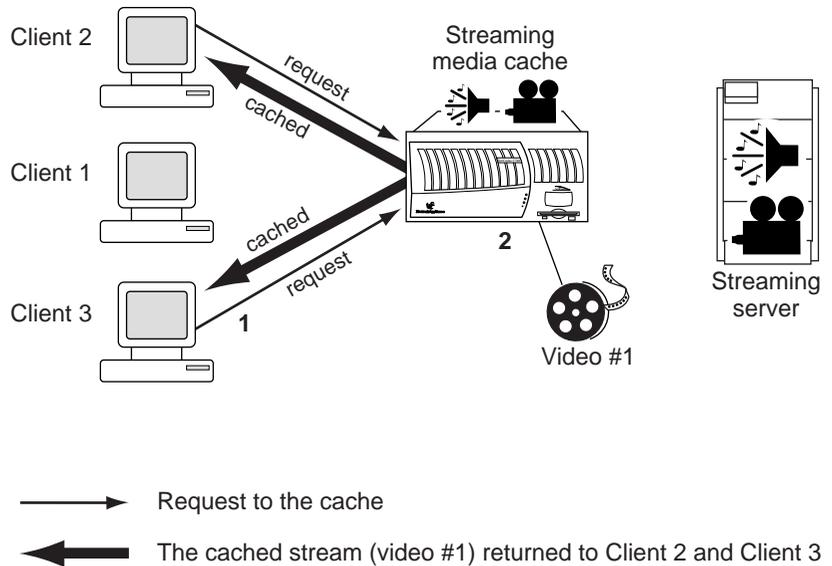
NetCache handles the first request for an on-demand stream differently than it handles subsequent requests for the same on-demand stream.

**Path of the first request for an on-demand stream:** Typically, the first request for an on-demand stream will be proxied to the streaming server because the stream is not yet in the cache. The following illustration shows what happens when a client makes a request for an on-demand stream that has not been cached.



The streaming media cache intercepts the streaming request sent by the client. Because the stream has not yet been cached, the cache asks the streaming server to deliver the requested stream. As it receives the requested stream, NetCache caches the stream and delivers it to the client.

**Path of subsequent requests for the same stream:** The following illustration shows the path for additional requests for the *same unique* stream that Client 1 requested.



The flow of requests and streams in the preceding illustration is as follows:

1. Assume that Client 1 already requested an on-demand stream and that NetCache cached the stream after fetching it from the streaming server. Client 2 and Client 3 now request the same unique stream that Client 1 requested.
2. Because the streaming media cache previously delivered the requested stream to Client 1, the stream is already in the cache and NetCache can satisfy the requests from Client 2 and Client 3 directly from the cache.

If Client 2 or 3 had requested the same streaming media content that Client 1 requested but at a bit rate encoding that had not been cached, the streaming media cache would have fetched that stream from the streaming server, as it did for Client 1. (Each bit rate encoding represents a different unique stream).

Caching streaming media content and delivering it directly to clients from a local source results in superior playback because it is more likely that packets will be delivered smoothly and on time. Additionally, considerable bandwidth is saved because streams can be served from the cache instead of traveling the entire distance from the streaming server.

## Section C: Deployment considerations

---

**About this section** This section provides issues to consider when planning for streaming media caching.

**Contents of this section** This section contains the following topics:

- ◆ [“Considerations for bandwidth”](#) on page 127
- ◆ [“Considerations for deploying NetCache multicast support”](#) on page 130
- ◆ [“Planning the number of streaming media caches needed”](#) on page 132
- ◆ [“Planning for client access to streaming media caches”](#) on page 133
- ◆ [“Planning for failover for streaming media service”](#) on page 135
- ◆ [“Considerations for firewalls and streaming media service”](#) on page 137
- ◆ [“Prefilling streaming media caches”](#) on page 140

## Considerations for bandwidth

---

### Plan for sustained streaming media bandwidth requirements

Bandwidth is a critical issue for streaming media because streaming media demands a consistent amount of bandwidth and can consume a great amount of bandwidth over time. All connections between the server and the streaming media cache remain open for the length of time that the server takes to deliver the stream to the cache, plus the length of time the cache takes to deliver the stream to the last client viewing the stream. Poor performance for other network traffic can result if you do not have sufficient bandwidth to accommodate both streaming media, with its long open connections, and other traffic.

In contrast, Web objects are bursty, typically consuming large amounts of bandwidth over a relatively brief period of time. The following table describes the basic differences between Web and streaming media delivery.

Web object delivery	Streaming object delivery
All data must be reliably transferred. Packet loss is not acceptable.	Some packet loss is acceptable. Some data loss is acceptable
Delivery is at network speed. No real-time requirement exists.	Delivery is paced and synchronized. Real-time constraints exist.
A single connection is opened per object.	Data and control connections are usually separate.
Small amounts of data.	Large amounts of data.
Sequential whole-file access.	Sometimes random and partial access.

Unlike Web objects, streaming media requires a minimum amount of *sustained bandwidth*. Streaming media packets must be received in a reasonable amount of time because audio and video must arrive on time and in sequence.

For example, if your streaming clients require 500 kbps of sustained bandwidth and network congestion reduces the bandwidth to 300 kbps, the audio and video will become choppy and will eventually fail. Conversely, Web objects will burst to the bandwidth that is currently available. A user might notice that a Web page is loading more slowly if the network becomes congested, but eventually the Web page will appear. Such latency might be acceptable to Web users, but it is not

acceptable to streaming media users who anticipate seeing or hearing the next action in the same smooth and vivid way they have experienced with television or radio.

---

**Note**

When you estimate the amount of bandwidth you need, be sure to take into account the amount of bandwidth that is used on your network at peak times.

---

**Bandwidth and the location of streaming media caches**

When you deploy a streaming media cache, the bulk of your bandwidth savings results from eliminating the need for each client to individually fetch the stream from the streaming server. The closer you deploy a streaming media cache to your clients, the more bandwidth you will save. This savings might be enough to eliminate the need to purchase more bandwidth or to reduce the amount of bandwidth that must be purchased. When you plan your streaming media deployment, consider several possible locations for your streaming media caches and evaluate the amount of bandwidth you should save with a cache in each location.

You can save bandwidth by pushing content that you anticipate your users will request to one or more NetCache appliances that you select. See [“Prefilling streaming media caches”](#) on page 140 and Appendix D, [“Considerations When Pushing Content,”](#) on page 277 for more information.

**Bandwidth and your requirements for stream quality**

Bandwidth has a direct relationship to the quality of streaming media presentations that you can provide. While you are estimating your bandwidth needs, you must also take into account the quality of service that you want to provide. If you need to provide high-quality service, plan for enough bandwidth to ensure that your network does not become congested.

By pushing streaming media content to your streaming media caches, you can reduce the impact of network congestion on streaming media delivered to users. As the distance the streaming media data needs to travel lessens, the possibilities of network problems that would affect the streaming media quality are likely to be fewer. Additionally, you are copying streaming media files to the streaming media cache rather than caching a stream that could have been affected by network conditions (for example, a thinned stream). See [“Prefilling streaming media caches”](#) on page 140 and Appendix D, [“Considerations When Pushing Content,”](#) on page 277 for more information.

**Note**

---

Quality of service is a particular challenge for an ISP because many users continue to use modems with speeds of 28.8, 33.6, and 56 kbps. Modems of 28.8 kbps are too slow to allow large-screen, good-quality video.

---

## Considerations for deploying NetCache multicast support

### Is multicast the right solution for your organization?

This section will help you determine whether using multicast transmission is the right solution for your organization. Using multicast is just one way of conserving bandwidth. Keep in mind that multicast can be used only for live streams and with network devices that support multicast. For some organizations, prepopulating NetCache appliances with content meets their requirements for bandwidth conservation. See [“Prefilling streaming media caches”](#) on page 140 for more information.

### Organizations that can benefit from using multicast transmission

The following table shows the types of organizations that can benefit from using multicast transmission.

Types of organizations	Usage example
Multicast-enabled enterprises	<p>Considerable bandwidth can be saved if NetCache passes streams to a multicast-enabled network instead of splitting the streams.</p> <p>NetCache can also enable an enterprise to optimize the transmission of Internet traffic. A majority of enterprises have no access to multicast from the Internet, although they might have set up multicast on their intranets. Such enterprises cannot, therefore, obtain a multicast stream through an Internet access point. If a NetCache appliance is set up for multicast and is located at an Internet access point, the appliance can convert unicast transmissions that it receives to multicast. See <a href="#">“Scenario: Deploying multicast in an enterprise”</a> on page 157.</p>
Internet CDNs	<p>Internal bandwidth usage in the CDN data center can be reduced by using multicast. See <a href="#">“Scenario: Multicast support for a CDN”</a> on page 163.</p>

Types of organizations	Usage example
Organizations using a satellite link	Organizations can stream corporate meetings over a satellite link to make them available worldwide, thereby reserving the corporate WAN for business-critical traffic. See <a href="#">“Scenario: Multicast support for transmission over a satellite link”</a> on page 160.
Web sites that NetCache appliances accelerate	Internal bandwidth usage can be reduced by using multicast.

**Multicast settings according to client location and activity**

The following table helps you determine which multicast settings, if any, are appropriate, depending on the location of clients and their settings for multicast.

If...	Then...
Clients will make requests only from origin servers in the same enterprise domain and the enterprise domain and clients are set up for multicast	<p>Typically, it is not necessary to configure a NetCache appliance for multicast in this situation. The reason is that the origin servers and clients in the enterprise can already communicate through the multicast channel.</p> <p>However, if your origin servers handle RTSP requests and you want to use NetCache features such as logging and bandwidth management, enable multicast on the NetCache appliance and configure NetCache for multicast output and multicast tunneling.</p>
Clients that are set up for multicast cannot reach an origin server directly, for example, because the origin server is located on the Internet	Enable multicast on the NetCache appliance and configure NetCache for multicast output. Do not set NetCache for multicast input (NetCache will use unicast input by default).

## Planning the number of streaming media caches needed

---

- Factors to consider** No one formula exists to determine the number of streaming media caches that you need. Your Network Appliance sales engineer can help you determine the number of caches that are appropriate for your organization, weighing factors such as the following:
- ◆ Size of your organization, the number of branch offices of your organization, and the distance between branch offices
  - ◆ Number of simultaneous connections you want to support

A linear relationship exists between the number of connections that a streaming media cache can handle and bandwidth usage per connection. NetCache should be able to split twice as many connections at 28.8 kbps than at 56 kbps.
  - ◆ Amount of bandwidth savings you want to achieve

When you are planning your deployment, consider how deploying streaming media caches in different locations would affect bandwidth. If, for example, you deploy only one streaming media cache and multiple clients request the same live stream, only one connection to the server is established. If you deploy two streaming media caches and clients of each cache request the same unique stream, each cache must establish a connection to the server. You might, for example, want to deploy regional streaming media caches in areas of the world where bandwidth is expensive.
  - ◆ Your desire for quality media streams

If your bandwidth usage is high and you do not want to purchase more streaming media caches, you must be willing to accept low-quality streams.
  - ◆ Whether you want your streaming media cache to function also as a Web cache or news cache

For deployments with heavy traffic loads, Network Appliance recommends that you configure a NetCache appliance for a single service, for example, only for streaming media service or only for Web service. Deploying dedicated NetCache appliances improves hit rates and bandwidth savings. Small sites with lighter traffic loads, such as at branch offices in the enterprise, might find that running multiple services on a NetCache appliance does not affect performance.

## Planning for client access to streaming media caches

---

### Client access through transparent proxying

If you are not using the NetCache Global Request Manager feature, the recommended way to deploy NetCache with streaming media service is to use transparent proxying, which involves configuring both of the following:

- ◆ Your L4 or L7 switch or WCCP router to redirect specific port traffic from clients to the NetCache appliance
- ◆ NetCache appliances for transparent proxying

For streaming media, set up transparent redirection on the L4 or L7 switch or WCCP router for the protocols for which you are deploying streaming, as shown in the following table.

Type of streaming traffic	Standard port traffic to redirect
RealNetworks (RTSP)	TCP 554 UDP 554
QuickTime (RTSP)	TCP 554 UDP 2001
Windows Media (MMS)	TCP 1755 UDP 1755
HTTP	TCP port 80 (or an alternative port for HTTP)

In addition to redirecting the streaming protocols, you must also redirect TCP port 80 (or an alternative port for HTTP) because some media players might be set for HTTP transport. Additionally, if your firewall administrator will not open ports on the firewall for the streaming protocols, streaming media must be transported over HTTP.

See Chapter 2, “[Strategies for Client Access to NetCache](#),” on page 17 for more details about transparent proxying.

**Direct (nontransparent) client access methods**

The following table summarizes the nontransparent methods for clients to access a streaming media cache.

Type of streaming	Summary
RTSP-based streaming	The user of a RealPlayer media player and the QuickTime player can manually configure the player to point to the streaming media cache.
MMS-based streaming	<p>If your streaming server is a Windows Media server, access to the streaming media caches can be set up as follows, depending on the Windows media player used:</p> <ul style="list-style-type: none"> <li>◆ The user of a Windows Media Player 7 (WMP 7) or later must manually configure the media player to point to the streaming media cache.</li> <li>◆ To support media players prior to WMP 7, you must deploy a NetCache appliance running as a Web cache and configure it for Windows Media metafile rewriting. You must also deploy a streaming media cache. (Both services can run on the same NetCache appliance.)</li> </ul> <p>Windows Media metafile rewriting enables MMS requests to be sent to the streaming media cache instead of to the Windows Media streaming server. Pre-WMP 7 media player users cannot configure their media players for access to a streaming media cache. See <a href="#">“Scenario: Deployment with a Windows Media server”</a> on page 150 for an example of this deployment.</p>

## Planning for failover for streaming media service

---

### No failover is possible for streams being delivered

You cannot set up a failover mechanism to handle streams that are in the process of being delivered to clients. Streaming connections are dropped if a problem occurs between the streaming media cache and the streaming server or between the streaming media cache and the clients. Ensuring that your network is sound should reduce the number of dropped connections.

The server streams the media stream to the cache, which then streams the media stream to the client. A connection between the server and streaming media cache remains open until the last client viewing the stream finishes viewing it. If a streaming media cache goes down, all connections that the streaming media cache had open between it and the streaming server are dropped. Users with connections that had been open must start playing the stream again.

If the origin server goes down, NetCache stops serving clients from that stream. New clients generate a new server connection.

### Building redundancy into your streaming media deployment

You can build redundancy into your streaming media deployment in the following ways:

- ◆ Add an extra streaming media cache.  
You might want to plan for one streaming media cache in addition to the number you think that you need to handle the streaming traffic. That way, if one streaming media cache goes down, the remaining streaming media caches can still handle the volume of streaming traffic.
- ◆ Use transparent proxying.  
If one streaming media cache goes down, the L4 or L7 switch or WCCP router distributes *new* streaming media requests over the remaining streaming media caches. Note that this deployment does not prevent existing connections from being dropped.
  - ❖ If you are using transparent proxying, configure your L4 or L7 switch or WCCP router to fail over to the Internet if all the streaming media caches go down.
  - ❖ If you are using transparent proxying with an L4 or L7 switch, you can deploy a switch failover pair if you are concerned about the failure of your switch.
- ◆ Use NetCache takeover if you are not using transparent proxying.

## Section C: Deployment considerations

- ◆ Deploy more than one streaming server with the same content. Then, if a cache miss occurs, the server does not become the point of failure.

## Considerations for firewalls and streaming media service

---

### About this section

Some administrators are reluctant to open ports on their firewalls because they are concerned that doing so will compromise security. This section compares NetCache vulnerability to attacks to those of computers running general operating systems and discusses the reasons why blocking streaming media-specific ports presents a problem. Network Appliance recommends that you share the information in this section with your firewall administrator.

### Firewall security and a NetCache appliance

Security breaches are less likely to occur through a NetCache appliance than through a general-purpose operating system such as Solaris. One reason is that NetCache has fewer UDP ports open than the general-purpose operating systems have open. Additionally, the Data ONTAP™ operating system is not as widely distributed as an operating system such as Solaris. Therefore, it is not as likely that a potential intruder will be able to write a Data ONTAP-specific script to breach security.

### Firewall requirements in a multicast streaming environment

Multicast uses UDP with special IP addresses. UDP and IGMP must be enabled on the firewall to ensure that multicast can work through a firewall.

### Correspondence between stream quality and protocols

Media players attempt to contact the streaming server using the protocols in the following table, in the order shown. As the table shows, blocking UDP or TCP streaming ports, or both, seriously degrades the quality of the streaming media you can deliver to clients.

Order	Protocol	Quality for streaming media
1	UDP	Best
2	TCP	Not as good as when UDP is used
3	HTTP	Poorest

**Need for an HTTP port for streaming service**

You must open a TCP port for HTTP traffic on the firewall (for example, TCP port 80), whether or not you have blocked streaming media TCP and UDP ports on the firewall, for the following reasons:

- ◆ If streaming media TCP and UDP ports are blocked on the firewall, client media players will try to use HTTP for streaming service. No streaming service is possible if the HTTP port is also blocked.
- ◆ Some client media players might be configured for HTTP transport. If TCP port 80 is blocked, no streaming service is possible for those clients.

**Note**

---

If you are using transparent proxying to redirect requests to a streaming media cache, you must set your L4 or L7 switch or WCCP router to redirect TCP port 80 requests also.

---

**Effect of blocked ports on stream caching and splitting**

The effect of blocking streaming media TCP and UDP ports on the streaming service that NetCache can provide depends on the location of the firewall in relation to the clients and the streaming media cache, as the following table shows.

<b>Location of the streaming media cache</b>	<b>TCP and UDP streaming ports blocked; TCP port 80 open</b>
One or more firewalls are between the clients and the streaming media cache.	Splitting and caching of streaming requests occur.  (NetCache contacts the streaming server using TCP and encapsulates the streams it returns to the client in HTTP.)
The streaming media cache and the clients are inside the firewall.	Only proxying of streaming requests over HTTP occurs. No splitting or caching can occur.

In both cases, because HTTP is used for streaming service, the quality of the streams returned to clients is degraded.

**Obtaining details about port settings for streaming media**

See the streaming media chapter in the *Guide to Caching Protocols and Services* for specific information about the minimum ports to open on your firewall to support streaming media caching. The requirements differ, depending on whether you are using MMS, RealNetworks RTSP, or Apple QuickTime RTSP.

**NetCache configuration related to firewalls**

How you configure your streaming media cache behind a firewall depends on whether the firewall is transparent or nontransparent, as the following table shows.

Type of firewall	Streaming media cache configuration
Transparent	No explicit configuration is necessary to pass requests through the firewall.
Nontransparent, running as a proxy	You must include the firewall in the cache hierarchy of the streaming media cache.

## Prefilling streaming media caches

---

### Consider prefilling caches

Typically, a streaming media cache is populated only as a result of user requests. You should consider whether prefilling your streaming media caches with content you expect your users will request will benefit your organization.

### Benefits of prefilling caches with streaming media

The benefits of prefilling the cache with streaming media include the following:

- ◆ Saves bandwidth (Streaming media files are very large.)
- ◆ Improves delivery time
- ◆ Reduces the quality problems that network congestion causes for streaming media

As the distance the streaming media data needs to travel lessens, the possibilities of network problems that would affect the streaming media quality are likely to be fewer.

See Appendix D, “[Considerations When Pushing Content](#),” on page 277 for more details about the benefits of prefilling streaming media caches.

### Examples

A few examples of the type of content some organizations push to caches are as follows:

- ◆ Employee training videos
- ◆ Corporate meetings that were recorded
- ◆ Static Web pages
- ◆ Advertising videos
- ◆ Graphic-rich documents
- ◆ Movies and movie trailers

### Prefilling caches

You can use the NetCache CLI command (`confill`) to prefill the caches of NetCache appliances. The streaming media chapter in the *Guide to Caching Protocols and Services* contains information about the `confill` command.

## Section D: Deployment scenarios

---

**About this section** This section includes several scenarios to illustrate the deployment of streaming media caches.

**Contents of this section** This section contains the following topics:

- ◆ [“Scenario: ISP adding streaming media caches”](#) on page 142
- ◆ [“Scenario: Distributed streaming media caches”](#) on page 146
- ◆ [“Scenario: Deployment with a Windows Media server”](#) on page 150
- ◆ [“Scenario: Prefilling content of corporate NetCache appliances”](#) on page 155
- ◆ [“Scenario: Deploying multicast in an enterprise”](#) on page 157
- ◆ [“Scenario: Multicast support for transmission over a satellite link”](#) on page 160
- ◆ [“Scenario: Multicast support for a CDN”](#) on page 163

## Scenario: ISP adding streaming media caches

---

**About this scenario** The ISP in this scenario provides Web caching service to its customers. The ISP has Web caches in its POPs and in the Data Center. The ISP now wants to add streaming media service.

Currently, all traffic from the POPs is routed through the Data Center. The ISP wants to continue with this routing scheme when streaming media service is added.

**ISP's requirements** The ISP's requirements when deploying streaming media service are as follows:

- ◆ To limit bandwidth consumption by streaming media

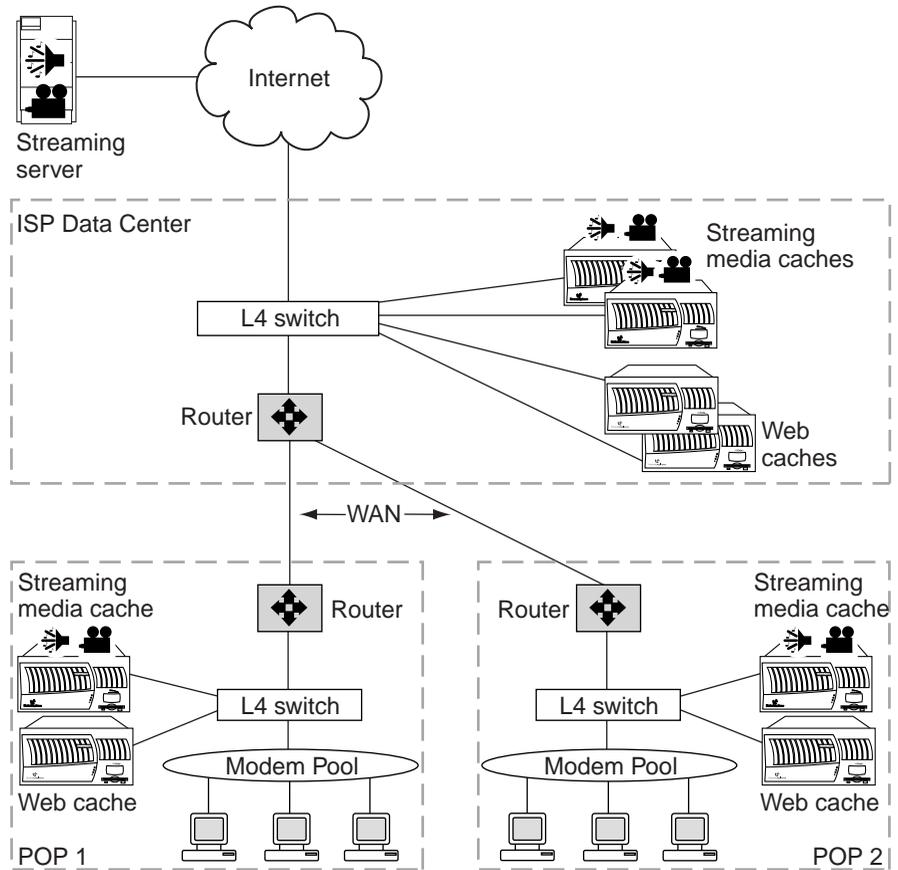
The deployment of Web caches in the POPs significantly reduced Web traffic across the WANs to the Data Center. The ISP wants to implement a similar scheme for streaming media. The reason is that the ISP expects that the addition of streaming media service will cause traffic congestion across the WANs, resulting in slow service for all protocols. Additionally, the ISP managers want a deployment scheme that does not mandate purchasing additional bandwidth.

- ◆ To provide good quality streaming media service for customers using modems of various speeds

The ISP's customers have modems of various speeds, with many customers still using 28.8-kbps modems. The ISP wants to be able to provide the best-quality streaming media service that is possible for each of the common modem speeds.

### **Deployment illustrated**

The following illustration shows how the ISP deployed streaming media caches. For simplicity, only two POPs are shown in the illustration.



**Note**

For this scenario, all traffic from the POPs to the Internet is routed through the Data Center.

**Deployment at the Data Center:**

- ◆ A hierarchical deployment was created by adding streaming media caches in the Data Center to handle streaming misses from the POPs. This deployment significantly reduces the number of connections to the streaming server. The reason is that if multiple POPs request the same stream, a Data Center streaming media cache needs to connect to the streaming server only once for each unique streaming media. For live streaming media, the cache can then respond to additional requests for the same unique stream by splitting the stream for any additional requests for the same stream.

- ◆ All NetCache appliances at the Data Center are dedicated streaming media caches or dedicated Web caches.

In this scenario, using dedicated appliances in the Data Center is especially desirable because the Data Center NetCache appliances must handle high volumes of each type of traffic from many POPs.

- ◆ Multiple streaming media caches were deployed at the Data Center for redundancy. The ISP managers do not want streaming service to be unavailable if a streaming media cache goes down.

The managers realize that if a streaming media cache goes down, any connections between a streaming server and the cache that are open will be lost. However, subsequent connection attempts could be handled by another streaming media cache.

The number of streaming media caches deployed in the Data Center was determined by considering the following:

- ◆ The volume of streaming traffic that the managers expect at the Data Center  
No clients send streaming requests to the Data Center streaming media caches directly. All streaming traffic is a result of streaming misses at the POPs.
- ◆ The need for redundancy
- ◆ The capability of the network connection to the Internet to handle streaming requests

The more streaming media caches deployed, the more separate connections to streaming servers in the Internet would be established.

Because the ISP does not own any streaming servers, shielding the streaming server from too many connections is not an issue for the ISP. A company that owns the streaming server and is setting up streaming service would need to consider the number of connections that the streaming server could handle.

#### **Deployment at the POPs:**

- ◆ The Web caches that were previously deployed at the POPs remain dedicated to Web caching.
- ◆ One dedicated streaming media cache was deployed at each POP to reduce the amount of streaming traffic over the WANs.
- ◆ The L4 switch at each POP sends streaming requests to the Data Center if the streaming media cache at the POP goes down. In this case, the L4 switch at the Data Center passes the streaming requests to a streaming media cache at the Data Center.

A second streaming media cache could have been deployed at each POP to provide failover locally in case the first streaming media cache went down. If

all streaming media caches go down, the L4 switch at the POP sends the streaming requests to the Data Center.

As an alternative, configuring a hierarchy on the two streaming media caches in the POP could eliminate the need for an L4 switch in the data center. If a hierarchy was configured on a POP streaming media cache, cache misses would be sent from the POP cache up the hierarchy to the logical parent cluster in the data center. The drawback is that if both streaming media caches in the POP became unavailable, client traffic would be sent directly to the Internet instead of to a streaming media cache in the data center.

## Scenario: Distributed streaming media caches

---

**About this scenario** The managers of NetDevices, Inc., want to provide live training sessions to their employees in offices around the world. They want to deploy streaming media service transparently, with L4 switches deployed with streaming media caches.

### Organization's requirements

The organization's requirements are as follows:

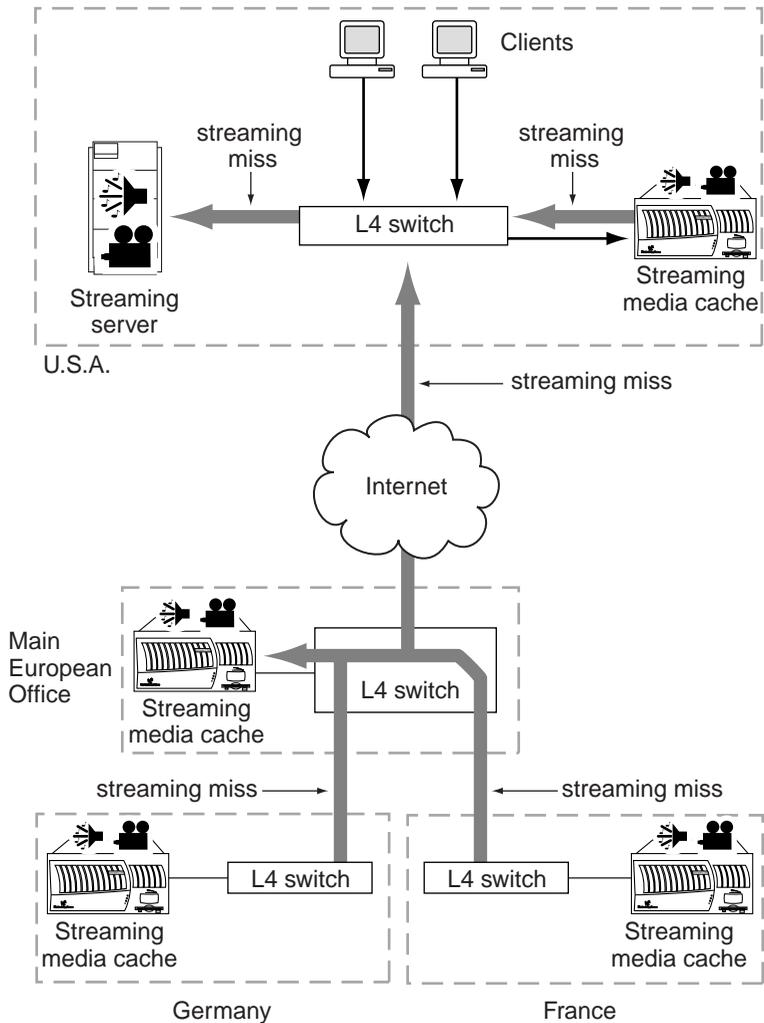
- ◆ Use bandwidth efficiently
  - ❖ The planner wants a deployment scheme that uses bandwidth as efficiently as possible.  
Bandwidth is quite expensive in some of the countries in which the company's offices are located. The planner wants to avoid purchasing additional bandwidth or, if that is not possible, to minimize the amount of bandwidth that must be purchased.
  - ❖ The planner anticipates that the company's current bandwidth cannot support the traffic that would result if clients from all branch offices sent requests directly to the streaming server in the U.S.A. The planner wants to be able to deploy streaming media service so that most requests from the European regions are resolved without sending the requests to the U.S.A.
- ◆ Protect the streaming server from becoming overloaded  
The planner expects that the streaming server in the U.S.A. cannot handle the number of connections that would result if all clients connect directly to the server. The planner, therefore, needs a solution that "protects" the streaming server.
- ◆ Maintain the same quality of service for nonstreaming traffic as that was provided before deploying streaming media service  
The planner wants to be sure that the increase in traffic resulting from streaming media service does not adversely affect the quality of service for other types of network traffic.
- ◆ Provide good quality streaming media

### Deployment illustrated

The following illustration shows how streaming media caches were deployed at NetDevices, Inc., and their relationship with the streaming server. The streaming media caches are deployed in regions to be able to serve streams closer to the users.

**Note**

For simplicity, network devices other than the L4 switches, streaming servers, and streaming media caches are not shown in the illustration.



The arrows in the previous illustration show, for each streaming media cache, where a streaming request is sent next if the streaming media cache in the region is not already delivering the stream to a client (that is, if a streaming miss occurs).

### Deployment of transparency with L4 switches

An L4 switch was deployed in each region in front of the streaming media cache. Each L4 switch is set up for transparent proxying, configured to pass requests for port 1755 (MMS) and port 554 (RTSP) to the streaming media cache. (In this example, the organization supports streaming media for both MMS and RTSP.) The switch is also configured to pass HTTP (port 80) traffic to the streaming media caches to support client media browsers that are set for HTTP transport only.

---

#### Note

A WCCP router could have been deployed in front of the streaming media caches instead of an L4 switch.

---

### Hierarchical routing

In this organization, routing is set up in a hierarchical fashion. Traffic of any type from the German and French regions destined for the U.S.A. must be sent through the European region.

---

#### Note

Alternatively, you can create a logical NetCache hierarchy to distribute requests that a NetCache appliance cannot resolve to one or more proxy-cache servers higher in the logical hierarchy. See the *Administration Guide* for details about hierarchies.

---

### Request resolution

Only a subset of the company's streaming requests must be sent over the Internet to the streaming server because the regional streaming media caches attempt to resolve streaming requests that they receive.

#### Request resolution by the French and German streaming media caches

- ◆ If the French or German streaming media cache receives a request for a live stream that it is already delivering to a client, it splits the stream for any additional clients that send requests for the same live stream.
- ◆ If the French or German streaming media cache is not already delivering a stream that a client requests, the cache sends the request to the next level up in the hierarchy—to the European streaming media cache. For this reason,

the hierarchical routing scheme works especially well for the streaming media deployment in Europe.

### **Streaming misses sent to the European region**

- ◆ If the European streaming media cache is already delivering a requested live stream to the German cache and the French cache sends a request for the same stream, the European cache splits the live stream and delivers the stream to the French cache. The same is true if the German cache sends a streaming request to the European cache and the European cache is already delivering the stream to the French cache.
- ◆ If the European region streaming media cache is *not* already delivering the requested stream, it connects to the streaming server in the U.S.A. region to obtain the stream and streams it to the cache that requested it.

### **Streaming misses sent to the U.S.A. region**

The L4 switch in the U.S.A. region passes streaming misses from the European region directly to the streaming server. The L4 switch in the U.S.A. region passes requests from clients in the U.S.A. region to the L4 switch, and sends cache misses to the streaming server.

## **Failover**

Failover works as follows for this scenario:

- ◆ **Current connections**

If any of the streaming media caches goes down while it is delivering streams, all connections to clients that are viewing those streams are lost. The client must initiate the connection again. If the regional cache is still down when the client tries to initiate the connection again, the L4 switch passes the streaming request to the region above it in the hierarchy.
- ◆ **New requests**

If the streaming media cache in a region goes down, the L4 switch passes any *new* requests to the region above it in the hierarchy.

## Scenario: Deployment with a Windows Media server

---

**About this scenario** This scenario provides two examples of the deployment of a streaming media cache and a Windows Media streaming server. In both examples, the deployment is nontransparent; that is, no L4 or L7 switch, WCCP router, or policy-based router is deployed to pass requests to the streaming media cache. Instead, Windows Media metafile rewriting is used to pass requests to the streaming media cache. The scenarios in this section illustrate the use of Windows Media metafile rewriting in two nontransparent streaming media deployments.

Network Appliance recommends deploying streaming media service by using transparent proxying.

---

### Note

An alternative to the deployment examples in this scenario is described in [“Scenario: ISP adding streaming media caches”](#) on page 142 and [“Scenario: Distributed streaming media caches”](#) on page 146 for RTSP-based transparent streaming.

---

### Windows Media metafile rewriting enables the request to be sent to the cache

Streams on a Windows Media server are contained in one of a few types multimedia files—.asf, .wmv., or .wma. A text file (.asx, .wmx, or .wax) acts as a link from a Web page to the .asf, .wmv., or .wma file on a Windows Media server. Clicking an .asx, .wmx, or .wax file link on a Web page transfers control of the data from the Web browser to the Windows Media player so that the data can stream. An example of an ASX file follows:

```
<asx version = "3.0">  
<ref href = "mms://server/content/movie.asf"/>  
</asx>
```

For a streaming media cache to be able to intercept a nontransparent MMS streaming request when transparent proxying is not deployed, the origin address in the text file must be changed to the address of the cache.

The NetCache Windows Media metafile rewriting feature, used for a nontransparent streaming media deployment with a Windows Media server, prepends the IP address of the streaming media cache to the origin address.

After NetCache applies Windows Media metafile rewriting to the previous ASX file, the following entries are displayed.

```
<asx version = "3.0">  
<ref href = "mms://streaming-cache-ip/server/content/movie.asf"/>  
<ref href = "mms://server/content/movie.asf"/>  
</asx>
```

---

**Note**

The NetCache Windows Media metafile rewriting feature is necessary for Windows Media media players prior to version 7 but can be used with later versions.

---

**About the company in this scenario**

ABC, Inc., is a small company with offices that are close geographically. Currently, the company has one NetCache appliance running as a Web cache. Client browsers are manually configured to point to the Web cache; the company is not interested in using transparent proxying for client access to the NetCache appliance.

The managers are interested in providing their employees with access to streaming media. They have purchased a Windows NT server that is to function as a Windows Media streaming server.

**Company requirements**

The company's requirements are as follows:

- ◆ Provide good quality streaming media presentations
- ◆ Protect the streaming server from becoming overloaded with too many connections
- ◆ Do not add any network hardware other than a NetCache appliance and the Windows Media server.

**Providing client access to the streaming media cache**

The key points to understand about setting up a nontransparent deployment with a Windows Media server are as follows:

- ◆ The streaming request must be sent to a Web cache first.  
Windows Media software prior to version 7 does not provide any way to point a Windows Media media player to a cache. Therefore, in a nontransparent streaming media deployment with a Windows Media server, client Web browsers must be configured to point to the Web cache.

- ◆ The Web cache must pass the streaming request to the streaming media cache.

The Web cache must rewrite the origin server address to that of the streaming media cache so that the streaming request can reach the cache. The Web cache rewrites the address through Windows Media metafile rewriting.

- ◆ You must configure the information that the Web cache needs to rewrite the origin server address to that of the streaming media cache.
- ◆ Windows Media metafile rewriting works whether you are deploying a single NetCache appliance configured to run as both a Web cache and a streaming media cache or whether you are deploying a dedicated streaming media cache.

Configuration is slightly different in these two types of deployments, as described in the examples in the following section.

---

**Note**

Network Appliance recommends deploying dedicated NetCache appliances. However, small organizations might find that one NetCache appliance can handle the traffic for multiple protocols.

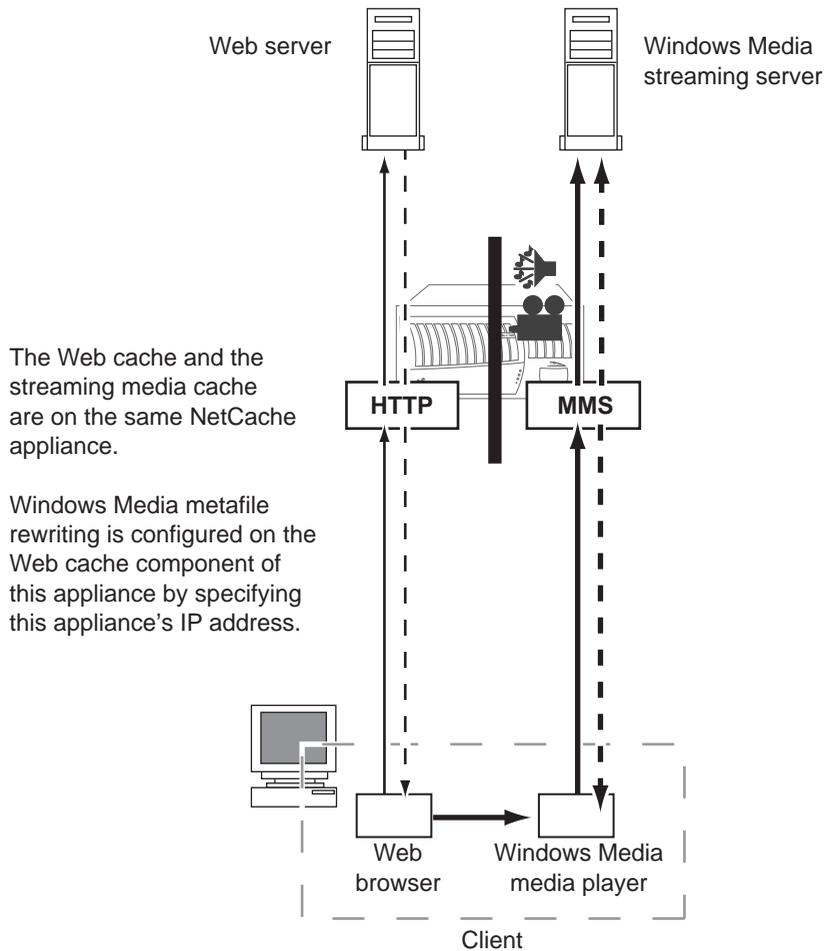
---

## Deployment examples illustrated

This section contains the two examples of nontransparent streaming media deployments in which a Windows Media server running pre-version 7 software is deployed, as follows:

- ◆ Example 1 shows a NetCache appliance that is configured as both a Web cache and a streaming media cache.
- ◆ Example 2 shows one NetCache appliance deployed as a dedicated Web cache and another NetCache appliance deployed as a dedicated streaming media cache.

**Example 1: a single NetCache appliance configured as both a Web cache and a streaming media cache:** In the following example, client browsers are configured to point to the Web cache, which in this case is on the same NetCache appliance as the streaming media cache.

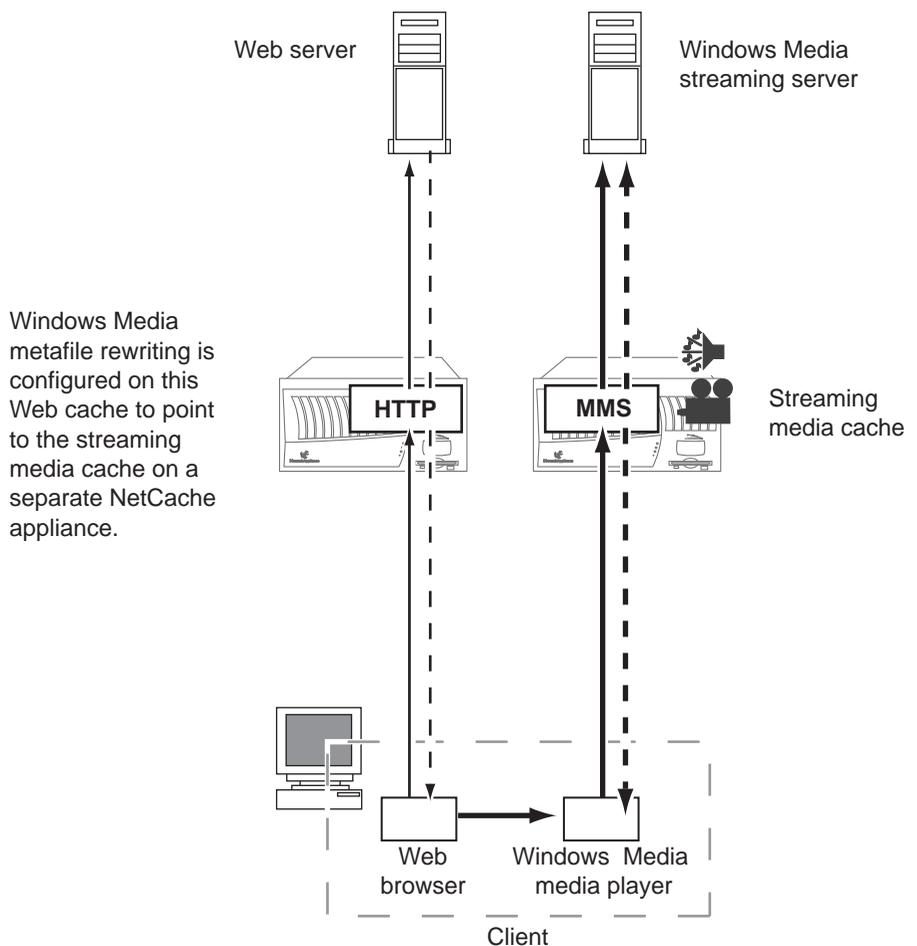


In the preceding example, notice that NetCache handles the HTTP and MMS protocols separately. The Web cache side of the NetCache appliance must communicate with the Web server to obtain the ASX file that NetCache needs to translate the origin server address of the streaming server to the origin server address of the streaming media cache.

After the Web cache rewrites the origin server address, the Web browser passes the request to the Windows Media player on the client, which then passes the request to the streaming media cache. The streaming media cache connects to the Windows Media server for any requested streams that it is not already delivering.

**Example 2: dedicated NetCache appliances:** The following example shows two dedicated NetCache appliances. One is configured as a Web cache and the other is configured as a streaming media cache. The other is configured as a streaming media cache.

As in Example 1, client browsers must be configured to point to the Web cache. The handling of the request from the Web cache to the streaming server is exactly the same as the process described in Example 1. The difference between this example and Example 1 is where you configure Windows Media metafile rewriting. In this example, you must configure Windows Media metafile rewriting on the dedicated *Web cache*. Doing so enables the Web cache to rewrite the IP address of the streaming server to the IP address of the dedicated streaming media cache.



## Scenario: Prefilling content of corporate NetCache appliances

---

**About this scenario** The managers of XYZ Corporation want to provide training videos to their employees in offices across the United States. They want employees to be able to view these videos at any time (video on-demand).

### Organization's requirements

The organization's requirements are as follows:

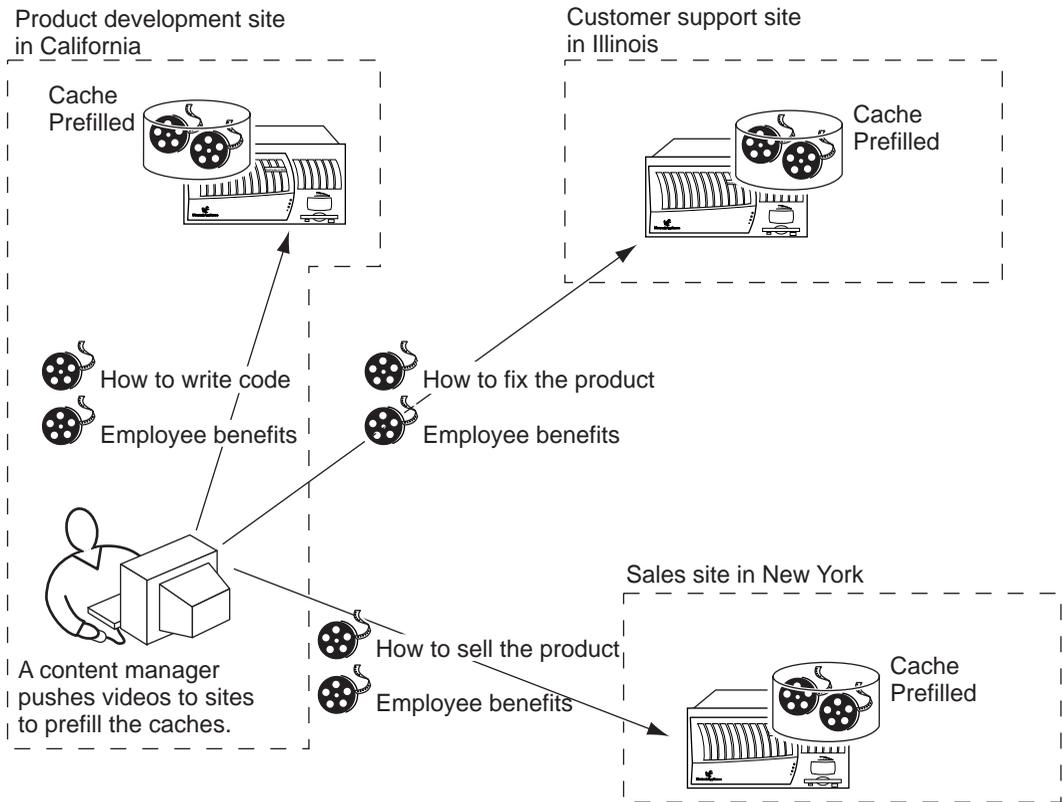
- ◆ Reduce the amount of traffic on the corporate LAN and WAN
- ◆ Provide good quality video quickly
- ◆ Serve training material specific to a particular site from the NetCache appliance at that site

Different corporate sites have different training needs and, therefore, require different video tapes.

- ◆ Provide all employees in the United States the same corporate information, for example, videos to keep employees informed about the corporation and videos about employee benefits

### Deployment illustrated

The following illustration shows content pushed out to different sites of XYZ Corporation.



As the previous illustration shows, the content manager pushes the site-specific streaming media to the three company sites. Additionally, the content manager pushed the same video about corporate news to each site. The content manager pushes the content to the caches as batch jobs at times when the WAN is not busy, for example, at 2:00 a.m.

The requirement to reduce bandwidth use is achieved because employee requests for videos appropriate for their sites can be satisfied from the local NetCache appliance instead of from a remote appliance.

Data can reach the employees more quickly because it is close to the users. Additionally, as the distance the streaming media data needs to travel lessens, the possibilities of network problems that would affect the streaming media are likely to be fewer.

This scenario showed three U.S. sites of a company. If the sites were in different countries, you could push language-specific videos to each site.

## Scenario: Deploying multicast in an enterprise

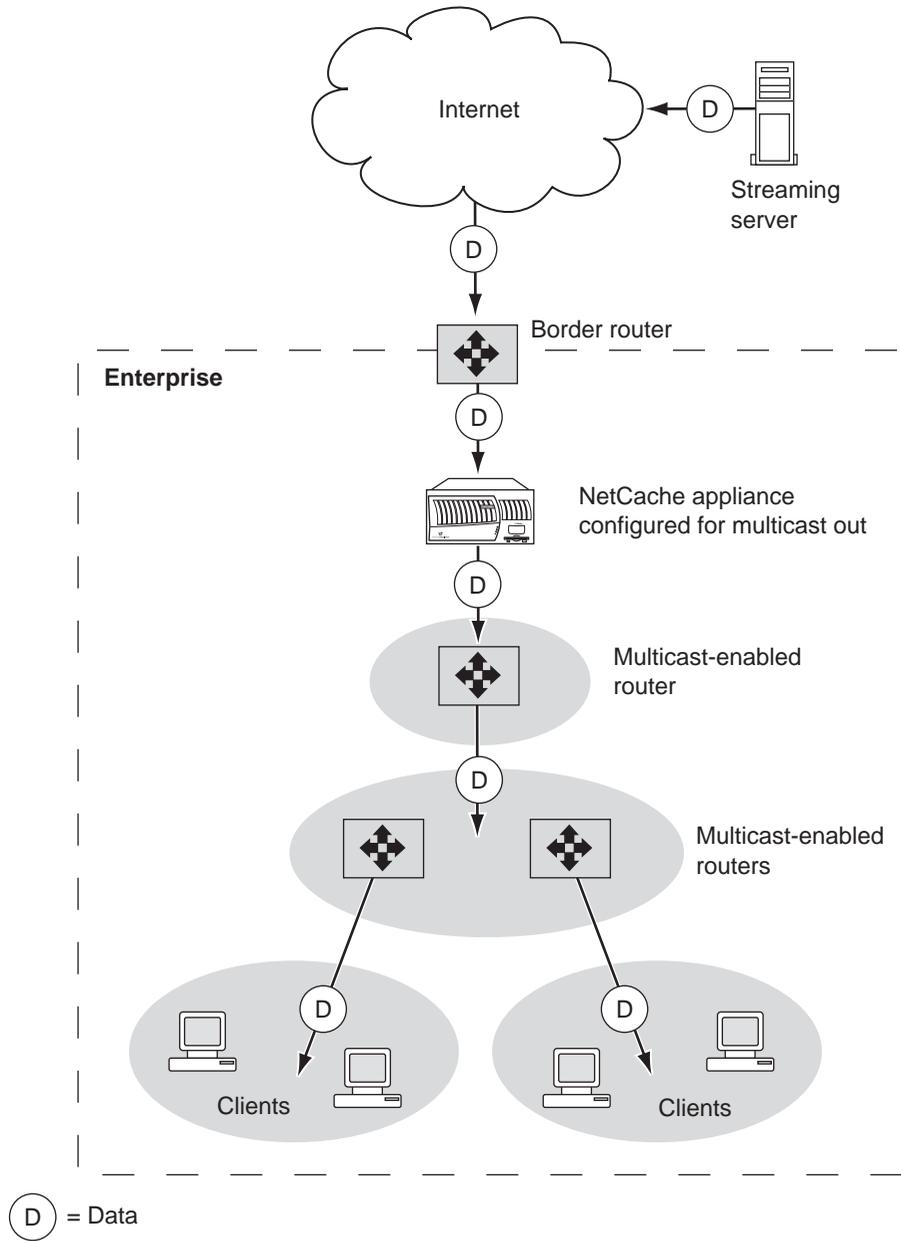
---

**About this scenario** This scenario describes a multicast-enabled NetCache appliance deployed in an enterprise that is multicast enabled.

**Organization's requirements** The organization's requirements are as follows:

- ◆ Reduce bandwidth consumption on the network
- ◆ Protect the routers from traffic overload
- ◆ Streams from origin servers on the Internet are to be distributed on the company's multicast-enabled network using multicast transmission

**Deployment illustrated** The following illustration shows a deployment in which a NetCache appliance configured for multicast output was deployed at the Internet access point.



**Use of a NetCache appliance configured for multicast versus a multicast-enabled router:** The NetCache appliance in the previous illustration is configured to convert the unicast stream from the Internet to multicast (multicast out) for distribution on the company network. A multicast-enabled router could have been deployed at the Internet access point. However, the advantages of deploying a NetCache appliance at the Internet access point are as follows:

- ◆ NetCache sends only one request for the same unique live stream to the origin server, no matter how many clients request that stream. A router would send each client's streaming request individually to the origin server.
- ◆ NetCache provides a number of features that are not available with a router, for example, logging and access controls.
- ◆ Most streams from the Internet are not multicast. If a multicast-enabled router was used, source streams would have to be multicast because a multicast router cannot convert unicast transmissions to multicast.

**Client-to-server traffic in the illustration:** The NetCache appliance sends only one request for the same unique stream to the origin server, no matter how many clients request that stream. This behavior accomplishes the company's goals to conserve bandwidth. Additionally, the border router is protected from traffic overload because only one request is sent through the border router on the way to the origin server. A separate request for each client would be sent through the border router if the NetCache appliance was not deployed inside the border router.

**Server-to-client traffic in the illustration:** The origin server responds with only one copy of the requested unique stream (because it received only one request from the NetCache appliance), thereby reducing bandwidth usage. The NetCache appliance, which is configured for multicast output, passes the stream to the multicast channel instead of splitting it. Each multicast router in the routing path determines whether to pass the stream through to the next branch of routers or split the stream. The use of multicast, therefore, protects the routers on the company's network from traffic overload. Clients configured for multicast listen for multicast traffic. Only one copy of the data is passed to the network of multicast-enabled clients.

## Scenario: Multicast support for transmission over a satellite link

---

**About this scenario** This scenario describes a deployment for a company with multiple offices across the world. NetCache appliances are configured for multicast input.

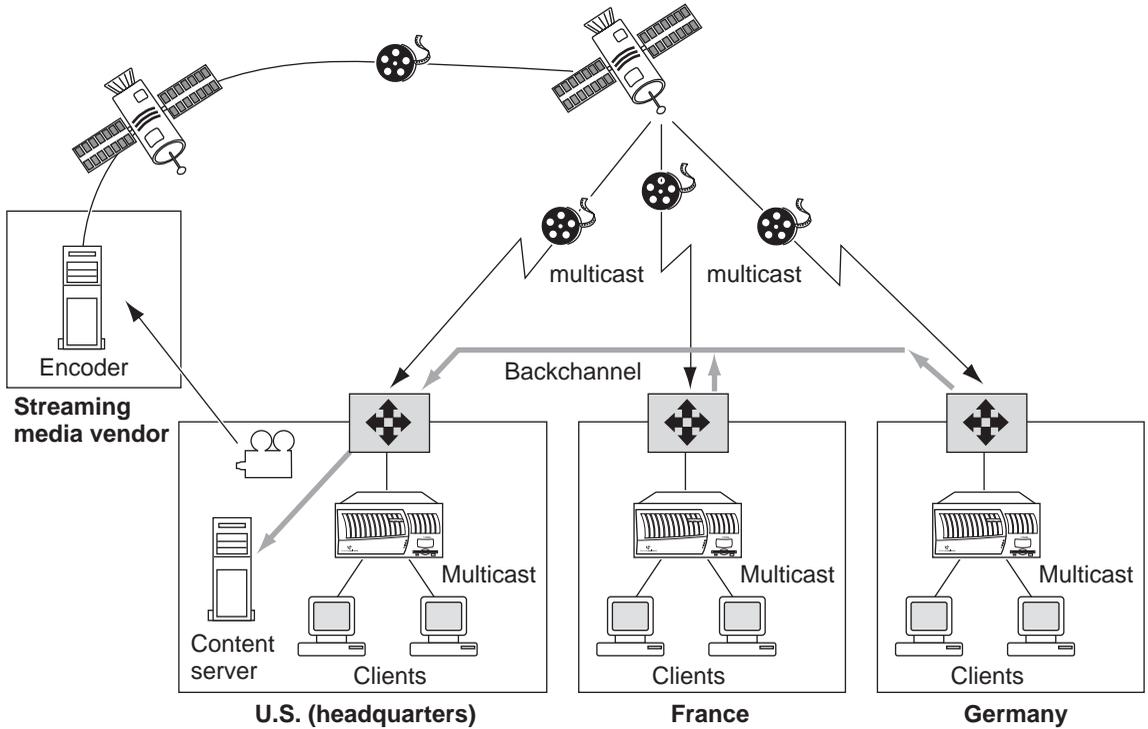
### Organization's requirements

The company's requirements are as follows:

- ◆ Reserve its primary data link for business-critical traffic
  - ◆ Use multicast transmission over a satellite link to send non-business-critical traffic, such as streaming media of company meetings and training videos
- Use of a satellite link eliminates the possibility of non-business critical traffic causing network congestion, which would interrupt company business and reduce the quality of streaming media. Use of a satellite link for non-business-critical traffic will also reduce costs because it is less expensive than the primary data link that the company uses.

### Deployment illustrated

The following illustration shows a multicast broadcast over a satellite link.



Assume that the corporate headquarters is broadcasting a live corporate meeting. A camera in the headquarters records the meeting and transmits it to a third-party vendor to encode the stream and upload the stream to a satellite link for transmission over multicast. The corporate offices in France and Germany download the stream from the satellite so that their employees can view the corporate meeting live.

Clients at the U.S. site could view the stream through the corporate LAN, or the stream could be downloaded from the satellite.

Configuration of the NetCache appliances in France and Germany is set up as follows:

- ◆ The NetCache appliances in France and Germany are configured for multicast input so that they can receive multicast transmissions from the multicast-enabled router that downloads the stream from the satellite link.
- ◆ The NetCache appliances in France and Germany are also configured for multicast output. The reason is that each corporate site is multicast-enabled. By configuring the NetCache appliances for multicast out, the appliances can pass streams to the next-hop routers in the corporation without splitting

the stream. Each router determines whether the stream should be split at that point.

---

**Note**

Typically, communication over the satellite is asymmetric, that is, only from the satellite transmission point to the destination at the end of the satellite link. Requests from France and Germany for the live broadcast must be sent over the terrestrial backchannel link.

---

## Scenario: Multicast support for a CDN

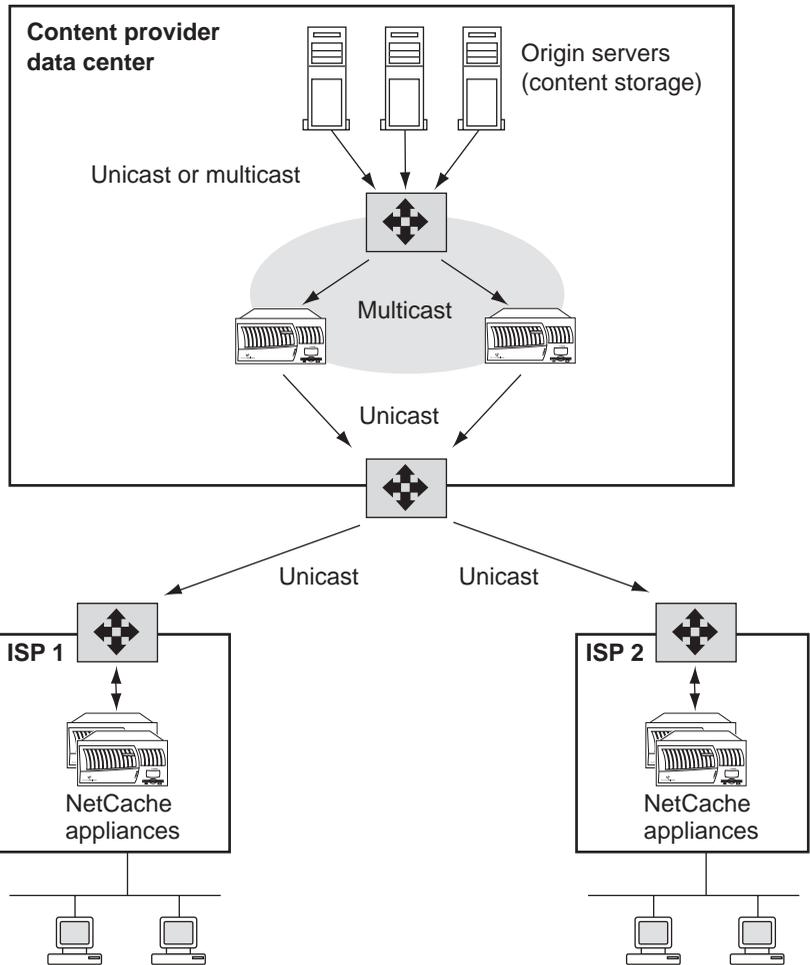
---

**About this scenario** This scenario describes a deployment in which NetCache appliances owned by a CDN are configured for multicast input. The CDN in this scenario provides services for several ISPs.

**Organization's requirements** The organization's requirements are as follows:

- ◆ Minimize internal bandwidth use in the CDN data center by using multicast
- ◆ Manage multicast streaming requests based on preconfigured priorities
- ◆ Log streaming activity by each ISP

**Deployment illustrated** The following illustration shows the traffic flow from the streaming server after it receives a multicast stream request.



The benefits of using NetCache multicast support are described in the following paragraphs.

**If the router between the content provider data center and the POPs is a unicast router:** In this scenario, the router at the POP is a unicast router. Configuring the NetCache appliances for multicast input (and using the default of unicast out) has the following benefits:

- ◆ A unicast router cannot route multicast transmissions. Configuring NetCache appliances for multicast input enables the appliances to listen for multicast transmissions from the multicast-enabled routers in the data center.

The NetCache appliances can then convert the multicast transmissions to unicast for routing to the unicast router at the POP.

- ◆ Forcing all multicast streams to be sent through the NetCache appliances enables the CDN to take advantage of NetCache services that will help achieve the CDN's other goals. For example:
  - ❖ NetCache can log streaming activity. The ContentReporter reporting application can be used to consolidate logs from each NetCache appliance and report that data in report formats that the CDN requires. The CDN might want data for billing each ISP for services.
  - ❖ If the NetCache bandwidth management feature is configured, NetCache can manage bandwidth based on the configured priorities.

**If the router between the content provider data center and the POPs is multicast-enabled:** It would not be necessary to route streams through multicast-enabled NetCache appliances to achieve the CDN's goal to minimize bandwidth consumption. However, forcing all multicast streams to be sent through the NetCache appliances enables the CDN to take advantage of NetCache logging and bandwidth management features, as previously described.

Most ISPs cannot accept multicast transmissions from the Internet. If the ISPs for which a CDN provides services can accept multicast transmissions, for example, because a satellite link exists between the CDN and the ISP, the CDN could configure its NetCache appliances in the data center for multicast output. The CDN would then need to advise administrators of the ISPs to configure their NetCache appliances for multicast input.



**About this chapter** A NetCache appliance can be used to accelerate a Web site or streaming media site. This chapter defines a NetCache accelerator and presents several scenarios in which NetCache accelerators are used.

**Chapter contents** This chapter contains the following sections:

- ◆ “[What is a NetCache accelerator?](#)” on page 168
- ◆ “[Strategies for client access to an accelerator](#)” on page 172
- ◆ “[Scenario: an accelerator outside the firewall](#)” on page 174
- ◆ “[Scenario: NetCache as a distributed Web site accelerator](#)” on page 176
- ◆ “[Scenario: multiple accelerators accelerating a single server](#)” on page 179
- ◆ “[Scenario: single accelerator accelerating multiple servers](#)” on page 181
- ◆ “[Scenario: allowing limited access from another company](#)” on page 183
- ◆ “[Scenario: accelerator for an historical stock performance Web site](#)” on page 185

# What is a NetCache accelerator?

## Description of an accelerator

A NetCache appliance can be configured as either of the following types of accelerators:

- ◆ Web accelerator for HTTP and HTTPS requests
- ◆ Streaming accelerator for RTSP and MMS requests

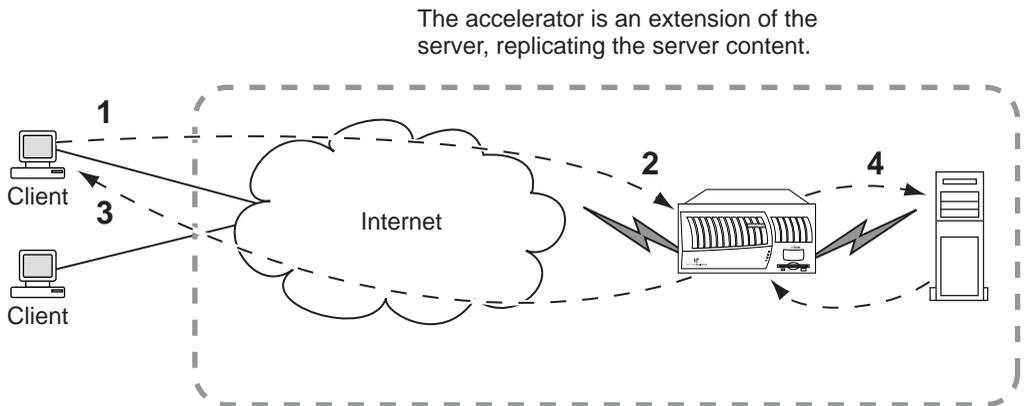
When NetCache is configured as an accelerator, it caches content from one or more *Web server* or *streaming servers* that you identify, and provides that content to clients that request it. The accelerator is, therefore, an extension of the Web server or streaming server (as applicable). To the outside world, the accelerator *is* the Web server or streaming server. In contrast, when NetCache is configured as a Web cache or streaming media cache, it acts as an agent for the browser. An accelerator is located close to the origin servers that it services.

### Note

An accelerator might also be referred to as a reverse proxy.

## Request flow with an accelerator

The following illustration shows the flow of a request from the client to the accelerator, and back to the client.



The following table describes the path that a request travels when an accelerator is deployed.

Stage	Description
1	Users send Web or streaming media requests.
2	Because the accelerator has become an extension of the Web server or streaming server, the accelerator receives the requests.
3	If the accelerator has the requested content in its cache, the accelerator returns that content to the client directly.
4	If the accelerator does not have the requested content in its cache, it fetches the content from the Web server or streaming server, and then returns it to the client while caching the data, if it is cacheable.

In the typical deployment of an accelerator, a Web browser does not have direct communication with the Web server and a media player does not have direct communication with the streaming server.

### Advantages of using an accelerator

The advantages of using NetCache as an accelerator include the following:

- ◆ An accelerator off-loads client traffic from your Web or streaming servers.
- ◆ An accelerator shields Web servers and streaming servers from the outside world.

Web servers and streaming servers are vulnerable to break-ins, whereas a NetCache appliance is more secure from them. The advantage of using an accelerator is that a Web server's or streaming server's content is always secure because an intruder cannot reach the server. If an intruder breaks into the accelerator, the intruder can only disable the accelerator; the intruder cannot modify the data in any way.

---

#### Note

Locating a Web server or streaming server inside the corporate firewall is often desirable to protect the site's content from unauthorized changes. However, depending on its complexity, a corporate firewall might introduce substantial delays or it might have limited bandwidth. NetCache enables you to provide Web content and streaming media content more quickly to users and saves bandwidth because duplicate requests to the same accelerator for the same content are satisfied from the cache.

---

- ◆ Work is off-loaded from the Web server or streaming server onto the accelerator. Therefore, you might need fewer Web servers or streaming servers.

### **What an accelerator caches**

A Web accelerator caches only cacheable objects. A Web accelerator proxies noncacheable objects (for example, CGI and private pages).

A streaming accelerator caches on-demand streaming media and splits live streaming media.

### **Can an accelerator be run in more than one mode**

You can run a NetCache appliance as both a Web accelerator and a Web cache. Likewise, you can run a NetCache appliance as both a streaming accelerator and a streaming media cache.

### **Hit rate when NetCache is an accelerator**

The hit rate for a NetCache appliance running as an accelerator is significantly greater than the rate for a NetCache appliance running as a Web cache or streaming media cache. The reason is that the Web server or streaming server that NetCache accelerates has a limited amount of data, compared to the World Wide Web, which has nearly an infinite amount of data. With an accelerator, it is likely that many users will send requests for the same data from the Web server or streaming server.

### **Load balancing over servers that NetCache is accelerating**

If NetCache is configured to accelerate a pool of Web or streaming servers with the same content, NetCache uses a weighting technique based on a history of connection round-trip times to balance the load of requests across the servers. Server CPU load and other server resources are not considered. Initially, all servers have the same chance of being selected. If a server is consistently slower in responding to the NetCache appliance, over time, the chance decreases that the server will be selected.

**Scenario:** A server that the appliance is configured to accelerate becomes unavailable. Subsequently, NetCache picks that server from the pool of servers to send the request to. However, the connection fails because the server is down. NetCache records a large round-trip time for this server, which reduces its weight and, therefore, the likelihood that NetCache will pick the server the next time. A chance still exists that NetCache will select that server again because NetCache uses a “weighted random” algorithm. However, over time, the probability that

NetCache will select the server decreases. The benefit of this algorithm is that the server is not removed from consideration after a single failure; that failure might be temporary.

# Strategies for client access to an accelerator

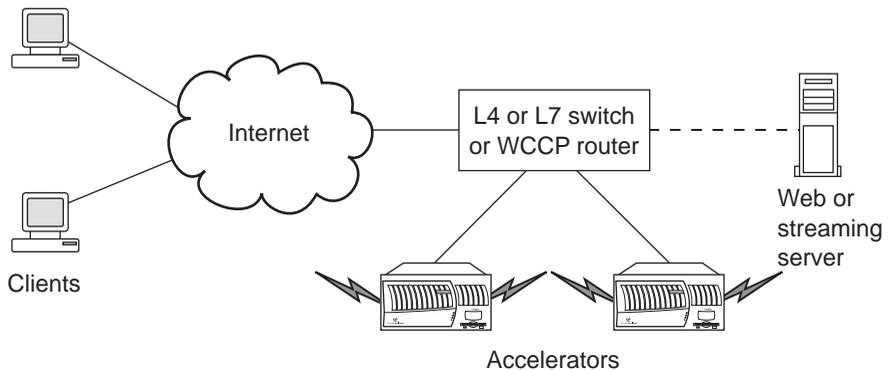
## Ways to provide client access to an accelerator

Two ways to provide client access to an accelerator are as follows:

- ◆ Set up transparent proxying on the NetCache appliance and on an L4 or L7 switch or a WCCP router
- ◆ Configure DNS to support an accelerator

## Using transparent proxying to direct traffic to an accelerator

When you are using an L4 or L7 switch or a WCCP router to provide access to an accelerator, you deploy the switch or router close to the accelerator. (In contrast, when an L4 or L7 switch or a WCCP router is deployed with a forward proxy, it is located close to the clients.) The following illustration shows the deployment of an L4 or L7 switch or a WCCP router and an accelerator.



The switch or router uses its hashing function to determine the accelerator to which to send the request. If both accelerators fail, the switch or router sends the request to the Web server or streaming server (as indicated by the dashed line in the illustration).

**Optimizing request distribution for an accelerator:** You can set a switch or WCCP router to optimize distribution for acceleration, as described in [“Request distribution with an L4 or L7 switch”](#) on page 30 and [“Request distribution with a WCCP router”](#) on page 39.

**An L7 switch and noncacheable objects:** If you deploy an L7 switch, you can configure it to directly access the server it is accelerating for objects that cannot be cached, for example, CGI and private pages.

**Allowing HTTP QuickTime encapsulation:** To allow QuickTime HTTP encapsulation, you must redirect TCP port 7070. However, the RealNetworks legacy protocol PNA also used TCP port 7070 and it is not fully supported by NetCache. Therefore, if you configure an L4 or L7 switch or a WCCP router to redirect TCP port 7070 requests to a NetCache appliance, the appliance might also receive PNA traffic. NetCache proxies all PNA traffic to the origin server.

## Using DNS to direct traffic to an accelerator

If you do not use transparent proxying to provide access to an accelerator, you can configure DNS to support client access to the accelerator. When you set up DNS, you configure the accelerator's IP addresses and IP address aliases in DNS so that the accelerator can receive requests for the servers it is accelerating. You could set up DNS in the following ways:

- ◆ The accelerator handles all requests for the servers it is accelerating. Many organizations want the accelerator to handle all the requests for the Web or streaming server. However, you might want to split requests between the accelerator and the Web server or streaming server in the following situations:
  - ❖ If the link between the client and the Web server or streaming server cannot handle the load of requests to the server
  - ❖ If neither the Web server or streaming server nor the accelerator can handle the full load
- ◆ Requests are split between the accelerator and the servers it is accelerating. This type of deployment provides failover because all requests are sent to the Web server or streaming server if the accelerator is unavailable. Some organizations add multiple accelerators over time, further splitting the load on the Web server or streaming server. The accelerator essentially becomes another Web server or streaming server and shares the load.

See the *Guide to Caching Protocols and Services* for more details about how to set up DNS to support an accelerator.

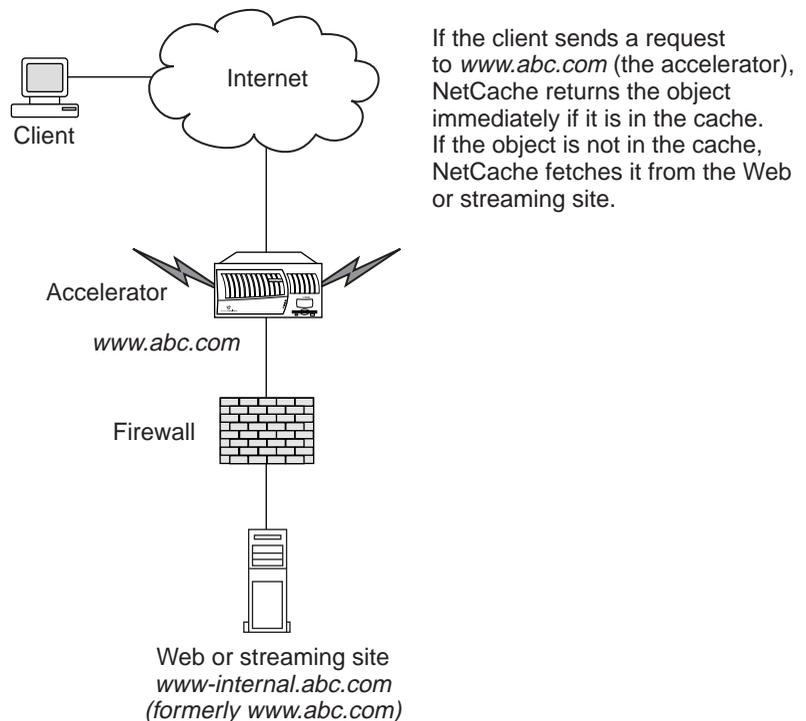
## Scenario: an accelerator outside the firewall

---

**About this scenario** In this scenario, the company wants to secure the data on the server (Web or streaming) from intruders. The company moved the server inside the firewall and deployed a NetCache appliance running as an accelerator outside the firewall. Because an intruder could not reach the server, the content would be secure. If an intruder broke into the accelerator, the intruder could not modify the data in any way.

### Deployment illustrated

The following illustration shows a single accelerator.



To make this deployment work, the company could use either of the following methods:

- ◆ Rename the Web server or streaming server and assign the accelerator the name that was previously assigned to the server. You can see in the previous

illustration that the accelerator is named *www.abc.com*, which was the name of the server. As far as the users know, they are sending requests directly to the server because the URL they are entering is the same.

- ◆ Define a local DNS name that NetCache can use to go to the IP address of the server that it is accelerating.

For example, *www.abc.com* would be advertised in the public DNS as the accelerator host name. The company would point NetCache to a DNS inside the local domain. The accelerator would have a virtual host entry for *www.abc.com* as the inbound host name and a matching destination of *www.abc.com*. The DNS that the accelerator points to would resolve *www.abc.com* as another address.

## Scenario: NetCache as a distributed Web site accelerator

---

**About this scenario** In this scenario, a large media site/hosting provider is interested in caching for the following reasons:

- ◆ Bandwidth use is growing. The organization wants to constrain the cost of bandwidth.
- ◆ The organization needs a way to scale sites that are growing rapidly.

A large amount of the data that clients request from this provider is static. Therefore, caching that data would result in significant savings in bandwidth use.

---

### Note

This example shows a deployment with Web accelerators. Streaming accelerators can also be distributed as shown in this example.

---

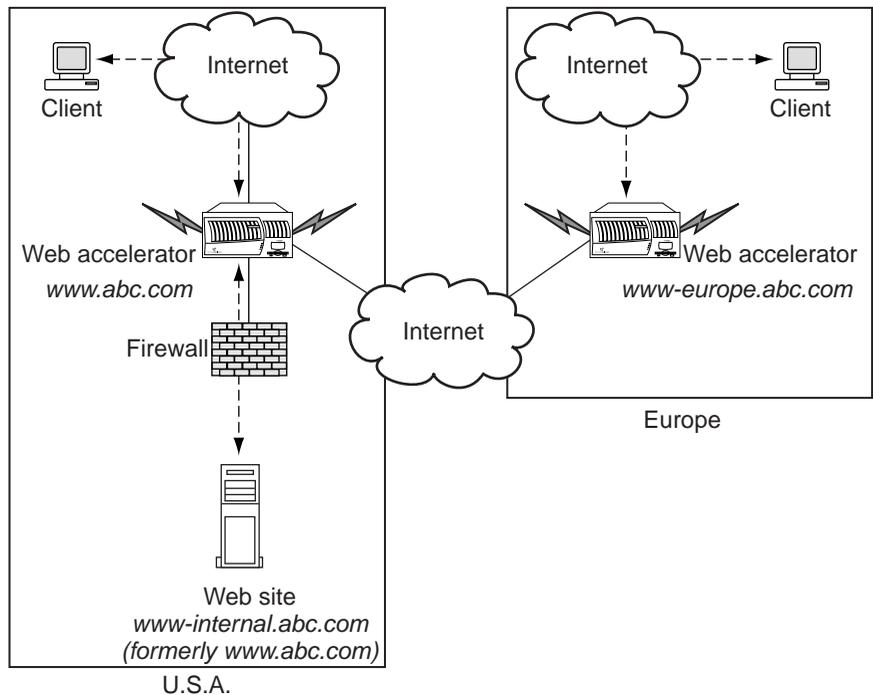
### Organization's requirements

The organization's requirements are as follows:

- ◆ The caching system must be reliable.
- ◆ The content served must be *fresh*; that is, if the content of a Web page changes, users requesting that page must be able to get the newest version very soon after it is changed.
- ◆ The media site/hosting provider must be able to log access to the data, for example, to bill an advertiser or perform a security audit.

### Deployment illustrated

The following illustration shows a deployment for this provider that includes two NetCache appliances running as Web accelerators.



**Goals for locating the Web accelerators:** The U.S.A. side of the illustration is like that in “[Scenario: an accelerator outside the firewall](#)” on page 174, with the accelerator outside the firewall. In this scenario also, the goal of deploying the Web accelerator outside the firewall and the Web server inside the firewall is to secure the Web site content.

A Web accelerator is deployed in Europe so that data from the Web site can be cached locally, thereby saving expensive bandwidth. Initially, the European Web accelerator must obtain the Web objects, and incur bandwidth cost. However, after the objects are in the cache in Europe, the objects can be served locally until they become stale. If the Web accelerator determines that an object is stale, it fetches a newer version of the object.

**About the European Web accelerator:** The provider can configure the European NetCache appliance in two different ways:

1. Identify the Web server *www-internal.abc.com* as the Web server for the European Web accelerator, just as it did for the U.S.A. Web accelerator. In this case, if the European Web accelerator does not have a requested object in its cache, it goes to the Web server to obtain the object.

2. Identify the U.S.A. Web accelerator as the “Web server” for the European Appliance Web accelerator. In this case, if the European Web accelerator does not have a requested object in its cache, it goes to the U.S.A. Web accelerator to obtain the object. The European Web accelerator does not go to the real Web server, *www-internal.abc.com*. This method is preferable because the provider does not have the cost of the slow firewall or the bottleneck that prompted the provider to deploy the U.S.A. Web accelerator.

**Sending requests to the appropriate Web accelerator:** The provider needs to determine the method to use so that user requests are sent to the appropriate Web accelerator, that is, the accelerator that the provider wants the each client to use. This request distribution is outside the control of NetCache.

Different methods for ensuring that requests are sent to the correct Web accelerator are as follows:

- ◆ Notify users about the URL they must enter so that the request is sent to the Web accelerator that the provider wants.  
Users in the U.S.A. access the Web server by entering *www.abc.com*. Users in Europe must be instructed to enter the URL for the European Web accelerator, *www-europe.abc.com*, not the URL for the U.S.A. Web accelerator. Otherwise, their requests are sent to the U.S.A. Web accelerator, with a higher bandwidth cost.
- ◆ Use a product, such as a switch or Server Load Balancer (SLB), that can distribute requests to the correct Web accelerator.
- ◆ Use DNS routing.  
3DNS (from F5 Lab), Distributed Director (from Cisco), and WSD-DS (from RND Network) are examples of DNS servers that assign IP addresses to ensure that the request is sent to the closest Web accelerator.

**Logging:** Logging features in NetCache enable system administrators to identify clients that are requesting objects and to use that information for billing. ContentReporter software can be used to gather logs from NetCache appliances. That data can then be imported into third-party reporting applications, for example, to create billing reports.

## Scenario: multiple accelerators accelerating a single server

---

**About this scenario** In this scenario, the Web or streaming site is very big and very busy. One server cannot handle the amount of traffic. One solution is to have multiple servers handle the site and perform DNS round robin or server load balancing to distribute requests over the site. The problem with this method is the need to keep the replicas synchronized with each other.

Another solution is to use a single Web server or streaming server, as applicable, to handle the site and use multiple accelerators to accelerate the server. You would use one of the methods described in Section C, “[Direct \(nontransparent\) client access methods](#),” on page 51 to distribute the requests over the accelerators. For example, you could use a Server Load Balancer (SLB) or transparent proxying with an L4 or L7 switch or WCCP router. The SLB, switch, or router would distribute the requests over the accelerators.

### **Freshness of content**

Replication works correctly in this case because the accelerators pick up new content as the objects’ Time-To-Live in the cache expires. Even in the case of Web pages whose Time-To-Live is a few minutes, if the pages are accessed frequently, considerable load can be taken off the server if many requests are received within that period of time.

---

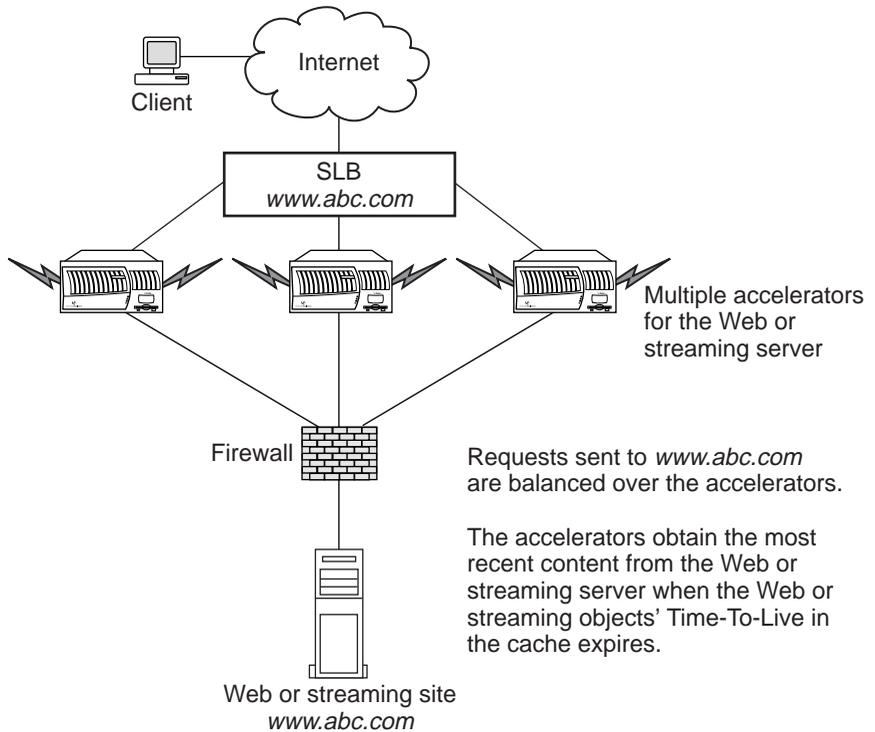
### **Note**

NetCache does not delete an object from the cache when its Time-To-Live expires. The next time a request for an object is received, NetCache checks the server to determine whether a newer version of the object exists. If the object did not change and the server can understand If-Modified-Since queries, the verification of the object’s status is less expensive than obtaining the object from the server.

---

### **Deployment illustrated**

The following illustration shows multiple accelerators accelerating a single server (Web server or streaming server). In this scenario, an SLB is used to distribute requests over the accelerators. Alternatively, you could use an L4 or L7 switch or a WCCP router.



**If you are using DNS round robin and NetCache appliances**

If you are using DNS round robin and NetCache appliances, you can use NetCache takeover pairs to enable one accelerator to take over proxy services for another if one accelerator in the pair fails. See Section B, “[Failover by using NetCache appliance takeover pairs](#),” on page 78.

## Scenario: single accelerator accelerating multiple servers

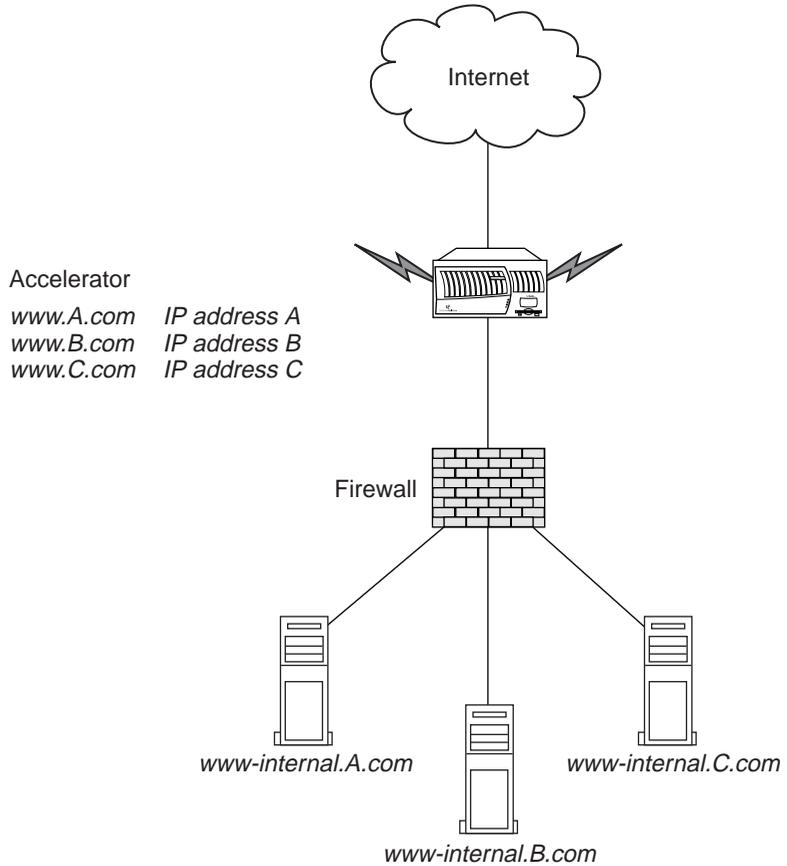
---

**About this scenario** In this scenario, one accelerator is deployed to accelerate three servers, which could be Web servers or streaming servers. Only one network card is installed in the accelerator, and multiple IP addresses are associated with the card. The administrator's goal is to have a separate interface on the accelerator handle requests for each server.

This deployment differs from the scenarios in which an accelerator was accelerating a single server because the administrator must provide a way for the accelerator to distribute requests to the correct servers.

### **Deployment illustrated**

The following illustration shows a single accelerator servicing multiple servers.



Notice that in this scenario the accelerator was assigned multiple host names and IP addresses. When you assign multiple host names and IP addresses to the accelerator, you can configure accelerator options in the Appliance Manager to ensure that NetCache can send cache misses through a particular interface to a specific server.

---

**Note**

NetCache supports a single IP interface and a single IP address. NetCache also supports one IP address and any number of IP address aliases for the accelerator, no matter how many cards and interfaces the accelerator has.

---

# Scenario: allowing limited access from another company

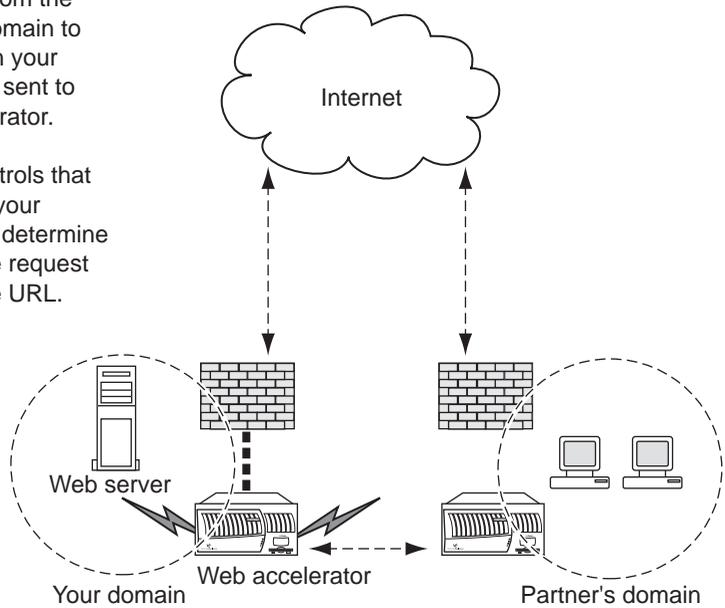
**About this scenario** You can allow certain clients in a partner company to access a predetermined set of URLs on specified servers in your domain. To set this up, you configure your NetCache appliance as an accelerator for a Web server or streaming server. In this case, when NetCache is configured as an accelerator, it is part of the firewall strategy.

## Deployment illustrated

In the following illustration, the NetCache appliance in your domain is configured as an accelerator. It appears to the NetCache appliance in the partner's domain as a server.

Requests from the partner's domain to URLs within your domain are sent to your accelerator.

Access controls that you set on your accelerator determine whether the request reaches the URL.



## Requirements

The requirements for configuring the NetCache appliance to enable a partner company to have limited access to your Web server or streaming server are as follows:

- ◆ Configure the NetCache appliance in your domain as an accelerator.

- ◆ Set access controls on the accelerator in your domain to control the IP addresses from the partner's domain that can access certain URLs in your domain. Be sure that when you set up your access controls, you are not compromising your firewall security by allowing access from other outside companies.

## Scenario: accelerator for an historical stock performance Web site

---

**About this scenario** This scenario describes locating a Web accelerator inside the firewall so that most client access to a Web site can be served without traversing the expensive firewall. You can use NetCache as an accelerator in any situation in which accessing the Web server (or streaming server) is expensive, not just with firewalls.

In this scenario, a Web site offering charts of historical stock performance would not want to regenerate an updated set of charts for every ticker symbol at the close of every market day. The reason is that generating the graphic is relatively expensive and most of the charts would probably not be referenced before the next market close renders them obsolete. However, a chart that is referenced is likely to be referenced again soon, and it is costly to regenerate the chart for each reference. In this case, you can configure NetCache as a Web accelerator so that the real server needs to generate a given chart only once. Service to the customer is faster, and the use of the chart-rendering server is less expensive.



**About this chapter** This chapter describes deploying NetCache as a news cache and includes a scenario in which NetCache is deployed as a news cache.

**Chapter contents** This chapter contains the following sections:

- ◆ [“Introduction to NetCache news caching”](#) on page 188
- ◆ [“Interaction between the news cache and news server”](#) on page 191
- ◆ [“Software, clients, and features that NetCache supports”](#) on page 193
- ◆ [“About news data that is cached”](#) on page 194
- ◆ [“Deployment considerations”](#) on page 195
- ◆ [“Scenario: news caches at an ISP Data Center and POPs”](#) on page 199

# Introduction to NetCache news caching

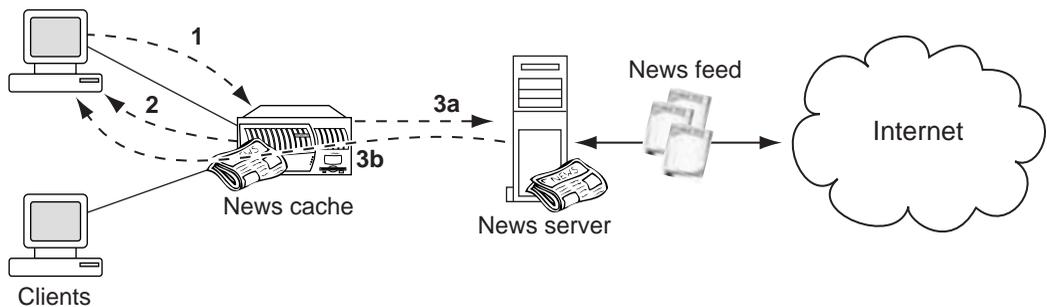
## What is a NetCache news cache?

Network Appliance uses the term *news cache* to describe a NetCache appliance that is configured to provide news caching service for news servers that support the Network News Transport Protocol (NNTP). A news cache requests news data from a news server on behalf of a client and caches that data. The news cache and news server communicate through NNTP.

A news cache might also be configured for DNS caching.

## Path that a news request takes

The following illustration shows the flow of a news request from the client to the news server by way of the news cache and from the news server back to the client.



1

Users use client newsreaders to send NNTP requests to a news server. The request goes to the news cache.

2

If the news cache has the requested data in its cache, it returns that data to the client directly.

3a, 3b

If the news cache does not have the requested data in its cache, it initiates a connection with the news server (if one is not already available), fetches the requested data from the news server (3a), returns the data to the client (3b), and then caches the data in its cache if the data is cacheable.

## News distribution to news servers and news caches

In NNTP news transport, the primary role of news servers is to receive news from and send news to other news servers. One news server creates a connection to a downstream news server and offers news to that server. The downstream server responds as to whether the upstream server should send particular articles. News servers routinely accept very large *news feeds* from other servers, which requires an administrator to closely monitor disk usage on the news storage device.

The news cache, however, cannot accept a news feed. It receives news only when it establishes a connection to the news server. If a client requests an article that is not in the news cache, the news cache initiates a connection to the news server to fetch the article. As shown in the previous illustration, the fetched article is returned to the client and is cached by the news cache.

## Number of news servers to which a news cache can connect

A NetCache news cache can connect to only one news server. Multiple news caches can connect to the same news server, however.

## Benefits of deploying a news cache

Including a news cache in your news environment can have the following benefits:

- ◆ Reduced bandwidth consumption over WANs

If news clients have been connecting to a server over a WAN, deploying a news cache close to the clients reduces traffic over the WAN. The reason is that multiple requests for the same news data can be fulfilled by the cache rather than having to send each request across the WAN to the news server.

Without a news cache, each client would open a TCP/IP connection to the news server for every news session, resulting in a high volume of traffic over the WAN.

Even in the case of cache misses, you save bandwidth by deploying a news cache. The cache can send many cache misses over a connection that it establishes with the news server. TCP/IP setup is not necessary for each cache miss.

- ◆ Improved quality of service

If you deploy a news cache close to your clients, you can provide news more quickly to your users.

- ◆ Increased scalability

For some NetNews software, a point is reached at which you cannot easily expand the news server to handle more news requests. Deploying a news cache can significantly reduce the server resources required for a given

number of clients. The reason is that a news cache can handle many hundreds of connections, whereas Reader servers typically can handle only a small number of connections.

---

**Note**

---

A Reader server is the component of the news server that handles client connections for news. The Reader server software might be installed on the same computer as other news server components, or it might be on a dedicated computer.

---

◆ More efficient handling of requests

Requests for the same article are handled more efficiently by a news cache than by some news servers. For some news servers, it is “expensive” to handle multiple requests for the same article. The reason is that each time a client requests an article, a news server must perform a lookup through its entire directory structure.

◆ “Lights out” administration of a news cache

A news cache does not accept or send news feeds. Therefore, a news cache administrator does not have to closely monitor disk usage, as is necessary for a news server. Nor does the administrator need to manage any issues in regard to the news cache’s interaction with news servers. After you configure your news caches, administration of the news caches is minimal.

## Licensing requirements

To deploy a NetCache appliance as a news cache, you must obtain a license from Network Appliance for NNTP.

## About the cost to process news requests

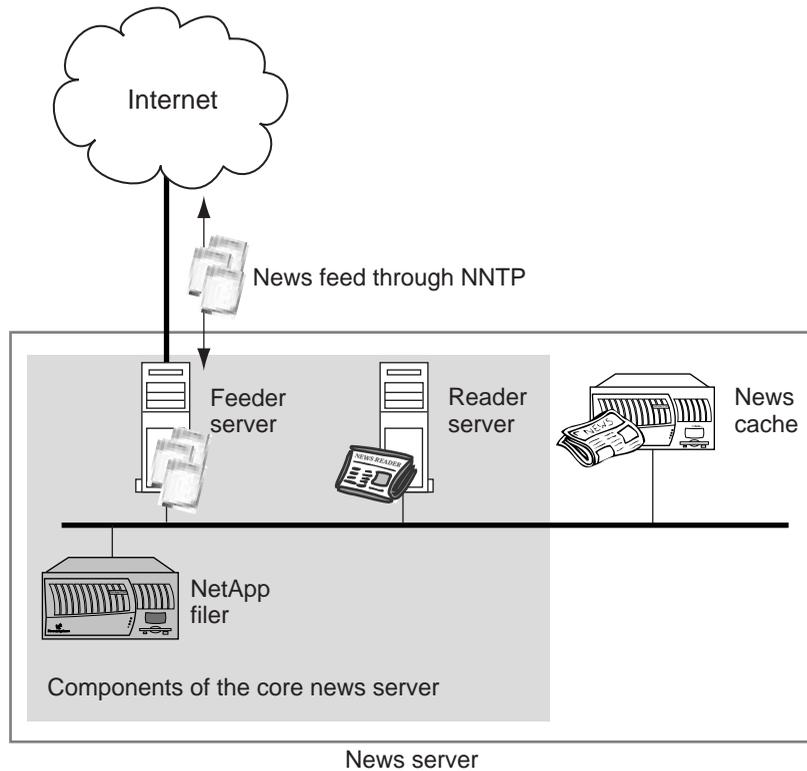
The most widely deployed news server software requires approximately 1 MB per client connection. Taking file system buffers, the operating system, and other overhead into consideration, a Reader server with 1 GB of memory might struggle to handle about 500 client connections. NetCache is far more efficient. Using similar hardware, NetCache can handle thousands of client connections. With a news cache deployed, significantly fewer connections to the news server are required because NetCache can resolve duplicate requests from the cache.

## Interaction between the news cache and news server

---

### How NetCache fetches news from the news server

The news cache is an extension of the news server. When the news cache does not have the requested news in its cache, it connects to the Reader server component of the news server through NNTP. The Reader server obtains the data that the news cache requests from the news storage device, in this case, a NetApp® filer.



---

### Note

The Feeder server and Reader server components of the news server might be on separate servers, as shown in the previous illustration, or they might be on the same server.

---

## **Updating news cache data**

NetCache provides controls that enable you to set the rate at which the news cache checks the news server for new articles and canceled articles.

## **News cache behavior if the news server is unavailable**

The news cache continues to service news requests, even if the news server is unavailable. If the client requests news data that is in the cache and has not yet expired, NetCache returns that data to the client. If the client requests news data that is not in the cache, NetCache returns the message “400 service discontinued, server disconnected” to the client. The connection between the client and the cache is then closed.

If the news server is unavailable while NetCache is checking for new or canceled articles, the client receives the “400 service discontinued, server disconnected” message, and the connection between the news cache and the client is closed.

News caching resumes as usual when the news server is available again.

## Software, clients, and features that NetCache supports

---

### NetNews software that NetCache supports

NetCache supports news servers running the following NetNews software:

- ◆ INN
- ◆ bCandid - Typhoon
- ◆ Diablo

### Types of clients that NetCache supports

NetCache supports the following client news software:

- ◆ tnr
- ◆ Outlook Express
- ◆ Free Agent
- ◆ Netscape
- ◆ Most other NNTP-compliant clients

### NetCache features available for use with news caching

You can use the following NetCache features with news caching:

- ◆ Authentication

For more information, see the *Security Guide*.

- ◆ Transparent proxying

You can deploy news caching transparently. For more information about transparent proxying, see Chapter 2, “[Strategies for Client Access to NetCache](#),” on page 17 and the *Administration Guide*.

Additionally, features such as those for setting up your NetCache appliance on the network and ensuring security for your NetCache appliance are available for news caching.

## About news data that is cached

---

### What a news cache caches

NetCache running as a news cache caches the following:

- ◆ News OverView (NOV) data for an article  
Overview data is header information about the article, which includes information such as the message ID (the unique identifier for the article), date, and subject.
- ◆ Group list  
The group list is a list of available newsgroups. Each line in the list includes the name of the newsgroup, the first and last known article in the newsgroup, and a flag indicating whether the server allows posting articles to the newsgroup.
- ◆ Group information  
The group information consists of the numbers of the first and last articles in the group and an estimate of the number of articles in the group.
- ◆ News articles  
NetCache caches the entire article that the client requested, which consists of the header and the body. The article header usually contains more detailed information about the article than is available from the Overview data.

### What a news cache does not cache

NetCache proxies commands for which it receives only partial data, for example, the NEWNEWS command, and sends them to the news server. NetCache does not cache commands sent by the news server.

### Keeping news fresh

NetCache provides controls that enable you to set the rate at which NetCache checks the news server for new articles and canceled articles.

## Deployment considerations

---

### Strategies for client access to your NetCache appliances

Setting up client access to news caches and distributing requests over news caches are much the same for Web caches and news caches. You can set up client access to a news cache in the following ways:

- ◆ Through transparent proxying
- ◆ Through a nontransparent client access
  - You would need to manually configure client browsers to point directly to one of the following:
    - ❖ A NetCache appliance
    - ❖ A device such as a Server Load Balancer (SLB) that is in front of the NetCache appliances and is configured to be aware of the NetCache appliances

See Chapter 2, “[Strategies for Client Access to NetCache](#),” on page 17.

### Determining how many news caches you need

Some considerations that affect how many NetCache appliances you need are as follows:

- ◆ Your goals for deploying news caching
  - If, for example, your goal is to reduce bandwidth consumption, deploying a news cache close to your users can reduce the volume of news requested over a WAN.
- ◆ The number of news requests that you expect
- ◆ Whether you want the same NetCache appliance to handle traffic of multiple protocols
  - You can configure a NetCache appliance to operate in more than one mode—for example, as a Web cache and a news cache. However, if you expect your news cache to handle a substantial load, for example, the volume of an ISP Data Center, Network Appliance recommends that you dedicate a NetCache appliance to news caching. By doing so, you avoid conflicts between different software programs, and performance and reliability are better.
- ◆ Whether you can afford to have news service unavailable, that is, whether you need a failover strategy

Your Network Appliance sales engineer can help you determine the number of NetCache appliances you need to meet your goals and handle the number of client connections that you expect for news traffic.

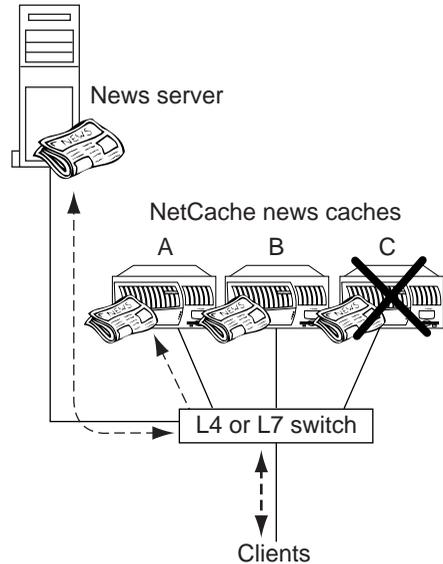
## **Failover strategies for news caching**

If you need uninterrupted news service, you can implement one or more of the following failover strategies:

- ◆ **Deploy a switch failover pair if you are using transparent proxying**  
If you are using an L4 or L7 switch and you are concerned about failure of your switch, you can deploy two switches in a failover pair.
- ◆ **Add an extra news cache for redundancy**  
When determining the number of news caches that you need, plan for one news cache in addition to the number you think you need to handle the news traffic. That way, if one news cache goes down, the remaining news caches can still handle the volume of news traffic.
- ◆ **Failover to the Reader server**  
It might not be possible to set up failover in case all news caches go down. Typically, the load of connections that a Reader server can handle is vastly less than the number of connections that a news cache can handle. If your Reader server can handle the volume of client requests that your news caches can handle, you can set up your deployment so that news requests fail over to the Reader server if all news caches go down. However, if your Reader server cannot handle the traffic that your news caches can handle, your news service will be unavailable or news service performance will be poor if the Reader server takes over.

The following illustration shows an example of news caches and failover when transparent proxying is deployed with an L4 or L7 switch.

If the switch is configured to be aware of A, B, and C and one goes down, the switch distributes requests over the remaining systems.



You could set up your switch so that if all news caches go down, news service fails over to the news server. However, because news servers typically cannot handle nearly the number of connections that a news cache can, you run the risk of overloading your news server.

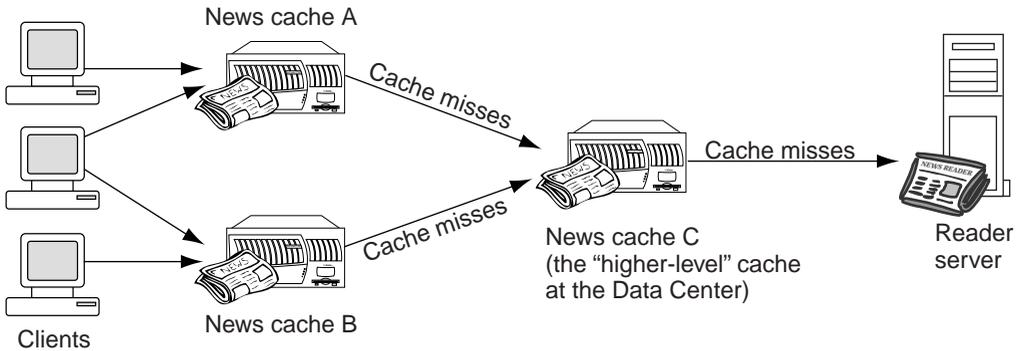
### Load balancing across multiple news caches

If you want to load balance the news service traffic across multiple news caches, you can do the following:

- ◆ Advertise a single IP address for the news service and use round-robin DNS to load balance news requests over multiple news caches.
- ◆ Advertise a single IP address for the news service and use an L4 or L7 switch to load balance news requests over the news caches.

### Strategy for maximizing the news hit rate and protecting the Reader server

You can deploy your news cache so that one news cache points to another news cache, and the “higher-level” news cache points to the Reader server component of the news server. This type of deployment enables you to add an extra “level” of news caching, which can result in a better cache hit rate and protect your news server’s Reader server from traffic overload. In the following illustration, News cache A and News cache B are deployed at the POPs and News cache C is deployed at the data center.



Deploying news caches in a hierarchy is helpful for an ISP with many POPs (points of presence). If the news caches at the POPs are experiencing a low hit rate, the hit rate can most likely be increased by deploying a news cache in the ISP's data center also and having the POPs send cache misses to the data center news cache. Although cache misses from the POPs could be sent directly to the Reader server, the Reader server might not be able to handle the number of connection attempts.

To implement this type of deployment, in your news cache configuration you identify the news server (or higher-level news cache) to which the news cache connects. This identification enables the news cache to send cache misses to the news server (or higher-level news cache).

### Caution about deploying chains of news caches

Deploying one news cache to point to another news cache, as shown in the previous illustration, can help protect your Reader servers from too much traffic. However, avoid deploying a chain of news caches for the following reasons:

- ◆ Delays in resolving news requests can occur.  
The configuration of each news cache includes settings for how often NetCache is to check the news server for new articles and how often NetCache is to check for canceled articles. If there is a chain of caches, the time period for checking for new articles is the total of the specified time period on all the caches. Likewise, the time period for checking for canceled articles is the total of the specified time period on all caches.
- ◆ Inconsistencies in expiration times for news articles can occur.  
News articles become inconsistent on the news caches if the time periods for the check for new articles and the check for canceled articles are not set the same way on all the news caches.

## Scenario: news caches at an ISP Data Center and POPs

---

**About this scenario** This section provides an example of an ISP that provides both Web service and news service to its customers. The ISP has a number of POPs. The news server is at the Data Center.

### Reasons for deploying news caches

The ISP decided to deploy a news cache at the Data Center to gain the following benefits:

- ◆ Reduce the number of Reader servers that are necessary to handle the volume of news traffic. Often, one news cache can handle the load of multiple Reader servers.
- ◆ Decrease the “expense” of the news server handling multiple requests for the same article. Currently, with the news servers deployed in this ISP example, each time a client requests a particular article, the Reader server must search through the directory structure for the article. NetCache can handle such requests more efficiently than the Reader servers at this ISP.

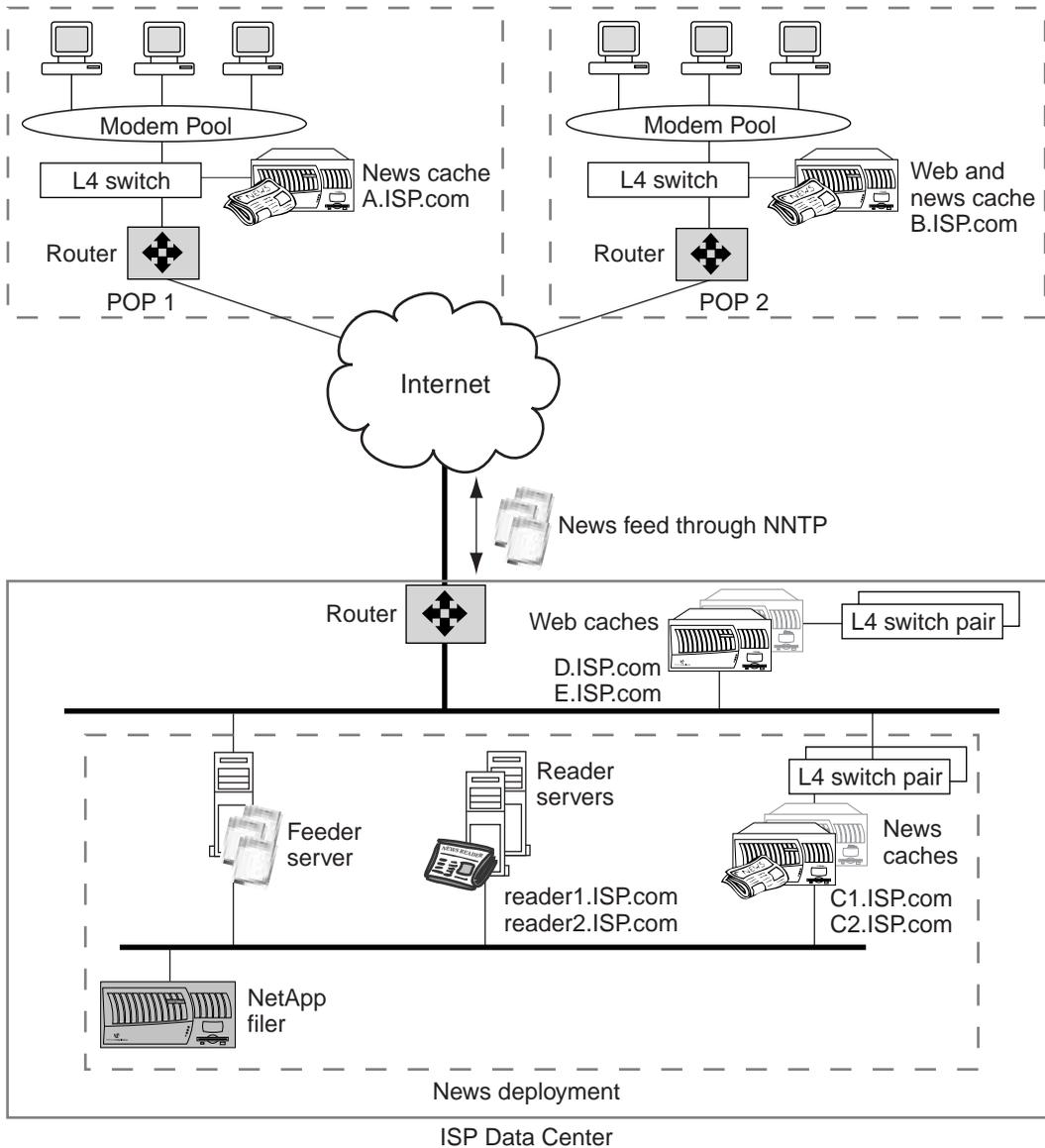
The ISP decided to deploy a news cache at each POP to gain the following benefits:

- ◆ Reduce the amount of bandwidth required for news service. Currently, the POP must send all news requests to the Data Center.
- ◆ Save bandwidth and improve quality of service by fulfilling client requests faster.

The administrator of the POP is concerned with quality of service and bandwidth usage. Sending every news request to the ISP Data Center uses considerable bandwidth and increases the response time. Deploying a news cache at the POP reduces bandwidth usage and response time because multiple requests for news can be resolved by the POP news cache. Additionally, any connectivity problems that might occur between the POP and Data Center will have less impact on news service.

### Deployment illustrated

The following illustration shows the news caching deployment for the ISP. For simplicity, only two POPs are shown in the illustration.



## **Types of NetCache appliances that were deployed**

The ISP deployed NetCache appliances as follows:

- ◆ **Deployment at the POPs**

The NetCache appliance at POP 1 handles only news traffic, whereas the NetCache appliance at POP 2 handles both Web traffic and news traffic. The deployments shown at the POPs are intentionally different to illustrate that each type of deployment is possible. The ISP could also have dedicated news caches and dedicated Web caches at the POPs.

- ◆ **Deployment at the Data Center**

All NetCache appliances are dedicated to either news caching or Web caching. The reason is that the Data Center NetCache appliances must handle higher volumes of each type of traffic than the NetCache appliances at the POPs handle. The Data Center appliances must handle cache misses from many POPs. Additionally, the ISP wanted the maximum level of performance and reliability possible.

## **Why POP caches are not sending cache misses directly to the news server**

This scenario describes news caches at the POPs sending cache misses to a news cache at the ISP Data Center, rather than the POPs sending cache misses to the news server directly. The ISP took this approach because the Reader server could not handle the number of connections that would result from many POPs sending cache misses to it.



**About this chapter** This chapter discusses using a NetCache appliance as an Internet Content Adaptation Protocol (ICAP) client. In this role, a NetCache appliance interacts with an ICAP server to adapt content to local policies.

**Chapter contents** This chapter contains the following topics:

- ◆ Section A, “[NetCache support for ICAP](#),” on page 204
- ◆ Section B, “[ICAP deployment considerations](#),” on page 212
- ◆ Section C, “[ICAP service scenario](#),” on page 221

## Section A: NetCache support for ICAP

---

**About this section**      This section defines ICAP and describes how NetCache interacts with ICAP servers to provide value-added services in remote offices.

**Contents of this section**      This section contains the following topics:

- ◆ “[Learning about ICAP](#)” on page 205
- ◆ “[When ICAP services are invoked](#)” on page 208

## Learning about ICAP

---

### What is ICAP?

ICAP is an open-ended protocol that is designed for easy extensibility of the capabilities of a proxy-cache server and offloading of specific Internet-based content to dedicated servers for adaptation of that content.

The NetCache ICAP feature works in conjunction with ICAP servers running ICAP applications to provide content adaptation services. When configured for ICAP service, a NetCache appliance directs content, or requests for content, to an ICAP server that can provide the necessary content adaptation service. The ICAP server then executes the service and responds to the NetCache appliance with the adapted content or request.

### Examples of ICAP services

ICAP is an emerging technology that supports flexible and diverse services for content adaptation. Network Appliance is collaborating with vendors on an ongoing basis as they add ICAP services.

Examples of current ICAP services are described in the following paragraphs.

**Virus scanning:** Historically, the receiving network or computer has often been responsible for virus scanning, which can result in the same object being scanned multiple times. With ICAP virus scanning, the scanning takes place before the object is passed to the client.

For example, if an ICAP virus scanning service is deployed and a user requests an *.exe* (executable) file, NetCache passes the *.exe* file returned by the origin server to an ICAP server. The ICAP server checks the *.exe* file for viruses and removes any viruses it finds or denies the user access. After the ICAP server performs its service, it returns the (possibly modified) content to the NetCache appliance for caching and delivery to the client.

**Content filtering:** Content filtering provides the ability to check requests to determine if the type of content requested is authorized.

Historically, content filtering has been performed by a proxy-cache server based on information the administrator manually configured or based on content filtering database files downloaded to the proxy-cache server, for example, SmartFilter or WebWasher DynaBLocator.

ICAP content filtering applications expand your choices for what to use for content filtering. You can easily switch ICAP content filtering applications later, if you find a better solution after deploying your proxy-cache server and ICAP server.

Unique to ICAP is that ICAP content filtering can work in conjunction with other ICAP services. For example, you can feed the results of content categorization returned by an ICAP content filter into a virus scanning ICAP service.

## Benefits of ICAP

Value-added services through ICAP provides the following benefits:

- ◆ ICAP is an open protocol. Any server or application provider can develop ICAP services. New applications are being developed on an ongoing basis.
- ◆ Access to the Web servers is faster because resource-intensive services are off-loaded to servers dedicated to content adaptation.
- ◆ Implementation with ICAP is scalable, supporting 100 to 100,000 desktops.
- ◆ Many services can be supported without deploying chains of proxy-cache servers, which can affect performance.
- ◆ Overhead and management to add additional services is low.

## Possible issues for ICAP

Possible issues for ICAP are as follows:

- ◆ Adding ICAP services to your network might increase network traffic, as described in “[Considering network traffic when determining ICAP services](#)” on page 217. Therefore, you need to weigh the benefits of providing ICAP services against the impact on network performance.
- ◆ Increased latency might occur as a result of trips on the network to and from the ICAP servers.

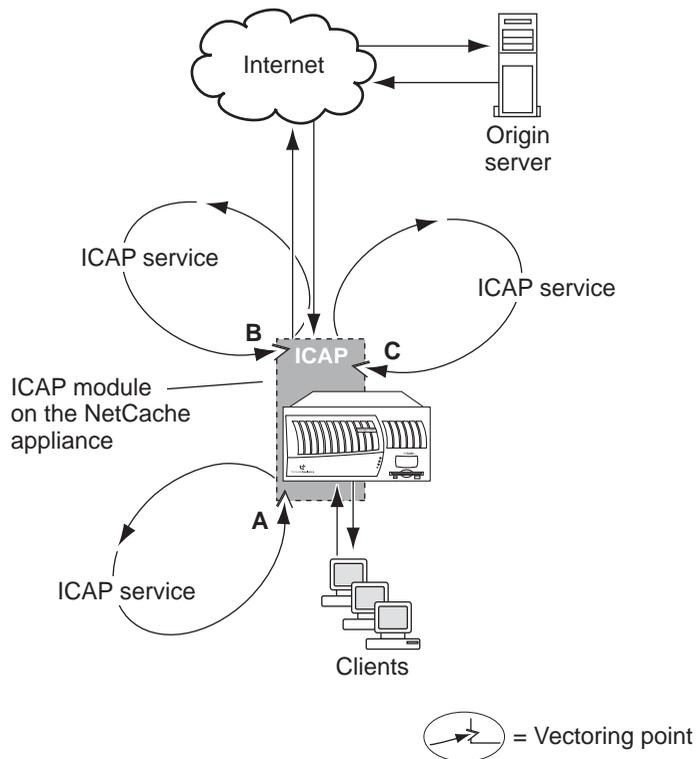
## Version supported

NetCache supports ICAP version 1.0.

## When ICAP services are invoked

### Points at which the ICAP server can be invoked

NetCache can send a request to an ICAP server at either of two different points before the request reaches the origin server, as well as sending content to the ICAP server after the origin server returns the content. Therefore, you can include ICAP services in your deployment that are performed on the request before it reaches the origin server, for example, content filtering. The following illustration shows the points at which a NetCache appliance can invoke an ICAP service.

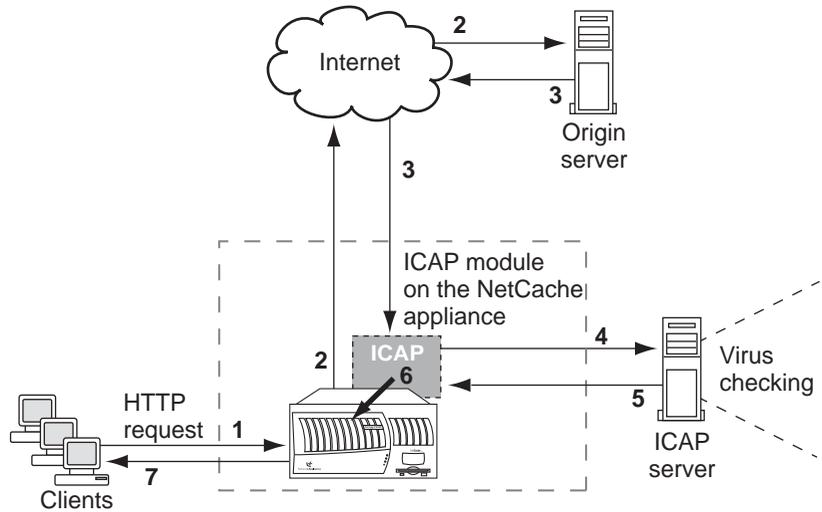


The following table describes the possible vectoring points for ICAP service in the request flow shown in the previous illustration (labeled A, B, and C), that is, the points at which the NetCache appliance can invoke ICAP services. The same content can be adapted by multiple ICAP services. Therefore, multiple vectoring points could be invoked during the flow of the same request.

Vectoring point	Common services at this vectoring point	Example of use
<b>Request modification stage</b>		
(A) After NetCache checks authentication and ACLs but before it checks its cache	Content filtering	Assume that you want to set a policy so that only employees in the finance department can access stock quotes on the Internet. You set the NetCache ICAP feature to send requests, at this vectoring point, to an ICAP content filtering service that is set up to block access to stock quotes. You also use ACLs to allow access to stock quotes for the finance department employees but no other employees.
(B) After NetCache checks its cache and a cache miss occurs	Content filters that are being applied the same way to all users	Assume that you want to block access to sexually explicit sites for all users. You could set the NetCache ICAP feature to send requests to an ICAP content filtering service at this vectoring point.
<b>Response modification stage</b>		
(C) After the origin server returns the content that the client requested	Virus scanning	See “ <a href="#">Example: modification of responses</a> ” on page 209 for a detailed example of the interaction between the NetCache appliance and an ICAP server at this vectoring point.

**Example:  
modification of  
responses**

The following illustration provides a detailed look at the flow of an HTTP request when the vectoring point for an ICAP service is in the response modification stage. In the response modification stage, NetCache passes the response from the origin server to an ICAP server for processing, in this case, for virus scanning.



The following table describes the flow of a request.

Stage	Description	
1	The client makes an HTTP request to download an executable file, which is sent to the NetCache appliance.	
2	<b>If...</b>	<b>Then...</b>
	If the NetCache appliance has the requested object	The appliance sends the object to the client. No ICAP transaction for virus scanning is performed. (Other ICAP services might already have been performed before the content was cached.)
	If the NetCache appliance does not have the requested object	The appliance sends the request to the origin server and the process continues, as described in the following steps.
3	The origin server returns the requested object to the NetCache appliance.	

<b>Stage</b>	<b>Description</b>
<b>4</b>	The ICAP module on the NetCache appliance passes the object to the ICAP server without handling the object.
<b>5</b>	The ICAP server performs content adaptation services. The ICAP configuration on the NetCache appliance controls the ICAP service that the ICAP server performs.
<b>6</b>	The ICAP server sends the object, possibly modified, to the NetCache appliance.
<b>7</b>	<p>The NetCache appliance caches the object and sends the object to the client. These operations are performed concurrently. Therefore, caching adds almost no cost in latency.</p> <p>When it receives additional requests for the same object, NetCache serves the ICAP-modified object from the cache.</p>

## Section B: ICAP deployment considerations

---

**About this section** This section provides information that will be helpful when you plan your ICAP deployment.

**Contents of this section** This section contains the following topics:

- ◆ [“Deployment overview”](#) on page 213
- ◆ [“Feature summary”](#) on page 214
- ◆ [“Planning for ICAP services and ICAP servers”](#) on page 216
- ◆ [“Security and ICAP”](#) on page 219

## Deployment overview

---

### Deployment planning tasks

Planning for deploying ICAP services involves the tasks listed in the following table.

Task	See...
Determine the ICAP services that you want to provide.	<a href="#">“Planning for ICAP services and ICAP servers”</a> on page 216 <a href="#">“Considering network traffic when determining ICAP services”</a> on page 217
Determine how many ICAP servers you need for the services that you have selected and where to locate them.	<a href="#">“Planning for ICAP services and ICAP servers”</a> on page 216 <a href="#">“Network link recommendation”</a> on page 218 <a href="#">“Location of the NetCache appliances and ICAP servers”</a> on page 218
Ensure that you have a strategy for failover of ICAP service if an ICAP server becomes unavailable.	<a href="#">“Failover if an ICAP server goes down”</a> on page 218
Ensure that security to support ICAP transactions is adequate.	<a href="#">“Security and ICAP”</a> on page 219

See the *ICAP Services Guide* for information about NetCache support for ICAP services.

## Feature summary

---

### Features

The following table summarizes ICAP features.

Element	Feature
Protocols from browsers and origin servers vectored to ICAP servers	HTTP, streaming media encapsulated in HTTP, limited FTP
When ICAP services can be invoked (vectoring point)	<p>At any of following vectoring points in the request path:</p> <p>Request modification</p> <ul style="list-style-type: none"> <li>◆ NetCache can send the request to the ICAP server before checking its cache for the object.</li> <li>◆ If a cache miss occurs, NetCache can send the request to an ICAP server.</li> </ul> <p>Response modification</p> <ul style="list-style-type: none"> <li>◆ After the origin server responds to the request, NetCache sends the request to the ICAP server. This is the only vectoring point for FTP.</li> </ul>
Access controls	<p>NetCache ACLs are used to implement secure policies and control access to services. NetCache ACLs must be enabled for ICAP services, and rules can be applied to do the following:</p> <ul style="list-style-type: none"> <li>◆ Direct requests for ICAP services</li> <li>◆ Control the order of ICAP services</li> <li>◆ Control who receives content</li> </ul>
Number of ICAP servers supported	Multiple
Load balancing	Load balancing between the same service running on multiple ICAP servers is available.

## Section B: ICAP deployment considerations

<b>Element</b>	<b>Feature</b>
NetCache logging of activity	An ICAP log and status information are available.

The following sections provide information that will help you determine the features you need for your organization.

## Planning for ICAP services and ICAP servers

---

### Determining the ICAP services you need

To determine the ICAP services you need, consider the following:

- ◆ Your organization’s requirements for content adaptation
- ◆ Whether using ICAP services for some or all of your content adaptation needs is the best strategy

In some cases, ICAP services enable you to adapt content in ways that have not been possible previously. In other cases, ICAP services might provide a superior alternative to methods you are using now. See “[Examples of ICAP services](#)” on page 205 for historical remarks about some services.

---

### Note

To learn about ICAP applications that are available at a given time, check <http://www.i-cap.org>. Vendors are developing new ICAP applications on an ongoing basis.

---

**Determining how to distribute ICAP services:** After you determine the ICAP services that you need, you need to determine how to distribute them. The size and scale of the computers used as ICAP servers and the specific services you select will influence your options for distributing ICAP services. Consult your ICAP vendors for recommendations.

The following table shows methods of distributing ICAP services.

You can run...	Comments
A single service on one ICAP server	A common recommendation is to deploy ICAP services in this manner.
Multiple services on one ICAP server	This type of distribution of ICAP services is likely to be uncommon. An ICAP server must have the capacity to support multiple ICAP applications. Overloading an ICAP server will affect performance of ICAP services.  Compatibility issues might exist between ICAP services from different vendors.

You can run...	Comments
<p>The same service on multiple ICAP servers</p>	<p>You might want to run the same service on more than one ICAP server. For example, you might want to run more than one copy or more than one version of a specific virus scanning application to handle the volume of content you expect origin servers to return in response to requests.</p> <p>Based on information you provide in your NetCache ICAP configuration, NetCache then balances the traffic load over all copies and versions of the same application. You specify how load balancing is to occur by service—round robin, IP address, or least usage.</p>

**Multiple ICAP services adapting the same content:** Multiple ICAP services can adapt the same content. For example, you might want the same content to be filtered and then checked for viruses. When you configure your NetCache appliances for ICAP service, you specify the order in which the ICAP services that are invoked at the same vectoring point will be applied.

---

**Note**  
Ordering applies only when the services are installed on different ICAP servers.

---

**Considering network traffic when determining ICAP services:** Ensure that your network can accommodate the number of ICAP services that you want to provide.

Adding ICAP services to your network might increase network traffic. The reason is that the NetCache appliance must pass a request it receives or content it fetched from the origin server to the ICAP server before it processes the request or content. For example, if a single ICAP service processed every byte of every page, it might create an ICAP traffic load that is twice as large as your HTTP origin server fetching load. However, if your NetCache appliance has a 50 percent hit rate, ICAP would be invoked only 50 percent of the time.

**Considerations for ICAP servers**

Consult your ICAP vendors for recommendations about the number of services that can run on a particular vendor's ICAP server and how many ICAP servers are required for a particular ICAP application.

**Network link recommendation**

The link to the ICAP services should have as much bandwidth capacity as the link to the clients and the upstream network.

**Location of the NetCache appliances and ICAP servers**

For the highest level of security and performance, locate the ICAP server and NetCache appliance on a switched network or private network segment. For best performance, ensure that the ICAP server remains well connected to the client; that is, do not locate the ICAP server at another site, which could cause high latency. Also, ensure that a slow link does not exist between the ICAP server and the client.

If a NetCache appliance and the ICAP servers it works with are on opposite sides of a firewall, work with your firewall administrators to ensure that ports on the firewalls are configured to allow the NetCache appliance to send traffic to and receive traffic from the ICAP servers.

**Failover if an ICAP server goes down**

If an ICAP server becomes unavailable, the NetCache default behavior is to automatically bypass ICAP services if they become unavailable. Optionally, you can configure your NetCache appliance so that it does not bypass an ICAP service but, instead, returns an error message to the client.

ICAP supports multiple ICAP servers. Therefore, you can install the same ICAP service on more than one ICAP server. Then, if one ICAP server running the service is unavailable, requests are balanced over the remaining ICAP servers running the service.

**Note**

---

Carefully consider the effects of suspending or bypassing specific services. For example, if you have content filtering software set up to block access to stock quotes on the Internet for all employees but employees in the finance department, is it acceptable to you that all employees could access stock quotes if the ICAP server becomes unavailable? If you are using ICAP virus scanning and the ICAP server becomes unavailable, would you want to allow documents into your network without checking them for viruses?

---

## Security and ICAP

---

### Security risks with ICAP

As with any client-server interaction, the link between the NetCache appliance and the ICAP server must be guarded for both privacy and security. Ensure that the network between the NetCache appliance and the ICAP server is at least as secure as the network between the user's browser and the NetCache appliance. The reason is that the current release of ICAP can expose some private data that NetCache security features strive to protect.

### Recommendations for security and privacy protection

Network Appliance recommends that you employ the following security measures for ICAP server deployment with a NetCache appliance:

- ◆ Deploy the ICAP server and the NetCache appliance on a private network, ideally a nonbroadcast network and possibly a switched hub network.  
Deploying the ICAP server and the NetCache appliance on the same network has the added advantage of improving performance.
- ◆ Do not share the NetCache password.
- ◆ Take security precautions appropriate to the operating system and applications running on the ICAP server computer.
- ◆ Locate the ICAP server and the NetCache appliance in a secure computer room.

One concern is that a small change to the ICAP server could redirect HTTP traffic to a remote hostile server. For this reason, the ICAP server should receive the same level of security protection as the NetCache appliance.

### ICAP services are not invoked with secure sockets transactions

NetCache cannot monitor transactions made through the secure sockets layer (SSL). SSL is designed to prevent a “man in the middle” attack on an HTTP transaction. Because RSA public-key cryptography is used to encode secure sockets transactions, only the origin server can decode the contents of a secure sockets transaction.

NetCache uses the following methods for handling secure traffic:

- ◆ Tunnel  
NetCache never sends tunneled traffic to ICAP servers. Instead, NetCache sends tunneled traffic directly to origin servers and receives responses directly from them. (NetCache cannot decipher tunneled traffic.)

- ◆ Terminate (NetCache communicates with the browser using the SSL handshake, thereby functioning as if the appliance is the origin server).  
NetCache sends HTTPS traffic to ICAP servers in the same way it sends any other HTTP traffic. However, HTTPS traffic is not encrypted between the appliance and the ICAP server. (Current ICAP servers cannot be configured for secure connections.)

If you want virus checking over HTTP to be secure, do not allow downloading of any executable files through SSL. The reason is that a virus might pass through the NetCache appliance when ICAP services are not invoked. The virus could infect a local computer system.

**Note**

---

The only way to enforce a policy whereby executable files are not downloaded through secure sockets is to block the SSL port and disable CONNECT requests.

---

For more information about SSL, go to the following URL:  
<http://developer.netscape.com/tech/security/ssl/howitworks.html>.

## Section C: ICAP service scenario

---

**About this section** This section provides scenarios that show how some organizations might use ICAP services.

**Contents of this section** This section contains the following topics:

- ◆ [“Scenario: virus checking in an enterprise”](#) on page 222

## Scenario: virus checking in an enterprise

---

**About this scenario** This scenario describes how an ICAP server running virus checking software checks an executable file that an origin server is returning in response to a client request.

### Organization's requirements

LMN, Incorporated, an enterprise, has the following goals:

- ◆ Reduce bandwidth consumption by caching HTTP objects and objects requested through HTTP

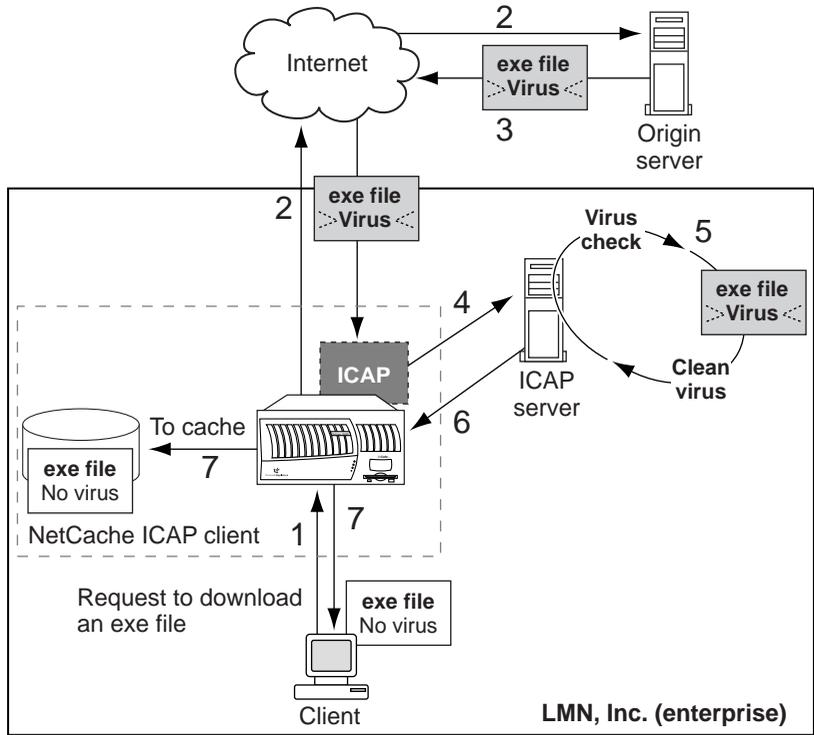
LMN wants to cache downloadable files, such as PDF files, to reduce the amount of bandwidth that results when multiple clients request the same downloadable files.

- ◆ Add a virus checking service to prevent viruses from being downloaded to the network

LMN cannot trust each origin server to provide virus-free data. The reason is that origin servers might have been compromised and virus-infected files might be installed.

### Deployment illustrated

The deployment for LMN is shown in the following illustration.



As the previous illustration shows, when a user sends a request to download an executable file, the NetCache ICAP client passes the file it receives from the origin server to the ICAP server for virus checking before caching the file.



## About this chapter

This chapter contains a number of possible strategies for deploying a NetCache appliance with your firewalls. A *firewall* separates networks and enforces security policies about communication between networks. When you deploy NetCache appliances in your network, you need to make decisions about where to locate NetCache appliances in relation to the firewall. Depending on the location you choose and the type of firewall you are using, you might need to configure the NetCache appliance to support your deployment.

If you are deploying a streaming media cache, also read “[Considerations for firewalls and streaming media service](#)” on page 137.

---

### Note

Keep in mind that a NetCache appliance is not a substitute for a firewall or an electronic commerce server.

---

## Chapter contents

This chapter contains the following sections:

- ◆ “[Deploying NetCache parallel to a firewall](#)” on page 226
- ◆ “[Deploying NetCache inside a firewall](#)” on page 227
- ◆ “[Deploying NetCache inside multiple firewalls](#)” on page 230
- ◆ “[Relationship between the firewall and NetCache authentication](#)” on page 232
- ◆ “[Scenario: access to a company Web server outside a firewall](#)” on page 234

# Deploying NetCache parallel to a firewall

## About deploying a NetCache appliance parallel to a firewall

Deploying a NetCache appliance parallel to a firewall provides a way to off-load traffic from the firewall. Off-loading traffic from a firewall is an advantage in cases in which the firewall is slow.

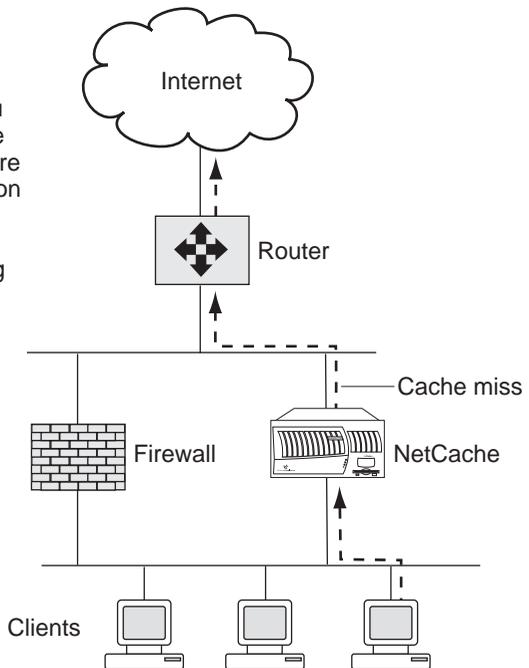
## What about security?

The security risk with a NetCache appliance parallel to the firewall is negligible. The NetCache appliance, for example, does not contain any other software on it (for example, Telnet or RSH) that would enable an intruder to access the network. Additionally, because NetCache does not expose objects to users, intruders cannot change data.

## Example: NetCache appliance parallel to a firewall

The following illustration shows a NetCache appliance deployed parallel to a firewall.

When a NetCache appliance is deployed parallel to a firewall, you do not need to configure the appliance to be aware of the firewall. The reason is that NetCache uses its default route to route the traffic out, bypassing the firewall completely.



## Deploying NetCache inside a firewall

---

### About deploying a NetCache appliance inside a firewall

Some organizations have a security policy that does not allow other devices to be added as part of the security interface to their company. You can easily add a NetCache appliance to the network without disturbing the existing security scheme. In this type of deployment, NetCache is just an internal service.

What you need to do to deploy a NetCache appliance inside a firewall depends on the type of firewall you have and how you want to use it.

### NetCache appliance deployed inside a transparent firewall

If you are deploying a NetCache appliance inside a *transparent* firewall, you use the firewall as a default gateway for NetCache, as the following illustration shows. You do not have to perform any additional configuration in NetCache to send requests through the firewall to the Internet. However, to avoid a traffic loop, you must configure the firewall to recognize the NetCache IP addresses and to *not* apply the diversion rules.

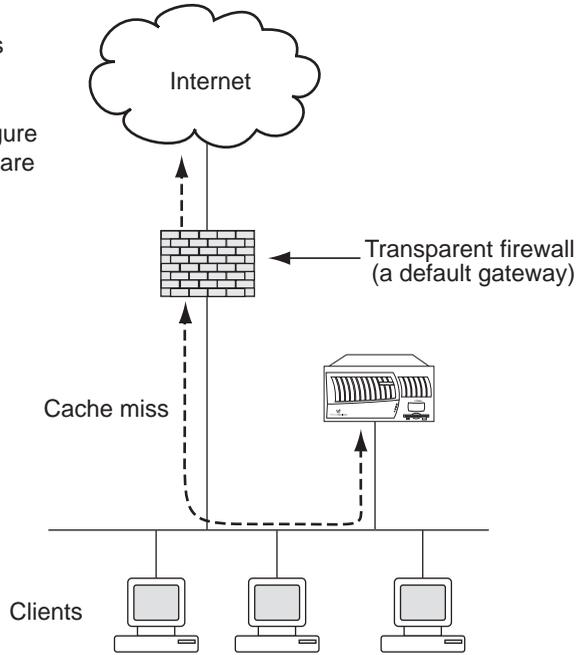
---

#### Note

A transparent firewall is a firewall that NetCache can connect through without having to know the IP address of the firewall.

---

When the firewall is used as a default gateway, it is not necessary to configure NetCache to be aware of the firewall.



### NetCache appliance deployed inside a nontransparent firewall

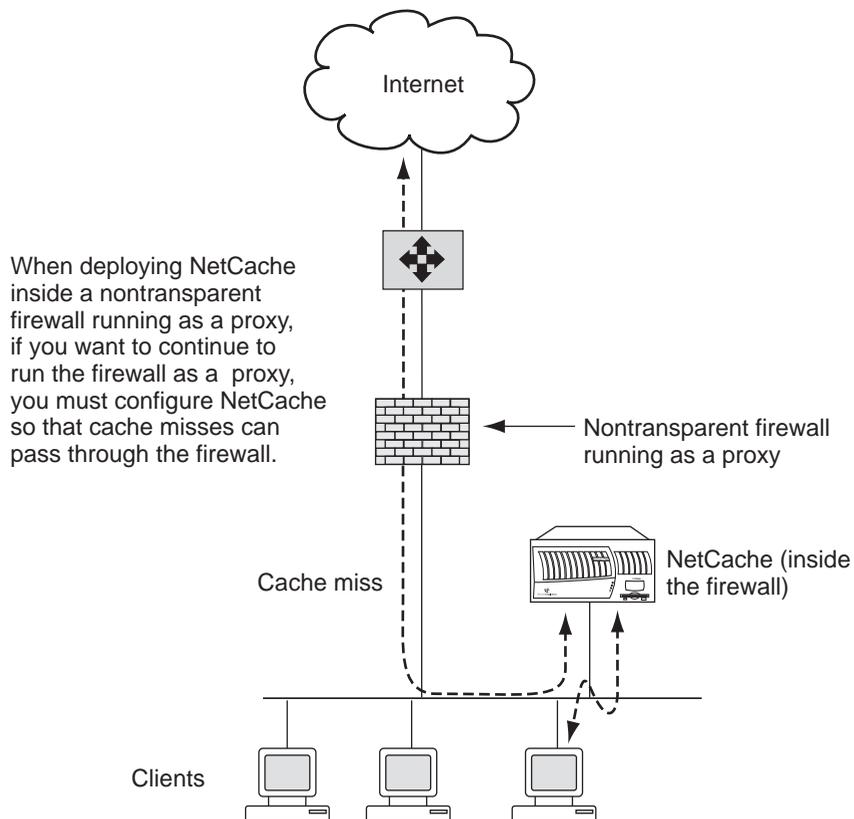
The following example shows a NetCache appliance deployed inside a nontransparent firewall running as a proxy.

---

#### Note

A nontransparent firewall is a firewall that NetCache must be aware of in order to connect through it. That is, NetCache must send requests directly to the correct firewall IP address and port.

---



In the NetCache configuration, you set up a logical hierarchy. This hierarchy enables NetCache to send cache misses to the nontransparent firewall for routing to the Internet. See Section A, “[Request resolution hierarchies](#),” on page 70 for more information about hierarchies.

---

**Note**

Defining logical clusters is not possible for NNTP.

---

**If origin servers are inside your firewall**

To speed up performance, you want NetCache to fetch URLs directly from any origin servers inside the firewall instead of trying to fetch the URLs outside the firewall. See information in the *Administration Guide* about hierarchies for details.

## Deploying NetCache inside multiple firewalls

---

### Goals when deploying NetCache inside multiple firewalls

When you deploy one or more NetCache appliances inside multiple firewalls, you want the following to occur:

- ◆ Distribution of traffic over the firewalls
- ◆ Failover of traffic to another firewall if one firewall goes down

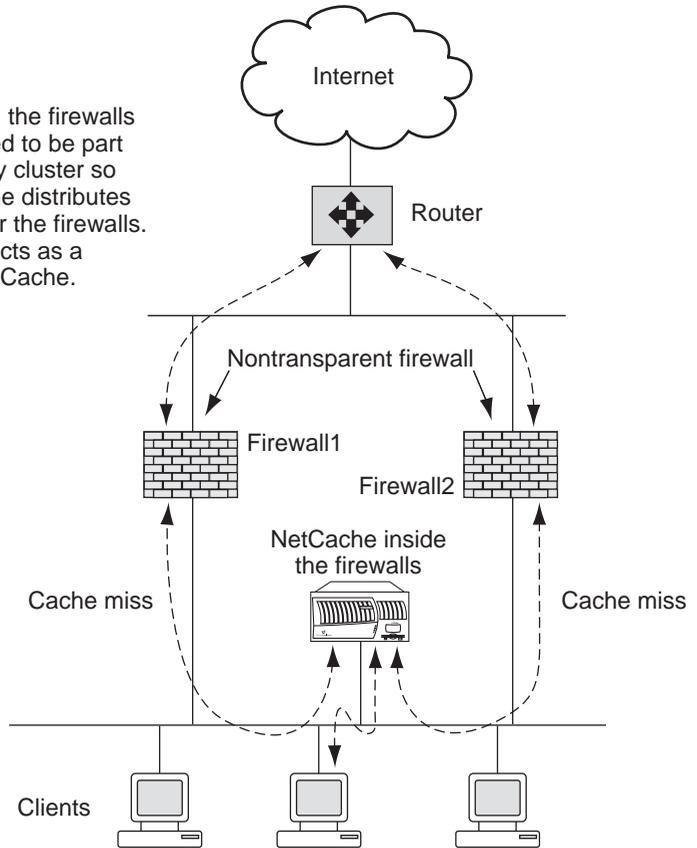
### Difference between transparent and nontransparent firewalls

If the firewall is transparent, routing automatically sends requests to the Internet. You do not need to configure NetCache for this to occur. However, if the firewall is nontransparent and running as a proxy, you must configure NetCache to distribute requests over the firewalls and have traffic fail over to the other firewall if one firewall goes down.

### Example: NetCache inside parallel nontransparent firewalls

The following illustration shows a deployment with a NetCache appliance inside two parallel firewalls.

In NetCache, the firewalls are configured to be part of a hierarchy cluster so that NetCache distributes requests over the firewalls. The cluster acts as a parent to NetCache.



By defining (in the NetCache configuration) that the firewalls are part of a cluster in a logical hierarchy, you enable NetCache to distribute requests over the two firewalls. Otherwise, NetCache might use one firewall continually until it that firewall goes down and, only then, use the other firewall. See Section A, [“Request resolution hierarchies,”](#) on page 70 for an explanation of clusters.

**Are there differences if there are multiple NetCache appliances?**

No. A request is sent to one of the NetCache appliances, as determined by how the client traffic request distribution is set up. That NetCache appliance then interacts with the firewalls as described in this section. To make this deployment work, each NetCache appliance is configured with the same logical hierarchy configuration.

# Relationship between the firewall and NetCache authentication

---

## About this section

Information in this section applies only to Web caching.

### Note

---

If a news server requires authentication, the news server handles the authentication. If a streaming server requires client authentication, NetCache establishes a connection to the streaming server and forwards the authentication information given by the client.

---

## If a NetCache appliance is parallel to the firewall

If a NetCache appliance is parallel to a firewall, Web requests bypass the firewall completely. Therefore, if you want protocol authentication for your users, you must enable authentication in NetCache and specify the authorized protocols for individual users.

## If a NetCache appliance is inside the firewall

If a NetCache appliance is inside the firewall, you can set up authentication in one of the following ways:

- ◆ The NetCache appliance authenticates the client.
- ◆ The firewall authenticates the client. If you do not want to disturb the security of your firewall, continue to have the firewall authenticate clients.

**If you want the firewall to authenticate users:** You do not need to configure protocol authentication in NetCache for individual users. In NetCache, you add user names and passwords only for the system administrators who need to use the Appliance Manager. You do not enable authentication in NetCache.

**If you want NetCache to authenticate users:** You enable protocol authentication in NetCache and set up user authentication in one of the following ways:

- ◆ Use the NetCache user database, in which case you need to configure protocol authentication for individual users.
- ◆ Use an LDAP server for user authentication.
- ◆ Use a RADIUS server for user authentication.
- ◆ Use the NTLM authentication protocol for user authentication.
- ◆ Use the Kerberos authentication protocol for user authentication.

If NetCache performs the authentication, the firewall is authenticating non-Web traffic only.

### **Requirement for access to an LDAP or RADIUS server**

If you want NetCache to use an LDAP server or RADIUS server for authenticating requests, you need to do the following:

- ◆ If the LDAP server or RADIUS server is outside the firewall
  - ❖ Configure NetCache to point to the LDAP server or RADIUS server.
  - ❖ Ensure that the firewall is set up so that NetCache can reach the LDAP server or RADIUS server through the firewall.
- ◆ If the LDAP server or RADIUS server is inside the firewall
  - ❖ Configure NetCache to point to the LDAP server or RADIUS server.

### **Requirements when using the NTLM or Kerberos authentication protocol**

If you are using the NTLM or Kerberos authentication protocol, you need to do the following:

- ◆ If the Microsoft Windows authentication environment is outside the firewall
  - ❖ Configure NTLM or Kerberos options in NetCache to enable NetCache to communicate with the Windows authentication environment.
  - ❖ Ensure that the firewall is set up so that NetCache can reach the Windows authentication environment through the firewall.
- ◆ If the Microsoft Windows authentication environment is inside the firewall
  - ❖ Configure NTLM or Kerberos options in NetCache to enable NetCache to communicate with the Windows authentication environment.

## Scenario: access to a company Web server outside a firewall

---

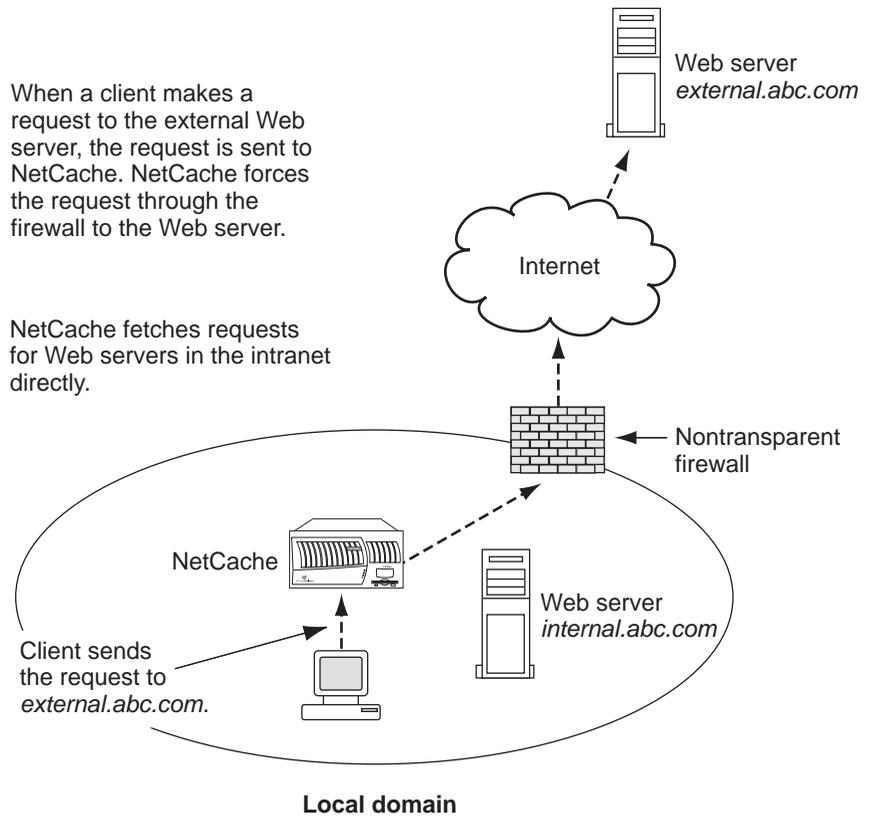
**About this scenario** This scenario describes deployment when an organization's Web server is outside a firewall and users in the organization need to access it.

**Difference for transparent and nontransparent firewalls**

If the firewall is transparent, routing ensures that requests are sent to the company's Web server when it is outside the firewall. Requests to the external Web server are handled at the IP level. Therefore, you do not need to configure NetCache to ensure that requests can reach the external Web server.

However, when the firewall is nontransparent and running as a proxy, you must configure NetCache to ensure that requests can reach the Web server. The reason is that when NetCache processes the domain name in the URL, it expects the Web server to be in the local domain.

**Example: forcing Web server access** In the following illustration, the organization's Web server, *external.abc.com*, is outside the firewall. It is accessible through an ISP, such as UUNET.





**About this chapter** This chapter provides examples of some of the more complex NetCache routing deployments that Network Appliance has received questions about. If you are using a single NetCache appliance, or if you have a simple network, these examples might not be of interest to you.

---

**Note**

TCP/IP routing is a complex topic and beyond the scope of this chapter.

---

**Chapter contents** This chapter contains the following sections:

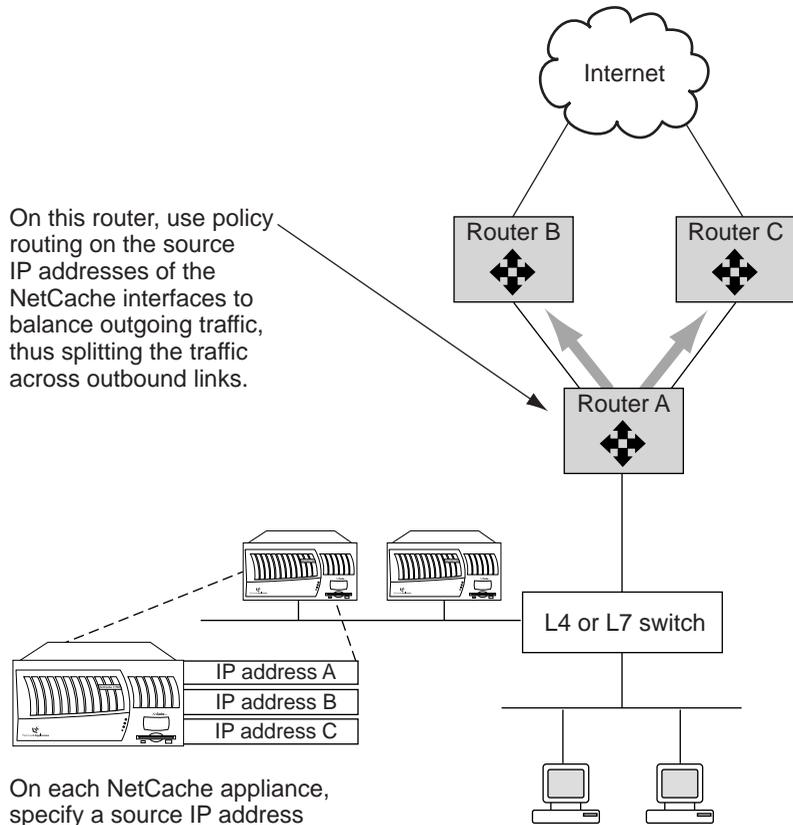
- ◆ “[Distributing outgoing traffic over multiple links](#)” on page 238
- ◆ “[Distributing incoming traffic over multiple links](#)” on page 240

**Routing summary** During NetCache appliance setup, you identify the default router. Subsequently, you typically do not need to add routes to the routing table because NetCache learns explicit routes through ICMP redirect messages that it receives from the default router. A NetCache appliance relies on the default route and explicit routes for routing its own packets. It does not route other packets. Details about how routing works on the NetCache appliance is discussed in the *Administration Guide*.

# Distributing outgoing traffic over multiple links

## Example: splitting outgoing traffic over multiple links

The following example illustrates NetCache configuration and the use of policy routing to split HTTP traffic over outgoing links.



### Explanation:

- ◆ You need to configure each NetCache appliance so that the IP addresses of its network cards show different source addresses. You can then set up your policy routing so that the traffic from NetCache can be split over multiple links.

- ◆ In this example, an L4 or L7 switch is deployed on the network to balance client traffic over the NetCache appliances. However, if the NetCache appliances were directly connected to the router in your deployment, you would set up source IP addresses on the NetCache appliances and policy routing on the router the same way as in this example.
- ◆ This example shows multiple routers to make it easier for you to see that there are multiple links. This is not meant to imply, however, that you must have multiple routers to be able to split outgoing traffic over links.

## Distributing incoming traffic over multiple links

---

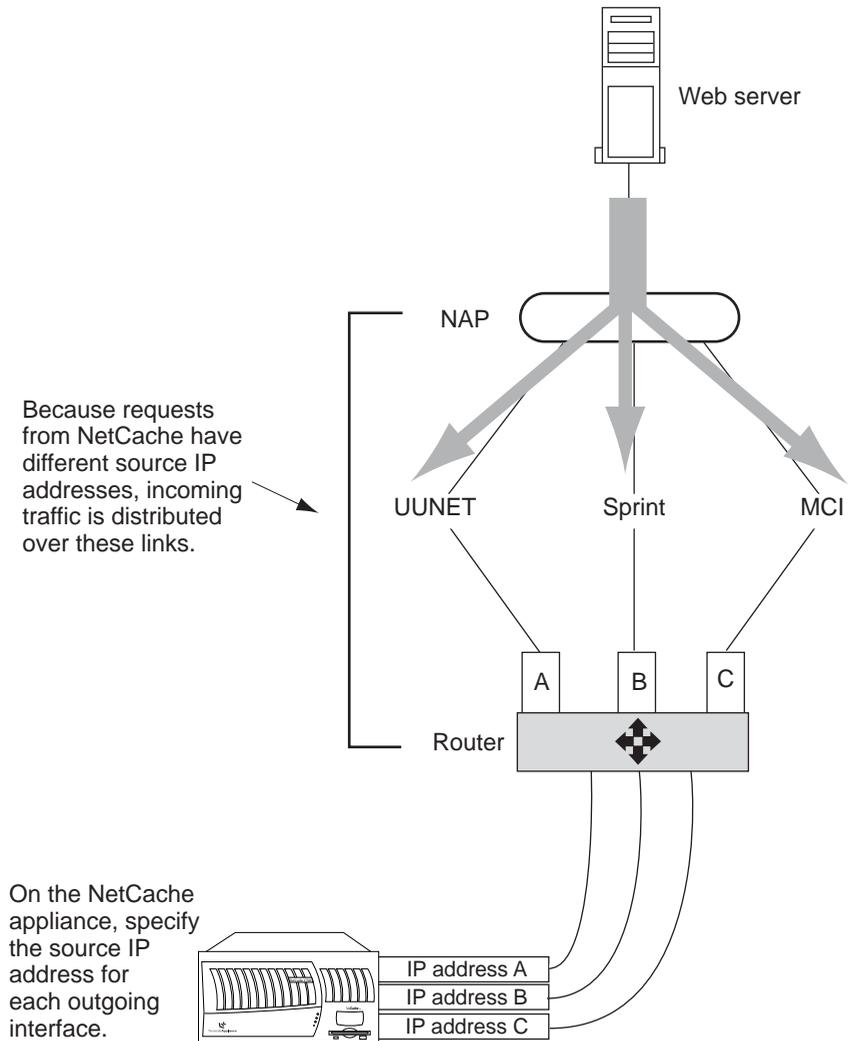
### **What the example illustrates**

The goal for the situation described in this example is to influence the route that a *reply* to a request takes, rather than to have the world's routing tables determine the "best" route. This goal is desirable because replies to requests can be quite large. Distributing replies across various routes helps to avoid congesting routes or having all replies slowed down by routes that are already congested.

### **Example: incoming traffic is distributed over WAN links**

In this example, as in the splitting of outgoing traffic over multiple links, you configure NetCache so that the IP addresses of its network cards show different source addresses. This configuration is beneficial if you want incoming traffic distributed over multiple links. The reason is that NetCache generates different source addresses, Web servers can respond to different source addresses, and replies can then take different routes through the Internet. In this example, the illustration shows the replies splitting at the Network Access Point (NAP) over three routes.

When replies can take different routes through the Internet, a greater chance exists that all the inbound links reach good utilization.





**About this chapter** This chapter provides information about the NetCache Global Request Manager (GRM), which is the Network Appliance request routing solution for Content Delivery Networks (CDNs) for enterprises and telcos. GRM can direct content requests from clients in a CDN to the NetCache appliances that are closest to the clients.

See the *Guide to Global Request Manager* for additional details about GRM.

**Chapter contents** This chapter contains the following sections:

- ◆ “[About Global Request Manager](#)” on page 244
- ◆ “[Request redirection with GRM](#)” on page 248
- ◆ “[Scenario: telco movie delivery to home subscribers](#)” on page 253
- ◆ “[Scenario: enterprise CDN spanning continents](#)” on page 256
- ◆ “[Additional GRM features](#)” on page 258

# About Global Request Manager

---

## Overview

As the Internet is maturing, content providers, publishers, and e-business operators are looking to the Internet to be a high-performance and reliable delivery mechanism for bandwidth-intensive, rich multimedia content. Service providers and enterprises are building CDNs in order to deliver high-quality content to users quickly and efficiently.

CDNs require a number of components specifically geared toward providing premium service to users, including routing, caching, content management, and reporting services. Global Request Manager (GRM), a feature of a NetCache appliance, can direct content requests to the NetCache appliances that are the closest to the clients; no L4 switches or WCCP routers are required. NetCache appliances in a GRM deployment can also provide caching of content.

NetCache works seamlessly with other products in the Network Appliance content delivery suite that enable telcos and service providers to build robust CDNs. See [“Integration with other content delivery products”](#) on page 258 for a description of how the WebWasher Corporation ContentReporter application can be used to create reports of cache activity.

## Advantages of directing requests to caches close to clients

A request routing scheme is necessary so that clients can find the content. In traditional caching deployments, L4 switches or WCCP routers are deployed on the network, or client browsers are manually configured to direct requests to NetCache appliances. These solutions are not ideal for CDNs, however, for the following reasons:

- ◆ CDNs want to provide premium service.  
In order to provide premium service, CDNs want to serve content as close to clients as possible. The reason is that the closer the device that serves the content is to the client, the faster the delivery. Additionally, the quality of the content served is likely to be higher. The content is less vulnerable to network problems because it travels a shorter distance.
- ◆ CDNs typically do not want to reconfigure customer networks to deploy CDN service.  
GRM provides an alternative to using L4 switches, WCCP routers, and manual configuration of client browsers to direct content requests to NetCache appliances—an alternative that is especially suited to the needs of CDNs. Often, NetCache appliances configured as GRM agents can be

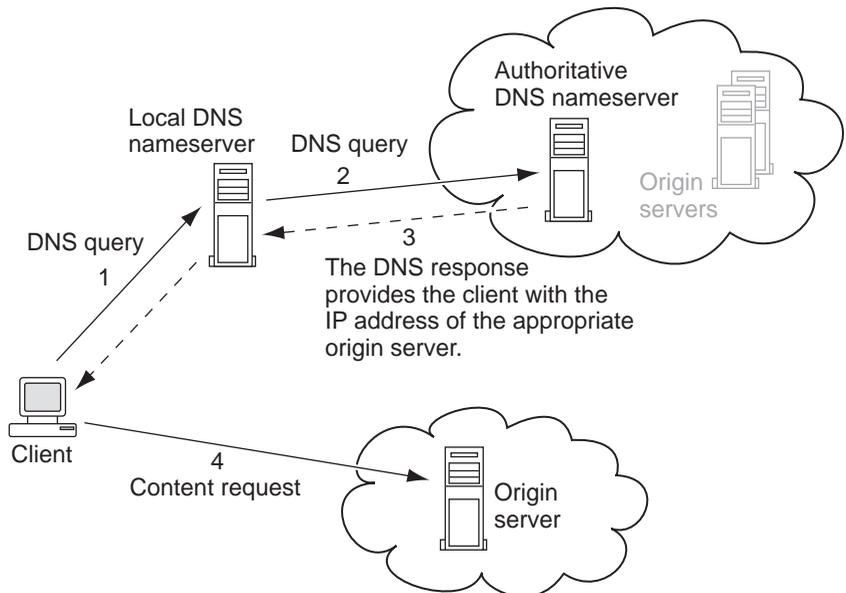
“dropped” into the customer environment; reconfiguration of the customer’s network environment is not necessary to support GRM. Reconfiguration might be necessary if your GRM deployment is not typical, for example, if you configure GRM agent caches to fetch content from random origin servers on the Internet as well as from the CDN data centers (not the recommended approach).

### Importance of serving content close to users

With the growing demand for streaming media in particular, serving content stored close to users is especially important because network congestion significantly degrades the quality of streaming media. Additionally, streaming media files, which are very large, consume a great amount of network bandwidth. NetCache appliances deployed close to users, working on behalf of the CDN content servers, can store and serve CDN content to clients. A significant amount of bandwidth is saved because the cache can fulfill multiple requests for the same content instead of the content servers at the other end of the WAN.

### Standard DNS request resolution

To understand how GRM works, it is useful to understand how standard DNS resolution works. The following illustration shows how standard DNS resolution works.



The purpose of DNS is to provide the mapping between host names, which humans find convenient, and Internet addresses, which are required for routing to devices on computer networks. The previous illustration shows the path of a request issued by the client, which is described in the following table.

<b>Stage</b>	<b>Description</b>
<b>1</b>	When a user sends a request for content, the client queries the local DNS nameserver to try to determine the Internet location of the origin server on which the requested content is stored.
<b>2</b>	DNS is designed to make host information available throughout the Internet, distributed among many sites and organizations, not just in a single central site. Therefore, the local DNS nameserver must pass the DNS query up the DNS nameserver hierarchy until the query reaches a DNS nameserver that is authoritative for (that is, has knowledge of) the DNS domain to which the origin server belongs.
<b>3</b>	The authoritative DNS nameserver returns to the client the IP address associated with the host name in the URL.
<b>4</b>	The client sends the content request (the application-level request) to the IP address of the origin server.

Standard DNS has a number of limitations. The following table contrasts the limitations of standard DNS to GRM.

<b>Standard DNS</b>	<b>GRM</b>
Standard DNS nameservers have no knowledge of proximity.	The GRM returns IP addresses of caches close to clients.
Standard DNS nameservers are unaware of the availability of the Web servers whose IP addresses they return.	The GRM returns only the IP addresses of caches that are available.

### **Is GRM the right solution for you?**

To determine whether GRM is the right solution for you, ask yourself the questions in the following table.

Questions to ask yourself	Considerations
Do you need multiple delivery points for content?	<p>If, for example, all your content is on one Web server and clients from all over the world access that content, delivery is likely to be slow.</p> <p>Using a global request routing solution to serve content from multiple delivery points would result in faster access to content. Additionally, moving content close to users results in fast delivery of high-quality content.</p> <p>GRM is a solution for providing content from multiple locations at optimal speed. To that end, the goal of GRM is to direct clients to the closest delivery point.</p>
Should you use transparency or a global request routing solution?	<p>If you deliver content from multiple locations, you need to determine how content can be served to clients from the delivery point closest to the clients.</p> <p>To use transparency, you must have control of the networks on which clients are located so that you can deploy the L4 switches or WCCP routers needed to redirect requests to the NetCache appliances. In contrast, a global request routing solution does not require control over the networks on which the clients are located.</p>
If a global request routing solution is appropriate, do you want to deploy one product or multiple products to provide request routing and caching services?	A NetCache appliance can provide caching services for multiple protocols plus the GRM feature—all on the same machine. The result is simpler configuration and management.

## Request redirection with GRM

---

### GRM components

The keys to achieving fast delivery of high-quality content are

- ◆ Moving content stored in central locations close to end users
- ◆ Directing a client to the delivery point closest to that client

NetCache appliances configured as *GRM servers* provide redirection services, redirecting client requests to NetCache appliances configured as *GRM agent caches*. GRM agent caches, which are located close to clients, cache content and resolve content requests on behalf of clients.

Typically, GRM agent caches are configured to handle only CDN traffic so that they can provide premium service to CDN subscribers. A GRM agent can be used to access content on the Internet, as well as to handle CDN content. However, Network Appliance recommends that you use your GRM agent caches to handle only CDN content delivery, which provides your customers with the most preferential service possible.

**Center caches and edge caches:** GRM agent caches in the data center (in a *center group*) are referred to as *center caches*. GRM agent caches in defined groups outside of the data center (*edge groups*) are referred to as *edge caches*. When a client in an edge group makes a request, the GRM server returns only the IP addresses of edge caches in that edge group. A center cache can provide failover for edge caches. A center cache can also provide content for clients that subscribe to CDN services but are not associated with an edge group.

### GRM server redirection methods

GRM supports two different redirection methods for GRM servers:

- ◆ DNS-based redirection services (GRM DNS server)
- ◆ L7 redirection services (GRM L7 server)

When you configure a NetCache appliance as a GRM server, you configure general GRM server options plus specific options for the type of redirection that you need. Depending on your redirection requirements, you might deploy all GRM DNS servers, all GRM L7 servers, or a combination of the two types of servers (a mixed hierarchy). Typically, multiple GRM servers are deployed, for example, for redundancy.

No limit exists for the number of GRM agent caches that a GRM server can support.

## About a GRM DNS server

A GRM DNS server provides standard DNS redirection plus a layer of intelligence that is not available with standard DNS— including support for proximity measurements, detection of unavailable GRM agents, and filtering. As with standard DNS, the client’s local DNS nameserver communicates with the GRM DNS server through a DNS request, and the GRM DNS server responds with a DNS response.

The GRM DNS server returns to the client a list of IP addresses of the available GRM agents that are *close to the local DNS nameserver that the client uses*. The client browser then selects a GRM agent cache from the list returned by the GRM DNS server. The cache that the browser selects might or might not be the closest cache to the client.

A GRM server can assume the identity of the existing authoritative DNS nameserver for the CDN name space, that is, the GRM server can assume the IP address that was previously assigned to the authoritative DNS nameserver. The authoritative DNS nameserver is then reconfigured with a new IP address. The authoritative DNS nameserver becomes the back-end to the GRM server in this case. Alternatively, URL rewriting can be used to direct content to the CDN.

## About a GRM L7 server

Like the GRM DNS server, the GRM L7 server includes intelligence to support proximity measurements, detection of unavailable GRM agents, and filtering. However, the GRM L7 server can provide more precise redirection than the GRM DNS server. The reason is that the GRM server redirection is *based on the client’s IP address* (rather than on the local DNS nameserver that the client uses). The GRM L7 server operates at Layer 7, the application layer of the OSI model, to rewrite a client request with the IP address of the agent cache that it determines is closest to the requesting client.

When you deploy a GRM L7 server, you must include in your deployment a means for client requests to locate the GRM L7 server. See “[Providing a means for locating a GRM L7 server](#)” on page 250.

---

### Note

If a proxy.pac file is used to direct client requests to a GRM L7 server, the GRM L7 server might return more than one GRM agent cache close to the client. If other methods described in “[Providing a means for locating a GRM L7 server](#)” on page 250 are used, the GRM L7 server rewrites the client request with the IP address of one GRM agent cache.

---

The same NetCache appliance can be configured as a GRM server and a GRM agent. If you are deploying GRM DNS servers to provide the means of locating GRM L7 servers, configuring the GRM L7 server as a GRM agent also enables the GRM DNS servers to select the closest GRM L7 server to a client.

## General GRM deployment guidelines

The following table provides general guidelines for determining when you should include a GRM L7 server in your deployment.

If...	Then...
The local DNS nameserver and the clients are physically close.	You can use all GRM DNS servers in your deployment.
The local DNS nameserver and the clients it services are <i>not</i> physically close.	You will want to include one or more GRM L7 servers in your deployment.  You must provide a means for redirecting client requests to the GRM L7 servers. See <a href="#">“Providing a means for locating a GRM L7 server”</a> on page 250.

For example, some enterprise CDNs are spread over continents, with clients located on continent that is different from the continent on which their local DNS nameserver is located. If only GRM DNS servers are used in such cases, the GRM agent IP addresses that GRM DNS servers return would be on the same continent as the local DNS nameserver because the GRM DNS server sees only the IP address of the local DNS nameserver, not the IP address of the client.

If a GRM L7 server is deployed for redirection service to clients, the GRM L7 server will redirect requests based on client IP address. Therefore, the GRM agent IP address returned to the client will be truly close to the requesting client.

## Providing a means for locating a GRM L7 server

If you are using a GRM L7 server, you must provide some means for the client requests to reach the GRM L7 server that is closest to the client. You can choose the methods shown in the following table.

<b>Methods for directing requests to a GRM L7 server</b>	<b>Advantages and disadvantages</b>
Using a standard DNS nameserver	<p><b>Advantage</b></p> <ul style="list-style-type: none"> <li>◆ This is a simple solution.</li> </ul> <p><b>Disadvantages</b></p> <ul style="list-style-type: none"> <li>◆ There is no way to guarantee that the GRM L7 server to which the DNS nameserver redirects requests is the closest one to the client.</li> </ul> <p>For example, if one GRM L7 server is in Australia and another is in the United States, the DNS nameserver round-robin process might direct requests from clients in the United States to the GRM L7 server in Australia.</p> <ul style="list-style-type: none"> <li>◆ The GRM L7 server can become overloaded if all client requests are sent to the same GRM L7 server.</li> </ul>
Deploying a GRM DNS server	<p><b>Advantages</b></p> <ul style="list-style-type: none"> <li>◆ The GRM DNS server acts as an intelligent means of locating the closest GRM L7 server to the clients.</li> <li>◆ The GRM DNS server can also be used to monitor the health of the GRM L7 servers.</li> </ul> <p><b>Disadvantages</b></p> <ul style="list-style-type: none"> <li>◆ This solution is more complicated than using only a DNS nameserver.</li> <li>◆ This solution requires additional NetCache appliances.</li> </ul>
Using a locator solution in your network infrastructure	<p><b>Advantage</b></p> <ul style="list-style-type: none"> <li>◆ An existing locator device in the network infrastructure can be used. Some locator devices provide sophisticated features, for example, features for load balancing over GRM L7 servers.</li> </ul> <p><b>Disadvantages</b></p> <ul style="list-style-type: none"> <li>◆ You must rely on the existing network infrastructure.</li> <li>◆ You must have devices in your network infrastructure to support this function.</li> </ul>

<b>Methods for directing requests to a GRM L7 server</b>	<b>Advantages and disadvantages</b>
Using a proxy.pac file	<p><b>Advantage</b></p> <ul style="list-style-type: none"> <li>◆ This solution is inexpensive, scalable, and easy to implement.</li> </ul> <p><b>Disadvantages</b></p> <ul style="list-style-type: none"> <li>◆ Client browsers must be manually configured to point to the proxy.pac file on the GRM L7 server that is closest to the clients.</li> <li>◆ Load balancing must be programmed into the proxy.pac file (for example, hash on URL).</li> </ul>

## Licensing

Each GRM server and GRM agent cache requires a license. Contact your Network Appliance sales engineer or the Installed Base Group at [ibg@netapp.com](mailto:ibg@netapp.com) to purchase a license.

## Scenario: telco movie delivery to home subscribers

---

**About this scenario** In this scenario, a company named Webmovie.com wants to provide movies to home subscribers in the United States and Canada. Webmovie.com has contacted Edgeserve.com, a telco, to provide the bandwidth and CDN services that it needs for its movie subscription business.

In this scenario, assume that clients in each country use a local DNS nameserver that is close to the clients; that is, clients in the United States use a local DNS nameserver in the United States and clients in Canada use a local DNS nameserver in Canada. Therefore, a simple way of deploying GRM is to use just GRM DNS servers for redirection.

### Organization's requirements

Webmovie.com's requirements are as follows:

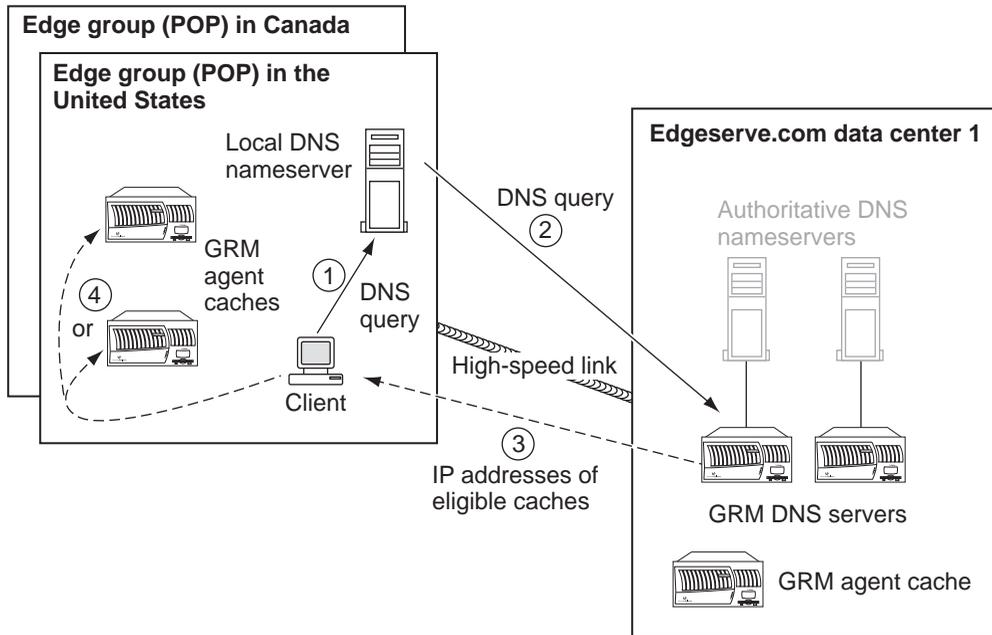
- ◆ Support both RTSP and MMS video streams
- ◆ Provide the highest quality video possible
- ◆ Support multiple bit-rate video

Webmovie.com customers will be connecting over a variety of types of connections, from slow-speed modems to DSL and cable modems.

- ◆ Provide fast and reliable service

### Deployment illustrated

The following illustration shows the deployment for Edgeserve.com and the path of a request from a client to a GRM agent cache that serves content to the client.



As the previous illustration shows, multiple GRM DNS servers are deployed in the data center so that content delivery service is not interrupted if one GRM DNS server becomes unavailable. Each GRM DNS server has been assigned the IP address that was previously assigned to an authoritative DNS nameserver in the data center.

Edgeservice.com deployed a GRM agent cache in the data center to provide failover in case all edge caches in an edge group become unavailable at the same time.

Edgeservice.com deployed edge caches configured for MMS and RTSP service in each of the two POPs (edge groups). The edge caches service only the clients in the edge group in which they are located. When a client in the POP makes a request, the GRM DNS server that receives the query from the local DNS nameserver returns the IP addresses of all eligible edge caches in the edge group with which the client is associated.

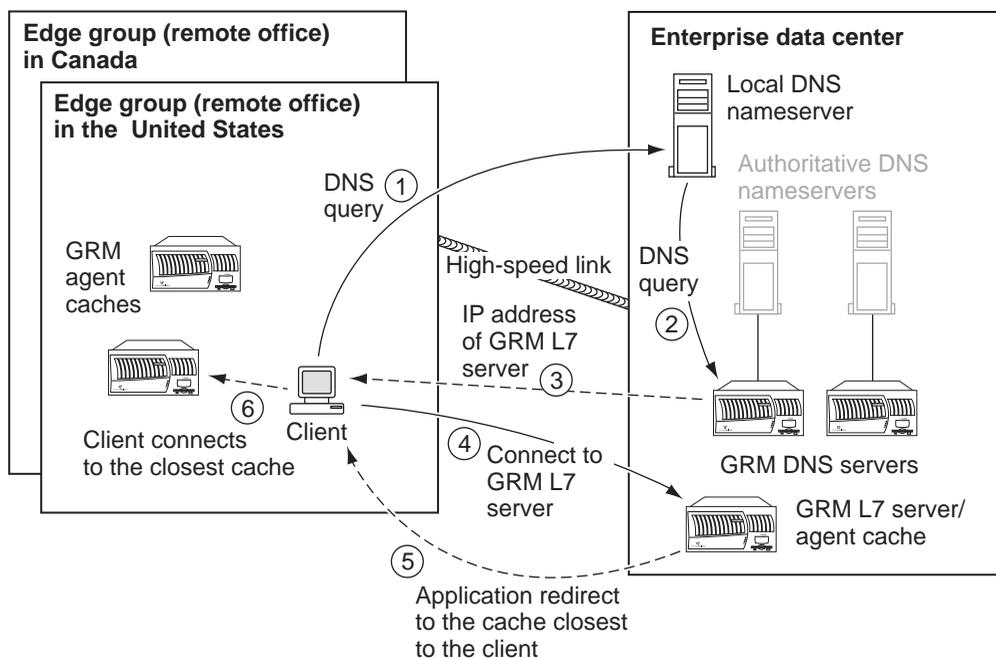
The center cache could be used to service content requests from the edge groups (instead of deploying edge caches in the POPs). However, this deployment was not chosen because Edgeservice.com requires high-quality video. If all clients are serviced from the center cache, the greater is the distance that content travels, the greater is the latency and the greater is the possibility of degradation of content

quality due to problems such as network congestion. Streaming media is especially vulnerable to network congestion, which can cause jagged video and unsynchronized audio and video. Additionally, serving content from edge caches rather than center caches significantly reduces bandwidth consumption over the WAN. The bulk of bandwidth savings results from eliminating the need for each client to individually fetch a media stream from the content server at the other end of the WAN.

## Scenario: enterprise CDN spanning continents

---

<b>About this scenario</b>	This scenario describes an enterprise that is distributed across continents. The clients in the remote offices in the United States and Canada use the local DNS nameserver in Europe to resolve their DNS queries.
<b>Organization's requirements</b>	<p>The enterprise's requirements are as follows:</p> <ul style="list-style-type: none"><li>◆ Use ELearning to train employees The managers want to create new content and deliver it to a large number of employees quickly. They want employees to view training videos from their desktops instead of traveling to attend training.</li><li>◆ Support both RTSP and MMS video streams</li><li>◆ Provide the highest quality video possible</li><li>◆ Provide fast and reliable service</li></ul>
<b>Deployment illustrated</b>	The following illustration shows the deployment for the enterprise and the path of a request from a client to the closest GRM agent cache to the client that can serve the requested content.



In this scenario, a GRM L7 server is deployed in the data center so that requests from clients in the remote offices can be served by the GRM agent caches that are closest to those clients. Because GRM L7 server redirection is based on the client IP address, the GRM L7 server can determine which GRM agent caches are truly close to the clients. In contrast, a GRM DNS server sees only the IP address of the DNS server, not the IP address of the client.

The goal with GRM L7 redirection is to force all client requests to the GRM L7 server so that the GRM L7 server can redirect the requests based on client IP address. In this scenario, GRM DNS servers are deployed to force client requests to the GRM L7 server. This deployment is referred to as a *mixed hierarchy*.

The GRM L7 server is also configured as a GRM agent cache, which enables the cache to send proximity data to the GRM DNS servers. (Only GRM agents can send proximity data to GRM servers.) The GRM DNS server determines the closest GRM L7 server to direct the client to based, on the IP address of the client's local DNS nameserver and the proximity information it receives from each GRM DNS agent.

## Additional GRM features

---

### **Features for optimizing request routing**

The GRM feature supports several failover and load balancing mechanisms. In addition, the GRM feature enables you to filter the IP addresses returned to clients based on protocols and location. See the *Guide to Global Request Manager* for details.

### **Integration with other content delivery products**

NetCache works seamlessly with the WebWasher Corporation ContentReporter products which enable telcos, service providers, and enterprises to build robust CDNs.

The WebWasher Corporation ContentReporter application gathers log files from each NetCache appliance and records the information in the ContentReporter database. You can use third-party applications to create reports that help you with billing and with planning content to push. Contact your Network Appliance sales representative or the WebWasher Corporation to obtain ContentReporter.

**About this chapter** This chapter provides illustrated examples of deployments in which NetCache is used as a cache and as a server accelerator in IPv6 networks.

**Chapter contents** This chapter includes the following topics:

- ◆ “[Proxy cache deployments in IPv6 networks](#)” on page 260
- ◆ “[NetCache as an accelerator in v4/v6 client and v4 server networks](#)” on page 261

**Before reading further** This chapter assumes that you have read information about proxy caching and server acceleration discussed previously in this guide. It also assumes that you have a working knowledge of how IPv6 networks work and how they are implemented. For more information about IPv6 networks, see the following sites:

- ◆ The IPv6 Information page at <http://www.ipv6.org>
- ◆ The 6bone IPv6 Testbed Network at <http://www.6bone.net>

You can obtain the IPv6-related specifications from the Internet Engineering Task Force Web site at <http://www.ietf.org/rfc>.

To obtain connectivity to the 6bone (IPv6) Internet, you must set up IPv6-over-IPv4 tunneling. IPv6-over-IPv4 tunneling enables the encapsulation of IPv6 packets within IPv4 headers to carry packets over IPv4 routing infrastructures. For information about IPv6-over-IPv4 tunneling, see to the following specifications:

- ◆ RFC 2893: Transition Mechanisms for IPv6 Hosts and Routers
- ◆ RFC 3056: Connection of IPv6 Domains via IPv4 Clouds

# Proxy cache deployments in IPv6 networks

---

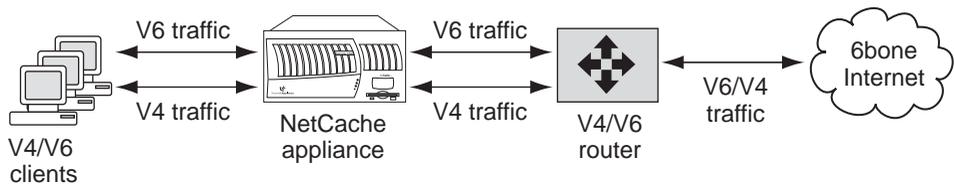
## About proxy caching in IPv6 networks

A NetCache appliance can be configured to proxy IPv4- and IPv6-enabled client HTTP and FTP-over-HTTP requests through a network that provides native IPv6 transport services.

The following example illustrates and describes how NetCache provides this type of connectivity.

## Deployment illustrated

In this example, to serve both its IPv4- and IPv6-enabled clients, an IPv6-based ISP has deployed a NetCache appliance that can support both IPv4 and IPv6 traffic to routers and to the IPv6 Internet.



# NetCache as an accelerator in v4/v6 client and v4 server networks

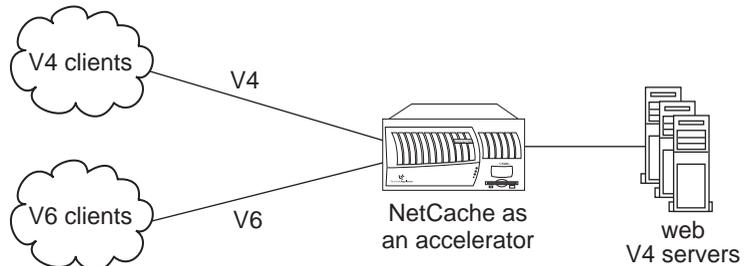
---

## About NetCache as an accelerator for IPv4-based legacy Web servers

As an accelerator, the NetCache appliance can be used to cache content from one or more IPv4-based Web servers and provide that content to either IPv4- or IPv6-enabled clients.

## Deployment illustrated

The following illustration shows an IPv4 and IPv6 client to IPv4 Web server deployment with the NetCache appliance configured as an accelerator.



After the appliance receives client requests, it fetches content from the Web servers that it does not have available in its cache. NetCache translates IPv6 client requests for the IPv4-based Web servers.



**About this appendix** Using an automatic proxy configuration file is discussed in this guide as a strategy you can use for the following:

- ◆ Providing user access to a NetCache appliance deployed as a Web cache
- ◆ Providing failover for Web caching
- ◆ Distributing requests over multiple NetCache appliance Web caches and third-party Web caches

Rather than repeating the description of automatic proxy configuration in each of these discussions, the basic information about this file is presented in this appendix. Discussions related to automatic proxy configuration in other parts of this guide refer to this appendix for the basic information.

---

**Note**

A proxy.pac file can be used in a GRM deployment to direct client requests to the closest GRM agent caches. That GRM agent cache can serve Web or streaming objects. See the *Guide to Global Request Manager* for more information.

---

**Appendix contents** This appendix contains the following sections:

- ◆ [“Introduction to automatic proxy configuration”](#) on page 264
- ◆ [“Examples of automatic proxy configuration files”](#) on page 267

# Introduction to automatic proxy configuration

---

## What is automatic proxy configuration?

You can use an automatic proxy configuration file to specify, in one place, the Web proxy configurations for your entire organization. Clients in your network must be configured to point to the file instead of to a specific Web cache.

The automatic proxy configuration file consists of a JavaScript function. You create the file by entering the JavaScript function in the file. Each time a client requests a URL, the JavaScript function is executed and it returns the IP address that should be used for retrieving the resource.

---

### Note

Typically, administrators create a proxy.pac file manually. Javascript debuggers are available to help debug the javascript used to create a proxy.pac file. You can, for example, type > 'javascript:' in the Netscape URL window to access a javascript debugging tool.

---

The ways you can customize the file include the following:

- ◆ Always use a Web cache.
- ◆ Use a Web cache for all requests to the Internet.
- ◆ Use different Web caches for different types of requests.
- ◆ Use different Web caches for URL ranges, using wildcard pattern matching.

## Advantages of automatic proxy configuration

Using an automatic proxy configuration file has the following advantages:

- ◆ It is inexpensive.
- ◆ It is scalable.
- ◆ It is relatively fail-safe.
- ◆ It is easy to implement.
- ◆ It works with all NetCache products and versions.
- ◆ You can include failover and intelligent distribution of requests over NetCache appliances and third-party Web caches.
- ◆ You can make any necessary changes to the automatic proxy configuration file without having to reconfigure client browsers to match the changes.
- ◆ Because the client browsers are not configured to point to the NetCache appliances, you can swap out NetCache appliances (for example, for maintenance) without affecting client requests.

## Disadvantages of automatic proxy configuration

Using an automatic proxy configuration file has the following disadvantages:

- ◆ Using an automatic proxy configuration file for failover and request distribution is equivalent to having an L4 or L7 switch, but the Web browser must detect failure. Not all browsers detect failure well. For example, a browser might not fail over at all or it might take several minutes.
- ◆ The system administrator must set up how request distribution occurs.
- ◆ Client browsers must be configured. If users do not configure their browsers correctly, you can receive many calls for help.
- ◆ If you had a Web cache previously and client browsers are already configured to point to it, these browsers must be reconfigured.
- ◆ Sophisticated users can bypass the NetCache appliance by changing the configuration in the browser, unless the firewall is configured to prevent direct access to the Internet.
- ◆ Failover does not occur if a Web cache is overloaded.

## Achieving finer load balancing than automatic proxy configuration provides

To achieve finer load balancing and to retain control over failover from the browser, you can combine the use of the automatic proxy configuration file with a Server Load Balancer (SLB). Combining the two lets you partition the workload, and failover is predictable because the SLB's method is predictable. For more information, refer to [“Using a Server Load Balancer for request distribution”](#) on page 65.

## When use of automatic proxy configuration is not recommended

If you are a dial-up ISP, do not use an automatic proxy configuration file because of startup problems with some browsers. With some versions of Netscape Navigator and Microsoft Internet Explorer, the browser times out before a PPP session is established. The user then receives an error message that might be confusing.

Network Appliance recommends that dial-up ISPs either use transparent proxying or start with an SLB and then move to the next generation of SLBs, which should help with partitioning.

## Browsers that support automatic proxy configuration

Automatic proxy configuration is supported by both Netscape Navigator and Microsoft Internet Explorer browsers.

## File location

You load the automatic proxy configuration file on a Web server or file server that is directly accessible to the clients, or on the NetCache appliance.

**Caution**

---

Before you configure client browsers, decide the name and location of the automatic proxy configuration file, then do not change them. Otherwise, client browsers must be reconfigured to match the changes.

---

# Examples of automatic proxy configuration files

---

## What this section contains

This section contains examples of automatic proxy configuration files that you can edit for use at your organization. The example for multiple Web caches includes comments to help you identify what to change and what to leave as is.

### Note

---

See the *Guide to Global Request Manager* for example proxy.pac files for use with GRM.

---

## Examples: single Web cache

For the next two examples, assume that the organization has only one NetCache appliance, which is functioning as a Web cache. This example shows a simple automatic proxy configuration file that is set up so that the Web cache is always used. If the Web cache goes down, failover to the Internet does not occur.

The PROXY statement, shown in the following example, has the IP address of the Web cache and the port. Change those to the IP address and port for your NetCache appliance.

```
function FindProxyForURL(url,host)
{
    return "PROXY 192.168.1.1:8080";
}
```

In the following example for this single NetCache appliance, the DIRECT statement is appended to the PROXY statement that is in the first example. The DIRECT statement sets up the file so that if the Web cache is unavailable, the client should attempt to connect to the Internet directly; that is, failover to the Internet occurs.

```
function FindProxyForURL(url,host)
{
    return "PROXY 192.168.1.1:8080; DIRECT";
}
```

## Example: multiple Web caches

The following example illustrates setting up the file for distributing Web requests over multiple servers and for failover.

```
function FindProxyForURL(url,host)
{
    var hash = 0;
    hash = (host.length % 4);
    if (url.substring(0,5) == "https")
        return "DIRECT"
    if (url.substring(0,5) == "snews")
        return "DIRECT"
    if (hash == 0)
        return "PROXY 192.168.1.1:8080;
                PROXY 192.168.1.2:8080;
                DIRECT"
    if (hash == 1)
        return "PROXY 192.168.1.2:8080;
                PROXY 192.168.1.3:8080;
                DIRECT"
    if (hash == 2)
        return "PROXY 192.168.1.3:8080;
                PROXY 192.168.1.4:8080;
                DIRECT"
    else
        return "PROXY 192.168.1.4:8080";
                "PROXY 192.168.1.1:8080";
                DIRECT;
}
```

← Leave this part as is.

Change the number after the modulo operator (%) to match the number of the Web caches on your network.

← Leave this part as is.

← In each of the "if" statements, replace the IP addresses and port numbers with the appropriate addresses and port numbers. The first PROXY statement is for the primary cache and the second is for a secondary cache, to which the primary cache can fail over.

If you want failover to the Internet when the secondary cache does not have the requested object, include the DIRECT statement, as shown.

The previous example is available on the NOW Web site with the NetCache product manuals.

**How requests are distributed:** The method used in this file to distribute requests over the Web caches is the string length of the URL's host name. The reason this method is used is that if the selection method was not restricted to the host name, performance would drop because subsequent URLs within a page might be sent to different caches. Each cache would then need to establish its own persistent TCP connection to the requested Web server, which reduces performance and adds latency.

**How request distribution is accomplished:** This file is set up so that requests are distributed over four NetCache appliances that are running as Web caches (192.168.1.1:8080, 192.168.1.2:8080, 192.168.1.3:8080, and 192.168.1.4:8080).

You configure distribution of requests over the Web caches as follows:

- ◆ Change the number in the `hash = (host.length % 4);` statement to match the number of Web caches over which the requests are to be distributed.
- ◆ As the example shows, include an “if” statement for all but one of the Web caches and an “else” statement for the last Web cache. In each “if” and “else” statement, include a PROXY statement with the IP address and port number for the Web cache. (In the example, this is the first PROXY statement.)

**How failover is accomplished:** This file includes parameters for failover, which you might or might not want to have in your automatic proxy configuration file.

- ◆ The second PROXY statement contains the IP address and port number for the cache to which you want the request to fail over if the primary cache does not have the object.
- ◆ The DIRECT statement indicates that if the secondary cache is unavailable, the browser is to resolve the URL directly from the source. If you do not want failover to the Internet, do not include this statement.

## Balancing the load of authenticated users across NetCache appliances

In this example, an enterprise company wants to authenticate users. The company wants to use a *proxy.pac* file to balance the load of requests across two NetCache appliances, but the company does not want users to have to authenticate more than once.

To eliminate the need for users to authenticate to each NetCache appliance, the *proxy.pac* file was set up so that last digit of the client IP address was used for hashing. Requests for clients for which the last digit of their IP address is odd are sent to one NetCache appliance and requests for clients for which the last digit of their IP address is even are sent to the other NetCache appliance.

The following example shows the *proxy.pac* file to support this scenario.

```

function FindProxyForURL(url,host)
{
if (url.substr(0,5) == "http:")
{
// take the last digit of the client's IP address and calculate
// its value modulo the number of available caches (2 in this example)
var hashValue, hash;
var myAddress = myIpAddress();
hashValue = parseInt(myAddress.charAt(myAddress.length - 1), 10);
hash = hashValue % 2;
if (hash == 0)
return "PROXY 10.70.20.18:3128; PROXY 10.70.20.19:3128; DIRECT";
else
return "PROXY 10.70.20.19:3128; PROXY 10.70.20.18:3128; DIRECT";
}
}
}

```

The **DIRECT** statement is included so that if neither cache is available, the browser can resolve the URL directly from the source. If you do not want failover to the Internet, do not include this statement. See [“Example: multiple Web caches”](#) on page 268 for an example of the use of the **DIRECT** statement.

**About this appendix** This appendix lists the requirements for implementing transparent proxying when you are using an L4 or L7 switch or a WCCP 2.0-based router. This appendix also provides key features to look for when purchasing switches.

**Requirements for transparent proxying with a switch** To deploy transparent proxying with an L4 or L7 switch, meet the requirements listed in the following table. Network Appliance also recommends that you comply with the “best practice” recommendations that are listed in the table. Items in the table are requirements unless otherwise noted.

Category	Requirements and recommendations
<b>NetCache appliance</b>	<ul style="list-style-type: none"><li>◆ A network interface is required for each switch attached.</li><li>◆ The NetCache appliance should be directly connected to the switch (best practice).</li></ul>
<b>Network layout</b>	<ul style="list-style-type: none"><li>◆ The switch must be in a location to detect all traffic for the clients it is expected to serve.</li></ul>
<b>L4 or L7 switch configuration</b>	Switch configuration involves the following high-level steps: <ul style="list-style-type: none"><li>◆ Identify traffic to be redirected to the NetCache appliances and the request distribution method to be used.</li><li>◆ Identify the NetCache appliances to which traffic is to be redirected.</li></ul> Contact your switch vendor for specific switch configuration settings.
<b>NetCache configuration</b>	For each protocol for which you want to set up transparent proxying, enable the transparency feature.  See the online Help for configuration details.

## Requirements for transparent proxying with a WCCP 2.0 router

To deploy transparent proxying with a WCCP router, meet the requirements listed in the following table.

Category	Requirements and recommendations
<b>NetCache appliance</b>	<ul style="list-style-type: none"> <li>◆ A network interface is required for each router attached.</li> </ul>
<b>Network layout</b>	<ul style="list-style-type: none"> <li>◆ The router must be in a location to detect all traffic for the clients it is expected to serve.</li> </ul>
<b>Router configuration</b>	<ul style="list-style-type: none"> <li>◆ Router configuration includes configuring WCCP service group information to correspond to service group configuration on the NetCache appliances. See Cisco documentation for information about setting up WCCP 2.0.</li> </ul>
<b>NetCache configuration</b>	<p>Configuring the NetCache appliance for transparent proxying with a WCCP router includes the following:</p> <ul style="list-style-type: none"> <li>◆ For each protocol for which you want to set up transparent proxying, enable the transparency feature.</li> <li>◆ Configure service group information, which includes identifying traffic to be redirected to the NetCache appliances and the request distribution method to be used. NetCache can then send information to the router that router can use to configure itself.</li> </ul> <p>See the online Help for configuration details.</p>

## Key features for switches

When evaluating switches to use for transparent proxying, look for the following key features:

- ◆ For Web caching or streaming media caching, failover routing to the local or remote server when all Web caches or streaming media caches fail
- ◆ A flexible algorithm for distributing requests among the NetCache appliances
- ◆ The capability to limit the number of connections to a particular NetCache appliance
- ◆ For Web caching service and streaming media caching, support for access control lists so that service can be disabled for particular Web servers, streaming servers, and clients
- ◆ The capability to detect failure of the NetCache appliance by using HTTP to poll the cache

- ◆ The capability to detect and avoid potential routing loops from NetCache appliance redirection that can occur between the switch, routing equipment, and NetCache appliances
- ◆ Administrative security
- ◆ Redundant switch failover deployment
- ◆ The capability to load balance to NetCache appliances on different subnets

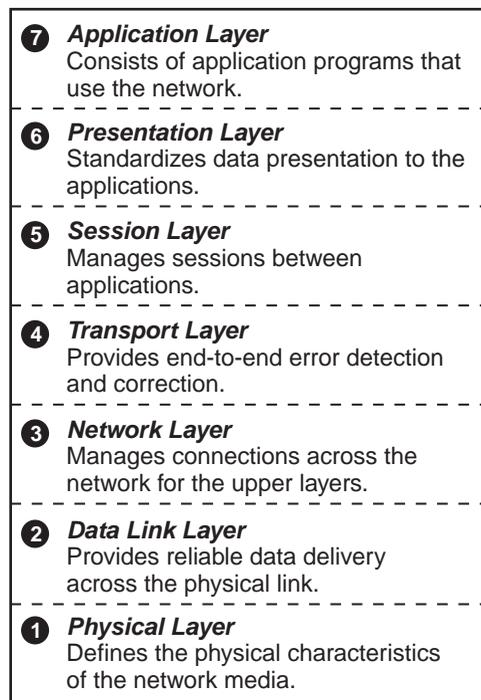


# OSI Model Relationship with Switches

**About this appendix** This appendix introduces the Open Systems Interconnection (OSI) Reference architectural model to briefly provide background information about the two types of switches that are discussed in this guide—the Layer 4 (L4) and the Layer 7 (L7) switches. These two switches function at different layers of the OSI model and, as a result, have some different capabilities.

## What is the OSI model?

The OSI model was developed by the International Standards Organization. It is used to describe the structure and function of data communications protocols. The OSI model contains seven layers that define the functions of communications protocols, as shown in the following illustration.



TCP/IP does not implement Layer 5 or Layer 6 of the OSI model. It is implemented on top of Layer 4.

## Layer 4 switch

An L4 switch operates at Layer 4, the *Transport layer* of the OSI model. An L4 switch looks at port numbers in a TCP/IP packet and, based only on the port number, passes port 80 (HTTP), port 119 (NNTP), port 1755 (MMS), port 554 (RTSP), port 21 (FTP), and port 53 (DNS) traffic to one of the NetCache appliances that it has been configured to be aware of. For Web caching and streaming media caching, an L4 switch can be configured to pass non-HTTP traffic directly to the Web.

These switches are called Layer 4 devices because they base their switching decisions on information in the TCP header, and TCP is a protocol that resides at Layer 4 in the OSI seven-layer model.

## Layer 7 switch

An L7 switch operates at Layer 7, the *Application layer*, of the OSI model. The Application layer includes all processes that use the Transport layer (L4) protocols to deliver data. The most widely known and implemented protocols at the Application layer are Telnet, FTP, SMTP, and HTTP. Because these switches operate at Layer 7, they can understand URLs and can understand much more about the traffic than an L4 switch can.

An L7 switch provides the same features that an L4 switch provides, plus additional, more sophisticated features. Whereas an L4 switch can only examine port numbers in a TCP/IP packet, an L7 switch examines the port number in a TCP/IP packet to determine if it needs to examine the packet further.

For HTTP requests (typically port 80), the L7 switch examines the request and determines whether the object is cacheable. This capability allows you to set up your switch so that requests for obviously uncacheable objects, such as CGI and URLs with cookies, can bypass the cache completely. Uncacheable objects are obtained directly from a Web server.

**About this appendix** This appendix describes considerations for using the DataFabric® Manager distribution management feature to push content to NetCache appliances.

**Appendix contents** This appendix contains the following sections:

- ◆ [“Reasons for pushing content to NetCache appliances”](#) on page 278
- ◆ [“Content distribution and management using DataFabric Manager”](#) on page 280

## Reasons for pushing content to NetCache appliances

---

### Centralized Internet data delivery is becoming less viable

In the traditional model of content delivery with caching services, a user requests content, then the NetCache appliance obtains it from the origin server and delivers the content to the requesting users while caching it (if it is cacheable). The content is then available for the next client that requests it. The cache, therefore, is populated only as a result of user requests. This is the *pull* model of content delivery.

As the Internet is maturing, however, requirements by organizations that provide content to users are changing. Providers of content (for example, enterprises, content delivery networks (CDNs), service providers, and e-commerce vendors) want to be able to provide high-quality data reliably and quickly. With the growing use of multimedia and interactive content in particular, the centralized model of providing data from the center is no longer working well. The reason is that file sizes for multimedia are extremely large (unlike HTTP) and require large amounts of network bandwidth. Additionally, network congestion greatly affects the quality of the content that is delivered.

Providers of content are becoming increasingly interested in controlling content so that they can provide the best possible experience for their end users. They want to *push* content from the center of the network to remote offices, closer to end users, and they want to track the use of that content.

### Benefits of pushing content close to users

Some advantages of pushing content to NetCache appliances rather than relying on the traditional caching services model for populating a cache are as follows:

- ◆ You can target specific content for particular locations and make the content available before peak demand. For example, you might want a training film for human resources personnel at the main corporate site and a training film for sales personnel at a remote site. ContentReporter reports can help you plan content to push based on comprehensive usage reporting. ContentReporter is available from the WebWasher Corporation.
- ◆ You can proactively distribute content at off-peak hours, which enables you to optimize bandwidth use and ensure that users enjoy fast response times.
- ◆ For enterprises, bandwidth limitations, a variety of links, a variety of clients, and single points of failure are among the problems that affect an enterprise's ability to deliver data efficiently. Pushing content close to users helps reduce the effects of these problems on enterprise content delivery.
- ◆ You can reduce the impact on quality that network congestion causes for streaming media. As the distance the streaming media data travels lessens,

the possibilities of network problems that would affect the streaming media quality are likely to be fewer. Additionally, you are copying streaming media files to the NetCache appliance rather than caching a stream that could have been affected by network conditions (for example, thinned).

- ◆ Streaming media is transferred at wire speed rather than at playback speed; that is, the time it takes for a two-hour video to be transferred to a NetCache appliance is the time it takes to complete the file transfer rather than the two hours it would take to play the video.

# Content distribution and management using DataFabric Manager

---

## About distribution management

The distribution management feature of DataFabric Manager automates the distribution of content from web servers to NetCache appliances.

## Supported content types

DataFabric Manager can distribute the following types of content:

- ◆ HTTP based, such as HTML, XML, PDF, JPEG, GIF, and NetCache software update
- ◆ Streaming based on MMS or RTSP protocols, such as Real, Windows Media, and Quicktime, including Real content that has been packaged for digital rights management (.rms files).

For details about the DataFabric Manager distribution management feature, see the *DataFabric Manager Administration Guide*.

# Glossary

---

## **authoritative DNS nameserver**

In standard DNS, an authoritative DNS nameserver is one that has knowledge of the DNS domain to which the origin server belongs.

In GRM deployments, GRM DNS servers interact with authoritative DNS nameservers to resolve DNS queries for unrecognized domains and to perform other functions, such as zone transfers.

In the context of GRM, the GRM DNS server can appear to be the authoritative DNS server for a content provider, if a GRM DNS server takes over DNS for the existing CDN namespace. In this case, the GRM DNS server assumes the identity of the authoritative DNS nameserver (also referred to as the back-end DNS nameserver), and subsequently resolves all DNS queries for domains associated with the CDN namespace. DNS queries for non-CDN domains and other DNS operations are proxied through the GRM DNS server to the back-end DNS nameserver.

## **back-end DNS nameserver**

The authoritative DNS nameserver for the CDN. The GRM DNS server must use the back-end DNS nameserver to resolve DNS queries for non-CDN domains and to perform other DNS operations. These operations are proxied through the GRM DNS server to the back-end DNS nameserver.

## **cache**

The location in which objects are stored until they expire or are replaced by an updated version.

## **cache hit**

The process by which a NetCache appliance receives a request for an object and can resolve the request from its cache rather than having to fetch the object from the origin server.

## **cache miss**

The process by which a NetCache appliance receives a request for an object but that object is not in the cache. The NetCache appliance then attempts to fetch the object from the origin server.

<b>center caches</b>	Network appliances on which the GRM agent software is enabled that have been identified in the GRM server configuration as a center cache. Edge caches are located in <i>edge groups</i> , such as POPs, that are remote from the data center. Edge caches in a particular edge group serve only the clients associated with that edge group.
<b>CIDR</b>	See Classless Inter-Domain Routing.
<b>Classless Inter-Domain Routing (CIDR)</b>	A type of network routing that uses the network mask instead of the address class to determine the destination network.
<b>cluster</b>	Two or more NetCache appliances or firewalls over which the NetCache appliance distributes requests for the purposes of load balancing and failover.
<b>Content Delivery Network (CDN)</b>	A network service provider that distributes content and services to optimize content delivery and charges a fee to do so.
<b>content provider</b>	Any company in the business of aggregating or producing content for delivery via the Internet.
<b>cookie</b>	Information sent by a Web server to a client about a particular Web page. This information is generally used to maintain user-specific information and to adjust the appearance of a particular page accordingly, allowing for more personalized content. However, a Web page with a cookie attached to it requires special attention and might be uncacheable.
<b>Domain Name Service (DNS)</b>	Domain Name Service (DNS) is the standard mechanism on the Internet for advertising and accessing a variety of information about computer hosts. DNS handles mapping between host names, which humans find convenient, and Internet addresses, which computers require. DNS makes host information available throughout the Internet, distributed among many sites and organizations, not just in a single central site.

<b>DynaBLocator</b>	One of two types of software on a NetCache appliance that restricts access to Web sites. It identifies the sites that contain content in various categories, maintains lists of those sites in control files, and restricts access accordingly.
<b>edge caches</b>	Network appliances on which the GRM agent software is enabled that have been identified in the GRM server configuration as an edge cache. Edge caches are located in edge groups, such as POPs, in a location remote from the data center. Edge caches in a particular edge group serve only the clients associated with that edge group.
<b>edge group</b>	<p>A group remote from the data center in which edge caches are located, such as a POP, are referred to as <i>edge groups</i>. Edge groups are identified in the GRM server configuration.</p> <p>Edge caches in a particular edge group serve only the clients associated with that edge group.</p>
<b>failover</b>	The process by which an alternate system takes over and emulates the primary system if the primary system becomes unusable. See also takeover.
<b>Feeder server</b>	The component of a news server that accepts news feeds from other news servers. The Feeder server might be on the same computer as other news server software components or it might be on a different computer.
<b>firewall</b>	A server or device deployed between a LAN and the external WAN to control all inbound and outbound traffic, thus the enhancing security of the network behind the firewall.
<b>flap</b>	See network flap.
<b>forward proxy</b>	From the client's perspective, a forward proxy operates on behalf of a client Web browser or media player. A forward proxy can handle requests for a virtually unlimited number of servers. Forward proxy servers are located close to the client. NetCache appliances operating as Web caches, news caches, and streaming media caches are forward proxies.

<b>FTP</b>	File Transfer Protocol.
<b>fully qualified domain name (FQDN)</b>	The complete name of a specific computer on the Internet, consisting of the computer's host name and its domain name. A host's FQDN consists of its host name and its domain name, separated by a period. For example, the host euler, located in the domain netapp.com, has an FQDN of euler.netapp.com.
<b>Gopher</b>	A predecessor of the HTTP protocol, and is seldom used today.
<b>GRM agent caches</b>	A generic term to refer to NetCache appliances on which the GRM agent software is enabled. In the GRM server configuration, a cache on which the GRM agent software is installed can be identified as a center cache or an edge cache.
<b>GRM server</b>	A NetCache appliance that is typically located in a CDN data center and is configured as a GRM server. A NetCache appliance can be configured as a GRM DNS server, if DNS-based redirection is desired, or as a GRM L7 server, if application-layer (L7) based redirection is desired.
<b>heartbeat</b>	A repeating signal transmitted from one appliance to another that indicates that the appliance is in operation. Heartbeat information is also stored on disk.
<b>HTTPS</b>	Hypertext Transport Protocol Secure. A variant of HTTP used for accessing secure Web servers. See also Secure Sockets Layer (SSL).
<b>ICAP</b>	Internet Content Adaptation Protocol. A high-level protocol that provides a common format for requesting services from a server.
<b>ICP</b>	Internet Cache Protocol. The method by which a hierarchy of NetCache appliances can communicate with each other. A derivative of UDP, ICP enables NetCache to query other caches for individual Web pages.

<b>IP address</b>	The unique numerical address of a computer that is attached to a TCP/IP network, for example, 198.95.226.66.
<b>IP spoofing</b>	You can use the NetCache appliance IP spoofing feature to configure NetCache to use the client IP address as the source address when communicating with servers. Requests originating from a client retain the client's source address even if requests are passed through a chain of proxy-cache servers. The requests, therefore, appear to originate from the client rather than from the NetCache appliance.
<b>L4 or L7 switch</b>	Switches typically include multiple methods for handling traffic. When this guide refers to L4 or L7 switches, it refers to a feature on a switch that can support transparent proxying; that is, a feature that operates on the L4 or L7 layer of the OSI model. Switch names vary by vendor.
<b>LDAP</b>	Lightweight Directory Access Protocol server. LDAP is a client-server protocol for accessing a directory service. NetCache can be configured to point to an LDAP server for authentication of user requests.
<b>live media stream</b>	A live media stream occurs in real time, like the news program on a television set. Some organizations record a live media stream and then broadcast the media stream to their employees or customers at a specified time. All users who have requested the media stream see the same media stream at the same time. Users are not able to rewind or fast-forward a live media stream.  See also stream, on-demand media stream, and streaming media.
<b>load balancing router</b>	See Server Load Balancer.
<b>megabyte (MB)</b>	1,048,576 bytes, or 1,024 KB. Unit of measure for computer storage.
<b>Microsoft Media Streaming (MMS)</b>	The streaming media control protocol used in Windows Media 3 & 4 streams.

**MMS**

See Microsoft Media Streaming.

**multicast in (input)**

Process by which NetCache receives a live stream through the multicast transport protocol from a streaming server that is configured to support multicast streaming. If set for multicast input for streaming media, NetCache always attempts to receive data from the upstream streaming server through multicast.

**multicast out (output)**

Process by which NetCache sends the live stream to clients through the multicast transport protocol regardless of how the stream is received from the streaming server. If set for multicast output for streaming media, when a client requests a multicast stream, NetCache always sends the received data to the client through multicast, regardless of how the stream was received.

**multicast streaming**

Process by which NetCache delivers live content to a large number of clients. In the multicast streaming model, NetCache receives one copy of the requested stream through an IP multicast channel from the streaming server. A network of routers then splits the stream to all clients on the local network that are listening for a multicast broadcast at the group address. See also unicast streaming.

**Network Access Point (NAP)**

The Internet encompasses more than 95,000 networks worldwide. The networks in the Internet are linked together in the United States at several major intersection points. One type of intersection point is a Network Access Point. There are three NAPs. These NAPs ensure the continued broad-based access to the Internet.

A Route Arbiter server is located at each NAP. The server provides access to the Routing Arbiter Database. ISPs can query Route Arbiter servers to validate the reachability information advertised by an autonomous system. (An autonomous system is a collection of networks and gateways with its own internal mechanism for collecting routing information and passing it to other independent systems.)

**network flap**

Network routes that change rapidly over a short period of time. Flapping routes strain routers and can lead to broken connections if clients' packets switch between paths with different characteristics; for example, paths that include transparent redirection to a NetCache appliance and paths that do not implement transparency.

<b>news cache</b>	A NetCache appliance that is configured to accept news requests from clients and to fetch NNTP objects from a news server or another NetCache appliance that connects to a news server.
<b>NNTP</b>	Network News Transfer Protocol. A news transmission protocol for the Internet that can be used to transport news from one host to another. Also, newsreader software can use NNTP to access the news database on a server host. One great advantage that NNTP has over UCP, which preceded NNTP for news transport, is that an NNTP host receiving multiple news feeds generally receives only one copy of each article.
<b>nontransparent firewall running as a proxy</b>	A type of firewall NetCache must be aware of in order to forward requests through it to the Internet. That is, NetCache must be configured to send requests directly to the correct firewall IP address and port.
<b>NTLM</b>	NT LAN Manager. A Windows NT authentication protocol; supported by NetCache.
<b>on-demand media stream</b>	A previously recorded media stream that users can request at a time most convenient to them. Users can rewind and fast-forward on-demand media streams. Also known as video-on-demand or VOD.  See also stream, live media stream, and streaming media.
<b>OSI model</b>	Open Systems Interconnection (OSI) Reference architectural model. The OSI model contains seven layers that define the functions of communications protocols.
<b>point of presence (POP)</b>	A remote location that provides network connectivity back to the core of the network for users within its service area. Service providers deploy POPs to give customers an entry point to their networks from their local loops.
<b>POP</b>	See point of presence.

<b>policy routing</b>	See routing policies.
<b>presentation</b>	See streaming media presentation.
<b>RADIUS</b>	Remote Authentication Dial-in User Service. RADIUS is a client-server protocol for accessing a directory service. A NetCache appliance can be configured to point to a RADIUS server for authentication of user requests.
<b>Reader server</b>	The component of a news server that obtains the data that the news cache requests from a news storage device, such as a Network Appliance filer. When the news cache does not have requested news in its cache, it connects to the Reader server component of the news server, through NNTP. The Reader server might be on the same computer as other news server software components or it might be on a different computer.
<b>Real Time Streaming Protocol (RTSP)</b>	An open standard for delivery of real-time media over the Internet. See also streaming media.
<b>Real-Time Transport Control Protocol (RTCP)</b>	A feedback protocol that provides quality of service information to the streaming server. The RTCP feedback mechanism periodically describes the state of the network so that the streaming server and the client can negotiate the optimum bit rate for current network conditions. By default, RTCP uses UDP for transport.
<b>reverse proxy</b>	<p>A reverse proxy server (also referred to as an accelerator) handles requests on behalf of the origin server, acting as an extension of the origin server. A reverse proxy, unlike a forward proxy, services one or a few origin servers. Random servers cannot be accessed through a reverse proxy server. Clients use the reverse proxy to access all origin servers that the reverse proxy is servicing.</p> <p>A reverse proxy server is usually operated by the same organization that operates the origin servers it services and it is located close to the origin server.</p>
<b>routing policies</b>	Rules that control what routes are accepted and what routes are advertised.

<b>RTCP</b>	See Real-Time Transport Control Protocol.
<b>RTSP</b>	See Real Time Streaming Protocol.
<b>Secure Sockets Layer (SSL)</b>	A protocol that is a standard method for secure, encrypted communication over the Internet. See also HTTPS.
<b>SecureAdmin</b>	A security product that enables you to administer a NetApp appliance in a nontrusted environment by creating a secure exchange between a client and the NetApp appliance using SSH and SSL protocols. Developed by Network Appliance.
<b>Server Load Balancer (SLB)</b>	<p>A device that can balance requests over servers and hosts, such as NetCache appliances. SLBs are likely to provide strict load-balancing algorithms (such as least connections, weighted percentage, fastest connections, and round-robin) rather than to distribute requests based on IP address.</p> <p>This device might also be referred to as a <i>load-balancing router</i>.</p>
<b>service group</b>	See WCCP service group.
<b>SmartFilter</b>	One of two types of software on a NetCache appliance that restricts access to Web sites. It identifies the sites that contain content in various categories, maintains lists of those sites in control files, and restricts access accordingly.
<b>splitting</b>	The process of delivering a unique stream to multiple clients simultaneously. Splitting reduces the number of requests sent across the network to the streaming server. See also stream and streaming media.
<b>stream</b>	A streaming media file being transmitted over a network. Streams are of two types, live media streams and on-demand media streams. See also live media stream, on-demand media stream, streaming media, and unique stream.

**streaming accelerator**

A NetCache appliance that caches content from one or more streaming servers that you identify. A streaming accelerator provides that content to clients that request it. To the outside world, the streaming accelerator is the streaming server. In contrast, when NetCache runs as a streaming media cache, it acts as an agent for the media player.

**streaming media**

A term used to describe media files that begin playing while they are being transmitted over the network to the media player on the client computer. In contrast, conventional Web files must be downloaded entirely before the user can view them. Commonly requested types of streaming media are video and audio. Streaming media also includes interactive media, cartoon-like animations, panoramic data, and more.

**streaming media cache**

A streaming media cache is a specially configured NetCache appliance that intercepts client requests for streaming media (RTSP, MMS, or both). When a streaming media cache is deployed between clients and a streaming media server, streaming requests that would otherwise have been sent directly to the streaming server are sent to the cache. For live streaming media presentations, bandwidth savings are realized when multiple clients request the same unique stream. NetCache caches on-demand media streams. See also [splitting](#), [streaming media](#), [stream](#), and [on-demand media stream](#).

**streaming media presentation**

Streaming media presentation is a general term used to describe the delivery of live or on-demand streaming media. Multiple unique streams can make up a streaming media presentation. See also [stream](#), [unique stream](#).

**takeover**

The configuration of one specific NetCache appliance to take over NetCache servicing functions of one other NetCache appliance in the event of failure.

**thinning**

In streaming media, the streaming server's process of dropping a consistent number of video or audio frames to try to make the data being delivered to the client more efficiently during playback.

See also [streaming media](#).

**Time To Live (TTL)**

The time limit on how long a copy of an object remains in the cache before NetCache verifies the object with the originating Web server.

<b>traditional caching service model</b>	<p>In this model, the proxy-cache server cache is populated as a result of user requests. If the content a user requests is not in the proxy-cache server, the proxy-cache server obtains the requested content from the origin server and delivers it to the requesting users, and at the same time saves a copy of the object in its cache (if the content is cacheable). The content is then available for the next client that requests it. The cache, therefore, is populated only as a result of user request. This method can be thought of as the pull concept of content delivery.</p> <p>Contrast this model with prefilling the cache, that is, pushing content to a NetCache appliance.</p>
<b>transparent firewall</b>	<p>A firewall that NetCache can connect through without having to know the IP address of the firewall.</p>
<b>transparent proxying</b>	<p>A method for providing access to NetCache appliances in which users do not have to manually configure Web browsers or media players for requests to be sent to the NetCache appliances. Transparent proxying is typically deployed with an L4 switch, and L7 switch, or a WCCP router. Less commonly, it is deployed with a policy-based router.</p>
<b>TTL</b>	<p>See Time To Live.</p>
<b>UDP</b>	<p>User Datagram Protocol. A connectionless network protocol that provides simple but unreliable transmission. An alternative to TCP that does not contain error checking.</p>
<b>unicast streaming</b>	<p>One-to-one transmission in which each client opens a connection to a NetCache appliance to retrieve streaming content. In unicast streaming, NetCache receives a copy of the requested stream from a streaming server and sends a separate copy to each requesting client. See also multicast streaming.</p>
<b>unique stream</b>	<p>A stream whose connection speed, media type, or thinning parameters are different from another stream that is part of the same streaming media presentation. See also stream and streaming media.</p>

<b>URL</b>	Uniform Resource Locator. To find a Web resource, you enter a name called a URL in the Location field of your Web browser, for example, <a href="http://www.netapp.com/">http://www.netapp.com/</a> .
<b>video on-demand</b>	A previously recorded media stream that users can request at a time most convenient to them. Users can rewind and fast-forward on-demand media streams. Also known as an on-demand stream. See also stream, live media stream, and streaming media.
<b>VIF</b>	A single virtual interface that is created from multiple physical interfaces by using the EtherChannel technology.
<b>virtual interface (VI)</b>	An architecture designed to allow bulk data transfer directly to or from application buffers. VI also allows applications to access VI-capable hardware directly without operating system intervention. See also VIF.
<b>VOD</b>	See video on-demand
<b>WCCP 2.0-based router or WCCP router</b>	A Cisco router running WCCP 2.0 software.
<b>WCCP service group</b>	In WCCP, traffic redirection and distribution is based on logical WCCP service groups. A service group is a group of one or more routers and one or more proxy-cache servers that can work together in traffic redirection and distribution because they have been defined, through a service group definition, with the same settings.
<b>Web accelerator</b>	A NetCache appliance that caches content from one or more Web servers that you identify. A Web accelerator provides that content to clients that request it. To the outside world, the Web accelerator is the Web server. In contrast, when NetCache runs as a Web cache, it acts as an agent for the browser.

**Web cache**

A NetCache appliance configured as an intermediary server that accepts requests from clients and forwards them to a Web server or, if appropriate, services the requests from its own cache. A Web cache acts as an agent for the client browser. A NetCache appliance configured as a Web cache handles any or all of HTTP, FTP, Gopher, and Tunnel (for example, HTTPS and SSL) requests.

**WebWasher  
DynaBLocator**

See DynaBLocator

**Windows Media  
Technologies  
(WMT)**

A collective term for Microsoft's streaming audio and video technologies.



# Index

---

## A

accelerator. *See* Web accelerator, streaming  
accelerator  
access controls

- blocking Web sites 85
- controlling NetCache access by IP address 184
- NetCache feature described 9
- with NetCache takeover 81

Apple corporation, QuickTime

- NetCache support for 111

Application layer, and L7 switches 276  
articles (news)

- caching of 194
- how a news server obtains 189
- how the news cache obtains 189
- message ID for 194

audio, bandwidth use 104  
authentication

- firewall and NetCache 232
  - with LDAP server 233
  - with NTLM 233
  - with RADIUS server 233
- IP-based, caution 24
- NetCache feature described 9
- news caching 232
- streaming media caching 113, 232
- with NetCache takeover 81

authoritative DNS nameserver

- defined 281
- interaction with GRM server 249

automatic proxy configuration file

- advantages of 56, 264
- browsers that support 265
- described 54, 263, 264
- disadvantages of 56, 265
- examples 267, 270
- failover capabilities 54, 267
  - advantages of 56
  - setting up 267
- file location 265
- how requests are distributed 55, 268
- setting up request distribution 268

use with GRM 249, 252  
when not to use 57, 265

## B

back-channel multicast

- defined 122
- NetCache support for 122

back-end DNS nameserver

- defined 281

bandwidth

- savings with proxy-cache servers 3
- with news caching
  - amount required for 199
  - saving bandwidth 189
  - usage scenario 199
- with streaming media caching
  - between clients and cache 128
  - effect on quality 104, 112
  - object delivery over 127
  - saving bandwidth 112
  - use, high 104, 127
- with Web caching
  - object delivery over 127
  - saving bandwidth 85, 89

bandwidth allocation feature

- described 10

bCandid software, supported by NetCache 193  
bit rates

- client changing request for stream 102
- compared to delivery bit rate 102
- encoding in a media stream 102

browsers

- configuring to point to NetCache 58
- supported by NetCache 2
- supporting an automatic proxy configuration file 265

## C

cache

- defined 3, 281
- how populated 3

- Cache Array Protocol (CARP) 73
- cache hit, defined 281
- cache miss, defined 281
- caching. *See* Web cache, news cache, streaming media cache
- CDNs
  - benefit of multicast 130, 163
  - defined 282
  - requirements 244
- CIDR, defined 282
- client access to a NetCache appliance
  - direct (nontransparent)
    - automatic proxy configuration file 54, 263
    - DNS round robin with 63
    - methods for 51, 52
    - pointing browsers to 58
    - request distribution with 62
    - SLB with 65
    - streaming media cache 60
  - DNS cache, with transparent proxying 25
  - global request routing (GRM) 68
  - news cache 188
    - transparent proxying 25
  - strategies for 17
    - summary 18
  - streaming accelerator 168, 172
    - controlling by IP address 184
    - with DNS 173
  - streaming media cache 133
    - nontransparent methods 60, 134, 150
    - pre-WMP 7.0 media player 60, 150
    - transparent proxying 25
  - transparent proxying 21
    - DNS cache 21
    - news cache 21
    - streaming accelerator 172
    - streaming media cache 21
    - Web accelerator 172
    - Web cache 21
  - Web accelerator 168, 172
    - controlling by IP address 184
    - with DNS 173
  - Web cache
    - transparent proxying 25
- client access to a Windows Media server
  - nontransparent method 60
- cluster
  - defined 282
  - failover in 74
  - request distribution among caches 73
- connections
  - how a news cache decreases 189
  - number a news cache supports 190
  - number a news server supports 190
  - persistent 96
  - streaming media
    - described 127
    - effect of cache failure on 27
    - HTTP for 118
    - to server 132
  - Web server
    - contrast with streaming media 127
- content
  - pushing to NetCache appliances 5, 278
  - serving in remote locations
    - importance of 245, 279
- content adaptation. *See* ICAP
- content delivery
  - populating a cache 3, 278
  - pushing content
    - benefits 5, 140, 278
    - type of streaming media 140
- content distribution 280
- content filtering
  - ICAP service described 205
  - SmartFilter 9
  - WebWasher DynaBLocator 9
- content provider
  - controlling content for 278
  - defined 282
- content pushing
  - benefits 140, 278
  - reasons for 140, 278
  - streaming media
    - quality impact 128, 140
    - scenario 155
  - use of ContentReporter reports with 278
- ContentReporter
  - report use for content pushing 278
- cookies

- defined 282
- problems with DNS round robin 64
- setting up switch to bypass cache 276

## D

- deployment considerations 14

- deployment examples

  - Data Center 48, 75, 93

  - DNS cache 43

  - enterprise environment 89

  - firewalls

    - accessing Web server outside 234

    - NetCache inside multiple firewalls 230

    - NetCache inside nontransparent firewall 228

    - NetCache inside transparent firewall 227

    - NetCache parallel to 226

    - using NetCache with 225

  - global carrier and ISP 91

  - GRM 253, 256

  - high latency links 95

  - ISP 93

  - NetCache takeover pairs 78

  - Network Hub 91, 93

  - news caching 187

    - ISP Data Center and POPs 199

  - routing 237

    - incoming traffic over multiple links 240

    - outgoing traffic over multiple links 238

  - SLB and NetCache 65

  - streaming accelerator 167, 174

  - streaming media 35, 45

    - with cache prefill 155

    - with enterprise 146, 155

    - with ISP 142

    - with Windows Media server 150

  - transparent proxying 43

    - DNS proxy cache 43

    - failover to the Web 26

    - global carrier and ISP 91

    - multiple NetCache appliances 47

    - POP and Data Center 48

    - streaming media and news caches 46

      - streaming media cache 45

      - Web accelerator 167, 174

      - Web caching 89

        - over satellite links 95

  - deployment planning

    - defining goals 12

    - describe environment 13

  - Diablo software, supported by NetCache 193

  - distribution

    - supported content types 280

  - DNS

    - client access to accelerators 172, 173

    - contrast standard with GRM routing 246

    - defined 282

    - how standard DNS works 246

    - purpose of 246

    - requests, transparent proxying with 21

    - supported by NetCache 2

    - traffic

      - how a switch handles 25

      - how a WCCP router handles 25

  - DNS cache

    - deployment example 43

    - nontransparent proxying 43

    - transparent proxying 25

  - DNS nameserver

    - authoritative

      - in standard DNS routing 246

      - limitations of 246

  - DNS round robin

    - advantages 64

    - described 63

    - disadvantages 64

    - load balancing 63

    - setting up 63

    - streaming accelerators

      - NetCache takeover pairs with 180

    - Web accelerators

      - NetCache takeover pairs with 180

  - DNS routing, GRM use of 243

  - Domain Name Service. *See* DNS, DNS round robin

  - DynaBLocator, NetCache feature described 283

- E**
- edge caches, defined 282, 283
  - edge groups, defined 283
  - electronic commerce sites
    - caution with transparent proxying 24
  - enterprise environment
    - Web caching deployment example 89
  - error messages
    - 400 service discontinued 192
- F**
- failover
    - automatic proxy configuration file 54, 264
    - defined 283
    - in a NetCache logical cluster 74
    - NetCache takeover pairs 78
      - described 78
      - interface requirements 80
    - news cache 196
      - when possible to Reader server 196
    - streaming media cache
      - strategies for 135
      - streams being delivered 135
      - when possible to streaming server 135
    - transparent proxying
      - failover to the Web 26
      - switch failover pair 29
      - with a switch 26
    - Web cache 86
  - Feeder server
    - definition of 191
    - illustration of 191
  - Feeder server, defined 283
  - firewalls
    - accessing Web server outside of 234
    - authentication 232
      - Kerberos machine inside of 233
      - Kerberos machine outside of 233
      - LDAP server inside of 233
      - LDAP server outside of 233
      - NTLM machine inside of 233
      - NTLM machine outside of 233
      - RADIUS server inside of 233
      - RADIUS server outside of 233
    - defined 283
    - multicast support requirements 137
    - NetCache deployment with 225
    - NetCache inside of
      - multiple 230
      - nontransparent firewall 228
      - transparent firewall 227
    - NetCache parallel to 226
    - nontransparent, defined 228, 287
    - security and NetCache appliances 137, 226
    - streaming media through 137
    - transparent, defined 227, 291
  - forward proxy
    - defined 6
    - request distribution with
      - switch 31
      - WCCP router 40
  - FQDN, defined 284
  - FTP
    - defined 284
    - requests
      - over HTTP 84
      - transparent proxying with 21, 84
    - supported by NetCache 2
    - traffic
      - how a switch handles 22, 25
      - how a WCCP router handles 22, 25
- G**
- global carrier deployment example 91
  - Global Request Manager. *See* GRM
  - Gopher
    - defined 284
    - supported by NetCache 2
  - GRM
    - center caches, defined 248
    - center groups, defined 248
    - components 248
    - content handled 248
    - deployment examples
      - enterprise 256
      - telco 253
    - deployment planning 14
    - determining if it is the right solution 246

- edge caches, defined 248
- edge groups, defined 248
- GRM agent caches
  - defined 248
- GRM DNS servers
  - defined 249
  - when to use 250
- GRM L7 servers
  - defined 249
  - methods for directing requests to 250
  - when to use 250
- GRM servers
  - defined 248
  - types 249
- how standard DNS resolution works 245
- integration with NetApp products 258
- licensing requirements 252
- overview of 244
- redirection methods 248
- versus transparent proxying 247
  - advantage 244
- with distributed enterprises 250

GRM agent caches

- defined 284

GRM routing

- contrast with DNS routing 246

GRM servers

- defined 284

IP address

- of authoritative DNS nameserver 249

types

- guidelines for 250

## H

hierarchy

- cluster
  - as a parent 75
  - defined 72
  - example, ISP 91, 93
- description of 71
- ensuring requests go through the firewall 228
- planning 72

high bandwidth links, deployment example 95

high latency links, deployment example 95

hit rate

- request distribution
  - accelerator 32, 41
  - switch 31
  - WCCP router 40
  - with an accelerator 170

HTTP

- requests, transparent proxying with 21
- supported by NetCache 2
- traffic
  - how a switch handles 25
  - how a WCCP router handles 25
  - how firewalls handle 225

HTTP encapsulation

- client side 118
- server side 118
- with accelerators 173

HTTPS

- requests, transparent proxying with 21

HTTPS, defined 284

## I

ICAP 10, 203

- benefits of 207
- cost of adding to network 217
- defined 205, 284
- deployment planning tasks 213
- failover 218
  - planning for 213
- feature summary 214
- issues for 207
- location of devices 218
- NetCache appliances
  - location of 218
- network type for 218
- security
  - planning for 213, 219
  - recommendations 219
  - risks with 219
- servers, ICAP
  - location of 218
  - NetCache interaction with 205
  - planning for 213, 218
- services

- content filtering, described 205
  - distributing 216
  - examples 205
  - multiple, for same content 217
  - planning for 213, 216
  - virus scanning, described 205
  - virus scanning, process example 210
  - virus scanning, scenario 222
  - when invoked 208
  - SSL, no services with 219
  - traffic resulting from 217
  - vectoring points 208
    - request modification 209
    - response modification 209
    - response modification example 210
    - usage examples 209
  - version
    - supported by NetCache 207
  - ICAP servers. *See* ICAP
  - ICAP services. *See* ICAP
  - ICP, defined 284
  - INN software, supported by NetCache 193
  - Internet Content Adaptation Protocol. *See* ICAP
  - Internet Explorer, use with NetCache 2
  - IP address hashing
    - switch 31
    - WCCP router 40
  - IP addresses
    - access controls for accelerator 184
    - authoritative DNS nameserver
      - to GRM server 249
    - defined 285
    - streaming accelerator
      - aliases for 182
      - setting up DNS 175
    - Web accelerator
      - aliases for 182
      - setting up DNS 175
  - IP spoofing
    - benefits 24
    - defined 285
    - for IP-based authenticated sites 24
    - symmetric routing and 38
  - IP takeover. *See* NetCache takeover pairs
  - IP-based authentication Web sites
    - caution with transparent proxying 23, 24
    - configure router to bypass NetCache 24
    - configure switch to bypass NetCache 24
  - IPv6
    - packet encapsulation 259
    - proxy cache example 260
    - web accelerator example 261
  - IPv6-over-IPv4 tunneling 259
  - ISP
    - deployment example 91, 93
      - streaming media service 142
    - when not to use automatic proxy configuration
      - file 57, 265
- ## K
- Kerberos authentication protocol 232
- ## L
- LDAP server
    - authentication
      - for news service 193
      - NetCache takeover and 81
      - when a firewall exists 233
    - defined 285
    - NetCache access when a firewall exists 233
  - license
    - for GRM 252
    - for news caching feature 190
    - for streaming media feature 111
  - Lightweight Directory Access Protocol server. *See* LDAP server
  - LDAP server
  - live media streams 101
    - defined 285
    - how NetCache handles 114
      - multicast 120
      - unicast 115
    - NetCache splitting of 117, 125
    - NetCache support for 111
    - path NetCache takes 116, 123, 124
    - typical client-server model 104
  - load balancing
    - DNS round robin 63
    - over NetCache network interfaces 42
    - SLB, described 65

- load balancing router, defined 285
  - logs, NetCache feature described 9
  - L4 or L7 switch
    - advantages of using 28
    - key features for 272
    - location in relation to clients 29
    - location in relation to NetCache 29
    - news cache 25
    - number needed 29
    - performance comparison 28
    - port redirection 25, 133
    - redirecting traffic to NetCache 25
    - relationship to OSI model 28
    - request distribution
      - IP address hashing 31
    - request distribution, IP address hashing 31
    - streaming media cache 25, 135
      - traffic to redirect 133
    - use with accelerators 172
  - L4 switch
    - how it operates 28
    - relationship to OSI model 276
    - versus GRM 244
    - versus L7 switch 28
  - L7 redirection with GRM 248
  - L7 switch
    - how it operates 28
    - noncacheable objects 172
      - setting up switch to bypass NetCache 276
    - noncacheable objects, bypassing cache 85
    - relationship to OSI model 276
    - request distribution by URL 32
    - versus L4 switch 28
- M**
- media players
    - interaction with NetCache 60, 134
    - supported by NetCache 112
    - traffic redirection with HTTP only setting 138
  - media streams
    - encoded at different bit rates 102
    - handled by NetCache 7, 111
    - live 101
      - how NetCache handles 114
      - how NetCache handles multicast 120
      - how NetCache handles unicast 115
      - typical client-server model 104
    - on-demand 101
      - how NetCache handles 123
    - message ID for an article 194
    - Microsoft Internet Explorer, use with NetCache 2
    - Microsoft Media Streaming. *See* MMS
    - MMS
      - directing streams to NetCache 133
      - license for feature 111
      - NetCache support for 2, 111
      - requests, transparent proxying of 21
      - See also* streaming media, streaming media cache
    - traffic
      - how a switch handles 25
      - how a WCCP router handles 25
  - multicast
    - back-channel, NetCache support for 122
    - benefits 108
    - defined 286
    - deployment considerations 130
    - device requirements 107
    - features of 106, 107
    - firewall support for 137
    - how NetCache handles 120
    - how switches handle 108
    - limitations 108
    - NetCache conversion from unicast 120
    - organizations that benefit from 130
    - scalable, NetCache support for 122
    - scenario
      - CDN 163
      - enterprise 157
      - satellite link 160
    - settings according to client activity 131
    - settings according to client location 131
    - types NetCache supports 122
    - when NetCache uses 120
  - multicast input, defined 120
  - multicast output, defined 120
  - multicast-enabled routers
    - traffic handled 120
  - multiple bit rates

encoding in a media stream 102

## N

NAP, defined 286

NetCache appliance

as a news cache 8, 187

overview 188

as a streaming accelerator 8, 168

overview 168

as a streaming media cache 7, 110

overview 7, 111

as a Web accelerator 7, 168

overview 168

as a Web cache 83

overview 2

in a GRM deployment 243

modes of operation 6

services provided to network clients 2

Web browsers to use with 2

NetCache features 9

NetCache takeover pairs

adding network fault tolerance 79

automatic proxy configuration file, with 81

described 78

interface requirements 80

use with transparent proxying 79

when feature is useful 79

NetCache user database

authentication when a firewall exists 232

Netegrity SiteMinder 9

NetNews software, supported by NetCache 193

Netscape Navigator, use with NetCache 2

Network Hub, deployment example 91

Network News Transport Protocol. *See* NNTP,

news cache

network routing flap

defined 286

effect on transparent proxying 23

news cache

authentication with 193

avoid chains of 198

bandwidth savings with 189

bandwidth usage scenario 199

benefits of deploying 189

caching articles 194

client types NetCache supports 193

communication with news server 188

connects to one news server only 189

cost to process requests 190

defined 188, 287

deployment considerations 86, 195

client access to 195

efficiency of versus news server 190

failover 196

available strategies 196

when possible to Reader server 196

how it gets news 189

how many are needed 195

keeping news fresh 194

license for feature 190

L4 switch with 25

maximizing hit rate 197

NetNews software, support for 193

News Overview for article 194

newsgroups, available 194

overview 188

quality of service 189

Reader server 190

relationship with a news server 189

scalability 189

transparent proxying with 21

what it caches 194

what it does not cache 194

news caching

cache-to-cache interaction 197

when news server goes down 192

news feeds, not accepted by NetCache 189

news server

components of 191, 288

cost to process requests 190

effect of going down 192

efficiency of versus news cache 190

how it gets news 189

Reader server component 190

relationship with a news cache 189

news storage device 191

newsgroups

availability of 194

information about 194

## NNTP

- defined 287
- how a switch handles traffic 25
- how a WCCP router handles traffic 25
- NetCache support for 2
- news cache to news server communication 188
- See also* news cache

## NNTP-compliant clients

- supported by NetCache 193

## noncacheable objects

- bypassing NetCache for 276
- cached in Web cache 85
- fetching through a hierarchy 72
- L7 switch with 32, 172

## nontransparent firewall

- defined 228, 287
- NetCache deployed inside 228

## NTLM

- authentication protocol 81, 232
- defined 287

## O

### objects

- hit rate with an accelerator 170
- noncacheable, switch bypasses NetCache 276

### on-demand media streams

- defined 287
- how NetCache handles 123
- NetCache support for 111

Open Systems Interconnect Reference model. *See*

OSI model

OSI model 28, 275, 287

## P

### persistent connections

- optimizing over satellite link 96

### planning for deployment

- defining goals 12
- describing environment 13

PNA traffic 173

### policy routing

- defined 288
- splitting NetCache traffic over links 238

### policy-based routers

setting up request distribution 22

transparent proxying 22

setting policy statements 22

POP, defined 287

prefilling a cache 5, 155, 278

presentations, streaming media 101, 290

protocols supported by NetCache 2

for streaming media 111

proxy.pac file. *See* automatic proxy configuration file

pushing content. *See* content pushing

## Q

### quality of service

caching 3

news caching 189, 199

streaming media 112

relationship to bandwidth use 3, 104, 128, 132, 278

quality of streaming media 128, 278

effect of closing firewall ports 137

querying other caches for objects 10

### QuickTime player

interaction with NetCache 60, 134

QuickTime, NetCache support for 111

## R

### RADIUS server

authentication for 233

news service 193

defined 288

NetCache access when a firewall exists 233

NetCache takeover and authentication 81

### Reader server

definition of 190, 191, 288

failover to, when it is possible 196

protecting from overload 197

Real Time Streaming Protocol. *See* RTSP

RealNetworks, NetCache support for 111

### RealPlayer media player

interaction with NetCache 60, 134

RealSystem, NetCache support for 111

redirection methods with GRM 248

redundancy

- news deployment 196
- providing with switches 29
- Remote Authentication Dial-in User Service. *See* RADIUS server
- request distribution
  - automatic proxy configuration file 55
  - NetCache hierarchy 10
    - cluster, defined 72
    - use for 71
  - news caching 188
  - over firewalls 225
  - policy-based routers 22
  - SLB 65
  - switches
    - hashing function 31
    - hit rate 31
    - methods available 31
    - on URL 31
  - transparent proxying
    - switch 30
    - WCCP router 39
  - WCCP routers 39, 41
    - hashing function 39
    - hit rate 40
    - methods available 39
    - weighting 41
- request forwarding, uses 24
- request modification. *See* ICAP, vectoring points
- request routing
  - with GRM 243
  - with L4 switches 244
- response modification. *See* ICAP, vectoring points
- reverse proxy
  - defined 6
  - request distribution with 31, 40
  - See also* Web accelerator or streaming accelerator
- router. *See* policy-based routers, WCCP routers, multicast-enabled routers
- routing
  - overview of GRM 244
  - with GRM 244
- routing deployment examples 237
  - incoming traffic over multiple links 240
  - outgoing traffic over multiple links 238

- routing flaps
  - defined 286
  - effect on transparent proxying 23
- routing policies, defined 288
- RTCP, defined 288
- RTSP
  - defined 288
  - directing streams to NetCache 133
  - how a switch handles traffic 25
  - how a WCCP router handles traffic 25
  - license for feature 111
  - NetCache support for 2, 111
  - requests, transparent proxying of 21
  - See also* streaming media, streaming media cache

## S

- satellite link
  - NetCache multicast support for 160
  - scenario showing 95
- scalability
  - NetCache and streaming media 112
  - news cache 189
- scalable multicast
  - defined 122
  - support by NetCache 122
- Secure Sockets Layer
  - defined 289
  - no ICAP services with transactions 219
  - supported by NetCache 2
- SecureAdmin feature 289
- security
  - firewall and NetCache authentication 232
  - ICAP 219
    - requirements 219
    - risks 219
  - NetCache feature described 10
  - SecureAdmin feature 289
- Server Load Balancer. *See* SLB
- service groups. *See* WCCP service groups
- single-mode virtual interface 80
- SiteMinder 9
- SLB
  - advantages 67

- defined 289
- disadvantages 67
- how to deploy 66
- using to avoid browser reconfiguration 58
- SmartFilter
  - NetCache feature described 9, 289
  - speed of response, improving with caching 3
  - splitting, defined 289
- SSL. *See* Secure Sockets Layer
- stealth mode, with transparent proxying 23
- stream splitting by NetCache 117, 125, 289
- stream upgrade requests 119
- streaming accelerator
  - advantages of 169
  - assigning multiple IP addresses to 182
  - client access to 172
    - DNS 172, 173
    - transparent proxying 172
  - described 8, 168, 290
  - hit rate 170
  - L4 or L7 switch with
    - optimizing 172
  - NetCache takeover pairs with 180
  - overview 168
  - request distribution
    - switch 32
    - WCCP router 40
  - scenarios
    - limited access from a partner 183
    - multiple 176
    - multiple accelerators 179
    - multiple streaming servers 181
    - single accelerator outside firewall 174
  - WCCP router with 172
    - optimizing 172
  - what it caches 170
- streaming media 99
  - audio, bandwidth use 104
  - bandwidth
    - contrasted with Web objects 127
    - duration of connections 127
    - savings 112
    - use, between cache and server 127
    - use, between clients and cache 128
    - use, defining goals 132
    - use, effect on quality 104
  - benefits of NetCache for 112
  - bit rates
    - advantages of multiple 102
    - compared to delivery bit rate 102
    - encoding in a media stream 102
  - compared to Web files 101
  - content pushing
    - quality improvement 128, 140
  - defined 101, 290
  - delivery methods 101
    - supported by NetCache 101
  - encoded at different bit rates 102
  - handling in typical client-server model 104
  - live media streams 101
    - how NetCache handles multicast 120
    - how NetCache handles unicast 115
    - how NetCache handles, overview 114
  - NetCache connection to origin server 114
  - NetCache delivery to clients 114
  - network conditions 102
  - number of NetCache appliances for 132
  - on-demand media streams 101
  - port redirection 25, 133
  - prefilling the cache 155, 278
  - presentations, defined 101, 290
  - pushing content, benefits 278
  - quality 128, 278
    - effect of closing firewall ports 137
  - stream splitting by NetCache 117, 125
  - stream upgrade requests 119
  - streams with multiple bit rates 102
  - thinning, defined 102
  - through firewalls 137
  - transmission methods 106
  - unicast versus multicast 106
  - unique stream
    - characteristics of 101
    - defined 102
    - how NetCache handles 114
- streaming media cache
  - as a streaming accelerator also 170
  - authentication 113
  - benefits of 112
  - client access to

- transparent proxying 25, 133
- Windows Media metafile rewriting 60, 134, 150
- connections
  - described 127
  - effect of cache failure on 27
- dedicate NetCache appliance for 132
- defined 7, 290
- deployment considerations 126
  - scalability 112
- deployment example 35, 45
- deployment scenarios 141
  - enterprise 146, 155
  - ISP 142
  - multicast for CDN 163
  - multicast in an enterprise 157
  - multicast over satellite link 160
  - with Windows Media server 150
- eliminate need for new streaming server 112
- failover
  - strategies for 135
  - streams being delivered 135
- firewalls, NetCache configuration for 139
- how many you need 132
- license for feature 111
- live media streams 124
  - path NetCache takes 116, 123
- logging connection data 113
- L4 switch with 25
- media players
  - set to HTTP only 138
  - supported 112
- MMS
  - directing streams to 133
  - NetCache support for 111
- multicast
  - settings by client activity 131
  - settings by client location 131
  - when NetCache uses 120
- NetCache configured as 7, 99, 111
- on-demand media streams
  - path NetCache takes 123
- providing access to 133
- pushing content, benefits 155, 278
- QuickTime, NetCache support for 111

- redirect HTTP traffic to 133, 138
- role with authentication 113
- RTSP
  - directing streams to 133
  - NetCache support for 111
  - RealSystem supported 111
- TCP and quality 137
- TCP support 117
- transparent proxying with 25, 133
- UDP and quality 137
- UDP support 117
- unicast
  - transports for 117
  - when NetCache uses 115
- WCCP router with
  - load balancing over NetCache interfaces 42
- streaming server
  - failover to, when it is possible 135
- stream, defined 289
- stream, unique, defined 291
- switch. *See* L7 switch, L4 switch

## T

- takeover (NetCache)
  - connection recommendation 79
  - described 78
  - networks feature supports 79
  - not applicable to news cache 78
  - takeover pairs 78
- TCP
  - and streaming media quality 137
  - connections
    - optimizing over satellite link 96
  - large windows
    - using to achieve higher throughput 97
  - support for streaming media 117
  - transmission, optimizing 95
- thinning, defined 102, 290
- throughput, increasing over satellite links 95
- Time To Live
  - what happens when it expires 179
- traffic loops
  - avoiding when using an L4 or L7 switch 29

- transmission methods
    - streaming media 106
      - multicast 106
      - unicast 106, 107
  - transparent firewall
    - defined 227, 291
    - NetCache deployed with 227
  - transparent proxying
    - access to
      - accelerators 172
      - DNS cache 25
      - news cache 25
      - streaming media cache 133
      - Web cache 25
    - caution for electronic commerce sites 24
    - defined 21, 291
    - devices to deploy with 22
    - drawbacks 23
    - error messages 23
    - examples of deployments 43
      - combination Web and news cache 47
      - multiple NetCache appliances 47
      - news cache 47
      - one NetCache appliance 43
      - POP and Data Center 48, 75, 93
      - WCCP routers 43
    - failover 26
      - how it works 26
    - IP spoofing, benefits 24
    - load balancing
      - over NetCache network interfaces 42
    - L4 switch. *See also* L4 or L7 switch
    - L7 switch. *See also* L4 or L7 switch
    - overview 21
    - policy-based routers
      - methods 22
      - performance impact 22
    - request distribution 30, 39
    - requirements for 271
    - stealth mode 23
    - traffic loops 29
    - versus GRM 244, 247
    - WCCP routers 22
  - Transport layer, and L4 switches 276
  - TTL. *See* Time To Live
  - tunnel (SSL), supported by NetCache 2
  - Typhoon software, supported by NetCache 193
- ## U
- UDP
    - and streaming media quality 137
    - defined 291
    - support for streaming media 117
  - uncacheable objects
    - fetching through a hierarchy 72
    - L7 switch with 32
  - unicast
    - defined 291
    - described 106
    - features of 106
    - how NetCache handles 115
    - NetCache conversion to multicast 120, 164
    - transports for 117
    - when NetCache uses 115
  - Uniform Resource Locator. *See* URL
  - unique stream
    - defined 101, 102, 291
    - how NetCache handles live 114
  - upgrade requests for streaming media 119
  - URL, defined 292
- ## V
- value-added services. *See* ICAP
  - vectoring points, ICAP 208
  - video on-demand, defined 292
  - virtual interfaces, single-mode 80
  - virus scanning
    - ICAP scenario with 222
    - ICAP service described 205
- ## W
- WCCP
    - defined 22
    - IOS images it runs with 22
    - IP spoofing for 38
    - protocol available for 22
    - version NetCache supports 22
  - WCCP routers 40

- connection to NetCache appliance 33
- defined 292
- deploying transparent proxying with 22
- deployment examples 35, 43
- failover 26
- load balancing over NetCache interfaces 42
- location in relation to clients 33
- NetCache load redirection 41
- number in a service group 36
- port redirection 25, 133
- redirecting traffic to NetCache 25
- request distribution
  - IP address hashing 40
  - See also* WCCP service groups
  - streaming media cache
    - traffic to redirect to 133
  - weighting distribution 41
  - with accelerators 172
- WCCP service groups
  - defined 292
  - effect on request distribution 39
  - number of proxy-cache servers in 36
  - number of routers in 36
  - planning for 37
- Web accelerator
  - advantages of 169
  - assigning multiple IP addresses to 182
  - client access to 172
    - DNS 172, 173
    - transparent proxying 172
  - defined 292
  - described 7, 168
  - hit rate 170
  - IPv6 deployment 261
  - L4 or L7 switch with 172
  - multiple Web servers, single accelerator 181
  - naming the Web accelerator 174
  - naming the Web server 174
  - NetCache takeover pairs with 180
  - overview 168
  - request distribution
    - switch 32
    - WCCP router 40
  - scenarios
    - distributing requests over 176, 178
    - historical performance site 185
    - limited access from a partner 183
    - multiple NetCache appliances, single Web server 176, 179
    - single accelerator outside firewall 174
    - WCCP router with 172
- Web browsers
  - types supported 2
  - use with NetCache 2
- Web cache 83
  - advantages of 85
  - as a Web accelerator also 170
  - bandwidth savings 85
  - blocking access to Web sites 85
  - defined 293
  - deployment considerations 86
    - client access to 86
    - how many you need 86
    - when behind a firewall 87
  - deployment scenarios 89, 91, 93, 95
  - L4 switch with 25
  - L7 switch for 85
  - optimizing Web site for caching 87
  - protocols it handles 84
  - quality of service 85
  - traffic
    - how a switch handles 25
    - how a WCCP router handles 25
    - what is cached 85
- Web Cache Control Protocol. *See* WCCP
- WebWasher DynaBLocator
  - NetCache feature described 9
- Windows Media metafile rewriting 60, 134
- Windows Media Player 7
  - interaction with NetCache 60, 134
- Windows Media server
  - deployment example
    - streaming media service 150
  - nontransparent client access method for 60, 150