

Content Server - Reverse Proxy Server Resource Guide
10g Release 3 (10.1.3.3.0)

March 2007

Content Server - Reverse Proxy Server Resource Guide, 10g Release 3 (10.1.3.3.0)
Copyright © 2007, Oracle. All rights reserved.

Contributing Authors: Sandra Christiansen

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

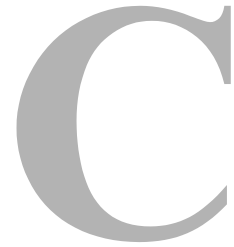
U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Table of Contents



Chapter 1: Introduction

About Content Delivery	1-1
Dynamic Delivery of Content	1-2
Static Delivery of Content	1-2
Dynamic and Static Delivery	1-2
About Proxy Caching	1-3
Advantages and Disadvantages	1-4
About Hierarchical Cache Systems	1-5
Reverse Proxy Considerations	1-6
Using the 8.3 Naming Convention	1-6
Defining Port Assignments	1-6
Site Studio and Firewall Considerations	1-7
About This Guide	1-7
Audience	1-7
Conventions	1-7
Document Organization	1-8
Additional Documentation	1-9

Chapter 2: Using Squid

Obtaining and Installing Squid	2-1
Installing Squid on UNIX	2-1
Installing Squid on Microsoft Windows	2-2
Configuration Options for Squid	2-3
Basic Configuration Options	2-3
Configuration for Dynamic Server Pages	2-3
Configuring for Reverse Proxy	2-4
Purging Objects from the Squid Cache	2-5
Sample squid.conf file	2-5
Starting Squid	2-6

- Verifying and Tuning System Performance 2-6
 - Testing Squid 2-6
 - Setting Refresh Properties 2-7
- Disabling Proxy Caching 2-7

Chapter 3: Using Microsoft ISA Server

- Pre-Installation Considerations 3-1
 - Multi-Networking 3-1
 - Terms and Concepts 3-2
 - Caching 3-3
 - Users and Privileges 3-3
- Obtaining Microsoft ISA 3-4
- Installing Microsoft ISA 3-4
- Verifying and Tuning System Performance 3-4
 - Verifying System Performance 3-5
 - Setting the Refresh Rate 3-5
- Disabling Caching 3-5

Chapter 4: Using Sun Java System

- Obtaining Sun Java System 4-1
- Installing Sun Java System 4-2
 - Installing on Microsoft Systems 4-2
 - Installing on UNIX Systems 4-3
- Installing Additional Proxy Servers 4-4
- Setting Up a Reverse Proxy Server 4-5
 - Setting Up A Reverse Proxy 4-5
- Additional Notes 4-6
- Verifying and Tuning System Performance 4-7
 - Verifying System Performance 4-7
 - Monitoring with Log Files 4-7
 - Other Monitoring Methods 4-7
 - Setting the Refresh Rate 4-8
- Disabling Caching 4-8

Chapter 5: Proxy Considerations with Site Studio

- Site Studio Design Considerations 5-1
- Configuring Site Studio 5-1

Site Studio Designer Changes	5-1
Changes for Versions Prior to 7.5	5-2
Content Server Changes	5-7
Host Web Server Changes	5-7
Configuring the ssBaseServerAddress Variable	5-8
Older Versions of Site Studio (pre-7.2.1)	5-8

Chapter 6: Tips and Tricks

Squid	6-1
Squid and Site Studio	6-1
Setting Appropriate Directory Permissions	6-1
Setting File Descriptor Limits	6-2
Using Swap Directories	6-2
Querying Other Cache Servers	6-2
Controlling Which Pages are Cached	6-3
Microsoft ISA	6-3
Using Windows Media Series	6-3
Network Adapters	6-3
Caching Considerations	6-3
Setting Cache Parameters	6-4
Sun Java System Web Proxy Server	6-4
Up-To-Date Checks	6-4
Proxy Timeouts and CGI Scripts	6-5
Caching Pages Requiring User Authentication	6-5

Appendix A: Third Party Licenses

Apache Software License	A-1
W3C® Software Notice and License	A-1
Zlib License	A-3
General BSD License	A-4
General MIT License	A-4
Unicode License	A-5
Miscellaneous Attributions	A-7

Glossary

Index

INTRODUCTION

OVERVIEW

This document discusses the use of different web proxy server software products that can be used to improve web performance by caching or by providing controlled web access to applications and sites behind a firewall.

The following products are discussed in this guide:

- ❖ Squid 2.5, an open source product which can be used on UNIX and Microsoft Windows platforms (NT 4.0, 2000, XP and 2003).
- ❖ Microsoft Internet Security and Acceleration (ISA) Server, which can be used with Microsoft Windows products.
- ❖ Sun Java System Web Proxy Server, which can be used with UNIX, Solaris and Microsoft Windows NT or Windows 2000 server.

This document does not provide detailed instructions for use of these products but rather provides an overview of product interaction with Content Server and related products. It also provides recommendations for best practices of use. For details about the use of these products, see [Additional Documentation](#) (page -9).

This chapter discusses the following topics:

- ❖ [About Content Delivery](#) (page -1)
- ❖ [About Proxy Caching](#) (page -3)
- ❖ [About Hierarchical Cache Systems](#) (page -5)
- ❖ [Reverse Proxy Considerations](#) (page -6)

- ❖ [About This Guide](#) (page -7)
- ❖ [Additional Documentation](#) (page -9)

ABOUT CONTENT DELIVERY

There are three standard approaches to content delivery: dynamic delivery, static delivery, and a combination of both (reverse proxy). The section describes those three approaches.

Dynamic Delivery of Content

Dynamic delivery of content occurs when all content requests go directly to the Content Server and are delivered from a Content Server to the client.

In an environment of this type, a contribution server exists behind a firewall. A consumption server and web server exist outside that firewall and behind another firewall to the Internet. Content is mirrored or replicated from the contribution server to the consumption server. The consumption Content Server distributes content to the outside world.

This type of delivery allows you to aggregate content at one server and deliver from that site. The speed of delivery is somewhat slower because client requests are managed by the Content Server.

Static Delivery of Content

In a static delivery system, a replicated update package of content from a contribution server is generated on a schedule. The contribution server is kept behind a firewall and the package is deployed to a file system and web server behind another firewall to the Internet. Clients accessing content access the scheduled copy that is kept at the web server. This content does not change as it is updated, but instead changes statically -- on a scheduled basis.

This type of delivery provides faster performance and “friendly URLs” (URLs which do not contain question marks). Using this scenario, people work dynamically but the content is delivered statically.

Dynamic and Static Delivery

A reverse proxy server combines dynamic and static delivery of content items. Like a dynamic system, the contribution server is updated continually and is replicated to a consumption site with immediate updates. The proxy cache acts as a static site, filling requests from clients as though it was the consumption server. The web pages that are served contain headers that dictate how long the page should be kept in the cache, when it should be refreshed, and whether or not it can be cached.

Contribution does not happen through the proxy server; it only delivers the content. You can configure sections of a web site so they are not cached, thus enabling forms contribution. You can also set different sections of web pages to be cached at different rates, thus enabling a combination of dynamic and static delivery.

ABOUT PROXY CACHING

Proxy caching can provide several benefits to sites that have a large load of web processing requests.

A *forward proxy* acts as a gateway for a client's browser, sending HTTP requests on behalf of the client when requests come from the Internet. When a request is processed, the IP address of the proxy is used, rather than the client's actual address. This hides the IP address of the network from the outside world.

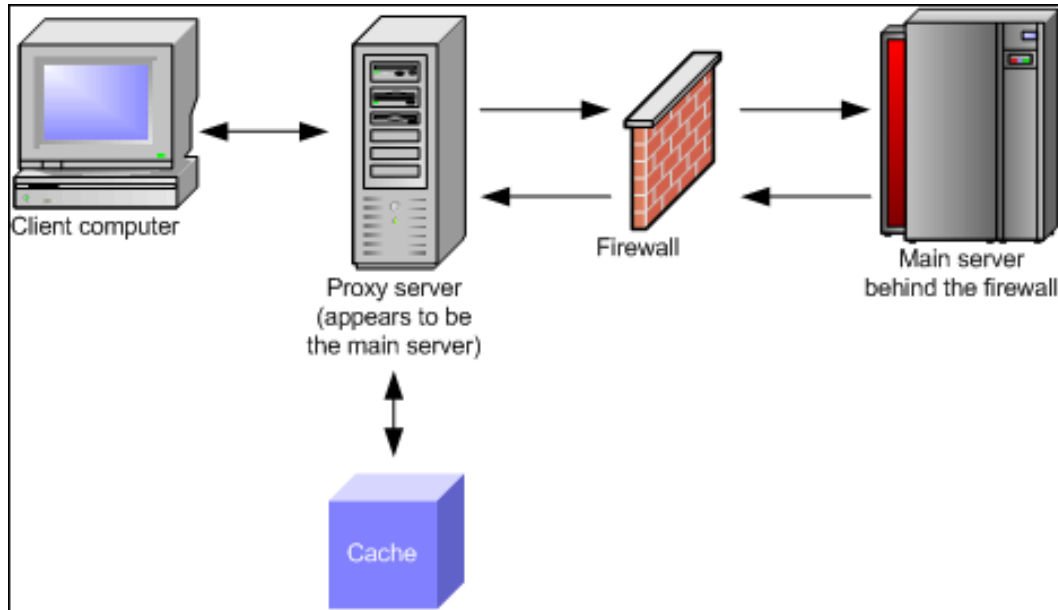
A *reverse proxy* server issues requests on behalf of the backend HTTP server, not on behalf of the client. Because of this, a client configuration is not needed. Clients access the server as if it were a regular web site. A reverse proxy server acts as a gateway to an HTTP server, and is the final IP address for requests from the outside.

In addition to this protective feature, a proxy cache stores documents close to the user, thus eliminating wait time for retrieval. The client web browser is configured to send all requests to the proxy server. The proxy server, located close to the client, caches the web content, thus providing faster access to common sites and pages. Caching proxy servers reduce traffic from the local site to the Internet, saving valuable bandwidth and reducing connection costs.

A reverse proxy server can also cache requests that it serves from the backend server. When a request is received for a page, the reverse proxy server forwards the request to the backend server and caches the page in addition to returning the page to the client. Subsequent requests to that page can be served from the cache, as long as the cache hasn't expired.

The following illustration shows how a reverse proxy server ‘hides’ the identity of the main server from those systems which are making requests.

Figure 1-1 Reverse proxy and main server identity



The firewall is configured to allow a specific server on a specific port to have access to the secure server. When a client makes a request, the request goes through the proxy server, which passes the request through the firewall to the secure server. A result is passed back through the firewall to the proxy.

If an error is returned, the proxy server intercepts the message and changes the URLs listed in the headers before sending the message on to the client. In this way, external clients do not receive redirection to URLs to the secure server.

Multiple proxy servers can be used for load-balancing by taking advantage of the caching features of the proxy server. If you have a web server that has active traffic, proxy servers can take some of the load from the web server, making network access more efficient. After an initial starting period (in which the proxy servers retrieve documents for the first time), the number of requests to the actual web server will drop as the proxy server cache is used instead.

Advantages and Disadvantages

A site can benefit in several ways by implementing a reverse proxy server:

- ❖ Avoid the expense of installing another web server. A reverse proxy server increases the capacity of existing servers.
- ❖ Serve more requests for static content and thus free up bandwidth to serve more dynamic content.
- ❖ Reduce operating expense by increasing bandwidth.
- ❖ Provide a single point of control over who can access HTTP servers, and which servers can be accessed.
- ❖ Decrease response time of web pages and accelerate download time, enhancing the experience of web site users.
- ❖ Provide another layer of protection by hiding the internal IP address.

Some disadvantages in using reverse proxies include the following:

- ❖ If the reverse proxy is compromised and a failover is not in place, the site's HTTP presence is shut down.
- ❖ If an outside attacker compromises the reverse proxy server, the attacker may also be able to get access into your HTTP server architecture. If the HTTP servers are inside your firewall, an attacker could compromise your internal network.
- ❖ A large amount of translations must be done between the reverse proxy and the firewall, so initial requests may be filled slightly slower.
- ❖ Cache content must have a single security access point and cannot be personalized.
- ❖ Because all HTTP traffic is going through a reverse proxy, content filtering must be added to ensure that HTTP requests are not actually attacks.

ABOUT HIERARCHICAL CACHE SYSTEMS

In a hierarchical proxy system, your proxy can communicate with other proxies to verify if the other proxy has the relevant requested page. The ability to act as a peer or sibling to another server can save valuable access time (by designating another machine at another site as a peer, your site can communicate directly with that site).

When querying more than one cache, all queries are sent out together without waiting for replies. The client request is put on hold until the first positive reply from a sibling is

received. The proxy server will then retrieve the object from the fastest-replying cache server. Because the earlier returning reply packet is usually the fastest link, the page is returned quickly.

REVERSE PROXY CONSIDERATIONS

Before setting up a reverse proxy, you should be aware of some planning considerations that should be addressed.

Using the 8.3 Naming Convention

You should set up a reverse proxy server first in a test environment and ensure that it is working as planned. When you are certain it is performing as expected, you can then move it to a production environment.

If you are running the Content Server on a dedicated system, the 8.3 naming convention may have been disabled at installation. Disabling the naming convention ensures that long file names in the vault directory and the weblayout directory are not compressed using a tilde (~).

If the system has been installed with the 8.3 naming convention option disabled, the Sun Java System Reverse Proxy server will not install properly and cannot be used. When the 8.3 convention is disabled, a registry key is set and it must be manually changed; it cannot be enabled through a new installation. After the key is changed, you will once again be given the option to disable the 8.3 naming convention during a Content Server installation.

To re-set the toggle, use the Registry Editor on Microsoft Windows systems to delete the following registry key (or set it to 0):

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem\NtfsDisable8dot3NameCreation
```

After the key has been reset, reboot your system and reinstall the Content Server. You are prompted again to decide whether to disable the naming convention or not. In order to use a Proxy Server, choose **No** when prompted.

Defining Port Assignments

If you use the Site Studio Publishing Utility, several default port numbers are used by the utility and should not be used by the reverse proxy server. See the Site Studio documentation for a complete list of default port numbers.

Site Studio and Firewall Considerations

The Site Studio Publishing Utility administration server normally communicates only internally, not across a firewall. If both the Publishing Utility and the Subscription Client are behind a firewall, the ports can remain closed. If they are separated by a firewall, ports 8890 and 8891 must be opened to allow the Subscription Client to communicate with the Publishing Utility.



Important: If you are already running a proxy server, you must install the new proxy server to a different port. After the new proxy server is installed and configured, shut down the old proxy server and change the new server to use the appropriate port.

ABOUT THIS GUIDE


This section discusses the audience for this guide, the conventions used and the organization of this guide.





Audience

This guide is intended for use by high-level system administrators who are familiar with network configurations, filtering, and firewalls.

Conventions

- ❖ The notation `<install_dir>/` is used to refer to the location on your system where Content Server is installed.
- ❖ Forward slashes (/) are used to separate the directory levels in a path name. This is true when referring to files on a Windows file system or on a UNIX system. A forward slash will always appear after the end of a directory name.
- ❖ Notes, technical tips, important notices, and cautions use these conventions:

Symbol	Description
	Note: Brings special attention to information.

Symbol	Description
	Tech Tip: Identifies information that can be used to make your tasks easier.
	Important: Identifies a required step or required information.
	Caution: Identifies information that might cause loss of data or serious system problems.
	Identifies information that is new to this version of the software.

Document Organization

This document is organized into the following chapters:

- ❖ [Chapter 1 \(Introduction\)](#), which describes proxy server concepts and provides an overview of the documentation.
- ❖ [Chapter 2 \(Using Squid\)](#), which describes how to use the open source Squid product for reverse proxy configuration.
- ❖ [Chapter 3 \(Using Microsoft ISA Server\)](#), which describes how to use the Microsoft Internet Security and Acceleration software.
- ❖ [Chapter 4 \(Using Sun Java System\)](#), which describes how to install and use the Sun Java System Web Proxy Server.
- ❖ [Chapter 5 \(Proxy Considerations with Site Studio\)](#), which describes some configuration options that can be used with Site Studio.
- ❖ [Chapter 6 \(Tips and Tricks\)](#), which provides information about shortcuts and other issues when using reverse proxy servers.

In addition, a [Glossary](#) (page -1) is available which explains the technical terms used in this documentation.

ADDITIONAL DOCUMENTATION



Note: This document may contain URLs which were valid when originally published, but may link to sites or pages that no longer exist.

- ❖ Squid is a well-known open source product and as such there are many third-party books available to document its use. Links to documentation can be found at <http://www.squid-cache.org/Doc/>.
- ❖ Microsoft Internet Security and Acceleration (ISA) product documentation can be found at <http://www.microsoft.com/isaserver/techinfo/books/default.msp>. In addition, the [isaserver.org](http://www.isaserver.org) and the <http://www.microsoft.com/isaserver/community> web sites contain many useful articles, FAQs, and links to information about configuring and using the product.
- ❖ Sun Java System Web Proxy Server documentation can be found at <http://docs.sun.com/app/docs>. In addition, the <http://swforum.sun.com> web site contains user forums where installations such as reverse proxies are discussed.

The scope of this document focuses on the interaction of these products with the Content Server and its associated products such as Site Studio. It does not provide complete details about installation and use of the products themselves.

USING SQUID

OVERVIEW

This chapter provides an overview of usage for the Squid open source proxy software. It covers the following topics:

- ❖ [Obtaining and Installing Squid](#) (page -1)
- ❖ [Configuration Options for Squid](#) (page -3)
- ❖ [Sample squid.conf file](#) (page -5)
- ❖ [Starting Squid](#) (page -6)
- ❖ [Verifying and Tuning System Performance](#) (page -6)
- ❖ [Disabling Proxy Caching](#) (page -7)

OBTAINING AND INSTALLING SQUID



Note: Depending on your system configuration, you may need to tune your system before you install Squid. See [Setting File Descriptor Limits](#) (page -2) for more information about tuning options.

Squid is open source software that is available to download from the Internet. This section discusses where to find the Squid downloadable software and how to install it, depending on the type of operating system used (UNIX or Windows).

- ❖ A UNIX tar file of pre-compiled binaries is available at <http://www.squid-cache.org>. See [Installing Squid on UNIX](#) (page -1) for details about untarring the file and installing the software.
- ❖ The Windows binary is not available at the Squid web site. A Windows NT/2000/XP version can be downloaded from <http://www.acmeconsulting.it/SquidNT.htm>. See [Installing Squid on Microsoft Windows](#) (page -2) for details about Windows installation.

Installing Squid on UNIX

You can download the Squid source archive as a gzipped tar file and then untar the file using the following commands:

```
% tar -xvzf squid-*.src.tar.gz
% cd squid-*
```

Next, enter the commands that are need to configure, compile, and install Squid:

```
% ./configure configuration options
% make
% make install
```

The `configure` command with the specified options determines the configuration of the Squid files. See [Configuration Options for Squid](#) (page -3) for recommendations about configuring Squid for your site by changing the options in the `squid.conf` file.

The `make` and `make install` commands compile and install the binaries. Squid is installed by default into the `/usr/local/squid` directory (the working directory).

To install Squid on a dedicated cache machine, you may want to put all Squid files into the `/usr/local` directory. Use the `prefix` option to the `configure` command:

```
% ./configure -prefix=/usr/local
```

This changes the default location for the installation.

To view all available configuration options, use the `-help` argument with the `configure` command:

```
% ./configure -help
```

See the *Squid User Guide* at http://www.deckle.co.za/squid-users-guide/Main_Page for details about configuration options that can be used at installation. See the *Squid Configuration Manual* at <http://www.visolve.com/squid/index.php> for a description of all configuration options.

Installing Squid on Microsoft Windows



Important: If you are upgrading from SquidNT 2.3, uninstall the old service and merge the old squid.conf file with the new squid.conf configuration file.

The Windows binary is available as a gzipped file from <http://www.acmeconsulting.it/SquidNT/>. You need to download and extract the Squid files into a directory on your Windows system. Configuration changes can be kept to a minimum if the binary is extracted to the C: drive root directory.

After it is extracted, a Squid directory is set up with subdirectories for the various Squid files.

See [Configuration Options for Squid](#) (page -3) for recommendations about changing the options specified in the squid.conf file to set up Squid for your site.

CONFIGURATION OPTIONS FOR SQUID

While there are several different configuration options that can be used to configure Squid, specific options need to be set to use Squid for basic proxy use or for reverse proxy operations.

Basic Configuration Options

After installation, two default configuration files are stored in the default Squid location (/usr/local/squid/etc on UNIX systems or c:\squid\etc on Windows systems). These files, squid.conf.default and mime.conf.default, are used to set configuration options for Squid.

Copy those files, and rename the copied files to squid.conf and mime.conf. Make all edits to these renamed files.

Many of the configuration options in squid.conf are originally commented out. You need to remove the comment designation (#) to use those configurations.

Uncomment and edit the following lines in the squid.conf file to set up Squid for use as a reverse proxy:

- ❖ `cache_dir`: set the cache directory to a location that has a large amount of hard disk space to be used for caching.
- ❖ `http_port`: set this to the port number where Squid will listen. The default is 3128.

- ❖ `http_access`: this determines the Access Control List (ACL) control of proxy access. Rules can be set to allow or deny access from specific sets of users or you can set this to deny all or allow all, as in the following example:

```
http_access allow local host
http_access deny all
```

- ❖ `cache_effective_user` and `cache_effective_group`: set these options to a user and group who will have access to the cache. It is recommended that you create a Squid user and a Squid group; you can use those user IDs here. The user should have the permission to read and write in the cache directory and in the log files. See [Setting Appropriate Directory Permissions](#) (page -1) for more details.

Configuration for Dynamic Server Pages

Additional configuration options should be changed to make certain that Squid will cache dynamic pages (such as those generated by Site Studio) properly.

- ❖ Comment out or modify the following lines that are set by default:

```
acl QUERY urlpath_regex cgi-bin\?
no_cache deny QUERY
```

These lines in the `squid.conf` file disable caching of any URLs containing `cgi-bin` or `?` in their path. This would disable caching for any Site Studio pages or other dynamic pages that receive URL parameters. Comment out these lines to enable caching. Make similar changes for any Content Server `cgi` query prefixes.

- ❖ To disable caching for POST requests, add the following lines to the `squid.conf` file:

```
acl METHODLIST method POST
no_cache deny METHODLIST
```

- ❖ To log the `QUERY_STRING` terms passed in the URL, add the following line in the `squid.conf` file (this is set to on by default):

```
strip_query_terms off
```

Logging the query strings allows you to debug the caching and to verify that caching is working properly.

Configuring for Reverse Proxy

Squid is configured in proxy mode by default. To run Squid as an accelerator (reverse proxy), you have to define specific parameters:

- ❖ `http_port`: It is common practice to set your port to a low number. Example:

```
http_port 80
```

- ❖ `httpd_accel_host`: this is the name or the IP address of the web server to be proxied.

For example:

```
httpd_accel_host mysite.com
```

If you are using Squid as an accelerator for a virtual host system, do not use a host name. Instead use the keyword `virtual`, as in this example:

```
httpd_accel_host virtual
```

- ❖ `httpd_accel_port`: this is the port number of the web server to be proxied. The default is 80. Example:

```
httpd_accel_port 80
```

- ❖ `httpd_accel_single_host`: if Squid is being run as a reverse proxy and the site has a single backend server, this should be set to `on`. This causes Squid to forward the request to this server regardless of what is specified by re-directors or host headers. The default is `off`.

```
httpd_accel_single_host on
```

Purging Objects from the Squid Cache

It is sometimes useful to force a refresh of the squid cache when a specific content item (for example, an urgent press release) needs to be published regardless of the cache age within the reverse proxy.

Squid does not allow you to purge objects unless it is configured with access controls in the `squid.conf` file. To purge a request which comes from the local host and deny all other purge requests, add the following lines to the `.conf` file:

```
acl PURGE method PURGE
acl localhost src 127.0.0.1
http_access allow PURGE localhost
http_access deny PURGE
```

To purge an object, use the `squidclient` program, which is a command line interface to retrieve URLs through the squid object cache:

```
squidclient -m PURGE <URL>
```

If the purge was successful, a “200 OK” response is returned. If the object was not found, a “404 Not Found” response is returned.

See <http://www.penguin-soft.com/penguin/man/1/squidclient.html> for details on usage of the `squidclient` program.

SAMPLE SQUID.CONF FILE

The following is a sample squid.conf file, showing entries discussed in this chapter:

```
httpd_accel_host stellent.com
httpd_accel_port 80
httpd_accel_single_host on
httpd_accel_uses_host_header off
http_access allow all
#
#the following 2 lines are commented out on purpose
#
#acl QUERY urlpath_regex cgi-bin\?
#no_cache deny QUERY
#
#these lines replace the previous ones
#
acl METHODLIST method POST
no_cache deny METHODLIST
strip_query_terms off
```

STARTING SQUID

Before using Squid with Site Studio, you need to make a few further configuration changes. See [Chapter 5 \(Proxy Considerations with Site Studio\)](#) for details.

After installing and configuring Squid and Site Studio, you should start the program with the `-z` option to create the necessary swap files.

On a UNIX command line:

```
% /usr/local/squid/bin/squid -z
```

In a Windows command prompt window:

```
.\squid -z
```

See [Using Swap Directories](#) (page -2) for more details about swap directory use.

After swap directory creation completes, you can start Squid by typing the above commands with no additional options.

VERIFYING AND TUNING SYSTEM PERFORMANCE

Several different methods can be used to test that Squid is running properly and to tune its performance. This section describes those procedures.

Testing Squid

Testing the caching performance can be difficult because pressing **Reload** or **Refresh** in a browser doesn't re-fetch the page, it forces the cache not to serve the cached page. Therefore the server has to download the page again from the original server.

To test the cache, set up two machines that can access the cache. Choose a site off the local network, and access it from the first machine. After a page has downloaded, change to the second machine and download the page.

After the page has downloaded there, check the access.log file, which logs all incoming requests. Check that the second download was marked as a HIT in the access.log. If the entry was marked as a MISS, it is probably because the origin server asked Squid to not cache the page. You can also check the cache.log file for details about caching activity.

Setting Refresh Properties

The `refresh_pattern` configuration entry can be used to set the refresh pattern for Squid. This entry specifies the time an object without an explicit expiry time should be considered fresh.

This configuration entry has the following form:

```
refresh_pattern regex min percent max options
```

This entry can be added to the `squid.conf` file. The `min` option specifies the time (in minutes) an object should be considered fresh. The `percent` option is a percentage of the object's age (the time since last modification). The `max` option is an upper limit for how long objects without explicit expiry time are considered fresh.

DISABLING PROXY CACHING

On Windows systems, choose the Squid Service name then click **Stop** from the Services panel to stop the Squid service.

On UNIX systems, use the following command:

Using Squid

```
% squid -n servicename | shutdown
```


USING MICROSOFT ISA SERVER

OVERVIEW

This chapter provides an overview of the installation and configuration of Microsoft Internet Security and Acceleration (ISA) Server. See <http://www.microsoft.com/isaserver/default.asp> for product details.

This chapter covers the following topics:

- ❖ [Pre-Installation Considerations](#) (page -1)
- ❖ [Obtaining Microsoft ISA](#) (page -4)
- ❖ [Installing Microsoft ISA](#) (page -4)
- ❖ [Verifying and Tuning System Performance](#) (page -4)
- ❖ [Disabling Caching](#) (page -5)

See [Chapter 5 \(Proxy Considerations with Site Studio\)](#) for details about configuring Site Studio to work with a reverse proxy server.



Note: This chapter describes using the Windows Internet Security and Acceleration Server with reverse proxy. For information about using the Windows Media Server with reverse proxy, see [Chapter 6 \(Tips and Tricks\)](#).

PRE-INSTALLATION CONSIDERATIONS

The Microsoft ISA product contains not only proxy software but firewall and network configuration software as well. This section provides an overview of the different parts of the ISA software and how they interact with a reverse proxy server.



Important: Export your current configuration and save it in case you want to quickly revert to a previous network configuration.

Multi-Networking

ISA configurations use a concept of multi-networking, in which internal servers work with not only external networks from other domains, but also work with users accessing the network with mobile computers (thereby appearing to be part of an “external” network). With the multi-networking features of ISA, you can limit communication between clients, even those within your own organization.

You can define relationships between these networks and determine how computers in each network communicate with each other. You can also group computers into ISA server network objects and configure access policies to each network object. Each network is isolated from the others and is made accessible when you configure rules to allow communication.

Terms and Concepts

The following terms are used when configuring an ISA server:

- ❖ *network*: a rule element that contains one or more ranges of IP addresses and domains.
- ❖ *network objects*: after they are created, networks can be grouped into sets such as subnets, URL sets, or domain name sets. Rules can be applied to networks or network objects.
- ❖ *network rules*: network rules can be configured to define and describe a network topography. Rules determine if connectivity is allowed and what type of connectivity is allowed. A network can be connected in one of two ways:
 - *Network Address Translation (NAT)*. With this type of connectivity, the IP address of the client on the source network is replaced by the ISA Server’s address.
 - *Route*: With this type of connectivity, the client requests from the source network are directly relayed to the destination network.

ISA Server is preconfigured with the following networks:

- ❖ external: including all computers that are not associated with any other internal network. This network cannot be deleted.
- ❖ internal: includes all computers associated with the internal network address card on the ISA server computer.
- ❖ local host: a network representing the ISA server computer. This cannot be modified or deleted.
- ❖ quarantined VPN clients: a network containing the addresses of VPN clients that are not yet approved to access the corporate network. These computers are usually allowed limited access to the corporate network.
- ❖ VPN clients: a network containing the addresses of VPN clients that are currently connected. This network cannot be deleted.

On installation, the following rules are created:

- ❖ Local host access: a rule defining the route relationship between the Local Host network and all other networks.
- ❖ VPN Clients to Internal Network: a rule defining the relationship between the two VPN networks (Clients and Quarantined) and the Internal network.
- ❖ Internet access: a rule defining a NAT relationship between the Internal and the External network.

Caching

ISA server supports forward caching for outgoing requests and reverse caching for incoming requests. It includes a Web Proxy filter, which handles the caching function.

When caching is enabled, you can set *cache rules*. These rules let you specify if content should be stored or retrieved from the ISA server cache. Rules can be applied to content from all sites or to specified sites, and to all content groups or limited to specific content types. In addition, you can limit the amount of time that objects are considered valid, and the way the cache handles expired objects.

You can set an object's Time To Live (TTL) in the HTTP or FTP cache property, or in the object itself. Depending on the rule's configuration property, the ISA server will retrieve the object in one of the following ways:

- ❖ retrieve an object only if the object is still valid.
- ❖ retrieve the object regardless of whether it is still valid.

- ❖ never route the request.

When ISA Server is initially installed, the default cache rule specifies that only valid requested objects are retrieved from the cache.



Note: If you install ISA server in Firewall mode, the server does not maintain a cache.

Users and Privileges

Administrator roles define the privileges on levels of ISA server management. Some users are set up as administrators of an entire enterprise while others may administer specific arrays. You should determine beforehand how these roles will interact with the roles you've established in the Content Server.

There are three types of ISA administrative users:

- ❖ **Monitoring:** users who can perform basic monitoring tasks.
- ❖ **Extended monitoring:** users who can perform monitoring tasks and configuration related to monitoring.
- ❖ **Full administration:** users who can perform all monitoring and administration tasks.

See the Microsoft ISA Server documentation for complete details about setting user and administrative privileges.

OBTAINING MICROSOFT ISA

The Microsoft Internet Security and Accelerator is available for purchase and download from the Microsoft web site at <http://www.microsoft.com/isaserver>. Depending on the size of your site, you will purchase either the enterprise edition or standard edition.

INSTALLING MICROSOFT ISA

Microsoft installation is done through an installation wizard which will take you through the steps. Keep the following points in mind when doing an installation:

- ❖ Microsoft ISA server is a combination of products, including a firewall, virtual private network, and caching server. Because of this, you are not just configuring the system for reverse proxy, but you must also configure it first for other operations.

- ❖ You can use pre-configured templates that may be applicable for your network. These templates provide default rules and cache behavior, which you can later change as needed.

When you install ISA server, caching is disabled because no space has been specified to be used for caching. When you specify the space to be used on a drive for caching, caching is effectively enabled.

VERIFYING AND TUNING SYSTEM PERFORMANCE

You can use the programs provided with Microsoft Windows to monitor system performance.

Verifying System Performance

You can check the status of cache performance in the operating system Performance Monitor. To do this, click **Start — Run**. Enter **perfmon** in the **Open** dialog box.

Right click in the graph pane and select **Add Counters**. In the Performance Object drop-down list, select **ISA Server Cache**. Then select the counters to view from the list.

Cache counters are discussed in detail in the ISA Server online help.

Setting the Refresh Rate

The Time to Live (TTL) rate determines how long an object is kept in the cache. This is set in the object's properties or in the HTTP properties pane when you are setting rules for the network.

DISABLING CACHING

When you enable caching, both forward and reverse caching are enabled. To disable reverse caching for specific sites, create a cache rule that disables the caching.

USING SUN JAVA SYSTEM

OVERVIEW

The Sun Java System Web Proxy Server can run on UNIX operating systems as well as Microsoft NT, Microsoft Windows 2000, Linux and Solaris.

Complete documentation for the Sun products discussed in this chapter is available at <http://docs.sun.com/app/docs/prod/sl.webproxys#hic>. This chapter provides an overview of that information, but does not provide complete details about usage.

This chapter covers the following topics:

- ❖ [Obtaining Sun Java System](#) (page -1)
- ❖ [Installing Sun Java System](#) (page -2)
- ❖ [Installing Additional Proxy Servers](#) (page -4)
- ❖ [Setting Up a Reverse Proxy Server](#) (page -5)
- ❖ [Additional Notes](#) (page -6)
- ❖ [Verifying and Tuning System Performance](#) (page -7)
- ❖ [Disabling Caching](#) (page -8)

See [Chapter 5 \(Proxy Considerations with Site Studio\)](#) for details about configuring Site Studio to work with a reverse proxy server.



Important: Changing the cache structure after installation and configuration requires that you reformat the cache structure and relocate files, which can be a time-consuming process. If you aren't sure what cache size to use, use 2GB as the default value in the installation process. This default value can hold more than 2GB of data if needed, and can be used with 3-5GB caches.

OBTAINING SUN JAVA SYSTEM

You can purchase and download the Sun Java System Web Proxy Server from http://www.sun.com/software/products/web_proxy/home_web_proxy.xml.

INSTALLING SUN JAVA SYSTEM

The Sun Java System can be installed on Microsoft operating systems and on UNIX-based systems. This section provides an overview of both installations.

Installing on Microsoft Systems

Change to the directory where the file was downloaded and double click the setup.exe file. The Installation Wizard takes you through the installation process. You will need to specify the following information when prompted:

- ❖ Destination directory: the directory where the proxy server files will be installed.
- ❖ Directory service: the service used to store user and group information. You can use either an LDAP-based service or the local directory service. If you choose an LDAP-based service, you must enter the directory server's name, ports, and its BDN (Base Distinguished Name).
- ❖ Administrator's username and password: these are the username and password used to access the admin server.
- ❖ Administration server port: the port address that the admin server will use.
- ❖ Cache partition root: the location of the proxy server's cache.
- ❖ Proxy server port: the port address that the proxy server will use. The standard proxy port is 8080.

At the end of the installation, you can access the Netscape Server Administration home page, where you can further customize the server settings. There are two main sections on this page: **General Administration** and **Servers Supporting General Administration**, also known as **Server Administration**.

The General Administration section is used to change user accounts, cluster management, and network settings.

Figure 4-2 General administration menu



To view or the configuration for the newly installed server, click the server name in the Server Administration section of the home page:

Figure 4-3 Server name



The Web Proxy Server main page is displayed. From this page you can change URL settings, caching information, and other configuration options.

Installing on UNIX Systems

Prior to installing the Sun Web Proxy Server software, you should set up a UNIX user account for the proxy server to run as that user. Because the server should have restricted access to the system resources, set up and run the proxy with a nonprivileged system user account. See your system manual for details on creating a new user account.

When the proxy server starts and runs, it runs with the UNIX user account that you specify during installation. It is strongly recommended that you use a dedicated user account for the proxy server.



Note: While it is possible to install your proxy server on the same machine as another Sun server, it is not recommended. Running two or more Sun servers on the same machine can cause a significant impact on the server's performance.

To install the software, log in as root or superuser and untar the distribution file. Type the ns-setup command to start the installation. See the installation guide which ships with the software for details about the installation process.



Note: When installing the software, you receive error messages if you don't have sufficient permission to the server root directory. If this happens, you have to either change the directory to install to, change your user permissions, or log in as root or superuser.

The installation process requires the following information:

- ❖ Server root: the directory where the proxy server will be installed.
- ❖ Machine name: the fully qualified host name of the machine where the proxy server is installed.
- ❖ SuiteSpot user: the UNIX user account that the proxy server will run as.
- ❖ Directory service: the service used to store user and group information. You can use either an LDAP-based service or the local directory service. If you choose an LDAP-based service, you must enter the directory server's name, ports, and its BDN (Base Distinguished Name).
- ❖ AdministrationServer User: the user that the administration server will run as.
- ❖ Server Administrator's username and password: these are the username and password used to access the admin server.
- ❖ Administration server port: the port address that the admin server will use.
- ❖ Web browser: the command-line name of the web browser used to configure the proxy server.

INSTALLING ADDITIONAL PROXY SERVERS

If you want to install another proxy server, click **Create New iPlanet Web Proxy Server** on the Server Administration main page. The installation page is displayed and you use it to set up the additional proxy system.

You will need to specify the following information. This information was filled in for you by default based on information you specified during the initial setup and download of the first proxy server.

- ❖ Host name where the proxy server is installed
- ❖ IP address (“bind address”) of the server.
- ❖ Server port where you want the proxy to listen to.
- ❖ A name used to identify the specific proxy server. If you are installing more than one proxy server on the computer, include a port number in the identifier (for example, myproxy-8080).

You will also need to decide on several configuration options such as the following:

- ❖ How IP addresses should be resolved.
- ❖ What kind of log format to use.
- ❖ What protocols the proxy should handle.
- ❖ What protocols will support SSL tunneling (HTTPS or SNEWS).
- ❖ Whether caching will be enabled and if so, what directory to use, the cache size, and other cache configuration options.
- ❖ A cache refresh policy and expiration policy

When you finish configuring the system, click **OK**.

SETTING UP A REVERSE PROXY SERVER

Setting up a Java Web Proxy Server as a reverse proxy server is done after the web proxy server is installed. Because a reverse proxy server can allow access to internal hosts (via a security breach) the firewall should be configured so it allows connections from the reverse proxy server to the main web servers and not to any other internal resources. It should allow only reverse proxy requests and remap them as follows:

- ❖ *a regular mapping* to redirect requests to the main server. This is also known as *request URL remapping*. The reverse proxy server must map the request URLs to URLs that point to the secure web server. The reverse proxy server only sends the path portion of the URL.

For example, if the full URL is the following:

```
http://machine.yoursite.com/path/file.html
```

the backend web server only receives `/path/file.html`. The proxy server uses the mapping to determine where to get the actual document.

- ❖ *a reverse mapping* used by the proxy server to trap redirects from the server. This is also known as *response header remapping*. The response headers may contain information that points to the main web server, such as `Location:` (which is used with redirections). A reverse proxy server remaps this reference from the main server to its own address.

Setting Up A Reverse Proxy

The following directions describe how to use the options on the Server Administration **URL** menu to create a regular proxy mapping and a reverse proxy mapping. The web server in this example is called `http://http.site.com`, and the reverse proxy server is `http://revproxy.site.com/`.

1. Click on **Create Mappings**.
2. To create a regular proxy mapping, select **regular**.
3. In the **Map source prefix** box, enter `http://revproxy.site.com`.
4. In the **Map destination** box, enter `http://http.site.com`.

Click **OK** when done then click **Save and Apply**.

5. Return to the form and create another new mapping for the reverse proxy. In the **Map source prefix** box, enter `http://http.site.com`. In the **Map destination** box, enter `http://revproxy.site.com`.

Click **OK** when done then click **Save and Apply**.

To view your mappings, click **View/Edit Mappings**.

ADDITIONAL NOTES

If the web server has DNS aliases, each alias should have a regular mapping. If the web server performs redirects to itself, each of those aliases should have a corresponding reverse mapping.

Authored content for the web server will also be served by the reverse proxy. Therefore, all links to files on the web server should use relative paths (for example, `/abc/def`) as opposed to fully qualified host names (`http://http`).

The proxy server never runs CGI applications; they always run on the origin server. However, if the CGI script indicates that the result can be cached (by implying a non-zero time-to-live, done by issuing a Last-modified or Expires header), the proxy caches the result.



Note: The information in this section is a summary of the instructions needed to install a reverse proxy server. See the Administration guides for UNIX and Windows systems at <http://docs.sun.com/app/docs/prod/s1.webproxys> for details.

In addition, see <http://www.sun.com/blueprints/1103/817-4402.pdf> for further details about adding a reverse proxy server and the appropriate configuration entries that must be made.

VERIFYING AND TUNING SYSTEM PERFORMANCE

You can use different methods to test that the proxy server is running properly and to tune its performance. This section describes those procedures.

Verifying System Performance

There are several ways to check the activity of the cache server, so you can verify that it is working as expected.

Monitoring with Log Files

On the Server Manager page, choose **Server Status**. Different log files are listed, which you can use to check system performance:

- ❖ **Error Log:** this log contains errors encountered as well as informational messages about the server.
- ❖ **Access log:** this file contains information about request types, protocols used, status codes, and bytes transferred. You can filter this log file to limit the amount of information you can see.
- ❖ **Archive log:** this screen is used to archive log files and to view previously created log files.

You can change your logging preferences from the Server Manager by selecting **Server Status** then selecting **Log Preferences**.

In addition to the actual log files, you can use the Log Analyzer to generate statistics about the server and summaries of activities. To run the Log Analyzer from the Server Manager, click **Server Status** then click **Generate Report**. Enter the name of the server, the log file to be analyzed, and the specifics of the report. This report can be printed to the screen or to a file. For large log file analysis, you should save the results to a file for later printing.

Other Monitoring Methods

You can also monitor the server by using the Simple Network Management Protocol, used to exchange data about network activity. Using SNMP, data travels between a managed device and a network management station.

Each proxy server also has its own Management Information Base (MIB), in a file called `ns-proxy.mib` stored in the `server-root/plugins/snmp` directory. This file contains the definitions for variables that are used for network management for the server.

See the Administration Guide for your system (UNIX or Windows), located at <http://docs.sun.com/app/docs/prod/s1.webproxys>, for complete details about using SNMP and the MIB.

Setting the Refresh Rate

HTTP documents have a descriptive header section that the proxy server uses, comparing the document in the cache to the one on the remote server. You can set a Cache Expiration setting for HTTP documents; this setting tells the proxy to estimate if the HTTP documents should be checked for updates before sending the request to the server. The estimate is based on the document's modified date, found in the header.

You can also use a refresh setting. This specifies if the proxy always does an up-to-date check or waits a specific period of time before doing a check. Using a refresh setting can decrease latency and save bandwidth.

You can configure most cache settings, including the refresh rate, by using the Server Manager and choosing **Cache** then **Specifics** from the menu that is displayed.

DISABLING CACHING

To disable caching, choose **Cache** from the Server Manager then **Specifics** from the menu that is displayed. Click **disabled**.

To disable the entire proxy server, choose **Server Preferences** from the Server Manager then click **Server Off**.

PROXY CONSIDERATIONS WITH SITE STUDIO

OVERVIEW

This chapter discusses the interaction of reverse proxy software with Content Server and Site Studio versions prior to the 7.5 version. It contains these sections:

- ❖ [Site Studio Design Considerations](#) (page -1)
- ❖ [Configuring Site Studio](#) (page -1)
- ❖ [Older Versions of Site Studio \(pre-7.2.1\)](#) (page -8)

The primary use of a reverse proxy is to provide HTTP acceleration. This documentation does not discuss the use of a reverse proxy with the Site Studio Publishing Utility.

Additional configuration changes may be needed for the proxy packages to work correctly with Site Studio. See [Chapter 6 \(Tips and Tricks\)](#) for details.

SITE STUDIO DESIGN CONSIDERATIONS

The HTTP headers used by Site Studio offer control over how the browser caches requests and how the proxy will handle the requests.

You can create a custom fragment that will insert the correct HTTP header in the page template. By using a custom node property, you can control the cache refresh rate and allow for custom refresh rates per section of the web page.

CONFIGURING SITE STUDIO

Before using Site Studio with any reverse proxy application, you will need to make changes to the Site Studio configuration. In some cases, you may also need to make changes to the proxy software to use Site Studio properly.

Site Studio Designer Changes

In Site Studio 7.5, the MaxAge section property is built into the product. Therefore you do not need to define it or create the custom fragment. This parameter will be used to define the expiration time of content for the `cache-control` HTTP header. If you are using Site Studio 7.5, see the information in step 2 on page -2 about adding the appropriate values for each node.

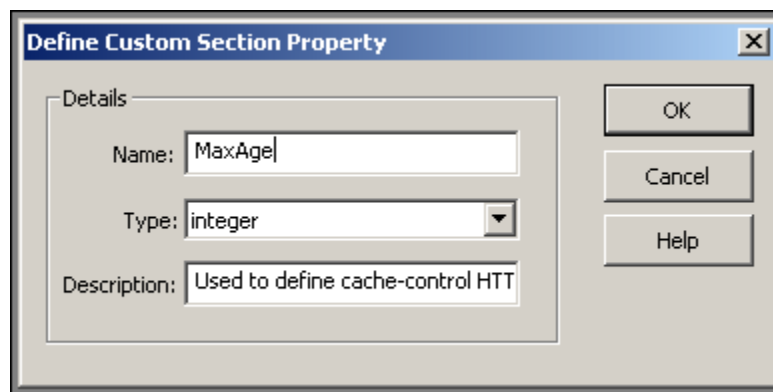
Changes for Versions Prior to 7.5

In the Site Studio Designer application **prior to version 7.5**, make the following changes:

1. Add a Custom Section Property (also known as a *Custom Node Property*) called `MaxAge`. Select **Tools—Define Custom Section Properties**.

The Define Custom Section Property screen is displayed.

Figure 5-4 Define Custom Section Property



Add the following information for this property:

- a name for the property which clearly identifies its use (in this example, `MaxAge`).
- its type. In this case, choose `integer` to define the parameter in seconds
- a brief description of the property.

2. After creating the property, it is added to the nodes in the site hierarchy. Set appropriate values (in seconds) for each node at your site.

To add the value for the parameter, select the node. If the Property sheet is not already open in Designer, open the sheet by selecting **View—Properties**. Enter a value for MaxAge.



Note: For testing purposes, set the MaxAge parameter to a high value (such as 160 or 240, to equal 2 or 3 minutes). This will allow you to verify that proxy caching is occurring as you planned.

3. Create a custom fragment that will contain a snippet. The fragment will be inserted on each layout page and will reference the MaxAge parameter. Select **File—Fragments—New**.

The New Fragment screen is displayed.

Figure 5-5 New Fragment screen

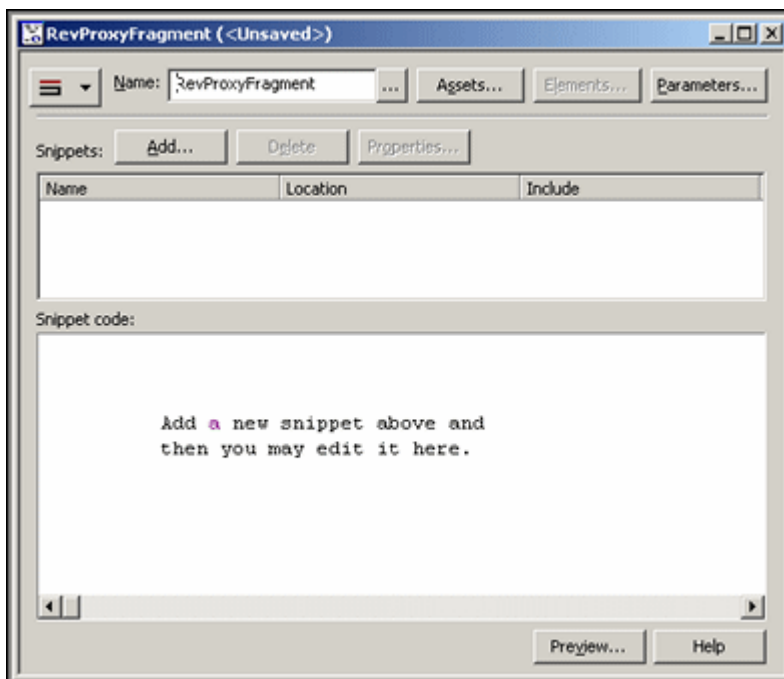
Property	Name
Name	
ID	
Language	idoc
Type	other
Library Name	<Unsaved>

Name
Enter a name for the fragment. This is the name that will appear in the toolbox.

4. Enter the following information:
 - Give the fragment a meaningful name.
 - Enter an ID. The ID is also an XML attribute and so cannot contain spaces or special characters.

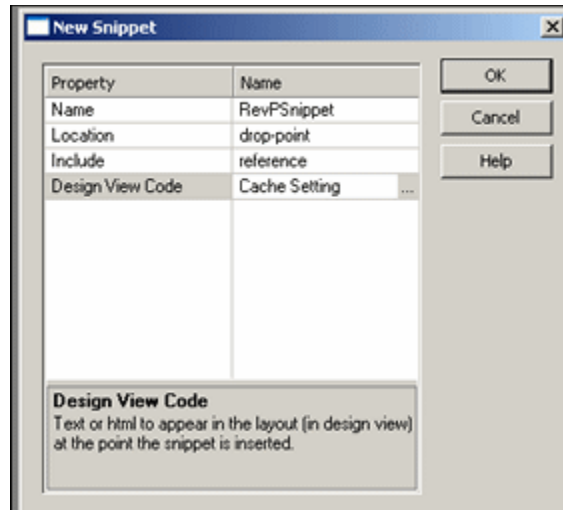
- You can choose either `idoc` or `jsp` for the language type. For this application, use `idoc`.
 - Select `Other` from the type list. This determines the fragment's location in the Toolbox.
5. When done, click **OK**.
- The Fragment Name screen is displayed.

Figure 5-6 Fragment Name screen



6. Click the **Add...** button in the Snippets section of this screen to add a new snippet to the fragment. The New Snippet screen is displayed.

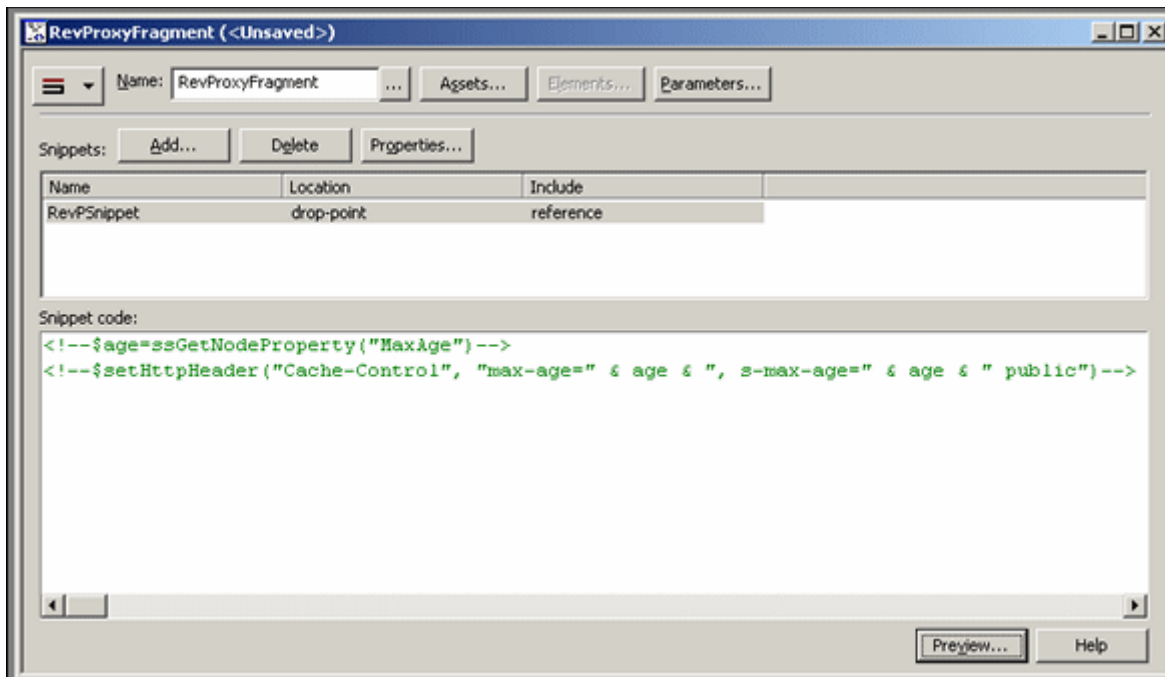
Figure 5-7 New Snippet screen



7. Add the following information:
 - Name: enter a meaningful name.
 - Location: choose a location from the pull-down menu where the snippet will be used in the layout. For this application, select **drop-point** to place the snippet in any location you choose.
 - Include: select how the snippet will be included in the layout. Select from simple, inline, or reference. A simple include inserts the code of the snippet permanently onto the page. An inline include also adds the code onto the page, but it retains its identity as a separate fragment. A reference include adds a pointer to the fragment on the page. For this application, select **reference**.
 - Design View Code: enter the text or HTML that will appear in the Design view of the layout where the snippet is inserted.
8. When done, click **OK**.
9. Add the following lines of code to the snippet (at the bottom of the Fragment Name screen):

```
<!--$age=ssGetNodeProperty("MaxAge")-->
<!--$setHTTPHeader("Cache-Control", "max-age=" & age & ", s-max-age=" & age &
" public")-->
```

Figure 5-8 Fragment Name screen (bottom)



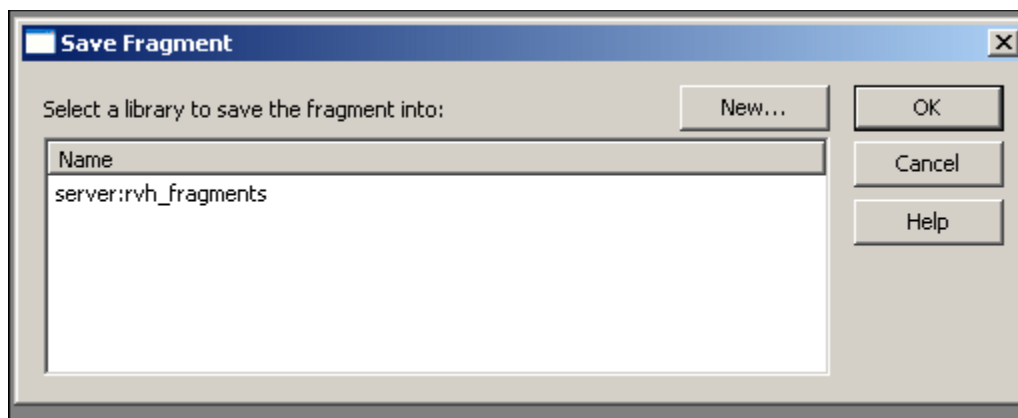
By setting the `max-age` value using the above code, you can set independent expiration times for different sections of the web site even if you reuse the same template. This is a better solution than hard-coding the `max-age` or setting it as a fragment parameter.

Other useful values are:

- `no-cache`: forces the cache to submit the request to the origin server for validation before releasing a cached copy.
- `must-revalidate`: tells the cache to obey freshness information given about an object.
- `proxy-revalidate`: similar to `must-revalidate` except that it applies only to proxy caches.
- `public`: marks the request as cacheable, even if it would normally be uncacheable. Many reverse proxy servers will not cache objects if they are not marked as `public`.

10. To save this snippet, click **File—Save**. Select a library in which to store the fragment or create a new library for the fragment by clicking **New...**

Figure 5-9 Save fragment screen



11. A Content Server **Assign Info** form is displayed. This is similar to a check-in form, but it is used for content related to your web site. Enter the appropriate information for the fragment (Content ID, type, title, and so on). Click **Assign Info** when done.
12. Your newly saved fragment appears on the Designer screen in the Fragment Type section that you chose in Step 3.
13. Add the fragment to each layout page used in your site. Click the layout page name in the Site Hierarchy panel and insert the snippet by clicking the snippet name in either the Design view or the Source view.
14. You can also put an `index.htm` file on the default web site of your Site Studio web server, redirecting users to the appropriate home page.

The following example shows how this redirection can be coded:

```
<HTML>
<HEAD><TITLE>HTTP Redirect</TITLE>
<META http-equiv="refresh" content=
"0;URL=http://localhost:3218/stellent/idcplg?IdcService=
SS_GETPAGE&nodeID=1000002">
</HEAD><BODY></BODY>
</HTML>
```



Important: Note that the URL is pointing to `localhost:3218` and not to `localhost`. It is important to redirect to the Squid server and not directly back to the backend server. In this case, 3128 is the port where the Squid server is running.

Content Server Changes

Perform the following operations in the Content Server that is used with Site Studio:

1. Select **Filter Administration** from the Content Server Administration menu.
2. Set **Disable GZIP Compression** to `TRUE`.

Figure 5-10 Configure Web Server Filter page

Configure Web Server Filter

General Options

Cache Timeout
This value specifies the number of minutes the web server filter will cache user data.

Default Authentication
This value specifies the default authentication method to apply to users who have never visited the Content Server before. The two valid choices are 'NTLM' and 'Basic'. 'NTLM' will use the Microsoft Login method to login users while 'Basic' will attempt to log users into the Content Server.

Disable GZIP Compression
By default, the content server compresses the HTML response pages for performance reasons. You may wish to disable it if CGI_RECEIVE_DUMP or CGI_DEBUG is enabled.

Host Web Server Changes

Depending on your operating environment, you may need to make changes to your host web server's cache, which is used by Site Studio. Change the web server's cache to make certain the .js files used by Site Studio are cached properly.

Also, if you created a default HTML page containing the HTTP redirect in step 14, make sure that it is included in your default web site properties.

Configuring the `ssBaseServerAddress` Variable

When using the Content Server with a reverse proxy server, you will need to set the `ssBaseServerAddress` variable in the `<install_dir>/config/config.cfg` file.

This is needed for any system using a reverse proxy server that forwards requests to the content server with HTTP headers that identify the content server as the HTTP server, rather than the proxy server.

Setting this variable allows you to specify that the base tag included in all Site Studio web pages uses a specified web address. This ensures the web site links function correctly when a content server is used behind a reverse proxy server. A typical value for `ssBaseServerAddress` is the full address of the proxy server, such as `http://www.myproxy.com`.

OLDER VERSIONS OF SITE STUDIO (PRE-7.2.1)

When using older versions of Site Studio, load the `SiteStudioSetBaseHref.zip` component. This is available for download from the support web site.

This component overrides the `ss_layout_head_info` resource include, which contains common declarations of server and client-side variables and methods.

The `SiteStudioSetBaseHref` component checks for a new configuration parameter named `ReverseProxyServer`. It uses that to construct the `<base href>` coding in the HTTP header. If the parameter is not found, the `<base href>` is constructed in the usual fashion by using `HttpServerAddress`.

To use a Reverse Proxy configuration, set `ReverseProxyServer=yes` in the configuration file provided with the component.



Important: If you have created other custom components that override the same resource include (`ss_layout_head_info`), use `SiteStudioSetBaseHref` with care. It will override the values you have set.

TIPS AND TRICKS

OVERVIEW

This chapter discusses some considerations to keep in mind as you use reverse proxy servers. It contains these sections:

- ❖ [Squid](#) (page -1)
- ❖ [Microsoft ISA](#) (page -3)
- ❖ [Sun Java System Web Proxy Server](#) (page -4)

SQUID

Several shortcuts can be followed when using Squid.

Squid and Site Studio

After installing and configuring Squid you will need to make some configuration changes for Squid to work correctly with Site Studio.

In the Squid directory (/usr/local/squid/etc on UNIX systems or c:\squid\etc on Windows systems), change the `httpd_accel_host` configuration entry in the `squid.conf` file to point to the server that is hosting the Site Studio site.

After configuring Site Studio as described in [Configuring Site Studio](#) (page -1), start Squid. If you used the recommended `MaxAge` of 240 for testing, you will need to change that value after testing is completed.

Setting Appropriate Directory Permissions

Because Squid normally starts when your system boots (by adding an entry to your startup files), Squid will run as root. Good security practices dictate that programs should run as root only if absolutely necessary, and for this reason, Squid switches to a default user (nobody) with no special privileges after it is started.

If you do not have root access to the machine and thus are not starting Squid as root, Squid runs with the user ID used to start Squid.

The `cache_effective_user` and `cache_effective_group` configuration options can be used to direct Squid to a new ID. See [Basic Configuration Options](#) (page -3) for information about these options.

Setting File Descriptor Limits

A file descriptor is a number that represents an open file or socket. Some systems have a relatively low per-process limit or low system-wide limit. The number of file descriptors needed for Squid depends on how many users are on the system, the size of the cache, and the particular features that are installed. Some common file descriptor uses are for TCP connections, log files, or idle HTTP connections.

If Squid is compiled and installed, check the `cache.log` file for a line similar to the following:

```
2005/01/05 09:10:23| With 1024 file descriptors available
```

This indicates the file descriptor limit on your system. If a file descriptor shortage occurs, a warning is inserted into the `cache.log`, similar to the following:

```
WARNING! Your cache is running out of file descriptors
```

The method of increasing file descriptor limits varies between operating systems. Consult your system documentation for details.

Using Swap Directories

If you receive `permission denied` errors while running the `squid -z` command, check that you are logged in as root and also check that the `cache_effective_user` and `cache_effective_group` configurations in the `squid.conf` file are both set to `squid` (or the name of the alias for the root squid user).

If you are still having problems accessing the swap directories, check the directory permissions of the `/usr/local` or `/usr/local/squid` directories (on UNIX). They should also have read and execute permissions for the `squid` user and group.

Querying Other Cache Servers

Web caches can be arranged in a hierarchical configuration or in a mesh configuration. If the cache structure is somewhat flat, it is a mesh configuration; a tree-like structure is a hierarchical configuration.

The `cache_peer` option allows you to specify the proxy servers that your server should communicate with. This option takes five arguments: the hostname or IP of the cache to be queried, the type of relationship between the two caches (for example, sibling, parent, and so on), the HTTP port of the destination server, the ICP (UDP) query port and any specified keywords.

See [About Hierarchical Cache Systems](#) (page -5) for an overview of hierarchy systems. See <http://squid.visolve.com/squid/> for details about Squid configuration options.

Controlling Which Pages are Cached

Access Control Lists (ACLs) are used to control which pages will be cached. For example, to avoid caching any URL containing the string 'database', add the following to the `squid.conf` file:

```
acl DATABASE urlpath_regex database
no_cache deny DATABASE
```

MICROSOFT ISA

This section discusses considerations to keep in mind when using Microsoft ISA server for a reverse proxy.

Using Windows Media Series

For a Windows Media server to act as a reverse proxy, the client must not be configured to use the server as a proxy. In addition, the server must contain a cache proxy plug-in that directs it how to handle content. For details, see the MSDN article at <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmsrvsdk/htm/implementingreverseproxy.asp>.

Network Adapters

ISA server computers require network adapters for communication with the different networks with which they have a connection. In a multi-server array, additional network adapters are required for intra-array communication.

Caching Considerations

If a web site has not appropriately configured its cache control directives for content that should not be cached, the ISA server may return the content, possibly to a malicious user.

ISA server caches objects to RAM and to the disk. By default, 10 percent of the RAM is used for caching objects and additional objects are cached only to the disk. This default can be changed when you configure the cache.

To use the ISA Server caching feature, you must install ISA on a computer with at least one partition formatted as an NTFS volume. For best performance, it is recommended that the cache partition be installed on a different drive than the one used for the ISA Server or the OS software.

If the current partition uses FAT partitions, you must convert the partitions to NTFS.

Setting Cache Parameters

You can create and edit cache rules which will specify the Time To live (TTL) of an object, as well as what type of objects you want cached (dynamic content, content

requiring user authentication, and so on). These rules are created by selecting the **Configuration** option then selecting **Cache** and **Create a Cache Rule** or **Configure a Cache Rule**.

In addition, you can set the cache so that frequently requested objects are automatically refreshed. This ‘active caching’ features tells the server to pre-fetch certain objects before they expire.

SUN JAVA SYSTEM WEB PROXY SERVER

This section discusses considerations to keep in mind when using Sun Java System.

Up-To-Date Checks

In order to improve performance, do not configure the proxy server to check the up-to-date status of a document every time the document is requested. Frequent checks of this type may be unnecessary and can consume network resources.

To improve performance and ensure a document is up-to-date, you can set a reasonable document lifetime in conjunction with the Last-Modified Factor (LMF). The LMF is a parameter which can be set to ensure that recently changed documents are checked more often than old documents. See the Administrator Guide for your operating system for details about setting this parameter.

Proxy Timeouts and CGI Scripts

The proxy timeout parameter sets the length of time that the server waits before aborting an idle process. A high timeout value could commit a proxy process to a potentially dead client for a long time. A low timeout value aborts CGI scripts that take a long time to produce their results (for example, a database query gateway).

If the proxy will handle many database queries and CGI scripts or if it will handle such a small amount of requests that it can spare a process at any given time, then you can set a high timeout value. The highest recommended value is 1 hour.

You can set the value by selecting the **Server Administration** page, then selecting **Server Preferences** and **System Specifics**.

Caching Pages Requiring User Authentication

Sun Java System Web Proxy Server can cache files that require user authentication. If you choose to set this type of caching, the server tags the files in the cache so if a user asks for them, it knows that the files require authentication from the remote server.

The proxy server does not know how remote servers authenticate. It forces an up-to-date check with the remote server each time a request is made for a document that requires authentication. If the user has accessed that server earlier in the session, the authentication information is automatically sent.

If you do not enable caching of pages that require authentication, the proxy assumes the default, which is not to cache them. You can set the policy for caching pages that require authentication by selecting the **Server Administration** page, then selecting **Caching and Configuration**.

See <http://docs.sun.com/app/docs/prod/s1.webproxys#hic> for complete details about setting up caching options.



THIRD PARTY LICENSES

OVERVIEW

This appendix includes a description of the Third Party Licenses for all the third party products included with this product.

- ❖ [Apache Software License](#) (page -1)
- ❖ [W3C® Software Notice and License](#) (page -1)
- ❖ [Zlib License](#) (page -3)
- ❖ [General BSD License](#) (page -4)
- ❖ [General MIT License](#) (page -4)
- ❖ [Unicode License](#) (page -5)
- ❖ [Miscellaneous Attributions](#) (page -7)

APACHE SOFTWARE LICENSE

```
* Copyright 1999-2004 The Apache Software Foundation.  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*   http://www.apache.org/licenses/LICENSE-2.0  
*
```

Third Party Licenses

- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.

W3C® SOFTWARE NOTICE AND LICENSE

- * Copyright © 1994-2000 World Wide Web Consortium,
- * (Massachusetts Institute of Technology, Institut National de
- * Recherche en Informatique et en Automatique, Keio University).
- * All Rights Reserved. <http://www.w3.org/Consortium/Legal/>
- *
- * This W3C work (including software, documents, or other related items) is
- * being provided by the copyright holders under the following license. By
- * obtaining, using and/or copying this work, you (the licensee) agree that
- * you have read, understood, and will comply with the following terms and
- * conditions:
- *
- * Permission to use, copy, modify, and distribute this software and its
- * documentation, with or without modification, for any purpose and without
- * fee or royalty is hereby granted, provided that you include the following
- * on ALL copies of the software and documentation or portions thereof,
- * including modifications, that you make:
- *
- * 1. The full text of this NOTICE in a location viewable to users of the
- * redistributed or derivative work.
- *
- * 2. Any pre-existing intellectual property disclaimers, notices, or terms
- * and conditions. If none exist, a short notice of the following form
- * (hypertext is preferred, text is permitted) should be used within the
- * body of any redistributed or derivative code: "Copyright ©
- * [\$date-of-software] World Wide Web Consortium, (Massachusetts

* Institute of Technology, Institut National de Recherche en
* Informatique et en Automatique, Keio University). All Rights
* Reserved. <http://www.w3.org/Consortium/Legal/>
*
* 3. Notice of any changes or modifications to the W3C files, including the
* date changes were made. (We recommend you provide URIs to the location
* from which the code is derived.)
*
* THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS
* MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT
* NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR
* PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE
* ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.
*
* COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR
* CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR
* DOCUMENTATION.
*
* The name and trademarks of copyright holders may NOT be used in advertising
* or publicity pertaining to the software without specific, written prior
* permission. Title to copyright in this software and any associated
* documentation will at all times remain with copyright holders.
*

ZLIB LICENSE

* zlib.h -- interface of the 'zlib' general purpose compression library
version 1.2.3, July 18th, 2005

Copyright (C) 1995-2005 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages

arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly jloup@gzip.org

Mark Adler madler@alumni.caltech.edu

GENERAL BSD LICENSE

Copyright (c) 1998, Regents of the University of California

All rights reserved.

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

"Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

"Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

"Neither the name of the <ORGANIZATION> 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 COPYRIGHT HOLDERS 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 COPYRIGHT OWNER 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.

GENERAL MIT LICENSE

Copyright (c) 1998, Regents of the Massachusetts Institute of Technology

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

UNICODE LICENSE

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories <http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>, and <http://www.unicode.org/cldr/data/>. Unicode Software includes any source code published in the Unicode Standard or under the directories <http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>, and <http://www.unicode.org/cldr/data/>.

Third Party Licenses

NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright © 1991-2006 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

Unicode and the Unicode logo are trademarks of Unicode, Inc., and may be registered in some jurisdictions. All other trademarks and registered trademarks mentioned herein are the property of their respective owners

MISCELLANEOUS ATTRIBUTIONS

Adobe, Acrobat, and the Acrobat Logo are registered trademarks of Adobe Systems Incorporated.

FAST Instream is a trademark of Fast Search and Transfer ASA.

HP-UX is a registered trademark of Hewlett-Packard Company.

IBM, Informix, and DB2 are registered trademarks of IBM Corporation.

Jaws PDF Library is a registered trademark of Global Graphics Software Ltd.

Kofax is a registered trademark, and Ascent and Ascent Capture are trademarks of Kofax Image Products.

Linux is a registered trademark of Linus Torvalds.

Mac is a registered trademark, and Safari is a trademark of Apple Computer, Inc.

Microsoft, Windows, and Internet Explorer are registered trademarks of Microsoft Corporation.

MrSID is property of LizardTech, Inc. It is protected by U.S. Patent No. 5,710,835. Foreign Patents Pending.

Oracle is a registered trademark of Oracle Corporation.

Portions Copyright © 1994-1997 LEAD Technologies, Inc. All rights reserved.

Portions Copyright © 1990-1998 Handmade Software, Inc. All rights reserved.

Portions Copyright © 1988, 1997 Aladdin Enterprises. All rights reserved.

Portions Copyright © 1997 Soft Horizons. All rights reserved.

Portions Copyright © 1995-1999 LizardTech, Inc. All rights reserved.

Red Hat is a registered trademark of Red Hat, Inc.

Sun is a registered trademark, and Sun ONE, Solaris, iPlanet and Java are trademarks of Sun Microsystems, Inc.

Sybase is a registered trademark of Sybase, Inc.

UNIX is a registered trademark of The Open Group.

Verity is a registered trademark of Autonomy Corporation plc

Third Party Licenses

Glossary

G

GLOSSARY

Array

Used in Microsoft ISA proxy software. An array represents one or more ISA server computers running the services that are physically connected and share the same configuration.

Cache

An HTTP proxy that caches Internet requests.

Cache Array Routing Protocol

Used in Microsoft ISA proxy software to provide scaling and efficiency when using multiple ISA server computers that are arrayed in a single logical cache. CARP uses hash-based routing to determine the best path through an array to resolve a request.

Cache Hit

A cache hit indicates that a valid copy of a requested object exists in a cache.

Cache Miss

A cache miss indicates that a valid copy of a requested object does not exist in a cache or that the copy in the cache is no longer valid.

CARP

See [Cache Array Routing Protocol](#) (page -1).

Child Cache

Part of a cache hierarchy in which the parent cache receives requests from subordinate child caches. See [Parent cache](#) (page -3).

Configuration Storage Server

Used in Microsoft ISA proxy software. A configuration server is one on which the configuration for all the arrays in an enterprise are stored.

Enterprise

Used in Microsoft ISA proxy software. An enterprise is a collection of ISA Server services that are grouped into arrays. An enterprise includes one or more Configuration Storage servers that store ISA server information, such as the arrays and servers which are part of the enterprise.

Hierarchical Topology

A web cache structure which is tree-like, composed of nodes and branches.

Internet Cache Protocol (ICP)

A method of inter-cache communication in which one cache checks another to see if it has a valid copy of a named object. This improves the chances that a neighbor cache will return a cache hit.

ISA Server Management

Used with Microsoft ISA proxy software. The console through which an administrator manages an ISA enterprise.

ISA Server Services

Used with Microsoft ISA proxy software. This is the computer that runs the firewall, VPN and caching functions of the ISA Server. It is connected to a Configuration Storage server, which stores the configuration information.

Mesh Topology

A relatively flat web cache structure.

NAT

See [Network Address Translation](#) (page -2).

Neighbor

Used to describe web caches in a hierarchical cache topology. Neighbor caches can also be related to each other. See [Sibling Relationship](#) (page -4).

Network Address Translation

A type of network connection used with Microsoft ISA proxy software. When this type of connection is specified the ISA server replaces the IP address of the client on the source network with its own IP address.

Network Load Balancing

Used in Microsoft ISA proxy software to balance network traffic going into the array. NLB uses virtual IP addresses that are shared among all array servers. Each array member can pick packets sent to a virtual IP address, according to a specified NLB algorithm.

Network Object

Used in Microsoft ISA proxy software. Networks which are grouped into sets such as subnets, URL sets, or domain name sets. Rules can be applied to network objects.

Network Rules

Used in Microsoft ISA proxy software. Network rules can be set to define and describe a network topography.

NLB

See [Network Load Balancing](#) (page -3).

Object

Generic term for a document, image, or piece of content available on the Internet.

Origin Server

The source location for an object. This is the hostname portion of a URL.

Parent cache

Part of a parent/child relationship in a cache hierarchy. A child cache forwards requests to its parent cache. See [Child Cache](#) (page -1).

Peer

See [Neighbor](#) (page -2).

Refresh Rate

The refresh rate of a cache determines how long an item should be considered 'fresh' in the cache.

Regular Mapping

A Sun Java System Proxy Server configuration setting which redirects requests to the main server.

Request Method

The method used to retrieve HTTP requests.

Request URL remapping

See [Regular Mapping](#) (page -3).

Response Header Remapping

See [Reverse Mapping](#) (page -4).

Reverse Mapping

A Sun Java System Proxy Server configuration setting used by the proxy server to trap redirects from the server.

Route

A type of network connection used with Microsoft ISA proxy software. When this type of connection is specified, client requests from the source network are directly relayed to the destination network and the source client address is included in the request.

Round Trip Time

The time interval between sending the first byte of an HTTP request until the last byte of the server response has arrived at the requesting web client.

Sibling Relationship

Part of a hierarchical cache structure in which web server caches are considered peers. A sibling cannot forward cache misses on behalf of the peer.

Simple Network Management Protocol

A protocol used to exchange data about network activity. Using SNMP, data travels between a managed device and a management station where users remotely manage the network.

SNMP

See [Simple Network Management Protocol](#) (page -4).



A

- adding a snippet, 5-5
- administrative roles, 3-3
- administrative users
 - extended monitoring, 3-4
 - full administration, 3-4
- administrative users
 - monitoring, 3-4

C

- cache querying, 1-5
 - process, 1-5
- cache rules, 3-3
- cache testing, 2-6
- cache_dir, 2-3
- cache_dir configuration option, 2-3
- cache_effective_group, 2-3
- cache_effective_user, 2-3
- cache_peer, 6-2
- changes needed to host web server, 5-7
- combination delivery type, 1-2
- configuration considerations, 2-3
- content delivery approaches, 1-1
- Content Server
 - changes needed to use proxy, 5-7
- Custom Node Property, 5-2
- Custom Section Property, 5-2

D

- directory permissions, 6-1
- Disable GZIP Compression, 5-7
- disabling POST caching, 2-4
- document audience, 1-7
- document conventions, 1-7
- dynamic delivery, 1-2

F

- file descriptor limits, 6-2
- files
 - mime.conf, 2-3
 - mime.conf.default, 2-3
 - sample squid.conf, 2-5
 - squid.conf, 2-3
 - squid.conf.default, 2-1
- Filter Administration, 5-7
- forward proxy, 1-3

H

- hierarchical cache systems, 1-5
- hierarchical proxy querying, 1-5
- http_access, 2-3
- http_port, 2-3, 2-4
- httpd_accel_host, 2-4
- httpd_accel_port, 2-4
- httpd_accel_single_host, 2-4

I

- installation
 - Microsoft ISA, 3-4
 - Squid, 2-1
- ISA installation, 3-4

M

- MaxAge parameter, 5-2
- Microsoft ISA
 - RAM caches, 6-4
- Microsoft ISA
 - administrative users, 3-3
 - cache rules, 3-3
 - caching overview, 3-3
 - installed network rules, 3-3
 - multi-networking, 3-1

- network address translation, 3-2
- network communication, 3-2
- network definition, 3-2
- network object, 3-2
- network objects, 3-2
- network preconfiguration, 3-2
- network rules, 3-2
- performance tuning, 3-5
- pre-installation, 3-1
- refresh rate, 3-5
- route connectivity, 3-2
- terms, 3-2
- tips, 6-3
- TTL setting, 3-3
- verifying performance, 3-5
- Web Proxy filter, 3-3
- web site caches, 6-3

mime.conf file, 2-3

multi-networking, 3-1

N

NAT, 3-2

P

peer, 1-5

POST requests, 2-4

proxy caching

- introduction, 1-3

proxy server

- benefits, 1-3

proxy timeouts, 6-5

purging squid cache, 2-5

Q

QUERY_STRING logging, 2-4

querying cache servers, 6-2

R

regular mapping, 4-5

remapping proxy requests, 4-5

remapping requests, 4-5

request URL remapping, 4-5

response header remapping, 4-5

reverse mapping, 4-5

reverse proxy

- benefits, 1-4
- definition, 1-3

reverse proxy configuration, 2-4

- Sun Java Systems, 4-5

ReverseProxyServer variable, 5-8

S

sample squid.conf, 2-5

setting max-age, 5-6

sibling, 1-5

Site Studio

- cache snippet code, 5-5
- changing node values, 5-2
- configuration details, 5-1
- custom fragment, 5-2
- Custom Section Property, 5-2
- Designer application, 5-1
- MaxAge value, 5-6
- pre-7.5 versions, 5-8
- redirect, 5-7
- setting expiration for headers, 5-2
- squid configuration, 6-1
- ssBasseServerAddress variable, 5-8

snippets

- adding, 5-5

Squid

- basic configuration options, 2-3
- configuration help
 - UNIX, 2-2
- configuration options
 - acl, 2-4
 - acl configuration option, 2-4
 - cache_dir, 2-3
 - cache_effective_group, 2-3
 - cache_effective_user, 2-3
 - cache_peer, 6-2
 - http_port, 2-3, 2-4
 - httpd_accel_host, 2-4
 - httpd_accel_single_host, 2-4
 - reverse proxy, 2-4
 - Site Studio, 6-1
- default configuration files, 2-3
- directory permissions, 6-1
- file descriptor limits, 6-2
- installation
 - UNIX, 2-1
 - Windows, 2-2
- log files, 2-4
- querying servers, 6-2
- swap directories, 6-2
- tips, 6-1

squid

- purging cache, 2-5

squid binary file, 2-2

- squid.conf file, 2-3
 - sample, 2-5
- ssBaseServerAddress, 5-8
- static delivery, 1-2
- Sun Java System
 - additional proxy servers, 4-4
 - and other Sun servers, 4-3
 - cache configuration, 6-5
 - configuration options, 4-4
 - downloading, 4-1
 - General Administration section, 4-2
 - installation information needed, 4-2
 - installation on Microsoft systems, 4-2
 - log files, 4-7
 - monitoring performance, 4-7
 - remapping, 4-5
 - remapping URLs, 4-5
 - reverse mapping, 4-5
 - Server Administration section, 4-3
 - setting up Reverse Proxy Server, 4-5
 - supported operating systems, 4-1
 - tips, 6-4
 - UNIX environments, 4-3
 - UNIX installation requirements, 4-3
 - UNIX user account, 4-3
 - up-to-date checks, 6-4
 - URL menu, 4-5
 - user authentication, 6-5, 6-5
 - Web Proxy main page, 4-3
- SunJava System
 - proxy timeout, 6-5
- swap directories, 6-2
- system tuning, 2-1

T

- testing cache
 - Squid, 2-6
- Time To Live setting, 3-3
- tips
 - Microsoft ISA, 6-3
 - squid, 6-1
 - Sun Java System, 6-4
- TTL
 - setting, 3-5

U

- UNIX
 - Squid installation, 2-1
- up-to-date checks, 6-4
- user authentication, 6-5, 6-5

V

- verifying performance
 - Squid, 2-6
 - Sun Java System, 4-7
- verifying system performance
 - Sun Java System, 4-7

W

- web proxy filter, 3-3
- Windows
 - installing ISA, 3-4
 - Squid installation, 2-2
 - Squid installation location, 2-2

