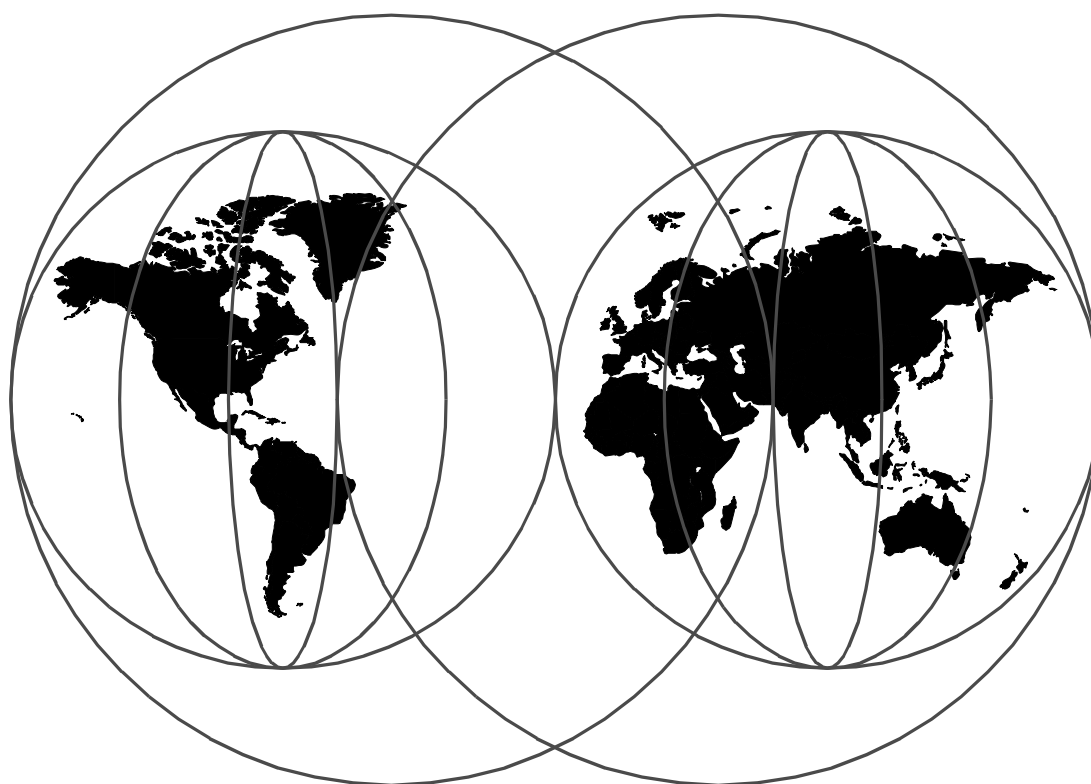


Caching and Filtering to Manage Internet Traffic and Bandwidth Demand

Marco Pistoia



International Technical Support Organization

<http://www.redbooks.ibm.com>

Take Note!

Before using this information and the product it supports, be sure to read the general information in "Special Notices" on page 25.

January 1999

This white paper applies to the Caching and Filtering component of IBM WebSphere Performance Pack Version 1.0 for use with the AIX, Solaris and Windows NT operating systems.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HZ8A Building 678
P.O. Box 12195
Research Triangle Park, NC 27709-2195

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Preface

The Internet is rapidly reaching almost every business and home in the world. Millions of people have the ability to create their own Web sites and put anything they want on them. As more businesses and individuals publish Web sites, the traffic on the Internet is increasing, while the content of the material varies widely. The Caching and Filtering component of IBM WebSphere Performance Pack is a caching proxy server that minimizes the time you spend waiting for data retrieval and allows you to focus on the sources deemed to be most productive. It accomplishes this through innovative caching, persistent connections between the client and the proxy, improved scalability, and filtering of nonproductive or inappropriate URLs.

This white paper gives you a clear understanding of the features of the Caching and Filtering component of IBM WebSphere Performance Pack and shows complex scenarios that demonstrate how this component can be successfully integrated with the other WebSphere Performance Pack functions to build record-breaking Web sites.

Who Wrote This White Paper

Marco Pistoia is an International Technical Support representative, working as a project leader at the International Technical Support Organization Raleigh Center. He writes extensively and teaches IBM classes worldwide on all areas of the e-Business Application Framework, Java and Internet security. Marco holds a degree with honors in Pure Mathematics from the University of Rome and a masters degree in Computer Science. Before joining the ITSO, he was a System Engineer in IBM Italy. He received an Outstanding Technical Achievement Award in 1996.

Thanks to the following people for their invaluable contributions:

Karen Gelveles
IBM, Integrated Solutions Marketing, Boca Raton

Susan Hanis
IBM, Research Triangle Park, North Carolina

Gail Christensen, Linda Robinson
IBM, International Technical Support Organization, Raleigh Center

Caching and Filtering with IBM WebSphere Performance Pack

The explosion of the Internet over the past few years has resulted in a rapid expansion in bandwidth and other resources deployed by Internet service providers (ISPs) to support this unprecedented demand for access. ISPs are now looking to contain the costs associated with this growth. One approach is proxy caching, which can significantly reduce bandwidth costs by retaining highly used Web objects locally. This white paper describes all the details of the Caching and Filtering component of IBM WebSphere Performance Pack, which has been designed by IBM to meet the needs of ISPs worldwide to continue their growth while effectively managing bandwidth costs.

Internet Growth and Bandwidth Management

In five short years the Internet has gone through an entire generation of usage. Simple e-mail and file transfers gave way to the phenomenal growth resulting from the ease of use associated with Web browsers. Adventurous entrepreneurs have explored the potential of browser technology to reach customers and provide a variety of information. The success of such early ventures has led many more marketing groups in traditional businesses to exploit the Internet for new advertising and customer information channels. Consumer response to such capability has been so positive that the Internet today is considered an essential marketing or fulfillment channel for the majority of businesses, large or small. This, in turn, has made the Internet one of the most competitive business environments and one upon which many organizations are increasingly dependent.

The dependency on and desirability of using the Internet has led to recent debates about the ability of the Internet to handle the expectations and load of existing information and emerging applications. While some problems and outages have indeed occurred and are frequently cited by pessimists as proof of the limitations of the Internet, there have also been significant successes and developments that permit thoughtful network managers to deploy more robust and scalable Internet infrastructures.

The Problem to Be Solved

As we said, Internet growth over the past five years has been exponential both in terms of numbers of users and in terms of traffic being generated by those users. Network managers in both private and public networks represent the groups most aware of the implications of such growth and the need to provide infrastructure for their users to handle the growth.

ISPs need to be able to handle thousands of concurrent users and to maintain satisfactory performance and access for those users consistently during peak usage periods as well as normal usage. They also need to address demands of businesses customers and consumers to restrict distribution of certain types of information. In the future, they will want to differentiate their services by offering greater value through functional enhancements that enable users to get more or better information from individual requests.

Enterprise network managers may not need to support as many concurrent users as most ISPs, but they need to be able to support traditional business data traffic as well as intranet traffic and access to external Internet sites. They need to be able to ensure that precious network resources can be optimized differently according to the time of day and priorities of the business for different types of traffic.

Record-Breaking Web Sites with IBM WebSphere Performance Pack

IBM has gained significant experience with the problems and technologies needed for severely demanding Internet environments through its support of major events such as the Olympic Games and the Deep Blue Chess Match. From this real-life environment it has developed IBM WebSphere Performance Pack, one of the most robust and scalable infrastructure product available today.

IBM WebSphere Performance Pack is Web infrastructure software that addresses the scalability, reliability and performance needs of e-business applications in both local and geographically distributed environments. Its functions incorporate dynamic traffic load balancing, content replication and distribution management, and superior caching and filtering functions for bandwidth management. These functions may be used individually and together to meet the specific needs of various networks, and compensate for the inherent weakness of the Internet to support critical business applications and expectations.

IBM WebSphere Performance Pack has been developed using IBM's extensive experience with very demanding Web sites. It is composed of three main components, which permit you to reduce Web server congestion, increase content availability and improve Web server general performances:

1. File Sharing

The File Sharing component, also known as the AFS file system, is an enterprise file system that enables cooperating hosts (client and servers) to efficiently share file system resources across both local area networks and wide area networks. It provides nondisruptive real-time replication of information across multiple servers, which guarantees data consistency, availability, global stability and administrative efficiency, required by large distributed Web sites or by Web sites with volatile content, which generate considerable administrative effort to maintain content links and URL to file I/O mapping.

2. Caching and Filtering

The Caching and Filtering component, also known as Web Traffic Express (WTE), is a caching proxy server that provides highly scalable caching and filtering functions associated with receiving requests and serving URLs. With tunable caching capable of supporting high cache hit rates, this component can reduce bandwidth costs and provide more consistent rapid customer response times. Using the Caching and Filtering component, `Unable to Connect to Server` errors, which today are so frequent, may be dramatically reduced.

3. Load Balancing

The Load Balancing component, also known as eNetwork Dispatcher (eND), is a server that is able to dynamically monitor and balance TCP servers and applications in real time. The main advantage of the Load Balancing

component is that it allows heavily accessed Web sites to increase capacity, since multiple TCP servers can be dynamically linked in a single entity that appears in the network as a single logical server.

The above components, which were previously unavailable in a single Internet software offering, can increase the scalability, availability and reliability of your Web site, while reducing infrastructure costs.

Installation procedures permit selection of which components to install, and specification of which machine(s) the selected component(s) should be located on. Subject to installation needs and operating platforms, components can coexist on a single machine or can be distributed over multiple machines.

IBM WebSphere Performance Pack allows you to design several architectures to enhance the performance of your Web site. IBM used the WebSphere Performance Pack technology to create a scalable and reliable system that efficiently handled unprecedented traffic volumes. On February 17, 1998, at 12:41 (Japan Standard Time), the Official Web site of the Olympic Winter Games in Nagano made Internet history by logging a staggering 98,226 hits per minute. Less than a week later, a new all-time record was established with a peak load of more than 103,400 hits per minute, while still providing normal response time. The Internet site of the 1998 Nagano Olympic Winter Games is recognized by the Guinness Book of World Records.

The Caching and Filtering Component

The Caching and Filtering component of IBM WebSphere Performance Pack is a proxy server with caching capabilities and with the ability to filter Internet content at the proxy server level:

- It is a proxy server, which means that it assumes the responsibility for retrieving Internet data for multiple browser clients. Client requests are sent to the Web servers through the proxy. In other words, the client is configured to send its request to the proxy first, and then it is the proxy that forwards the client's request to the Web server, acting on behalf of the originating client. The Web server does not even see the IP address of the client, but only the address of the proxy server. Once the proxy receives the information from the Web server, it forwards the information to the requesting client. This way the responsibility and the machine load associated with making the URL request is transferred to a dedicated machine.
- A traditional proxy server receives a request for a URL from a client and then forwards the request to the destination Web server. The Caching and Filtering component of IBM WebSphere Performance Pack does something more: it has a caching functionality, which means that it can save, or cache, the Web documents it retrieves that are considered cacheable according to the HTTP protocol and administrator defined overrides. It can then serve subsequent requests for cached documents from its local cache. The client gets the information faster and network bandwidth utilization is reduced.
- It has a content filtering functionality, implemented through Platform for Internet Content Selection (PICS) labels, which rate Web material by criteria such as language, nudity or violence. This is a consistent way to implement filtering on a broad scale and control the content you are providing. You can apply these filters in addition to, or instead of, the filtering set by the browsers,

It is especially useful where end users do not (or should not) have access to those controls.

This component of IBM WebSphere Performance Pack uses an innovative caching scheme, just released from IBM's research laboratories, to improve end-user response time. It allows you to customize its caching features to your own benefit. You can specify which pages are cached, when the information on a page will expire, how large to make the cache and when to update it. It offers key features of advanced caching, such as:

- The ability to handle very large caches.
- Enhanced caching algorithm that accommodates the variable size and arrival characteristics of Web objects in order to optimize your end user response time.
- An option to automatically refresh the cache with the most frequently accessed pages. These can be identified by an administrator or determined by a system from the cache logs.
- The possibility to cache even those pages where the header information says to fetch them every time.
- Configurable cache maintenance that includes a daily, rather than continual, garbage collection for improved server performance.
- HTTP 1.1 compatibility, including support of persistent connections between the client and the proxy server.
- Capability to exploit the power of today's symmetrical multiprocessor (SMP) machines.

Notice that client browsers must be configured to automatically send their requests to a proxy server. When a caching proxy server receives a request, it first attempts to serve the request from its own cache. If the requested document is available in the cache, the proxy server returns it to the Web client. If the document is not in the cache or is old, the proxy server forwards the request to the appropriate Web server. When the proxy server receives the response, it relays the response to the Web client. If the proxy server determines that the response can be cached, it stores it, so that it is available for subsequent requests.

Moreover, the Caching and Filtering component of IBM WebSphere Performance Pack allows you to set content filtering at the proxy server level, rather than or in addition to the browser level, where it could be easily compromised or overridden. This way network managers can control which Internet services, sites and information their users can access. For example, filters can be configured to prevent client machines from accessing nonproductive or offensive URLs.

Content filtering in the Caching and Filtering component of IBM WebSphere Performance Pack can use:

- Platform for Internet Content Selection (PICS) rules guiding use of rating labels - such as Recreational Software Advisory Council on the Internet (RSACi) criteria for inappropriate language, nudity or violence - placed in HTML or HTTP headers or third-party content rating label distributions
- Lists of URLs/sites for which access is to be blocked
- APIs for filtering applications

The Caching and Filtering component of IBM WebSphere Performance Pack also has the functionality of a basic Web server. This feature is very useful for server remote administration and monitoring purposes. In fact the administrator can configure the caching and filtering proxy server either by editing the configuration files or by remotely accessing the Configuration and Administration Forms through the Web, using a common Web browser. However, we discourage using the Caching and Filtering component as a pure Web server. Although this would be possible, a more powerful Web server, such as the Lotus Domino Go Webserver or Apache HTTP Server, is recommended, especially for its capability to be integrated with IBM WebSphere Application Server.

Other important features of the Caching and Filtering component of IBM WebSphere Performance Pack include:

- Flexible-client SOCKS that allows requests for specific IP addresses to go directly to the destination server instead of being routed through the SOCKS server.
- An option to increase client anonymity by configuring the proxy server to strip or modify HTTP header information, `From:` fields and client IP addresses.
- A Proxy Activity Monitor that provides summary information and recent entries from the cache and proxy access logs. You can use this data to configure the caching features and to improve your server's performance.
- Files transferred by the FTP protocol are cached *only* when a complete file is received. Incomplete FTP files are not cached. Additionally, FTP files are maintained in the cache in the same way as HTTP files. The Caching and Filtering component of IBM WebSphere Performance Pack generates a `Last-Modified:` header based on the date of the file, which is obtained from an FTP directory listing, and uses this date to calculate an expiration time for the file. When a cached FTP file expires, the caching and filtering proxy server compares the current file date for the file, which is obtained by doing an FTP directory listing for the file, to the FTP directory listing file date obtained earlier, and decides whether to serve the cached file or request a new one from the FTP server.

When used in conjunction with the Load Balancing component of IBM WebSphere Performance Pack, this caching and filtering proxy server becomes a massively scalable solution. Add IBM eNetwork Firewall to your architecture and you will have a scalable and secure solution for caching and filtering Internet information.

Who Can Benefit

The Caching and Filtering component of IBM WebSphere Performance Pack provides a valuable and scalable solution to some of the major traffic management problems. These are the main advantages it offers:

- Reduction of costs and constraints on network bandwidth, particularly during periods of peak concurrent activity
- Scaleable infrastructure that provides cost-effective growth paths and essentially unlimited capacity potential with minimum redesign or disruption
- Bandwidth management capabilities based upon content filtering and proxy functions

- Content management capabilities based upon industry-standard filtering technologies to restrict information or to enhance information provided to users
- Functional openness to permit evolution and exploitation of emerging Internet capabilities with minimal impact on overall network architecture
- Multiple platform support to simplify implementation planning and skill requirements

Caching Proxy Function

The caching proxy function provided by the Caching and Filtering component of IBM WebSphere Performance Pack is valuable to customers and/or ISPs needing to optimize line costs and performance associated with accessing remote Web sites. Customers that can benefit from using the caching and proxy function of a caching and filtering proxy server include:

- Service providers needing to provide good response time to clients from Web sites accessible only via expensive or distant links that carry significant propagation delay time.

They need to be able to provide nondisruptive access to information on servers located within their networks as well as those external to their networks. Many ISPs must have infrastructures capable of cost-effective expansion to handle growth rates greater than 10% per month.

- Enterprises with significant external Web access (Internet and/or intranet) wanting to optimize wide area line usage.

Large corporations and university campuses are the most likely to benefit from caching, to improve response time while optimizing external links.

Filtering Function

The filtering function of the Caching and Filtering component of IBM WebSphere Performance Pack is particularly valuable to customers and/or ISPs wanting to be able to filter out content from Web sites on the basis of defined codes, rules or APIs. Such customers include:

- Service providers wanting to enable PICS filtering for consumers or to enforce regulations prohibiting certain types of content.

- Service providers wanting to be able to filter content to suit the needs of closed user groups, such as subscriber-based communities of interest.

Such filtering would likely reflect non-standard rules, programmed to suit the preferences of a particular interest group.

- Service providers wanting to be able to filter or intercept content to enable additional processing such as premium content (authentication or billing) or language translation.

- Business network managers wanting to be able to minimize traffic on backbone network links during peak periods by filtering traffic according to application or content priorities.

- Education institutions, particularly grade schools, wanting to control student access to nonapproved content.

An Overview of Caching Functions

Caching technologies have long been used within computer systems to streamline and optimize I/O operations and memory access. Spooling, buffering, and/or paging technologies are all variations of caching functions. However, when applied to networking environments, and particularly to Web server environments, simple adaptation of these technologies do not necessarily have the same or desired effectiveness.

Most simple caching technologies define a basic cache or buffer capacity that can be used to capture and store information likely to require re-use. Within a single operating system, centralization of resource management functions permit optimal management and awareness of the caching resources. In addition, most I/O or memory access designs process information in fixed lengths to optimize transfer times, using segmentation or chaining in a controlled environment to ensure the integrity of the data. Thus, the buffers or caches can be optimized for *look ahead* or *least recently used* algorithms to improve their effectiveness.

In a wide area network, and on the Internet, the controlled and centralized resource management is not available, nor is the fixed-length attribute of information being transferred. Thus the effectiveness of the caching function is significantly impacted by such additional factors as:

- The capacity and scalability of the cache itself
- The way information in the cache is stored and retrieved, including indexing and searching mechanisms
- The number of concurrent accesses that can be supported
- The number and speed of communication links or ports that can be enabled
- The algorithms used to purge content or re-use storage within the cache
- The ability to recognize and manage perishable or time-dependent content
- The ability to expand or redefine caching function nondisruptively
- The ability to analyze caching performance to gain useful information for tuning or planning network resources
- The ability to share caching information effectively in a distributed configuration to maximize performance and availability while reducing content storage redundancy and costs

The Caching and Filtering component of IBM WebSphere Performance Pack provides highly scalable caching function on multiple platforms suitable to a wide variety of network environments from smaller caching systems to essentially unlimited capacity, and very complex network infrastructures.

Increasing Hit Bytes and Hit Rates

Caching effectiveness is typically expressed in terms of *hit bytes* or *hit rates*, referring to the number of Bytes retrieved from a cache (and therefore not transmitted over the network) and the number of cached objects retrieved as a percentage of all objects served. In fact, hit bytes are more commonly utilized as a measurement in operational environments such as backbone networks, where bandwidth and uplink network access costs are the reason for caching. Hit rates,

on the other hand, are the measurement most likely to be referenced in those environments such as user access points, where consistency and speed of user response time is the key reason for caching.

The Caching and Filtering component of IBM WebSphere Performance Pack has several features that result in an enhancement of caching effectiveness:

- Increase cache capacity

The Caching and Filtering component supports multiple platforms, including highly scalable clustered nodes such as the SP2 and can be combined with the IBM ADSTAR Distributed Storage Manager, an archiving system with essentially unlimited capacity.

- Preload cache with known high demand content

The Caching and Filtering component permits preloading of information on the basis of named content and pre-fetching of content with automatic location and fetching of linked objects to a pre-specified *depth*.

- Monitor usage to support tuning

The Caching and Filtering component supports logging that can be used as input to preloading and other administrative activities.

- Purge content least costly to retrieve and keep content most expensive to retrieve (that is, consider size of object in addition to usage indicators)

The Caching and Filtering component enables network administrators to specify weighting criteria to reflect their specific cost trade-offs between cache capacity and bandwidth costs. The Caching and Filtering component also permits administrators to purge or keep content based upon *time-to-live* headers and the relative size of the objects.

- In larger networks, use caching servers in structured hierarchy, that is, second level caching

The Caching and Filtering component of IBM WebSphere Performance Pack can be configured to recognize a *parent cache* that will be searched next in the event a specific request is not found in the local cache. Multiple tiers can be supported, subject to trade-offs in latency for the end users between cache processing time versus network access.

Cache Content Management

Most network managers using a caching function to minimize network bandwidth costs want to get the best *hit rate* possible, that is, to increase the probability that user requests can be satisfied from content within the cache versus a network request. Hit rates of 30-50% for Internet traffic can significantly improve both bandwidth demand as well as user-perceived response times. However, if reduction in bandwidth is the goal, a better measurement would be hit bytes, which is probability expressed in terms of bytes rather than absolute requests. Thus a hit Byte ratio of 30% would be equivalent to a 30% increase in network bandwidth.

There are several parameters where the administrator can act:

- Which documents are kept in the cache
- How many documents can be cached

- How long they are considered current
- How the documents are indexed
- When the cache is refreshed

This section gives an overview of how caching works.

Controlling Which Documents Are Kept in the Cache

By configuring the Caching and Filtering component of IBM WebSphere Performance Pack, you can specify which documents should be cached, how long they should be cached and which documents should never be cached.

Notice, however, that some files are never cached:

- Documents that were requested through HTTP methods other than `GET`, such as `POST` or `PUT`
- All the documents that were obtained after authentication or payment
- All the documents that were dynamically generated by CGI-BIN scripts or Java servlets
- Any information passed on an SSL connection, because the proxy cannot decrypt the data passing through it
- Any URL containing a question mark `?` in it

Some documents, even though they do not belong to the above categories, have a `Cache-control: no-cache` directive in their header. This directive can appear in two places:

1. It can be included with the document when it is returned by the Web server, instructing a caching proxy not to cache the file in the cache.
2. It can be included in a request from a client, instructing the caching proxy to get a fresh copy of the document from the current server, even if a copy in the document exists in the cache.

The administrator of the Caching and Filtering component of IBM WebSphere Performance Pack can configure the caching proxy server to override the `Cache-control: no-cache` directive that is received as part of the request from the client. In this way, if a request is received from a Web browser that specifies `Cache-control: no-cache`, this directive in the request will be ignored and the document will be served from the proxy cache, assuming it exists there.

Cache Freshness

An important issue that the administrator of the caching and filtering proxy server must face is to ensure that cached documents are consistent with the original documents located at the originating Web server. In other words, the administrator must ensure *cache freshness*.

For each document that has been cached, the Caching and Filtering component of IBM WebSphere Performance Pack computes a time at which the document will expire:

- For HTTP documents, it is the header of the document generated by the Web server that contains the expiration information.

- For FTP documents, it is the Caching and Filtering component of IBM WebSphere Performance Pack that generates its own `Last-Modified:` header information to compute expiration times. The reason for this is that the FTP protocol does not include expiration information, unlike the HTTP protocol.

The Web server can indicate the expiration time in several ways, putting header information in the HTTP response. The permitted header information is in the following order of preference:

1. The Web server can specify the time the document is good after it has been received.
2. The Web server can specify the exact time at which the document should be considered expired.
3. The Web server can specify the time when the document was last modified. Then the caching and filtering proxy server performs a sequence of operations based on a class of parameters specified in the configuration file and calculates how long the document will be good.

When a document is found in the cache, but it has expired, the Caching and Filtering component of IBM WebSphere Performance Pack issues a special request to the Web server. This request is known as `if-modified-since`. The Web server sends back the document to the caching and filtering proxy server only if such a document has been modified since it was last received by the proxy. Otherwise, the Web server only sends a message indicating that the document has not been modified, and does not send the entire file. At this point, the caching and filtering proxy server can serve the page to the client.

The caching and filtering proxy server administrator is also allowed to specify how long to keep unused cached files.

Cache Size and Garbage Collection

Disk space and file maintenance are common concerns when using a cache. The Caching and Filtering component of IBM WebSphere Performance Pack allows you to control the amount of disk space used for the entire cache.

Another important feature implemented in the Caching and Filtering component of IBM WebSphere Performance Pack is the nightly cleanup process known as *garbage collection*. This process examines the files in the cache directory and attempts to remove old, expired or unused files to make room for more current files.

There are two algorithms that the garbage collection process can use when deciding which files to remove and which files to keep in the cache. One algorithm maximizes the cache to improve user response time and the other maximizes the cache to minimize network bandwidth:

1. When you are tuning your cache to minimize response time, larger files are given a higher priority for deletion and, therefore, are more likely to be removed during garbage collection.
2. When you are tuning your cache to minimize network bandwidth, larger files are given a lower priority for deletion and they are less likely to be removed during garbage collection.

Cache Indexing

The Caching and Filtering component of IBM WebSphere Performance Pack implements a cache directory structure and lookup methods that are different from many other proxy servers. It creates an index of the files in the cache to keep in memory as each page is added. RAM memory is used instead of other media, so that the lookup operation and retrieval times are faster.

The index separates the cached files into a set of sub-caches, and for each file in the cache the index stores in memory the file name, URL and expiration information. For this reason, the RAM memory required is directly proportional to the number of files in the cache.

When the caching and filtering proxy server receives a request from a client, the proxy checks the index in memory for that particular URL:

- If the file is not in the index, the request is made to the destination server. The retrieved URL is then checked to ensure that the document is cacheable, and the document is cached if this operation is permitted. The index is then updated with the new URL, sub-cache and expiration information.
- If the file is in the index, the expiration information is checked to see if the URL is stale. If the URL has expired, the caching and filtering proxy server contacts the content destination server, and the URL is replaced by the newly retrieved document with expiration and sub-cache information updated in the index. If the URL is still consistent, the document is served.

The cache contains shadow files that mirror the index information for the proxy to use only when the proxy server is started. The garbage collection process updates the cached document index files.

Automatic Cache Refreshing

Typically, the most common proxy servers cache a particular page only after a user requests it. The Caching and Filtering component of IBM WebSphere Performance Pack, in addition to this default caching, has a *cache agent* that provides automatic caching and gives more control to the administrator. The cache agent can retrieve specified URLs even before they are effectively requested and refresh the cache automatically. The cache refresh takes place when the proxy server activity is low (by default, every night at midnight, local time) and all the retrieved pages are ready in the cache to provide faster service even the first time a user requests them.

The automatic cache refreshing has two sources it can use to refresh the cache:

1. It can load specific URLs defined by the administrator. In this way, the administrator can specify a certain set of pages that must be loaded by the cache agent when it starts.
2. It can load the most popular URLs from the previous day's activity. To obtain this, the cache agent checks the cache access log, sorts it by frequency of requests and then picks the most popular pages. It can refresh the top number of pages as specified by the administrator.

Notice that the cache agent can use both sources of input.

Optionally, the cache agent can follow a specified level of HTML links on the pages it is loading and cache all of those linked pages. This operation is also

known as *delving*. It is not necessary that the linked pages reside on the same host as the source page, since the cache agent can retrieve them even if they reside on other hosts.

The cache agent offers a very useful service. Using the cache agent, caching is performed even before cached pages are effectively requested, so the average response time is minimized. Moreover the cache is built before user activity gets busy, typically at night. However, turning the cache agent on forces the caching and filtering proxy server machine to be busy even during hours of low activity. Moreover, configuring the cache agent to perform delving requires more control to the caching and filtering proxy server administrator. For example disabling delving from high-level pages such as Web indexes or search sites, is recommended, or multiple requests for large numbers of pages will be generated.

Flexible-Client SOCKS

The Caching and Filtering component of IBM WebSphere Performance Pack has a particular feature, named *flexible-client SOCKS*, that allows the caching and filtering proxy server to reside behind a firewall or SOCKS server without sharing the same physical machine. Requests going to the proxy can then be routed directly to the destination Web server, instead of sending all requests through the SOCKS server. None of the components of IBM WebSphere Performance Pack include a SOCKS server. We recommend that you install the SOCKS server provided by IBM eNetwork Firewall.

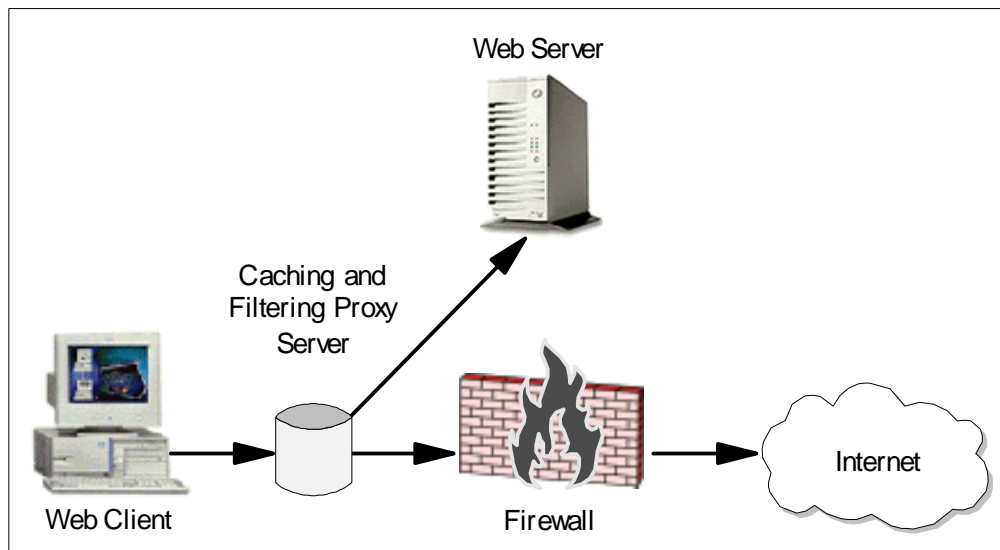


Figure 1. Flexible-Client SOCKS in the Caching and Filtering Component

The flexible-client SOCKS functionality helps security by allowing a firewall server to be isolated from the proxy server, even if this requires additional hardware and can produce higher latency on requests. The load on the firewall is reduced by having the caching and filtering proxy server handle internal requests. Moreover the administrator can easily specify which requests the caching and filtering proxy server sends to the SOCKS server and which requests it redirects back to the local domain.

A traditional proxy server installed behind a firewall would route all the requests to the firewall itself. On the contrary, the flexible-client SOCKS provided by the Caching and Filtering component of IBM WebSphere Performance Pack lets you specify which IP addresses or domains should be contacted directly by the proxy server and which ones should be contacted through the SOCKS server.

Proxy Chaining

Proxy chaining is a mechanism that allows you to create a hierarchical chain of proxies, each proxy belonging to a certain level in the hierarchy. If a proxy server in the lowest level of the hierarchy cannot serve the requested URL from its cache, it does not forward the client's request to the content Web server, but to the proxy server that has been configured as the proxy server of the immediately higher level in the hierarchy. The higher the level in the chain, the larger the number of users that access that proxy, so the possibility that a proxy server at a higher level in the chain finds the requested document in its cache becomes greater.

The proxy server at the top level of the chain contacts the Web server to retrieve the documents if such a document was not in its cache. After that, it passes the document back down in the hierarchy, and all the proxies under it cache the document, if this is cacheable, until the lowest proxy caches the document and serves it to the requesting client.

The following is a graphical representation of the proxy chaining process:

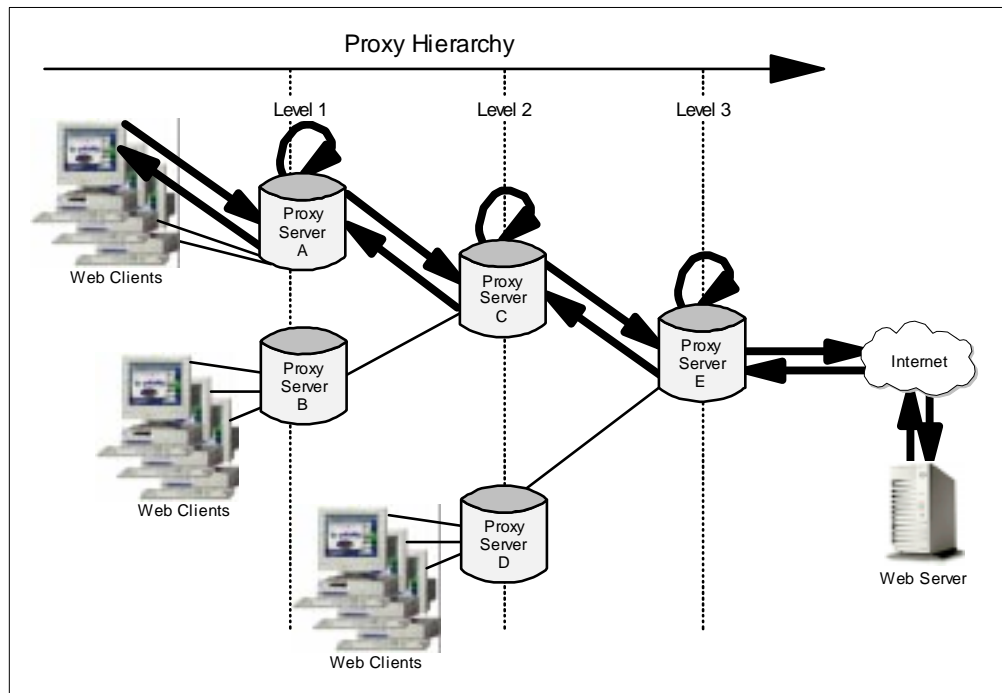


Figure 2. Graphical Representation of the Proxy Chaining

Notice that the request goes through the daisy chain of servers and the response passes through the same proxies but in the reverse way. If intermediate proxies have caching enabled, each of them would search its own cache for the

requested resource and return a cached copy, without forwarding the request to the originating Web server.

This architecture offers the following advantages:

- Proxies at a lower level, which are closer to the client that originated the request, benefit from the caches of the higher-level proxies.
- Proxy chaining reduces the load on the highest level proxy (typically, the proxy closest to the firewall) and ultimately on the Web server, since lower-level proxies may already have the document cached.
- The larger the number of users, the higher the probability that the proxy server already has the document in its cache. Considering that high-level proxies serve a larger number of clients, many requests that cannot be honored by a low-level proxy can be resolved by higher-level proxies, since other groups of clients may have already requested the same files.

However, it should be kept in mind that:

- A high-level proxy should have a larger cache, since it has to honor the requests of a large number of users.
- Proxy chaining greatly increases response time for requests, especially for those files that have not been cached yet by any proxies in the hierarchy.
- The risk of failure in a chain increases with each additional node.

The Caching and Filtering component of IBM WebSphere Performance Pack allows the creation of proxy chains based on the protocol (HTTP, FTP or Gopher) of the request to be forwarded. In other words, a proxy server can be configured to send all incoming requests with a particular protocol to a higher level proxy server in the chain.

PICS Filtering at the Proxy Server Level

Platform for Internet Content Selection (PICS) is a technological standard that builds up a content labeling and filtering system for Web information. It is formed by a set of specifications used to create and manage ratings for the information published on any given Web site. The idea behind PICS is relatively simple. Since people have all kinds of different preferences and values, there should be a ratings and labeling standard that people can use to choose what content they would welcome seeing, or wish to exclude, based on certain parameters.

PICS development began in mid-1995, when the computing and online industries became sensitized to the possibility of online content censorship by the U.S. government and other governments around the world. Working under the aegis of the World Wide Web Consortium (W3C), the interested parties came together to create a technological solution that would support different rating systems. The PICS standard was adopted in May 1996. For the most up-to-date PICS information, see the World Wide Web Consortium's PICS Web site at the URL <http://www.w3.org/PICS/>.

PICS is a standard that lists rules for rating the information contained on any given Web site. Rating decisions are made on the basis of particular categories, usually violence, pornography, language and nudity. Some Web browsers, such as Microsoft Internet Explorer 4.0, are PICS-compliant and are able to filter

content information as content is received from the destination server. However, relying on browser settings may not be a safe solution, because such settings can be adjusted at the browser, and may be easily compromised.

With the Caching and Filtering component of IBM WebSphere Performance Pack, you can implement PICS filtering at the proxy server level. This removes the responsibility away from the client, and your proxy server administrators can directly prevent certain types of information from being served to specific browsers (or to groups of browsers). Browsers with PICS filter settings defined will then be able to perform further filtering of the Web pages they will be served. This method ensures that clients will get only the level of content specified at the proxy. Interaction between the client and the proxy administrator would be required to change the sensitivity of the filter.

Using this centralized approach, the PICS filtering process is transparent to all the users. When a user requests an HTML page, they will see the requested URL or get an error message similar to the following:

Error 403
Blocked by Filtering Rule

The Caching and Filtering component of IBM WebSphere Performance Pack allows an administrator to specify filtering rules based on PICS labels. When a URL is accessed, the caching and filtering proxy server uses these rules to determine if it passed or failed. The PICS labels can be supplied to the caching and filtering proxy server in several ways. They can be stored locally on the proxy's hard disk, supplied by a label bureau or even provided by the Web server. Some URLs have the label embedded within their HTML files under the `<META>` tag or in the HTTP response header. Security mechanisms, such as message digests and digital signatures, can be incorporated in the label creation, to grant label validity.

The diagram below shows the logical flow of the PICS filtering at a proxy server level, supposing that the PICS labels are provided by a label bureau:

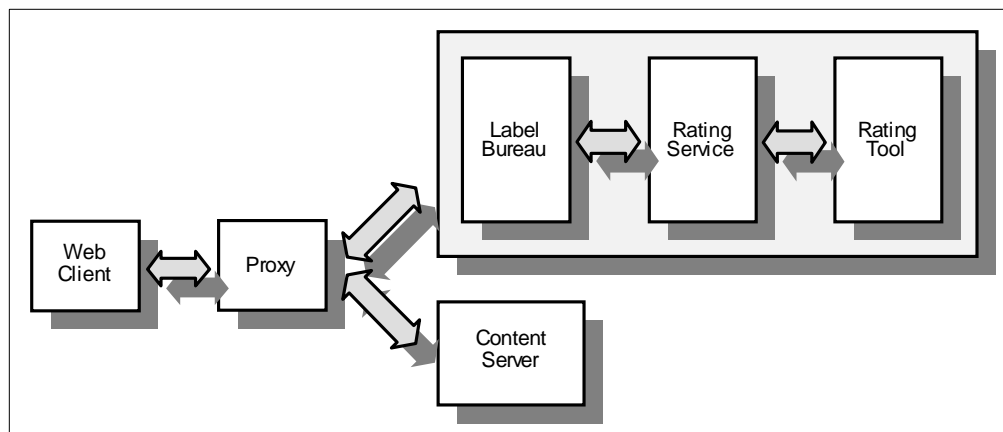


Figure 3. PICS Filtering at the Proxy Server Level

Here, the proxy server administrator has set up and enabled a filtering profile on the proxy server. To decide whether a particular document will be passed or blocked, the proxy filtering profile uses the values contained in the PICS labels supplied by a label bureau server, which is in turn managed by a rating service.

The rating service might own a rating tool to examine particular URLs and create labels describing those URLs. This rating tool can implement a procedure to discover a new site as soon as it goes online, examine the site and store a PICS label at the label bureau server.

Later, when content from this site is requested by the user's client software, the PICS label is requested from the label bureau server by the proxy. If the profile establishes that the values in this PICS label mean that the content is not wanted, then the proxy will send an HTML page back to the client, explaining the reason why the content is not being delivered. It may or may not include instructions on how to bypass the blocking or how to initiate the correction of a faulty rating if necessary. However, in most cases, the values in the label mean the content is acceptable, and the content is fetched from the Web server to the proxy and then forwarded to the user.

If the proxy caching is enabled, during further fetches the user sees no delay, since the proxy server caches only pages that have previously been accepted, and then serves them directly from its cache.

The label bureau server might tell the proxy that this site has not yet been rated, and the profile determines whether the user is sent the content anyway or a `Not Yet Rated` page by the proxy is sent. The reviewer's tool is notified by the label bureau that this site should be rated as soon as possible. Some label bureau servers in the future will fetch the content and run a program to create an interim rating that will be returned, pending the reviewer's more accurate site evaluation.

The diagram shown in Figure 3 on page 15, which we have just explained, is simplified if the Web server is enabled to embed PICS labels in the Web documents it serves (either in the HTTP header or in the HTML header), and the proxy server is configured to accept such labels without the need to contact an external label bureau.

Other Functions

Beside the caching and filtering functionality, this component of IBM WebSphere Performance Pack has other useful features, which we describe in the following.

Handling Header Information

The Caching and Filtering component of IBM WebSphere Performance Pack allows you to increase client anonymity by configuring the caching and filtering proxy server to strip or modify HTTP header information.

Proxy Activity Monitor

The Proxy Activity Monitor consists of multiple pages that contain information about the activity of the proxy server. It provides summary information and recent entries from the cache and proxy access logs. You can use this data to configure the caching features and to improve your server's performance.

Caching and Filtering Proxy Server Access Protection

The Caching and Filtering component of IBM WebSphere Performance Pack can be configured to protect access to its resources, when it works as a typical Web server. It is also possible to configure it to require user ID and password

authentication to all the users that try to access the proxy function. It is possible for you to specify, for example, that only authorized users can access your caching and filtering proxy server, by requiring them to authenticate with user ID and password. As another example, you can restrict access to your caching and filtering proxy server, allowing only requests generated by users belonging to specified domains to come.

SSL Tunneling

The Caching and Filtering component of IBM WebSphere Performance Pack supports Secure Socket Layer (SSL) connections. SSL secure connections involve encryption and decryption processes and are established directly between the client browser and the destination Web server. The caching and filtering proxy server does not make any attempt to cache or decrypt the information that the client and the proxy server exchange during an SSL connection, but it establishes a connection to the destination Web server and passes the requests to it without looking at the data.

A Complementary Offering: IBM Distributed Web Traffic Express

The Caching and Filtering component of IBM WebSphere Performance Pack is not the only solution that IBM offers to cache and filter Internet content. IBM has developed Distributed Web Traffic Express (DWTE), a complementary offering that can be integrated with the Load Balancing and File Sharing components of IBM WebSphere Performance Pack in order to build complex record-breaking scenarios.

DWTE uses the same code base as WTE, the Caching and Filtering component of IBM WebSphere Performance Pack, but has specific feature enhancements. The complementary function used by DWTE is called Remote Cache Access (RCA), and allows proxy servers to cooperate to form cache arrays. Using RCA, multiple proxy servers can distribute the cache contents across their combined, logical cache to improve hit rates and reduce redundancy of cached content.

DWTE uses new caching algorithms and information-sharing technologies to enable an ISP to manage its servers more efficiently by storing information where it is more likely to be needed and delivering it more efficiently to customers. These enhancements reduce transmission costs and eliminate the need for ISPs to replicate information in redundant proxy servers.

How Distributed Web Traffic Express Works

RCA enables peer WTE proxy servers, in close physical proximity to each other, to share the contents of their caches utilizing a shared file system, such as AFS, DFS, NFS, or Windows NT file sharing. We recommend the use of AFS, the File Sharing component of IBM WebSphere Performance Pack, since it offers nondisruptive real-time replication of information across multiple servers, data consistency, availability, global stability and data consistency.

The peer servers must be configured as an array. RCA uses the CARP protocol to determine which peer in the array should process the incoming request. If the requested file is not in the proxy's cache, it will query the other peers in the array to determine which proxy might have the object cached. If the object is in another cache, the owning proxy indicates where the file can be found in the shared file

system. If the object is not in any cache, the proxy processing the request will get the file and then cache it.

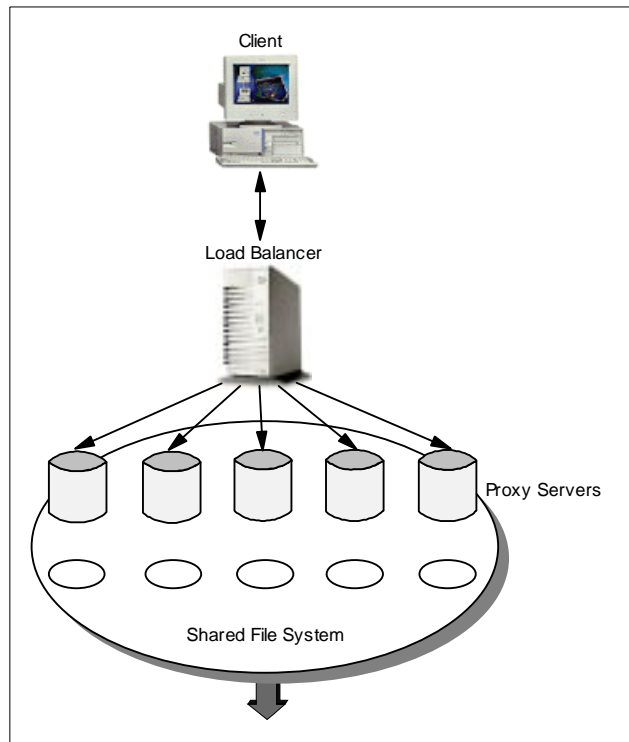


Figure 4. How Distributed Web Traffic Express Works

Improved Cache Hit Rates with Innovative Algorithms

DWTE uses enhanced caching algorithms aimed at increasing the likelihood that a request for an object on the Web will find it in the first cache it searches. The algorithms improve *hit probability* by streamlining the ways this local cache obtains and temporarily stores information from remote caches.

Traditional caching algorithms were designed to manage objects strictly on the basis of frequency of use, independent of their size and movement across a network. In reality, Internet objects vary in size, as do the costs of keeping them in cache and distributing them to local subscribers. The new algorithms used by DWTE are optimized for ISP environments, where large-scale performance and bandwidth costs are critical. These innovative algorithms guide the proxy server or servers to manage objects based on ISP requirements for reducing bandwidth costs and improving performance for local subscribers.

Flexible Caching Configuration

DWTE enables ISPs to distribute the caching function across multiple servers to improve performance and response time. Providing more local caching can also result in savings in line costs. In addition, DWTE also improves information flow by keeping each server in the system aware of what is available in the cache of its peers. This cache array enhancement reduces storage costs by minimizing the need for redundant data, and improves performance with the addition of servers in the cache array as required. Because it is scalable - readily expandable in size through the addition of servers as needed - the solution allows ISPs to grow as

their subscriber base grows. CARP-like replication further enhances performance by minimizing the number of index hits required for the average search.

This combination of central and local caching gives the network administrator the control and flexibility to maintain a high performance network.

Who Can Benefit

ISPs can use the transactional orientation of DWTE to evolve from Web access and hosting into new areas of growth, such as e-business. Using this solution, an ISP with a large customer base can save many millions of dollars a year on long-distance bandwidth alone.

Large ISPs that serve widely dispersed customers will benefit from the reduced bandwidth requirements of DWTE. This solution will be of great value to international ISPs dealing with high volume of subscribers.

Scenarios

IBM WebSphere Performance Pack allows you to design and perform multiple complex architectures. The scenarios described in this section provide specific examples of how various ISP implementations can benefit from the use of IBM WebSphere Performance Pack. In particular they demonstrate how the Caching and Filtering component of IBM WebSphere Performance Pack can be successfully integrated with the other components.

The Caching and Filtering component of IBM WebSphere Performance Pack offers you much flexibility in terms of architecture design, since you can easily integrate it in different areas of the network, depending on your needs:

1. It can be installed as an ordinary proxy on the client side of the network architecture. With this architecture, the Web server is accessed only to retrieve pages that are not found in the local cache of the proxy server. Static pages are retrieved from the Web server only the first time they are requested or when they have expired. Then they are cached and are served directly from the local cache of the proxy server. The workload on the destination server is mainly generated by requests for dynamic pages, which are not cached.

This particular choice offers the following advantages:

- It minimizes the number of machines that connect to the firewall or gateway
 - It minimizes the network traffic that crosses the firewall or gateway
 - It reduces the utilization of the network segment that connects the firewall to the Web server (and this is particularly useful to optimize line costs and performance associated with accessing remote Web sites)
 - It dramatically reduces the end-user response time
2. The Caching and Filtering component of IBM WebSphere Performance Pack can be installed on the server side of the network architecture. A proxy server placed in the server side is called a *reverse proxy server*. This architecture is particularly indicated to reduce the workload on the Web server machines for frequently accessed Web sites.

Placing the Caching and Filtering component of IBM WebSphere Performance Pack immediately before the Web servers offers several advantages:

- The performances of the Web site are strongly enhanced, since the caching and filtering proxy server contacts the destination server only for those pages that are not found in the local cache, such as dynamic pages or static pages that are found expired or are requested for the first time. In all the other cases, Web pages are served directly from the proxy server's local cache.
 - The number of times the Web site is effectively accessed is minimized.
 - The end-user response time is dramatically minimized.
 - Client machines do not need to be configured to forward their requests to the proxy server. The proxy presence is transparent to clients.
 - The proxy server acts on behalf of multiple clients, which in this way do not need to know even the IP address of the Web server. In other words, the reverse proxy server hides the structure of the Web site to the clients, and this offers the possibility to use a private address for the Web server. This feature is particularly useful when the Web site is composed of multiple Web servers, load-balanced by the Load Balancing component of IBM WebSphere Performance Pack.
3. In a proxy chaining configuration, caching and filtering proxy servers at multiple levels can be installed in the network architecture (see "Proxy Chaining" on page 13).

This feature offers several advantages:

- The load on the higher level proxy (typically, the proxy nearest to the firewall) and ultimately on the Web server is reduced, since lower proxies may already have the requested document cached.
- The larger the number of users, the higher the probability that the proxy server already has the document in its cache. The client that originated the request benefits from the cache of the higher level proxies, and this reduces the end-user response time.

When used in conjunction with the Load Balancing component of IBM WebSphere Performance Pack, the caching and filtering proxy server becomes a massively scalable solution. However, multiple proxy servers cannot share the same cache and a single Web page needs to be retrieved from the originating Web server multiple times before it is available in the cache of all the proxy servers.

This problem can be solved by using DWTE, which has the same code base as the Caching and Filtering component of IBM WebSphere Performance Pack, but has specific feature enhancements. RCA, the complementary function used by DWTE, offers support for reducing redundancy of cached content (see "A Complementary Offering: IBM Distributed Web Traffic Express" on page 17 and Figure 4 on page 18). In the following, we show how some of the scenarios below may also need the functions of remote caching to reduce redundant page storage.

Content Hosting Internet Service Providers

Content hosting ISPs can effectively use all the components of IBM WebSphere Performance Pack to more effectively support and distribute the content from their own Web sites, and to provide more response and cost-effective access to other sites. Within a content hosting *server farm*, proxy caching can be used to

provide more responsive access to content from other sites, as well as to optimize backbone network traffic capacity, and Web content filtering can be applied to prevent end users from accessing offensive or nonproductive contents.

Thus content hosting ISPs or corporate webmasters can benefit from the nondisruptive replication and distribution capabilities of the file sharing functions, local and wide area load balancing, and proxy caching. Flexible configuration of these components can ensure that requests are directed to the most appropriate local or remote location, and can enable location outages or routine maintenance schedules to be handled without disrupting customers.

The IBM WebSphere Performance Pack components can be used in conjunction with firewalls and authentication gateways to provide secure access where desired.

Corporate Web Sites and Content Aggregators

Use of IBM WebSphere Performance Pack by corporate Web sites and content aggregators is similar to that of content hosting ISPs. Caching and filtering proxy servers may be deployed to filter or optimize access from within the corporation to external Web sites.

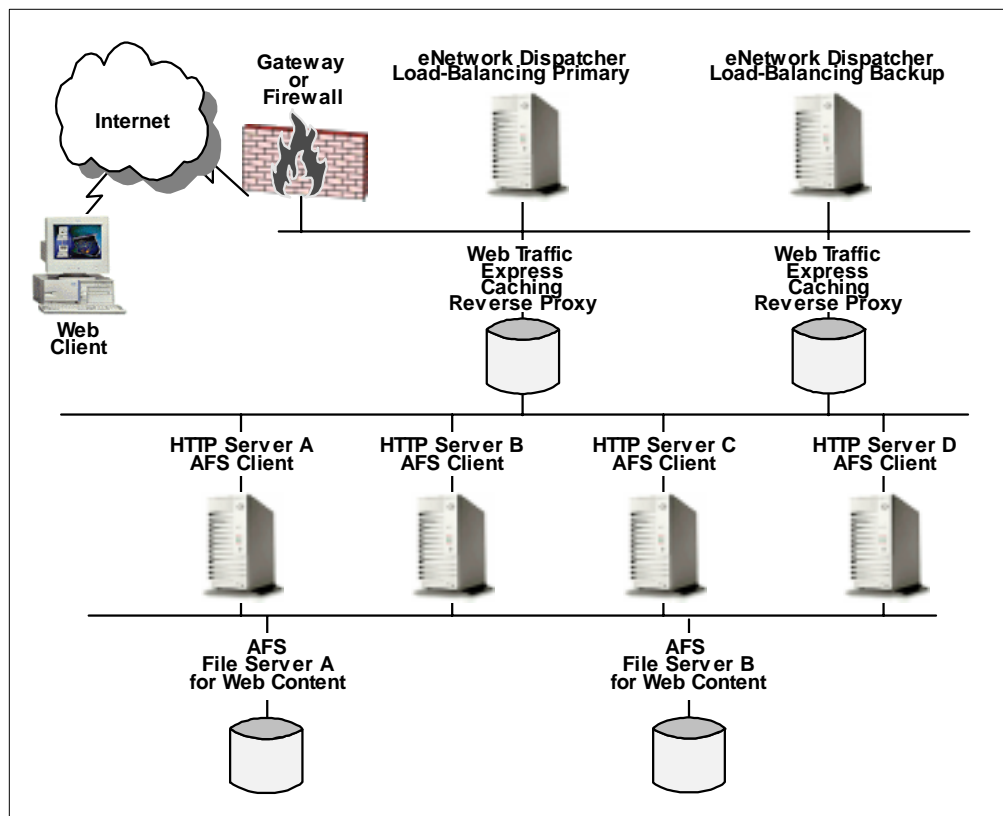


Figure 5. Corporate Web Sites and Content Aggregators

Many corporate Web sites are located at head office or regional locations, while branch offices or business partners may need frequent access to the content. Most offices of this type have relatively low line speed (somewhat less than 1.5 Mbps) network access to the regional or corporate sites. Relatively few users can, with concurrent usage, use all the available bandwidth with resulting erratic

response times. At these locations HTTP servers with AFS clients combined with general-purpose caching can provide more consistent user response time.

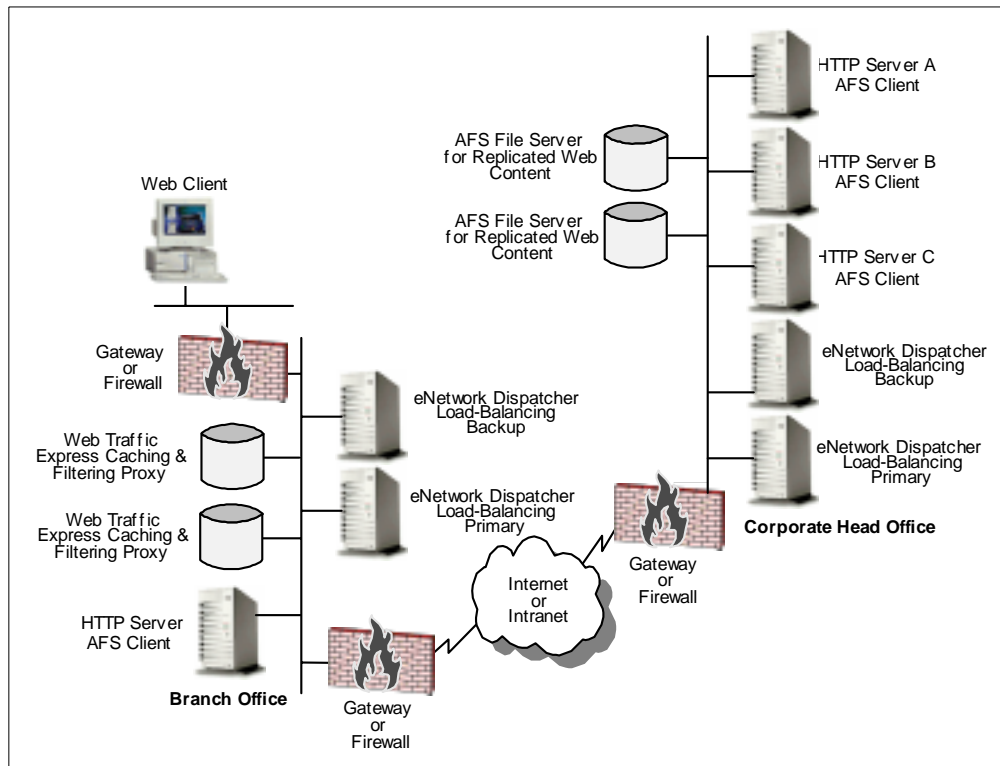


Figure 6. Head Office - Branch Office

Some industries have both small branch offices in addition to larger branch or regional offices. A simple deployment of IBM WebSphere Performance Pack caching and filtering on a single platform, eventually combined with a firewall, is probably sufficient to provide more consistent response time for employees in these offices. In the larger branches or regional offices, it may be desirable to have more than one caching and filtering proxy server and to use the Load Balancing functionality to provide better resource management.

Two approaches can be used to reduce redundant page storage and to address the effectiveness of caching in these locations:

1. Hierarchical caching

Optimize a primary (initial) cache for higher page hit rate by favoring more files of a smaller size, and a hierarchical cache for higher hit byte rate by favoring fewer files of a larger size.

2. Distributed Web Traffic Express

Configure RCA together with the Load Balancing function and with AFS shared file storage to reduce the index size of each proxy and to increase the scalability of the caching storage.

Corporate Headquarters Buildings or Large Campuses

On large campuses or in corporate headquarter buildings, the size of the campus or number of personnel frequently leads to the creation of smaller local area

networks interconnected by routers and a backbone LAN. Busy LAN servers combined with increasing use of Web server applications can result in congestion on the backbone LAN segments. This can be reduced by installing AFS client-enabled HTTP servers and general proxy caching on user LAN segments. The AFS clients can provide caching for corporate Web content, while the proxy server can provide caching and filtering for external Internet access. Design considerations for the smaller LAN segments are similar to those for small branch offices above.

Backbone Internet Service Providers

Backbone ISPs typically provide co-location and/or peering services for other ISPs in addition to content hosting for large national or international corporations. In many cases they may provide *virtual ISP services* for other ISPs such as content hosting ISPs. Backbone ISP customers are increasingly demanding both high availability and differentiated service levels and backbone ISPs are responding by enhancing their Internet infrastructures.

Features demanded by backbone ISPs include proxy caching. To minimize the effects of *hot potato* routing and Web traffic surges, backbone ISPs are typically installing highly scalable caches at major peering points and network interconnections points such as NAPs.

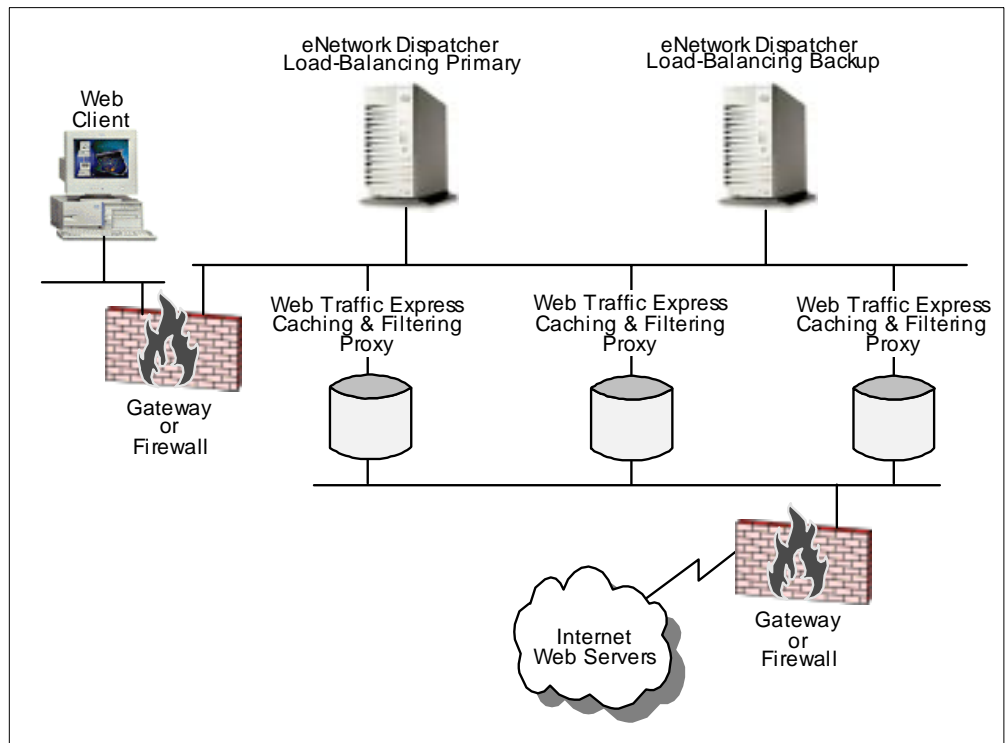


Figure 7. Hierarchical Caching

International ISPs in particular need caching to reduce the costs associated with trans-oceanic links. Such installations can make very effective use of the RCA feature of DWTE, a variation of the caching function that used together with load balancing and shared file storage can reduce redundancy of page storage.

Typical configurations for backbone ISPs would include load balancing for caching proxy servers. Because of the amount of traffic and the desire for high availability, such caching servers would likely use the RCA feature and would thus also be configured with AFS clients. Therefore there would also be an AFS server with the file content.

Access Internet Service Providers

Access ISPs need to provide both more consistent response time to their customers and to conserve their backbone network link and access charges. Thus caching at points of presence would address these needs. These configurations would be similar to those for backbone ISP solutions.

Because access ISPs target many of the small to medium-size business customers, there is also an opportunity for them to create revenue producing services, by deploying smaller caching devices on customer premises but configuring and managing them as an ISP service. For this scenario, the caching would look much as it does for corporate branch offices.

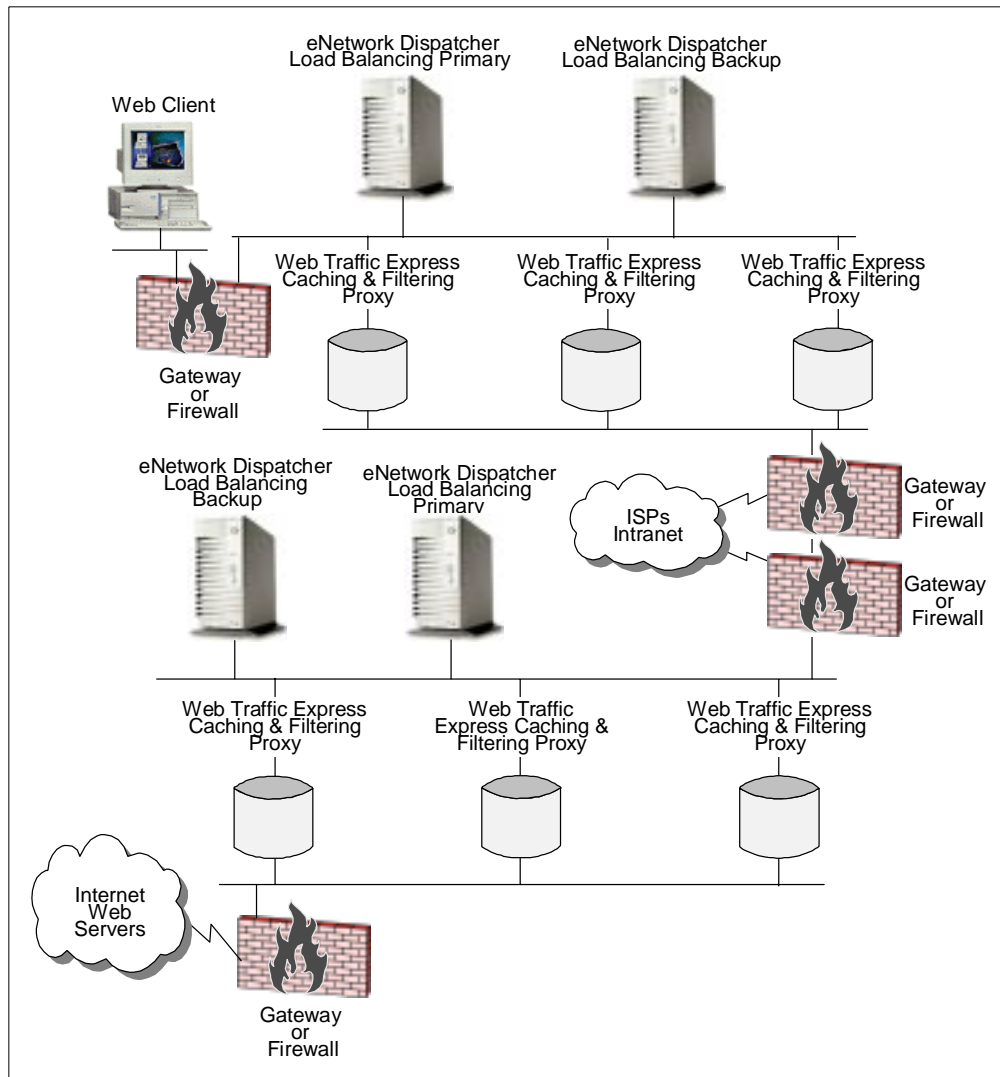


Figure 8. Access Internet Service Providers

Special Notices

This publication is intended to help IBM customers, system engineers and I/T architects to plan for and use the Caching and Filtering component of IBM WebSphere Performance Pack. The information in this publication is not intended as the specification of any programming interfaces that are provided by IBM WebSphere Performance Pack. See the PUBLICATIONS section of the IBM Programming Announcement for IBM WebSphere Performance Pack for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this publication was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

IBM ®	WebSphere
WebSphere Performance Pack	WebSphere Application Server
eNetwork Firewall	Distributed Web Traffic Express
ADSTAR	Deep Blue
SP2	AIX

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

Pentium, MMX, ProShare, LANDesk, and ActionMedia are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

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

Other company, product, and service names may be trademarks or service marks of others.

Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

International Technical Support Organization Publications

For information on ordering ITSO publications see “How to Get ITSO Redbooks” on page 29.

- *IBM WebSphere Performance Pack Usage and Administration*, SG24-5233

Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
Lotus Redbooks Collection	SBOF-6899	SK2T-8039
Tivoli Redbooks Collection	SBOF-6898	SK2T-8044
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
RS/6000 Redbooks Collection (PDF Format)	SBOF-8700	SK2T-8043
Application Development Redbooks Collection	SBOF-7290	SK2T-8037

Other Publications

This publication is also relevant as further information source:

- *Web Traffic Express for Multiplatforms User's Guide*, GC31-8645

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at <http://www.redbooks.ibm.com/>.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Redbooks Web Site on the World Wide Web**

<http://w3.itso.ibm.com/>

- **PUBORDER** – to order hardcopies in the United States
- **Tools Disks**

To get LIST3820s of redbooks, type one of the following commands:

```
TOOLCAT REDPRINT
TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
```

To get BookManager BOOKs of redbooks, type the following command:

```
TOOLCAT REDBOOKS
```

To get lists of redbooks, type the following command:

```
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
```

To register for information on workshops, residencies, and redbooks, type the following command:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1998
```

- **REDBOOKS Category on INEWS**
- **Online** – send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (<http://www.redbooks.ibm.com/redpieces.html>). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.