IBM @server

Redbooks Redpaper

Brian R. Smith
Gaia Banchelli
Monica Maria Echeverry
Axel Lachmann

A Feature Based Comparison Between HTTP Server (original) and HTTP Server (powered by Apache)

HTTP Server (original and powered by Apache) features

Both HTTP Server (original) and HTTP Server (powered by Apache) on the IBM @server iSeries Server have their own advantages, but which product should you choose? This section provides a functional comparison about these products to help answer this question.

Tip: It is the goal of IBM to eventually replace the HTTP Server (original) with the HTTP Server (powered by Apache). For long-term thinkers, HTTP Server (powered by Apache) is the better choice.

Note: The information found in this Redpaper was abstracted from the IBM Redbook *HTTP Server (powered by Apache): An Integrated Solution for IBM eServer iSeries Servers*, SG24-6716. For more details about the HTTP Server (powered by Apache) please refer to this redbook.

HTTP Version 1.1

Both original and Apache

Both products support HTTP Version 1.1. The HTTP protocol implementation in Apache was chiefly architected by one of the HTTP Version 1.1 authors. Most current versions of popular Web browsers support HTTP Version 1.1. Apache is normally configured to detect popular browsers that do not properly support HTTP Version 1.1, and use only HTTP Version 1.0.

GUI configuration and administration

Both original and Apache

You can configure and administrate HTTP server instances from Web browsers. To show the configuration and administration screen, type the following URL from a Web browser:

```
http://hostname:2001
```

Figure 1 and Figure 2 show the configuration pages for the HTTP Server (original) and HTTP Server (powered by Apache), respectively. The configuration pages are different between the HTTP Server (original) and the HTTP Server (powered by Apache), but you can reach each page from the same URL.

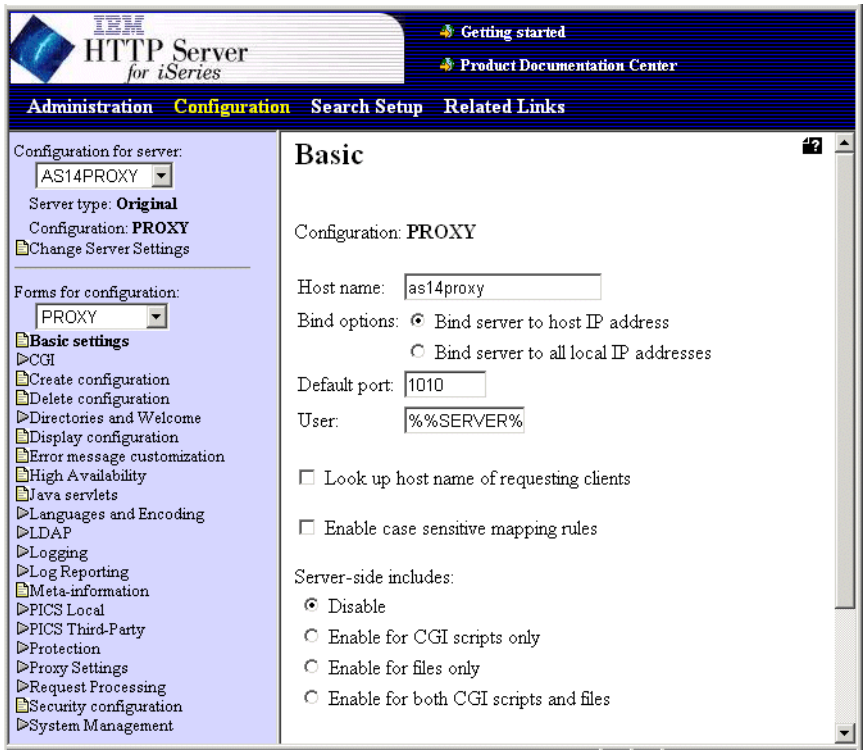

Figure 1 Original server configuration page

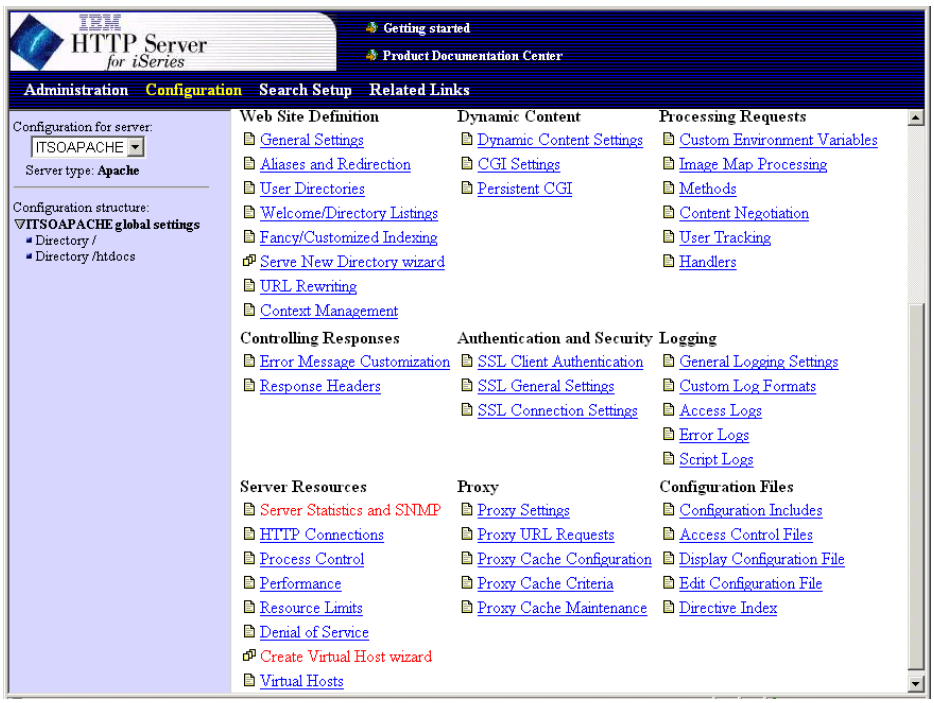

Figure 2 Apache server configuration page

The HTTP Server (original) and the HTTP Server (powered by Apache) can coexist. That is, you may have zero, one, or many original servers running at the same time you have zero, one, or many HTTP Server (powered by Apache) servers running. The administration screen allows you to create and manage HTTP servers. Figure 3 shows the GUI Administration page.

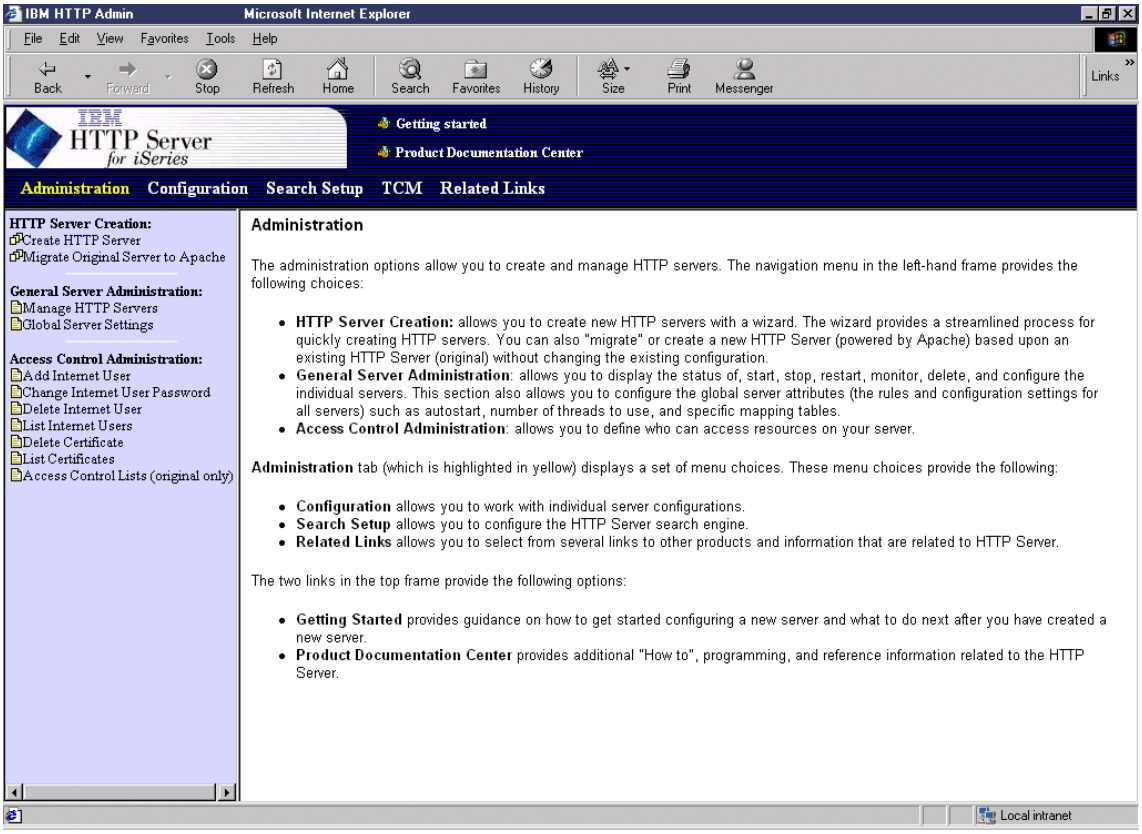

Figure 3 IBM HTTP Server configuration GUI

The navigation menu in the left-hand frame provides the following choices:

- **HTTP Server Creation**: Allows you to create new HTTP servers with a wizard. The wizard provides a streamlined process for quickly creating HTTP servers. You can also “migrate” or create a new HTTP Server (powered by Apache) based on an existing HTTP Server (original) without changing the existing configuration.
- **General Server Administration**: Allows you to display the status of, start, stop, restart, monitor, delete, and configure the individual servers. This section also allows you to configure the global server attributes (the rules and configuration settings for all servers) such as autostart, number of threads to use, and specific mapping tables.
- **Access Control Administration**: Allows you to define who can access resources on the server.

Persistent connections

Both original and Apache

When you enter a URL into your browser's address line or click a link on a Web page, you open a connection between your browser and the HTTP server. Prior to the availability of persistent connections, each file referenced on the Web page was retrieved using a separate connection. This type of retrieval is tremendously costly for the HTTP server and the network since overhead is required to establish and terminate each connection. Persistent connections are the default behavior for an HTTP server that implements the HTTP 1.1 protocol.

Virtual hosts

Both original and Apache

You can enable virtual hosting. This allows you to host any number of Web sites through one communications adapter. With virtual hosting, you do not need to assign a unique port to each Web site. Virtual hosting is useful if you need to provide multiple "top-level" URLs for your Web sites or if you provide ISP services to clients.

Dynamic virtual hosting

Only in Apache

The dynamic virtual host allows you to dynamically add Web sites (host names) by adding directories of content. This approach is based on automatically inserting the IP address and the contents of the Host: header into the pathname of the file that is used to satisfy the request.

Proxy caching

Both original and Apache

The IBM HTTP Server for iSeries can be configured as a non-caching or caching proxy server. When used as a non-caching proxy, the primary benefit of enabling proxy services is that the IP addresses used on the internal network are not sent out of your network. The proxy service forwards the request from your internal network using the IP address of the proxy server, not the address of the original requester. When the proxy server receives the response, it forwards the response to the original requester.

With caching enabled, the proxy server can act as a high-speed local store of previously accessed Web pages. When you configure the server as a proxy caching server, you can improve performance. You can also allow users of your internal network to access documents on the Internet. For example, if you frequently access the same set of Web pages from one or more sites, it may be advantageous to activate the caching feature. The retrieved Web page is stored locally on your iSeries server. Any subsequent accesses to the page occur at LAN speed, rather than at Internet speed.

Web pages can be encoded with a "no-cache" attribute or a specific expiration date. You can also configure the IBM HTTP Server for iSeries proxy service so that it periodically performs "garbage collection" to remove expired files from the cache.

Another use of the proxy service (with or without caching) is to log client requests. Some of the data available includes:

- Client IP address
- Date and time
- URL requested
- Byte count
- Success code

With this information, you can construct reports to account for the use of your Web site. For example, the information can be used in a charge-back system to understand and track marketing trends.

Local memory cache

Both original and Apache

A proxy cache is traditionally most beneficial to clients on your network since it lets you store files that were retrieved from other Web sites. You can provide a caching service for files on your site using the local memory cache configuration options.

To use a local memory cache, you identify an amount of memory to allocate and a set of files to be cached. When the IBM HTTP Server for iSeries is started, the files are read into the local memory cache up to the limit of the amount of memory allocated or the limit of the number of files that you allow to be cached.

When a request is received at your IBM HTTP Server for iSeries, the local memory cache is checked first to determine if it has a copy of the requested file. If so, the file is served from the cache, which is significantly faster than if the file is retrieved from disk storage.

Server-side includes

Both original and Apache

Server-side includes (SSI) enable the server to process some of the Web pages before the server sends the page to the client. The current date, size of the file, and the last change date of a file are examples of the kind of information that you can include in Web pages that you send to the client.

Tip: There is very little difference between the HTTP Server (original) and HTTP Server (powered by Apache) implementation of SSI. The reason? The Apache Software Foundation initially designed the syntax for SSI that IBM later used in its HTTP Server (original).

CGI programming

Both original and Apache

Corporations and other customers benefit from interacting with browser users by sending and receiving data. In the Web presence arena, this type of transaction is simple, such as collecting the name and address of a browser user who wants to receive a catalog. In general, these interactions start with a form – a Web page that contains input-capable fields and push buttons (like function keys). The server needs to hand the input from the form to a program for processing.

Typically, on the iSeries server (and most other platforms), this program is a CGI program. The CGI program receives the form data from the browser, accesses business data and business logic on the iSeries server, updates or stores information (if required by the transaction), and then builds the Web page that the HTTP server returns to the browser user in response.

CGI programs written for the HTTP Server (original) function the same way for the HTTP Server (powered by Apache).

LDAP support

Both original and Apache

The HTTP Server uses Lightweight Directory Access Protocol (LDAP) to store configuration information and user authentication information. LDAP integration allows you to use directory services for server configuration and authentication.

Webserver Search Engine

Both original and Apache

The HTTP Server search engine allows you to perform full text searches on HTML and text files stored in the iSeries file system from any Web browser. The iSeries Webserver Search Engine is available at no charge with IBM HTTP Server for iSeries (5769-DG1 or 5722-DG1) starting at OS/400 V4R4. You can control what options are available to the user and how the search results are displayed through customizable Net.Data macros.

Some of the features of the Webserver Search Engine include:

- Indexes documents for fast searching:

  The iSeries Webserver Search Engine indexes HTML or text files into a format that allows a large number of documents to be searched quickly. Multiple indexes can be created, and documents from multiple directories can be placed in a single index.

- Exact word indexing:

  The Webserver Search Engine uses an exact word indexing scheme rather than a keyword indexing scheme used by many search engines. All words are indexed; nothing is left out. Exact word indexing provides for faster index building and more precise searching than keyword indexing, but requires additional disk space. Documents are searched using consecutive character matching, which is essential for proper support of double-byte languages.

- Advanced search functions:

  The iSeries Webserver Search Engine supports advanced search capabilities such as exact search, fuzzy search, wild card search, proximity search, English word stemming, case-sensitive search, boolean search, and document ranking.

- Customizable search forms:

  The search forms and search results form are completely customizable by the end user using the Net.Data scripting language. This gives the user the ability to specify the type of search to be done and how the results are to be displayed. The information that can optionally be displayed on the results page includes the number of documents satisfying the search, number of occurrences of the search term, number of documents returned on this page, the URL associated with each document, the document's ranking, and the last modified date and size. Any and all of this information can be displayed however the user chooses.

- Web-based administration:

  Administration of the search indexes is handled as part of the IBM HTTP Server Configuration and Administration Web pages. The search administration forms allow you to create and delete search indexes, update search indexes when documents are modified, and view the status of an index.

- Multiple language support:

  The Webserver Search Engine supports multiple national languages including double-byte languages Chinese, Japanese, and Korean.

On the iSeries, the search engine comes in two logical pieces that are related to each other.

Web-based Distributed Authoring and Versioning (WebDAV)

Both original and Apache

WebDAV provides a network protocol for creating interoperable, collaborative applications. Major features of the protocol include:

- Locking (concurrency control):

Long-duration exclusive and shared write locks prevent the problem of overwriting, where two or more collaborators write to the same resource without first merging changes. To achieve robust Internet-scale collaboration, where network connections may be disconnected arbitrarily, and for scalability, since each open connection consumes server resources, the duration of DAV locks is independent of any individual network connection.

- Properties:

  XML properties provide storage for arbitrary metadata, such as a list of authors on Web resources. These properties can be efficiently set, deleted, and retrieved using the DAV protocol. The DAV Searching and Locating (DASL) protocol provides searches based on property values to locate Web resources.

- Namespace manipulation:

  Since resources may need to be copied or moved as a Web site evolves, DAV supports copy and move operations. Collections, similar to file system directories, may be created and listed.

For more information about WebDAV, refer to: http://www.webdav.org/

Access log reporting and Web usage mining

Only in original

The HTTP Server (original) provides the log reporting and Web usage mining function. If you are using HTTP Server (powered by Apache), you can obtain the IBM WebSphere Site Analyzer to provide a similar function.

Platform for Internet Content Selection (PICS)

Only in original

PICS support enables labels (metadata) to be associated with Internet content. Originally designed to help parents and teachers control what children access on the Internet, it also facilitates other uses for labels, including code signing and privacy.

Tip: PICS never found wide spread support in the industry. This is why you will not find this support on the HTTP Server (powered by Apache) and other Apache servers.

Domino plug-in

Only in original

The Domino plug-in allows the HTTP server to access documents stored in Notes. See *Domino and WebSphere Integration on the IBM @server iSeries Server*, SG24-6223, for details in regards to the HTTP Server (original) only.

Tip: Any announcements of future Domino for iSeries and HTTP Server (powered by Apache) integration will be made at:
http://www.ibm.com/servers/eserver/iseries/domino/

WebSphere Application Server plug-in

Both original and Apache

The IBM HTTP Server for iSeries handles static content, CGI program invocations, and proprietary plug-ins. The run-time environment (WebSphere Application Server) plugs into IBM HTTP Server for iSeries using plug-in APIs. This extends the support of the HTTP Server to include an implementation of the Java 2 Platform Enterprise Edition (J2EE) specification from SUN Microsystems.

Apache Software Foundation’s Jakarta Tomcat

Only in Apache

The HTTP Server (powered by Apache) includes an industry-standard Java servlet and JavaServer Pages (JSP) engine based on technology from the Apache Software Foundation's Jakarta Tomcat open source code base. Lightweight and easy-to-use software extends the HTTP Server (powered by Apache) server and is compliant with the Java Servlet 2.2 and JavaServer Pages 1.1 specifications from SUN Microsystems.

Apache Software Foundation's Jakarta Tomcat for iSeries support can be used as a simple starting point for business partners and customers interested in learning about or piloting Java servlet and JSP applications.

Original Server API

Only in original

HTTP Server (original) APIs are not supported on HTTP Server (powered by Apache). The strategic direction of IBM is to extend the function of the Web server using Java servlets rather than with modules or server APIs. This function can only be used in HTTP Server (original).

Apache Portable Runtime (APR)

Only in Apache

The design of the Apache HTTP server is one that defines *modules*. Modules are operating system objects that can be dynamically linked and loaded to extend the nature of the Apache HTTP server. Depending on the operating system, this is similar to:

- Window's Dynamic Link Libraries (DLL)
- UNIX's shared object libraries
- OS/400's ILE Service Programs

In this way, the Apache modules provide a way to extend a server's function. Functions commonly added by optional modules include:

- Authentication
- Encryption
- Application support
- Logging
- Support for different content types
- Diagnostics

Support for the TRCTCPAPP command

Only in Apache

The Trace TCP/IP Application (TRCTCPAPP) command can be used to trace the server, but only one instance at a time. It can be started while the server is running.

Note: The old -vv (very verbose) still works at startup much like the original server (and -vi and -ve, which stand for informational and error tracing, respectively). The Dump User Trace (DMPUSRTRC) and Display Physical File Member (DSPPFM) commands can be used to see the results, but TRCTCPAPP is the suggested trace method.

In addition to these functions, the following functions are provided only in HTTP Server (powered by Apache):

- *Headers control*: Has the ability to control headers. It has the ability to control expires and other headers.
- *SSL*: This is more specific customization of SSL

- *CGI*:
  - Controls the number of CGI jobs started with the server and their user profile
  - OS/400 PASE CGI programs (UNIX binaries)
- More customization of directory listings
- Automatic restart of multi-threaded child job monitored by parent job
- Configuration file support in threadsafe IFS file systems (not just QSYS.LIB)

Triggered Cache Manager (TCM)

Both original and Apache

Triggered Cache Manager provides a mechanism to manage dynamically-generated Web pages. TCM is a separate server that can be used in conjunction with the HTTP Server to allow a Web designer to build dynamic pages. It only updates the cache when the underlying data changes, thereby improving the performance of a Web site.

Highly available HTTP server

Both original and Apache

If Web serving is a critical aspect of your business, you may want high availability for your Web server environment. A highly available Web server takes advantage of iSeries clustering technology and makes it possible to build a highly available Web site. This improves the availability of business-critical Web applications built with Common Gateway Interface (CGI) programs.

References

For more information, refer to the IBM HTTP Server for iSeries Web site at (which contains many components including the HTTP Server (original) and the HTTP Server (powered by Apache)): http://www.ibm.com/servers/eserver/iseries/software/http/

For greater details about the HTTP Server (powered by Apache) see the IBM Redbook *HTTP Server (powered by Apache): An Integrated Solution for IBM eServer iSeries Servers*, SG24-6716.

This document created or updated on April 11, 2002.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. JLU Building 107-2
3605 Highway 52N
Rochester, Minnesota 55901-7829

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.

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 document 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, North Castle Drive, Armonk, NY 10504-1785.

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 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.

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

e (logo)®	Net.Data®	Notes®
Redbooks (logo)™	OS/400®	Domino™
IBM®	SP™	
iSeries™	WebSphere®	

Other company trademarks

The following terms are trademarks of other companies:
C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

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