

COMPARATIVE ANALYSIS OF MODERN WEB SERVERS

BY

AYESANMI OWOLABI TEMITOPE

(RUN/CMP/24/17883)

COMPUTER SCIENCE DEPARTMENT

COLLEGE OF POSTGRADUATE STUDIES

REDEEMER'S UNIVERSITY

EDE, OSUN STATE, NIGERIA

CSC 828 ASSIGNMENTS

LECTURER-IN-CHARGE

DR. S.A. ADEPOJU

1.0. INTRODUCTION

Web servers form the backbone of internet infrastructure, serving content to billions of users daily. This paper provides a comprehensive comparison of five prominent web servers: Apache HTTP Server, Nginx, Microsoft IIS, LiteSpeed, and Apache Tomcat. Each server has distinct architectural approaches, performance characteristics, and ideal use cases that make them suitable for different web hosting scenarios. Understanding these differences is crucial for system administrators and developers when selecting the appropriate web server for their specific needs.

2.0. ARCHITECTURE AND PROCESSING MODEL

2.1. Apache HTTP Server

The Apache HTTP Server utilizes a modular architecture with Multi-Processing Modules (MPMs) that determine its connection handling approach (The Apache Software Foundation, 2023). The traditional prefork MPM uses multiple child processes with single threads, while the worker MPM employs a hybrid multi-process/multi-threaded model. The event MPM, introduced in version 2.4, implements an asynchronous, event-driven model that improves performance for keep-alive connections. This flexibility allows Apache to adapt to various workloads but requires careful tuning for optimal performance.

2.2. Nginx

Nginx employs an event-driven, asynchronous architecture that differs fundamentally from Apache's process-based model (Nginx, Inc., 2023). Instead of creating new processes or threads for each connection, Nginx uses a single master process that manages multiple worker processes. Each worker handles thousands of connections simultaneously through an efficient event loop. This design makes Nginx particularly resource-efficient for serving static content and handling numerous concurrent connections.

2.3. Microsoft IIS

Microsoft Internet Information Services (IIS) follows a thread-pool based architecture tightly integrated with the Windows operating system (Microsoft, 2023). IIS leverages the Windows kernel-mode HTTP.sys driver for request processing and caching, which provides kernel-level caching and request queuing. The user-mode worker processes (w3wp.exe) handle application requests, with each application pool running in isolation. This architecture is optimized for Windows environments but lacks cross-platform compatibility.

2.4. LiteSpeed

LiteSpeed combines an event-driven architecture similar to Nginx with Apache compatibility features (LiteSpeed Technologies, 2023). Its event-driven design allows it to handle thousands of concurrent connections with low memory usage. LiteSpeed's unique approach includes a queuing mechanism that prevents worker processes from being overwhelmed during traffic spikes. The

server maintains compatibility with Apache configurations, including .htaccess files and mod_rewrite rules, easing migration from Apache.

2.5. Apache Tomcat

Apache Tomcat implements a Java-centric architecture designed specifically for serving Java-based web applications (The Apache Software Foundation, 2023). Unlike general-purpose web servers, Tomcat includes a servlet container that implements the Java Servlet and JavaServer Pages (JSP) specifications. Its connector architecture allows different implementations for handling HTTP requests, with the NIO (Non-blocking I/O) connector providing the best performance for concurrent connections.

3.0. PERFORMANCE AND SCALABILITY

Performance characteristics vary significantly among these web servers, particularly in their handling of static versus dynamic content and their behavior under heavy loads.

3.1. Static Content Performance

Nginx and LiteSpeed demonstrate superior performance for static content delivery. In benchmark tests, Nginx served approximately 2.5 times more static requests per second than Apache when using the event MPM (TechEmpower, 2023). LiteSpeed's performance is comparable to Nginx for static content, with the added benefit of Apache configuration compatibility. Microsoft IIS shows strong performance for static content on Windows platforms, though it generally trails Nginx in cross-platform benchmarks.

3.2. Dynamic Content Handling

For dynamic content, the performance landscape changes. Apache's `mod_php` integration provides excellent performance for PHP applications, though at the cost of higher memory usage per process. Nginx typically requires FastCGI Process Manager (PHP-FPM) for PHP processing, which adds some overhead but maintains better stability under load. IIS demonstrates exceptional performance for ASP.NET applications due to its tight integration with the Windows web stack. Tomcat naturally excels with Java servlets but performs poorly for non-Java content.

3.3. Scalability Under Concurrent Loads

In high-concurrency scenarios, Nginx and LiteSpeed maintain consistent performance while Apache's process-based models may struggle. Tests show Nginx handling over 10,000 concurrent connections with minimal resource impact, while Apache's memory usage grows linearly with connections (Netcraft, 2023). IIS scales well within Windows environments but requires careful application pool configuration. Tomcat's performance depends heavily on JVM tuning and the specific Java application being served.

4.0. FEATURES AND MODULES

4.1. Apache HTTP Server

Apache boasts an extensive module ecosystem with over 70 core modules and countless third-party extensions (The Apache Software Foundation, 2023). Key modules include `mod_rewrite` for URL manipulation, `mod_security` for web application firewall capabilities, and `mod_ssl` for HTTPS support. The `.htaccess` system allows directory-level configuration, providing flexibility at the cost of some performance overhead. Apache's modularity makes it highly customizable but can lead to configuration complexity.

4.2. Nginx

Nginx emphasizes essential functionality with fewer but more optimized modules (Nginx, Inc., 2023). Its core features include reverse proxying, load balancing, and FastCGI support. Nginx's configuration syntax is generally considered more consistent than Apache's, though less flexible in some scenarios. The lack of `.htaccess` equivalent means all configuration must be handled at the server level, improving security and performance but reducing per-directory flexibility.

4.3. Microsoft IIS

IIS provides deep integration with Microsoft technologies like ASP.NET, MS SQL, and Windows Authentication (Microsoft, 2023). Its graphical management console simplifies configuration for Windows administrators. IIS supports advanced features like application

request routing, dynamic IP restrictions, and centralized SSL certificate management. The Web Platform Installer simplifies adding common web applications and components.

4.4. LiteSpeed

LiteSpeed offers unique features like built-in page caching (LSCache) and HTTP/3 support (LiteSpeed Technologies, 2023). Its Apache-compatible configuration system allows easy migration from existing Apache setups. LiteSpeed WebADC provides load balancing capabilities, and the server includes built-in anti-DDoS features. The commercial version offers GUI management and WordPress acceleration features.

4.5. Apache Tomcat

Tomcat focuses specifically on Java servlet and JSP support (The Apache Software Foundation, 2023). It includes connectors for linking with web servers like Apache and Nginx, clustering capabilities for high availability, and JMX-based monitoring. Tomcat lacks many features expected in general-purpose web servers but excels in its specialized role as a Java application server.

5.0. COMPATIBILITY AND PLATFORM SUPPORT

Platform compatibility significantly influences web server selection, particularly in heterogeneous environments.

5.1. Cross-Platform Support

Apache and Nginx lead in cross-platform compatibility, running natively on Windows, Linux, and macOS (W3Techs, 2023). Apache has slightly broader Unix-like system support, while Nginx performs better on modern Linux kernels. LiteSpeed is available for Linux and FreeBSD, with Windows support in the enterprise version. IIS is exclusively Windows-based, while Tomcat runs anywhere Java is supported.

5.2. Language and Framework Support

Apache supports the widest range of programming languages through CGI, FastCGI, and modules like `mod_php` and `mod_wsgi`. Nginx relies more on external processors for dynamic content. IIS naturally supports Microsoft technologies best, though PHP and other languages can run via FastCGI. Tomcat exclusively serves Java applications, often requiring a front-end web server for static content.

5.3. Cloud and Container Integration

Nginx leads in cloud-native adoption, with strong Kubernetes integration and a microservices-friendly architecture (TechEmpower, 2023). Apache remains popular in traditional hosting environments. IIS integrates seamlessly with Azure services, while Tomcat is commonly deployed in Java-oriented cloud environments. LiteSpeed offers specific integrations with cPanel and CloudLinux for shared hosting providers.

6.0. USE CASE SUITABILITY

6.1. High-Traffic Static Content

For content delivery networks and static-heavy sites, Nginx is the preferred choice due to its efficient event-driven architecture. LiteSpeed provides comparable performance with easier Apache migration paths. Apache can serve static content adequately but requires more resources at scale.

6.2. Enterprise Web Applications

IIS dominates for Windows-based enterprise applications using ASP.NET and other Microsoft technologies. Its tight Active Directory integration and management tools suit corporate environments. For Java enterprises, Tomcat remains the standard application server.

6.3. Dynamic Web Applications

Apache's flexibility makes it ideal for shared hosting environments with diverse applications. PHP-based sites often perform best on Apache or LiteSpeed. Nginx paired with PHP-FPM works well for high-traffic PHP sites. Python and Ruby applications typically deploy behind Nginx with uWSGI or similar interfaces.

6.4. Microservices and API Gateways

Nginx's lightweight footprint and reverse proxy capabilities make it ideal for microservices architectures. Its advanced load balancing and service discovery integrations support modern distributed systems. Apache can fill this role but with greater resource overhead.

7.0. MERIT AND DEMERIT IN SUMMARY

SERVER	MERIT	DEMERIT
Apache HTTP	Extremely flexible configuration; Mature and stable with extensive documentation; Broad module ecosystem; .htaccess per-directory configuration	Resource-intensive under high load; Complex configuration syntax; Performance lags behind event-driven servers
Nginx	Excellent performance and scalability; Low memory footprint; Strong reverse proxy capabilities; Simple, consistent configuration	Less flexible for dynamic content; Smaller module ecosystem; No per-directory configuration
Microsoft IIS	Excellent Windows integration; Graphical management tools; Strong ASP.NET performance; Enterprise-grade features	Windows platform lock-in; Licensing costs for commercial use; Less community support than open-source options
LiteSpeed	Apache compatibility with better performance;	Commercial licensing for full features;

	Built-in caching and optimization; HTTP/3 support; Good security features	Smaller community than Apache/Nginx Primarily Linux-focused
Apache Tomcat	Optimal Java application support; Lightweight for servlet/JSP workloads; Good clustering capabilities; Pure Java implementation	Poor static file performance; Limited to Java ecosystem; Requires front-end server for production

8.0. CONCLUSION AND RECOMMENDATIONS

Selecting the appropriate web server depends on specific application requirements, technical environment, and performance needs. For general-purpose web hosting with mixed content types, Apache remains a robust choice due to its flexibility and extensive feature set. High-traffic sites serving primarily static content should prioritize Nginx or LiteSpeed for their superior performance and scalability.

Windows-centric organizations running ASP.NET applications will find IIS delivers the best integration and performance. Java applications naturally belong on Tomcat, though often behind an Nginx or Apache reverse proxy for static content and SSL termination.

Emerging architectures like microservices and API gateways benefit from Nginx's lightweight footprint and advanced proxying capabilities. LiteSpeed presents an attractive alternative for Apache users seeking performance improvements without configuration changes.

Ultimately, many production environments benefit from combining these technologies - using Nginx as a reverse proxy front-end with Apache or application servers handling dynamic content. This layered approach leverages each server's strengths while mitigating their weaknesses.

REFERENCES

The Apache Software Foundation. (2023). *Apache HTTP Server documentation*. <https://httpd.apache.org/docs/>

Nginx, Inc. (2023). *Nginx documentation*. <https://nginx.org/en/docs/>

Microsoft. (2023). *Internet Information Services (IIS) documentation*. <https://docs.microsoft.com/en-us/iis/>

LiteSpeed Technologies. (2023). *LiteSpeed Web Server documentation*. <https://www.litespeedtech.com/docs/>

The Apache Software Foundation. (2023). *Apache Tomcat documentation*. <https://tomcat.apache.org/tomcat-10.1-doc/>

TechEmpower. (2023). *Web Framework Benchmarks*. <https://www.techempower.com/benchmarks/>

Netcraft. (2023). *Web Server Survey*. <https://news.netcraft.com/archives/category/web-server-survey/>

W3Techs. (2023). *Usage statistics of web servers*. https://w3techs.com/technologies/overview/web_server