Server programming languages have grown greatly over time, with PHP maintaining a dominant position in online development. According to recent estimates, PHP powers about 77.4% of websites using recognized server-side programming languages (W3Techs, 2025). This remarkable prevalence stems from its ease of use, extensive community support, and rich ecosystem of frameworks and tools.

Many prominent websites and platforms rely on PHP as their backbone. Facebook, which was first constructed solely in PHP, has since grown to employ a variety of technologies while still retaining PHP components. WordPress, which runs more than 43% of all websites on the internet, was built mostly with PHP. Other significant platforms include Drupal, Joomla, and MediaWiki (the software that powers Wikipedia). These platforms demonstrate PHP's scalability and reliability in handling large-scale applications.

The advanced features of PHP have expanded considerably since its inception. Modern PHP frameworks, such as Laravel and Symfony, have introduced advanced architectural patterns and development methodologies. PHP's object-oriented programming has improved, with features like characteristics, namespaces, and interfaces allowing for better code organization and reuse. According to Taylor (2024), PHP's type system has substantially improved with the addition of strict typing, return type declarations, and union types, making code more dependable and manageable.

Memory management in PHP has become more efficient through optimizations in the Zend Engine. The language now supports asynchronous programming through extensions like ReactPHP and Swoole, enabling developers to handle concurrent connections more effectively. Built-in support for package management through Composer has revolutionized dependency handling and project organization.

Security features have also seen substantial improvements. PHP now includes built-in protection against common web vulnerabilities, such as cross-site scripting (XSS) and SQL injection attacks. The password hashing API provides secure password handling out of the box, while the Filter extension offers robust data validation and sanitization capabilities.

Database integration in PHP has evolved beyond basic MySQL support. The PDO (PHP Data Objects) extension provides a uniform interface for accessing numerous database systems, whereas contemporary ORMs (Object-Relational Mappers) such as Doctrine provide advanced database abstraction layers.

PHP's performance capabilities have been boosted by features such as OpCache, which caches precompiled script bytecode in memory. Just-In-Time (JIT) compilation, which was introduced in PHP 8.0, has significantly improved execution speed for computationally expensive activities.

Web service integration has become more streamlined with built-in support for JSON, XML, and SOAP protocols. RESTful API development is facilitated through modern frameworks that provide elegant routing and middleware systems.

References:

Taylor, F. (2024, November 24). Modern PHP Development in 2024 - Frederick Taylor - Medium. *Medium*. https://medium.com/@Fredtaylor1/modern-php-development-in-2024-77236ac85f28

W3Techs. (2025). Usage statistics of server-side programming languages for websites. Retrieved from https://w3techs.com/technologies/overview/programming_language