



How Migrating a PHP Application to .NET Solves Business Critical Issues



AMain Performance Hobbies is one of the leading online retailers selling RC and cycling products in the United States. Founded in 2004, the company quickly grew and is now considered an industry leader.

Industry

Electronics

Key Challenges

Delays, crashes, security breaches, no agile development

“PHP has a lot of value due to the enormous amount of code out there written in it, but it has a lot of weaknesses, as well.”

- Kendall Bennett, CEO
AMain Hobbies

Background

The amain.com site originally ran on an open source project called **osCommerce**, which was written in PHP, and a MySQL database on the back end. As the company grew in popularity and the number of users increased drastically, AMain CEO Kendall Bennett began to see a number of issues arising from the complicated system architecture.

Business Challenge

Using a fairly complex mix of the Zend Framework, XML-RPC as the communication mechanism and PHP on the back end, amain.com ran into a few troubles:

- ❗ There was a significant **delay** in the information displayed on the company’s desktop application compared to viewing the same data in the browser. It turned out that the information was gathered from the database and put into a PHP array in a fraction of a second, but the Zend Framework took a disproportionately longer time to turn these PHP arrays into XML packets.
- ❗ As soon as a particular concurrent user load was reached (Kendall Bennett estimated it to be about 5000), the entire site started to **slow down** and parts of it were **crashing**. This, of course, has a severely detrimental impact on any e-shop.
- ❗ After deciding to switch to C# from C++, amain.com immediately noticed that the **PHP/MySQL architecture did not run as smoothly** as expected on Windows.
- ❗ In spite of the well-deserved popularity of PHP, the language is **vulnerable** due to the source code being readily available to anyone. Amain.com experienced this first hand when a vulnerability in an extension was **used by hackers to break into the system** and dump code in to get full access to the company’s database.

Solution

Amain.com decided to solve the issues by compiling their PHP code to .NET using **Phalanger**. This provided multiple advantages for the company:

- 🌀 **Agile development:** by sequentially porting subsets of the application from PHP to .NET rather than rewriting the entire code base, amain.com could gradually shift to .NET without ever shutting down its business-critical checkout process pages in PHP.
- 🌀 **Performance increase:** compiling the PHP code to .NET resulted in a significant increase of the application's performance.
- 🌀 **Security:** the migration process enabled amain.com to compile their source code into a .DLL file, preventing hackers from exploiting vulnerabilities in the PHP code.
- 🌀 **Interoperability:** thanks to the architecture of the solution, the PHP code was both-way interoperable with .NET, offering amain.com the option of flexibility and agile development. Parts could be re-implemented in .NET, while other parts could remain in PHP, all functioning within the same ecosystem.

Business Impact

The benefits of compiling PHP code to .NET went beyond mere programming aspects. The migration had an extensive impact on the company's business operations, as well.

The site's performance increase meant that more concurrent users could be served without crashes and the customers had a more comfortable experience. The site could now perform at **two times the speed**.

Amain.com was able to move away from osCommerce in an agile flexible manner, which would otherwise have required several years of work by multiple developers. By compiling to .NET, amain.com therefore saved **\$500,000-1,000,000** in development costs.

Finally, business continuity was ensured through a considerably **more secure website**. The source code is compiled and obfuscated, preventing hackers from exploiting vulnerabilities in PHP.

2x
faster
site

\$500k-
1M
saved

Agile
develop-
ment

Improved
security