



# blackfire.io

## Case Study

# WPML adopting Blackfire.io for consistent performance results



## Introduction by Blackfire.io

This case study has been written by Dimitris Mitsis. He's QA Leader at WPML, a multilingual WordPress plugin built by OnTheGoSystems. OTGS is a 100% remote company, working in 39 different countries. It creates some of the world's most popular WordPress plugins such as WPML and Toolset.

Dimitris and his team adopted Blackfire Enterprise in 2016. It's part of their development and continuous integration workflow. This includes performance of the company websites, and over 800,000 sites using OnTheGoSystems' plugins.

Blackfire has been a key part of every release over the last few years. It's built into their CI, and unit, integration and acceptance tests.

## Achieving Consistent Measurements

Performance improvements are an important target for us. We pledged to our clients that we'd help with any performance problems they encountered. So performance issues reported to our support teams receive top priority. We're even offering a 12-month money back performance guarantee for WPML. That's how seriously we take the matter.

Prior to Blackfire we struggled for consistent performance-related data from our tools. This made it difficult, if not impossible, to meet our performance targets. We were never quite sure if we were hitting them or not.

Our metrics vary, as we monitor pages on both the front and back end of WordPress. These have very different types and amounts of content. So unreliable results from our performance management tools wasn't acceptable. You can't promise great performance if you don't know what results you're getting.

This meant finding a tool that would give us reliable data. We began by testing the Blackfire trial, to see if it could remove this obstacle to our performance commitments.

## Blackfire and GitLab

Our first task was adding Blackfire to our existing workflow.

We've been using GitLab and the GitLab Runner for almost five years. It hosts our codebase and continuous integration. At the time, Gitlab was free even for private repositories. We also liked that we could host the server ourselves. We can run everything from the same environment as our codebase.

So sticking with GitLab was a necessity for us. But Blackfire wasn't natively compatible with GitLab.

The Blackfire support team was incredibly helpful. They

worked with us to integrate the two platforms. This provided the automated performance testing we needed, using the tools we wanted. It allowed us to begin building better performing code from an earlier stage in the cycle.

GitLab integration is now officially supported by the Blackfire team.

## Blackfire on our own environment

In heavy-use cases, like a multilingual e-commerce store, there can be tens of thousands of strings to translate. This can have a real effect on a site's performance.

We've been using Blackfire for new versions of our multilingual WordPress plugins.

Our first test is to run beta versions of the plugins on our own production sites. We always do this before making them available publicly. Our sites are quite

*« During our first encounter with Blackfire we were really impressed by the UI. It's easy to identify unneeded calls and queries that can be improved or cached. »*

complex, have a lot of content in multiple languages and receive a lot of traffic.

We monitor this first stage of our performance test carefully. It's where we look to find the first signs of possible performance issues. We began using Blackfire at this stage to get early insights into how well our plugins work in a live environment.

It now helps us to scope out any potential bottlenecks long before our products are made publicly available.

## Development and support

Naturally we make use of it in the same way when providing customer support.

We set up Blackfire to run every day in our continuous integration platform on GitLab. This simulates client setups using our daily development builds. It's become a very useful tool for our supporters, when dealing with clients' reported performance issues.

They will regularly clone a client's WordPress site to troubleshoot issues on local environments with Blackfire. This lets us fault find performance problems and pinpoint any signs of congestion on sites built by other people.

It's a great way to find systemic issues, but also to drill down and find rare issues in fringe cases.

## Automation and Overhead

During our first encounter with Blackfire we were really impressed by the UI. It's easy to identify unneeded calls and queries that can be improved or cached. The filtering abilities, the call graph and the timeline have made our lives a lot easier. We've even been using those tools for local development, before merging



### Feature Focus

Blackfire is a unique SaaS-delivered Performance Management Solution; it allows your development and DevOps teams to measure, compare, and improve performance at every step of the application development lifecycle. It is built for use in production, staging, test pipelines and development machines.



### Feature Focus

Blackfire supports integrations with GitLab, GitHub and Bibucket. Check the many ways of automating profiling, and receiving notifications when results are available.



### Feature Focus

Blackfire's core technology a profiler. It measures how PHP code consumes resources, and enables to visualize how it is executed. Its graphs highlight hot paths and bottlenecks in the code in order to find the root cause. Profile comparisons make it possible for developers to verify the impact of their code iterations.



## Feature Focus

Profiling with Blackfire is on-demand. This means that Blackfire adds no overhead for your end users, which makes it safe to use in production.



## Feature Focus

Blackfire Player is a powerful Web Crawling, Web Testing, and Web Scraper application. It provides a nice DSL to crawl HTTP services, assert responses, and extract data from HTML/XML/JSON responses. Combined with webhooks and native integrations, it enables to automatically profile and test code performance in any testing pipeline, and in production.



## Market Insight

WordPress is not just for blogging or corporate websites. WooCommerce powers 1/4th of all eCommerce sites according to BuiltWith

<https://trends.builtwith.com/shop>

*« Using the Blackfire player we can even test pages that require a login. Zero overhead when profiling production websites was another important factor. »*

changes in our main development branches.

A deciding factor in becoming an Enterprise customer was the automation tools. These allow us to easily measure loading times, queries, execution times and memory use. This works both in the front-end of WordPress and in the admin. Using the Blackfire player we can even test pages that require a login. Zero overhead when profiling production websites was another important factor.

## Blackfire's results

Combined, these functions gave us the foundation we needed to understand WPML's performance. Getting consistent results from the tools we were using had been problematic. We'd lacked this grounding until this point, and were always working with a degree of uncertainty.

Loading time isn't the most stable factor to quantify performance, either. It's too dependent on external

factors; networking, server resources, physical location, even bad weather! We prefer to monitor and improve the number of queries as it's a more absolute comparison. Blackfire is useful in helping us profile this essential factor.

So you can see how it's taken care of the problem that first prompted us to trial Blackfire. It takes multiple profiles and calculates the average, so our results are now logical and reliable. It gives us the baseline we need to know whether performance has improved, stayed the same, or dropped.

We've achieved approximately 55% improvement since we started using Blackfire in the WordPress admin pages. And we've seen a 50% improvement on the front-end. That's comparing WPML 3.5 (the first version where we integrated Blackfire) to one of our latest releases, WPML 4.2.8.

Other major performance improvements are also built into the new 4.3.0 version of WPML.

## Example 1

A WPML client had a website with 13 duplicated languages, and six translated languages. When saving pages it took too long due to the logic for the duplication synchronization. CloudFlare wouldn't wait long enough for the server to respond, causing an error.

The client was operating a mid-sized WooCommerce site. Blackfire helped us to identify bottlenecks caused by database calls. We refactored the to avoid around 20,000 database requests per product update. A cache and indexes were also added. This reduced page loading times by around 40% and eliminated the error.

## Example 2

We noticed a performance issue when WPML and Query Monitor were both in use. This plugin measures a site's performance from the WordPress admin.

Query Monitor reported slow loading times for some WPML clients. Blackfire identified numerous database calls being made for each taxonomy. This wasn't a big issue, and we reduced the number of calls by over 70 percent.

But Query Monitor compounded the problem on the client's site. Its increased activity (caused by the database calls) further reduced performance.

By fixing the first issue identified by Blackfire, the second issue took care of itself.

## Example 3

A WPML-translated site used the popular WooCommerce Bookings plugin. Adding a booking to the shopping cart was taking a long time to complete.

Blackfire helped to highlight that there was no check for hierarchical post types. New bookings were being checked across all posts and parents to validate the hierarchy. This was unnecessary and caused delays of up to 30 seconds.

By pinpointing and fixing this process, Blackfire helped us to reduce page loading times by 90%. We removed 17,000 database requests when WPML and WooCommerce Bookings were used together.

## Conclusion

The support provided by Blackfire has been invaluable. We now get the measurable results we

need. We've been provided with first-rate assistance on our requests.

Combined with our innovative changes in WPML 4.3.0, Blackfire has helped our performance improve by leaps and bounds. Our plugin has never been faster or more efficient. Blackfire has become an essential aspect in all our testing and development processes.

## About OnTheGoSystems

OnTheGoSystems has been developing WordPress plugins since 2007 and now has a team of approximately 100 people distributed all over the globe.

## Contact Blackfire

[blackfire.io](https://blackfire.io)

[customer.success@blackfire.io](mailto:customer.success@blackfire.io)

[@blackfireio](https://twitter.com/blackfireio)

---

## Contact WPML

WPML team

[hello@wpml.org](mailto:hello@wpml.org)