ProGet

Case Study

Here's How Symbotic Uses ProGet to Eliminate Its Reliance on Public Repositories and Maintain Flawless Load Balancing Across All Its Infrastructure

☑ **Zero Down-time**

☑ **Zero Reliance on External Feeds & Sites**

☑ **100% Source Control Between On- and Off-site Teams**

☑ **100% Load Balancing Success**

☑ **24/7 Builds with Zero Latency**

The Challenge

Customers rely on Symbotic to streamline their distribution centers and increase speed-to-shelf time through robotics. Symbotic's internal IT systems therefore need to stay fully functional at all times. However, they relied on popular, third-party, public repositories like NuGet, which can (often) crash under the stress. Every time that happened, employees were forced to completely stop working to reduce the load. And because Symbotic relies on its IT team for builds 24/7, any downtime whatsoever hurt the bottom line.

In addition to downtime concerns, Symbotic also found sharing binaries using source control or file share to manage dependencies slowed its productivity. Having to check in to source control is a manual process that, when used in a largely automated system, reduces the speed and safety benefits of automation. Furthermore, having to manage dependencies in source control meant Symbotic's source control was performing more work than it was designed to do, increasing disk usage and reducing productivity.

Finally, Symbotic also kept running into integration problems that blocked maximum build efficiency and made the lives of their Development team difficult. For example, Symbotic needed a Helm feed integration to do their work with Docker and Kubernetes packages. Moreover, they faced many other problems that seemed too small to matter to anyone outside their team but that still reduced development efficiency and frustrated the team.

"We have a lot of builds and a lot of projects that are in a dependence, and the best way to get dependencies between each project was using ProGet"

Scott Cusson,
Senior Release Engineer, Symbotic

The Solution:

Symbotic needed to find a solution to eliminate their reliance on third-party repositories prone to crashing, manage heavy loads from a widely dispersed team, and unclog their source control while still sharing code.

To eliminate its reliance on crash-prone external feeds, Symbotic uses ProGet to maintain full, on-site control of their repositories. Because ProGet is on-site, there is no risk that an external factor will take Symbotic's productivity offline. This gives Symbotic 100% up-time to get exactly what they want in each release.

To provide additional support to Symbotic's heavy traffic loads, ProGet Enterprise's load balancing functionality allows Symbotic to create as many nodes as needed to handle the load effectively. With multiple nodes balancing the load, ProGet Enterprise creates a High Availability configuration that supports Symbotic's round-the-clock builds. Members of their geographically dispersed team can now to do their work without fear of downtime or interruption.

To avoid over-taxing their source code while still managing dependencies, Symbotic uses ProGet to enable automated code sharing, to monitor dependency management with its NuGet integration, and to use careful versioning on published build packages.

Inedo Innovates to Meet Symbotic's Needs

In addition to its repository management needs, Symbotic also needed a solution that would allow it to remain cutting edge by integrating with Helm. However, ProGet did not offer this functionality at the time. So, although they were convinced ProGet offered the solution to their packages problem, the lack of Helm support presented a serious barrier to adoption.

Not wanting to miss out on ProGet's value as a repository management system, Senior Release Engineer Scott Cusson reached out to Inedo's Customer Support team. The Engineering team took Scott's concerns to heart and worked hard to deliver the value Symbotic the needed, ultimately creating a new integration at a speed that exceeded his expectations.

Today, Helm is fully integrated into ProGet's functionality—a demonstration of Inedo's commitment to responding quickly to customers' needs.

⊚ Overall Results:

Eliminated downtime due to third-party site overload and crashing

24/7 uninterrupted access, no matter how many employees require simultaneous access

Complete dependency management and load-balancing without ever experiencing system crashes

Inedo DevOps tools maximize developer time, minimize release risk, and empower stakeholders to bring their vision to life faster. All with the people and technology you have right now.

inedo