TIDELIFT | Case study: Distributive

How Distributive uses Tidelift to maximize the security and resilience of its open source application components

The founders of Distributive deal in dark matter. At least, that's one of the scientific computing use cases Dan Desjardins was trying to solve for when he and Wes Garland came up with the idea for the company: how can we harness the power of all the computers in the university library to model galaxy formation and understand dark matter's critical role in the evolution of the universe?

Dan is the CEO of Distributive, which was founded in 2017 to create the Distributive Compute Protocol (DCP), a distributed computing platform built on open source web technologies, namely Javascript. Wes is the CTO, and he explains that when people ask, "Why Javascript?"—he has many reasons: it's growing, it's fast, and it's portable, to name a few.

"We've been building a complete distributed computing ecosystem," Dan said. "The goal is to give scientists and researchers—specializing, for example, in astrophysics, drug discovery, fluid dynamics, financial simulations—a powerful tool that spreads millions of calculations out over hundreds of computers and devices already found in a building—on a university campus, in a hospital, or in a manufacturing plant—to compute the answers quicker."

Distributive's model harkens back to SETI@home—that is, the search for extraterrestrial intelligence, a project of the Berkeley SETI Research Center started in the 90s. SETI@home was an Internet-based computing project that connected a network of volunteer computers to share the work of analyzing radio signals using latent compute cycles, searching for signs of extraterrestrial intelligence.

Instead of creating a network of volunteer computers to search for ET, Distributive's software makes use of an organization's latent compute cycles to run all sorts of experiments. One example: Jomo Kenyatta University of Agriculture and Technology in Nairobi uses DCP to power AI applications aiming to optimize food supply chains, detect long queues in public areas, characterize crops, and more. Quietly, in the background, the campus computers are calculating which towns need which produce and how much, and which are the most efficient logistics to transport it before it spoils.

An outgrowth of cloud computing

In many ways, Distributive is a direct reply to the cloud computing model—instead of paying a cloud computing company millions of dollars a year for computational power, why not make use of the latent cycles from the computers, servers, and devices your organization already owns? Think hospitals, universities, governments, enterprises: all own thousands of computers that aren't making full use of their available capacity. Why not run computational workloads in the background and after hours to solve complex scientific and societal problems?

Of course, one thing weighing on Wes's mind was the security of building an application mostly on Javascript. "We deliver our product server backend on Node.js," Wes said. "Their package management is centralized, uncurated, unvetted."

Distributive works with many highly regulated industries like hospitals and governments, so hoping for the best and letting unmanaged dependencies run free was not an option. Wes knows the code he's building on the DCP server is secure, but what about all the Node.js dependencies?

A multilayered approach to risk mitigation and open source security

"Security is a multilayered approach," Wes said. "[We can make sure] DCP is secure on servers and that no untrusted workload ever gets touched by anything other than our extremely secure firewall."

That's where Tidelift comes in. Wes asked Doug Stewart, co-founder and Director of Special Operations at Distributive, to research ways to ensure their Javascript dependencies are audited and vetted, a process which would otherwise be a full-time hire.

"Without doing what Tidelift is doing internally, [we wouldn't have time for this]. This is one place where outsourcing does save an organization of our size money," Wes said. "[Tidelift creates] that supply chain resilience and [helps us] keep overall costs down, [so that] Trojan horses stay outside of the city."

The Tidelift solution is simple: it's a multilayered "defense in depth" approach to application health and security. The bottom layer is an organization's own code, surrounded by container security, software composition analysis, and testing—we think of these approaches as reactive. Tidelift cocoons your application and these reactive approaches in a proactive layer, where organizations like Distributive can apply governance, policies, and standards to the open source components they use.

Open source software supply chain resilience

Tidelift provides a proactive, people and software-powered approach to managing open source effectively by working directly with the people who know open source components best—the maintainers who build the libraries themselves.

Once Distributive decided on Tidelift, the onboarding process was a breeze. The team, including software engineer Bryan Hoang, identified some quick wins. "Tidelift identified some packages and CVEs that npm audits didn't identify," Bryan said. "We fixed those issues in the CLI tool."

With the Tidelift Subscription, Wes and his team now have a curated list of maintained and vetted dependencies they can use to make DCP as light and fast as possible, so they can focus on the myriad complex problems for which DCP was created: galaxy modeling, optimizing supply chains, and maximizing surgical throughput in hospitals, to name a few.