

SUPPORTING MULTIPLE DISPARATE TOOLS WHILE OPTIMIZING STORAGE

Why Mercedes Benz Research and Development (MBRDNA) chose JFrog Artifactory as its Universal Artifact Repository Manager



Siva Mandadi,

Sr. DevOps engineer, Connected Cars, Mercedes Benz Research and Development

Next Generation UI for Connected, Autonomous Vehicles

Mercedes-Benz is a household name that is equated with luxury and quality in the automotive industry. MBRDNA division has more than 300 engineers working on UI. Mercedes' R&D facility in Sunnyvale, California conducts research in the field of IoT, machine learning, user interface and informatics for the novel and innovative technology of autonomous, connected cars. Engineers at this facility are working on the next generation UI that will go far beyond self-driving vehicles.

Self-driving technology is only the beginning, and new features are adding intelligence to the car and the driving experience in general, allowing drivers to do everything that they now do on a cell phone from their connected car. For example, you will be able to use voice commands to organize an online calendar, and if an appointment includes a GPS location, your car will take you there autonomously. But the technology goes beyond just getting you to your destination. As the car drives you to your appointment, it will check your mood and play music accordingly. And if you think "IoT", as your car drives you home, you'll be able to tell it to switch on your coffee machine as you get close, and turn on the hallway lights when you arrive.

Before
<ul style="list-style-type: none">• 1000s of artifacts/week• 1 TB of data/week• 60 daily builds stored• 6-hour CI builds
After
<ul style="list-style-type: none">• 10000s of artifacts/week• 2 TB of data per week• 600 daily builds stored• CI builds within minutes

The Challenge: Supporting Polyglot Programming

As with any typical software development organization today, the MBRDNA engineers were using multiple development and build technologies. Conan for C/C++ development, Maven for Java development, and Docker for containerization to name a few. Some teams used Jenkins CI while others were using Bamboo to run their continuous integration builds. The wide variety of technologies created several problems. Managing dependencies for all the different development technologies was complicated and required detailed configuration to access all the required public remote repositories. CI builds were very slow since Gigabyte-sized dependencies had to be downloaded, and could take 5 - 6 hours to run. Then, once builds succeeded, there was the challenge of sharing build artifacts between the different teams that had to collaborate on a project. Eventually, storage on the Network Attached Storage (NAS) also maxed out, and the teams needed different plugins to free up space and had to limit the accumulation of daily builds to 50 - 60 per month. The teams were generating thousands of artifacts amounting to 1TB of data per week and needed a solution that would support the tens of thousands of artifacts amounting to at least 2TB of data per week that they expected to reach within a few months.

The Solution: A Universal Repository - JFrog Artifactory

Siva's team had previously used Apache Archiva and was currently using Sonatype Nexus to manage binary artifacts, however, previous experience with Artifactory made it an easy choice. Artifactory's full support for all major package formats and build tools meant that the MBRDNA teams could do all their dependency management and builds through Artifactory. In addition, Artifactory's checksum-based storage and remote repositories made both storage and dependency management much more efficient. CI servers could now access Gigabyte-sized dependencies stored in remote repository caches on the local network, within minutes, dramatically reducing both build times and storage requirements. With these

improvements, the team was able to accumulate 500 - 600 daily builds per month, and an added benefit was the fine-grained access control that Artifactory provided giving the team an easy way to define who could upload and download artifacts. But there were three things that clinched it for MBRDNA. The first was the intuitive UI offering advanced search and a clear view into the exhaustive build information that Artifactory stores. The second was the ease in which Artifactory can be configured using a simple YAML configuration file. The third was JFrog support that provided unbeatable response times to questions that arose during the initial usage period.

I had used Archiva and Nexus previously, but once I started using Artifactory, I understood that Artifactory is to binary artifact management as Jenkins is to CI.

Siva Mandadi, Sr. DevOps engineer, Connected Cars, Mercedes Benz R&D Division

Promoting Faster, Collaborative, Scalable Development

As a result of adopting Artifactory, MBRDNA's teams' development processes dramatically improved. With Artifactory sitting in the center of the development ecosystem, onboarding new team members and getting them to contribute to projects became much easier. Shorter build times meant shorter development cycles and earlier discovery of bugs which could then be fixed earlier on in the development process. Using Artifactory made it much easier for teams to collaborate safely using the right access privileges which also made it much easier to scale the teams to double their initial sizes.

Why MBRDNA Selected Artifactory

- Intuitive UI with easy search and exhaustive build information
- Easy to configure and set up
- Unbeatable support

Summary

The teams working on the next-generation UI for MBRDNA's self-driving, connected cars were suffering from the limitations imposed by their development infrastructure. The wide variety of technologies they were using made their daily lives difficult with complex configuration, slow builds, and maxed-out storage. Adopting JFrog Artifactory made their lives much easier. Setting up Artifactory with a simple configuration file was easy. As a Universal Repository Manager, Artifactory fully supported all the development technologies that the MBRDNA UI teams were using, making configuration and dependency management of their development tools much simpler. Artifactory's remote repository cache storing large dependencies locally made CI builds run much more quickly. As an added bonus, MBRDNA's teams experienced unbeatable response times from JFrog support whenever they had a question or needed guidance in their adoption of Artifactory.