



Sonatype Nexus Repository Review From A Customer



From IT Central Station, the leading review site for enterprise technology solutions.

Review by a Real User

Verified by IT Central Station



DevOps Practitioner at a financial services firm with 5,001-10,000 employees

Kulbhushan Mayer

WHAT IS OUR PRIMARY USE CASE?

We are using this tool for our Java, .NET, AngularJS and Node.js. Apart from that, we have recently built a solution to utilize this tool for Docker images as well.

WHAT IS MOST VALUABLE?

The first good feature is the searching capability. They have recently really improved it. When I started using it, their search was quite slow. They have improved in that area very effectively, and I really like it. The second valuable feature is that in a single tool we are managing multiple central repositories. The more support we have, the more the tool will be effective because we are able to manage multiple requirements from multiple domains in a single tool. Initially, when I started using this tool, I was mainly using it for Java. Then I onboarded .NET, then I onboarded JS. And about six or eight months back, I onboarded Python. And I am about to onboard Docker. The availability of integrations allows me to do this. Of course, we use this solution to manage binaries, build artifacts, and release candidates across the DevOps pipeline. I can't say how our company is doing so, that's company information, but we are using it to manage artifacts across our pipeline, starting from the lower environment and ending in production.

WHAT NEEDS IMPROVEMENT?

One feature that needs changing is their pricing model. They are charging a huge amount. The way they charge it's too much. In addition, they should have some feature where we can move a specific repository from one instance of Nexus to another instance of Nexus. As of now, this feature doesn't exist. With the recent upgrade, when they moved from 2.x to 3.x, they made a couple of changes in the backend regarding how data is saved. That, again, makes it a bit difficult to move the changes. So the feature that I would suggest is the capability to move repositories that people have configured in their systems from one instance to the other. If they had this feature, it would be very effective.



FOR HOW LONG HAVE I USED THE SOLUTION?

More than five years.

WHAT DO I THINK ABOUT THE STABILITY OF THE SOLUTION?

It's reliable, of course. That's why we have been using this tool for the last eight years. It is stable, but one thing that's a drawback is that, whenever they release a new major version, like 3.12 or 3.14 or 3.15, what generally happens is that we wait for at least a month or so before upgrading to that version. Recently-released versions are definitely not stable, and they definitely have room for the improvement there.

WHAT DO I THINK ABOUT THE SCALABILITY OF THE SOLUTION?

The scalability is effective, but it's not as effective as other tools available. Currently, they are only supporting a single data center. They should have the ability to support multiple data centers. That is actual scalability and, in effect, high-availability. For example, if I have three different instances and I want to maintain high-availability and the scalability of my tool, it needs to be scalable across my DCs, rather than on a single DC. If that single DC goes down for some reason, then my tool goes down. If I have ten servers being maintained in a DC, I have to maintain ten others in DC2 and ten more in DC3. If DC1 goes down, I have to bring DC2 up with all ten servers. That means I'm managing 30 servers, where I could be managing ten. And they should have the capability for automation to refresh from one instance to the other.

HOW ARE CUSTOMER SERVICE AND TECHNICAL SUPPORT?

Technical support is effective.

IF YOU PREVIOUSLY USED A DIFFERENT SOLUTION, WHICH ONE DID YOU USE AND WHY DID YOU SWITCH?

This tool was here when I joined the team.

HOW WAS THE INITIAL SETUP?

The setup was very easy. The deployment took somewhere between 30 minutes and one hour. The install takes about 30 minutes and the rest is the upgrade process. The time for upgrading depends on the data, if there have been multiple changes. But the deployment process should not take more than a half hour, and that's on the high side.

WHAT ABOUT THE IMPLEMENTATION TEAM?

We did it by ourselves. We just went ahead with the documentation and tried to understand it. If we had any queries, we just raised them with their support team, got them clarified, and then moved ahead with the upgrade. Our internal team was comprised of three people, including me.

WHAT WAS OUR ROI?

It's expensive, so the return on investment is not that much.



WHAT'S MY EXPERIENCE WITH PRICING, SETUP COST, AND LICENSING?

One thing about this tool that I found difficult is that it's quite expensive. They are charging around \$110 or \$120 per user, per year. It's quite expensive in comparison to the other tools available in the market, not just for these requirements but for other requirements as well. Those tools are very extensively used but they are charging much less compared to Nexus. And there are a couple of more tools that Sonatype has that are even more costly when compared to Sonatype Nexus. Those should be available by default in the tool or available at a much cheaper cost if we're already using one of their tools.

WHICH OTHER SOLUTIONS DID I EVALUATE?

There are a couple of solutions that I have used in the past, before using Nexus. They're as effective as Nexus. When I joined the team, they were already using Nexus and it was being used very effectively in the company, within our team, and across all the different users. I didn't see any reason to ask that we go ahead and switch a different tool.

WHAT OTHER ADVICE DO I HAVE?

If you have the ability to implement this tool within your team by yourself, go with the open-source solution that they have, rather than going ahead with the paid solution. I started with the 2.2 version but two weeks back I upgraded to 3.15. Before that, I was using 3.14 and prior to that I was on 3.9. The clean-up policies have been improved: How we manage our changes, how we manage our artifacts, how much we need to keep, and how much we want to remove. That has effectively improved. In addition, slowly they have onboarded multiple technologies. As I mentioned, I started just with Java and I am now supporting six technologies on this. Three of them have been onboarded in the last one-and-a-half years. They are improving a lot in almost all areas. With every release they have made a couple of changes. Searching was the main capability that they needed to improve and, recently, with the latest upgrade that I did done one month back, they improved it a lot. They have really made the search very fast. We have a license for about 200 users and we have around 100 to 150 users. We have a spare of fifty so that if we increase we can cater to that. I am not able to share the roles of our users. Usage is quite extensive and is increasing day by day, as more and more technologies are being onboarded. We are amazingly maintaining it within my team, which consists of six people. But a single person is maintaining it, not the entire team. We have multiple instances of this tool running in our company. Most of the instances are open-source. One instance is a paid instance and that is being managed by my team. That instance is a benchmark for the entire company. The rest of the teams are utilizing the open-source version. I would rate this solution at seven out of ten. It needs the ability to be available across DCs, across geographies. In addition, it needs the capability to do an auto-sync between the between different centers or there should be some automated trigger for that. They need the capability to move repositories from one instance to the other instance, as well. Last but not least, they should have a bit lower pricing.