

Run **Web Start** based application after the release of **Java 11**

Java Web Start will die!

Java Web Start (JWS) was deprecated in Java 9 and starting with Java 11 Oracle will remove JWS altogether. According to the current roadmap this will happen in September 2018. From this time on, it will be impossible to use JWS-based applications on clients that have the latest version of Java installed. For more information, please take a look at the „Java Client Roadmap Update“ (Whitepaper, Oracle, March 2018) or the DZone article „What Future Java Releases Mean for Legacy Desktop Apps“.

Solution

The team at Karakun plans to implement an open-source successor of Java Web Start. Our intention is to provide native installers for all major platforms (Windows, Linux & MacOS). With this solution, applications based on Java Web Start can be used with all future versions of Java - while no code changes are required.

Technical details

The product will be based on the JNLP-specification defined in JSR-56 and will implement the most commonly used features of Java Web Start. It will be able to handle any typical JWS-based application and will support all future versions of Java starting from Java 11. The tool will provide exactly the same workflow as Java Web Start, therefore nothing will change from the point of view of your users:

- A JNLP-file that describes the application is downloaded to the client, typically either by clicking a link on a webpage or by an automatic provisioning process.
- Our tool registers itself for the JNLP-file-extension and the MIME-type „application/x-java-jnlp-file“, therefore by double clicking the JNLP-file, our tool launches.
- It parses the JNLP-file, downloads all needed resources (JARs, native libraries and images) and stores them in a cache.
- Once all resources are available, the application starts.

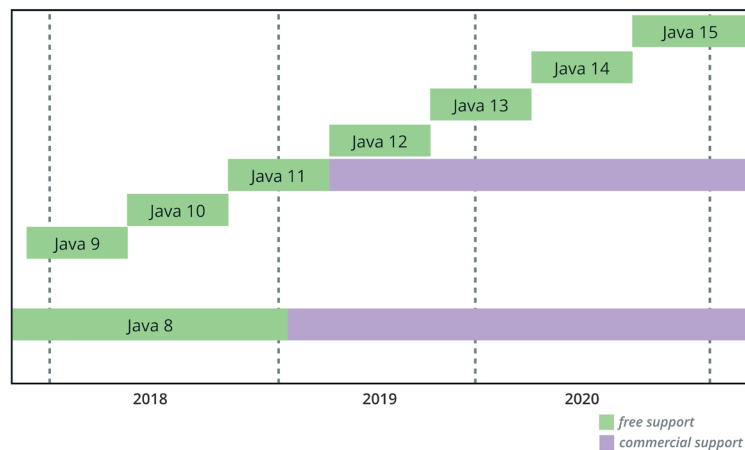
The tool automatically checks if a newer version of your application is available and downloads updated resources automatically. This guarantees that your users will always run the latest version of your applications on any client. The focus of the tool are rich desktop applications. It will not address the Java Applet technology. Therefore no browser plugins will be part of the tool. The application will come with a native installer, which ensures a simple installation process. It needs to be installed on a client only once, because it will contain a modern update mechanism to download and install required updates automatically.

Alternative Solutions

There are certainly other solutions available but to our knowledge all of them apply only to specific scenarios and have other downsides. Sticking to Java 8 may appear as a possible solution. If all clients continue to use Java 8, one can continue to use the official Java Web Start. This requires that the configuration of all client computers can be controlled. Unfortunately this also means, that no Java applications - even those that do not require Java Web Start - can take advantage of newer versions of Java.



Free support for Java 8 will end soon, as you can see in the following image that shows an overview of the Java release train beginning with 2018:

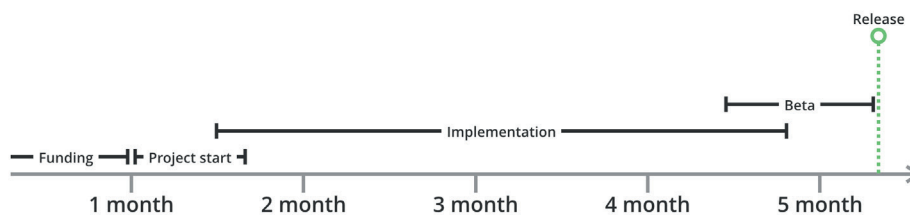


If you continue to use Java 8 after free support has ended, you either have to pass on security updates or buy commercial support from Oracle. Commercial support for Java 8 will cost about 30-40\$ per client instance that needs to run a Java Web Start based application, i.e. if your application runs on 500 client machines, you may have to pay up to 20.000\$ just to support this single application. Another option is the migration to an alternative mechanism for installing and starting your applications, e.g. building native installers. These solutions require updates to the build process for every application and distinct builds for each supported platform. This is only an option, if you have to support a limited number of applications. Deployment is not part of this solution and needs to be resolved differently. Compared to the alternative solutions our proposal has the following advantages:

- no refactoring of your applications or build pipeline is required because we plan a Java Web Start compatible approach
- the best alternative tools are closed source
- the commercial support from Oracle is not sustainable because it just delegates the problem to the future, concretely at the time when Java 8 commercial support will end
- no commercial license per client instance is required

Roadmap

Currently we are looking for funding. Once funding is secured, we will start development. The following diagram shows the roadmap:



Interested?

If you want to learn more or if you would like to discuss your specific needs, get in touch with us at info@karakun.com.