

---

# Java 9 and 10 (11) support and the SmartClient

---

Servoy supports through the admin page Java9 from 815/82 on. We have then the configuration option to choose java9 as minimum or maximum.

For bootstrap.jnlp you can add the <j2se> tag yourself but the version is not 1.9 (like 1.7/1.8 before) but now just 9, so without the 1. before it.

To run the smart client some vm args must be given to so that the Windows Look and Feel will work (and possible others should be added if more exceptions are seen as shown below:

The exception that the java console log will give under windows is something like:

```
java.lang.IllegalAccessException: class com.servoy.j2db.LAFManager cannot access class com.sun.java.swing.plaf.windows.WindowsLookAndFeel (in module java.desktop) because module java.desktop does not export com.sun.java.swing.plaf.windows to unnamed module @6c6fdd7c
```

For this we need to add on the admin page (vm args section of the smartclient) or the bootstrap.jnlp file the vm args of the <j2se> tag:

```
--add-exports=java.desktop/com.sun.java.swing.plaf.windows=ALL-UNNAMED
```

```
--add-exports=java.desktop/sun.java2d=ALL-UNNAMED
```

So it could be that more IllegalAccessException like that are shown with

```
class X cannot access class <some.package>.Class in module <some.module>
```

that means we need to add another --add-exports in this syntax:

```
--add-exports=<some.module>/<some.package>=ALL-UNNAMED
```

See <http://openjdk.java.net/jeps/261> for more info

Also Java9 and the developer is here <https://wiki.servoy.com/display/DOCS/Java+9+support+for+the+developer>

The other problem is the bug Oracle did introduce in Java 9 (as a regression from Java 8) that they until now don't plan to fix for Java 9 (but only Java 10). We don't have a workaround for that because we heavily depend on this to resolve the media:/// urls (media from the solution or from the database through the blobloader syntax)

<https://bugs.openjdk.java.net/browse/JDK-8178799>

So all kinds of media:/// urls directly set on the components or inside <html> markup that is shown on labels are not resolving to the right image.

For Java 11 there is a bigger problem because there is no JRE anymore that integrates into the system that also supplies webstart.

So the only way to start it there is to use the [bootstrapper](#) in standalone mode and you need to install java and setup the paths correctly so you can run `java -jar bootstrap.jar`