
Subject: Interest in a Mavenised form of the code?
Posted by [mjw99](#) on Wed, 20 Jul 2022 09:44:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

I've managed to code a Mavenised fork of datawarrior. The CI on this via GitHub Actions also works; it builds automatically with every commit. This in turn, depends on a Mavenised fork of fxmolviewer that I have also created. I can't post the links due to limitations of the forum software, but they are under my github repo (mjw99).

Is there any interest in merging this? I think it will make building and CI easier.

I see that Actelion/openchemlib is Mavenised and this is wonderful to use with other projects via Maven Central.

Thanks,

Mark

Subject: Re: Interest in a Mavenised form of the code?
Posted by [mjw99](#) on Wed, 20 Jul 2022 09:51:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

The repos are at:

<https://github.com/mjw99/datawarrior>
<https://github.com/mjw99/fxmolvviewer>

Subject: Re: Interest in a Mavenised form of the code?
Posted by [thomas](#) on Wed, 20 Jul 2022 10:06:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Mark,

thank you for your support. There is certainly a lot of interest, because Maven is a kind of de-facto standard. I personally don't have much experience in it, because I don't use Maven. I use simple shell scripts the old-fashioned way. Some of DataWarrior's dependencies are rather large jar files which I manually reduce substantially in size before adding them to the lib folder. Therefore, I would favor a dual option approach: offer both, a maven and a bash-script based build process as default options for the user to select from.

If you agree, lets merge the maven build into the original DataWarrior and FXMolviewer projects. I would give you access. We can discuss that directly via VC or e-mail?!

Best wishes, Thomas

Subject: Re: Interest in a Mavenised form of the code?

Posted by [mjw99](#) on Wed, 20 Jul 2022 12:28:01 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Thomas,

Thank you for the reply.

I can certainly help with this; my email is mjw@mjw.name.

Thanks,

Mark

Subject: Re: Interest in a Mavenised form of the code?

Posted by [mjw99](#) on Thu, 08 Feb 2024 14:16:15 GMT

[View Forum Message](#) <> [Reply to Message](#)

Dear Thomas,

I have recently been investigating JPackage[1] as a tool for creating platform dependent installers of Java programs. It first appeared in Java 15 (I think) and I have been using it in the LTS JDK 17. I have also found a Maven plugin for this [2] and have managed to apply it to a branch of my DataWarrior Maven fork[3].

An aspect of JPackage is that it can only build binary installers on the platform that it is executed on. Hence I have three different containers in my GitHub actions config for each platform[4]. The upshot is that these each build an installer[5] and this final binary installer is deposited as an artifact[5], which can be downloaded. This entire process is triggered every time one commits to the repository.

Further tuning of icons and file associations can also be done with JPackage[6,7]. I have tested Windows and Linux and it seems to work. I've not tested Mac yet, since I do not have access to one.

This needs a bit more optimisation at the moment, but I thought I would share this with you and please let me know what you think of this.

Thanks,

Mark

[1] <https://docs.oracle.com/en/java/javase/19/jpackage/packaging-overview.html#GUID-C1027043-587D-418D-8188-EF8F44A4C06A>

[2] <https://github.panteleyev.org/jpackage-maven-plugin/>

[3] <https://github.com/mjw99/datawarrior/tree/jpackage>

[4] <https://github.com/mjw99/datawarrior/blob/1c459a7b23c69fbe1de6c3fa112c081bd9e91ed1/.github/workflows/maven.yml#L14>

[5] <https://github.com/mjw99/datawarrior/actions/runs/7829134115>

[6] <https://github.com/mjw99/datawarrior/blob/1c459a7b23c69fbe1de6c3fa112c081bd9e91ed1/pom.xml#L295>

[7] <https://github.com/mjw99/datawarrior/tree/jpackage/src/main/resources/jpackage>

Subject: Re: Interest in a Mavenised form of the code?

Posted by [thomas](#) on Thu, 08 Feb 2024 21:04:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

Dear Mark,

many thanks for your investigation on jpackage. My way of creating all three installers using three completely methods on the three platforms is historic and only done, because at that time no cross platform solution supported registering file types and corresponding icons. They have all been pretty useless. For the recent update I used the old mechanism because I didn't have the time to evaluate more modern solutions. I had considered Hydraulic Conveyor, because with it you can build on one platform installers for all platforms, it supports icons and file types and it simplifies application signing, which might be necessary in the future. It is commercial, but free for open source projects. I didn't have the time to try it yet and finally don't know, which solution is the most promising. Essential are platform and icon registration (double click on the OS level omust open file, second file must open in first open app, except Linux), splash screen, multi architecture support (e.g. universal binary for MacOS). Of course, ideally, everything is defined once and then translated by the tool into every platform installer.

Of course it would be fantastic, if you could create a Github repository that creates proper applications and installers for Mac,Linux,Windows just from the datawarriorlauncher.jar and datawarrior_all.jar. Do you think, JPackage can do that? Or rather Conveyor? A link would be <https://conveyor.hydraulic.dev/3.0/faq/comparisons/#1-jpacka> ge

Best wishes,

Thomas

Subject: Re: Interest in a Mavenised form of the code?

Posted by [mjw99](#) on Tue, 26 Mar 2024 10:43:06 GMT

Dear Thomas,

Apologies for the delay in replying to this.

>Hydraulic Conveyor

I've not seen this tool, but it looks interesting; I'll have a read of it.

>Of course it would be fantastic, if you could create a Github repository that creates proper applications and installers for Mac,Linux,Windows just from the datawarriorlauncher.jar and datawarrior_all.jar. Do you think, JPackage can do that? Or rather Conveyor? A link would be <https://conveyor.hydraulic.dev/3.0/faq/comparisons/#1-jpackage>

I am not sure if this is possible offhand with JPackage. You might be able to short circuit to the package phase, having placed those jars in ./target/libs/. I'll have a play to see if I can get that to work.
