Item 276
**git_comments:**

**git_commits:**

1. **summary:** PHOENIX-3020 Bulk load tool is not working with new jars
   **message:** PHOENIX-3020 Bulk load tool is not working with new jars

**github_issues:**

**github_issues_comments:**

**github_pulls:**

**github_pulls_comments:**

**github_pulls_reviews:**

**jira_issues:**

1. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
2. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
3. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
   **label:** code-design
4. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
   **label:** code-design
5. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
   **label:** code-design
6. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
   **label:** code-design
7. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
8. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
9. **summary:** Bulk load tool is not working with new jars
   **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
10. **summary:** Bulk load tool is not working with new jars
    **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars
11. **summary:** Bulk load tool is not working with new jars
    **description:** [~rajeshbabu] and [~sergey.soldatov] has identified an issue related to bulk load tool with latest jars

**jira_issues_comments:**

1. Yep. The problem is that META-INF dir has {{LICENSE}} file and {{license}} directory. Somehow runJar is not case sensitive when unpacking jar and fails with an error that license is not a directory. [~elserj] also found that if we build against signed packages, the certificate comes to META-INF dir and cause problem with using such jars. Will provide fix shortly.
2. Fixed bulk load clash of license files, filtered signature files.
3. **body:** {code} - <relocation> - <pattern>com.sun.jersey</pattern> - <shadedPattern>${shaded.package}.com.sun.jersey</shadedPattern> - </relocation> + <!--<relocation>--> + <!--<pattern>com.sun.jersey</pattern>--> + <!--<shadedPattern>${shaded.package}.com.sun.jersey</shadedPattern>--> + <!--</relocation>--> {code} Why the undo of this shading? If we don't want to do this for some reason, let's remove instead of comment. Aside, have you looked at the shaded/fat jars to verify that they still have the necessary information in META-INF/LICENSE and META-INF/NOTICE per ASF policies? It's not clear to me just looking at the patch whether or not those files will still be present. bq. Somehow runJar is not case sensitive when unpacking jar and fails with an error that license is not a directory Is this something else that we should look into fixing (for >4.8.0)?
   **label:** code-design
4. **body:** With shaded jersey MR jobs fails with {noformat} 016-06-22 01:32:49,669 WARN [main] org.mortbay.log: failed guice: com.sun.jersey.spi.service.ServiceConfigurationError: com.sun.jersey.spi.HeaderDelegateProvider: The class org.apache.phoenix.shaded.com.sun.jersey.core.impl.provider.header.LocaleProvider implementing provider

interface com.sun.jersey.spi.HeaderDelegateProvider could not be instantiated: Cannot cast org.apache.phoenix.shaded.com.sun.jersey.core.impl.provider.header.LocaleProvider to com.sun.jersey.spi.HeaderDelegateProvider 2016-06-22 01:32:49,669 WARN [main] org.mortbay.log: failed org.mortbay.jetty.webapp.WebAppContext@351f2244{/,jar:file:/Users/ssoldatov/bin/hadoop/share/hadoop/yarn/hadoop-yarn-common-2.7.2.jar!/webapps/mapreduce}: com.sun.jersey.spi.service.ServiceConfigurationError: com.sun.jersey.spi.HeaderDelegateProvider: The class org.apache.phoenix.shaded.com.sun.jersey.core.impl.provider.header.LocaleProvider implementing provider interface com.sun.jersey.spi.HeaderDelegateProvider could not be instantiated: Cannot cast org.apache.phoenix.shaded.com.sun.jersey.core.impl.provider.header.LocaleProvider to com.sun.jersey.spi.HeaderDelegateProvider {noformat} Once we shade jersey, shading plugin also shading all related records in META-INF/services and it becomes used by MR's jetty. I will remove it (actually thought to keep it to remember that we are not shading it and possible we will need to address it in the future. Like removing services file if we do not use it) our LICENSE/NOTICE files are copying to the root of the jar. Actually some clean up is required for this mess of licenses from dependencies, but at the moment there are no changes between how we handling them now and how we did it before shading stuff, so it can/should be done in a separate jira later. As for the runJar - it happens during unpacking, so it's not related to our code.
**label:** code-design

5. **body:** With shaded jersey MR jobs fails with {noformat} 016-06-22 01:32:49,669 WARN [main] org.mortbay.log: failed guice: com.sun.jersey.spi.service.ServiceConfigurationError: com.sun.jersey.spi.HeaderDelegateProvider: The class org.apache.phoenix.shaded.com.sun.jersey.core.impl.provider.header.LocaleProvider implementing provider interface com.sun.jersey.spi.HeaderDelegateProvider could not be instantiated: Cannot cast org.apache.phoenix.shaded.com.sun.jersey.core.impl.provider.header.LocaleProvider to com.sun.jersey.spi.HeaderDelegateProvider 2016-06-22 01:32:49,669 WARN [main] org.mortbay.log: failed org.mortbay.jetty.webapp.WebAppContext@351f2244{/,jar:file:/Users/ssoldatov/bin/hadoop/share/hadoop/yarn/hadoop-yarn-common-2.7.2.jar!/webapps/mapreduce}: com.sun.jersey.spi.service.ServiceConfigurationError: com.sun.jersey.spi.HeaderDelegateProvider: The class org.apache.phoenix.shaded.com.sun.jersey.core.impl.provider.header.LocaleProvider implementing provider interface com.sun.jersey.spi.HeaderDelegateProvider could not be instantiated: Cannot cast org.apache.phoenix.shaded.com.sun.jersey.core.impl.provider.header.LocaleProvider to com.sun.jersey.spi.HeaderDelegateProvider {noformat} Once we shade jersey, shading plugin also shading all related records in META-INF/services and it becomes used by MR's jetty. I will remove it (actually thought to keep it to remember that we are not shading it and possible we will need to address it in the future. Like removing services file if we do not use it) our LICENSE/NOTICE files are copying to the root of the jar. Actually some clean up is required for this mess of licenses from dependencies, but at the moment there are no changes between how we handling them now and how we did it before shading stuff, so it can/should be done in a separate jira later. As for the runJar - it happens during unpacking, so it's not related to our code.
**label:** code-design

6. **body:** added better handling notice files, removed commented part.
**label:** code-design

7. [~elserj]/[~enis], can you please review this, Is this how other projects are handling or we have not experience this elsewhere yet?

8. bq. Once we shade jersey, shading plugin also shading all related records in META-INF/services and it becomes used by MR's jetty. I will remove it (actually thought to keep it to remember that we are not shading it and possible we will need to address it in the future. Like removing services file if we do not use it) Gotcha. So, someone is calling this from a jar other than our shaded client jar (and is looking for the non-shaded packages)? I'll admit, I don't know what you're referring to with "runJar". Is that somethign in Phoenix? A Hadoop/YARN thing for submitting a job? bq. our LICENSE/NOTICE files are copying to the root of the jar. Actually some clean up is required for this mess of licenses from dependencies, but at the moment there are no changes between how we handling them now and how we did it before shading stuff, so it can/should be done in a separate jira later. Yeah, tbqh, I'm a little fearful of rc0 and verifying the ASF licensing requirements... bq. can you please review this, Is this how other projects are handling or we have not experience this elsewhere yet? Most projects don't have a good story here AFAIK :) {noformat} + <!--<relocation>--> + <!--<pattern>com.sun.jersey</pattern>--> + <!--<shadedPattern>${shaded.package}.com.sun.jersey</shadedPattern>--> + <!--</relocation>--> {noformat} Looks like you forgot to remove this commented block in your 2nd patch. Otherwise, if you're having success with this, +1 from me.

9. [~elserj] thanks for the review. Removed commented part and committed to master and 4.x branches.

10. SUCCESS: Integrated in Phoenix-master #1290 (See [https://builds.apache.org/job/Phoenix-master/1290/]) PHOENIX-3020 Bulk load tool is not working with new jars (ssa: rev 3e69b90d8666ffb8017cf5537d987522ef485d57) * phoenix-queryserver-client/pom.xml * phoenix-server/pom.xml * phoenix-queryserver/pom.xml * phoenix-client/pom.xml