

Item 114

git_comments:

git_commits:

1. **summary:** Merge pull request #569 from pwendell/merge-fixes.
message: Merge pull request #569 from pwendell/merge-fixes. Fixes bug where merges won't close associated pull request. Previously we added "Closes #XX" in the title. Github will sometimes linbreak the title in a way that causes this to not work. This patch instead adds the line in the body. This also makes the commit format more concise for merge commits. We might consider just dropping those in the future. Author: Patrick Wendell <pwendell@gmail.com> Closes #569 and squashes the following commits: 732eba1 [Patrick Wendell] Fixes bug where merges won't close associated pull request.

github_issues:

github_issues_comments:

github_pulls:

1. **title:** Fix SPARK-1629: Spark should inline use of commons-lang `SystemUtils.IS_...
body: ...OS_WINDOWS`

github_pulls_comments:

1. Can one of the admins verify this patch?
2. Jenkins, test this please. Looks good to me, pending tests.
3. Jenkins, test this please.
4. Merged build triggered.
5. Merged build started.
6. Merged build finished. All automated tests passed.
7. All automated tests passed. Refer to this link for build results:
<https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/14526/>
8. I am still catching up on PR's and bugs. Why was this changed ? Hacky solutions based on string parsing of properties lead to fragility in case of changes in future. Since we are already depending on apache commons, makes sense to delegate to that.
9. The problem was that the project was not depending on commons lang already. The code is the same as is contained in commons lang. The property being referenced is a standard JVM property. On May 3, 2014 9:34 PM, "Mridul Muralidharan" notifications@github.com wrote: > I am still catching up on PR's and bugs. > Why was this changed ? > Hacky solutions based on string parsing of properties lead to fragility in > case of changes in future. > Since we are already depending on apache commons, makes sense to delegate > to that. > > — > Reply to this email directly or view it on GitHub<https://github.com/apache/spark/pull/569#issuecomment-42115677> > .
10. Maven dependency still shows org.apache.commons:commons-lang3:jar - am I missing something here ? Btw, we do depend on it for repl tests too ...
11. Yes, but those are transitive dependencies. `core` happens to bring in both lang and lang3, and this was using the old lang dependency even. It would be correct-er to either depend directly on it, or not use it. To me it seems better to inline what amounts to one-liner access of a standard system property. I think it's also true that Guava is the go-to utility package and would be better to use Guava where possible rather than encourage use of lang too. The use of lang3 in ReplSuite should also either be 'inlined', or else supplemented with a direct test-scope dependency. It's just a test, so not as big of an issue. I think the one use can even be replaced with a simple escape, which I presume is there to handle backslash on Windows.
12. If we are depending on something and bundling it, we might as well use it instead of duplicating code and having to maintain the changes : assuming it is intuitive to use. Since apache commons is used by whole bunch of projects, bugs would be better reported and fixed there than hacked up code within spark. The long term maintenance cost adds up for these 'small' changes. Things like os.name.toLowerCase startswith windows or file.separator ==\ are hacks we should avoid and let others handle if being maintained.
Regards Mridul On May 4, 2014 12:55 PM, "Sean Owen" notifications@github.com wrote: > Yes, but those are transitive dependencies. core happens to bring in both > lang and lang3, and this was using the old lang dependency even. It would > be correct-er to either depend directly on it, or not use it. To me it >

seems better to inline what amounts to one-liner access of a standard > system property. > > I think it's also true that Guava is the go-to utility package and would > be better to use Guava where possible rather than encourage use of lang too. > > The use of lang3 in ReplSuite should also either be 'inlined', or else > supplemented with a direct test-scope dependency. It's just a test, so not > as big of an issue. I think the one use can even be replaced with a simple > escape, which I presume is there to handle backslash on Windows. > > — > Reply to this email directly or view it on GitHub<https://github.com/apache/spark/pull/569#issuecomment-42126473> > .

13. That makes sense. Since Tachyon now brings in `lang3`, it is no extra burden to use it. It would rationalize the use in `ReplSuite`. Anyone second a PR a to bring _back_ use of Commons Lang, but the latest `lang3` version?
14. Yes, lang3 - not lang : some of the methods (for example, the escape method used in repl) is actually broken in lang, but works in lang3. On Sun, May 4, 2014 at 10:09 PM, Sean Owen notifications@github.com wrote: > That makes sense. Since Tachyon now brings in lang3, it is no extra > burden to use it. It would rationalize the use in ReplSuite. Anyone > second a PR a to bring _back_ use of Commons Lang, but the latest lang3version? > > — > Reply to this email directly or view it on GitHub<https://github.com/apache/spark/pull/569#issuecomment-42137398> > .
15. I'm ambivalent between inlining this and fixing up the dependency. In general we should probably rely on other utilities where possible, but if the JVM ever changes the way windows machines identify themselves in a non-backwards-compatible fashion, I will eat my shoe.

github_pulls_reviews:

1. @srowen ``` scala def isWindows(): Boolean = { try { val osName = System.getProperty("os.name") osName != null && osName.startsWith("Windows") } catch { case e: SecurityException => (log a warning and return false) } } ``` You think here will be thrown a SecurityException .Why?
2. Regarding `SecurityException`, `System.getProperty` will often be disallowed in an environment where the security manager has been enabled. I strongly suspect that the rest of Spark fails already in such an environment for other reasons, and you could argue that the only reasonable response here is to fail rather than "guess" that the environment is not Windows. But I put it in mostly to mimic `SystemUtils`. It could go either way.

jira_issues:

jira_issues_comments: