

git_comments:**git_commits:**

1. **summary:** SOLR-10912: fix routing of Solr non-contrib build output dirs (e.g. solr/core -> ../build/solr-core; previously -> ../build/core)
message: SOLR-10912: fix routing of Solr non-contrib build output dirs (e.g. solr/core -> ../build/solr-core; previously -> ../build/core)

github_issues:**github_issues_comments:****github_pulls:****github_pulls_comments:****github_pulls_reviews:****jira_issues:**

1. **summary:** Adding automatic patch validation
description: Proposing introduction of automated patch validation, similar what Hadoop or other Apache projects are using (see link). This would ensure that every patch passes a certain set of criterions before getting approved. It would save time for developer (faster feedback loop), save time for committers (less step to do manually), and would increase quality. Hadoop is currently using Apache Yetus to run validations, which seems to be a good direction to start. This jira could be the board of discussing the preferred solution.

jira_issues_comments:

1. [~manokovacs] - I'm on board with this. There was some discussion already at <https://lists.apache.org/thread.html/21ad23ec5449171739d87681cf2d011f20d39c39420c1e610a3b1751@%3Cdev.lucene.apache.org%3E> but I didn't have time to drive it further.
2. This would be awesome
3. **body:** [~manokovacs] - was there supposed to be a QA run for the patch you attached? Looks like some integration is still missing.
label: test
4. [~mdrob], thanks for the followup :) Well, there is no automated integration yet. I just wanted to have a test-run, but eventually I used SOLR-10783 for that. When I am finished, I upload and run a test-patch for each scenario to demonstrate.
5. (x) *{color:red}-1 overall{color}* | \ \ \ \ \ Vote || Subsystem || Runtime || Comment || \ \ \ \ \ {color:brown} Prechecks {color} || | {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 00s{color} | {color:green} The patch does not contain any @author tags. {color} || | {color:red}-1{color} | {color:red} test4tests {color} | {color:red} 0m 00s{color} | {color:red} The patch doesn't appear to include any new or modified tests. Please justify why no new tests are needed for this patch. Also please list what manual steps were performed to verify this patch. {color} || \ \ \ \ \ {color:brown} master Compile Tests {color} || \ \ \ \ \ {color:brown} Patch Compile Tests {color} || | {color:green}+1{color} | {color:green} Check examples refer correct lucene version {color} | {color:green} 0m 04s{color} | {color:green} the patch passed {color} || | {color:green}+1{color} | {color:green} Release audit (RAT) {color} | {color:green} 0m 04s{color} | {color:green} the patch passed {color} || | {color:green}+1{color} | {color:green} Check forbidden APIs {color} | {color:green} 0m 04s{color} | {color:green} the patch passed {color} || | {color:green}+1{color} | {color:green} Check licenses {color} | {color:green} 0m 04s{color} | {color:green} the patch passed {color} || \ \ \ \ \ {color:brown} Other Tests {color} || | {color:red}-1{color} | {color:red} unit {color} | {color:red} 3m 47s{color} | {color:red} solr in the patch failed. {color} || | {color:black}{color} | {color:black} {color} | {color:black} 7m 01s{color} | {color:black} {color} | \ \ \ \ \ Subsystem || Report/Notes || | JIRA Issue | SOLR-10912 || | JIRA Patch URL | <https://issues.apache.org/jira/secure/attachment/12885918/SOLR-10912.sample-patch.patch> || | Optional Tests | javac unit || | uname | Darwin MunawAir.local 15.6.0 Darwin Kernel Version 15.6.0: Fri Feb 17 10:21:18 PST 2017; root:xnu-3248.60.11.4.1~1/RELEASE_X86_64 x86_64 || | Build tool | ant || | Personality | /Users/munaw/repos/lucene-solr/dev-tools/test-patch/solr-yetus-personality.sh || | git revision | master / 31eab31 || | Default Java | 1.8.0_45 || | unit | artifact/patchprocess/patch-unit-_.solr.txt || | modules | C: solr/solrj U: solr/solrj || | Powered by | Apache Yetus 0.6.0- SNAPSHOT <http://yetus.apache.org> | This message was automatically generated.
6. (x) *{color:red}-1 overall{color}* | \ \ \ \ \ Vote || Subsystem || Runtime || Comment || \ \ \ \ \ {color:brown} Prechecks {color} || | {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 00s{color} | {color:green} The patch does not contain any @author tags. {color} || | {color:red}-1{color} | {color:red} test4tests {color} | {color:red} 0m 00s{color} | {color:red} The patch doesn't appear to include any new or modified tests. Please justify why no new tests are needed for this patch. Also please list what manual steps were performed to verify this patch. {color} || \ \ \ \ \ {color:brown} master Compile Tests {color} || \ \ \ \ \ {color:brown} Patch Compile Tests {color} || | {color:green}+1{color} | {color:green} Check examples refer correct lucene version {color} | {color:green} 0m 06s{color} | {color:green} the patch passed {color} || | {color:green}+1{color} | {color:green} Release audit (RAT) {color} | {color:green} 0m 06s{color} | {color:green} the patch passed {color} || | {color:green}+1{color} | {color:green} Check forbidden APIs {color} | {color:green} 0m 06s{color} | {color:green} the patch passed {color} || | {color:green}+1{color} | {color:green} Check licenses {color} | {color:green} 0m 06s{color} | {color:green} the patch passed {color} || \ \ \ \ \ {color:brown} Other Tests {color} || | {color:red}-1{color} | {color:red} unit {color} | {color:red} 9m 08s{color} | {color:red} solr in the patch failed. {color} || | {color:black}{color} | {color:black} {color} | {color:black} 12m 55s{color} | {color:black} {color} | \ \ \ \ \ Reason || Tests || | Failed junit tests | solr.cloud.AliasIntegrationTest || | solr.cloud.ChaosMonkeyNothingIsSafeWithPullReplicasTest || | solr.cloud.FullSolrCloudDistribCmdsTest || | solr.cloud.MoveReplicaTest || | solr.cloud.MultiThreadedOCPTTest || | solr.cloud.OverseerRolesTest || | solr.cloud.RemoteQueryErrorTest || | solr.cloud.ShardRoutingTest || | solr.cloud.ShardSplitTest || | solr.cloud.TestDeleteCollectionOnDownNodes || | solr.cloud.TestLeaderElectionWithEmptyReplica || | solr.cloud.TestMiniSolrCloudClusterSSL || | solr.cloud.TestSolrCloudWithDelegationTokens || | solr.cloud.TestStressCloudBlindAtomicUpdates || | solr.cloud.TestTolerantUpdateProcessorRandomCloud || | solr.cloud.UnloadDistributedZkTest || | solr.core.snapshots.TestSolrCoreSnapshots || | solr.DistributedIntervalFacetingTest || | solr.handler.PingRequestHandlerTest || | solr.metrics.reporters.solr.SolrShardReporterTest || |

solr.search.facet.TestJsonFacetRefinement ||| solr.search.join.BlockJoinFacetDistribTest ||| solr.search.join.TestCloudNestedDocsSort ||| solr.search.mlt.CloudMLTQParserTest ||| solr.TestDistributedGrouping ||| solr.TestDistributedMissingSort ||| solr.TestDistributedSearch ||| solr.TestTolerantSearch ||| solr.update.processor.TemplateUpdateProcessorTest ||| Subsystem || Report/Notes || JIRA Issue | SOLR-10912 || JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12885918/SOLR-10912.sample-patch.patch || Optional Tests | javac unit | | uname | Darwin MunawAir.local 15.6.0 Darwin Kernel Version 15.6.0: Fri Feb 17 10:21:18 PST 2017; root:xnu-3248.60.11.4.1~1/RELEASE_ARM_T8020 RELEASE_ARM_T8020 || Build tool | ant || Personality | /Users/munaw/repos/lucene-solr/dev-tools/test-patch/solr-yetus-personality.sh || git revision | master / 708ed8f || Default Java | 1.8.0_45 || unit | artifact/patchprocess/patch-unit-_.solr.txt || modules | C: solr/solrj U: solr/solrj || Powered by | Apache Yetus 0.6.0-SNAPSHOT http://yetus.apache.org | This message was automatically generated.

7. bq. solr in the patch failed would be good to see if we can say solr-core or solr-solrj or solr-contrib failed (and similar for lucene)
8. Yeah, that would make sense. Do you think it would be enough just to test the module changed?
9. Uploading a new patch with solj, contrib module, and lucene module changes to verify unittests in changed modules only.
10. There's some dependency work here. A change in solr-core shouldn't need to re-run the solrj unit tests, but a solrj change would benefit from running the solr unit tests. Or do I have that backwards?
11. | (x) *{color:red}-1 overall{color}* | \\ \\ \\ \\ Vote || Subsystem || Runtime || Comment || || || || {color:brown} Prechecks {color} || | {color:green}+1{color} | | {color:green} @author {color} | | {color:green} 0m 00s{color} | | {color:green} The patch does not contain any @author tags. {color} || | {color:red}-1{color} | | {color:red} test4tests {color} | | {color:red} 0m 00s{color} | | {color:red} The patch doesn't appear to include any new or modified tests. Please justify why no new tests are needed for this patch. Also please list what manual steps were performed to verify this patch. {color} || || || || {color:brown} master Compile Tests {color} || || || || {color:brown} Patch Compile Tests {color} || | {color:green}+1{color} | | {color:green} Check examples refer correct lucene version {color} | | {color:green} 0m 05s{color} | | {color:green} the patch passed {color} | | {color:green}+1{color} | | {color:green} Release audit (RAT) {color} | | {color:green} 0m 05s{color} | | {color:green} the patch passed {color} | | {color:green}+1{color} | | {color:green} Check forbidden APIs {color} | | {color:green} 0m 05s{color} | | {color:green} the patch passed {color} | | {color:green}+1{color} | | {color:green} Check licenses {color} | | {color:green} 0m 05s{color} | | {color:green} the patch passed {color} | | || || || {color:brown} Other Tests {color} || | {color:green}+1{color} | | {color:green} unit {color} | | {color:green} 1m 24s{color} | | {color:green} facet in the patch passed. {color} || | {color:red}-1{color} | | {color:red} unit {color} | | {color:red} 2m 46s{color} | | {color:red} analytics in the patch failed. {color} || | {color:red}-1{color} | | {color:red} unit {color} | | {color:red} 4m 25s{color} | | {color:red} solrj in the patch failed. {color} || | {color:black}{color} | | {color:black} {color} | | {color:black} 12m 14s{color} | | {color:black} {color} | \\ \\ \\ \\ Subsystem || Report/Notes || JIRA Issue | SOLR-10912 || JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12886505/SOLR-10912.solj-contrib-facet-error.patch || Optional Tests | javac unit | | uname | Darwin MunawAir.local 15.6.0 Darwin Kernel Version 15.6.0: Fri Feb 17 10:21:18 PST 2017; root:xnu-3248.60.11.4.1~1/RELEASE_ARM_T8020 RELEASE_ARM_T8020 || Build tool | ant || Personality | /Users/munaw/repos/lucene-solr/dev-tools/test-patch/solr-yetus-personality.sh || git revision | master / 708ed8f || Default Java | 1.8.0_45 || unit | artifact/patchprocess/patch-unit-solr_contrib_analytics.txt || unit | artifact/patchprocess/patch-unit-solr_solrj.txt || modules | C: lucene/facet solr/contrib/analytics solr/solrj U: . || Powered by | Apache Yetus 0.6.0-SNAPSHOT http://yetus.apache.org | This message was automatically generated.
12. **body:** The latest test-patch only tested the three module was changed (either my intentional error was not placed correctly, or we need more tests there). bq. A change in solr-core shouldn't need to re-run the solrj unit tests, but a solrj change would benefit from running the solr unit tests. Yeah, solr depends on solrj, so the direction is correct. It is matter of view to say that a module's API should be validated through it's unittests or the tests of other modules depending on it (in this case solr->solrj). But for start, if that won't cause too much load on the jenkins, we can tests rather more than less. Could we define all the transitive triggers for unittest between modules? * solrj->solr-core * lucene->solr? Any other?
label: test
13. One thing I'd like to see added to validation is an indication when patches are missing doc updates (i.e., the patch contains no edits to any .adoc file), assuming that's possible. I don't think the validation should fail if they are missing (so could be a "-0", not impacting the final score), and some patches will not require doc updates which would be hard to for such a tool to be able to determine. But, it would help to ensure possible doc changes are considered at early stages of patch review, and when changes are warranted they'd have a better chance of being included before the final commit of the patch is done.
14. **body:** I agree with Cassandra, and would go even further in saying that maybe a -1 _is_ appropriate. We already have test4tests, why not test4docs? Also, need to make sure that test4tests doesn't complain about doc-only patches.
label: test
15. bq. One thing I'd like to see added to validation is an indication when patches are missing doc updates (i.e., the patch contains no edits to any .adoc file), ... +1 bq. I don't think the validation should fail if they are missing ... some patches will not require doc updates which would be hard to for such a tool to be able to determine. ... I agree, validation should not fail if doc updates are not (yet) present. If doc updates are not (yet) present then it might also be nice for the tool to try and point towards which parts of the documentation might need changing. E.g. if the patch changes the {{ReplicationHandler.java}} class, the word {{ReplicationHandler}} appears currently in three *.adoc files and the tool could just mention them: implicit-requesthandlers.adoc index-replication.adoc making-and-restoring-backups.adoc
16. bq. If doc updates are not (yet) present then it might also be nice for the tool to try and point towards which parts of the documentation might need changing. E.g. if the patch changes the {{ReplicationHandler.java}} class, the word {{ReplicationHandler}} appears currently in three *.adoc files and the tool could just mention them: implicit-requesthandlers.adoc index-replication.adoc making-and-restoring-backups.adoc +1, I like that idea.
17. [~ctargett], [~cpoerschke], [~mdrob], thank you for the inputs. It clearly shows how much potential there is in this tool/method. I am trying to find the MVP for this to start, so with your permission, I would like to move the topic about documentation validation to SOLR-11419. Could you help me specify the details of the validation there? Meanwhile, I am trying to get this into a usable state so we can start some test-run with it.
18. bq. I would like to move the topic about documentation validation to SOLR-11419. Could you help me specify the details of the validation there? +1, sure, we'll tackle it as an add-on once we work out the details so you can still move forward [~manokovacs].
19. | (x) *{color:red}-1 overall{color}* | \\ \\ \\ \\ Vote || Subsystem || Runtime || Comment || || || || {color:brown} Prechecks {color} || | {color:green}+1{color} | | {color:green} @author {color} | | {color:green} 0m 00s{color} | | {color:green} The patch does not contain any @author tags. {color} || | {color:green}+1{color} | | {color:green} test4tests {color} | | {color:green} 0m 00s{color} | | {color:green} The patch appears to include 1 new or modified test files. {color} || || || || {color:brown} master Compile Tests {color} || || || || {color:brown} Patch Compile Tests {color} || | {color:green}+1{color} | | {color:green} Check examples refer correct lucene version {color} | | {color:green} 0m 06s{color} | | {color:green} the patch passed {color} | | {color:green}+1{color} | | {color:green} Release audit (RAT) {color} | | {color:green} 0m 06s{color} | | {color:green} the patch passed {color} | | {color:green}+1{color} | | {color:green} Check

forbidden APIs {color} | {color:green} 0m 06s {color} | {color:green} the patch passed {color} | | {color:green}+1 {color} |
{color:green} Check licenses {color} | {color:green} 0m 06s {color} | {color:green} the patch passed {color} | | | | {color:brown}
Other Tests {color} | | {color:red}-1 {color} | {color:red} unit {color} | {color:red} 255m 09s {color} | {color:red} core in the patch
failed. {color} | | {color:red}-1 {color} | {color:red} unit {color} | {color:red} 13m 11s {color} | {color:red} solrj in the patch failed.
{color} | | {color:black} {color} | {color:black} {color} | {color:black} 272m 02s {color} | {color:black} {color} | | | Reason | Tests |
| Failed junit tests | solr.analysis.TestWordDelimiterFilterFactory | | solr.cloud.CdcrBootstrapTest | | solr.cloud.DeleteStatusTest | |
solr.cloud.FullSolrCloudDistribCmdsTest | | solr.cloud.hdfs.HdfsCollectionsAPIDistributedZkTest | |
solr.cloud.LeaderElectionIntegrationTest | | solr.cloud.ShardRoutingTest | | solr.cloud.ShardSplitTest | |
solr.cloud.SharedFSAutoReplicaFailoverUtilsTest | | solr.cloud.SyncSliceTest | | solr.cloud.TestHdfsCloudBackupRestore | |
solr.cloud.TestLeaderElectionWithEmptyReplica | | solr.cloud.TestLocalFSCloudBackupRestore | |
solr.cloud.TestTolerantUpdateProcessorCloud | | solr.cloud.UnloadDistributedZkTest | | solr.core.TestCorePropertiesReload | |
solr.core.TestCustomStream | | solr.core.TestInfoStreamLogging | | solr.core.TestSimpleTextCodec | |
solr.core.TestSolrDeletionPolicy1 | | solr.handler.loader.JavabinLoaderTest | | solr.handler.TestSolrConfigHandlerCloud | |
solr.handler.V2StandaloneTest | | solr.handler.XsltUpdateRequestHandlerTest | | solr.highlight.TestPostingsSolrHighlighter | |
solr.internal.csv.CharBufferTest | | solr.internal.csv.CSVUtilsTest | | solr.metrics.JvmMetricsTest | | solr.MinimalSchemaTest | |
solr.schema.DateFieldTest | | solr.schema.MultiTermTest | | solr.schema.PolyFieldTest | | solr.schema.TestBinaryField | |
solr.schema.TestBulkSchemaConcurrent | | solr.schema.TestManagedSchema | | solr.schema.UUIDFieldTest | |
solr.search.AnalyticsMergeStrategyTest | | solr.search.function.distance.DistanceFunctionTest | |
solr.search.similarities.TestPerFieldSimilarity | | solr.search.TestPayloadScoreQParserPlugin | | solr.search.TestSolrJ | |
solr.servlet.DirectSolrConnectionTest | | solr.servlet.HttpSolrCallGetCoreTest | | solr.spelling.SpellingQueryConverterTest | |
solr.spelling.suggest.SuggesterTest | | solr.spelling.suggest.TestHighFrequencyDictionaryFactory | |
solr.spelling.suggest.TestPhraseSuggestions | | solr.TestHighlightDedupGrouping | | solr.util.CircularListTest | | solr.util.FileUtilsTest
| | solr.util.PrimUtilsTest | | solr.util.TestUtils | | solr.client.solrj.io.stream.StreamExpressionTest | |
solr.common.cloud.TestCollectionStateWatchers | | solr.analysis.TestWordDelimiterFilterFactory | | solr.cloud.CdcrBootstrapTest | |
solr.cloud.DeleteStatusTest | | solr.cloud.FullSolrCloudDistribCmdsTest | | solr.cloud.hdfs.HdfsCollectionsAPIDistributedZkTest | |
solr.cloud.LeaderElectionIntegrationTest | | solr.cloud.ShardRoutingTest | | solr.cloud.ShardSplitTest | |
solr.cloud.SharedFSAutoReplicaFailoverUtilsTest | | solr.cloud.SyncSliceTest | | solr.cloud.TestHdfsCloudBackupRestore | |
solr.cloud.TestLeaderElectionWithEmptyReplica | | solr.cloud.TestLocalFSCloudBackupRestore | |
solr.cloud.TestTolerantUpdateProcessorCloud | | solr.cloud.UnloadDistributedZkTest | | solr.core.TestCorePropertiesReload | |
solr.core.TestCustomStream | | solr.core.TestInfoStreamLogging | | solr.core.TestSimpleTextCodec | |
solr.core.TestSolrDeletionPolicy1 | | solr.handler.loader.JavabinLoaderTest | | solr.handler.TestSolrConfigHandlerCloud | |
solr.handler.V2StandaloneTest | | solr.handler.XsltUpdateRequestHandlerTest | | solr.highlight.TestPostingsSolrHighlighter | |
solr.internal.csv.CharBufferTest | | solr.internal.csv.CSVUtilsTest | | solr.metrics.JvmMetricsTest | | solr.MinimalSchemaTest | |
solr.schema.DateFieldTest | | solr.schema.MultiTermTest | | solr.schema.PolyFieldTest | | solr.schema.TestBinaryField | |
solr.schema.TestBulkSchemaConcurrent | | solr.schema.TestManagedSchema | | solr.schema.UUIDFieldTest | |
solr.search.AnalyticsMergeStrategyTest | | solr.search.function.distance.DistanceFunctionTest | |
solr.search.similarities.TestPerFieldSimilarity | | solr.search.TestPayloadScoreQParserPlugin | | solr.search.TestSolrJ | |
solr.servlet.DirectSolrConnectionTest | | solr.servlet.HttpSolrCallGetCoreTest | | solr.spelling.SpellingQueryConverterTest | |
solr.spelling.suggest.SuggesterTest | | solr.spelling.suggest.TestHighFrequencyDictionaryFactory | |
solr.spelling.suggest.TestPhraseSuggestions | | solr.TestHighlightDedupGrouping | | solr.util.CircularListTest | | solr.util.FileUtilsTest
| | solr.util.PrimUtilsTest | | solr.util.TestUtils | | solr.client.solrj.io.stream.StreamExpressionTest | |
solr.common.cloud.TestCollectionStateWatchers | | | Subsystem | | Report/Notes | | JIRA Issue | SOLR-10912 | JIRA Patch URL |
<https://issues.apache.org/jira/secure/attachment/12889953/SOLR-10912.ok-patch-in-core.patch> | Optional Tests | javac unit | | uname |
Darwin manomac.local 16.7.0 Darwin Kernel Version 16.7.0: Thu Jun 15 17:36:27 PDT 2017; root:xnu-
3789.70.16~2/RELEASE_ARM64_T8020 Build tool | ant | Personality | /Users/manokovacs/repos/solr-yetus/dev-tools/test-
patch/solr-yetus-personality.sh | | git revision | master / 266cc82df0 | Default Java | 1.8.0_101 | unit | artifact/patchprocess/patch-unit-
solr_core.txt | | unit | artifact/patchprocess/patch-unit-solr_solrj.txt | | modules | C: solr/core solr/solrj U: solr | Powered by | Apache
Yetus 0.6.0-SNAPSHOT <http://yetus.apache.org> | This message was automatically generated.

20. (x) *{color:red}-1 overall {color}* | | | | Vote | Subsystem | Runtime | Comment | | | | {color:brown} Prechecks {color} | |
{color:green}+1 {color} | {color:green} @author {color} | {color:green} 0m 00s {color} | {color:green} The patch does not contain any
@author tags. {color} | | {color:green}+1 {color} | {color:green} test4tests {color} | {color:green} 0m 00s {color} | {color:green} The
patch appears to include 1 new or modified test files. {color} | | | | {color:brown} master Compile Tests {color} | | | |
{color:brown} Patch Compile Tests {color} | | {color:green}+1 {color} | {color:green} Check examples refer correct lucene version
{color} | {color:green} 0m 07s {color} | {color:green} the patch passed {color} | | {color:green}+1 {color} | {color:green} Release audit
(RAT) {color} | {color:green} 0m 07s {color} | {color:green} the patch passed {color} | | {color:green}+1 {color} | {color:green} Check
forbidden APIs {color} | {color:green} 0m 07s {color} | {color:green} the patch passed {color} | | {color:green}+1 {color} |
{color:green} Check licenses {color} | {color:green} 0m 07s {color} | {color:green} the patch passed {color} | | | | {color:brown}
Other Tests {color} | | {color:red}-1 {color} | {color:red} unit {color} | {color:red} 62m 41s {color} | {color:red} core in the patch
failed. {color} | | {color:red}-1 {color} | {color:red} unit {color} | {color:red} 11m 07s {color} | {color:red} solrj in the patch passed.
{color} | | {color:black} {color} | {color:black} {color} | {color:black} 77m 37s {color} | {color:black} {color} | | | Reason | Tests |
| Failed junit tests | solr.cloud.CdcrBootstrapTest | | solr.cloud.FullSolrCloudDistribCmdsTest | | solr.cloud.ReplaceNodeTest | |
solr.cloud.TestCloudRecovery | | solr.cloud.TestLeaderElectionWithEmptyReplica | | solr.cloud.TestRebalanceLeaders | |
solr.cloud.CdcrBootstrapTest | | solr.cloud.FullSolrCloudDistribCmdsTest | | solr.cloud.ReplaceNodeTest | |
solr.cloud.TestCloudRecovery | | solr.cloud.TestLeaderElectionWithEmptyReplica | | solr.cloud.TestRebalanceLeaders | | |
Subsystem | Report/Notes | | JIRA Issue | SOLR-10912 | JIRA Patch URL |
<https://issues.apache.org/jira/secure/attachment/12889953/SOLR-10912.ok-patch-in-core.patch> | Optional Tests | javac unit | | uname |
Darwin manomac.local 16.7.0 Darwin Kernel Version 16.7.0: Thu Jun 15 17:36:27 PDT 2017; root:xnu-
3789.70.16~2/RELEASE_ARM64_T8020 Build tool | ant | Personality | /Users/manokovacs/repos/solr-yetus/dev-tools/test-
patch/solr-yetus-personality.sh | | git revision | master / fd2b4f3f86 | Default Java | 1.8.0_101 | unit | artifact/patchprocess/patch-unit-
solr_core.txt | | unit | artifact/patchprocess/patch-unit-solr_solrj.txt | | modules | C: solr/core solr/solrj U: solr | Powered by | Apache
Yetus 0.6.0-SNAPSHOT <http://yetus.apache.org> | This message was automatically generated.
21. **body:** As you can see from the last comment above, the minimal setup is getting ready. Steps remaining: * Multiple module test
currently duplicates the failed test result, fixing shortly. * Add test to call {{-validate-source-patterns}} * Finalize the runner script to
setup Yetus, similarly to Hadoop. * Verify the test-patch with Github PR * Add Patch Available status for SOLR project (and update the

script to look for that). [~steve_rowe], regarding to our IRL conversation, could you help me with adding that status in JIRA? * Add {{Precommit-SOLR}} and {{Precommit-LUCENE}} jenkins jobs Open questions: # As you can see, a patch with two log entries had 6 (flaky) test failures. Assuming flaky tests will not go away very soon, would it still be useful to have this test-patch? # I propose starting with the smallest set of tests (ie. affected modules) and extend them later to dependent modules.

label: test

22. <delurks> * If the plan is to use <https://builds.apache.org/job/PreCommit-admin/> to trigger patches, the jenkins jobs have to be called "PreCommit-<PROJECT NAME>-Build". * Be aware that hadoop's precommit uses a flat directory structure in its workspace. History has shown that this is less than ideal. At some point I'll likely rework it so that's more structured like hadoop's qbt jobs. <relurks>
23. bq. Add Patch Available status for SOLR project [...] Steve Rowe, regarding to our IRL conversation, could you help me with adding that status in JIRA? I don't have karma to make this change myself, so I asked on Infrastructure's hipchat channel, and Daniel Takamori said I should create a JIRA to ask for this: INFRA-15213.
24. Thank you [~aw], [~steve_rowe]!
25. [~manokovacs]: I'd like to push this forward. What are the next steps here? Where can I find your code that produced the reports in comments here?
26. Hey Steve, that would be great as I couldn't work on this lately and probably will be busy in the next weeks. The current change is here: [<https://github.com/apache/lucene-solr/compare/master...manokovacs:feature/SOLR-10912>] Also, I have the list with the remaining work I was planning to do and the open questions. {quote}Steps remaining: * Multiple module test currently duplicates the failed test result, fixing shortly. * Add test to call {{-validate-source-patterns}} * Finalize the runner script to setup Yetus, similarly to Hadoop. * Verify the test-patch with Github PR * Add Patch Available status for SOLR project (and update the script to look for that). [~steve_rowe], regarding to our IRL conversation, could you help me with adding that status in JIRA? * Add {{Precommit-SOLR}} and {{Precommit-LUCENE}} jenkins jobs Open questions: 1. As you can see, a patch with two log entries had 6 (flaky) test failures. Assuming flaky tests will not go away very soon, would it still be useful to have this test-patch? 2. I propose starting with the smallest set of tests (ie. affected modules) and extend them later to dependent modules. {quote} Let me know if I can be of any help.
27. comment/request from someone who hasn't really followed the progress here much but does use jira every day... can we *PLEASE* rename the "Submit Patch" button to something else since it does not actually give you the power to "submit a patch" -- ie: clicking that button does nothing to enable me to upload a patch file from my computer. that functionality is burried in the "More" menu as "Attach Files" AFAICT the _true_ functionality of the current button is "Submit issue for automated Patch Review" -- so can we please change the button text to say that? Likewise i would suggest changing the "Cancel Patch" button to something like "Cancel Automated Patch Review" (If it's _also_ possible to add a top level button that *actually* triggers the "Attach Files" file selector, then by all means let's do that and give it text like "Submit Patch" or "Attach New Path")
28. bq. can we PLEASE rename the "Submit Patch" button +1, I've also found myself confused by that option, though I didn't realize it was tied to this JIRA until now.
29. bq. can we PLEASE rename the "Submit Patch" button to something else since it does not actually give you the power to "submit a patch" See INFRA-16094
30. I'm confused what the state is right now. In JIRA I see these new workflow states. Do they do anything right now, or _not yet_ pending further TODOs above? Will this feature support both JIRA attached patch files as well as linked GitHub PRs (e.g. as seen here: SOLR-11976) ?
31. bq. I'm confused what the state is right now. In JIRA I see these new workflow states. Do they do anything right now, or not yet pending further TODOs above? Not yet. I'm working on it. bq. Will this feature support both JIRA attached patch files as well as linked GitHub PRs (e.g. as seen here: SOLR-11976) ? Attached JIRA patch files, yes. I don't know about GitHub PRs... [~manokovacs] ? (Note his todo item, which sounds like the answer is yes for PRs: "Verify the test-patch with Github PR")
32. Github PR support is sort of there. test-patch does. It can take either a github PR directly on the command line or passed via a JIRA. If it gets told to test a JIRA that references a github PR, it will defer to the PR as the source of the patch. In other words, if a JIRA issue references a github PR and has a patch attached, it will use the github PR and ignore the attachments. However! The job on Jenkins that feeds test-patch is *NOT* github aware. The original version was built before github integration existed. To make matters worse, that code was locked away in a repository no one really had access to modify. As of a month or so ago, that code is now part of Apache Yetus (<https://github.com/apache/yetus/blob/master/precommit/jenkins/jenkins-admin.py>), so there is an opportunity for us to fix this problem and add better asf<->github integration.
33. Some thoughts: If there's not an entry in CHANGES.txt that mentions the issue number (either the lucene or solr file as appropriate), that should be a -1. How about a -1 if a SOLR patch makes changes to lucene, or vice versa? If there is an entry in the appropriate CHANGES.txt file for the issue, turn that into a -0. That way, we have better assurance that if a commit for one part of the project requires changes to the other part, there will be a release note. I'm pretty sure that votes made by this QA mechanism wouldn't be binding, but it would be a good idea to achieve a +1 from it if possible, and when it's not, there should be a very good and well-documented reason.
34. bq. If there's not an entry in CHANGES.txt that mentions the issue number (either the lucene or solr file as appropriate), that should be a -1. I think -0 is warranted, but not -1; some committers' workflows order CHANGES additions after the initial commits, and non-committers rarely include CHANGES entries (maybe partly because committers have to change it, minimally to include their name). bq. How about a -1 if a SOLR patch makes changes to lucene, or vice versa? If there is an entry in the appropriate CHANGES.txt file for the issue, turn that into a -0. That way, we have better assurance that if a commit for one part of the project requires changes to the other part, there will be a release note. Some issues require changes in both places. Is there some issue you're trying to address besides release noting both projects? I ask because Solr users really need to pay attention to Lucene CHANGES regardless.
35. On what to do if there's no CHANGES.txt entry: The goal is to make sure the changelog update doesn't fall through the cracks and get forgotten. If the -0 accomplishes that, then that's great. bq. Some issues require changes in both places. Is there some issue you're trying to address besides release noting both projects? I ask because Solr users really need to pay attention to Lucene CHANGES regardless. I'm not trying to prevent one issue changing both sections of the codebase, just trying to make sure that if it happens, there's an explicit note in both changelogs. Obviously a lot of LUCENE changes directly affect Solr even if they don't change Solr code, and for those, only updating the Lucene changelog is the right thing to do, and it's up to Solr users to read Lucene's changelog. But if an issue on one part of the project actually makes changes to the code in the other, I think the documentation bar should be a little bit higher. Additional idea: Record a -1 if a SOLR issue *only* makes changes to Lucene code, or vice versa. But when this happens, a changelog entry won't be enough to adjust to -0.
36. **body:** {quote}The current change is here: [<https://github.com/apache/lucene-solr/compare/master...manokovacs:feature/SOLR-10912>] {quote} I've attached a patch with a modified version of Mano's branch, and I plan on committing it shortly in order to start testing the Jenkins job I've set up at [<https://builds.apache.org/view/PreCommit+Builds/job/PreCommit-LUCENE-Build/>] (the script in this job's config, a copy of which is in the patch, expects the Yetus personality to be committed to the Git repo). h2. From Mano's "Steps

remaining": {quote}Multiple module test currently duplicates the failed test result, fixing shortly. {quote} Fixed. {quote}Add test to call -validate-source-patterns {quote} Added. FYI, I had to rename the Ant task to remove the leading dash so that it can be invoked from the cmdline. {quote}Finalize the runner script to setup Yetus, similarly to Hadoop. {quote} Done. {quote}Verify the test-patch with Github PR {quote} Since [~aw] wrote above that "The job on Jenkins that feeds test-patch is NOT github aware", I don't plan on doing this verification. I'll include this on a TODO list below. {quote}Add Patch Available status for SOLR project (and update the script to look for that). {quote} Done. {quote}Add Precommit-SOLR and Precommit-LUCENE jenkins jobs {quote} I've added a PreCommit-LUCENE-Build job (linked above), and once I've committed the patch I've attached here (on master only initially), I'll manually invoke it for testing (via "Build with Parameters"). h2. From Mano's "Open questions": {quote}1. As you can see, a patch with two log entries had 6 (flaky) test failures. Assuming flaky tests will not go away very soon, would it still be useful to have this test-patch? {quote} I think now is a perfect time to do this, given the current efforts focused on reducing test flakiness. {quote}2. I propose starting with the smallest set of tests (ie. affected modules) and extend them later to dependent modules. {quote} +1. h2. Stuff I changed from Mano's branch that is not already mentioned above: # Renamed the personality file (was {{solr-yetus-personality.sh}}, now {{lucene-solr-yetus-personality.sh}}). # Improved handling of Lucene modules. # Added basic ref guide validation, via {{ant bare-bones-html-validation}}; note that this is not the kind of missing-doc validation discussed above in this issue; that idea was spun off as SOLR-11419 # Standardized Lucene/Solr-specific plugins to make them run only if they need to. # Added some user- and dev-level documentation to the local runner script and personality files. h2. Short-term TODO: # Commit current patch to master # Manually test the Jenkins precommit job # Iterate above two steps until testing is successful # Backport the patch to branch_7x # Copy the Precommit-LUCENE-Build job to a new PreCommit-SOLR-Build job. # Request ASF Infrastructure to add LUCENE and SOLR to the list of projects that use the PreCommit-Admin Jenkins job to enqueue precommit runs for new patches on LUCENE/SOLR JIRAs with the "Patch Available" state. (I'll make a JIRA for this and link it to this issue.) # Attach a patch with the finalized Lucene/Solr personality on YETUS-537, for inclusion in future Yetus releases *edit*: Added Shawn's improvement ideas to the TODO list below: h2. Longer-term TODO (to be dealt with on further issues): # Solr missing-doc validation: SOLR-11419. # Add handling of module dependencies and build ordering. # Currently when any test or non-test file is changed in a module, all unit tests are run in that module. I think a faster version of this is possible when there are test-only changes: just the changed test suites could be run, rather than all of the module's tests. # If there's not an entry in CHANGES.txt that mentions the issue number (either the lucene or solr file as appropriate), that should be a -1 (or maybe -0?) # How about a -1 if a SOLR patch makes changes to lucene, or vice versa? If there is an entry in the appropriate CHANGES.txt file for the issue, turn that into a -0. That way, we have better assurance that if a commit for one part of the project requires changes to the other part, there will be a release note.

label: code-design

37. Commit 12372530a8366ab35834b8a93c39775ab87564ed in lucene-solr's branch refs/heads/master from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=1237253>] SOLR-10912: Add scripts for automatic patch validation
38. Commit cc1ad49e72bb6753997711e3cd7fb05d3c319598 in lucene-solr's branch refs/heads/master from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=cc1ad49>] SOLR-10912: excluding apparently unnecessary plugins: 'junit' and 'javac' (the 'unit' and 'compile' plugins are producing output but the 'j' ones aren't)
39. Commit 63fde153d3ca195f07a7fc2cc5999327411f3cc5 in lucene-solr's branch refs/heads/master from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=63fde15>] SOLR-10912: updating copy of Jenkins precommit job script: proc max limit code seems not to work on jenkins slaves, so put it at a fixed 10k; added customization of artifact url so console output links in the JIRA comment report work properly; no longer attempting to cache the yetus download, since it always downloads every time anyway.
40. Commit bd20e36d2dd58cb291b0cb3b59bd9e240136a5d2 in lucene-solr's branch refs/heads/master from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=bd20e36>] SOLR-10912: attempted personality plugins fix: trying just junit/javac instead of unit/compile (which didn't actually do anything at all)
41. **body:** compile does a basic compile test. javac looks in the output log for specific java warnings/errors generated from the compile phase when java's lint mode is activated.
label: code-design
42. Commit ac9559d70de5e2454e0515a08774890ad4012eef in lucene-solr's branch refs/heads/master from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=ac9559d>] SOLR-10912: reverting personality plugins changes to include junit+unit and javac+compile, since this combo works, and neither one individually does.
43. Commit 298063eee7d9f43cea1321b4f26ad58846ba28a2 in lucene-solr's branch refs/heads/master from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=298063e>] SOLR-10912: personality fix: in solr-ref-guide, don't run compile and unit plugins. Also, consistently use curly brackets when interpolating variables
44. bq. compile does a basic compile test. javac looks in the output log for specific java warnings/errors generated from the compile phase when java's lint mode is activated. Thanks [~aw], good to know. Unfortunately the plugins are pretty lightly documented: AFAICT only a subset of the built-in plugins' individual *methods* are documented in the api ref, with no whole-builtin-plugin docs I could find; I haven't dug into the source much yet, other than test-patch.sh itself and some of the bundled example personalities.
45. Commit 3e0d245cf1152fa12ebb52a846bb3deff2d6ea7f in lucene-solr's branch refs/heads/branch_7x from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=3e0d245>] SOLR-10912: Add scripts for automatic patch validation
46. Commit dd3ace8ed68987bfc5562474bac455b923e2e14f in lucene-solr's branch refs/heads/branch_7x from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=dd3ace8>] SOLR-10912: excluding apparently unnecessary plugins: 'junit' and 'javac' (the 'unit' and 'compile' plugins are producing output but the 'j' ones aren't)
47. Commit 285bc554a6bdd0dec686ad4e5b02256836ce61e6 in lucene-solr's branch refs/heads/branch_7x from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=285bc55>] SOLR-10912: updating copy of Jenkins precommit job script: proc max limit code seems not to work on jenkins slaves, so put it at a fixed 10k; added customization of artifact url so console output links in the JIRA comment report work properly; no longer attempting to cache the yetus download, since it always downloads every time anyway.
48. Commit 077f19bdc5bc6b4a27d5604f6e83f166457374a7 in lucene-solr's branch refs/heads/branch_7x from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=077f19b>] SOLR-10912: attempted personality plugins fix: trying just junit/javac instead of unit/compile (which didn't actually do anything at all)
49. Commit a3980add396e878b8b5693c8f8b6e0b89a0ebda0 in lucene-solr's branch refs/heads/branch_7x from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=a3980ad>] SOLR-10912: reverting personality plugins changes to include junit+unit and javac+compile, since this combo works, and neither one individually does.
50. Commit 1c31773fa303e5a48efb1e8af3efae8cd844d760 in lucene-solr's branch refs/heads/branch_7x from [~steve_rowe] [<https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=1c31773>] SOLR-10912: personality fix: in solr-ref-guide, don't run compile and unit plugins. Also, consistently use curly brackets when interpolating variables
51. bq. 6. Request ASF Infrastructure to add LUCENE and SOLR to the list of projects that use the PreCommit-Admin Jenkins job to enqueue precommit runs for new patches on LUCENE/SOLR JIRAs with the "Patch Available" state. (I'll make a JIRA for this and link

it to this issue.) Done: INFRA-16194

52. **body:** The short-term TODO is all taken care of: {quote} # Commit current patch to master # Manually test the Jenkins precommit job # Iterate above two steps until testing is successful # Backport the patch to branch_7x # Copy the Precommit-LUCENE-Build job to a new PreCommit-SOLR-Build job. # Request ASF Infrastructure to add LUCENE and SOLR to the list of projects that use the PreCommit-Admin Jenkins job to enqueue precommit runs for new patches on LUCENE/SOLR JIRAs with the "Patch Available" state. (I'll make a JIRA for this and link it to this issue.) # Attach a patch with the finalized Lucene/Solr personality on YETUS-537, for inclusion in future Yetus releases {quote} I'll leave this issue open until after automatic validation has started.
label: code-design
53. {quote} bq. 6. Request ASF Infrastructure to add LUCENE and SOLR to the list of projects that use the PreCommit-Admin Jenkins job to enqueue precommit runs for new patches on LUCENE/SOLR JIRAs with the "Patch Available" state. (I'll make a JIRA for this and link it to this issue.) Done: INFRA-16194 {quote} This is now completed. The {{PreCommit-Admin}} job is scheduled to run every 10 minutes (can stretch to 40 minutes or longer though, depending on executor availability), and in the first runs after INFRA-16194 was done, two Lucene/Solr qualifying issues (i.e. with "Patch Available" status and updated some time in the last 2 weeks) were submitted: LUCENE-8197 and SOLR-11331. Unfortunately, I had not properly configured the auth token on the {{PreCommit-Admin}} jobs -- {{PreCommit-Admin}} always supplies token 'hadoopqa' when it triggers all {{PreCommit-*}} jobs, and I had configured the jobs to expect 'lucenesolrqa'; I've since fixed this -- and as a result the builds didn't kick off, but {{PreCommit-Admin}}'s database of submitted patches now includes the attachments that were submitted as already dealt with, so those patches won't be validated until somebody uploads new patches there. I re-opened and switched status to "Patch Available" on the two test issues I created to manually test the new {{PreCommit}} jobs (LUCENE-8210 and SOLR-12106). {{PreCommit-Admin}} has now run again and has queued the corresponding {{PreCommit}} jobs to validate the patches on those two issues (once they run the results will be available at [https://builds.apache.org/view/PreCommit%20Builds/job/PreCommit-LUCENE-Build/10/] and [https://builds.apache.org/view/PreCommit%20Builds/job/PreCommit-SOLR-Build/6/]).
54. | (x) *{color:red}-1 overall{color}* | \\ \\ || Vote || Subsystem || Runtime || Comment || | {color:red}-1{color} | {color:red} patch {color} | {color:red} 0m 6s{color} | {color:red} SOLR-10912 does not apply to master. Rebase required? Wrong Branch? See https://wiki.apache.org/solr/HowToContribute#Creating_the_patch_file for help. {color} | \\ \\ || Subsystem || Report/Notes || | JIRA Issue | SOLR-10912 | | JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12914632/SOLR-10912.patch | | Console output | https://builds.apache.org/job/PreCommit-SOLR-Build/7/console | | Powered by | Apache Yetus 0.7.0 http://yetus.apache.org | This message was automatically generated.
55. bq. I re-opened and switched status to "Patch Available" on the two test issues I created to manually test the new PreCommit jobs (LUCENE-8210 and SOLR-12106). PreCommit-Admin has now run again and has queued the corresponding PreCommit jobs to validate the patches on those two issues (once they run the results will be available at https://builds.apache.org/view/PreCommit%20Builds/job/PreCommit-LUCENE-Build/10/ and https://builds.apache.org/view/PreCommit%20Builds/job/PreCommit-SOLR-Build/6/). Those runs finished as expected, after waiting about 14 hours in the queue. I think it would be good for this kind of job to be executed ASAP rather than waiting their turn, but I don't know how to set that up. The patch on this issue was also run under {{PreCommit-SOLR-Build}}, as the auto comment above shows. (I assume the patch didn't apply because it was already committed/pushed.) I'll resolve this issue for now. We can reopen (or create a new issue) if problems surface later.
56. Commit c6ef6b67b10c74e9e427860873320f0d77b3fb3b in lucene-solr's branch refs/heads/branch_7x from [~steve_rowe] [https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=c6ef6b6] SOLR-10912: fix routing of Solr non-contrib build output dirs (e.g. solr/core -> ../build/solr-core; previously -> ../build/core)
57. Commit 51a6bec48d0bf6d9b972c69ba87a12ac44f485e4 in lucene-solr's branch refs/heads/master from [~steve_rowe] [https://git-wip-us.apache.org/repos/asf?p=lucene-solr.git;h=51a6bec] SOLR-10912: fix routing of Solr non-contrib build output dirs (e.g. solr/core -> ../build/solr-core; previously -> ../build/core)
58. [~sarowe], regarding ??Since Allen Wittenauer wrote above that "The job on Jenkins that feeds test-patch is NOT github aware", I don't plan on doing this verification. I'll include this on a TODO list below.?? Can it be YETUS-614, which is fixed in Yetus 0.8.0? Note, I tried to use github PR link in precommit, but got just this https://builds.apache.org/job/PreCommit-SOLR-Build/641/console {code:java} [Sat Dec 28 08:14:15 UTC 2019 DEBUG]: jira_http_fetch: https://issues.apache.org/jira/browse/SOLR-12490 [Sat Dec 28 08:14:17 UTC 2019 DEBUG]: jira_locate_patch: SOLR-12490 seemed like a Github PR, but there was a failure. {code} How we can evaluate Yetus 0.8.0 or the latest one?