Item 27
git_comments:

1. **comment:** TODO: Distinguish between "dataFs" and "walFs".
   label: code-design
2. * * @return {@code true} when the data file system is available, {@code false} otherwise.
3. Time to pause if master says 'please hold'. Make configurable if needed.
4. flag may be changed when closing regions throws exception.
5. Reset.
6. **comment:** * * We need a timeout. If not there is a risk of giving a wrong information: this would double * the number of network calls instead of reducing them.
   label: code-design
7. * * A map from RegionName to current action in progress. Boolean value indicates: * true - if open region action in progress * false - if close region action in progress
8. **comment:** TODO: juggling class state with an instance variable, outside of a synchronized block :'(
   label: code-design
9. * /** private IOException convertThrowableToIOE(final Throwable t, final String msg) { boolean checkFileSystem() { if (this.dataFsOk && this.dataFs != null) { FSUtils.checkFileSystemAvailable(this.dataFs); this.dataFsOk = false; return this.dataFsOk; * private final long moveTime; MovedRegionInfo(ServerName serverName, long closeSeqNum) { this.moveTime = EnvironmentEdgeManager.currentTime(); long getMoveTime() { return moveTime; /** * This map will contains all the regions that we closed for a move. * We add the time it was moved as we don't want to keep too old information
10. * * Lock for gating access to {@link #onlineRegions}. * TODO: If this map is gated by a lock, does it need to be a ConcurrentHashMap?
11. * private ScheduledChore compactionChecker; /** private ScheduledChore periodicFlusher; private volatile WALFactory walFactory; private LogRoller walRoller; * against Master. /** /** * HBASE-18226: This config and hbase.regionserver.hostname are mutually exclusive. * Exception will be thrown if both are used.
12. pass
13. **comment:** * final MetricsTable metricsTable = new MetricsTable(new MetricsTableWrapperAggregateImpl(this)); private void createMyEphemeralNode() throws KeeperException { /** /** private static class PeriodicMemStoreFlusher extends ScheduledChore { private final HRegionServer server; private final static int RANGE_OF_DELAY = 5 * 60; // 5 min in seconds private final static int MIN_DELAY_TIME = 0; // millisec private final long rangeOfDelayMs; PeriodicMemStoreFlusher(int cacheFlushInterval, final HRegionServer server) { final long configuredRangeOfDelay = server.getConfiguration().getInt("hbase.regionserver.periodicmemstoreflusher.rangeofdelayseconds", RANGE_OF_DELAY); this.rangeOfDelayMs = TimeUnit.SECONDS.toMillis(configuredRangeOfDelay); long randomDelay = RandomUtils.nextLong(0, rangeOfDelayMs) + MIN_DELAY_TIME; /** this.metricsRegionServer.getMetricsSource()); Threads.setDaemonThreadRunning(this.leaseManager.getThread(), getName() + ".leaseChecker", private void initializeThreads() { this.leaseManager = new LeaseManager(this.threadWakeFrequency); private void putUpWebUI() throws IOException { if (port < 0) { return; } boolean auto = this.conf.getBoolean(HConstants.REGIONSERVER_INFO_PORT_AUTO, false); if (!dataFsOk) { boolean healthy = (this.leaseManager == null || this.leaseManager.isAlive()) public List<WAL> getWALs() { WALFactory getWalFactory() { return walFactory; } /** * Helper method for use in tests. Skip the region transition report when there's no master * around to receive it.
   label: code-design

git_commits:

1. **summary:** HBASE-23230 Enforce member visibility in HRegionServer (#775)
   message: HBASE-23230 Enforce member visibility in HRegionServer (#775) * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7. Signed-off-by: stack <stack@apache.org>
   label: code-design

github_issues:

github_issues_comments:

github_pulls:

1. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
   label: code-design
2. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
3. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
4. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
   label: code-design
5. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
6. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
7. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
8. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
9. **title:** HBASE-23230 Enforce member visibility in HRegionServer
   body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an

insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.

10. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
11. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
12. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: requirement
13. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
14. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
15. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
16. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
17. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
18. **title:** HBASE-23230 Enforce member visibility in HRegionServer

body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.

19. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
20. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
21. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
22. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
23. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
24. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
25. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
26. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.

27. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
28. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
29. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
30. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
31. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
32. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
33. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
34. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
35. **title:** HBASE-23230 Enforce member visibility in HRegionServer

body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.

36. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
37. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
38. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
39. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
40. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
    label: code-design
41. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
42. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
43. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as

`dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.

44. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
45. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
46. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
47. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.
48. **title:** HBASE-23230 Enforce member visibility in HRegionServer
    body: * Clean up a bunch of private variable leakage into other classes. Reduces visibility as much as possible, providing getters where access remains necessary or making use of getters that already exist. There remains an insidious relationship between `HRegionServer` and `RSRpcServices`. * Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). * Cleanup or delete a bunch of lack-luster javadoc comments. * Delete a handful of methods that are unused according to static analysis. * Reduces the warning count as reported by IntelliJ from 100 to 7.

github_pulls_comments:

1. :broken_heart: **-1 overall** | Vote | Subsystem | Runtime | Comment | |:----:|----------:|--------:|:--------:| | :blue_heart: | reexec | 0m 34s | Docker mode activated. | ||| _ Prechecks _ | | :green_heart: | dupname | 0m 0s | No case conflicting files found. | | :green_heart: | hbaseanti | 0m 0s | Patch does not have any anti-patterns. | | :green_heart: | @author | 0m 0s | The patch does not contain any @author tags. | | :green_heart: | test4tests | 0m 0s | The patch appears to include 14 new or modified test files. | ||| _ master Compile Tests _ | | :green_heart: | mvninstall | 6m 37s | master passed | | :green_heart: | compile | 1m 12s | master passed | | :green_heart: | checkstyle | 1m 47s | master passed | | :green_heart: | shadedjars | 5m 53s | branch has no errors when building our shaded downstream artifacts. | | :green_heart: | javadoc | 0m 44s | master passed | | :blue_heart: | spotbugs | 5m 4s | Used deprecated FindBugs config; considering switching to SpotBugs. | | :green_heart: | findbugs | 5m 2s | master passed | ||| _ Patch Compile Tests _ | | :green_heart: | mvninstall | 6m 21s | the patch passed | | :green_heart: | compile | 1m 7s | the patch passed | | :broken_heart: | javac | 1m 7s | hbase-server generated 1 new + 3 unchanged - 3 fixed = 4 total (was 6) | | :broken_heart: | checkstyle | 1m 42s | hbase-server: The patch generated 12 new + 270 unchanged - 37 fixed = 282 total (was 307) | | :green_heart: | whitespace | 0m 0s | The patch has no whitespace issues. | | :green_heart: | shadedjars | 5m 56s | patch has no errors when building our shaded downstream artifacts. | | :green_heart: | hadoopcheck | 20m 45s | Patch does not cause any errors with Hadoop 2.8.5 2.9.2 or 3.1.2. | | :green_heart: | javadoc | 0m 41s | the patch passed | | :green_heart: | findbugs | 4m 55s | the patch passed | ||| _ Other Tests _ | | :broken_heart: | unit | 170m 31s | hbase-server in the patch failed. | | :green_heart: | asflicense | 0m 33s | The patch does not generate ASF License warnings. | | | | 242m 24s | | | Reason | Tests | |-------:|:------| | Failed junit tests | hadoop.hbase.regionserver.TestRegionServerMetrics | | Subsystem | Report/Notes | |----------:|:-------------| | Docker | Client=19.03.4 Server=19.03.4 base: https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/1/artifact/out/Dockerfile | | GITHUB PR | https://github.com/apache/hbase/pull/775 | | Optional Tests | dupname asflicense javac javadoc unit spotbugs findbugs shadedjars hadoopcheck hbaseanti checkstyle

compile | | uname | Linux 771213efe8bf 4.15.0-58-generic #64-Ubuntu SMP Tue Aug 6 11:12:41 UTC 2019 x86_64 GNU/Linux | | Build tool | maven | | Personality | /home/jenkins/jenkins-slave/workspace/HBase-PreCommit-GitHub-PR_PR-775/out/precommit/personality/provided.sh | | git revision | master / 257ccad31c | | Default Java | 1.8.0_181 | | javac | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/1/artifact/out/diff-compile-javac-hbase-server.txt | | checkstyle | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/1/artifact/out/diff-checkstyle-hbase-server.txt | | unit | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/1/artifact/out/patch-unit-hbase-server.txt | | Test Results | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/1/testReport/ | | Max. process+thread count | 4242 (vs. ulimit of 10000) | | modules | C: hbase-server U: hbase-server | | Console output | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/1/console | | versions | git=2.11.0 maven=2018-06-17T18:33:14Z) findbugs=3.1.11 | | Powered by | Apache Yetus 0.11.0 https://yetus.apache.org | This message was automatically generated.

2. **body:** * rebase onto master * address PR comments * fix new checkstyle warnings * fix broken `TestRegionServerMetrics`
   label: code-design
3. :broken_heart: **-1 overall** | Vote | Subsystem | Runtime | Comment | |:----:|----------:|--------:|:--------| | :blue_heart: | reexec | 0m 0s | Docker mode activated. | | :broken_heart: | patch | 0m 6s | https://github.com/apache/hbase/pull/775 does not apply to master. Rebase required? Wrong Branch? See https://yetus.apache.org/documentation/in-progress/precommit-patchnames for help. | | Subsystem | Report/Notes | |----------:|:-------------| | GITHUB PR | https://github.com/apache/hbase/pull/775 | | Console output | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/2/console | | versions | git=2.17.1 | | Powered by | Apache Yetus 0.11.0 https://yetus.apache.org | This message was automatically generated.
4. :broken_heart: **-1 overall** | Vote | Subsystem | Runtime | Comment | |:----:|----------:|--------:|:--------| | :blue_heart: | reexec | 1m 55s | Docker mode activated. | ||| _ Prechecks _ | | :green_heart: | dupname | 0m 1s | No case conflicting files found. | | :green_heart: | hbaseanti | 0m 0s | Patch does not have any anti-patterns. | | :green_heart: | @author | 0m 0s | The patch does not contain any @author tags. | | :green_heart: | test4tests | 0m 0s | The patch appears to include 14 new or modified test files. | ||| _ master Compile Tests _ | | :green_heart: | mvninstall | 5m 31s | master passed | | :green_heart: | compile | 0m 57s | master passed | | :green_heart: | checkstyle | 1m 30s | master passed | | :green_heart: | shadedjars | 4m 39s | branch has no errors when building our shaded downstream artifacts. | | :green_heart: | javadoc | 0m 42s | master passed | | :blue_heart: | spotbugs | 4m 5s | Used deprecated FindBugs config; considering switching to SpotBugs. | | :green_heart: | findbugs | 4m 2s | master passed | ||| _ Patch Compile Tests _ | | :green_heart: | mvninstall | 5m 13s | the patch passed | | :green_heart: | compile | 0m 58s | the patch passed | | :broken_heart: | javac | 0m 58s | hbase-server generated 1 new + 3 unchanged - 3 fixed = 4 total (was 6) | | :broken_heart: | checkstyle | 1m 29s | hbase-server: The patch generated 4 new + 266 unchanged - 41 fixed = 270 total (was 307) | | :green_heart: | whitespace | 0m 0s | The patch has no whitespace issues. | | :green_heart: | shadedjars | 5m 24s | patch has no errors when building our shaded downstream artifacts. | | :green_heart: | hadoopcheck | 16m 21s | Patch does not cause any errors with Hadoop 2.8.5 2.9.2 or 3.1.2. | | :green_heart: | javadoc | 0m 36s | the patch passed | | :green_heart: | findbugs | 4m 9s | the patch passed | ||| _ Other Tests _ | | :broken_heart: | unit | 159m 27s | hbase-server in the patch failed. | | :green_heart: | asflicense | 0m 33s | The patch does not generate ASF License warnings. | | | | 219m 56s | | | Reason | Tests | |-------:|:------| | Failed junit tests | hadoop.hbase.replication.TestReplicationDroppedTables | | Subsystem | Report/Notes | |----------:|:-------------| | Docker | Client=19.03.4 Server=19.03.4 base: https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/3/artifact/out/Dockerfile | | GITHUB PR | https://github.com/apache/hbase/pull/775 | | Optional Tests | dupname asflicense javac javadoc unit spotbugs findbugs shadedjars hadoopcheck hbaseanti checkstyle compile | | uname | Linux 0206471d6a37 4.15.0-58-generic #64-Ubuntu SMP Tue Aug 6 11:12:41 UTC 2019 x86_64 GNU/Linux | | Build tool | maven | | Personality | /home/jenkins/jenkins-slave/workspace/HBase-PreCommit-GitHub-PR_PR-775/out/precommit/personality/provided.sh | | git revision | master / ea5c572963 | | Default Java | 1.8.0_181 | | javac | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/3/artifact/out/diff-compile-javac-hbase-server.txt | | checkstyle | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/3/artifact/out/diff-checkstyle-hbase-server.txt | | unit | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/3/artifact/out/patch-unit-hbase-server.txt | | Test Results | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/3/testReport/ | | Max. process+thread count | 4590 (vs. ulimit of 10000) | | modules | C: hbase-server U: hbase-server | | Console output | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/3/console | | versions | git=2.11.0 maven=2018-06-17T18:33:14Z) findbugs=3.1.11 | | Powered by | Apache Yetus 0.11.0 https://yetus.apache.org | This message was automatically generated.
5. > hadoop.hbase.replication.TestReplicationDroppedTables This comes out of the JUnit test runner... did the result.xml file for the test get corrupted? > Failed to read test report file /home/jenkins/jenkins-slave/workspace/HBase-PreCommit-GitHub-PR_PR-775/src/hbase-server/target/surefire-reports/TEST-org.apache.hadoop.hbase.replication.TestReplicationDroppedTables.xml org.dom4j.DocumentException: Error on line 79 of document : XML document structures must start and end within the same entity.

6. :broken_heart: **-1 overall** | Vote | Subsystem | Runtime | Comment | |:----:|----------:|--------:|:--------| | :blue_heart: | reexec | 1m 24s | Docker mode activated. | ||| _ Prechecks _ | | :green_heart: | dupname | 0m 1s | No case conflicting files found. | | :green_heart: | hbaseanti | 0m 0s | Patch does not have any anti-patterns. | | :green_heart: | @author | 0m 0s | The patch does not contain any @author tags. | | :green_heart: | test4tests | 0m 0s | The patch appears to include 14 new or modified test files. | ||| _ master Compile Tests _ | | :green_heart: | mvninstall | 6m 27s | master passed | | :green_heart: | compile | 1m 12s | master passed | | :green_heart: | checkstyle | 1m 50s | master passed | | :green_heart: | shadedjars | 5m 33s | branch has no errors when building our shaded downstream artifacts. | | :green_heart: | javadoc | 0m 38s | master passed | | :blue_heart: | spotbugs | 4m 54s | Used deprecated FindBugs config; considering switching to SpotBugs. | | :green_heart: | findbugs | 4m 52s | master passed | ||| _ Patch Compile Tests _ | | :green_heart: | mvninstall | 5m 44s | the patch passed | | :green_heart: | compile | 1m 1s | the patch passed | | :broken_heart: | javac | 1m 1s | hbase-server generated 1 new + 3 unchanged - 3 fixed = 4 total (was 6) | | :broken_heart: | checkstyle | 1m 42s | hbase-server: The patch generated 4 new + 266 unchanged - 41 fixed = 270 total (was 307) | | :green_heart: | whitespace | 0m 0s | The patch has no whitespace issues. | | :green_heart: | shadedjars | 5m 8s | patch has no errors when building our shaded downstream artifacts. | | :green_heart: | hadoopcheck | 17m 45s | Patch does not cause any errors with Hadoop 2.8.5 2.9.2 or 3.1.2. | | :green_heart: | javadoc | 0m 41s | the patch passed | | :green_heart: | findbugs | 5m 20s | the patch passed | ||| _ Other Tests _ | | :broken_heart: | unit | 250m 26s | hbase-server in the patch failed. | | :green_heart: | asflicense | 0m 27s | The patch does not generate ASF License warnings. | | | | 317m 19s | | | Subsystem | Report/Notes | |----------:|:-------------| | Docker | Client=19.03.4 Server=19.03.4 base: https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/4/artifact/out/Dockerfile | | GITHUB PR | https://github.com/apache/hbase/pull/775 | | Optional Tests | dupname asflicense javac javadoc unit spotbugs findbugs shadedjars hadoopcheck hbaseanti checkstyle compile | | uname | Linux cd93459eb2c6 4.15.0-66-generic #75-Ubuntu SMP Tue Oct 1 05:24:09 UTC 2019 x86_64 GNU/Linux | | Build tool | maven | | Personality | /home/jenkins/jenkins-slave/workspace/HBase-PreCommit-GitHub-PR_PR-775/out/precommit/personality/provided.sh | | git revision | master / b8a4504a26 | | Default Java | 1.8.0_181 | | javac | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/4/artifact/out/diff-compile-javac-hbase-server.txt | | checkstyle | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/4/artifact/out/diff-checkstyle-hbase-server.txt | | unit | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/4/artifact/out/patch-unit-hbase-server.txt | | Test Results | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/4/testReport/ | | Max. process+thread count | 5129 (vs. ulimit of 10000) | | modules | C: hbase-server U: hbase-server | | Console output | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/4/console | | versions | git=2.11.0 maven=2018-06-17T18:33:14Z) findbugs=3.1.11 | | Powered by | Apache Yetus 0.11.0 https://yetus.apache.org | This message was automatically generated.
7. Test suite passes for me when I run locally, flakies excluded, and `-fae -Dsurefire.rerunFailingTestsCount=2`. Feeling comfortable to commit.
8. :broken_heart: **-1 overall** | Vote | Subsystem | Runtime | Comment | |:----:|----------:|--------:|:--------| | :blue_heart: | reexec | 0m 46s | Docker mode activated. | ||| _ Prechecks _ | | :green_heart: | dupname | 0m 0s | No case conflicting files found. | | :green_heart: | hbaseanti | 0m 0s | Patch does not have any anti-patterns. | | :green_heart: | @author | 0m 0s | The patch does not contain any @author tags. | | :green_heart: | test4tests | 0m 1s | The patch appears to include 14 new or modified test files. | ||| _ master Compile Tests _ | | :green_heart: | mvninstall | 5m 50s | master passed | | :green_heart: | compile | 1m 0s | master passed | | :green_heart: | checkstyle | 1m 30s | master passed | | :green_heart: | shadedjars | 4m 46s | branch has no errors when building our shaded downstream artifacts. | | :green_heart: | javadoc | 0m 39s | master passed | | :blue_heart: | spotbugs | 4m 34s | Used deprecated FindBugs config; considering switching to SpotBugs. | | :green_heart: | findbugs | 4m 31s | master passed | ||| _ Patch Compile Tests _ | | :green_heart: | mvninstall | 5m 12s | the patch passed | | :green_heart: | compile | 0m 55s | the patch passed | | :broken_heart: | javac | 0m 55s | hbase-server generated 1 new + 3 unchanged - 3 fixed = 4 total (was 6) | | :broken_heart: | checkstyle | 1m 23s | hbase-server: The patch generated 4 new + 266 unchanged - 41 fixed = 270 total (was 307) | | :green_heart: | whitespace | 0m 0s | The patch has no whitespace issues. | | :green_heart: | shadedjars | 4m 33s | patch has no errors when building our shaded downstream artifacts. | | :green_heart: | hadoopcheck | 16m 30s | Patch does not cause any errors with Hadoop 2.8.5 2.9.2 or 3.1.2. | | :green_heart: | javadoc | 0m 36s | the patch passed | | :green_heart: | findbugs | 4m 18s | the patch passed | ||| _ Other Tests _ | | :green_heart: | unit | 166m 49s | hbase-server in the patch passed. | | :green_heart: | asflicense | 0m 36s | The patch does not generate ASF License warnings. | | | | 226m 38s | | | Subsystem | Report/Notes | |----------:|:-------------| | Docker | Client=19.03.4 Server=19.03.4 base: https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/5/artifact/out/Dockerfile | | GITHUB PR | https://github.com/apache/hbase/pull/775 | | Optional Tests | dupname asflicense javac javadoc unit spotbugs findbugs shadedjars hadoopcheck hbaseanti checkstyle compile | | uname | Linux 6f264290adb3 4.15.0-58-generic #64-Ubuntu SMP Tue Aug 6 11:12:41 UTC 2019 x86_64 GNU/Linux | | Build tool | maven | | Personality | /home/jenkins/jenkins-slave/workspace/HBase-PreCommit-GitHub-PR_PR-775/out/precommit/personality/provided.sh | | git revision | master / f58bd4a7ac | | Default Java | 1.8.0_181 | | javac | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/5/artifact/out/diff-compile-javac-hbase-server.txt | | checkstyle | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/5/artifact/out/diff-checkstyle-hbase-server.txt | | Test Results | https://builds.apache.org/job/HBase-

PreCommit-GitHub-PR/job/PR-775/5/testReport/ | | Max. process+thread count | 4243 (vs. ulimit of 10000) | | modules | C: hbase-server U: hbase-server | | Console output | https://builds.apache.org/job/HBase-PreCommit-GitHub-PR/job/PR-775/5/console | | versions | git=2.11.0 maven=2018-06-17T18:33:14Z) findbugs=3.1.11 | | Powered by | Apache Yetus 0.11.0 https://yetus.apache.org | This message was automatically generated.

github_pulls_reviews:

1. Should do parametrized logging if you are making changes... i.e. "hbase.rootdir={}..."
2. getDataRootDir is diff from getRootDir?
3. What version we targetting this patch against? HRS is LimitedPrivate for Tools. Wonder if any tools using these data members. For 3.0.0 or 2.3.0?
4. You were able to remove this? Great.
5. Ok.
6. These dead data members? If so, great.
7. You were able to make-do w/ locals for these? They didn't have to be data members? If so, great.
8. **body:** Oh boy. Dead feature never finished.
   label: requirement
9. Final on params is not the fashion any more. Compiler doesn't do anything w/ them IIRC
10. Great you figured this can be removed.
11. Smile
12. Better.
13. Move out of this class?
14. Will this change the possible delays? Wonder if nextInt was intentionally chosen?
15. Odd on different line....
16. If not subclassed, yeah, kill it. Was probably added for Master.
17. Good
18. **body:** Not used? Moved?
    label: code-design
19. Good
20. Good
21. This is part of > Rename `fs` to `dataFs`, `rootDir` as `dataRootDir` so as to distinguish from the new `walFs`, `walRootDir` (and make it easier to spot bugs). I think I see some bugs...
22. I'm thinking this is for master and branch-2. Any further and these kind of refactor/cleanup patches end up not being "back ports" but pre-branch rewrites. I could be persuaded, but you'll have to be pretty decorated in Debate Club.
23. I suppose I should go read the fine-print re: `IA.LimitedPrivate(Tools)`...
24. it is also ok if not change?
25. **body:** Looks like this is use a different code formatter, should we use unified?
    label: code-design
26. **body:** `final` on params makes me feel better at night knowing we're not trying to use "out" parameters. Overall I find that pushing toward a more functional style leads to more reliable and maintainable code, and making as many things final as possible makes for good general hygiene.
    label: code-design
27. Yeah perhaps in a subsequent patch. Trying to keep this scope narrow on "OOP 101" stuff.
28. **body:** I'm using the code formatter in `dev-support/hbase_eclipse_formatter.xml` and checkstyle doesn't complain about it. The newlines between method arguments make for simpler reading and shorter diffs in the long run. My rule of thumb is that if the parameter list need to wrap, they should probably be broken out into one line each. What formatter do you use? It would be great if we didn't have to talk about this at all, if there was a tool like rust or go that just does it.
    label: code-design
29. **body:** @binlijin mentioned the same about my newline habits. You don't like the `) {` on their own line? I like this style because (1) it unclutters the argument list, when they would need to line wrap, to have each on their own line; (2) moving down the `) {` makes room for a `throws` clause with its list of exceptions; and (3) keeps indentation levels clear re: what is argument list and what is method body. Example with throws arguments: ```java private ReportRegionStateTransitionRequest createReportRegionStateTransitionRequest(final RegionStateTransitionContext context) throws ThisException, ThatException { // method body } ``` I don't want to rock the boat too badly for you lads, let me undo these radical, divergent styles ;)
    label: code-design
30. **body:** @saintstack @binlijin Here's an example of this style with a throws clause.
    label: code-design
31. No I don't believe it will. I think `int` was used down here because the value is read out as an `int` using `conf.getInt()` -- which it still uses. The value will remain as whatever was read from the config. I'm simply converting the value to a `long` slightly earlier.

32. **body:** This field is flagged as unused by static analysis. It's possible that something in the scope of `HIA.TOOLS` is reaching in for it, but if the region server itself never uses it, then it's a variable that could be local to the tool that's using it.
    label: code-design
33. Yep, a value is set in the constructor and then never read.
34. **body:** Another field assigned but never used. `RpcControllerFactory` instances do existing the codebase, they're just not pulled from `HRegionServer` any more.
    label: code-design
35. **body:** Yep, never used.
    label: code-design
36. **body:** Static analysis says it's not used. deleted.
    label: code-design
37. True. Breaking up these compound statements makes it easier to see NPEs when they happen. I guess such stylistic choices will become obsolete once Java14 lands. https://bugs.openjdk.java.net/browse/JDK-8218628
38. Great.
39. OK, i am also use the code formatter in dev-support/hbase_eclipse_formatter.xml with intellij idea, but it does not format the code into it.
40. OK
41. What did you figure reading fine-print?
42. Yeah, code looks unusual. See code around you.
43. I see you added it on the end.

jira_issues:

1. **summary:** Enforce member visibility in HRegionServer
   description: {{HRegionServer}} leaks member variables quite a lot. Lock down our interface -- member variables should be {{private}} (unless intentionally {{protected}} for subclasses) and use getters with proper accessibility constraints.
2. **summary:** Enforce member visibility in HRegionServer
   description: {{HRegionServer}} leaks member variables quite a lot. Lock down our interface -- member variables should be {{private}} (unless intentionally {{protected}} for subclasses) and use getters with proper accessibility constraints.
   label: code-design
3. **summary:** Enforce member visibility in HRegionServer
   description: {{HRegionServer}} leaks member variables quite a lot. Lock down our interface -- member variables should be {{private}} (unless intentionally {{protected}} for subclasses) and use getters with proper accessibility constraints.

jira_issues_comments:

1. Results for branch master [build #1530 on builds.a.o|https://builds.apache.org/job/HBase%20Nightly/job/master/1530/]: (x) *{color:red}-1 overall{color}* ---- details (if available): (x) {color:red}-1 general checks{color} -- For more information [see general report|https://builds.apache.org/job/HBase%20Nightly/job/master/1530//General_Nightly_Build_Report/] (x) {color:red}-1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report|https://builds.apache.org/job/HBase%20Nightly/job/master/1530//JDK8_Nightly_Build_Report_(Hadoop2)/] (x) {color:red}-1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report|https://builds.apache.org/job/HBase%20Nightly/job/master/1530//JDK8_Nightly_Build_Report_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (/) {color:green}+1 client integration test{color}
2. Merged to master, branch-2. Thanks for the reviews [~stack], [~binlijin].
3. Results for branch branch-2 [build #2349 on builds.a.o|https://builds.apache.org/job/HBase%20Nightly/job/branch-2/2349/]: (x) *{color:red}-1 overall{color}* ---- details (if available): (/) {color:green}+1 general checks{color} -- For more information [see general report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2/2349//General_Nightly_Build_Report/] (x) {color:red}-1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2/2349//JDK8_Nightly_Build_Report_(Hadoop2)/] (x) {color:red}-1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2/2349//JDK8_Nightly_Build_Report_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (/) {color:green}+1 client integration test{color}