

Item 145

git_comments:

1. * Default: 0.0.0.0
2. * Mandatory field
3. * * Class to construct instances of RPC server with specific options.
4. * Default: false
5. * Default: -1
6. * * Build the RPC Server. * @throws IOException on error * @throws HadoopIllegalArgumentException when mandatory fields are not set
7. * Default: null
8. * Default: 0
9. * Default: 1
10. Test mandatory field instance
11. Test mandatory field conf
12. Test mandatory field protocol

git_commits:

1. **summary:** HADOOP-8736. Merging change r1379652 from trunk to branch-2. Contributed by Brandon Li.
message: HADOOP-8736. Merging change r1379652 from trunk to branch-2. Contributed by Brandon Li. git-svn-id: <https://svn.apache.org/repos/asf/hadoop/common/branches/branch-2@1388575> 13f79535-47bb-0310-9956-ffa450edef68

github_issues:

github_issues_comments:

github_pulls:

github_pulls_comments:

github_pulls_reviews:

jira_issues:

1. **summary:** Add Builder for building an RPC server
description: There are quite a few variants of `getServer()` method to create an RPC server. Create a builder class to abstract the building steps and avoid more `getServer()` variants in the future.

jira_issues_comments:

1. -1 overall. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12542664/HADOOP-8736.patch> against trunk revision . -1 patch. The patch command could not apply the patch. Console output: <https://builds.apache.org/job/PreCommit-HADOOP-Build/1369//console> This message is automatically generated.
2. Uploaded a new patch which is rebased with the HEAD of trunk.
3. -1 overall. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12542693/HADOOP-8736.patch> against trunk revision . +1 @author. The patch does not contain any @author tags. +1 tests included. The patch appears to include 8 new or modified test files. -1 javac. The applied patch generated 2069 javac compiler warnings (more than the trunk's current 2059 warnings). +1 javadoc. The javadoc tool did not generate any warning messages. +1 eclipse:eclipse. The patch built with eclipse:eclipse. +1 findbugs. The patch does not introduce any new Findbugs (version 1.3.9) warnings. +1 release audit. The applied patch does not increase the total number of release audit warnings. +1 core tests. The patch passed unit tests in hadoop-common-project/hadoop-common. +1 contrib tests. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-HADOOP-Build/1370//testReport/> Javac warnings: <https://builds.apache.org/job/PreCommit-HADOOP->

Build/1370//artifact/trunk/patchprocess/diffJavacWarnings.txt Console output:

<https://builds.apache.org/job/PreCommit-HADOOP-Build/1370//console> This message is automatically generated.

4. **body:** The new javac warnings are due to the new deprecated `getServer()` method in the patch.
label: code-design
5. **body:** Thanks Brandon for doing this. This cleanup improves the readability of the code. I have not reviewed the patch completely. Here are my early comments: # Can you add javadoc to the newly added class `{{Builder}}` # Instead of deprecating `getServer()` methods, why not get rid of those methods altogether? You could also leave those methods for now and get rid of them in another jira. # Typo: method name "setProt" should be "setProto" # Some of the values while building protocol engine, such as protocol, instance, etc. cannot have default values. It would be good to either add check for that in Builder or ensure it is checked in `{{getProtocolEngine()}}` implementation.
label: documentation
6. **body:** bq. Typo: method name "setProt" should be "setProto" Agree it's a typo, but I think it should be "setPort". bq. Some of the values while building protocol engine, such as protocol, instance, etc. cannot have default values. It would be good to either add check for that in Builder or ensure it is checked in `getProtocolEngine()` implementation. Or perhaps those required fields should be set as part of the Builder constructor, while the other fields are set via the builder `set*()` methods?
label: documentation
7. **body:** bq. but I think it should be "setPort". Correct. bq. Or perhaps those required fields should be set as part of the Builder constructor, while the other fields are set via the builder `set*()` methods? Actually I prefer `set*` methods instead of Builder constructor. That way adding another constructor for another required field is not required. Thinking more about this, I think the way code is done is probably fine. For required fields, setting to null that results in `NullPointerException` to indicate programming error is fine. Not sure if there are any other fields which will not cause runtime exception where the default value cannot be valid.
label: code-design
8. **body:** In the patch, I tried to keep the Builder constructor simple(taking only Configuration as input). I think it can give the Builder more flexibility. For example, suppose the constructor takes 3 arguments, conf, protocol and instance. Suppose later on we realized more fields can not be null, we have to either modify the constructor or create a few new constructor, which make it like `getServer()` again. If we want to avoid `NullPointerException`, we could do a sanity check in the last step `build()` and throw an exception. The sanity check can be more comprehensive as more needs added later on.
label: code-design
9. I see your points, but I'd still prefer to have the truly required fields in the constructor. That way, if we later add another required field, developers' IDEs will pick this up or it will be seen at compile time, instead of waiting for a RTE like NPE or some specific exception we put in the `build()` method. For what are truly required fields, it seems strictly better to me to catch this at compile time instead of later. I don't feel super strongly about this. Feel free to roll with whatever you'd prefer.
10. {quote} That way, if we later add another required field, developers' IDEs will pick this up or it will be seen at compile time, instead of waiting for a RTE like NPE or some specific exception we put in the `build()` method.{quote} I agree that bigger constructor has certain merits. However, there are two problems. Firstly, the "truly required fields" are just the truly required for now, and it's hard to predict future. Secondly, even we have a constructor with all the current required fields, the developer can still pass null pointers by mistake. Doing param checking and throwing exceptions in last step doesn't seem too strange. Another example doing this is in google library <https://developers.google.com/protocol-buffers/docs/reference/java/com/google/protobuf/MessageLite.Builder#build%28%29>
11. Uploaded a new patch to address the above comments.
12. -1 overall. Here are the results of testing the latest attachment
<http://issues.apache.org/jira/secure/attachment/12542869/HADOOP-8736.patch> against trunk revision .
+1 @author. The patch does not contain any @author tags. +1 tests included. The patch appears to include 8 new or modified test files. +1 javac. The applied patch does not increase the total number of javac compiler warnings. +1 javadoc. The javadoc tool did not generate any warning messages. +1 eclipse:eclipse. The patch built with eclipse:eclipse. +1 findbugs. The patch does not introduce any new Findbugs (version 1.3.9) warnings. +1 release audit. The applied patch does not increase the total number of release audit warnings. -1 core tests. The patch failed these unit tests in hadoop-common-project/hadoop-common: org.apache.hadoop.ha.TestZKFailoverController +1 contrib tests. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-HADOOP->

Build/1376//testReport/ Console output: <https://builds.apache.org/job/PreCommit-HADOOP-Build/1376//console> This message is automatically generated.

13. The test failure is not introduced by this patch, and it passed in my local test.
14. bq. Firstly, the "truly required fields" are just the truly required for now, and it's hard to predict future. Sure, but if/when that future comes, what will have to happen with each of these approaches? In the constructor approach, you'll change the constructor signature and then things won't compile until you've fixed all of the call sites. That is good. In the multi-method builder approach, everything will compile, but you'll have to run all of the tests to find the call sites that you missed when adding a new builder method call, and also will have to hope that the tests in fact do cover all of the call sites. That is bad. bq. Secondly, even we have a constructor with all the current required fields, the developer can still pass null pointers by mistake. Of course that's always a possibility, but that seems less likely than a developer forgetting to call a builder method that is in fact required. That's also a possibility with the multi-method builder approach as well - the developer might pass null values by mistake. But like I said, you can go with whatever you prefer. You haven't convinced me that this is the right way to go, but I'm not going to stop you from doing it.
15. Aaron, I appreciate your review and comments. Discussion helps. :-)
16. **body:** bq. But like I said, you can go with whatever you prefer. You haven't convinced me that this is the right way to go, but I'm not going to stop you from doing it. I am fine with the current approach being taken. If there are issue due to this approach, because of frequent changes in Server, we could always revisit this. Here are comments for the patch (mostly nits): # build() method, please follow the coding guidelines and have {} after if. # Throwing HadoopIllegalArgumentException is fine. But if you are doing that for two of the parameters, I suggest doing the same for handlerCount and conf parameter as well. # In javadoc for build() method please add @throws and say if mandator fields are not set, the build method will throw HadoopIllegalArgumentException. # Please add a testcase where you create RPC server without mandatory fields and ensure exceptions are thrown. We could perhaps add this to TestIPC.
label: code-design
17. {quote}build() method, please follow the coding guidelines and have {} after if.{quote} done {quote}Throwing HadoopIllegalArgumentException is fine. But if you are doing that for two of the parameters, I suggest doing the same for handlerCount and conf parameter as well.{quote} Done for conf, handlerCount has a default as "1". {quote} In javadoc for build() method please add @throws and say if mandator fields are not set, the build method will throw HadoopIllegalArgumentException.{quote} Done. {quote} Please add a testcase where you create RPC server without mandatory fields and ensure exceptions are thrown. We could perhaps add this to TestIPC.{quote} Done in TestRPC.java.
18. +1 overall. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12543003/HADOOP-8736.patch> against trunk revision . +1 @author. The patch does not contain any @author tags. +1 tests included. The patch appears to include 8 new or modified test files. +1 javac. The applied patch does not increase the total number of javac compiler warnings. +1 javadoc. The javadoc tool did not generate any warning messages. +1 eclipse:eclipse. The patch built with eclipse:eclipse. +1 findbugs. The patch does not introduce any new Findbugs (version 1.3.9) warnings. +1 release audit. The applied patch does not increase the total number of release audit warnings. +1 core tests. The patch passed unit tests in hadoop-common-project/hadoop-common. +1 contrib tests. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-HADOOP-Build/1385//testReport/> Console output: <https://builds.apache.org/job/PreCommit-HADOOP-Build/1385//console> This message is automatically generated.
19. **body:** Minor comments: # Please add information which of the fields are mandatory # Javadoc for build() method should use java convention @throws and explain in that HadoopIllegalArgumentException is thrown when mandatory fields are not set.
label: documentation
20. Updated the patch. Thanks!
21. -1 overall. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12543312/HADOOP-8736.patch> against trunk revision . +1 @author. The patch does not contain any @author tags. +1 tests included. The patch appears to include 8 new or modified test files. +1 javac. The applied patch does not increase the total number of javac compiler warnings. +1 javadoc. The javadoc tool did not generate any warning messages. +1 eclipse:eclipse. The patch built with eclipse:eclipse. +1 findbugs. The patch does not introduce any new Findbugs (version 1.3.9) warnings. +1 release audit. The applied patch does not increase the total number of release audit warnings. -1 core tests. The patch failed these unit tests in hadoop-common-project/hadoop-common: org.apache.hadoop.ha.TestZKFailoverController +1 contrib tests. The patch

passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-HADOOP-Build/1392/testReport/> Console output: <https://builds.apache.org/job/PreCommit-HADOOP-Build/1392/console> This message is automatically generated.

22. The test failure is not introduced by this patch.
23. Integrated in Hadoop-Hdfs-trunk-Commit #2732 (See [<https://builds.apache.org/job/Hadoop-Hdfs-trunk-Commit/2732/>]) HADOOP-8736. Add Builder for building RPC server. Contributed by Brandon Li (Revision 1379652) Result = SUCCESS suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1379652> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ha/ZKFCRpcServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ipc/RPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/MiniRPCBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/RPCCallBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestMultipleProtocolServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestProtoBufRpc.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPCCompatibility.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestSaslRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/security/TestDoAsEffectiveUser.java
24. Integrated in Hadoop-Common-trunk-Commit #2669 (See [<https://builds.apache.org/job/Hadoop-Common-trunk-Commit/2669/>]) HADOOP-8736. Add Builder for building RPC server. Contributed by Brandon Li (Revision 1379652) Result = SUCCESS suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1379652> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ha/ZKFCRpcServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ipc/RPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/MiniRPCBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/RPCCallBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestMultipleProtocolServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestProtoBufRpc.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPCCompatibility.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestSaslRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/security/TestDoAsEffectiveUser.java
25. Integrated in Hadoop-Mapreduce-trunk-Commit #2694 (See [<https://builds.apache.org/job/Hadoop-Mapreduce-trunk-Commit/2694/>]) HADOOP-8736. Add Builder for building RPC server. Contributed by Brandon Li (Revision 1379652) Result = FAILURE suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1379652> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ha/ZKFCRpcServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ipc/RPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/MiniRPCBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/RPCCallBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestMultipleProtocolServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestProtoBufRpc.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPC.java *

- /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPCCompatibility.java *
- /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestSaslRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/security/TestDoAsEffectiveUser.java
26. Integrated in Hadoop-Hdfs-trunk #1152 (See [<https://builds.apache.org/job/Hadoop-Hdfs-trunk/1152/>]) HADOOP-8736. Add Builder for building RPC server. Contributed by Brandon Li (Revision 1379652) Result = SUCCESS suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1379652> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ha/ZKFCRpcServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ipc/RPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/MiniRPCBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/RPCCallBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestMultipleProtocolServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestProtoBufRpc.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPCCompatibility.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestSaslRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/security/TestDoAsEffectiveUser.java
27. Integrated in Hadoop-Mapreduce-trunk #1183 (See [<https://builds.apache.org/job/Hadoop-Mapreduce-trunk/1183/>]) HADOOP-8736. Add Builder for building RPC server. Contributed by Brandon Li (Revision 1379652) Result = SUCCESS suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1379652> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ha/ZKFCRpcServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/main/java/org/apache/hadoop/ipc/RPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/MiniRPCBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/RPCCallBenchmark.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestMultipleProtocolServer.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestProtoBufRpc.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestRPCCompatibility.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/ipc/TestSaslRPC.java * /hadoop/common/trunk/hadoop-common-project/hadoop-common/src/test/java/org/apache/hadoop/security/TestDoAsEffectiveUser.java
28. Consider merging to branch-2? Seems like a straight-forward change and will help with merge conflicts.
29. Merged this change to branch-2.
30. Integrated in Hadoop-Hdfs-trunk-Commit #2816 (See [<https://builds.apache.org/job/Hadoop-Hdfs-trunk-Commit/2816/>]) Moving HADOOP-8736 to release 2.0.3 section (Revision 1388578) Result = SUCCESS suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1388578> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt
31. Integrated in Hadoop-Common-trunk-Commit #2753 (See [<https://builds.apache.org/job/Hadoop-Common-trunk-Commit/2753/>]) Moving HADOOP-8736 to release 2.0.3 section (Revision 1388578) Result = SUCCESS suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1388578> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt
32. Integrated in Hadoop-Mapreduce-trunk-Commit #2775 (See [<https://builds.apache.org/job/Hadoop-Mapreduce-trunk-Commit/2775/>]) Moving HADOOP-8736 to release 2.0.3 section (Revision 1388578) Result = FAILURE suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1388578>

SVN&view=rev&rev=1388578 Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt

33. Integrated in Hadoop-Hdfs-trunk #1173 (See [<https://builds.apache.org/job/Hadoop-Hdfs-trunk/1173/>]) Moving HADOOP-8736 to release 2.0.3 section (Revision 1388578) Result = SUCCESS suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1388578> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt
34. Integrated in Hadoop-Mapreduce-trunk #1204 (See [<https://builds.apache.org/job/Hadoop-Mapreduce-trunk/1204/>]) Moving HADOOP-8736 to release 2.0.3 section (Revision 1388578) Result = FAILURE suresh : <http://svn.apache.org/viewcvcs.cgi/?root=Apache-SVN&view=rev&rev=1388578> Files : * /hadoop/common/trunk/hadoop-common-project/hadoop-common/CHANGES.txt