Item 148
git_comments:

git_commits:

1. **summary:** HBASE-2018 Updates to .META. blocked under high MemStore load
   message: HBASE-2018 Updates to .META. blocked under high MemStore load git-svn-id: https://svn.apache.org/repos/asf/hadoop/hbase/trunk@887054 13f79535-47bb-0310-9956-ffa450edef68

github_issues:

github_issues_comments:

github_pulls:

github_pulls_comments:

github_pulls_reviews:

jira_issues:

1. **summary:** Updates to .META. blocked under high MemStore load
   description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.
2. **summary:** Updates to .META. blocked under high MemStore load
   description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.
3. **summary:** Updates to .META. blocked under high MemStore load

description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.

4. **summary:** Updates to .META. blocked under high MemStore load
   description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.
5. **summary:** Updates to .META. blocked under high MemStore load
   description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.

6. **summary:** Updates to .META. blocked under high MemStore load
   description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.
7. **summary:** Updates to .META. blocked under high MemStore load
   description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.
   label: code-design
8. **summary:** Updates to .META. blocked under high MemStore load
   description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently

1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.

9. **summary:** Updates to .META. blocked under high MemStore load
   description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.
10. **summary:** Updates to .META. blocked under high MemStore load
    description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false 2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.
    label: code-design
11. **summary:** Updates to .META. blocked under high MemStore load
    description: I discovered this on Lars' cluster. The symptom was the good old: {code} 09/11/30 08:10:26 INFO mapred.JobClient: Task Id : attempt_200911250121_0011_r_000010_1, Status : FAILED org.apache.hadoop.hbase.client.RetriesExhaustedException: Trying to contact region server Some server, retryOnlyOne=true, index=0, islastrow=false, tries=9, numtries=10, i=14, listsize=20, region=prev-docs,de68fb97795ef3d936a3f10ff8790253,1259573366564 for region prev-docs,ccea967e66ccb53d83c48849c3a23f21,1259542138868, row 'ccff8cd4ca871c41f4fa7d44cffed962', but failed after 10 attempts. Exceptions: at org.apache.hadoop.hbase.client.HConnectionManager$TableServers$Batch.process(HConnectionManager.java:1120) at org.apache.hadoop.hbase.client.HConnectionManager$TableServers.processBatchOfRows(HConnectionManager.java:1201) at org.apache.hadoop.hbase.client.HTable.flushCommits(HTable.java:605) at org.apache.hadoop.hbase.client.HTable.put(HTable.java:470) at org.apache.hadoop.hbase.mapreduce.TableOutputFormat$TableRecordW {code} But the load wasn't that heavy, just lots of splitting going on. Looking at the logs, I see a split taking more than 4 minutes which is explained by this happening on the RS hosting .META. : {code} 2009-11-30 08:08:39,922 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,2c9d51e57b20decd5c6419d23ede822b,1259542273901 because global memstore limit of 1.6g exceeded; currently 1.6g and flushing till 1021.9m ... 2009-11-30 08:12:33,743 DEBUG org.apache.hadoop.hbase.regionserver.HRegion: Finished memstore flush of ~22.9m for region prev-docs,c8fea4fbbc41e746d960854ed4d41dd6,1259587143838 in 14160ms, sequence id=13677, compaction requested=false

2009-11-30 08:12:33,744 INFO org.apache.hadoop.hbase.regionserver.MemStoreFlusher: Forced flushing of prev-docs,39c2995d955c041d21f4dc4a0d0dbf6c,1259587061295 because global memstore limit of 1.6g exceeded; currently 1.0g and flushing till 1021.9m {code} So we should not block updates to .META. for any reason. I'm pretty sure this issue explains other issues we've seen on the mailing list.

jira_issues_comments:

1. This patch adds a check before calling cacheFlusher.reclaimMemStoreMemory so that we don't go waiting on the synchronized method if it's a .META. update. I'm currently running the tests.
2. +1 Agree with the issue priority also.
3. All the tests pass. I would love if Lars could try out my patch before committing.
4. Testing now, takes a few hours to ramp up through the map phase. Results forthcoming...
5. So testing for me is not going too well. Overall the patch is not harming anything, I say +1. For my testing I am struggling apparently with a too small cluster :(
6. Should this check be for all operations on meta regions, deletes also? See patch -v2.
7. **body:** New patch makes sense. Also Lars just reported that his job finally was successful (by tweaking other stuff). At least I think this patch covers a very bad corner case.
   label: code-design
8. +1 on v2 of patch.
9. Committed -v2 patch to trunk and 0.20 branch.