MongoDB Routing Info Storage & Refresh Optimization

Author: Tencent MongoDB team

Preface

In recent releases of MongoDB, there's a couple optimization been made around Refreshing Routing Info, however the performance issue caused by it wasn't rooted out for good that big sharded clusters would still suffer slow queries due to it.

Tencent MongoDB team have (or hope so) come up with an optimization solution to solve the problem by utilizing Two-Dimensional Sorting & Search. With the optimization, there'd be no latency caused by refreshing routing info, that the refreshing time cost would remain at around 2ms regardless of the data size of the shreded cluster.

1. Background

Tencent MongoDB team has noticed unusual slow queries on many sharded clusters out of blue while there's no sign of any system resource shortage (CPU, RAM, I/O, etc) in the past several years. Further looking into this symptom, the team figured out retrieving incremental routing info would take a lot of time when total chunk number exceeds certain threshold on shared clusters.

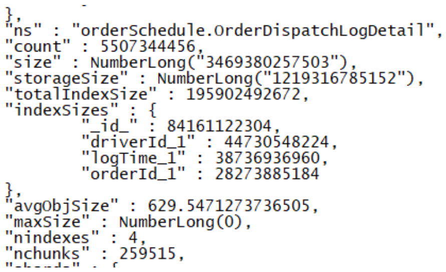
For instance, a sharded cluster with 250k chunks requires around 300ms to refresh routing info; For larger clusters, like a cluster with 1 million chunks, it'd take seconds to do the refresh, severely blocking other queries.

Other than that, refreshing routing info could consume significantly more system resources – If a cluster has multiple sharded collections, CPU spikes will occur much more often when refreshing routing info for multiple collections at the same time.

Below are some of the cases we found in production & testing environment.

Case 1: v4.0 Product Cluster with 250k chunks

- **Cluster Info**



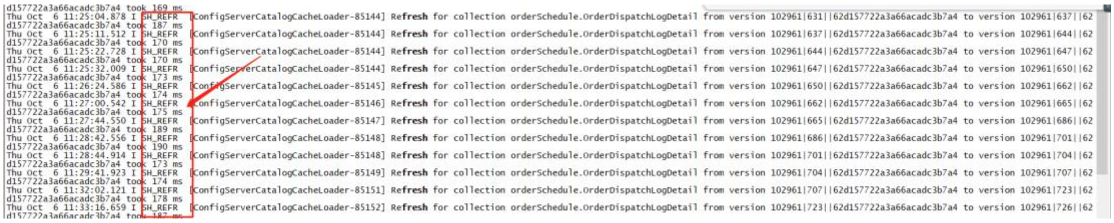
```
},
"ns" : "orderSchedule.OrderDispatchLogDetail",
"count" : 5507344456,
"size" : NumberLong("3469380257503"),
"storageSize" : NumberLong("1219316785152"),
"totalIndexSize" : 195902492672,
"indexSizes" : {
        "_id_" : 84161122304,
        "driverId_1" : 44730548224,
        "logTime_1" : 38736936960,
        "orderId_1" : 28273885184
},
"avgObjSize" : 629.5471273736505,
"maxSize" : NumberLong(0),
"nindexes" : 4,
"nchunks" : 259515,
```

Data size: 5.5 billion docs, 1.2TB in total size;

Chunk number: 250k;

Refreshing routing info duration: 200ms for mongos, 300ms for mongod.

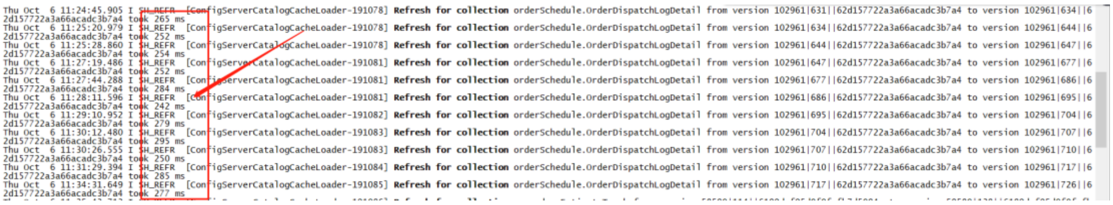
- **Related mongos logs**



```
d157722a3a66acadc3b7a4 took 169 ms
Thu Oct  6 11:25:04.878 I SH_REFR  [ConfigServerCatalogCacheLoader-85144] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|631||62d157722a3a66acadc3b7a4 to version 102961|637||62
d157722a3a66acadc3b7a4 took 187 ms
Thu Oct  6 11:25:11.512 I SH_REFR  [ConfigServerCatalogCacheLoader-85144] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|637||62d157722a3a66acadc3b7a4 to version 102961|644||62
d157722a3a66acadc3b7a4 took 170 ms
Thu Oct  6 11:25:22.728 I SH_REFR  [ConfigServerCatalogCacheLoader-85144] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|644||62d157722a3a66acadc3b7a4 to version 102961|647||62
d157722a3a66acadc3b7a4 took 170 ms
Thu Oct  6 11:25:32.009 I SH_REFR  [ConfigServerCatalogCacheLoader-85144] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|647||62d157722a3a66acadc3b7a4 to version 102961|650||62
d157722a3a66acadc3b7a4 took 173 ms
Thu Oct  6 11:26:24.586 I SH_REFR  [ConfigServerCatalogCacheLoader-85145] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|650||62d157722a3a66acadc3b7a4 to version 102961|662||62
d157722a3a66acadc3b7a4 took 174 ms
Thu Oct  6 11:27:00.542 I SH_REFR  [ConfigServerCatalogCacheLoader-85146] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|662||62d157722a3a66acadc3b7a4 to version 102961|665||62
d157722a3a66acadc3b7a4 took 175 ms
Thu Oct  6 11:27:44.550 I SH_REFR  [ConfigServerCatalogCacheLoader-85147] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|665||62d157722a3a66acadc3b7a4 to version 102961|686||62
d157722a3a66acadc3b7a4 took 189 ms
Thu Oct  6 11:28:42.556 I SH_REFR  [ConfigServerCatalogCacheLoader-85148] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|686||62d157722a3a66acadc3b7a4 to version 102961|701||62
d157722a3a66acadc3b7a4 took 190 ms
Thu Oct  6 11:28:44.914 I SH_REFR  [ConfigServerCatalogCacheLoader-85148] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|701||62d157722a3a66acadc3b7a4 to version 102961|704||62
d157722a3a66acadc3b7a4 took 173 ms
Thu Oct  6 11:29:41.923 I SH_REFR  [ConfigServerCatalogCacheLoader-85149] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|704||62d157722a3a66acadc3b7a4 to version 102961|707||62
d157722a3a66acadc3b7a4 took 174 ms
Thu Oct  6 11:32:02.121 I SH_REFR  [ConfigServerCatalogCacheLoader-85151] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|707||62d157722a3a66acadc3b7a4 to version 102961|723||62
d157722a3a66acadc3b7a4 took 178 ms
Thu Oct  6 11:33:16.659 I SH_REFR  [ConfigServerCatalogCacheLoader-85152] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|723||62d157722a3a66acadc3b7a4 to version 102961|726||62
d157722a3a66acadc3b7a4 took 187 ms
```

- Thu Oct 6 11: 28: 42.556 I SH_REFR [ConfigServerCatalogCacheLoader-85148] Refresh **for** collection orderSchedule.OrderDispatchLogDetail from version 102961|686||62d157722a3a66acadc3b7a4 to version 102961|701||62d157722a3a66acadc3b7a4 took 190 ms
- Thu Oct 6 11: 28: 44.914 I SH_REFR [ConfigServerCatalogCacheLoader-85148] Refresh **for** collection orderSchedule.OrderDispatchLogDetail from version 102961|701||62d157722a3a66acadc3b7a4 to version 102961|704||62d157722a3a66acadc3b7a4 took 183 ms
- Thu Oct 6 11: 29: 41.923 I SH_REFR [ConfigServerCatalogCacheLoader-85149] Refresh **for** collection orderSchedule.OrderDispatchLogDetail from version 102961|704||62d157722a3a66acadc3b7a4 to version 102961|707||62d157722a3a66acadc3b7a4 took 194 ms
- Thu Oct 6 11: 32: 02.121 I SH_REFR [ConfigServerCatalogCacheLoader-85151] Refresh **for** collection orderSchedule.OrderDispatchLogDetail from version 102961|707||62d157722a3a66acadc3b7a4 to version 102961|723||62d157722a3a66acadc3b7a4 took 198 ms

- **Related mongod logs**



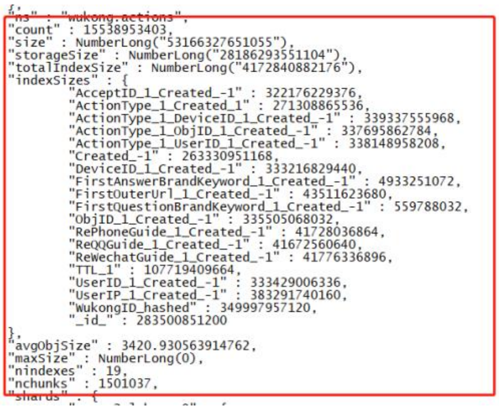
```
Thu Oct  6 11:24:45.905 I SH_REFR  [ConfigServerCatalogCacheLoader-191078] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|631||62d157722a3a66acadc3b7a4 to version 102961|634||6
2d157722a3a66acadc3b7a4 took 265 ms
Thu Oct  6 11:25:20.979 I SH_REFR  [ConfigServerCatalogCacheLoader-191078] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|634||62d157722a3a66acadc3b7a4 to version 102961|644||6
2d157722a3a66acadc3b7a4 took 252 ms
Thu Oct  6 11:25:28.860 I SH_REFR  [ConfigServerCatalogCacheLoader-191078] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|644||62d157722a3a66acadc3b7a4 to version 102961|647||6
2d157722a3a66acadc3b7a4 took 254 ms
Thu Oct  6 11:27:19.486 I SH_REFR  [ConfigServerCatalogCacheLoader-191081] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|647||62d157722a3a66acadc3b7a4 to version 102961|677||6
2d157722a3a66acadc3b7a4 took 252 ms
Thu Oct  6 11:27:44.288 I SH_REFR  [ConfigServerCatalogCacheLoader-191081] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|677||62d157722a3a66acadc3b7a4 to version 102961|686||6
2d157722a3a66acadc3b7a4 took 284 ms
Thu Oct  6 11:28:11.596 I SH_REFR  [ConfigServerCatalogCacheLoader-191081] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|686||62d157722a3a66acadc3b7a4 to version 102961|695||6
2d157722a3a66acadc3b7a4 took 242 ms
Thu Oct  6 11:29:10.952 I SH_REFR  [ConfigServerCatalogCacheLoader-191082] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|695||62d157722a3a66acadc3b7a4 to version 102961|704||6
2d157722a3a66acadc3b7a4 took 279 ms
Thu Oct  6 11:30:12.480 I SH_REFR  [ConfigServerCatalogCacheLoader-191083] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|704||62d157722a3a66acadc3b7a4 to version 102961|707||6
2d157722a3a66acadc3b7a4 took 295 ms
Thu Oct  6 11:30:26.555 I SH_REFR  [ConfigServerCatalogCacheLoader-191083] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|707||62d157722a3a66acadc3b7a4 to version 102961|710||6
2d157722a3a66acadc3b7a4 took 250 ms
Thu Oct  6 11:31:29.394 I SH_REFR  [ConfigServerCatalogCacheLoader-191084] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|710||62d157722a3a66acadc3b7a4 to version 102961|717||6
2d157722a3a66acadc3b7a4 took 285 ms
Thu Oct  6 11:34:31.649 I SH_REFR  [ConfigServerCatalogCacheLoader-191085] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|717||62d157722a3a66acadc3b7a4 to version 102961|726||6
2d157722a3a66acadc3b7a4 took 277 ms
```

- Thu Oct 6 11: 24: 11.358 I SH_REFR [ConfigServerCatalogCacheLoader-191078] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|603||62d157722a3a66acadc3b7a4 to version 102961|628||62d157722a3a66acadc3b7a4 took 262 ms

- Thu Oct 6 11: 24: 21.306 I SH_REFR [ConfigServerCatalogCacheLoader-191078] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|628||62d157722a3a66acadc3b7a4 to version 102961|631||62d157722a3a66acadc3b7a4 took 285 ms
- Thu Oct 6 11: 24: 45.905 I SH_REFR [ConfigServerCatalogCacheLoader-191078] Refresh for collection orderSchedule.OrderDispatchLogDetail from version 102961|631||62d157722a3a66acadc3b7a4 to version 102961|634||62d157722a3a66acadc3b7a4 took 265 ms
- Thu Oct 6 11: 25: 20.979 I SH_REFR [ConfigServerCatalogCacheLoader-191078] Refresh for collection **orderSchedule.OrderDispatchLogDetail** from version 102961|634||62d157722a3a66acadc3b7a4 to version 102961|644||62d157722a3a66acadc3b7a4 took 252 ms

Case 2: v4.2 Product Cluster with 1.5m chunks

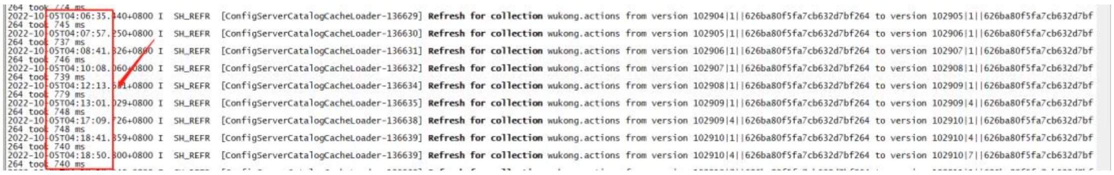
Cluster Info



```
{,
"ns" : "wukong.actions",
"count" : 15538953403,
"size" : NumberLong("53166327651055"),
"storageSize" : NumberLong("28186293551104"),
"totalIndexSize" : NumberLong("4172840882176"),
"indexSizes" : {
        "AcceptID_1_Created_-1" : 322176229376,
        "ActionType_1_Created_1" : 271308865536,
        "ActionType_1_DeviceID_1_Created_-1" : 339337555968,
        "ActionType_1_ObjID_1_Created_-1" : 337695862784,
        "ActionType_1_UserID_1_Created_-1" : 338148958208,
        "Created_-1" : 263330951168,
        "DeviceID_1_Created_-1" : 333216829440,
        "FirstAnswerBrandKeyword_1_Created_-1" : 4933251072,
        "FirstOuterUrl_1_Created_-1" : 43511623680,
        "FirstQuestionBrandKeyword_1_Created_-1" : 559788032,
        "ObjID_1_Created_-1" : 335505068032,
        "RePhoneGuide_1_Created_-1" : 41728036864,
        "ReQQGuide_1_Created_-1" : 41672560640,
        "ReWechatGuide_1_Created_-1" : 41776336896,
        "TTL_1" : 107719409664,
        "UserID_1_Created_-1" : 333429006336,
        "UserIP_1_Created_-1" : 383291740160,
        "WukongID_hashed" : 349997957120,
        "_id_" : 283500851200
},
"avgObjSize" : 3420.930563914762,
"maxSize" : NumberLong(0),
"nindexes" : 19,
"nchunks" : 1501037,
"shards" : {
```

Data size: 15.5 billion docs， 22.5TB in total size;
Chunk number: 1.5 million;
Refreshing routing info duration: 800ms for mongos, 1.2s for mongod.

Related mongos logs

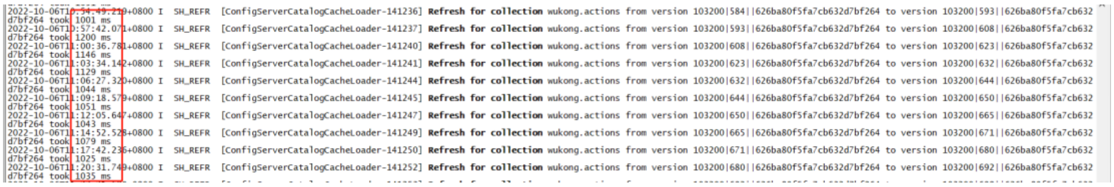


```
264 took 774 ms
2022-10-05T04:06:35.440+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136629] Refresh for collection wukong.actions from version 102904|1||626ba80f5fa7cb632d7bf264 to version 102905|1||626ba80f5fa7cb632d7bf
264 took 745 ms
2022-10-05T04:07:57.250+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136630] Refresh for collection wukong.actions from version 102905|1||626ba80f5fa7cb632d7bf264 to version 102906|1||626ba80f5fa7cb632d7bf
264 took 737 ms
2022-10-05T04:08:41.326+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136631] Refresh for collection wukong.actions from version 102906|1||626ba80f5fa7cb632d7bf264 to version 102907|1||626ba80f5fa7cb632d7bf
264 took 746 ms
2022-10-05T04:10:08.060+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136632] Refresh for collection wukong.actions from version 102907|1||626ba80f5fa7cb632d7bf264 to version 102908|1||626ba80f5fa7cb632d7bf
264 took 739 ms
2022-10-05T04:12:13.581+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136634] Refresh for collection wukong.actions from version 102908|1||626ba80f5fa7cb632d7bf264 to version 102909|1||626ba80f5fa7cb632d7bf
264 took 779 ms
2022-10-05T04:13:01.029+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136635] Refresh for collection wukong.actions from version 102909|1||626ba80f5fa7cb632d7bf264 to version 102909|4||626ba80f5fa7cb632d7bf
264 took 748 ms
2022-10-05T04:17:09.726+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136638] Refresh for collection wukong.actions from version 102909|4||626ba80f5fa7cb632d7bf264 to version 102910|1||626ba80f5fa7cb632d7bf
264 took 748 ms
2022-10-05T04:18:41.359+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136639] Refresh for collection wukong.actions from version 102910|1||626ba80f5fa7cb632d7bf264 to version 102910|4||626ba80f5fa7cb632d7bf
264 took 740 ms
2022-10-05T04:18:50.800+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-136639] Refresh for collection wukong.actions from version 102910|4||626ba80f5fa7cb632d7bf264 to version 102910|7||626ba80f5fa7cb632d7bf
264 took 740 ms
```

- 2022-10-05T04: 18: 41.359+0800 I SH_REFR [ConfigServerCatalogCacheLoader-136639] Refresh for collection wukong.actions from version 102910|1||626ba80f5fa7cb632d7bf264 to version 102910|4||626ba80f5fa7cb632d7bf264 took 788 ms
- 2022-10-05T04: 18: 50.800+0800 I SH_REFR [ConfigServerCatalogCacheLoader-136639] Refresh for collection wukong.actions from version 102910|4||626ba80f5fa7cb632d7bf264 to version 102910|7||626ba80f5fa7cb632d7bf264 took 780 ms
- 2022-10-05T04: 19: 18.546+0800 I SH_REFR [ConfigServerCatalogCacheLoader-136639] Refresh for collection wukong.actions from version 102910|7||626ba80f5fa7cb632d7bf264 to version 102911|1||626ba80f5fa7cb632d7bf264 took 778 ms

```
● 2022-10-05T04: 20: 01.105+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-
  136640] Refresh for collection wukong.actions from version 102911|1||626ba80f5fa7cb632d7bf264 to version 102912|1||626ba80f5fa7cb632d7bf264 took 781 ms
```

- **Related mongod logs**



```
2022-10-06T10:54:49.219+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141236] Refresh for collection wukong.actions from version 103200|584||626ba80f5fa7cb632d7bf264 to version 103200|593||626ba80f5fa7cb632
d7bf264 took 1001 ms
2022-10-06T10:57:42.071+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141237] Refresh for collection wukong.actions from version 103200|593||626ba80f5fa7cb632d7bf264 to version 103200|608||626ba80f5fa7cb632
d7bf264 took 1200 ms
2022-10-06T11:00:36.781+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141240] Refresh for collection wukong.actions from version 103200|608||626ba80f5fa7cb632d7bf264 to version 103200|623||626ba80f5fa7cb632
d7bf264 took 1146 ms
2022-10-06T11:03:34.142+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141241] Refresh for collection wukong.actions from version 103200|623||626ba80f5fa7cb632d7bf264 to version 103200|632||626ba80f5fa7cb632
d7bf264 took 1129 ms
2022-10-06T11:06:27.320+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141244] Refresh for collection wukong.actions from version 103200|632||626ba80f5fa7cb632d7bf264 to version 103200|644||626ba80f5fa7cb632
d7bf264 took 1044 ms
2022-10-06T11:09:18.579+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141245] Refresh for collection wukong.actions from version 103200|644||626ba80f5fa7cb632d7bf264 to version 103200|650||626ba80f5fa7cb632
d7bf264 took 1051 ms
2022-10-06T11:12:05.647+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141247] Refresh for collection wukong.actions from version 103200|650||626ba80f5fa7cb632d7bf264 to version 103200|665||626ba80f5fa7cb632
d7bf264 took 1043 ms
2022-10-06T11:14:52.528+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141249] Refresh for collection wukong.actions from version 103200|665||626ba80f5fa7cb632d7bf264 to version 103200|671||626ba80f5fa7cb632
d7bf264 took 1079 ms
2022-10-06T11:17:42.236+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141250] Refresh for collection wukong.actions from version 103200|671||626ba80f5fa7cb632d7bf264 to version 103200|680||626ba80f5fa7cb632
d7bf264 took 1025 ms
2022-10-06T11:20:31.749+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-141252] Refresh for collection wukong.actions from version 103200|680||626ba80f5fa7cb632d7bf264 to version 103200|692||626ba80f5fa7cb632
d7bf264 took 1035 ms
```

```
● 2022-10-06T10: 54: 49.219+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-
  141236] Refresh for collection wukong.actions from version 103200|584||626ba80f5fa7cb632d7bf264 to version 103200|593||626ba80f5fa7cb632d7bf264 took 10
  01 ms
● 2022-10-06T10: 57: 42.071+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-
  141237] Refresh for collection wukong.actions from version 103200|593||626ba80f5fa7cb632d7bf264 to version 103200|608||626ba80f5fa7cb632d7bf264 took 12
  00 ms
● 2022-10-06T11: 00: 36.781+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-
  141240] Refresh for collection wukong.actions from version 103200|608||626ba80f5fa7cb632d7bf264 to version 103200|623||626ba80f5fa7cb632d7bf264 took 11
  46 ms
● 2022-10-06T11: 03: 34.142+0800 I  SH_REFR  [ConfigServerCatalogCacheLoader-
  141241] Refresh for collection wukong.actions from version 103200|623||626ba80f5fa7cb632d7bf264 to version 103200|632||626ba80f5fa7cb632d7bf264 took 11
  29 ms
```

Case 3: v3.6 Product Cluster with 4.3m chunks

- **Cluster Info**

Data size: 120 billion docs, 80TB data size in total;
Chunk number: 430 million;
Refreshing routing info duration: 4s for mongos, 4.6s for mongod.

Case 4: v5.0 Test Cluster with 2m chunks

- **Cluster Info**

Set up a v5.0 sharded cluster with 2 million chunks -- Use “id” field as shard key, ranging from 0 to 100,000,000, generate 2M chunks by pre-splitting chunks.

- **Related mongos logs**

> //Refreshing routing info

```
{"t": {"$date": "2022-10-06T17: 46: 25.479+08: 00"},"s": "I",  "c": "SH_REFR",  "id": 4619901, "ctx": "CatalogCache-2","msg": "Refreshed cached collection","attr": {"namespace": "test.test2","lookupSinceVersion": {"0": {"$timestamp": {"t": 49182,"i": 1}},"1": {"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i": 6}}},"newVersion": {"chunkVersion": {"0": {"$timestamp": {"t": 49182,"i": 1}},"1": {"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i": 6}}},"forcedRefreshSequenceNum": 21,"epochDisambiguatingSequenceNum": 18},"timeInStore": {"chunkVersion": : {"0": {"$timestamp": {"t": 49182,"i": 1}},"1": {"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i": 6}},"forcedRefreshSequenceNum": 20,"epochDisambiguatingSequenceNum": 17},"durationMillis": 896}}
```

Case 5: v5.0 Test Cluster with 5m chunks

- **Cluster Info**

  Set up a v5.0 sharded cluster with 5 million chunks -- Use “id” field as shard key, generate 5m chunks by pre-splitting chunks.

- **Related mongod logs**

```
1.  {"t":{"$date":"2022-10-17T16:15:56.209+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-3","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinceVersion":{"0":{"$timestamp":{"t":49188,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":49189,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":15,"epochDisambiguatingSequenceNum":17},"timeInStore":{"chunkVersion":{"0":{"$timestamp":{"t":49189,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":15,"epochDisambiguatingSequenceNum":16},"durationMillis":2442}}
2.  {"t":{"$date":"2022-10-17T16:15:56.238+08:00"},"s":"I",  "c":"COMMAND",  "id":51803,   "ctx":"conn31","msg":"Slow query","attr":{"type":"command","ns":"test.test2","appName":"MongoDB Shell","command":{"find":"test2","filter":{"id":12},"lsid":{"id":{"$uuid":"60e64fb3-900e-4dc4-8295-0b41f30f9782"}},"$clusterTime":{"clusterTime":{"$timestamp":{"t":1665994442,"i":1}},"signature":{"hash":{"$binary":{"base64":"AAAAAAAAAAAAAAAAAAAAAAAAAAA=","subType":"0"}},"keyId":0}},"$db":"test"},"nShards":1,"cursorExhausted":true,"numYields":0,"nreturned":3,"reslen":663,"readConcern":{"level":"local","provenance":"implicitDefault"},"remote":"127.0.0.1:34674","protocol":"op_msg","durationMillis":3136}}
```

2. Problem Impact

Refreshing routing info happens under a lot of circumstances on mongos & mongod, e.g. splitting & moving chunks, routing requests for read/write queries, adding/removing a shard, etc. Efficiency of refreshing is crucial to MongoDB sharded cluster’s core functionalities.

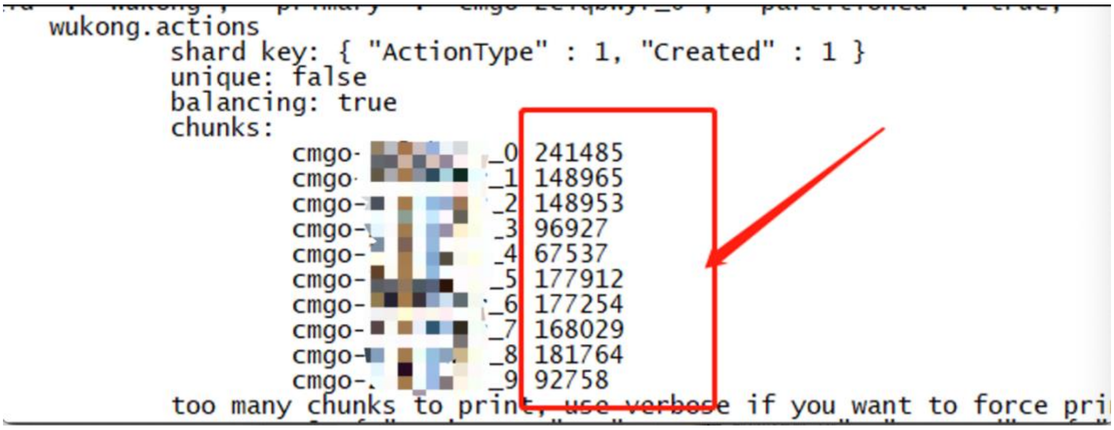
In production clusters, chunk numbers grow rapidly with data keeps flowing in, resulting longer refreshing duration, which in turn caused troubles for users:

- **Cluster performance degradation**

All requests will be blocked when mongos/mongod is retrieving incremental routing info, henceforth the bigger the cluster is, the more un-responsive it could be.

- **Uneven data distribution among shards**

One user had to disable balancer on a sharded cluster with 1.4 million chunks except for several hours during midnight, due to the extreme slow queries caused by refreshing routing info. However the balancing progress made during midnight never caught up to close the gap, and the shards imbalance ends up like below figures, and is still worsening:



```
  wukong.actions
          shard key: { "ActionType" : 1, "Created" : 1 }
          unique: false
          balancing: true
          chunks:
                  cmgo-[redacted]_0 241485
                  cmgo-[redacted]_1 148965
                  cmgo-[redacted]_2 148953
                  cmgo-[redacted]_3 96927
                  cmgo-[redacted]_4 67537
                  cmgo-[redacted]_5 177912
                  cmgo-[redacted]_6 177254
                  cmgo-[redacted]_7 168029
                  cmgo-[redacted]_8 181764
                  cmgo-[redacted]_9 92758
          too many chunks to print, use verbose if you want to force pri
```

- **Increasing development & maintenance complexity**

In order to avoid serious service degradation caused by routing problems, some users would strictly limit the data size of collections. When the data size in a collection exceeds certain threshold (say 4TB), they may need to split the collection manually.

This would add a lot of complexity for development & maintenance and counteracts the advantage of MongoDB's data distribution capability.

- **CPU spikes**

If multiple collections' routing info is being refreshed at the same time, CPU resources would easily exhaust.

3. MongoDB Routing Info Refresh Mechanism Limitation Analysis

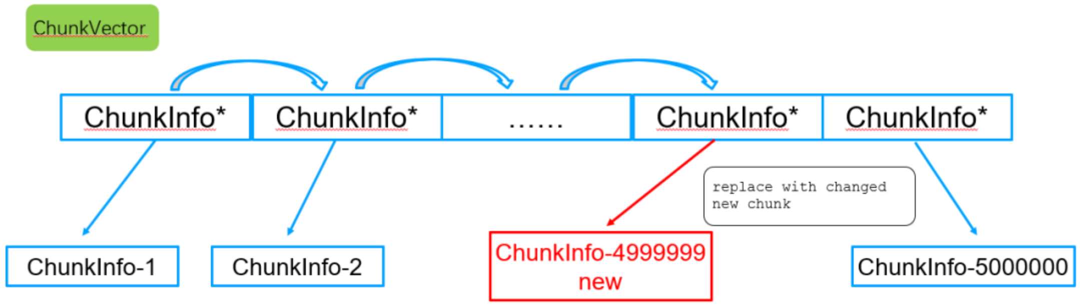
This section briefly introduces the mechanism and its limitation of refreshing routing info in MongoDB v5.0.

3.1 Mongos/mongod incremental routing info retrieval workflow

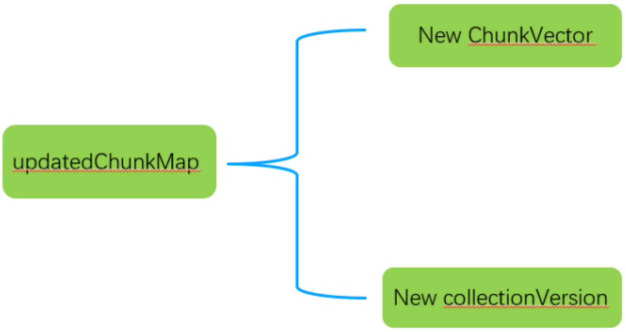
Step 1: Get changedChunk from config server

Retrieve ChunkInfo from config server if its _lastmod is greater than local collectionVersion, then generate changedChunks.

Step 2: Iterate all ChunkVectors' members in old ChunkMap and compare with changedChunks to generate updatedChunkMap.

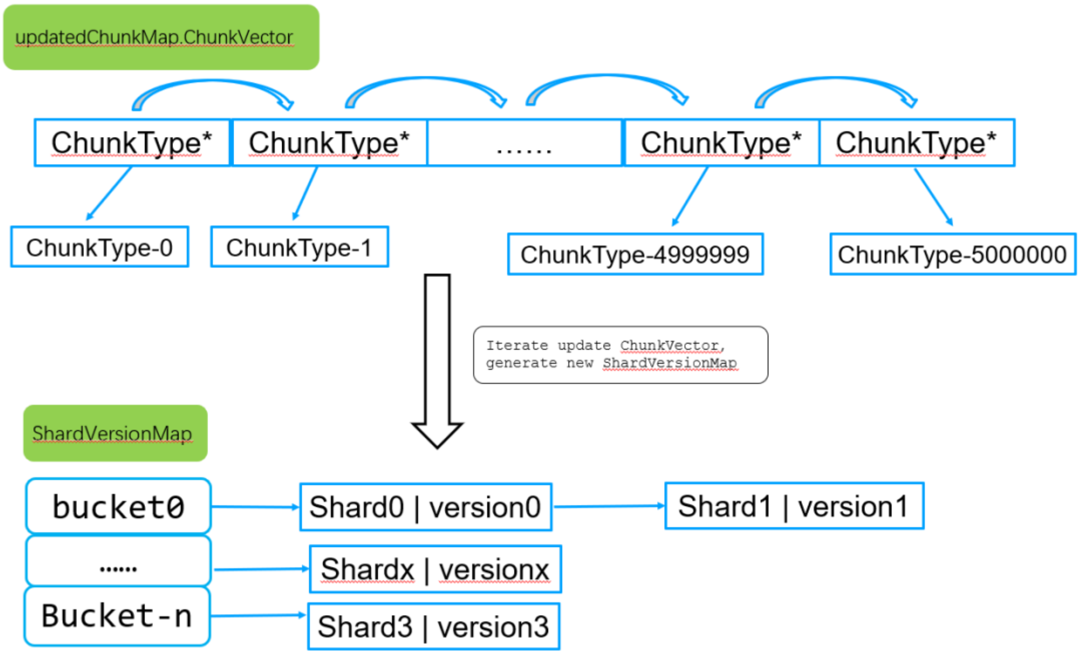


Based on full history routing info in old ChunkMap, and incremental info in changedChunks, generate a new updatedChunkMap where ChunkVector and collectionVersion (or ShardVersionMap) are updated.

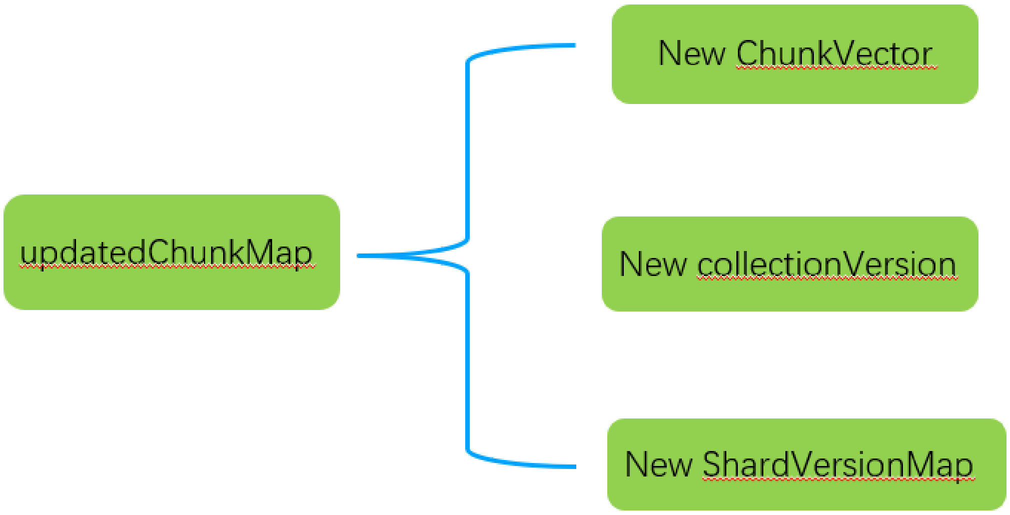


Step 3: Re-iterate updatedChunkMap and generate new ShardVersionMap

Re-iterate chunkInfo in updatedChunkMap.ChunkVector, generate new shardVersion hash table and store it in updatedChunkMap.ShardVersionMap, as below:



As of now a complete updatedChunkMap is generated.



Step 4: Iterate old ChunkVector and free reference counts of shared_ptr

This step also requires a traversal of ChunkVector.

3.2. mongos/mongod incremental routing info performance bottleneck

According to the workflow described in last section, it’s certain that even retrieving just one incremental chunk info requires thre ChunkVector traversals which are obvious performance bottlenecks:

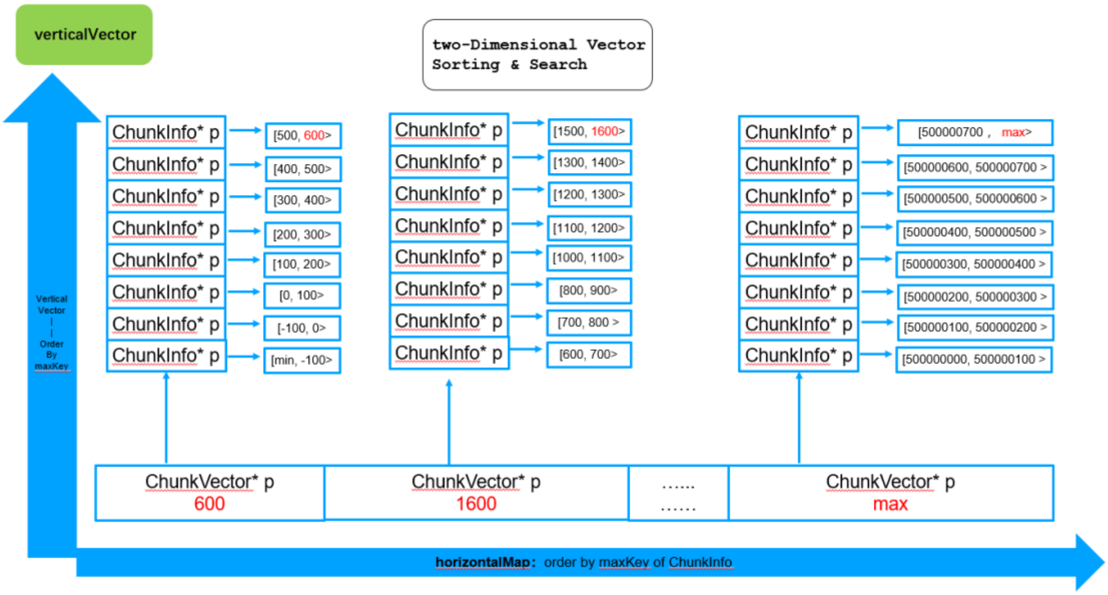
- **Bottleneck 1: Generating updatedChunkMap requires ChunkVector traversal**
- **Bottleneck 2: Generating ShardVersionMap requires ChunkVector traversal**
- **Bottleneck 3: Freeing reference counts of ChunkVector's shared_ptr， requires ChunkVector traversal as well**

Time cost increases exponentially as chunk number grows when these processes are performed, leading to slow user queries.

4. Proposed MongoDB Incremental Routing Info Refresh Method: Two-Dimensional Sorting & Search

This section introduces key processes of the proposed method on ChunkInfo storage, search and update.

4.1 Store Full Routing Info in Two-Dimensional Data Structure



The new method stores and manipulate routing info in ChunkMap with below objects:

- **Horizontal Map**

horizontalMap: std::map< std::pair<std::string, ChunkVector *>>

Each map element stores the maxKey of the ChunkInfo, horizontalMap is sorted by maxKey in ascending order.

- **Vertical Vector**

verticalVector: std::vector<std::shared_ptr<ChunkInfo>>

Each Vector element stores a portion of ChunkInfo ascendingly. Within the verticalVector, chunkInfo is also stored in ascending order.

- **Horizontal Search**

horizontalCursor: Locate target verticalVector among horizontal map by maxKey utilizing Binary Search.

- **Vertical Search**

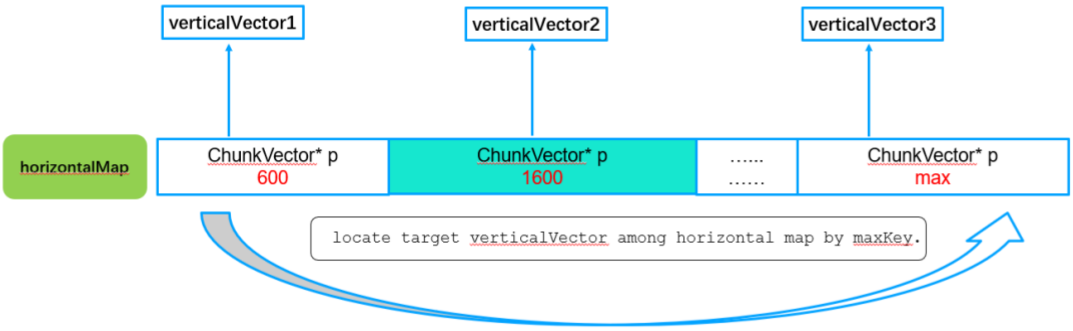
verticalCursor: Also utilize Binary Search to locate chunkInfo among vertical vectors by maxKey.

4.2 Two-Dimensional Search: Point Search + Range Search

Point Search

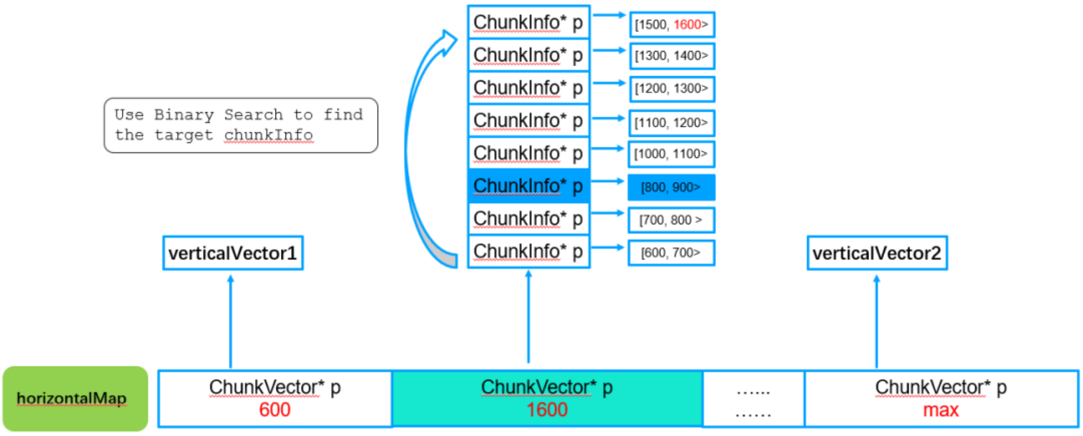
Take search MQL: db.xxx.find({id: 805}) as example:

Step 1: Horizontal search to locate target horizontalMap's cursor



Horizontal binary search: Check std::lower_bound to find the first member whose maxKey being greater than 805, here we find the member with maxKey 1600.

Step 2: Vertical search to locate target ChunkInfo

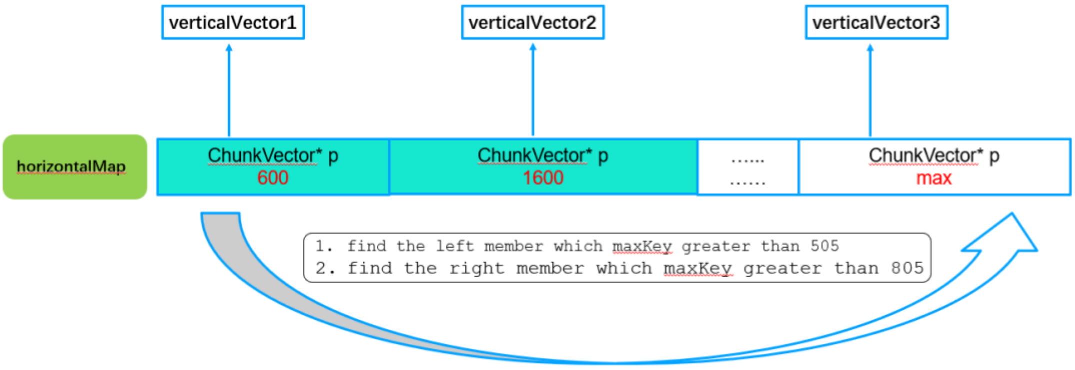


Vertical binary search: Find the first chunkInfo where 805 falls into its key range.

Range Search

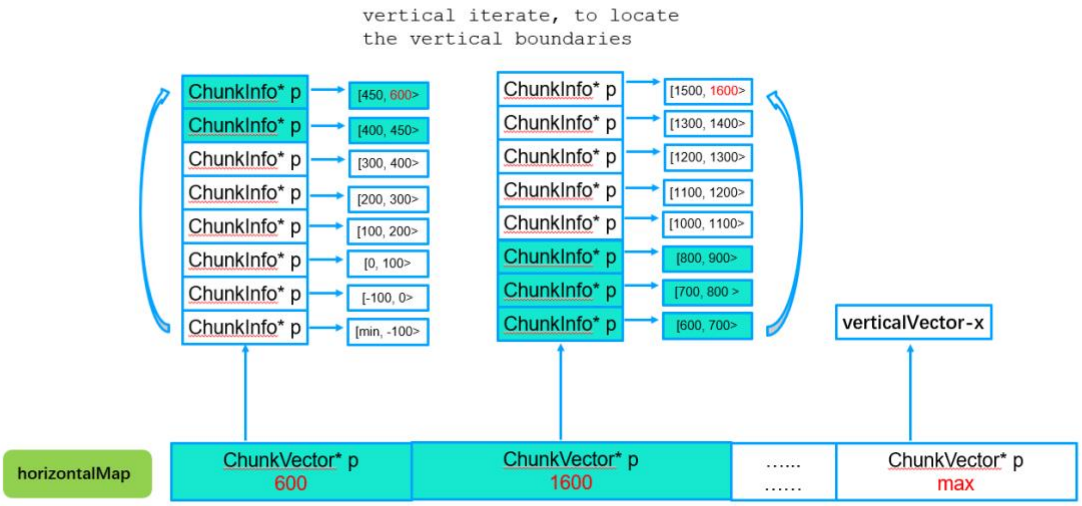
Take search MQL: db.xxx.find({id: {$gte: 505, $lte: 805}) as example:

Step 1: Horizontal search to locate lower and upper boundaries for ChunkVector



Determine the lower and upper boundaries for the cursor of horizontal ChunkVectors in the search range first.

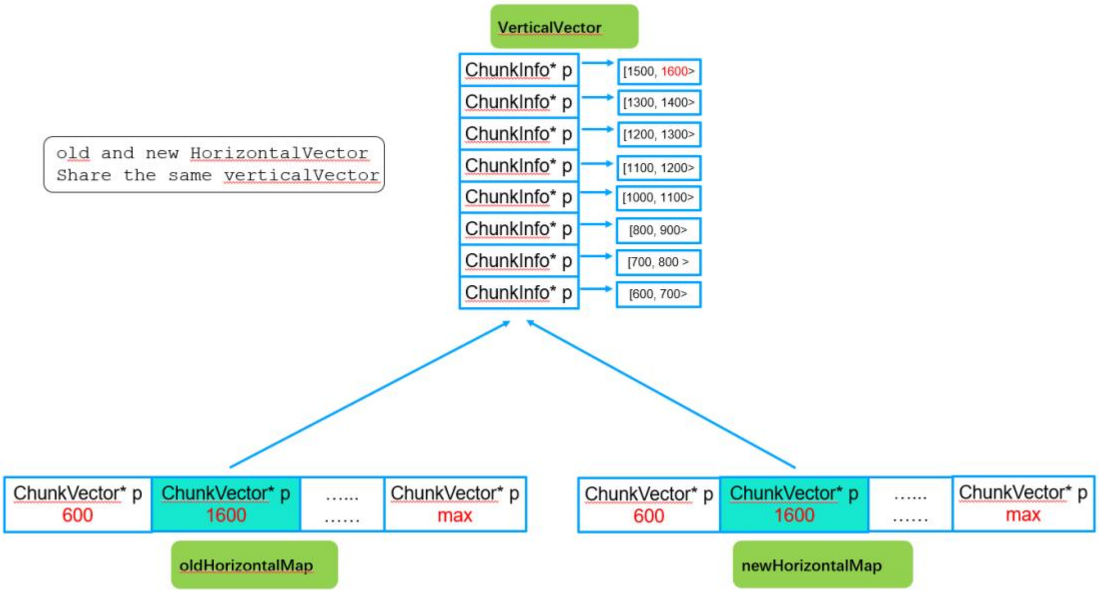
Step 2: Vertical search to locate the boundaries for the ChunkInfo



Then locate the vertical boundaries to get the corresponding ChunkInfo.

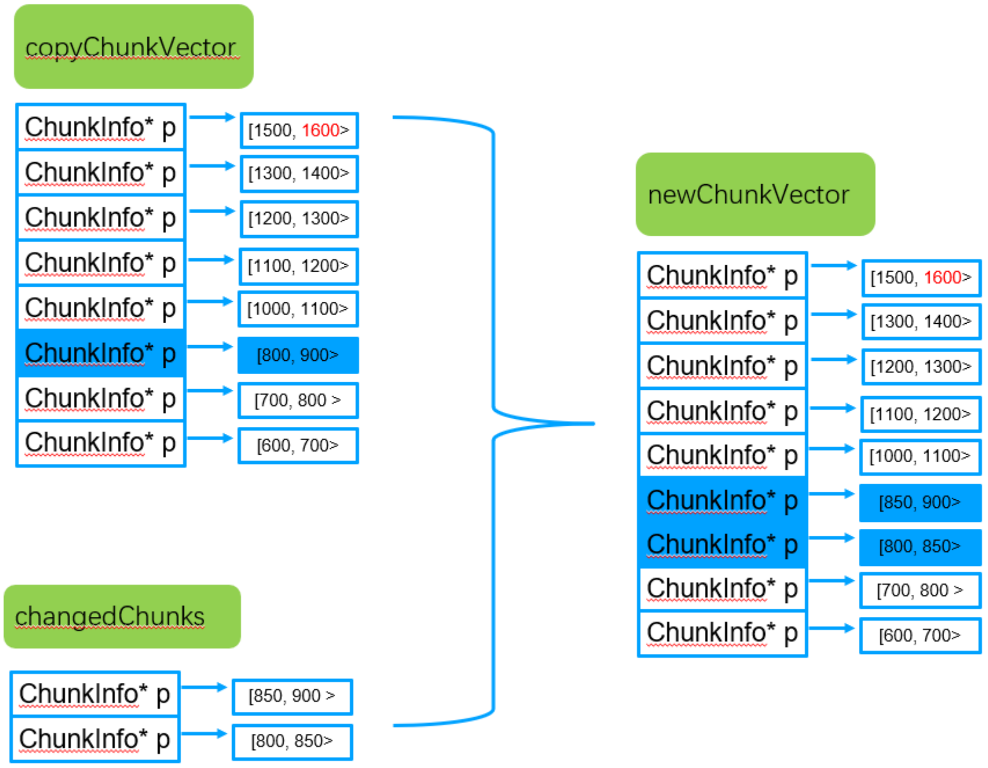
4.3 Generate New ChunkMap Using Incremental Routing Info

Step 1: Duplicate a new horizontalMap



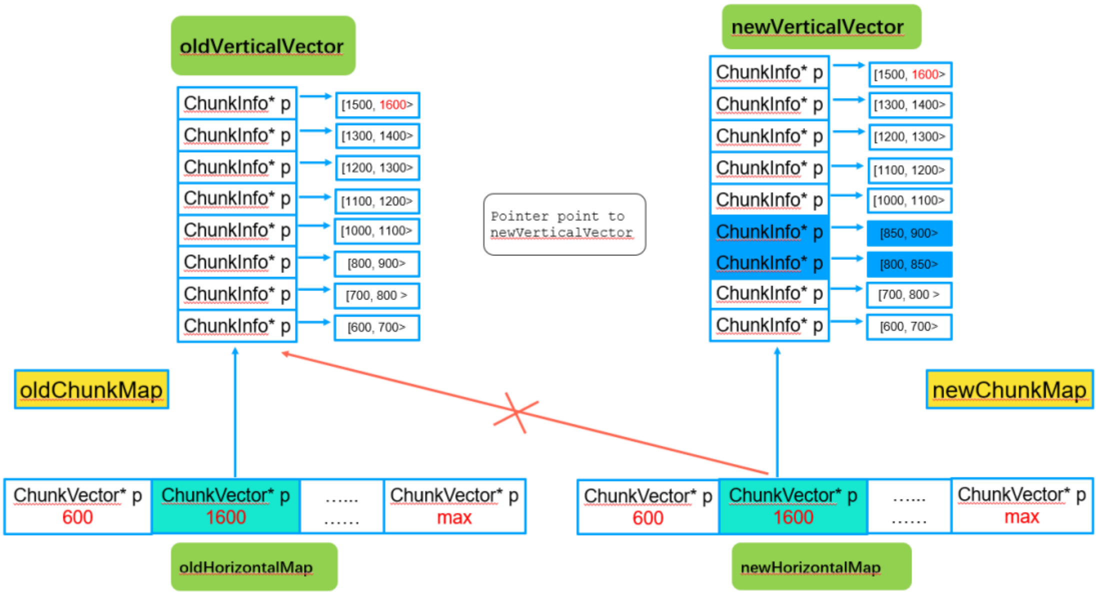
*Duplicate a new horizontal map, with pointers pointing to the same verticalVectors.

Step 2: Generate a new ChunkVector by merging the untouched ChunkInfo and the updated ChunkInfo.



*Use createMerged method to populate the new ChunkVector.

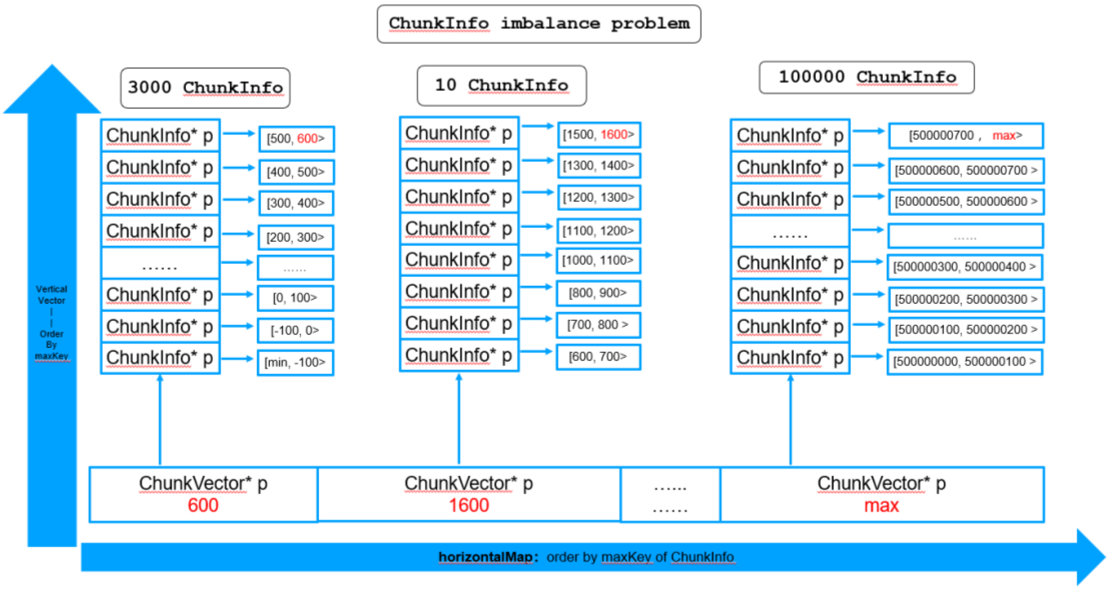
Step 3: Redirect ChunkVector's pointer in the horizontalMap to the newChunkVector



After redirection, the new ChunkMap with updated ChunkInfo is generated.

4.4 Rebalance Mechanism: Vertical Vectors Split

As routing info keeps getting updated, some chunks may have been split too many times while others not, causing imbalance between vertical vectors’ sizes (or depths in the chart), as below:

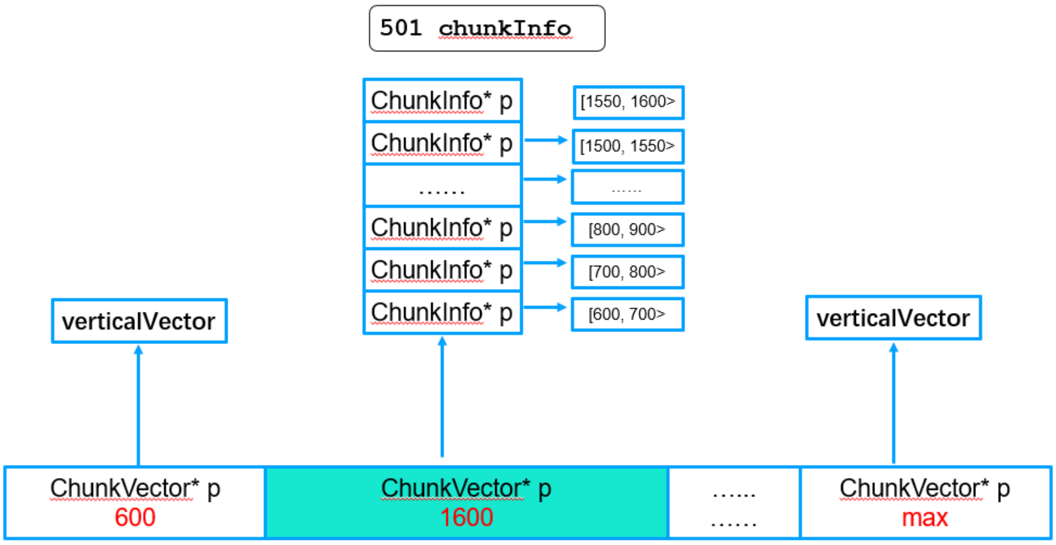


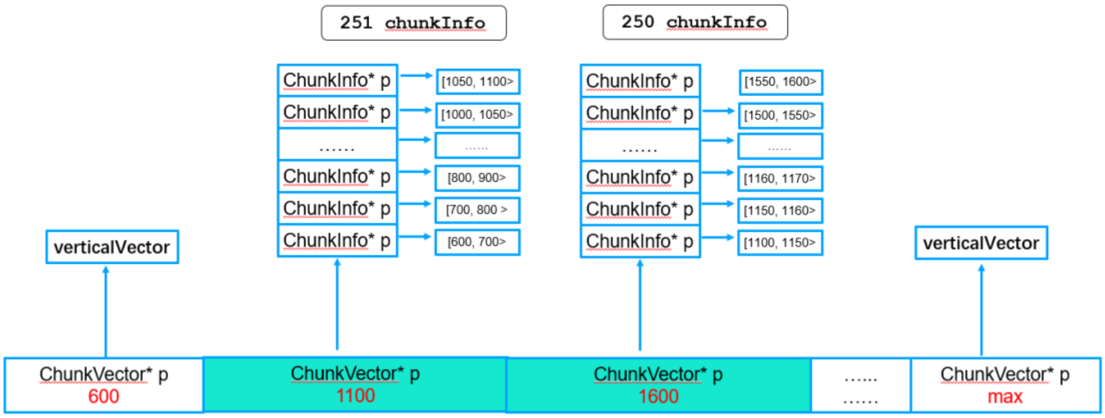
Traversal against the big vertical vector could take exceptionally long time and cause performance issue again. In order to avoid such imbalance, if one vertical vector size is too big, we can split it with following steps:

- **Add a configurable parameter to control vertical vector’s sizes**

Add “routeRefreshCacheVerticalDepth” as a startup parameter for mongos & mongod, default to 500.

- **Automatic balancing - Split ChunkVector**



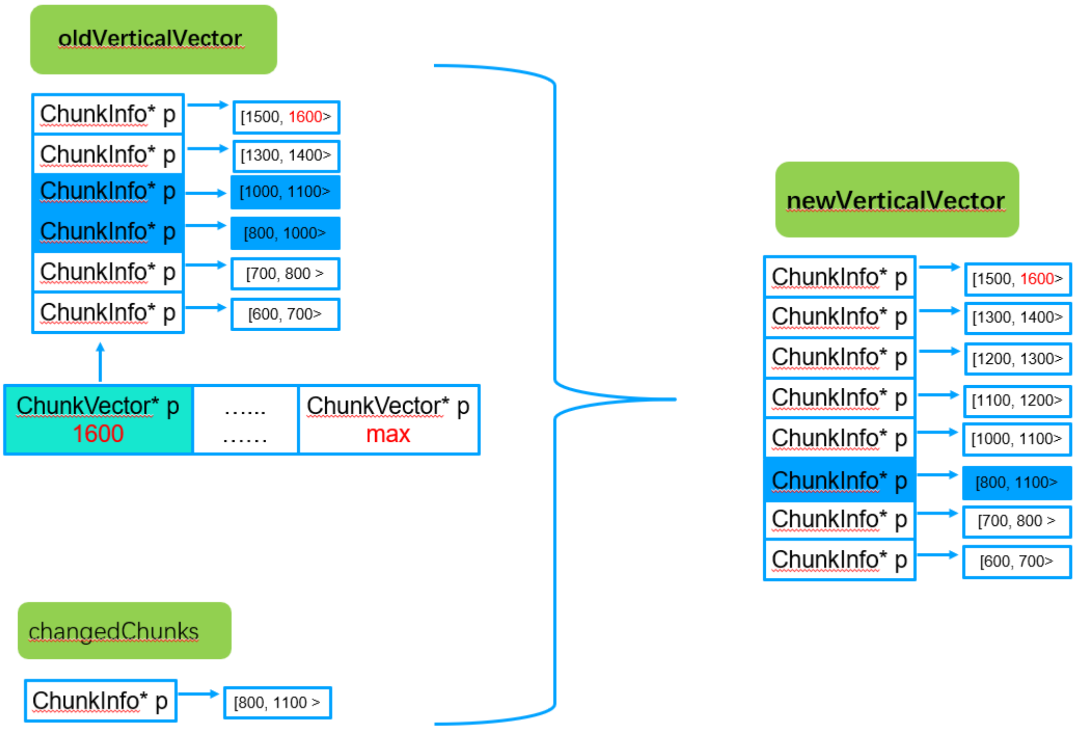


When a verticalVector's size exceeds routeRefreshCacheVerticalDepth, we split it to two vectors.

4.5 Rebalance Mechanism: Vertical ChunkInfo Merge

There're two types of vertical ChunkInfo Merge: merge in single verticalVector and merge between multiple verticalVectors.

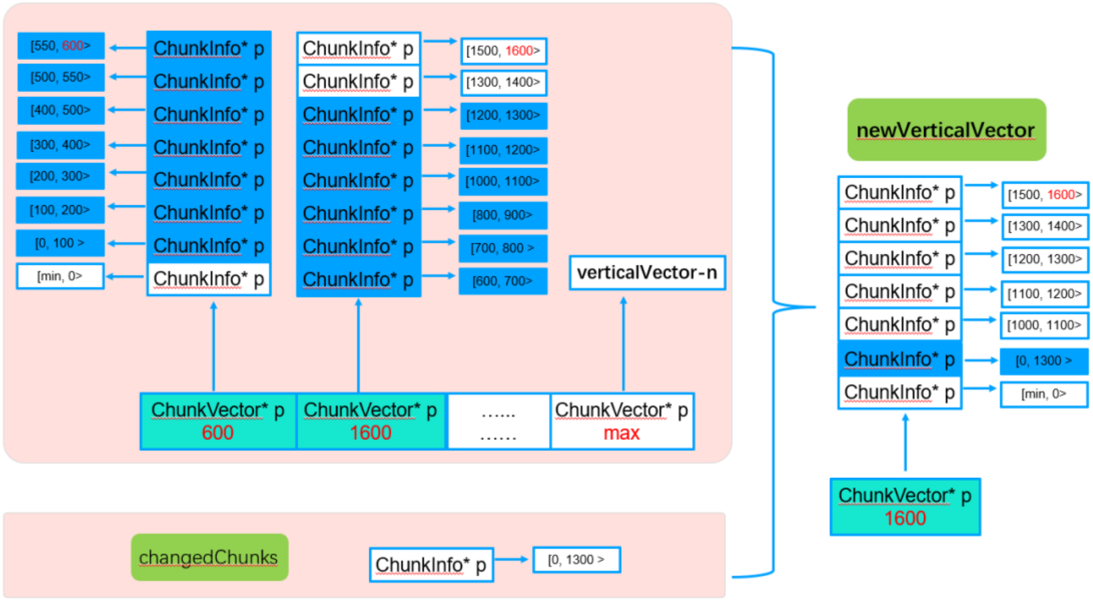
- **Single verticalVector's ChunkInfo merge**



*Iterate oldVerticalVector and changedChunks to generate newVerticalVector. (createMerged)

- **Multiple verticalVectors' ChunkInfo merge**

Besides merge in single verticalVectors, ChunkVectors in horizontalMap will be merged



4.6 Routing Info Validation

When handling updated ChunkInfo, the ChunkMap is updated and its underlying horizontalMap and verticalVector's structure will be changed. The integrity of the updated rounting info needs to be validated along with the ChunkInfo updates. Here's the main checklist:

- **Adjacent ChunkInfo boundaries validation**
  Upper boundary of a ChunkInfo should equal to lower boundary of the next ChunkInfo.
- **Epoch check**

  ChunkInfo from the same collection should have the same epoch.
- **ChunkInfo version check**

  Updated ChunkInfo version should equal to or be greater than existing collectionVersion.
- **MinKey、MaxKey validation**

  In the ChunkMap, the lowest ChunkInfo boundary must be the value defined in macro MinKey, and the uppermost ChunkInfo boundary must be MaxKey's value.

4.7 Summary of Optimization

In the official release, refreshing routing info requires iterating full ChunkInfo in the ChunkMap twice, plus iterating ChunkVector once to free shared pointers – This could be very time- & resource-consuming if the chunk size exceeds certain threshold.

The updated _chunkMap and algorithm in the proposed method requires only one iteration on a very small portion of ChunkInfo based on the changed Chunks to update routing info: _chunkMap, _collectionVersion & _shardVersions.

Pull Request has been created: https://github.com/mongodb/mongo/pull/1505

5. Performance Comparison and Optimization Benefit

- **Performance comparison before and after optimization**

MongoDB Version	Total Data Size(TB)	Total Chunk Number(M)	Elapsed Time of queries(ms)	Elapsed Time after optimization (ms)
3.6	80	4.5	4500	2

4.0	1.2	0.25	300	2
4.2	25	1.5	1200	2
5.0	30	2	910	2
5.0	80	5	2600	2

After optimization, refreshing incremental routing info’s time cost is around 2ms, and most of the elapsed time is spent on retrieving changed chunks from the Config Server, while generating the new ChunkMap only takes a very short period (< 1ms).

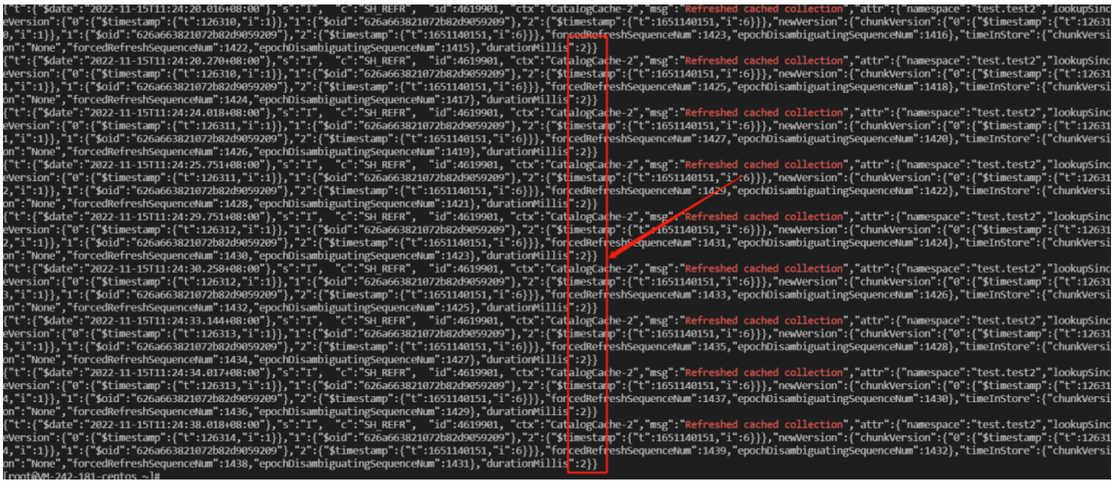
- **Logs before and after optimization(5M chunk size)**

Logs before optimization:

```
{"t": {"$date": "2022-10-17T11: 15: 56.209+08: 00"},"s": "I",  "c": "SH_REFR",  "id": 4619901, "ctx": "CatalogCache-3","msg":
"Refreshed cached collection","attr": {"namespace": "test.test2","lookupSinceVersion": {"0": {"$timestamp": {"t": 49188,"i": 1}},"1":
{"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i": 6}}},"newVersion": {"chunkVersion": {"0":
{"$timestamp": {"t": 49189,"i": 1}},"1": {"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i":
6}}},"forcedRefreshSequenceNum": 15,"epochDisambiguatingSequenceNum": 17},"timeInStore": {"chunkVersion": {"0":
{"$timestamp": {"t": 49189,"i": 1}},"1": {"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i":
6}}},"forcedRefreshSequenceNum": 15,"epochDisambiguatingSequenceNum": 16},"durationMillis": 2442}}
{"t": {"$date": "2022-10-17T11: 15: 56.238+08: 00"},"s": "I",  "c": "COMMAND",  "id": 51803,   "ctx": "conn31","msg":
"Slow query","attr": {"type": "command","ns": "test.test2","appName": "MongoDB Shell","command": {"find": "test2","filter": {"id":
12},"lsid": {"id": {"$uuid": "60e64fb3-900e-4dc4-8295-0b41f30f9782"}},"$clusterTime": {"clusterTime": {"$timestamp": {"t":
1665994442,"i": 1}},"signature": {"hash": {"$binary": {"base64": "AAAAAAAAAAAAAAAAAAAAAAAAAAA=","subType":
"0"}},"keyId": 0}},"$db": "test"},"nShards": 1,"cursorExhausted": true,"numYields": 0,"nreturned": 3,"reslen": 663,"readConcern":
{"level": "local","provenance": "implicitDefault"},"remote": "127.0.0.1: 34674","protocol": "op_msg","durationMillis": 3136}}
```

Logs after optimization:

```
{"t": {"$date": "2022-10-17T15: 40: 01.742+08: 00"},"s": "I",  "c": "SH_REFR",  "id": 4619901, "ctx": "CatalogCache-6","msg":
"Refreshed cached collection","attr": {"namespace": "test.test2","lookupSinceVersion": {"0": {"$timestamp": {"t": 49185,"i": 1}},"1":
{"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i": 6}}},"newVersion": {"chunkVersion": {"0":
{"$timestamp": {"t": 49186,"i": 1}},"1": {"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i":
6}}},"forcedRefreshSequenceNum": 27,"epochDisambiguatingSequenceNum": 29},"timeInStore": {"chunkVersion": {"0":
{"$timestamp": {"t": 49186,"i": 1}},"1": {"$oid": "626a663821072b82d9059209"},"2": {"$timestamp": {"t": 1651140151,"i":
6}}},"forcedRefreshSequenceNum": 27,"epochDisambiguatingSequenceNum": 28},"durationMillis": 2}}
{"t": {"$date": "2022-10-17T15: 40: 01.781+08: 00"},"s": "I",  "c": "COMMAND",  "id": 51803,   "ctx": "conn30","msg":
"Slow query","attr": {"type": "command","ns": "test.test2","appName": "MongoDB Shell","command": {"find": "test2","filter": {"id":
6},"lsid": {"id": {"$uuid": "f77ce42d-af82-4770-bacf-5e754f74eb6f"}},"$clusterTime": {"clusterTime": {"$timestamp": {"t":
1665992391,"i": 1}},"signature": {"hash": {"$binary": {"base64": "AAAAAAAAAAAAAAAAAAAAAAAAAAA=","subType": "0"}},"keyId":
0}},"$db": "test"},"nShards": 1,"cursorExhausted": true,"numYields": 0,"nreturned": 5,"reslen": 737,"readConcern": {"level":
"local","provenance": "implicitDefault"},"remote": "127.0.0.1: 34268","protocol": "op_msg","durationMillis": 3}}
```



```
{"t":{"$date":"2022-11-15T11:24:20.016+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126310,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
0,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1423,"epochDisambiguatingSequenceNum":1416},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1422,"epochDisambiguatingSequenceNum":1415},"durationMillis":2}}
{"t":{"$date":"2022-11-15T11:24:20.270+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126310,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
1,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1425,"epochDisambiguatingSequenceNum":1418},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1424,"epochDisambiguatingSequenceNum":1417},"durationMillis":2}}
{"t":{"$date":"2022-11-15T11:24:24.018+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126311,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
1,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1427,"epochDisambiguatingSequenceNum":1420},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1426,"epochDisambiguatingSequenceNum":1419},"durationMillis":2}}
{"t":{"$date":"2022-11-15T11:24:25.751+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126311,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
2,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1429,"epochDisambiguatingSequenceNum":1422},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1428,"epochDisambiguatingSequenceNum":1421},"durationMillis":2}}
{"t":{"$date":"2022-11-15T11:24:29.751+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126312,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
2,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1431,"epochDisambiguatingSequenceNum":1424},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1430,"epochDisambiguatingSequenceNum":1423},"durationMillis":2}}
{"t":{"$date":"2022-11-15T11:24:30.258+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126312,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
3,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1433,"epochDisambiguatingSequenceNum":1426},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1432,"epochDisambiguatingSequenceNum":1425},"durationMillis":2}}
{"t":{"$date":"2022-11-15T11:24:33.144+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126313,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
3,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1435,"epochDisambiguatingSequenceNum":1428},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1434,"epochDisambiguatingSequenceNum":1427},"durationMillis":2}}
{"t":{"$date":"2022-11-15T11:24:34.017+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126313,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
4,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1437,"epochDisambiguatingSequenceNum":1430},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1436,"epochDisambiguatingSequenceNum":1429},"durationMillis":2}}
{"t":{"$date":"2022-11-15T11:24:38.018+08:00"},"s":"I",  "c":"SH_REFR",  "id":4619901, "ctx":"CatalogCache-2","msg":"Refreshed cached collection","attr":{"namespace":"test.test2","lookupSinc
eVersion":{"0":{"$timestamp":{"t":126314,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"newVersion":{"chunkVersion":{"0":{"$timestamp":{"t":12631
4,"i":1}},"1":{"$oid":"626a663821072b82d9059209"},"2":{"$timestamp":{"t":1651140151,"i":6}}},"forcedRefreshSequenceNum":1439,"epochDisambiguatingSequenceNum":1432},"timeInStore":{"chunkVersi
on":"None","forcedRefreshSequenceNum":1438,"epochDisambiguatingSequenceNum":1431},"durationMillis":2}}
[root@VM-242-181-centos ~]#
```