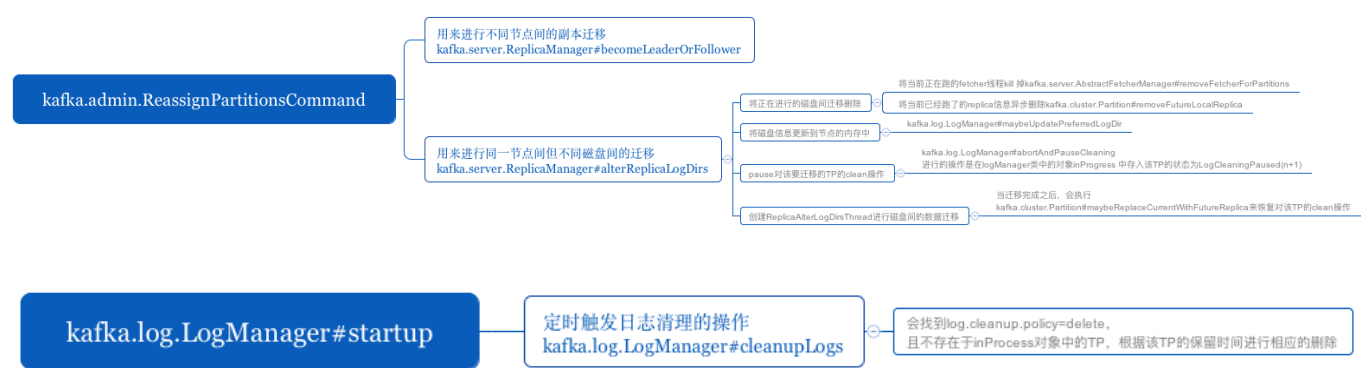


# Kafka - 2.1.1版本迁移bug

- 一、基本理论
- 二、操作过程流程图
- 三、执行过程
  - 3.1执行的命令
  - 3.2第一次迁移内容
  - 3.3第二次迁移内容
  - 3.4结果
  - 3.5导致问题出现的代码块
  - 3.6fix方式

## 一、基本理论



## 二、操作过程流程图



## 三、执行过程

### 3.1执行的命令

```
./kafka-reassign-partitions.sh --zookeeper zkAddress --bootstrap-server  
bootstrap --reassignment-json-file /mnt/storage00/Nora/reassignment211.  
json --execute
```

### 3.2第一次迁移内容

```
{ "partitions":  
  [{ "topic": "lancer_ops_billions_all_log_json_billions",  
    "partition": 1,  
    "replicas": [6,15],  
    "log_dirs": ["any", "/data/mnt/storage02/datum/kafka_data"] } ]  
}
```

### 3.3第二次迁移内容

```
{ "partitions":  
  [{ "topic": "lancer_ops_billions_all_log_json_billions",  
    "partition": 1,  
    "replicas": [6,15],  
    "log_dirs": ["any", "/data/mnt/storage03/datum/kafka_data"] } ]  
}
```

```
Map(lancer_ops_billions_all_log_json_billions-1 -> LogCleaningPaused(1))
```

### 3.4结果

lancer\_ops\_billions\_all\_log\_json\_billions-1

该TP的状态改变情况

从

None

---> Map(lancer\_ops\_billions\_all\_log\_json\_billions-1 -> LogCleaningPaused(1))

---> Map(lancer\_ops\_billions\_all\_log\_json\_billions-1 -> LogCleaningPaused(2))

---> Map(lancer\_ops\_billions\_all\_log\_json\_billions-1 -> LogCleaningPaused(1))

最后保持在 LogCleaningPaused(1)状态。

### 3.5导致问题出现的代码块

```
//futureresume
if (cleaner != null && !isFuture) {
    trace(s"the cleaner is not null and the isfure is false ")

    cleaner.abortCleaning(topicPartition)
    cleaner.updateCheckpoints(removedLog.dir.getParentFile)
}
```

### 3.6fix方式

```
//futuretpresume clean
```