[Elasticsearch: 权威指南](https://elasticsearch.cn/book/elasticsearch_definitive_guide_2.x/%20)

1. **ES概念**

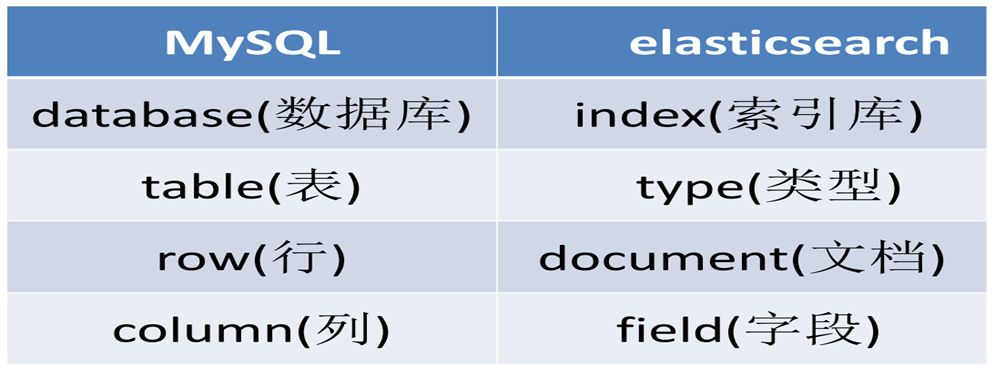


ES是一个最终一致性系统

ES是密集使用磁盘的应用

cluster（集群），node(节点)，index（索引），type(类型)，document（文档），field（字段）,mapping(定义了index中各个字段的类型和属性等，相当于MySQL的table定义语句)

MySQL 与 elasticsearch 的概念对应关系：



（注意，ES中一个index下最好只创建一个type，不要创建多个type！）

### 如何确定分片数

* 1. ES是静态分片，一旦分片数在创建索引时确定那么后继不能修改
  2. 数据量在亿级别，8或者16分片够用，**分片数最好是2的n次方**。
  3. 如果后继数据量的增长超过创建索引的预期，那么需要创建新索引并重灌数据
  4. ES维护团队后继会提供重灌数据的服务
  5. 创建mapping是请自行制定分片数，否则创建的索引的分片数是ES的默认值。这其实并不符合需求。
  6. 副本数：一般设置为1

1. **如何提高写性能**

[Performance Considerations for Elasticsearch Indexing](https://www.elastic.co/blog/performance-considerations-elasticsearch-indexing)

1. **如何提高读性能**

<https://www.elastic.co/blog/found-optimizing-elasticsearch-searches>

# ES性能调优

## 减少refresh次数

LUCENE在新增数据时，为了提高写的性能，采用的是延迟写入的策略，即将数据先写入到内存，当延时超过1秒（默认）时，会出发一次refresh，refresh回把要写入的数据以段的形式刷新到操作系统的文件缓存系统中。当数据以段的形式刷新到文件缓存系统后才可以搜索，所以在搜索的时效性要求不高时，可以增加延时时间，这样可以减少段的数量，为后面段合并减少压力。但是也会消耗更多的Heap内存和搜索延时。

默认的刷新时间间隔是1s，对于写入量很大的场景，这样的配置会导致写入吞吐量很低，适当提高刷新间隔，可以提升写入量，代价就是让新写入的数据在60s之后可以被搜索，新数据可见的及时性有所下降。

"index.refresh\_interval": "60s"

## 减少Flush的次数

当Tranlog的数据量达到512MB或者30分钟（默认），会出发一次刷新（Flush）。刷新主要是将文件缓存系统中的段持久化到磁盘中，因为持久化到磁盘是一个比较耗时的操作。可以通过配置增加Tranlog的缓存数据量。

Index.Tranlog.flush\_threshold\_size=1GB

这样配置意味着文件缓存系统要存储更多的数据，所以要减少Heap的内存空间，为操作系统的文件缓存系统留下更多的空间

# Java中Es操作

## 查询

### Id查询

client.prepareGet(*INDEX\_NAME*, *TYPE\_NAME*, docId)  
 .execute()  
 .actionGet();

### [QueryBuilder简单查询](https://blog.csdn.net/lom9357bye/article/details/52852533)

**matchAllQuery()**

matchQuery("filedname","value")匹配单个字段，匹配字段名为filedname,值为value的文档

**multiMatchQuery(Object text, String... fieldNames)**

**wildcardQuery(String name，String query)模糊查询，name中包含query**

### 指定前缀查询

QueryBuilders.prefixQuery("spell", entity.getSpell())

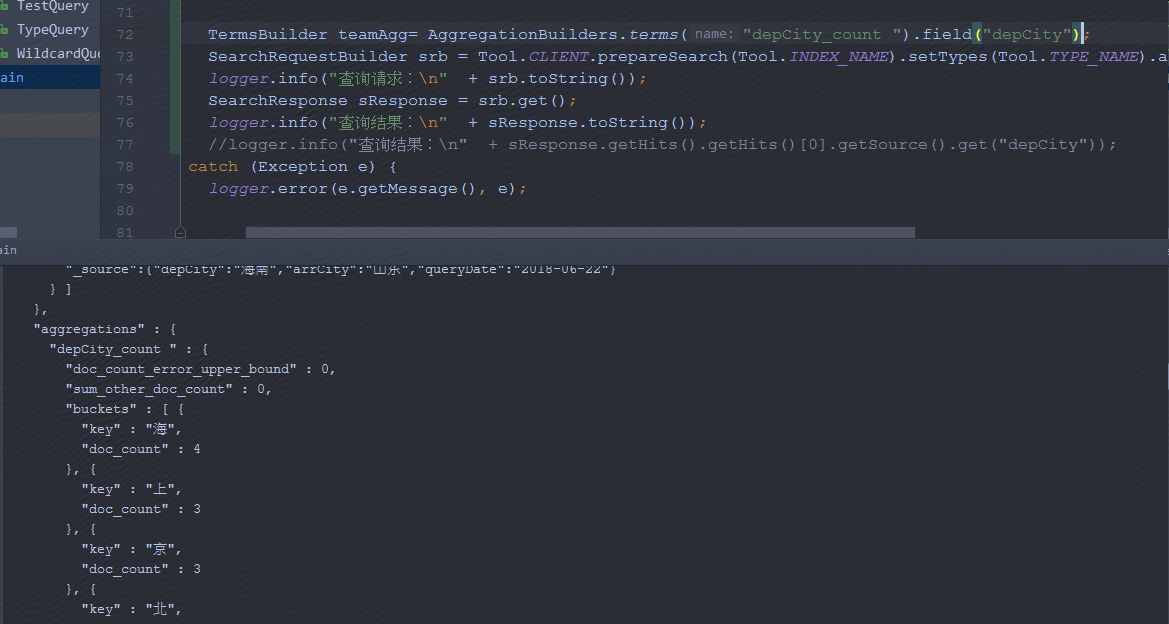
### TermQuery

### [elasticsearch 查询（match和term）](https://www.cnblogs.com/yjf512/p/4897294.html)

### 聚合查询Aggregation

[ElasticSearch AggregationBuilders java api常用聚会查询](https://www.cnblogs.com/xionggeclub/p/7975982.html)

[ElasticSearch 的 聚合（Aggregations）](https://blog.csdn.net/mlljava1111/article/details/70598519)



"aggregations" : {

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "海",

"doc\_count" : 4

}, {

"key" : "上",

"doc\_count" : 3

}, {

"key" : "京",

"doc\_count" : 3

}, {

"key" : "北",

"doc\_count" : 2

}, {

"key" : "南",

"doc\_count" : 2

}, {

"key" : "安",

"doc\_count" : 1

}, {

"key" : "西",

"doc\_count" : 1

} ]

}

}

## 更新（insert、update、delete）

对一个文档进行覆盖，修改，删除，它的版本值都会增加。好好体会一下。

新文档的版本从1开始递增。如果删除一个文档后，再重新插入相同id的文档，那么它的版本并不是从1开始。注意。看下面的例子。

Update：

Tool.*CLIENT*.prepareUpdate(*IndexName*,*TypeName*,"1").setDoc("{\"age\":2}").get();

Insert：

IndexResponse response = Tool.*CLIENT*.prepareIndex(indexName, typeName, docId)  
 .setSource(json)  
 .execute()  
 .actionGet();

Delete

DeleteResponse dResponse = Tool.*CLIENT*.prepareDelete(*IndexName*,*TypeName*,"1").get();

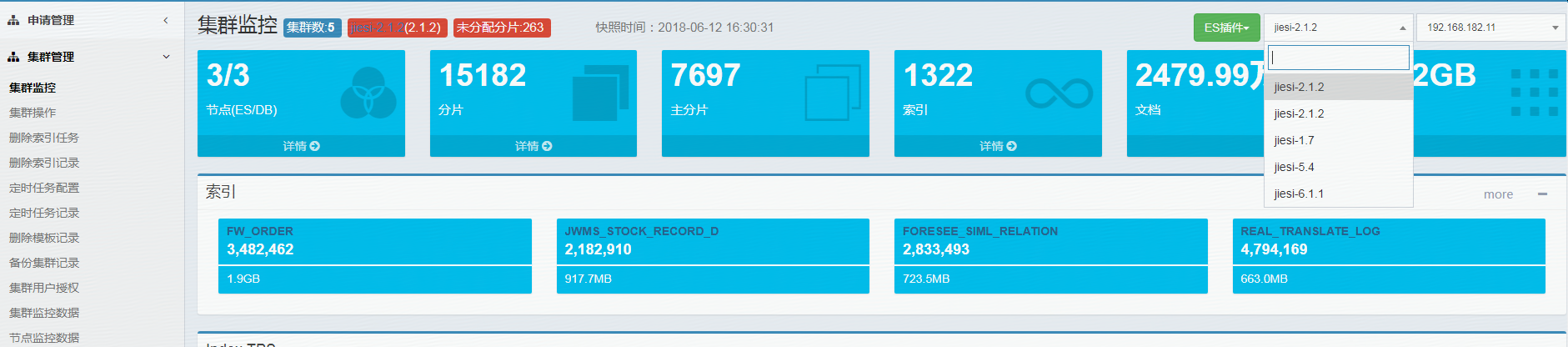
## 分词器：

# [Elasticsearch 中文分词器 IK 配置和使用](https://blog.csdn.net/jam00/article/details/52983056)

# 实战

测试环境：test.esm.jd.com

集群：jiesi-2.1.2



节点：点击详情



## 新建mapping



{

"settings": {

"index": {

"number\_of\_shards": "4",

"number\_of\_replicas": "1"

}

},

"mappings": {

"air\_flight\_query\_type": {

"\_ttl": {

"enabled": false

},

"dynamic": false,

"\_all": {

"enabled": false

},

"properties": {

"depCity": {

"type": "string",

"index": "not\_analyzed"

},

"arrCity": {

"type": "string",

"index": "not\_analyzed"

},

"queryDate": {

"type": "date",

"format": "yyyy-MM-dd HH:mm:ss",

"index": "not\_analyzed"

},

"userPin": {

"type": "string",

"index": "not\_analyzed"

}

}

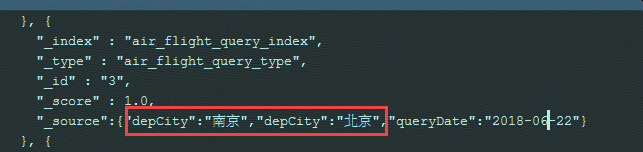
}

}

}

## 数据准备（插入文档）

public static void prepareTestDoc(){  
  
 String json1 = "{" +  
 "\"depCity\":\"北京\"," +  
 "\"arrCity\":\"上海\"," +  
 "\"queryDate\":\"2018-06-22 00:00:00\"," +  
 "\"userPin\":\"huyanxia\"" +  
 "}";  
  
 String json2 = "{" +  
 "\"depCity\":\"北京\"," +  
 "\"arrCity\":\"三亚\"," +  
 "\"queryDate\":\"2018-06-21 00:00:00\"," +  
 "\"userPin\":\"huyanxia1\"" +  
 "}";  
  
 String json3 = "{" +  
 "\"depCity\":\"南京\"," +  
 "\"arrCity\":\"北京\"," +  
 "\"queryDate\":\"2018-06-25 00:00:00\"," +  
 "\"userPin\":\"huyanxia2\"" +  
 "}";  
  
 String json4 = "{" +  
 "\"depCity\":\"西安\"," +  
 "\"arrCity\":\"延安\"," +  
 "\"queryDate\":\"2018-06-22 00:00:00\"," +  
 "\"userPin\":\"huyanxia\"" +  
 "}";  
  
 String json5 = "{" +  
 "\"depCity\":\"海南\"," +  
 "\"arrCity\":\"山东\"," +  
 "\"queryDate\":\"2018-06-22 00:00:00\"," +  
 "\"userPin\":\"huyanxia\"" +  
 "}";  
  
 String json6 = "{" +  
 "\"depCity\":\"上海\"," +  
 "\"arrCity\":\"北京\"," +  
 "\"queryDate\":\"2018-06-19 00:00:00\"," +  
 "\"userPin\":\"huyanxia\"" +  
 "}";  
  
 String json7 = "{" +  
 "\"depCity\":\"上海\"," +  
 "\"arrCity\":\"北京\"," +  
 "\"queryDate\":\"2018-06-22 00:00:00\"," +  
 "\"userPin\":\"huyanxia\"" +  
 "}";  
  
 String json8 = "{" +  
 "\"depCity\":\"上海\"," +  
 "\"queryDate\":\"2018-06-22 00:00:00\"," +  
 "\"userPin\":\"huyanxia5\"" +  
 "}";  
  
  
 *indexWithStr*(Tool.*INDEX\_NAME*, Tool.*TYPE\_NAME*,"1",json1);  
 *indexWithStr*(Tool.*INDEX\_NAME*, Tool.*TYPE\_NAME*,"2",json2);  
 *indexWithStr*(Tool.*INDEX\_NAME*, Tool.*TYPE\_NAME*,"3",json3);  
 *indexWithStr*(Tool.*INDEX\_NAME*, Tool.*TYPE\_NAME*,"4",json4);  
 *indexWithStr*(Tool.*INDEX\_NAME*, Tool.*TYPE\_NAME*,"5",json5);  
 *indexWithStr*(Tool.*INDEX\_NAME*, Tool.*TYPE\_NAME*,"6",json6);  
 *indexWithStr*(Tool.*INDEX\_NAME*, Tool.*TYPE\_NAME*,"7",json7);  
 *indexWithStr*(Tool.*INDEX\_NAME*, Tool.*TYPE\_NAME*,"8",json8);  
}



## 查询

### 全部查询

QueryBuilder qb = QueryBuilders.matchAllQuery();

### 根据docId查询

GetResponse response = Tool.*CLIENT*.prepareGet(Tool.*INDEX\_NAME*,Tool.*TYPE\_NAME*,"1").get();  
*logger*.info("查询结果：\n" + response.getSourceAsMap());

查询结果：

{userPin=huyanxia, queryDate=2018-06-22 00:00:00, arrCity=上海, depCity=北京}

### 精确复合查询

QueryBuilder queryBuilder1 = QueryBuilders.matchQuery("depCity", "上海");  
 QueryBuilder queryBuilder2 = QueryBuilders.matchQuery("arrCity", "北京");  
  
 BoolQueryBuilder boolQueryBuilder = QueryBuilders.boolQuery();  
 boolQueryBuilder.must(queryBuilder1);  
 boolQueryBuilder.must(queryBuilder2);

### date日期范围

Sql:

Select \* from table where queryDate>="2018-06-21 00:00:00" and queryDate<="2018-06-24 00:00:00"

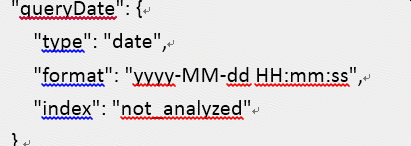
Java api:

BoolQueryBuilder boolQueryBuilder = QueryBuilders.*boolQuery*();  
boolQueryBuilder.must(QueryBuilders.*rangeQuery*("queryDate").from("2018-06-21 00:00:00").to("2018-06-24 00:00:00"));

SearchRequestBuilder srb = Tool.*CLIENT*.prepareSearch(Tool.*INDEX\_NAME*).setTypes(Tool.*TYPE\_NAME*).setQuery(boolQueryBuilder).setSize(20);//setSize(3) 可以设置查询的文档数的大小，默认10

下面也可以，可以控制是开还是闭空间、

Mapping中存储的是



查询使用年月日，默认后面时分秒是00:00:00

Select \* from table where queryDate>"2018-06-21 00:00:00" and queryDate<"2018-06-24 00:00:00"

boolQueryBuilder.must(QueryBuilders.*rangeQuery*("queryDate").format("yyyy-MM-dd").gt("2018-06-21").lt("2018-06-24"));

### 一个Group by 和一个order by

Sql:

Select count(\*) as depCity\_count from table where queryDate>="2018-06-21 00:00:00" and queryDate<="2018-06-24 00:00:00" group by depCity order by depCity\_count asce

Java Api:

BoolQueryBuilder boolQueryBuilder = QueryBuilders.*boolQuery*();  
boolQueryBuilder.must(QueryBuilders.*rangeQuery*("queryDate").from("2018-06-21 00:00:00").to("2018-06-24 00:00:00"));

TermsBuilder teamAgg= AggregationBuilders.*terms*("depCity\_count ").field("depCity").subAggregation(AggregationBuilders.*count*("depCity\_count").field("depCity")).order(Terms.Order.*aggregation*("depCity\_count", true));

SearchRequestBuilder srb = Tool.*CLIENT*.prepareSearch(Tool.*INDEX\_NAME*).setTypes(Tool.*TYPE\_NAME*).setQuery(boolQueryBuilder).addAggregation(teamAgg).setSize(20);//setSize(3) 可以设置查询的文档数的大小，默认10  
*logger*.info("查询请求：\n" + srb.toString());  
SearchResponse sResponse = srb.get();  
*logger*.info("查询结果：\n" + sResponse.toString());

结果：先是查询结果然后再聚合

{

"took" : 1,

"timed\_out" : false,

"\_shards" : {

"total" : 4,

"successful" : 4,

"failed" : 0

},

"hits" : {

"total" : 6,

"max\_score" : 1.0,

"hits" : [ {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "7",

"\_score" : 1.0,

"\_source":{"depCity":"上海","arrCity":"北京","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "8",

"\_score" : 1.0,

"\_source":{"depCity":"上海","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia5"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "1",

"\_score" : 1.0,

"\_source":{"depCity":"北京","arrCity":"上海","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "2",

"\_score" : 1.0,

"\_source":{"depCity":"北京","arrCity":"三亚","queryDate":"2018-06-21 00:00:00","userPin":"huyanxia1"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "4",

"\_score" : 1.0,

"\_source":{"depCity":"西安","arrCity":"延安","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "5",

"\_score" : 1.0,

"\_source":{"depCity":"海南","arrCity":"山东","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

} ]

},

"aggregations" : {

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "海南",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

}, {

"key" : "西安",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

}, {

"key" : "上海",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

}, {

"key" : "北京",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

} ]

}

}

}

### 多个group，多个order

Sql:

Select count(\*) as depCity\_count from table where queryDate>="2018-06-21 00:00:00" and queryDate<="2018-06-24 00:00:00" group by queryDate,depCity order by queryDate asc,depCity\_count desc

BoolQueryBuilder boolQueryBuilder = QueryBuilders.*boolQuery*();  
 boolQueryBuilder.must(QueryBuilders.*rangeQuery*("queryDate").from("2018-06-21 00:00:00").to("2018-06-24 00:00:00"));  
 TermsBuilder teamAgg= AggregationBuilders.*terms*("depCity\_count ").field("depCity").subAggregation(AggregationBuilders.*count*("depCity\_count").field("depCity")).order(Terms.Order.*aggregation*("depCity\_count", true));  
 TermsBuilder dateAgg= AggregationBuilders.*terms*("queryDate\_count").field("queryDate").order(Terms.Order.*term*(true)).subAggregation(teamAgg);

SearchRequestBuilder srb = Tool.*CLIENT*.prepareSearch(Tool.*INDEX\_NAME*).setTypes(Tool.*TYPE\_NAME*).setQuery(boolQueryBuilder).addAggregation(dateAgg).setSize(20);//setSize(3) 可以设置查询的文档数的大小，默认10  
  
 *logger*.info("查询请求：\n" + srb.toString());  
 SearchResponse sResponse = srb.get();  
 *logger*.info("查询结果：\n" + sResponse.toString());

**数据一的请求与结果：**

**请求：**

**{**

**"size" : 20,**

**"query" : {**

**"bool" : {**

**"must" : {**

**"range" : {**

**"queryDate" : {**

**"from" : "2018-06-21",**

**"to" : "2018-06-24",**

**"format" : "yyyy-MM-dd",**

**"include\_lower" : true,**

**"include\_upper" : true**

**}**

**}**

**}**

**}**

**},**

**"aggregations" : {**

**"queryDate\_count" : {**

**"terms" : {**

**"field" : "queryDate",**

**"order" : {**

**"\_term" : "desc"**

**}**

**},**

**"aggregations" : {**

**"depCity\_count " : {**

**"terms" : {**

**"field" : "depCity",**

**"order" : {**

**"depCity\_count" : "desc"**

**}**

**},**

**"aggregations" : {**

**"depCity\_count" : {**

**"value\_count" : {**

**"field" : "depCity"**

**}**

**}**

**}**

**}**

**}**

**}**

**}**

**}**

结果：

"aggregations" : {

"queryDate\_count" : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : 1529712000000,

"key\_as\_string" : "2018-06-23 00:00:00",

"doc\_count" : 3,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "上海",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

}, {

"key" : "上",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

} ]

}

}, {

"key" : 1529625600000,

"key\_as\_string" : "2018-06-22 00:00:00",

"doc\_count" : 4,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "北京",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

}, {

"key" : "海南",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

}, {

"key" : "西安",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

} ]

}

}, {

"key" : 1529539200000,

"key\_as\_string" : "2018-06-21 00:00:00",

"doc\_count" : 2,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "北京",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

} ]

}

} ]

}

}

}

数据二的请求与结果：

查询请求：

{

"size" : 20,

"query" : {

"bool" : {

"must" : {

"range" : {

"queryDate" : {

"from" : "2018-06-21",

"to" : "2018-06-24",

"format" : "yyyy-MM-dd",

"include\_lower" : false,

"include\_upper" : false

}

}

}

}

},

"aggregations" : {

"queryDate\_count" : {

"terms" : {

"field" : "queryDate",

"order" : {

"\_term" : "asc"

}

},

"aggregations" : {

"depCity\_count " : {

"terms" : {

"field" : "depCity",

"order" : {

"depCity\_count" : "desc"

}

},

"aggregations" : {

"depCity\_count" : {

"value\_count" : {

"field" : "depCity"

}

}

}

}

}

}

}

}

2018-06-14 09:39:01,628 main INFO [com.jd.es.example.Main:main] - 查询结果：

{

"took" : 1,

"timed\_out" : false,

"\_shards" : {

"total" : 4,

"successful" : 4,

"failed" : 0

},

"hits" : {

"total" : 7,

"max\_score" : 1.0,

"hits" : [ {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "6",

"\_score" : 1.0,

"\_source":{"depCity":"上","arrCity":"北京","queryDate":"2018-06-23 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "7",

"\_score" : 1.0,

"\_source":{"depCity":"上海","arrCity":"北京","queryDate":"2018-06-23 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "8",

"\_score" : 1.0,

"\_source":{"depCity":"上海","queryDate":"2018-06-23 23:00:00","userPin":"huyanxia5"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "9",

"\_score" : 1.0,

"\_source":{"depCity":"北京","arrCity":"上海","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "1",

"\_score" : 1.0,

"\_source":{"depCity":"北京","arrCity":"上海","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "4",

"\_score" : 1.0,

"\_source":{"depCity":"西安","arrCity":"延安","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "5",

"\_score" : 1.0,

"\_source":{"depCity":"海南","arrCity":"山东","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

} ]

},

"aggregations" : {

"queryDate\_count" : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : 1529625600000,

"key\_as\_string" : "2018-06-22 00:00:00",

"doc\_count" : 4,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "北京",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

}, {

"key" : "海南",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

}, {

"key" : "西安",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

} ]

}

}, {

"key" : 1529712000000,

"key\_as\_string" : "2018-06-23 00:00:00",

"doc\_count" : 2,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "上",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

}, {

"key" : "上海",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

} ]

}

}, {

"key" : 1529794800000,

"key\_as\_string" : "2018-06-23 23:00:00",

"doc\_count" : 1,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "上海",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

} ]

}

} ]

}

}

}

### 时间以天分组

Sql:

Select count(\*) as depCity\_count from table where queryDate>="2018-06-21 00:00:00" and queryDate<"2018-06-24 00:00:00" group by queryDate的天,depCity order by queryDate asc,depCity\_count desc

//时间范围查询  
BoolQueryBuilder boolQueryBuilder = QueryBuilders.*boolQuery*();  
boolQueryBuilder.must(QueryBuilders.*rangeQuery*("queryDate").format("yyyy-MM-dd").gte("2018-06-21").lt("2018-06-24"));  
  
//group by depCity order by count(\*) desc  
TermsBuilder teamAgg= AggregationBuilders.*terms*("depCity\_count ").field("depCity").subAggregation(AggregationBuilders.*count*("depCity\_count").field("depCity")).order(Terms.Order.*aggregation*("depCity\_count", false));  
  
//日期以天分组升序，与teamAgg作为子聚合  
DateHistogramBuilder dateAgg= AggregationBuilders.*dateHistogram*("queryDate\_count")  
 .field("queryDate")  
 .interval(DateHistogramInterval.*DAY*).order(Histogram.Order.*KEY\_ASC*).subAggregation(teamAgg);  
//先根据日期范围查询，再group by queryDate（具体到秒）并升序，再group by depCity order by count(\*) desc  
SearchRequestBuilder srb = Tool.*CLIENT*.prepareSearch(Tool.*INDEX\_NAME*).setTypes(Tool.*TYPE\_NAME*).setQuery(boolQueryBuilder).addAggregation(dateAgg).setSize(20);//setSize(3) 可以设置查询的文档数的大小，默认10  
*logger*.info("查询请求：\n" + srb.toString());  
SearchResponse sResponse = srb.get();  
*logger*.info("查询结果：\n" + sResponse.toString());

查询请求：

{

"size" : 20,

"query" : {

"bool" : {

"must" : {

"range" : {

"queryDate" : {

"from" : "2018-06-21",

"to" : "2018-06-24",

"format" : "yyyy-MM-dd",

"include\_lower" : true,

"include\_upper" : false

}

}

}

}

},

"aggregations" : {

"queryDate\_count" : {

"date\_histogram" : {

"field" : "queryDate",

"interval" : "1d",

"order" : {

"\_key" : "asc"

}

},

"aggregations" : {

"depCity\_count " : {

"terms" : {

"field" : "depCity",

"order" : {

"depCity\_count" : "desc"

}

},

"aggregations" : {

"depCity\_count" : {

"value\_count" : {

"field" : "depCity"

}

}

}

}

}

}

}

}

2018-06-14 10:16:00,047 main INFO [com.jd.es.example.Main:main] - 查询结果：

{

"took" : 2,

"timed\_out" : false,

"\_shards" : {

"total" : 4,

"successful" : 4,

"failed" : 0

},

"hits" : {

"total" : 9,

"max\_score" : 1.0,

"hits" : [ {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "6",

"\_score" : 1.0,

"\_source":{"depCity":"上","arrCity":"北京","queryDate":"2018-06-23 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "7",

"\_score" : 1.0,

"\_source":{"depCity":"上海","arrCity":"北京","queryDate":"2018-06-23 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "8",

"\_score" : 1.0,

"\_source":{"depCity":"上海","queryDate":"2018-06-23 23:00:00","userPin":"huyanxia5"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "9",

"\_score" : 1.0,

"\_source":{"depCity":"北京","arrCity":"上海","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "1",

"\_score" : 1.0,

"\_source":{"depCity":"北京","arrCity":"上海","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "2",

"\_score" : 1.0,

"\_source":{"depCity":"北京","arrCity":"三亚","queryDate":"2018-06-21 00:00:00","userPin":"huyanxia1"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "4",

"\_score" : 1.0,

"\_source":{"depCity":"西安","arrCity":"延安","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "5",

"\_score" : 1.0,

"\_source":{"depCity":"海南","arrCity":"山东","queryDate":"2018-06-22 00:00:00","userPin":"huyanxia"}

}, {

"\_index" : "air\_flight\_query\_index",

"\_type" : "air\_flight\_query\_type",

"\_id" : "10",

"\_score" : 1.0,

"\_source":{"depCity":"北京","arrCity":"三亚","queryDate":"2018-06-21 00:00:00","userPin":"huyanxia1"}

} ]

},

"aggregations" : {

"queryDate\_count" : {

"buckets" : [ {

"key\_as\_string" : "2018-06-21 00:00:00",

"key" : 1529539200000,

"doc\_count" : 2,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "北京",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

} ]

}

}, {

"key\_as\_string" : "2018-06-22 00:00:00",

"key" : 1529625600000,

"doc\_count" : 4,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "北京",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

}, {

"key" : "海南",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

}, {

"key" : "西安",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

} ]

}

}, {

"key\_as\_string" : "2018-06-23 00:00:00",

"key" : 1529712000000,

"doc\_count" : 3,

"depCity\_count " : {

"doc\_count\_error\_upper\_bound" : 0,

"sum\_other\_doc\_count" : 0,

"buckets" : [ {

"key" : "上海",

"doc\_count" : 2,

"depCity\_count" : {

"value" : 2

}

}, {

"key" : "上",

"doc\_count" : 1,

"depCity\_count" : {

"value" : 1

}

} ]

}

} ]

}

}

}

### 分页查询

[Elasticsearch——分页查询From&Size VS scroll](https://blog.csdn.net/z69183787/article/details/70139894)

From&Size是深翻，scroll是快照