Elasticsearch Reference

Indices APIs

Postman Shared Collection

http://goo.gl/Px9jf2

Indices

Create Indices

```
$ curl -XPUT localhost:9200/twitter/

$ curl -XPUT localhost:9200/twitter/ -d '
index :
  number_of_shards : 3
  number_of_replicas : 2
'
```

More about Index Settings => **Index Modules**

```
$ curl -XPUT localhost:9200/twitter/ -d '
{
  "settings" : {
    "index" : {
      "number_of_shards" : 3,
      "number_of_replicas" : 2
    }
  }
}
'
```

or, simply

```
$ curl -XPUT localhost:9200/twitter/ -d '
{
  "settings" : {
    "number_of_shards" : 3,
    "number_of_replicas" : 2
  }
}
'
```

Delete Indices

```
$ curl -XDELETE localhost:9200/twitter/
```

Indices Exist?

```
$ curl -XHEAD localhost:9200/twitter
```

Open/Close Indices

```
$ curl -XPOST localhost:9200/twitter/_close

$ curl -XPOST localhost:9200/twitter/_open
```

Update Index Settings

```
$ curl -XPUT localhost:9200/twitter/_settings -d '
{
  "index" : {
    "number_of_replicas" : 4
  }
}
'
```

Get Index Settings

```
$ curl -XGET localhost:9200/twitter/_settings
```

Supports multiple indices.

```
$ curl -XGET localhost:9200/twitter,kimchy/_settings

$ curl -XGET localhost:9200/_all/_settings

$ curl -XGET localhost:9200/2013-*/_settings
```

Filter settings with `prefix` and `name` options.

```
$ curl -XGET localhost:9200/twitter/_settings?prefix=index.

$ curl -XGET localhost:9200/_all/_settings?prefix=index.routing.allocation.

$ curl -XGET localhost:9200/2013-*/_settings?name=index.merge.*

$ curl -XGET localhost:9200/2013-*/_settings/index.merge.*
```

Mappings

Put Mappings

```
$ curl -XPUT localhost:9200/twitter/tweet/_mapping -d '
{
  "tweet" : {
    "properties" : {
      "message" : {"type" : "string", "store" : true }
    }
  }
}
'
```

More about Mappings => **Mapping**

Supports multi indices.

```
$ curl -XPUT localhost:9200/twitter,facebook/tweet/_mapping -d '
{
  "tweet" : {
    "properties" : {
      "message" : {"type" : "string", "store" : true }
    }
  }
}
'
```

Get Mappings

```
$ curl -XGET localhost:9200/twitter/tweet/_mapping
```

Also, supports multi indices and types.

```
$ curl -XGET localhost:9200/twitter,facebook/_mapping

$ curl -XGET localhost:9200/_all/tweet,post/_mapping
```

Get Field Mappings

```
$ curl -XGET localhost:9200/twitter/tweet/_mapping/field/text
```

Of course, also supports multi indices, types and fields.

```
$ curl -XGET localhost:9200/twitter,facebook/_mapping/field/message

$ curl -XGET localhost:9200/_all/tweet,post/_mapping/field/message,user.id

$ curl -XGET localhost:9200/_all/tw*/_mapping/field/*.id
```

Types Exist?

```
$ curl -XHEAD localhost:9200/twitter/tweet
```

Delete Mappings

```
$ curl -XDELETE localhost:9200/twitter/tweet/_mapping
$ curl -XDELETE localhost:9200/twitter/tweet
```

Deleting Mapping == Deleting Types

Aliases

Index Aliases

Alias == **View** (in RDB)

Add Aliases

```
$ curl -XPOST localhost:9200/_aliases -d '
{
  "actions" : [
    { "add" : { "index" : "twitter", "alias" : "alias1" } }
  ]
}
'
```

Get Aliases

```
$ curl -XGET localhost:9200/_aliases
```

Remove Aliases

```
$ curl -XPOST localhost:9200/_aliases -d '
{
  "actions" : [
    { "remove" : { "index" : "twitter", "alias" : "alias1" } }
  ]
}
'
```

Multi Actions

```
$ curl -XPOST localhost:9200/_aliases -d '
{
  "actions" : [
    { "remove" : { "index" : "twitter", "alias" : "alias1" } },
    { "add" : { "index" : "twitter", "alias" : "alias2" } }
  ]
}
'

$ curl -XPOST localhost:9200/_aliases -d '
{
  "actions" : [
    { "add" : { "index" : "twitter", "alias" : "alias1" } },
    { "add" : { "index" : "facebook", "alias" : "alias1" } }
  ]
}
'
```

Filtered Aliases

```
$ curl -XPOST localhost:9200/_aliases -d '
{
   "actions" : [
      {
         "add" : {
            "index" : "twitter",
            "alias" : "alias2",
            "filter" : { "term" : { "user" : "kimchy" } } }
         }
      }
   ]
}
'
```

More about Filters => **Query DSL**

Filter by **routing values**

```
$ curl -XPOST localhost:9200/_aliases -d '
{
  "actions" : [
    {
      "add" : {
        "index" : "twitter",
        "alias" : "alias1",
        "routing" : "1"
      }
    }
  ]
}
'
```

Different routing values for searching and indexing.

```
$ curl -XPOST localhost:9200/_aliases -d '
{
  "actions" : [
    {
      "add" : {
        "index" : "twitter",
        "alias" : "alias2",
        "search_routing" : "1,2",
        "index_routing" : "2"
      }
    }
  ]
}
'
```

Analyze

Analyze

Performs the **analysis process on a text** and return the tokens breakdown of the text.

```
$ curl -XPOST localhost:9200/_analyze?analyzer=standard -d 'this is a test'

$ curl -XPOST 'localhost:9200/_analyze?tokenizer=keyword&filters=lowercase' \
-d 'this is a test'

$ curl -XPOST 'localhost:9200/_analyze?tokenizer=keyword&token_filters=lowercase
&char_filters=html_strip' -d 'this is a <b>test</b>'

$ curl -XGET localhost:9200/twitter/_analyze?text=this+is+a+test
```

Templates

Index Templates

Index templates allow to define **templates** that will **automatically** be applied to **new indices created**.

Create Templates

```
$ curl -XPUT localhost:9200/_template/template_1 -d '
{
  "template" : "te*",
  "settings" : {
    "number_of_shards" : 1
  },
  "mappings" : {
    "type1" : {
      "_source" : { "enabled" : false }
    }
  }
}
'
```

Will be applied to the indices with **te*** name pattern.

{index} placeholder

```
$ curl -XPUT localhost:9200/_template/template_1 -d '
{
  "template" : "te*",
  "settings" : {
    "number_of_shards" : 1
  },
  "aliases" : {
    "alias1" : {},
    "alias2" : {
      "filter" : {
        "term" : {"user" : "kimchy" }
      },
      "routing" : "kimchy"
    },
    "{index}-alias" : {}
  }
}
'
```

Delete Templates

```
$ curl -XDELETE localhost:9200/_template/template_1
```

Get Templates

```
$ curl -XGET localhost:9200/_template/template_1
$ curl -XGET localhost:9200/_template/temp*
$ curl -XGET localhost:9200/_template/template_1,template_2
$ curl -XGET localhost:9200/_template/
```

Multiple Template Matching by order

```
$ curl -XPUT localhost:9200/_template/template_1 -d '
{
  "template" : "*",
  "order" : 0,
  "settings" : {
    "number_of_shards" : 1
  },
  "mappings" : {
    "type1" : {
      "_source" : { "enabled" : false }
    }
  }
}
'

$ curl -XPUT localhost:9200/_template/template_2 -d '
{
  "template" : "te*",
  "order" : 1,
  "settings" : {
    "number_of_shards" : 1
  },
  "mappings" : {
    "type1" : {
      "_source" : { "enabled" : true }
    }
  }
}
'
```

Config

Index templates can also be placed within **config/templates** directory.

Warmers

Warmers

Index warming allows to run registered search requests to *warm up the index* before it is available for search.

Warmup searches typically include requests that require **heavy loading** of data, such as faceting or sorting on specific fields.

Index Creation with Warmers

```
$ curl -XPUT localhost:9200/test -d '
{
  "warmers" : {
    "warmer_1" : {
      "types" : [],
      "source" : {
        "query" : {
          "match_all" : {}
        },
        "facets" : {
          "facet_1" : {
            "terms" : {
              "field" : "field"
            }
          }
        }
      }
    }
  }
}
'
```

Put Warmers

```
$ curl -XPUT localhost:9200/test/_warmer/warmer_1 -d '
{
  "query" : {
    "match_all" : {}
  },
  "facets" : {
    "facet_1" : {
      "terms" : {
        "field" : "field"
      }
    }
  }
}
'
```

Delete Warmers

```
$ curl -XDELETE localhost:9200/test/_warmer/warmer_1
```

Get Warmers

```
$ curl -XGET localhost:9200/test/_warmer/warmer_1
$ curl -XGET localhost:9200/test/_warmer/warm*
$ curl -XGET localhost:9200/test/_warmer/
```

More GETs

Status

The indices status API allows to get a **comprehensive status information** of one or more indices.

```
$ curl -XGET localhost:9200/twitter/_status
$ curl -XGET localhost:9200/twitter,kimchy/_status
$ curl -XGET localhost:9200/_status
```

Stats

Indices level stats provide **statistics on different operations** happening on an index.

```
$ curl -XGET localhost:9200/twitter/_stats
$ curl -XGET localhost:9200/twitter,kimchy/_stats
$ curl -XGET localhost:9200/_stats
```

Segments

Provides **low level segments information** that a Lucene index (shard level) is built with.

```
$ curl -XGET localhost:9200/twitter/_segments
$ curl -XGET localhost:9200/twitter,kimchy/_segments
$ curl -XGET localhost:9200/_segments
```

Recovery

The indices recovery API provides insight into **on-going shard recoveries**. Recovery status may be reported for specific indices, or cluster-wide.

```
$ curl -XGET localhost:9200/twitter/_recovery
$ curl -XGET localhost:9200/twitter,kimchy/_recovery
$ curl -XGET localhost:9200/_recovery

$ curl -XGET localhost:9200/twitter/_recovery?detailed=true
```

More POSTs

Clear Cache

The clear cache API allows to **clear either all caches or specific caches** associated with one ore more indices.

```
$ curl -XPOST localhost:9200/twitter/_cache/clear
$ curl -XPOST localhost:9200/twitter,kimchy/_cache/clear
$ curl -XPOST localhost:9200/_cache/clear
```

Cache Types

— **filter**

— **field_data**

— **id_cache**

```
$ curl -XPOST localhost:9200/twitter/_cache/clear?filter=true
```

cf. The `filter` cache will be cleared within 60 seconds.

Flush

The **flush** process of an index basically **frees memory** from the index by flushing data to the *index storage* and clearing the internal *transaction log*.

```
$ curl -XPOST localhost:9200/twitter/_flush
$ curl -XPOST localhost:9200/twitter,kimchy/_flush
$ curl -XPOST localhost:9200/_flush
```

Refresh

The refresh API allows to explicitly **refresh** one or more index, *making all operations performed* since the last refresh available for search.

```
$ curl -XPOST localhost:9200/twitter/_refresh
$ curl -XPOST localhost:9200/twitter,kimchy/_refresh
$ curl -XPOST localhost:9200/_refresh
```

Optimize

The **optimize** process basically optimizes the index *for faster search* operations.

The optimize operation allows to *reduce the number of segments* by merging them.

```
$ curl -XPOST localhost:9200/twitter/_optimize
$ curl -XPOST localhost:9200/twitter,kimchy/_optimize
$ curl -XPOST localhost:9200/_optimize
```

Thank You!

by Daniel Ku (http://kjunine.net)