1、定制 dynamic 策略

true：遇到陌生字段，就进行 dynamic mapping
false：遇到陌生字段，就忽略
strict：遇到陌生字段，就报错

```
PUT my_index
{
  "mappings": {
    "my_type":{
        "dynamic":"strict",
        "properties": {
            "title": {
                "type": "text"
            },
            "stash":{
                "type": "object",
                "dynamic":true
            }

        }
    }
  }
}
```

现在插入一个条数据：

```
POST /my_index/my_type
{
  "title":"this is my title",
  "content":"this is my coontent",
  "stash": {
    "province":"jiangxi",
    "city":"nanchang"
  }
}
```

因为把 dynamic 设置为 strict 因此会报以下错误：

```
1  {
2    "error": {
3      "root_cause": [
4        {
5          "type": "strict_dynamic_mapping_exception",
6          "reason": "mapping set to strict, dynamic introduction of [content]
   within [my_type] is not allowed"
7        }
8      ],
9      "type": "strict_dynamic_mapping_exception",
10      "reason": "mapping set to strict, dynamic introduction of [content] within
   [my_type] is not allowed"
11    },
12    "status": 400
13  }
```

2、定制 dynamic mapping 策略

1、date_detection

　　默认会按照一定格式识别 date，比如 yyyy-MM-dd。但是如果某个 field先过来一个 2017-01-01的值，就会被自动 dynamic mapping成 date，后面如果再来一个 "hello world" 之类的值，就会报错。可以手动关闭某个 type 的 date_detection，如果有需要，自己手动 指定某个 field 为 date 类型。

```
1  PUT /my_index/_mapping/my_type
2  {
3    "date_detection": false
4  }
```

2、定制自己的 dynamic mapping template (type level)

```
1  PUT /my_index
2  {
3    "mappings": {
4      "my_type": {
5        "dynamic_templates": [                  // 如果是 匹配上了 通配符 *_es 就采用自己
   定制的 dynamic mapping
6          {
```

```
 7              "en":{
 8                "match":"*_es",
 9                "match_mapping_type":"string",
10                "mapping":{
11                  "type": "text",
12                  "analyzer": "english"
13                }
14              }
15            }
16          ]
17        }
18      }
19  }
```

放入两条数据并执行搜索：

```
 1  POST /my_index/my_type
 2  {
 3    "content": "this is a es test"
 4  }
 5
 6
 7  POST /my_index/my_type
 8  {
 9    "content_es":" this is a test,this one would not be searched out, beacuse IS
    will be filted out."
10  }
11
12  GET /my_index/my_type/_search?q="is"
```

此时结果只有 第一条插入的记录会被查询出来，因为 content 没有匹配到 创建的 dynamic 模板，所有使用的分词器就是默认的 standard 分词器，不会过滤停用词，因此 is 进入倒排索引，所以用 is 搜索是可以搜索出来的。

3、定制自己的default mapping template（index level）es 6.x 不支持

```
 1  PUT /my_index                       // es 6.x 不支持
 2  {
 3    "mappings": {
 4      "_default_":{
```

```
 5        "_all":{
 6            "enabled":false
 7         }
 8      },
 9      "blog":{
10        "_all": {
11            "enabled": true
12         }
13      }
14
15    }
16 }
```