

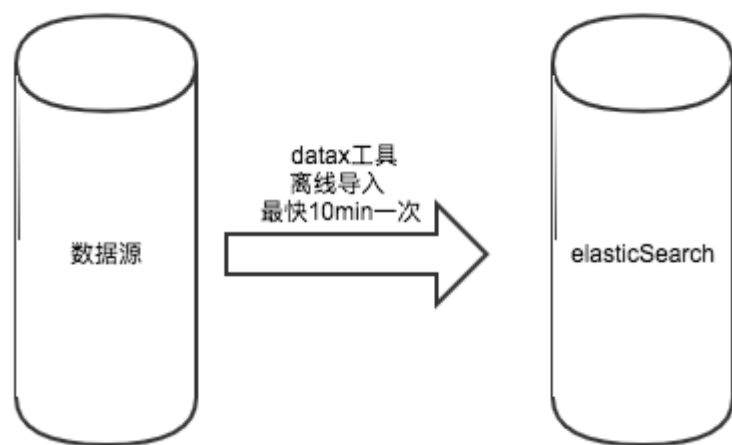
4-es在ether的架构和接入

Created by 杨超, last modified on 2018 Sep 29

- 1 db数据导入
- 2 监控系统
- 3 导入脚本
- 4 具体使用
 - 1 composer
 - 2 config
 - 3 es的ActiveRecord使用
 - 简单用法
 - 高级用法
 - 测试代码1
 - 测试代码2
 - 测试代码3
- 5 todo

1 db数据导入

支持多种 mysql/odps/file



如图所示, es可以看做一个db, 数据源可以来自mysql/odps等各种, 一般来说使用datax工具写一些脚本来进行导入, 目前最快支持10min一次的数据导入
说明两点

1 因为华北1区 阿里云没有资源，所以目前线下的es搭建在newdev上

2 监控系统

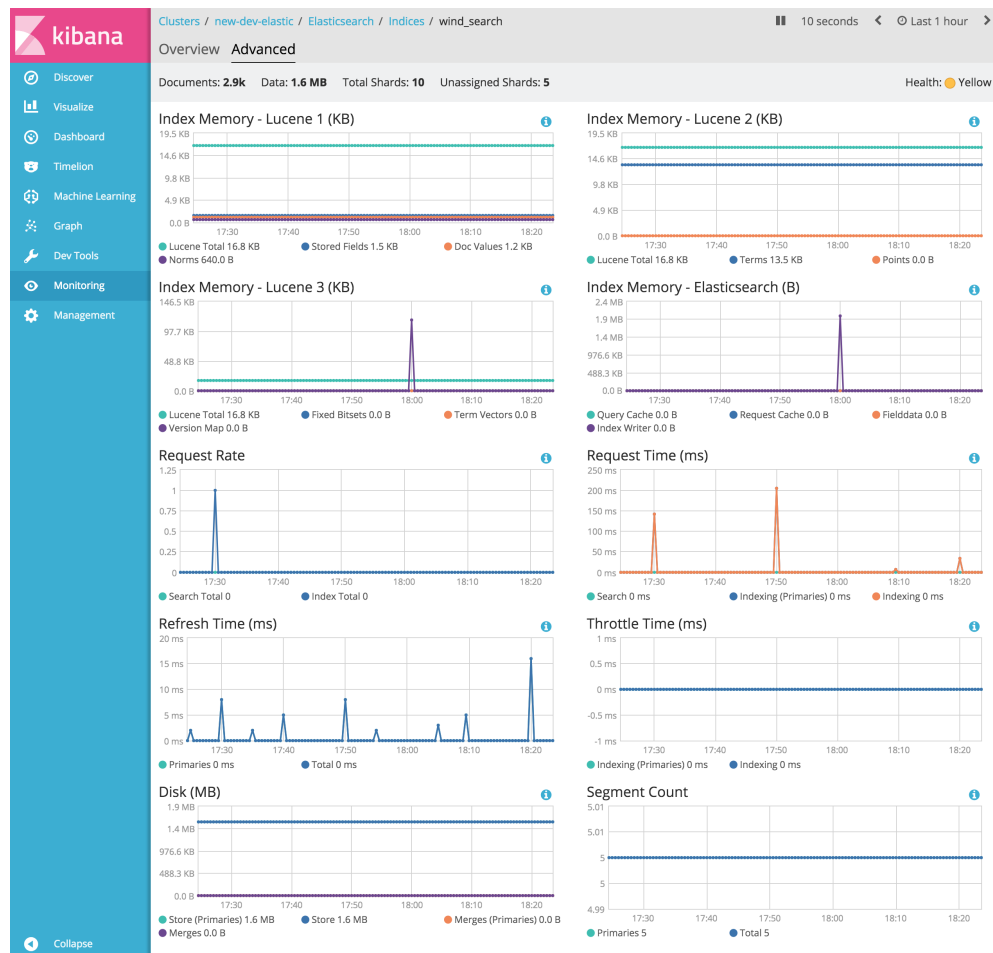
es有配套的kibana监控系统，如图所示 可以监控目前 的索引表数量，耗时等指标

因为kibana权限较高，有需要的同学请找我要账号密码

dev: [http://kibana.newdev.ethercap.com/app/monitoring#/overview?_g=\(cluster_uuid:LQP8GoZ9T_iQ5Q3RsWRBsg\)](http://kibana.newdev.ethercap.com/app/monitoring#/overview?_g=(cluster_uuid:LQP8GoZ9T_iQ5Q3RsWRBsg))

stage/prod: [https://es-cn-v0h0q2d7l0001spov.kibana.elasticsearch.aliyuncs.com:5601/app/monitoring#/overview?_g=\(cluster_uuid:RaWLNuUfTLa5eaqrTVZUsg\)](https://es-cn-v0h0q2d7l0001spov.kibana.elasticsearch.aliyuncs.com:5601/app/monitoring#/overview?_g=(cluster_uuid:RaWLNuUfTLa5eaqrTVZUsg))

Indices <input type="text" value="Filter Indices"/> 2 of 2 <input type="checkbox"/> Show system indices						
Name	Status	Document Count	Data	Index Rate	Search Rate	Un
wind_search	Yellow	2.9k	1.6 MB	0 /s	0 /s	
twitter	Yellow	3	14.2 KB	0 /s	0 /s	



3 导入脚本

主要是datax的数据导入脚本，比较简单，仿照 数据集成-elastic_search/wind_search 编写即可

sample

```
{
  "configuration": {
    "reader": {
      "plugin": "mysql",
      "parameter": {
```

```
    "datasource": "rds_social",
    "column": [
      "uid",
      "uid",
      "content",
      "attr"
    ],
    "where": "",
    "splitPk": "",
    "table": "wind_search"
  }
},
"writer": {
  "plugin": "elasticsearch",
  "parameter": {
    "accessId": "elastic",
    "endpoint": "http://newdev.ethercap.com:9200",
    "indexType": "default",
    "accessKey": "//问我啊",
    "cleanup": true,
    "discovery": false,
    "column": [
      {
        "name": "id",
        "type": "id"
      },
      {
        "name": "uid",
        "type": "keyword"
      },
      {
        "analyzer": "ik_max_word",
        "name": "content",
        "type": "text"
      }
    ]
  }
}
```

```
        "name": "attr",
        "type": "keyword"
    }
],
"index": "wind_search",
"batchSize": 1000,
"splitter": ",",
}
},
"setting": {
    "errorLimit": {
        "record": "0"
    },
    "speed": {
        "throttle": false,
        "concurrent": 1,
        "mbps": "1",
        "dmu": 1
    }
}
},
"type": "job",
"version": "1.0"
}
```

4 具体使用

如果数据已经导入es之后

es有直接的yii2-sdk, 但是我们现在的es版本是5.3, 这个yii2的分支还在开发中, 所以需要工程支持dev的composer

1 composer

```
"minimum-stability": "dev",  
"yii2-elasticsearch": "~2.1.0"
```

2 config

注意dev和stage/prod不同，因为阿里云的es是不允许直连nodes

dev配置

```
'components' => [  
    'elasticsearch' => [  
        'class' => 'yii\elasticsearch\Connection',  
        'autodetectCluster' => true,  
        'nodes' => [  
            // configure more hosts if you have a cluster  
            [  
                'http_address' => 'newdev.ethercap.com:9200',  
                'auth' => ['username' => 'elastic', 'password' => 'sanzhilaoshu123'],  
            ],  
        ],  
    ],  
],
```

stage/prod配置

```
'components' => [  
    'elasticsearch' => [  
        'class' => 'yii\elasticsearch\Connection',  
        'autodetectCluster' => false,  
        'nodes' => [  
            // configure more hosts if you have a cluster  
            [  
                'http_address' => 'es-cn-v0h0q2d7l0001spov.elasticsearch.aliyuncs.com:9200',  
                'auth' => ['username' => '***', 'password' => '***'],  
            ],  
        ],  
    ],  
],
```

```
    ],  
    ],
```

3 es的ActiveRecord使用

走到这里，就可以直接用yii2-es-sdk了

es有对应的ActiveRecord和ActiveQuery, 然后基本的使用和 普通的 ActiveQuery没有太大区别，高阶的用法需要对es有相当的了解

简单使用参考这个: <https://github.com/yiisoft/yii2-elasticsearch/blob/master/docs/guide/README.md>

简单用法

先写出 WindSearch Model， 查询之后返回的结果就是es的ActiveRecord 支持各种event/behavior，使用起来相当方便（比如attrBehavior等）

```
<?php  
  
namespace common\es;  
  
class WindSearch extends \yii\elasticsearch\ActiveRecord  
{  
    public static function index()  
    {  
        return 'wind_search';  
    }  
  
    public static function type()  
    {  
        return 'default';  
    }  
  
    /**  
     * @return array the list of attributes for this record  
     */  
    public function attributes()
```

```
{
    // path mapping for '_id' is setup to field 'id'
    return [
        'id',
        'uid',
        'content',
        'attr',
    ];
}
```

测试代码， 搜索content字段 和 '好'有相似的models， 并且高亮content字段

```
$customers = WindSearch::find()
->where(['content' => '好'])
->highlight([
    'fields' => ['content' => ['type' => 'plain']],
    'force_source' => true,
])
->asArray()->limit(2)->all();
//$customers = WindSearch::find()->where(['content' => $dumb])->limit(2000)->all();
var_dump($customers);
die;
```

结果:

```
array(2) {
    [0]=>
    array(6) {
        ["_index"]=>
        string(11) "wind_search"
        ["_type"]=>
        string(7) "default"
        ["_id"]=>
        string(12) "14af4ffc49b9"
```



```
["_score"]=>
float(1)
["_source"]=>
array(3) {
  ["uid"]=>
  string(12) "14af4ffc49b9"
  ["attr"]=>
  string(98) "39693615e74f28c069e8564e238cdfa31b4d7f5a07956050b1eaa6a9e29e6589c8d9050eaccc82bf42c20a4c6ee47caf49"
  ["content"]=>
  string(9) "大家好"
}
["highlight"]=>
array(1) {
  ["content"]=>
  array(1) {
    [0]=>
    string(18) "大家<em>好</em>"
  }
}
[1]=>
array(6) {
  ["_index"]=>
  string(11) "wind_search"
  ["_type"]=>
  string(7) "default"
  ["_id"]=>
  string(12) "dfe48797d930"
  ["_score"]=>
  float(1)
  ["_source"]=>
  array(3) {
    ["uid"]=>
    string(12) "dfe48797d930"
    ["attr"]=>
    string(96) "697e1c33a06a4d609278e25837c8fe419ca8047b5c29ac90dc2371056ecf7571b6b1237ea817e1f44c5574667f517238"
```

```
["content"]=>
string(56) "裤腿来lol考虑图实际KKK咯哦下午兔兔里Hi好"
}
["highlight"]=>
array(1) {
    ["content"]=>
    array(1) {
        [0]=>
        string(65) "裤腿来lol考虑图实际KKK咯哦下午兔兔里Hi<em>好</em>"
    }
}
}
```

高级用法

这个需要对es语法有相当了解，普通使用者直接问我吧

FaaSRecScoreInvestorTag Model

```
<?php

namespace common\es\query;

class FaaSRecScoreInvestorTagQuery extends \yii\elasticsearch\ActiveQuery
{
    // ?tagIds=116&type=3&searchType=2&page=1&tagInnerJoinType=1&ts=1
    public static function type($type)
    {
        return ['match' => ['type' => $type]];
    }

    public static function ts($ts)
    {
        return ['match' => ['ts' => $ts]];
    }
}
```

```
public static function tagId($tagId)
{
    return ['match' => ['array_tagids' => $tagId]];
}

public static function tagIds($tagIds)
{
    return ['terms' => ['array_tagids' => $tagIds]];
}
}
```

测试代码1

1 这个表示，查出ts=1 and type=3 and tagId in (116,8546) and tagId != 117

```
// complex search
$customers = FaasRecScoreInvestorTag::find()->query([
    'bool' => [
        'must' => [
            FaasRecScoreInvestorTagQuery::ts(1),
            FaasRecScoreInvestorTagQuery::type(3),
            FaasRecScoreInvestorTagQuery::tagIds([116, 8546]),
        ],
        'must_not' => [
            FaasRecScoreInvestorTagQuery::tagId(117),
        ],
    ],
]->all();
```

测试代码2

根据id来获取单个或者批量的数据，很像findOne和findAll

```
$customer = FaasRecScoreInvestorTag::get('21071_3_3'); // get a record by pk
```

```
$customers = FaasRecScoreInvestorTag::mget(['21071_3_3', '21071_3_5']); // get multiple records by pk
```

测试代码3

查出name字段和 '我有一匹马' 模糊匹配的字段

```
// http://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-flt-query.html
$query = FaasRecScoreInvestorTag::find()->query(
    [
        'match' => [
            'name' => [
                'query' => '我有一匹马',
                'fuzziness' => 1,
            ],
        ],
    ],
);
$query->all(); // gives you all the documents
```

5 todo

es的语法和用途还有很多，有待大家去发现

<https://www.elastic.co/guide/index.html>

Like Be the first to like this

No labels

地址：北京市朝阳区建国路86号佳兆业广场北塔6层梦想加空间601室

以太资本由艾普拉斯投资顾问(北京)有限公司运营，提供早期互联网项目的投融资对接服务

©2014-2017 以太资本 京ICP备14028208号

