阿里云 Elasticsearch

最佳实践

文档版本: 20190806

为了无法计算的价值 | [] 阿里云

<u>法律声明</u>

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读 或使用本文档,您的阅读或使用行为将被视为对本声明全部内容的认可。

- 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档,且仅能用于自身的合法 合规的业务活动。本文档的内容视为阿里云的保密信息,您应当严格遵守保密义务;未经阿里云 事先书面同意,您不得向任何第三方披露本手册内容或提供给任何第三方使用。
- 未经阿里云事先书面许可,任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分 或全部,不得以任何方式或途径进行传播和宣传。
- 3. 由于产品版本升级、调整或其他原因,本文档内容有可能变更。阿里云保留在没有任何通知或者 提示下对本文档的内容进行修改的权利,并在阿里云授权通道中不时发布更新后的用户文档。您 应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
- 4. 本文档仅作为用户使用阿里云产品及服务的参考性指引,阿里云以产品及服务的"现状"、"有缺陷"和"当前功能"的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引,但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的,阿里云不承担任何法律责任。在任何情况下,阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害,包括用户使用或信赖本文档而遭受的利润损失,承担责任(即使阿里云已被告知该等损失的可能性)。
- 5. 阿里云网站上所有内容,包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计,均由阿里云和/或其关联公司依法拥有其知识产权,包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意,任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外,未经阿里云事先书面同意,任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称(包括但不限于单独为或以组合形式包含"阿里云"、Aliyun"、"万网"等阿里云和/或其关联公司品牌,上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司)。
- 6. 如若发现本文档存在任何错误,请与阿里云取得直接联系。

通用约定

格式	说明	样例
•	该类警示信息将导致系统重大变更甚至 故障,或者导致人身伤害等结果。	禁止: 重置操作将丢失用户配置数据。
A	该类警示信息可能导致系统重大变更甚 至故障,或者导致人身伤害等结果。	▲ 警告: 重启操作将导致业务中断,恢复业务所需 时间约10分钟。
	用于补充说明、最佳实践、窍门等,不 是用户必须了解的内容。	道 说明: 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定。
courier 字体	命令。	执行 cd /d C:/windows 命令,进 入Windows系统文件夹。
##	表示参数、变量。	bae log listinstanceid Instance_ID
[]或者[a b]	表示可选项,至多选择一个。	ipconfig[-all -t]
{}或者{a b }	表示必选项,至多选择一个。	<pre>swich {stand slave}</pre>

目录

计体生的	т
法律严明	·····I
通用约定	I
1 阿里云Elasticsearch通过Beats搭建可视化运维系统	1
2 Curator操作指南	6
3 数据同步及迁移	
3.1 云上数据导入	9
3.2 使用DataWorks实现Hadoop数据同步到阿里云Elasticsearch	13
3.3 同步 MySQL 数据库到 Elasticsearch 中并进行搜索分析	
3.4 RDS for MySQL与阿里云ES实时同步数据	
3.5 使用DataWorks实现MaxCompute与阿里云ES数据同步	
3.6 通过ES-Hadoop将Hadoop数据写入阿里云Elasticsearch	65
3.7 Logstash部署	
3.8 自建Elasticsearch迁移	

1 阿里云Elasticsearch通过Beats搭建可视化运维系统

本文档以使用Metricbeat采集Mac电脑的指标信息,投递到阿里云Elasticsearch上,并 在Kibana中生成对应Dashborard的场景为例,为您介绍通过Beats和阿里云Elasticsearch搭建 可视化运维系统的方法。

教程概述

Beats平台集合了多种单一用途的数据采集器,这些采集器安装后可用作轻量型代理,从成百上千 或成千上万台机器向Logstash或Elasticsearch发送数据。

Metricbeat是一个轻量级的指标采集器,用于从系统和服务中收集指标。从CPU到内存,从Redis 到Nginx,Metricbeat能够以一种轻量型的方式,输送各种系统和服务的统计数据。

本案例为您演示如何使用Metricbeat采集一台Mac电脑的指标信息,投递到阿里 云Elasticsearch上,并且在Kibana中生成对应的Dashborard,整体步骤如下。

- 1. 准备工作。
- 2. 配置阿里云Elasticsearch。
- 3. 配置Metricbeat。
- 4. 在Kibana中查看Dashboard。

📕 说明:

- · 您也可以参考本案例的步骤,使用Metricbeat采集一台Linux系统或Windows系统电脑的指标信息,投递到阿里云Elasticsearch上。
- ·本案例参考文档:借助Beats快速搭建可视化运维系统。

注意事项

本案例使用了0.0.0.0/1,128.0.0.0/1的Elasticsearch实例公网白名单,这个配置将导致您的阿里 云Elasticsearch基本上完全暴露在公网中,在进行同样配置前请先评估是否可以接受这个风险。

准备工作

在开始本案例前,您需要完成以下准备工作。

· 购买阿里云Elasticsearch实例。



如果您需要通过阿里云Elasticsearch实例的内网地址来访问,还需要先购买一台与阿里 云Elasticsearch实例相同VPC和Region的阿里云ECS实例进行访问操作。

- · 下载Metricbeat。
 - MAC系统的Metricbeat安装包下载地址。
 - 32位Linux系统的Metricbeat安装包下载地址
 - 64位Linux系统的Metricbeat安装包下载地址
 - 32位Windows系统的Metricbeat安装包下载地址
 - 64位Windows系统的Metricbeat安装包下载地址

配置阿里云Elasticsearch

- 1. 登录阿里云Elasticsearch控制台,单击实例ID > 安全配置。
- 打开公网地址开关,待配置生效后,单击公网地址访问白名单右侧的修改,将您MAC机器对外的公网IP配置到公网地址访问白名单中。

<	es-cn-v=	修	改公网访问白名单	×
基本信息	集时网络设置		② 支持配置单个ip成p网站的形式。格式为192.168.0.1或192.168.0.0/24,多个ip用能交运号 周开:127.0.0.1代表想上所有px+地址出向,0.0.0.0(代表的上所有px+地址出向,目前	
活件配置	ISERIARE BEE		杭州区或波增公阁ipvd抵起访问,并可以配置ipvd后张单,推式为2401b180;10002458 2401b180:1000;48:::1代表基止所有ipv688世访问,::/V代表作所有ipv688世访问,详 最参考文档	
集群监控		的上记	=1,127.0.0.1,128.0.0.0/1	
日志查询	(E/ATTPS:16%): 💭 🛛			
安全配置				
数据备份				
可视化控制				
 ■ 智能活用 				
集群概况				
建成沙斯				
历史报告				

!) 注意:

如果您使用的是公司或WIFI等网络,需要将公网出口的跳板机IP配置进去。如果获取不到,建 议配置0.0.0.0/1,128.0.0.0/1来开放尽可能多的IP(本篇以此为例)。需要特别注意这个配置 将导致您的阿里云Elasticsearch基本上完全暴露在公网中,需要先评估下是否可以接受这个风 险。

3. 返回实例的基本信息页面,获取您阿里云Elasticsearch实例的公网地址备用。

基本信息	基本信息	转包平包月
ES集群配置		AUR###30.004の日本日本日 40.57.00
插件配置	Angle to Contraction for the second for	初連月间、2019年3月0日 10.57.39
集群监控	- Arky, IDESMIGRATOT_UV_JUNWER 編編	
日志童询	Eldsutsealth 版本。5.5.5_will_A-Fatx	可用尺: cn-hanozhou-b
安全配置	专有网络:vp	VSwitch信息: vsv
数据备份	内网地址: es-	内网端曰: 9200
▼ 智能运维	公网地址: es	公网满口: 9200
集群概况	配置信息	集群介配
健康诊断	数据节点规格: elasticsearch.n4.small(1核 2GB)	数据节点数量:2
历史报告	存储频格 高效云盘	存储容量: 20 GB

4. 切换到ES集群配置页面,单击YML文件配置右侧的修改配置,将自动创建索引设置为允许自动 创建索引。

es-cn-		YML参数配置	×
分词配置		自动创建素	I: 不允许自动创建索引 ● 允许自动创建索引 ● 用電义 Auto
F YML文件配置	11.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1	勤除 索引描定名4	
自动创建来引 Auditlog要引	l: 大)F国政治理查引 🕢	集別描定記 Auditlog 条: 开启Watch	:
其他configure配置	• 0	开启Watch	m. ● 关闭
		其他configure的	ē. Ø



此配置需要重启您的Elasticsearch实例才能生效,为保证您的业务不受影响,请谨慎操作。

5. 勾选该操作会重启实例,请确认后操作,单击确认。

此过程约持续30分钟左右,请您耐心等待。重启完成后,即可完成Elasticsearch实例的配置。

配置Metricbeat

1. 将在准备工作中下载的Metricbeat安装包解压缩,并进入Metricbeat文件夹。

zhaohongyangdeMacBook-Pro:Desktop zhaohongyang\$ cd metricbeat-6.3.1-darwin-x86_64 zhaohongyangdeMacBook-Pro:metricbeat-6.3.1-darwin-x86_64 zhaohongyang\$ ls							
LICENSE.txt	data	logs	<pre>metricbeat.yml</pre>				
NOTICE.txt	fields.yml	metricbeat	modules.d				
README.md	kibana	<pre>metricbeat.reference.yml</pre>					
zhaohongyangdeMacBook-Pro:metricbeat-6.3.1-darwin-x86.64 zhaohongyang\$							

2. 打开并编辑metricbeat.yml中Elasticsearch output部分内容,并取消对应内容的注释

状态。



- hosts:为阿里云Elasticsearch实例的公网/内网地址(本篇以阿里云Elasticsearch实例 的公网地址为例)。
- · protocol: 需要配置为http。
- ・username: 默认是elastic。
- · password: 为购买阿里云Elasticsearch实例时填写的登录密码。
- 3. 执行以下命令, 启动Metricbeat。

```
./metricbeat -e -c metricbeat.yml
```

启动成功后,Metricbeat就开始向您的阿里云Elasticsearch推送数据了。

zhaohongyangdeMacBook-Pro:metricbeat-6.3.1-darwin-x86_64 zhaohongyang\$./metricbeat -e -c metricbeat.yml[]

在Kibana中查看Dashboard

进入您阿里云Elasticsearch实例的Kibana控制台,单击左侧导航栏的Dashboard,进入Dashboard页面查看相关信息。



如果Kibana控制台中没有创建过Index Patterns,切换到Dashboard页面后可能无法正常展示 对应信息。此时可在Kibana控制台的Management页面单击Index Patterns,并按照提示创建 一个Index Patterns,再切换到Dashboard页面查看对应内容。

・各类相关指标列表。

X	Dashboard	
∅	⊘ Q Search	+ 1-20 of 20 < >
\odot	Image: Second	
8	C Golang: Heap	
۲	Kubernetes overview	
	Metricbeat - Apache HTTPD server status	
۶	Metricbeat CPU/Memory per container	
	Metricbeat Docker	
\$	Metricbeat Hosts Overview	
	Metricbeat MongoDB	
	Metricbeat MySQL	
	Metricbeat filesystem per Host	
	Metricbeat host overview	
	Metricbeat system overview	
2	▲ Metricbeat-Rabbitmg	
Ð	- Metricbeat-cpu	
٥	Metricbeat-filesystem	

· Metricbeat-cpu指标信息。



▋ 说明:

您可以将数据定义成5s刷新一次,并且可以生成对应的报表,接入webhook对异常进行告警。

2 Curator操作指南

Curator是Elasticsearch官方的一个索引管理工具,提供了删除、创建、关闭、段合并索引等功能。本文档为您介绍Curator的使用方法,包括安装、单命令行执行、定时执行、冷热数据分离实践、索引跨节点迁移等。

概述

本文档为您讲解了如何安装Elasticsearch Curator,如何使用单命令行执行,以及如何使用 curator命令完成crontab定时执行、冷热数据分离实践、将索引从hot节点迁移到warm节点等 操作。

准备工作

在安装Curator前,首先要购买一台阿里云ECS实例(本文档以CentOS 7.3 64位的ECS为

例),所购买的实例需要与您的阿里云Elasticsearch实例在同一个VPC下。

安装Elasticsearch Curator

在您购买的ECS命令行界面,执行以下命令安装Curator。

pip install elasticsearch-curator

🗾 说明:

- ·建议您安装5.6.0版本的Curator,它可以支持阿里云Elasticsearch 5.5.3和6.3.2版本。
- · 请参考官方文档查看Curator版本与Elasticsearch版本的兼容性。

安装成功后,执行以下命令查看Curator版本。

curator --version

正常情况下的返回结果如下。

curator, version 5.6.0

单命令行执行

您可以使用curator_cli命令执行单个操作,命令行使用方式请参考官方文档。

📋 说明:

- ・curator_cli只能执行一个操作。
- ·并不是所有的操作都适用于单命令行执行,例如Alias和Restore操作。

crontab定时执行

您可以通过crontab和curator命令实现定时执行一系列操作。

curator命令格式如下。

```
curator [OPTIONS] ACTION_FILE
Options:
    --config PATH Path to configuration file. Default: ~/.curator/
curator.yml
    --dry-run    Do not perform any changes.
    --version    Show the version and exit.
    --help    Show this message and exit.
```

执行curator命令时需要指定config.yml文件(官方参考文档)和action.yml文件(官方参考文 档)。

冷热数据分离实践

详细操作方法请参考使用Curator进行冷热数据迁移(官方文档)。

将索引从hot节点迁移到warm节点

1. 在/usr/curator/路径下创建config.yml文件, 配置内容参考如下示例。

```
client:
  hosts:
    - http://es-cn-0pxxxxxxxxxx234.elasticsearch.aliyuncs.com
  port: 9200
 url_prefix:
  use_ssl: False
  certificate:
  client_cert:
  client_key:
  ssl_no_validate: False
 http_auth: user:password
 timeout: 30
 master_only: False
logging:
  loglevel: INFO
 logfile:
  logformat: default
  blacklist: ['elasticsearch', 'urllib3']
```

· hosts:将该参数值替换为需要访问的阿里云Elasticsearch实例地址(此处

以Elasticsearch内网地址为例)。

- · http_auth:将该参数值替换为对应阿里云Elasticsearch实例的账号和密码。
- 2. 在/usr/curator/路径下创建action.yml文件, 配置内容参考如下示例。

```
actions:
    1:
        action: allocation
        description: "Apply shard allocation filtering rules to the
specified indices"
```

```
options:
 key: box_type
 value: warm
 allocation_type: require
 wait_for_completion: true
 timeout_override:
 continue_if_exception: false
  disable_action: false
filters:
filtertype: pattern
 kind: prefix
  value: logstash-
- filtertype: age
  source: creation_date
 direction: older
  timestring: '%Y-%m-%dT%H:%M:%S'
 unit: minutes
 unit_count: 30
```

以上示例按照索引创建时间,将30分钟前创建在hot节点以logstash-开头的索引迁移到warm

节点中。您也可以根据实际场景自定义配置action.yml文件。

3. 执行以下命令,验证curator命令能否正常执行。

```
curator --config /usr/curator/config.yml /usr/curator/action.yml
```

正常情况下会返回类似如下所示的信息。

```
Preparing Action ID: 1, "
2019-02-12 20:11:30,607 INFO
allocation"
2019-02-12 20:11:30,612 INFO
                                  Trying Action ID: 1, "allocation
": Apply shard allocation filtering rules to the specified indices
2019-02-12 20:11:30,693 INFO
                                  Updating index setting {'index.
routing.allocation.require.box_type': 'warm'}
2019-02-12 20:12:57,925 INFO
                                  Health Check for all provided keys
passed.
2019-02-12 20:12:57,925 INFO
                                  Action ID: 1, "allocation"
completed.
2019-02-12 20:12:57,925 INFO
                                  Job completed.
```

4. 执行以下命令,使用crontab实现每隔15分钟定时执行curator命令。

```
*/15 * * * * curator --config /usr/curator/config.yml /usr/curator/
action.yml
```

3数据同步及迁移

3.1 云上数据导入

```
阿里云上数据导入阿里云ES(离线)
```

阿里云上拥有丰富的云存储、云数据库产品。如果您希望针对这些产品中的数据进行分析和 搜索,可以通过 数据集成(Data Integration)来实现最快 5min 一次的离线数据,同步 到Elasticsearch的需求。

支持数据源

- · 阿里云云数据库(MySQL、PG、SQL Server、PPAS、MongoDB、HBase)
- ・阿里云DRDS
- ・阿里云MaxCompute(ODPS)
- 阿里云OSS
- 阿里云Table Store
- · 自建HDFS、Oracle、FTP、DB2,及上述云数据库的自建版本



做数据同步时可能会产生公网流量费用,请您知晓。

操作步骤

完成离线数据导入,需要您完成以下几步操作:

- · 您需要有一台可以与 VPC 内的 Elasticsearch 交互的ECS,这台 ECS 将获取数据源数据并执行写ES数据的"Job"(该任务将由"Data Integration系统统一下发")。
- · 您需要开通 Data Intergration 的服务,并且将ECS作为一个可以执行 Job 的"资源"注册到 Data Intergration 的服务中去。
- ·您需要配置一个"数据同步的脚本",并且让其可以周期性的执行起来。

详细步骤

 购买一台与Elasticsearch服务处于同一个VPC内的ECS服务器,并分配一个公网IP合或开通弹 性IP,为了节省您的成本,您可以复用已有的ECS服务器,(如何购买ECS,请参考创建ECS实 例文档)。

📕 说明:

- ·建议使用 centos6、centos7 或者 aliyunos。
- · 如果您添加的 ECS 需要执行 MaxCompute 任务或者同步任务,需要检查当前 ECS 的 python 版本是否是 python2.6或2.7 的版本 (centos5 的版本为 2.4,其余 os 自带了 2.6 以上版本)。
- ・请确保 ECS 有公网 IP。
- 2. 前往Data Integration控制台,并进入工作区。

如果您已经开通过Data Integration 或者 DataWorks 产品,您将会看到如下页面:

Θ	管理控制台		搜索	Q 消息 ⁸⁷ 费用	工单备案企业	支持与服务 😕 🍹 简体	Þ文 🁰
		概览	工作空间列表 调度	资源列表 计算引擎	列表		
6 6	0						
8	DataWorks	数据集成・数据开发	·数据服务		9		
Ŧ	快速入口					DataV与DataWorks 数据服务无缝对接	DataWorks
Ø	数据开发	数据集成	运维中心		数据服务	XX3610X737042X33X	~ 4>
? ©	工作空间				全部工作空间	Stream Studio	开启邀测
0	Bulk_Datasource 华	BulkTest	华东2	dla_project	华东1	and the factor second	
○ ■	创建时间:2018-12-28 15:03:49 计算引擎:MaxCompute 服务模块数据开发 数据集成 数据管理 数据服务 运线	创建时间:2018-12-28 15: 计算引擎:MaxCompute 服务模块数据开发数据集	10:26 或数据管理数据服务 运维	创建时间:2019-04-23 16 计算引擎:无 服务模块数据开发数据集	:47:31 5成 数据管理 数据服务 运堆	常见问题 如何生成AccessKey并绑定 如何创建成员以及赋权	● 咨 询 · 建 议
ଷ୍	工作空间配置 进入数据开发	工作空间配置	进入数据开发	工作空间配置	进入数据开发	如何进行本地数据上传下载	
0	进入数据服务 进入数据集成	进入数据服务	进入数据集成	进入数据服务	进入数据集成	大数据服务计费说明	
N	常用功能					数操集成产品计费说明	
×	☞ 创建工作空间 × 一键CDN						

如果您未开通过Data Integration 或者 DataWorks 产品,您需要按照步骤进行 Data Integration 的开通,此开通动作 会产生费用,请您按照费用提示进行预算评估。

← -	→ C 🔒 安全 http	s://workbench.data.aliyun.com/console?spm=517	6.7944453.751670.btn2.51074560hFVnl	B0#/	☆ 📀
(-)	管理控制台	产品与服务 ▼	Q、搜索 单 🤒 费用	工单 备案 企业	支持 简体中文
			概览 项目列表	调度咨询列表	
•	云计算基础服务		佩见 项目为权	则反风加小小社	
-	大数据(数加)				
•	数加控制台概览	🜀 DataWorks 数据集	成 ・数据开发 ・ MaxComput	te	
0	DataWorks			E	
۲	Quick BI	新用户引导 🕢 实名认证	✓ 创建AccessKey ──	十算资源 ——— 4 创建项目	产品升级功能列表
E.	机器学习				不能错过的新功能,新体验!
N	大数据计算服务	* 选择区域:	○ 华东2		
ŗ,	智能语音交互	* 付费方式:	○ CU预付费 ○ I/O后付费		多种开发模式,支持更多的数据通道
00	数据集成	* 项目名称:	字母或下划线开头,只能包含字母下划线和数字	Ζ	
co	阿里云Elasticsearch	日二々。	初田安培 副计指示日本		常见问题
•	安全(云盾)	亚小石:	如汞个块,氯认为项目有		如何生成AccessKey并绑定
•	域名与网站(万网)	项目描述:			如何进行本地数据上传下载
•	云市场			1	如何使用系统参数与自定义参数
_			确定		大数据服务计费说明
_					

3. 进入Data Integration的项目管理-调度资源管理页面,将您之前VPC内的ECS配置成为一个调 度资源。详细配置参考新增任务资源。

资源组管理 输入调度资	源名称进行搜索				新增资源组
资源组名称	网络类型	服务器	已使用DMU	付费类型	攝作
默认资源组	-			按量付费	

- 4. 在Data Integration中配置数据同步脚本,具体配置请参考脚本模式配
 - 置。Elasticsearch的Config规则请参考 配置Elasticsearch Writer。

$\leftarrow \rightarrow C$ e 安全 htt	tps://di-cn-shanghai.data.aliyun.c		☆ 2011 🐼 🕸 🛛
C DataWorks	数据写ES测试 -	数据集成 数据开发 数据管理 运维中心 项目管理 机器学习平台	· · · 中文 ▼
● Dataworks ■ ■ ■ ● <t< th=""><th> ① 注 ご ② ● 型 数据同步 ● @ cdp_es_test 我锁定 2017- </th><th>Xda 元 2 Xda 元 2 Xda 元 2 Xda 日 2 24単中の 切目 単</th><th> ↓ ↓ ↓ ↓ ⊙ = ₽</th></t<>	 ① 注 ご ② ● 型 数据同步 ● @ cdp_es_test 我锁定 2017- 	Xda 元 2 Xda 元 2 Xda 元 2 Xda 日 2 24単中の 切目 単	↓ ↓ ↓ ↓ ⊙ = ₽
	l	36 "record": "0" 37 },	

📋 说明:

- ·同步脚本的配置分为三个部分,Reader是配置您上游数据源(待同步数据的云产品)的 config,Writer是配置ES的config,还有一个setting是配置同步中的一些丢包和最大并发 等设置
- · ES Writer中accessId和accessKey需要配置您的Elasticsearch的访问"用户名"和"密 码"

5. 脚本配置完成后,将数据同步Job进行提交,按照需求填写相应的周期性执行配置,并点击"确 定"完成提交动作。

G	DataWorks	数据写ES测试	÷	数据集成	数据开发	数据管理	运维中心	项目管理	机器学习平台	elasticsea 👻 中文 🗸
		Q 🖻 :	提交						×	: ₪
-	离线同步	∨ 🛅 数据同步				周期属性			→ 提交	
8≣	同步任务	● 🕢 cdp_es_test 我锁		* 调度	类型: 💿 周期i	周度 🔵 一次性调	度			3
8	数据源			* 自动:	重跑: 自动	重跑 ⑦				
• n	客户端数据采集			* 生效	日期: 1970-0	01-01 – 2116-	11-18			
u				* 调度	周期: 分钟	小时	天周	月		
				* 起始	时间: 00:00	C)			
						(依赖属性		1		
				* 添加	依赖: 数据写图	ES测试 丶	请选择上游任务	5 ~	.yuncs.com:92	00",
				项目:	名称	任务名利	R	操作		
						没有依赖上游的	E务			
						跨周期依赖	0		-	
				● 不依	K赖上一调度周期		14+>=-<=			
					★●案, ●19 上一····································]期结束,才能继续	::xw=1」 [运行	\sim		
								确认	取消	
				日志						

说明:

- 如果您希望周期性调度,需要配置周期任务,具体的Job开始执行时间,周期间隔,Job生命 周期,均需要在这个弹窗中配置
- ·周期任务将于配置任务开始的第二天00:00,按照您的配置规则生效执行
- 6. 最后一步提交完成后,请务必前往运维中心-周期任务,找到您提交的Job,将其调度资源从默 认修改为您配置好的调度资源。

G	DataWorks	数据写ES测试 → 数据集成	戈 数据开发	数据管理	运维中心	项目管理 机器学习平台	elasti	csea ▾ 中文
	=	周期任务						
8	运维概览							
-	任务列表	节点任务 > 工作流名称或节点任务	名称 Q 任务类型:	全部任务	◇ 责任/	人 elasticsearchte ∨ ✓ 爻 我的任	务 今日修改的任务	冻结任务
6	周期任务	名称	修改日期↓	1	任务类型	责任人	调度类型 扌 操作	
ŵ	手动任务	C cdp_es_test	2017-11-2	10:53:11	数据同步	elasticsearchtest2pd	日调度 测试	补数据 更多 ▼
•	任务运维	project_etl_start	2017-11-1	8 14:02:45	虚节点	elasticsearchtest2pd	测试	冻结
6	周期实例							查看实例
B	手动实例							添加报警
	测试实例							修改责任人修改资源组
:ē	补数据实例							
•	报警							
¢	报警记录							
<i>L</i> ,	报警设置							

实时数据导入

功能开发中,敬请期待。

3.2 使用DataWorks实现Hadoop数据同步到阿里云Elasticsearch

本文向您详细介绍如何通过DataWorks数据同步功能,将Hadoop数据同步到阿里 云Elasticsearch上,并进行搜索分析。

背景信息

您也可以使用Java代码进行同步,具体请参考通过ES-Hadoop将Hadoop数据写入阿里 云Elasticsearch和在E-MapReduce中使用ES-Hadoop。

环境准备

1. 搭建Hadoop集群。

在进行数据同步前,您需要保证自己的Hadoop集群环境正常。本文使用阿里云EMR服务自动 化搭建Hadoop集群,详细过程请参见步骤三:创建集群。

EMR Hadoop的版本信息如下。

- ・ EMR版本: EMR-3.11.0
- ・集群类型: HADOOP
- · 软件信息: HDFS2.7.2 / YARN2.7.2 / Hive2.3.3 / Ganglia3.7.2 / Spark2.3.1 / HUE4.1.
 0 / Zeppelin0.8.0 / Tez0.9.1 / Sqoop1.4.7 / Pig0.14.0 / ApacheDS2.0.0 / Knox0.13.0

Hadoop集群使用VPC网络,区域为华东1(杭州),主实例组ECS计算资源配置公网及内网IP,高可用选择为否(非HA模式),具体配置如下图所示。

首页 〉 集群管理 〉 集群 (C-10) >	详情								
集群基础信息 当前集群: C-10	/ ES_to	est_hadoop					扩容 自动	续去管理 按量转包月	同步主机信息	释放
集群信息.										
ID: C-1C 地域: cn-hangzhou 开始时间: 2018-09-29 14:34:41			欽件配置: 10代化 量 海可用: 否 安全機式 标准		付誘共型: 按量付费 当前状态: 空闲 运行时间: 14分19秒			引导操作/软件配置: EMR- ECS应用角色: AliyunEmrEc	3.11.0 sDefaultRole	
软件信息						网络信息				
EMR版本: EMR-3.11.0 集群进型: HADOOP 软件信息: HDFS2.7.2 / YARN2.7.	EMR版本: EMR-3110 原則時型: HADGOP 叙州 信息: HOFS27.2 / HRN27.2 / Hive23.3 / Ganglia3.7.2 / Spark2.2.1 / HUE41.0 / Zeppelin0.7.3 / Tex0.9.1 / Sqoop1.4.6 / Pig0.14.0 / ApacheDS20.0 / Know0.13.0					区域ID: cn 网络类型: 安全组ID: 专有网络/:	-hangzhou-f vpc sg-bp 远後仍: vpc-bp			
主机信息		G	主实例组 🚭							
主实例组(MASTER)		按量付费	ECS ID	状态	:	2m	内网	创建时间		
主机款量: 1 内存: 8GB 数据盘配置: SSD云盘80GB*1块	CPU: 4核		ibp	●正常			192	2018-09-29	14:34:48	
核心实例组(CORE)	011.44	按量付费								
土tilaxæ: 2 内存: 8GB 数据盘配置: SSD云盘80GB*4块	UPU: 41%									

2. 购买和配置Elasticsearch。

登录Elasticsearch控制台,参考购买和配置,购买一个Elasticsearch实例。选择与EMR集群 相同的区域和VPC网络配置,如下图所示。

阿	里云Elastics	search (按量付	费)				
	预付费	后付费					
	地域	华东1	华北2	华东2	华南1	亚太南部 1 (孟买)	亚太东南 1 (新加坡)
讃		香港	美国西部 1 (硅谷)	亚太东南 3 (吉隆坡)			
-		检研印度					
	-2401-2						
	版本	5.5.3 with X-Pack	6.3 with X-Pack				
		当前版本为5.5.3					
	网络米刑	夫 方网络					
	网络关生	专有网络					
	专有网络	jing-vpc	•				
		创建VPC/子网(交换机)	, 创建完成后请刷新页	面			
1993 1993	虚拟交换机	jing-sw	•				
EU.							
	实例规格	1核2G	•				
		1核2G规格只适合测试	,不适用于生产环境,	不在SLA售后保障范围内	I		
	数量	3					
		两个节点集群有脑裂风	脸,谨慎选择				
	去古于共占						
	~ 비고 마						

3. 创建DataWorks工作空间。

创建DataWorks项目,区域选择华东1区。本文直接使用已经存在的项目bigdata_DOC。

	概览 项目列表	调度资源列表
🜀 DataWorks 🛛	y据集成・数据开发・MaxCompute	
快速入口		
数据开发	数据集成	运维中心
项目		全部项目
bigdata_DOC 华东1	MaxCompute_DOC 华东2	PAItest 华东2
创建时间:2018-09-02 10:26:59 计算引擎: MaxCompute 服务模块数据开发数据集成数据管理 运维中心	创建时间:2018-07-19 09:12:37 计算引擎:MaxCompute 服务模块数据开发数据集成数据管理运维中心	创建时间:2018-05-23 13-32-29 计算引擎:MaxCompute PA计算引擎 服务模块数据开发数据集成数据管理运维中心
项目配置 进入数据开发 进入数据集成	项目配置 进入数据开发 进入数据集成	项目配置 进入数据开发 进入数据集成
常用功能 ♀ 创建项目 × 一键CDN		

数据准备

在Hadoop集群中创建测试数据,步骤如下。

- 1. 进入EMR控制台界面,单击左侧菜单栏的交互式工作台。
- 2. 选择文件 > 新建交互式任务。

3. 在Notebook对话框中,输入交互式任务的名称,选择默认类型以及关联集群,单击确认。 本文新建一个名为es_test_hive的交互式任务,默认类型为Hive,关联集群为环境准备中创建的EMR Hadoop集群。

Notebook		\times
* 名称:	es_test_hive 长度限制为1-64个字符,只允许包含中文、字母、数字、-、_	
* 默认类型:	 ● Spark ● Spark SQL ● Hive 交互式任务中,在不指定任务类型的情况下,该交互式任务将会以默认的类型运行 	
关联集群:	ES_test_hadoop v	
	确认	取消

4. 在代码编辑区域中,输入Hive建表语句,单击运行。

本文档使用的建表语句如下。

CREATE TABLE IF NOT EXISTS hive_esdoc_good_sale(create_time timestamp, category STRING, brand STRING, buyer_id STRING, trans_num BIGINT, trans_amount DOUBLE, click_cnt BIGINT) PARTITIONED BY (pt string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' lines terminated by '\n'

表创建成功后,系统会提示Query executed successfully。

	大开式的第5条的1条件系与目前数,据试输用4006000,FullXT的分为100000。							
E-MapReduce管理控制台	EMR3.11.0之后的集群,请使用	数据开发->临时查询						
概览	交互式工作台							
集群	交互式任务列表	es_test_hive (HIVE) bafc38d8-5ba0-4b2d-bc0d-e922c2e27ac9						
交互式工作台	es_test_hive							
表管理	bank_data							
PENV	bigdata							
执行计划	hive							
数据开发 NEW	DOC1 doc	CREATE TABLE IF NOT EXISTS hive <u>esdoc_good_sale(</u> create_time timestamp, category TSENG:						
 报答 		brand STRING, buyer_id STRING,						
帮助		trans_num BIGINT, trans_amount DOUBLE,						
		click_cat BIGINT						
		/ PARTITIONED BY (pt string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' lines terminated by '\n'						
		▶运行						
		运行结果:						
		Query executed successfully. Affected rows : -1						
		状态: FINISHED , 运行 0秒 , 完成时间:Sep 29, 2018 4:30:37 PM						

5. 单击文件 > 新建段落,在段落编辑区域输入SQL语句,单击运行,插入测试数据。



您可以选择从OSS或其他数据源导入测试数据,也可以手动插入少量的测试数据。本文使用手动 插入数据的方法,脚本如下。

insert into hive_esdoc_good_sale PARTITION(pt =1) values('2018-08-21','外套','品 牌A','lilei',3,500.6,7),('2018-08-22','生鲜','品牌B','lilei',1,303 ,8),('2018-08-22','外套','品牌C','hanmeimei',2,510,2),(2018-08-22 ,'卫浴','品牌A','hanmeimei',1,442.5,1),('2018-08-22','生鲜','品牌D',' hanmeimei',2,234,3),('2018-08-23','外套','品牌B','jimmy',9,2000,7),(' 2018-08-23','生鲜','品牌A','jimmy',5,45.1,5),('2018-08-23','外套','品 牌E','jimmy',5,100.2,4),('2018-08-24','生鲜','品牌G','peiqi',10,5560 ,7),('2018-08-24','卫浴','品牌F','peiqi',1,445.6,2),('2018-08-24','外

套','品牌A','ray',3,777,3),('2018-08-24','卫浴','品牌G','ray',3,122,3),('2018-08-24','外套','品牌C','ray',1,62,7);

6. 使用同样的方式新建段落,并在段落编辑区域输入select * from hive_esdoc
 _good_sale where pt =1;语句,单击运行。

此操作可以检查Hadoop集群表中是否已存在数据可用于同步,运行成功结果如下。

						■ 保存段)	告 - 隐藏结果 × 册
> select * from hive	_esdoc_good_sale where	pt =1;					
b)= (=							
▶ 运行							
室行结果:							
hive_esdoc_good_sale.c reate_time	hive_esdoc_good_sale.c ategory	hive_esdoc_good_sale.b rand	hive_esdoc_good_sale.b uyer_id	hive_esdoc_good_sale.tr ans_num	hive_esdoc_good_sale.tr ans_amount	hive_esdoc_good_sale.cl ick_cnt	hive_esdoc_good_sale.p
2018-08-21 00:00:00.0	外赛	品牌A	lilei	3	500.6	7	1
2018-08-22 00:00:00.0	生鮮	品牌B	lilei	1	303.0	8	1
2018-08-22 00:00:00.0	外赛	品牌C	hanmeimei	2	510.0	2	1
null	卫浴	品牌A	hanmeimei	1	442.5	1	1
2018-08-22 00:00:00.0	生鮮	品牌D	hanmeimei	2	234.0	3	1
2018-08-23 00:00:00.0	外赛	品牌B	jimmy	9	2000.0	7	1
2018-08-23 00:00:00.0	生鮮	品牌A	jimmy	5	45.1	5	1
2018-08-23 00:00:00.0	外赛	品牌E	jimmy	5	100.2	4	1
2018-08-24 00:00:00.0	生鮮	品牌G	peiqi	10	5560.0	7	1
2018-08-24 00:00:00.0	卫浴	品牌F	peiqi	1	445.6	2	1
2018-08-24 00:00:00.0	外套	品牌A	ray	3	777.0	3	1
	TINS	忌聴ら	rav	3	122.0	3	1

数据同步



由于DataWorks项目所处的网络环境与Hadoop集群中的数据节点(Data Node)网络通常 不可达,因此您可以通过自定义资源组的方式,将DataWorks的同步任务运行在Hadoop集群 的Master节点上(Hadoop集群内Master节点和数据节点通常可达)。

- 1. 查看Hadoop集群的数据节点。
 - a) 在EMR控制台上,单击左侧菜单栏的集群。
 - b) 选择您的集群,单击右侧的管理。
 - c) 在集群管理控制台上, 单击左侧菜单栏的主机列表, 查看集群master节点和数据节点信息。

👯 E-MapReduce	集群管理 数据开发 告留的	性护 操作日志 帮助						
ES_test_hadoop 🖙	前页 〉 集群管理 〉 集群 (C-1C) > 主机列表						
88 集群基础信息	主机列表 当前黄酥: C-1C	/ ES_test_hadoop						同步主机供息
◎ 集群与服务管理			内阁印	外网IP	查询			
⑥ 主机列表	ECS ID	主机名	IP度思	角色 🏹	所屬机器組	付壽类型	规格	受(期月11日)
 ▶ 東新都本 № 访问触报与第日 △ 用户管理 	і-бр	emr-header-1	内障:192 204 外局:	MASTER	MASTER	按量付费	CPU-4 核 内存 8.6 ECS 规格 eca.n4.xlarge 数振曲配置 550 元曲 80 X 1块 系统曲配置 550 元曲 120 X 1块	
〇 3单位的名称	ibp	emr-worker-2	内际:192 206	CORE	CORE	按量付费	CPU-4 核 1 内存-86 ECS 原格ecs.n4.xlarge 数据曲配置-550元曲 1 80 X 4块 系统曲配置-550元曲 1 80 X 1块	
	iðp	emr-worker-1	内际192 205	CORE	CORE	按量付费	CPU-4 核1内存9G ECS 現俗 ecs.n4.xiarge 数据曲配置 SSD 元曲180 X 4块 系统曲配置 SSD 元曲180 X 1块	

📋 说明:

通常非HA模式的EMR上Hadoop集群的Master节点主机名为emr-header-1, Data Node主机名为emr-worker-X。

d) 单击上图中Master节点的ECS ID, 进入ECS实例详情页。单击远程连接进入ECS服务

器, 通过hadoop dfsadmin -report命令查看数据节点信息。

DFS Remaining: 665931456512 (620.20 GB) DFS Used: 209780736 (200.06 MB) DFS Used:: 0.03% Under replicated blocks: 0 Blocks with corrupt replicas: 0 Missing blocks: 0 Missing blocks (with replication factor 1): 0 Live datanodes (2): Name: 192. 206:50010 (emr-worker-2.cluster-77026) Hostname: emr-worker-2.cluster-77026 Decommission Status : Normal Configured Capacity: 333373341696 (310.48 GB) DFS Used: 104890368 (100.03 MB) Non DFS Used: 302723072 (288.70 MB) DFS Remaining: 332965728256 (310.10 GB) DFS Used:: 0.03/ DFS Remaining%: 99.88% Configured Cache Capacity: 0 (0 B) Cache Used: 0 (0 B) Cache Remaining: 0 (0 B) Cache Used: 100.00: Cache Remaining%: 0.00% Xceivers: 1 Last contact: Sat Sep 29 17:37:46 CST 2018 Name: 192. 205:50010 (emr-worker-1.cluster-77026) Hostname: emr-worker-1.cluster-77026 Decommission Status : Normal Configured Capacity: 333373341696 (310.48 GB) DFS Used: 104890368 (100.03 MB) Non DFS Used: 302723072 (288.70 MB) DFS Remaining: 332965728256 (310.10 GB) DFS Used:: 0.03/ DFS Remaining%: 99.88% Configured Cache Capacity: 0 (0 B) Cache Used: 0 (0 B) Cache Remaining: 0 (0 B) Cache Used%: 100.00% Cache Remaining%: 0.00% Xceivers: 1 Last contact: Sat Sep 29 17:37:46 CST 2018

- 2. 新建自定义资源组。
 - a) 进入DataWorks的数据集成页面,选择资源组 > 新增资源组。

DataWorks	bigdata_DOC 🚽 🚦	数据集成 数据开发	数据管理 运输中心	项目管理			• 中文•
= → 数据集成概范	资源组管理 输入调度资源名称	进行搜索					新潮资源组
	资源坦名称	网络类型		服务器	已使用DMU	付總逆型	操作
→ 项目空间	數认资源组					按量付器	
→ 項目空间概況	hdfe	专有网络			0	按量付票	服务器初始化 管理 删除
▼ 高线同步							
8 ≈***							
() X3538							
- 同步资源管理							
品 密源组							

关于自定义资源组的详细信息请参见新增任务资源。

b) 根据界面提示,输入资源组名称和服务器信息。此服务器为您EMR集群的Master节点,服 务器信息说明如下。

新增资源组				×
	2		+0	
的建筑标组		scazAgeni	154	
* 网络类型: 服务器1	🥑 专有网络 💕			
* ECS UUID :	请输入UUID,非服	务器名称	0	
* 机器IP:	请输入内网机器IP		0	
∗ 机器CPU(核):				
∗ 机器内存(GB):				
添加服务器				
			上一步	下一步

参数	说明
网络类型	选择专有网络。
	 注意: 对于专有网络类型,需输入服务器UUID。对于经典网络类型,需输入服务器名称。目前仅DataWorks V2.0华东2区支持经典网络类型的调度资源添加,对于其他区域,无论您使用的是经典网络还是专有网络类型,在添加调度资源组时都请选择专有网络类型。
ECS UUID	登录EMR集群的Master节点,执行 dmidecode grep UUID,取返回值。
机器IP/机器CPU(核)/机 器内存(GB)	您Master节点的公网IP/CPU/内存。您可以在Master节点的ECS控制台上单击实例名称,在配置信息模块,找到相关信息。

! 注意:

完成添加服务器后,您需要保证Master Node与DataWorks网络可达。

- · 如果您使用的是ECS服务器, 需设置服务器的安全组。
- ·如果您使用的内网IP互通,需要添加安全组。
- ·如果您使用的是公网IP,可直接设置安全组公网出入方向规则。

由于本文档的EMR集群使用的是VPC网络,且与DataWorks在同一区域下,因此不需要进行安全组设置。

c) 按照提示安装自定义资源组Agent。

注意: 由于本文使用的是VPC网络类型,因此不需开通8000端口。

观察到当前状态为可用时,说明新增自定义资源组成功。如果状态为不可用,您可以登录Master Node,使用tail -f/home/admin/alisatasknode/logs/heartbeat. log命令查看DataWorks与Master Node之间心跳报文是否超时。

[root@emr-header-1 logs]# hdfs di	fs -ls /user/hive/warehouse/hive_doc_good_sale/
Found 1 items	
drwxr-xx - hive hadoop	0 2018-09-03 17:46 /user/hive/warehouse/hive_doc_good_sale/pt=1
[root@emr-header-1 logs]# tail -	f /home/admin/alisatasknode/logs/heartbeat.log
2018-09-06 21:47:34,440 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:104] [] - heartbeat start, current status:2
2018-09-06 21:47:34,465 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:133] [] - heartbeat end∎ cost time:0.025s
2018-09-06 21:47:39,465 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:104] [] - heartbeat start, current status:2
2018-09-06 21:47:39,491 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:133] [] - heartbeat end∎ cost time:0.026s
2018-09-06 21:47:44,491 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:104] [] - heartbeat start, current status:2
2018-09-06 21:47:44,515 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:133] [] - heartbeat end∎ cost time:0.024s
2018-09-06 21:47:49,516 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:104] [] - heartbeat start, current status:2
2018-09-06 21:47:49,538 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:133] [] - heartbeat end∎ cost time:0.022s
2018-09-06 21:47:54,539 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:104] [] - heartbeat start, current status:2
2018-09-06 21:47:54,555 INFO [pd	ool-6-thread-1] [HeartbeatReporter.java:133] [] - heartbeat end∎ cost time:0.016s

3. 新建数据源。

a) 在DataWorks的数据集成页面,单击数据源 > 新增数据源,在弹框中选择HDFS类型的数据 源。

DataWorks	bigdata_DOC -	数据集成 数据开发	数据管理	运维中心	项目管理					• 中文•
- - 数据集成概范	救援源典型: 全部	✓ 数据源名称:	9/178	数据源				×		新建数据源
2 资源消耗监控	数据源名称	数据源类型	19 198	東型数据库	8	m			数据源描述	操作
▼ 項目空间	odps_first	ODPS	OD OD	MySQL	SQL Server	PostgroSQL	ORACLE [,]	(2) IZB	connection from odpa calc engine 1144	
→ 項目空间概法			Ao	MySQL	SQL Server	PostgreSQL	Oracle	DM		
2 资源消耗监控	HDEST	HDFS	det	8						544g milet
▼ 案线同步				DRDS						く 正一页 1 下一页 >
8 ####			17	大数描存储						
() #380#				\sim	\diamond	×				
- 同步资源管理			М	axCompute (ODPS)	AnalyticDB (ADS)	Datahub				
an 1920 an			14	科结构化存储						
				055	HDFS	D				
			I N	1oSQL						
				MongoDB	Memcache (OCS)	Redis	Table Store (OTS)			
			19	LogHub						
								R 56		

b) 在新增HDFS数据源页面中,填写数据源名称和defaultFS。

编辑HDFS数据源		×
* 数据源名称	HDFS_data_source	
数据源描述	Elasticsearch 测试	
* defaultFS :	hdfs://47. 100:9000	?
测试连通性	测试连通性	
	完成	取消

! 注意:

对于EMR Hadoop集群而言,如果Hadoop集群为非HA集群,则此处地址为hdfs:// emr-header-1的IP:9000。如果Hadoop集群为HA集群,则此处地址为hdfs://emrheader-1的IP:8020。在本文中,emr-header-1与DataWorks通过VPC网络连接,因 此此处填写内网IP,且不支持连通性测试。

4. 配置数据同步任务。

- a) 在DataWorks数据集成页面,单击左侧菜单栏的同步任务,选择新建 > 脚本模式。
- b) 在导入模板对话框中,选择数据源类型如下,单击确认。

导入模板			\times
* 来源类型:	Hdfs	~ ?	
* 数据源 :	HDFS_data_source (hdfs) 新增数据源	\sim	
* 目标类型:	Elasticsearch	~ ?	
		确认	取消

c) 完成导入模板后,同步任务会转入脚本模式,本文中配置脚本如下,相关解释请参见脚本模式配置, Elasticsearch的配置规则请参考配置Elasticsearch Writer。

1 -	{	2 Hdfe Deeder
2 -	"configuration": {	Mais Reader
3 -	"reader": {	
4	"plugin": "hdfs",	
5 -	"parameter": {	
6	"path": "/user/hive/warehouse/hive_esdoc_good_sale/",	
/	"datasource": "HUFS_data_source",	
8 T	"COLUMN": [
10	l "index": 0	
10	"type": "string"	
12	Lype , Stilling	
13 -	د ا ک	
14	"index": 1.	
15	"type": "string"	
16	},	
17 -	{	
18	"index": 2,	
19	"type": "string"	
20	},	
21 -	{	
22	"index": 3,	
23	"type": "string"	
24	},	
25 -	{	
26	"index": 4,	
27	"type": "long"	
28		
29 *	l Vindev‼: E	
31	Index : D, "type": "double"	
32	Cype : double	
33 -	1) {	
34	"index": 6.	
35	"type": "long"	
36	}	
37],	
38	"defaultFS": "hdfs:// 9000",	
39	"fieldDelimiter": ",",	
40	"encoding": "UTF-8",	
41	"fileType": "text"	
42	}	
43	},	
44 -	"writer": {	
45	"plugin": "elasticsearch",	
46 -	"parameter": {	
47	accessia : ,	com: 0200"
40	"indevType": "elasticsearch"	.com.5200 ,
50	"accessKev": "	
51	"cleanup": true.	
52	"discoverv": false.	
53 -	"column": [
54 -	{	
55	"name": "create_time",	
56	"type": "string"	
57	},	
58 -		
59	"name": "category",	
60	"type": "string"	
61	},	
62 -		
63	"name": "brand", "two-", "stairs"	
64	"type": "string"	
66 -		
67	l "name": "buver id"	
68	"type": "string"	
69	lype i bering	
70 -	4	
71	"name": "trans num".	
72	"type": "long"	
73	},	
74 -	{	
75	"name": "trans_amount",	
76	"type": "double"	
77		
文档版本: 2019780	5	25
79	"name": "click_cnt",	20
80	"type": "long"	
81	}	
82		

- · 同步脚本的配置分为三个部分, Reader用来配置您上游数据源(待同步数据的云产 品)的config, Writer用来配置 Elasticsearch的config, setting用来配置同步中的一 些丢包和最大并发等。
- *path*为数据在Hadoop集群中存放的位置,您可以在登录master node后,使用hdfs
 dfs -ls /user/hive/warehouse/hive_esdoc_good_sale命令确认。对于分区
 表,您可以不指定分区,DataWorks数据同步会自动递归到分区路径。
- ・由于Elasticsearch不支持timestamp类型,本文档将creat_time字段的类型设置 为string。
- endpoint为Elasticsearch的内网或外网地址。如果您使用的是内网地址,请
 在Elasticsearch的集群配置页面,配置Elasticsearch的系统白名单。如果您是用的是
 外网地址,请在Elasticsearch的网络配置页面,配置Elasticsearch的公网地址访问白
 名单(包括DataWorks服务器的IP地址和您所使用的资源组的IP地址)。
- Elasticsearch Writer中accessId和accessKey需要配置您的Elasticsearch的访问用
 户名(默认为elastic)和密码。
- · index为Elasticsearch实例的索引,您需要使用该索引名称访问Elasticsearch的数据。
- ・ 在创建同步任务时, DataWorks的默认配置脚本中, errorLimit的record字段值 为0, 您需要将其修改为大一些的数值,比如1000。
- d) 完成配置后,单击页面右侧的配置任务资源组,选择您创建的资源组名称,完成后单击运行。

```
如果提示任务运行成功,则说明同步任务已完成。如果运行失败,可通过复制日志进行进一
步排查。
```

结果验证

- 进入Elasticsearch控制台,单击实例名称 > 可视化控制,在Kibana区域中,单击右下角的进入控制台。
- 2. 输入用户名和密码,单击登录进入Kibana控制台,选择Dev Tools。
- 3. 在Console控制台中,执行如下命令,查看已经同步过来的数据。

```
POST /hive_doc_esgood_sale/_search?pretty
{
"query": { "match_all": {}}
```

}

hive_doc_esgood_sale为您同步数据时,设置的index字段的值。

Libore	Dev Tools
kidana 🔪	Console Search Profiler Grok Debugger
Ø Discover	1 POST /hive_doc_esgood_sale/_search?pretty > 1 - {
Visualize	2 * { 3 "query": { "match_all": {}} 3 "timed_out": false,
	4 * } 4 * "_shards": { 5 "total": 5
Oashboard	6 "successful": 5,
😨 Timelion	7
Machine Learning	9• "hits": { 10
	11 "max_score": 1,
🛠 Graph	12 * ThIS: [13 * {
🔎 Dev Tools	14 "_index": "hive_doc_esgood_sale", 15 "_type": "elasticsearch"
	16 "_id': "AWZ2421uvdLQQ0x23xYB",
• Monitoring	17
🄅 Management	19 "create_time": "2018-08-23 00:00:00", ""
	20 trans_num : 5, 21 "click cnt": 5,
	22
	23 "buyer_id": "jimmy",
	24 (trans_mount: 45.1, 25 "brand": "Brand"
	: 26 ^ }
	27^ },
	29mex. nive_uor_esgoou_sate,
	31 "id": "AWZ242luvdLQQ0x23xYF",
	32 "_score": 1,
	33 → "_source": {
	34 Create_lime: 2018-08-24 00:00:00 , 35 "trans num": 3
	36 "click cnt": 3,
	37 "category": "外套",
	38 "buyer_id": "ray",
	40 "brand": "R@A"
	41 + }
	42 * },
	43* { 44 "index"· "hive dor econod cale"
	45 "_type": "elasticsearch",
	46 "_id": "AWZ242luvdLQQ0x23xYK",
🚨 elastic	47 <u>"_scope": 1,</u>
and the second	40 ⁴
Logout	50 "category": "外套",
 Collapse 	51

数据搜索与分析

1. 在Console控制台中,执行如下命令,返回品牌为A的所有文档。

```
POST /hive_doc_esgood_sale/_search?pretty
{
    "query": { "match_phrase": { "brand":"品牌A" } }
```

٦		
}		
	7	Dev Tools
	kibana	
		Console Search Profiler Grok Debugger
Ø		1 POST /hive_doc_esgood_sale/_search?pretty > / 1 - {
la I		2 ~ 1 2 ~ "took": 16, 3 3 "timed out": false,
		4 "query": { "match_phrase": { "brand":"品牌A" } } 4 - "_shards": { "_shards": {
\odot		5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
8		7 "failed": 0
		9 • "hits": {
КÀ	Machine Learning	10 "total": 8, 11 "max_score": 1.5866871,
网络		12 * "hits": [
بو	Dev Tools	13) ¹ 14 "_index": "hive_doc_esgood_sale",
· ·		15 <u>"_type": "elasticsearch",</u> 16 id": <u>'AM27421uud1008x37x7</u> .
0		17 "_score": 1.5866871,
•	Management	18 *
		20 "trans_num": 3,
		21 CITCK_ETT: - /, 22 "category": "外套",
		23 "buyer_id": "lilei",
		24 「Trans_amount: 590.6, 25 『Prand": 「兄娘A"
		: 26 + }
		29 'index": "hive_doc_esgood_sale",
		30 "_type": "elasticsearch",
		3110 : AW22421UV01UQ00x23XA-, 32 " score": 0, 7954041.
		33 • "_source": {
		34 "create_time": "\\N",
		36 "click_cnt": 1,
		37 "category":"卫浴",
		38 DUYE_10: narmeimei, 39 "trans amount": 442.5.
		40 "brand": "品牌A"
		41 - 3
		43 - {
		44 "_index": "hive_doc_esgood_sale",
		45Ype": "elasticsearch", 46 "id": "AWZ42Uvd100x23xYT".
		47 "_score": 0.7954041,
		48 - "_source": { 49 - "_create time": "2018-09-21 20:00:00"
÷		50 "category": "外套",
0		51

2. 在Console控制台中,执行如下命令,按照点击次数进行排序,判断各品牌产品的热度。

```
POST /hive_doc_esgood_sale/_search?pretty
{
  "query": { "match_all": {} },
  "sort": { "click_cnt": { "order": "desc" } },
  "_source": ["category", "brand","click_cnt"]
```

}			
	kibana	Dev Tools Console Search Profiler Grok Debugger	
	Discover Visualize Dashboard Timelion Machine Learning Graph Dev Tools Monitoring Management	<pre>POST /hive_doc_esgood_sale/_search?pretty 2 { 3 "query": { "match_all": {} }, 4 "sort": { "click_cnt": { "orden": "desc" } }, 5 "_source": ["category", "brand", "click_cnt"] 6 * } </pre>	<pre>1 - { 2 "tock": 11, 3 "timed_out": false, 4 "_shards": { 5</pre>
. ⊡			23 · "sort": [24 · 8 25 ·] 26 ·

更多命令和访问方式,请参见阿里云Elasticsearch官方文档和Elastic.co官方帮助中心。

3.3 同步 MySQL 数据库到 Elasticsearch 中并进行搜索分析

阿里云上拥有丰富的云存储、云数据库产品。如果您希望针对这些产品中的数据进行分析和搜 索,可以通过DataWorks的数据集成服务,将离线数据同步到Elasticsearch中,最快可达到5分 钟一次。



做数据同步时可能会产生公网流量费用,请您知晓。

准备工作

完成离线数据的分析与搜索,需要您完成以下几步操作:

・创建一个数据库,您可以选择使用阿里云的RDS数据库,也可以在本地服务器上自建数据库。本 文档以RDS MySQL数据库为例,数据库字段及数据如下图所示。

create_time	▼ category	/ 🔻 brand 🔻	buyer_id	▼ trans_num ▼	trans_amount 💌	click_cnt 🔻
2018-08-21 00:00:00	外	品	1	3	500.6	7
2018-08-22 00:00:00	生	品	1	1	303	8
2018-08-22 00:00:00	ቃኑ	品	h	2	510	2
1970-01-01 08:00:00	P	品	h	1	442.5	1
2018-08-22 00:00:00	生	品	h	2	234	3
2018-08-23 00:00:00	ቃኑ	品	j	9	2000	7
2018-08-23 00:00:00	生	品	j	5	45.1	5
2018-08-23 00:00:00	外	品	j	5	100.2	4
2018-08-24 00:00:00	生	品	р	10	5560	7
2018-08-24 00:00:00	P	品	р	1	445.6	2
2018-08-24 00:00:00	外	品	r	3	777	3
2018-08-24 00:00:00	P	品	r	3	122	3
2018-08-24 00:00:00	外	品	r	1	62	7

- ·购买一台可以与VPC内的Elasticsearch交互的ECS,这台ECS将获取数据源数据并执行写 Elasticsearch数据的任务(该任务将由数据集成系统统一下发)。
- · 开通DataWorks的数据集成服务,并且将ECS作为一个可以执行任务的资源,注册到数据集成 服务中去。
- · 配置一个数据同步的脚本,并且让其可以周期性的执行起来。
- · 创建一个Elasticsearch实例,用来存储数据集成系统同步成功的数据。

操作步骤

数据同步

- **1.** #unique_32_°
- 2. 进入Elasticsearch控制台,单击创建,创建一个Elasticsearch实例。





地域、专有网络、虚拟交换机与您第一步中创建的专有网络保持一致。

3. 购买一台与Elasticsearch服务处于同一个VPC内的ECS服务器,并分配一个公网IP合或开通弹 性IP,为了节省您的成本,您可以复用已有的ECS服务器。



- ・建议使用 centos6、centos7 或者 aliyunos。
- · 如果您添加的 ECS 需要执行 MaxCompute 任务或者同步任务,需要检查当前 ECS 的 python 版本是否是 python2.6或2.7 的版本 (centos5 的版本为 2.4,其余 os 自带了 2.6 以上版本)。
- ・ 请确保 ECS 有公网 IP。

4. 进入DataWorks 控制台,并进入工作区。

·如果您已经开通过DataWorks数据集成产品,您将会看到如下页面:

	概览 项目列表 调度资	源列表
DataWorks	数据集成・数据开发・MaxCompute	
快速入口	数探集成	运練中心
项目		全部项目
bigdata_DOC	华东1 PAltest 华东2	MaxCompute_DOC 华东2
 ・(18:09-02-10-22:5-59) ・ ・ 日前第:1-MacCompate 服务模块数据开发数层集成数层管理运用中心 項目範置 進入数据开设 進入数据集成 	 ・⑪建学師:2018-05-23 13:32:29 ・ ・ ・	 ・101807-19-09-12-37 ・ ・ ・
常用功能		

·如果您未开通过DataWorks数据集成产品,您将会看到如下页面。您需要按照步骤开通数据 集成服务,此开通动作会产生费用,请您按照费用提示进行预算评估。

\leftarrow	← → C 🗎 安全 https://workbench.data.aliyun.com/console?spm=5176.7944453.751670.btn2.51074560hFVnB0#/ 😒 📀							
C-)	管理控制台	产品与服务 ▼ Q 搜索 ▲ 199 费用 工单	备案 企业 支持 简体中文					
	Ш	概览 项目列表 调度	资源列表					
→ ÷	云计算基础服务							
• ;	大数据(数加)							
•	数加控制台概览	DataWorks 数据集成・数据开发・MaxCompute						
0	DataWorks	[][]						
\$	Quick Bl	新用户引导 🕢 实名认证 ————————————————————————————————————	④ 创建项目 产品升级功能列表					
厚	机器学习		不能错过的新功能,新体验!					
м	大数据计算服务	* 选择区域: 〇 华东2	数据集成正式上线					
,	智能语音交互	* 付费方式: 〇 CU预付费 〇 I/O后付费	多种开发模式,支持更多的数据通道					
Co	数据集成	* 项目名称: 字母或下划线开头,只能包含字母下划线和数字						
60	阿里云Elasticsearch	日二次。如何不能言說「後」不同少	常见问题					
• 5	安全(云盾)	重小有: 如果不喝,MAA切口台	如何生成AccessKey并绑定 如何创建成员以及赋权					
• 3	或名与网站(万网)	项目描述:	如何进行本地数据上传下载					
•	云市场		如何使用系统参数与自定义参数					
		确定	大数据服务计费说明					

5. 单击DataWorks项目下方的进入数据集成。

6. 在数据集成页面,选择左侧导航栏中的资源组,单击新增资源组。
7. 根据界面提示,输入资源组名称和服务器信息。此服务器为您已经购买的ECS服务器,服务器信 息说明如下:

新增资源组				\times
1 创建资源组	2 不不不不不不不不不不不不不不不不不不不不不不不不不不不不不不不不不不不不	3 安装Agent	4 检查联通	
* 网络类型: 服务器1	💿 专有网络 🛛 👔			
* ECS UUID :	请输入UUID,非哪	务器名称	0	
* 机器IP:	请输入内网机器IP		0	
* 机器CPU(核):				
* 机器内存(GB):				
添加服务器				
			上—步 下—	步

- ・ ECS UUID: #unique_33 服务器,执行 dmidecode | grep UUID, 取返回值。
- ・机器 IP/机器CPU(核)/机器内存(GB): 您ECS实例的公网IP/CPU/内存。您可以 在ECS控制台上单击实例名称,在配置信息模块,找到相关信息。
- · 按照界面提示,完成安装Agent步骤。其中第五步为开通服务器的8000端口,可以跳过,保 持系统默认即可。
- 8. 配置数据库白名单,添加该资源组的IP地址和DataWorks服务器的IP地址,到您的数据库白名 单中。配置方法请参见添加白名单。
- 9. 资源组创建成功后,选择左侧导航栏的数据源,单击新增数据源。

编辑MySQL数据源		×
* 数据源类型	阿里云数据库 (RDS) ~	
* 数据源名称	es_test_rdsmysql	
数据源描述		
* RDS实例ID	rm-bp	?
* RDS实例主帐号ID	107	?
* 数据库名	xi	
* 用户名	xi	
* 密码		
测试连通性	测试连通性	
0	需要先添加RDS白名单才能连接成功, <mark>点我查看如何添加白名单。</mark> 确保数据库可以被网络访问	
	确保数据库没有被防火墙禁止 确保数据库域名能够被解析	
	确保数据库已经启动	
	完成	取消

10.单击MySQL,进入新增MySQL数据源页面,填入数据源信息,如下图所示。

数据源类型:本文档以阿里云数据库(RDS)为例,您也可以选择有公网IP和无公网IP。各配置项的详细信息请参见配置MySQL数据源。

11.选择左侧导航栏的同步任务,单击新建,选择脚本模式。

12.在导入模板对话框中,选择数据源类型 > MySQL,数据源为您第10步中新增的数据源名称,目标类型为Elasticsearch,完成后单击确认。

新建	一个同步任务:			
导入模板			×	<u></u>
* 来源类型:	MySQL	\sim ?)脚本模式
* 数据源:	es_test_rdsmysql (mysql) 新增数据源	\checkmark	JE	高效 _{度调优} , 支持全部数据源
* 目标类型:	Elasticsearch	~ ?	- 1	
		确认	取消	

13.配置数据同步脚本。具体配置请参考脚本模式配置,Elasticsearch 的配置规则请参考配 置Elasticsearch Writer。

	し新建	📑 导入模板	□ 保存	() 运行	① 停止	믬 格式化	
	1 • { 2 • "co 3 • " 4 5 • 6 7 • 8 9 10 11 12 13 14	nfiguration": { reader": { "plugin": "mysq "parameter": { "datasource": "column": ["create_tim "category", "brand", "buyer_id", "trans_amou "click cot"	l", "es_test_rdsr e", nt",	nysql",			
	15 16 17 18 19 20 } 21 - "], "where": "", "splitPk": "", "table": "good } , writer": { "plugin": "elast	, d_sale" ticsearch",				
	23 ▼ 24 25 26	"parameter": { "accessId": " "endpoint": " "indexType":	elastic", http://es-cn-	1 .		aliyuncs	.com:9200",
	20 27 28 29 30 - 31 - 32	"accessKey": "cleanup": fa "discovery": "column": [{ { "name": "	false,	,			
	33 34 35 ∓ 36 37	"type": " }, { "name": " "type": "	date" category", string"				
	39 - 40 41 42 43 -	}, { "name": " "type": " }, {	brand", string"				
	44 45 46 47 - 48 49	"name": " "type": " }, { "name": " "type": "	buyer_id", string" trans_num", long"				
	50 51 - 52 53 54	}, { "name": " "type": "(},	trans_amount" double"	,			
	56 57 58 59 60	' "name": " "type": " }]. "index": "tes	click_cnt", long" trds",				
	61 62 63 64 } 65 ~ "	<pre>"batchSize": "splitter": " } setting": { """""""""""""""""""""""""""""""""""</pre>	1000, ,"				
	66 ▼ 67 68 69 ▼ 70	"errorLimit": { "record": "0" }, "speed": { "throttle": file	alse.				
文档版本:	2071 90806 73 74 75	"concurrent": "mbps": "1", "dmu": 1 }	1,				37

📙 说明:

- · 同步脚本的配置分为三个部分, Reader用来配置您上游数据源(待同步数据的云产品)的 config, Writer用来配置 Elasticsearch的config, setting用来配置同步中的一些丢包和 最大并发等。
- endpoint为Elasticsearch的内网或外网地址,如果您使用的是内网地址,请
 在Elasticsearch的集群配置页面,配置Elasticsearch的系统白名单。如果您是用的是外
 网地址,请在Elasticsearch的网络配置页面,配置Elasticsearch的公网地址访问白名
 单(包括DataWorks服务器的IP地址和您所使用的资源组的IP地址)。
- Elasticsearch Writer中accessId和accessKey需要配置您的Elasticsearch的访问用户
 名(默认为elastic)和密码。
- · index为Elasticsearch实例的索引,您需要使用该索引名称访问Elasticsearch的数据。
- 14.同步脚本配置完成后,单击页面右侧的配置任务资源组,选择您第7步创建的资源组名称,完成 后单击运行,将MySQL中的数据同步到Elasticsearch中。

数据搜索分析

- 1. 进入Elasticsearch控制台,单击右上角的kibana控制台,选择Dev Tools。
- 2. 执行如下命令,查看已经同步过来的数据。

```
POST /testrds/_search?pretty
{
"guery": { "match_all": {}}
```

}

testrds为您同步数据时,设置的index字段的值。

	kibana	Dev Tools
	KIDalla	Console Search Profiler Grok Debugger
Ø	Discover	1 POST /testrds/_search?pretty
Ŀ	Visualize	2 1000: 2, 3 "timed_out": false, 4 "showt" ("motch all": ())
$^{\odot}$	Dashboard	4 query : { match_all : {}} 5 } 5 "total": 5, 6 "rureneful": 5
8	Timelion	7 "skipped": 0, 8 "failad": 0
ø	Machine Learning	9 + }, 10 - "hits": {
÷	APM	11 "total": 13, 12 "max score": 1,
<u>4</u>	Graph	13 • "hits": [14 • {
بر	Dev Tools	<pre>15 "_index": "testrds", 16 "_type": "elasticsearch",</pre>
<i>~</i>	Monitoring	17 "_id": "fVQJ@mUBNq0pXuST1IUW", 18 "_score": 1,
,	wontoning	19 * "_source": { 20 "create_time": "2018-08-22T00:00:00.000+08:00",
*	Management	21 "trans_num": 2, 22 "click_cnt": 2,
		23 "category" "//", 24 "buyer_id" "h",
		25 "trans_amount": 510, 26
		27 ~ } 28 ~ },
		29 - { 30 "index": "testrds".
		31 "_type": "elasticsearch",
		321 : TAUSMINGNQDAUSIIIW,
		34 * "source": {
		36 "trans.num": 2,
		37 "click cnt": 3,
		39 "buyer id": "h",
		40 "trans_amount": 234,
		4.1 "brand": "語" 4.2
		43 ^ },
_		44~ { 45 index". "testrds"
		46 "_type": "elasticsench",
2	elastic	47 "_id": "gVQJ@mUBNq0pXuST1IUW",
-a	Logout	401 49 ~
	Logout	50 create_time": "2018-08-23T00:00:00.000+08:00",

3. 执行如下命令,按照trans_num字段对文档进行排序。

```
POST /testrds/_search?pretty
{
    "query": { "match_all": {} },
    "sort": { "trans_num": { "order": "desc" } }
}
```

4. 执行如下命令, 搜索文档中的category和brand字段。

```
POST /testrds/_search?pretty
{
"query": { "match_all": {} },
"_source": ["category", "brand"]
}
```

5. 执行如下命令,搜索category为生的文档。

```
POST /testrds/_search?pretty
{
"query": { "match": {"category":"生"} }
```

```
}
```

```
Ł
 "took": 10,
  "timed_out": false,
  " shards": {
   "total": 5,
   "successful": 5,
   "skipped": 0,
   "failed": 0
  },
  "hits": {
   "total": 4,
    "max score": 0.6931472,
    "hits": [
     {
        "_index": "testrds",
       "_type": "elasticsearch",
       " id": "f1QJ0mUBNqOpXuST1IUW",
        " score": 0.6931472,
        "_source": {
        "create_time": "2018-08-22T00:00:00.000+08:00",
         "trans_num": 2,
          "click_cnt": 3
         "category": "生",
         "buyer_id": "h",
          "trans_amount": 234,
          "brand": "品"
        }
      },
      {
        " index": "testrds",
       " type": "elasticsearch",
        "_id": "gVQJ0mUBNqOpXuST1IUW",
       "_score": 0.6931472,
        ....
         source": {
         "create time": "2018-08-23T00:00:00.000+08:00",
          "trans_num": 5,
          "click_cnt": 5,
         "category": "生",
         "buyer_id": "j",
          "trans amount": 45.1,
         "brand": "品"
       }
      },
      {
        " index": "testrds",
         type": "elasticsearch",
        "_id": "g1QJ0mUBNqOpXuST1IUW",
        "_score": 0.6931472,
        " source": {
        "create time": "2018-08-24T00:00:00.000+08:00",
          "trans num": 10,
                                                          文档版本: 20190806
```

更多命令和访问方式,请参考ES访问测试和Elastic.co官方帮助中心。

常见问题

· 同步过程中出现无法连接数据库的相关错误。

解决方法:将您资源组中所使用的ECS服务器的内网IP和外网IP,都添加到您数据库的白名单中。

· 同步过程中无法连通Elasticsearch实例的相关错误。

解决方法:按照下面步骤进行排查。

- 检查在运行同步脚本之前,是否在页面右侧的配置任务资源组中选择了您前面步骤创建的资源组。
 - 是,执行下一步。
 - 否,单击页面右侧的配置任务资源组,选择您前面步骤创建的资源组。完成后单击运行。
- 2. 检查是否在Elasticsearch实例的白名单中,添加了DataWorks服务器的IP地址和您所使用 的资源组的IP地址。
 - 是,执行下一步。
 - 否,将DataWorks服务器的IP地址和您所使用的资源组的IP地址,添加到 Elasticsearch 实例的白名单中。

🗾 说明:

如果您使用的是内网地址,请在Elasticsearch的集群配置页面,配置Elasticsearch的 系统白名单。如果您是用的是外网地址,请在Elasticsearch的网络配置页面,配 置Elasticsearch的公网地址访问白名单(包括DataWorks服务器的IP地址和您所使用 的资源组的IP地址)。

 检查您的同步脚本配置是否正确。包括endpoint(您 Elasticsearch 实例的内网或外网 地址)、accessId(Elasticsearch 实例的访问用户名,默认为elastic)和accessKey(Elasticsearch实例的访问密码)。

3.4 RDS for MySQL与阿里云ES实时同步数据

数据传输服务 DTS (以下简称 DTS)支持RDS for MySQL与阿里云Elasticsearch实时同步 数据,通过 DTS 提供的 RDS for MySQL->阿里云Elasticsearch实时同步功能,可以将企业线 上RDS for MySQL中的生产数据实时同步到阿里云Elasticsearch中进行搜索。本小节介绍如 何使用 DTS 快速创建RDS for MySQL->阿里云Elasticsearch的实时同步作业,实现RDS for MySQL数据到阿里云Elasticsearch的实时同步。

支持实时同步类型

同一个阿里云账号下 RDS for MySQL->阿里云Elasticsearch实例。

支持SQL操作类型

主要支持的SQL操作类型如下:

- Insert
- · Delete
- Update

▋ 说明:

目前暂不支持 DDL同步,如果同步过程中遇到DDL操作,DTS会忽略掉。

如果后续遇到DDL某个表,则对应表的DML操作可能失败,修复方法为:

1. 参考减少同步对象先将这个对象从同步列表中摘除。

- 2. 删除阿里云Elasticsearch中这个表对应的索引。
- 3. 参考 新增同步对象,修改这个同步作业,将这个表重新添加到同步对象中,进行重新初始化。

如果是修改表、新增列的DDL,建议DDL的操作顺序为:

- 1. 先在阿里云Elasticsearch中手动修改对应表的mapping,新增列。
- 2. 再在源RDS for MySQL实例中手动修改表结构,新增列。
- 3. 暂停DTS同步实例,重启DTS同步实例让DTS重新加载阿里云Elasticsearch中修改后的 mapping关系。

配置步骤

下面详细介绍创建RDS for MySQL实例到阿里云Elasticsearch实例同步链路的具体步骤。

1. 购买同步链路

进入数据传输服务 DTS控制台,进入数据同步界面,点击控制台右上角创建同步作业先购买一 个同步链路,购买完同步链路后返回DTS控制台,进行配置同步链路。



在配置同步链路之前需要先购买一个同步链路,同步链路目前支持包年包月及按量付费两种付 费模式,可以根据需要选择不同的付费模式。

购买界面参数

・功能

选择数据同步。

・源实例

选择MySQL。

- ・源实例地域
 - 本示例为RDS for MySQL, 需选择RDS for MySQL实例所在地域。
- ・日标实例

选择Elasticsearch。

- ・目标实例地域
 - 阿里云Elasticsearch实例所在地域,订购后不支持更换地域,请谨慎选择。
- ・同步拓扑

选择单项同步。

・网络类型

默认为专线,目前仅支持专线模式。

同步链路规格

同步链路规格影响了链路的同步性能,同步链路规格跟性能之间的对应关系详见数据同步规 格说明。

- ・订购时长
 - 如果是预付费,默认为1个月,支持勾选开启自动续费功能。
- ・购买数量

默认为1,根据业务实际需要进行选择。

📃 说明:

DTS控制台的同步实例按照地域展示,刚才购买的同步实例所属的地域为同步实例的目标地 域。例如上面购买的是 杭州RDS for MySQL->杭州阿里云Elasticsearch的同步实例,那 么这个同步实例在DTS的杭州地区。进入杭州区域的实例列表,查找刚才购买的同步实 例,然后点击新购实例右侧的配置同步作业开始配置实例。

数据传输		MySQL到Elasticsearch	数据实时同步正式上线,基于№	NySQL Binlog同步,实现室秒级同	步延迟,了解更多>>			
概览		同步作业列表 华东	[11] (杭州) 华东2(上海)	华北1(青岛) 华北2(北3	(1) 华南1(深圳) 华	は13(张家口) 香港		
数据迁移		美国	1(硅谷) 美国(弗吉尼亚) 新加坡 阿联酋(迪拜)	德国(法兰克福) 马来	R西亚(吉隆坡)	(所选地域为同步作业目标实例所在的地	或)
数据订阅		演大	利亚(悉尼) 印度(孟买) 英国(伦敦) 日本(东京	(1) 印度尼西亚(雅加达) 华北5 (呼和浩特)		
数据同步							○ 刷新	创建同步作业
▶ 文件导入导出		同步作业名称 ▼		搜索	排序: 默认排序	▼ 状态: 全部	¥	
操作日志	÷		4大(今部)	- 同小概/J	けまたゴ	同步初均(合部)		15.Pc
数据备份			4人23(土中)	▼ [P]22*160/76	22003601	(回公)第四回(主印) *		1361 F
产品文档		dtstofmh4t4uddk rdsnew-to-es6	未配置		按量付费	单向同步	配置同步链路	转包年包月 升级
解决方案		dts1hv605hf2nxp rdsold-to-es5	od 0 同步中	延时:0 窒秒 速度:0TPS(0.00MB/	s) 技量付费	单向同步	暂停同步	转包年包月 升级 查看更多
		1 暫停同步	释放同步				共有2条,每页显示:20条 «	< 1 > »

2. 配置同步链路

同步作业名称

同步作业名称没有唯一性要求,为了更方便识别具体的作业,建议选择一个有业务意义的作业名称,方便后续的链路查找及管理。

源实例信息

本示例采用数据源为 RDS for MySQL,需要配置RDS实例的ID、数据库账号、数据库密码。

同步作业名称:	rdsnew-to-es6	
源实例信息		
实例类型:	RDS实例	•
实例地区: * 实例ID:	华东1(杭州) rm-bp19c0thdimptg52	其他阿里云账号下的RDS实例
* 数据库账号:	root 帐号需要具备 Replication slave, Replication client 及所有同步对象的 Select	
* 数据库密码:	••••••]
* 连接方式:	● 非加密连接 ● SSL安全连接	

目标实例信息

目标实例信息中需要配置阿里云Elasticsearch的实例ID,及访问阿里云ES实例账号密码。

目标实例信息		
实例类型:	Elasticsearch	
实例地区:	华东1(杭州)	
* Elasticsearch	es-cn-mp90]
* 数据库账号:	elastic]
	帐号需要具备同步对象的 ALL 权限	-
* 数据库密码:	•••••	

以上内容配置完成后,点击授权白名单并进入下一步进行RDS for MySQL及阿里 云Elasticsearch的白名单添加。

3. 授权实例白名单



如果是RDS for MySQL, DTS会自动添加白名单或安全组。

如果源实例为RDS for MySQL,那么DTS将自身的IP段添加到RDS实例的白名单的安全组中,避免因为RDS实例设置了白名单,DTS服务器连接不上数据库导致同步作业创建失败。为了保证同步作业的稳定性,在同步过程中,请勿将这些服务器 IP 从 RDS实例的白名单的安全组中删除。

当白名单授权后,点击下一步,进入同步账号创建。

4. 选择同步对象

当白名单授权完成后,即进入同步对象的选择步骤。在这个步骤可以配置需要同步的表列,以及 索引的命名规则。

a. 索引名称命名规则可以选择:表名、库名_表名。

- ·如果选择了表名,那么索引名称同表名。
- · 如果选择了库名表名,那么索引名称的命名格式为:库名表名。例如,库名为:dbtest ,表名为:sbtest1,那么这张表同步到阿里云Elasticsearch后,对应的索引名称为: dbtest_sbtest1。
- ・如果需要同步的不同库中存在相同名称的表名,建议索引名称命名规则选择:库名_表
 名。
- b. 选择具体需要同步的库表列,实时同步的同步对象的选择粒度可以支持到表级别,即用户可以选择同步某些库或某几张表。

实时同步的同步对象的选择粒度可以支持到表级别,即用户可以选择同步某些库或某几张 表。

同步架构:单向同步	
索引名称: 库表_表名 \$	
源库对象	已选择对象 (鼠标移到对象行,点击编辑可修改对象名或过滤条件)详情点我
🖃 💼 dts 🖸 🎦 Tables	i dts_(1个对象) I dts_dts_test
	> <
全选	全选

c. 默认所有表的docid为表的主键,如果部分表没有主键,那么对于这部分配置docid 对应的 源表的列。在右侧-已选择对象 框中,将鼠标挪到对应表上,点击右侧的 编辑 入口,进入这 个表的高级设置界面。

注意:编辑表名或列名	后,目标数据库的表名列	」名将为修改后的名	称。	
* 索引名称: dts_	dts_test			
* Type名称: dts_	test			
是否分区: 〇是	●否			
_id取值: 表的	回主键列(联合主键合并用	戊──列 ♦		
□ 全选 列名	类型	字段参数	字段参数值	
✓ C1	bigint(20)	index \$	false \$	
	varchar(20	index \$	false \$	添加参数
	valonai(20	Analyzer \$	Standarc \$	
✓ xmltest	text	index 🜲	false 🜲	添加参数

d. 在高级配置中可以设置:

索引名称、Type名称、分区列及分区数定义、_id取值列。其中_id 取值如果选择 业务主键,那么需要选择对应的业务主键列。

- e. 配置完同步对象后,进入高级配置步骤。
- 5. 高级配置

主要配置

a. 同步初始化类型,建议选择结构初始化+全量数据初始化,由DTS自动进行索引的创建及 全量数据的初始化。如果不选择结构初始化,那么需要在同步创建之前,先手动在阿里 云Elasticsearch中完成索引mapping的定义。如果不选择全量数据初始化,那么DTS同步 增量数据的起始时间点为:启动同步的时间点。

- b. 索引分片配置, 默认为5个分片, 1个副本。可以根据业务需要进行调整, 一旦调整后, 所有 的索引按照这个配置定义分片。
- c. 字符串analyzer定义,可以选择字符串的analyzer,默认为Standard Analyzer。取 值包括: Standard Analyzer、Simple Analyzer、Whitespace Analyzer、Stop Analyzer、Keyword Analyzer、English Analyzer、Fingerprint Analyzer,所有索 引的字符串字段按照这个配置定义Analyzer。

创建同步作业 🕯 返回数据同步列表	
1.选择同步通道的源及目标实例	2.选择同步对象 3.高级设置 4.预检查
同步初始化:	☑ 结构初始化 ☑ 全量数据初始化
分片配置:	配置主分片数, 默认值为5 配置分片副本数, 默认值为1
字符串Index:	analyzed 🗘 Standard Analyzer 💠
时区:	+8:00 \$
DOCID取值:	就认主键,无主键表使用Elasticsearch自动生成D

d. 时区,可以配置同步到阿里云Elasticsearch中的时间字段存储的时区,默认为东八区。

6. 预检查

同步作业配置完成后,DTS会进行预检查,当预检查通过后,可以点击 启动 按钮,启动同步作 业。

同步作业启动后,即进入同步作业列表,此时刚启动的作业处于同步初始化状态。初始化的时间 长度取决于源实例中同步对象的数据量大小,初始化完成后,同步链路即进入同步中的状态,此 时源跟目标实例的同步链路才真正建立。

7. 数据效验

以上任务完执行成后,登录阿里云ES控制台,确认对应阿里云ES实例中有无创建对应索引,及 同步的数据是否符合预期。

3.5 使用DataWorks实现MaxCompute与阿里云ES数据同步

阿里云上拥有丰富的云存储、云数据库产品。如果您希望对这些产品中的数据进行分析和搜索,可 以通过DataWorks的数据集成服务,将离线数据同步到阿里云Elasticsearch中进行搜索分析,最 快可达到5分钟一次。



做数据同步时可能会产生公网流量费用,请您知晓。

准备工作

完成离线数据的分析与搜索,您需要完成以下几步操作:

· 创建和查看表,并导入数据。实际情况中,您可以将Hadoop数据迁移MaxCompute最佳实 践再进行同步,本案例使用的表结构和部分数据如下:

列名 ♣	类型 ♦
create_time	STRING
category	STRING
brand	STRING
buyer_id	STRING
trans_num	BIGINT
trans_amount	DOUBLE
click_cnt	BIGINT
pt	STRING

create_time	category	brand	buyer_id	trans_num	trans_amount	click_cnt	pt
2018-08-21 00:00:00	外套	品牌A	null	null	null	null	1
2018-08-22 00:00:00	生鮮	品牌B	null	null	null	null	1
2018-08-22 00:00:00	外套	品牌C	null	null	null	null	1
	卫浴	品牌A	null	null	null	null	1
2018-08-22 00:00:00	生鮮	品牌D	null	null	null	null	1
2018-08-23 00:00:00	外套	品牌B	null	null	null	null	1
2018-08-23 00:00:00	生鮮	品牌A	null	null	null	null	1
2018-08-23 00:00:00	外套	品牌E	null	null	null	null	1
2018-08-24 00:00:00	生鮮	品牌G	null	null	null	null	1
2018-08-24 00:00:00	卫浴	品牌F	null	null	null	null	1
2018-08-24 00:00:00	外套	品牌A	null	null	null	null	1
2018-08-24 00:00:00	卫浴	品牌G	null	null	null	null	1
2018-08-24 00:00:00	外套	品牌C	null	null	null	null	1

- · 创建一个阿里云Elasticsearch实例,用来存储数据集成系统同步成功的数据。
- · 购买一台与阿里云Elasticsearch相同VPC的阿里云ECS,这台ECS将获取数据源数据并执行写 阿里云Elasticsearch数据的任务(该任务将由数据集成系统统一下发)。
- ・开通DataWorks数据集成服务,并且将ECS作为一个可以执行任务的资源,注册到数据集成服务中去。
- ・配置一个数据同步脚本,并让其周期性执行。

操作步骤

- 1. 创建阿里云Elasticsearch和ECS实例
 - a. #unique_32,本案例创建了一个位于华东1区,名称为es_test_vpc的专有网络,对应的交换机名称为es_test_switch。
 - b. 进入阿里云Elasticsearch控制台,创建一个阿里云Elasticsearch实例。





地域、专有网络、虚拟交换机与您第一步中创建的专有网络保持一致。

c. 购买一台与阿里云Elasticsearch实例处于同一个VPC内的ECS服务器,并分配一个公网IP 或开通弹性IP,为了节省您的成本,您可以复用已有且符合条件的ECS服务器。

本案例创建了一个位于华东1,可用区F的ECS实例,使用CentOS 7.4 64位系统,并勾选分配公网地址,网络配置如下:



· 建议使用CentOS 6、CentOS 7 或者 Aliyun Linux。

- · 如果您添加的ECS需要执行MaxCompute任务或者同步任务,需要检查当前ECS的 python版本是否是python2.6或2.7 的版本(CentOS 5 的版本为2.4,其余CentOS自 带了2.6以上版本)。
- ・ 请确保 ECS 有公网 IP。

2. 配置数据同步

- a. 进入DataWorks控制台创建项目,本案例使用名称为bigdata_DOC的DataWorks项目。
 - · 如果您已经开通过DataWorks数据集成产品,您将会看到如下页面:

		概觉 项目列表	调度贷)	源列表	
G DataWorks	数据集	『成・数据开发・MaxCompute			0
快速入口					
数据开发		数据集成		运继中心	
项目					全部项目
bigdata_DOC	华东1	PAitest	半东2	MaxCompute_DOC	华东2
创造时間:2016-09-02-10-26-59 计算可障: MacComput 服务機快 数据开发 数据集成 数据管理 退時中心 项目截置 进入数据开发 进入数据集成		创意时间:2018-05-23 13-22-29 计預引等: MaxCompute PA计算引等 至約使決 数据开发 数据集成 数据管理 近時中心 双目配置 进入数据开发 进入数据集成		会議时前:2018-07-19 09:12:37 计器引導: MasCompute 服务機体 数据开波 数据集成 数据管理 运体中心 項目配置 進入数据开放 進入数据集成	
常用功能					

·如果您未开通过DataWorks数据集成产品,将会看到如下页面。您需要按照步骤开通数 据集成服务,此开通动作会产生费用,请您按照费用提示进行预算评估。

← -	→ C ● 安全 http:	s://workbench.data.aliyun.com/console?s			☆ 📀
(-)	管理控制台	产品与服务 ◄	Q 搜索 单 🤒 费用	工单 备案 企业	支持 简体中文
	ш		概览 项目列表	调度资源列表	
•	云计算基础服务				
•	大数据(数加)	0			A State Stat
•	数加控制台概览	🌀 DataWorks 🐲	数据集成・数据开发・MaxCompu	ute	
0	DataWorks			I	
۲	Quick BI	新用户引导 🕢 实名认证		計算资源 —— 4 创建项目	产品升级功能列表
ख	机器学习				不能错过的新功能,新体验!
N	大数据计算服务	* រ៉	选择区域: 〇 华东2		数据集成正式上线
¢,	智能语音交互	* (付费方式: CU预付费 1/O后付费		多种开发模式,支持更多的数据通道
00	数据集成	* []	项目名称: 字母或下划线开头,只能包含字母下划线和数字	字	ᄊᇚᅿᇏ
c:)	阿里云Elasticsearch		启云之 , 加思不慎 默认为项目之		常见问题
•	安全(云盾)		320/01 · X0201299 · MARC2222011		如何生成AccessKey并绑定 如何创建成员以及赋权
•	域名与网站(万网)	IJ	项目描述:		如何进行本地数据上传下载
•	云市场				如何使用系统参数与自定义参数
			确定		人政黨國為打算说明

- b. 单击DataWorks项目下方的进入数据集成。
- c. 创建资源组。
 - A. 在数据集成页面,选择左侧导航栏中的资源组,单击新增资源组。
 - B. 按照以下步骤,完成资源组的添加:

A. 创建资源组: 自定义输入资源组名称,本案例的资源组名称为es_test_resource。

	资源组管理 输入调度资源名称进行搜索							85.00 (M 10
→ 数据集成概范								
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	资源组名称	网络类型	8	法職		已使用DMU	付勝英型	操作
▼ 项目空间	數以資源這						按量付聘	
→ 項目空间概然	And a state of the	专有网络	新增资源组			×	按量付募	服务器初始化 管理 删除
2 资源消耗监控			0					
→ 商线同步			创建资源组	添加服务器	安始Agent	检查获通		
8 = ====			 资源归名称: 	es_test_resource				
() XXXXX								
➡ 同步资源管理								
A 2010								
					取詞	⊼− ∌		

B. 添加服务器。

新增资源组				×
1 创建资源组	2 不同的 2 不	3 安装Agent	4 检查联通	
* 网络类型: 服务器1	💿 专有网络 🛛 🕜			
* ECS UUID :	请输入UUID,非服	务器名称	0	
* 机器IP:	请输入内网机器IP		0	
∗ 机器CPU(核):				
* 机器内存(GB):				
添加服务器				
			上—步 下—2	ŧ

• ECS UUID: #unique_33 服务器,执行 dmidecode | grep UUID,取返回 值。



- · 机器 IP/机器CPU(核)/机器内存(GB): ECS实例的公网IP/CPU/内存。进 入ECS控制台,单击实例名称链接,在配置信息模块,可以找到相关信息。
- C. 安装Agent:按照界面提示,完成安装Agent步骤。由于本案例使用的是VPC网络,不需要开通服务器的8000端口。
- D. 检查联通:联通成功后,状态会显示为可用。如果状态为不可用,您可以登录该ECS服务器,使用tail -f /home/admin/alisatasknode/logs/heartbeat.log命令查看DataWorks与该ECS服务器之间心跳报文是否超时。
- d. 添加数据源。
 - A. 在数据集成页面,选择左侧导航栏中的数据源,单击新增数据源。
 - B. 选择数据源类型为MaxCompute。



C. 输入数据源信息,本案例创建的数据源名称为odps_es,如下所示。

新增MaxCompute (ODP	S)数据源	×
* 数据源名称	odps_es	
数据源描述	test	
* ODPS Endpoint	http://service.odps.aliyun.com/api	
* ODPS项目名称	bigdata_DOC	
* Access Id		?
* Access Key		
测试连通性	测试连通性	
	上一步	完成

· ODPS空间名称: 在DataWorks的数据开发页面,表对应的空间名称显示在左上角图标右侧,如下图所示:

Datal	DataStudio	bigdata_DOC	~				
	表管理	[‡ C	im hive_doc_good	_ sale × 🏢 b	ank_data 🔅	× Di 314	× Di
$\langle \rangle$	表名/描述	V.	DDL模式	从生产环境加速	载 提3		
*	✓ ■ 表管理						
a	◆ 🖿 其他				4	表名 hive_do	c_good_sale
0	🏢 bank_data		其太届性				
G	🛗 bank_data	_01					
×	🗰 demo_trad	e_amount		中文名:			
⊞	hive_doc_g	ood_sale			语选择		1
==	Hive_esdoo	_good_sale		72.1182.	비난의부		J
==		le_1548120003070018		描述:			
fx	eutput_tab	le 154822526534301a					
	₩ output_tab	_ le_154838757651615e					
Σ	utput_tab	le_1548743722536a9c					
-	🗰 result_table	5	物理模型设计	_			
	🗮 system_9c	676e75c4324f75b5430		分区类型:	💿 分区表 🛛	○ 非分区表	
	₩ t1				121412		1
	test			层级:	南选择		J
	test1			表类型:			
	userlog1						
			表结构设计				

· Access Id/Access Key: 鼠标移至您的用户名称上,选择用户信息,如下图所示:

DotaW	DataStudio	bigdata_DOC	~						跨项目克隆	运维中心	dtplus_d	ocs 中文
ш	表管理	C C	📻 hive_doc_good_sale × 📻	bank_data 🗙 🖸						🛔 wo 🔒		444
S			DDL模式从生产环境加							用户信息	版本历史	用户手册
*	✓ ■ 表管理									*		
Q	> 🛅 其他			表名	hive_doc_good_se					提交工单	退出	
©			基本雇性								关于DataWorks	
			中文名:								<u>42660</u>	
■			一级主题:	请选择		二级主题	: 请选择	~ 新建	E±≣ C			3
≣			描述:									Q6
f×												В,

在个人信息页面, 鼠标移至您的用户头像上, 单击accesskeys进行获取, 如下图所示:

支持与服务 🔄 简体中文 💮
<u>@</u>
基本资料实名认证安全设置
♥ 安全管控
B 访问控制
accesskeys
☺ 会员积分
■ 推荐返利后台
退出管理控制台

e. 配置同步任务。

A. 在数据开发页面,单击左侧菜单栏中的数据开发,打开业务流程导航栏:



B. 右键单击导航栏中的数据集成,选择新建数据集成节点 > 同步节点,输入同步任务名称:



C. 成功创建同步节点后,单击新建同步节点右上角的转换脚本,选择确认即可进入脚本模式:

			$\left[\uparrow \right]$	<u>ل</u> ا		ľ	< <u>\</u>						
01	选择数												
							王王帝王教据的新			」以是默认	人的数据》		
		* 数据》								?			* 数排
02	字段明				?	提您	示 确定要将向导档	模式转化为	脚本模式	℃吗?— <u></u>	目转化将升	こ法撤销!	×
											确认	取消	

D. 单击脚本模式右上角的导入模板,在弹框中分别选择读取端的来源类型和数据源、写入端的目标类型和数据源,单击确认生成初始脚本:

1 { *type": "job", stream Reader 帮助文档 stream Writer 帮助 3 "steps": [\$ \$ \$ 4 { \$ \$ \$ 9 ///// "paramet \$ \$ \$ 10 *col * * *源类型: ODPS ? 10 * * * * * * * * * * * * * * * * * * *			٤]		C P		X					
2 "type": "job", 3 "steps": [4 { 5 { 7 stepTyp 7 paramet 7 col 8 * 来源类型: ODPS												
3 "steps": [4 {		"type"						A				
4 { 5 "stepTyp 6 "paramet 7 "col 8 9 9 9 10 11 * 来源类型: ODPS		"steps										
5 "stepTyp 导入模板 × 6 "paramet" "col 7 "col * 9 * * 10 * * 11 * * 12 * * 13 * * 14 * * 15 * * 16 * * 17 * 数据源: 18 * *												
6 "paramet 7 "col 8 9 10 11 12 13 14 14 14 15 16 17 18 19 19 10 10 10 10 10 10 11 12 13 14 14 15 16 17 18 19 19 19 19 19 19 19 19 19 19			ерТур	导入	模板							×
7 "col 9 * 来源类型: ODPS ? 10 * 数据源: 请选择 ~ 11 * 目标类型: ODPS ? 13 * 目标类型: ODPS ? 14 * 目标类型: ODPS ? 15 * 新增数据源: 请选择 ~ 16 * 数据源: 请选择 ~ 17 新增数据源: 18			ramet									
8 * 来源类型: 00PS ? 10 * 数据源: 请选择 ✓ 11 * 数据源: 請选择 ✓ 12 新增数据源 3 13 * 目标类型: 00PS ? 14 * 目标类型: 00PS ? 15 * 数据源: 请选择 ✓ 16 * 数据源: 請选择 ✓ 17 13 新增数据源 18 19 19												
9 10 11 12 13 14 14 14 15 16 17 18 19 10 10 10 10 10 10 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 11 10 10								De				
10 * 数据源: 请选择 、 11 * 数据源: 请选择 、 12 新增数据源 13 * 目标类型: ODPS 、 14 * 目标类型: 「请选择 、 15 * 16 * 数据源: 请选择 、 17 新增数据源 18 19						* 米冰尖尘	. 00	22			9	
11 * 数据源: 请选择 * 12 新增数据源 13 * 14 * 目标类型: ODPS ? 15 * 数据源: 请选择 * 16 * 数据源: 新增数据源 17												
12 新增数据源 13 *目标类型: ODPS · (?) 14 *目标类型: ODPS · (?) 15 · (?) · (?) 16 * 数据源: 请选择 · (?) 17 · (?) · (?) 18 · (?) · (?)						* 数据源	: 请	战				
13 14 15 16 17 18 19							新增	数据源				
14 *目标类型: ODPS ? 15 *数据源: 请选择 ? 17 新增数据源 19												
15						* 目标类型	: OD	PS			?	
16 * 数据源: 请选择 ~ ~ 17 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7												
17 [13] 18] 新增数据源 19] [19] [19] [19] [19] [19] [19] [19] [* 数据源	: 请	先择				
							空后他	鐵桿源				
							dy121⊨					
22 确认 取消											取消	

E. 配置数据同步脚本,具体配置请参考脚本模式配置,Elasticsearch的配置规则请参考配置Elasticsearch Writer。

"reader": {	2 Odne Deader #
"plugin": "odps",	
"parameter": {	
"partition": "pt=1",	
"datasource": "odps_es",	
"column": [
"create_time",	
"category",	
"brand",	
"buyer_id",	
"trans_num",	
"trans_amount",	
"click_cnt"	
],	
"table": "hive_doc_good_sale"	
}	
}, Nacitaria (
"Writer": {	
prugrn": "elasticsearch",	
parameter": { "accessId": "alastic"	
accessid : eldstic , "endnoint": "http://es.co.wo	sch alivuncs com:0200"
"indevType": "elasticsearch"	ch.allyuncs.com;9200 ,
"accessKev", "	
"cleanup": true.	
"discoverv": false.	
"column": [
{	
"name": "create time".	
"type": "string"	
},	
{	
"name": "category",	
"type": "string"	
},	
{	
"name": "brand",	
"type": "string"	
},	
{	
"name": "buyer_id",	
"type": "string"	
},	
{	
"name": "trans_num",	
"type": "long"	
3) (
i "name", "tease amount"	
name": "trans_amount",	
cype : double	
1) 7	
l "name": "click ont"	
"type": "long"	
cype . TouR	
۲ ۲	
"index": "es index".	
"batchSize": 1000.	

说明:

- ·同步脚本的配置分为三个部分,Reader用来配置您上游数据源(待同步数据的云产品)的config,Writer用来配置阿里云Elasticsearch的config,setting用来配置同步中的一些丢包和最大并发等。
- · endpoint为阿里云Elasticsearch的内网或外网地址,本案例使用内网地址,所以 不用配置白名单。如果您是用的是外网地址,请在阿里云Elasticsearch的网络配

置页面,配置阿里云Elasticsearch的公网地址访问白名单(包括DataWorks服务器的IP地址和您所使用的资源组的IP地址)。

- Elasticsearch Writer中accessId和accessKey需要配置您的阿里 云Elasticsearch的访问用户名(默认为elastic)和密码。
- index为阿里云Elasticsearch实例的索引,您需要使用该索引名称访问阿里 云Elasticsearch的数据。本案例中的index名为es_index。
- ・如果您的ODPS表是一个分区表,需要在partition字段中设置分区信息,本案例中
 的分区信息为pt=1。

配置代码示例如下:

```
"configuration": {
"reader": {
"plugin": "odps",
"parameter": {
  "partition": "pt=1",
  "datasource": "odps_es",
  "column": [
    "create_time",
    "category",
    "brand"
    "buyer_id"
    "trans_num",
     "trans_amount",
     "click_cnt"
  ],
  "table": "hive_doc_good_sale"
}
},
"writer": {
  "plugin": "elasticsearch",
"parameter": {
  "accessId": "elastic",
"endpoint": "http://es-cn-mpXXXXXX.elasticsearch.aliyuncs.
com:9200",
  "indexType": "elasticsearch",
"accessKey": "XXXXXX",
"cleanup": true,
"discovery": false,
  "column": [
     {
       "name": "create_time",
       "type": "string"
    },
     {
       "name": "category",
       "type": "string"
    },
     {
       "name": "brand"
       "type": "string"
    },
     {
       "name": "buyer_id",
       "type": "string"
```

```
},
{
                                                               "name": "trans_num",
"type": "long"
                                          },
                                             {
                                                               "name": "trans_amount",
"type": "double"
                                          },
{
                                                               "name": "click_cnt",
"type": "long"
                                           }
                     ],
"index": "es_index",
                      "batchSize": 1000,
"splitter": ","
  }
 j,
"setting": {
"errorLimit": {
                      "record": "0"
},
"speed": {
    "speed": {

                      "throttle": false,
                      "concurrent": 1,
                      "mbps": "1",
                      "dmu": 1
  }
}
  },
"type": "job",
"version": "1.0"
  }
```

F. 同步脚本配置完成后,单击运行,将ODPS中的数据同步到阿里云Elasticsearch中。

 数据开发 组件管理	文件名称/创建人	T		Þ	ᡗ	5		R	22		
	> 解决方案	88	(59)								
			(01)(12)	ype	vpe": "job",			<u> </u>	stream Reader 帮助又档	stream Writer 帮助又档	

3. 结果验证

- a. 进入阿里云Elasticsearch控制台,单击右上角的kibana控制台,选择Dev Tools。
- b. 执行如下命令,查看数据是否已经同步到ES中。

```
POST /es_index/_search?pretty
{
"query": { "match_all": {}}
```

}

es_index为您同步数据时,设置的index字段的值。

```
如果数据同步成功, 会显示以下界面:
```

7	Dev Tools	Dev Tools								
kiba	Console Search Profiler Grok Debugger	Console Search Profiler Grok Debugger								
	2 · {	2 "took": 26,			- î					
Usualize	3 "query": { "match_all": {}} 4 * }	3 "timed out": false, 4 v " shards": (- 1					
	1	5 Total": S. 6 "successful": S.			- 1					
😨 Timelion		7 "failed": 0			. 1					
ស៍ Machine I	earning	9~ "hits": { 18 "total": 26.			- 1					
		11 "max_score": 1,			- 1					
					- 1					
🎾 Dev Tools		14 "_index": "esindex", 15 "_true": "elaticsearch".			- 1					
Monitorin		16			- 1					
		1/			- 1					
Managerr	ent	19 "create_time": "2018-08-23 00:00:00", 20 "trans gum": 0								
		21 "Click_crt" 7,								
		22 "category": "外章", 								
		23 object_a0 : jiamy, 24 "trans anount": 2000.								
		25 "brand": "品牌8"								
		264								
		$2^{2} + 3^{2}$								
		29 "_index": "es_index",								
		30 "_type": "elasticsearch",								
		31								
		33 •								
		34 "create_time": "2018-08-23 00:00:00",								
		35 "trans_num": 5,								
		37 "category": "牛鲜".								
		38 "buyer_id": "jinmy",								
		39 "trans_anount": 45.1,								
		40 Drang: "高城県A" 41 - 3								
		42*								
		43 * {								
		44index": "es_index", 45true": "es_intex",								
		46 " 10': "AMAGNEW/FUJ2070804Y".								
🔍 elastic		47 "_score": 1,								
		49 * " Source": {								
- Logout		49 Create_lime: 2018-08-24 00:00:00 , 50 "category": "D'XX".								
· · · · · · · · · · · · · · · · · · ·		51 "brand": "品條6"								

c. 执行如下命令,按照trans_num字段对文档进行排序。

```
POST /es_index/_search?pretty
{
    "query": { "match_all": {} },
    "sort": { "trans_num": { "order": "desc" } }
}
```

d. 执行如下命令, 搜索文档中的category和brand字段。

```
POST /es_index/_search?pretty
{
    "query": { "match_all": {} },
    "_source": ["category", "brand"]
}
```

e. 执行如下命令, 搜索category为生鲜的文档。

```
POST /es_index/_search?pretty
{
"query": { "match": {"category":"生鲜"} }
}
```

更多命令和访问方式,请参考ES访问测试和Elastic.co官方帮助中心。

常见问题

无法连通阿里云ES实例相关报错

- 检查在运行同步脚本之前,是否在页面右侧的配置任务资源组中选择了您前面步骤创建的资源 组。
 - ・是、执行下一步。
 - · 否, 单击页面右侧的配置任务资源组, 选择您前面步骤创建的资源组, 完成后单击运行。
- 检查您的同步脚本配置是否正确,包括endpoint(您的阿里云Elasticsearch实例的内网或外 网地址,使用外网地址需要配置公网地址访问白名单)、accessId(阿里云Elasticsearch实例 的访问用户名,默认为elastic)和accessKey(阿里云Elasticsearch实例的访问密码)。

3.6 通过ES-Hadoop将Hadoop数据写入阿里云Elasticsearch

本文档为您演示如何使用E-MapReduce,通过ES-Hadoop直接将数据写入阿里 云Elasticsearch中。

支持版本

阿里云Elasticsearch 5.5.3 with Commercial Feature。

准备工作

在开始本案例前,您需要首先开通如下的阿里云产品:

- · 专有网络VPC:由于通过公网访问推送数据安全性较差,为保证阿里云Elasticsearch访问环境 安全,对应区域下必须要有VPC和虚拟交换机,因此需开通VPC专有网络。
- · OSS: 在本示例中将E-MapReduce的日志存储在OSS上,在开通配置E-MapReduce前需开通 OSS并创建完成Bucket。
- \cdot Elasticsearch
- E-MapReduce

请参考以下步骤开通上述阿里云产品:

- 1. 开通阿里云VPC。
 - a. 登录阿里云首页,选择产品 > 云计算基础 > 网络 > 创建网络环境 > 专有网络VPC,然后单击立即开通。
 - b. 进入到VPC管理控制台界面,新建专有网络。
 - c. 创建完成之后在控制台中可以进行管理。

🗾 说明:

更多关于专有网络VPC的文档请参专有网络VPC。

- 2. 开通专有网络OSS。
 - a. 登录阿里云首页,选择产品 > 云计算基础 > 存储服务 > 云存储 > 对象存储 OSS,然后单击立即开通。
 - b. 进入到OSS管理控制台界面,单击新建 Bucket。



Bucket的区域要和E-MapReduce集群的区域一致,本示例将区域均选择为华东1区。

- c. 根据界面提示完成Bucket创建。
- 3. 开通阿里云Elasticsearch。
 - a. 登录阿里云首页后选择产品 > 大数据 > 大数据搜索与分析 > Elasticsearch,进入阿里云Elasticsearch产品界面。

📋 说明:

新用户可以免费试用30天。

- b. 购买成功后,在Elasticsearch控制台可以看到新创建的Elasticsearch集群实例。
- 4. 开通阿里云E-MapReduce。
 - a. 登录阿里云首页,选择产品 > 大数据 > 大数据计算 > E-MapReduce,进入E-MapReduce产品页面。
 - b. 单击立即购买, 根据界面提示完成参数配置。
 - c. E-MapReduce集群创建成功后在集群列表中查看,也可通过以下操作验证集群创建结果:
 - · 公网IP可以直接访问,远程登录:

ssh root@你的公网IP

· 使用jps命令查看后台进程:

```
[root@emr-header-1 ~]# jps
16640 Bootstrap
17988 RunJar
19140 HistoryServer
18981 WebAppProxyServer
14023 Jps
15949 gateway.jar
16621 ZeppelinServer
1133 EmrAgent
15119 RunJar
17519 ResourceManager
1871 Application
19316 JobHistoryServer
1077 WatchDog
17237 SecondaryNameNode
16502 NameNode
16988 ApacheDsTanukiWrapper
```

18429 ApplicationHistoryServer

写数据到ES的MR作业开发

推荐使用maven来进行项目管理,操作步骤如下:

1. 安装 Maven。

首先确保计算机已经正确安装maven。

2. 生成工程框架。

在工程根目录处执行如下命令:

```
mvn archetype:generate -DgroupId=com.aliyun.emrtoes -DartifactId=
emrtoes -DarchetypeArtifactId=maven-archetype-quickstart -Dinteracti
veMode=false
```

maven会自动生成一个空的Sample工程,工程名为emrtoes(和指定的artifactId一

- 致),里面包含一个简单的pom.xml文件和App类(类的包路径和指定的groupId一致)。
- 3. 加入Hadoop和ES-Hadoop依赖。

使用任意IDE打开这个工程,编辑pom.xml文件。在dependencies内添加如下内容:

4. 添加打包插件。

由于使用了第三方库,需要把第三方库打包到jar文件中,在pom.xml中添加maven-assembly-plugin插件的坐标。

```
<plugins>
   <plugins>
    <plugin>
        <artifactId>maven-assembly-plugin</artifactId>
        <configuration>
            <archive>
                <manifest>
                <manifest>
                <manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
               </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
                </manifest>
```

```
</descriptorRefs>
     </configuration>
     <executions>
       <execution>
         <id>make-assembly</id>
         <phase>package</phase>
         <goals>
           <goal>single</goal>
         </goals>
       </execution>
     </executions>
   </plugin>
   <plugin>
     <proupId>org.apache.maven.plugins</proupId>
     <artifactId>maven-shade-plugin</artifactId>
     <version>3.1.0</version>
     <executions>
       <execution>
         <phase>package</phase>
         <goals>
           <goal>shade</goal>
         </goals>
         <configuration>
           <transformers>
              <transformer implementation="org.apache.maven.plugins.
shade.resource.ApacheLicenseResourceTransformer">
              </transformer>
           </transformers>
         </configuration>
       </execution>
     </executions>
   </plugin>
 </plugins>
```

5. 编写代码。

在com.aliyun.emrtoes包下和App类平行的位置添加新类EmrToES.java,内容如下。

```
package com.aliyun.emrtoes;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.util.GenericOptionsParser;
import org.elasticsearch.hadoop.mr.EsOutputFormat;
import java.io.IOException;
public class EmrToES {
     public static class MyMapper extends Mapper<Object, Text,
NullWritable, Text> {
         private Text line = new Text();
         @Override
         protected void map(Object key, Text value, Context context)
                 throws IOException, InterruptedException {
             if (value.getLength() > 0) {
                 line.set(value);
                 context.write(NullWritable.get(), line);
             }
         }
     }
```
```
public static void main(String[] args) throws IOException,
ClassNotFoundException, InterruptedException {
           Configuration conf = new Configuration();
           String[] otherArgs = new GenericOptionsParser(conf, args).
getRemainingArgs();
           //阿里云 Elasticsearch X-PACK用户名和密码
           conf.set("es.net.http.auth.user", "你的X-PACK用户名");
conf.set("es.net.http.auth.pass", "你的X-PACK密码");
           conf.setBoolean("mapred.map.tasks.speculative.execution",
false);
           conf.setBoolean("mapred.reduce.tasks.speculative.execution
", false);
           , conf.set("es.nodes", "你的Elasticsearch内网地址");
conf.set("es.port", "9200");
conf.set("es.nodes.wan.only", "true");
           conf.set("es.nodes.wan.onty", "true");
conf.set("es.resource", "blog/yunqi");
conf.set("es.mapping.id", "id");
conf.set("es.input.json", "yes");
Job job = Job.getInstance(conf, "EmrToES");
           job.setJarByClass(EmrToES.class);
           job.setMapperClass(MyMapper.class);
           job.setInputFormatClass(TextInputFormat.class);
           job.setOutputFormatClass(EsOutputFormat.class);
           job.setMapOutputKeyClass(NullWritable.class);
           job.setMapOutputValueClass(Text.class);
           FileInputFormat.setInputPaths(job, new Path(otherArgs[0]));
           System.exit(job.waitForCompletion(true) ? 0 : 1);
      }
 }
```

上述代码的相关说明请参见下文的API分析。

6. 编译并打包。

在工程的目录下,执行如下命令:

mvn clean package

执行完毕以后,可在工程目录的target目录下看到一个emrtoes-1.0-SNAPSHOT-jar-withdependencies.jar,这个就是作业jar包。

•	🗕 🗧 🖉 ei	mrtoes [~/Docun	nents/mycode/emr	toes]/src/mai	in/java/com/aliyun/em	nrtoes/EmrToES	.java [emrtoes]					
15	emrtoes $ angle$ has src $ angle$ has main $ angle$ have	aliyun 🔪 🖿 emrto	oes 🗟 💣 EmrToES 🤇					↓ ⁰¹ 01	▼ ▶ ∰	⊗ ■	•	Q,
ect	Project ▼ ⊕ ≑ ∳	m emrtoes ×	🕏 EmrToES.java 🛛	$_{\rm III}$ blog.json $ imes$	dependency-reduce	ed-pom.xml ×	🕏 App.java 🛛					*
II 1: Proje	 memtoes ~/Documents/mycode/emrtoes idea isrc java java mein java EnrroES target archive-tmp classes maven-archiver maven-status surefire-reports test-classes enrtoes-1.0-SNAPSHOT.jar enrtoes-1.0-SNAPSHOT.jar original-emrtoes-1.0-SNAPSHOT.jar blog.json dependency-reduced-pom.xml memtoes.iml 	34 35 35 36 37 38 39 40 41 42 43 44 44 45 46 50 51 52 53 54 55 55 56 57 58 59	<pre>//阿里云 conf.set conf.set conf.set conf.set conf.set conf.set conf.set job.setJ job.setJ job.setM job.setW fileInpu System.e</pre>	Elasticsearn ("es.net.ht" Boolean(nan Golean(nan ("es.nodes", ("es.nodes", ("es.nodes", ("es.resourn ("es.input.] arByClass(Er apperClass(fr nputFormatC apportlass(fr apperClass(fr nputFormats) apOutputKey apOu	ch X—PACK用户名; tp.auth.user", tp.auth.pass", he: "mapred.red "mapred.red "y200"); tr wan.only", "tr" g.id", "id"); json", "yes"); stance(conf, " nrToES.class); MyMapper.class tlass(TextInput; Class(EsOutput) Class(TextInput; JuputPaths(job, tForCompletion	和密码 "你的X-PAC("你的X-PAC(.tasks.spec uce.tasks.s tzce000vre4 ue"); qi"); obName: "Em); Format.clas able.class) lass); , new Path((verbose: tr	<pre>(用户名,默认 (密码"); ulative.ex peculative (5.elastics rToES"); (5.); (5.); (5.); (7</pre>	elastic"); ecution", .execution earch.aliyu 0])); 1);	value: false ', value: fa incs.com");	<pre>>); ilse);</pre>	×	🛿 Ant Build 🛛 🕅 Database 🛛 E Maven Projects
¥ 2: Favorites 🛛 🔀 2: Structure	Terminal (INFO] Replacing /Users/yaopan oss/target/emrtoes-1.0-SNAPSHO [INFO] Dependency-reduced POM [INFO] [INFO] BUILD SUCCESS [INFO] Total time: 18.156 s [INFO] Total time: 18.156 s [INFO] Finished at: 2018-06-21 [INFO] pandeMacBook-Pro:emrtoes yaopa	/Documents/ /T-shaded.ja written at: 	mycode/emrto ır /Users/yaop 	es/target/em	irtoes–1.0–SNAF	PSHOT.jar w es/dependen - -	ith /Users, cy-reduced	/yaopan/Doc -pom.xml	uments/myc	¢- ode/emr	t.	
	🥌 📴 Topo 🔤 Java Enterprise 🛛 🕅 Terminal								00.0 15	Event I	Log	•
									60:2 LF	2 01F-8¢	Ó	Ŷ

EMR中完成作业

1. 测试数据。

a. 把下面的数据写入到blog.json中。

```
{"id":"1","title":"git简介","posttime":"2016-06-11","content":"svn
与git的最主要区别..."}
{"id":"2","title":"ava中泛型的介绍与简单使用","posttime":"2016-06-12
","content":"基本操作: CRUD ..."}
{"id":"3","title":"SQL基本操作","posttime":"2016-06-13","content":"
svn与git的最主要区别..."}
{"id":"4","title":"Hibernate框架基础","posttime":"2016-06-14","
content":"Hibernate框架基础..."}
```

{"id":"5","title":"Shell基本知识","posttime":"2016-06-15","content ":"Shell是什么..."}

b. 使用scp远程拷贝命令,将blog.json文件上传到阿里云E-MapReduce集群中。

scp blog.json root@你的弹性公网IP:/root

c. 执行以下命令,将blog.json文件上传至HDFS。

hadoop fs -mkdir /work

hadoop fs -put blog.json /work

2. 上传JAR包。

执行以下命令,将上文创建的maven工程target目录下的jar包上传至阿里云E-MapReduce集群。

scp target/emrtoes-1.0-SNAPSHOT-jar-with-dependencies.jar root@
YourIP:/root

3. 使用以下命令,执行MR作业。

hadoop jar emrtoes-1.0-SNAPSHOT-jar-with-dependencies.jar /work/blog
.json

运行成功后,控制台会输出如下图所示的信息。

1. root@emr-header-1:~ (ssh) [root@emr-header-1 ~]# hadoop jar emrtoes-1.0-SNAPSHOT-jar-with-dependencies.jar /work/blog.json SLF4J: Class path contains multiple SLF4J bindings. SLF4J: Found binding in [jar:file:/opt/apps/ecm/service/hadoop/2.7.2-1.2.11/package/hadoop-2.7.2-1.2.11/share/had oop/common/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class] SLF4J: Found binding in [jar:file:/opt/apps/ecm/service/tez/0.8.4/package/tez-0.8.4/lib/slf4j-log4j12-1.7.10.jar! /org/slf4j/impl/StaticLoggerBinder.class] SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation. SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory] 18/06/21 15:53:18 INFO impl.TimelineClientImpl: Timeline service address: http://emr-header-1.cluster-67561:8188/ ws/v1/timeline/ 18/06/21 15:53:18 INFO client.RMProxy: Connecting to ResourceManager at emr-header-1.cluster-67561/192.168.0.103: 8032 18/06/21 15:53:19 INFO input.FileInputFormat: Total input paths to process : 1 18/06/21 15:53:19 INFO mapreduce.JobSubmitter: number of splits:1 18/06/21 15:53:19 INFO Configuration.deprecation: mapred.reduce.tasks.speculative.execution is deprecated. Instea d, use mapreduce.reduce.speculative 18/06/21 15:53:19 INFO Configuration.deprecation: mapred.map.tasks.speculative.execution is deprecated. Instead, use mapreduce.map.speculative 18/06/21 15:53:19 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1529566866753_0001 18/06/21 15:53:19 INFO impl.YarnClientImpl: Submitted application application_1529566866753_0001 18/06/21 15:53:20 INFO mapreduce.Job: The url to track the job: http://emr-header-1.cluster-67561:20888/proxy/app lication_1529566866753_0001/ 18/06/21 15:53:20 INFO mapreduce.Job: Running job: job_1529566866753_0001 18/06/21 15:53:28 INFO mapreduce.Job: Job job_1529566866753_0001 running in uber mode : false 18/06/21 15:53:28 INFO mapreduce.Job: map 0% reduce 0% 18/06/21 15:53:34 INFO mapreduce.Job: map 100% reduce 0% 18/06/21 15:53:40 INFO mapreduce.Job: map 100% reduce 14% 18/06/21 15:53:41 INFO mapreduce.Job: map 100% reduce 57% 18/06/21 15:53:42 INFO mapreduce.Job: map 100% reduce 71% 18/06/21 15:53:43 INFO mapreduce.Job: map 100% reduce 86% 18/06/21 15:53:44 INFO mapreduce.Job: map 100% reduce 100% 18/06/21 15:53:44 INFO mapreduce.Job: Job job_1529566866753_0001 completed successfully 18/06/21 15:53:44 INFO mapreduce.Job: Counters: 66 File System Counters FILE: Number of bytes read=412 FILE: Number of bytes written=1024771 FILE: Number of read operations=0 FILE: Number of large read operations=0 FILE: Number of write operations=0 HDFS: Number of bytes read=635 HDFS: Number of bytes written=0 HDFS: Number of read operations=2 HDFS: Number of large read operations=0 HDFS: Number of write operations=0

查看结果

执行以下命令验证数据是否成功写入Elasticsearch。

```
curl -u elastic -XGET es-cn-v0h0jdp990001rta9.elasticsearch.aliyuncs.
com:9200/blog/_search?pretty
```

```
验证成功后,结果如下图所示。
```

1. root@emr-header-1:~ (ssh)

```
• • •
[root@emr-header-1 ~]# curl -u elastic -XGET es-cn-4590nukw4000xuig3.elasticsearch.aliyuncs.com:9200/blog
/_search?pretty
Enter host password for user 'elastic':
{
  "took" : 17,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "failed" : 0
  },
  "hits" : {
    "total" : 5,
    "max_score" : 1.0,
    "hits" : [
      {

"_index" : "blog",

"_type" : "yunqi",

"5"
         "_id" : "5",
        "_score" : 1.0,
"_source" : {
          "id": "5",
"title": "Shell基本知识",
           "posttime" : "2016-06-15",
           "content": "Shell是什么..."
        }
      },
      {
        "_index" : "blog",
         "_type" : "yunqi",
        "_id" : "4",
"_score" : 1.0,
          _source" : {
           "id" : "4",
"title" : "Hibernate框架基础",
           "posttime" : "2016-06-14",
           "content": "Hibernate框架基础...."
        }
```

},

您也可以在Kibana控制台进行验证,结果如下图所示。

kibana Console Search Profiler Grok Debugger Image: Search Profiler 1 GET / 1 1 Image: Search Profiler 1 1 1 1 1 Image: Search Profiler 1 1 1 1 1 1 1 Image: Search Profiler 1 </th <th>y Journey i</th> <th></th>	y Journey i	
Oliscover 1 GET / I 1 GET / I GET / 2 Get / 3 Get / O Dashboard 5 "query": { Imathen 7 } Imathen 7 } Imathen 7 } Imathen 8 } Imathen 8 } Imathen 8 } Imathen 8 > Imathen 8 > Imathen 8 > Imathen 8 > Imathen 10 Imathen Imathen 11 Imathen Imathen 11 Imathen Imathen 12 Imathen Imathen 12 Imathen Imathen 12 Imathen Imathen 13 Imathen Imathen 15 Imathen Imathen 11 Imathen Imathen 12 Imathen Imathen Imath		
1 Visualize 2 GET _search 3 "timed_out": false, (*) Dashboard 5- "query": { 4- { "_shards": { (*) Dashboard 5- "query": { 4- * "_shards": { (*) matchall": {} 7- } 8- } * (*) Machine Learning 0 6 **/ * * (*) Machine Learning 10 GET blog/_search 10 "total": 5, (*) Graph 11 11 **/ * * (*) Dev Tools 12 12 **/ * * (*) Monitoring 12 **/ * * *		
Image: Second		
So Dashboard 5 - "query": { 5 "total": 5, So Timelion 7 - } } 6 "successful": 5, So Machine Learning 9 8 - } 8 - } 9 - "hits": { Machine Learning 9 10 GET blog/_search 9 "total": 5, Machine Learning 9 10 GET blog/_search 9 "hits": { Dev Tools 12 12 "and example		
Graph 10 GET blog/_search 6 "successful": 5, Y 10 GET blog/_search 7 "failed": 0 ** Graph 10 GET blog/_search 9 11 12 "max_score": 1, 12 * Dev Tools 14 "_inidex": "blog", 0 Monitoring 15 "_type": "yunqi",		
6 infinited in the second seco		
(i) Machine Learning 9 (ii) GET blog/_search 9 (iii) GET blog/_search 10 (iiii) Dev Tools 11 (iv) Monitoring 12		
image: Second interview image: Second interview image: Second interview image: Second interview <th></th> <th></th>		
A supplin II III IIII IIII IIII IIIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII		
Dev Tools 13 - 14 { Monitoring 15 "_index": "blog", "_type": "yunqi",		
Monitoring 14 "_index": "blog", 15 "_type": "yunqi",		
16 "_id": "5",		
Management 17 "_score": 1,		
18		
20 "title": "Shell基本知识",		
21 "posttime": "2016-06-15",		
22 "content": "Shell是什么" 22 3		
24 - 3,		
25 - {		
26 "_index": "blog", 37 " tyme": "yunsi"		
27		
29 ".score": 1.		
30 - "_source": {		
31 31 "id": "4",		
_ elastic 32 "title": "Hibernate框架基础",		
33 "posttime": "2016-06-14",		
Content": "Hibernate框架基础"		
Collapse		

API分析

Map过程按行读入, input key的类型为Object, input value的类型为Text。输出的key为 NullWritable类型, NullWritable是Writable的一个特殊类, 实现方法为空实现, 不从数据流 中读数据, 也不写入数据, 只充当占位符。

MapReduce中如果不需要使用键或值,就可以将键或值声明为NullWritable,这里把输出的key设置NullWritable类型,输出的value为BytesWritable类型,将json字符串序列化。

```
论明:
在本示例中只需要写入,因此没有Reduce过程。
参数配置说明
conf.set("es.net.http.auth.user", "你的X-PACK用户名")
设置X-PACK的用户名。
conf.set("es.net.http.auth.pass", "你的X-PACK密码")
设置X-PACK的密码。
conf.setBoolean("mapred.map.tasks.speculative.execution", false)
关闭mapper阶段的执行推测。
conf.setBoolean("mapred.reduce.tasks.speculative.execution", false)
关闭reducer阶段的执行推测。
```

conf.set("es.nodes", "你的Elasticsearch内网地址")

配置Elasticsearch的IP和端口。

- conf.set("es.resource", "blog/yunqi")
- 设置索引到Elasticsearch的索引名和类型名。
- conf.set("es.mapping.id", "id")

设置文档id,这个参数id是文档中的id字段。

• conf.set("es.input.json", "yes")

指定输入的文件类型为json。

• job.setInputFormatClass(TextInputFormat.class)

设置输入流为文本类型。

• job.setOutputFormatClass(EsOutputFormat.class)

设置输出为EsOutputFormat类型。

• job.setMapOutputKeyClass(NullWritable.class)

设置Map输出的key类型为NullWritable类型。

• job.setMapOutputValueClass(BytesWritable.class)

设置Map输出的value类型为BytesWritable类型。

• FileInputFormat.setInputPaths(job, new Path(otherArgs[0]))

设置传入HDFS上的文件路径。

3.7 Logstash部署

Logstash是一个开源的数据收集引擎,具有实时传输数据的能力。它可以统一过滤来自不同源的数据,并按照您制定的规范将过滤的数据输出到目标源中。本文档为您介绍在ECS上部署Logstash的方法,并通过一个简单的示例为您演示Logstash的使用步骤。

概述

本文档为您介绍了部署安装Logstash、使用Logstash同步增量数据以及监控Logstash节点的方法,并在最后给出了操作过程中的常见问题。

准备工作

 购买阿里云Elasticsearch以及能够同时访问自建集群和阿里云Elasticsearch的ECS实例(已 符合条件的ECS不需要重复购买)。

您可以购买经典网络的ECS实例,前提是该ECS实例能够通过经典网络访问VPC内的阿里 云Elasticsearch服务。

2. 安装JDK, 要求JDK版本为1.8及以上版本。

部署安装Logstash

1. 下载5.5.3版本的Logstash。

```
在Elastic官网页面中,下载与ElasticSearch版本一致的Logstash(建议下载5.5.3版本)。
```

2. 对下载的Logstash压缩包进行解压缩。

```
tar -xzvf logstash-5.5.3.tar.gz
```

ElasticSearch从5.x版本之后,会对配置文件进行严格校验。

使用Logstash同步增量数据

- 1. 创建数据接入的用户名和密码。
 - a. 创建角色。

在您购买的ECS的命令行界面,执行以下命令,创建角色。

```
curl -XPOST -H "Content-Type: application/json" -u elastic:es-
password http://***instanceId***.elasticsearch.aliyuncs.com:9200/
_xpack/security/role/***role-name*** -d '{"cluster": ["manage_ind
ex_templates", "monitor"],"indices": [{"names": [ "logstash-*" ],
    "privileges":["write","delete","create_index"]}]}'
```

- · es-password: 阿里云Elasticsearch实例的密码,即您登录Kibana控制台的密码。
- ***instanceId***: 阿里云Elasticsearch实例的ID,可在实例的基本信息页面获
 取。
- · ***role-name***: 您想使用的角色名称。

📕 说明:

Logstash默认的索引名称以logstash-当前日期命名,所以在添加用户角色的时候,需
 要有对logstash-*索引开放读写权限。

kibana Users Roles	/ Roles	
Discover	New Role	
💶 Visualize	Name	
S Dashboard	logstash-writer-role	
3 Timelion	Cluster Privileges	
Machine Learning		
es meening	C monitor	
🔆 Graph	anage	
🖌 Dev Tools	manage_security	
 Monitoring 	manage_pipeline	
🗱 Management	manage_ingest_pipelines	
	transport_client	
	manage_ml	
	monitor_ml	
	manage_watcher	
	monitor_watcher	
	Run As Privileges	
	Add a user	
	Index Privileges	
	Indices Privileges	
	logstash* × read × create × delete × write × create_index	×
	Granted Documents Query Optional Granted Fields Optional	

b. 创建用户。

在您购买的ECS的命令行界面,执行以下命令,创建用户。

```
curl -XPOST -H "Content-Type: application/json" -u elastic:es
-password http://***instanceId***.elasticsearch.aliyuncs.com:
9200/_xpack/security/user/***user-name*** -d '{"password" : "***
logstash-password***","roles" : ["***role-name***"],"full_name" :
"***your full name***"}'
```

- · es-password: 阿里云Elasticsearch实例的密码,即您登录Kibana控制台的密码。
- ***instanceId***: 阿里云Elasticsearch实例的ID,可在实例的基本信息页面获
 取。
- · ***user-name***: 您想创建的数据接入的用户名。
- · ***logstash-password***: 您创建的数据接入用户的密码。
- · ***role-name***: 您上一步创建的角色的名称。
- ・***your full name***: 当前用户名的全名描述。



您也可以在Kibana控制台中创建用户。

kibana Management / Security / Users	
KIDANA Users Roles Image: Second	New User Username aliyun-logstash-write Password Password Again, Please Full Name helloworld Email helloworld@aliyun Roles togstash-writer-role × Save Cancel

上图中的Roles需要选择您上一步创建的角色名称。

2. 编写conf文件。

```
在您购买的ECS中创建test.conf文件,并参考以下内容进行配置。
```

```
input {
    file {
        path => "/your/file/path/xxx"
        }
}
filter {
        output {
        elasticsearch {
            hosts => ["http://***instanceId***.elasticsearch.aliyuncs.com:
9200"]
        user => "***user-name***"
        password => "***logstash-password***"
    }
}
```

```
🗾 说明:
```

Logstash提供了丰富的input、filter、output插件,只需要简单的配置就可是实现数据的流转,详情请参见官方配置文件结构文档。

- · ***instanceId***: 阿里云Elasticsearch实例的ID, 可在实例的基本信息页面获取。
- ·***user-name***: 您上一步中创建的数据接入的用户名。
- · ***logstash-password***: 您上一步中创建的数据接入用户的密码。

(!) 注意:

用户名和密码需要使用英文引号,防止特殊字符在启动logstash时报错。

3. 执行logstash命令。

按照上一步中配置的conf文件,执行logstash命令。

bin/logstash -f path/to/your/test.conf

命令执行成功后,系统会自动通过Logstash获取file中的变化,并提交到Elasticsearch集群。只要监控的file文件有新增内容,Logstash就会自动索引到Elasticsearch集群中。

监控Logstash节点

您可以通过以下步骤监控Logstash节点,并收集监控日志。

1. 为Logstash安装X-Pack插件。

单击<mark>此处</mark>下载X-Pack插件,完成后执行以下命令对X-Pack进行部署安装。

```
bin/logstash-plugin install
file:///path/to/file/x-pack-5.5.3.zip
```

2. 创建Logstash监控用户。

阿里云Elasticsearch集群默认会禁掉logstash_system用户,因此您需要创建一个角色为logstash_system的用户名(用户名不可以配置为logstash_system)。本文档以logstash_system_monitor用户名为例,为您讲解以下两种方式创建用户的方法。

通过命令行方式添加用户。

```
curl -u elastic:es-password -XPOST http://***instanceId***.
elasticsearch.aliyuncs.com:9200/_xpack/security/user/logstash_s
ystem_monitor -d '{"password" : "***logstash-monitor-password
```

```
***","roles" : ["logstash_system"],"full_name" : "your full name
"}'
```

- es-password: 阿里云Elasticsearch实例的密码,即您登录Kibana控制台的密码。
- ***instanceId***: 阿里云Elasticsearch实例的ID,可在实例的基本信息页面获
 取。
- ***logstash-monitor-password***: 您创建的logstash_system_monitor用户 的密码。
- · 通过Kibana控制台添加监控用户。
 - a. 进入Kibana控制台, 单击Management > Users。

	kibana	Management	
Ø	Discover	Version: 5.5.3	
	Visualize	Security	
\odot	Dashboard	Lisors	Polos
8	Timelion	USEIS .	Roles
ø	Machine Learning		
*	Graph	Elasticsearch	
يو	Dev Tools	Watcher	
o	Monitoring		
۵	Management	📕 Kibana	

b. 在Users页面, 单击Create user。

KIDdild Users Roles Discover Visualize Dashboard Timelion Machine Learning Graph Dev Tools Monitoring		libono	Manage	ement / Security			
 Discover Visualize Dashboard Timelion Machine Learning Graph Dev Tools Monitoring 		KIDana	Users	Roles			
Image: Wisualize Q. Search Image: Dashboard No use Image: Timelion No use Image: Machine Learning Image: Comparison of the tearning Image: Graph Image: Comparison of the tearning Image: Comparison of the tearning Image: Comparison of tearning Image: Comparison of tearning Image: Compa	Ø		_				
 Dashboard Timelion Machine Learning Graph Dev Tools Monitoring 	[10]			Q Search			
 Timelion Machine Learning Graph Dev Tools Monitoring 	0					No us	ers
 Inmetion Machine Learning Graph Dev Tools Monitoring 							
Machine Learning Graph Dev Tools Monitoring							
Image: Graph Image: Dev Tools Image: Monitoring	••	Machine Learning					
لو Dev Tools O Monitoring	- ×						
Monitoring	يو.						
	o						
🐼 Management	ф	Management					

c. 输入下图所示的配置信息, 单击Save。



常见问题

・更改集群自动创建索引配置。

阿里云Elasticsearch为了保证用户操作数据的安全性,默认把自动创建索引的配置设置为不允许。

Logstash在上传数据的时候,使用的是提交数据的方式创建索引,而不是使用create index API的方式。所以在使用Logstash上传数据之前,需要首先把集群的自动创建索引设置为允许。

YML文件配置		
自动创建索引	允许自动创建索引 🕖	删除素引指定名称 删除时明确素引名称 🥹
Auditlog索引:	不开启auditlog索引 ?	开启Watcher: 关闭 📀
其他configure配置	0	

<u>!</u>注意:

修改配置并确认后,阿里云Elasticsearch会自动重启,为保证您的业务不受影响,请谨慎操作。

・创建索引时提示没有权限。

[2017	7-12-01T15:	01:11,523][INF	-0][logst	ash.outputs.el	.asticsearch]	Retrying	individual	bulk actio	ons that	failed or	[,] were reje	ected by t	he previous	; bulk requ	est. {:co	unt=>1}				
[2017	7-12-01T15:	01:13,534][INF	0][logst	ash.outputs.el	asticsearch]	retrying	failed act	ion with re	esponse c	:ode: 403	({"type"=:	>"security	_exception"	, "reason"	=>"action	[indices:ad	dmin/create]	is un	authorized	for
user	 [logstash 	-writer-user]	'})																	
[2017	7-12-01T15:	01:13,534][INF	0][logst	ash.outputs.el	asticsearch]	Retrying	individual	bulk actio	ons that	failed or	• were reje	ected by t	he previous	bulk requ	est. {:co	unt=>1}				
[2017	7-12-01T15:	01:17,549][INF	0][logst	ash.outputs.el	asticsearch]	retrying	failed act	ion with re	esponse c	:ode: 403	({"type"=:	>"security	_exception"	, "reason"	=>"action	[indices:ad	dmin/create]	is un	authorized	for
user	· [logstash	-writer-user]	'})																	
[2017	7-12-01T15:	01:17,549][INF	0][logst	ash.outputs.el	asticsearch]	Retrying	individual	bulk actio	ons that	failed or	r were reje	ected by t	he previous	; bulk requ	est. {:co	unt=>1}				
[2017	7-12-01T15:	01:25,567][INF	0][logst	ash.outputs.el	asticsearch]	retrying	failed act	ion with re	esponse c	code: 403	({"type"=:	>"security	_exception"	, "reason"	=>"action	[indices:ad	dmin/create]	is un	authorized	for
user	 [logstash 	-writer-user]	'})																	
[2017	7-12-01T15:	01:25,567][INF	0][logst	ash.outputs.el	asticsearch]	Retrying	individual	bulk actio	ons that	failed or	r were reje	ected by t	he previous	; bulk requ	est. {:co	unt=>1}				
[2017	7-12-01T15:	01:41,592][INF	0][logst	ash.outputs.el	asticsearch]	retrying	failed act	ion with re	esponse c	code: 403	({"type"=>	>"security	_exception"	', "reason"	=>"action	[indices:ad	dmin/create]	is un	authorized	for

请检查您创建的接入数据的用户拥有的角色,是否具有write、delete、create_index权限。

・系统提示内存不足。



Logstash默认配置的是1GB的内存,如果您申请的ECS内存不足,可以修改config/jvm. options中的内存配置,适当调小Logstash内存的使用。

· 配置test.conf时,用户名和密码没有添加引号。

[rootet2pplaroody>30mH35ae521 logstash-5.5.3]# bin/logstash -f task/test.comf ERROR Statuslagger No log42: configuration file found. Using default configuration: logging only errors to the console. Sending logstash's logs to /root/tmp/logstash-5.5.3/logs which is now configured via log4j2.properties [2017-12-01TJs18:02.034] [ERROR]]logstash.agent] Cannot create pipeline {reason="Executed one of #, {, } at line 12, column 22 (byte 261) after output {\n elasticsearch {\n hosts → [\"http://or pipeline {reason="Executed" on passand → Ac ""} [2017-12-01TJs18:02.045][INF0][logstash.outputs.elasticsearch] Elasticsearch pol URLs updated {:changes→{:removed→[], :added→[http://logstash_system_monitor:xxxxxxee=.cn=mp90cbsy1002ejbtn.elasticsearch]

如果在配置您的任务文件时(上文提到的test.conf文件),用户名和密码中有特殊字符但是 又没有用引号括起来,就会出现上述的错误信息。

3.8 自建Elasticsearch迁移

本文档为您介绍将ECS自建的Elasticsearch迁移至阿里云Elasticsearch的方法,包括创建索引和 数据迁移。

教程概述

本案例的整体步骤如下。

- 1. 创建索引。
- 2. 数据迁移。

同时本文档也为您介绍了一些操作过程中可能遇到的问题和解决方法,详情请参见常见问题。

前提条件

参考本文档做迁移前必须先满足以下条件,如果不满足需要通过其他方案进行迁移,详情请参见Logstash部署。

- · 自建Elasticsearch所在的ECS必须是VPC网络(不支持Classiclink方式打通的ECS),并且 自建Elasticsearch必须与阿里云Elasticsearch在同一个VPC下。
- ·您可以通过中控机器(或者任意一台机器)执行文档中的脚本,前提是该中控机器可以同时访问 新旧Elasticsearch集群的9200端口。
- · 自建Elasticsearch所在的ECS的VPC安全组不能限制IP白名单,并且需要开启9200端口。
- · 自建Elasticsearch所在的ECS的VPC安全组不能限制阿里云Elasticsearch实例的各节点IP(Kibana控制台可查看阿里云Elasticsearch实例各节点的IP)。
- ・自建Elasticsearch与阿里云Elasticsearch实例已经连通。可以在执行脚本的机器上使用curl
 -XGET http://<host>:9200进行验证。

创建索引

参考旧集群中需要迁移的索引配置,提前在新集群中创建索引。或者为新集群开启自动创建索引和 动态映射(不建议)功能。

以Python为例,使用如下脚本在新集群中批量创建旧集群索引,默认新创建的索引副本数为0。

```
#!/usr/bin/python
# -*- coding: UTF-8 -*-
# 文件名: indiceCreate.py
import sys
import base64
import time
import httplib
import json
## 老集群host (ip+port)
oldClusterHost = "old-cluster.com"
## 老集群用户名,可为空
oldClusterUserName = "old-username"
## 老集群密码,可为空
oldClusterPassword = "old-password"
## 新集群host (ip+port)
newClusterHost = "new-cluster.com"
## 新集群用户名,可为空
newClusterUser = "new-username"
## 新集群密码,可为空
newClusterPassword = "new-password"
DEFAULT_REPLICAS = 0
def httpRequest(method, host, endpoint, params="", username="",
password=""):
    conn = httplib.HTTPConnection(host)
   headers = \{\}
   if (username != "") :
        'Hello {name}, your age is {age} !'.format(name = 'Tom', age =
 '20')
       base64string = base64.encodestring('{username}:{password}'.
format(username = username, password = password)).replace('\n', '')
       headers["Authorization"] = "Basic %s" % base64string;
    if "GET" == method:
        headers["Content-Type"] = "application/x-www-form-urlencoded"
        conn.request(method=method, url=endpoint, headers=headers)
    else :
       headers["Content-Type"] = "application/json"
```

```
conn.request(method=method, url=endpoint, body=params, headers
=headers)
    response = conn.getresponse()
    res = response.read()
    return res
def httpGet(host, endpoint, username="", password=""):
return httpRequest("GET", host, endpoint, "", username, password)
def httpPost(host, endpoint, params, username="", password=""):
    return httpRequest("POST", host, endpoint, params, username,
password)
def httpPut(host, endpoint, params, username="", password=""):
    return httpRequest("PUT", host, endpoint, params, username,
password)
def getIndices(host, username="", password=""):
    endpoint = "/_cat/indices"
    indicesResult = httpGet(oldClusterHost, endpoint, oldCluster
UserName, oldClusterPassword)
    indicesList = indicesResult.split("\n")
    indexList = []
for indices in indicesList:
         if (indices.find("open") > 0):
              indexList.append(indices.split()[2])
    return indexList
def getSettings(index, host, username="", password=""):
    endpoint = "/" + index + "/_settings"
    indexSettings = httpGet(host, endpoint, username, password)
print index + " 原始settings如下: \n" + indexSettings
    settingsDict = json.loads(indexSettings)
    ## 分片数默认和老集群索引保持一致
    number_of_shards = settingsDict[index]["settings"]["index"]["
number_of_shards"]
    ## 副本数默认为0
    number_of_replicas = DEFAULT_REPLICAS
    newSetting = "\"settings\": {\"number_of_shards\": %s, \"
number_of_replicas\": %s}" % (number_of_shards, number_of_replicas)
    return newSetting
def getMapping(index, host, username="", password=""):
    endpoint = "/" + index + "/_mapping"
    indexMapping = httpGet(host, endpoint, username, password)
    print index + " 原始mapping如下: \n" + indexMapping
    mappingDict = json.loads(indexMapping)
    mappings = json.dumps(mappingDict[index]["mappings"])
newMapping = "\"mappings\" : " + mappings
    return newMapping
def createIndexStatement(oldIndexName):
    settingStr = getSettings(oldIndexName, oldClusterHost, oldCluster
UserName, oldClusterPassword)
    mappingStr = getMapping(oldIndexName, oldClusterHost, oldCluster
UserName, oldClusterPassword)
    createstatement = "{\n" + str(settingStr) + ",\n" + str(mappingStr
) + "\n}"
    return createstatement
def createIndex(oldIndexName, newIndexName=""):
    if (newIndexName == "")
         newIndexName = oldIndexName
    createstatement = createIndexStatement(oldIndexName)
    print "新索引 " + newIndexName + " 的setting和mapping如下: \n" +
createstatement
    endpoint = "/" + newIndexName
    createResult = httpPut(newClusterHost, endpoint, createstatement,
newClusterUser, newClusterPassword)
    print "新索引 " + newIndexName + " 创建结果: " + createResult
## main
```

```
indexList = getIndices(oldClusterHost, oldClusterUserName, oldCluster
Password)
systemIndex = []
for index in indexList:
    if (index.startswith(".")):
        systemIndex.append(index)
    else :
        createIndex(index, index)
if (len(systemIndex) > 0) :
    for index in systemIndex:
        print index + " 或许是系统索引, 不会重新创建, 如有需要, 请单独处理~"
```

数据迁移

以下提供了三种数据迁移的方式供您参考。您可以根据迁移的数据量大小以及具体的操作情况,选 择合适的方式进行数据迁移。

🕛 注意:

- ·为保证数据迁移前后的一致性,需要上游业务停止旧集群的写操作,读服务才可以正常进行。
 迁移完毕后,直接切换到新集群进行读写。如果不停止写操作可能会存在迁移前后数据不一致
 的情况。
- ・使用下述方案迁移时,如果是通过IP + Port的方式访问旧集群,则必须在新集群的yml文件
 中配置reindex白名单(加入旧集群的IP地址),例如reindex.remote.whitelist: 1.1
 .1.1:9200,1.2.*.*:*。

·如果使用域名访问,则不允许通过http://host:port/path这种带path的形式访问。

・数据量小。

使用reindex.sh脚本。

```
#!/bin/bash
# file:reindex.sh
indexName="你的索引名"
newClusterUser="新集群用户名"
newClusterPass="新集群密码'
newClusterHost="新集群host"
oldClusterUser="老集群用户名"
oldClusterPass="老集群密码"
# 老集群host必须是[scheme]://[host]:[port], 例如http://10.37.1.1:9200
oldClusterHost="老集群host"
curl -u ${newClusterUser}:${newClusterPass} -XPOST "http://${
newClusterHost}/_reindex?pretty" -H "Content-Type: application/json"
 -d'{
"source": {
"comote
         "remote": {
              "host": "'${oldClusterHost}'",
"username": "'${oldClusterUser}'",
"password": "'${oldClusterPass}'"
         },
"index": "'${indexName}'",
         "query": {
              "match_all": {}
```

·数据量大、无删除操作、有更新时间。

数据量较大且无删除操作时,可以使用滚动迁移的方式,减少停止写服务的时间。滚动迁移需要 有一个类似于更新时间的字段代表新数据的写时序。可以在数据迁移完成后,再停止写服务,快 速更新一次。即可切换到新集群,恢复读写。

```
#!/bin/bash
# file: circleReindex.sh
# CONTROLLING STARTUP:
# 这是通过reindex操作远程重建索引的脚本, 要求:
# 1. 新集群已经创建完索引,或者支持自动创建和动态映射。
# 2.
    新集群必须在yml里配置IP白名单 reindex.remote.whitelist: 172.16.123
.*:9200
# 3. host必须是[scheme]://[host]:[port]
USAGE="Usage: sh circleReindex.sh <count>
      count: 执行次数, 多次(负数为循环)增量执行或者单次执行
Example:
        sh circleReindex.sh 1
        sh circleReindex.sh 5
        sh circleReindex.sh -1"
indexName="你的索引名"
newClusterUser="新集群用户名"
newClusterPass="新集群密码"
oldClusterUser="老集群用户名"
oldClusterPass="老集群密码"
## http://myescluster.com
newClusterHost="新集群host"
# 老集群host必须是[scheme]://[host]:[port],例如http://10.37.1.1:9200
oldClusterHost="老集群host"
timeField="更新时间字段"
reindexTimes=0
lastTimestamp=0
curTimestamp=`date +%s`
hasError=false
function reIndexOP() {
    reindexTimes=$[${reindexTimes} + 1]
    curTimestamp=`date +%s
    ret=`curl -u ${newClusterUser}:${newClusterPass} -XPOST "${
newClusterHost}/_reindex?pretty" -H "Content-Type: application/json"
 -d '{
        "source": {
            "remote": {
               "host": "'${oldClusterHost}'",
               "username": "'${oldClusterUser}'",
"password": "'${oldClusterPass}'"
           },
"index": "'${indexName}'",
            "query": {
                "range" : {
                    "'${timeField}'" : {
                        "gte" : '${lastTimestamp}',
                       "lt" : '${curTimestamp}'
                   }
               }
```

```
}
       },
"dest": {
           "index": "'${indexName}'"
       }
   }'`
    lastTimestamp=${curTimestamp}
   echo "第${reindexTimes}次reIndex,本次更新截止时间 ${lastTimestamp
} 结果: ${ret}"
    if [[ ${ret} == *error* ]]; then
        hasError=true
       echo "本次执行异常,中断后续执行操作~~,请检查"
   fi
}
function start() {
   ## 负数就不停循环执行
   if [[ $1 -lt 0 ]]; then
       while :
       do
           reIndex0P
       done
   elif [[ $1 -gt 0 ]]; then
       k=0
       while [[ k -lt $1 ]] && [[ ${hasError} == false ]]; do
           reIndex0P
           let ++k
       done
   fi
}
## main
if [ $# -lt 1 ]; then
   echo "$USAGE"
   exit 1
fi
echo "开始执行索引 ${indexName} 的 ReIndex操作"
start $1
echo "总共执行 ${reindexTimes} 次 reIndex 操作"
```

数据量大、无删除操作、无更新时间。

当数据量较大,且索引的mapping中没有定义更新时间字段时,需要由上游业务修改代码添加 更新时间字段。添加完成后可以先将历史数据迁移完,然后再使用上述的第二种方案。

```
#!/bin/bash
# file:miss.sh
indexName="你的索引名"
newClusterUser="新集群用户名"
newClusterPass="新集群密码
newClusterHost="新集群host"
oldClusterUser="老集群用户名"
oldClusterPass="老集群密码"
# 老集群host必须是[scheme]://[host]:[port],例如http://10.37.1.1:9200
oldClusterHost="老集群host"
timeField="updatetime"
curl -u ${newClusterUser}:${newClusterPass} -XPOST "http://${
newClusterHost}/_reindex?pretty" -H "Content-Type: application/json"
-d '{
    "source": {
        "remote": {
            "host": "'${oldClusterHost}'",
            "username": "'${oldClusterUser}'"
            "password": "'${oldClusterPass}'"
```

・不停止写服务。

敬请期待。



您也可以使用Logstash进行数据迁移,详情请参见 Logstash迁移Elasticsearch数据方法解读。

常见问题

```
・问题:执行curl命令时提示{"error":"Content-Type header [application/x-www-
form-urlencoded] is not supported","status":406}。
```

解决方法:可以在curl命令中添加-H "Content-Type: application/json"参数重试。

```
// 获取老集群中所有索引信息,如果没有权限可去掉"-u user:pass"参数,
oldClusterHost为老集群的host, 注意替换。
 curl -u user:pass -XGET http://oldClusterHost/ cat/indices | awk
 '{print $3}'
 // 参考上面返回的索引列表,获取需要迁移的指定用户索引的setting和mapping,注
意替换indexName为要查询的用户索引名。
 curl -u user:pass -XGET http://oldClusterHost/indexName/_settings,
_mapping?pretty=true
 // 参考上面获取到的对应索引的_settings,_mapping信息、在新集群中创建对应索
引,索引副本数可以先设置为0,用于加快数据同步速度,数据迁移完成后再重置副本数为1
0
 //其中newClusterHost是新集群的host. testindex是已经创建的索引名.
testtype是对应索引的type。
 curl -u user:pass -XPUT http://<newClusterHost>/<testindex> -d '{
   "testindex" : {
       "settings": {
          "number_of_shards": "5", //假设老集群中对应索引的shard数是5
个
          "number_of_replicas": "0" //设置索引副本为0
        }
       },
       "mappings" : { //假设老集群中对应索引的mappings配置如下
          "testtype" : {
              "properties" : {
                 "uid" : {
                     "type" : "long"
                 },
```

·问题:数据同步速度太慢。

解决方法:如果单索引数据量比较大,可以在迁移前将目的索引的副本数设置为0,刷新时间 为-1。待数据迁移完成后,再更改回来,这样可以加快数据同步速度。

```
1 说明:
```

本文档部分内容参考了官方文档。