

## ArkControl 搭建部署

### 检查是否需要安装 YUM 源

#### 安装 YUM 源介绍

Ark-control 部署时需要安装一些依赖，假如没有外网的话或者 Yum 源，需要搭建 Arkcontrol 提供的 yum 源。以保证正常安装。

#### Ark-control-Yum 部署

- 1、保证 Arkcontrolserver 和 yum server 部署在同一台机器
- 2、在即将安装 Arkcontrol server 机器的任意目录下，解压 ark\_yum\_repos.tar-gz (tar -xf ark\_yum\_repos.tar.gz)
- 3、执行 ./arkyum\_server.sh start
- 4、查看运行状态 (默认端口 6033)

```
[root@zsh01 ark_yum_repos]# ps -fe |grep SimpleHTTPServer
root      11443      1  0 16:15 pts/0    00:00:00 python -m SimpleHTTPServer 6033
```

#### Ark-control 安装前

##### 关闭防火墙

```
systemctl stop firewalld.service
systemctl disable firewalld.service
```

##### 关闭 selinux

```
sestatus

setenforce 0
```

**设置时区为东八区 (部署时检查配置, 如果配置库和 control server 不在一个机器, 确保时区一致, 且为本地时区)**

```
[ark@zsh01 log]$ date -R #查看是否为东 8 区
Mon, 06 Jan 2020 16:01:30 +0800

[root@zsh02 ~]# ll /etc/localtime #查看是否为上海
/etc/localtime -> ../usr/share/zoneinfo/Asia/Shanghai

[ark@zsh01 log]$ tzselect
1 [ark@zsh01 log]$ tzselect
2 Please identify a location so that time zone rules can be set correctly.
3 Please select a continent or ocean.
4 1) Africa
5 2) Americas
6 3) Antarctica
7 4) Arctic Ocean
8 5) Asia
9 6) Atlantic Ocean
10 7) Australia
11 8) Europe
12 9) Indian Ocean
13 10) Pacific Ocean
14 11) none - I want to specify the time zone using the Posix TZ format.
15 #? 5
16 Please select a country.
17 1) Afghanistan 18) Israel 35) Palestine
18 2) Armenia 19) Japan 36) Philippines
19 3) Azerbaijan 20) Jordan 37) Qatar
20 4) Bahrain 21) Kazakhstan 38) Russia
21 5) Bangladesh 22) Korea (North) 39) Saudi Arabia
22 6) Bhutan 23) Korea (South) 40) Singapore
23 7) Brunei 24) Kuwait 41) Sri Lanka
```

```
24  8) Cambodia          25) Kyrgyzstan        42) Syria
25  9) China             26) Laos              43) Taiwan
26 10) Cyprus            27) Lebanon           44) Tajikistan
27 11) East Timor       28) Macau             45) Thailand
28 12) Georgia          29) Malaysia          46) Turkmenistan
29 13) Hong Kong        30) Mongolia          47) United Arab Emirates
30 14) India             31) Myanmar (Burma)   48) Uzbekistan
31 15) Indonesia        32) Nepal             49) Vietnam
32 16) Iran             33) Oman              50) Yemen
33 17) Iraq             34) Pakistan
34 #? 9
35 Please select one of the following time zone regions.
36 1) Beijing Time
37 2) Xinjiang Time
38 #? 1
39
40 The following information has been given:
41
42     China
43     Beijing Time
44
45 Therefore TZ='Asia/Shanghai' will be used.
46 Local time is now:   Mon Jan  6 16:01:14 CST 2020.
47 Universal Time is now: Mon Jan  6 08:01:14 UTC 2020.
48 Is the above information OK?
49 1) Yes
50 2) No
51 #? 1
52
53 You can make this change permanent for yourself by appending the line
54     TZ='Asia/Shanghai'; export TZ
55 to the file '.profile' in your home directory; then log out and log in again.
56
57 Here is that TZ value again, this time on standard output so that you
```

```
58 can use the /bin/tzselect command in shell scripts:
59 Asia/Shanghai
60 [ark@zsh01 log]$ date -R
61 Mon, 06 Jan 2020 16:01:30 +0800
```

## 预留

## 部署 Ark-control

### 解压安装包

```
tar -xf arkcontrol-v2.7.tar.gz
```

```
1 [root@zsh01 arkcontrol-v2.7]# ll
2 total 1084612
3 -rw-r--r--. 1 root root 1095225286 Dec 31 18:51 arkcontrol.tar.gz
4 -rwxr-xr-x. 1 root root 15393631 Dec 31 18:44 arkdeploy #安装程序
5 drwxr-xr-x. 5 root root 4096 Dec 25 14:47 arksentinel
6 drwxr-xr-x. 4 root root 4096 Dec 2 18:58 dependency
7 -rw-r--r--. 1 root root 340 Dec 31 18:44 deploy.yaml
8 -rw-r--r--. 1 root root 1076 Dec 2 18:58 README.md
```

### 执行安装脚本

```
./arkdeploy
```

```
1 [root@zsh01 arkcontrol-v2.7]# ./arkdeploy
2 -----
3 | Welcome to the World of Arkcontrol! |
4 -----
5 欢迎使用 Arkcontrol 产品...
6 内存不足 4G,可能有未知错误,建议更换机器,是否继续? [y/N]y
7 创建[mysql, ark]用户...
```

- 8 Arkcontrol Deploy 需要使用 2 个相邻端口, 默认前后端: [9527/9528]
- 9 Arkcontrol Server 需要使用 2 个相邻端口, 默认前后端: [13306/13307]
- 10 请确保端口未被占用, 或者是否需要手动指定? [y/N]N
- 11 默认安装目录 (自动创建): /data, 是否手动指定: [y/N]N
- 12 安装目录已选择: /data/arkcontrol
- 13 - 正在拷贝文件至安装目录.
- 14 - 文件拷贝成功...
- 15 - 安装过程可能需要花费较长时间, 请耐心等待...
- 16 - 开始安装 Yum 依赖...
- 17 - numactl 安装成功
- 18 - rsync 安装成功
- 19 - screen 安装成功
- 20 - libaio 安装成功
- 21 - libaio-devel 安装成功
- 22 - perl 安装成功
- 23 - perl-devel 安装成功
- 24 - perl-Time-HiRes 安装成功

- 25 - perl-DBD-MySQL 安装成功
- 26 - 开始配置 Nginx...
- 27 - Nginx 配置成功...
- 28 Deploy `Server` 启动中...
- 29 请访问 <http://10.0.0.231:9527> 继续执行安装...

## 打开 web 管理界面

### 点击下载依赖包

#### Arkcontrol部署

##### Arkcontrol产品介绍与依赖

**介绍:** Arkcontrol产品是极数云舟独立自研的一套全面解决MySQL日常运维和集群管理的自动化平台，它提供MySQL的集群管理、实例管理、监控备份、日常巡检、参数建议等多种功能，同时可以提供对公有云数据库的API调用管理，可以帮助所有使用MySQL的企业提升工作效率，降低故障时间，提升数据安全。

**依赖工具包:**

- 1、MySQL数据库，用于存储相关资源信息、配置数据、巡检数据、慢查询等数据。
- 2、监控数据存储（ES），用于监控相关数据库存储。
- 3、其他工具，用于备份、分析、以及产品依赖的其他相关工具包

下载依赖包

## web 界面安装处理

### 数据库及 ES 配置 (Arkcontrol 采集依赖 ES)

#### 配置库

MySQL:  搭建数据库 (推荐)  已有数据库(5.7+) 升级请选此项

密码:

端口:

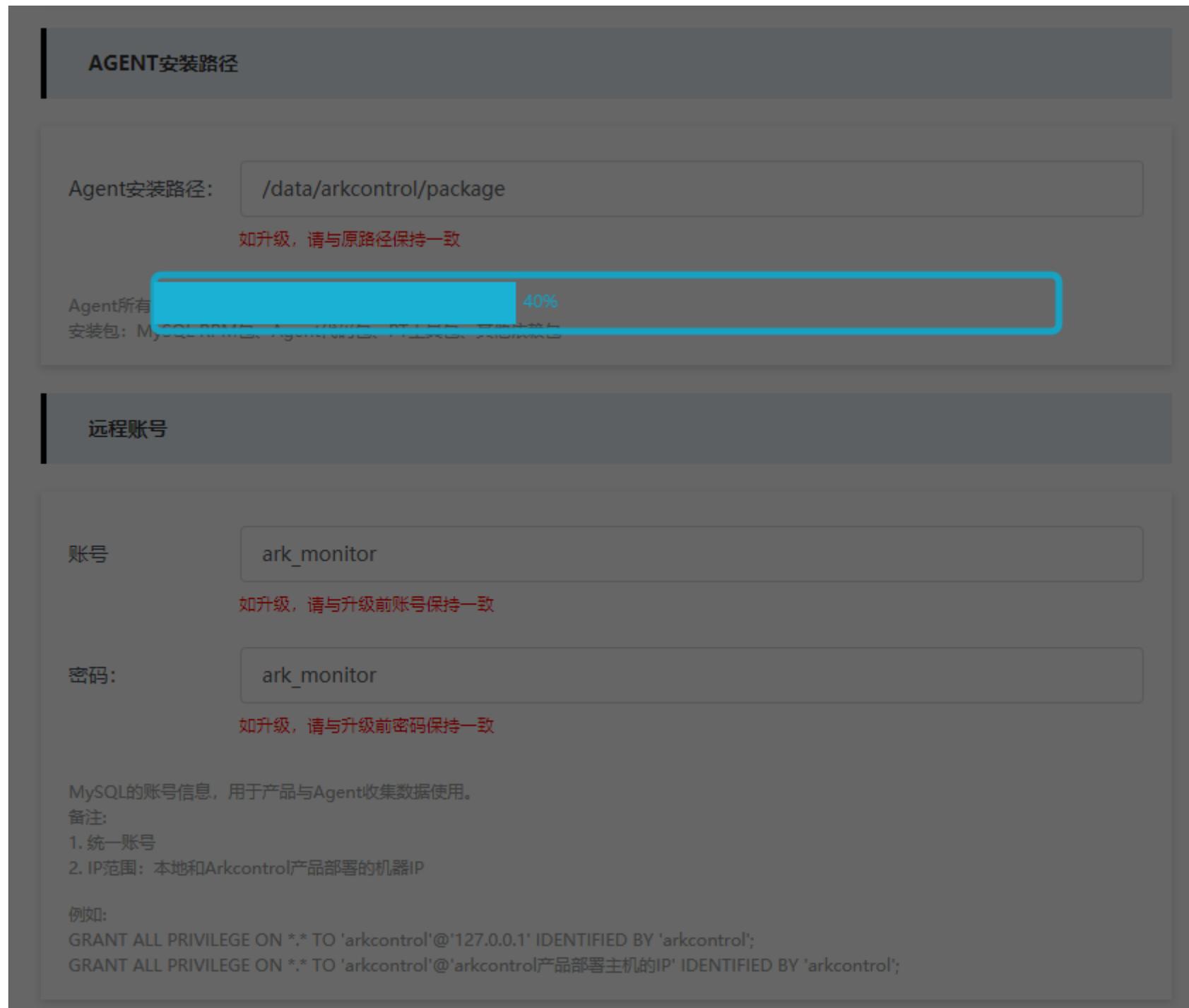
#### ES配置

ES:  搭建ES (推荐)  已有ES(5.6+) 升级请选此项

端口:

Arkcontrol数据采集存储依赖ES

## Agent 安装及远程帐号



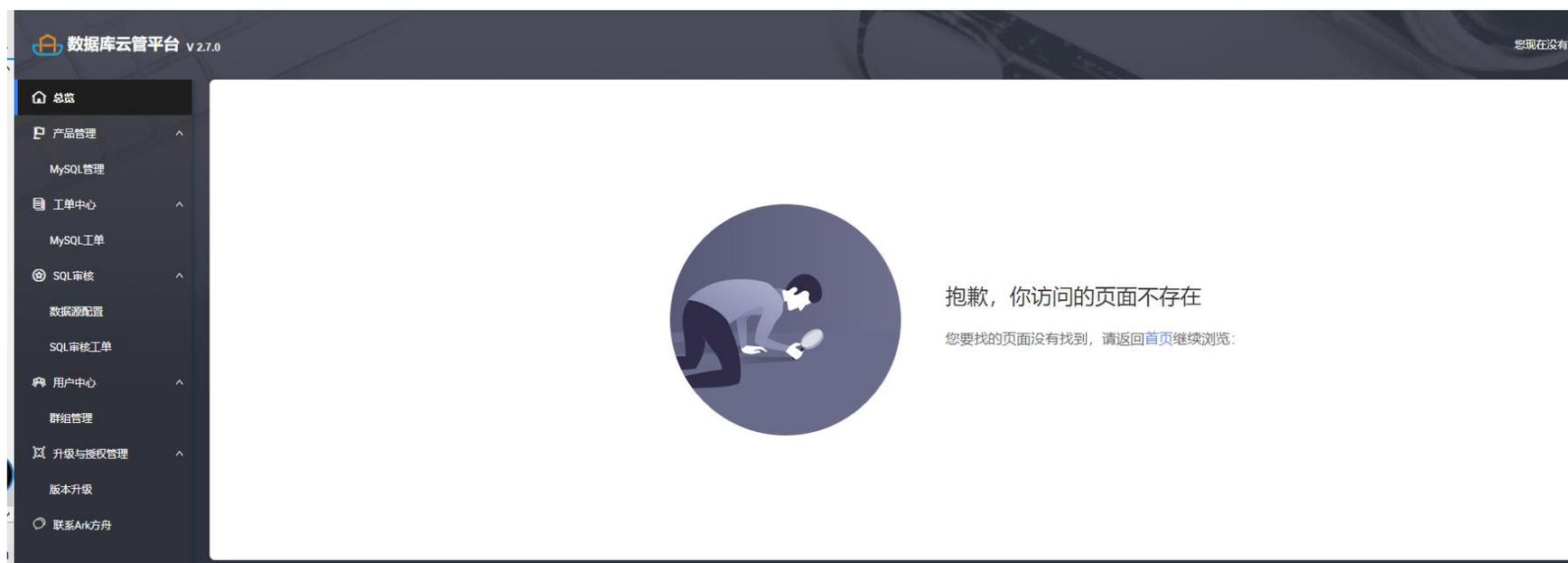
### 同时 Shell 界面会有提示

- 1 请勿退出本程序, 安装结束后会自动退出...
- 2 - 开始安装 pt 工具...
- 3 - pt 工具安装成功...
- 4 - 开始安装 Arkcontrol 所需配置库...
- 5 - 配置库安装成功
- 6 - 开始初始化配置库
- 7 - 配置库初始化成功
- 8 - 安装 arkit.so 成功
- 9 - 配置监控...可能需要较长时间, 请耐心等待
- 10 -- 开始配置 elasticsearch...
- 11 -- elasticsearch 配置成功...
- 12 -- 开始配置 grafana...
- 13 -- grafana 配置成功...
- 14 - 监控配置完毕
- 15 - 启动 Arkcontrol Server...
- 16 - 启动成功, 等待页面自动跳转...

## 部署成功后跳转页面

<http://10.0.0.231:13306/#/user/login>

## 通过注册用户，进入到首页



## ArkControl 启动及管理

### 启动、关闭

#### Arkserver: Arkcontrol 核心服务

```
cd /data/arkcontrol
```

```
arkcontrol.sh start/stop #启停脚本
```

```
[root@zsh01 ark_yum_repos]# ps -fe |grep ark[server,alert] #检测 server 和 alert 服务是否正常启动
```

```
root      9498  9493  0 15:27 pts/1    00:00:01 ./arkserver start
root      9503  9502  0 15:27 pts/1    00:00:00 ./arkalert start
```

### MySQL: Arkcontrol 配置库, 存储相关资源信息、配置数据、巡检数据、慢查询

```
/data/arkcontrol/software/mysql/arkdb/3306  
./mysql.server -P 3306 start / stop
```

切换用户到 ark 才可以使用 MySQL 命令

```
su - ark
```

登陆 MySQL 库

```
mysql -uroot -p -S /data/arkcontrol/software/mysql/arkdb/3306/socket/mysql.sock
```

### ES: Arkcontrol 监控数据存储

切换到 ark 用户, 才可以使用 ES 功能

```
su - ark
```

启动停止

```
/data/arkcontrol/software/elasticsearch/start.sh  
/data/arkcontrol/software/elasticsearch/stop.sh
```

测试

```
[ark@zsh01 elasticsearch]$ curl 10.0.0.231:9200  
{  
  "name" : "FPGAkJd",  
  "cluster_name" : "ark_es",  
  "cluster_uuid" : "DZnG3au4RmO2_zGxXdu7_w",  
  "version" : {  
    "number" : "5.6.9",  
    "build_hash" : "877a590",  
    "build_date" : "2018-04-12T16:25:14.838Z",  
    "build_snapshot" : false,  
    "lucene_version" : "6.6.1"  
  },  
}
```

```
"tagline" : "You Know, for Search"  
}
```

### Grafana: 旧版监控数据展示 (2.0 废弃)

用户名密码

```
admin
```

```
admin
```

切换到 ark 用户, 才可以使用 grafana 功能

```
su - ark
```

启动停止

```
/data/arkcontrol/software/grafana/start.sh
```

```
/data/arkcontrol/software/grafana/stop.sh
```

登陆: 默认端口 3000

如果有数据说明 grafana 没问题

检查 ES, 重启 ES, 出图

## 更新, 升级, 降级

### 确保 arkserver 和 arkalert 进程已经停止

```
arkcontrol.sh stop
```

计划任务里面可能有定时拉起的计划任务, 关掉, 保证 arkalert 和 arkserver 全部停了

```
arkalert.sh stop
```

```
arkserver.sh stop
```

### 确保 Arkcontrol 存储服务 and 配置服务正常运行，不需要停止，且必须保证服务正常运行

MySQL: 3306

ES: 9200, 确保 es 的 IP 和端口号一致

### 安装如何采用同一个目录，需要选择新目录进行安装

- 1、也可以采用原目录，Arkcontrol 只会将二进制文件覆盖，但是不会覆盖 MySQL 和 ES 的数据
- 2、确保版本可以平滑升级，否则表结构不一致导致页面无法访问
- 3、删除 arkcontrol 文件夹，解压需要的包，执行可执行文件部署即可。

### web 界面操作注意事项

- 1、配置库选择已有数据库，并输入配置库地址和 IP，帐号密码进行连接测试
- 2、ES 选择已有 ES，因为 ES 有可能是集群模式，IP 可选任意一台即可，默认端口 9200，并进行连接测试（确保 IP 和端口号一致）
- 3、Agent 安装路径务必保证与之前版本保持一致（若用户遗忘之前的配置，可进入配置库中，访问 `arkmate.init_mate_conf` 中的 `source_dir` 字段）
- 4、远程帐号密码务必保证与之前版本保持一致（若遗忘，访问配置库 `arkmate.init_mate_conf` 的 `agent_userheagent_pass`）

## ES 安装失败问题

### 原因

部分 Centos `sysctl -p` 报错 `unknown key`，要重新部署监控

## 解决

执行/data/arkcontrol/conf/monitor/ini\_sys\_properties.sh

su - ark

./start.sh #启动 ES

获取 initMonitor.tar.gz 解压

./init\_monitor \

-es\_ip=es 集群中任意一台 ip \

-es\_httpport=es 的 http port \

-es\_user=选填 \

-es\_passwd=选填 \

-grafana\_ip=Arkcontrol 所在机器 ip \

-grafana\_port=默认 3000 \

-grafana\_user=默认 admin \

-grafana\_passwd=默认 admin

## ES 运维插件

实际项目部署中,经常会遇到 ES 部署或者监控的问题,所以提供一个 cerebro 插件,方便对 ES 进行观察和运维.

插件安装方式:

```
tar -zxvf cerebro.tar.gz && ./start.sh
```

确保机器上安装有 java.访问 9000 端口来查看服务是否正常.

## 监控图没有数据问题

- 1、登陆 3000 端口，查看 grafana 是否有图，如果没图说明 ES 有问题，可以重启 ES (记得 su - ark)
- 2、如果 3000 有图，arkcontrol 没图，查看 arkcontrol 时区是否有问题，data -R dzselect 9511
- 3、如果时区一致，无法出图，重启 arkcontrol server 即可

## ES 清空部分数据

### 查看单签 ES 中的索引列表

```
curl -XGET 'http://10.0.0.231:9200/_cat/indices?v'
```

```
1 [root@zsh01 ~]# curl -XGET 'http://10.0.0.231:9200/_cat/indices?v'
```

2	health	status	index	uuid	pri	rep	docs.count	docs.deleted	store.size	pri.store.size
3	yellow	open	ark-monitor-mysql-2020.01	3FbFowrqRi27vkJ7dWlw9A	3	1	433362	0	59.1mb	59.1mb
4	yellow	open	ark-monitor-machine-2020.01	kgwMPchASf2cwLFH31-wIg	3	1	5625581	0	901.7mb	901.7mb
5	yellow	open	slowquery-2020.01	-CEleLXcR9mP739icR6Fyw	3	1	2753458	0	986.3mb	986.3mb

第三列是索引名字

### 查看集群健康状态

```
curl 'http://10.0.0.231:9200/_cluster/health?pretty'
```

```
1 [root@zsh01 ~]# curl 'http://10.0.0.231:9200/_cluster/health?pretty'
```

```
2 {
```

```
3   "cluster_name" : "ark_es",
```

```
4   "status" : "yellow", #yellow 说明当前集群分片太多,red 说明有损坏的分片
```

```
5   "timed_out" : false,
```

```
6   "number_of_nodes" : 1,
```

```
7   "number_of_data_nodes" : 1,
```

```
8     "active_primary_shards" : 9,  
9     "active_shards" : 9,  
10    "relocating_shards" : 0,  
11    "initializing_shards" : 0,  
12    "unassigned_shards" : 9,  
13    "delayed_unassigned_shards" : 0,  
14    "number_of_pending_tasks" : 0,  
15    "number_of_in_flight_fetch" : 0,  
16    "task_max_waiting_in_queue_millis" : 0,  
17    "active_shards_percent_as_number" : 50.0  
18 }
```

### 将某个 open 的索引 close 掉

```
curl -XPOST 'http://10.0.0.231:9200/slowquery-2020.0/_close'
```

### 删除某个索引

```
curl -Xdelete 'http://10.0.0.231:9200/index_name'
```

### 查看 ES 的 template

```
curl 'http://10.0.0.231:9200/_template'
```

### ES xpack 认证

部分用户 ES 集群开启了 xpack 的权限认证，需要保证 ES 的帐号拥有[manager,read,write,create\_index]等四种权限，原因：

read: 查询数据

write: 写入数据

create\_index: 自动创建索引

manager: 创建索引模板

## 监控数据不全, 15 分钟, 1 小时监控面板无数据

- 1、tzselect
- 2、echo "TZ='Asia/Shanghai'; export TZ" >> ~/.bash\_profile
- 3、~/.bash\_profile

```
ll /etc/localtime
```

```
timedatectl list-timezones |grep Shanghai #查找中国时区的完整名称
```

```
Asia/Shanghai
```

```
timedatectl set-timezone Asia/Shanghai #其他时区以此类推
```

数据库时间

```
select now();
```

```
set global time_zone='+8:00';
```

```
select now();
```

时区调整完毕, 重启 arkcontrol server 即可

```
./stop.sh
```

```
./start.sh
```

## pxc 安装版本不匹配

如果要更新 Galera 包,只需要将新版本包 [ark-galera57-5.7.26-1.el6.x86\\_64.rpm](#) 放到 arkcontrol 机器上的 /data/arkcontrol/package/下

登录到 arkcontrol 配置库执行

```
update arkmate.mate_resource_name set resource_name='ark-galera57-5.7.26-1.el6.x86_64.rpm' where package=1 and  
server_type=13 and version='5.7';
```

新添加的部署任务即使用新包

### arkcontrol 高可用中心配置时直接报错。登陆配置库执行 SQL

```
insert into arksentinel.sentinel_config(raft_config,config_cluster,notifier_config) values('{}','{}','{}');
```

执行 SQL 后页面配置高可用哨兵正常

## ArkControl 部署不同版本 MySQL

### 升级 MySQL 版本

将新版本包例如 `ark-mysql80-8.0.16-1.el7.x86_64.rpm` 放在 arkcontrol 的 `/data/arkcontrol/package` 目录下

```
1 [ark@zsh01 package]$ ll ark-mysql*  
2 -rw-r--r--. 1 root root 28221992 Dec 2 18:58 ark-mysql56-5.6.36-1.el6.x86_64.rpm  
3 -rw-r--r--. 1 root root 41388244 Dec 2 18:58 ark-mysql57-5.7.20-1.el6.x86_64.rpm  
4 -rw-r--r--. 1 root root 40829428 Dec 2 18:58 ark-mysql57-5.7.26-1.el6.x86_64.rpm  
5 -rw-r--r--. 1 root root 42376020 Dec 2 18:58 ark-mysql80-8.0.16-1.el7.x86_64.rpm
```

### 登陆到 arkcontrol 配置库执行

```
update arkmate.mate_resource_name set resource_name='ark-mysql80-8.0.16-1.el7.x86_64.rpm' where package=1  
and server_type=0 and version='8.0';
```

新添加的部署任务即使用新的包

### 更新 Galera 包

将新版本包例如 `ark-galera57-5.7.26-1.el6.x86_64.rpm` 放在 `arkcontrol` 的 `/data/arkcontrol/package` 目录下

```
1 [ark@zsh01 package]$ ll ark-galera*
2 -rw-r--r--. 1 root root 34305048 Dec  2 18:58 ark-galera56-5.6.43-1.el6.x86_64.rpm
3 -rw-r--r--. 1 root root 55095800 Dec  2 18:58 ark-galera57-5.7.26-1.el6.x86_64.rpm
```

登陆到 `arkcontrol` 配置库执行

```
update arkmate.mate_resource_name set resource_name='ark-galera57-5.7.26-1.el6.x86_64.rpm' where package=1
and server_type=13 and version='5.7';
```

新添加的部署任务即使用新的包

## 总览界面

### 主机



## MySQL



## Redis



## 产品管理

### MySQL 管理

## 基本信息 (查看当前所有 MySQL 实例大致信息)

### 基本信息

集群名称:	<input type="text" value="请输入集群名称"/>	架构类型:	<input type="text" value="请选择架构"/>	实例名称:	<input type="text" value="请输入实例名称"/>
实例版本:	<input type="text" value="请选择实例版本"/>	实例归属组:	<input type="text" value="请选择实例归属组"/>	运行状态:	<input type="text" value="请选择"/>
到期时间:	<input type="text" value="请选择日期"/>				

[搜索](#)[重置](#)

集群名称	实例归属组	实例名称	角色	架构类型	运行状态	到期时间	操作
三类项目	DB部门	10.0.0.232_3307	master	MS Cluster	实例正常	2030-01-01	<a href="#">详情</a>
三类项目	DB部门	10.0.0.46_3307	slave	MS Cluster	实例正常	2030-01-01	<a href="#">详情</a>
三类项目	DB部门	10.0.0.233_3307	slave	MS Cluster	实例正常	2030-01-01	<a href="#">详情</a>

### 查看详情-基本信息

可以查看到实例的大致信息以外的基本信息，MySQL 版本，端口，集群类型，当前实例角色、配置信息、备份信息等。

## | 基本信息

## 基本信息

## 参数变更

## 巡检

## 慢查询

## 监控告警

## 基本信息

集群名称:	三类项目	实例归属组:	DB部门	架构类型:	MS Cluster
实例版本:	Percona Server (GPL) 5.7.26-29-log	实例名称:	10.0.0.232_3307	端口:	3307
角色:	master	可用区:	北1机房		

## 配置信息

CPU:	1 core	可维护时间段:	1:00~2:00	存储类型:	SSD
到期时间:	2030-01-01				

## 运行状态

创建时间:	2020-01-09 11:06:50	集群备份状态:	-
-------	---------------------	---------	---

## 使用量统计

集群备份使用量: 16791206423

## 备份设置

是否开启:	OFF	预计下次备份时间:	-
-------	-----	-----------	---

[查看详情-参数变更](#)

可以修改 MySQL 的一些基本参数，有些参数可以动态修改，有些参数不可以（根据 **需要** 和 **不需要** 来判断）

## | 基本信息

基本信息

参数变更

巡检

慢查询

监控告警

当前参数 变更历史

参数名	参数默认值	参数运行值	是否重启	可修改参数值
character_set_server	utf8mb4	utf8mb4	需要	[utf8mb4   utf8   latin1]
collation_server	utf8mb4_general_ci	utf8mb4_general_ci	需要	[utf8mb4_general_ci   utf8_general_ci]
back_log	50	50	需要	[1-65535]
innodb_strict_mode	ON	ON	需要	[ON   OFF]
max_allowed_packet	33554432	33554432	需要	[1024-1073741824]
max_connect_errors	1000000	1000000	需要	[1-18446744073709551615]
interactive_timeout	900	900	需要	[1-28800]
wait_timeout	900	900	需要	[1-31536000]
interactive_timeout	900	900	需要	[1-28800]
skip_name_resolve	ON	ON	不需要	[ON   OFF]

当对参数进行修改的时候，会提一个工单给组负责人进行审批

当前参数 变更历史

变更时间:

请选择日期



参数名	变更前参数值	变更后参数值	是否生效	变更时间
open_files_limit	65535	65534	等待修改	2020-02-01 11:54:46

< 前一页 1 下一页 >

## 查看详情-巡检

**巡检 (根据巡检配置来进行自定义巡检)**

告警可以通过配置中心、用户中心进行配置，将告警发送给组管理员

[巡检](#) [巡检配置](#)

巡检项	巡检时间	状态	详细信息
实例数据文件大小	2020-02-01 10:28:43	正常	⊖
binlog文件大小	2020-02-01 10:28:43	正常	⊖
表自增键使用率	2020-02-01 10:28:43	警告	⊖

巡检信息



集群名: 三类项目  
IP: 10.0.0.232  
port: 3307  
数据库名称: -  
巡检项: 实例数据文件大小  
巡检值: 15.33GB  
状态: 正常  
详细信息: 实例数据文件大小 15699 M

关闭

## 巡检配置 (可以自定义巡检配置项)

巡检 [巡检配置](#)

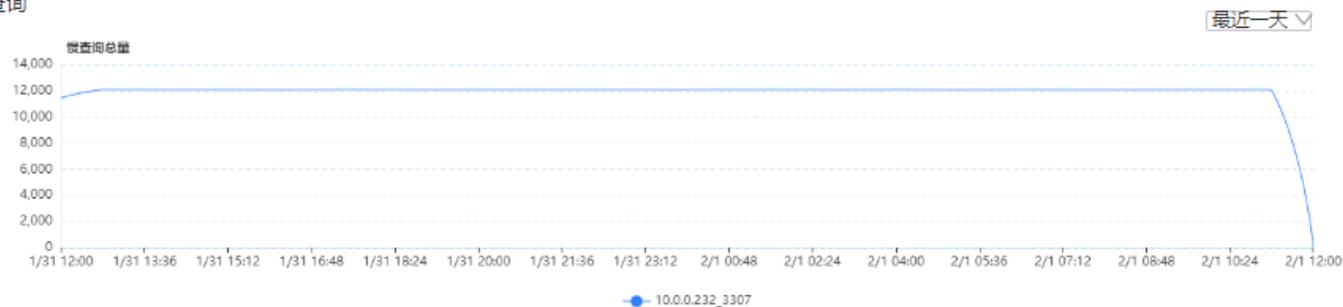
巡检项	巡检阈值	巡检说明
实例数据文件大小	10MB	实例数据文件大小超过阈值时警告
binlog文件大小	1MB	binlog文件大小超过阈值时警告
表自增键使用率	88%	自增键使用率超过阈值时警告
MylSAM表检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
大表检查	0MB	表大小超过阈值时警告
表碎片率	82%	表碎片率超过阈值时警告
表索引个数	1	表索引个数超过阈值时警告
冗余索引检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
从库可写检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
临时表检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
外键检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
没有主键表检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查

[查看详情-慢查询](#)

## 慢查询 (图表展示, 基本信息)

慢查询的数量曲线, 以及 SQL 样本, 执行总时间, 平均时间, 执行次数等。

## 实例慢查询



来源主机:  数据库:  用户名:

时间范围:  排序类型:

来源主机	用户名	数据库	SQL样本	总时间	总次数	最大执行时间	平均时间	操作
127.0.0.1	ark_monitor		select max(time) as ... ax_time from inform...	7.21	8625	0.00	0.00	<a href="#">详情</a> <a href="#">优化建议</a>

## 慢查询详情

查看某个慢查询更加详细的信息，如下图：

**慢查询详情** >

访问来源: 127.0.0.1	数据库: -	用户名: ark_monitor
总时间: 7.21s	总次数: 8625	平均时间: 0.00s
最小时间: 0.00s	最大执行时间: 0.00s	最后出现: 2020-02-01 11:59:49
最大扫描行数: 92	平均扫描行数: 83	最小扫描行数: 82

SQL样本:  
select max(time) as max\_time from information\_schema.PROCESSLIST where command = 'Query'

SQL语句: select max(time) as max\_time from information\_schema.processlist where command = ?

## 慢查询优化建议 (调用小米的 *soar*, 给出优化建议)

优化建议

Query: 18541CF73E488945

★★★★★ 100分

```
SELECT
  authentication_string as PASSWORD, Host, Super_priv
FROM
  mysql. USER
WHERE
  USER = 'e13947744ffe63fd'
```

OK

sql重写建议

```
SELECT
  authentication_string as PASSWORD, Host, Super_priv
FROM
  mysql. USER
WHERE
  USER = 'e13947744ffe63fd';``
```

[查看详情-监控告警](#)

### 监控

可以进行当前 MySQL 的基本信息的查看，如果图表信息展示不完整，或者无数据，查看 ArkControl 启动及管理界面

## | 基本信息

基本信息

参数变更

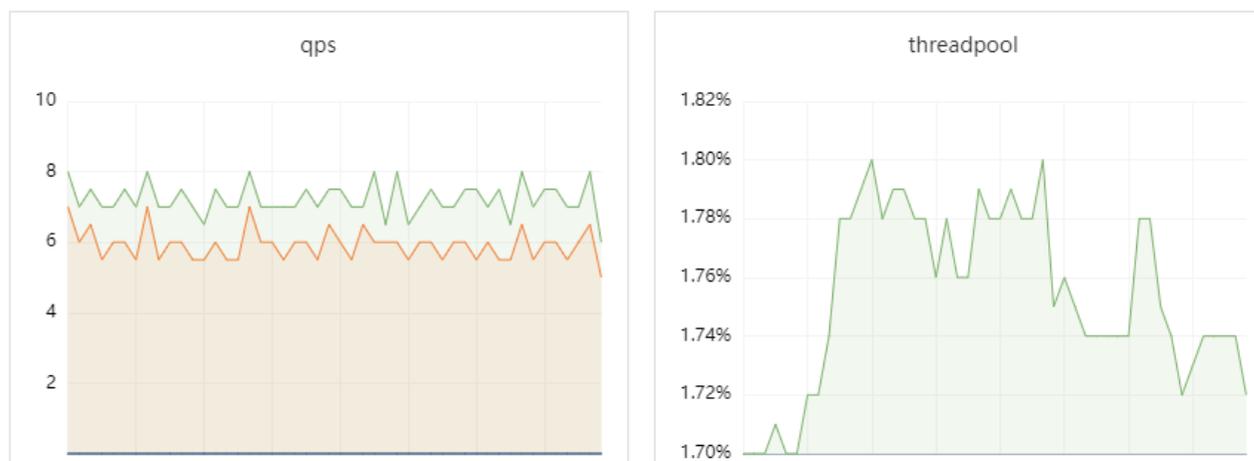
巡检

慢查询

监控告警

监控 告警

时间: 15分钟 1小时 1天 7天 1个月 自定义

**告警**

如果配置了告警模板，会在此处显示告警，通过 [MySQL 管理-监控告警配置](#)

监控 告警

监控项

报警规则

状态

没有数据

**Redis 管理**

## 实例列表及基本信息

## 实例列表

集群名称:  架构类型:  角色:

实例归属组:  可用区:  运行状态:

集群名称	实例名称	角色	架构类型	版本	实例归属组	可用区	运行状态	到期时间	操作
测试	10.0.0.233_6380	slave	主从集群	4.0.14	DB部门	北1机房	在线	2030-12-31	<a href="#">详情</a>
测试	10.0.0.232_6380	slave	主从集群	4.0.14	DB部门	北1机房	在线	2030-12-31	<a href="#">详情</a>
测试	10.0.0.231_6380	master	主从集群	4.0.14	DB部门	北1机房	在线	2030-12-31	<a href="#">详情</a>

## 实例详情

## 基本信息(查看集群更加详细的信息)

## 集群相关信息

## 基本信息

巡检

慢查询

监控告警

## 实例信息

集群名称:	测试	归属组:	DB部门	可用区:	北1机房
实例名称:	10.0.0.233_6380	端口:	6380	角色:	slave
主机名称:	10.0.0.233	架构类型:	主从集群	实例版本:	4.0.14
实例状态:	在线	创建时间:	2020-02-01 17:52:11	到期时间:	2030-12-31
备注:	-				

## 实例配置信息

maxmemory(G):	1	maxmemory-policy:	volatile-lru	AOF:	开启
appendfsync:	-	安装路径:	/data/redis/redis6380		

## 巡检

## 巡检

## 巡检

巡检配置

巡检项:

请选择

搜索

重置

集群名称	实例名称	主机名称	端口	数据大小	巡检结果	巡检时间
------	------	------	----	------	------	------

没有数据

## 巡检配置(进行巡检的配置设置)

### 巡检

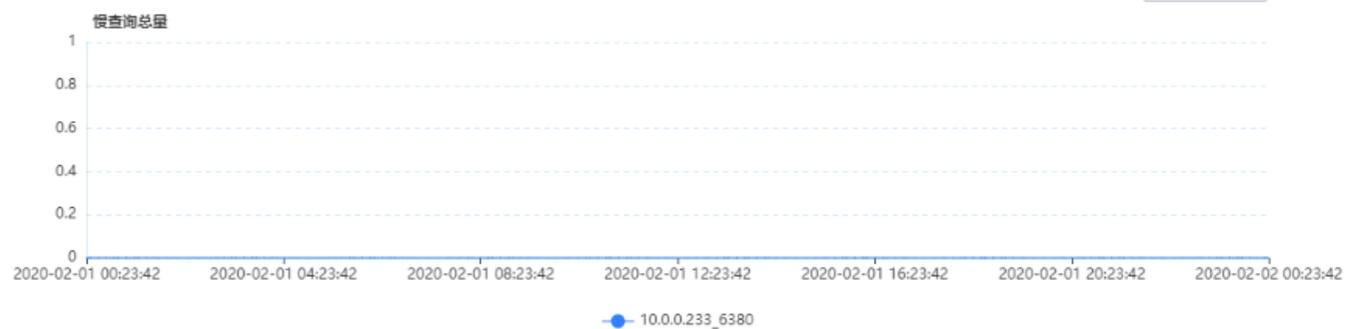
#### 巡检配置

巡检项	巡检阈值	巡检说明
主机时间是否正确	2s 	主机时间是否正确
Swap使用率	0% 	Swap使用超过一定比例
是否关闭内存大页	<input checked="" type="checkbox"/>	内存大页是否关闭 1 表示开启巡检项, 0 表示关闭巡检项
系统内存使用比例	80% 	当系统内存使用比例超过一定阈值时报警
内存碎片率	5 	Redis 实例的内存碎片率超过一定的阈值时报警
从库是否只读	<input checked="" type="checkbox"/>	当实例的slave-read-only 参数为no 时报警 1 代表开启巡检, 0 代表关闭巡检
连接使用百分比	70% 	连接使用百分比

## 慢查询

## 实例慢查询

最近一天 ▾

数据类型: 命令: KEY: 时间范围: 排序:  

类型	命令	语句样本	总次数	总耗时(ms)	平均时间(ms)	操作
----	----	------	-----	---------	----------	----

没有数据

## 监控告警(显示实例的监控图表)

### 集群相关信息

基本信息

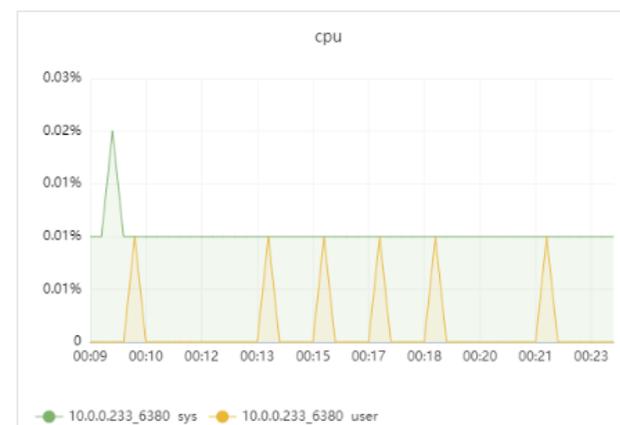
巡检

慢查询

监控告警

### 监控信息

时间: 15分钟 1小时 1天 7天 1个月 自定义



## 工单中心

MySQL 工单 (只提供接口功能, 还无实现具体操作)

## 申请实例

### 申请实例

* 归属组	<input type="text" value="请选择"/>
* 可用区	<input type="text" value="请选择"/>
* 数据库版本	<input type="text" value="请选择"/>
* 架构	<input type="text" value="请选择"/>
* 存储类型	<input type="text" value="请选择"/>
* 规格	<input type="text" value="请选择"/>
* 使用时长(月)	<input type="range" value="1"/> <input type="text" value="1"/>
* 存储空间(GB)	<input type="range" value="1"/> <input type="text" value="1"/>

## 申请数据库

根据实际要求进行数据库配置的选择，然后提交申请，MySQL 属组的管理员进行审批即可。

## 申请数据库

\* 集群名称

\* 数据库名称

\* 预估数据容量(GB)

\* 预估未来两年容量(GB)

\* 预估QPS

读写比  :

实例工单 数据库工单 权限工单 参数变更

工单编号:  集群名称:  申请时间:

	工单编号	集群名称	预估QPS	读写比	预估数据容量(GB)	状态	申请时间	操作
+	1	三类项目	2000	5:1	10	已完成	2020-02-01 12:12:51	

## 申请权限

### 新增帐号

#### 新建帐号

#### 申请权限

申请类型	<input checked="" type="radio"/> 新增账号 <input type="radio"/> 扩展IP <input type="radio"/> 扩展DB <input type="radio"/> 扩展权限
* 集群名称	<input type="text" value="10.0.0.231_3306"/>
* 用户名	<input type="text" value="arktest1"/>
* 数据库名称	<input type="text" value="db1"/>
* 连接范围	<input type="text" value="127.0.0.1"/>
* 权限	<input checked="" type="checkbox"/> SELECT <input checked="" type="checkbox"/> INSERT,UPDATE,DELETE <input type="checkbox"/> CREATE <input type="checkbox"/> DROP <input type="checkbox"/> REPLICATION CLIENT <input type="checkbox"/> REPLICATION SLAVE
SQL与备注	<input type="text" value="SQL与备注"/> 

提交

#### 审批流程

工单编号	归属组	权限类型	申请人	状态	工单处理人	申请时间	操作
2	系统默认组	新增账号	ark_control	审批中		2020-01-08 11:39:24	详情

### 审批详情

#### 基本信息

工单编号:	3	权限类型:	新增账号
申请人:	ark_control	状态:	审批中
驳回理由:	-		

#### 申请信息

集群名称:	db1	用户名:	arktest1
连接范围:	127.0.0.1	权限:	SELECT

[执行](#) [驳回](#) [查询授权信息](#)

### 执行、驳回、查询授权信息

授权语句 ✕

```
待执行语句: GRANT SELECT,INSERT,UPDATE,DELETE ON `db1`.* TO 'arktest1'@'127.0.0.1'  
IDENTIFIED BY PASSWORD  
*'BE1BDEC0AA74B4DCB079943E70528096CCA985F8'
```

### 扩展 IP

#### 查询已有权限

申请权限

申请类型  新增账号  扩展IP  扩展DB  扩展权限

1 查看已有权限

2 选择一个你现在的匹配的账号信息

\* 集群名称 db1

\* 用户名 arktest1

IP列表	密码	数据库	权限
<input type="radio"/>	127.0.0.1	*BE1BDEC0AA74B4DCB079943E7052809 6CCA985F8	db1 CREATE USER 'arktes...

上一步 下一步

### 选择一个你现在匹配的账号信息

V 2.7.0 请求参数错误 您有257个未

申请权限

申请类型  新增账号  扩展IP  扩展DB  扩展权限

1 查看已有权限

2 选择一个你现在的匹配的账号信息

3 提交申请

\* 集群名称 db1

\* 用户名 arktest1

IP列表	密码	数据库	权限
<input checked="" type="radio"/>	127.0.0.1	*BE1BDEC0AA74B4DCB079943E7052809 6CCA985F8	db1 CREATE USER 'arktes...

\* 待扩展IP范围 10.0.0.232

上一步 提交

### 提交申请

审批流程一样

### 扩展 DB

操作和上面一致

### 扩展权限

操作和上面一致

### 查看已申请的工单详情 (其余的工单审批流程一致)

工单编号	归属组	可用区	架构	规格	状态	申请时间	操作
—	2	DB部门	北1机房	MS Cluster	1核 1G	已完成	2020-01-07 16:46:43
申请人:ark_control		归属组:DB部门		状态: 已完成		可用区:北1机房	
存储类型:普通IO		数据库版本: 5.7.26		存储空间(GB): 1			
申请时间: 2020-01-07 16:46:43		更新时间: 2020-01-07 16:46:46		使用时长(月): 1月		备注:	

## Redis 工单

申请集群(提交之后走审批流程)

集群申请

\* 集群名称  3/20

\* 归属组 ? :

\* 可用区 ? :

\* 架构 ? :

\* 使用类型 ? :  Cache  Storage

\* 版本

\* QPS预估

\* 存储空间(GB)

\* 使用时长(月)

提交

回收集群(可以回收集群,然后选择备份或者不备份数据)

集群回收

\* 集群名称

\* 数据备份 ②:  备份  不备份

集群扩缩容

集群扩缩容

集群扩缩容

\* 集群名称  [现有集群信息](#)

\* 变更类型 ②:  扩容  缩容

\* 变更后容量(G) ②:

\* 变更后QPS预估 ②:

## 现有集群信息

▼ 现有集群信息

架构类型: 主从集群 存储类型: Cache  
版本: 4.0.14 总容量(G): 2  
当前使用量(G): 0.06 实例个数: 2

## 查看已申请的工单详情 (其余的工单审批流程一致)

集群申请 **集群回收** 集群扩缩容

归属组:  工单编号:  申请时间:

	工单编号	归属组	集群名称	数据备份	状态	申请时间	操作
+	2	DB部门	测试	<span>备份</span>	<span>审批中</span>	2020-02-02 00:14:36	<input type="button" value="审批"/>
+	1	DB部门	测试	<span>备份</span>	<span>已完成</span>	2020-02-02 00:10:16	

## SQL 审核

### 参数设置

一些 SQL 审核的参数设置, 以及 PTOSC 和 GHOSC 的审核参数设置

[执行选项](#)   [审核参数](#)   [PT-OSC参数](#)   [GH-OSC参数](#)

## 执行选项列表

参数名称	参数值	参数可修改值	操作者	更新时间	参数说明
osc_type	pt_osc 	[build_in_osc,direct_alter,pt_osc,gh_osc]	ark_control	2019-06-11 06:48:31	osc改表 方法
sleep_interval	0 	0-1000	ark_control	2019-06-12 08:32:46	SQL语句执行间隔时间,单位是ms
force_exec	<input type="checkbox"/>	[ON OFF]	ark_control	2019-06-05 08:21:08	执行过程中遇到错误是否继续执行
ignore_warn	<input checked="" type="checkbox"/>	[ON OFF]	ark_control	2019-06-05 08:21:14	执行过程中是否忽略警告信息
backup	<input checked="" type="checkbox"/>	[ON OFF]	ark_control	2019-06-05 08:21:14	是否打开备份, 备份SQL执行后的回滚语句

◀ 1 ▶

## 数据源配置

## 新增数据源

添加数据源 ✕

\* 归属组

\* 数据源名称

\* 数据库地址

\* 数据库端口

\* 数据库用户

\* 数据库密码

\* 同构库  否  是

备注

0/200

## 编辑, 删除数据源

数据源名称	归属组	数据库地址	数据库端口	数据库账号	备注	操作者	更新时间	操作
数据源1	DB部门	10.0.0.40	13307	arkproxy		ark_control	2020-02-01 12:34:16	<input type="button" value="编辑"/> <input type="button" value="删除"/>

## SQL 审核工单

申请工单->预检查, 会生成 SQL 审批

## 申请工单

操作实例

来自平台中  来自配置数据源  手动输入

\* 集群名称

test01

节点 10.0.0.233:3306

提交方式

文本  附件

\* SQL语句

```
create database db1;
```

预检查

错误代码

错误信息

警告

Set charset to one of 'utf8mb4' for table 'db1'.

### 预检查结果 (调用 inception SQL 审核)

检查SQL ×

您的SQL语句存在一定的瑕疵，但不影响执行，您可以返回修改，或者直接去点击【下一步】

错误代码	错误信息	SQL语句	影响行数
<span style="background-color: #ffc107; padding: 2px;">警告</span>	Set charset to one of 'utf8mb4' for table 'db1'.	create database db1	0

< 上一页 1 下一页 >

返回修改 下一步

### 检查 SQL 下一步

检查SQL

<p>* 主题</p>	<input style="width: 90%;" type="text" value="创建db7 表"/>
<p>描述</p>	<input style="width: 90%;" type="text"/>
<p>* 执行人</p>	<input style="width: 90%;" type="text" value="ark_control"/>

取消 提交

## 提交, 生成 SQL 审批流程

基本信息 回滚信息

### 工单详情

主题:	创建db7表	归属组:	DB部门	创建人:	ark_control
集群名称:	-	数据源名称:	db1	工单状态:	待审核
实例地址:	10.0.0.231	实例端口:	3307	工单描述:	-
创建时间:	2020-01-08 12:03:23	更新时间:	2020-01-08 12:03:23		

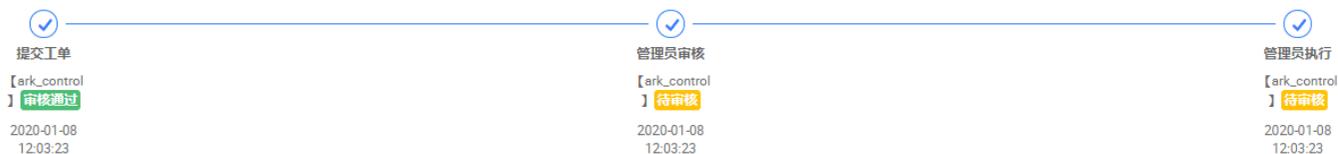
### 检查结果

SQL总条数: 1 预计影响行数: 0

参与审核的SQL语句: [预览](#) [下载](#) [检查结果](#)

### 工单进度

[修改工单](#) [审批](#)



### 处理历史

阶段名	处理人员	处理状态	处理描述	时间
		没有数据		

## 查看审批结果, 预览, 以及下载

### 检查结果

SQL总条数: 1

参与审核的SQL语句: [预览](#) [下载](#) [检查结果](#)

### 工单进度-修改工单 (返回创建工单界面)

#### 申请工单

操作实例

来自平台中  来自配置数据源  手动输入

\* 集群名称

db1

节点 10.0.0.231:3307

提交方式

文本  附件

\* SQL语句

```
create database db7 char set =utf8mb4;
```

预检查

### 工单进度-审批

工单审批 ×

审批状态  通过  不通过

审批信息

确认 取消

### 撤销工单

#### 工单进度

修改工单

撤销

## 执行工单

基本信息 SQL执行 回滚信息

### 执行工单

待执行SQL文件名: ./uploads/20200108/61e8dc33-442d-43d7-b104-9a6ad98fee91\_1.sql

预览

下载

语句类型: DML

执行选项:  备份  忽略警告  强制执行

执行时间间隔(ms): 0

操作: [立即执行](#) [标记完成](#) [定时执行](#)

执行人: -

执行耗时(ms): 0

执行状态: **待执行**

## 执行结果

SQL语句执行结果

ID	阶段	执行代码	状态	信息	SQL语句	备份数据库	备份序列号	实际影响行数	执行时间
1	EXECUTED	Success	Execute Successfully	None	create database db6 char set=utf8mb4		1578459147_20524_1	1	

< 上一页 1 下一页

## 回滚信息

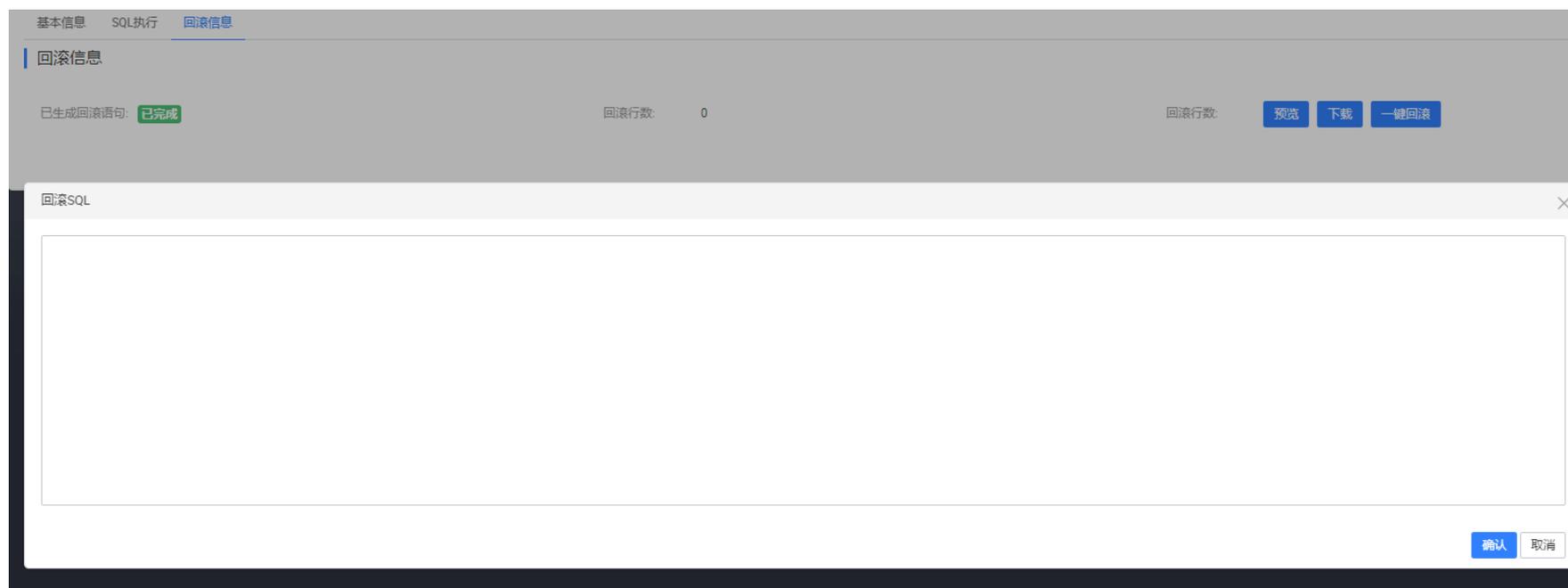
基本信息 SQL执行 回滚信息

### 回滚信息

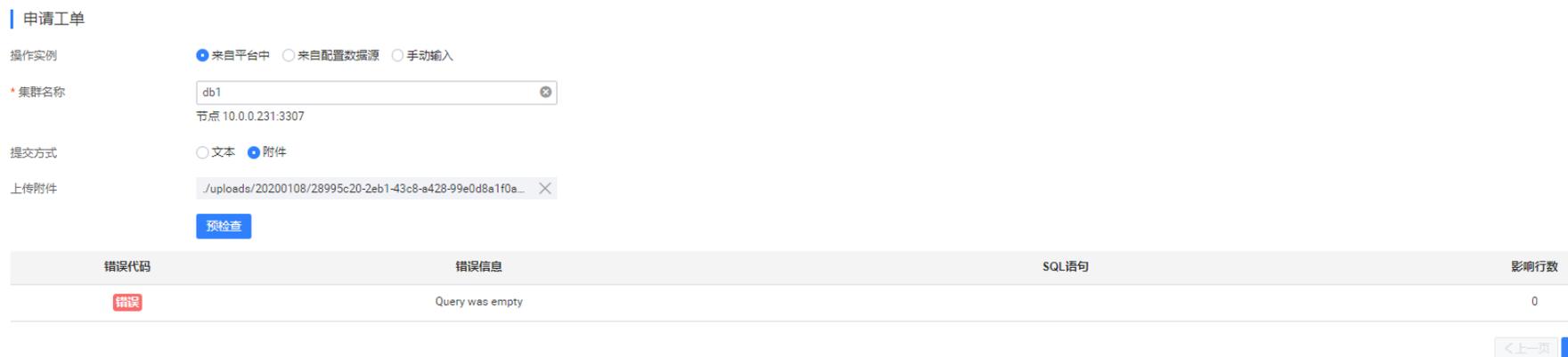
已生成回滚语句: **未回滚**

回滚行数: 0

## DDL 生成回滚语句 ( )



## DDL 因为没有语句导致一键回滚失败



## 创建表

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| db1 |
| db2 |
| db3 |
| db7 |
| mysql |
| performance_schema |
| sys |
+-----+
```

## DML 操作可以生成回滚语句



## 主机管理

### 主机录入

## 主机录入

### 录入信息

* IP地址	<input type="text" value="127.0.0.1"/>
* 录入描述	<input type="text" value="ark_control"/> 11/20
* 可用区	<input type="text" value="北1机房"/>
* SSH用户名 <sup>?</sup>	<input type="text" value="ark_control"/>
* SSH端口	<input type="text" value="22"/>
* 认证方式	<input checked="" type="radio"/> 密码 <input type="radio"/> 私钥
* SSH密码	<input type="password" value="....."/>

录入

[人工录入](#)

### SSH 用户创建

```
useradd ark_control
```

```
passwd ark_control
```

//账号名和密码由用户自己设置:

```
usermod -g root ark_control
```

```
vim /etc/sudoers
```

//添加如下行:

```
ark_control ALL=(ALL) NOPASSWD: ALL
```

## 录入状态

IP地址列表	录入描述	录入者	状态	录入时间	操作
127.0.0.1	arkcontrol	ark_control	正在录入	2020-01-07 15:38:16	<a href="#">详情</a> <a href="#">删除</a>

## 录入主机

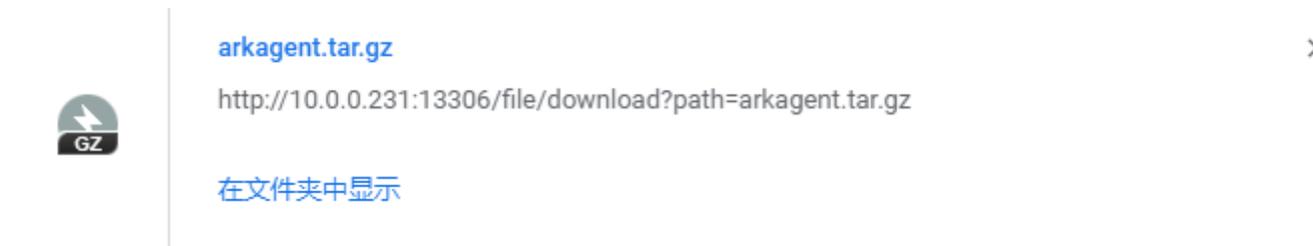
录入主机，会在主机的 /data/arkcontrol 目录下安装 arkagent 软件，用来收集和操纵主机

## 录入错误重试



## agent 软件下载

会下载当前使用的 agent 版本



## 主机录入历史删除

删除掉录入历史信息，不会删除掉主机

## 主机管理（管理已经录入的主机）

### Agent 全部升级

如何操作

### 录入主机信息

#### 主机管理

关键字:  可用区:

◇ AGENT全部升级

主机名	IP地址	CPU	内存容量	磁盘容量	可用区	AGENT版本	AGENT启动时间	AGENT状态	监控	操作
zesh02	10.0.0.232	1	1.80GB	38.38GB	北2机房	2.7.0	2020-01-06 17:10:37	normal		<a href="#">详情</a>
zesh03	10.0.0.233	1	1.80GB	38.38GB	北1机房	2.7.0	2020-01-06 17:10:33	normal		<a href="#">详情</a>

< 上一页 1 下一页 >

## 录入主机详情

## 基本信息（可以对主机所属的机房进行修改）

## | 基本信息

主机名:	zsh02	IP信息:	10.0.0.232	可用区:	北2机房 ✓
操作系统:	linux	CPU核数:	1	CPU信息:	Intel(R) Xeon(R) CPU E5-2650 v3 @ 2.30GHz
内存:	1.80GB	磁盘容量:	38.38GB	备注:	-

## | ANAGENT信息

AGENT版本:	2.7.0	状态:	normal	更新时间:	2020-01-07 15:59:37
----------	-------	-----	--------	-------	---------------------

磁盘信息 实例信息 监控 告警配置

挂载点	文件系统	磁盘容量	使用比例	已用	可用	驱动	更新时间
/	xfs	7.99GB	25%	1.98GB	6.01GB	/dev/mapper/centos-root	2020-01-07 15:59:50
/data	ext4	29.40GB	13%	3.75GB	24.14GB	/dev/sdb	2020-01-07 15:59:50
/boot	xfs	0.99GB	14%	0.14GB	0.85GB	/dev/sda1	2020-01-06 17:10:38

## 修改机房信息

可用区:	<input type="text" value="请选择"/>	✓	↻
CPU信息:	北1机房	v3 @ 2.30GHz	
备注:	北1灾备		
	北2机房		
	北2灾备		
更新时间:	2020-01-07 15:59:37		

## agent 信息

## 磁盘信息

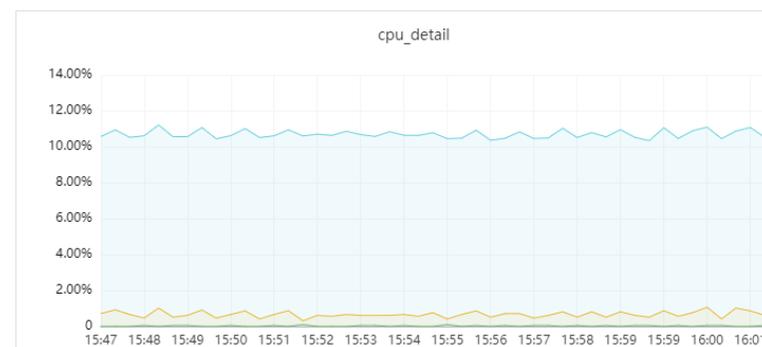
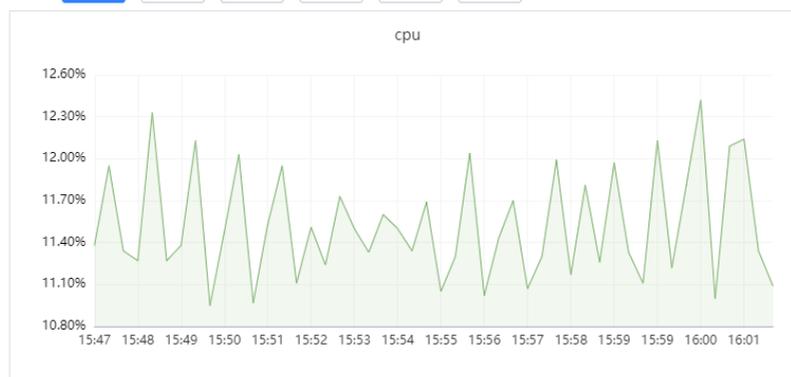
[磁盘信息](#) [实例信息](#) [监控](#) [告警配置](#)

挂载点	文件系统	磁盘容量	使用比例	已用	可用	驱动	更新时间
/	xfs	7.99GB	25%	1.98GB	6.01GB	/dev/mapper/centos-root	2020-01-07 16:01:50
/data	ext4	29.40GB	13%	3.75GB	24.14GB	/dev/sdb	2020-01-07 16:01:50
/boot	xfs	0.99GB	14%	0.14GB	0.85GB	/dev/sda1	2020-01-06 17:10:38

## 实例信息

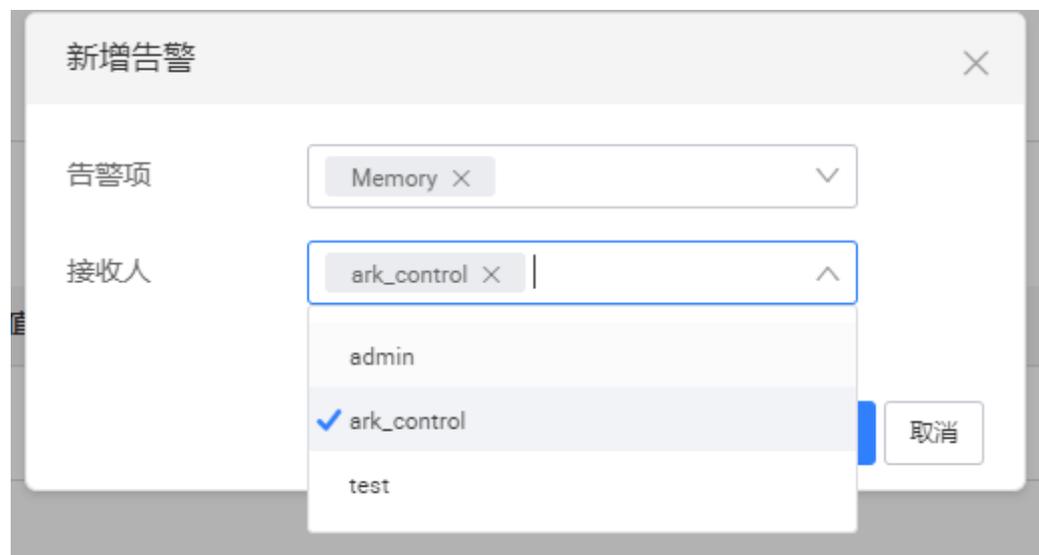
集群名	实例归属组	实例名称	角色	实例版本	实例规格	架构类型	监控	Uptime
test01	系统默认组	10.0.0.232_3306	master	Percona Server (GPL) 5.7.2 6-29-log	-	MM Cluster		5时4分0秒

## 监控信息

[磁盘信息](#) [实例信息](#) [监控](#) [告警配置](#)时间: [15分钟](#) [1小时](#) [1天](#) [7天](#) [1个月](#) [自定义](#)

## 告警配置

## 新建告警



### 修改告警配置信息

#### 预警阈值

预警阈值

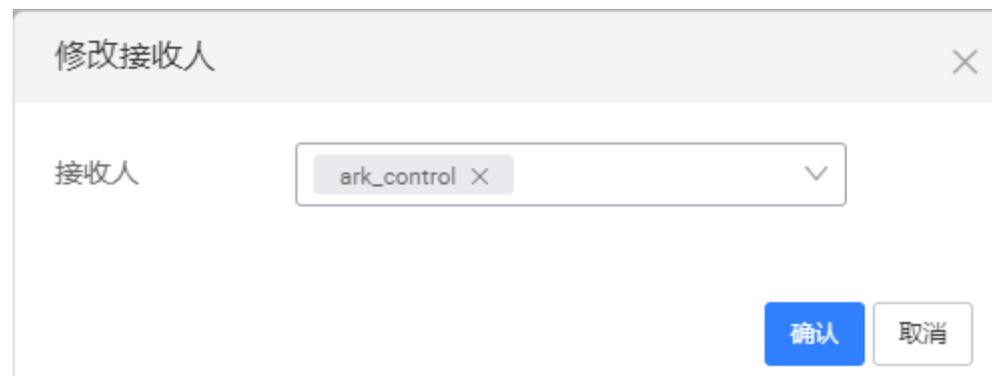
 ✓ ↻

#### 报警阈值

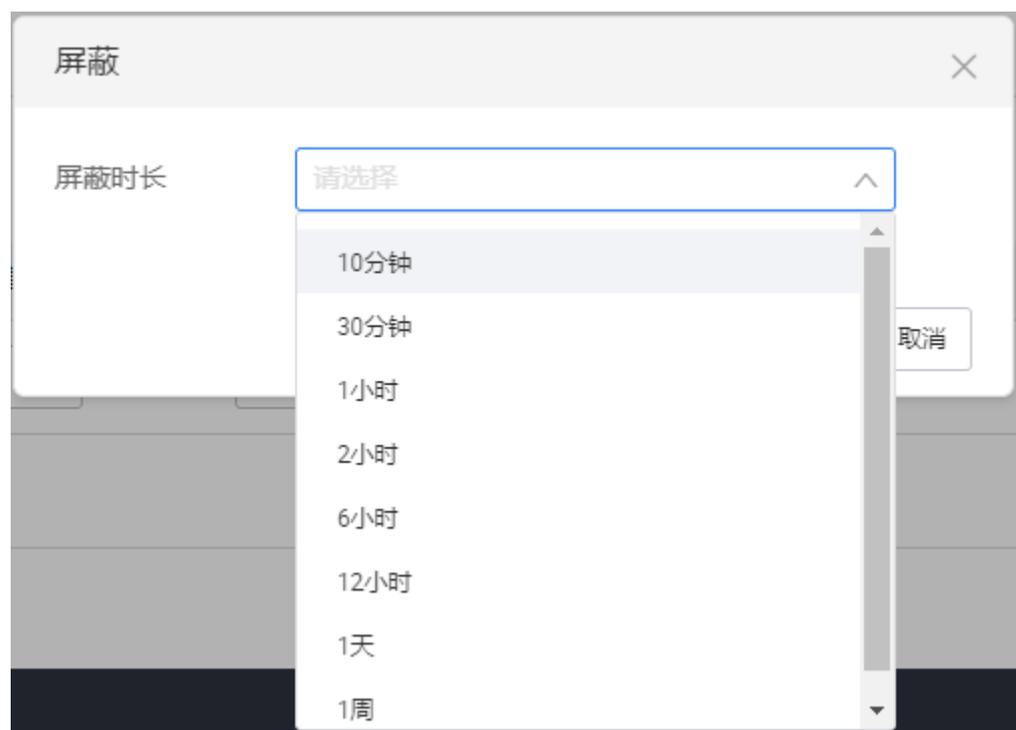
报警阈值

 ✓ ↻

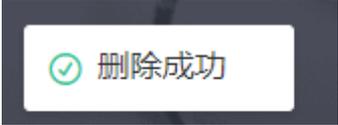
#### 接收人修改



**屏蔽告警 (可选屏蔽时长,屏蔽后,接收告警通知的用户不会再收到告警信息)**



**删除告警**



录入主机的监控 ([跳到监控告警选项栏](#))

## 主机巡检

### 巡检周期设置

巡检周期设置可以通过访问 [配置中心 - 通知配置- Agent 配置信息](#) 来进行配置间隔

### 巡检 (可以看到历史告警信息)

#### 巡检

[巡检](#) [巡检配置](#)

巡检项:  关键字:  可用区:

搜索

重置

主机名称	主机IP	可用区	使用率	巡检结果	详细信息
zsh03	10.0.0.233	北1机房	11%	巡检警告	内存使用率 11%
zsh02	10.0.0.232	北2机房	11%	巡检警告	内存使用率 11%

< 上一页 1 下一页 >

## 巡检配置 (可以自定义设置巡检阈值)

### 巡检

巡检 [巡检配置](#)

巡检项	巡检阈值	巡检说明
内存使用率	1% 	内存使用率大小超过阈值时警告
swap使用率	50% 	swap使用率超过阈值时警告
磁盘使用率	85% 	磁盘使用率超过阈值时警告
inode使用率	85% 	inode使用率超过阈值时警告
TCP连接数	2000 	TCP连接数超过阈值时警告

## 监控告警

告警可以选择机房，选择机房内的任意主机，以及选择显示时间范围

## 监控告警

监控 告警





查看已经产生的告警（可以选择屏蔽时长）

#### 监控告警

监控 告警

主机名	可用区	IP地址	监控项	告警信息	状态	最近报警时间	操作
10.0.0.233	北机房	10.0.0.233	Load	> 0.662222	<b>fatal</b>	2020-01-07 16:10:00	屏蔽

< 上一页 1 下一页

## MySQL 管理-MySQL 部署

### MySQL 部署

### 部署列表 (查询部署的机器列表)

部署列表    参数模板

#### 部署列表

集群名称:

架构类型:

归属组:

部署状态:

## 部署集群

### 部署实例基本信息

1

实例基本信息

2

安装方式配置

集群名称	
* 集群名称	三类
* 归属组	DB部门
* 可用区	北1机房 ×
* 数据源类型	MySQL
* 架构类型	Master Slave
* 版本号	5.7.26
* 端口	3306
* Master地址	10.0.0.231
* Slave地址	10.0.0.232 × 10.0.0.233 ×
* 复制账号	repl
* 复制密码	repl

[下一步](#)

## 部署实例安装方式配置 (根据需求填写配置)

实例基本信息 2 安装方式配置

安装方式

\* 部署路径 <sup>?</sup>:

\* 初始化账号 <sup>?</sup>:

\* 初始化密码 <sup>?</sup>:

## 部署实例参数配置 (根据需求填写参数配置)

实例基本信息 3 安装方式配置 实例参数配置

核心参数

参数配置 <sup>?</sup>:  默认  从模板中获取

innodb\_buffer\_pool\_size  G

sync\_binlog <sup>?</sup>:

innodb\_flush\_log\_at\_trx\_commit <sup>?</sup>:

expire\_log\_days <sup>?</sup>:

## 部署进度查看

### 基本信息

集群名称:	db1	归属组:	DB部门	架构类型:	MS Cluster
可用区:	北1机房	版本号:	5.7.26	部署状态:	部署成功
IP地址:	10.0.0.231	端口:	3307	创建时间:	2020-01-07 18:16:49
部署者:	ark_control				

### 核心参数

部署路径:	/data/arkdb/3307/data	innodb_flush_log_at_trx_commit:	1	sync_binlog:	1
innodb_buffer_pool_size(G):	1	expire_log_days:	1	log_queries_not_using_indexes:	-

### 部署进度

<h4>主节点</h4> <p>IP地址: 10.0.0.231</p> <p>部署状态: 部署成功</p> <p>最新进度: 部署完成</p> <p>清理</p> <p>查看更多</p>	<h4>从节点</h4> <p>IP地址: 10.0.0.232</p> <p>部署状态: 部署成功</p> <p>最新进度: 部署完成</p> <p>清理</p> <p>查看更多</p>	<h4>从节点</h4> <p>IP地址: 10.0.0.233</p> <p>部署状态: 部署成功</p> <p>最新进度: 部署完成</p> <p>清理</p> <p>查看更多</p>
--	--	--

**部署进度中任意节点查看**

进度信息 ✕

节点数检查通过  
**执行成功**

端口范围检测通过  
**执行成功**

端口是否可用检测通过  
**执行成功**

basedir检测通过  
**执行成功**

Datadir检测通过  
**执行成功**

MySQL实例数检测通过  
**执行成功**

内存剩余363M，不足1024M  
**执行失败**

磁盘空间检测通过  
**执行成功**

检测安装MySQL所需用户mysql:mysql  
通过  
**执行成功**

安装MySQL完成  
**执行成功**

修改配置文件完成  
**执行成功**

启动MySQL实例  
**执行成功**

对MySQL授权完成  
**执行成功**

部署完成  
**执行成功**

**确认** **取消**

## 清理机器

点击清理按钮即可，数据目录等都会被清理

主节点 [查看更多](#)

---

IP地址: 10.0.0.232  
部署状态: **清理完成**  
最新进度: 等待部署

[重试](#) [清理](#)

**部署参数模板 (可以手动上传模板或者自定义模板, 可以选择设置更多参数)**

## 创建参数模板

* 模板名称	<input type="text" value="三类项目专用"/> 6/20
* 数据源类型	<input type="text" value="MySQL"/>
* 版本 ?:	<input type="text" value="5.7.26"/>
* 参数来源	<input checked="" type="radio"/> 平台默认配置 <input type="radio"/> 手动上传文件
* innodb_buffer_pool_size	<input type="text" value="1"/> G
* sync_binlog ?:	<input type="text" value="1"/>
* innodb_flush_log_at_trx_commit ?:	<input type="text" value="1"/>
* expire_log_days ?:	<input type="text" value="1"/>
* log_queries_not_using_indexes ?:	<input checked="" type="radio"/> on <input type="radio"/> off
<a href="#">更多参数选择</a>	
模板描述	<input type="text" value="三类项目专用模板"/> 8/200
	<input type="button" value="保存"/> <input type="button" value="取消"/>

## 集群列表

## 查看所有已经创建的集群

集群列表

集群名称:  到期时间:  架构类型:

归属组:  运行状态:

集群名称	架构类型	归属组	运行状态	到期时间	操作
db1	MS Cluster	DB部门	集群正常	2030-01-01	<input type="button" value="详情"/>
10.0.0.231_3306	Single Instance	系统默认组	集群正常	2030-01-01	<input type="button" value="详情"/>

< 上一页 1 下一页

## 集群详情

## 实例信息 (集群内实例的基本信息, 以及监控告警等)

实例信息

实例名称	端口	角色	主机名	监控	实例版本	运行状态	到期时间	操作
10.0.0.232_3307	3307	slave	zsh02	<input checked="" type="checkbox"/>	Percona Server (GPL) 5.7.26-29-log	实例正常	2030-01-01	<input type="button" value="告警"/>
10.0.0.233_3307	3307	slave	zsh03	<input checked="" type="checkbox"/>	Percona Server (GPL) 5.7.26-29-log	实例正常	2030-01-01	<input type="button" value="告警"/>
10.0.0.231_3307	3307	master	zsh01	<input checked="" type="checkbox"/>	Percona Server (GPL) 5.7.26-29-log	实例正常	2030-01-01	<input type="button" value="告警"/>

## 集群信息 (集群名称, 集群备注, 集群归属组等信息)

集群信息

集群名称: db1

架构类型: MS Cluster

集群备注:

高可用配置: db1

归属组: DB部门

## 集群信息-添加从库 (切勿添加相同名称的机器)

## 给已有集群添加从库

集群现状

架构类型	角色	实例名称	端口	主机名	实例版本	运行状态	到期时间
Single Instance	master	10.0.0.231_3306	3306	10.0.0.231	Percona Server (GPL) 5.7.20-19-log	实例正常	2030-01-01

1 实例基本信息 2 实例参数配置

添加从库

\* Master地址

\* 可用区

\* 新增节点地址

\* 部署路径

\* 端口

\* 数据源类型

\* 版本号

\* 同步账号

\* 同步密码

\* 传输速度  MB/s

下一步

## 添加从库的参数设置

实例基本信息

### 添加从库

#### 核心参数

##### \* 参数配置

默认  从模板中获取

\* innodb\_buffer\_pool\_size

1

\* sync\_binlog ?):

1

\* innodb\_flush\_log\_at\_trx\_commit ?):

1

\* expire\_log\_days ?):

15

\* max\_binlog\_files ?):

200

log\_queries\_not\_using\_indexes ?):

ON  OFF

更多参数选择

上一步

提交

## 添加集群的进度信息

进度信息

check if port is free

部署成功

check if basedir is empty

部署成功

check if datadir is empty

部署成功

check if memory is bigger than MemThreshold

部署成功

check if memory is bigger than DiskThreshold

部署成功

run rpm install cmd

部署成功

run install instance cmd

部署成功

modify instance config file

部署成功

start innobackupex begin backup

部署成功

start arkncat wait receive backups

部署成功

prepare backups

部署成功

start instance

部署成功

check binlog position info

部署成功

change master

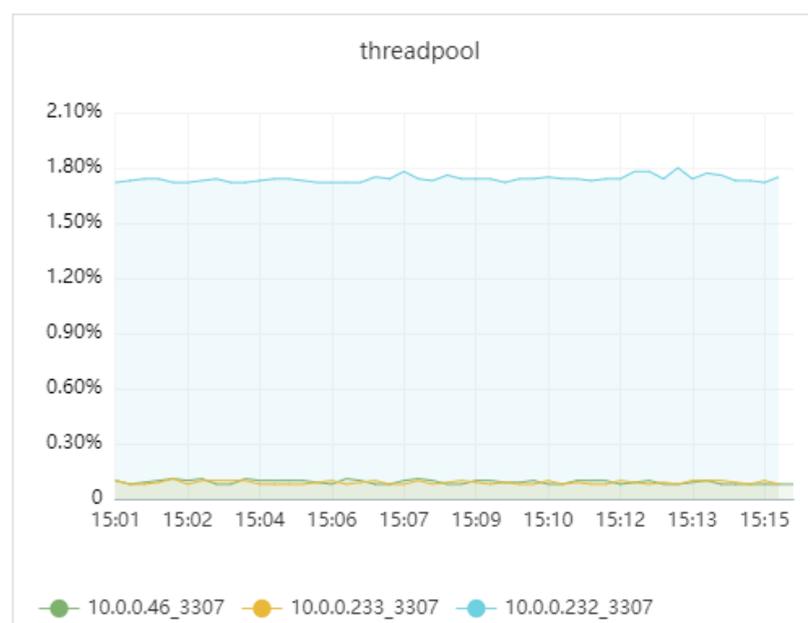
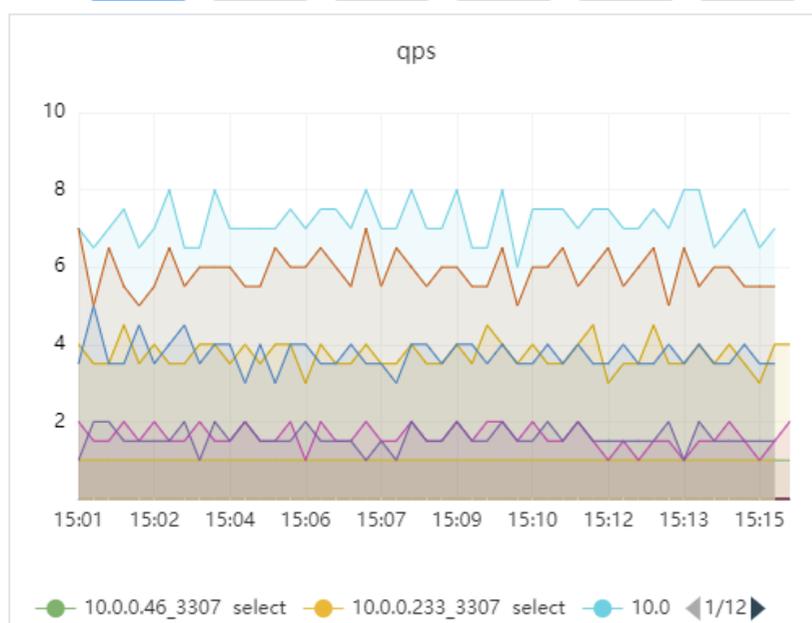
部署成功

扩容完成的集群信息，可以在 [\(扩容列表中查看\)](#)

监控告警 [\(查看整个集群的监控信息\)](#)

### 监控告警

时间:



### 扩容列表

## 显示已经扩容的集群的状态

**扩容列表**

集群名称:  IP地址:  实例归属组:

扩容类型:  部署状态:  部署状态:

[搜索](#) [重置](#)

集群名称	IP地址	扩容类型	归属组	版本	端口号	创建时间	部署状态	操作
10.0.0.231_3306	10.0.0.231	添加从库	系统默认组	5.7.26	3308	2020-01-08 09:50:20	部署成功	<a href="#">详情</a> <a href="#">删除</a>

< 上一页 1 下一页 >

## 查看,删除已经扩容的集群信息

### 基本信息

集群名称:	三类项目	归属组:	DB部门	端口:	3307
可用区:	北1机房	版本号:	5.7.26	部署状态:	部署成功
创建时间:	ark_control	更新时间:	2020-01-09 14:26:45	同步实例:	10.0.0.232_3307

### 核心参数

部署路径:	/data/arkdb/3307/data	innodb_flush_log_at_trx_commit:	1	sync_binlog:	1
innodb_buffer_pool_size(G):	1	log_queries_not_using_indexes:	ON		

### 部署进度

#### 新增节点

[查看更多](#)

IP地址: 10.0.0.233

端口: 3307

部署状态: 部署成功

## 实例列表

## 查看当前已存在的实例（运行和不运行的都可以看到）

实例列表

关键字:  角色:  实例归属组:

架构类型:  可用区:  运行状态:

集群名称	实例名称	主机名称	角色	架构类型	版本	实例归属组	可用区	监控	运行状态	操作状态	操作
10.0.0.231_3306	10.0.0.231_3308	zsh01	unknown	MS Cluster	5.7	系统默认组	北1机房	<input checked="" type="checkbox"/>	实例正常	•	<input type="button" value="详情"/> <input type="button" value="停止"/>
db1	10.0.0.231_3307	zsh01	master	MS Cluster	5.7	DB部门	北1机房	<input checked="" type="checkbox"/>	实例正常	•	<input type="button" value="详情"/> <input type="button" value="停止"/>
db1	10.0.0.233_3307	zsh03	slave	MS Cluster	5.7	DB部门	北1机房	<input checked="" type="checkbox"/>	实例正常	•	<input type="button" value="详情"/> <input type="button" value="停止"/>
db1	10.0.0.232_3307	zsh02	slave	MS Cluster	5.7	DB部门	北2机房	<input checked="" type="checkbox"/>	实例正常	•	<input type="button" value="详情"/> <input type="button" value="停止"/>
10.0.0.231_3306	10.0.0.231_3306	zsh01	master	MS Cluster	5.7	系统默认组	北1机房	<input checked="" type="checkbox"/>	实例正常	•	<input type="button" value="详情"/>

< 上一页 1 下一页 >

## 可以查看当前的实例在集群中的角色是什么 (master, slave, unknown)

角色
unknown
master
slave
slave
master

## 查看已存在实例的详细信息

## 基本信息

基本信息:	10.0.0.231_3306	实例归属组:	系统默认组	架构类型:	MS Cluster
实例版本:	Percona Server (GPL) 5.7.26-29-log	实例名称:	10.0.0.231_3308	端口:	3308
角色:	unknown	主机名称:	zsh01	可用区:	北1机房

## 配置信息

CPU:	1 core	存储类型:	SSD	可维护时间段:	1:00~2:00
到期时间:	2030-01-01				

## 运行状态

运行状态:	<span style="background-color: #28a745; color: white; padding: 2px;">实例正常</span>	创建时间:	2020-01-08 09:52:15	集群备份状态:	-
-------	--	-------	---------------------	---------	---

[连接管理](#) [账号管理](#) [INNODB STATUS](#) [死锁信息](#) [监控](#) [告警配置](#)

用户	来源主机	连接数
ark_monitor	10.0.0.231	1
system user		2

## 连接管理

[连接管理](#) [账号管理](#) [INNODB STATUS](#) [死锁信息](#) [监控](#) [告警配置](#)

用户	来源主机	连接数量
arkproxy	10.0.0.44	2
ark_monitor	10.0.0.231	1
monitor	10.0.0.45	1
system user		6

## 账号信息

[连接管理](#) [账号管理](#) [INNODB STATUS](#) [死锁信息](#) [监控](#) [告警配置](#)

用户	IP地址	权限信息
ark_monitor	10.0.0.231	CREATE USER IF NOT EXISTS 'ark_monitor'@'%ip'; ALTER USER 'ark_monitor'@'%ip' IDENTIFIED WITH 'mysql_native_password' AS '*6FEE718633EE5F94BEF58E013776D193F2C9AF9C' REQUIRE NONE PASSWORD EXPIRE DEFAULT ACCOUNT UNLOCK; GRANT ALL PRIVILEGES ON *.* TO 'ark_monitor'@'%ip' WITH GRANT OPTION;
	10.0.0.232	
	10.0.0.233	
	10.0.0.44	
	10.0.0.45	
	10.0.0.46	
arkproxy	127.0.0.1	
	10.0.0.231	CREATE USER IF NOT EXISTS 'arkproxy'@'%ip'; ALTER USER 'arkproxy'@'%ip' IDENTIFIED WITH 'mysql_native_password' AS '*A093DDB56DE15A7BC8CBE4A1D2BCAB483EFE2B02' REQUIRE NONE PASSWORD EXPIRE DEFAULT ACCOUNT UNLOCK; GRANT ALL PRIVILEGES ON *.* TO 'arkproxy'@'%ip' WITH GRANT OPTION;
	10.0.0.232	
	10.0.0.233	
	10.0.0.44	
	10.0.0.45	
10.0.0.46		

## InnoDB\_status

连接管理    账号管理    **INNODB STATUS**    死锁信息    监控    告警配置

```
=====
2020-02-01 02:46:43 0x7f0d827ff700 INNODB MONITOR OUTPUT
=====
Per second averages calculated from the last 60 seconds
-----
BACKGROUND THREAD
-----
srv_master_thread loops: 1709730 srv_active, 0 srv_shutdown, 277912 srv_idle
srv_master_thread log flush and writes: 1987596
-----
SEMAPHORES
-----
```

## 死锁信息

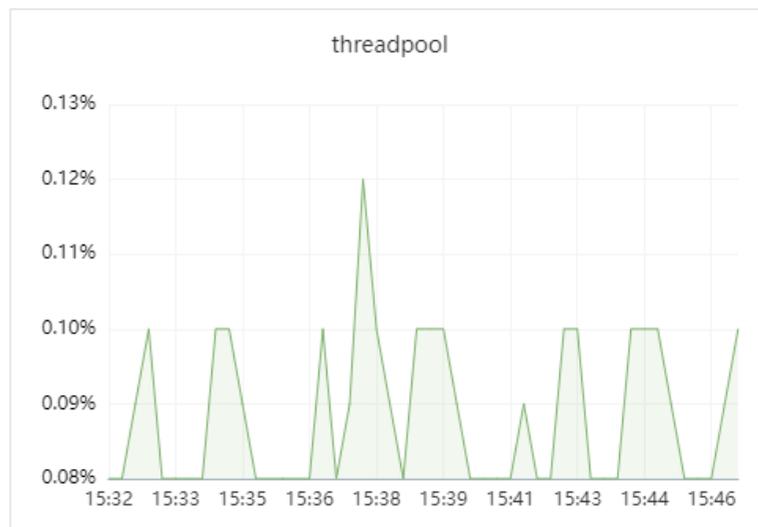
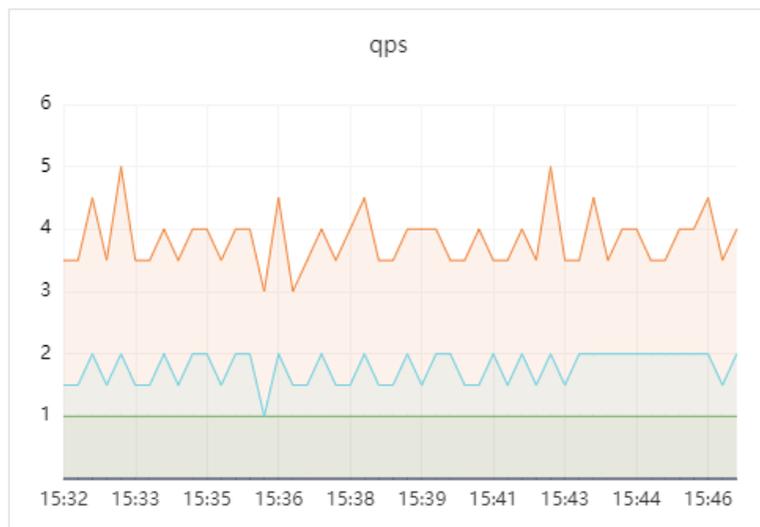
连接管理    账号管理    INNODB STATUS    **死锁信息**    监控    告警配置

暂无数据

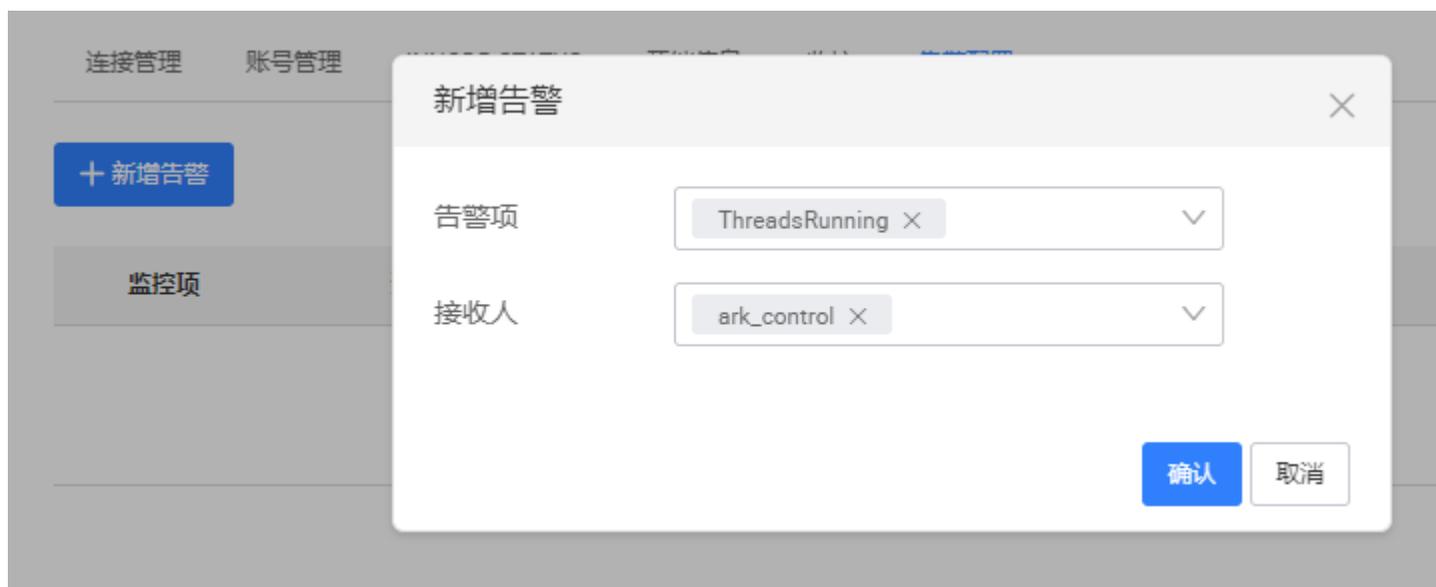
### 监控

连接管理 账号管理 INNODB STATUS 死锁信息 **监控** 告警配置

时间: **15分钟** 1小时 1天 7天 1个月 自定义



## 告警配置



监控项	预警阈值	报警阈值	报警说明	状态	接收人	操作
ThreadsRunning	20	50	MySQL连接池并发数	normal	ark_control	<a href="#">屏蔽</a> <a href="#">删除</a>

启动关闭已经停止的实例（建议先查看详情，然后再启动实例）

操作状态	操作
●	<a href="#">详情</a> <a href="#">停止</a>
●	<a href="#">详情</a>

< 上一页 1 下一页 >

## MySQL 巡检

### 巡检

[巡检](#) [巡检报告](#) [巡检配置](#)

| 巡检

巡检项:

集群名称:

[搜索](#) [重置](#)

集群名称	实例名称	主机名称	端口	数据大小	巡检结果	详细信息
10.0.0.231_3306	10.0.0.231_3306	zeh01	3306	3.07GB	巡检警告	实例数据文件大小 3142 M
db1	10.0.0.232_3307	zeh02	3307	3.05GB	巡检警告	实例数据文件大小 3126 M
db1	10.0.0.233_3307	zeh03	3307	3.05GB	巡检警告	实例数据文件大小 3126 M
db1	10.0.0.231_3307	zeh01	3307	3.05GB	巡检警告	实例数据文件大小 3126 M
10.0.0.231_3306	10.0.0.231_3308	zeh01	3308	3.05GB	巡检警告	实例数据文件大小 3122 M

< 上一页 1 下一页 >

## 巡检报告 (可以根据需求, 生成巡检内容, 然后生成报告)

巡检 [巡检报告](#) [巡检配置](#)

### 创建巡检报告

\* 巡检集群

\* 巡检项:

<input type="checkbox"/> 实际数据大小	<input type="checkbox"/> BINLOG文件大小	<input type="checkbox"/> 表自增键使用率
<input type="checkbox"/> MyISAM表检查	<input type="checkbox"/> 大表检查	<input type="checkbox"/> 表碎片率
<input type="checkbox"/> 表索引个数	<input type="checkbox"/> 冗余索引检查	<input type="checkbox"/> 从库可写检查
<input type="checkbox"/> 临时表检查	<input type="checkbox"/> 外键检查	<input type="checkbox"/> 没有主键表检查

\* 报告格式  pdf  docx

备注

[生成并下载](#)

### 巡检报告列表

文件名	生成时间	生成报告用户	备注	下载报告
全部集群_20200106191150.pdf	2020-01-06 19:11:50	ark_control		<a href="#">下载报告</a>

< 上一页 1 下一页

## 巡检配置

[巡检](#) [巡检报告](#) [巡检配置](#)

## | 巡检配置

巡检项	巡检阈值	巡检说明
实例数据文件大小	10MB 	实例数据文件大小超过阈值时警告
binlog文件大小	1MB 	binlog文件大小超过阈值时警告
表自增键使用率	88% 	自增键使用率超过阈值时警告
MyISAM表检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
大表检查	0MB 	表大小超过阈值时警告
表碎片率	82% 	表碎片率超过阈值时警告
表索引个数	1 	表索引个数超过阈值时警告
冗余索引检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
从库可写检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
临时表检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
外键检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查
没有主键表检查	<input checked="" type="checkbox"/>	ON为开启检查,OFF为关闭检查

## 监控告警

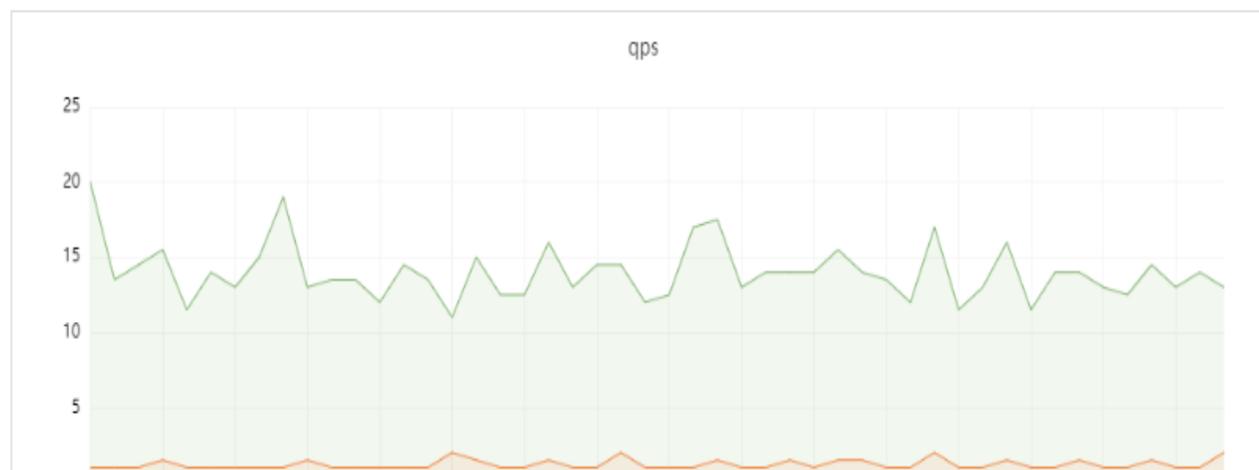
### 监控告警 (MySQL 相关监控和告警)

#### 监控告警

监控 告警

架构类型: 基础版(主从) v

时间: 15分钟 1小时 1天 7天 1个月 自定义



实例列表 10.0.0.231\_3306 x ^

- 10.0.0.231\_3306
- 10.0.0.232\_3307
- 10.0.0.233\_3307
- 10.0.0.231\_3307
- 10.0.0.231\_3308



备份中心

## 物理备份(基于备份配置)

### 备份列表

物理备份 逻辑备份 配置

#### 备份列表

集群名称:  是否部署:  备份状态:

搜索 重置

集群名称	归属组	备份	备份状态	最近一次备份	下一次备份	BINLOG备份	操作
db1	DB部门	没有部署	•			-	部署备份
10.0.0.231_3306	系统默认组	没有部署	•			-	部署备份

< 上一页 1 下一页 >

## 部署备份

## 部署备份

## 基础信息

- \* 备份实例
- \* MySQL配置文件路径
- \* 本地备份存储临时目录
- \* 备份命令路径

## 备份机器配置

- \* 备份机地址
- \* 备份机数据存储目录
- \* 备份池  否  是

## 备份策略

- \* 传输速度(M)
- \* 数据备份保留(天) ①  天
- \* 备份周期
- \* 备份时间范围 ②
- \* 日志备份  否  是
- \* 日志备份保留(天)  天
- \* 日志备份周期(秒)  秒

确认

取消

## 备份详情

**基本信息** (可以选择立即备份, 进行备份, 也可以重新配置一下备份信息)

## 备份详情

基本信息 数据备份 BINLOG备份 备份日志

## 备份信息

立即备份

集群名称: db1 归属组: DB部门 备份实例: 10.0.0.233\_3307 [↗](#)  
 主机名: 10.0.0.233 端口: 3307 最近数据备份: **未开始**  
 最近日志备份: **备份成功** 最近备份时间: 1970-01-02 12:00:00

## 配置信息

保存 取消

备份机器: 10.0.0.231 备份机存储目录: /data/backup  
 本地临时目录: /data/backup  
 备份池:  不开启  开启  
 备份周期: 每天备份一次 备份时间范围: 00:00:00-01:00:00 数据保留(天): 1 天  
 下次备份时间: 2020-01-08 传输速度: 1000 M 备份命令路径: /usr/bin  
 BINLOG备份:  不开启  开启 BINLOG保留(天): 2 天 日志备份周期(秒): 100 秒  
 配置文件路径: /data/arkdb/3307/etc/my.cnf

**立即备份** (可以看到当前状态, 备份中, 或者选择终止备份)

## 备份信息

终止备份

集群名称: db1 归属组: DB部门 备份实例: 10.0.0.233\_3307 [↗](#)  
 主机名: 10.0.0.233 端口: 3307 最近数据备份: **备份中**  
 最近日志备份: **备份成功** 最近备份时间: 1970-01-02 12:00:00

## 数据备份

## 备份详情

基本信息 数据备份 BINLOG备份 备份日志

存储机器	备份集名称	数据大小(B)	备份日期	是否压缩	Binlog名称	Binlog位置	备份集的GTID	备份集清理日期
10.0.0.231	10.0.0.233_3307.full.20200107213724	3255522439	2020-01-08 10:37:59	未压缩	mysql-bin.000005	154		2020-01-08 21:37:59

## BINLOG 备份

**备份详情**

基本信息 数据备份 **BINLOG备份** 备份日志

备份状态:  开始时间:

实例名称	开始时间	更新时间	BINLOG文件大小之和	备份状态	备份详情
10.0.0.233_3307	2020-01-08 10:31:51	2020-01-08 10:32:13	266	备份成功	<a href="#">查看详细日志</a>

< 上一页 1

**备份日志**

存储机器	BINLOG文件名称	BINLOG文件记录的开始时间	BINLOG文件记录的结束时间	文件大小(B)	Binlog状态	清理日期
10.0.0.231	mysql-bin.000001.tar.gz	2020-01-07 05:17:40	2020-01-07 21:31:52	266	可用	2020-01-09 21:31:52

## 备份日志

集群名称	实例名称	开始时间	更新时间	备份状态	压缩状态	传输到备份池	详细日志
db1	10.0.0.233_3307	2020-01-08 10:37:21	2020-01-08 10:37:24	备份中	未压缩	未传输	<a href="#">查看详细日志</a>

## 备份日志详情

### 备份日志

日志来源组件	日志信息	时间
发送备份集端	Backuping start.	2020-01-08 10:37:21
接受备份集端	Receive backupset start.	2020-01-08 10:37:24
接受备份集端	Receive cmdDir:/data/backup/full_backup/db1/20200107/10.0.0.233_3307.ful.2020010...	2020-01-08 10:37:24
发送备份集端	Wait NC ready... count:0. msg:<nfil>	2020-01-08 10:37:31
发送备份集端	do innobackupex   arkat cmd to send bakset	2020-01-08 10:37:33
接受备份集端	Receive bakset data success!	2020-01-08 10:37:50
接受备份集端	Update trans_status to 5 finished.	2020-01-08 10:37:50
接受备份集端	Update last_backup_time success, and apply log start.	2020-01-08 10:37:50
发送备份集端	Host:10.0.0.233 send bakset data to backup_machine:10.0.0.231 success!	2020-01-08 10:37:51
接受备份集端	Apply log success.	2020-01-08 10:37:59
接受备份集端	Record bakset info success!	2020-01-08 10:37:59
接受备份集端	full hot backup succeeded!	2020-01-08 10:37:59

### Binlog 备份文件目录 (IP 命目录, 压缩打包)

```
[root@zsh01 10.0.0.233_3307]# pwd
/data/backup/binlog_backup/db1/10.0.0.233_3307
[root@zsh01 10.0.0.233_3307]# ll
total 4
-rw-r--r--. 1 root root 266 Jan  8 10:32 mysql-bin.000001.tar.gz
[root@zsh01 10.0.0.233_3307]#
```

## 全备目录命名

```
[root@zsh01 10.0.0.233_3307.full.20200107213724]# ll
total 3166272
-rw-rw----. 1 root root      494 Jan  8 10:37 backup-my.cnf
-rw-rw----. 1 root root      413 Jan  7 18:17 ib_buffer_pool
-rw-rw----. 1 root root 1073741824 Jan  8 10:37 ibdata1
-rw-r-----. 1 root root  536870912 Jan  8 10:37 ib_logfile0
-rw-r-----. 1 root root  536870912 Jan  8 10:37 ib_logfile1
-rw-r-----. 1 root root  536870912 Jan  8 10:37 ib_logfile2
-rw-r-----. 1 root root  536870912 Jan  8 10:37 ib_logfile3
-rw-r-----. 1 root root  12582912 Jan  8 10:37 ibtmp1
drwxr-xr-x. 2 root root      4096 Jan  8 10:37 mysql
drwxr-xr-x. 2 root root      4096 Jan  8 10:37 performance_schema
drwxr-xr-x. 2 root root     12288 Jan  8 10:37 sys
-rw-rw----. 1 root root        22 Jan  8 10:37 xtrabackup_binlog_info
-rw-r--r--. 1 root root        21 Jan  8 10:37 xtrabackup_binlog_pos_innodb
-rw-rw----. 1 root root       135 Jan  8 10:37 xtrabackup_checkpoints
-rw-rw----. 1 root root       718 Jan  8 10:37 xtrabackup_info
-rw-rw----. 1 root root   8388608 Jan  8 10:37 xtrabackup_logfile
-rw-r--r--. 1 root root         1 Jan  8 10:37 xtrabackup_master_key_id
-rw-rw----. 1 root root        68 Jan  8 10:37 xtrabackup_slave_info
[root@zsh01 10.0.0.233_3307.full.20200107213724]# pwd
/data/backup/full_backup/db1/20200107/10.0.0.233_3307.full.20200107213724
[root@zsh01 10.0.0.233_3307.full.20200107213724]#
```

## 逻辑备份

## 逻辑备份设置

## 部署逻辑备份

* 备份任务名称	每日导出
* 备份集群	db1
* 备份实例	10.0.0.233_3307
备份数据库/表	<a href="#">选择数据库表</a>
* 数据存储目录	/data/backup
* 数据备份保留(天)	2
* 备份周期(分钟)	60
备份开始时间	2020-1-1 00:00
	<a href="#">确定</a>

## 备份数据库, 表设置

备份数据库/表

可选数据库

点此搜索

mysql

可选数据表

点此搜索

columns\_priv  
 db  
 engine\_cost  
 event  
 func  
 general\_log  
 gtid\_executed  
 help\_category  
 help\_keyword  
 help\_relation

[确认](#) [取消](#)

## 逻辑备份详情

### 备份信息

[立即备份](#)

备份任务名称: 每日导出

任务创建者: ark\_control

下次备份时间: 2020-01-01 12:00:00

集群名称: db1

最近数据备份状态: 未开始

实例名称: 10.0.0.233\_3307

最近备份时间: -

### 配置信息

[编辑](#)

数据存储目录: /data/backup

备份周期(分钟): 60

数据备份保留(天): 2

数据库:

mysql

数据表:

mysql.\*

### 备份文件列表

备份任务名称	备份文件路径	数据大小	备份状态	备份开始时间	备份结束时间	备份文件清理时间	日志
没有数据							

## 立即开始逻辑备份

### 备份文件列表

备份任务名称	备份文件路径	数据大小	备份状态	备份开始时间	备份结束时间	备份文件清理时间	日志
每日导出	/data/backup/logic_backup/db1/202001/07/每日导出_mysqlL_20200107214821.sql	1.06MB	<span>备份成功</span>	2020-01-08 10:48:20	2020-01-08 10:48:20	2020-01-10 02:48:21	<a href="#">日志详情</a>

## 逻辑备份备份日志

备份日志

日志信息	
start backup /data/arkdb/3307/private/bin/mysqldump -uark_monitor -p'xxxxx' -P 3307 -h 127.0.0.1 --default-character-set=utf8mb4 --single-transaction=TRUE --databases mysql > /data/backup/logic_backup/db1/202001/07/每日导出_mysql_20200107214821.sql	2020-01-08 10:48:20
backup success	2020-01-08 10:48:20

确认 取消

## 逻辑备份目录

```
[root@zsh03 07]# pwd
/data/backup/logic_backup/db1/202001/07
[root@zsh03 07]# ll
total 1088
-rw-r--r--. 1 root root 1113036 Jan  8 10:48 每日导出_mysql_20200107214821.sql
```

## 配置

物理备份 逻辑备份 **配置**

<b>MySQL配置文件路径</b> + /data/mysql_home/ark tc/my.cnf ✓ tc/my.cnf ✓ tc/my.cnf ✓	<b>本地临时存储目录</b> + /data/backup	<b>备份命令路径</b> + /usr/bin	<b>备份机</b> + 10.0.0.231	<b>备份机数据存储目录</b> + /data/backup	<b>传输速度(M)</b> + 1000
<b>备份池</b> +	<b>备份池存储目录</b> +				

## 还原中心

## 还原

## 还原基本信息

## | 基本信息

- \* 任务描述
- \* 集群名称
- \* 还原主机
- \* 还原端口
- \* 传输速度(M)

[下一步](#)

## 还原类型

## | 还原类型信息

- \* 还原类型  还原为单实例(基于备份集)  还原为单实例(恢复到指定位置)  还原为单实例(恢复到指定时间点)  还原为从实例
- \* 主实例信息
- \* 复制用户
- \* 复制密码

[上一步](#)[下一步](#)

## 安装 MySQL

\* 安装MySQL方式 

默认  从模板中获取

\* Data路径



\* innodb\_buffer\_pool\_size

### 还原模板

#### 创建模板 ✕

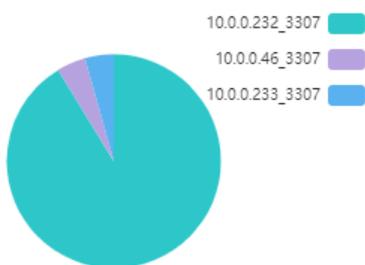
* 模板名称	<input style="width: 100%;" type="text" value="MySQL"/> <span>✕ 5/20</span>
* 数据源类型	<input style="width: 100%;" type="text" value="MySQL"/> <span>▼</span>
* 版本 <span>?</span> :	<input style="width: 100%;" type="text" value="5.7.26"/> <span>▼</span>
* 配置文件 <span>?</span> :	<input style="width: 100%;" type="text" value="/etc/my.cnf"/> <span>✕</span>
* 安装命令 <span>?</span> :	<input style="width: 100%;" type="text" value="mysqld --initialize --defaults-file=/etc//my.cnf"/> <span>✕</span>
* 启动命令 <span>?</span> :	<input style="width: 100%;" type="text" value="mysqld_safe--defaults-file=/etc/my.cnf &amp;"/> <span>✕</span>
* 关闭命令 <span>?</span> :	<input style="width: 100%;" type="text" value="mysql.server -P 3306 stop"/> <span>✕</span>
* BaseDir	<input style="width: 100%;" type="text" value="/usr/local/mysql"/> <span>✕</span>
* DataDir	<input style="width: 100%;" type="text" value="/data/mysql/data"/> <span>✕</span>

### 慢查询

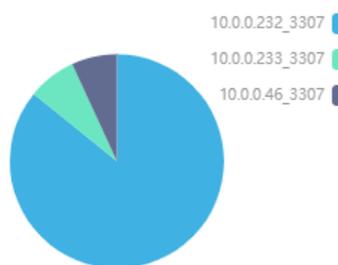
## 查看所有节点的慢查询

## 慢查询列表

实例慢查询个数



慢查询总时间(s)



慢查询总量 396079



最近一天

集群名称: 实例归属组: 时间范围: 

搜索

重置

## 查看个别节点的慢查询详情(慢查询详情和优化建议不赘述)

实例IP:  实例类型:

来源主机	用户名	数据库	SQL样本	总时间	总次数	最大执行时间	平均时间	操作
127.0.0.1	ark_monitor		select name, count from information_sche...	4.18	8632	0.01	0.00	<a href="#">详情</a> <a href="#">优化建议</a>
127.0.0.1	ark_monitor		select max(time) as max_time from inform...	3.87	8632	0.00	0.00	<a href="#">详情</a> <a href="#">优化建议</a>

&lt; 上一页 1 下一页 &gt;

## 高可用中心

## 认识集群状态

### 在线状态:

offline、online

### 状态:

problem、normal、maintain

### 角色:

master、standby、statistic

### 备注:

statistic 节点: 该角色的节点永远不能“提升为写”, 只能进行其他操作。如果提升为写角色, 需要先删除在重新添加该节点为 master 或者 standby 角色。

### 操作:

#### 上线节点:

将 offline+normal 状态的节点上线, 之后节点状态为 online+normal。只有 online 状态的节点才会参与高可用切换。

#### 添加节点:

新增一个集群节点, 初始化状态为 offline+normal (置为 online 时会判断同步情况和集群信息)。

#### 删除节点:

将 offline+maintain/problem 状态的节点删除。

**标记维护:**

将 online 状态的节点下线, 之后节点状态为 offline+maintain。

**标记正常:**

将 maintain/problem 状态的节点置为 normal 状态, 且原 online/offline 状态不变。

**提升为写:**

将 online 状态的 standby 节点提升为 master 角色, 原 master 节点变更为 standby 角色, 其他状态不变。

**强制下线:**

集群仅剩 master 节点时, 才能强制下线, 下线后节点状态为 offline+maintain。

**数据库高可用-配置高可用**

- 1、保证集群内的机器的个数为奇数, 基于 raft 协议投票机制
- 2、复制帐号即 mysql 中的 主从复制的帐号
- 3、switchover 脚本, 主要可以进行手动切换, 先要进行部署之后, 确定脚本的位置, 然后写入脚本的绝对路径
- 4、failover 脚本, 主要可以进行自动切换, 先要进行部署之后, 确定脚本的位置, 然后写入脚本的绝对路径
- 5、先点击新建一个哨兵站, 然后数据信息之后点击刷新, 就可以可, 更新配置信息完毕后, 点击 刷新也可

## 高可用(Arksentinel)配置详情

\* 由此查看您的哨兵站配置信息

三类哨兵

新建一个哨兵站

\* Arksentinel节点地址

\* Arksentinel通信端口

\* odown\_timeout(ms)

\* ping间隔时间(ms)

\* ping次数

\* 复制账号

\* 复制密码

switchover的脚本

failover的脚本

10.0.0.232 x 10.0.0.233 x 10.0.0.46 x

7977

10000

3000

3

repl

\*\*\*\*

/data/arkagent/package/arksentinel/bin/switchover

/data/arkagent/package/arksentinel/bin/failover

刷新

撤销新建

## 数据高可用-给项目增加高可用

## 增加 sentinel

- 1、只有写节点需要 写 VIP，其他节点可以不用写，默认也是读节点
- 2、新增的 sentinel 节点，是 offline 状态的，需要进行上线，方可进行正常使用，投票等

添加节点 ✕

\* 实例名称

\* 实例角色

读VIP

\* 网卡名称

\* 子网掩码

\* SSH用户名

\* SSH端口

\* 认证方式  密码  私钥

\* SSH密码

## 上线 sentinel

**上线前** (offline 状态, 且只有状态处于 normal 的机器才可以上线, 如果上线 写节点, 触发切换)

10.0.0.233 standby

---

IP地址: 10.0.0.233      端口: 3307

是否在线: offline      状态: normal

---

上线   删除

## 上线过程

上线节点 ×

实例名称	10.0.0.233_3307
角色	standby
角色	normal
备注	将指定的实例上线(只有normal状态的实例才允许上线)。如果是write节点, 点击上线后会导致切换。

确认   取消

**上线成功 (状态为 online, 且可以提升为, 写节点, 默认为 standby 备选节点状态)**

10.0.0.233 standby

---

IP地址: 10.0.0.233      端口: 3307

是否在线: **online**      状态: **normal**

---

**标记为维护**    **提升为写**

## 标记为维护

### 标记维护过程

- 1、normal 转换为 maintain 状态, online 转换为 offline
- 2、standby 和 statistic 可以进行维护下线
- 3、master 没有 维护选项, 只有其他节点被 提升为主节点 的时候才能进行维护



### 标记为正常

- 1、offline 不变 , maintain 转换为 normal
- 2、标记为正常以后, 实例是 offline 状态, 需要进行重新上线

### 3、重新上线后 offline 转换为 online 状态

10.0.0.233	standby
IP地址: 10.0.0.233	端口: 3307
是否在线: <b>offline</b>	状态: <b>normal</b>

**上线** **删除**

## 删除

- 1、只有标记为维护状态的实例 才能进行删除
- 2、删除的具体动机是什么？（是吧什么删除了，不能投票了吗？）

## 上线

10.0.0.233	standby
IP地址: 10.0.0.233	端口: 3307
是否在线: <b>online</b>	状态: <b>normal</b>

**标记为维护** **提升为写**

## 强制下线

- 1、强制下线，只能在 master 上进行操作
- 2、强制下线的条件，集群内只剩下 master 节点，下线后为 offline+maintain 状态

## 提升为写库

- 1、将 online 状态的 standby 节点提升为 master 角色，原 master 节点变更为 standby 角色，其他状态不变。
- 2、下线原主节点的 VIP ，将 VIP 漂移到新的主上

```
[root@zsh03 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.233 netmask 255.255.255.0 broadcast 10.0.0.255
    inet6 fe80::8a0:4f4a:250:6dfc prefixlen 64 scopeid 0x20<link>
    inet6 fe80::e2a4:96c4:3c34:e262 prefixlen 64 scopeid 0x20<link>
    inet6 fe80::455e:5102:61a2:5b9a prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:6d:5e:c2 txqueuelen 1000 (Ethernet)
    RX packets 40104094 bytes 17397898256 (16.2 GiB)
    RX errors 0 dropped 87 overruns 0 frame 0
    TX packets 12947370 bytes 5705006791 (5.3 GiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s3:45: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.45 netmask 255.255.255.0 broadcast 10.0.0.255
    ether 08:00:27:6d:5e:c2 txqueuelen 1000 (Ethernet)

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1 (Local Loopback)
    RX packets 1979859 bytes 656932759 (626.4 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1979859 bytes 656932759 (626.4 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

- 3、新节点状态 (online+normal+master)

10.0.0.233 master

---

IP地址: 10.0.0.233      端口: 3307

是否在线: online      状态: normal

写域名/VIP: 10.0.0.45

namespace: 三类项目

哨兵站: 三类哨兵

---

强制下线

#### 4、原节点状态变为 online + normal + standby

10.0.0.232 standby

---

IP地址: 10.0.0.232      端口: 3307

是否在线: online      状态: normal

---

标记为维护   提升为写

#### KIII 切换 (被动切换)

##### *kill slave 机器 看是否能拉起*

- 1、无法拉起, 没有自动拉起这个功能, 只有自动切换功能
- 2、参数控制?

3、启动之后，机器的 `super_read_only = OFF`，只有 `read_only` 和 `super_read_only` 都为 `ON` 才能使主机上线。

### kill master 机器看是否能切换

- 1、原节点下线 `offline` 状态变为 `problem`
- 2、新节点上线 `online`，VIP 漂移到新 master 上

#### 高可用详情

IP地址	角色	是否在线	状态	操作
10.0.0.232	master	online	normal	强制下线
10.0.0.46	standby	online	normal	标记为维护, 提升为写
10.0.0.233	standby	offline	problem	删除, 标记为正常

### 下线的机器重新加入集群? (原主库)

- 1、MySQL 管理 -> 实例列表 -> 清理下线节点

集群名称	实例名称	主机名称	角色	架构类型	版本	实例归属组	可用区	监控	运行状态	操作状态
三类项目	10.0.0.233_3307	zsh03	master	MS Cluster	5.7	DB部门	北1机房	📺	实例异常	详情 启动
三类项目	10.0.0.232_3307	zsh02	master	MS Cluster	5.7	DB部门	北1机房	📺	实例正常	详情 将会清

- 2、清理节点



### 3、集群添加从库

#### 实例信息

实例名称	端口	角色
10.0.0.46_3307	3307	slave
10.0.0.232_3307	3307	master
10.0.0.233_3307	3307	slave

### 4、设置 MySQL 的 只读

```
set global super_read_only = ON;
```

### 5、添加 sentinel



### 下线的机器重新加入集群? (原从库)

启动 mysql

```
./mysql.server -P 3307 start
```

开启 super\_read\_only

```
set global super_read_only = ON;
```

MySQL 管理 -> 实例管理 启动实例

高可用中心

删除 sentinel

添加 sentinel

### arkproxy 高可用

- 1、添加完成 sentinel 之后，会让添加 proxy 集群

添加Arkproxy节点

\* 节点地址

\* 访问方式  域名  VIP  其他

VIP

\* 网卡名称

\* 子网掩码

\* SSH用户名

\* 认证方式  密码  私钥

\* SSH密码

\* SSH端口

\* Arkproxy账号

\* Arkproxy密码

\* Arkproxy端口

\* Arkproxy管理端口

\* 最大连接数

2、mysql 和 proxy 共用一个 哨兵架构

## 用户中心

### 用户管理

### 可以查看所有用户的详细信息

关键字:

用户名	真实姓名	用户邮箱	电话	归属组	操作
—	admin			DB部门1 ...	<input type="button" value="编辑"/> <input type="button" value="删除"/>
用户名: admin 钉钉号:	电话: 真实姓名:	用户邮箱: 所属组: DB部门1,系统默认组	微信号:		
—	ark_control			DB部门1 ...	<input type="button" value="编辑"/> <input type="button" value="删除"/>
用户名: ark_control 钉钉号:	电话: 真实姓名:	用户邮箱: 所属组: DB部门1,系统默认组,DB部门	微信号:		

可以修改用户的基本信息，以及修改密码

### 基本设置

用户名	<input type="text" value="ark_control"/>
* 电话	<input type="text" value="电话"/>
* 邮箱	<input type="text" value="邮箱"/>
微信号	<input type="text" value="微信号"/>
钉钉号	<input type="text" value="钉钉号"/>
* 真实姓名	<input type="text" value="真实姓名"/>
所属组	<input type="text" value="DB部门1,系统默认组,DB部门"/>
<input type="button" value="提交"/>	

### 密码设置

原始密码	<input type="text" value="原始密码"/>
新密码	<input type="text" value="新密码"/>
<input type="button" value="更新密码"/>	

## 群组管理

## 创建组

创建组 ×

\* 组名称

描述

## 查看所有组 (管理员和超级管理员有管理权限, 和删除权限)

[+ 创建组](#)

组名称	描述	组管理员	组成员个数	管理组	操作
系统默认组		ark_control	2	<input checked="" type="checkbox"/>	<a href="#">详情</a> <a href="#">删除</a>
DB部门	DB部门	ark_control	1	<input type="checkbox"/>	<a href="#">详情</a> <a href="#">删除</a>
DB部门1	DB部门1	admin	2	<input type="checkbox"/>	<a href="#">详情</a>

**查看所有组 (普通用户, 只有查看权限)**

	用户名	真实姓名	用户邮箱	电话	归属组
+	test				系统默认组 ...
+	admin				DB部门1 ...
+	ark_control				DB部门1 ...

**是否可以管理组 (超级管理员拥有权限操作)**

管理组

是

否

---

否

---

**普通用户权限**

## 1、用户中心

只有群组管理

## 2、MySQL 管理

无

## 3、主机管理

无

#### 4、SQL 审核

数据源配置

sql 审核工单

#### 5、群组管理

群组管理只能看到，自己所属的组，不是的话无法查看

### **普通用户拥有组管理权限**

#### 1、用户中心，增加用户管理

只能查看，不能操作

#### 2、MySQL 管理

查看所有集群信息等，也可以操作

#### 3、主机管理

都可以操作

#### 4、SQL 审核

可以进行 审核 的参数设置

#### 5、产品管理

无变化

## 查看组的详细信息

### 组详情

#### 组基本信息

组ID: default      组名称: 系统默认组      组管理员: ark\_control  
管理组: 是      描述: -

编辑

#### 组成员

+ 添加成员

	用户名	电话	用户邮箱	微信号	真实姓名	操作
+	admin					删除
+	ark_control					

### 编辑组的信息

可以进行组名称,组描述,并进行组管理员的设置

编辑基本信息 ×

\* 组名称

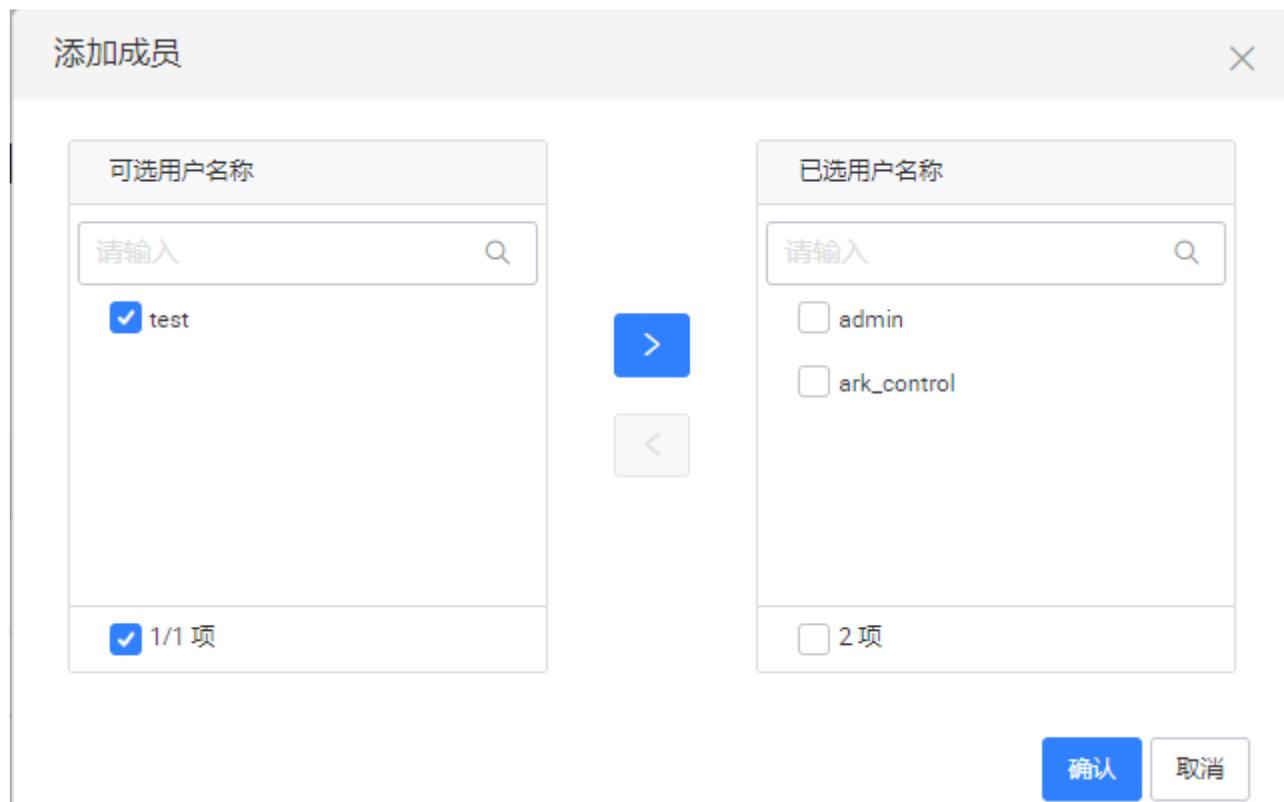
描述

\* 组管理员  ^

- admin
- ark\_control

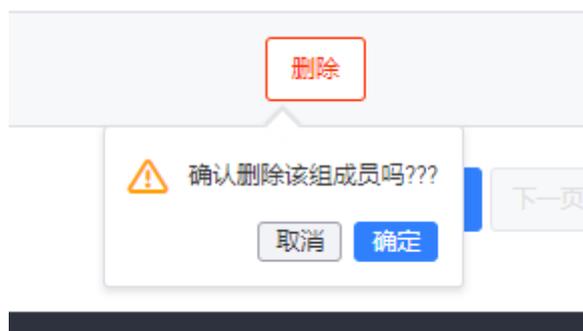
## 组成员管理

**超级管理员和组管理员可以添加组成员。**

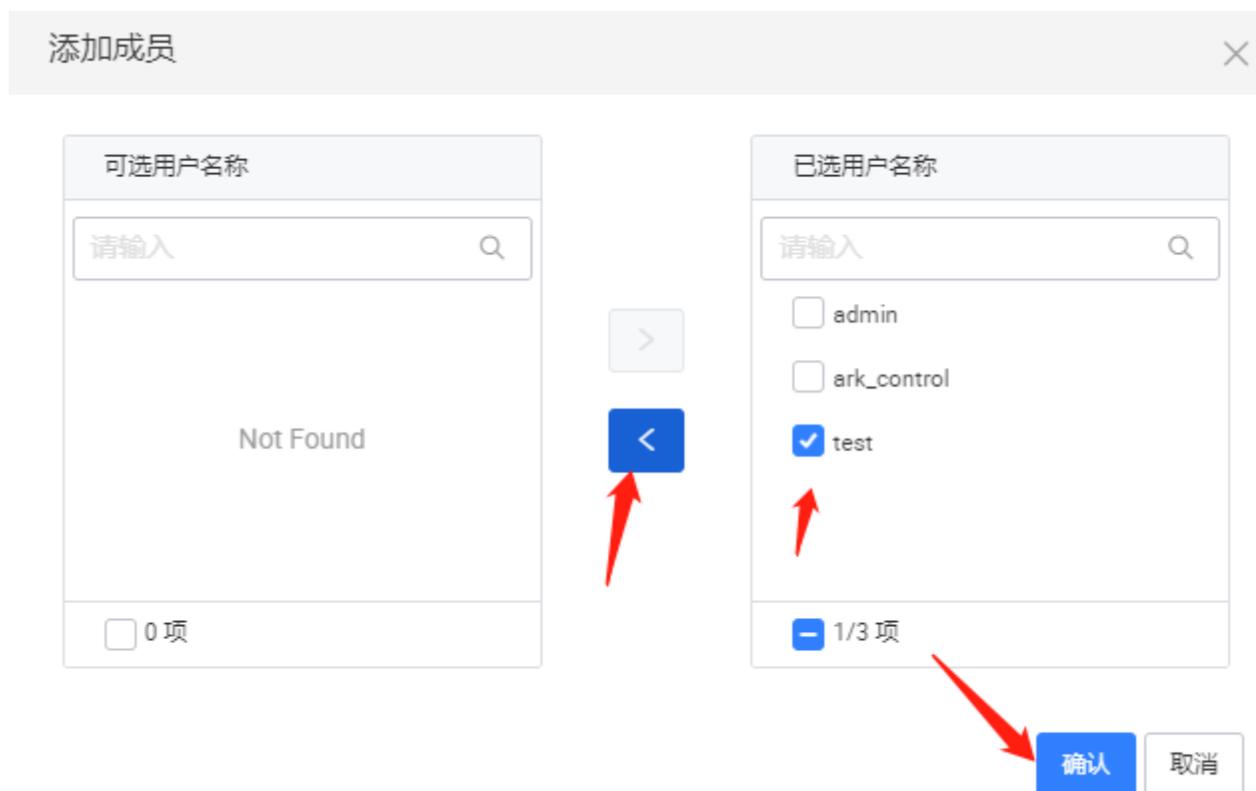


**超级管理员和组管理员可以删除组成员。**

### 1、直接删除

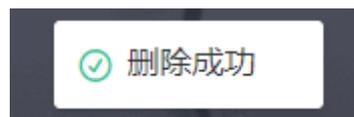


## 2、通过添加成员窗口删除

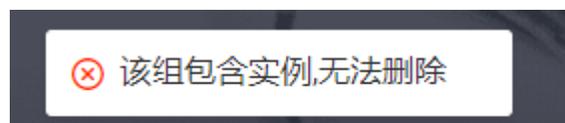


## 删除组

**超级管理员可以删除组（但是无法删除有实例的组）**



**如果组中包含实例，则无法删除**



## 配置中心

### 资源配置

#### 主机可用区

- 1、可以进行可用区的添加，删除，修改名称
- 2、如果该可用区内有机器的话，无法删除，但是可以修改，默认无法修改名称



### 资源规格设置\_MySQL

- 1、可以进行资源规格的设置，删除
- 2、可以在申请实例的时候进行选择



## 平台配置

### Arkcontrol 数据库管理帐号



## 配置库信息

配置库信息		?
数据库地址	端口	
10.0.0.231	3306	
账号: root		密码: 123456

## ES 信息

ES信息		?
ES地址	端口	
10.0.0.231	9200	
账号: -		密码: *****

## Agent 配置信息

Agent配置信息			?					
主机采集周期(秒)	<input type="text" value="60"/>		MySQL信息采集周期(秒)	<input type="text" value="60"/>		监控采集周期(秒)	<input type="text" value="10"/>	
备份任务轮询时间(秒)	<input type="text" value="10"/>		还原任务轮询时间(秒)	<input type="text" value="10"/>		BINLOG备份间隔时间(秒)	<input type="text" value="10"/>	
MySQL慢查询日志信息采集间隔(秒)	<input type="text" value="300"/>		巡检信息采集间隔(小时)	<input type="text" value="24"/>		巡检数据保留时间(天)	<input type="text" value="7"/>	

## 通知配置

提供了三个常用的通知渠道的配置

The screenshot displays three configuration panels for notifications:

- 邮件服务器配置 (Email Server Configuration):** Includes fields for '服务器地址' (Server Address) with a placeholder '服务器地址', '服务器端口' (Server Port) with a value of '0', '账号' (Account) with a placeholder '账号', and '密码' (Password) with a placeholder '密码'. It also features a '通知模块' (Notification Module) section with '报修通知' (Repair Notification) and '工单中心' (Ticket Center) buttons, and a '保存' (Save) button.
- 企业微信配置 (WeChat Enterprise Configuration):** Includes fields for 'Corp\_id' with a placeholder 'Corp\_id', 'Corp\_secret' with a placeholder 'Corp\_secret', and 'Agent\_id' with a value of '0'. It also features a '通知模块' (Notification Module) section with '报修通知' (Repair Notification) and '工单中心' (Ticket Center) buttons, and a '保存' (Save) button.
- 钉钉配置 (DingTalk Configuration):** Includes fields for 'Corp\_id' with a placeholder 'Corp\_id', 'Corp\_secret' with a placeholder 'Corp\_secret', and 'Agent\_id' with a value of '0'. It also features a '通知模块' (Notification Module) section with '报修通知' (Repair Notification) and '工单中心' (Ticket Center) buttons, and a '保存' (Save) button.

**登录配置** (Ldap,企业如果使用 Ldap 进行用户管理,则可以用此功能)

### LDAP服务器配置

启用LDAP



\* 服务器地址

ldap.example.com

\* 服务器端口

0

\* 账号 ?

请输入LDAP账号

\* 密码 ?

请输入密码



\* 绑定 DN ?

cn=user,dc=example,dc=com

\* 过滤条件 ?

例如: (objectClass=person)

\* 用户名字段 ?

uid

\* 登录字段 ?

uid,mobile

测试认证

(必须为一个正确的LDAP用户)

测试账号 ?

测试密码 ?

保存

测试

## Arkgate 管理

### MySQL 部署 Arkgate-cluster

## 新增 gate 基本信息

1 实例基本信息      2 安装方式配置      3 实例参数

集群名称

\* 集群名称

\* 归属组

\* 可用区

\* 数据源类型

\* 架构类型

\* 版本号

\* 端口

\* Master地址

\* Slave地址

\* 复制账号

\* 复制密码

下一步

## 新增 gate 安装方式

实例基本信息      2 安装方式配置

安装方式

\* 部署路径 ? :

\* 初始化账号 ? :

\* 初始化密码 ? :

生成随机密码

上一步    下一步

## 新增 gate 配置

实例基本信息      安装方式配置      3 实例参数配置

**核心参数**

参数配置 <sup>?</sup>:       默认     从模板中获取

innodb\_buffer\_pool\_size       G

sync\_binlog <sup>?</sup>:     

innodb\_flush\_log\_at\_trx\_commit <sup>?</sup>:     

expire\_log\_days <sup>?</sup>:     

[更多参数选择](#)

[上一步](#)    [提交](#)

如果内存不足,请跳过,表字段默认值问题,跳过

## 部署进度

**ArkGate节点**      [查看更多](#)

IP地址: 10.0.0.45

部署状态: **正在部署**

最新进度: 内存剩余1057M, 不足16384M

[清理](#)

ArkGate节点

查看更多

IP地址: 10.0.0.44

部署状态: **正在部署**

最新进度: Error 1101: BLOB, TEXT,  
GEOMETRY or JSON column  
'transfer\_columns' can't have a default value

清理

新增 gate

## 新增 gate 服务

### 基本信息设置

#### Arkgate部署

\* 任务名称  3/20

#### Gate服务

\* 归属组

\* Gate集群

\* 通道名称

#### 源端数据库信息

\* 源端来源  来自平台中  阿里云RDS  其他实例

\* 实例名称

\* 数据库账号

\* 数据库密码

\* Binlog文件

\* Binlog起始位置

#### 目的端数据库信息

## gate 服务

### Gate服务

* 归属组	DB部门
* Gate集群 ?	arkgate
* 通道名称 ?	td1

## gate 集群

如果未部署 gate 集群,请前往 MySQL 部署中部署 gate 集群,目前 gate 集群只能是 MySQL5.7 版本,架构选择 Arkgate cluster

### 通道名称

在 gate 服务中心,由于一套服务,可以支持多数据源,内部使用通道的模式,隔离资源,类似于 MySQL5.7 的多源复制

## 源端数据库信息

### 源端数据库信息

* 源端来源 ?	<input checked="" type="radio"/> 来自平台中	<input type="radio"/> 阿里云RDS	<input type="radio"/> 其他实例
* 实例名称	10.0.0.232_3307		
* 数据库账号	root	* 数据库密码	....
* Binlog文件	mysql-bin.000143	* Binlog起始位置	4

## 目标端数据库信息

目的端数据库信息 编辑 ?

数据库地址: 10.0.0.45      数据库端口: 3311      账号: root  
密码: 👁

## 数据同步详情

### 基本信息(开启或者关闭 gate, 添加备用节点等操作)

数据同步详情

基本信息 过滤规则 同步参数

同步基本信息 停止 开启

任务名称: ceshi1      通道名称: td1      归属组: DB部门  
运行状态: gate出错      延迟: 0s      创建人: ark\_control  
创建时间: 2020-02-02 01:11:29      更新时间: 2020-02-02 01:11:29

源端数据库信息 添加备用节点 ?

数据库地址: 10.0.0.232      数据库端口: 3307      账号: root  
密码: 👁      Binlog起始文件: mysql-bin.000143      Binlog起始位置: 4  
数据源来源: 来自平台中

目的端数据库信息 编辑 ?

数据库地址: 10.0.0.45      数据库端口: 3311      账号: root  
密码: 👁

### 过滤规则(对传输的 SQL 进行过滤)

#### 黑名单

db1 % 删除

新增 保存

#### 表黑名单

db1 t1 db2 t2 删除

新增 保存

#### 列过滤

db1 t2 pass 删除

新增 保存

#### DDL过滤

db1 t2 DROPTABLE 删除

## 同步参数

## arkgate 参数

## 数据同步详情

基本信息

过滤规则

同步参数

## arkgate参数:

- \* 同步线程
- \* 工作队列长度
- \* 数据库架构类型  原生MySQL  阿里云(RDS)
- \* 是否包含全部信息  包含  不包含
- \* 数据保留时长(小时)
- \* 清理数据线程数
- \* 每次清理行数

保存

## ETL 参数

### ETL参数:

* 工作线程	<input type="text" value="8"/>
* 开始位置	<input type="text" value="0"/>
* 工作步长	<input type="text" value="4000"/>
* 缓存大小	<input type="text" value="8000"/>
* 双活开关	<input checked="" type="checkbox"/> 
* 出错处理	<input checked="" type="radio"/> 跳过 <input type="radio"/> 覆盖执行 <input type="radio"/> 业务方提供异常处理接口 <input type="radio"/> 程序退出

## Redis 管理-BATE

### 部署中心

## 部署列表

[给已有集群添加从库](#)

[给已有集群添加从库](#)

## 部署集群

[部署新的集群](#)

### 填写实例的基本信息

1 实例基本信息      2 安装配置      3 节点选择

集群名称

\* 集群名称  2/20

\* 归属组 ? :  ▼

\* 可用区 ? :  ▼

\* 版本  ▼

\* 使用类型 ? :  Cache  Storage

\* 架构类型 ? :  ▼

下一步

### 填写 Redis 的集群的参数信息,以及 AOF

#### 安装方式

* maxmemory(G)	<input type="text" value="1"/>
* maxmemory-policy ?:	<input type="text" value="volatile-lru"/>
* AOF	<input checked="" type="radio"/> 开启 <input type="radio"/> 关闭
* appendfsync ?:	<input type="text" value="always"/>
* 密码 ?:	<input type="text" value="123456"/> 6/20

## 集群分片(首创)

实例基本信息      安装配置      3 节点选择

核心参数

\* 分片数 <sup>?</sup>:       \* 从库数量 <sup>?</sup>:       \* 使用机器数 <sup>?</sup>:       [智能推荐](#)

分片号	master地址	从库地址	端口	操作
分片号1	10.0.0.231	10.0.0.232,10.0.0.233	6379	
分片号2	10.0.0.231	10.0.0.232,10.0.0.233	6380	

[上一步](#)      [提交](#)

## 查看,删除已有集群

### 基本信息

集群名称:	测试	归属组:	DB部门	可用区:	北1机房
版本号:	4.0.14	使用类型:	Cache	架构类型:	主从集群
状态:	<span>部署成功</span>	部署者:	ark_control		

### 节点与部署状态信息

分片号	地址	角色	端口	状态	操作
	10.0.0.231	master	6379	<span>部署成功</span>	<a href="#">查看部署信息</a>
分片号1	10.0.0.232	slave	6379	<span>部署成功</span>	<a href="#">查看部署信息</a>
	10.0.0.233	slave	6379	<span>部署成功</span>	<a href="#">查看部署信息</a>

#### 进度信息

copy redis bin OK, id:1  
create redis config file OK, id:1  
deploy task success, id:1.

集群名称	使用类型	架构类型	版本	归属组	状态	部署者	部署时间	操作
测试	Cache	主从集群	4.0.14	DB部门	<span>部署成功</span>	ark_control	2020-02-01 17:52:20	<a href="#">详情</a> <a href="#">删除</a>

## 集群列表

## 集群基本信息

### 查看集群的基本信息

#### 集群列表

集群名称:  归属组:  架构类型:

运行状态:

集群名称	架构类型	归属组	运行状态	采集时间	操作
测试	主从集群	DB部门	在线	2020-02-01 18:00:33	<input type="button" value="详情"/> <input type="button" value="删除"/>

< 前一页 1 下一页 >

### 删除集群(不演示了)

## 集群详细信息

### 实例信息

可以查看集群内,所有实例的状态信息,以及监控告警,还可以对告警进行屏蔽等操作

## 实例信息

实例名称	端口	角色	主机名	监控	实例版本	运行状态	操作
10.0.0.231_6379	6379	master	10.0.0.231		4.0.14	在线	告警 ∨
10.0.0.232_6379	6379	slave	10.0.0.232		4.0.14	在线	告警 ∨
10.0.0.233_6379	6379	slave	10.0.0.233		4.0.14	在线	告警 ∨
10.0.0.231_6380	6380	master	10.0.0.231		4.0.14	在线	告警 ∨
10.0.0.232_6380	6380	slave	10.0.0.232		4.0.14	在线	告警 ∨
10.0.0.233_6380	6380	slave	10.0.0.233		4.0.14	在线	告警 ∨

## 集群信息

可以对集群的基本信息进行修改

## 集群信息

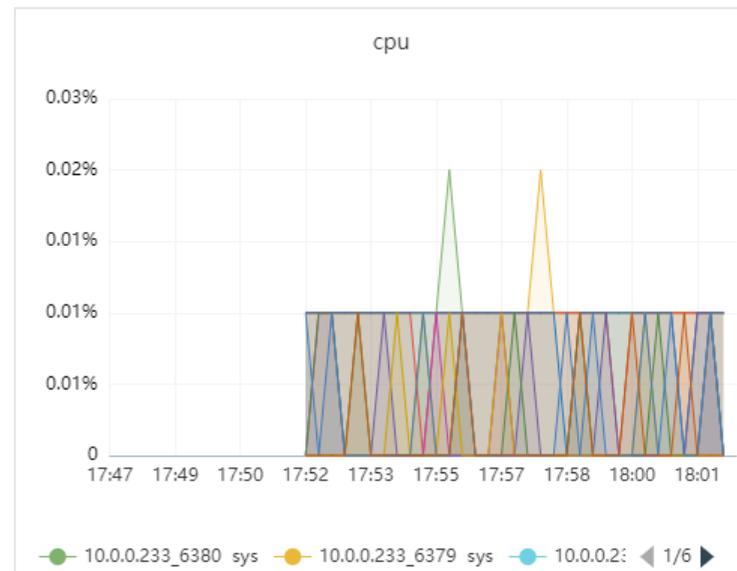
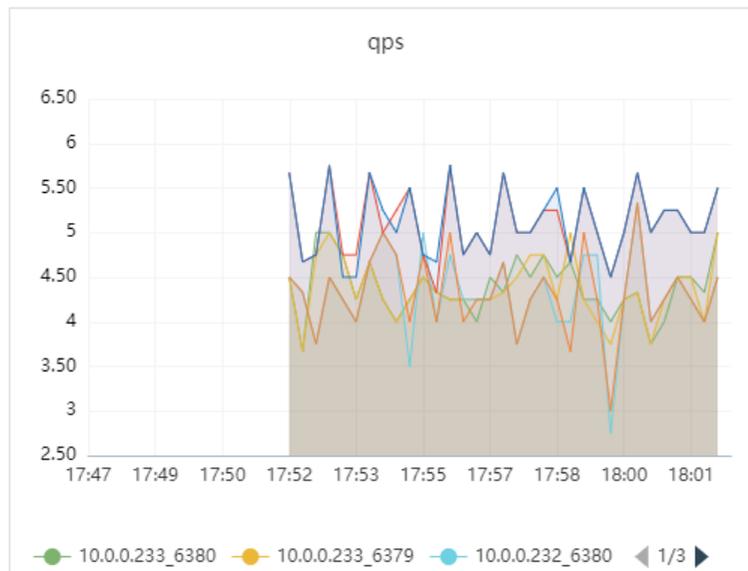
集群名称: 测试 / 集群备注: / 归属组: DB部门 /  
架构类型: 主从集群

## 监报告警

可以查看集群的 Redis 的监报告警图表

### 监控告警

时间: **15分钟** 1小时 1天 7天 1个月 自定义



### 实例列表

## 实例的基本信息展示

### 所有实例的基本信息,以及根据条件进行筛选

#### 实例列表

集群名称:	<input type="text" value="请输入集群名称"/>	架构类型:	<input type="text" value="请选择架构类型"/>	角色:	<input type="text" value="请选择角色"/>
实例归属组:	<input type="text" value="请选择归属组"/>	可用区:	<input type="text" value="请选择可用区"/>	运行状态:	<input type="text" value="请选择运行状态"/>
<input type="button" value="搜索"/> <input type="button" value="重置"/>					

集群名称	实例名称	角色	架构类型	版本	实例归属组	可用区	监控	运行状态	到期时间	操作
测试	10.0.0.233_6 380	slave	主从集群	4.0.14	DB部门	北1机房		在线	2030-12-31	<a href="#">详情</a>
测试	10.0.0.232_6 380	slave	主从集群	4.0.14	DB部门	北1机房		在线	2030-12-31	<a href="#">详情</a>

### 删除某个实例(不演示了)

## 实例详情信息

### 实例信息

显示实例的更详细的基本信息

### 实例信息

集群名称:	测试	归属组:	DB部门	可用区:	北1机房
实例名称:	10.0.0.232_6380	端口:	6380	角色:	slave
主机名称:	10.0.0.232	架构类型:	主从集群	实例版本:	4.0.14
实例状态:	<span style="background-color: #28a745; color: white; padding: 2px;">在线</span>	创建时间:	2020-02-01 17:52:11	到期时间:	2030-12-31
备注:	-				

### 实例配置信息

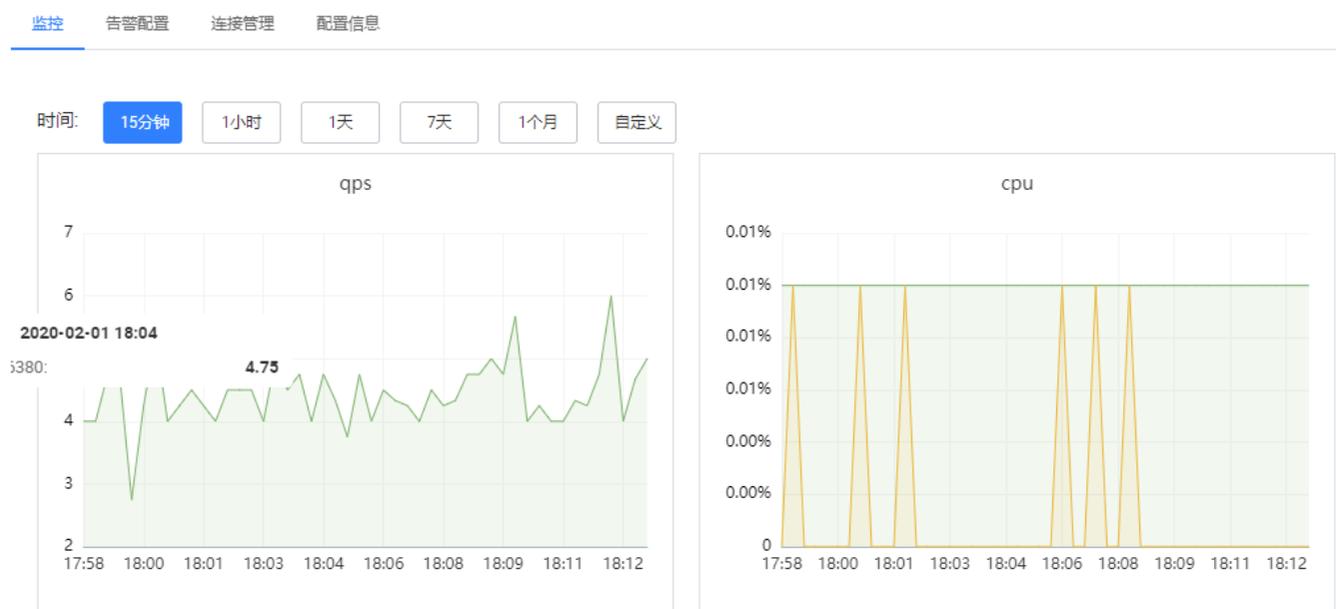
显示实例的一些配置信息

### 实例配置信息

maxmemory(G):	1	maxmemory-policy:	volatile-lru	AOF:	开启
appendfsync:	-	安装路径:	/data/redis/redis6380		

## 监控告警

## 监控(查看基本的状态告警)



## 告警配置(新增告警,告警通知负责人)

监控 告警配置 连接管理 配置信息

+ 新增告警

监控项	预警阈值	报警阈值	报警说明	状态	接收人	操作
Qps	50000	60000	Redis qps状态	normal	ark_control	<a href="#">屏蔽</a> <a href="#">删除</a>

## 连接管理(查看正在连接的连接)

监控 告警配置 连接管理 配置信息

来源IP	连接数量
10.0.0.231	2

&lt; 上一页 1 下一页 &gt;

## 配置信息(进行巡检的配置设置)

监控 告警配置 连接管理 配置信息

参数名称	参数值
maxmemory	1G
maxmemory-policy	volatile-lru
slowlog-max-len	128
repl-backlog-size	33554432
repl-backlog-ttl	3600
maxclients	4064
slave-priority	80
client-output-buffer-limit	normal 0 0 0 slave 0 0 0 pubsub 33554432 8388608 60

巡检中心

## 巡检信息

### 巡检

巡检 巡检配置

巡检项:  集群名称:

集群名称	实例名称	主机名称	端口	数据大小	巡检结果	巡检时间
------	------	------	----	------	------	------

没有数据

## 巡检配置

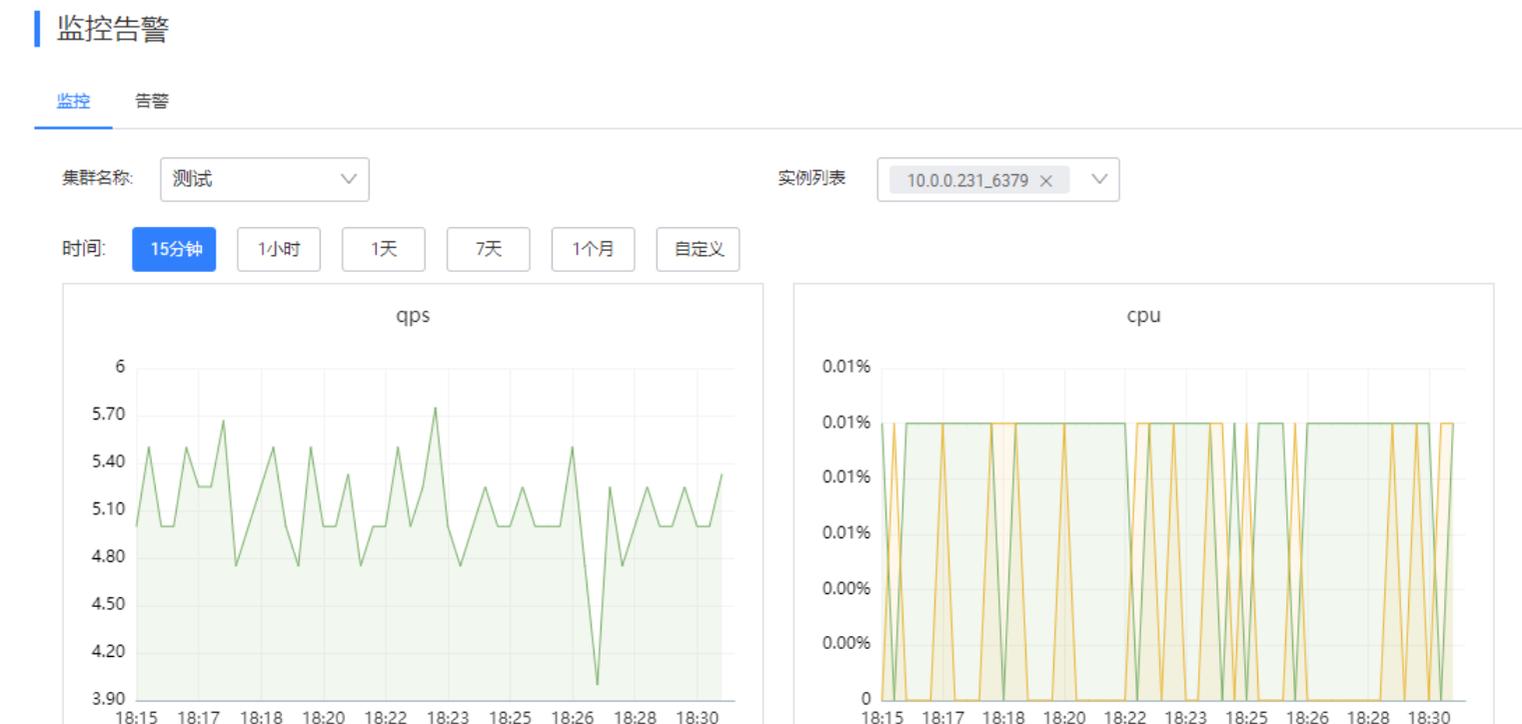
查看和设置巡检的配置,可以通过平台配置-巡检时间间隔等设置巡检操作

## | 巡检

巡检 [巡检配置](#)

巡检项	巡检阈值	巡检说明
主机时间是否正确	2s	主机时间是否正确
Swap使用率	0%	Swap使用超过一定比例
是否关闭内存大页	<input checked="" type="checkbox"/>	内存大页是否关闭 1 表示开启巡检项, 0 表示关闭巡检项
系统内存使用比例	80%	当系统内存使用比例超过一定阈值时报警
内存碎片率	5	Redis 实例的内存碎片率超过一定的阈值时报警
从库是否只读	<input checked="" type="checkbox"/>	当实例的slave-read-only 参数为no 时报警 1 代表开启巡检, 0 代表关闭巡检
连接使用百分比	70%	连接使用百分比

## 监控告警

**监控(查看 Redis 集群的告警,可以进行集群和集群内主机的选择)****告警(查看告警,屏蔽和删除告警)**

### 监控告警

监控 告警

主机名称	实例名称	归属组	集群名称	可用区	监控项	告警信息	最近报警时间	操作
没有数据								

**备份中心**

## 备份列表

## 查看备份列表

备份

## 备份列表

集群名称:  备份状态:  RDB备份:

AOF备份:

集群名称	归属组	架构类型	AOF备份	RDB备份	备份状态	集群状态	操作
测试	DB部门	主从集群	<span>开启</span>	<span>开启</span>	<span>备份成功</span>	<span>在线</span>	<input type="button" value="详情"/> <input type="button" value="暂停备份"/> <input type="button" value="删除备份"/>

## 暂停,删除备份(删除备份不演示了)

集群名称	归属组	架构类型	AOF备份	RDB备份	备份状态	集群状态	操作
测试	DB部门	主从集群	<span>开启</span>	<span>开启</span>	<span>备份成功</span>	<span>在线</span>	<input type="button" value="详情"/> <input type="button" value="开启备份"/> <input type="button" value="删除备份"/>

## 部署备份

## 部署备份设置

部署备份 ×

\* RDB备份  开启  未开启

\* RDB备份实例  主实例  从实例

\* RDB备份保留(天)  天

\* AOF备份  开启  未开启

\* AOF备份实例  主实例  从实例

\* AOF备份保留(天)  天

\* 备份周期  ▼

\* 远程存储机器  ▼

\* 远程存储目录

## 部署备份详情

### 基本信息(Redis 备份信息,可以进行设置)

#### 备份详情

基本信息 RDB备份 AOF备份 备份日志

#### 备份信息

集群名称:	测试	归属组:	DB部门	最近RDB备份:	备份成功
最近AOF备份:	备份成功	最近RDB备份时间:	2020-02-01 21:36:58	最近AOF备份时间:	2020-02-01 21:36:45
部署者:	ark_control				

#### 配置信息

编辑

AOF备份:	开启	AOF备份实例:	从实例	AOF备份保留(天):	7
RDB备份:	开启	RDB备份实例:	主实例	RDB备份保留(天):	7
远程存储机器:	10.0.0.231	远程存储目录:	/data	传输速度(M):	100
备份周期:	每隔三小时				

## RDB 备份(备份完成后,会进行压缩)

## 备份详情

基本信息 **RDB备份** AOF备份 备份日志

实例名称:  备份状态:

分片号	实例名称	存储机器	备份集名称	数据大小(B)	备份日期	备份状态	是否压缩	备份集待清理日期
2	10.0.0.2 31_6380	10.0.0.231	/data/10.0.0.231_6380_20200201083 658.rdb	177	2020-02-01 2 1:36:58	备份成功	未压缩	2020-02-08 0 8:36:58
1	10.0.0.2 31_6379	10.0.0.231	/data/10.0.0.231_6379_20200201083 645.rdb	177	2020-02-01 2 1:36:45	备份成功	未压缩	2020-02-08 0 8:36:45
2	10.0.0.2 31_6380	10.0.0.231	/data/10.0.0.231_6380_20200201053 646.rdb.tar.gz	306	2020-02-01 1 8:36:46	备份成功	压缩成功	2020-02-08 0 5:36:46
1	10.0.0.2 31_6379	10.0.0.231	/data/10.0.0.231_6379_20200201053 642.rdb.tar.gz	308	2020-02-01 1 8:36:42	备份成功	压缩成功	2020-02-08 0 5:36:42

## AOF 备份(备份完成后,会进行压缩)

## 备份详情

基本信息 RDB备份 **AOF备份** 备份日志

实例名称:  备份状态:

分片号	实例名称	存储机器	备份集名称	数据大小(B)	备份日期	备份状态	是否压缩	备份集待清理日期
2	10.0.0.2 32_6380	10.0.0.231	/data/10.0.0.232_6380_20200201083 645.aof	0	2020-02-01 2 1:36:45	备份成功	未压缩	2020-02-08 0 8:36:45
1	10.0.0.2 32_6379	10.0.0.231	/data/10.0.0.232_6379_20200201083 643.aof	0	2020-02-01 2 1:36:43	备份成功	未压缩	2020-02-08 0 8:36:43
2	10.0.0.2 32_6380	10.0.0.231	/data/10.0.0.232_6380_20200201053 639.aof.tar.gz	133	2020-02-01 1 8:36:39	备份成功	压缩成功	2020-02-08 0 5:36:39
1	10.0.0.2 32_6379	10.0.0.231	/data/10.0.0.232_6379_20200201053 637.aof.tar.gz	132	2020-02-01 1 8:36:37	备份成功	压缩成功	2020-02-08 0 5:36:37

## 备份日志

### 备份详情

基本信息 RDB备份 AOF备份 **备份日志**

实例名称:  备份类型:  备份状态:

备份类型	分片号	实例名称	开始时间	结束时间	备份状态	压缩状态	详细日志
RDB	2	10.0.0.231_6380	2020-02-01 21:36:58	2020-02-01 21:37:01	备份成功	未压缩	<a href="#">查看详细日志</a>
AOF	2	10.0.0.232_6380	2020-02-01 21:36:45	2020-02-01 21:36:47	备份成功	未压缩	<a href="#">查看详细日志</a>

### 详细备份日志

#### 备份日志

日志来源主机	日志来源组件	日志信息	时间
10.0.0.231	redis_backup:backup_rdb	Start Backup RDB for instance 10.0.0.231_6380	2020-02-01 21:36:59
10.0.0.231	redis_backup:backup_rdb	Execute BGSAVE Command	2020-02-01 21:36:59
10.0.0.231	redis_backup:backup_rdb	Backup RDB File Success	2020-02-01 21:37:01
10.0.0.231	redis_backup:Sending	Start Transfer file arkncat 10.0.0.231 7000 < /data/redis/redis6380/10.0.0.231_6...	2020-02-01 21:37:16
10.0.0.231	redis_backup:Sending	Transfer File /data/redis/redis6380/10.0.0.231_6380_20200201083658.rdb Success	2020-02-01 21:37:17
10.0.0.231	redis_backup:Receiving	Receiving File /data/10.0.0.231_6380_20200201083658.rdb Success	2020-02-01 21:37:17

## 配置信息(备份的配置)

### 配置信息

编辑

AOF备份:	开启	AOF备份实例:	从实例	AOF备份保留(天):	7
RDB备份:	开启	RDB备份实例:	主实例	RDB备份保留(天):	7
远程存储机器:	10.0.0.231	远程存储目录:	/data	传输速度(M):	100
备份周期:	每隔三小时				

## 慢查询中心

### 慢查询列表(展示当前集群的慢查询,可以进行时间的筛选)



### 慢查询详情(查看某个实例的慢查询详情)

类型	命令	语句样本	总次数	总耗时(ms)	平均时间(ms)	操作
没有数据						

## Redis 高可用(功能还未完善)

### 高可用列表

可以查看集群是否已经有了高可用配置,如果有会显示绿色

### 高可用列表

集群名称:  架构类型:  归属组:

运行状态:  哨兵站:

哨兵站	集群名称	架构类型	归属组	运行状态	高可用	操作
	测试	主从集群	DB部门	在线	未添加	<input type="button" value="详情"/>

### 新增高可用配置

#### 添加节点

添加节点 ×

\* 分片号

\* 实例名称

\* 实例角色

\* 哨兵站

\* Redis节点密码

访问方式  域名  VIP

写域名

读域名

## 节点操作

标记为上线

标记为维护

标记为正常

提升为写

强制下线

删除节点