

在线oj项目-Nacos

Nacos

<https://nacos.io/>

安装

拉取nacos镜像

```
1 docker pull nacos/nacos-server:v2.2.3
```

配置MySQL数据库

为什么要外置MySQL数据库

数据持久性：使用MySQL作为外置数据库可以确保数据被持久化存储，这对于确保服务的稳定性和数据的可靠性至关重要。

高可用性：Nacos支持集群部署，而使用MySQL作为共享的数据存储可以确保集群中各个节点之间的数据一致性。此外，MySQL自身也支持高可用性和故障转移，如使用主从复制或集群解决方案，从而进一步提高系统的可用性。

性能优化：使用nacos内置的数据库虽然能够简化部署，但是性能受到限制，外置的MySQL可以根据需要进行优化和扩展，满足更高的性能要求。

易于管理和维护：首先我们系统本身采用MySQL数据库，nacos外置数据库和我们系统采用同样的数据库可以保证技术使用的统一。简化了数据库的管理和维护工作，降低运维成本。

创建表sql

<https://github.com/alibaba/nacos/blob/master/distribution/conf/mysql-schema.sql>

```
1  /*****  
2  /*  库名称 = bitoj_nacos_local          */  
3  /*****/  
4  create database bitoj_nacos_local;  
5  
6  /*****/  
7  /*  表名称 = config_info              */  
8  /*****/
```

```

9 CREATE TABLE `config_info` (
10   `id` bigint(20) NOT NULL AUTO_INCREMENT COMMENT 'id',
11   `data_id` varchar(255) NOT NULL COMMENT 'data_id',
12   `group_id` varchar(128) DEFAULT NULL COMMENT 'group_id',
13   `content` longtext NOT NULL COMMENT 'content',
14   `md5` varchar(32) DEFAULT NULL COMMENT 'md5',
15   `gmt_create` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '创建时间',
16   `gmt_modified` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '修改时间',
17   `src_user` text COMMENT 'source user',
18   `src_ip` varchar(50) DEFAULT NULL COMMENT 'source ip',
19   `app_name` varchar(128) DEFAULT NULL COMMENT 'app_name',
20   `tenant_id` varchar(128) DEFAULT '' COMMENT '租户字段',
21   `c_desc` varchar(256) DEFAULT NULL COMMENT 'configuration description',
22   `c_use` varchar(64) DEFAULT NULL COMMENT 'configuration usage',
23   `effect` varchar(64) DEFAULT NULL COMMENT '配置生效的描述',
24   `type` varchar(64) DEFAULT NULL COMMENT '配置的类型',
25   `c_schema` text COMMENT '配置的模式',
26   `encrypted_data_key` text NOT NULL COMMENT '密钥',
27   PRIMARY KEY (`id`),
28   UNIQUE KEY `uk_configinfo_datagrouptenant` (`data_id`,`group_id`,`tenant_id`)
29 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin COMMENT='config_info';
30
31 /*****
32 /* 表名称 = config_info_aggr */
33 *****/
34 CREATE TABLE `config_info_aggr` (
35   `id` bigint(20) NOT NULL AUTO_INCREMENT COMMENT 'id',
36   `data_id` varchar(255) NOT NULL COMMENT 'data_id',
37   `group_id` varchar(128) NOT NULL COMMENT 'group_id',
38   `datum_id` varchar(255) NOT NULL COMMENT 'datum_id',
39   `content` longtext NOT NULL COMMENT '内容',
40   `gmt_modified` datetime NOT NULL COMMENT '修改时间',
41   `app_name` varchar(128) DEFAULT NULL COMMENT 'app_name',
42   `tenant_id` varchar(128) DEFAULT '' COMMENT '租户字段',
43   PRIMARY KEY (`id`),
44   UNIQUE KEY `uk_configinfoaggr_datagrouptenantdatum`
    (`data_id`,`group_id`,`tenant_id`,`datum_id`)
45 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin COMMENT='增加租户字段';
46
47
48 /*****
49 /* 表名称 = config_info_beta */
50 *****/
51 CREATE TABLE `config_info_beta` (
52   `id` bigint(20) NOT NULL AUTO_INCREMENT COMMENT 'id',
53   `data_id` varchar(255) NOT NULL COMMENT 'data_id',
54   `group_id` varchar(128) NOT NULL COMMENT 'group_id',

```

```

55 `app_name` varchar(128) DEFAULT NULL COMMENT 'app_name',
56 `content` longtext NOT NULL COMMENT 'content',
57 `beta_ips` varchar(1024) DEFAULT NULL COMMENT 'betaIps',
58 `md5` varchar(32) DEFAULT NULL COMMENT 'md5',
59 `gmt_create` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '创建时间',
60 `gmt_modified` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '修改时间',
61 `src_user` text COMMENT 'source user',
62 `src_ip` varchar(50) DEFAULT NULL COMMENT 'source ip',
63 `tenant_id` varchar(128) DEFAULT '' COMMENT '租户字段',
64 `encrypted_data_key` text NOT NULL COMMENT '密钥',
65 PRIMARY KEY (`id`),
66 UNIQUE KEY `uk_configinfobeta_datagrouptenant`
  (`data_id`,`group_id`,`tenant_id`)
67 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin
  COMMENT='config_info_beta';
68
69 /*****
70 /* 表名称 = config_info_tag */
71 *****/
72 CREATE TABLE `config_info_tag` (
73 `id` bigint(20) NOT NULL AUTO_INCREMENT COMMENT 'id',
74 `data_id` varchar(255) NOT NULL COMMENT 'data_id',
75 `group_id` varchar(128) NOT NULL COMMENT 'group_id',
76 `tenant_id` varchar(128) DEFAULT '' COMMENT 'tenant_id',
77 `tag_id` varchar(128) NOT NULL COMMENT 'tag_id',
78 `app_name` varchar(128) DEFAULT NULL COMMENT 'app_name',
79 `content` longtext NOT NULL COMMENT 'content',
80 `md5` varchar(32) DEFAULT NULL COMMENT 'md5',
81 `gmt_create` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '创建时间',
82 `gmt_modified` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '修改时间',
83 `src_user` text COMMENT 'source user',
84 `src_ip` varchar(50) DEFAULT NULL COMMENT 'source ip',
85 PRIMARY KEY (`id`),
86 UNIQUE KEY `uk_configinfotag_datagrouptenanttag`
  (`data_id`,`group_id`,`tenant_id`,`tag_id`)
87 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin
  COMMENT='config_info_tag';
88
89 /*****
90 /* 表名称 = config_tags_relation */
91 *****/
92 CREATE TABLE `config_tags_relation` (
93 `id` bigint(20) NOT NULL COMMENT 'id',
94 `tag_name` varchar(128) NOT NULL COMMENT 'tag_name',
95 `tag_type` varchar(64) DEFAULT NULL COMMENT 'tag_type',
96 `data_id` varchar(255) NOT NULL COMMENT 'data_id',
97 `group_id` varchar(128) NOT NULL COMMENT 'group_id',

```

```

98  `tenant_id` varchar(128) DEFAULT '' COMMENT 'tenant_id',
99  `nid` bigint(20) NOT NULL AUTO_INCREMENT COMMENT 'nid, 自增长标识',
100 PRIMARY KEY (`nid`),
101 UNIQUE KEY `uk_configtagrelation_configidtag` (`id`,`tag_name`,`tag_type`),
102 KEY `idx_tenant_id` (`tenant_id`)
103 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin
    COMMENT='config_tag_relation';
104
105 /*****
106  /* 表名称 = group_capacity */
107  *****/
108 CREATE TABLE `group_capacity` (
109  `id` bigint(20) unsigned NOT NULL AUTO_INCREMENT COMMENT '主键ID',
110  `group_id` varchar(128) NOT NULL DEFAULT '' COMMENT 'Group ID, 空字符表示整个集群',
111  `quota` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '配额, 0表示使用默认值',
112  `usage` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '使用量',
113  `max_size` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '单个配置大小上限, 单位为字节, 0表示使用默认值',
114  `max_aggr_count` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '聚合子配置最大个数, , 0表示使用默认值',
115  `max_aggr_size` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '单个聚合数据的子配置大小上限, 单位为字节, 0表示使用默认值',
116  `max_history_count` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '最大变更历史数量',
117  `gmt_create` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '创建时间',
118  `gmt_modified` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '修改时间',
119  PRIMARY KEY (`id`),
120  UNIQUE KEY `uk_group_id` (`group_id`)
121 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin COMMENT='集群、各Group容量信息表';
122
123 /*****
124  /* 表名称 = his_config_info */
125  *****/
126 CREATE TABLE `his_config_info` (
127  `id` bigint(20) unsigned NOT NULL COMMENT 'id',
128  `nid` bigint(20) unsigned NOT NULL AUTO_INCREMENT COMMENT 'nid, 自增标识',
129  `data_id` varchar(255) NOT NULL COMMENT 'data_id',
130  `group_id` varchar(128) NOT NULL COMMENT 'group_id',
131  `app_name` varchar(128) DEFAULT NULL COMMENT 'app_name',
132  `content` longtext NOT NULL COMMENT 'content',
133  `md5` varchar(32) DEFAULT NULL COMMENT 'md5',
134  `gmt_create` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '创建时间',
135  `gmt_modified` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '修改时间',
136  `src_user` text COMMENT 'source user',
137  `src_ip` varchar(50) DEFAULT NULL COMMENT 'source ip',

```

```

138 `op_type` char(10) DEFAULT NULL COMMENT 'operation type',
139 `tenant_id` varchar(128) DEFAULT '' COMMENT '租户字段',
140 `encrypted_data_key` text NOT NULL COMMENT '密钥',
141 PRIMARY KEY (`nid`),
142 KEY `idx_gmt_create` (`gmt_create`),
143 KEY `idx_gmt_modified` (`gmt_modified`),
144 KEY `idx_did` (`data_id`)
145 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin COMMENT='多租户改造';
146
147
148 /*****
149 /* 表名称 = tenant_capacity */
150 *****/
151 CREATE TABLE `tenant_capacity` (
152   `id` bigint(20) unsigned NOT NULL AUTO_INCREMENT COMMENT '主键ID',
153   `tenant_id` varchar(128) NOT NULL DEFAULT '' COMMENT 'Tenant ID',
154   `quota` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '配额, 0表示使用默认值',
155   `usage` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '使用量',
156   `max_size` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '单个配置大小上限, 单位为字节, 0表示使用默认值',
157   `max_aggr_count` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '聚合子配置最大个数',
158   `max_aggr_size` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '单个聚合数据的子配置大小上限, 单位为字节, 0表示使用默认值',
159   `max_history_count` int(10) unsigned NOT NULL DEFAULT '0' COMMENT '最大变更历史数量',
160   `gmt_create` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '创建时间',
161   `gmt_modified` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT '修改时间',
162   PRIMARY KEY (`id`),
163   UNIQUE KEY `uk_tenant_id` (`tenant_id`)
164 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin COMMENT='租户容量信息表';
165
166
167 CREATE TABLE `tenant_info` (
168   `id` bigint(20) NOT NULL AUTO_INCREMENT COMMENT 'id',
169   `kp` varchar(128) NOT NULL COMMENT 'kp',
170   `tenant_id` varchar(128) default '' COMMENT 'tenant_id',
171   `tenant_name` varchar(128) default '' COMMENT 'tenant_name',
172   `tenant_desc` varchar(256) DEFAULT NULL COMMENT 'tenant_desc',
173   `create_source` varchar(32) DEFAULT NULL COMMENT 'create_source',
174   `gmt_create` bigint(20) NOT NULL COMMENT '创建时间',
175   `gmt_modified` bigint(20) NOT NULL COMMENT '修改时间',
176   PRIMARY KEY (`id`),
177   UNIQUE KEY `uk_tenant_info_kptenantid` (`kp`, `tenant_id`),
178   KEY `idx_tenant_id` (`tenant_id`)
179 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin COMMENT='tenant_info';
180

```

```

181 CREATE TABLE `users` (
182     `username` varchar(50) NOT NULL PRIMARY KEY COMMENT 'username',
183     `password` varchar(500) NOT NULL COMMENT 'password',
184     `enabled` boolean NOT NULL COMMENT 'enabled'
185 );
186
187 CREATE TABLE `roles` (
188     `username` varchar(50) NOT NULL COMMENT 'username',
189     `role` varchar(50) NOT NULL COMMENT 'role',
190     UNIQUE INDEX `idx_user_role` (`username` ASC, `role` ASC) USING BTREE
191 );
192
193 CREATE TABLE `permissions` (
194     `role` varchar(50) NOT NULL COMMENT 'role',
195     `resource` varchar(128) NOT NULL COMMENT 'resource',
196     `action` varchar(8) NOT NULL COMMENT 'action',
197     UNIQUE INDEX `uk_role_permission` (`role`, `resource`, `action`) USING BTREE
198 );
199
200 INSERT INTO users (username, password, enabled) VALUES ('nacos',
201     '$2a$10$EuWPZHzz32dJN7jexM34M0eYirDdFAZm2kuWj7VE0JhhZkDrxfvUu', TRUE);
202
203 INSERT INTO roles (username, role) VALUES ('nacos', 'ROLE_ADMIN');

```

启动nacos容器

将以单例模式启动 开放

```

1 docker run -d -p 8848:8848 -p 9848:9848 --name oj-nacos -e MODE=standalone -e
  JVM_XMS=256m -e JVM_XMX=256m -e SPRING_DATASOURCE_PLATFORM=mysql -e
  MYSQL_SERVICE_HOST=${mysql_ip} -e MYSQL_SERVICE_PORT=${mysql_port} -e
  MYSQL_SERVICE_DB_NAME=${nacos_db_name} -e MYSQL_SERVICE_USER=${mysql_user} -e
  MYSQL_SERVICE_PASSWORD=${mysql_password} nacos/nacos-server:v2.2.3

```

项目中引入

引入依赖

```

1 <dependency>
2     <groupId>com.alibaba.cloud</groupId>
3     <artifactId>spring-cloud-starter-alibaba-nacos-config</artifactId>
4 </dependency>
5

```

```
6 <dependency>
7   <groupId>com.alibaba.cloud</groupId>
8   <artifactId>spring-cloud-starter-alibaba-nacos-discovery</artifactId>
9 </dependency>
```

修改bootstrap.yml

```
1 # Spring
2 spring:
3   application:
4     # 应用名称
5     name: oj-system
6   profiles:
7     # 环境配置
8     active: local
9   cloud:
10     nacos:
11       discovery:
12         namespace: biteoj
13         server-addr: http://localhost:8848
14       config:
15         namespace: biteoj
16         server-addr: http://localhost:8848
17         file-extension: yaml
```