

AC Serverless Architecture Design

What

无服务器模式（前期主要聚焦于groovy函数）：目的是通过把serverful开发模式逐渐转变为serverful+serverless并行的模式。提供sdk以代理模式与现有存量服务集成，在不改变被集成服务情况下，由执行框架统一调度动态脚本运行。在应用层面，由于groovy脚本的解释性，无需代码编译和部署就可以响应业务需求变化，显著加快研发效率和业务响应速度。

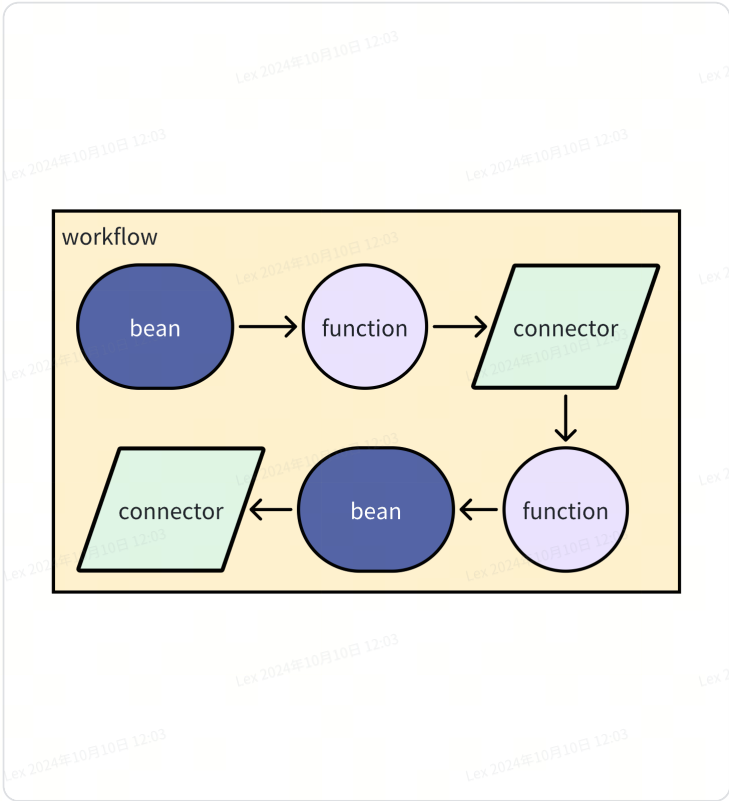
How

1. 组件抽象

- bean：bean是运行时动态创建出来的新的类，开发者在控制台写实现代码并发布。发布后的类将和宿主服务共享同一ClassLoader，因此不光可以实现新类之间互相调用而且可以在新类调用宿主服务已加载bean
- function：函数是一个通用的组件抽象，开发者可以直接在控制台写实现代码（其类名将自动重命名为【租户名+应用名+方法名】）。其中每个函数都有如下几个方法：
 - 匹配(match)：调用此函数的前提条件
 - 前置(onBefore)：当action成功执行后要执行的代码逻辑
 - 执行(run)：函数真正执行的代码逻辑，接受外部入参Map对象，也会返回出参Map对象。为了保证函数计算的单一职责、可观测和存算分离原则，函数内部仅内存操作，非必要不支持I/O操作，I/O相关由连接器(connector)承担
 - 后置(onAfter)：当action成功执行后要执行的代码逻辑
 - 错误(onException)：当action执行异常时要执行的代码逻辑
- hotfix：热更新类，开发者写更新逻辑后可以实时替换存量服务中的具体类的方法。每个热更新约定方法为：
 - hotfix：方法名不得修改，可以有入参列表，但必须和hotfix任务在控制台指定的入参保持一致。且方法返回类型必须和为原方法返回类型一致
- connector：连接器聚焦于连接不同的中间件进行读写，支持：
 - Openfeign connector：在同一注册中心下，注入被调用服务的feign接口bean，完成跨服务调用
 - Redis connector：封装redis组件完成相应环境下各种redis结构下的读与写操作，属于workflow流程编排的节点之一

- Mysql connector: 封装mysql组件完成相应环境下指定数据库与表的读与写操作, 属于 workflow 流程编排的节点之一
- RocketMq/kafka connector: 封装MQ组件完成相应环境下指定topic的消费与发布, 属于 workflow 流程编排的节点之一
- workflow: 工作流组件由一或多个函数、连接器编排而成的有向无环图(DAG)结构从而实现流水线有序作业, 必须包含至少一个函数

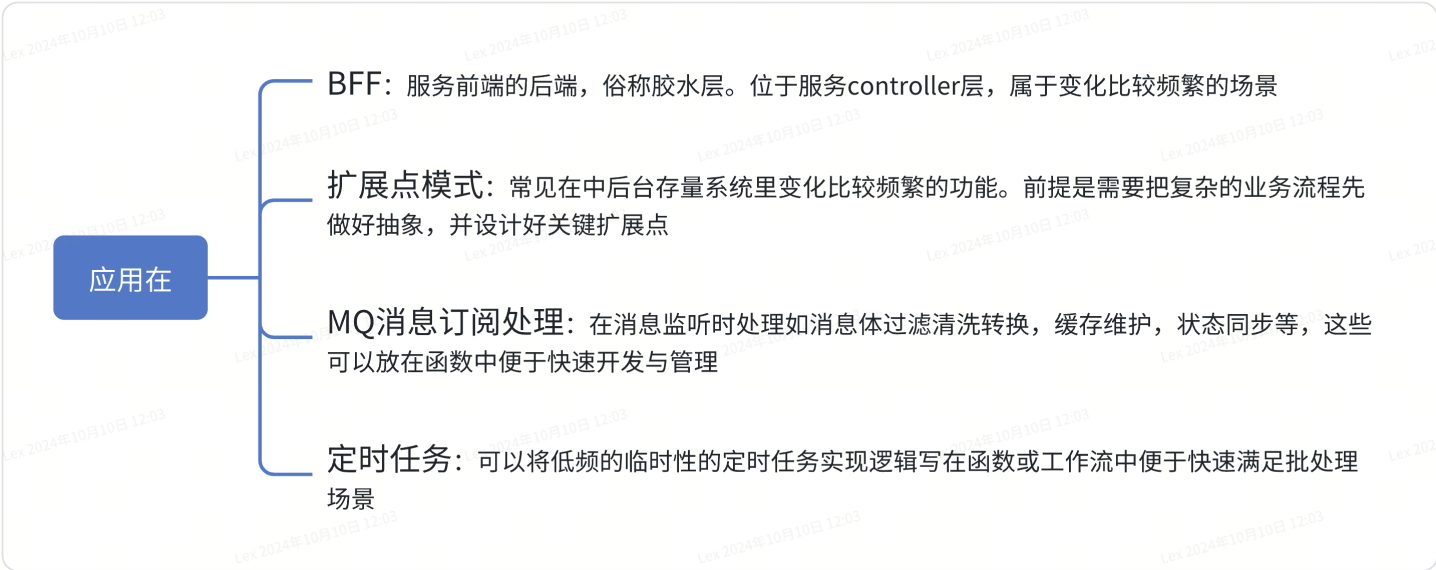
2. 组件关系(代码组织)



3. 触发条件

基于自定义注解, 可在服务任意方法中植入函数或工作流代码:

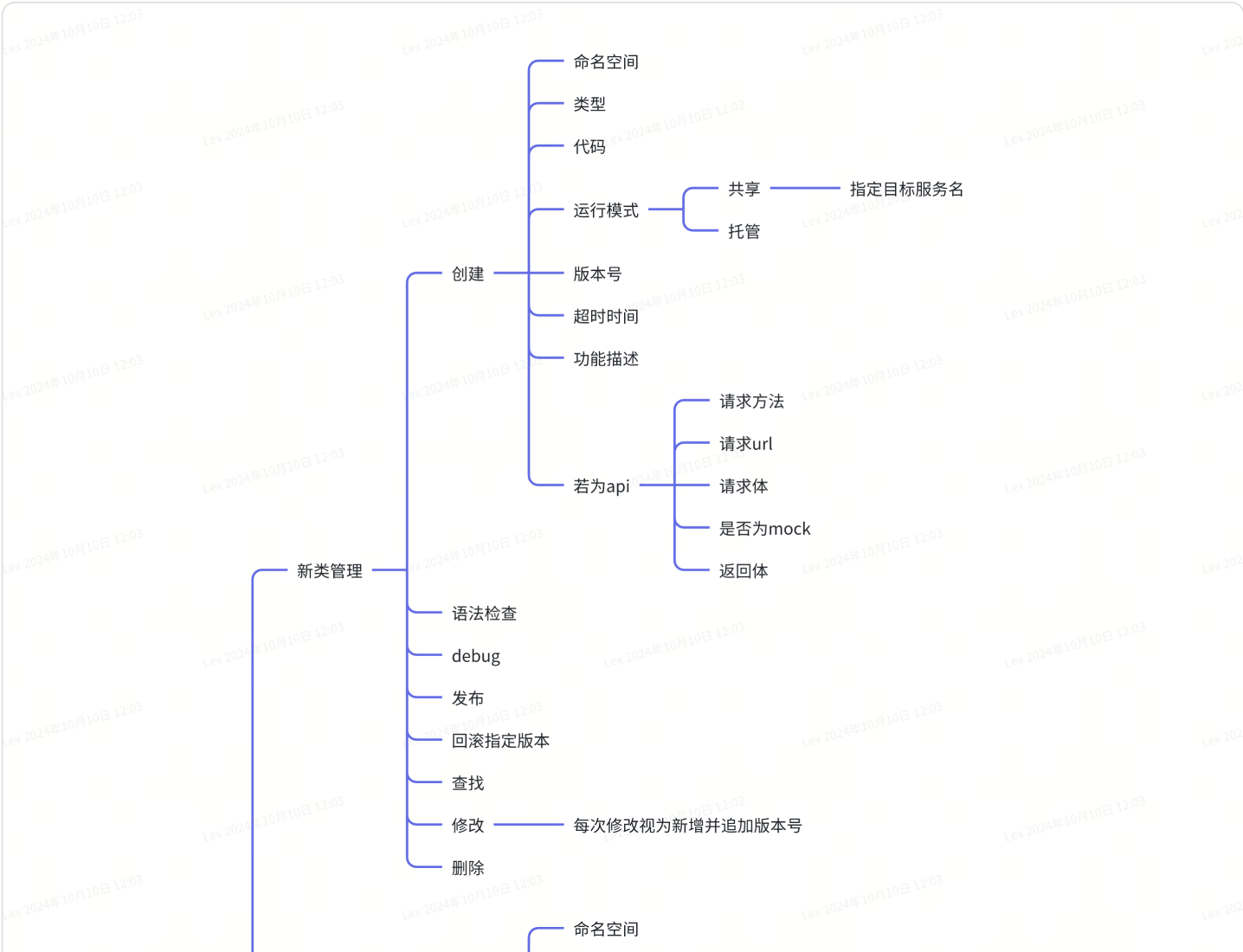
4. 应用场景



5. 资源管理

- 当前的scenario运行在被集成服务中，属于共享模式，即与调用方的服务共享服务器资源。风险是需要严格把关脚本的执行性能和稳定性，对cpu和内存占用过多的脚本应该拒绝发布
- 未来：基于k8s划分执行单元，可以做到不同scenario间的资源隔离、弹性扩缩容

6. 管控面设计



管理后台

函数管理

创建

- 唯一函数名
- 函数类型: groovy
- classpath
- 代码体
- 入参数集
- 出参数集
- 版本号
- 超时时间
- 功能描述

语法检查

debug

查找

修改 每次修改视为新增并追加版本号

删除

热修复管理

创建

- 命名空间
- 所属服务名
- fixed类路径
- fixed类方法
- 方法参数列表
- 代码体
- 功能描述

语法检查

查找

修改

删除

发布

回滚指定版本

运行日志

创建

- 命名空间
- 唯一连接器名
- 连接器类型: openfeign、redis、mysql、rocketmq、kafka
- classspath
- 代码体

连接器配置参数集

openfeign

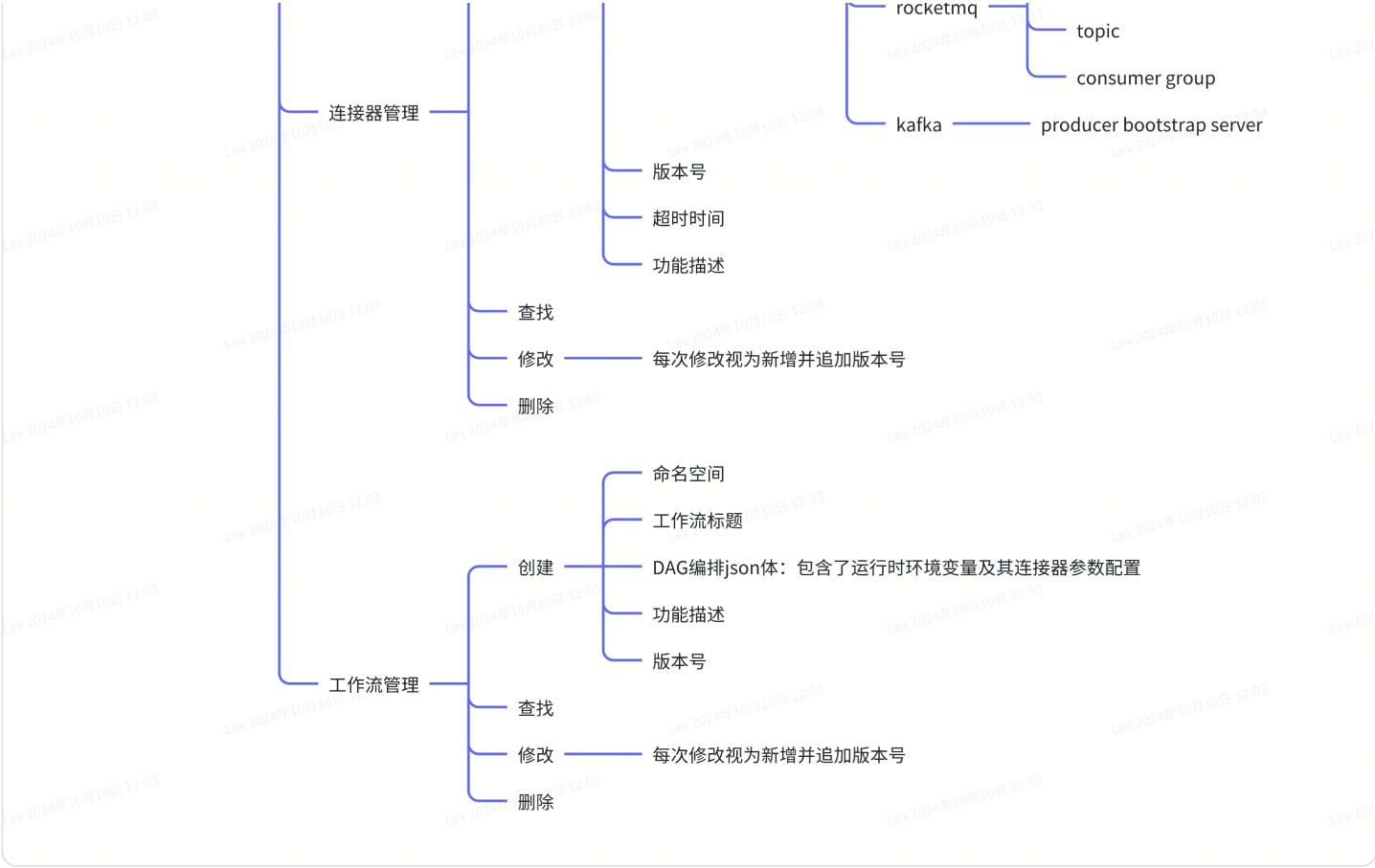
redis

- cluster nodes
- password

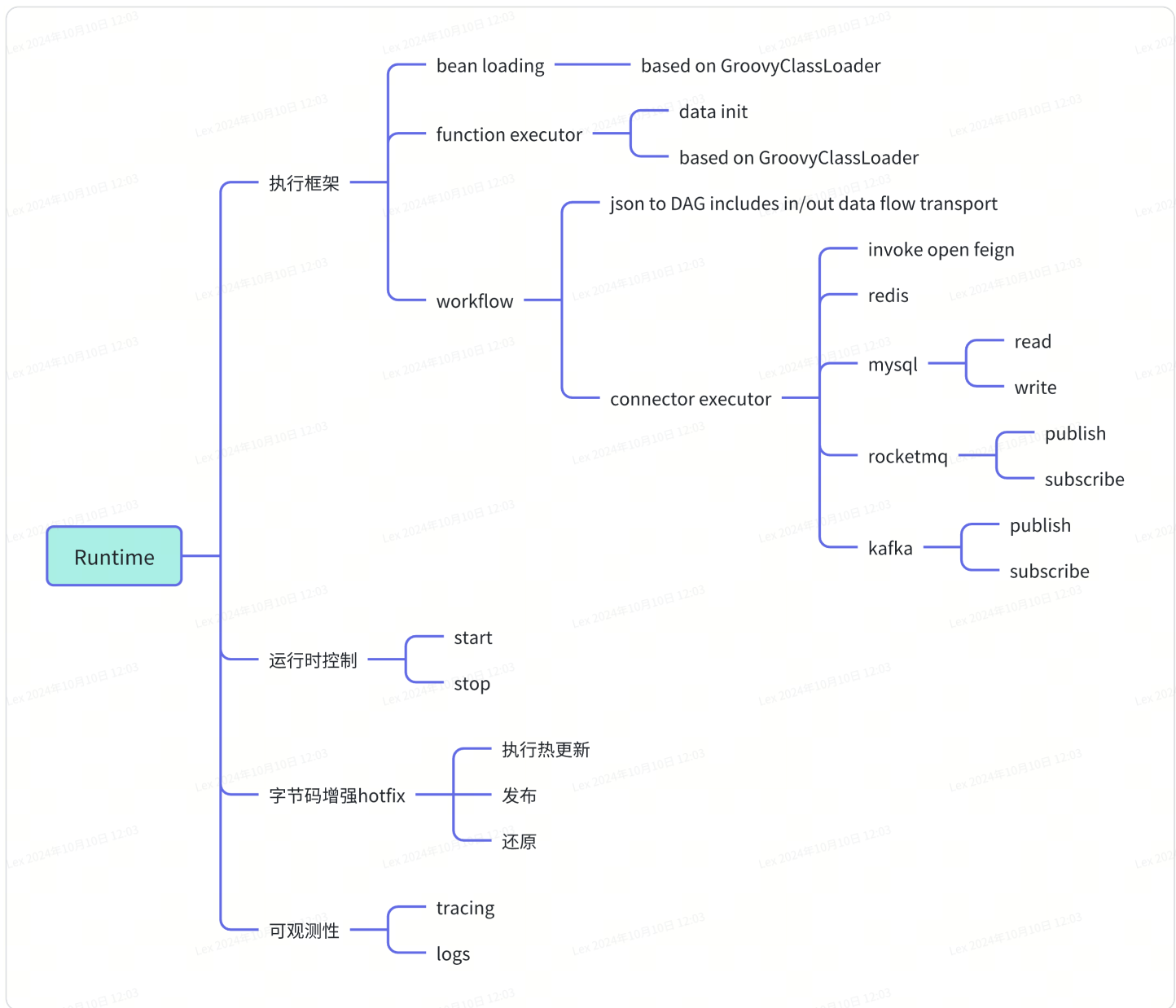
mysql

- jdbc connection
- username
- password
- driver classname

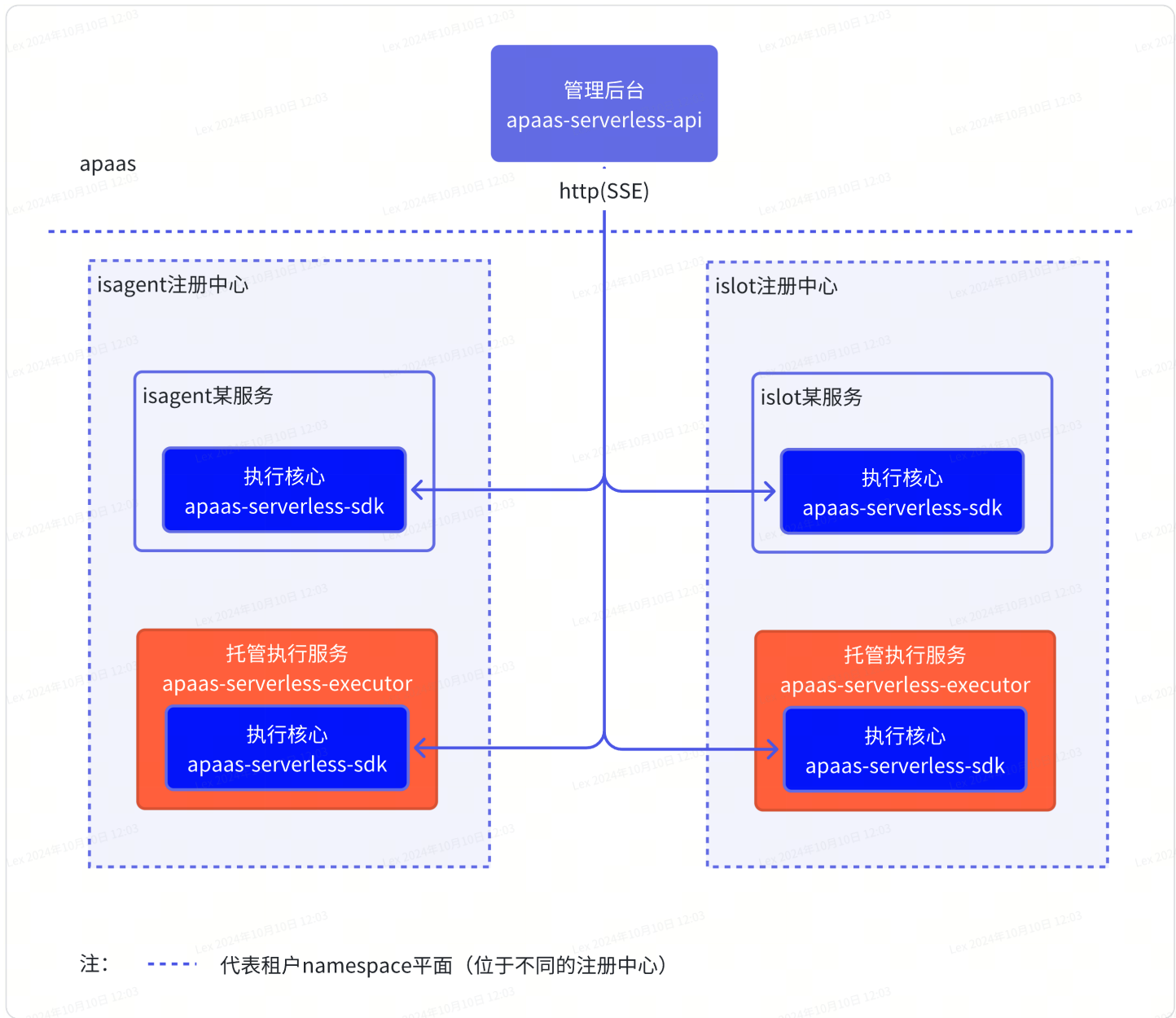
- binder nameserver
- binder group



7. 数据面设计



8. 系统架构



9. 表结构设计

UAT: islot-v4 ilum08

```
1 CREATE DATABASE `serverless` DEFAULT CHARACTER SET utf8mb4 COLLATE
  utf8mb4_0900_ai_ci;
2
3
4 CREATE TABLE `ac_serverless_function` (
5   `fid` bigint unsigned NOT NULL COMMENT '函数id',
6   `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
7   `appname` varchar(20) NOT NULL DEFAULT '' COMMENT '目标服务名与
  spring.application.name一致',
8   `function_name` varchar(20) NOT NULL DEFAULT '' COMMENT '唯一函数名',
9   `function_type` int unsigned NOT NULL DEFAULT 0 COMMENT '函数类型: 0-groovy',
10  `classpath` varchar(200) NOT NULL DEFAULT '' COMMENT '类路径',
```

```
11 `code_address` varchar(300) NOT NULL DEFAULT '' COMMENT '代码文件在对象存储的地址',
12 `in_var_names` varchar(200) NOT NULL DEFAULT '' COMMENT '请求参数名称集合逗号分割',
13 `out_var_names` varchar(200) NOT NULL DEFAULT '' COMMENT '返回参数名称集合逗号分割',
14 `version` int unsigned NOT NULL DEFAULT 0 COMMENT '版本号',
15 `applied` tinyint NOT NULL DEFAULT 0 COMMENT '是否应用默认0否 此时选择version最大的',
16 `timeout` int unsigned NOT NULL DEFAULT 5000 COMMENT '超时时间单位毫秒',
17 `description` varchar(500) NOT NULL DEFAULT '' COMMENT '功能描述',
18 `create_by` varchar(20) NOT NULL DEFAULT '' COMMENT '创建者',
19 `update_by` varchar(20) NOT NULL DEFAULT '' COMMENT '修改者',
20 `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建时间',
21 `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE CURRENT_TIMESTAMP(3) COMMENT '更新时间',
22 `deleted` tinyint NOT NULL DEFAULT 0 COMMENT '是否删除',
23 PRIMARY KEY (`fid`) USING BTREE,
24 KEY `idx_namespace` (`namespace`),
25 KEY `idx_appname` (`appname`)
26 ) ENGINE=InnoDB comment='函数组件表';
27
28
29 CREATE TABLE `ac_serverless_connector` (
30 `cid` bigint unsigned NOT NULL COMMENT '连接器id',
31 `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
32 `connector_name` varchar(20) NOT NULL DEFAULT '' COMMENT '唯一连接器名',
33 `connector_type` int unsigned NOT NULL DEFAULT 0 COMMENT '连接器类型: 0-openfeign 1-redis 2-mysql 3-rocketmq 4-kafka',
34 `classpath` varchar(200) NOT NULL DEFAULT '' COMMENT '类路径',
35 `code_address` varchar(300) NOT NULL DEFAULT '' COMMENT '代码文件在对象存储的地址',
36 `in_params` varchar(500) NOT NULL DEFAULT '' COMMENT '输入参数json',
37 `out_params` varchar(500) NOT NULL DEFAULT '' COMMENT '输出参数json',
38 `version` int unsigned NOT NULL DEFAULT 0 COMMENT '版本号',
39 `applied` tinyint NOT NULL DEFAULT 0 COMMENT '是否应用默认否',
40 `timeout` int unsigned NOT NULL DEFAULT 10000 COMMENT '超时时间单位毫秒',
41 `description` varchar(500) NOT NULL DEFAULT '' COMMENT '功能描述',
42 `create_by` varchar(20) NOT NULL DEFAULT '' COMMENT '创建者',
43 `update_by` varchar(20) NOT NULL DEFAULT '' COMMENT '修改者',
44 `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建时间',
45 `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE CURRENT_TIMESTAMP(3) COMMENT '更新时间',
46 `deleted` tinyint NOT NULL DEFAULT 0 COMMENT '是否删除',
47 PRIMARY KEY (`cid`) USING BTREE,
```



```
48 KEY `idx_namespace` (`namespace`),
49 UNIQUE KEY `uk_connector_name` (`connector_name`,`version`,`deleted`)
50 ) ENGINE=InnoDB comment='连接器组件表';
51
52
53 CREATE TABLE `ac_serverless_workflow` (
54 `wid` bigint unsigned NOT NULL COMMENT '工作流id',
55 `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
56 `workflow_name` varchar(20) NOT NULL DEFAULT '' COMMENT '唯一工作流名称',
57 `title` varchar(30) NOT NULL DEFAULT '' COMMENT '工作流标题',
58 `orchestration_address` varchar(300) NOT NULL DEFAULT '' COMMENT '编排json文件在
对象存储的地址',
59 `version` int unsigned NOT NULL DEFAULT 0 COMMENT '版本号',
60 `applied` tinyint NOT NULL DEFAULT 0 COMMENT '是否应用默认否',
61 `description` varchar(500) NOT NULL DEFAULT '' COMMENT '功能描述',
62 `create_by` varchar(20) NOT NULL DEFAULT '' COMMENT '创建者',
63 `update_by` varchar(20) NOT NULL DEFAULT '' COMMENT '修改者',
64 `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建时
间',
65 `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE
CURRENT_TIMESTAMP(3) COMMENT '更新时间',
66 `deleted` tinyint NOT NULL DEFAULT 0 COMMENT '是否删除',
67 PRIMARY KEY (`wid`) USING BTREE,
68 KEY `idx_namespace` (`namespace`),
69 UNIQUE KEY `uk_workflow` (`workflow_name`,`version`,`deleted`)
70 ) ENGINE=InnoDB comment='工作流组件表';
71
72
73 CREATE TABLE `ac_serverless_hotfix` (
74 `hid` bigint unsigned NOT NULL COMMENT 'hotfix id',
75 `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
76 `appname` varchar(20) NOT NULL DEFAULT '' COMMENT '目标服务名与
spring.application.name一致',
77 `classpath` varchar(100) NOT NULL DEFAULT '' COMMENT '要修复的类路径',
78 `method_name` varchar(30) NOT NULL DEFAULT '' COMMENT '要修复的类方法',
79 `arg_types` varchar(200) NOT NULL DEFAULT '' COMMENT '修复代码的方法参数类型集合',
80 `arg_names` varchar(200) NOT NULL DEFAULT '' COMMENT '修复代码的参数名称集合逗号分
割',
81 `arg_values` varchar(800) NOT NULL DEFAULT '' COMMENT '修复代码的参数值集合逗号分
割',
82 `return_type` varchar(100) NOT NULL DEFAULT '' COMMENT '返回类型',
83 `code_address` varchar(300) NOT NULL DEFAULT '' COMMENT '修复的代码文件在对象存储的
地址',
84 `description` varchar(500) NOT NULL DEFAULT '' COMMENT '功能描述',
85 `version` int unsigned NOT NULL DEFAULT 0 COMMENT '版本号',
86 `applied` tinyint NOT NULL DEFAULT 0 COMMENT '是否应用默认否 此时选择version最大
的',
```

```
87 `create_by` varchar(20) NOT NULL DEFAULT '' COMMENT '创建者',
88 `update_by` varchar(20) NOT NULL DEFAULT '' COMMENT '修改者',
89 `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建时
间',
90 `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE
CURRENT_TIMESTAMP(3) COMMENT '更新时间',
91 `deleted` tinyint NOT NULL DEFAULT 0 COMMENT '是否删除',
92 `uk` varchar(32) NOT NULL DEFAULT '' COMMENT 'ns:app:class:method:return串md5代
表特定类方法',
93 PRIMARY KEY (`hid`) USING BTREE,
94 KEY `idx_namespace` (`namespace`),
95 KEY `idx_appname` (`appname`),
96 KEY `uk` (`uk`)
97 ) ENGINE=InnoDB comment='热修复表';
98
99
100 CREATE TABLE `ac_serverless_envar` (
101 `eid` bigint unsigned NOT NULL COMMENT '环境变量id',
102 `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
103 `appname` varchar(20) NOT NULL DEFAULT '' COMMENT '目标服务名与
spring.application.name一致',
104 `var_name` varchar(30) NOT NULL DEFAULT '' COMMENT '环境变量名',
105 `var_classpath` varchar(60) NOT NULL DEFAULT '' COMMENT '环境变量类型',
106 `var_value` varchar(500) NOT NULL DEFAULT '' COMMENT '环境变量值',
107 `description` varchar(500) NOT NULL DEFAULT '' COMMENT '变量描述',
108 `create_by` varchar(20) NOT NULL DEFAULT '' COMMENT '创建者',
109 `update_by` varchar(20) NOT NULL DEFAULT '' COMMENT '修改者',
110 `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建时
间',
111 `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE
CURRENT_TIMESTAMP(3) COMMENT '更新时间',
112 `deleted` tinyint NOT NULL DEFAULT 0 COMMENT '是否删除',
113 PRIMARY KEY (`eid`) USING BTREE,
114 KEY `idx_namespace` (`namespace`),
115 KEY `idx_appname` (`appname`),
116 UNIQUE KEY `uk_var` (`namespace`, `appname`, `var_name`, `deleted`)
117 ) ENGINE=InnoDB comment='环境变量表';
118
119
120 CREATE TABLE `open_api_client` (
121 `client_id` bigint unsigned NOT NULL COMMENT '客户端id',
122 `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
123 `appname` varchar(30) NOT NULL DEFAULT '' COMMENT 'app名称',
124 `encrypt_algorithm` varchar(20) NOT NULL DEFAULT 'rsa' COMMENT '加密算法',
125 `expired_date` datetime COMMENT '过期时间',
126 `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建时
间',
```

```

127 `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE
    CURRENT_TIMESTAMP(3) COMMENT '更新时间',
128 `disabled` tinyint NOT NULL DEFAULT '0' COMMENT '是否禁用0为否',
129 PRIMARY KEY (`client_id`),
130 UNIQUE KEY `uk_namespace_app` (`namespace`, `appname`, `disabled`)
131 ) ENGINE=InnoDB comment='OPENAPI接入客户端信息表';
132
133
134 CREATE TABLE `ac_serverless_restriction` (
135 `id` bigint unsigned NOT NULL AUTO_INCREMENT COMMENT '主键id',
136 `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
137 `appname` varchar(20) NOT NULL DEFAULT '' COMMENT '目标服务名与
    spring.application.name一致',
138 `class_name` varchar(100) NOT NULL DEFAULT '' COMMENT '类名称',
139 `method_name` varchar(30) NOT NULL DEFAULT '' COMMENT '方法名',
140 `restricted` tinyint NOT NULL DEFAULT 0 COMMENT '0黑名单 1白名单',
141 `description` varchar(500) NOT NULL DEFAULT '' COMMENT '描述',
142 `create_by` varchar(20) NOT NULL DEFAULT '' COMMENT '创建者',
143 `update_by` varchar(20) NOT NULL DEFAULT '' COMMENT '修改者',
144 `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建时
    间',
145 `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE
    CURRENT_TIMESTAMP(3) COMMENT '更新时间',
146 `deleted` tinyint NOT NULL DEFAULT 0 COMMENT '是否删除',
147 PRIMARY KEY (`id`) USING BTREE,
148 KEY `idx_namespace` (`namespace`),
149 KEY `idx_appname` (`appname`)
150 ) ENGINE=InnoDB comment='黑白名单限制表';

```

10. 变更记录:

10.1 Changed Timeline: 20240916

```

1 --变更sql
2 ALTER TABLE sys_user ADD CONSTRAINT uk_namespace_username UNIQUE KEY
    (namespace,username,deleted);
3
4 ALTER TABLE sys_role MODIFY COLUMN role_lock char(1) DEFAULT '0' NOT NULL
    COMMENT '锁定标识 0-正常。1-锁定';
5 ALTER TABLE sys_role DROP COLUMN namespace;
6 ALTER TABLE sys_role ADD CONSTRAINT uk_role UNIQUE KEY (role_code,deleted);
7
8 ALTER TABLE sys_menu MODIFY COLUMN keep_alive char(1) DEFAULT '1' NOT NULL
    COMMENT '1-开启 0-关闭';
9 ALTER TABLE sys_menu DROP COLUMN namespace;

```

```
10 ALTER TABLE sys_menu ADD CONSTRAINT uk_permission UNIQUE KEY
   (permission,deleted);
11 ALTER TABLE sys_menu MODIFY COLUMN menu_type char(1) DEFAULT '0' NOT NULL
   COMMENT '菜单类型 0 目录 1菜单 2按钮';
12 ALTER TABLE sys_menu ADD component varchar(100) DEFAULT '' NOT NULL COMMENT '类
   型为菜单时指定前端组件路径';
13 ALTER TABLE sys_menu CHANGE component component varchar(100) DEFAULT '' NOT
   NULL COMMENT '类型为菜单时指定前端组件路径' AFTER menu_type;
14 ALTER TABLE sys_menu ADD component_name varchar(30) DEFAULT '' NOT NULL
   COMMENT '前端组件名不能重复';
15 ALTER TABLE sys_menu CHANGE component_name component_name varchar(30) DEFAULT
   '' NOT NULL COMMENT '前端组件名不能重复' AFTER component;
16
17 ALTER TABLE sys_tenant DROP KEY uk_namespace;
18 ALTER TABLE sys_tenant ADD CONSTRAINT uk_namespace UNIQUE KEY
   (namespace,deleted);
19
20 ALTER TABLE sys_user_role ADD CONSTRAINT uk_user_role UNIQUE KEY
   (user_id,role_id);
21
22 ALTER TABLE sys_role_menu ADD CONSTRAINT uk_role_menu UNIQUE KEY
   (role_id,menu_id);
23
24
25 ALTER TABLE ac_serverless_function ADD appname varchar(20) DEFAULT '' NOT NULL
   COMMENT '目标服务名与spring.application.name一致';
26 ALTER TABLE ac_serverless_function CHANGE appname appname varchar(20) DEFAULT
   '' NOT NULL COMMENT '目标服务名与spring.application.name一致' AFTER namespace;
27 ALTER TABLE ac_serverless_function DROP KEY uk_function_name;
28 CREATE INDEX idx_appname USING BTREE ON ac_serverless_function (appname);
29
30
31 ALTER TABLE ac_serverless_hotfix ADD version int unsigned NOT NULL DEFAULT 0
   COMMENT '版本号';
32 ALTER TABLE ac_serverless_hotfix CHANGE version version int unsigned DEFAULT 0
   NOT NULL COMMENT '版本号' AFTER description;
33 ALTER TABLE ac_serverless_hotfix ADD applied tinyint NOT NULL DEFAULT 0
   COMMENT '是否应用默认0否 此时选择version最大的';
34 ALTER TABLE ac_serverless_hotfix CHANGE applied applied tinyint NOT NULL
   DEFAULT 0 COMMENT '是否应用默认0否 此时选择version最大的' AFTER version;
35 ALTER TABLE ac_serverless_hotfix ADD uk varchar(32) NOT NULL DEFAULT ''
   COMMENT 'ns:app:class:method:return串md5代表特定类方法';
36 CREATE INDEX idx_uk USING BTREE ON ac_serverless_hotfix (uk);
37 ALTER TABLE ac_serverless_hotfix MODIFY COLUMN arg_values varchar(800) DEFAULT
   '' NOT NULL COMMENT '修复代码的参数值字符串数组';
```

```

1 # apaas-serverless-api.properties变更配置
2 # 修改值
3 secure.ignored-paths[1]=/web/v1/auth/register
4 # 修改key名 原key为openapi.authentication.enabled,更新为如下
5 openapi.manager.authentication.enabled
6 # 修改key名 原key为openapi.authentication.interception-path,更新为如下
7 openapi.manager.authentication.interception-path
8 # 修改key名 原key为openapi.authentication.ignored-paths[0],更新为如下
9 openapi.manager.authentication.ignored-paths[0]
10 # 修改key名 原key为openapi.authentication.algorithm,更新为如下
11 openapi.manager.authentication.algorithm
12 # 修改key名 原key为openapi.authentication.public-key,更新为如下
13 openapi.manager.authentication.public-key
14 # 修改key名 原key为openapi.authentication.private-key,更新为如下
15 openapi.manager.authentication.private-key
16
17 # 修改uat的值
18 secure.jwt-secret=hp1Urb8jLM5JtcsdAhGL5eJjmC24x724
19 # 修改prod的值
20 secure.jwt-secret=DRddFu7EB9BMarQQEWFFdgTbtQT3L5Gv

```

10.2 Changed Timeline: 20240923

```

1 CREATE TABLE `ac_serverless_bean` (
2   `bid` bigint unsigned NOT NULL COMMENT 'bean id',
3   `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
4   `appname` varchar(20) NOT NULL DEFAULT '' COMMENT '所在服务',
5   `bean_name` varchar(20) NOT NULL DEFAULT '' COMMENT '唯一类名',
6   `run_mode` tinyint NOT NULL DEFAULT 0 COMMENT '运行模式:0共享 1托管',
7   `category_id` int unsigned NOT NULL DEFAULT 0 COMMENT '类型id, 0:WebApi
8     1:Component 2:POGO 3:InnerApi 4:Helper 5:Configuration 6:Interface 7:Enum
9     8:Annotation',
10  `code_address` varchar(300) NOT NULL DEFAULT '' COMMENT '代码文件在对象存储的地址',
11  `version` int unsigned NOT NULL DEFAULT 0 COMMENT '版本号',
12  `applied` tinyint NOT NULL DEFAULT 0 COMMENT '是否应用默认0否 此时选择version最大的',
13  `timeout` int unsigned NOT NULL DEFAULT 5000 COMMENT '超时时间单位毫秒',
14  `description` varchar(500) NOT NULL DEFAULT '' COMMENT '功能描述',
15  `request_method` varchar(8) NOT NULL DEFAULT '' COMMENT 'api请求方法:get,post,put,delete',
16  `api_url` varchar(50) NOT NULL DEFAULT '' COMMENT 'api路由地址',
17  `request_body` varchar(500) NOT NULL DEFAULT '' COMMENT 'api请求体',
18  `api_mocked` tinyint NOT NULL DEFAULT 0 COMMENT 'api是否mock',

```

```

17 `response_body` varchar(500) NOT NULL DEFAULT '' COMMENT 'api返回体 mocked=1时为
    mock返回',
18 `create_by` varchar(20) NOT NULL DEFAULT '' COMMENT '创建者',
19 `update_by` varchar(20) NOT NULL DEFAULT '' COMMENT '修改者',
20 `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建时
    间',
21 `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE
    CURRENT_TIMESTAMP(3) COMMENT '更新时间',
22 `deleted` tinyint NOT NULL DEFAULT 0 COMMENT '是否删除',
23 PRIMARY KEY (`bid`) USING BTREE,
24 KEY `idx_namespace` (`namespace`),
25 KEY `idx_appname` (`appname`),
26 KEY `idx_bean_name` (`bean_name`),
27 KEY `idx_category_id` (`category_id`),
28 KEY `idx_run_mode` (`run_mode`),
29 UNIQUE KEY `uk_namespace_app_bean` (`namespace`, `appname`, `bean_name`,
    `version`, `deleted`)
30 -- UNIQUE KEY `uk_request_api` (`namespace`, `request_method`, `api_url`,
    `version`)
31 ) ENGINE=InnoDB comment='新类组件表';
32
33
34 CREATE TABLE `sys_tenant_application` (
35   `id` bigint unsigned NOT NULL AUTO_INCREMENT COMMENT '主键id',
36   `namespace` varchar(20) NOT NULL DEFAULT '' COMMENT '命名空间',
37   `appname` varchar(20) NOT NULL DEFAULT '' COMMENT '应用名',
38   `appname` varchar(20) NOT NULL DEFAULT '' COMMENT '包路径',
39   `create_by` varchar(20) NOT NULL DEFAULT '' COMMENT '创建者',
40   `update_by` varchar(20) NOT NULL DEFAULT '' COMMENT '修改者',
41   `create_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) COMMENT '创建
    时间',
42   `update_time` datetime(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3) ON UPDATE
    CURRENT_TIMESTAMP(3) COMMENT '更新时间',
43   `deleted` tinyint NOT NULL DEFAULT 0 COMMENT '是否删除',
44   PRIMARY KEY (`id`),
45   KEY `idx_namespace` (`namespace`),
46   UNIQUE KEY `uk_tenant_app` (`namespace`, `appname`, `deleted`)
47 ) AUTO_INCREMENT = 1 COMMENT='租户应用表';

```

10.3 Changed Timeline: 20241008

- 1 ALTER TABLE ac_serverless_bean MODIFY COLUMN appname varchar(30) DEFAULT '' NOT NULL COMMENT '所在服务';
- 2 ALTER TABLE ac_serverless_envar MODIFY COLUMN appname varchar(30) DEFAULT '' NOT NULL COMMENT '目标服务名与spring.application.name一致';


```
3 ALTER TABLE ac_serverless_function MODIFY COLUMN appname varchar(30) DEFAULT ''
  NOT NULL COMMENT '目标服务名与spring.application.name一致';
4 ALTER TABLE ac_serverless_hotfix MODIFY COLUMN appname varchar(30) DEFAULT ''
  NOT NULL COMMENT '目标服务名与spring.application.name一致';
5 ALTER TABLE ac_serverless_restriction MODIFY COLUMN appname varchar(30) DEFAULT
  '' NOT NULL COMMENT '目标服务名与spring.application.name一致';
6 ALTER TABLE sys_tenant_application MODIFY COLUMN appname varchar(30) DEFAULT ''
  NOT NULL COMMENT '应用名';
```