

好业务架构及实践

jqzhang

s_jqzhang@163.com

目录

- ▶ 好业务架构的基本原则有那些
- ▶ 运维通道在此架构下的实践之路

好业务架构的基本原则

稳定性

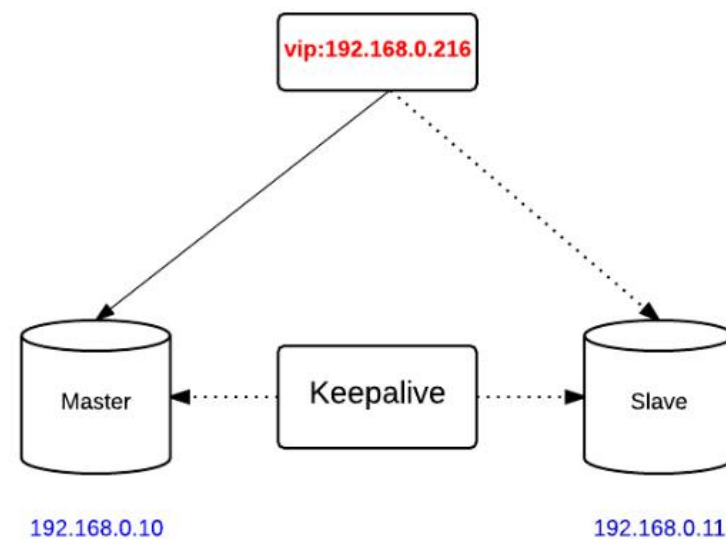
反例

- 裸奔（没什么保护措施）

正例

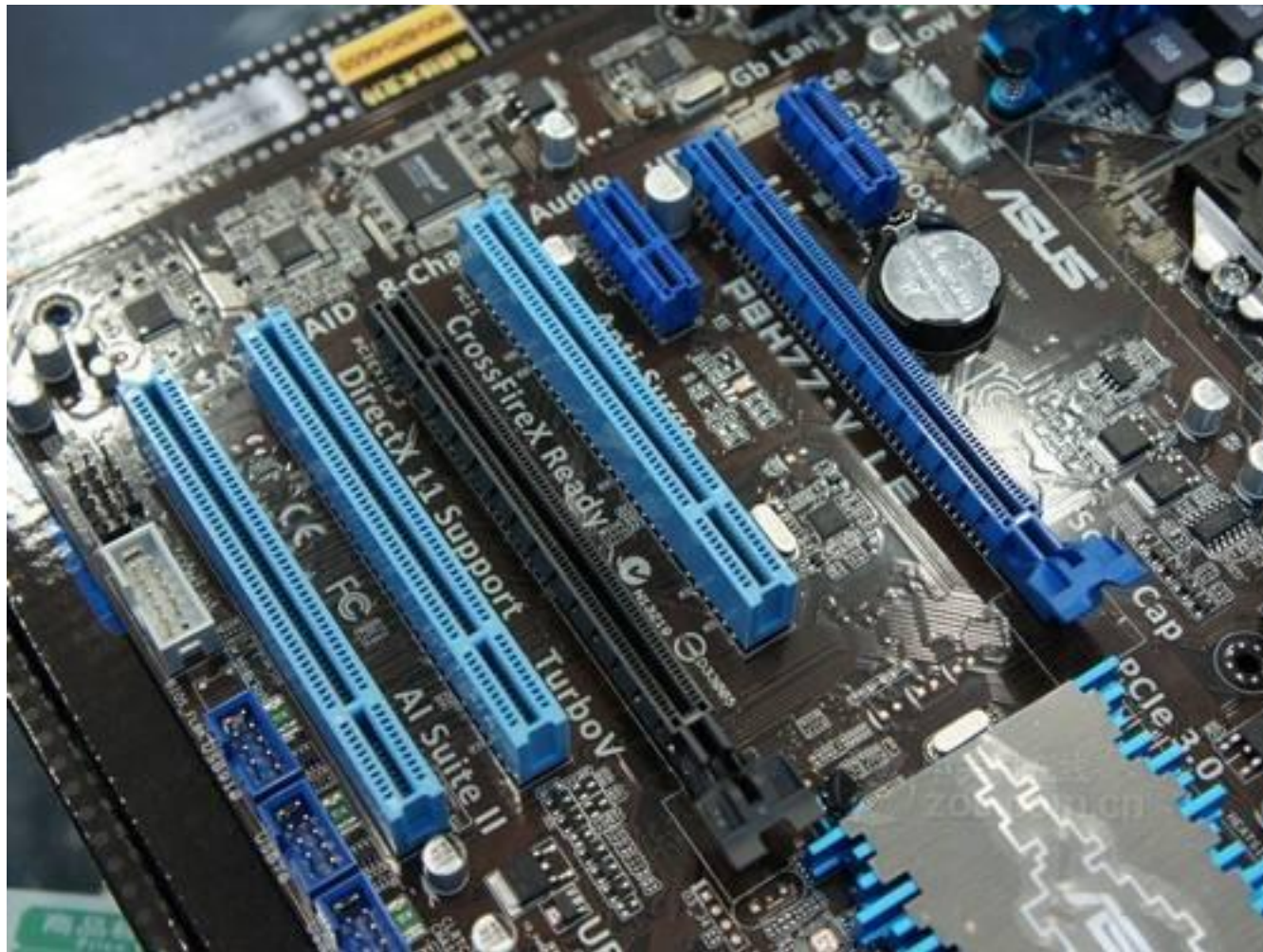
- 自动修复
- 监控先行
- 过载保护
- 高可用部署

OA选型：
OA系统稳定性压倒一切



扩展性

- ▶ 反例
 - ▶ 无分层
 - ▶ 无分治
- ▶ 正例
 - ▶ 业务可扩展
 - ▶ 容量可扩展



可配置性

- ▶ 反例
 - ▶ 硬编码（配置没抽离）
 - ▶ 重启更新配置
- ▶ 正例
 - ▶ 静态配置性
 - ▶ 动态配置性

MKX尊享版/MKC尊耀版 外观配置对比		
车型	MKX尊享版	MKC尊耀版
厂商指导价(万元)	449,800	438,800
LED氙气前大灯	●	●
前大灯自动开启/关闭功能	●	●
前大灯自动水平调节功能	●	●
前大灯自动清洗功能	●	●
自动远光灯控制	-	●
LED日间行车灯	●	●
整车转向信号灯	-	●
外置后视镜带防眩目功能(驾驶员侧)	●	●
外置后视镜加热和记忆功能	●	●
后门照地灯	●	●
后雾灯	●	●
带电动遮阳帘的电动开启式全景天窗	●	●
自动雨量感应式前雨刷 /2速后雨刷	-	●
注：“●”表示配备，“-”表示不配备		
制表：网通社(Internet Info Agency)		

```

{
  "debug": true,
  "benchmark": false,
  "use_zk_db": true,
  "is_gateway": true,
  "use_zk_redis": true,
  "result2db": true,
  "group": "cn",
  "builtin_redis": false,
  "static_dir": "static",
  "falcon_url": "http://127.0.0.1:1988/v1/push",
  "super_admin": [
    "jqzhang"
  ],
}
```

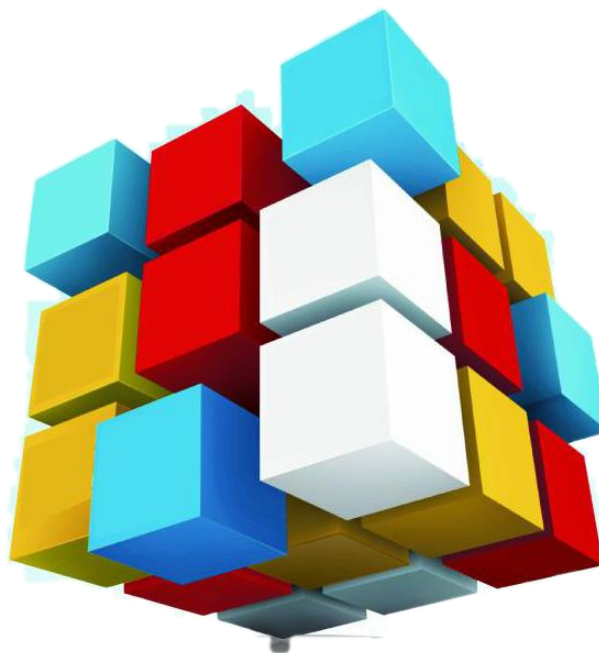
简单性

► 反例

- 开发复杂(设计复杂，定位问题困难)
- 运维复杂

► 正例

- 开发简单（增加功能）
- 运维简单（部署，配置，维护）



可测试性

- ▶ 反例
 - ▶ 一切在线上测试
- ▶ 正例
 - ▶ 性能测试
 - ▶ 功能回归



可维护性

► 反例

- 停服定位问题
- 重启更新配置

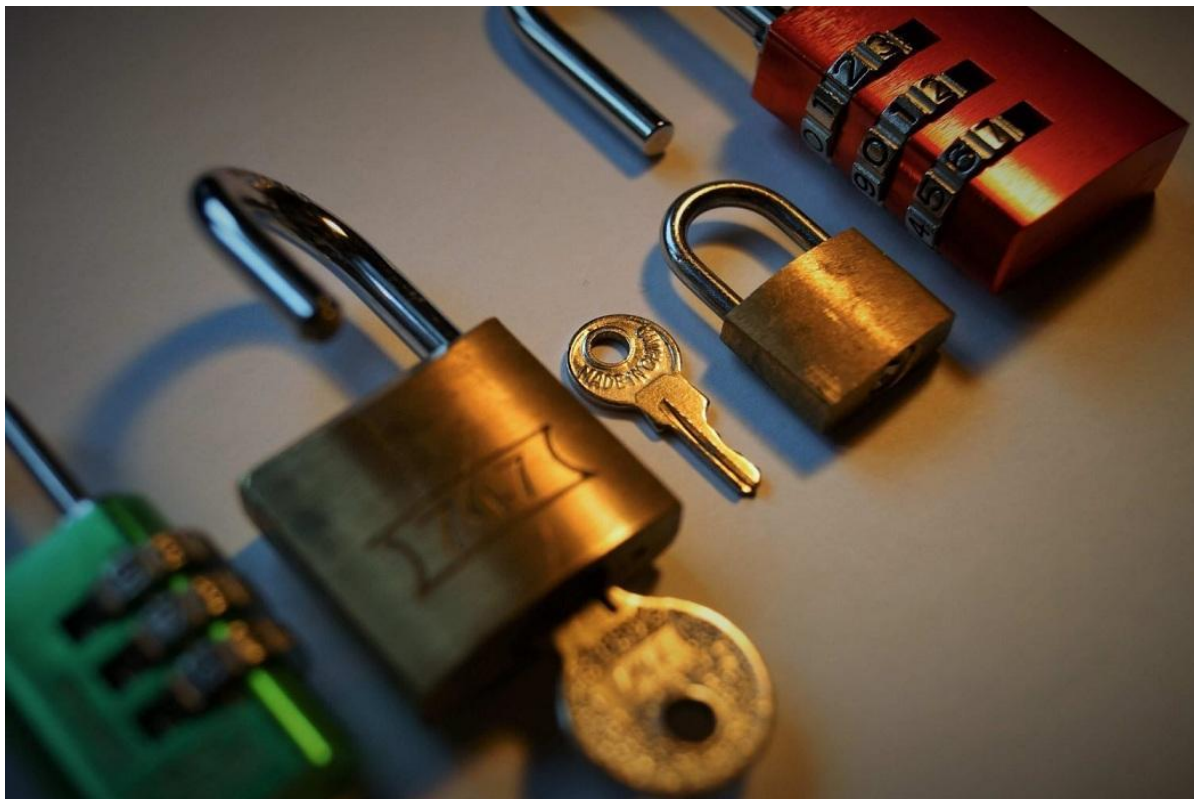
► 正例

- 快速部署
- 问题的定位是否快速方便
- 迁级是否平滑



安全性

- ▶ 反例（无安全意识）
 - ▶ 无环境隔离
 - ▶ 权限最大化（用户权限）
 - ▶ 运行用户**ROOT**
 - ▶ **SQL**注入
 - ▶ 跨站攻击
 - ▶ 敏感数据明文储存
 - ▶
- ▶ 正例（以上反着读）
 - ▶ 环境隔离
 - ▶ 权限最最小化（用户权限）
 - ▶ 非**ROOT**用户运行程序
 - ▶ **SQL**参数化
 - ▶ 防跨站攻击（过滤）
 - ▶ 敏感数据加密储存



闭环

- ▶ 运营数据可视化
- ▶ 监控数据可视化
- ▶ 数据驱动优化业务

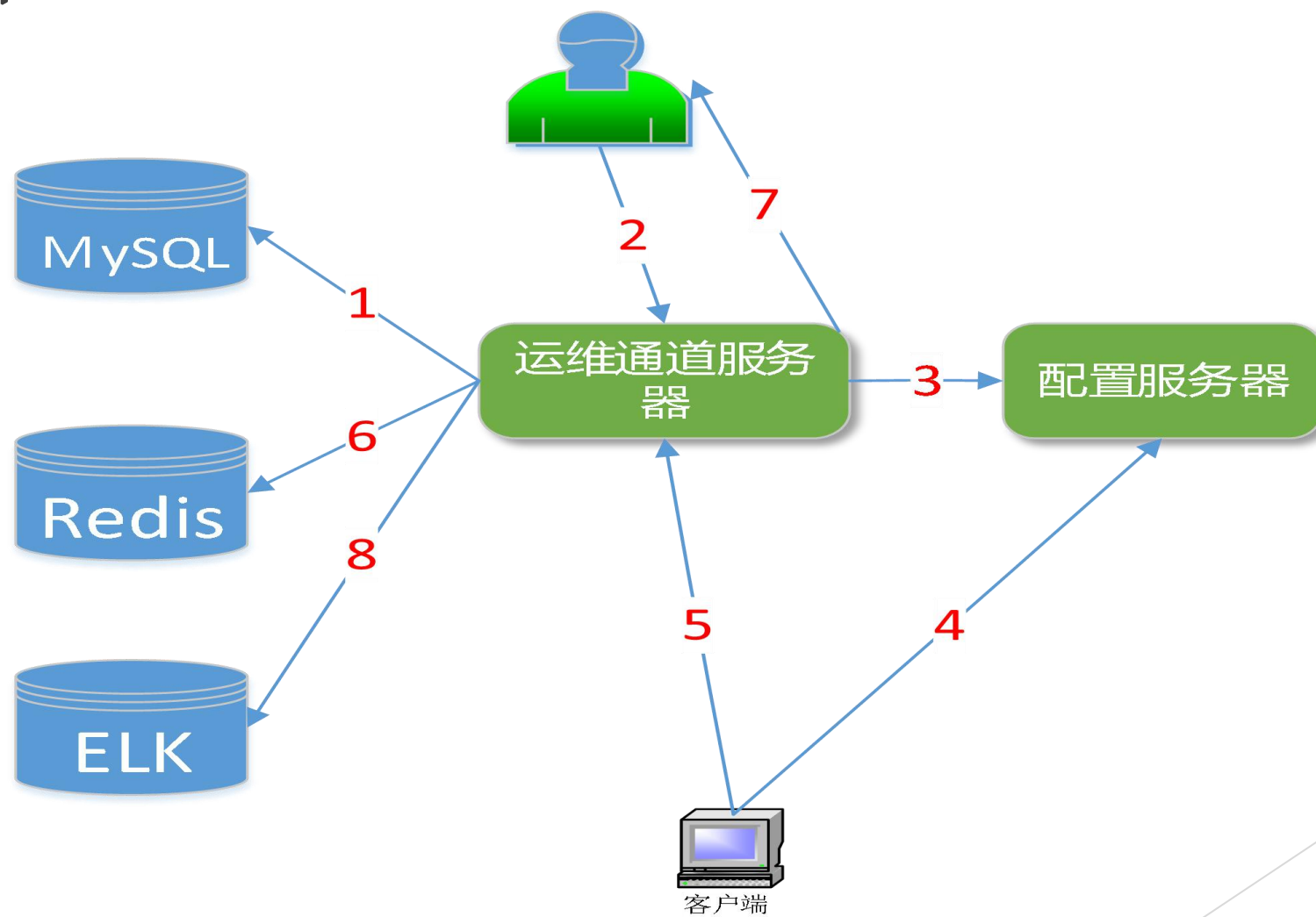
运维通道实践之路

运维通道是什么？

- ▶ 运维通是一座桥
- ▶ 为什么要自建运维通道？（效率，安全，可靠）



架构图




稳定性

- ▶ 自动修复 (**cron,timer**)
- ▶ 监控先行
- ▶ 过载保护
- ▶ 高可用部署(**keepalive**)

```
[root@10.1.50.90 ~]# time cli api -u root --sudo 1 -i [redacted] -c hostname
-----
10.1.50.90
OPER-dev-50-90
fails:

real    0m0.441s
user    0m0.094s
sys     0m0.029s
[root@10.1.50.90 ~]# time cli api -u root --sudo 1 -i [redacted] -c hostname
{
  "message": "(error) server overload"
}

real    0m0.130s
user    0m0.097s
sys     0m0.021s
```

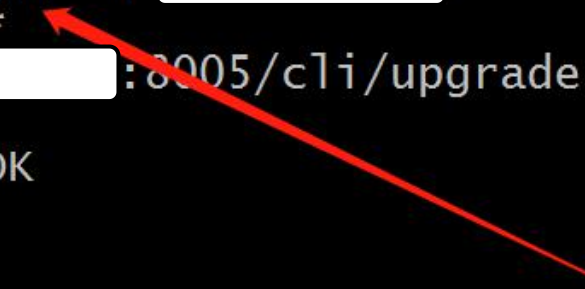


```
[root@OPER10.1.55.69 channel]# cli repair -i [redacted]
***** (success) 10.1.55.69 *****
--2019-05-02 09:48:36-- http://[redacted]:8005/cli/upgrade
正在连接 [redacted]:8005... 已连接。
已发出 HTTP 请求, 正在等待回应... 200 OK
长度: 未指定 [text/plain]
正在保存至: "/bin/cli"

      26,550      --.-K/s   in 0s

2019-05-02 09:48:36 (66.5 MB/s) - "/bin/cli" 已保存 [26550]

cli daemon restart
```



扩展性

- ▶ 功能增加通过插件（脚本）
- ▶ 增加机器增加处理能力

```
[root@OPER[REDACTED] backends]# cli rshell -f vm -d jqzhang -u root --sudo 1
-----
[REDACTED]
-a(arguments) require, example: -a 'name=test mem=8G cpu=4 disk=50G bridge=br0 i
fails:
```

可配置性

- ▶ 静态配置性
- ▶ 运行时配置

```
{
  "etcd": {
    "server": ["http://127.0.0.1:4001/v2/keys"],
    "user": "",
    "password": "",
    "prefix": "/keeper"
  },
  "mongo": {
    "host": ["127.0.0.1:27017"],
    "db": "test",
    "user": "",
    "password": "",
    "mechanism": "",
    "max_pool": 100,
    "table_prefix": ""
  },
  "mail": {
    "user": "abc@163.com",
    "password": "abc",
    "host": "smtp.163.com:25"
  },
  "group": "default",
  "debug": false,
  "delete_etcdkey_sync": false,
  "benchmark": false,
  "result2db": false,
  "auto_create_table": true,
  "api_overload_per_min": 60000,
  "use_zk": false,
  "use_kafka": false,
  "etcd_value_expire": 15,
  "use_api_salt": false,
  "auto_repair": true,
  "use_nfs": false,
  "use_mongo": false,
  "super_admin": ["admin"],
}
```

简单性

- ▶ 运维(ALL in One)
 - ▶ 一台机器能干完所有的事情
 - ▶ 一个二进制文件包含所有组件
- ▶ 开发(使有动态接口)
 - ▶ 增加新功能，客户端无需更新

```
[root@OP[REDACTED] backends]# cli upgrade
success
[root@OP[REDACTED] backends]# cli repair -i [REDACTED]
*****[REDACTED]*****
--2018-11-08 10:49:41-- http://[REDACTED]:8005/cli/upgrade
Connecting to [REDACTED] 8005... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]
Saving to: "/bin/cli"

      27,334      --.-K/s   in 0s

2018-11-08 10:49:41 (88.1 MB/s) - "/bin/cli" saved [27334]

cli daemon restart
```


可维护性

- ▶ 修复接口
- ▶ 日志查看接口
- ▶ 状态查询接口

```
~]# cli status
{
  "count": 7357,
  "offline": 1549,
  "online": 5808
}

~]# cli run_status
{
  "Redis.ActiveCount": "3",
  "Redis.MaxActive": "5000",
  "Sys.Alloc": "37060144",
  "Sys.Frees": "9573909715",
  "Sys.GCCPUFraction": "0.000147",
  "Sys.GCSys": "4034560",
  "Sys.HeapAlloc": "37060144",
  "Sys.HeapObjects": "257576",
  "Sys.NumGC": "24010",
  "Sys.TotalAlloc": "669384206120",
  "Sys.num_cpu": "8",
  "Sys.num_goroutine": "318"
}

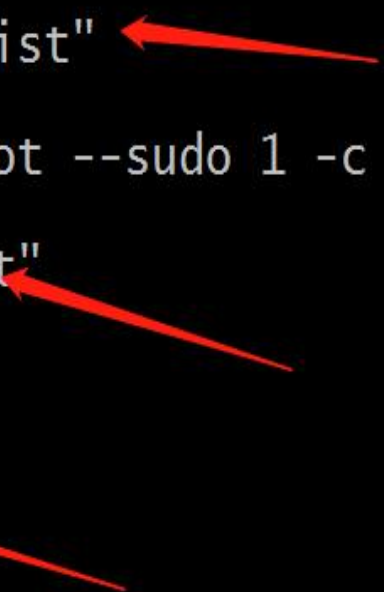
~]# cli check_status
{
  "db": "ok",
  "etcd": "ok",
  "redis": "ok"
}

~]#
```

安全性

- ▶ 环境隔离
- ▶ IP黑白名单
- ▶ token认证
- ▶ 用户名密码认证
- ▶ 使用SQL参数化

```
[root@localhost ~]# cli api -u root --sudo 1 -c hostname
{
  "message": "(error)token not exist"
}
[root@localhost ~]# cli api -u root --sudo 1 -c hostname
{
  "message": "(error)ip not permit"
}
[root@localhost ~]# cli login
please input username: jqzhang
please input password:
(error) password is error
[root@localhost ~]# cli login
please input username: jqzhang
please input password:
success
[root@localhost ~]#
```



Q & A