SSO重构项目复盘

OKR

O 构建架构更合理、服务更稳定、维护更简便的内部员工登录服务（两个双月）

KR1：整理SSO服务文档，完成B端服务、账号服务正式交接，日常运维人力减少80%（3.75）

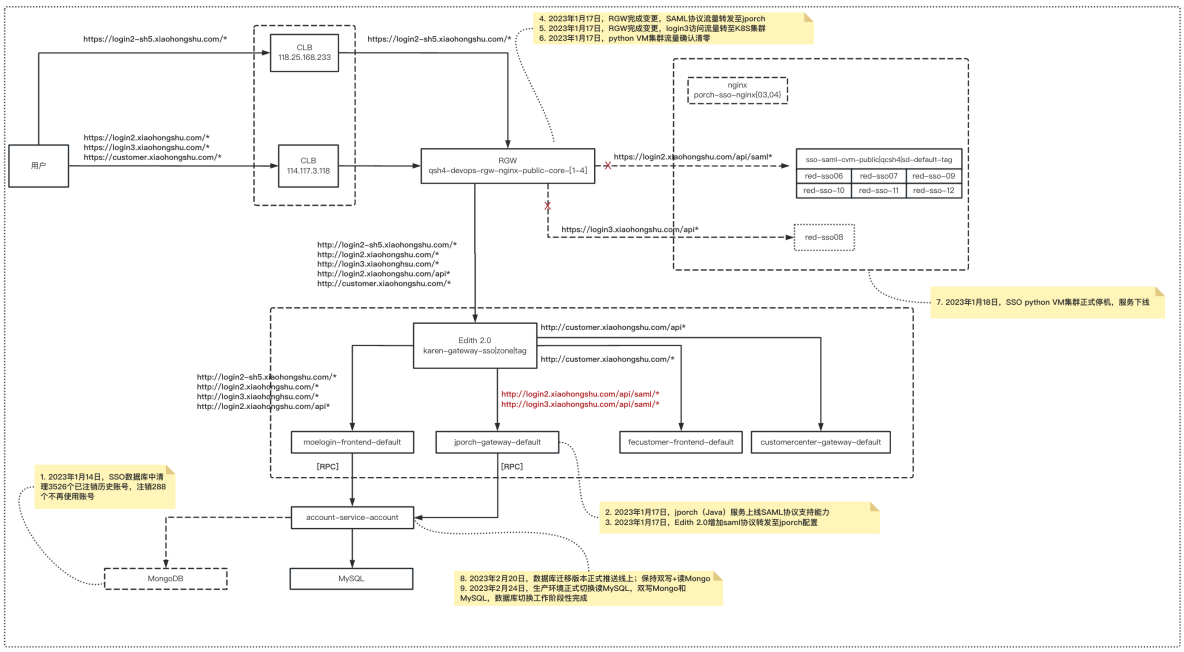
梳理PORCH系统整体架构和相关介绍，并和电商（真姬）、企效（宏峰）、IT（莱茵）对齐，整体介绍SSO的架构和相关功能。「SSO统一登录服务号」拆分成两个服务号：「商家账号服务号」和「员工登录服务号」，2月23号正式将「商家账号服务号」交接给电商团队。收集近一周服务号的咨询情况如下：

服务号	0220	0221	0222	0223	0224	0225	0226	0227	Avg
商家账号服务号	9次	6次	5次	8次	7次	4次	1次	10次	6.25
员工登录服务号	-	-	-	-	2次	0次	0次	1次	0.75

目前数据量偏少，整体看切换成「员工登录服务号」后，每日进线的咨询量会降低到非常低的阈值。

KR2：架构优化重构，全年SSO服务无P2及以上故障，P2以下故障数<3次（3.5+）**

技术架构调整主要改动点见下图：



核心里程碑节点：

- 1月14日，启动对历史无效账号的清理，注销部分无效账号（2月25日再次注销1000+客服无效账号）
- 1月17日，SAML协议正式迁移Java服务（JPorch），相关改动发布生产
- 1月17日，接入层链路改造，Java版本的SAML服务正式上线
- 1月17日，python集群流量归零，正式停机

- 2月20日，数据库迁移的代码正式发布生产
- 2月24日，数据库正式切换读MySQL + 双写

改造过程未引发RCA故障，相关问题和解决情况如下：

问题	说明	处理人	发现时间	解决时间
Java版本SAML服务上线后导致quip无法登陆	login3和login2使用的SAML证书不一致 & 存在bug；修复完bug后重新配置quip使用的证书后，问题修复。	高俊康	1月31日	1月31日
DTS数据迁移相关问题处理	**问题一**：大小写问题（Mongo区分大小写，MySQL默认不区分）**原因**：mysql 默认的 collate 配置与 mysql server 的版本有关，但都是不区分大小写的**解决方案**：使用这些值来避免问题： ○ utf8_general_cs (case sensitive) ○ utf8_bin (sort by binary sequence) ----------**问题二**：String文本前后空值问题（MySQL默认对字符串进行trim操作）**原因**：CHAR, VARCHAR, TEXT类型数据默认忽略右侧空格，导致 DTS 同步事出现 duplicate key。MySQL :: MySQL 5.7 Reference Manual :: 11.3.2 The CHAR and VARCHAR Types**解决方案**：DTS侧未提供迁移 log，通过编写diff工具定位出现问题的数据，并根据如下规则解决此问题： ● 迁移时过滤掉带空格的记录，直接丢弃。 ● 修改了 user 表的 collate 配置来支持区分大小写。----------**问题三**：DTS同步至MySQL中的部分字段被截断**原因**：部分数据长度超出 mysql 表字段设置长度，迁移存量数据时超过长度的部分会被截断**解决方案**：通过编写diff工具定位被截断的记录，并修改MySQL的ddl，重新同步数据；	廖立标高俊康郑剑周凡	2月9日	2月22日
密码缓存字段双库写入值不一致	diff工具比对数据发现密码缓存字段不一致，定位是该字段更新逻辑有问题，调整相关代码后该问题得到修复。	廖立标	2月23日	2月23日
头像和用户名更新逻辑存在BUG	diff工具比对数据发现双库用户信息存在不一致。分析后定位是原头像和用户名在Mongo数据库上的更新逻辑中存在并发问题。通过增加锁机制，调整相关代码后该问题得到修复。	廖立标周凡	2月23日	2月25日
生产user table存量数据迁移后user.auth_token出现不合法的"BsonNull" 值	dts 对 null 字段的转换缺陷，与 dts 开发同学沟通后修复。	廖立标周凡高俊康	2月24日	2月24日

问题	说明	处理人	发现时间	解决时间
线上部分 rpc 接口调用报错	Repository 层部分接口不合法的 null 入参导致拼出的 sql 存在 syntax 问题，对出现问题的 Repository 做了入参的兜底处理后解决。根本原因是数据层差异。 ● 对 mongodb 来说 null 作为查询条件是指该字段为 null or not exists，对 mysql 来说则会直接导致 sql 不合法。 ● 在 mongodb 中部分 collection 的字段的 null 值是有业务意义的，在 mysql 中使用空串模拟这些字段的 null 值意义，于是在入参和出参都需要做转化处理来适配以往的行为。	高俊康	2月25日	2月25日
porch2 登录后系统加载异常	sql 的缺陷，将对应 mysql repo 层接口逻辑与 mongo 同步后解决。	镜悬高俊康	2月25日	2月25日
小红书开放平台注册功能异常	相关逻辑执行的 sql 在 user table 上全表扫描，导致调用方调用超时。删除对应字段的查询对业务基本无影响，删掉后正常。	镜悬周凡	2月25日	2月25日
mysql 查询 operation 结果与 mongo 不一致	原因：当 action 字段为 -1 时，不应该用 action 作为查询条件，但是sql中并不符合以上逻辑，导致多出来的条件使查询结果为空。方案：根据逻辑更改sql，当 action 为 -1 时，不做为查询条件	郑剑	2月27日	2月27日

遗留问题清单

\1. jporch 中 SAML 登出接口已实现但未经业务系统测试，目前无业务系统使用 SAML 登出的逻辑，它们会自行维护登录态。

过程复盘

SAML 协议迁移 Java

迁移方案

基于 OpenSAML 开源包将 SAML 实现从 Python 迁移到 Java。

详细方案:SAML SSO实现现状与迁移方案

开发方案

- metadata 的获取
  - account rpc 新增接口查询 SP 信息，校验 SAML 请求。
- 测试验证
  - 本地反向代理，把待测接口的流量代理到本地，通过 sit 鉴权，测试线上 SP 服务。
  - sit gitlab 的 SAML 登入测试。

存在问题

- 对 SAML 协议不了解，开发同学本身也是摸着石头过河，影响开发进度。

Mongo 数据库迁移 Mysql

迁移方案

读写分离，迁移逻辑上线双写**增量数据**，dts 迁移**存量数据**，存量数据迁移完成后读流量切换到 Mysql 观察，确认无问题后下线 Mongodb。

详细方案: SSO存储迁移MySQL方案

开发方案

account 服务的 Repository 层封装了所有与 Mongodb 交互的逻辑，实现一套与 Mysql 交互的 Repository 来替换这一层，做好与 Mongodb Repository 的兼容从而做到迁移过程对上层（下文称 service）透明。

- 读写开关

实现一套基于配置的读写流量切换逻辑，能够做到线上读写调用的热切换。

读写开关的实现方案记录

- Repository 代理
- 为了令读写开关的切换对 service 层透明，实现了一套 Repository 位于 service - mongodbRepo/mysqlRepo 之间根据开关分发读写调用。
- 部分接口读写伴随在一起，无法分离，需要在不同的迁移阶段将逻辑提升到这一代理层进行更细粒度的读写调用分发。

验证方案

- 数据

写了工具来对比 prod 环境 db 的数据差异，repo -> gaojunkang/db-diff。

- Mysql Repository 层接口行为

运行时 compare mongo repo 与 mysql repo 的返回差异。

问题和改进

- 测试有些草率，QA 全程参与到项目中会对项目进度的推进有很大增益，包括立项、迁移方案的讨论制定、开发方案的讨论制定等。

QA 在基本了解开发方案的大部分细节，开发中遇到的主要问题的解决方案的大部分细节，以及对应服务上层的基本业务逻辑后，就能基本保证测试用例的有效性。

- 在逐步上 sit/staging/prod 环境的过程中暴露了越来越多的问题。
- 过去 Repository 层的接口设计和实现存在严重问题。
  - 复杂业务逻辑。
  - 在一个接口中既读又写，导致读写调用无法简单进行调用分发，提升了开发复杂度。
  - Repository 层反向调用到 service。

与 db 直接交互的抽象层应该保持纯洁性，不应包含业务逻辑，且应尽量少地或不包含从入参到写入 db 的数据的处理逻辑，从而尽量屏蔽数据层差异。

- 开发同学的单元测试覆盖率不够，忽略了一部分边界 case。