

Rehab 拆分方案设计

背景综述

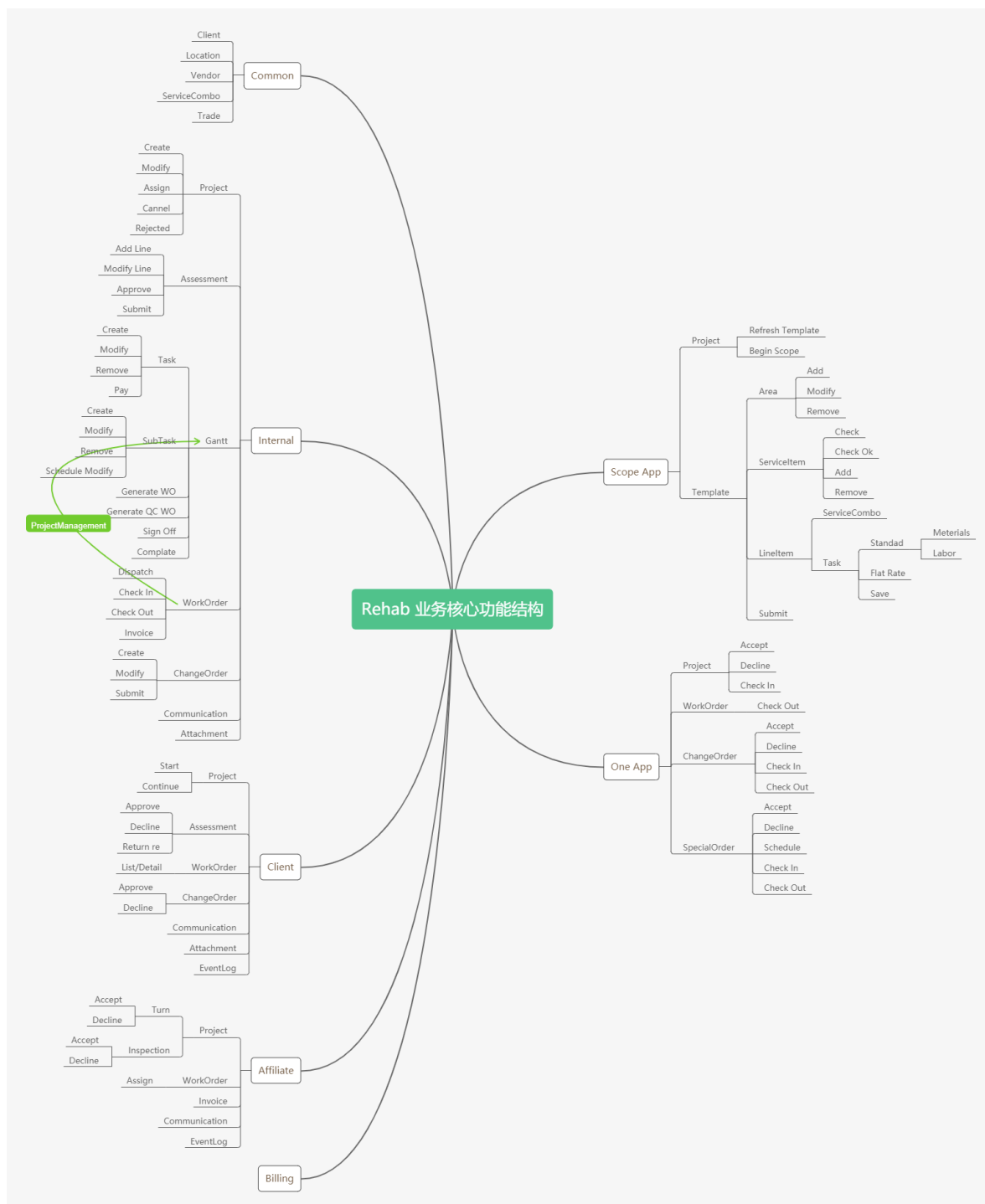
现在的Rehab系统由One的Inspection和Turn子系统整合演化而来，目前存在架构层面上与其它模块高度耦合，业务实体模型与实际业务发展要求脱节，已经出现开发成本提高，系统多处出现性能瓶颈等问题。为了更好支撑Rehab业务的发展，达到公司随时发布的目标，Rehab需要从One系统中独立拆分出来，本文档将描述拆分的整体方案思路与设计。

文档对象

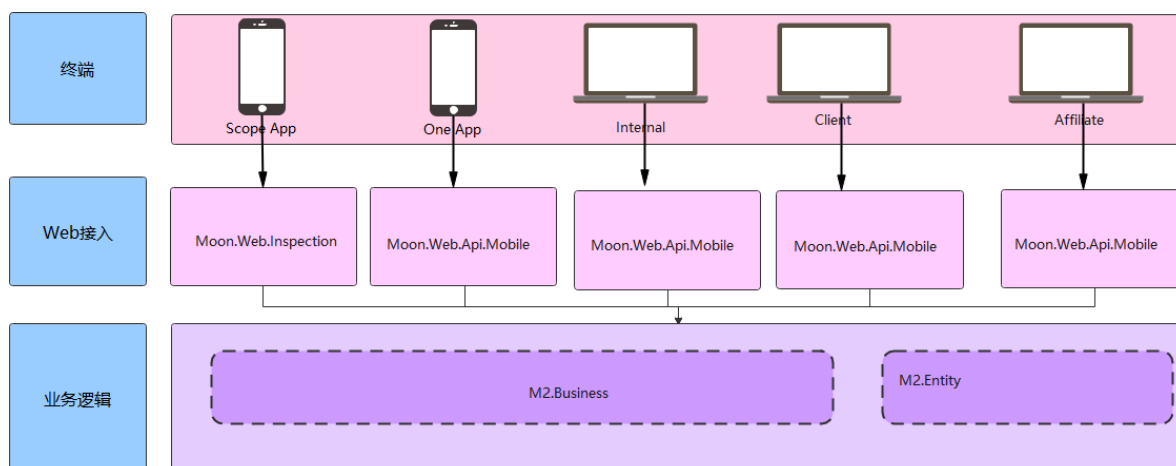
公司领导与项目组开发测试等相关人员

系统现状

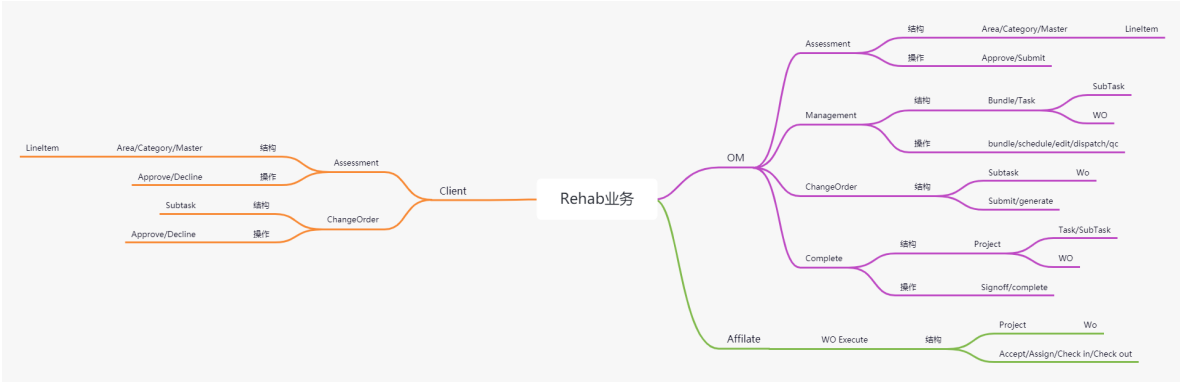
核心功能



逻辑架构



核心模型操作



存在的问题

Rehab核心模型关系是Project -> LineItem/Task/SubTask -> Wo 三级关系

在Assessment 阶段后, SubTask会生成对应的WO, 形成了互联耦合关系, 在很多业务场景下修改信息或状态变更都需要反向同步信息

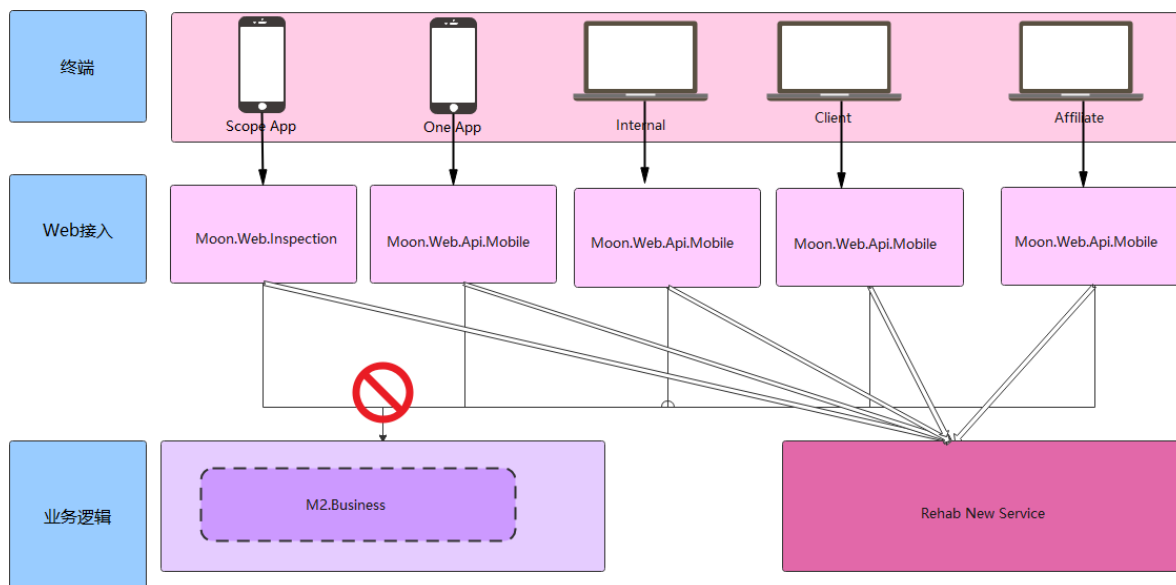
目前的One App应用, 有些业务场景是在Project级别, 会产生批量WO更新操作, 出现性能问题

拆分方案概述

方案名称	说明	优缺点(风险)
就地改造	从技术层面上出发将Rehab从One系统拨离开来, 将代码和数据进行分离, 实现可独立发布。实现方法是从Moon代码仓库复制一份源代码, 梳理代码和数据表, 只保留Rehab相关的逻辑代码和存储表。	分离的工作量比较大和复杂, 虽然可以实现了独立发布目标, 但对后续业务开发没有太大改善
WO解耦	现在的Rehab比较重度依赖One系统的WO功能, 包括Schedule, Dispatch, QC, Check in/out, Invoice等。由于WO是One通用的业务流程处理, 业务逻辑比较重量级和复杂, Rehab实际只使用其中一部分功能。目前许多的开发痛点与WO相关, 如批量WO操作性能差等。此方案的思路是将Rehab与One 的WO解耦, 定制适合Rehab自身的WO业务模型	需要梳理Rehab WO业务逻辑和UI效果, 单独拆分WO需要与现有Rehab进行数据同步, 对能提升后续业务开发有部分改善和部分实现随时发布的目标
核心重来, 保留外观	保留现有的终端外观界面, 后端的Rehab业务模型重新设计开发新的Rehab服务, 新的Rehab服务实现要满足现有的Rehab业务流程和效果, 在现有的Web接入层实现适配, 接入新的Rehab服务	需要整体梳理Rehab业务流程和逻辑, 开发周期和工作量会比较大, 基本可以实现独立的目标和提高后续的开发效率
洗心革面, 彻底重来	完全重新设计开发, 包适主要UI	

Rehab新服务设计

技术架构



实现方法

梳理现有的Rehab当前业务场景和流程逻辑，形成完整需求文档. 对现有需求优化和重新规划，设计新的Rehab Service，具体的实现策略可采用MVP，新开发的每个端API逐步适配接入等

迁移路径

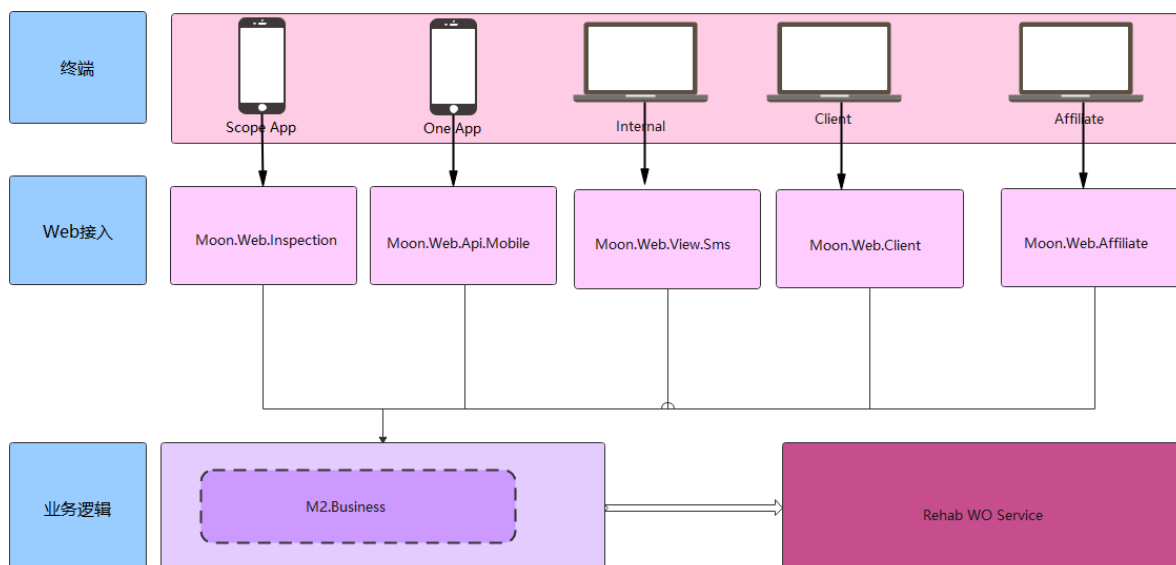
按Client?

按Project数据版本?

直接数据同步(新旧兼容)?

WO解耦方案

技术架构



涉及的业务场景

角色	场景描述	
OM	bundle, schedule, dispatch, change order, qc, signoff, complete	
Affiliate	accept, assign, check in/out	
Client	change order	

在Assessment阶段后，基本上比较核心的操作都牵涉WO

其它考虑

如果单独解耦WO, 由于存在关联关系，需要根据业务场景保持WO与Task数据同步

其它问题

- 1. 依赖的公共基础数据如何处理?
- 2. 依赖的其它公共模块?