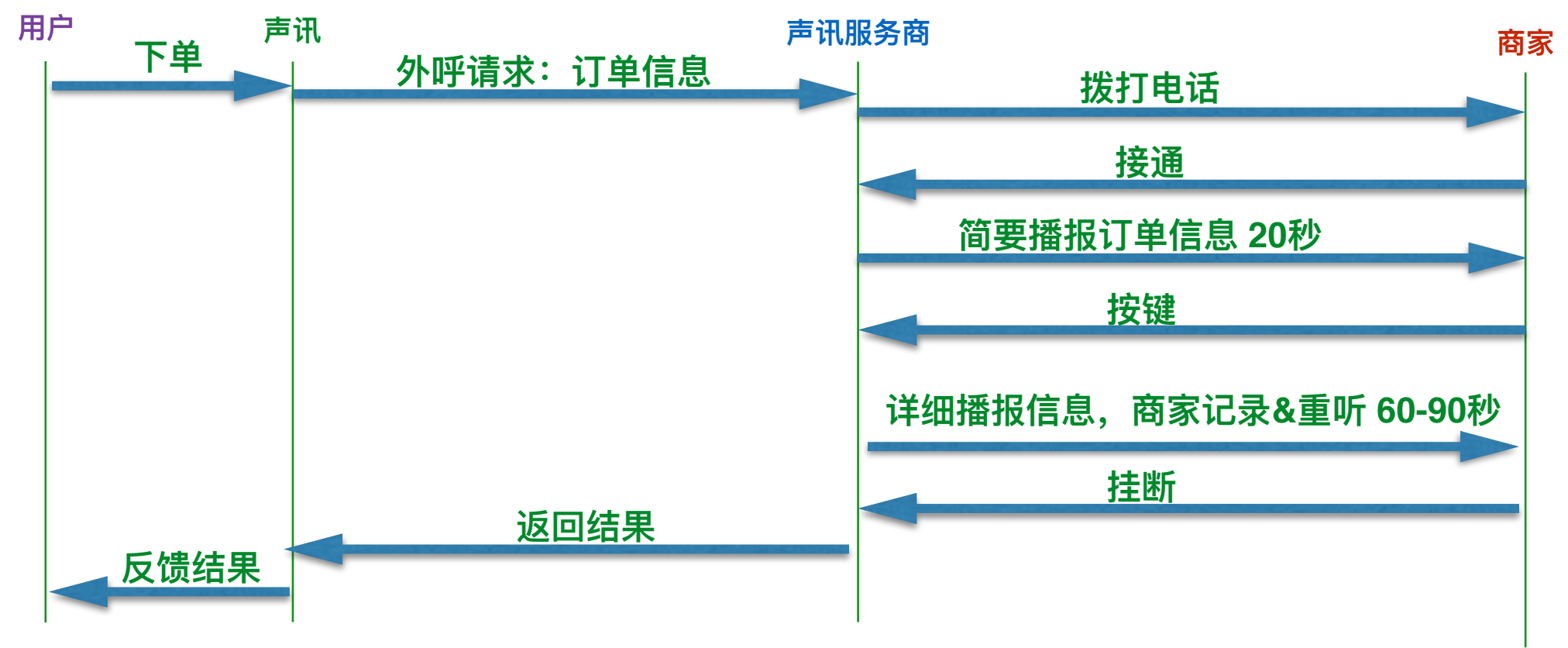


声讯平台串讲

李华卿@预订技术部 2017.12



天使轮



用户下单后，将订单信息，以声讯电话的形式触达商家，并提供接单、拒单等按键操作，使订单成交。

基本职能

- 外呼
- 处理回调
- 适配服务商

业务职能

- 维护、组装模板
- 重试机制
- 监控预警

老声讯缺乏抽象



合理抽象，**配置灵活**



领域模型设计

接单主要渠道



丰富重试，优化触达

对接多服务商



合理适配，**快速插拔**



代码结构设计

问题频发



问题自解决



业务指标制定

声讯 - 领域抽象

Template

- 外呼条件
- 动、静态文本 (SPEL)
- 重试规则
- 回调规则

树形结构

描述外呼文案

如何重试和回调

Trigger

- 执行的根模板
- 订单号、门店ID
- 状态
- 结果

记录任务及结果

驱动外呼

反馈给调用方

Call

- triggerId
- 第三方ID
- connect\press\hang状态及时间
- 结果
- 失败原因

记录每个电话详细过程

反馈给Trigger

Response

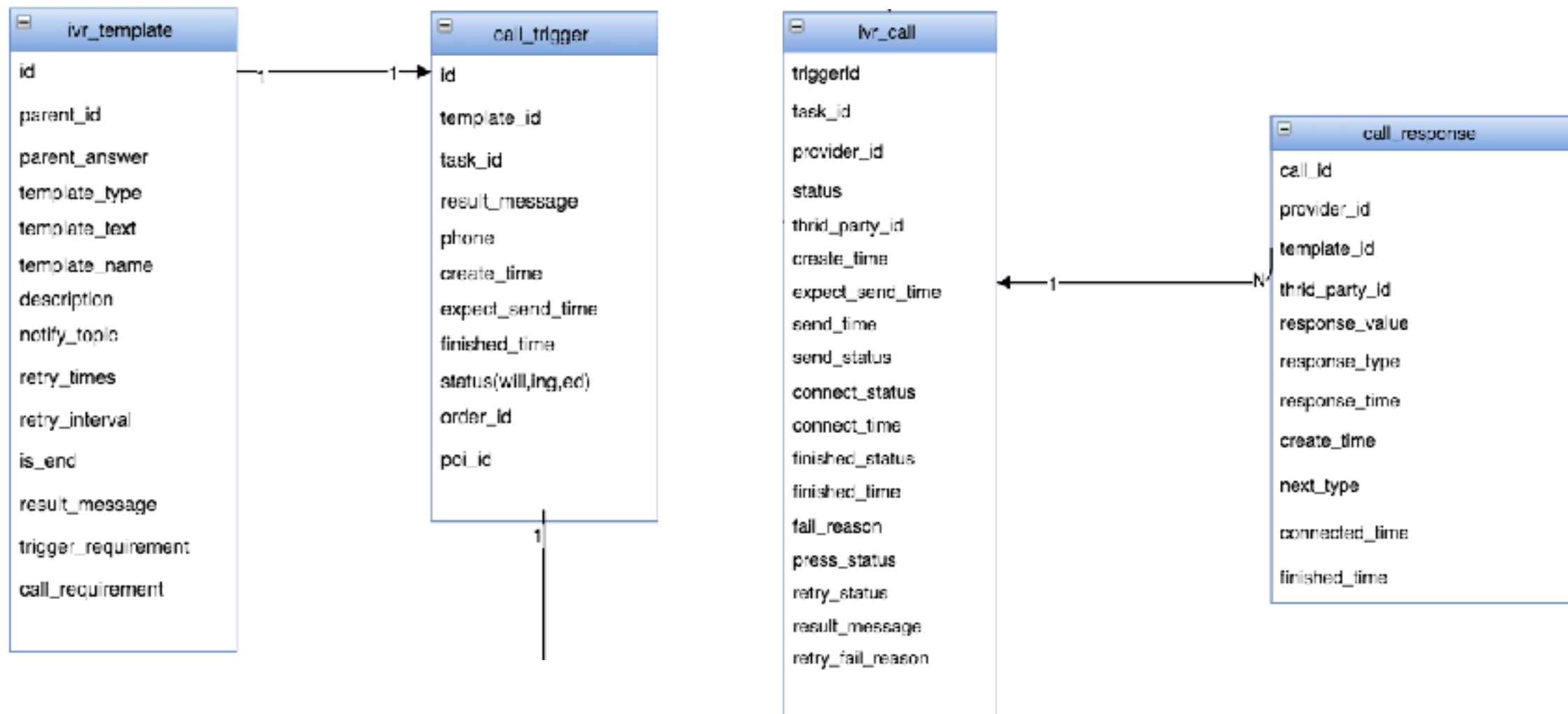
- callId
- 第三方ID
- 按键? 挂机?
- 按键结果
- 时间

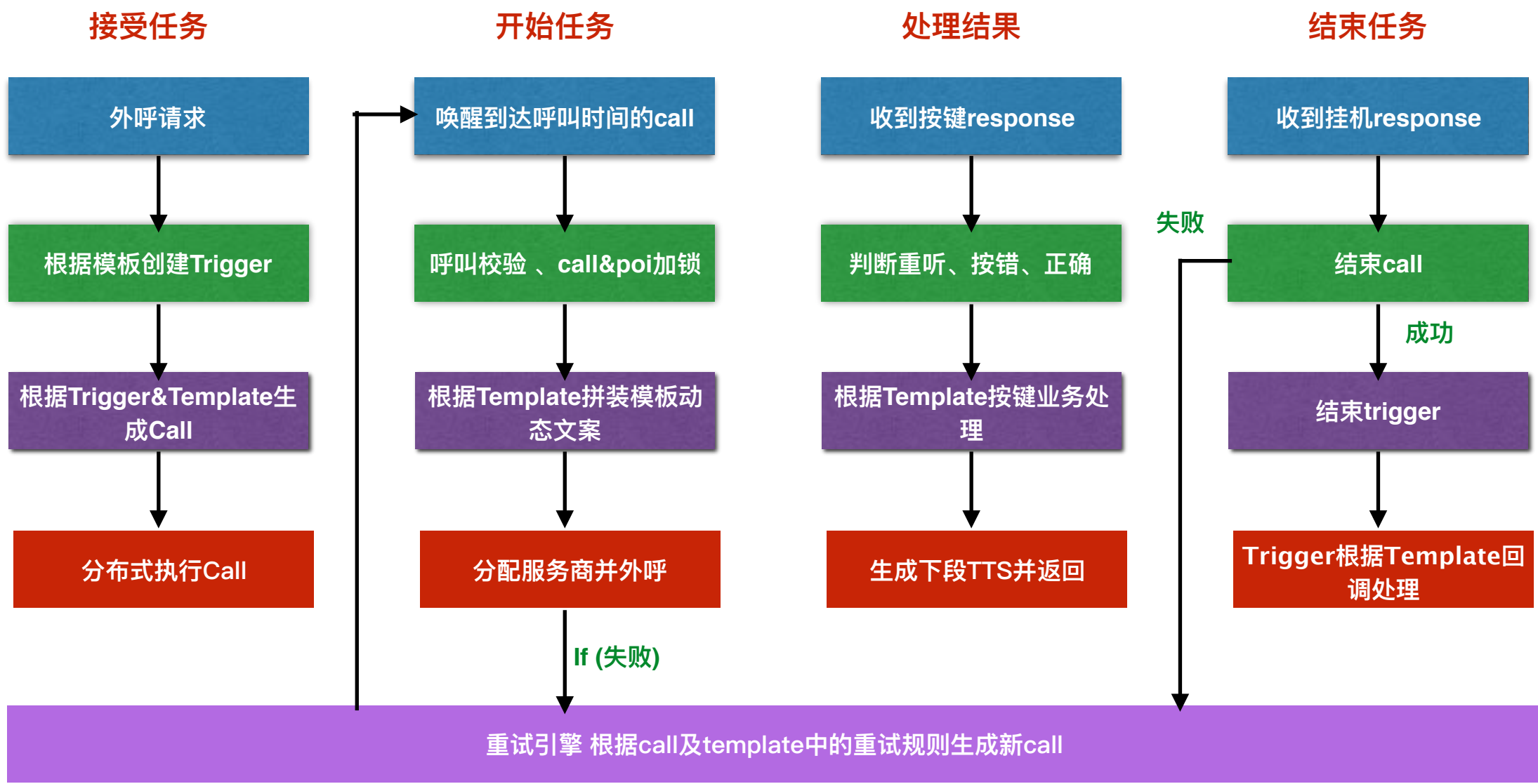
记录每个回调结果 (用户操作)

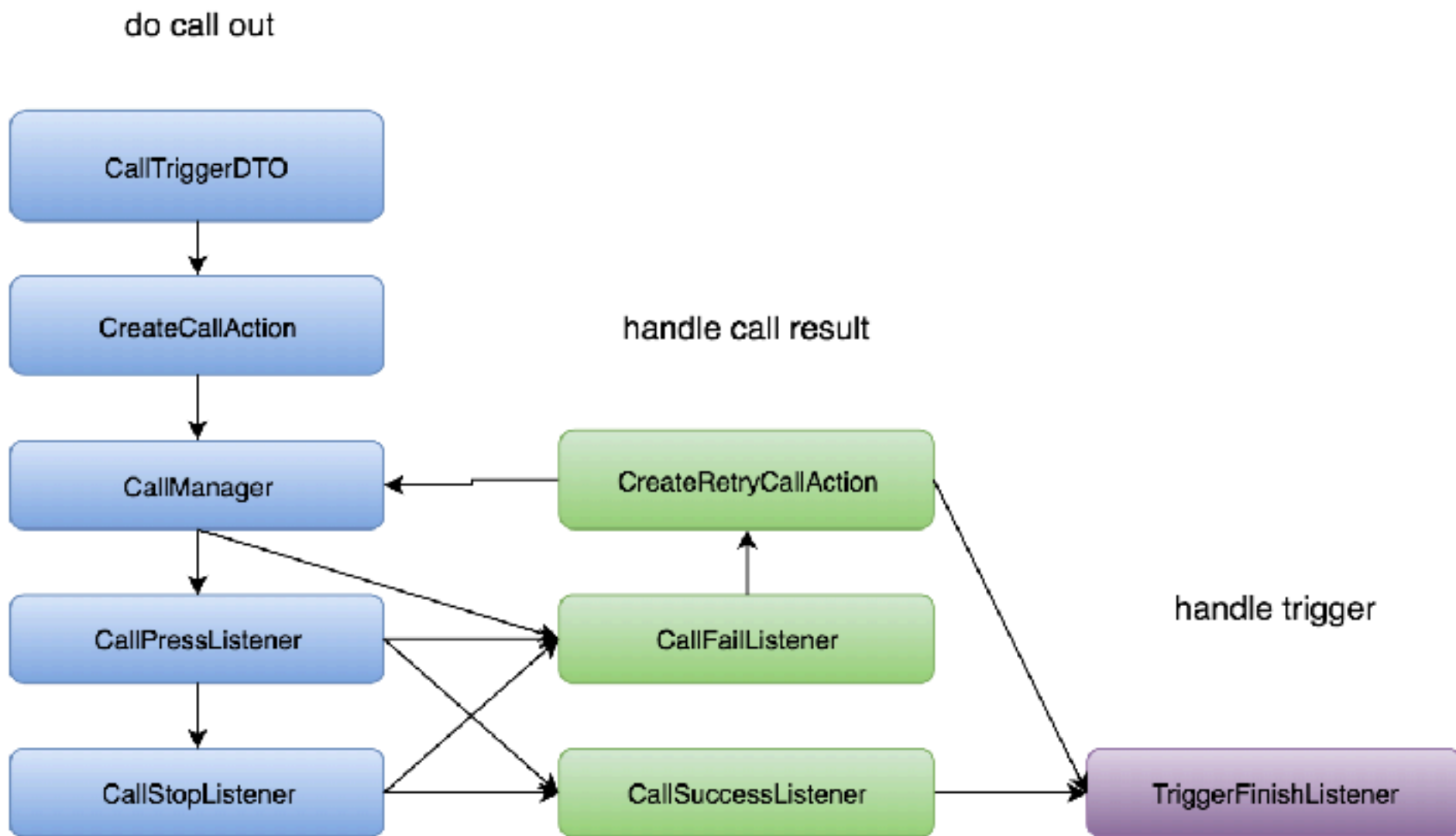
反馈给Call

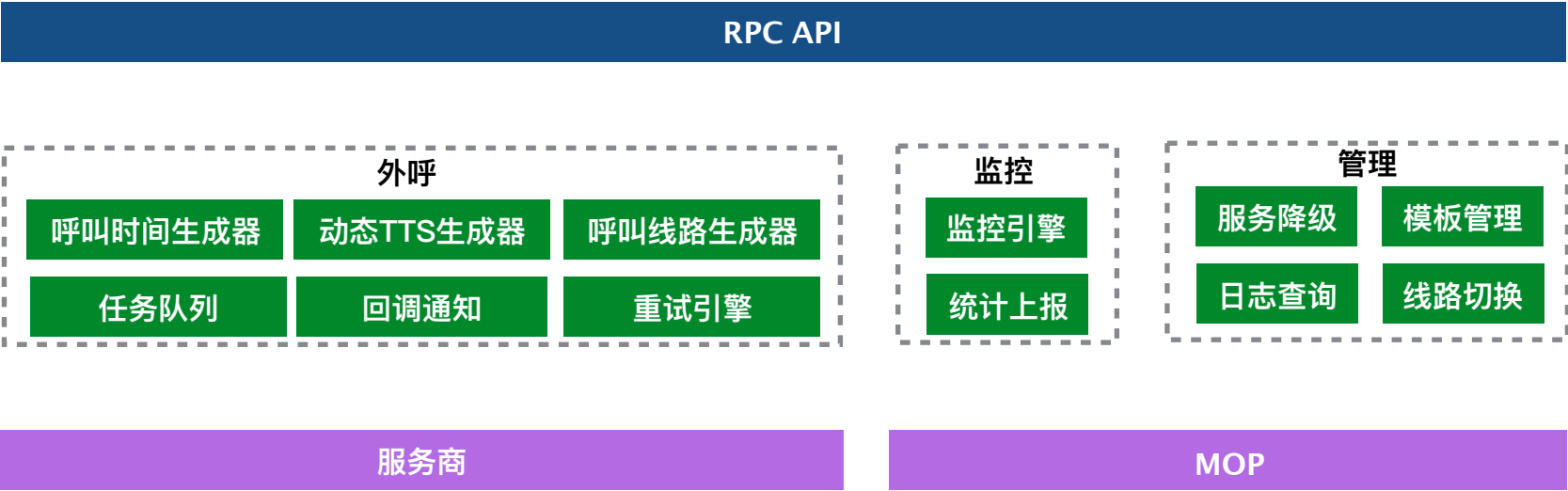
如同领域功能多需要拆分, 业务过程长也需细分, 核心:找驱动

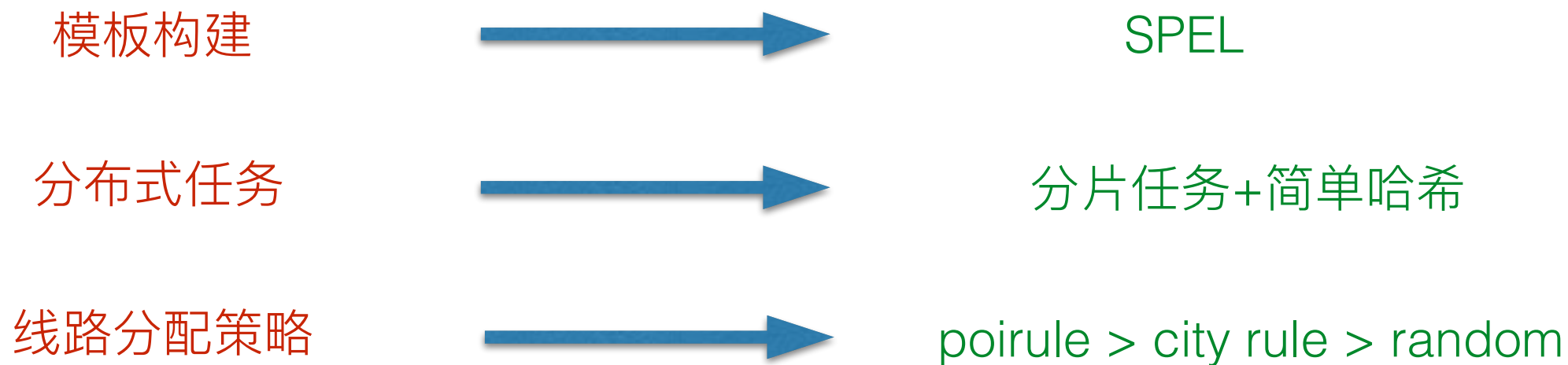
声讯 - 领域设计











声讯 - 模板构建方案选型

方案1: 手写

优点: 灵活, 容易上手, 配置简单随意, 发挥空间大

弊端: 容易出Bug

```
'美团点评预订, '  
+#@bookTimeMinPeople@#+  
    '人就餐, '  
+#@positionGrouponRemark@#+  
    '有座按1, 没座按2, 重听按9'
```

方案2: SPLE

优点: Spring原生功能, 可以与domain集成, 复用性高

弊端: 模板配置略显臃肿

'美团点评预

```
订, '+getPoilInfo().getPointName()  
    +getOrderInfo().getDeposit()  
    +getOrderInfo().getBookDate()  
    +getOrderInfo().getTimeType()  
    +getOrderInfo().getBookTime()  
+', '+getOrderInfo().getNumber()+ '人  
就餐, '+getOrderInfo().getPosition()  
+', '+getOrderInfo().getGroupon()  
    +getOrderInfo().getRemark()  
+getOrderInfo().getPreference()+ '有  
    座按1, 没座按2, 重听按9'
```

声讯 - 模板构建方案选型

```
private IvOrderInfoParam orderInfo;  
  
private StagePoiView poiInfo;  
  
private Integer unArrivalOrderCount;  
  
private String verifyCode;  
  
private IvBatchOrderParam ivrBatchOrderParam;  
  
private Long currentTime = System.currentTimeMillis();
```

```
/**  
 * 使用SPEL将IVRTemplate拼装为动态tts  
 * @param call  
 * @param ttsTemplate  
 */  
public String generateDynamicInfo(Call call, TTSTemplate ttsTemplate){  
    ExpressionParser expressionParser = new SpelExpressionParser();  
    Expression expression = expressionParser.parseExpression(ttsTemplate.getTemplateText());  
    IvDynamicParam ivrDynamicParam = ivrDynamicInfoBuilder.build(ttsTemplate, call);  
    EvaluationContext context = new StandardEvaluationContext(ivrDynamicParam);  
    return expression.getValue(context).toString();  
}
```

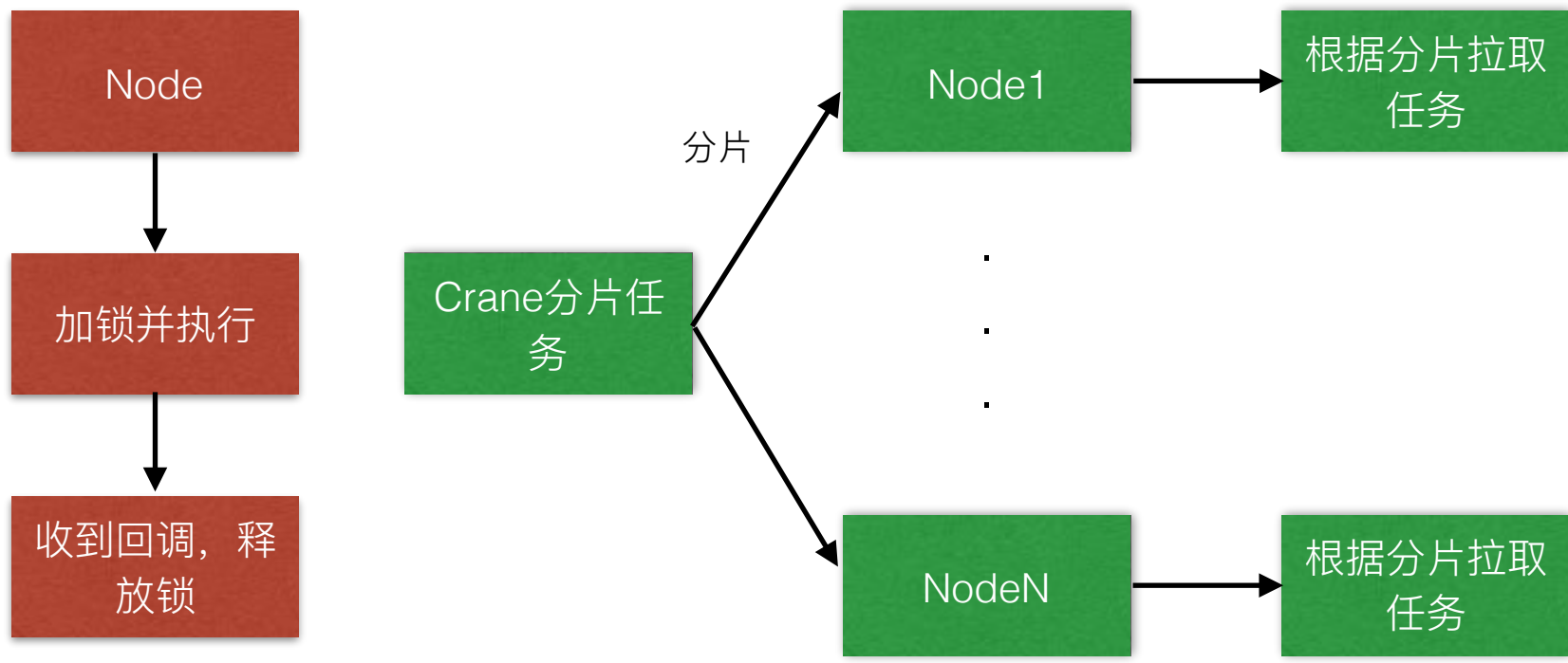
声讯-分布式任务技术选型

背景：高峰期任务并行数500，
任务要求高一致性，只执行一次
思路：高并发 -> 分布式任务
一致性 -> 分布式乐观锁

方案1：quartz
优点：现成成熟框架支持分布式任务
弊端：需要引入额外7张表，不适合微服务

方案2：crane分片任务
思路：拿到的分片作为page，分页处理任务
优点：避免了大规模锁竞争
弊端：无法保证数据拉取时的一致性

方案3：crane分片任务+简单hash
思路： $\text{poild} \bmod (\text{总片数}) = \text{得到的分片}$ 处理任务
优点：同时保证了一致性与平衡性，新增、重启机器不会导致任务阻塞



一致性 - 避免重复执行

平衡性 - 避免大规模竞争

声讯-线路分配策略

每个门店对线路、服务商的适应能力不同

核心规则：门店规则 > 城市规则 > 权重随机

权重随机方案：

方案1：实时读取权重，进行有权随机

优点：实现简单

弊端：随机性大，IO压力大，可能需要缓存

方案2：库存式扣减，根据权重生成相应库存，随机扣

优点：压力较小，稳定

弊端：库存在内存中

A轮

参数不一样



适配器+工厂模式

过程不一样



过程标准化，非标转换

思路：底层稳定，上层适配

```
/**
 * Created by lihuqing on 2017/3/2.
 */
public interface IvrserviceInterface {

    static final Logger logger = LoggerFactory.getLogger(IvrserviceInterface.class);

    default Boolean call(Call call){
        logger.error(" no support provider find call is {}",call.getId());
        return false;
    }

    Boolean isSupport(Integer providerId);

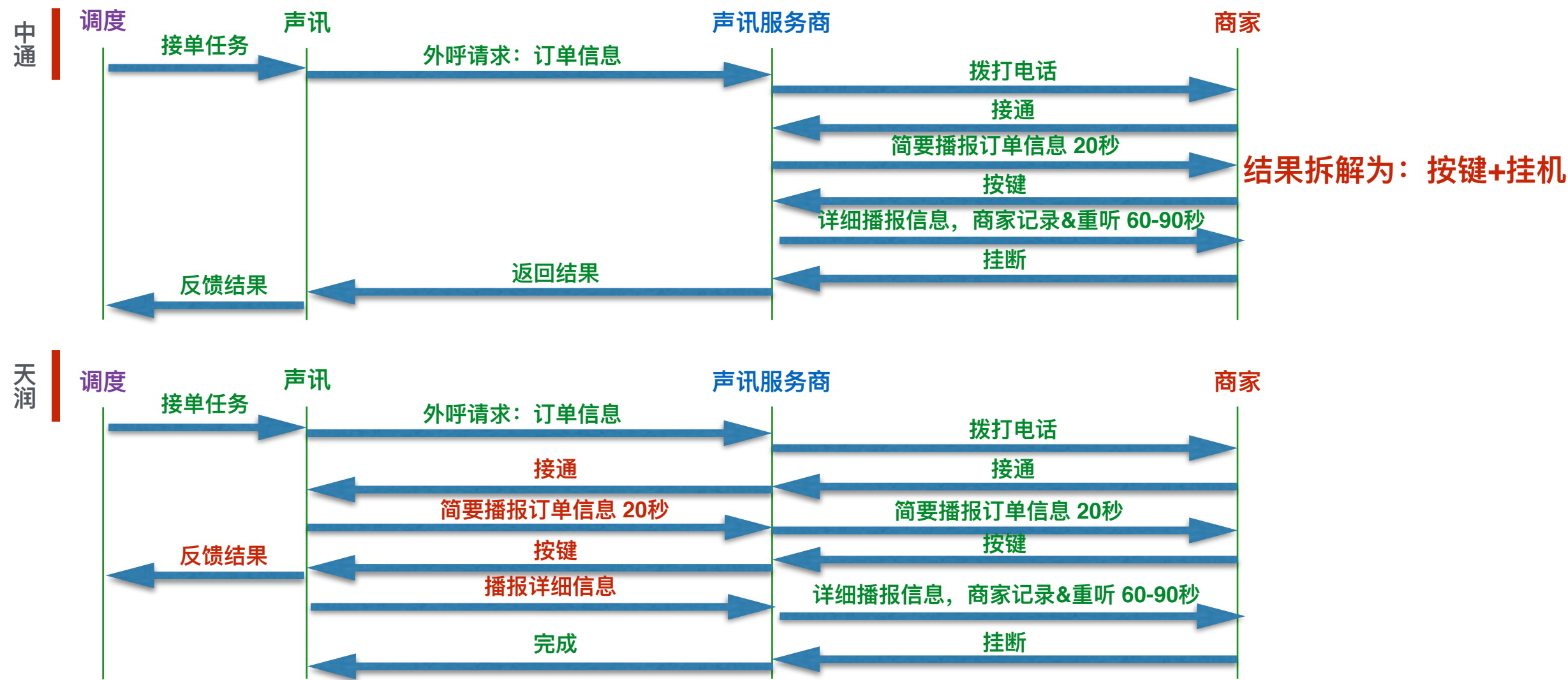
    //TODO
}
```

外呼使用适配器模式

```
CallResponseFactory  
  create(TianRunFinishNotificationModel): CallResponse  
  create(TianRunPressNotifycationModel): CallResponse  
  create(ZhongTongFinishNotificationModel): CallResponse  
  create(ZijianPressNotificationModel): CallResponse  
  create(ZijianStopNotificationModel): CallResponse
```

回调使用工厂模式

挑战1 - 适配过程



挑战2 - 领域日益臃肿

随着业务发展，规则更加细致，大量规则维护在ivr_template中

```
ivr_template
parent_id
parent_answer
template_name
template_type
template_text
notify_topic
retry_times
retry_interval
description
is_end
result_message
trigger_requirement
call_requirement
continuity_times
min_connect_time
no_press_retry_times
no_press_retry_interval
```

```
operation_text
repect_value
is_need_press
connect_short_retry_tim
connect_short_retry_int
call_time_type
is_immediately
right_answer_list
need_next_call
next_call_template_id
next_call_interval
dynamic_info_type
zhontong_template_id
```

挑战2 - 领域日益臃肿

Template

- 创建规则
- 呼叫规则
- 回调规则
- 时间规则
- 电话规则

描述类型单一
零散的规则

TTSTemplate

- templateId
- 动、静态文本
(SPEL)
- 按键列表
- 错误提示
- 第三方模板ID

树形结构

描述外呼文案

该怎么按键

如何提示

RetryRule

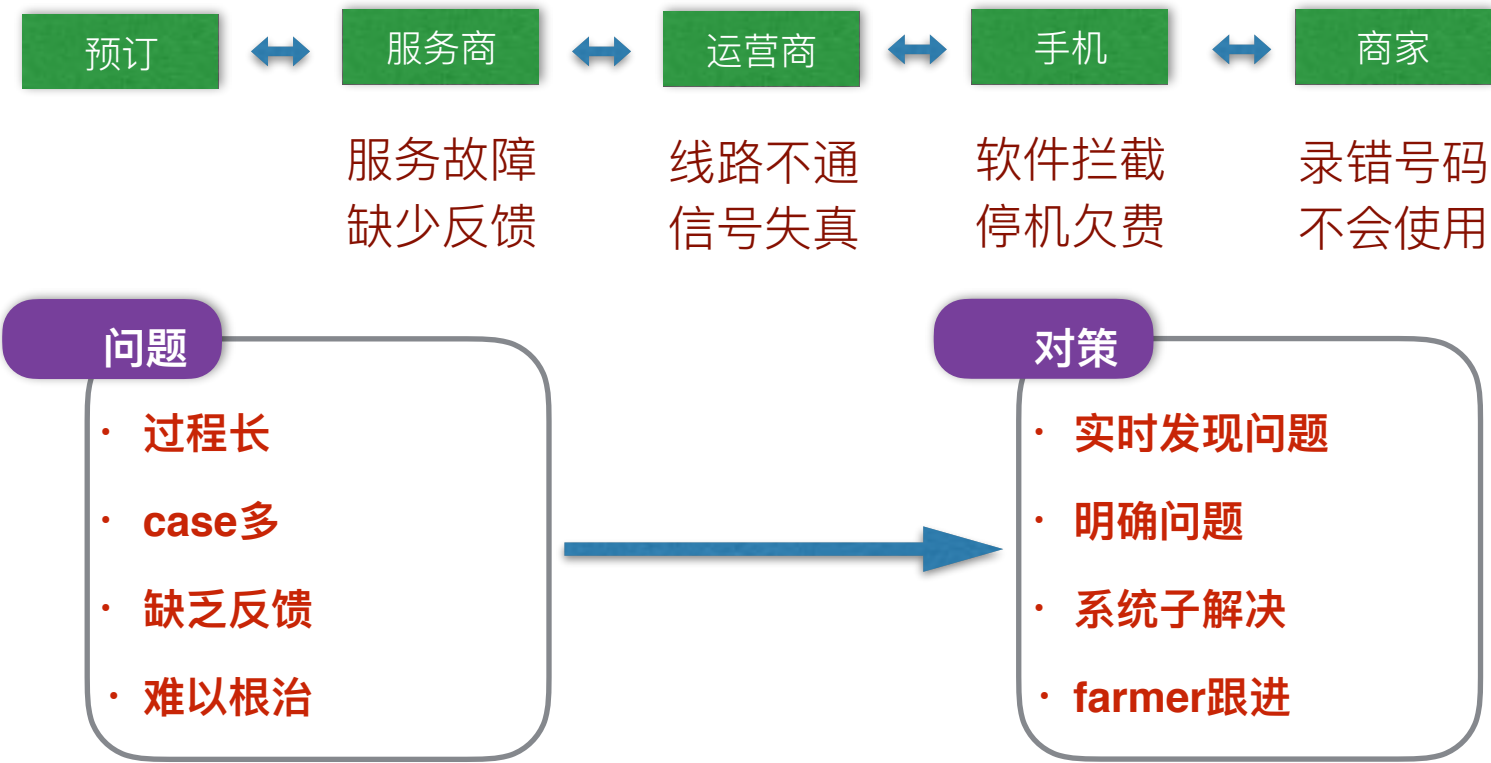
- templateId
- 未接通重试规则
- 未按键重试规则
- 接听时间过短重试规则

描述模板重试规则

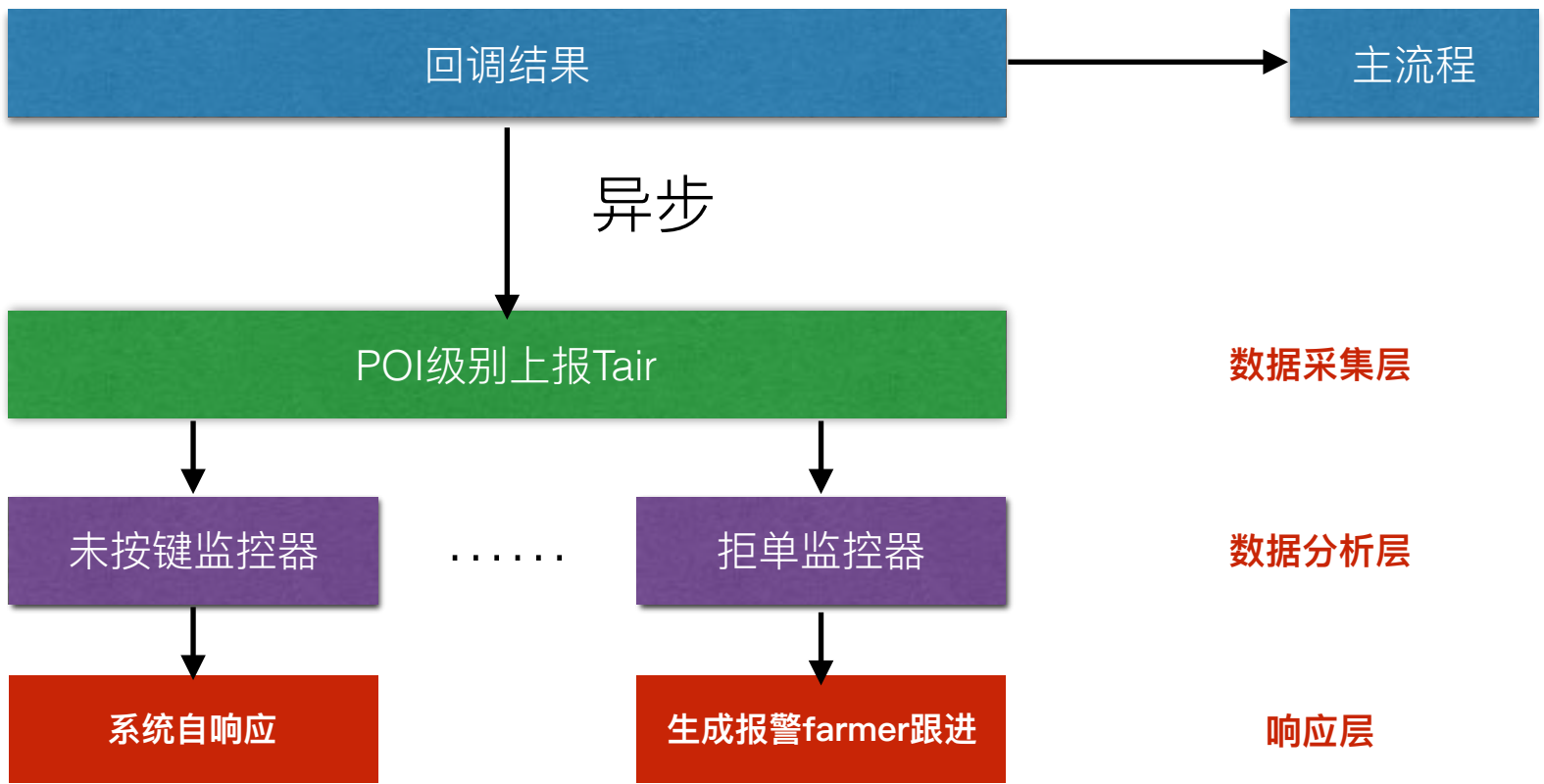
应该化繁为简，进行拆分，使功能更加内聚，核心:找维度

挑战3 - case多反馈少难根治

90%订单通过声讯接单



挑战3 - case多反馈少难根治





IVR服务商:自建

呼叫任务总数:10339

接通率为:59.13531289292968%

订单维度接通率为:88.11616354604509%

订单维度净接通率为:96.38453500522466%

按键率为:66.25776905462872%

订单维度按键率为:87.72766695576756%

平均反馈时间: 1.1min

IVR服务商:天润

呼叫任务总数:15198

接通率为:55.23095144097908%

订单维度接通率为:87.3445313501731%

订单维度净接通率为:93.00928454396505%

按键率为:78.56802477960449%

订单维度按键率为:93.70229007633588%

- 通话质量分析
- 细粒度监控：发现更多case，细化失败原因，丰富重试策略
- 精准性预测：寻找实时性与准确性平衡做监控
- 智能线路分配

谢谢

