

高德稳定性架构实践

阿里巴巴-高德 梁磊(雷娃)

leywar.liang@alibaba-inc.com

2015.10.10

Geekbang>

极客邦科技

全球领先的技术人学习和交流平台

扫我，码上开启新世界



Geekbang>

InfoQ | EGO NETWORKS | StuQ

InfoQ

专注中高端技术人员
的社区媒体

EGO NETWORKS

EXTRA GEEKS' ORGANIZATION
高端技术人员
学习型社交网络

StuQ

实践驱动的IT职业
学习和服务平台



促进软件开发领域知识与创新的传播



实践第一 案例为主

时间：2015年12月18-19日 / 地点：北京·国际会议中心

欢迎您参加ArchSummit北京2015, 技术因你而不同



ArchSummit北京二维码



【北京站】

2016年04月21日-23日

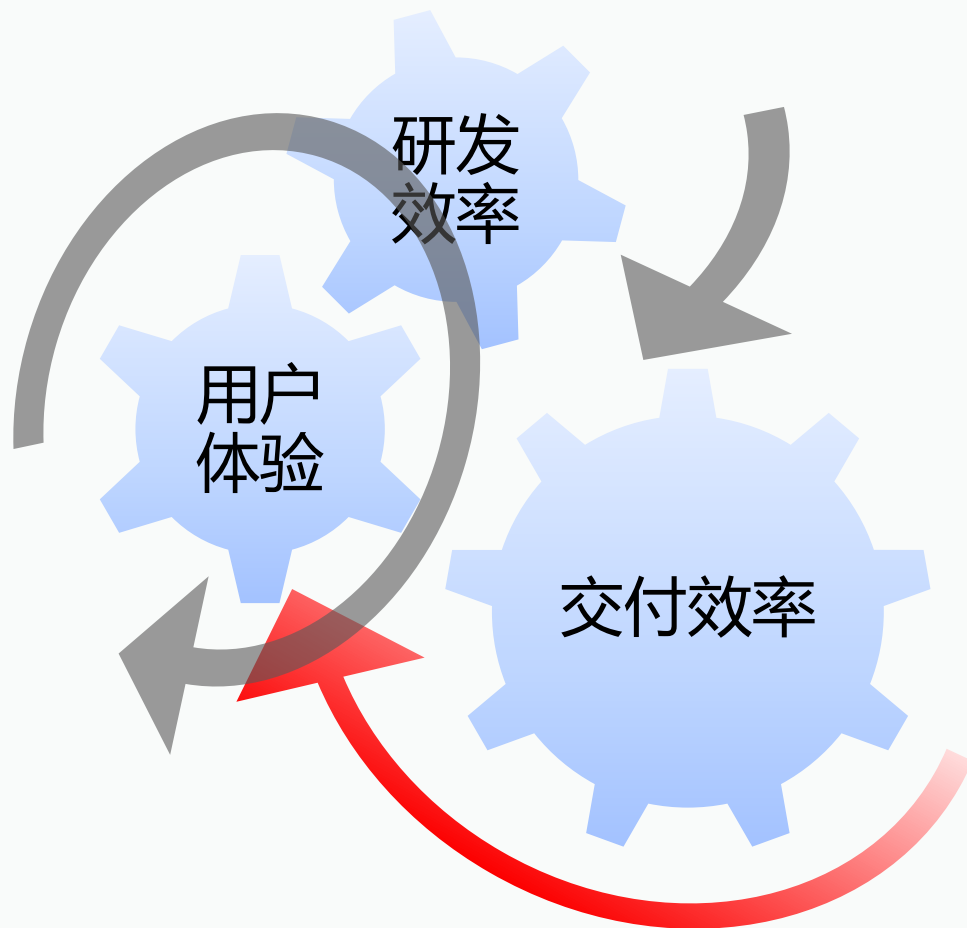


关注InfoQ官方信息
及时获取QCon演讲视频信息

概览

- 稳定性与在线系统
- 高德稳定性挑战
- 系统容灾一般原则
- 主动容灾架构设计
- Trace-问题诊断与跟踪
- 适应迭代的业务抽象
- Q/A

在线稳定性的影响



地图服务稳定性挑战

地图场景下的稳定性挑战

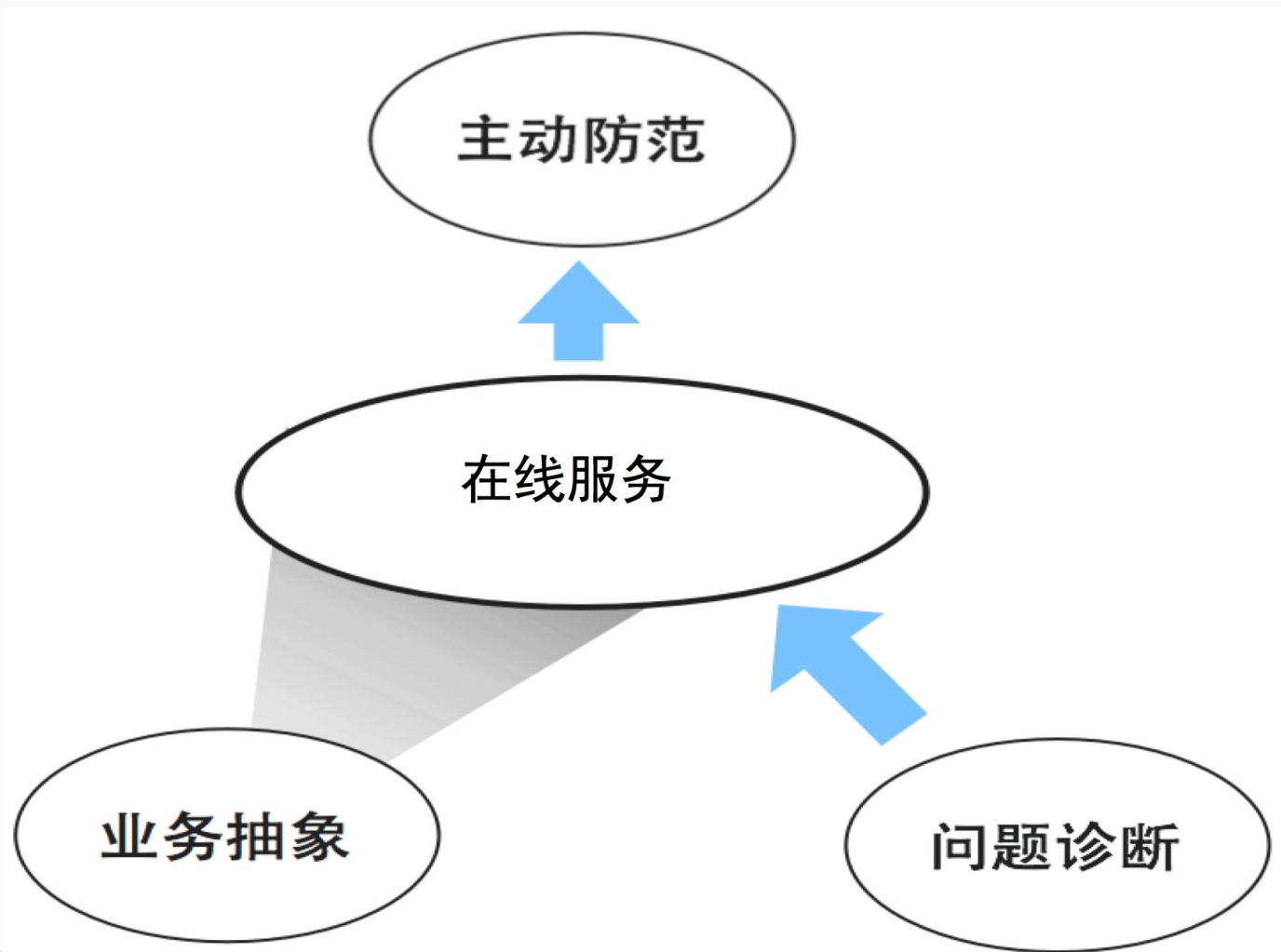
结果位置相关，位置的不可枚举性

Native应用快速发版与稳定性的矛盾(前向兼容/两周发版)

多上游业务差异化：

手机端/PC/Web MO/开放平台/苹果地图/车联网/导航…

稳定性架构设计一般原则



I. 主动防范体系架构

在线服务分层模型

在线服务分层抽象



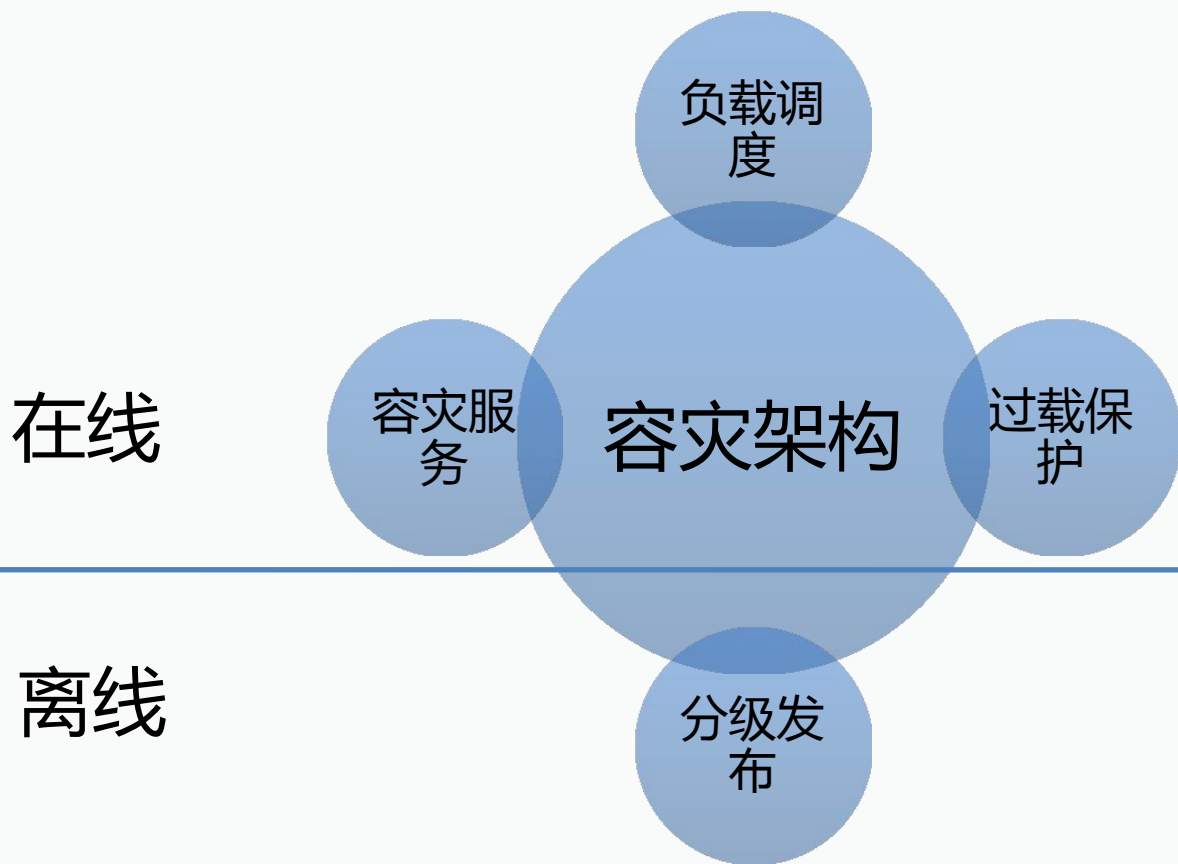
在线服务分层容灾能力

分层	容灾能力
系统级	<ol style="list-style-type: none">1. 防攻击能力2. 快速的流量切换与引导
接入层	<ol style="list-style-type: none">1. 有效的流控机制(TMD)2. 异常流量的检测与快速屏蔽机制3. 必要的降级预案
服务层	<ol style="list-style-type: none">1. 服务的自我保容能力2. 服务的自我容错机制3. 服务的快速诊断机制4. 服务的降级预案5. 服务的分级发布和快速回滚能力
数据层	<ol style="list-style-type: none">1. 数据的预上线与分级发布机制2. 数据的校验与异常保护机制

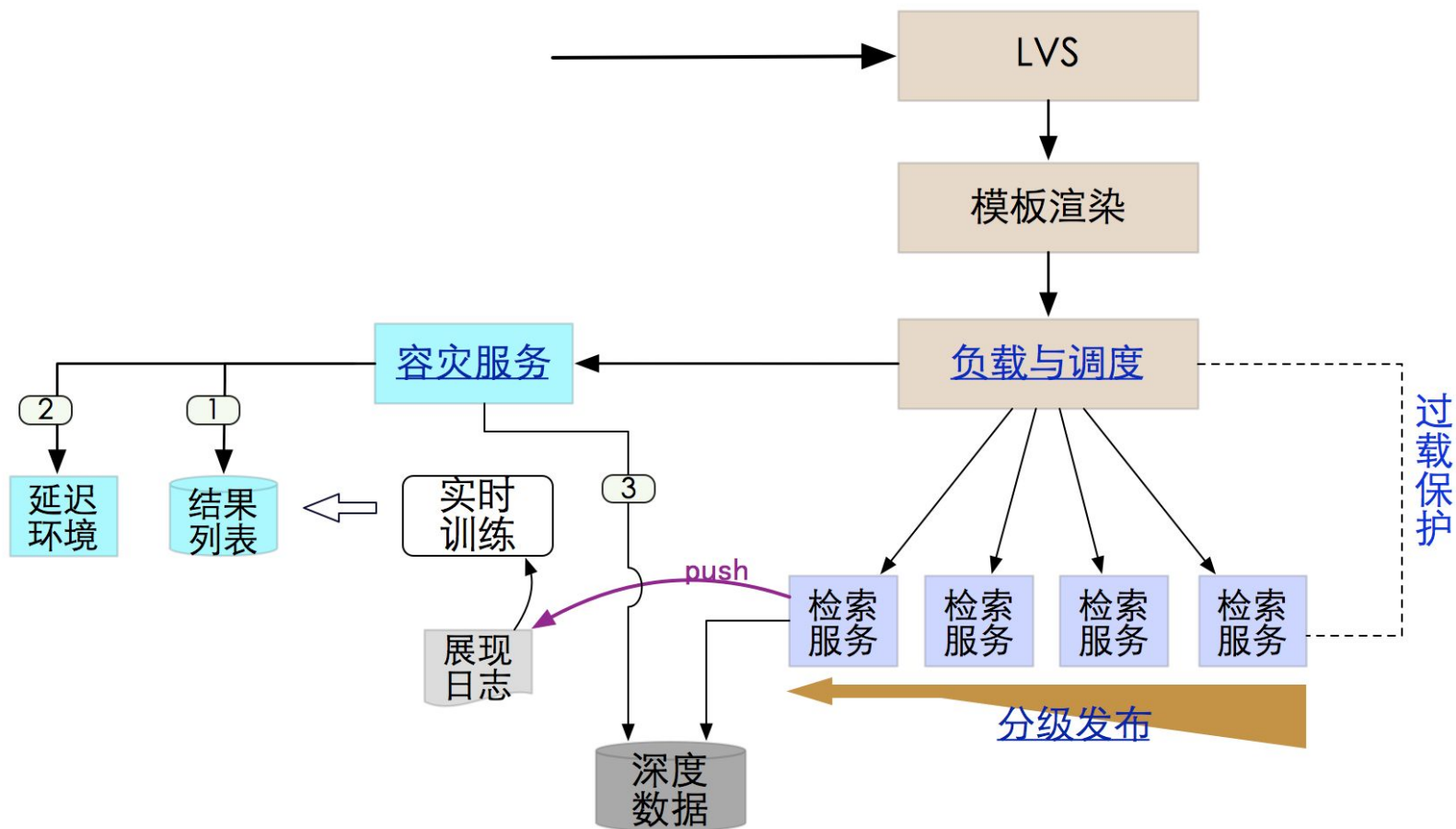
稳定性风险最高

搜索主动容灾系统

搜索容灾架构



搜索容灾架构



过载保护

核心目标

服务自我保护, 异常时主动进入过载保护状态
服务不能被打垮, 在保护的容量上限下稳定服务输出
(网络/交换机等异常除外)

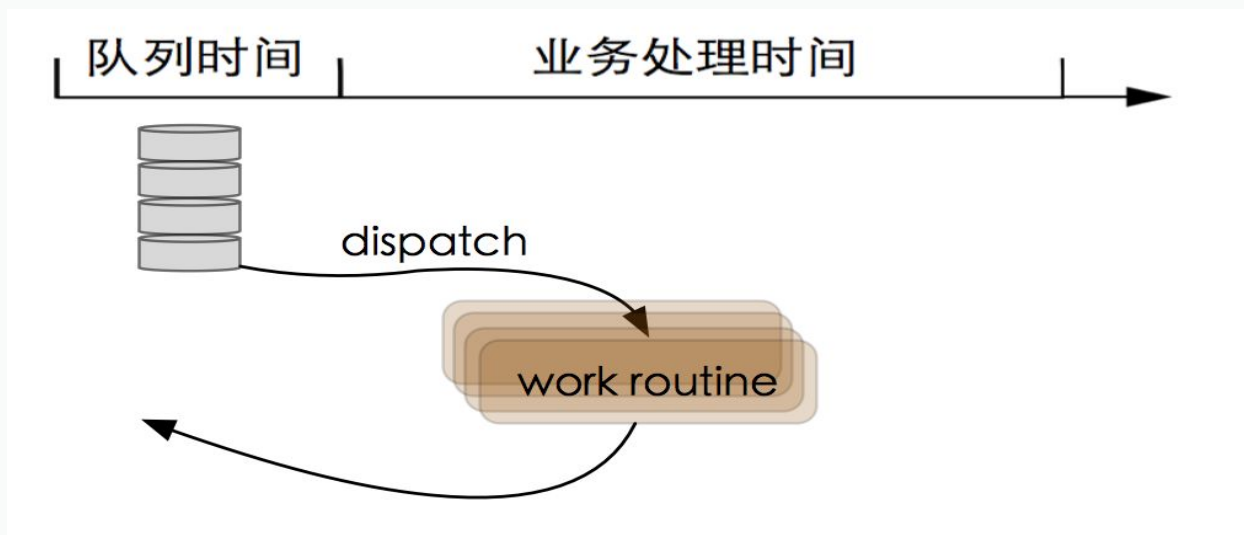
问题难点

指标的选择, 如何判定过载?
状态的判断, 基于单点/集群状态?

过载保护之指标选择

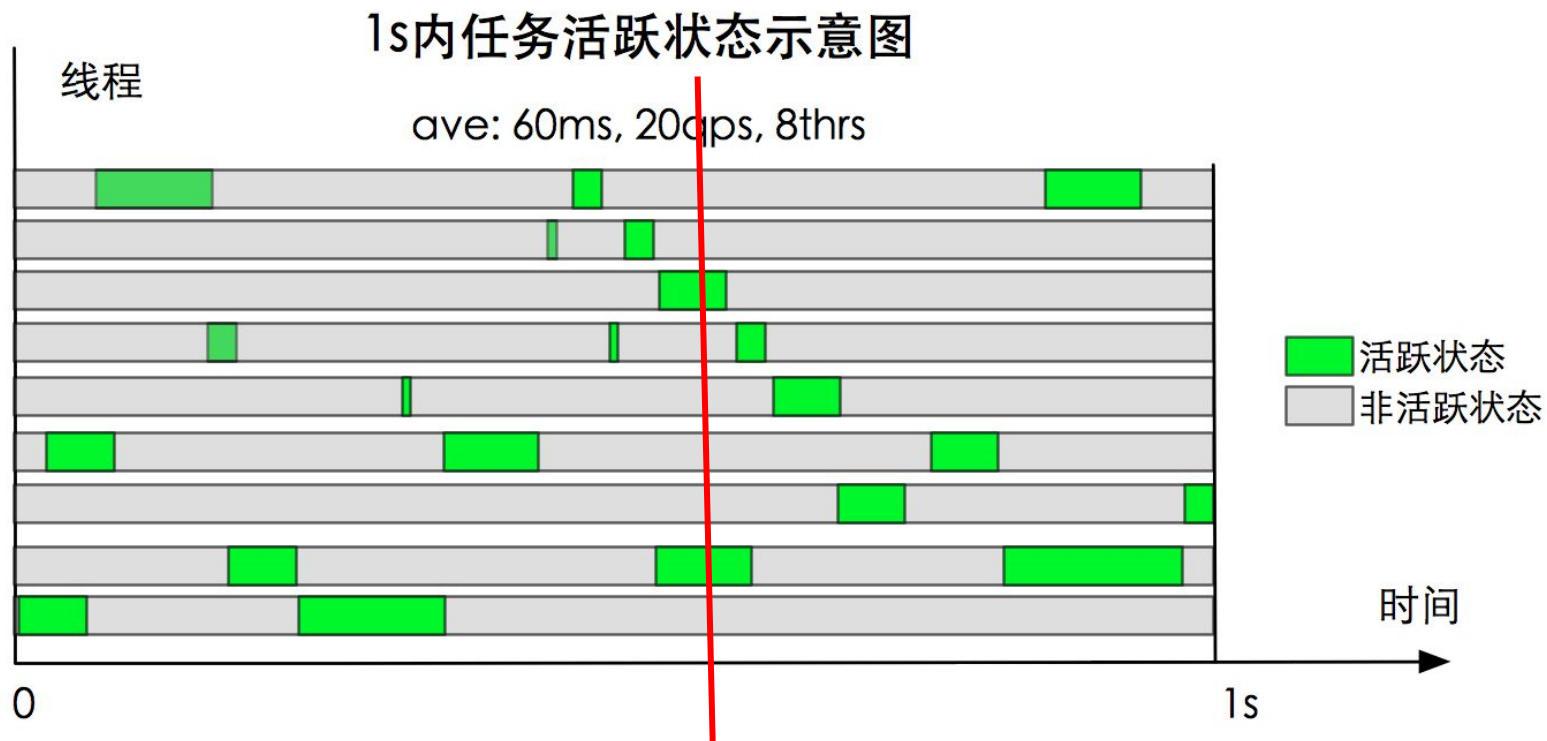
基于队列状态

往往在拥塞后作出响应, 弹性差, 甚至堵死
存在无法触发自我保护的容灾, 降级预案等的风险



基于CPU利用率?

过载保护之活跃任务



在一个时间切片下同时活跃的任务

过载保护

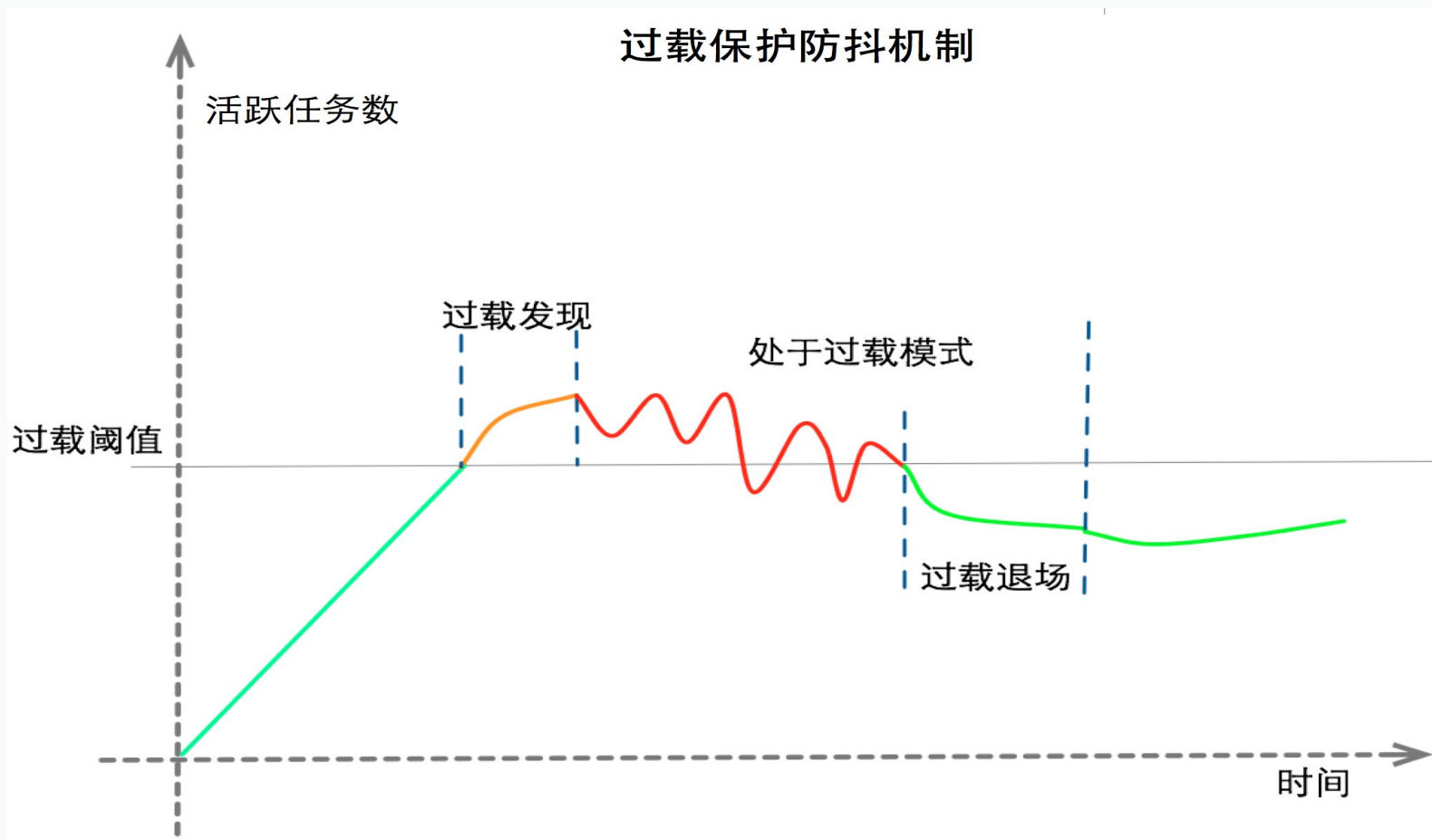
活跃任务状态

系统状态的实时负载反应
敏锐感知服务状态,非线程的CPU活跃状态

过载保护机制

过载发现, 规避队列控制的缺点
对系统异常同步检测, 配合以防抖动机制
是保容的前置条件, 保证容灾手段的绝对生效
准事前防范

过载保护



容灾Cache

一种有损的降级服务
基于用户请求实时训练而得

延迟环境

历史版本的服务/数据镜像
一个静态检索环境

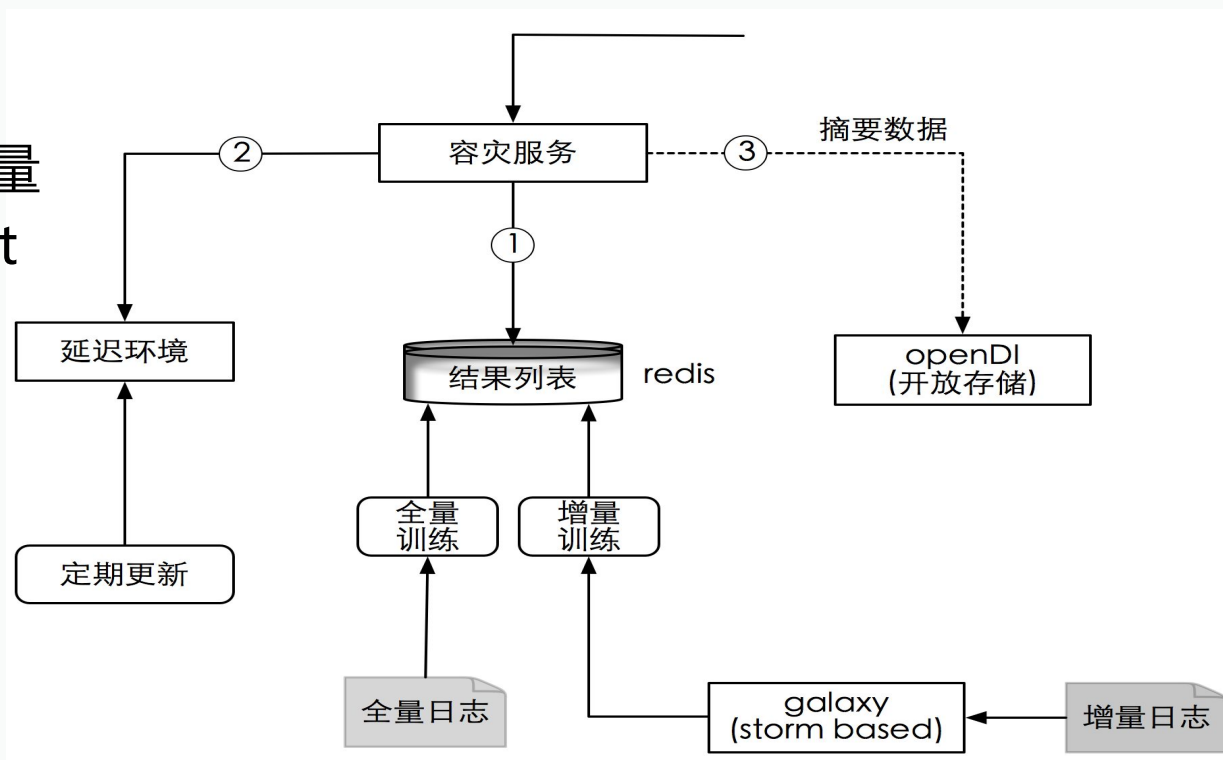
容灾服务

78%

容灾Cache承载大部分流量
展现日志训练key->IDList

22%

延迟环境相对稳定
承载Cache无结果流量



泛需求Cache训练



1. 山城辣妹子
2. 沸炉
3. 菜麻
4. 金真子烤肉
5. 正火烤肉
6. 必胜宅急送
7. 小肥羊火锅
8. 重庆老灶火锅
9. 水平有限螺蛳粉
10. 川蜀印象火锅



1. 正火烤肉
2. 山城辣妹子
3. 必胜宅急送
4. 金真子烤肉
5. 小肥羊火锅
6. 紫菜包饭先生
7. 达乐美比萨
8. 湘香湖南米粉
9. 千方老鸭汤
10. 邱老头麻辣香锅

负载均衡与调度

负载均衡

Nginx based

心跳检测

round robin

服务调度

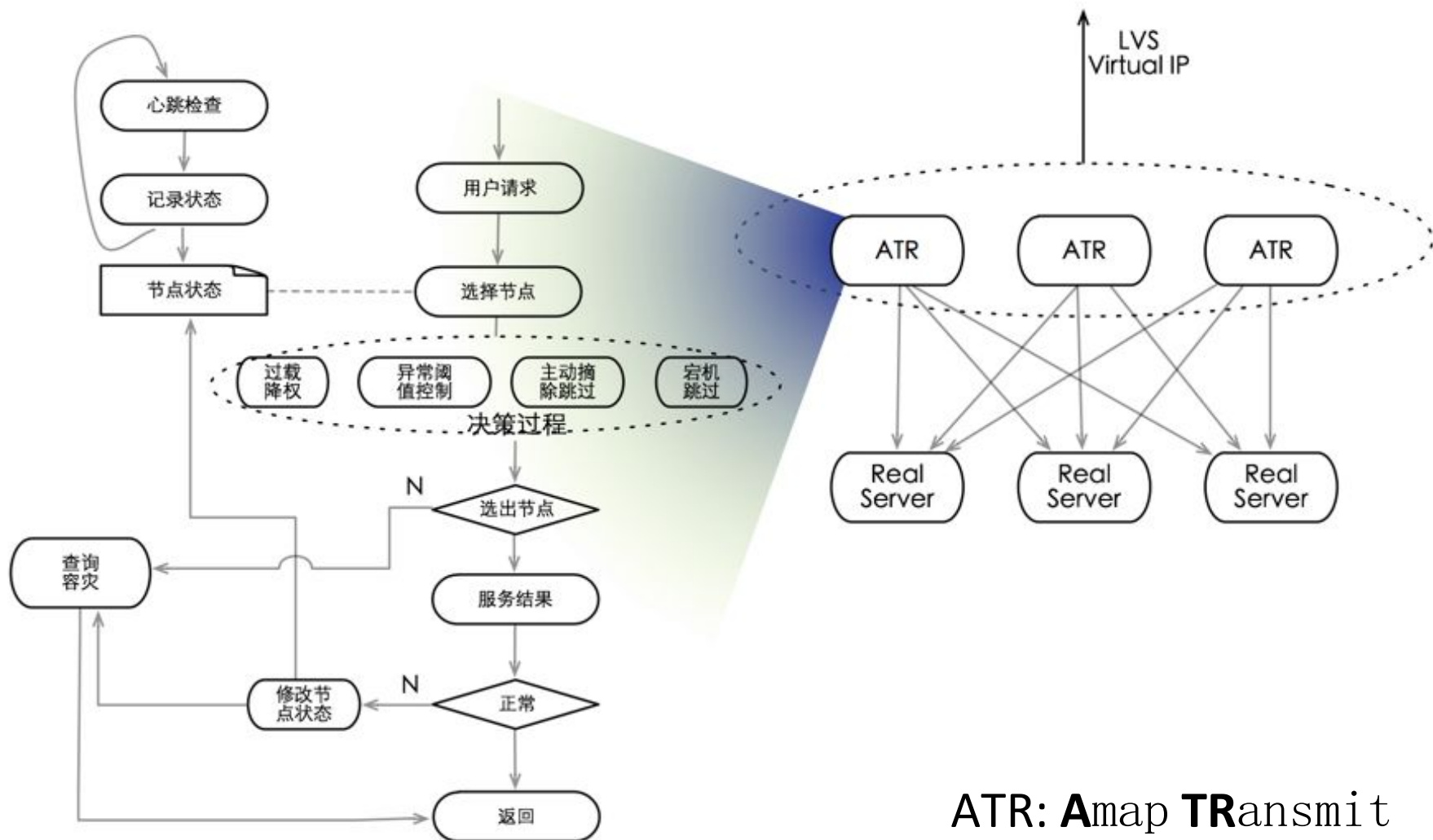
容灾调度

过载汇报

异常汇报

风险隔离

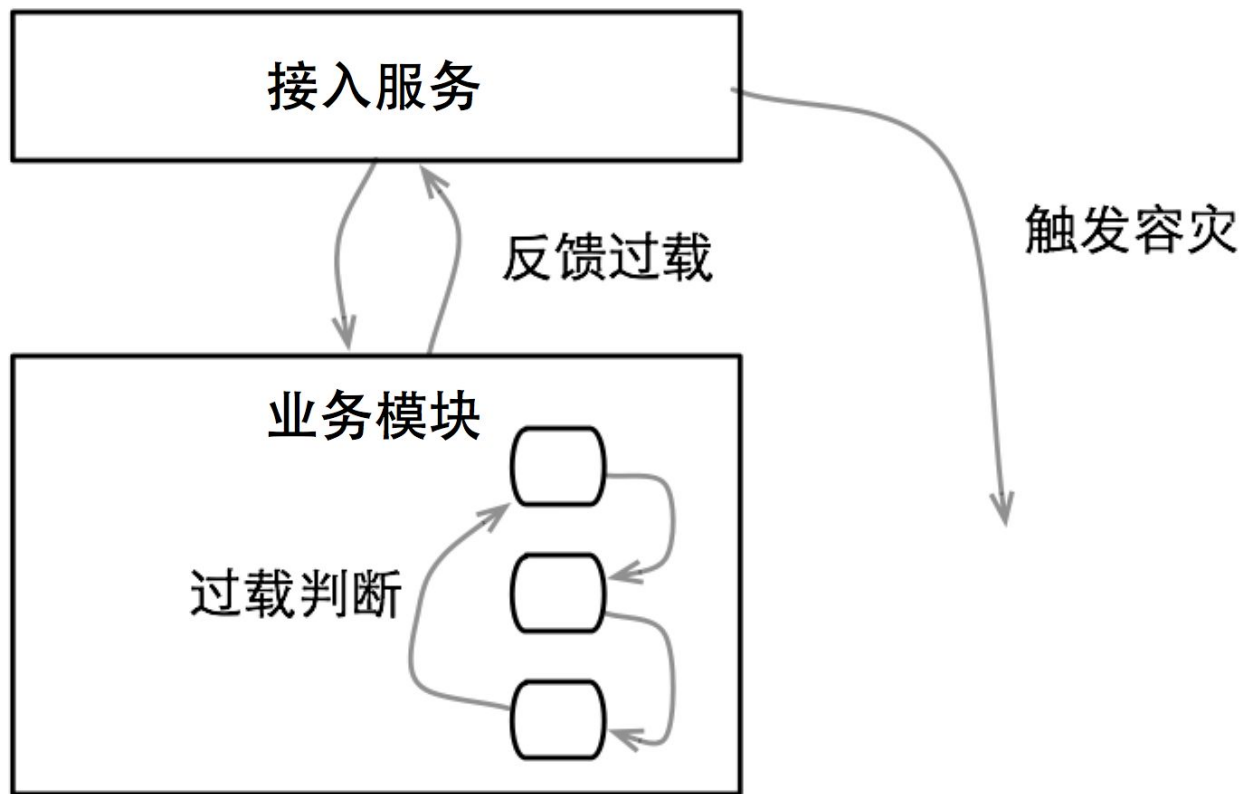
负载均衡与调度



ATR: **A**map **T**Ransmit

调度之过载触发过程

接入服务
过载反馈
保险丝熔断



过载触发过程

点到点状态判断，
而非集群状态判断

调度之健康检查过程-服务汇报链

服务状态汇报链

负载调度

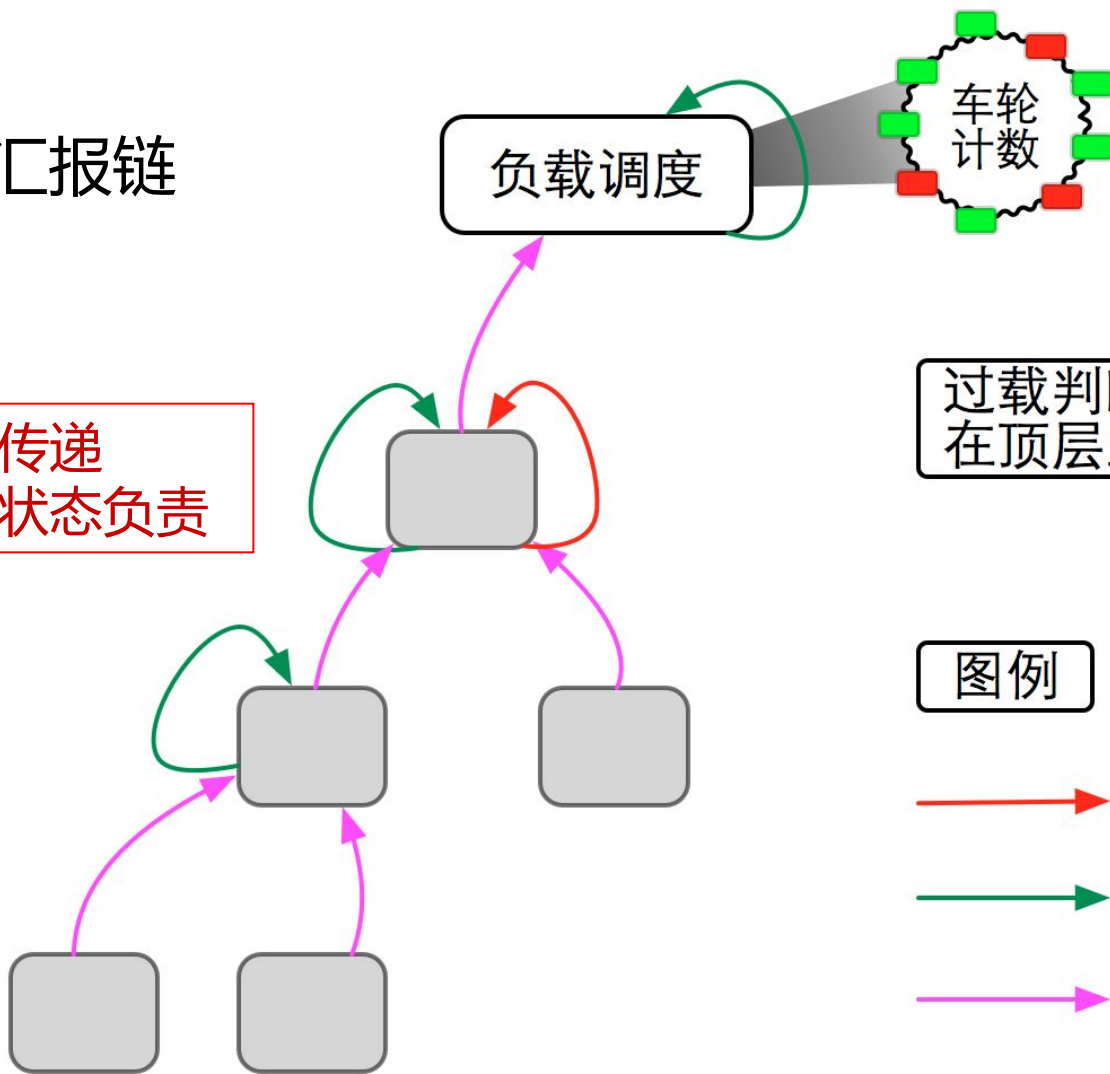
车轮
计数

状态不跨层传递
只对子节点状态负责

过载判断只发生
在顶层业务节点

图例

- 过载判断
- 状态判断/聚合
- 状态汇报



分级发布

数据

分级发布

新库测试

服务

分级发布(稳定性)

灰度发布(效果/稳定性)

II. 实时问题诊断

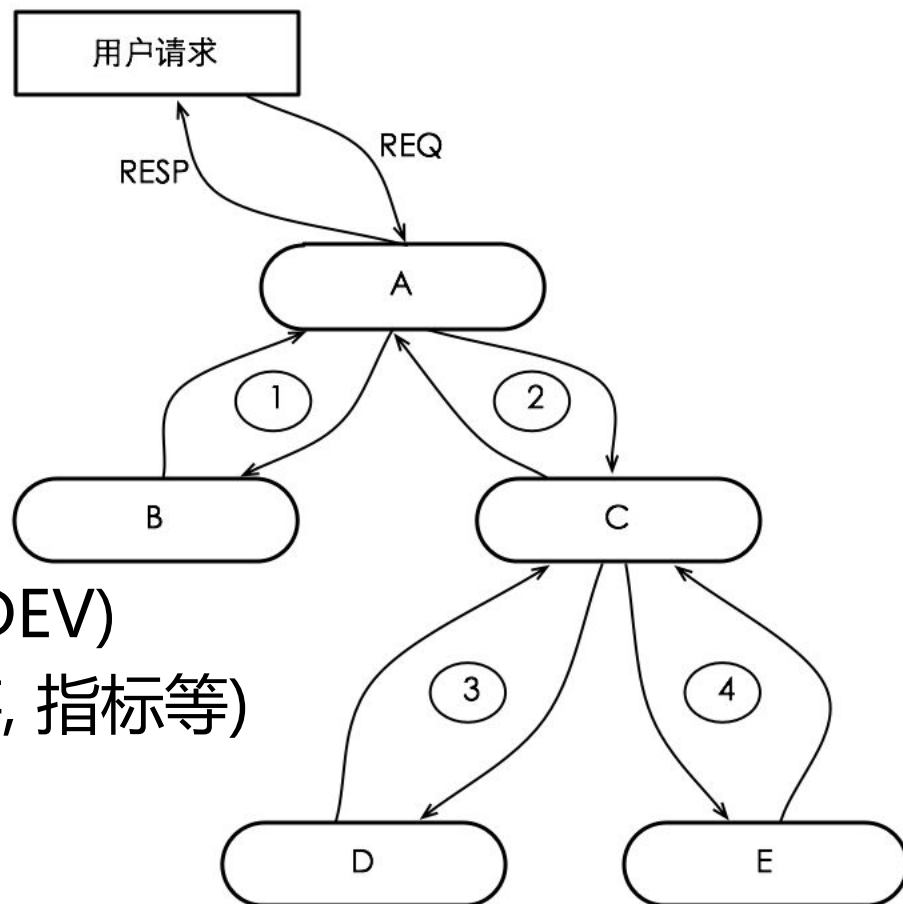
TRACE

背景

复杂系统下在线问题实时追溯
构建从系统到问题的最短路径
服务状态TimeLine

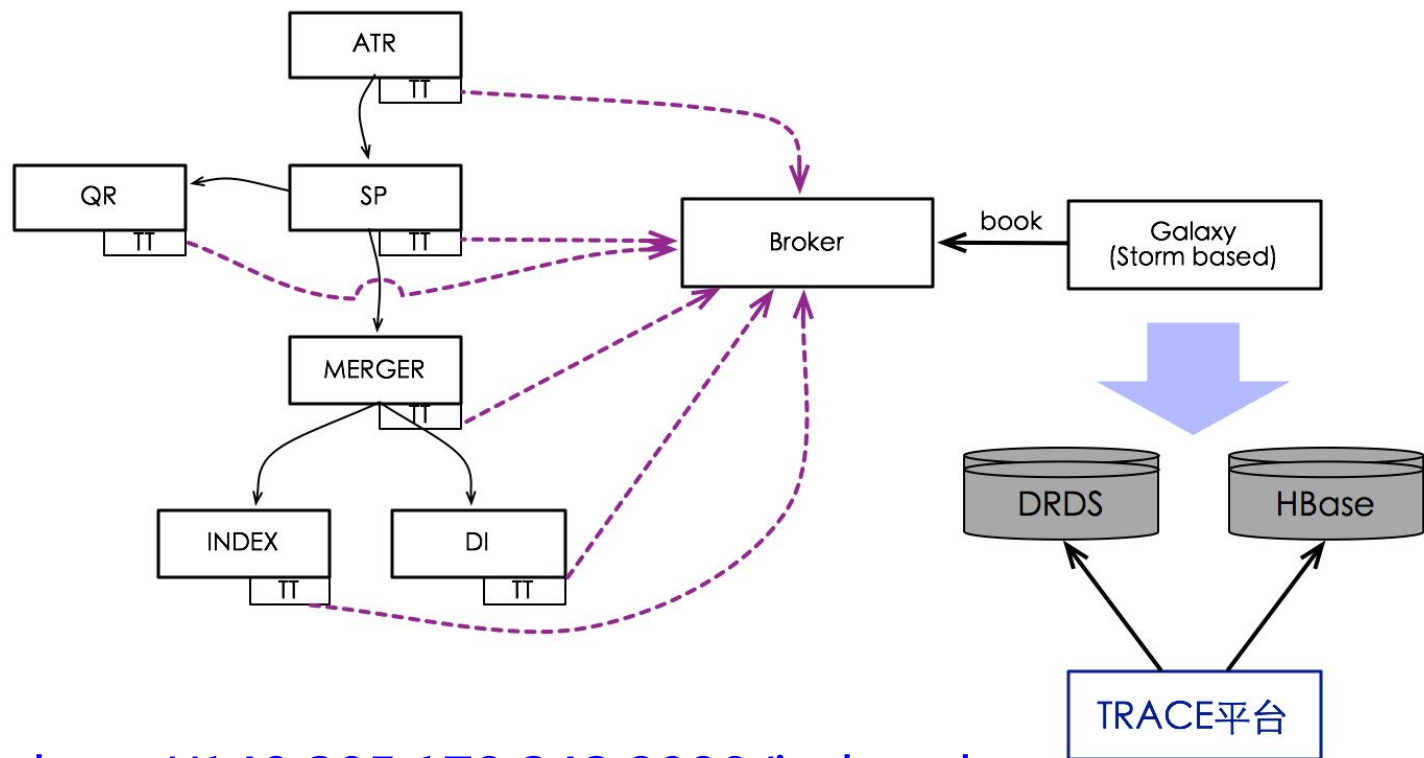
应用

服务哨兵-服务状态自我诊断(DEV)
监控追溯-波动分析(性能, 内存, 指标等)
现场镜像-结果级分析(DEV)
日志聚合-到达性分析



TRACE

Trace整体架构



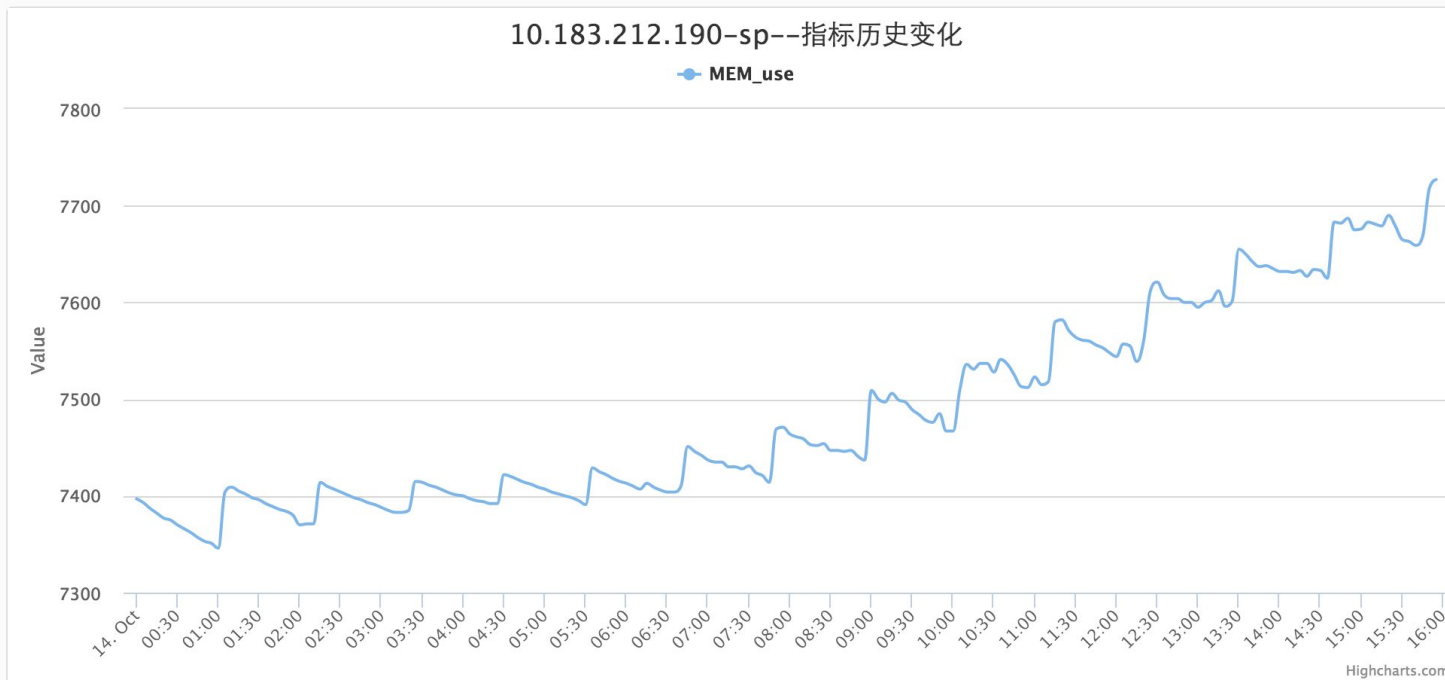
演示地址: <http://140.205.179.242:8090/index.php>

参考: [Google Dapper](#)

TRACE

指标详情

开始时间: 2015-10-14 00:00:00 结束时间: 确定



TIMEOUT_num	normal:0 mask:0 error:0 fatal:0 warn:0	2015-10-14 15:50:00	点出
WARNLOG_num	normal:6 mask:0 error:0 fatal:0 warn:0	2015-10-14 15:50:00	点击

III. 合理的业务抽象

展现业务抽象

70%

在线项目直接或间接与发版相关

<20%

20%的代码和产品迭代相关

重数据/重展现/轻策略

代码重复度高/改动琐碎/稳定性风险大

展现业务抽象



Native下复杂多变的展现样式该如何控制?

展现业务抽象-模版化

充分发挥NA/H5的优势:

NA的渲染效率

H5的变更灵活性

一个典型的结果模板

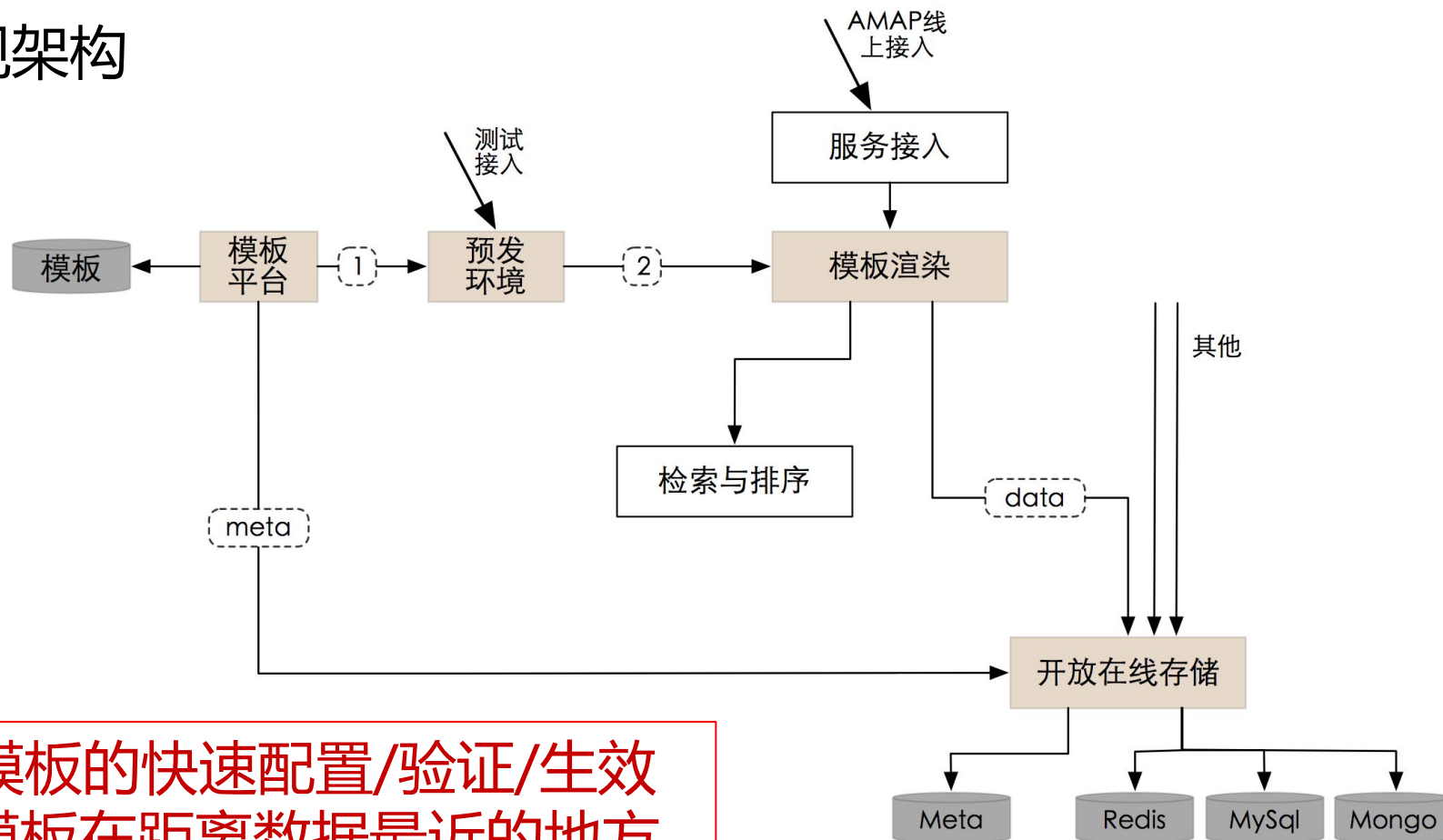
逻辑下放



```
<scenes>
  <scene id="101000,201000,100500,200500,101100,201100,100507,200507" grp_rule_name="grp_pk_grkeys">
    <domain id="1007" name="icon" max_match="3">
      <display gram="$groupbuy_flag!=0: value+=$groupbuy_flag; src+='resource://localhost/group1.png'"/>
      <display gram="$takeout_flag!=0: value+=$takeout_flag; src+='resource://localhost/takeout1.png'"/>
      <display gram="$discount_flag!=0: value+=$discount_flag; src+='resource://localhost/dis2.png'"/>
      <display gram="$quan_flag!=0: value+=$quan_flag; src+='resource://localhost/quan.png'"/>
      <display gram="$view_flag!=0: value+=$view_flag; src+='resource://localhost/view.png'"/>
      <display gram="$businesses_flag!=0: value+=$business_flag; src+='resource://localhost/bussiness.png'"/>
      <display gram="$taodiandian!=0: value+=$taodiandian_flag; src+='resource://localhost/taodiandian.png'"/>
    </domain>
    <domain id="1008" name="star" >
      <display gram="$category=10*&$src_star!=0: value=$src_star;"/>
      <display gram="$category!=100000&$groupbuy_flag!=0: value=$star;"/>
    </domain>
    <domain id="1009" name="tag" max_match="1">
      <display gram="$category=050000&$type!='': value=$type;"/>
      <display gram="$category=100000: value=$level;"/>
      <display gram="$category=100000: value=$type;"/>
      <display gram="$category=110000: value=$level;"/>
      <display gram="$category=110000: value=$type;"/>
    </domain>
  </scene>
</scenes>
```

展现业务抽象-模版化

展现架构



模板的快速配置/验证/生效
模板在距离数据最近的地方

展现业务抽象-模版化

数据特征

展现数据

- 灵活多样, 不断变化

特征数据

- 服务于rank, 相对稳定

核心思路

展现数据与特征数据分离

- 独立存储平台/数据视图
- 支持数据的快速扩展接入
- 数据视图与模板平台化对接

NA下可动态执行模板

- 列表数据/阿拉丁数据
- 服务端执行模板渲染

客户端只关注渲染，
服务端关注数据生产

Q/A

联系方式:

leywar.liang@alibaba-inc.com

105025295@qq.com

微信: (leywar)

