

大数据平台服务构建 - 之务虚心得

今天讲什么？

问问题：我是谁，我在哪，我该干什么？

讲道理：摆正价值观，全心全意为人民服务

举例子：瞎扯一下数据链路部分核心系统

晒经验：那些背锅的人，现在过得咋样？

卖东西：祖传秘方，客官要来一份不？

大数据平台基本架构组成？

无差别套用一下：H公司Hadoop发行版套件图



那么，各家公司的的大数据平台看起来都没啥本质区别？

现实问题和价值



数据平台的用户需要什么服务

你最烦的用户来找你

- 问题跟踪（来找茬）
- 技术支持（问答百事通）
- 压根不理你（你是谁？你们干啥的？）

客服+资源

你希望用户做到

- 思考各种组件优缺点，做出正确选择
- 学习各种优化的手段，别乱搞事
- 理解平台困难，讨论最佳实践，通情达理

顾问+专家

用户实际关心的

- 要什么有什么（我可以乱来）
- 稳定，高效，低成本（你别出幺蛾子）
- 快速便捷，容易理解，支持业务决策（易用，好用，有用）

管家+保姆



大数据平台的建设目标？

No, No, No!

- 谁的组件更丰富？谁跟进社区技术跟进得最快？
- 谁的团队技术能力最无敌，谁拥有最多的Committers？
- 只是手段，甚至都不一定是实现目标的最有效的手段。

Yes, Yes, Yes!

- 平台内部组件的横向联通能力
- 业务纵向贯穿打通上下游链路的能力
- 为用户解决了哪些问题，扫除了哪些障碍
- 提升了多少效率，产生了哪些增值收益

赋能+伙伴

四个现代化
指导方针



把青春献给四个现代化

组件工具化

就是写写集群日常维护脚本这点事么

- 所有自建的数据平台，都是从集群的运维管理开始
- 这件事情做得多了，你就会想要提高效率，最简单的，就是把一些常用的操作用脚本维护起来，沉淀经验，避免误操作

工具化的本质目标

- 降低学习成本，提高工作效率，减少犯错概率。
- 对组件细节的封装和简化，不仅仅从平台组件维护的角度，更是从用户应用开发的角度来说。

工具平台化

什么是平台化

- 将各种组件，工具，开发流程整合到一起，统一管理，提供成体系的开发运维管理途径

实践起来有什么障碍？

- 信任危机：没有收益，没有保障，自己玩呗
- 团队定位：业务方怎么用好集群构建业务，那是他们的问题，我是技术专家不是保姆
- 系统架构能力：对业务系统全貌不了解，缺乏整体规划的能力

平台服务化



和平台化 的区别

重要的是理念，服务是围绕着客户体验为中心展开的，
它的重点不在平台自身，在于用户

用户满意才是衡量服务水平的唯一标准

平台产品化

提供工具

- 码农的自我修养和追求

提供平台

- 领取微薄薪水的义务

提供服务

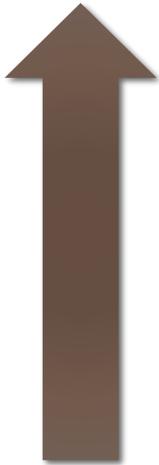
- 博取业务方大爷怜悯的方式

但是上面这些都没用！领导最终只关心你的价值产出：-)

底层团队不好干啊。。。

- 做得越多，错得越多，服务越好，负担越重，这种困境，只有依托良好的产品形态来换取可衡量的价值产出才能打破
- 产品化不是一个一厢情愿，埋头努力就能解决的问题，是对现实中各种问题的评估，妥协和取舍

构建数据平台服务的两条路径



针对具体的业务场景，一站到底式的支持

- 产品逻辑可以高度定制，最大程度匹配业务需求
- 流程复杂度相对可控，可以屏蔽与具体业务无关的内容，容易保障易用性
- 系统架构成型快，演进负担小
- 系统专用性较强，可拓展性差，可能存在重复建设



针对通用的功能需求，构建独立的系统组件

- 模块化建设，可拓展性较好
- 减少各业务系统的重复建设
- 架构方案有机会做到更加深入，完善
- 系统架构成型较慢，迭代演进负担较大
- 场景定制程度较低，易用性较难保障

LOOK INSIDE

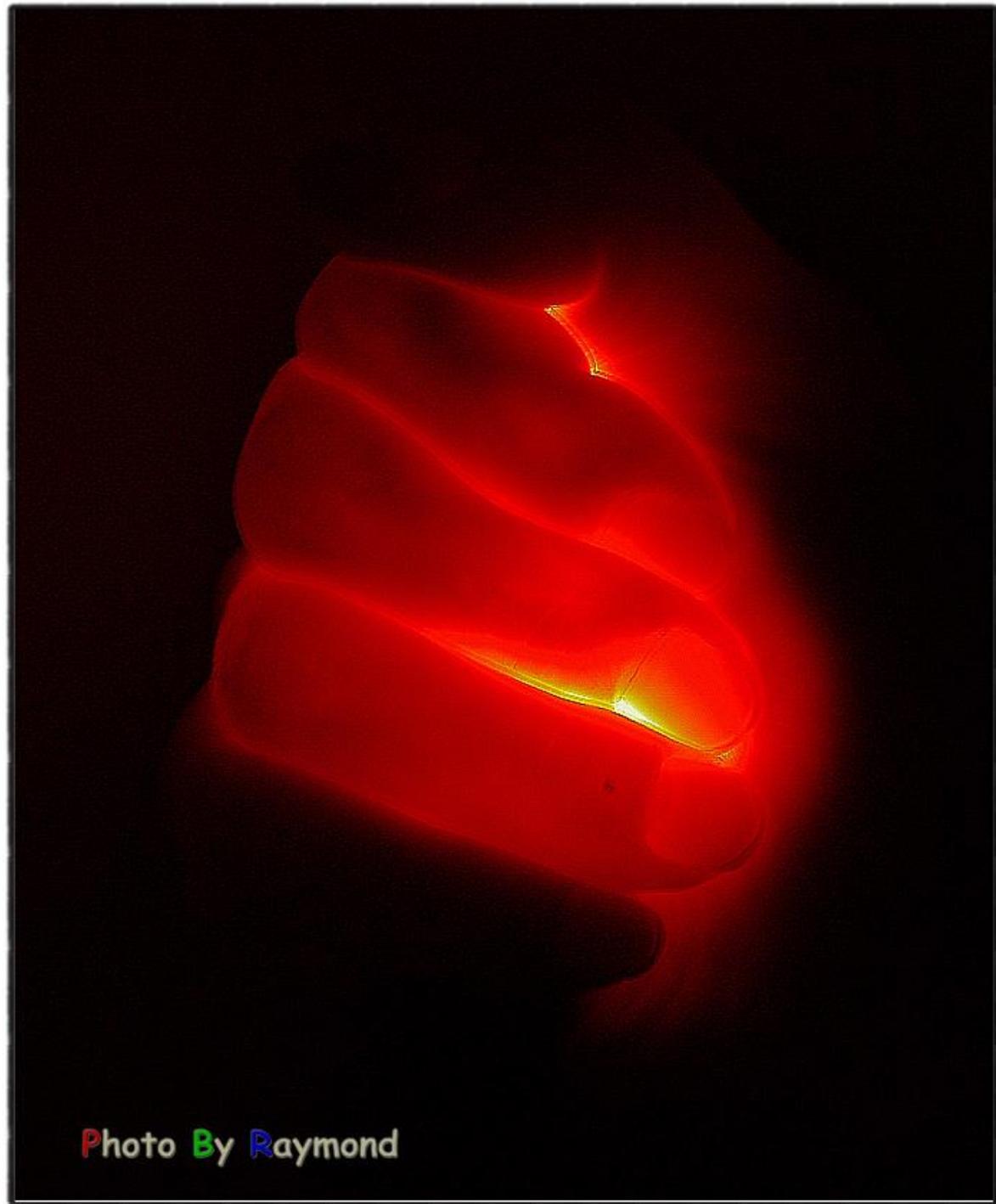


Photo By Raymond

数据采集链路

- 实际实现，可繁可简
 - scp, ftp, script, python/java
 - Flume, kafka
 - datax, sqoop
 - ELK
- 但是，如果你希望构建一个稳定可靠，用户友好的系统？
 - 数据质量问题
 - 传输采集问题
 - 分发传输问题
 - 系统维护代价问题
 - 业务价值问题
 - 易用性问题

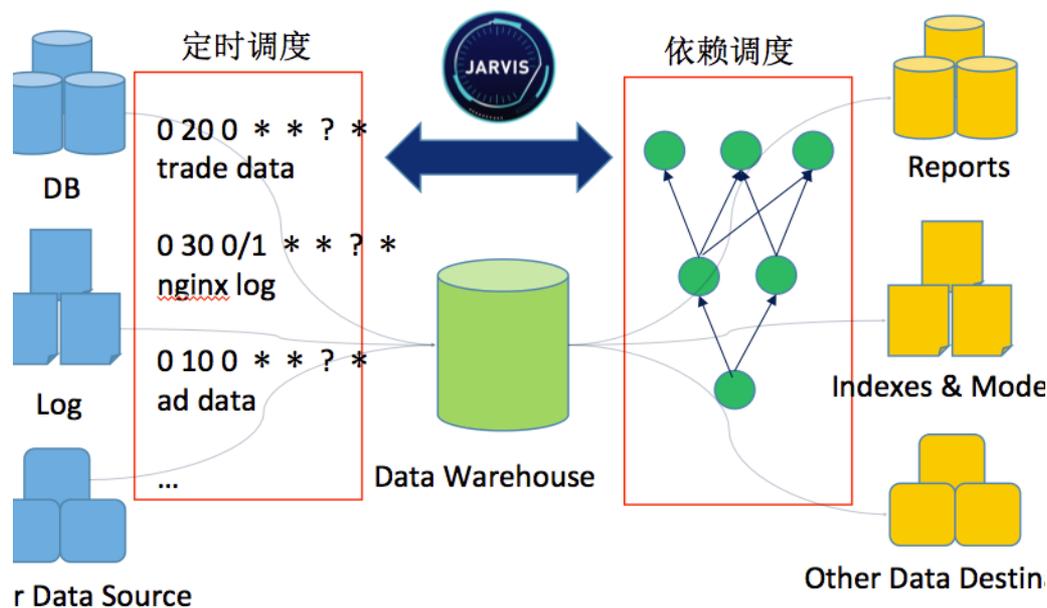
用户行为埋点和分析

- 前端埋点系统
 - Tracking方案?
 - 手动, 自动?
 - 规划, 管理, 测试, 跟踪?
- 数据存储, 加工和处理
 - 版本? 渠道? 活动?
 - 安卓/iOS/H5/小程序?
 - 离线, 实时?
- 数据展现
 - 日志? 报表? 漏斗? 链路?
- 产品形态
 - Google analytics
 - Growing IO / Talking Data

这些工作
用户关心哪部分?

作业调度系统

- 大数据平台运转的枢纽组件
 - 看似简单的一个系统
 - 有大量的开源实现，为什么还要自己做？
 - 真正做（用）得好的公司也没有几家，why？



作业调度系统实现

- Oozie
- Azkaban
- Chronos
- Zeus
- Lhotse
- SchedulerX
- Elastic-job
- Saturn
- Jarvis

问题

无效作业，耗费资源

无序使用，滥用集群

业务自身问题？

任务性能差，未优化

链路长，随机因素大

- 关键路径执行情况，报警策略？
- 任务变更操作纪录？异常行为监测？
- 配额负载分析？流量分配，优先级管控？
- 错误原因分析，任务性能诊断，数据质量监测？
- 这么多东西，谁来关注？怎么关注？如何关注得过来？

大数据集成开发环境

- 大数据集成开发环境所提供的主要服务，在用户看来，当然就是让他能够在写代码，然后运行
 - 所以不就是一个web IDE开发界面么，这有何难？
 - 但是还是有很多公司连这个都没有（做不到）
 - 开源：HUE，公有云：数加，DataWorks；TDF，TBDS
- 所提供的服务，需要贯穿大数据处理链路的全过程
 - 包括数据的采集，计算，管理，查询，展示等环节
 - 代码编辑器仅仅是支持其中部分环节所需要的服务之一
 - 串联各个系统，为用户提供一站式服务
 - 水平的高低，体现在各种组件融合的顺畅程度
 - 理想状态：用户对底层系统的完全无感知



开发平台建设/管理维度

脚本

- 脚本 V.S. 业务逻辑
- 使能增值收益
- 质量, 规范, 血缘等

任务

- 脚本 V.S. 任务
- 生命周期, 调度管理
- 完善度, 自动化程度

数据

- 数据 V.S. 元数据
- 高效的挖掘和使用数据

用户

- 统一登陆认证
- 平台/组件账号映射
- 多租户管理

权限

- 认证 V.S. 授权
- 方案众多环节各异
- 安全 V.S. 便利

流程

- 自觉 V.S. 规范
- 划定边界, 隔离风险
- 标准化, 提高效率

组件

- IaaS v.s. SaaS
- SDK v.s. Platform

集群

- 部署效率, 安全稳定
- 屏蔽细节, 监管能力

背锅心得



说好不打脸！

说好不打脸！

君子喻于义：“真心”服务用户

- 别让用户思考，别让用户有挫败感！
 - 不要对用户做任何知识假定
 - 把饭喂到用户嘴里
- 提供差异化，阶梯式产品服务
 - 面对现实，没有万能的产品
- 构建反馈式服务
 - 比起响应迟钝的系统，更让人崩溃的，是压根没有响应的系统
- 确保你的产品，可持续改进
 - 光有决心和能力，往往是不够的，你需要反馈和数据

小人喻于利：不怕谈论利益（价值）

- 主动思考，勇于放弃
 - 我们经常讨论的是，“是什么，怎么做，能不能做”
 - 很少考虑“为什么要做？做点别是不是收益更高？”
- 没有经验，找不到价值点，不知道如何评估收益怎么办？
 - 问题驱动（不是 bug 驱动）
- 必要的约定
 - 权益和责任的对等
 - 服务好不等同于单方面承诺，需要共同保障
- 开放的心态

取舍平衡

- 求逼格还是求实效
 - 不怕low，就怕不知道
 - LOW甚至可以是一种主动选择
 - 方向比逼格更重要
- 求发展还是求稳定
 - 问题导向
 - 面子怎么办？
- 技术驱动还是业务驱动？
 - 劳逸结合
 - 因人施教
 - 换位思考



NEVER STOP

大数据相关建设-经验对比&观察

- 工作过的公司
 - Intel
 - 蘑菇街 V.S. 美丽说
 - 51信用卡
- PK过（中）的平台/产品
 - 腾讯云
 - Growing IO
 - Tableau/永洪
- 交流过/了解过的公司/平台

关于体系，产品，
路径等观察

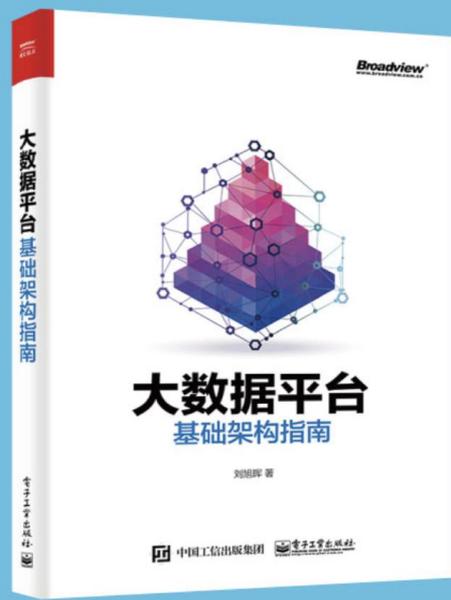
个人观点，不便成文
口述，不留证据 😊

王婆卖瓜

• 《大数据平台基础架构指南》

• <https://item.jd.com/12385129.html>

美丽联合集团|51信用卡两大CTO作序
及爱奇艺|唯品会|网易|携程|英特尔|
bilibili|PayPal|宅米网|Hortonworks
集体力荐



大数据平台 基础架构指南

本书不仅提供了互联网大数据平台建设的指导方针，还有针对关键组件的可落地设计思路，服务意识和产品思维的理念更贯穿全书，真正做到“授人予渔”。

刘俊晖 爱奇艺智能平台部高级技术总监

读这本书有种酣畅淋漓的快感，因为把很多我想说但说不清楚的东西以优美的文笔——呈现。对于有志于大数据平台开发的同学来说，这本书会是指路明灯，可以极大提升你的视野。

姜伟华 唯品会大数据平台负责人

本书从大数据平台全局视角切入，使得我们能够了解一个真正产品化的大数据平台的技术内幕，内容来自作者多年大数据平台实践经验总结，值得一读。

余利华 网易数据科学中心总监

大数据平台建设不等同于一堆开源工具的拼凑，需要很多方面的经验和知识。这本书可以给你指明前进的方向，让你少走弯路，适合广大大数据平台的从业者。

张翼 携程大数据平台技术总监

作者从产品服务的角度，总结了蘑菇街在大数据平台演化过程中的各种技术和非技术经验，对于大数据平台和应用开发人员均有很大的借鉴意义。

程浩 英特尔大数据技术经理

本书作者从整体方案的角度，深入阐述了如何建设一个可持续、可落地的大数据平台，更提炼出四个现代化的理念，非常值得学习和借鉴。

薛赵明 bilibili数据平台技术经理

相较于市场上许多源码解读书籍，本书作者分享的更多是设计的根源和思路。对于许多处在选型、架构设计阶段的技术人员，也更加具有启迪作用。

黄洁 Paypal企业数据服务部高级经理

如何将各种开源大数据技术整合改造，集成应用到自己公司的生产环境中，其中各种权衡利弊，只有亲身经历者才能有所体会。本书正是这样一本实践之作，为读者呈现一个与众不同的、大数据技术应用的落地视角。

李智慧 《大型网站技术架构：核心原理与案例分析》作者

本书从企业应用着手，详细探究了企业应用大数据之道。作为一线团队的实践总结，具有极强的指导意义。

邵赛赛 Hortonworks 资深技术专家

多卖两句

- 编辑说，销量感人呀。。。能不能推一把好快点再版呢？
 - 出书前就有心理准备，这不是一本入门扫盲的书，销量一定不行

编辑推荐

适读人群：大数据平台相关开发人员以及对底层平台感兴趣的数据业务开发同学。

- ✓ 集蘑菇街数据平台资深架构师数载功力之大成，基于大数据服务平台整体产品规划和架构设计真实案例。
- ✓ 独辟蹊径，着眼点凌驾于技术之上，从业务与产品宏观视角入手，提供平台服务构建整体思路与解决方案。
- ✓ 全面覆盖核心组件：工作流调度系统|集成开发/测试环境|元数据管理系统|交换/可视化/质量管理服务等。
- ✓ 以人为本，关注大数据从业者岗位要求、能力建设、职业发展与规划，既可开阔视野，更能避免弯路。

- 不能入门，没有干货，那有什么？
 - 大数据平台整体产品规划和真实的实践经验，
 - 相信是你在其它途径基本都找不到的内容，官网有的，别人写过的，时效性强的，容易过时的，坚决不写。
 - 着急的编辑比我会说

QUESTION ?

