

大家好！前面几个大佬都以产品角度来讲运维，我今天以干货的方式跟大家分享一下我在运维方面的经验。首先做一个自我介绍，我是广发银行数据中心的彭华盛，主要是交易渠道应用运维团队及自动化方面的负责人。

今天给大家分享的议题是运维一体化中的平台一体化，从标题看可以看到两个一体化，分别是运维一体化与平台一体化，其中运维一体化是我们数据中心在运维方面的运营体系，它包括三方面：人员组织一体化、流程一体化、平台一体化。这里面的平台一体化偏向于运维工具与自动化方面的建设，是今天分享的重点。

### 一、转型：

和目前大部从运维团队一样，我们的运维团队也打着“救火”、“背锅”、“低价值”等标签，团队的特点归纳下有四个特点：1、被动救火式，以被动保障业务系统运行，日常计划性工作容易被打断、搁置；2、问题驱动式，以系统可用性、可靠性、业务请求等问题驱动运维工作；3、操作运维，重复性、操作类点主要工作量的运维模式；4、经验式运维，由人工经验驱动的运维模式，尤其是一些经验丰富的老员工的离职在短期内会对运维质量带来一定的冲击。

针对上面四个特点我们提出了四个转型，分别是被动救火式向主动精细化转型，主动分析，主动优化，驱动开发，促进 DEVOPS 的落地；从问题驱动向价值驱动转型，以业务体验、服务满意度、促进业务更好发展；从操作运维向运维开发转型，通过为运维人员提供运维开发平台，降低运维开发门槛，快速落地一些紧迫的运维工具，降低操作性、重复性的运维工作；最后一个转型是从依靠经验向智能化驱动运维转型，结合数据分析、知识库、机器学习技术促进运维智能化。

在现有人力维持不变，运维质量要求还要提升的背景下，要实现上述四个转型目标，我们首先要解放生产力，解决生产力当前最主要的手段是自动化，所以下面再说说我们自动化方面遇到的 3 个困难：

-如何更好：虽然运维体系比较完整，但工具主要以商业软件为主，以烟囱式建设，信息无法互联互通，无法实现  $1+1>2$  的效果；

-如何更快：运维工具拿来较多，自主研发少，无法快速满足自身运维需求，还是存在较多人肉运维的工作；

-如何更重要：运维职业危机，传统的运维操作为主的方式注定会成为历史，取而代之的是精细化运营，如何既解决职业危机，又能创造更大的价值，是难题；

针对上面提到的四个转型目标，以及自动化目前遇到的困难，我们制定了相关的技术及管理的改进方案：运维一体化。

## 二、一体化

在讲一体化思路前，我先讲讲促进一体化思路形成的一些思路来源：

- 《架构即未来》这本书里提到的组织、流程、架构的三位一体和我们运维一体化很吻合；
- 腾讯蓝鲸、云霁科技在运维自动化方面的整体解决方案引导着我制定平台一体化的方案；
- google sre 高逼格的运维运营模式，促进我在运维开发方面的解决思路形成；

如上图所示，我们的运维一体化的核心是组织、流程、工具三位一体，具体来讲是以 CMDB 为基础，结合运维统一门户、运维分析平台、云平台、监控平台、流程平台、操作平台、审计归档平台，构建运维工具一体化（即平台一体化），并在工具一体化的同时，结合流程一体化，最终构建组织、流程、工具三位一体的运营一体化的运营体系。运维一体化以运维向主动精细化运维、价值驱动、运维开发、智能化转型为目标，为实现上述目标首要的工作是需要解放运维生产力，通过“监管控”运维自动化提高工作效率，通过自主的开发平台实现敏捷的开发能力，通过运维分析实现运维智能化，辅助运维决策。

再重点讲讲今天分享的重点：平台一体化，平台一体化的思路是：6 平台 + 1 门户 + 4 原则。

-6 平台是：云平台、监控平台、归档审计平台、流程平台、操作平台、分析平台这 6 个平台分别对标我们身体的骨、眼、神经系统、循环系统、手、脑。其中监控平台、归档审计平台、流程平台、操作平台又组成了我们常规讲到的自动化中的“监、管、控”。这 6 个平台中每一个平台又组成相关技术体系，具体每个平台包括的体系内容及技术架构这里因时间问题不做深入解释。

-1 门户是：统一门户是运维可视化的关键，它集成了工具的可视化层，提供多维用户、多种展示形式、以运维场景驱动建设等作用。

-4 原则是：自主化，工具化、服务化、可视化，这 4 个原则后面会进一步介绍。

通过归纳这个 6 平台+1 门户+4 原则的平台一体化思路，并将这个思路推广到数据中心工具建设团队中，使我们能否快速达成共识，提高协作效率。

下面这张 PPT 是对平台一体化的进一步细化，概括了目前平台一体化中的主要内容，各位有兴趣可以花时间看看。

### 三、建设原则

平台一体化规划重点解决各技术平台间的信息互联互通、统一展现和紧密联动，对于各个平台工具有着几个原则“服务化、可视化、自主化、工具化”，即：

- 自主化：构建运维开发平台降低运维工具开发门槛，促进平台工具开发更加自主可控，更加敏捷；
- 工具化：在运维团队中建设工具建设文化，促进运维开发文化建设；
- 服务化：拒绝推倒重建，整合好存量自动化工具，引入新的工具，实现工具间的互联互通，数据共享；
- 可视化：通过更加统一、清晰的可视化建设促进平台的效益的产生；

下面对这 4 个原则，结合我们目前工具建设阶段性的成效来做进一步的分享。

#### 1、原则 1：自主化

自主化包括开发能力自主化与架构自主化。

1) 开发能力的自主化，我们主要是通过建立运维开发平台，这个开发平台具有所见即所得的开发能力，PPT 这几张工具界面是我们目前实现的脚本与可视化开发工具，它们具备这些特点：

-所见即所得的脚本开发能力：

脚本开发环境（含环境、测试、部署，以及日志、权限、代码管理、脚本执行统计等一篮子解决方案）；

标准组件化脚本可供调用，运维人员在开发过程中可以不写具体的脚本，采用组件的方式在可视化界面上组合多个现成的脚本为一个脚本。

-所拖即所得的可视化开发能力：

实现运维工具的可视化展示（提供 HTML5 风格可视化运维工具控件的拖拉生成统一风格的代码，运维开发人员无需关注 HTML 标签代码与 CS 风格 S）；

实现可视化工具流程的配置能力，即工具上事件所需要的页面流；

设计标准组件提高可视化开发效率；

-运维一体化下的开发能力：

标准化脚本的服务化能力；

服务化脚本的接口注册；

2) 架构自主化，我们在平台建设过程中引入了互联网分布式的架构，开源的技术架构能让我们对技术架构更有可控性。以集中监控系统为例，我们对原有的应用主备、数据库主备的架构改造为分布式架构，现在这个系统采用 WEB、应用分布式，通过 MYCAT 分布式数据库中间件实现数据库分布式，采用 ZK 实现数据库主节点的选取，通过 MYSQL 实现读写分离，目前我们的数据库由 17 台 MYSQL 组成，数据存储由原来了一个月，到现在的一年以上，支持多指标、多形式、多并发的监控、分析的运维场景需要。

整体的架构参见 PPT 这张图。

我们的平台一体化对于工具及技术主要以开源、国产为主，以下这张 PPT 是截止目前我们平台的主要技术栈。

## 2、原则 2：工具化

我们将工具分为重量型工具和轻量型工具，大致以这个思路区分：

-重量型工具：包括监控类系统（集中监控、性能监控、基础监控、网络安全监控等）、自动化部署、日志系统、业务批次调度等等这些需要开发工作量比较大的系统；

-轻型工具：包括应用服务启停工具、数据维护工具、数据查询工具、业务运营活动实时报表等和运维日常操作工作结合比较紧密且相对简单的工具；

对于这两类工具，我们综合人员能力、投入产出等客观因素求个平衡点，其中重量型工具以引入成熟系统进行二次开发为主，轻量型工具逐步以自主开发工具为主。

下面针对上面两类工具分别举例。

首先是重量型工具，以集中监控为例，监控架构见这张 PPT，集中监控的建设思路主要是以“不漏报、不误报”加强“监”的能力，通过监控分析、学习能力补充自动化“控”的能力，实现智能化的主动预测、故障自愈、无人值守。目前我们的监控体系已覆盖从基础设施、服务器存储、系统软件（含虚拟化、容器、系统软件等）、应用可用性、客户体验五个层次的对象，这些监控对象由不同的监控工具实现监控数据的采集与事件分析。在监控工具之上，由集中监控实现监控数据整合、事件整合、子系统接入、统一可视化、数据源采集、智能学习、智能基线、事件协同处理、事件联动分析、新技术平台监控等平台能力。后续需要在平台能力之上建立智能学习型监控，实现主动预测故障、故障自愈、无人值守。

下面图中我取了几张我们监控有特色的几个功能，分别是：

- 集中的可视化，具备多用户视角、多系统整合展示、多形式展示（WEB 端、大屏，以及目前正在开发的手持端监控）
- 体系化整合，整合存量系统（比如基础监控、性能监控、应用监控等）、整合数据、整合事件等。
- 将监控能力下探到分行，可以实现分行终端的的监控管理，模拟柜面终端的操作回放。
- 深度关联分析，相似事件统一汇总，比如我们在 CMDB 的基础之上，建设应用配置库，应用配置库除了应用服务、版本、程序等 CI 项外，还将应用的纵向与横向关系通过可视化拖拉的方式实现。
- 利用好监控数据，比如将监控数据用于一键巡检，业务运营活动自定义报表等。
- 其它，比如这个事件丰富，我们将事件的信息展示，还将涉及的系统配置信息、关联事件、事件具体数据、事件应急、工单情况、涉及 OS 的资源、性能、事件处理情况、事件应急工具等信息集中在一个视图，以促进事件的快速定位与应急恢复。

讲完重量型工具，现在举例讲讲轻型工具，下面这张应用工厂的界面可以很好的解释我们的建设思路。参考 APPSTORE，在团队中建立工具开发文化，管理员可以开发工具，并发布到应用工厂并供其它管理员使用，其它管理员可以对工具提建议或打分，这些打分可以作为该工具欢迎度进行奖励。

下面这几张图我以轻量型工具中的服务启停为例，这个工具可以满足关机维护、应用投产、故障应急等场景，支持单个或多个服务的环境保存、进程常规情况下的启停、异常情况下启停、启停后多重形式的检查方法。

这些小工具的建设一方面提高了团队运维工作效率与标准化的落实，另一方面也有助于一些有想法、有能力的同事增加工作成就感。

### 3、原则 3：服务化

服务化是为了实现工具间通讯的互联互通，服务化一方面要求各工具对外提供 API 接口；另一方面是通过统一开发一个服务集成模块实现工具监控 API 接口的注册、发现、鉴权。

这个服务集成有 2 个主要功能：

- 服务通讯总线，相当于一个轻量型的 ESB，工具间的通讯需要经过服务集成模块；

- 服务注册与发现，提供可视化的界面为各个工具进行接口注册，为调用起提供接口入参说明、调用说明等功能；

这个服务集成还提供多种通讯方式，RPC、MQ 队列等，可根据不同的通讯需要进行调用。

### 4、原则 4：可视化

可视化方面我们主要以统一门户为载体，在技术上我们选择了以 H5 加 CSS3（为移动端化作准备），提供以下 3 个特性：

- 提供指定服务：统一门户提供所有工具菜单、生成访问用 TOKEN；

- 页面适应性改造：统一风格，并分步对存量工具进行风格改造

- 访问适应急改造：快速跳转、多标签单点登录、浏览器兼容；

在功能上，我们提供多种用户视角，专业团队视图、管理视图、业务视图，下在这张界面是我们统一门户待办的视图，它将不同用户角色所关心的指标集中在这个视图，作为用户控制台。

最后，接下来我们还会继续做好自动化，解决我们工作上的痛点，解放生产力。后续再结合大数据去放眼智能运维。谢谢大家！