

## 09 | 如何在CMDB中落地应用的概念？

2018-01-07 赵成

赵成的运维体系管理课

[进入课程 >](#)



讲述：黄洲君

时长 11:11 大小 5.13M



我们前面讲了应用是整个微服务架构体系下运维的核心，而 CMDB 又是整个运维平台的基石。今天我就讲讲在 CMDB 中如何落地应用这个核心概念，以及如何建立应用集群分组的思路。

### 如何有效组织和管理应用

微服务架构下会有很多应用产生出来，少则十几、几十个，多则上百甚至上千个。这时我们面临的第一个问题就是如何有效地组织和管理这些应用，而不是让它们在各处散乱，命名方式和层次结构可能还不统一。

你可能接触过“**服务树**”的概念，这个提法是小米在早期互联网运维实践的分享中传播出来的。我第一次听到这个概念是在 13 年阿里技术嘉年华大会上听小米运维的分享。再往前，

这个概念应该是从百度的运维体系中借鉴出来的。

这里的服务实际对应的就是我们前面提到的应用这个概念。据我了解，在阿里和腾讯都是叫作应用，现在业界比较通用的叫法也是应用。其实叫什么并不重要，关键还是要学习到对这个概念的管理方式。

从服务树这个名字中，我们就可以了解到，有效组织和管理应用的方式，就是把它组织成一个树形的层次结构。这种管理模式，无论是在 BAT，还是在其它的互联网公司，基本都是一样的思路和模式，所以叫法虽然不同，但是思路上是相通的，可谓异曲同工。

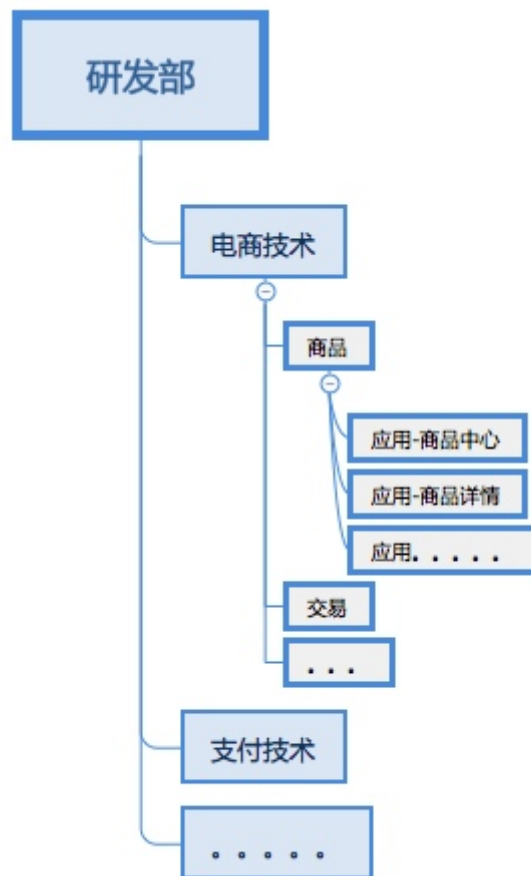
**基于业务维度的拆分，对应产生了我们的应用拆分原则。**比如对于电商公司，大的维度会有电商、支付、广告、流量和搜索等业务领域；进一步，电商业务领域里最典型的会有用户、会员、商品、交易、商家、店铺以及物流等；这里面还可以再进一步细分，比如商品会有详情、SKU、SPU、库存、评价、标签等。

讲到这里，我们再看一下技术团队的组织架构，基本上是对应着整个业务技术架构的拆分的。也就是**业务架构决定了技术架构，而技术架构又决定了一个研发团队的组织架构**，这个组织架构中不同的团队单元分别承担着对应业务的需求开发和实现职责。

上面这个组织架构建设的逻辑和思路，也是我们在组建团队和职责划分时可以参考的。

这样一个逻辑讲下来，我们的**应用管理思路**其实也就明晰了：**产品线 - 业务团队 - 应用。**

这里举个电商商品的例子就是：电商技术 - 商品团队 - 商品中心 - 商品详情等。



当然因为每个公司对组织架构定义的方式不同，也可以用一、二级部门这样的方式来指代。但是具体团队的分工和职责，一定是来自于业务架构决定的技术架构，只有这样，各业务团队才会职责清晰，配合协作才会顺畅起来。

对于应用名定义，要设定规范，比如：

应用名必须以大小写英文字母以及下划线组合；

应用名长度不超过 40 个字符，尽量简单易懂；

不允许出现机房代号和主机名称这样的信息。

简单举例，商品中心命名为 itemcenter，商品详情命名为 detail。

这里做个小结：**到了软件运维阶段，运维工作是否可以高效地组织开展，很大程度上，在前面的业务架构拆分阶段就决定了。也就是业务架构拆分得是否合理、职责是否明晰，决定了后续团队组织架构是否合理、团队职责是否明晰。如果这点没做好，到了运维阶段必然就是混乱的。**

这一点我在开篇词中也提到过，**运维能力的体现，一定是整体技术架构能力的体现，割裂两者单独去看，都是没有意义的。**同时，对于当前仍然把运维割裂建设的研发团队，也需要去思考一下在组织架构建设上的合理性了。

## 应用的集群服务分组建设

上述讲到的是应用的组织管理，看上去逻辑思路相对清晰，组织起来也不复杂，但是再往下，应用的集群服务分组建设就会相对复杂了。

为什么会有集群服务分组呢？我们一起来看这么几个需求场景。

### 场景一：多环境问题。

我们常见的环境会有开发联调环境、集成测试环境、预发环境、线上环境等等。后面我们讨论持续交付时会讲到，实际场景下所需要的环境会更多。

### 场景二：多 IDC 问题。

对于大型互联网业务，会做业务单元化，或者有海外业务拓展需求的场景，我们会在多个 IDC 机房部署应用，应用代码是相同的，但是配置可能会不同。

### 场景三：多服务分组问题。

这个场景就跟具体业务场景相关了。举个例子，比如商品中心 IC 这样一个核心应用，对外会有商品详情、交易下单、订单、购物车、评价、广告、秒杀活动、会场活动、商家、店铺等一系列应用依赖它，但是这些依赖它的应用优先级是不一样的。

**核心应用和非核心应用：**比如交易支付链路上的应用属于核心应用，任何时候都必须优先保障，但是对于评价、商家和店铺这些应用优先级就低一些。反过来理解就是一个应用出现故障，是不是会影响业务收入，如果影响就属于核心应用，如果不是或者影响非常小，那就属于非核心应用。所以 IC 这个应用下面，就会有 IC 的交易分组，IC 的广告分组、IC 的电商分组等，**这些分组就会相对固定和静态。**

**场景因素决定。**这个对于电商就会比较典型，比如大促时的秒杀场景，对于参加秒杀活动的商品，瞬时的访问量就会非常大，而不参加活动的商品就不会有这么大的访问量。所以这时为了隔离较大的流量，就需要有多个不同的秒杀 IC 分组，从资源层面进行隔离；同

时上层秒杀活动的应用在配置中心配置依赖时，就要配置到对应的秒杀 IC 集群分组上，这样即使秒杀 IC 出现问题，也不会影响正常的商品 IC 访问。所以根据场景，不同阶段就会有 IC 的大促秒杀分组，这种类型的分组就需要根据实际的业务场景来决定，是个动态调整的过程，需要开发和运维一起来讨论和验证。

一般情况下，集群服务分组会有以上三个维度中的一个或多个来决定。还是以商品中心 IC 为例，按照上面的介绍，就会对应如下关系：



至此，“应用 - 集群服务分组 - 资源”的对应关系就建立起来了。这里我们叫它“应用树”或者“服务树”都可以，不管叫什么，这个信息是 CMDB 中最为关键和核心的信息。为什么是最关键和核心的呢？

## CMDB 在基础服务体系中的核心位置

这里我们以应用为核心来看，CMDB 中会保存“应用 - 分组 - 资源”的对应关系，这个关系对于周边系统来说都是需要的，举例如下。

### 1.监控系统。

我们需要以上的对应关系，监控到每个应用、每个集群以及每台机器上的关键信息。

### 2.发布系统。

我们需要将每个应用对应的代码进行编译打包，然后发布到对应集群的主机上，也需要这个对应关系，这一点我在后面的持续交付中还会讲到。

### 3.服务化框架。

需要依赖应用和集群分组两个信息，其中主要是对应用名和集群分组名的依赖，对于服务化框架来说，更多的是通过其配置管理中心注册的应用名，来实现应用的服务和 API 管理，这里要做到与 CMDB 统一。同样，像 LVS 和 Nginx 这样的四七层负载，以及 ZK 这样的开源分布式配置管理，凡是涉及服务注册、服务发现以及服务上下线的基础服务，都是类似思路。

### 4.基础服务中。

如分布式 DB、分布式缓存和消息等，就需要应用的应用名，以及应用与资源 IP 的对应关系，或者集群分组与 IP 的对应关系。

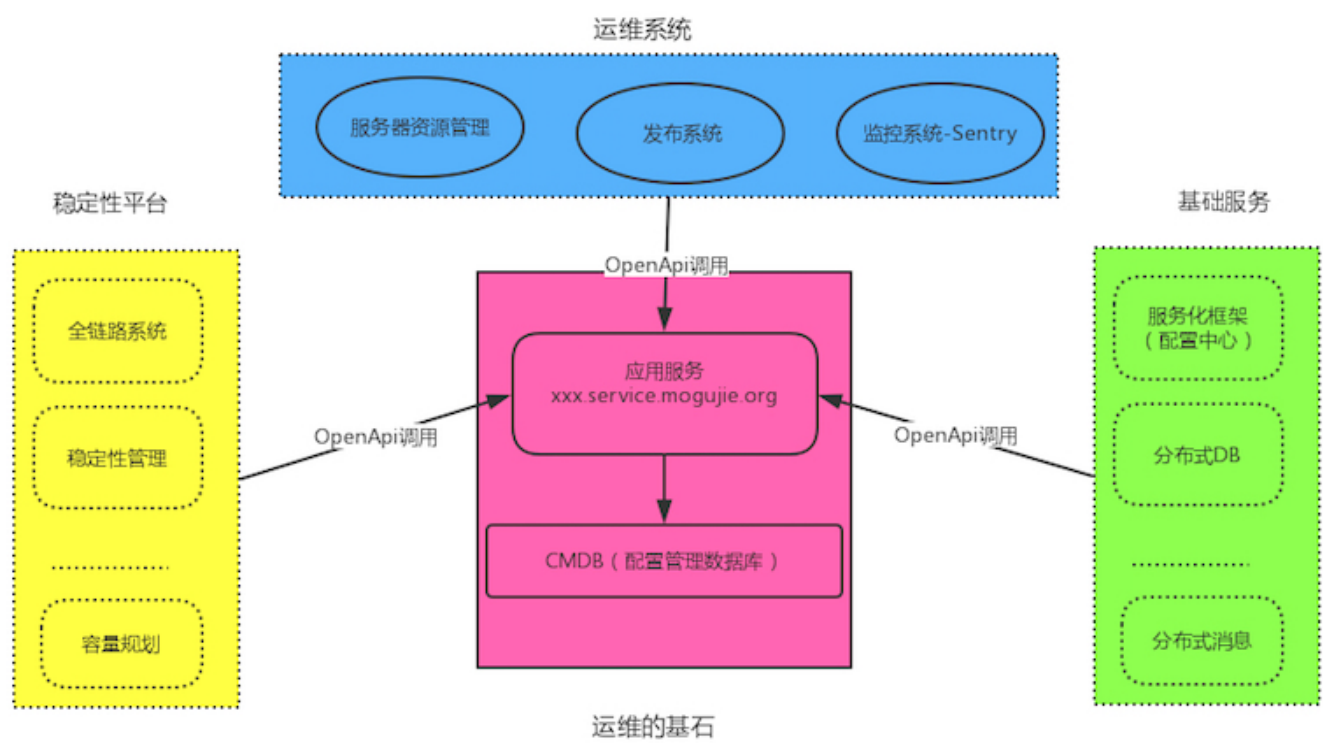
**应用名**，是因为要建立应用与分布式服务实例之间的关系。如应用与缓存 NameSpace 的对应关系，应用与消息 Topic 的对应关系等，以便于这些基础服务的生命周期管理和自动化开发。

**应用与资源的对应关系**，是因为有些核心资源是要做 ACL 访问控制的。比如对于用户、交易或支付这样非常敏感的数据，它们对应的数据库就不允许随意连接，而应该是仅限于授权过的应用访问。这时就要针对应用对应的 IP 地址进行白名单配置。一方面，可以通过分布式 DB 中间件进行配置；另一方面，也可以通过在 DB 层面进行设置，比如 MySQL 就可以直接配置白名单策略；同时也可以机器的 iptables 上配置，至于如何配置就看具体需求了，但是无论如何，应用与资源的对应关系是非常重要的。

5.稳定性保障平台，或者叫服务治理平台。

针对系统的稳定性，我们会在应用中做很多的降级限流和开关预案策略，这些都是跟应用直接关联的。而且按照我们前面介绍的，不同的集群分组，策略可能会有不同，所以又会跟集群分组相关。同时，这些策略最终下发到具体服务器上运行的应用实例上，所以这里就会需要应用、集群分组以及对应的资源关系。

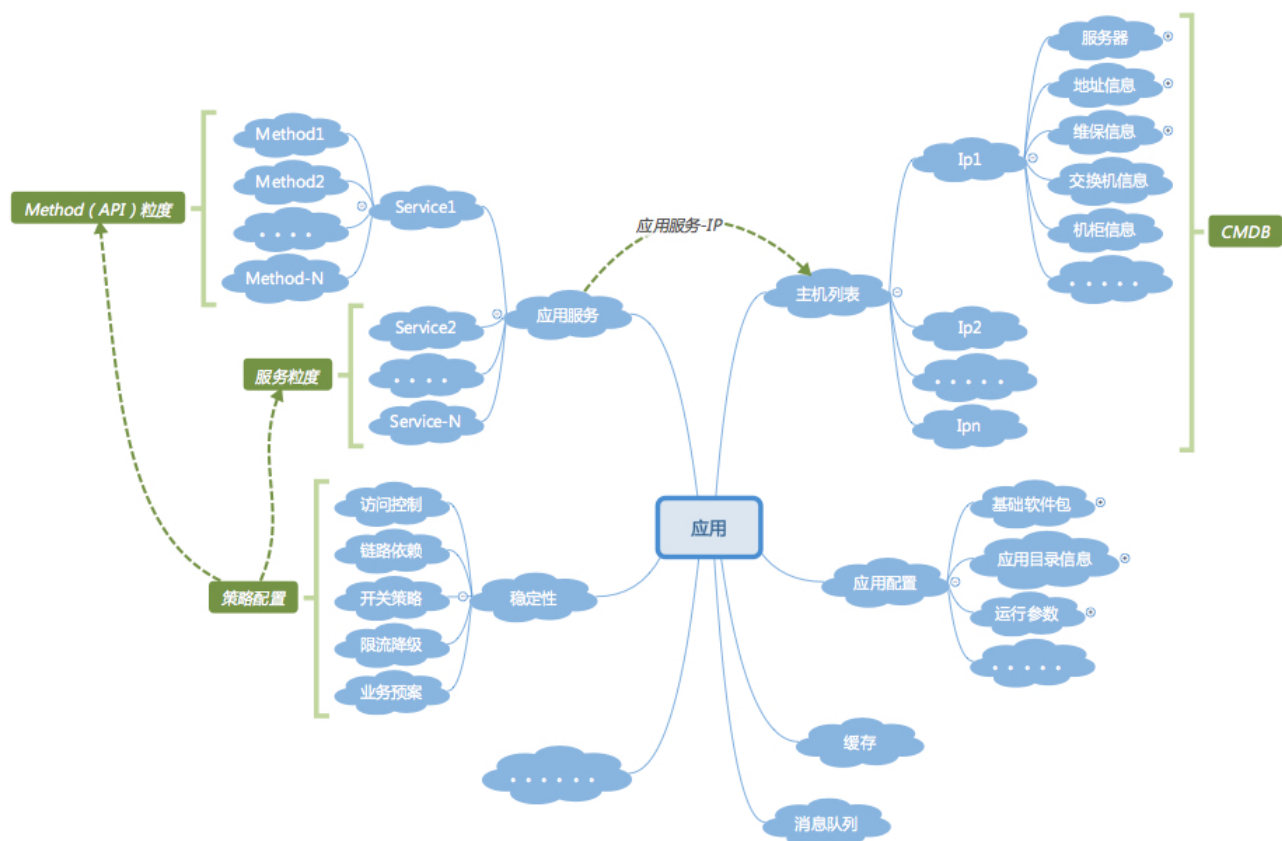
总结一下，简单示意图如下：



总结

通过上述的分析，我们可以看到**基于以应用为核心的 CMDB 中，又衍生出“应用 - 集群服务分组 - 资源”这样一个运维体系中的核心关系**。经过这三部分的分析，我们之前所说的基于应用为核心的运维视图就可以建立出来了，我们再次示意如下：





今天我们讨论的内容提到了，监控、发布、基础服务以及稳定性平台会依赖 CMDB 中“应用、集群服务分组 - 资源”的对应关系信息，但是当 CMDB 中的这些关系信息发生变化，比如新增一个 IP，或者下线一个 IP，这些信息是如何传递到其它平台的呢？这些平台又是如何查询这些关键信息的呢？欢迎你留言与我一起讨论。

如果今天的内容对你有帮助，也请你分享给身边的朋友，我们下期见！



# 赵成的运维体系管理课

带你直击运维的本质

赵成

美丽联合集团技术  
服务经理



新版升级：点击「👤 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 08 | 有了CMDB，为什么还需要应用配置管理？

下一篇 10 | 如何打造运维组织架构？

## 精选留言 (10)

写留言



赵成 置顶

2018-01-08

👍 2

本文重点，应用集群如何管理？怎么分组？以及组织架构如何与技术架构相匹配？

欢迎大家讨论。



圣诞使者

2018-01-10

👍 1

我想问下，cmdb需要体现应用的依赖关系吗？

展开 ▾

作者回复: 不需要, 应用的依赖更多体现在服务调用层面, cmdb里面的应用粒度还是会比较粗。这一点后面会有文章讲到。



宵伯特

2018-01-09

👍 1

对于小的开发团队或者初创的开发团队可能在基础设施架构的管理上并没有过多的资源, 而更偏向于使用云端的设施或架构, 甚至如serverless之类的计算服务。这方面的运维管理会有较大的差异或者模式上的差别吗?

展开 ▾

作者回复: 体量不同, 管理方式和采用的技术手段必然不一样, 就跟数据量大的表和量小的表, 查询方式和索引方式一定是不一样的。

量不大的时候可以怎么快怎么来, 即使没有运维, 问题也不大, 但是要有意识, 如果业务量开始快速增长了, 就要有规划和设计了。



kevinsu

2019-05-16

👍

如果业务大多是在云上呢? 是否只是需要针对公司的特定需求来做即可? 比如代码发布系统等等小系统, 而不是去整合成一个大的系统?



老牛

2018-12-15

👍

云环境下, 因为存在弹性的扩容缩容, 这样应用对应的资源(物理设备)是不是不固定的? 那么怎么保持这种对应关系呢?



张sir

2018-06-26

👍

你好, 文中提到“应用-集群服务分组-资源”, 请问下你们做服务化, 应用是最小的服务单元吗? 还是应用下的集群分组? 这样做有何用意及价值?

我们这边是“模块-应用-资源”, 模块是一个独立的子系统, 应用是最新单元, 资源就是这个应用所有环境的机器了, 比如测试, 预发, 线上。整体都是基于服务树的理念

展开 ▾



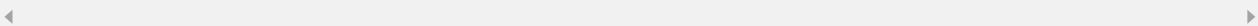
james

2018-03-26



应用所在容器不固定如何处理关系呢，比如弹性扩容所容以及应用down了重新启动一个docker镜像

作者回复: 应用跟运行载体不一定是ip，容器id或者pod的对应关系也可以



刘斌

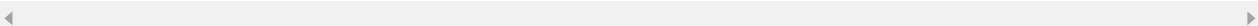
2018-01-24



之前看过您的文章提过“应用层的CMDB”，我感觉基础设施层CMDB不会涉及应用，还是以服务器为中心。应用所在的机器（及依赖的缓存、队列等），是在应用层的CMDB？

展开 ▾

作者回复: 其实不用纠结做在哪里，关键是要看解决什么问题，怎么解决问题，cmdb只是一个概念而已。



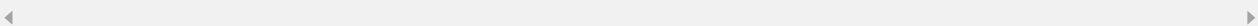
casper2dd

2018-01-16



cmdb提供API或者命令 用来查询应用 资源的信息 当cmdb信息有变化的时候 其他平台通过API或者命令 同步最新的信息

作者回复: 量大的话可以通过消息方式处理变更



天舟

2018-01-09



框架已经搭好，接下来就期待博主能讨论一个具体的实施方法了，比如说是纯粹基于name还是引入了tag，比如相关的组策略等等

作者回复: Tag模式适用于场景更复杂, 体量更大的情况下, 这种情况需要更为灵活的查询和分类, 我建议体量不大的话尽量简化设计。我们自己就当前情况, 一直是通过相对固定的分组模式来管理。

