

05 | 如何从生命周期的视角看待应用运维体系建设？

2017-12-29 赵成

赵成的运维体系管理课

[进入课程 >](#)



讲述：黄洲君

时长 10:48 大小 4.95M



还记得上周我们在讲标准化体系建设（上）的最后，我留了两个小问题，其中一个是这样的：

在对象属性识别过程中，我们进行了一些关键项的举例，但是如果换一个对象，我们有没有好的方法论来指导我们进行准确和全面的识别，而不至于遗漏？从我们今天的内容中，你有没有发现些规律呢？

这个问题的答案其实就是我们今天要讨论的内容，那就是从“**应用生命周期管理**”的角度分阶段去梳理。

简单理解下来就是，对于一个对象，既然有生命周期，就会有不同的生命周期阶段，那这个对象在不同的阶段，可能就会具备不同的属性、关系和场景。只要我们把一个对象的生命周

期阶段理清楚了，顺着这条主线分阶段进行分解，就可以分析得更加清晰、明确和具体了。

怎样理解生命周期

谈到生命，首先就会联想到我们自己，所以这里以人做一个简单的类比。我们人类从出生到死亡，就是一个生命周期，在这个周期的每一个阶段，我们都会有不同的属性、关系和所要面对的场景。

比如从人的学生时代开始，作为学生，我们就会具备学生的属性，会有所属学校、所属年级、所属班级、所属学号等等。这个时候我们周边的关系更多的是同学关系和师生关系。我们面临的场景可能就是读书、做作业和考试。当然学生时代细分下去还会有小学生、中学生、大学生以及研究生等阶段，每个阶段里面又会有不同的细分属性以及所要面临的场景，比如大学生毕业，就面对求职的场景等。

当一个学生毕业走入职场之后，这个时候就开启了生命周期里的另一个阶段，那就是职场生涯。这个时候我们身上的属性又发生了变化，具备所属公司、所谓职位、所谓层级等。这个时候的关系也更为复杂，如同事关系、上下级关系以及各种各样的社会关系。我们所面临的场景也变得复杂，要完成工作、晋升考核、领取薪酬以及离职跳槽、再次面试等等。

再往后，我们到了一定年纪，成为老年人，又会有老年人的属性、关系和场景，这里就不详细列举了。

围绕着人类的生命周期，我们国家和社会提供的解决方案，就必须要有的一系列对应的教育体系、职业体系、医疗体系、养老体系等。目的就是针对人生的不同阶段，提供不同形式的保障和支持，让每个人在社会体系下都够正常生存并发挥出自己的价值。

从上面的分析我们可以看到，人这个对象，在不同的生命周期阶段中，会有不同的角色属性和外部关系，同时要面对的社会和生存场景也不一样。所以，当我们谈论人这个对象的时候，一定是把这个对象放到具体的某个生命周期阶段中，才会有意义。

应用的生命周期分析

回到我们运维对象的生命周期上来，我们所面对的这些对象就相对规范、标准很多。

当一个场景下有多个对象时，就一定要找到那个核心的运维对象，这个核心对象的生命周期就会涵盖其它附属运维对象的子生命周期。结合我们前面讲过的内容，微服务架构下，一切

要以应用核心。因此，我们就找到了整个运维体系，或者说软件运行阶段的核心对象，那就是应用。

应用就类似于我们社会中的人，凡事都会以人为本，这里就是要以应用为本。那接下来按照上面我们对一个人的生命周期的阶段分解，我们也可以**对应用的生命周期阶段进行分解，大致分为五个部分，应用的创建阶段、研发阶段、上线阶段、运行阶段和销毁阶段**。我们依次来分析看一下。

1. 应用的创建阶段

这个阶段，最重要的工作，是确认应用的基础信息和与基础服务的关系，要同时固化下来，从应用创建之初，就将应用与各类基础服务的生命周期进行挂钩。

应用的基础信息，可以参考之前我们讲标准化的部分，基本上已经涵盖了比较全的信息，你可以按照生命周期的思路，再理解一下并做梳理。

对于同一类的应用，只需要做一次标准化即可，后续完全可以形成模板固化到工具平台上。

同时，**另外一个很重要的工作，就是要开启与应用相关的各类基础服务的生命周期**。比如这个应用需要用到缓存、消息队列和 DB 等，也可能需要域名 DNS 服务、VIP 配置等，这些就要从应用创建这个动作延伸出去，启动这些关联基础服务的创建，比如需要缓存就去申请容量空间，需要消息队列要申请创建新的 Topic 等等。

当然一个应用使用到哪些基础服务，应该是在架构设计和编码阶段就确定下来的，这里做的事情，就是把这些信息通过应用关联起来，与应用的生命周期挂钩。

2. 应用的研发阶段

应用的研发阶段主要是业务逻辑实现和验证的阶段。针对业务逻辑层面的场景就是开发代码和质量保证，但是这个过程就会涉及到代码的提交合并、编译打包以及在不同环境下的发布部署过程。同时，开发和测试在不同的环境下进行各种类型的测试，比如单元测试、集成测试以及系统测试等等，这整个过程就是我们常说的持续集成。

所以，这个阶段，我们要做的最重要的一个事情，就是为研发团队打造完善的持续集成体系和工具链支持，在后面我们会有专门一个部分讲解这个过程。

3. 应用的上线阶段

这是个过渡阶段，从应用创建过渡到线上运行。创建阶段，应用的基础信息和基础服务都已经到位，接下来就是申请到应用运行的服务器资源，然后将应用软件包发布上线运行，这个动作在下面的运行阶段也会持续迭代，我们直接看下面这个阶段。

4. 应用的运行阶段

这是应用生命周期中最重要、最核心的阶段。

从运维角度来看，应用在线上运行起来之后就已经变成一个线上运行的进程，那这个进程形态的应用应该有什么样的属性呢？你可能已经联想到，这个时候需要应用线上运行的各种指标的输出。所以这个阶段，应用最重要的属性就是应用本身以及相关的基础服务的各项运行指标。

这里，我们就需要制定每个运维对象的 SLI、SLO 和 SLA，同时要建设能够对这些指标进行监控和报警的监控体系。

从业务角度看，应用是线上业务逻辑的执行载体，但是我们的业务需求是在不断变化和迭代的，所以需要不断地去迭代更新我们的线上应用，这里仍然会依赖到上述应用研发阶段的持续集成过程，并最终与线上发布形成持续交付这样一个闭环体系。

从运行阶段应用的关系看，除了它跟基础服务之间相对固化的关系外，应用跟应用、以及应用包含的服务之间的调用关系也非常重要，而且这个关系可能随时都在变化，这个时候，我们应用之间依赖管理和链路跟踪的场景就出现了。

同时，应用线上运行还会面临外部业务量的各种异常变化，以及应用自身所依赖的基础设施、基础服务以及应用服务的各种异常状况。比如“双 11”大促，外部流量激增；微博上热点事件带来的访问量剧增；或者服务器故障、IDC 故障，DB 故障；又或者服务层面 API 的报错等等。这时就出现了线上稳定性保障的场景，比如流量激增时的限流降级、大促前的容量规划、异常时的容灾、服务层面的熔断等等。

通过上面的这个分析过程，我们可以看到，**日常接触到的各种技术解决方案，都是在解决应用生命周期不同阶段中应用自身或者应用与周边关系的问题，或者是所面对的场景问题。**

5. 应用的销毁阶段

这一部分就不难理解了。如果应用的业务职责不存在了，应用就可以下线销毁了。但是这里不仅仅是应用自身要销毁，我们说应用是整个运维体系的核心，所以围绕着某个应用所产生出来的基础设施、基础服务以及关联关系都要一并清理，否则将会给系统中造成许多无源（源头）的资源浪费。

我们在日常工作中，经常见到的缓存系统中，很多 NameSpace 不知道是谁的，消息系统中有很多 Topic 不知道是谁的，但是又不敢随意乱动，就只能让它无端占用着系统资源。

执行应用的销毁这一步动作，其实是取决于最前面应用与基础服务的关系模型分析和建设是否做得足够到位。

总结

今天我们分析了应用的生命周期，再结合之前讲的标准化内容，我们就找到了**做运维架构的切入点**，套路也就有了，总结一下就是：

从生命周期入手，划分阶段，提炼属性，理清关系，固化基础信息，实现运维场景。

同理，这个思路还可以运用到基础设施和基础服务对象的生命周期管理中，虽然它们只是子生命周期，但是具体到每个基础服务上面，同样需要这个管理手段和过程。

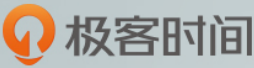
我已经介绍了很多和应用相关的内容，很大一部分的原因是希望能够帮助你梳理好思路，在思考问题和设计解决方案的时候，一定要从实际出发、从问题出发、从基础出发，理清自己的需求和痛点，然后再去寻求解决方案。

借鉴业界思路，千万不要一上来就去套用别人的解决方案。因为别人的思路和解决方案往往是建立在一个非常稳固的基础之上的，而这些基础，往往又因为太基础、太枯燥、太不够酷炫，所以常常是一带而过，甚至是略去不讲的。一旦忽略了这一点，再优秀的解决方案也是无源之水，无本之木，是实现不了的。

独立思考非常重要，共勉！

如果今天的内容对你有帮助，也请你分享给身边的朋友。

欢迎你留言与我一起讨论。



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 04 | 标准化体系建设（下）：如何建立基础架构标准化及服务化体系？

下一篇 06 | 我是如何走上运维岗位的？

精选留言 (10)

💬 写留言



赵成

2018-01-02

👍 15

做应用运维管理的套路:从应用生命周期入手，划分阶段，提炼属性，理清关系，固化基础信息，实现运维场景。



白开水

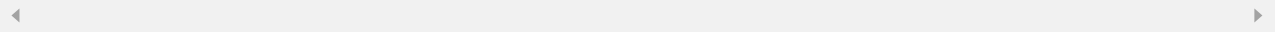
2018-01-04

👍 2

说得很透彻，令人茅塞顿开！

展开 ▾

作者回复: 对大家有帮助, 很开心!



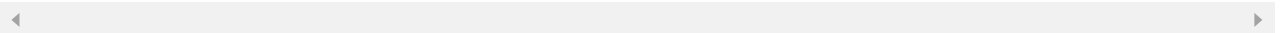
岑峯

2018-01-03

👍 2

从生命周期入手, 不仅能够让运维人员理清思路, 指导工作方向; 更是让运维管理者在有限的资源中选择正确的方向和事去做, 达到事半功半的效果

作者回复: 你抓住了重点。



Genius

2018-01-10

👍 1

前面两章内容终于串起来了, 棒

展开 ▾



杨家二少爷

2019-02-12

👍

说的真棒, 分析很到位、透彻, 谢谢

展开 ▾



守艺人·惜...

2018-10-18

👍

“梳理好思路, 在思考问题和设计解决方案的时候, 一定要从实际出发、从问题出发、从基础出发, 理清自己的需求和痛点, 然后再去寻求解决方案”。

受益匪浅。



hk

2018-07-03

👍

受益匪浅👍

展开 ▾



男儿当自强

2018-05-14



目前公司属于初创性质，也有5年多了，运维人员也就4个左右，现在集成还会存在对接不清楚情况，如果标准规范，相信可以更好



老陈醋

2018-03-29



以应用为中心没错 但是梳理对象属性的时候以应用的生命周期并不能涵盖所有的对象的属性 这个地方是不是主要是思路 以对象的生命周期去思考和梳理

作者回复: 对象属性的梳理是个持续过程，关键看线上运行阶段要关注那些属性，要管理那些属性，从生命周期的角度去看，不一定一步到位全部梳理清楚，但是这个思路提供了这样一个梳理方法，尽可能的确保全面



老陈醋

2018-03-29



没太明白应用的创建阶段具体指的是什么？是在gitlab里创建工程吗 在这个阶段一般不需要做各种资源和基础服务组件的申请呀 我的理解不对吗

作者回复: 应用创建时，应用名，责任人，gitlab地址，代码类型，配置基线，是否核心应用等等都要确认下来，这些都是后续应用管理的关键要素。

同时，应用是不是需要对接缓存，消息，db，需要应对多大流量，这些在应用架构设计时是应该提前设计好的，并不一定在创建时就申请，但是上线发布前，都应该申请联调完成。

