

# Dubbo 稳定性案例：Nacos 注册中心可用性问题复盘

作者：徐靖峰（岛风）

创作日期：2019-12-02

专栏地址：[【稳定大于一切】](#)

PDF 格式：[Dubbo 稳定性案例：Nacos 注册中心可用性问题复盘](#)

## 问题描述

上周四晚刚回到家，就接到了软负载同学的电话，说是客户线上出了故障，我一听“故障”两个字，立马追问是什么情况，经过整理，还原出线上问题的原貌：

客户使用了 Dubbo，注册中心使用的是 Nacos，在下午开始不断有调用报错，查看日志，发现了 Nacos 心跳请求返回 502

```
2019-11-15 03:02:41.973 [com.alibaba.nacos.client.naming454] -ERROR [com.alibaba.nacos.naming.beat.sender] request xx.xx.xx.xx failed.
com.alibaba.nacos.api.exception.NacosException: failed to req API: xx.xx.xx.xx:8848/nacos/v1/ns/instance/beat. code:502 msg:
```

此时还没有大范围的报错。随后，用户对部分机器进行了重启，开始出现大规模的 Nacos 连接不上的报错，并且调用开始出现大量 no provider 的报错。

## 问题分析

Nacos 出现心跳报错，一般会有两种可能：

- 用户机器出现问题，如网络不通
- Nacos Server 宕机

但由于是大面积报错，所以很快定位到是 Nacos Server 本身出了问题：由于磁盘老旧导致 IO 效率急剧下降，Nacos Server 无法响应客户端的请求，客户端直接接收到 502 错误响应。这个事件本身并不复杂，是一起注册中心磁盘故障引发的血案，但从这起事件，却可以窥探到很多高可用的问题，下面来跟大家一起聊聊这当中的细节。

## 问题复现

**Dubbo 版本：**2.7.4

**Nacos 版本：**1.1.4

**复现目标：**在本地模拟 Nacos Server 宕机，检查 Dubbo 的调用是否会受到影响。

**复现步骤：**

1. 本地启动 Nacos Server、Provider、Consumer，触发 Consumer 调用 Provider
2. kill -9 Nacos Server，模拟 Nacos Server 宕机，触发 Consumer 调用 Provider
3. 重启 Consumer，触发 Consumer 调用 Provider

**期望：**

3 个步骤均可以调用成功

**实际结果：**

1、2 调用成功，3 调用失败

问题成功复现，重启 Consumer 之后，没有调用成功，客户恰好遇到了这个问题。大家可能对这其中的细节还是有一些疑问，我设想了一些疑惑点，来和大家一起进行探讨。

## 为什么 Nacos 宕机后，仍然可以调用成功

我们都知道，一般聊到 Dubbo，有三个角色是必须要聊到的：服务提供者、服务消费者、注册中心。他们的关系不用我赘述，可以从下面的连通性列表得到一个比较全面的认识：

- 注册中心负责服务地址的注册与查找，相当于目录服务，服务提供者和消费者只在启动时与注册中心交互，注册中心不转发请求，压力较小
- 服务提供者向注册中心注册其提供的服务，此时间不包含网络开销
- 服务消费者向注册中心获取服务提供者地址列表，并根据负载算法直接调用提供者，此时间包含网络开销
- 注册中心，服务提供者，服务消费者三者之间均为长连接
- 注册中心通过长连接感知服务提供者的存在，服务提供者宕机，注册中心将立即推送事件通知消费者
- **注册中心宕机，不影响已运行的提供者和消费者，消费者在本地缓存了提供者列表**
- 注册中心可选的，服务消费者可以直连服务提供者

重点关注倒数第二条，Dubbo 其实在内存中缓存了一份提供者列表，这样可以方便地在每次调用时，直接从本地内存拿地址做负载均衡，而避免每次调用都访问注册中心。只有当服务提供者节点发生上下线时，才会推送到本地，进行更新。所以，Nacos 宕机后，Dubbo 仍然可以调用成功。

## Nacos 宕机不影响服务调用，为什么日志中仍然有调用报错

宕机期间，已有的服务提供者节点可能突然下线，但由于注册中心无法通知给消费者，所以客户端调用到下线的 IP 就会出现报错。

对于此类问题，Dubbo 也可以进行兜底

- Dubbo 会在连接级别进行心跳检测，当 channel 本身不可用时，即使没有注册中心通知，也会对其进行断连，并设置定时器，当该连接恢复后，再恢复其可用性
- 在阿里云商业版的 Dubbo -- EDAS 中，提供了「离群摘除」功能，可以在调用层面即时摘除部分有问题的节点，保证服务的可用性。

## 为什么期望 Consumer 重启之后，调用成功

Nacos Server 宕机后，Consumer 依旧可以调用成功，这个大家应该都比较清楚。但是为什么期望 Consumer 重启之后，依旧调用成功，有些人可能就会有疑问了，注册中心都宕机了，重启之后一定连不上，理应调用失败，怎么会期望成功呢？这就要涉及到 Nacos 的本地缓存了。

**Nacos 本地缓存的作用：**当应用与服务注册中心发生网络分区或服务注册中心完全宕机后，应用进行了重启操作，内存里没有数据，此时应用可以通过读取本地缓存文件的数据来获取到最后一次订阅到的内容。

例如在 Dubbo 应用中定义了如下服务：

```
<dubbo:service interface="com.alibaba.edas.xml.DemoService" group="DUBBO" version="1.0.0" ref="demoService" />
```

可以在本机的 `/home/${user}/nacos/naming/` 下看到各个命名空间发布的所有服务的信息，其内容格式如下：

```
{ "metadata": {}, "dom": "DEFAULT_GROUP@@providers:com.alibaba.edas.xml.DemoService:1.0.0:DUBBO", "cacheMillis": 10000, "useSpecifiedURL": false, "hosts": [ { "valid": true, "marked": false, "metadata": { "side": "provider", "methods": "sayHello", "release": "2.7.4", "deprecated": "false", "dubbo": "2.0.2", "pid": "5275", "interface": "com.alibaba.edas.xml.DemoService", "version": "1.0.0", "generic": "false", "revision": "1.0.0", "path": "com.alibaba.edas.xml.DemoService", "protocol": "dubbo", "dynamic": "true", "category": "providers", "anyhost": "true", "bean.name": "com.alibaba.edas.xml.DemoService", "group": "DUBBO", "timestamp": "157535563302" }, "instanceId": "30.5.122.3#20880#DEFAULT#DEFAULT_GROUP@@providers:com.alibaba.edas.xml.DemoService:1.0.0:DUBBO", "port": 20880, "healthy": true, "ip": "30.5.122.3", "clusterName": "DEFAULT", "weight": 1.0, "ephemeral": true, "serviceName": "DEFAULT_GROUP@@providers:com.alibaba.edas.xml.DemoService:1.0.0:DUBBO", "enabled": true } ], "name": "DEFAULT_GROUP@@providers:com.alibaba.edas.xml.DemoService:1.0.0:DUBBO", "checksum": "69c4eb7e03c03d4b18df129829a486a", "lastRefTime": 157535563862, "env": "", "clusters": "" }
```

为什么期望重启后调用成功？因为经过检查，发现线上出现问题的机器上，缓存文件一切正常。虽然 Nacos Server 宕机了，本地的缓存文件依旧可以作为一个兜底，所以期望调用成功。

## 为什么 Consumer 重启后，没有按照预期加载本地缓存文件

缓存文件正常，问题只有可能出现在读取缓存文件的逻辑上。

- 可能是 nacos-client 出了问题

- 可能是 Dubbo 的 nacos-registry 出了问题

一番排查，在 Nacos 研发的协助下，找到了 nacos-client 的一个参数：`namingLoadCacheAtStart`，该配置参数控制启动时是否加载缓存文件，默认值为 `false`。也就是说，使用 nacos-client，默认是不会加载本地缓存文件的。终于定位到线上问题的原因了：需要手动开启加载本地缓存，才能让 Nacos 加载本地缓存文件。

该参数设置为 `true` 和 `false` 的利弊：

- 设置为 `true`，认为可用性 & 稳定性优先，宁愿接受可能出错的数据，也不能因为没有数据导致调用完全出错
- 设置为 `false`，则认为 Server 的可用性较高，更能够接受没有数据，也不能接受错误的数据

无论是 `true` 还是 `false`，都是对一些极端情况的兜底，而不是常态。对于注册发现场景，设置成 `true`，可能更合适一点，这样可以利用 Nacos 的本地缓存文件做一个兜底。

## Dubbo 传递注册中心参数

Dubbo 中使用统一 URL 模型进行参数的传递，当我们需要在配置文件传递注册中心相关的配置参数时，可以通过键值对的形式进行拼接，当我们想要在 Dubbo 中开启加载注册中心缓存的开关时，可以如下配置：

```
<dubbo:registry address="nacos://127.0.0.1:8848?namingLoadCacheAtStart=true"/>
```

遗憾的是，最新版本的 Dubbo 只传递了部分参数给 Nacos Server，即使用户配置了

`namingLoadCacheAtStart` 也不会被服务端识别，进而无法加载本地缓存。我在本地修改了 Dubbo 2.7.5-SNAPSHOT，传递上述参数后，可以使得 1、2、3 三个阶段都调用成功，证明了

`namingLoadCacheAtStart` 的确可以使得 Dubbo 加载本地缓存文件。该问题将会在 Dubbo 2.7.5 得到修复，届时 Dubbo 中使用 Nacos 的稳定性将会得到提升。

## 问题总结

该线上问题反映出了 Nacos 注册中心可用性对 Dubbo 应用的影响，以及系统在某个组件宕机时，整体系统需要进行的一些兜底逻辑，不至于因为某个组件导致整个系统的瘫痪。

总结下现有代码的缺陷以及一些最佳实践：

- Dubbo 传递注册中心参数给 Nacos 时，只能够识别部分参数，这会导致用户的部分配置失效，在接下来的版本会进行修复。
- nacos-client 加载本地缓存文件的开关等影响到系统稳定性的参数最好设计成 `-D` 启动参数，或者环境变量参数，这样方便发现问题，及时止血。例如此次的事件，有缺陷的 Dubbo 代码仅仅依赖于参数的传递，无法加载本地缓存文件，而如果有 `-D` 参数，可以强行开始加载缓存，大大降低了问题的影响面。
- `namingLoadCacheAtStart` 是否默认开启，还需要根据场景具体确定，但 nacos-server 宕机等极端场景下，开启该参数，可以尽可能地降低问题的影响面。顺带一提，Nacos 本身还提供了一个本地灾

备文件，与本地缓存文件有一些差异，有兴趣的朋友也可以去了解一下。

## 推荐链接

---

- [Dubbo——国内最火的开源服务框架](#)
- [Nacos——开源的动态服务发现/配置管理平台](#)
- [EDAS——公有云应用托管和微服务管理的 PaaS 平台](#)

## 加入我们

---

【稳定大于一切】打造国内稳定性领域知识库，让无法解决的问题少一点点，让世界的确定性多一点点。

- [GitHub 地址](#)
- 钉钉群号：
  - 23168197（2群，推荐）
  - 23179349（1群，已满）
- 如果阅读本文有所收获，欢迎分享给身边的朋友，期待更多同学的加入！