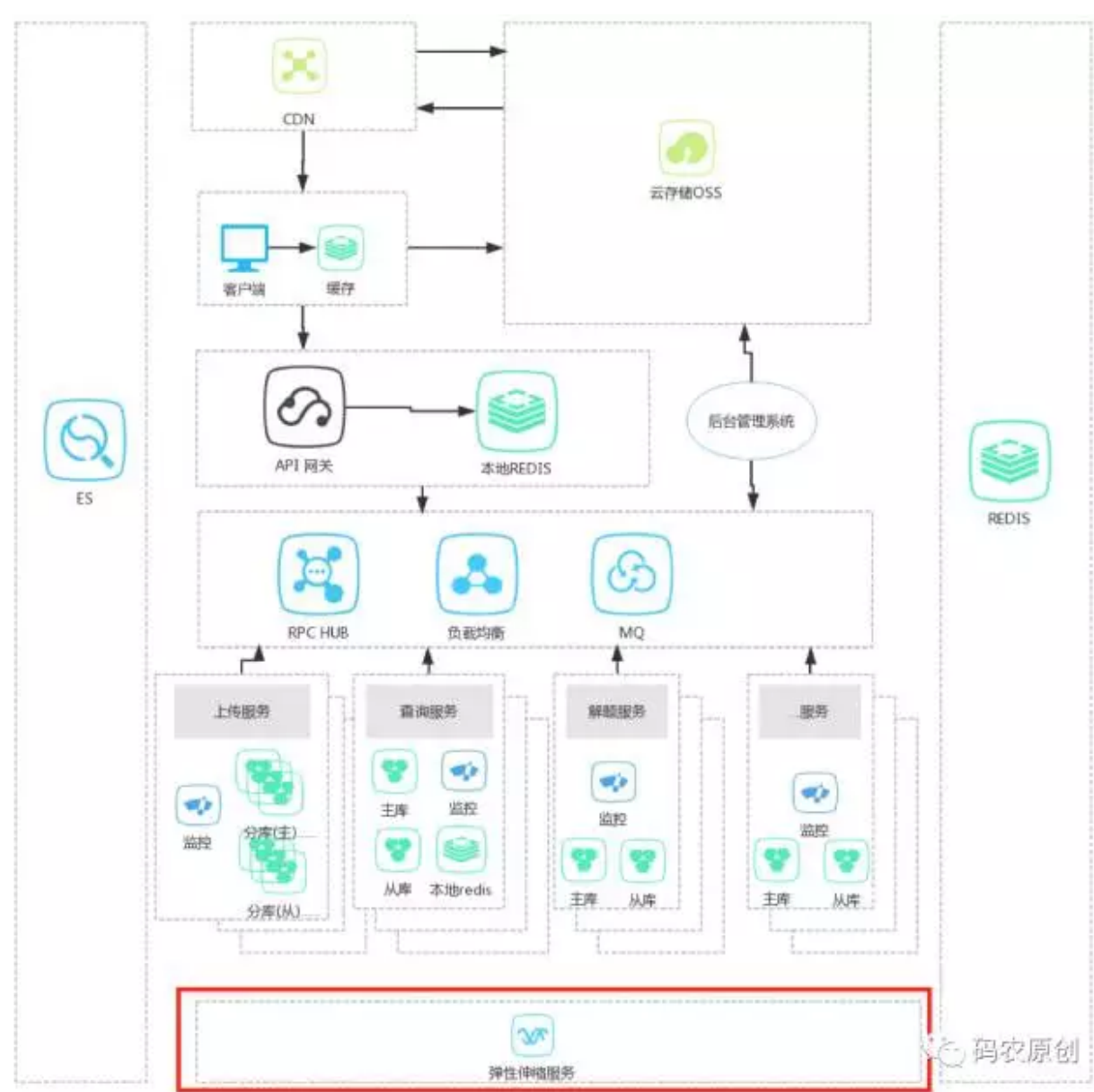


这种模式主要解决突发流量的到来，导致无法横向扩展或者横向扩展太慢，进而影响业务，全站崩溃的问题。这个模式是一种相对来说比较高级的技术，也是各个大公司目前都在研究、试用的技术。截至今日，有这种思想的架构师就已经是很不错了，能够拿到较高薪资，更别提那些已经实践过的，甚至实现了底层系统的那些，所以，你懂得.....

这种模式的一般设计见下图：



如上图所示，多了一个弹性伸缩服务，用来动态的增加、减少实例。原理上非常简单，但是这个模式到底解决什么问题呢？先说说由来和意义。

每年的双11、六一八或者一些大促到来之前，我们都会为大流量的到来做以下几个方面的工作：

1. 提前准备10倍甚至更多的机器，即使用不上也要放在那里备着，以防万一。这样浪费了大量的资源。
2. 每台机器配置、调试、引流，以便让所有的机器都可用。这样浪费了大量的人力、物力，更容易出错。
3. 如果机器准备不充分，那么还要加班加点的重复上面的工作。这样做特别容易出错，引来领导的不满，没时间回家陪老婆，然后你的老婆就.....(自己想)

在双十一之后，我们还要人工做缩容，非常的辛苦。一般一年中会有多次促销，那么我们会一直这样，实在是烦！

最严重的，突然间的大流量爆发，会让我们触不及防，半夜起来扩容是在正常不过的事情，为此，我们偷懒起来，要更多的机器备着，也就出现了大量的cpu利用率为1%的机器。

我相信，如果你是老板一定很震惊吧！！

哈哈，那么如何改变这种情况呢？请接着看

为此，首先把所有的计算资源整合成资源池的概念，然后通过一些策略、监控、服务，动态的从资源池中获取资源，用完后在放回到池子中，供其他系统使用。

具体实现上比较成熟的两种资源池方案是VM、docker，每个都有着自己强大的生态。监控的点有CPU、内存、硬盘、网络IO、服务质量等，根据这些，在配合一些预留、扩张、收缩策略，就可以简单的实现自动伸缩。怎么样？是不是很神奇？深入的内容我们会在的码农原创的公众号文章中详细介绍。

该总结一下这种模式的优缺点的了，如下：

**优点：**弹性、按需计算，充分优化企业计算资源。

**缺点：**应用要从架构层做到可横向扩展化改造、依赖的底层配套比较多，对技术水平、实力、应用规模要求较高。