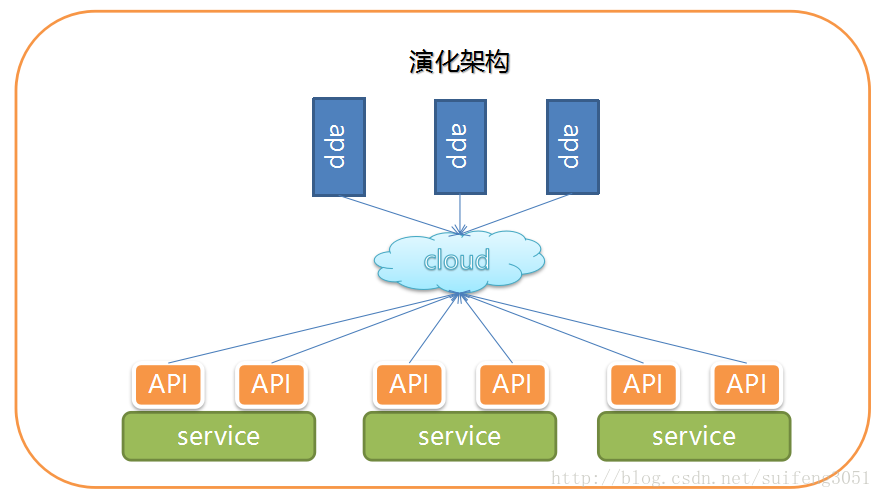
## 微服务

现在很多人都在谈微服务，那么到底什么是微服务呢？这里谈谈我对微服务的理解。

微服务有两个核心：

* **微：服务的粒度要细，即服务要细化到API**
* **服务：提供好服务，要让用户感到好用(要做到这一点很不容易)**

上面两个核心总结起来，可以用下面这幅图表示：



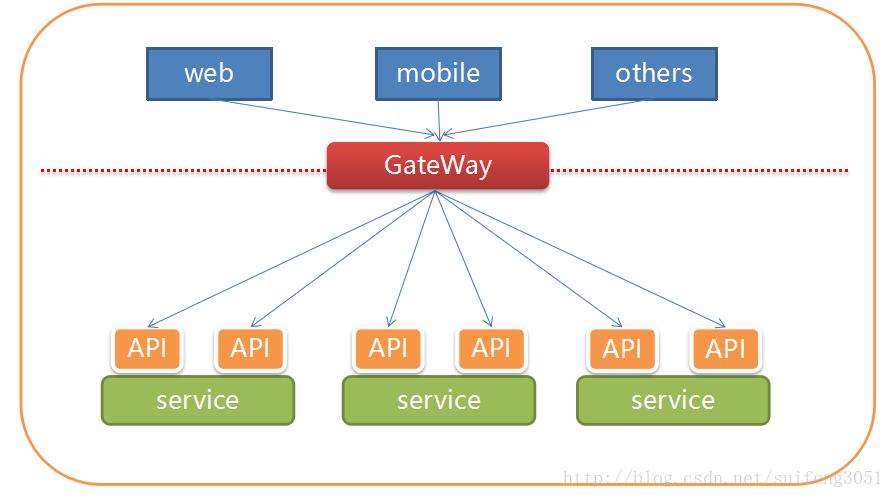
从上面这幅图看出，微服务特别简单（好的架构就应该简单），我们把服务再拆分成一个个API，**API是一个完整的功能。**然后我们把API扔到一个“云上”，然后用户就可以到“云上”**获取所有API的服务**，这个“云”保证能提供好的服务。

我们可以看到，有了微服务之后，服务对用户来说变得特别简单，而且dubbo的不足之处在微服务这里都解决了。**使用者不再需要依赖任何jar包，不再需要去注册中心查找服务，不再去做鉴权处理，不用担心服务挂掉，不用担心不会使用服务，所有的问题这个“云”都解决了**。

这也是微服务的核心之一，提供好服务。

说到这里，大家就应该大体知道该怎么做微服务了，**图中的“云”是关键**。下面我们就慢慢拨开这朵云。

### 微服务的实现



微服务的关键是**服务网关**，所以，上面提到的“云”就是服务网关。要做微服务，我们先定义一下微服务需要具备的特点。

#### 微服务的特点

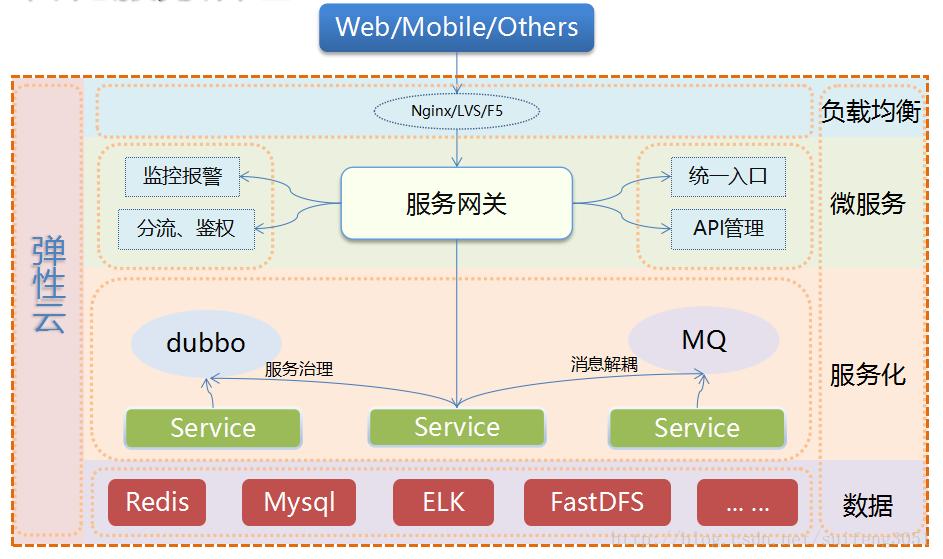
服务需要再细化成API（服务接口——>服务API）

* 每个服务由一组API组成
* 以API形式对外提供统一格式的服务
* 使用者无需任何配置，直接调用（http）
* 完整的API文档
* API服务安全可靠稳定

#### 微服务要解决的问题

上面提到了，dubbo还存在一些问题 ，其实dubbo存在的问题 就是 微服务要解决的问题，这里 再总结一下。当然，dubbo和微服务的侧重点不一样，dubbo侧重于内部接口之间的RPC(RPC，即 Remote Procedure Call（远程过程调用），说得通俗一点就是：调用远程计算机上的服务，就像调用本地服务一样。)，而**微服务则侧重于对外提供服务**。

#### 实现方案



* 负载均衡层：nginx/lvs/F5
* **微服务层 （所有请求的入口）**
  + **高性能服务网关**
  + **API配置管理(Spring cloud Bus)、统一入口、分流鉴权(Zurl，pre那种)、服务监控(Zurl)、协议转换(Zurl)**
  + **API映射、OAuth2.0、API文档管理**
  + **分布式、可拓展**
* 服务治理层   
  + **成熟的服务治理框架dubbo**
  + MQ服务之间解耦
* 弹性云   
  + 服务docker化
  + **基于访问压力的实时集群调度与管理**

#### 弹性云

这里简单介绍一下弹性云的概念，微服务要想提供好服务，保证API不能挂掉并且有好的性能，需要很高的运维要求。这里的弹性云便是**自动化运维解决方案，**对访问压力进行监控，根据**监控解决调度应用的发布和回收。**

