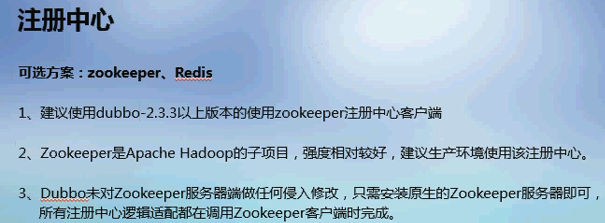
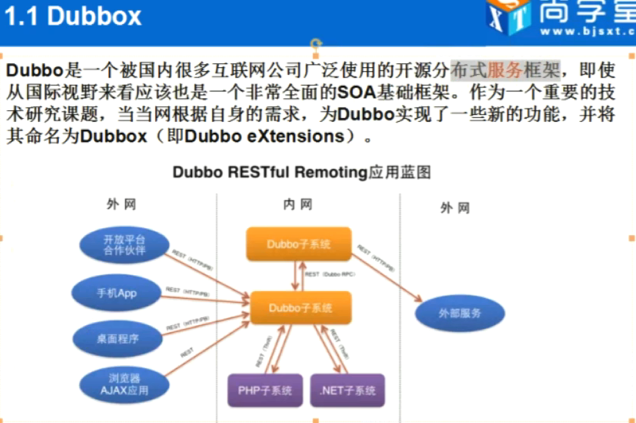
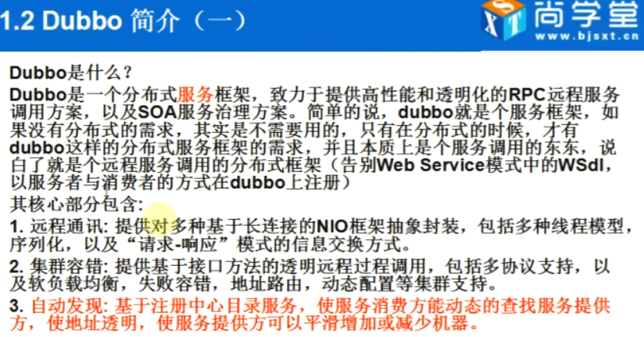
## Dubbo





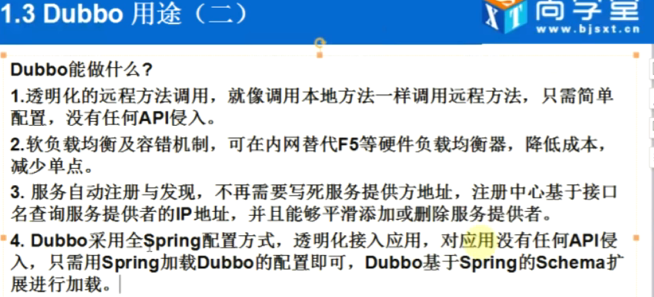
### 简介



Dubbo是一个分布式服务框架，致力于提供高性能和透明化的RPC远程服务调用方案，以及SOA服务治理方案。 其核心部分包含:

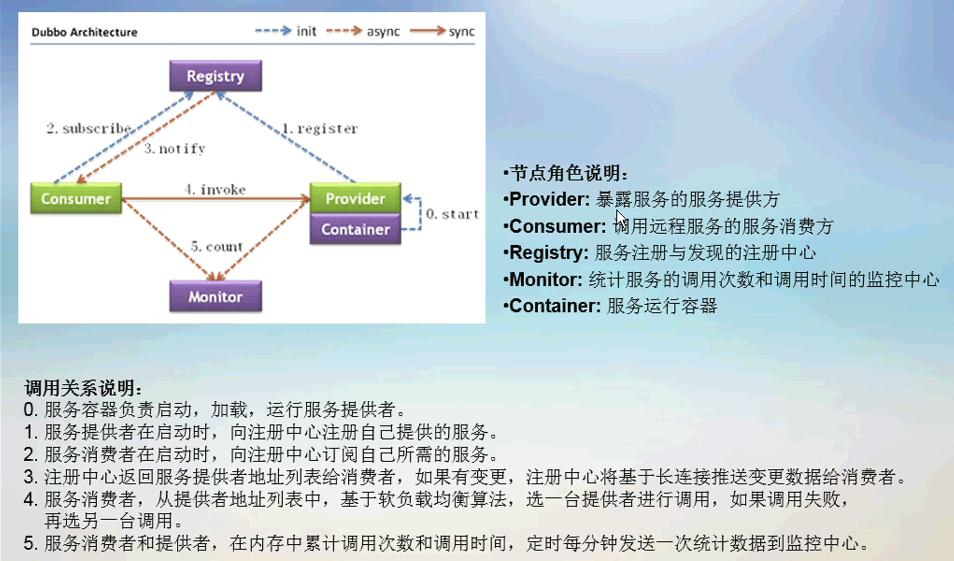
* 远程通讯: 提供对多种基于长连接的NIO框架抽象封装，包括多种线程模型，序列化，以及“请求-响应”模式的信息交换方式。
* 集群容错: 提供基于接口方法的透明远程过程调用，包括多协议支持，以及软负载均衡，失败容错，地址路由，动态配置等集群支持
* 自动发现: 基于注册中心目录服务，使服务消费方能动态的查找服务提供方，使地址透明，使服务提供方可以平滑增加或减少机器。

### 应用



* 透明化的远程方法调用，就像调用本地方法一样调用远程方法，只需简单配置，没有任何API侵入。
* 软负载均衡及容错机制，可在内网替代F5等硬件负载均衡器，降低成本，减少单点。
* 服务自动注册与发现，不再需要写死服务提供方地址，注册中心基于接口名查询服务提供者的IP地址，并且能够平滑添加或删除服务提供者。

### 架构

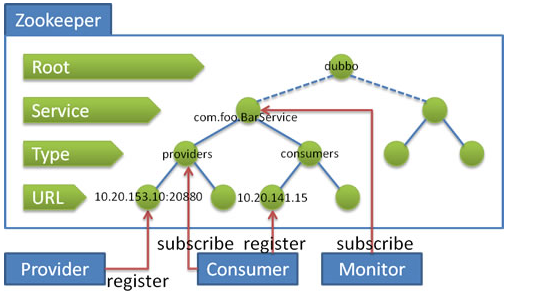


Admin管控台：可以查看有哪些消费者 提供者 以及服务，可以配置负载均衡策略、路由等。

1. 连通性： 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 监控中心负责统计各服务调用次数，调用时间等，统计先在内存汇总后每分钟一次发送到监控中心服务器，并以报表展示 服务提供者向注册中心注册其提供的服务，并汇报调用时间到监控中心，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，同时汇报调用时间到监控中心，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，监控中心除外 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 注册中心和监控中心全部宕机，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 注册中心和监控中心都是可选的，服务消费者可以直连服务提供者
2. 健状性： 监控中心宕掉不影响使用，只是丢失部分采样数据 数据库宕掉后，注册中心仍能通过缓存提供服务列表查询，但不能注册新服务 注册中心对等集群，任意一台宕掉后，将自动切换到另一台 注册中心全部宕掉后，服务提供者和服务消费者仍能通过本地缓存通讯 服务提供者无状态，任意一台宕掉后，不影响使用 服务提供者全部宕掉后，服务消费者应用将无法使用，并无限次重连等待服务提供者恢复。
3. 伸缩性： 注册中心为对等集群，可动态增加机器部署实例，所有客户端将自动发现新的注册中心 服务提供者无状态，可动态增加机器部署实例，注册中心将推送新的服务提供者信息给消费者
4. 升级性： 当服务集群规模进一步扩大，带动IT治理结构进一步升级，需要实现动态部署，进行流动计算，现有分布式服务架构不会带来阻力

### 注册中心

#### ZK



* 服务提供者启动时: 向/dubbo/com.foo.BarService/providers目录下写入自己的URL地址。
* 服务消费者启动时: 订阅/dubbo/com.foo.BarService/providers目录下的提供者URL地址。并向/dubbo/com.foo.BarService/consumers目录下写入自己的URL地址。
* 监控中心启动时: 订阅/dubbo/com.foo.BarService目录下的所有提供者和消费者URL地址。

功能

* 当提供者出现断电等异常停机时，注册中心能自动删除提供者信息。
* 当注册中心重启时，能自动恢复注册数据，以及订阅请求。
* 当会话过期时，能自动恢复注册数据，以及订阅请求。
* 当设置 <dubbo:registry check="false" /> 时，记录失败注册和订阅请求，后台定时重试。
* 可通过 <dubbo:registry username="admin" password="1234" /> 设置zookeeper登录信息。
* 可通过 <dubbo:registry group="dubbo" /> 设置zookeeper的根节点，不设置将使用无根树。
* 支持 \* 号通配符 <dubbo:reference group="\*" version="\*" />，可订阅服务的所有分组和所有版本的提供者。

#### Redis

使用的hash结构

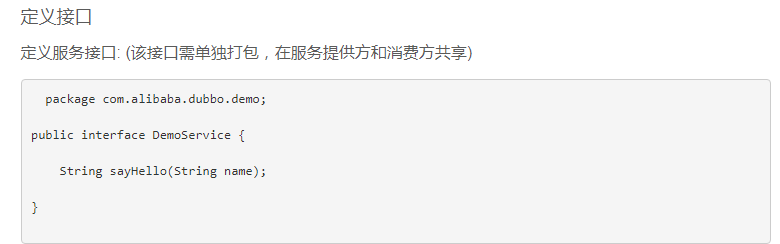
### Quick start

Eclipse中的dubbo代码

服务消费者需要和服务提供者需要有相同的接口，实际应用中是把这些接口单独作为一个应用，然后消费者和提供者都去引用这个应用接口。

服务提供者在启动时，会向zk中注册，创建节点名为 暴露接口的全名称。















## 官方文档

查看dubbo的官方文档，用户指南

<http://dubbo.io/user-guide/preface/background.html>