# Dubbo:

## Dubbo介绍

## Dubbo特性

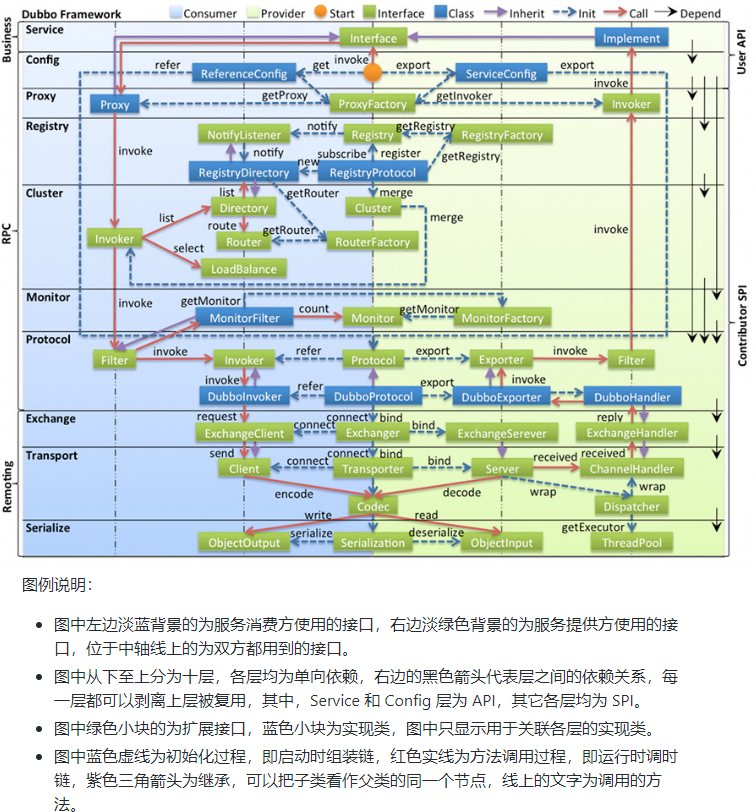
## Dubbo功能

## Dubbo原理

## Dubbo 理解

本次为了节约时间,只写这一点后期补充上完整的,目前完整的参看

[**http://dubbo.apache.org/#!/?lang=zh-cn**](http://dubbo.apache.org/#!/?lang=zh-cn)



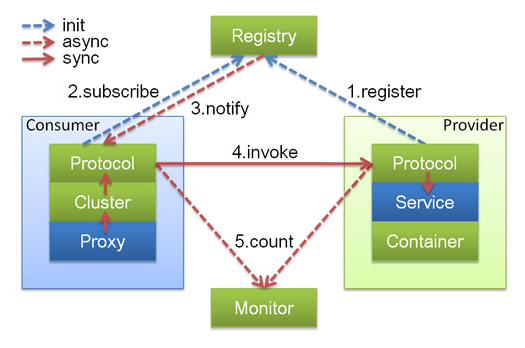
分析步骤:

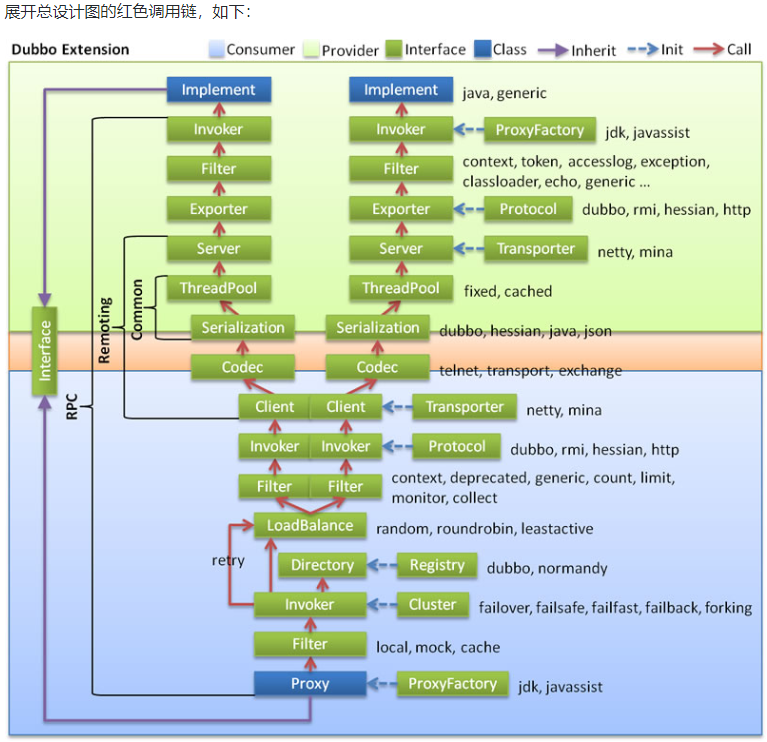
图中可拆分2部分分析:

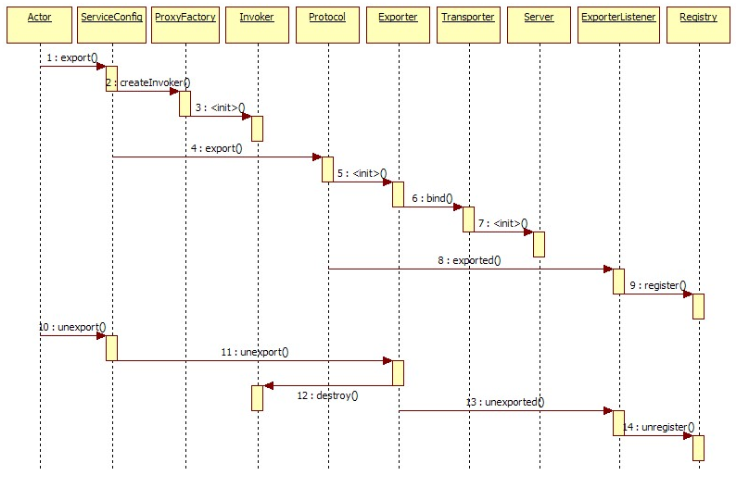
只看虚线蓝色箭头(初始化)

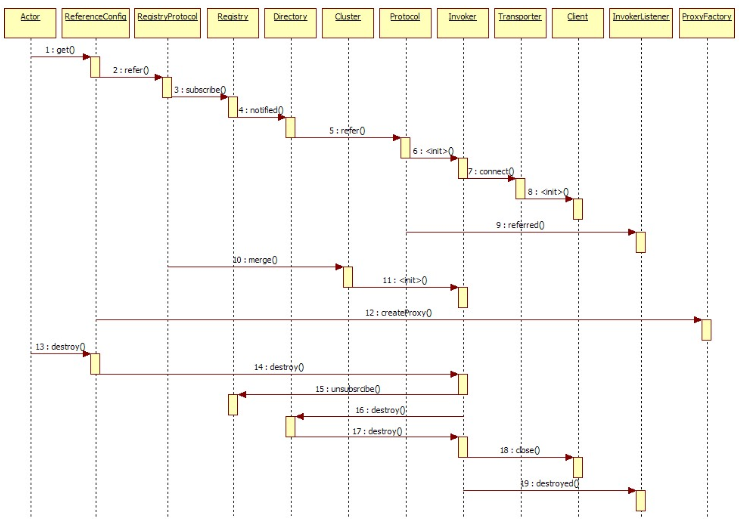
只看红色箭头指向(请求走向)

上图中的蓝色虚线框内容,具体可以拆分成下图

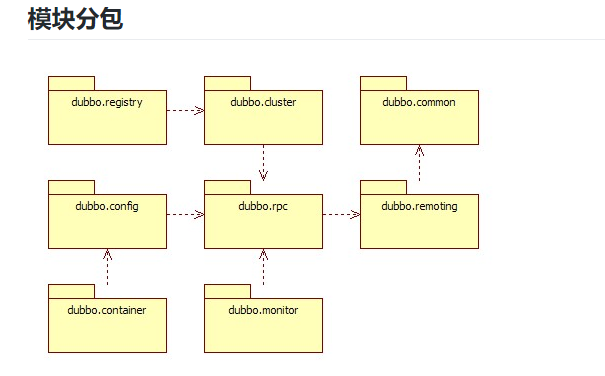








Git上的模块分包



模块说明：

dubbo-common 公共逻辑模块：包括 Util 类和通用模型。

dubbo-remoting 远程通讯模块：相当于 Dubbo 协议的实现，如果 RPC 用 RMI协议则不需要使用此包。

dubbo-rpc 远程调用模块：抽象各种协议，以及动态代理，只包含一对一的调用，不关心集群的管理。

dubbo-cluster 集群模块：将多个服务提供方伪装为一个提供方，包括：负载均衡, 容错，路由等，集群的地址列表可以是静态配置的，也可以是由注册中心下发。

dubbo-registry 注册中心模块：基于注册中心下发地址的集群方式，以及对各种注册中心的抽象。

dubbo-monitor 监控模块：统计服务调用次数，调用时间的，调用链跟踪的服务。

dubbo-config 配置模块：是 Dubbo 对外的 API，用户通过 Config 使用D ubbo，隐藏 Dubbo 所有细节。

dubbo-container 容器模块：是一个 Standlone 的容器，以简单的 Main 加载 Spring 启动，因为服务通常不需要 Tomcat/JBoss 等 Web 容器的特性，没必要用 Web 容器去加载服务。

整体上按照分层结构进行分包，与分层的不同点在于：

container 为服务容器，用于部署运行服务，没有在层中画出。

protocol 层和 proxy 层都放在 rpc 模块中，这两层是 rpc 的核心，在不需要集群也就是只有一个提供者时，可以只使用这两层完成 rpc 调用。

transport 层和 exchange 层都放在 remoting 模块中，为 rpc 调用的通讯基础。

serialize 层放在 common 模块中，以便更大程度复用。

## Dubbo总结