

服务发现开发

联系QQ: 2816010068, 加入会员群

目录

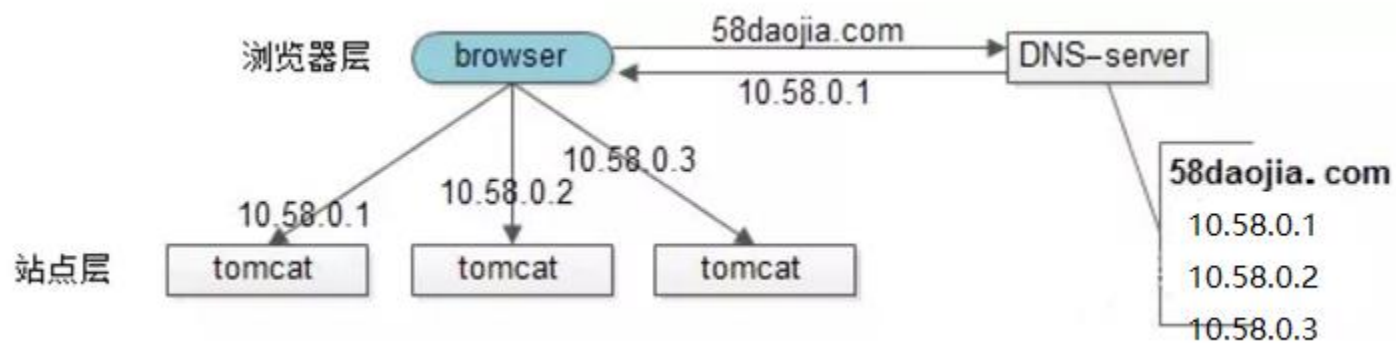
- 使用DNS进行服务发现
- 服务注册&发现原理
- 服务发现接口定义
- 基于etcd的服务发现开发

服务发现的两种方案

- 使用DNS进行服务发现
- 基于SDK的形式进行服务发现

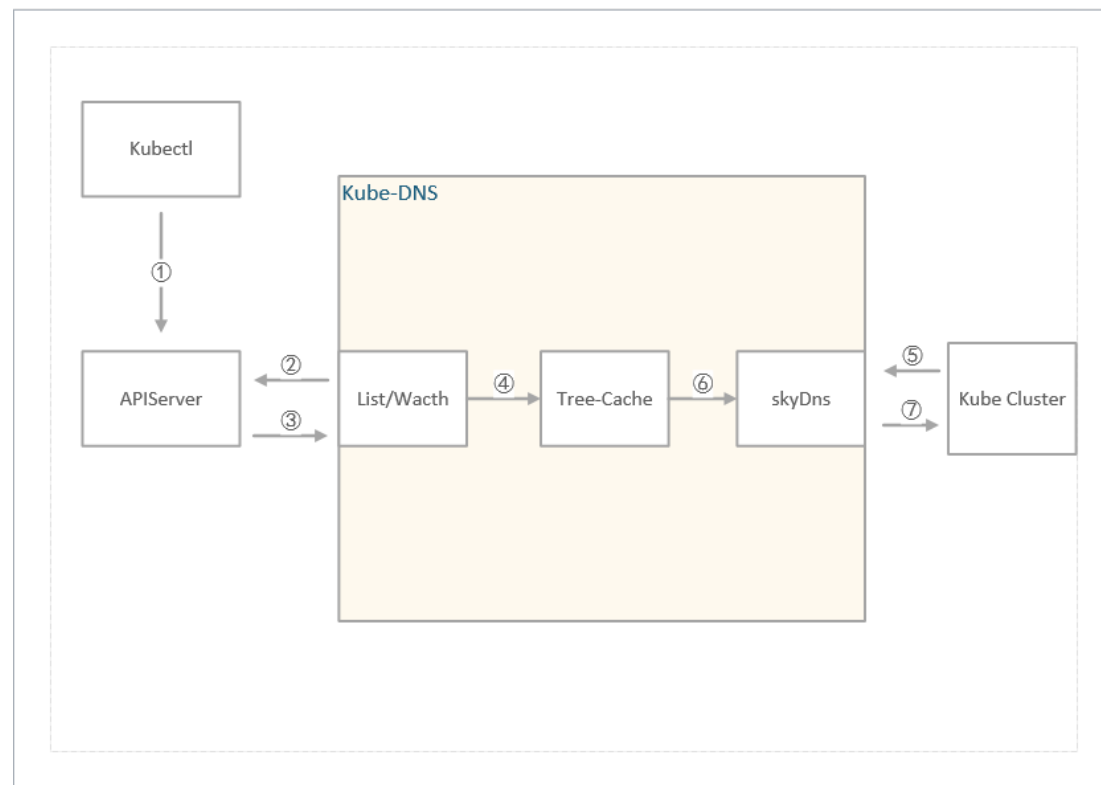
使用DNS进行服务发现

- 传统DNS方案
 - 非高可用
 - 不支持动态变更
 - 域名解析生效慢



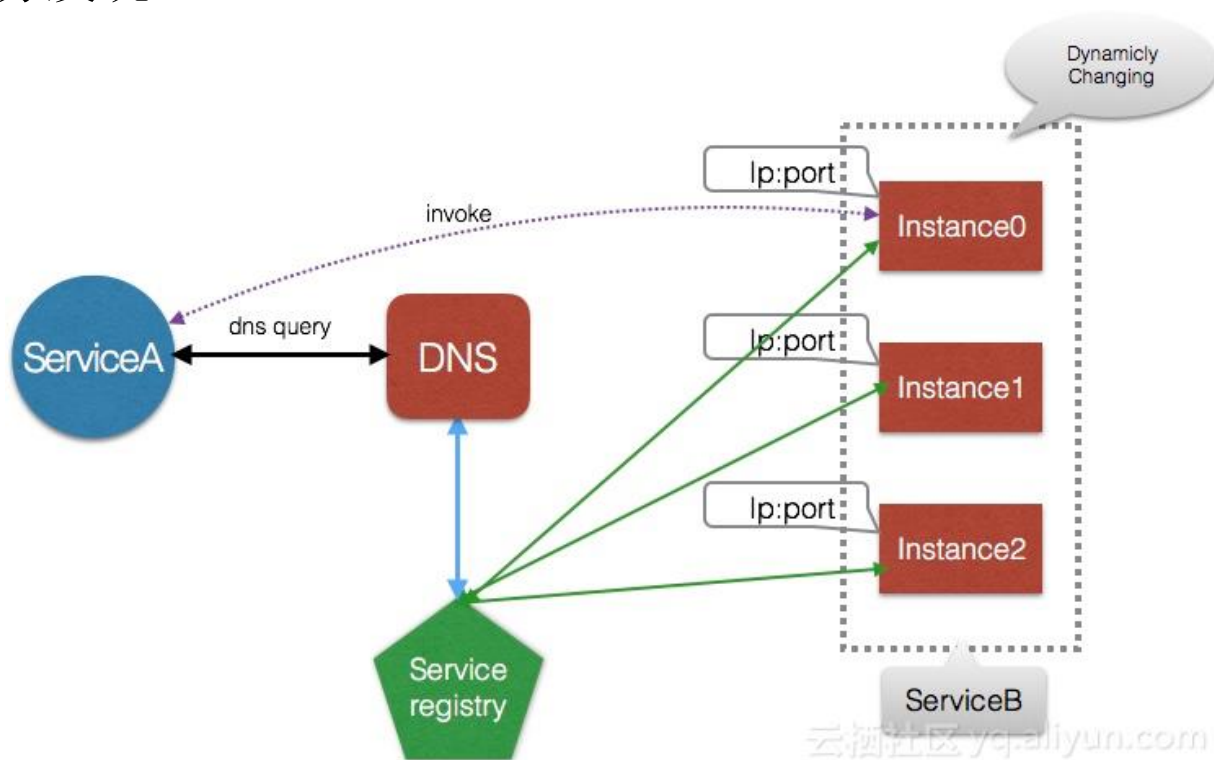
使用dns进行服务发现

- Kube-DNS如何解决这些问题？



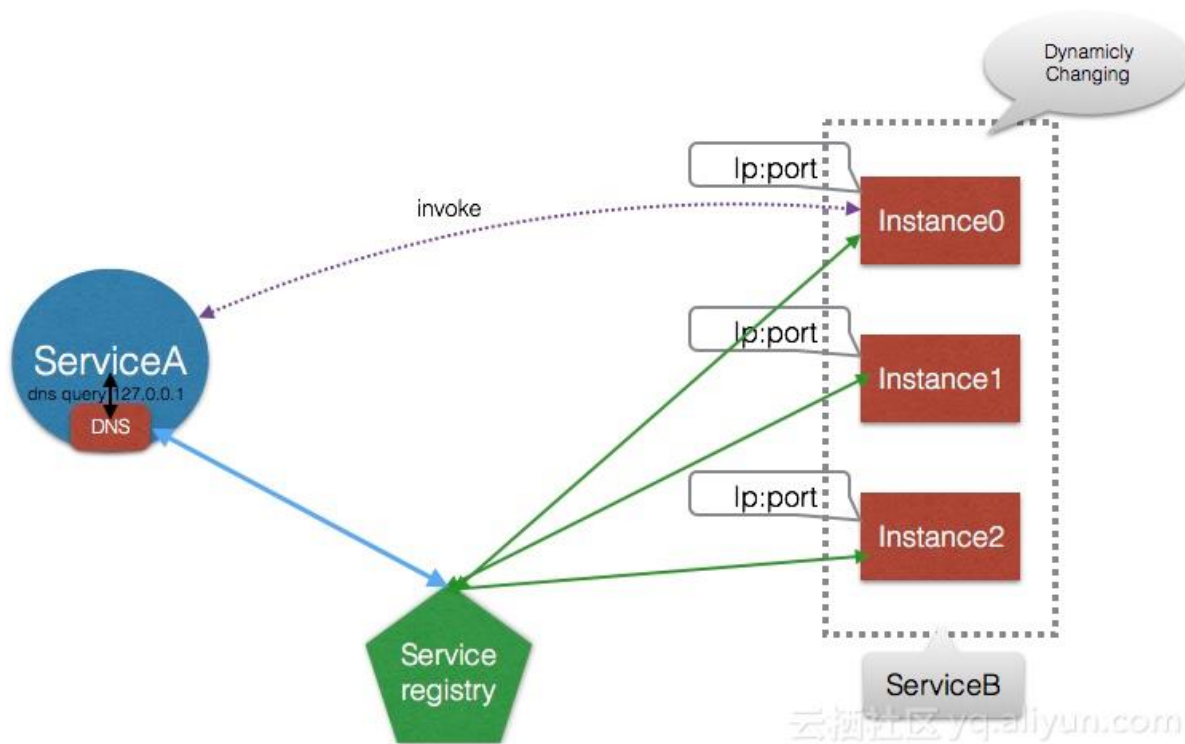
使用DNS进行服务发现

- 集中式的DNS进行服务发现
 - 系统存在单点



使用DNS进行服务发现

- 去掉单点故障

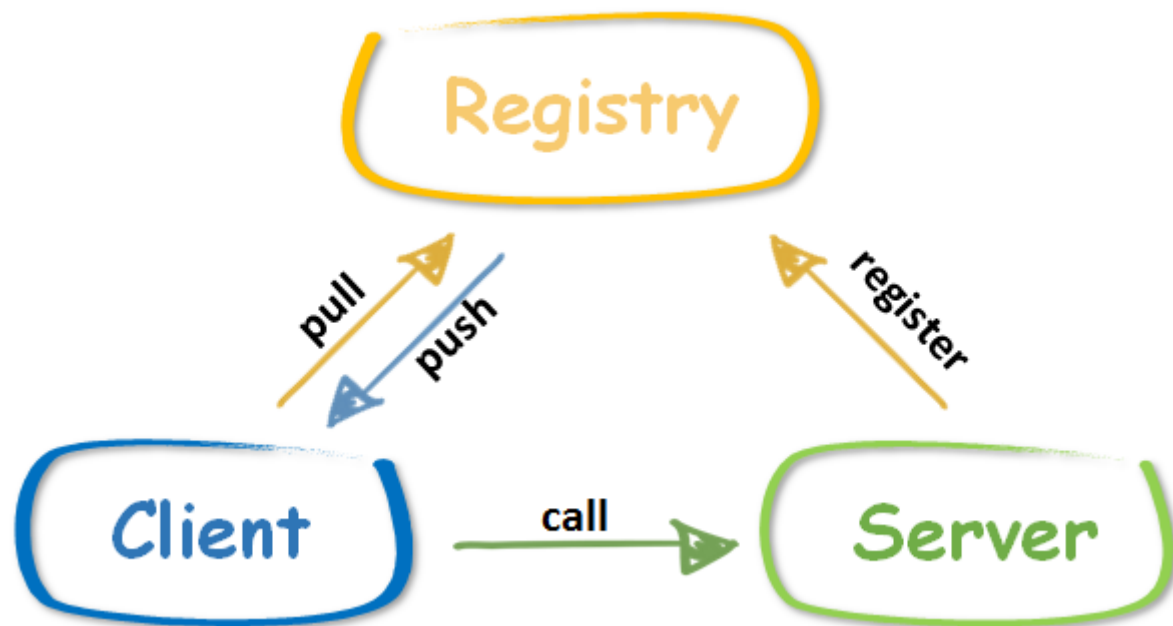


使用dns进行服务发现

- 参考资料

- <https://draveness.me/dns-coredns>
- <https://github.com/coredns/coredns>
- <https://www.cncf.io/community/webinars/introduction-to-coredns/>
- <http://www.servicemeshher.com/blog/ant-financial-service-mesh-adoption-plan/>

服务注册&发现原理



基于sdk的服务发现功能设计

- 目标
 - 不影响主流程
 - 节点准实时更新

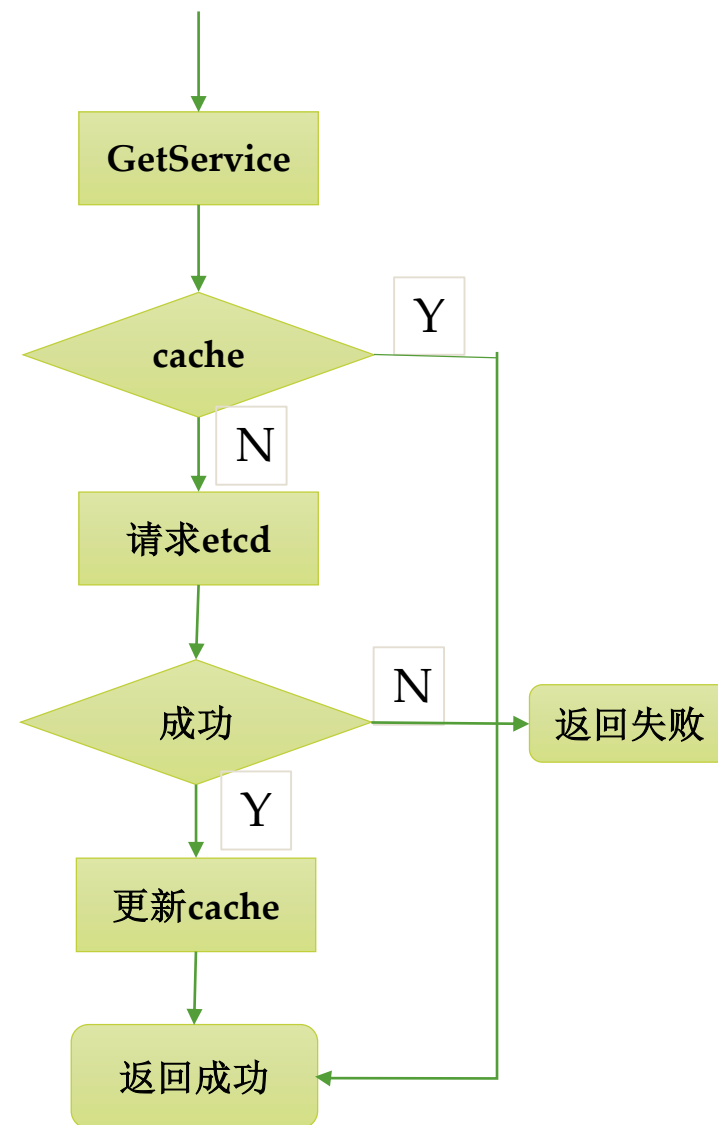
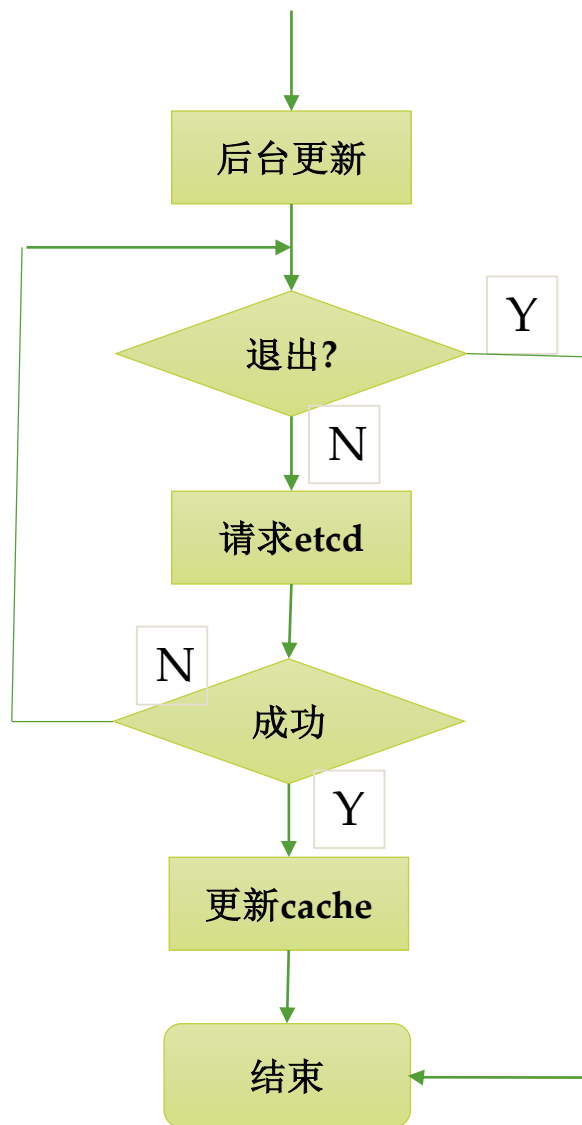
服务发现设计

- 接口定义

```
// 服务注册插件的接口
type Registry interface {
    // 插件的名字
    Name() string
    // 初始化
    Init(ctx context.Context, opts ...Option) (err error)
    // 服务注册
    Register(ctx context.Context, service *Service) (err error)
    // 服务反注册
    Unregister(ctx context.Context, service *Service) (err error)
    // 服务发现: 获取服务名对应的服务
    GetService(ctx context.Context, name string) (service *Service, err error)
}
```

服务发现

- GetService核心流程



基于sdk的服务发现功能开发

- 基于etcd进行开发