

# Kubernetes Informer

## 基本概念



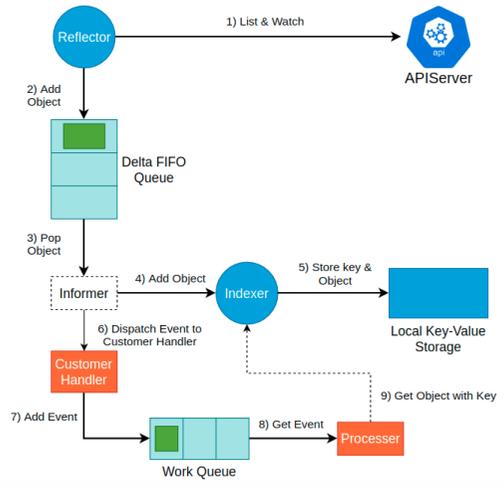
## 原理

List & Watch

Informer 的实现原理主要依赖于 API Server 的 List & Watch 机制, 其中 List 即表示查询某个资源的所有实例, 例如 List Pods, List Deployments, 属于标准 REST API 接口。

而 Watch 机制其实就是一种监听机制, 用户可通过 Watch API 来和 API Server 建立 HTTP 长连接, 当某个资源发生创建、更新或者是删除等事件时, API Server 会将这些事件通过建立的长连接推送至客户端, 使用 HTTP/1.1 分块传输实现

关于 Watch API 的具体格式, 可查阅官方文档 <https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.18/#strong-api-overview-strong->



首先, Reflector 包将会负责和 APIServer 建立长连接, 并使用 Watch API 监听资源的变化, 每一个变化都称之为一个“增量”(Delta), 将会被放置于一个称之为增量队列的先进先出队列中

其次, Informer 从该队列中 pop 事件进行处理, 如果是创建事件, 则在本地缓存中创建, 如果是更新事件, 则更新本地缓存, 删除事件亦然

除了更新本地缓存以外, Informer 还会根据用户定义的 AddFunc、UpdateFunc 以及 DeleteFunc 进行回调, 以此实现自定义控制器

Informer 的原理并不是特别的复杂, 只需要理解其两个作用: 缓存和实现自定义控制器, 缓存的目的在于减少 API Server 以及 Etcd 的负载, 使得可以通过 Local Storage 来获得集群信息