Micro课程-基础篇

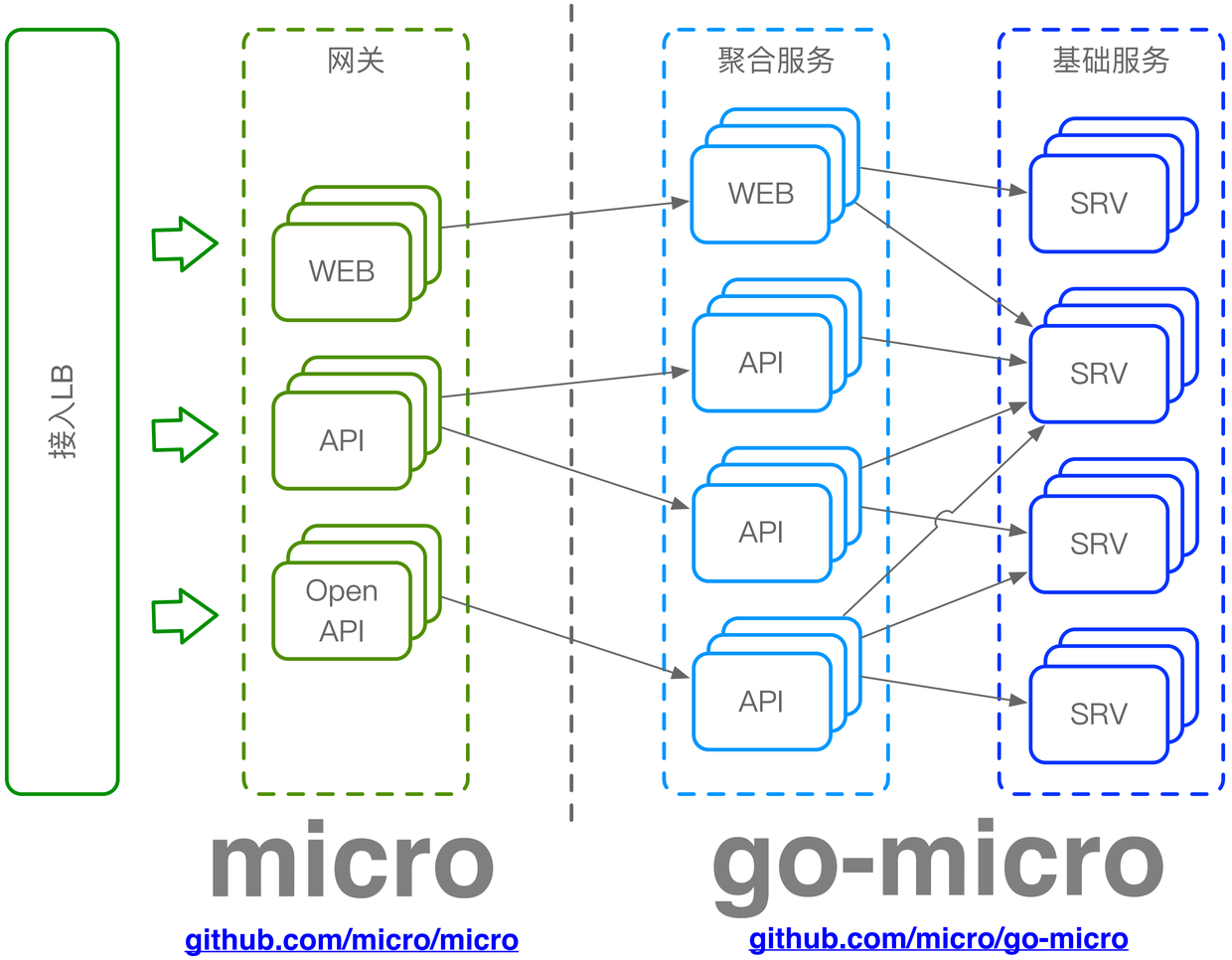
Micro中国站

微服务网关

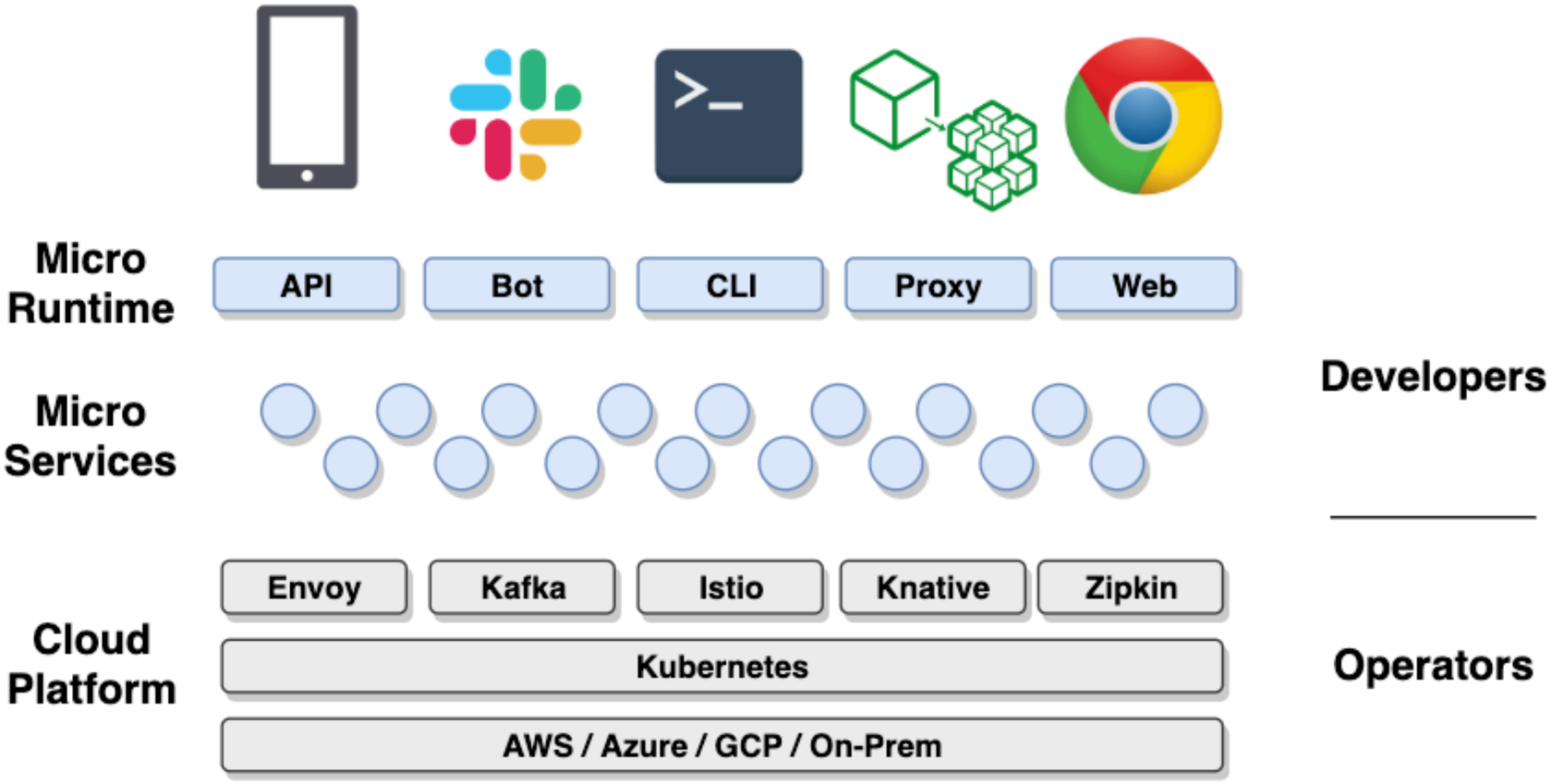
Micro中国 陈洪波

- 微服务架构-网关
- micro api/web
- 自定义网关

微服务架构-网关


接入LB
网关
WEB
API
Open
API
聚合服务
WEB
API
API
API
基础服务
SRV
SRV
SRV
SRV
micro
github.com/micro/micro
go-micro
github.com/micro/go-micro

Micro生态


Micro Runtime
API
Bot
CLI
Proxy
Web
Micro Services
Cloud Platform
Envoy
Kafka
Istio
Knative
Zipkin
Kubernetes
AWS / Azure / GCP / On-Prem
Developers
Operators

演示: 启动网关

micro api / micro web

https://github.com/micro-in-cn/learning-videos/tree/master/docs/Micro%20API

micro命令

- micro -h
- micro api -h
- micro web -h

global options

- --registry
  - Registry for discovery. etcd, mdns
- --registry_address
  - Comma-separated list of registry addresses
- --server_name
  - Name of the server. go.micro.srv.example
- --transport
  - Transport mechanism used; http
- ……

command options

- --address
  - Set the api address e.g 0.0.0.0:8080
- --handler
  - Specify the request handler to be used for mapping HTTP requests to services; {api, event, http, rpc}
- --namespace
  - Set the namespace used by the API e.g. com.example.api
- --resolver
  - Set the hostname resolver used by the API {host, path, grpc}
- --enable_rpc
  - Enable call the backend directly via /rpc

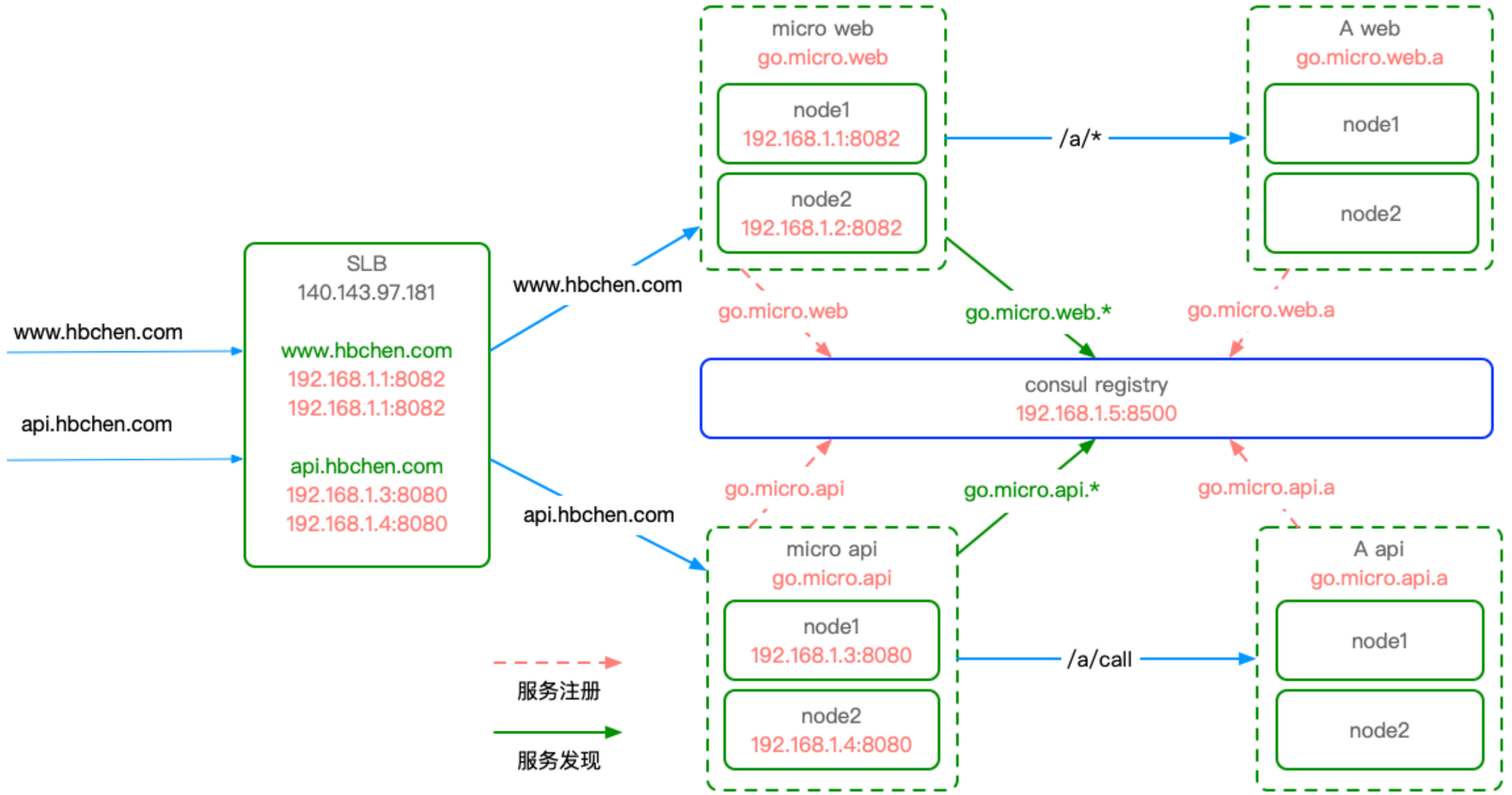
演示: options

micro api / micro web

https://github.com/micro-in-cn/tutorials/tree/master/examples/basic-practices/micro-api

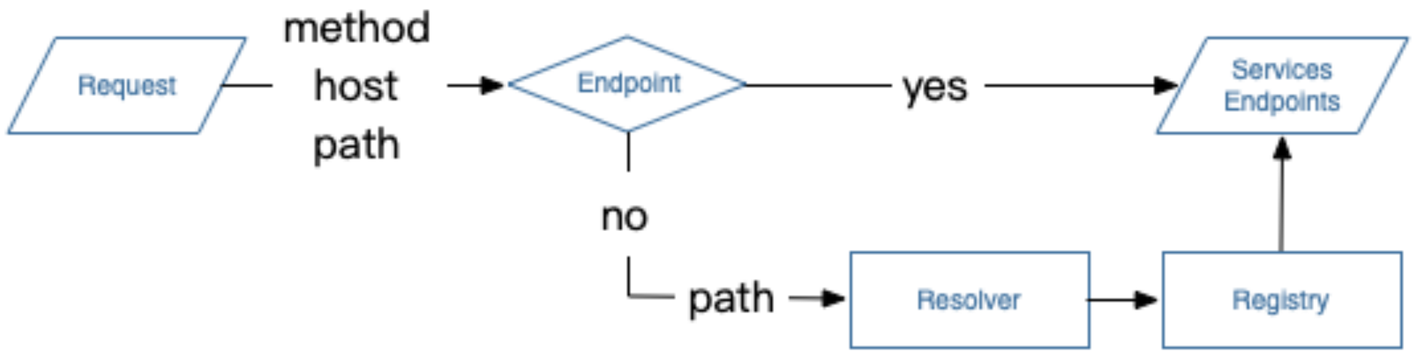
服务发现

- go.micro.api.*
- go.micro.web.*



路由

- 自定义endpoint
- Resolver



请求路径	后台服务	接口方法
/foo/bar	go.micro.api.foo	Foo.Bar
/v1/foo/bar	go.micro.api.v1.foo	Foo.Bar
/v1/foo/bar/baz	go.micro.api.v1.foo	Bar.Baz
/v2/foo/bar	go.micro.api.v2.foo	Foo.Bar
/v2/foo/bar/baz	go.micro.api.v2.foo	Bar.Baz

api handler

类型	说明
rpc	通过RPC向go-micro应用转送请求，只接收GET和POST请求，GET转发`RawQuery`，POST转发`Body`
api	与rpc差不多，但是会把完整的http头封装向下传送，不限制请求方法
http或proxy	以反向代理的方式使用**API**，相当于把普通的web应用部署在**API**之后，让外界像调api接口一样调用web服务
web	与http差不多，但是支持websocket
event	代理event事件服务类型的请求
meta*	默认值，元数据，通过在代码中的`Endpoint`配置选择使用上述中的某一个处理器，默认RPC

演示: API Handler

micro api —handler=api / micro api —handler=proxy

https://github.com/micro-in-cn/tutorials/tree/master/examples/basic-practices/micro-api

自定义网关

- Copy micro/main.go
  - 适合简单定制，如增加插件、go-micro组件

- Fork
  - 需要修改网关源码

- 不管需求大小都建议在项目中自己编译`micro`工具
  - 确保开发、生产等环境一致

go-micro组件

- registry
  - consul、kubernetes
- transport
  - tcp、grpc

- 有关k8s使用kubernetes注册中心的问题可以参考此文
  - http://hbchen.com/post/microservice/2019-06-27-go-micro-use-kubernetes-registry/

演示: Registry&Transport

micro/go-micro

https://github.com/micro-in-cn/learning-videos/tree/master/docs/Micro%20API/example

plugin

- 限流熔断
- CORS
- 认证鉴权
- 日志
- 监控
- 链路追踪
- 流量染色
- ……

micro/plugin


```
type Plugin interface {
  // Global Flags
  Flags() []cli.Flag
  // Sub-commands
  Commands() []cli.Command
  // Handle is the middleware handler for HTTP requests. We pass in
  // the existing handler so it can be wrapped to create a call chain.
  Handler() Handler
  // Init called when command line args are parsed.
  // The initialised cli.Context is passed in.
  Init(*cli.Context) error
  // Name of the plugin
  String() string
}
```

演示: Metrics

micro/micro plugin.Plugin

https://github.com/micro-in-cn/learning-videos/tree/master/docs/Micro%20API/example