

课程介绍

2018-11-23 8:54

1 微服务介绍与概念

2 protobuf

3 grpc

4 consul

5 框架micro

6 租房网的业务

操纵 fastdfs

网站短信验证

。 。 。

7 部署 docker

8 k8s的集群部署 基本介绍

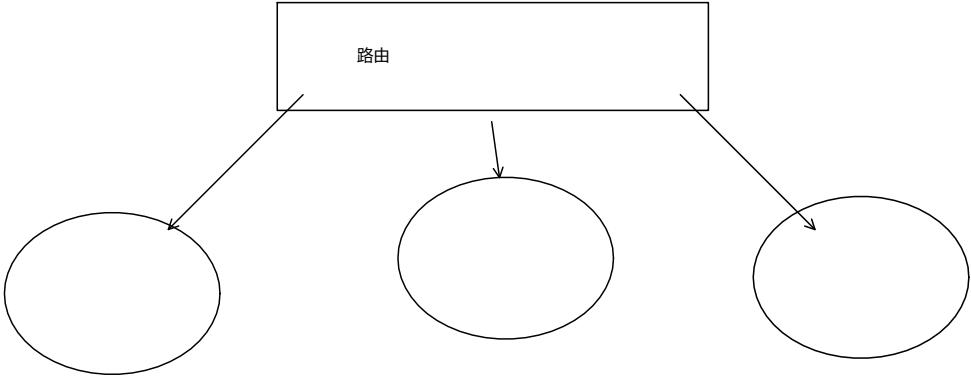
微服务

2018-11-23 9:33

微服务

微服务架构

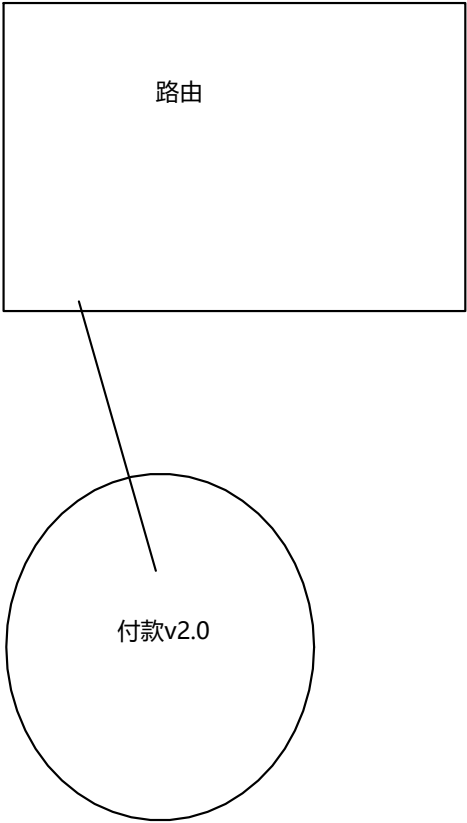
微服务框架：micro



单体式开发的缺点

2018-11-23 9:56

- 1 复杂性逐渐变高
- 2 技术债务逐渐上升
- 3 维护成本大
- 4 持续交付周期长
- 5 可扩展性差



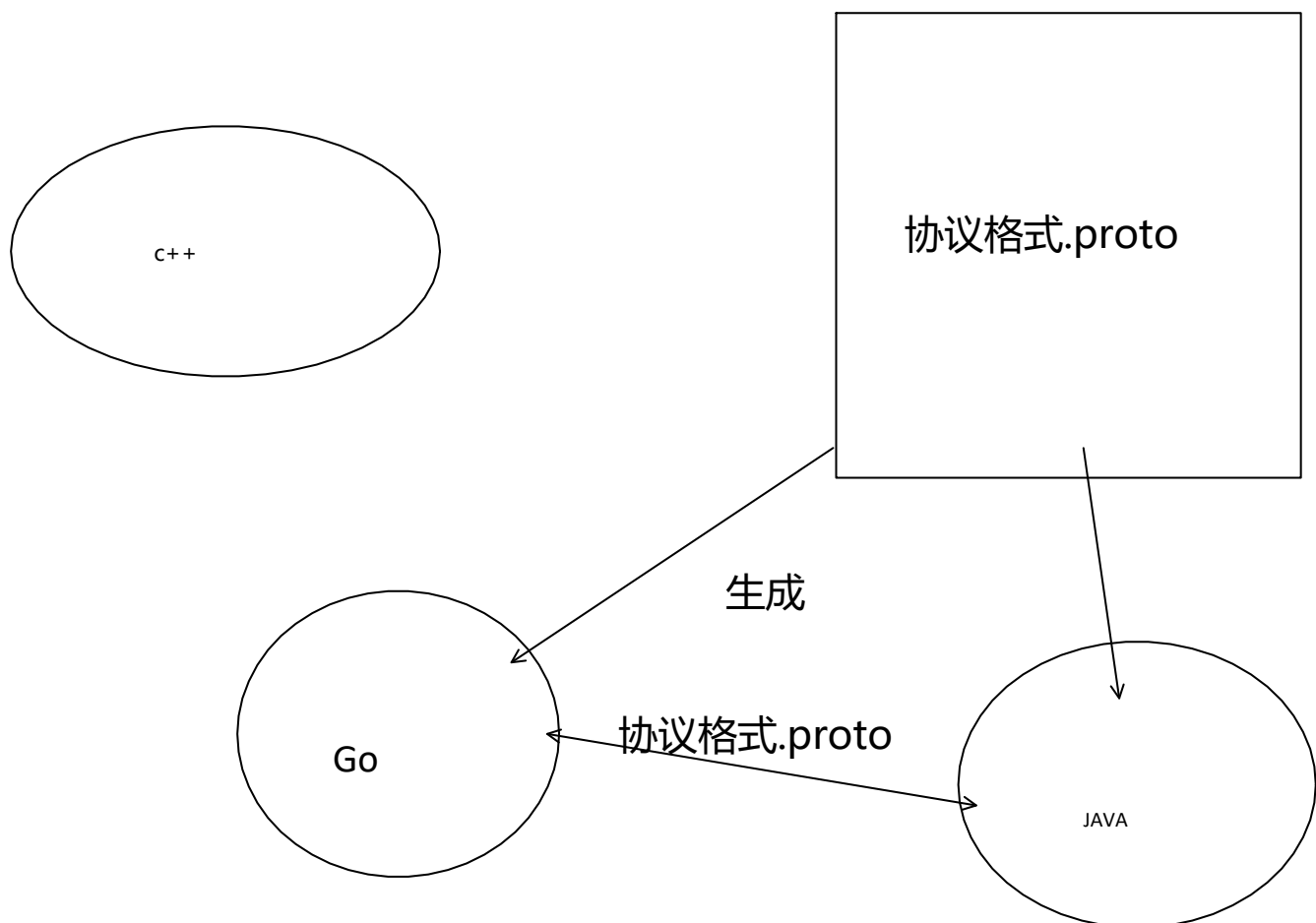
微服务课程的几个重要的组件

2018-11-23 10:41

跨语言，跨平台 通讯格式	protobuf
通讯协议	gRPC
调度管理服务发现	consul
微服务的框架	micro
部署	docker

protobuf

2018-11-23 10:48



protobuf安装

2018-11-23 11:29

下载protobuf

git clone <https://github.com/protocolbuffers/protobuf.git>

或者将准备好的压缩包进行拖入

解压

Unzip protobuf.zip

安装依赖库

\$ sudo apt-get install autoconf automake libtool curl make g++ unzip libffi-dev -y

进入protobuf文件

Cd protobuf

进行检测

./autogen.sh

./configure

进行编译

make

进行安装

sudo make install

刷新共享库

sudo ldconfig

测试

protoc -h

获取 proto包

go get -v -u github.com/golang/protobuf/proto

或者将 github.com-golang-protobuf.zip拖入 进行解压到 \$GOPATH/src/github.com/golang
进入到文件夹内进行编译

\$ cd \$GOPATH/src/github.com/golang/protobuf/protoc-gen-go/

\$ go build

#将生成的 protoc-gen-go可执行文件，放在/bin目录下

```
$ sudo cp protoc-gen-go /bin/
```



```
var pd int
/*
func (client *Client) Call(serviceMethod string, args interface{}),
*/
err =cli.Call( serviceMethod: "Panda.Getinfo", args: 10086,&pd)
if err!=nil {
    fmt.Println( a: "打call失败")
}

fmt.Println( a: "最后得到的值为:",pd)
```

```

25 type Panda int
26 // 打印服务端返回的内容，返回给对端内容，错误返回
27 func (this *Panda)Getinfo(argType int, replyType *int)error{
28     fmt.Println( a: "打印对端发送过来的内容为:",argType)
29     // 修改内容
30     *replyType = argType+ 12306
31     return nil
32 }
```

GRPC环境搭建

2018-11-23 17:32

将x.zip文件拖入到虚拟机中

将x.zip 解压到 \$GOPATH/src/golang.org/x 目录下

```
$ unzip x.zip -d /GOPATH/src/golang.org/x
```

-d 是指定解压目录地址

/home/itcast/go/src/golang.org

文件名为x

如果对应目录的文件夹不存在需要自行创建

将google.golang.org.zip 拖入虚拟机

将google.golang.org.zip 解压到 \$GOPATH/src/google.golang.org 目录下

如果对应目录的文件夹不存在需要自行创建