

Tarsjava入门教程

先前准备

- Linux 服务器，或者MacOs电脑
- 安装好Docker 服务
- JDK1.8及以上
- IntelliJ Idea （ 编译工具）

Tars的架构



拉取镜像

```
docker pull tarscloud/framework:latest  
docker pull tarscloud/tars-node:latest  
docker pull mysql:5.6
```


创建虚拟网络

```
docker network create -d bridge --subnet=172.25.0.0/16 --gateway=172.25.0.1 tars
```


启动框架

```
docker run -d \  
--net=tars \  
-e MYSQL_ROOT_PASSWORD="root@appinside" \  
--ip="172.25.0.2" \  
--name=tars-mysql \  
mysql:5.6  
sleep 30s  
docker run -d \  
--net=tars \  
-e MYSQL_HOST=172.25.0.2 \  
-e MYSQL_ROOT_PASSWORD='root@appinside' \  
-eREBUILD=false -eSLAVE=false \  
-e INET=eth0 \  
--ip="172.25.0.4" \  
-p 3000-3001:3000-3001 \  
tarscloud/framework
```


平台启动

```
docker run -d --net=tars --ip="172.25.0.3" -eWEB_HOST=http://172.25.0.4:3000 tarscloud/tars-node
```


实际操作

编写Jce

```
module TestApp  
{
```

```
    struct TestInfoEx  
    {  
        0 optional TestInfo bi;  
        1 optional vector<TestInfo> vbi;  
        2 optional map<string, TestInfo> mbi;  
    };
```

```
    interface Hello  
    {  
        string hello(int no, string name);  
        int helloJson(TestInfo tie, out TestInfoEx otie);  
    };
```

```
}
```


生成代码

操作发布服务

生成客户端

发布代码

查看日志

新建一个 Http

此外如果你想使用spring-boot来编写一个http服务, 而不使用tars接口的话也是可以的:

```
```java
@SpringBootApplication
@EnableTarsServer
@TarsHttpService("HttpObj")
@RestController
public class DemoApplication {
 public static void main(String[] args) {
 SpringApplication.run(DemoApplication.class, args);
 }

 @RequestMapping(path = "/test")
 public String test() {
 return "hello world";
 }
}
```



发布到平台



增加数据库读写操作



输出数据库内容



结束