1. 客户端负载均衡：Spring Cloud Ribbon

1 客户端负载均衡

●负载均衡的算法（比如线性轮询、按权重负载、按流量负载等）从维护的可用服务端清单中取出一台服务端的地址，然后进行转发。

●客户端负载均衡和服务端负载均衡最大的不同点在于上面所提到的服务清单所存储的位置。

●客户端负载均衡中，所有客户端节点都维护着自己要访问的服务端清单，而这些服务端的清单来自于服务注册中心。

●Ribbon配置：

org.springframework.cloud.netflix.ribbon.eureka.RibbonEurekaAutoConfiguration和

org.springframework.cloud.consul.discovery.RibbonConsulAutoConfiguration

●微服务架构中使用客户端负载均衡调用步骤：

1. 服务提供者只需要启动多个服务实例并注册到一个注册中心或是多个相关联的服务注册中心
2. 服务消费者直接通过调用@LoadBalanced注解修饰过的RestTemplate来实现面向服务的接口调用。

2 RestTemplate详解

●GET请求：

1. getForEntity函数【该对象是Spring对HTTP请求响应的封装】

例子：

RestTemplate restTemplate = new RestTemplate();

ResponseEntity<String> responseEntity = restTemplate.getForEntity (“http://USER-SERVICE/user?**Name=[1]**”, String.class, “didi”);

String body = responseEntity.getBody();

※“didi”所在位置是urlVariables参数，urlVariables参数是一个数组，所以顺序会对应url中占位符定义的数字顺序。

※urlVariables可以是Map类型，参数应该是一个key，如，name={name}

※getForEntity(URI uri, Class responseType);

1. getForObject函数【对getForEntity函数的再封装】

例子：

RestTemplate restTemplate = new RestTemplate();

String result = restTemplate.getForObject(uri, String.class);

※getForObject(String url, Class responseType, Object... urlVariables);

※getForObject(String url, Class responseType, Map urlVariables);

※getForObject(URI url, Class responseType);

●POST请求：

1. postForEntity函数
2. postForObject函数

●PUT请求：

例子：

RestTemplate restTemplate = new RestTemplate();

Long id = 10001L;

User user = new User(“didi”,40);

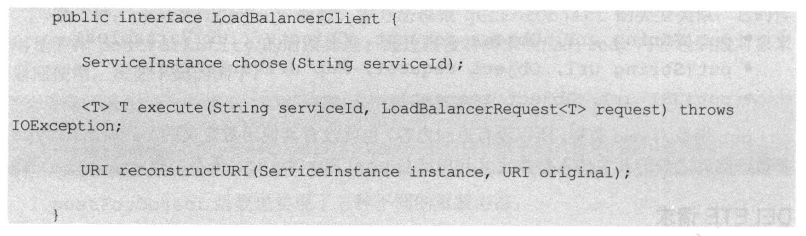
restTemplate.put(“http://USER-SERVICE/user/{1}”,user,id);

※put(String url, Class responseType, Object... urlVariables);

※put(String url, Class responseType, Map urlVariables);

※put(URI url, Class responseType);

20180323T16:04 @LoadBalanced源码分析





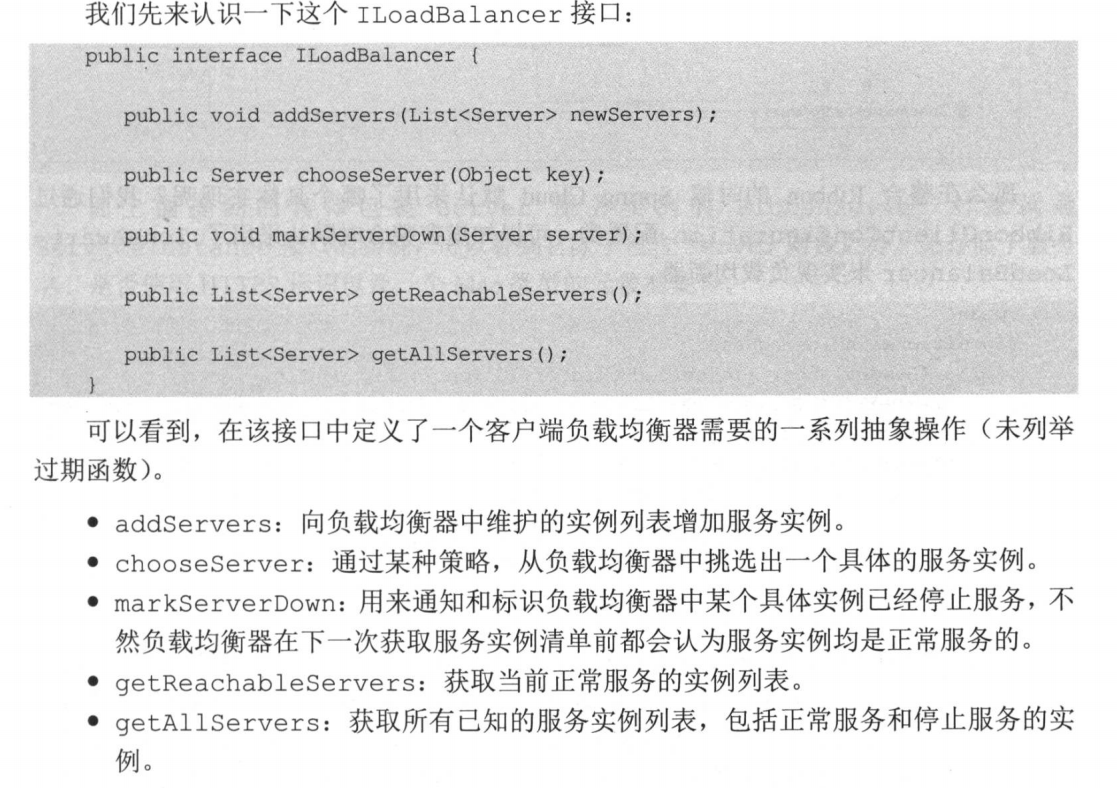
1 Ribbon实现的负载均衡自动化配置需要满足下面两个条件：

1. @ConditionalOnClass(RestTemplate.class): RestTemplate类必须存在于当前工程的环境中
2. @ConditionalOnBean(LoadBalancerClient.class): 在Spring的Bean工程中必须有LoadBalancerClient的实现Bean

2 自动化配置类中，主要做了三件事：

1. 创建了一个LoadBalancerInterceptor的Bean，用于实现对客户端发起请求时进行拦截，以实现客户端负载均衡
2. 创建了一个RestTemplateCustomizer的Bean，用于给RestTemplate增加LoadBalancerInterceptor拦截器
3. 维护了一个被@LoadBalanced注解修饰的RestTemplate对象列表，并在这里进行初始化，通过调用RestTemplateCustomizer的实例来给需要客户端负载均衡的RestTemplate增加LoadBalancerInterceptor拦截器

ILoadBalancer接口



地址转换

