

서비스 연결

📅 Created	@2021년 11월 10일
☰ Tags	micro service
▼ 스터디	학교

서비스 디스커버리

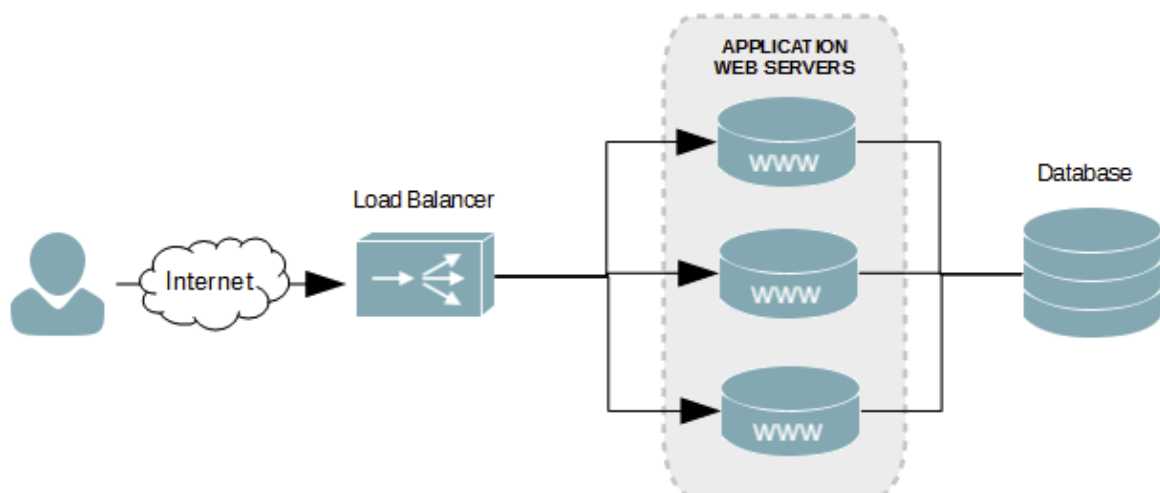
마이크로 서비스를 호출하기 위해서 서비스의 주소를 관리하는 시스템이 필요하다.

- 서비스 등록: 서비스를 서비스 디스커버리에 등록한다.
- 서비스 주소 검색: 등록된 서비스의 주소를 검색한다.
- 정보 공유: 서비스의 정보를 노드 간에 공유할 수 있다.
- 상태 모니터링: 서비스가 자신의 상태 정보를 서비스 디스커버리에 전달한다.

엔드 포인트 호출

1. Spring Discovery: 디스커버리 클라이언트인 표준 스프링 RestTemplate 클래스를 사용하여 서비스 호출
2. RestTemplate: 스프링 RestTemplate와 넷플릭스 리본을 사용하여 서비스 호출
3. Netflix Feign: 넷플릭스 Feign 클라이언트와 넷플릭스 리본을 사용하여 서비스 호출

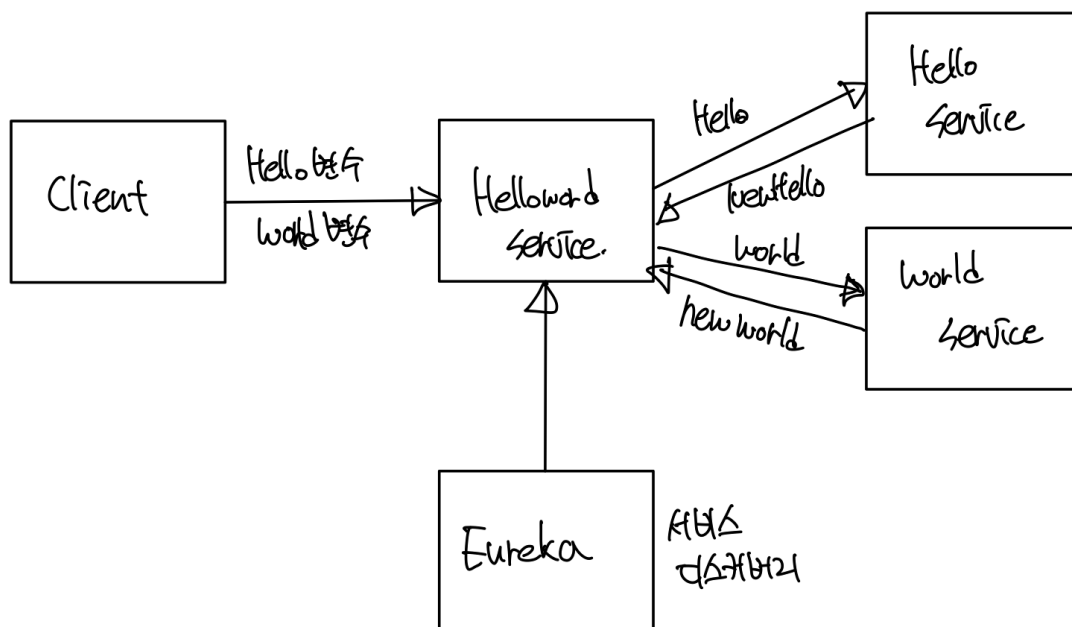
Ribbon



구성요소

- Rule
 - 요청을 보낼 서버를 선택하는 정책
 - Round Robbin: 돌아가면서 실행
 - Available Filtering: 에러가 많은 서버 제외
 - Weighted Response: 서버별 응답 시간에 따라 확률 조절
- Ping:
 - 서버리스트가 살아 있는지 체크
- ServerList:
 - static: configuration을 통해서 관리
 - dynamic: eureka와 같은 디스커버리를 사용하여 관리

구조



helloworld 서비스 호출 코드

```

14 @RestController
15 @RequestMapping(value = "api")
16 public class HomeworkController {
17     private static final Logger logger = LoggerFactory.getLogger(HomeworkController.class);
18
19     @Autowired
20     private HelloRestTemplateClient helloRestTemplateClient;
21
22     @Autowired
23     private WorldRestTemplateClient worldRestTemplateClient;
24
25     @RequestMapping(value = "helloworld", method = RequestMethod.GET)
26     public String helloworld(@RequestParam String hello, @RequestParam String world){
27         String newHello = helloRestTemplateClient.getHello(hello);
28         logger.info("hello: " + newHello);
29         String newWorld = worldRestTemplateClient.getWorld(world);
30         logger.info("world: " + newWorld);
31         return newHello + newWorld;
32     }
33 }
34

```

restTemplate

```

9 @Component
10 public class HelloRestTemplateClient {
11     @Autowired
12     private RestTemplate restTemplate;
13
14     public String getHello(String hello){
15         ResponseEntity<String> restExchange = restTemplate.exchange(
16             "http://licensing-service/license/v1/organizations/test0rg/licenses/hello?hello={hello}",
17             HttpMethod.GET, null, String.class, hello);
18
19         return restExchange.getBody();
20     }
21 }

```

호출과 결과

GET localhost:1234/api/helloworld?hello=test&world=world

Params ● Authorization Headers (7) Body Pre-request Script Test

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	hello	test
<input checked="" type="checkbox"/>	world	world
	Key	Value

Body Cookies (1) Headers (5) Test Results

Pretty Raw Preview Visualize Text ▾ ↺

1 TESTWORLD

helloworld 서버

```
2021-11-10 03:52:36.778 INFO 42238 --- [nio-1234-exec-5] c.e.h.controller.HomeworkController : hello: TEST
2021-11-10 03:52:36.794 INFO 42238 --- [nio-1234-exec-5] c.e.h.controller.HomeworkController : world: WORLD
```

hello 서버

```
2021-11-10 03:52:36.775 INFO 39708 --- [nio-8080-exec-9] c.e.l.c.LicenseServiceCon
troller : license server : test
```

world 서버

```
2021-11-10 03:52:36.784 INFO 39706 --- [io-8085-exec-10] c.e.o.controller.Organiza
tionController : organization server: world
```

