

Day10 CSE实战之微服务线程模型和性能统计

1 打卡任务

作业：

- 1、开启metrics模块查看服务的性能数据。
- 2、在consumer服务中增加一个reactive的REST方法，通过reactive模式调用provider服务

打卡：

- 1、截图consumer日志中的metrics统计日志
- 2、调用consumer的reactive方法成功并截图

打卡任务基于Day10的demo项目：



2 准备工作

- 1、正常运行Day10的demo

3 开启 Metrics

此处以consumer服务为例，其他服务也是用同样的方法开启metrics

- 1、在pom文件中引入metrics的依赖：

```

<dependencies>
  <dependency>
    <groupId>com.huawei.paas.cse</groupId>
    <artifactId>cse-solution-service-engine</artifactId>
  </dependency>
  <dependency>
    <groupId>org.apache.servicecomb</groupId>
    <artifactId>metrics-core</artifactId>
  </dependency>
</dependencies>

```

2、在microservice.yaml文件中配置开启metrics日志：

```

cse:
  metrics:
    publisher:
      defaultLog:
        enabled: true # 是否在默认的日志中打印metrics日志
        window_time: 10000 # metrics日志打印周期

```

此处配置的是每10秒钟打印一次metrics日志

3、运行edge、consumer、provider服务，调用consumer服务，观察日志输出的信息

```

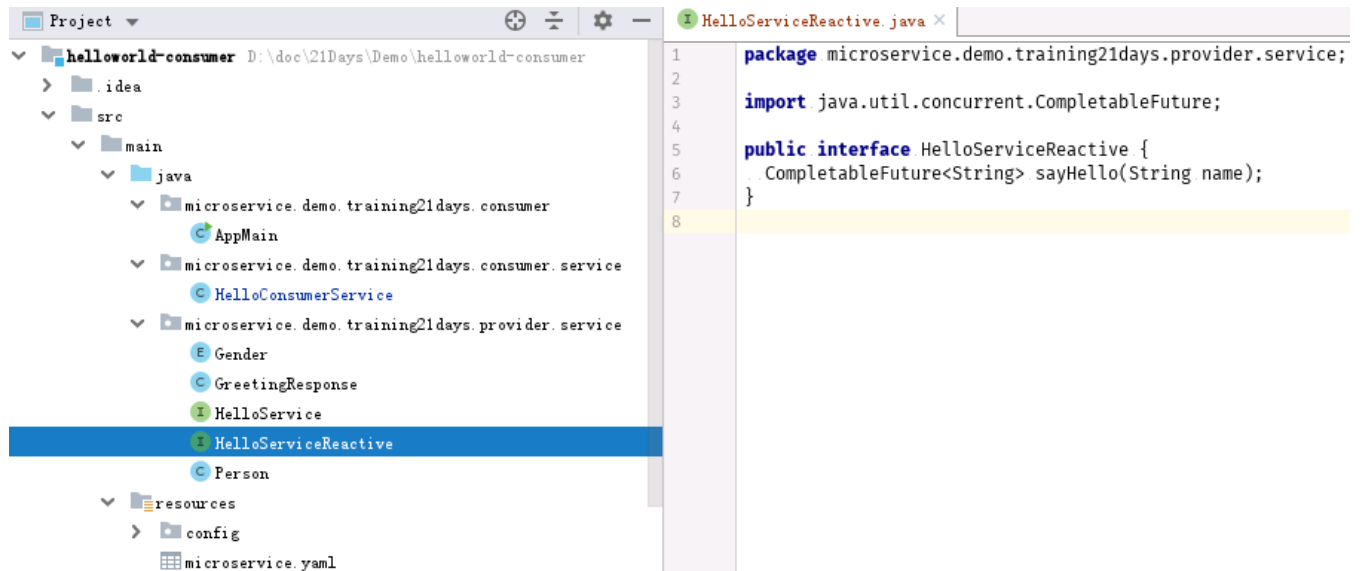
vertx:
  instances:
    name.....eventLoopContext-created
    registry...2
    transport...10
    monitor-center 2
    config-center 2
  transport:
    client.endpoints:
      remote.....connectCount.....disconnectCount.....connections.....send(Bps).....receive(Bps)
      (summary).....0.....0.....1.....0.....756
    server.endpoints:
      listen.....connectCount.....disconnectCount.....rejectByLimit.....connections.....send(Bps).....receive(Bps)
      0.0.0.0:9090.....0.....0.....0.....1.....756.....0
      (summary).....0.....0.....0.....1.....756.....0
  threadPool:
    corePoolSize maxThreads poolSize currentThreadsBusy queueSize taskCount completedTaskCount name
    8.....8.....0.....0.....0.....0.0.....0.0.....cse.executor.groupThreadPool-group1
    8.....8.....8.....0.....0.....25.6.....25.6.....cse.executor.groupThreadPool-group0
  consumer:
    simple:
      status.....tps.....latency.....operation
      rest.200.....25.....2.429/6.217.....provider.hello.sayHello
      25.....2.429/6.217.....(summary)
    details:
      rest.200:
        provider.hello.sayHello:
          prepare.....: 0.009/0.021.....handlersReq : 0.506/4.238.....cFiltersReq: 0.014/0.035.....sendReq.....: 0.398/0.674
          getConnect : 0.224/0.493.....writeBuf.....: 0.174/0.301.....waitResp...: 1.250/1.746.....wakeConsumer: 0.049/0.372
          cFiltersResp: 0.109/0.209.....handlersResp: 0.091/0.162
  producer:
    simple:
      status.....tps.....latency.....operation
      rest.200.....25.....3.024/6.802.....consumer.helloConsumer.sayHello
      25.....3.024/6.802.....(summary)
    details:
      rest.200:
        consumer.helloConsumer.sayHello:
          prepare: 0.049/0.206.....queue.....: 0.134/0.204.....filtersReq : 0.040/0.082.....handlersReq: 0.111/0.222
          execute: 2.532/6.328.....handlersResp: 0.019/0.086.....filtersResp: 0.084/0.505.....sendResp...: 0.048/0.195

```

从consumer日志中可以看到如上图所示的metrics信息

4 在 consumer 服务中开发一个 reactive 模式的 REST 方法

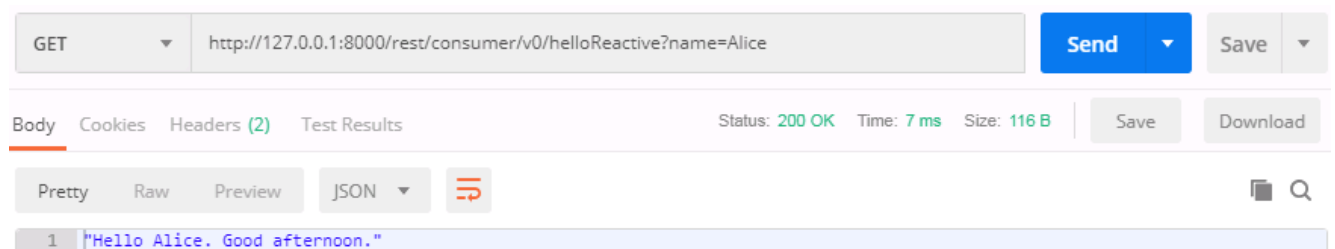
- 1、为了让consumer能够以reactive模式调用provider服务，我们先在consumer服务中定义一个Reactive风格的RPC接口类，供consumer服务发请求时使用



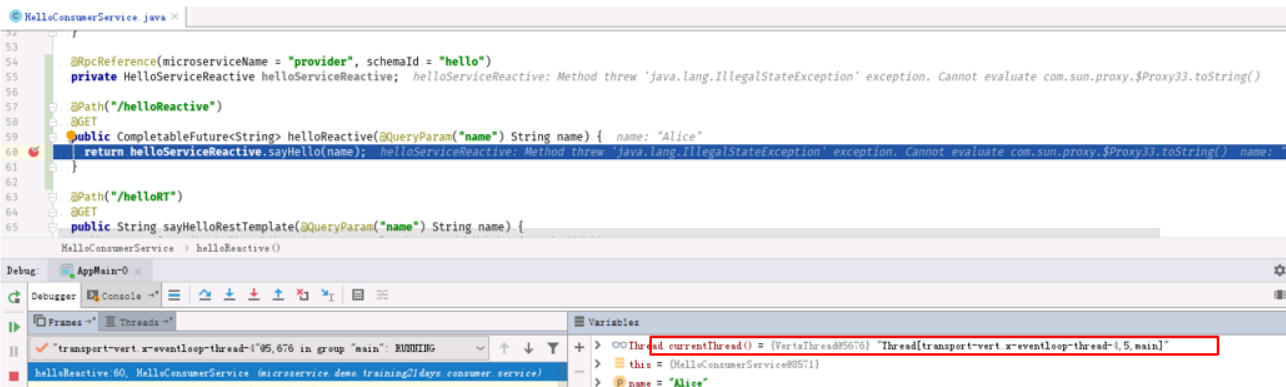
- 2、在HelloConsumerService类中定义reactive风格的REST方法

```
1 @RpcReference(microserviceName = "provider", schemaId = "hello")
2 private HelloServiceReactive helloServiceReactive;
3
4 @Path("/helloReactive")
5 @GET
6 public CompletableFuture<String> helloReactive(@QueryParam("name") String name) {
7     return helloServiceReactive.sayHello(name);
8 }
```

- 3、启动consumer服务，通过edge服务调用consumer的helloReactive方法，调用成功



- 4、如果以debug模式启动consumer服务，在helloReactive方法上打断点，调用consumer的helloReactive方法，可以看到该方法的执行线程是网络线程。即reactive模式的方法默认是执行在网络线程里的。



5 打卡截图

1、Consumer服务的metrics统计日志截图：

```
vertx:
..instances:
...name.....eventLoopContext-created
...registry...2
...transport...10
...monitor-center 2
...config-center 2
..transport:
...client.endpoints:
...remote.....connectCount.....disconnectCount.....connections.....send(Bps).....receive(Bps)
... (summary) .....0.....0.....1.....0.....756
...server.endpoints:
...listen.....connectCount.....disconnectCount.....rejectByLimit.....connections.....send(Bps).....receive(Bps)
... 0.0.0.0:9090.....0.....0.....0.....1.....756.....0
... (summary) .....0.....0.....0.....1.....756.....0
threadPool:
..corePoolSize maxThreads poolSize currentThreadsBusy queueSize taskCount completedTaskCount name
..8.....8.....0.....0.....0.....0.0.....0.0.....cse.executor.groupThreadPool-group1
..8.....8.....8.....0.....0.....25.6.....25.6.....cse.executor.groupThreadPool-group0
consumer:
..simple:
...status.....tps.....latency.....operation
...rest.200.....25.....2.429/6.217.....provider.hello.sayHello
...rest.200.....25.....2.429/6.217.....(summary)
..details:
...rest.200:
...provider.hello.sayHello:
...prepare.....: 0.009/0.021.....handlersReq : 0.506/4.238.....cFiltersReq: 0.014/0.035.....sendReq.....: 0.398/0.674
...getConnect.....: 0.224/0.493.....writeBuf.....: 0.174/0.301.....waitResp.....: 1.250/1.746.....wakeConsumer: 0.049/0.372
...cFiltersResp: 0.109/0.209.....handlersResp: 0.091/0.162
producer:
..simple:
...status.....tps.....latency.....operation
...rest.200.....25.....3.024/6.802.....consumer.helloConsumer.sayHello
...rest.200.....25.....3.024/6.802.....(summary)
..details:
...rest.200:
...consumer.helloConsumer.sayHello:
...prepare: 0.049/0.206.....queue.....: 0.134/0.204.....filtersReq : 0.040/0.082.....handlersReq: 0.111/0.222
...execute: 2.532/6.328.....handlersResp: 0.019/0.086.....filtersResp: 0.084/0.505.....sendResp.....: 0.048/0.195
```

2、调用consumer的reactive方法成功的截图：



GET

http://127.0.0.1:8000/rest/consumer/v0/helloReactive?name=Alice

Send

Save

Body

Cookies

Headers (2)

Test Results

Status: 200 OK

Time: 7 ms

Size: 116 B

Save

Download

Pretty

Raw

Preview

JSON

1

"Hello Alice. Good afternoon."

参考答案:



Demo-Day10-Ho
mework.zip