spring cloud Bus

1.rabbitmq实现消息总线

首先是安装rabbitmq

先安装erlang

再安装rabbitmq(应该是要装在c盘下面)

当然可以根据兴趣配置路径(这里没有配置,使用默认的配置)

然后需要安装rabbitmq的可视化管理界面的插件

在cmd窗口切换到rabbitmq安装目录下的sbin目录下

如下图

```
C:\Users\pc>cd C:\Program Files\RabbitMQ Server\rabbitmq_server-3.6.12

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.6.12>cd sbin

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.6.12\sbin>rabbitmq-plugins enable rabbitmq_management
The following plugins have been enabled:
  amqp_client
  cowlib
  cowboy
  rabbitmq_web_dispatch
  rabbitmq_management_agent
  rabbitmq_management

Applying plugin configuration to rabbit@pc-PC... started 6 plugins.

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.6.12\sbin>_
```

然后访问http://localhost:15672

默认端口号15672

即可进入rabbitmq管理界面

默认账号guest

默认密码guest

建议一下操作都用guest账号即可

有兴趣可以学习rabbitmq的adduser时权限的设置等这里不再赘述

首先可以先实现springboot与rabbitmq的整合

参考http://www.cnblogs.com/hlhdidi/p/6535677.html

另外:测试类写在自动生成的里面就行了

发送端与接收端如果用的

测试时应先启动接收端

2.spring cloud Bus 实现实时刷新git配置文件到所有的服务(实际上这次还是要手动刷新一次)

按照书中所写即可实现

这里主要说明注意事项:

在configclient配置文件必须有rabbitmq的信息即

```
spring.rabbitmq.host=localhost
spring.rabbitmq.port=5672
spring.rabbitmq.username=springcloud
spring.rabbitmq.password=123456
```

3.步骤如下:

- 添加依赖bus-amqp
- 配置信息(书要是amqp的信息)
- sender类

  添加@Component注解 自动注入

  并 注入模板

```
@Component
public class Sender {

    @Autowired
    private AmqpTemplate amqpTemplate;

    public void send(){
        String mess="hello "+new Date();
        System.out.println("Sender "+mess);
        //"hello"为routtingkey
        amqpTemplate.convertAndSend("hello", mess);
    }

}
```

添加配置类

```
@Configuration
public class SenderConf {

    @Bean
    public Queue queue() {
        return new Queue("hello2");
    }
}
```

这个hello就是队列的名字

测试类直接写在springboot自动生成的方法中即可,注解都是自动生成的不再多说,注入Sender类,调用send方法

```
@RunWith(SpringRunner.class)
@SpringBootTest
public class CloudEruakaBusSenderApplicationTests {
    @Autowired
    private Sender sender;

    /*   @Test
    public void testRabbit() {
        sender.send();
    }*/
    @Test
    public void contextLoads() {
        sender.send();
    }

}
```

- receiver应用
- receiver类

```
@Component
public class Reciever {
    @RabbitListener(queues="hello")     //监听器监听指定的Queue
    public void processC(String str) {
        System.out.println("Receive:"+str);
    }
}
```

依赖,配置信息(应用名不同),配置类都相同

测试步骤,先启动两个应用,在用Junit测试启动测试类,效果

接收应用的控制台:

```
Markers Properties Servers Data Source Explorer Snippets Console Progress Debug JUnit
rabbitmqTest2 - RabbitmqTestApplication [Spring Boot App] C:\Program Files\Java\jre1.8.0_121\bin\javaw.exe (2017年9月15日 下午3:46:21)
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::        (v1.5.6.RELEASE)

2017-09-15 15:46:21.882  INFO 4040 --- [           main] c.example.demo.RabbitmqTestApplication   : Starting RabbitmqTestApplication on pc-PC with PID 4040 (E:\cloudTest-workspace\rabbitmqTest2\target\classes sta
2017-09-15 15:46:21.884  INFO 4040 --- [           main] c.example.demo.RabbitmqTestApplication   : No active profile set, falling back to default profiles: default
2017-09-15 15:46:21.908  INFO 4040 --- [           main] s.c.a.AnnotationConfigApplicationContext : Refreshing org.springframework.context.annotation.AnnotationConfigApplicationContext@6a396c1e: startup date [Fri
2017-09-15 15:46:22.164  INFO 4040 --- [           main] trationDelegate$BeanPostProcessorChecker : Bean 'org.springframework.amqp.rabbit.annotation.RabbitBootstrapConfiguration' of type [org.springframework.amqp
2017-09-15 15:46:22.402  INFO 4040 --- [           main] o.s.j.e.a.AnnotationMBeanExporter        : Registering beans for JMX exposure on startup
2017-09-15 15:46:22.406  INFO 4040 --- [           main] o.s.j.e.a.AnnotationMBeanExporter        : Bean with name 'rabbitConnectionFactory' has been autodetected for JMX exposure
2017-09-15 15:46:22.408  INFO 4040 --- [           main] o.s.j.e.a.AnnotationMBeanExporter        : Located managed bean 'rabbitConnectionFactory': registering with JMX server as MBean [org.springframework.amqp.r
2017-09-15 15:46:22.426  INFO 4040 --- [           main] o.s.c.support.DefaultLifecycleProcessor  : Starting beans in phase -2147482648
2017-09-15 15:46:22.426  INFO 4040 --- [           main] o.s.c.support.DefaultLifecycleProcessor  : Starting beans in phase 2147483647
2017-09-15 15:46:22.447  INFO 4040 --- [cTaskExecutor-1] o.s.a.r.c.CachingConnectionFactory       : Created new connection: rabbitConnectionFactory#6bf1f62f:0/SimpleConnection@45b40ffd [delegate=amqp://guest@127.
2017-09-15 15:46:22.469  INFO 4040 --- [           main] c.example.demo.RabbitmqTestApplication   : Started RabbitmqTestApplication in 0.707 seconds (JVM running for 1.078)
Receive:hello Fri Sep 15 15:46:48 CST 2017
```

junit控制台:

```
2017-09-15 15:57:32.656  INFO 9484 --- [           main] o.s.i.endpoint.EventDrivenConsumer       : Adding {message-handler:inbound.springCloudBus.default} as a subscrib
2017-09-15 15:57:32.656  INFO 9484 --- [           main] o.s.i.endpoint.EventDrivenConsumer       : started inbound.springCloudBus.default
2017-09-15 15:57:32.656  INFO 9484 --- [           main] o.s.c.support.DefaultLifecycleProcessor  : Starting beans in phase 2147483647
2017-09-15 15:57:32.669  INFO 9484 --- [           main] e.d.CloudEruakaBusSenderApplicationTests : Started CloudEruakaBusSenderApplicationTests in 3.884 seconds (JVM ru
Sender hello Fri Sep 15 15:57:32 CST 2017
2017-09-15 15:57:32.715  INFO 9484 --- [       Thread-2] o.s.w.c.s.GenericWebApplicationContext   : Closing org.springframework.web.context.support.GenericWebApplication
```

注意当测试显示正确时,并不能说明数据被接收到,只能说明被发送出去了,也可以在接收端写一个测试类