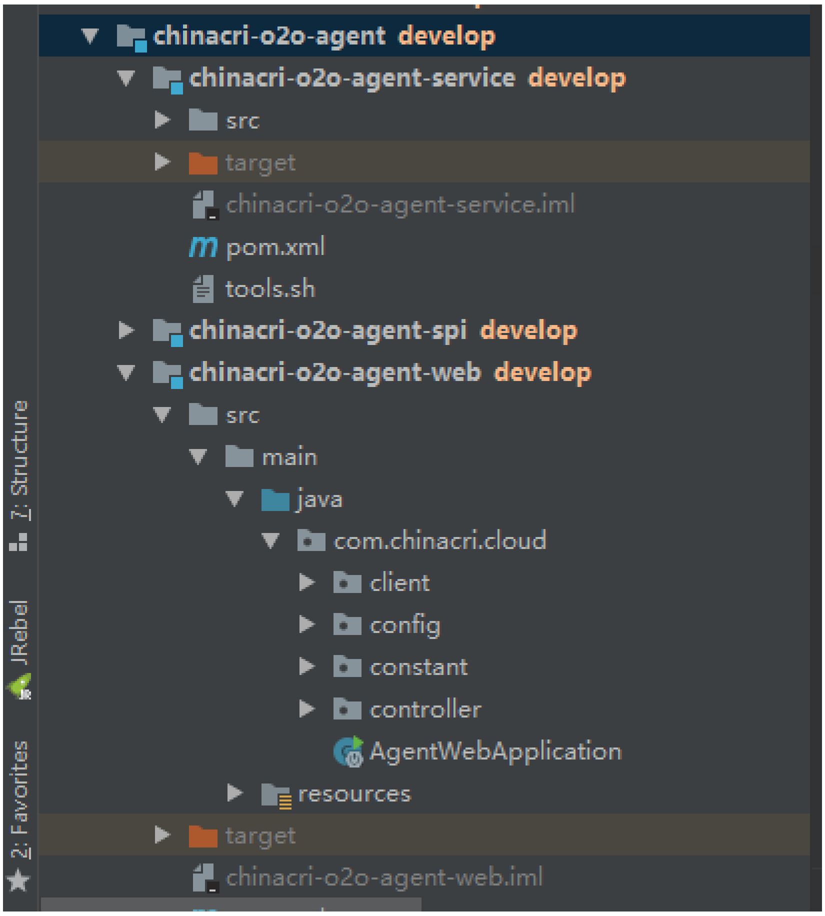
Spring Cloud+Dubbo对Feign进行RPC改造

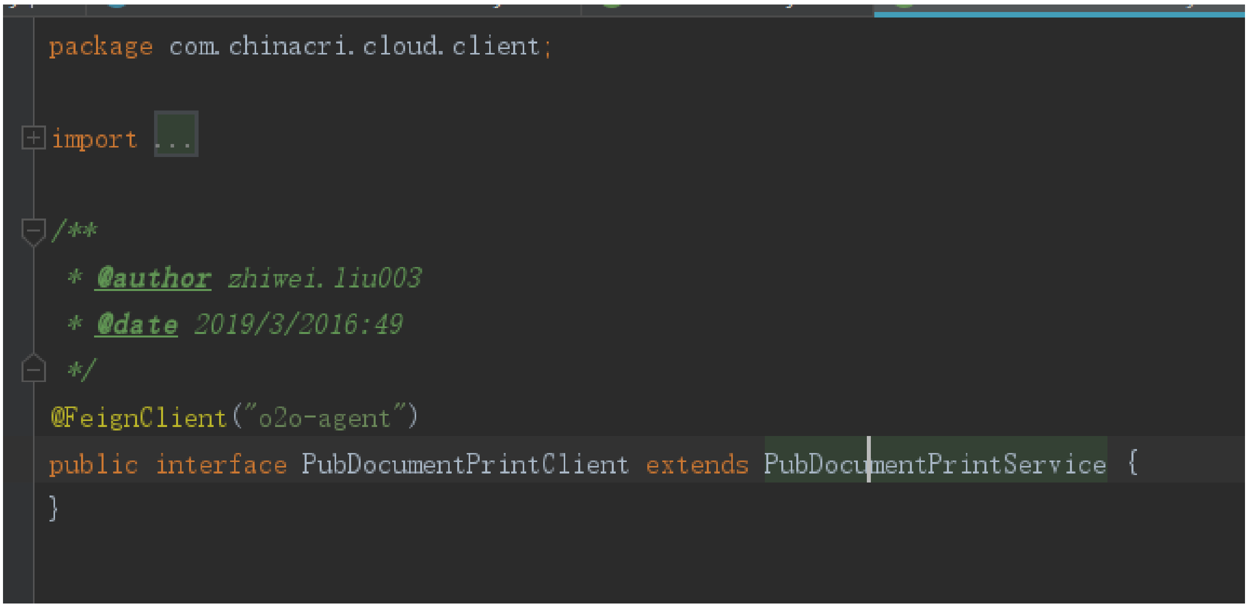
Spring Cloud Feign是基于Http Restful的调用，在高并发下的性能不够理想（基于Ribbon以及带有熔断机制，可以防止雪崩），成为性能瓶颈，对Feign进行Dubbo的RPC改造。

Spring Cloud的项目结构如下 web 调用 service 为例

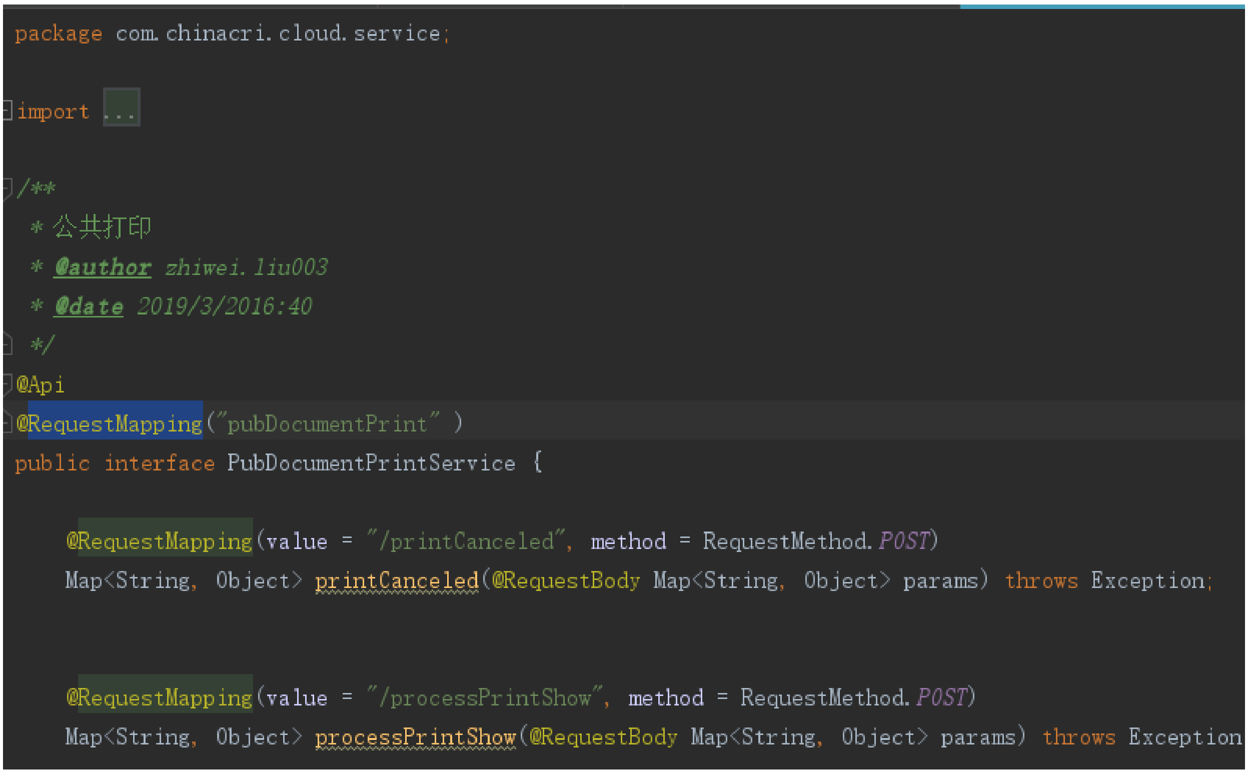


```
▼ chinacri-o2o-agent  develop
    ▼ chinacri-o2o-agent-service  develop
        ▶ src
        ▶ target
          chinacri-o2o-agent-service.iml
          pom.xml
          tools.sh
    ▶ chinacri-o2o-agent-spi  develop
    ▼ chinacri-o2o-agent-web  develop
        ▼ src
            ▼ main
                ▼ java
                    ▼ com.chinacri.cloud
                        ▶ client
                        ▶ config
                        ▶ constant
                        ▶ controller
                          AgentWebApplication
                ▶ resources
        ▶ target
          chinacri-o2o-agent-web.iml
```

公共打印接口

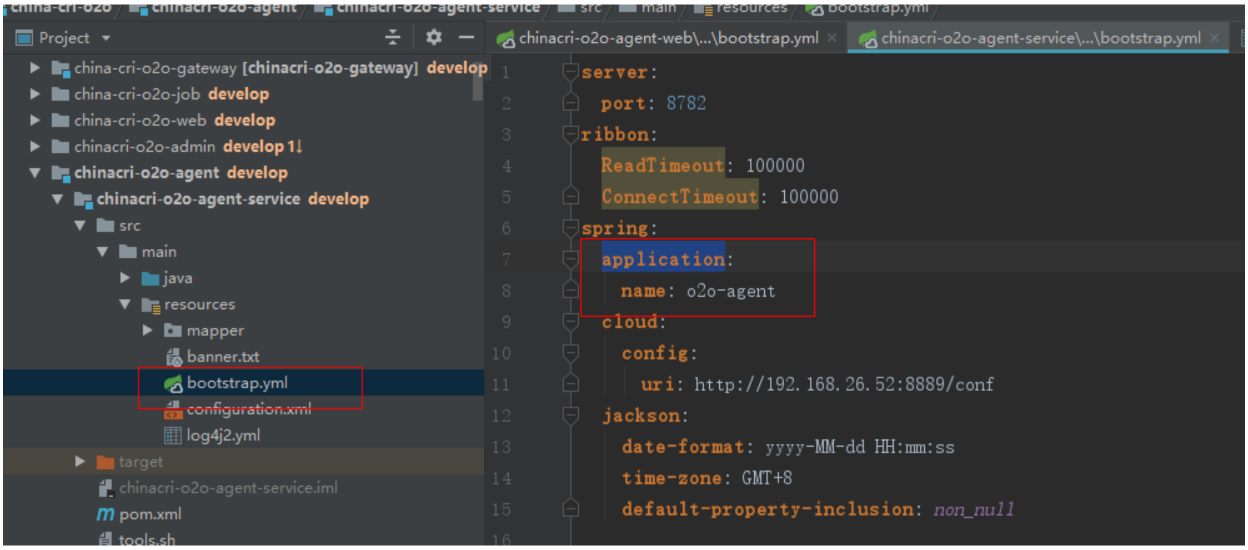


```java
package com.chinacri.cloud.client;

import ...

/**
 * @author zhiwei.liu003
 * @date 2019/3/2016:49
 */
@FeignClient("o2o-agent")
public interface PubDocumentPrintClient extends PubDocumentPrintService {
}
```



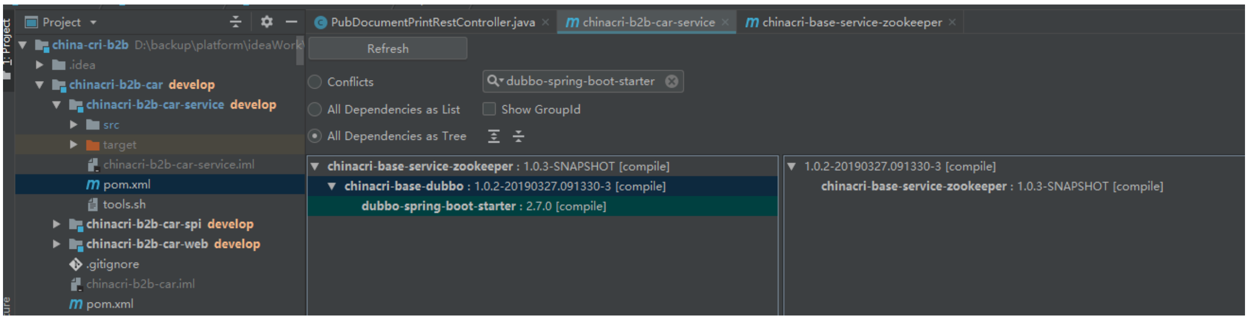
```java
package com.chinacri.cloud.service;

import ...

/**
 * 公共打印
 * @author zhiwei.liu003
 * @date 2019/3/2016:40
 */
@Api
@RequestMapping("pubDocumentPrint" )
public interface PubDocumentPrintService {

    @RequestMapping(value = "/printCanceled", method = RequestMethod.POST)
    Map<String, Object> printCanceled(@RequestBody Map<String, Object> params) throws Exception;


    @RequestMapping(value = "/processPrintShow", method = RequestMethod.POST)
    Map<String, Object> processPrintShow(@RequestBody Map<String, Object> params) throws Exception
```

@FeignClient("o2o-agent") 指定为Service application name


china-cri-o2o
chinacri-o2o-agent
chinacri-o2o-agent-service
src
main
resources
bootstrap.yml
Project
chinacri-o2o-agent-web\...\bootstrap.yml
chinacri-o2o-agent-service\...\bootstrap.yml
china-cri-o2o-gateway [chinacri-o2o-gateway] develop
china-cri-o2o-job develop
china-cri-o2o-web develop
chinacri-o2o-admin develop 1↓
chinacri-o2o-agent develop
chinacri-o2o-agent-service develop
src
main
java
resources
mapper
banner.txt
bootstrap.yml
configuration.xml
log4j2.yml
target
chinacri-o2o-agent-service.iml
pom.xml
tools.sh

```yaml
server:
  port: 8782
ribbon:
  ReadTimeout: 100000
  ConnectTimeout: 100000
spring:
  application:
    name: o2o-agent
  cloud:
    config:
      uri: http://192.168.26.52:8889/conf
  jackson:
    date-format: yyyy-MM-dd HH:mm:ss
    time-zone: GMT+8
    default-property-inclusion: non_null
```

Service 作为Dubbo提供者 pom 添加Dubbo引用

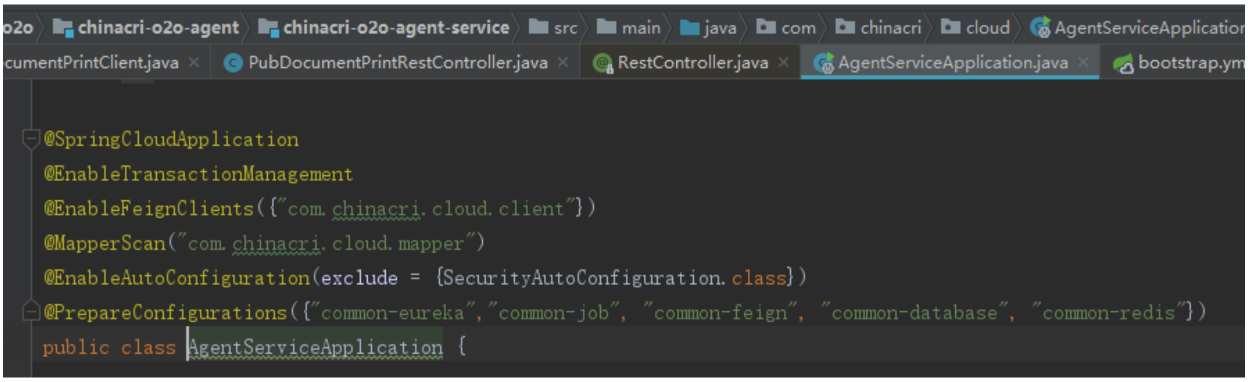

Project
china-cri-b2b D:\backup\platform\ideaWork
.idea
chinacri-b2b-car develop
chinacri-b2b-car-service develop
src
target
chinacri-b2b-car-service.iml
pom.xml
tools.sh
chinacri-b2b-car-spi develop
chinacri-b2b-car-web develop
.gitignore
chinacri-b2b-car.iml
pom.xml
PubDocumentPrintRestController.java
chinacri-b2b-car-service
chinacri-base-service-zookeeper
Refresh
○ Conflicts
○ All Dependencies as List
● All Dependencies as Tree
□ Show GroupId

dubbo-spring-boot-starter

```
▼ chinacri-base-service-zookeeper : 1.0.3-SNAPSHOT [compile]
  ▼ chinacri-base-dubbo : 1.0.2-20190327.091330-3 [compile]
      dubbo-spring-boot-starter : 2.7.0 [compile]
```

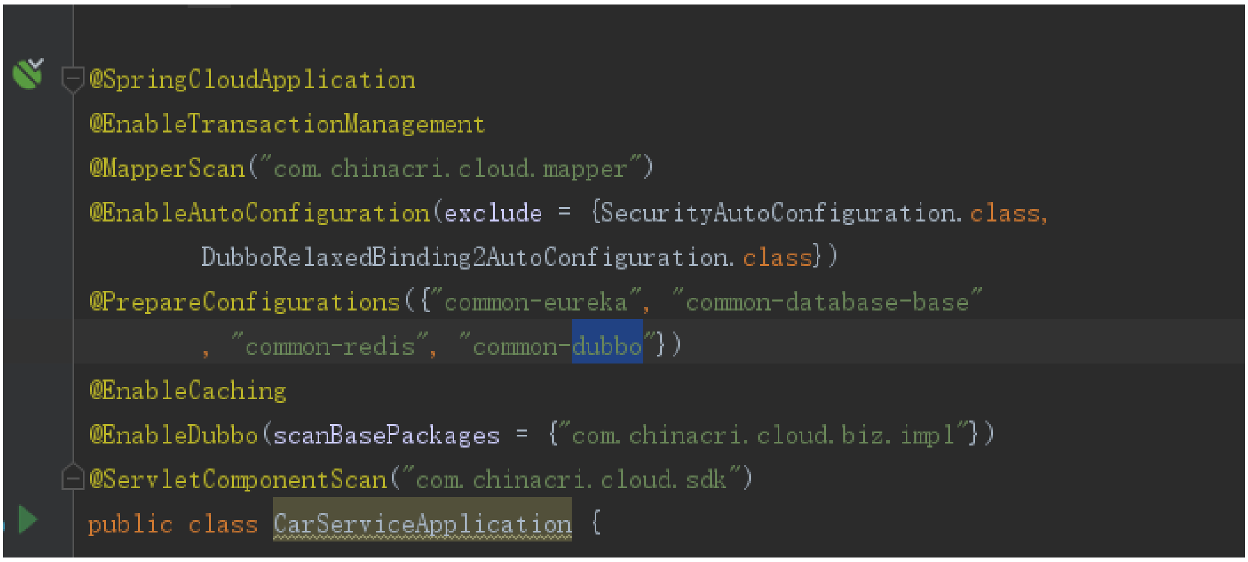
```
▼ 1.0.2-20190327.091330-3 [compile]
    chinacri-base-service-zookeeper : 1.0.3-SNAPSHOT [compile]
```

修改启动配置



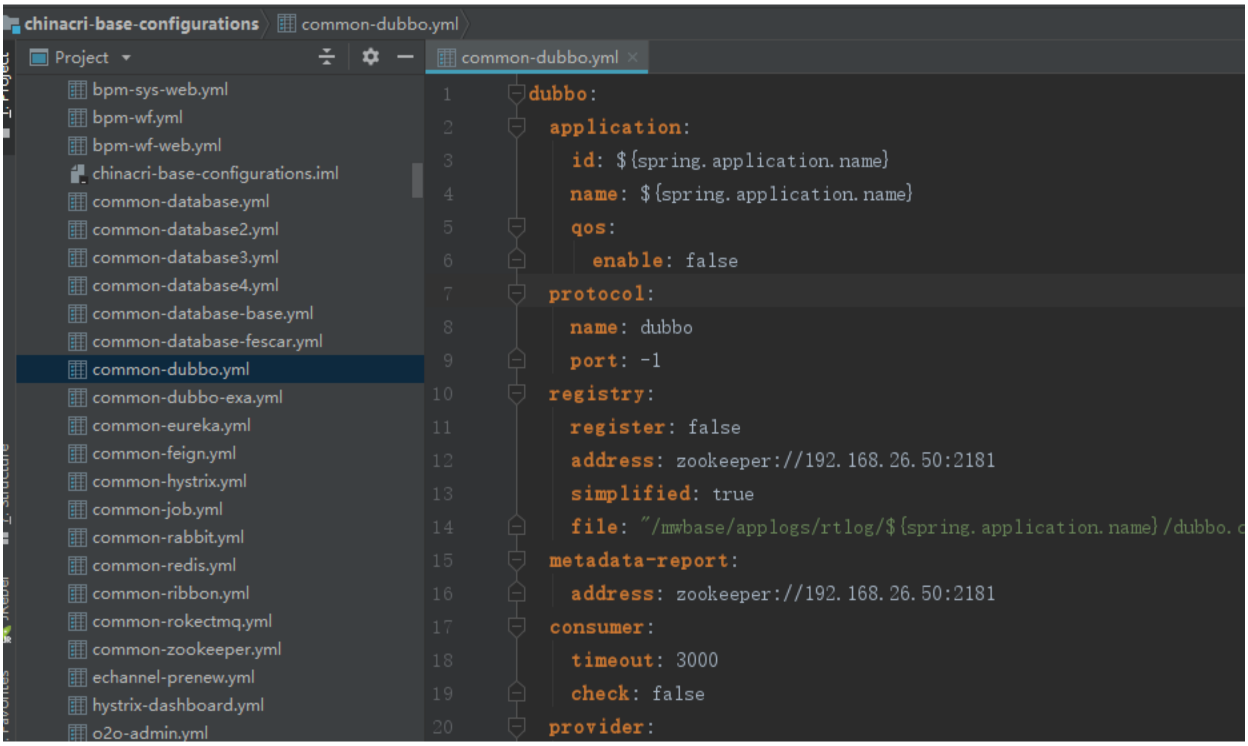
```java
@SpringCloudApplication
@EnableTransactionManagement
@EnableFeignClients({"com.chinacri.cloud.client"})
@MapperScan("com.chinacri.cloud.mapper")
@EnableAutoConfiguration(exclude = {SecurityAutoConfiguration.class})
@PrepareConfigurations({"common-eureka","common-job", "common-feign", "common-database", "common-redis"})
public class AgentServiceApplication {
```

添加common-dubbo ， 根据名称会去取配置服务器对应配置文件



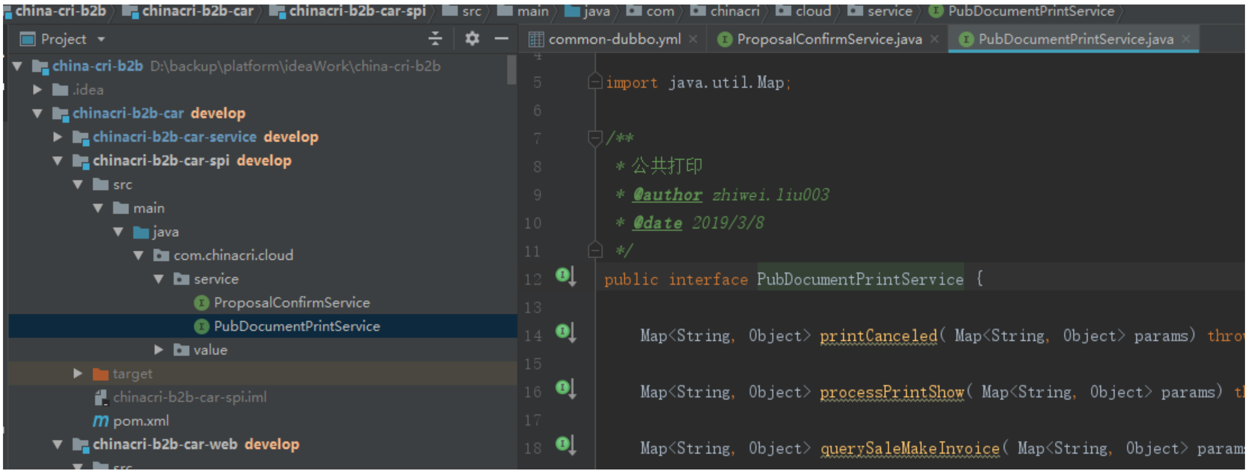
```java
@SpringCloudApplication
@EnableTransactionManagement
@MapperScan("com.chinacri.cloud.mapper")
@EnableAutoConfiguration(exclude = {SecurityAutoConfiguration.class,
        DubboRelaxedBinding2AutoConfiguration.class})
@PrepareConfigurations({"common-eureka", "common-database-base"
        , "common-redis", "common-dubbo"})
@EnableCaching
@EnableDubbo(scanBasePackages = {"com.chinacri.cloud.biz.impl"})
@ServletComponentScan("com.chinacri.cloud.sdk")
public class CarServiceApplication {
```

dubbo配置



```yaml
dubbo:
  application:
    id: ${spring.application.name}
    name: ${spring.application.name}
    qos:
      enable: false
  protocol:
    name: dubbo
    port: -1
  registry:
    register: false
    address: zookeeper://192.168.26.50:2181
    simplified: true
    file: "/mwbase/applogs/rtlog/${spring.application.name}/dubbo.c
  metadata-report:
    address: zookeeper://192.168.26.50:2181
  consumer:
    timeout: 3000
    check: false
  provider:
```

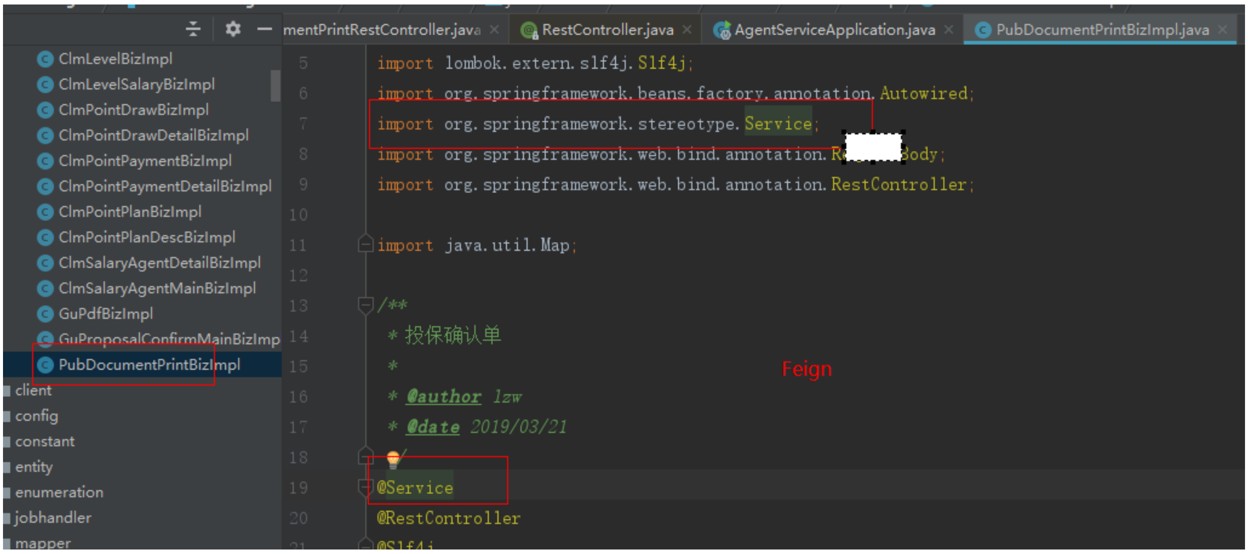
Service 接口放置在 spi中



```java
import java.util.Map;

/**
 * 公共打印
 * @author zhiwei.liu003
 * @date 2019/3/8
 */
public interface PubDocumentPrintService {

    Map<String, Object> printCanceled( Map<String, Object> params) thro

    Map<String, Object> processPrintShow( Map<String, Object> params) t

    Map<String, Object> querySaleMakeInvoice( Map<String, Object> param
```

接口实现类的标签修改 其中这个@Service已经不再是spring的标签，而需要使用，

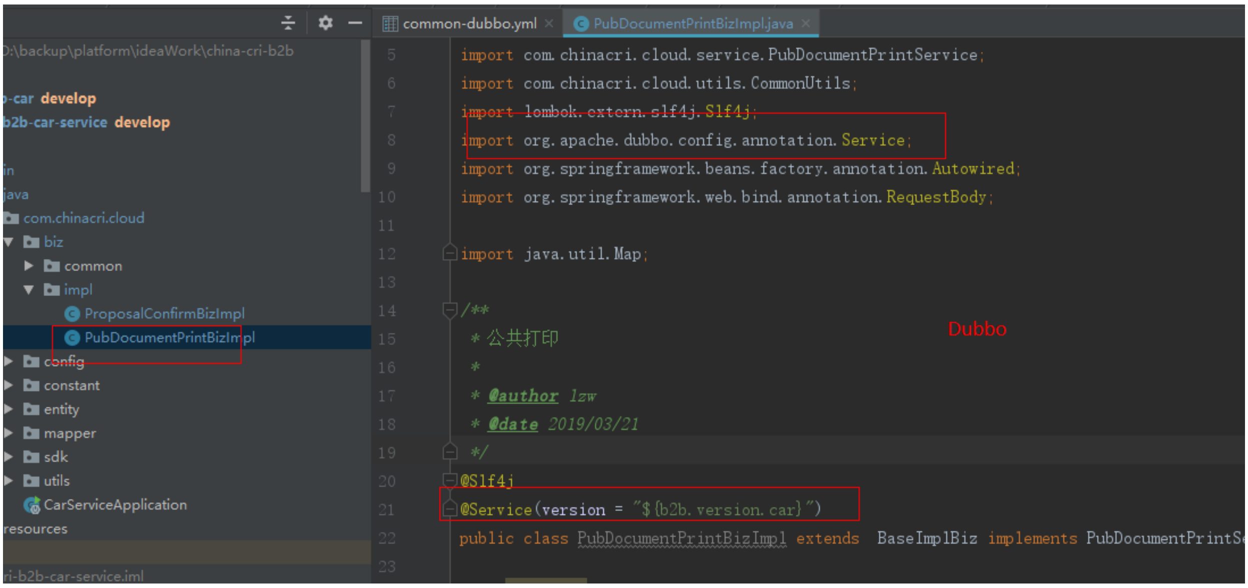
```
import com.alibaba.dubbo.config.annotation.Service;
```


mentPrintRestController.java
RestController.java
AgentServiceApplication.java
PubDocumentPrintBizImpl.java

ClmLevelBizImpl
ClmLevelSalaryBizImpl
ClmPointDrawBizImpl
ClmPointDrawDetailBizImpl
ClmPointPaymentBizImpl
ClmPointPaymentDetailBizImpl
ClmPointPlanBizImpl
ClmPointPlanDescBizImpl
ClmSalaryAgentDetailBizImpl
ClmSalaryAgentMainBizImpl
GuPdfBizImpl
GuProposalConfirmMainBizImp
PubDocumentPrintBizImpl
client
config
constant
entity
enumeration
jobhandler
mapper

```
import lombok.extern.slf4j.Slf4j;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import org.springframework.web.bind.annotation.R[redacted]Body;
import org.springframework.web.bind.annotation.RestController;

import java.util.Map;

/**
 * 投保确认单
 *
 * @author lzw
 * @date 2019/03/21
 */
@Service
@RestController
@Slf4j
```

Feign

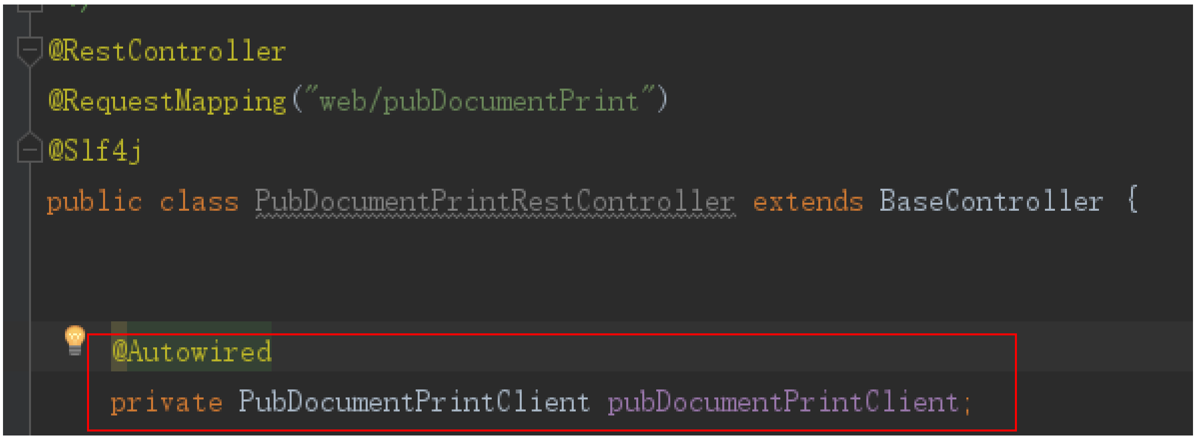


D:\backup\platform\ideaWork\china-cri-b2b
-car develop
b2b-car-service develop
in
java
com.chinacri.cloud
biz
common
impl
ProposalConfirmBizImpl
PubDocumentPrintBizImpl
config
constant
entity
mapper
sdk
utils
CarServiceApplication
resources
ri-b2b-car-service.iml

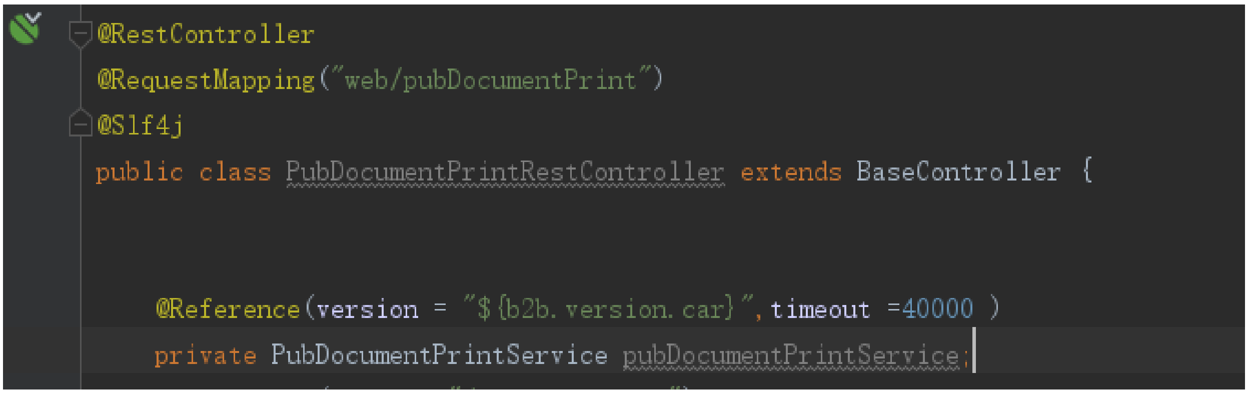
```
import com.chinacri.cloud.service.PubDocumentPrintService;
import com.chinacri.cloud.utils.CommonUtils;
import lombok.extern.slf4j.Slf4j;
import org.apache.dubbo.config.annotation.Service;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.RequestBody;

import java.util.Map;

/**
 * 公共打印
 *
 * @author lzw
 * @date 2019/03/21
 */
@Slf4j
@Service(version = "${b2b.version.car}")
public class PubDocumentPrintBizImpl extends  BaseImplBiz implements PubDocumentPrintS
```

Dubbo

在使用的Controller中注释掉之前的feign注入，使用Dubbo的接口



```
@RestController
@RequestMapping("web/pubDocumentPrint")
@Slf4j
public class PubDocumentPrintRestController extends BaseController {


    @Autowired
    private PubDocumentPrintClient pubDocumentPrintClient;
```



```
@RestController
@RequestMapping("web/pubDocumentPrint")
@Slf4j
public class PubDocumentPrintRestController extends BaseController {


    @Reference(version = "${b2b.version.car}",timeout =40000 )
    private PubDocumentPrintService pubDocumentPrintService;
```