## GODDESS项目详解

1. **结构及目录**

**common-- 通用包,该包下包含所有模块的通用基础包,可见目录:**

common-api :基础api模块,common-consumer,common-provider引用此模块

common-consumer:基础消费者模块,其他模块consumer均引用此模块

common-provider:提供者模块,其他模块provider均引用此模块

common-jpa :持久化模块,common-provider引用此模块来进行持久化

common-redis:缓存模块

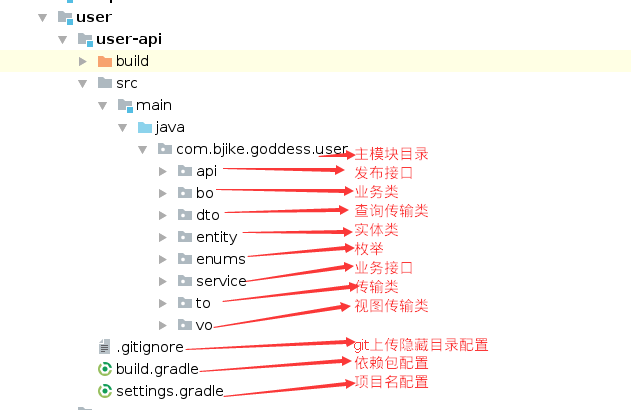
common-utils:工具包模块

**主目录配置**

****

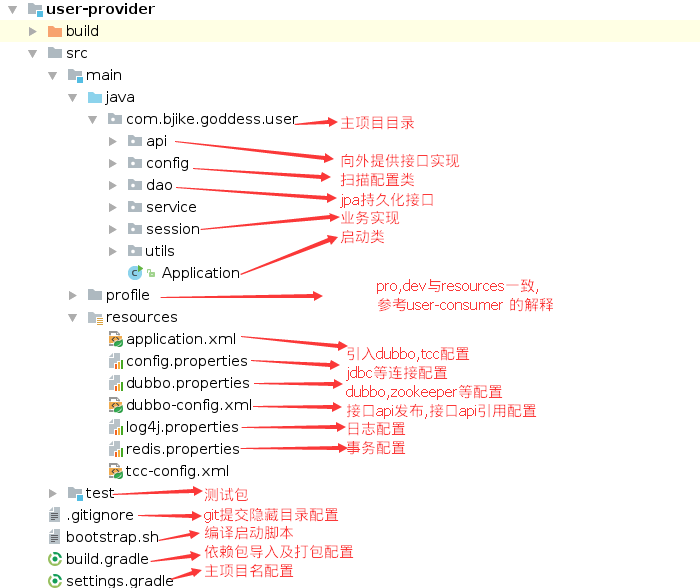
**modules -- 所有功能模块 [以user模块为例],可见目录**

**user-api(主接口)**

****

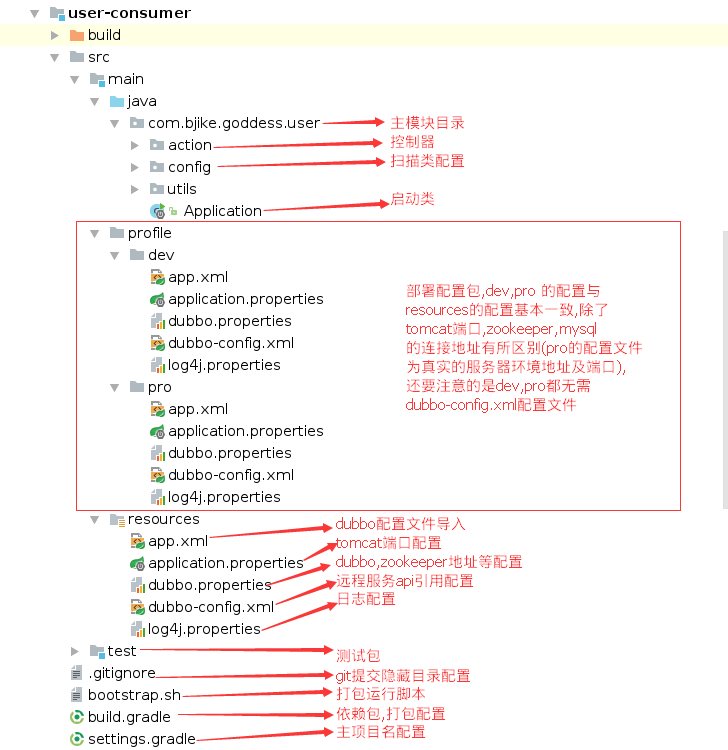
及依赖包

**user-provider(提供者)**

****

tcc-config.xm redis连接池,及事务配置,dev,pro的redis连接与resources不同,注意修改

**user-consumer(消费者)**

****

**Resource配置说明:**

1. **provider(以user-provider为例)**

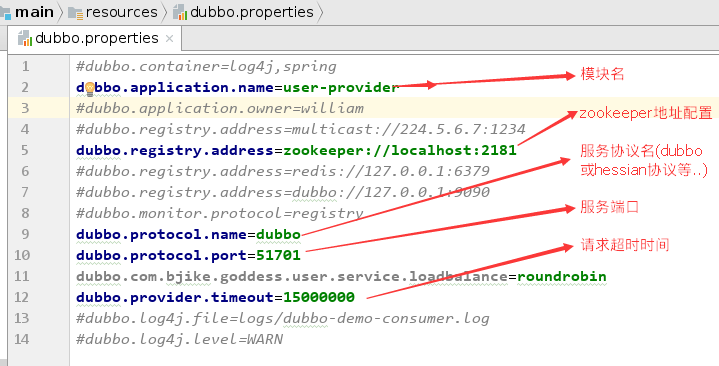
**注: provider里的profile/pro,profile/dev 与resource的配置文件基本一致,唯一不一样的是config.properties的mysql服务地址或端口,dubbo.properties的zookeeper地址 tcc.config.xml的redis地址 ,以及pro,dev都无需dubbo-config.xml文件**

application.xml 仅仅引入配置,不做详解

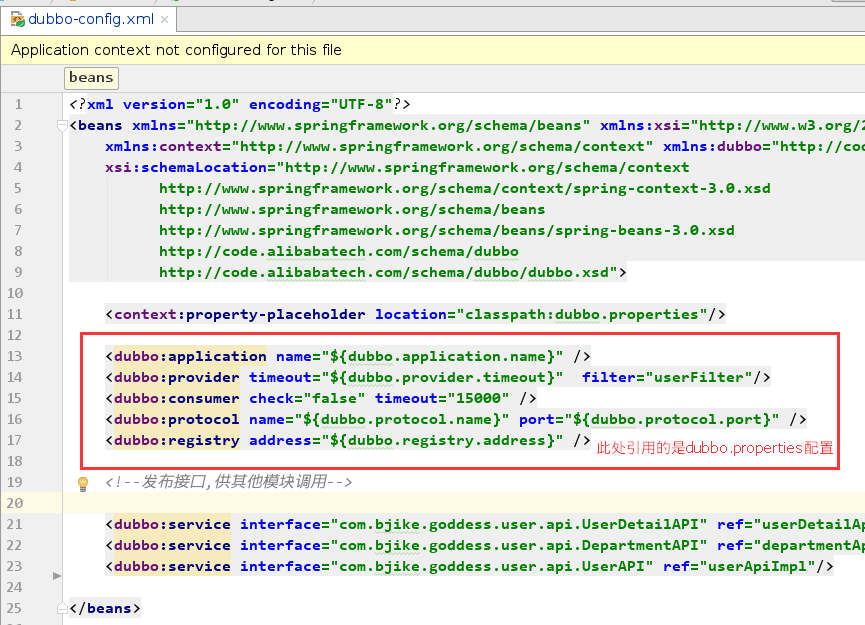
config.properties 数据库连接配置

tcc-config.xml 统一事务及redis连接池配置,无需了解

log4j.properties 统一日志配置,无需了解

dubbo.properties:

dubbo-config.xml:



1. **consumer(以user-consumer为例)**

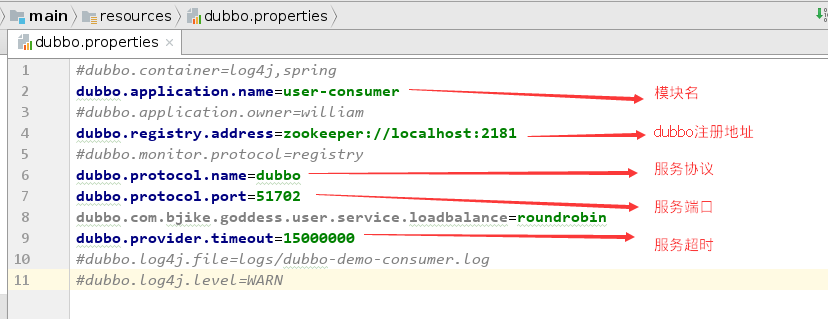
**注: consumer里的profile/pro,profile/dev 与resource的配置文件基本一致,唯一不一样的是application.properties的**server.port(tomcat端口)**,dubbo.properties的zookeeper地址 以及pro,dev都无需dubbo-config.xml文件**

app.xml仅仅引入配置,不做详解

application.properties tomcat 端口设置

log4j.properties 日志配置

dubbo.properties



dubbo-config.xml

