

测试环境与UAT环境发布

- 1 概述
- 2 测试发布流程
 - 2.1 修改POM中的版本
 - 2.2 修改application-test.yml文件
 - 2.3 git中pull request到release分支
 - 2.4 git打tag
 - 2.4.1 批量merge以及tag
 - 2.5 DBA同步数据库
 - 2.6 jenkins中depoly代码到私服
 - 2.7 openshift中build
- 3 测试系统的发布计划
- 4 UAT环境发布
 - 4.1 增加application-uatca.yml文件
 - 4.2 代码分支
 - 4.3 数据库脚本
 - 4.4 部署应用
 - 4.5 UAT访问地址
- 5 各组配置管理员

概述

问题	解决方案
版本，测试版本，代码版本，git版本如何管理	代码版本在maven的pom文件中修改（parent工程，api工程，server工程都需要改） 将develop分支merge到release分支 git版本在git代码库的release分支上打tag
测试环境提bug，在哪儿修改？	测试环境发现bug，先在develop改
每周发布几次？提测就发，还是固定时间发	一周三次，135发布 版本号规则，v0.0.20200303 maven中版本号每季度升级，0.1,0.2,0.3,0.4...1.0,与sprint中版本保持一致即可。 测试系统需要紧急发布，例如由于重大bug，导致测试无法进行，可以紧急在release分支修改，手工发布，记得将这个bug同步到develop分支
数据库如何同步？包括uat版本发布	一季度，dba帮着直接从开发刷测试，测试库保留1个月的备份，每周一次； 从二季度开始，开发人员需要准备变更的脚本，，dba将脚本发布到测试环境，脚本规范由dba定。
每个组安排发布人员	各组固定安排好发布人员，责任到人
uat环境代码版本	uat发布master版本

测试发布流程

修改POM中的版本

以下以主文件工程为例：

模块的父工程pom中的version字段，api工程中的pom中version字段，server工程中的version字段

```
<!-- spring boot -->
<parent>
    <groupId>com.acca</groupId>
    <artifactId>araf-parent</artifactId>
    <version>1.0.0</version>
</parent>

<groupId>com.acca.opra</groupId>
<artifactId>sys</artifactId>
<version>0.1</version>
<packaging>pom</packaging>
<description></description>

<properties>
    <araf.version>1.0.0</araf.version>
</properties>
```

代码版本与sprint版本保持一致，每个季度修改一次版本即可。

api与server工程修改pom中的版本信息，将parent的版本改为release version

```
<parent>
    <groupId>com.acca.opra</groupId>
    <artifactId>mas</artifactId>
    <version>0.1</version>
</parent>
```

系统中引用其他微服务的版本，请根据发布的版本自行确定

修改application-test.yml文件

目前测试环境启动，采用spring active profile的方式，指定test。所以需要修改测试环境的配置文件。

```
spring:
  cloud:
    consul:
      host: consul.ca-opra-test.svc.cluster.local
  redis:
    host: redis.ca-opra-test.svc.cluster.local
  data:
    elasticsearch:
      cluster-name: opra
      cluster-nodes: 10.1.19.223:9300
  elasticsearch:
    rest:
      uris: ["http://10.1.19.223:9200"]

spring.kafka.bootstrap-servers: my-cluster-kafka-bootstrap.ca-opra-test.svc:9092
```

修改redis, kafka, 数据库, es的测试环境地址

以后会采用配置服务器的方式

git中pull request到release分支

在bitbucket中, 提交pull request, 从develop到release分支。并merge。如果出现问题, 手工解决代码冲突。

git打tag

本地切换到release分支, 在git的release分支中打tag, 记录版本号。tag命令为

```
git tag v0.0.20200306
git push origin --tags
```

注意添加一个v, 用来表示版本

批量merge以及tag

涉及到多个工程的发布, 可以使用下面的脚本批量操作。

```
@echo off

set CODE_HOME=D:\Enterprise\GIT\RELEASE

set formatDate=%date:~0,4%%date:~5,2%%date:~8,2%
echo %formatDate%

setlocal enabledelayedexpansion
for %%i in (opra-sys,opra-sys-jobschedule,opra-sys-filemanagement,opra-
mas-global,opra-mas,opra-ui) do (
set subsystem=%CODE_HOME%\%%i
echo !subsystem!
cd !subsystem!
git checkout release
git pull
git tag -d v0.0.%formatDate%
git push origin :refs/tags/v0.0.%formatDate%
git merge origin/develop
git commit -m "merge release"
git tag v0.0.%formatDate%
git push origin
)

cd %CODE_HOME%
echo "tag finish!"
```

DBA同步数据库

初期，为了不增加大家的负担，由dba帮忙直接从开发库刷新到测试库，如果需要数据的同步，请提前通知dba

计划从二季度开始，开发库的变更需要由开发人员提供脚本，dba审核，再同步到测试库。脚本的规范由dba指制定。

jenkins中depoly代码到私服

jenkins地址10.1.21.24:8081/jenkins

自己建立相应的任务，发布代码。参见: <http://10.1.17.141/jenkins/view/opra-deploy/>

这里有个问题，目前发布jar到私服，与发布openshift是两个jenkins，因为openshift内的jenkins没法配置私服的用户名密码。外部的jenkins目前连不上openshift，插件装不上。

下面我们继续研究这个问题。

openshift中build

参见DevOps章节，可以将测试环境的build都做成一个jenkins任务。点击发布。参见：<https://jenkins-ca-opra-dev.apps.ocp.acca/view/opra-test/job/opra-sys-test-DEPLOY/>

在openshift平台中ca-opra-test空间中建立build config，注意代码分支使用release分支。参见<https://console-openshift-console.apps.ocp.acca/k8s/ns/ca-opra-test/buildconfigs>

涉及到jenkins都可以做成定时

测试系统的发布计划

每周三次，周一，周三，周五，下班时段发布。

版本号采用

{大版本号}. {日期}

的方式

例如：一季度发的版本：0.0.20200306

二季度发的版本：0.1.20200410

UAT环境发布

三大航的uat环境是独立部署的。

uat环境的bug需要在独立的bugfix分支修改

uat对应代码master分支

下面的配置已ca的uat环境为例

增加application-uatca.yml文件

```

spring:
  application:
    name: sys
  cloud:
    consul:
      host: consul.ca-opra-uat.svc.cluster.local
      port: 8500
      discovery:
        prefer-ip-address: true
        instance-id: ${spring.application.name}-${spring.cloud.client.ip-
address}:${server.port}
        serviceName: ${spring.application.name}
        healthCheckPath: /actuator/health
        healthCheckInterval: 15s
        healthCheckCriticalTimeout: 60s #
    redis:
      host: redis.ca-opra-uat.svc.cluster.local
      port: 6379
      database: 0
      pool:
        max-idle: 8
        min-idle: 0
        max-wait: 1
        max-active: 8
    data:
      elasticsearch:
        cluster-name: opra-es-uat
        cluster-nodes: 10.1.17.248:9300
    elasticsearch:
      rest:
        uris: ["http://10.1.17.248:9200"]

datasource:
  url: jdbc:postgresql://10.1.17.237:5432/oprasys
  username: oprasys
  password: accal234

# loggingdebug
logging:
  level:
    org.mybatis: INFO

spring.kafka.bootstrap-servers: my-cluster-kafka-bootstrap.ca-opra-uat.svc:
9092

```

各自配置管理员注意，修改自己的数据库连接，以及相关连接

es的地址:

MU-OPRA-U-ES-1	10.1.19.40
CZ-OPRA-U-ES-1	10.1.19.254
CA-OPRA-U-ES-1	10.1.17.248

代码分支

uat环境对应的分支为master分支，请各自代码打基线后，合并到master分支。

master分支中对应的版本tag为v0.1

数据库脚本

数据库初始化脚本以及数据文件对应的git仓库为：

`ssh://git@git.acca.com.cn:7999/opra-git/opra-db.git`

各组配置管理员将需要准备的脚本文件准备好，统一发给dba，由dba放到基线中。

部署应用

在openshift平台中的ca-opra-uat，mu-opra-uat，ca-opra-uat三个project中部署各自模块的应用

其中分支执行master

spring.profiles.active指定uat-ca

UAT访问地址

`https://10.1.17.14/opra-ui` ca

`https://10.1.19.205/opra-ui` cz

`https://10.1.19.222/opra-ui` mu

各组配置管理员

团队，模块	配置管理员	备份
负责销售模块和空白票模块开发（客运产品团队）	王宇	孔令昊
负责运输模块和配比模块开发（客运产品团队）	张赫	辛莉
负责IWB和OWB模块开发（客运产品团队）	张晓芳	杨霁轩
负责分摊引擎开发（计算引擎团队）	贾瑶丽	宋鹏飞
负责税费引擎及报税管理系统开发（计算引擎团队）	聂于智	刘海涛

负责系统技术架构、前端页面的开发，负责主文件开发、系统管理等的开发（技术平台团队）	张冬	赵兴
应收资金管理，账务子系统（业务平台团队）	李明易	杨凯
负责增值产品开发，例如CI和里程积分产品开发（增值产品团队）	张彩荣	孙同鑫
负责历史库存储以及数据服务产品开发（数据平台团队）	马新宇	