

# 数据库版本管理

## 一. 脚本规范定义

假设有版本号0.1.1, 0.1.2, 0.2前2个是紧急投产uat的脚本。



数据库脚本管理流程：

- (1) 需要记录issue号，或者jira号码与变更修改的代码关联
- (2) 需要记录版本号，决定变更投uat的版本

- 1. 需要发布到各个tag的变更：版本号为对应的tag号，比如当前0.1tag已完成，这个季度0.2 tag，此时变更版本号为0.2
- 2. 需要紧急发布到测试环境（否则影响后续测试）的数据库变更：版本号同1
- 3. 需要紧急发布到uat环境的变更：版本号为已完成tag的子版本，比如当前0.1tag已完成，版本号为0.1.x

脚本规范定义如下：

- 1. 提供每次发布的文件，文件命名：[模块名]-[schema-type]-[版本号]-yyyyMMdd-[3位序号]-[JIRA-ID].sql

|            |                 |   |        |      |
|------------|-----------------|---|--------|------|
| 最近访问的位置    | 2020/5/29 14:38 | OPRASYS-oprmas-0.1.1-20200512-001-OPRATPT-830.sql | SQL 文件 | 1 KB |
| 库          | 2020/5/29 14:38 | OPRASYS-oprmas-0.1.1-20200512-002-OPRATPT-830.sql | SQL 文件 | 1 KB |
| Subversion | 2020/5/29 14:38 | OPRASYS-oprmas-0.1.1-20200527-003-OPRATPT-590.sql | SQL 文件 | 1 KB |
| 视频         | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200416-001-OPRATPT-648.sql   | SQL 文件 | 1 KB |
| 图片         | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200421-002-OPRATPT-691.sql   | SQL 文件 | 1 KB |
| 文档         | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200427-003-OPRATPT-659.sql   | SQL 文件 | 2 KB |
| 音乐         | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200427-004-OPRATPT-669.sql   | SQL 文件 | 1 KB |
| 计算机        | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200430-001-OPRATPT-736.sql   | SQL 文件 | 2 KB |
| 本地磁盘 (C:)  | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200520-001-OPRATPT-774.sql   | SQL 文件 | 2 KB |
| 本地磁盘 (D:)  | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200521-001-OPRATPT-912.sql   | SQL 文件 | 4 KB |
| 网络         | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200525-001-OPRATPT-774.sql   | SQL 文件 | 1 KB |
| tsclient   | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200525-001-OPRATPT-821.sql   | SQL 文件 | 4 KB |
|            | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200525-001-OPRATPT-922.sql   | SQL 文件 | 1 KB |
|            | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200525-001-OPRATPT-942.sql   | SQL 文件 | 8 KB |
|            | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200526-001-OPRATPT-781.sql   | SQL 文件 | 1 KB |
|            | 2020/5/29 14:38 | OPRASYS-oprmas-0.2-20200526-002-OPRATPT-781.sql   | SQL 文件 | 1 KB |

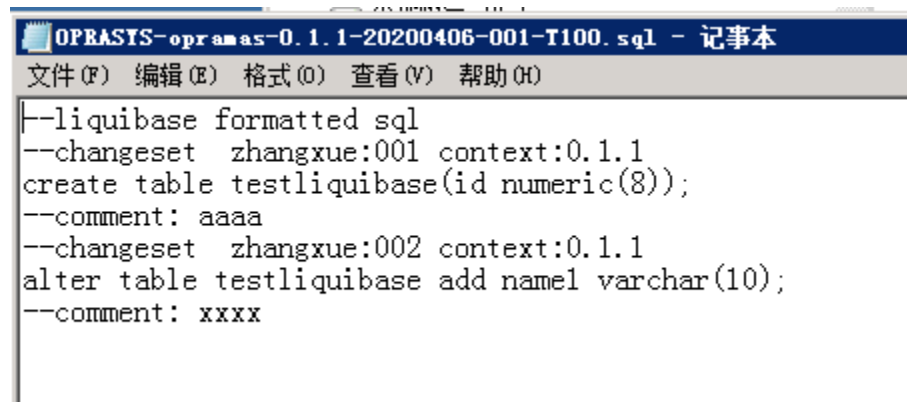
- 2. sql format要求&&脚本注意事项

### (1) 文件第一行

—liquibase formatted sql

(2) 每个SQL前一行标注 --changeset author:id, id使用[3位序号] (可以超过3位, 即超过999条SQL的变更可以接着1000...), context对应版本号

比如:



```
OPRASIS-opramas-0.1.1-20200406-001-T100.sql - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
--liquibase formatted sql
--changeset zhangxue:001 context:0.1.1
create table testliquibase(id numeric(8));
--comment: aaaa
--changeset zhangxue:002 context:0.1.1
alter table testliquibase add name1 varchar(10);
--comment: xxxx
```

以上的author和id会存放到数据管理表中, 一个sql文件中的id不能重复。

“--comment: ” 格式是冒号后面空一位空格, 然后输入注释内容, 注意这个只是前一个sql的注释, 跟comment每个表和列的SQL不同。

6. 保证sql文件不修改之前的文件, 每次发布之后新增sql文件。

7. sql 禁止包含schema名称

8. 禁止使用存储过程

9. 注意sql脚本文件编码格式必须为UTF-8

10. sql语句结尾处分号;必须使用英文符号

11. sql脚本运行成功后会在相应数据库中生成databasechangelog, databasechangeloglock 表请勿自己删除和修改

12. 新增和修改的变更请勿提前手动添加到数据库中

## 二. git管理脚本

develop分支仓库里建对应的schema-db文件夹, 子目录为版本号, 如v0.2, v0.1.1, 下图为目录结构

## Source

| <div> develop ▾ ...</div> <div>opra-mas /</div> |                                   |
|--|-----------------------------------|
| Source   | Description                       |
|  opra-mas-api                                   |                                   |
|  opra-mas-server                                |                                   |
|  opramas-db                                     |                                   |
|  .checkstyle                                    | 大洲信息                              |
|  .gitignore                                     | Merge remote-tracking branch 'ren |
|  pom.xml  | 增加管理大区的数据权限样例                     |

## Source

| <div> develop ▾ ...</div> <div>opra-mas / opramas-db /</div> |             |
|---|-------------|
| Source  | Description |
| ..  |             |
|  v0.1.1  |             |
|  v0.2  |             |

## Source

develop ... opra-mas / opramas-db / v0.2 /

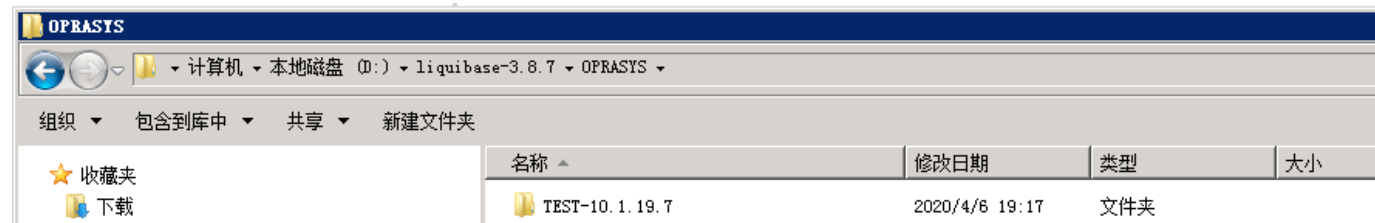
| Source                                  | Description |
|---|-------------|
| ..                                      |             |
| OPRASYS-opramas-0.2-20200416-001-OPR... | jiaoben     |
| OPRASYS-opramas-0.2-20200421-002-OPR... | 脚本提交        |
| OPRASYS-opramas-0.2-20200427-003-OPR... | 提交脚本        |
| OPRASYS-opramas-0.2-20200427-004-OPR... | tijiao      |

开发测试环境发布v0.2版本数据库脚本，uat发布v0.1.1的数据库脚本。

三. liquibase在jekin上配置和使用

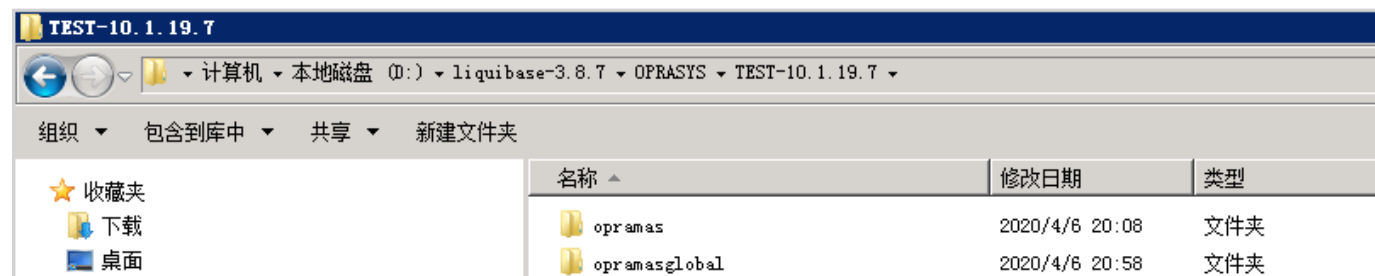
1. 更新sql脚本的配置

1.1 建数据库服务器目录，比如TEST-10.1.19.7



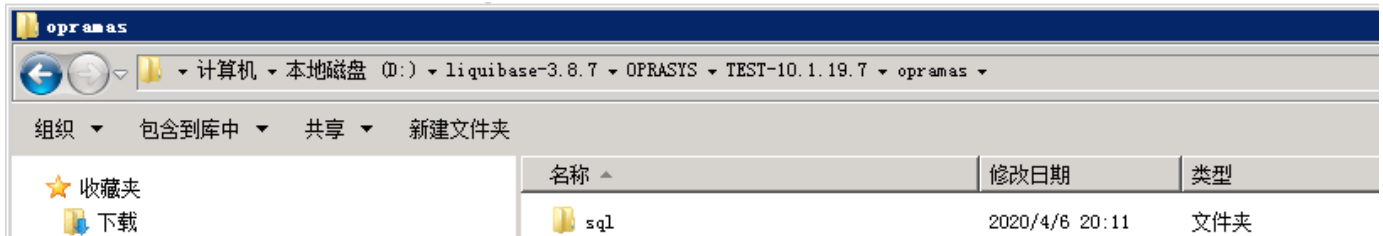
1.2 在数据库服务器目录下建schema对应目录, 比如 在

TEST-10.1.19.7目录下分别建opramas和opramasglobal两个子目录。

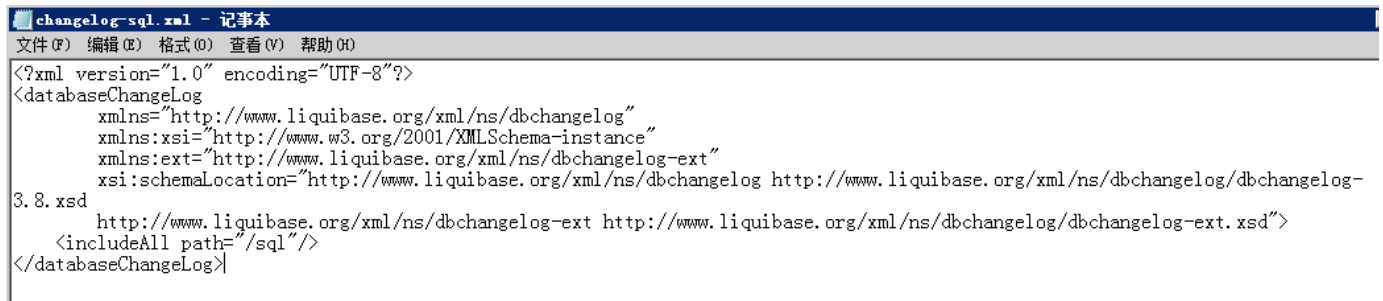


1.3 在schema子目录下建sql目录

此时的目录结构如下



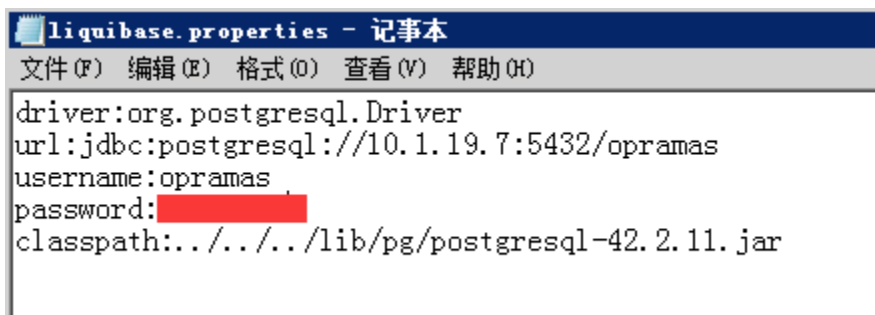
1.4建changelog-sql.xml文件，并输入内容如下：



```
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ext="http://www.liquibase.org/xml/ns/dbchangelog-ext"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-3.8.xsd
  http://www.liquibase.org/xml/ns/dbchangelog-ext http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-ext.xsd">
  <includeAll path="/sql"/>
</databaseChangeLog>
```

1.5建liquibase.properties文件，内容如下：

PG数据库连接配置



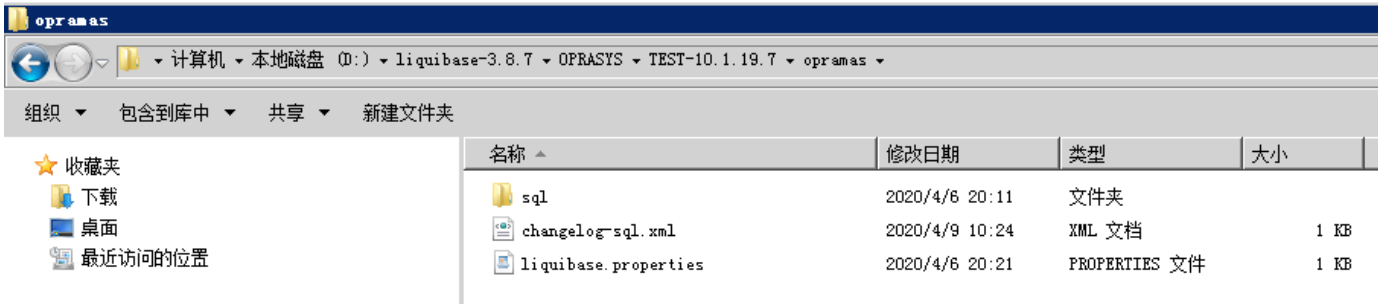
Oracle数据库连接配置：

```
liquibase.properties - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

driver:oracle.jdbc.OracleDriver
url:jdbc:oracle:thin:@//10.1.19.232:1521/opraitl
username:iwb
password:
classpath:../../lib/oracle/ojdbc8-19.3.0.0.0.jar
```

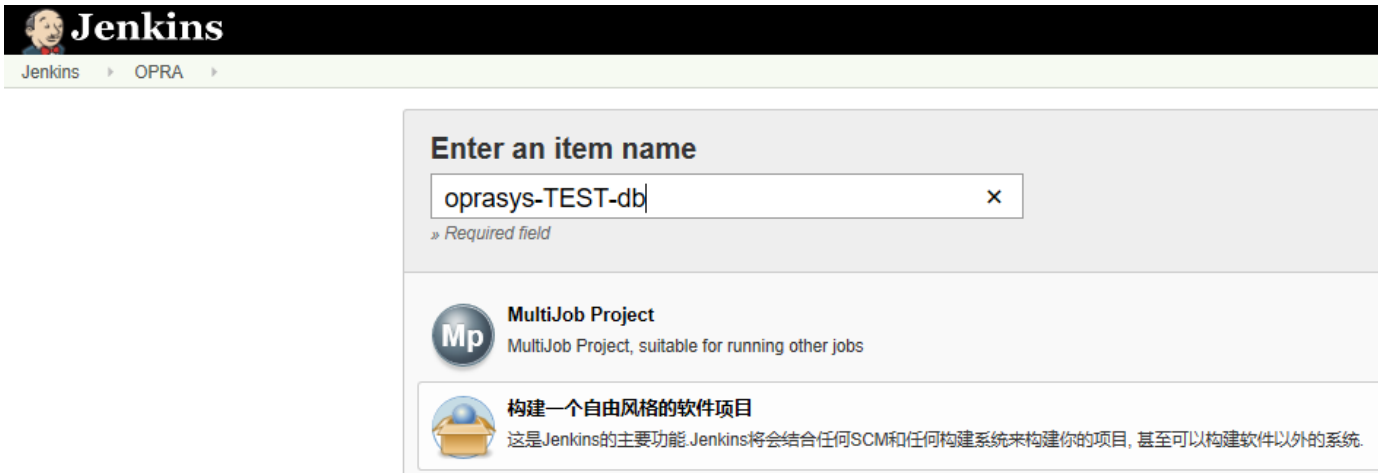
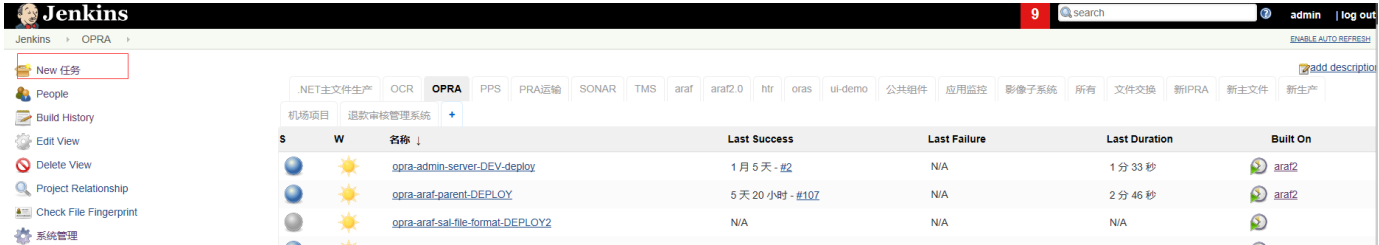
只需要修改url和username, password即可, 其他的配置无需修改。其他配置对应数据库版本的jdbc, 就是java连接数据库的驱动, oracle和pg对应版本的驱动我都从官网下载了, 所以只要是按照我设计的目录结构配置, 驱动信息无需修改。

此时目录结构如下:



1.6配置jenkins命令

1.6.1新建任务



1.6.2 配置git远程仓库和分支, 并配置下载目录 (当前目录的子目录)

使用Multiple SCMs里的Add GIT能够增加多个git的下载，需要配置对应的schema-name的子目录：

**Source Code Management**

☐ 无  
☐ CVS  
☐ CVS Projectset  
☐ Git  
☒ Multiple SCMs

**Git**

**Repositories**

Repository URL:  ?

Credentials:  Add

Advanced...  
Add Repository

**Branches to build**

Branch Specifier (blank for 'any'):  ?

Add Branch

**Repository browser**

?

**Additional Behaviours**

☒ Check out to a sub-directory ? 下载当前目录的schema子目录

Local subdirectory for repo:  ?

省略其他三个schema的git配置图。具体配置参考<http://10.1.17.143:8080/jenkins/job/oprasys-TEST-db/>

1.6.3 配置从git下载目录得到的数据库脚本拷贝到1.3的sql目录下

起始地址是上面“下载schema子目录”\schema-db\v版本号\\*

目的地址是对应服务器的1.3的sql目录

## Build

### 执行 Windows 批处理命令

Command

```
rd /s/q D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\opramas\sql
mkdir D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\opramas\sql
rd /s/q D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\opramasglobal\sql
mkdir D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\opramasglobal\sql
rd /s/q D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\oprajobschedule\sql
mkdir D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\oprajobschedule\sql
rd /s/q D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\oprasys\sql
mkdir D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\oprasys\sql
echo d | xcopy .\opramas\opramas-db\v0.1.1\* D:\liquibase-3.8.7\OPRASYS\TEST-
10.1.19.7\opramas\sql /s/e/y
echo d | xcopy .\opramas\opramas-db\v0.2\* D:\liquibase-3.8.7\OPRASYS\TEST-
10.1.19.7\opramas\sql /s/e/y
echo d | xcopy .\opramasglobal\opramasglobal-db\v0.1.1\* D:\liquibase-3.8.7\OPRASYS\TEST-
10.1.19.7\opramasglobal\sql /s/e/y
echo d | xcopy .\opramasglobal\opramasglobal-db\v0.2\* D:\liquibase-3.8.7\OPRASYS\TEST-
10.1.19.7\opramasglobal\sql /s/e/y
echo d | xcopy .\oprajobschedule\oprajobschedule-db\v0.1.1\* D:\liquibase-3.8.7\OPRASYS\TEST-
10.1.19.7\oprajobschedule\sql /s/e/y
echo d | xcopy .\oprajobschedule\oprajobschedule-db\v0.2\* D:\liquibase-3.8.7\OPRASYS\TEST-
10.1.19.7\oprajobschedule\sql /s/e/y
echo d | xcopy .\oprasys\oprasys-db\v0.1.1\* D:\liquibase-3.8.7\OPRASYS\TEST-
10.1.19.7\oprasys\sql /s/e/y
echo d | xcopy .\oprasys\oprasys-db\v0.2\* D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\oprasys\sql
/s/e/y
```

1.6.4 后续配置多个schema的liquibase update命令:





这个不属于脚本管理的功能，在开发测试环境需要进行数据库的结构同步时可以用。

2.1在1.2的目录下建changelog-diff.xml文件，内容如下：

```
changelog-diff.xml - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ext="http://www.liquibase.org/xml/ns/dbchangelog-ext"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-
3.8.xsd
  http://www.liquibase.org/xml/ns/dbchangelog-ext http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-ext.xsd">
</databaseChangeLog>
```

```
<?xml version="1.0" encoding="UTF-8"?>

<databaseChangeLog

xmlns="http://www.liquibase.org/xml/ns/dbchangelog"

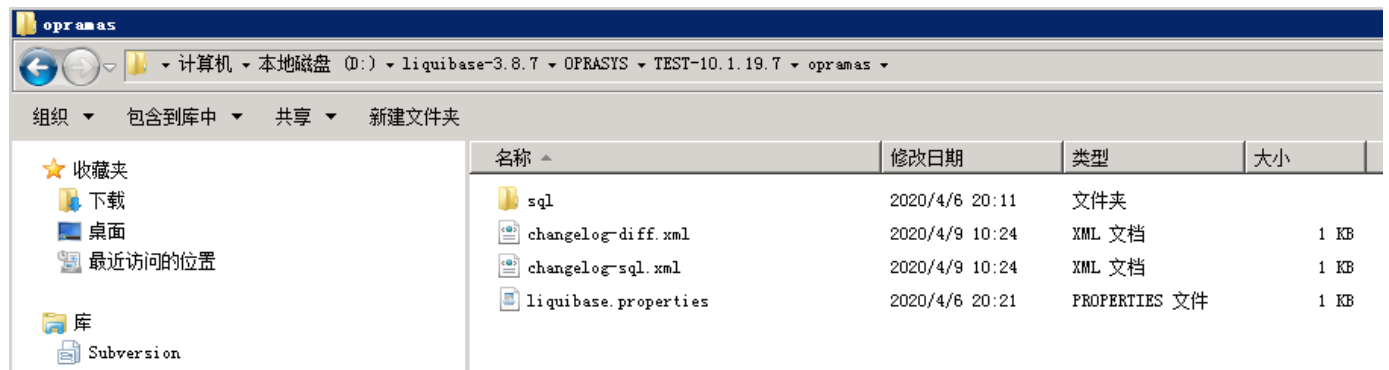
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:ext="http://www.liquibase.org/xml/ns/dbchangelog-ext"

xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-3.8.xsd
http://www.liquibase.org/xml/ns/dbchangelog-ext http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-ext.xsd">

</databaseChangeLog>
```

此时目录结构如下：



2.2建liquibase.properties文件，参考更新脚本第5步骤，如果已经有，省略此步骤。

liquibase.properties中为数据库连接地址和账号信息。

2.3配置jenkins命令

 **Jenkins**

Jenkins » OPRA »

### Enter an item name

opramas-TEST-diff

» Required field



**MultiJob Project**  
MultiJob Project, suitable for running other jobs



**构建一个自由风格的软件项目**  
这是Jenkins的主要功能.Jenkins将会结合任何SCM和任何构建系统来构建你的项目, 甚至可以构建软件以外的系统.

执行 Windows 批处理命令

Command

D:  
cd D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\opramasglobal  
copy ..\..\..\changelog-diff.xml changelog-diff.xml /y  
liquibase --changeLogFile=changelog-diff.xml --logLevel=debug diffChangeLog  
--referenceUrl=jdbc:postgresql://10.1.19.5:5432/opramasglobal --referenceUsername=opramasglobal --referencePassword=opramasglobal

See [the list of available environment variables](#)

Advanced...

执行 Windows 批处理命令

Command

D:  
cd D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\opramas  
copy ..\..\..\changelog-diff.xml changelog-diff.xml /y  
liquibase --changeLogFile=changelog-diff.xml --logLevel=debug diffChangeLog --referenceUrl=jdbc:postgresql://10.1.19.5:5432/opramas  
--referenceUsername=opramas --referencePassword=accal234

See [the list of available environment variables](#)

Advanced...

执行 Windows 批处理命令

Command

D:  
cd D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\opramasglobal  
liquibase --changeLogFile=changelog-diff.xml --outputFile=./changelog-diff.sql --overwriteOutputFile=true --logLevel=debug updateSQL

See [the list of available environment variables](#)

Advanced...

执行 Windows 批处理命令

Command

D:  
cd D:\liquibase-3.8.7\OPRASYS\TEST-10.1.19.7\opramas  
liquibase --changeLogFile=changelog-diff.xml --outputFile=./changelog-diff.sql --overwriteOutputFile=true --logLevel=debug updateSQL

See [the list of available environment variables](#)

Advanced...

第一步diffChangeLog是得到差异集合放在changlog-diff.xml中, 第二条updateSQL命令是把changlog-diff.xml里的差异生成sql文件格式, 加上--outputFile=./changelog-diff.sql 选项, 可以输出sql文件

diffChangeLog得到的changelog-diff.xml或者updateSQL根据changelog-diff.xml生成的changelog.sql不能直接执行update的原因:

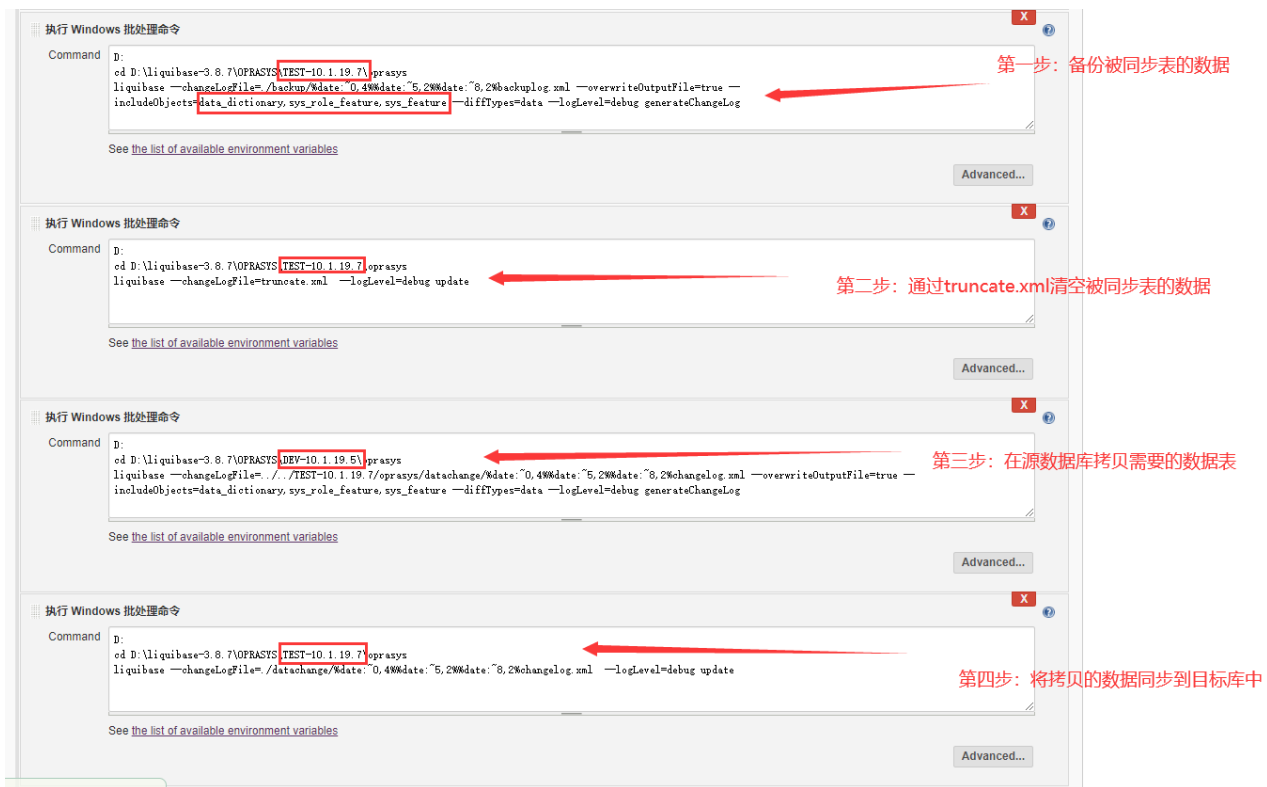
1. 字段重命名或者表重命名  
生成sql是直接删除旧的列, 创建新的列, 所以需要在生成脚本里改为alter table xxx rename 的语句, 否则当前数据和原始列一起删除。
2. 新的字段类型不符合当前数据

例如:

a. 精度改变, 注意整数位和小数位变小时, 数据库的数据不符合新的精度要求, 修改会报错, 需要增加修改数据的sql。

b. 可为空变非空, 原来null的值需要修改

2.4使用jenkins配置特定表的数据同步



1. 备份目录需提前创建，然后修改相应的路径，命令参考第一步。
2. 如果同步的表含有外键，尽量主子表同时同步、命令中按照（ includeObjects=子表，主表 ）的格式书写，否则会在数据库中发生错误，导致失败
3. truncate.xml需自己创建，并书写相应的执行语句。（可参照OPRASYS\TEST-10.19.7\oprasy\truncate.xml）

命令行学习参考

[https://www.liquibase.org/documentation/command\\_line.html](https://www.liquibase.org/documentation/command_line.html)