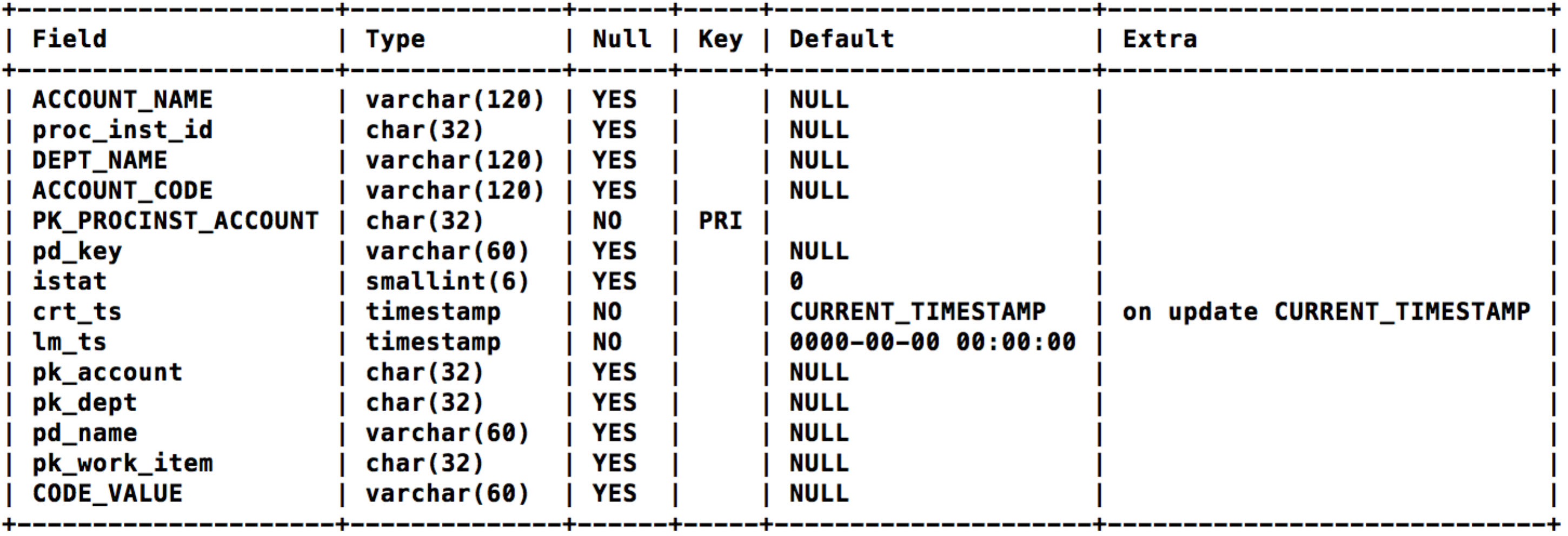
BPM服务

# 表结构

## rlt\_procinst\_account

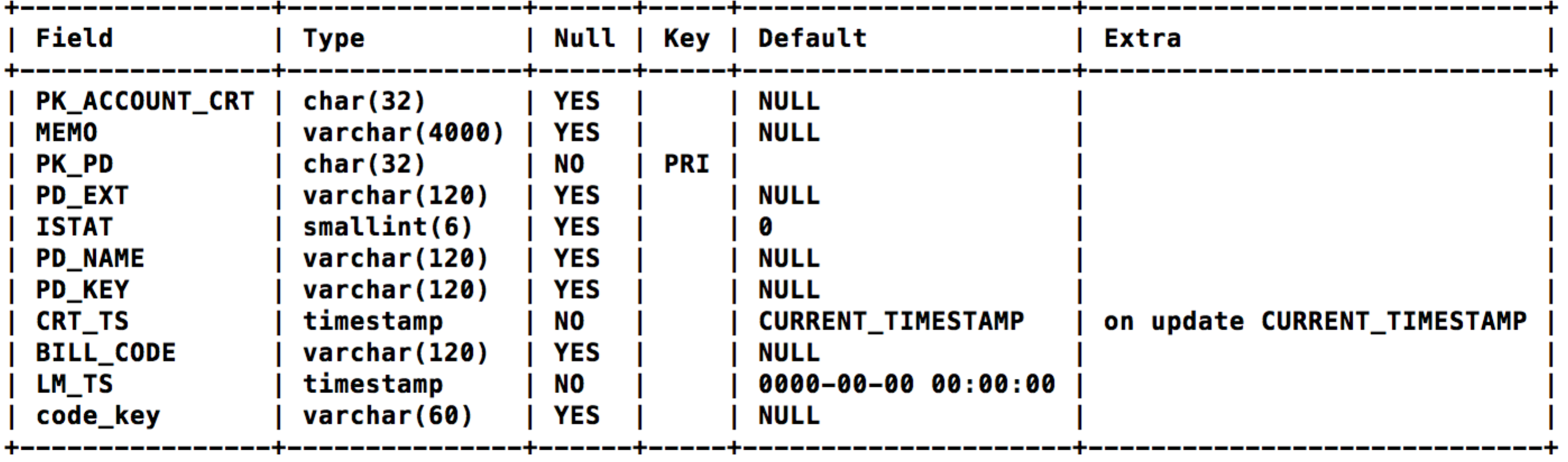
该表用于将用户信息-流程信息-单据主键关联起来，每个单据pk\_work\_item只能对应一个流程实例。



|  |  |  |
| --- | --- | --- |
| 字段 | 含义 | 说明 |
| PK\_PROCINST\_ACCOUNT | 主键 |  |
| PK\_ACCOUNT | 流程发起人主键 | MT\_OA\_ACCOUNT中的主键 |
| ACCOUNT\_NAME | 流程发起人名字 | 如董文婷 |
| ACCOUNT\_CODE | 流程发起人编码 | 如dongwenting02 |
| PK\_DEPT | 部门主键 | MT\_OA\_DEPT中的主键 |
| DEPT\_NAME | 部门名称 | 如开发组 |
| proc\_inst\_id | 流程实例ID |  |
| pd\_key | 流程定义ID | 在bpmn文件中定义的流程ID，如tm\_demand\_split |
| pd\_name | 流程定义名称 | 在bpmn文件中定义的流程名称，如要求管理 |
| pk\_work\_item | 记录主键 | 各单据实体的主键值 |
| crt\_ts | 创建时间 |  |
| lm\_ts | 最后修改时间 |  |
| istat | 状态 | 0(编辑):3(处理中):5(完成):7(已读) |
|  |  |  |

## mt\_oa\_pd

业务加工-流程注册，对应该表的内容，用于将单据信息和流程定义关联



|  |  |  |
| --- | --- | --- |
| 字段 | 含义 | 说明 |
| PK\_PD | 主键 |  |
| BILL\_CODE | 单据编码 | 如tm\_demand\_bill |
| PD\_KEY | 流程定义ID | 如tm\_demand\_split |
| PD\_NAME | 流程定义名称 | 如要求管理，这个是在界面配置的名称 |
| PD\_EXT | 流程辅助服务 | 流程执行前后须执行的操作，该类必须继承自ProcDefExtBizService |
| CODE\_KEY | 单据编号 | 如DEMAND\_CODE |
| MEMO | 备注 |  |
| PK\_ACCOUNT\_CRT | 创建人主键 | PK\_ACCOUNT\_CRT |
| crt\_ts | 创建时间 |  |
| lm\_ts | 最后修改时间 |  |
| istat | 状态 | 0:封存@3:发布 |
|  |  |  |

# 元数据

## BpmTestMeta

流程测试表

## BpmTestParamMeta

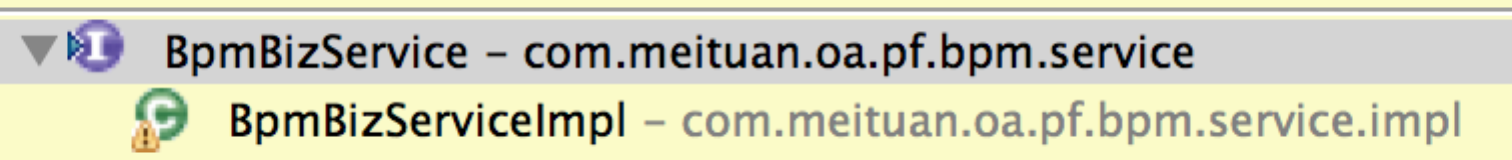
流程测试参数表

## ProcDefMeta

流程

## RltProcInstAccountMeta

流程实例与账号对应关系的元数据



# BpmManagementBizService

### deploy

部署bpmn文件

repositoryService载入流程文件

resources：bpmn文件列表

返回：部署ID

public String deploy(List<String> resources)

### unDeploy

根据部署ID，删除相关部署

public void unDeploy(String deploy\_id)

### deleteInstace

删除流程实例

processInstanceId：流程实例ID

reason：删除原因，可为null

public void deleteInstace(String processInstanceId,String reason)

### cleanDB

删除所有流程相关信息

public void cleanDB()

### fetchInstanceList

根据流程定义ID，返回流程实例列表

pd\_key：流程定义ID

public List<ProcessInstance> fetchInstanceList(String pd\_key)

### deleteHistoricProcInst

根据流程实例ID值，删除历史库和rlt\_procinst\_account中该流程的所有信息

private void deleteHistoricProcInst(String proc\_inst\_id)

### cleanProcInstByID

根据流程实例ID，首先查找是否有该实例，如果有，则获取该实例的ebs对象，ebs执行删除前的操作，删除该流程，ebs执行删除后的操作

public void cleanProcInstByID(String proc\_inst\_id,ClientEnv client)

### cleanProcInstByWorkItem

根据主键pk\_work\_item清除相应流程实例

public void cleanProcInstByWorkItem(String pk\_work\_item,ClientEnv client)

# BpmBizService

### \*start

先从rlt\_procinst\_account表中搜索是否有主键的当前流程实例，如果有则返回，然后从mt\_oa\_pd中搜索是否有相应的流程单据信息，执行流程启动前操作，启动流程实例，保存流程实例、流程定义、主键和单据号的信息到rlt\_procinst\_account中。

pd\_key：流程定义ID

workitem：GenericVO，发起流程所需的信息，必须包含pk\_main\_value，即主键。

message：发起流程的消息

env：用户信息

public String start(String pd\_key, GenericVO workitem, String message, ClientEnv env)

### fetchTaskList

获取任务列表

pd\_key：流程定义ID

env：用户信息

public List<Task> fetchTaskList(String pd\_key, ClientEnv env)

param：查询条件，可以为task\_id，start\_ts，end\_ts，TASK\_DESC

public List<Task> fetchTaskList(String pd\_key, GenericVO param, ClientEnv env)

### getFormValue

public String getFormValue(String task\_id, String key)

根据任务id，返回该任务中name在keyList中的FormValue

public GenericVO getFormValue(String task\_id, List<String> key\_list)

# CandiateBizService\*Impl

在BPMN文件中需要对参与者进行配置，如下

activiti:candidateUsers="#{candidate\_dept.fetch('资金组')}"

搜索candidate\_dept服务，可知其为CandidateBizServiceDeptImpl，

即调用public List<String> fetch(String pk\_dept)方法，传入部门名称，返回该部门所有ACCOUNT\_CODE。其他类都类似。

# 问题

**pk\_work\_item是什么意思？**

表中每条记录的主键值

**pd\_key和proc\_inst\_id区别？**

pd\_key是流程定义ID，即在bpmn文件中设置的Process的ID号。

proc\_inst\_id是流程实例ID，是流程变成流程实例后分配的ID号。

**配置文件加载路径？**

JUnit运行的测试类输出目录为test-classes，配置文件搜索路径首先为test-classes，找不到的话再找classes。

**优先级显示有问题，弹出对话框后，选中消失？**

主子表字段名不能一样