平台[PLAT]

数据库设计及备忘

|  |  |  |
| --- | --- | --- |
| 文件状态：  [√] 草稿  [ ] 正式发布  [ ] 正在修改 | **文件标识：** | PLAT-DATABASE(Simple User|Model|) |
| **当前版本：** | 0.1 |
| **作 者：** | 王晖 |
| **完成日期：** | 2014-3-19 |

**版 本 历 史**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本/状态** | **作者** | **参与者** | **起止日期** | **备注** |
| 0.1发布 | 王晖 |  | 2014-3-14 | 建立文档 |

**SQL文档**

**目 录**

[0. 文档介绍 4](#_Toc382897392)

[0.1. 文档目的 4](#_Toc382897393)

[0.2. 文档范围 4](#_Toc382897394)

[0.3. 读者对象 4](#_Toc382897395)

[0.4. 文档说明 4](#_Toc382897396)

[1. 数据库环境说明 4](#_Toc382897397)

[2. 数据库规范 5](#_Toc382897398)

[2.1. 表的命名规则 5](#_Toc382897399)

[2.2. 布尔值取值 5](#_Toc382897400)

[2.3. 键 5](#_Toc382897401)

[2.4. 字段/列 6](#_Toc382897402)

[2.5. 数值类型 7](#_Toc382897403)

[3. 编程注意 7](#_Toc382897404)

[4. 物理设计 7](#_Toc382897405)

[4.1. 表汇总 7](#_Toc382897406)

[4.2. 用户类(1) 7](#_Toc382897407)

[4.2.1. 用户[PLAT\_USER] 7](#_Toc382897408)

[4.3. 模块类(2) 8](#_Toc382897409)

[4.3.1. 系统模块[PLAT\_MODULE] 8](#_Toc382897410)

[4.3.2. 系统模块授权[PLAT\_MODULE\_ORG] 9](#_Toc382897411)

[4.4. 权限类(1) 9](#_Toc382897412)

[5. 数据逻辑视图 10](#_Toc382897413)

[6. 数据库管理与维护说明 10](#_Toc382897414)

1. 文档介绍
   1. 文档目的

描述情报平台底层数据对象实体，为产品建造提供核心信息架构。平台代码设计必须以数据库设计为依据。

* 1. 文档范围

包括底层平台的数据库及XML/json设计两类。

* 1. 读者对象

·产品开发组人员、测试人员、及产品相关人员；

·基于平台进行项目开发的人员；

* 1. 文档说明

·红色粗体字体部分为本此修改或需要注意的内容；

·蓝色粗体字体部分为需要完善和补充的内容；

·在表格中：黄色底的字段为主键；

·在表格中：淡蓝绿底的字段为外键或其值从其他表得到；

1. 数据库环境说明

希望满足跨数据库平台的要求，但目前只对MySql数据库。

若以Oracle为基准，表空间统一为platform，数据库用户统一为platform，口令platform，权限限制为资源访问（如：记录的增、删、改、查等）和应用级管理（建表、改表、删表、建索引等）。

**注意：各项目可以有自己的用户名，如海南项目——platform\_hn。**

1. 数据库规范
   1. 表的命名规则

·平台部分的表以“plat\_功能”命名，如plat\_user(用户表)；

·通用功能“common[\_模块]\_功能”命名，如common\_wcm\_channel(发布栏目表)；

·业务功能“app[\_模块]\_功能”命名，如app\_item\_person(人员信息)；

·表名最长为100个字符(由程序判断)，这受到PLAT\_SERIALNO表table\_name字段长度的限制。

·与权限相关的表用后缀“\_ORG”标明

* 1. 布尔值取值

·用数值类型表示布尔值，数据库种类型为“NUBMER”，java中类型为int；

·0（空）、1（是）、2（否）；说明：**由于java中数值int的默认值为0，为便于编程采用本条规则；**

·布尔字段包括包括字段：

1. **isValidate**—是否有效：1（有效）、2（无效），与del\_tag意思是相近的。

**b)has\_child**—是否有子节点：1（是）、2（不是）；

* 1. 键

·主键用id命名，UUID或长整型，若为长整型，则从1开始编号，若为UUID则长度为VARCHAR2(36)。

·父结点外p\_id命名；

·主子关系，子表到主表的外键用m\_id；

·其他外键命名：用功能缩写\_id，如u\_id或user\_id：用户ID，dd\_id：字典项ID；

**·实体表的ID用长整型，如用户表，情报表，公文表；内部关系表用UUID，如资产文件关联表，用户组用户关联表；**

* 1. 字段/列

1）全部小写：在Oracle中对象名称大小写不敏感，除非新建时用””框起来

2）每个英文单词或缩写与英文单词间用“\_”分开，如user\_name，un\_py，b\_code

3）是判断类型的用is+单词，并且单词首字母大写，如isValidate，isUnit

4）为了避免一些数据库系统的保留字，type，name，level均不作为表名或列明（包括其它一切用户对象的名称）

5）常用的命名及规范：

a)isValidate NUMBER //是否生效，1有效；2无效，默认值1

b)types NUMBER //类别，注意不用type，以免作为关键字；

c)descn VARCHAR2(2000) //说明字段

6）树结构通用规范，所有的数都要按这个建立：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段中文名** | **字段名** | **数据类型** | **空否** | **约束条件** | **说明** |
| 主键 | id | NUMBER | 非空 | 主键PK | 从PLAT\_SERIALNO生成 |
| 父结点ID | p\_id | NUMBER | 非空 | 默认值为0 | 用户组表中的ID(外键)  若是第一级用户组，此值为0 |
| 是否有子结点 | has\_child | NUMBER | 非空 | 默认值2无子 | 1有结点,2无子结点 |
| 字典项层数 | levels | NUMBER | 非空 | 默认值1无子 | 从1开始 |
| 排序 | sort | NUMBER | 非空 | 默认值0 | 数值大者靠前，从0-99998 |
| 索引编码 | index\_code | VARCHAR2(1100) | 非空 |  | 计算得到，每级10位长，前五位为本级排序，后五位为主键，支持100级 |
| 是否生效 | isValidate | NUMBER | 非空 | 默认值1有效 | 1有效；2无效 |

其中的index\_code，按照说明建立，如：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **P\_ID** | **NAME** | **SORT** | **INDEX\_CODE** |
| 183 | 0 | 互联网 | 9 | #9999000183 |
| 184 | 183 | 境内网站 | 9 | #9999000183#9999000184 |
| 185 | 183 | 境外网站 | 9 | #9999000183#9999000185 |

index\_code的规则为：[上级index\_code]连接[“#”+toStr(100000-(sort+1))+左补零至五位(id)]。通过这种编码规则可以用一个SQL语句按照排序方式变成深度树遍历的列表。

**排序字段越大越靠前，在树类型的排序中：**

**a)由于受到index\_code的影响，排序字段取值从0-99998，在录入和修改时要判断。**

**b)同结点的下一级结点项之间排序才有意义。**

* 1. 数值类型

所有用数值类型表示状态分类的字段，0都作为空值处理，这样避免程序中没有赋值的数值变量对数据库的影响。

1. 编程注意

·若主键采用长整型，并从PLAT\_SERIALNO生成，则SQLMap的namespace属性要和PLAT\_SERIALNO.table\_name字段的值项对应，否则不能争取获得序列号。**系统中强制规定：namespace、表明、PLAT\_SERIALNO.table\_name字段对应的值，三者一致。**

1. 物理设计
   1. 表汇总

下表按功能对表进行了划分，在具体介绍表的结构时也采用这一划分。

目前平台表分为如下：

|  |  |  |
| --- | --- | --- |
| **分类** | **表名** | **功能说明** |
| **1、用户类 (1表)** | PLAT\_USER（用户组） | 用户表 |
| **2、模块类(2表)** | PLAT\_MODULE（模块） | 系统模块 |
| PLAT\_MODULE\_ORG（模块授权） | 系统模块授权 |

* 1. 用户类(1)
     1. 用户[PLAT\_USER]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **表名** | **PLAT\_USER（用户）** | | | | |
| **字段中文名** | **字段名** | **数据类型** | **空否** | **约束条件** | **说明** |
| 用户ID | id | VARCHAR2(36) | 非空 | 主键PK | 从PLAT\_SERIALNO生成 |
| 登录名 | login\_name | VARCHAR2(100) | 非空 | 唯一索引 |  |
| 用户名 | user\_name | VARCHAR2(100) | 非空 |  |  |
| 密码 | password | VARCHAR2(100) | 可空 |  |  |
| **索引** | 1. id，主键 2. login\_name，IDX\_USER\_LNAME，保证登陆名称唯一   由于用户用户组信息进行缓存，因此除主键外，不建立其他索引 | | | | |
| **基本操作** |  | | | | |
| **程序使用** |  | | | | |

* 1. 模块类(2)

这些表用来维护系统模块关系，系统模块就是系统的基本组件。其中有权限相关结构。

* + 1. 系统模块[PLAT\_MODULE]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **表名** | **PLAT\_MODULE（系统模块）** | | | | |
| **字段中文名** | **字段名** | **数据类型** | **空否** | **约束条件** | **说明** |
| 模块ID | id | VARCHAR2(36) | 非空 | 主键PK | 从PLAT\_SERIALNO生成 |
| 父结点ID | pId | VARCHAR2(36) | 非空 | 默认值为0 | 本表中的ID(外键)  若是第一级模块，此值为0 |
| 层数 | levels | NUMBER | 非空 | 默认值1无子 | 从1开始 |
| 排序 | sort | NUMBER | 非空 | 默认值0 | 数值大者靠前，从0-999999 |
| 是否显示 | isValidate | NUMBER | 非空 | 默认值1有效 | 1有效；2无效  这里作为是否显示的意义 |
| 模块类型 | moduleTypes | NUMBER | 非空 | 默认值2用户定义模块 | 1：系统模块，2：用户定义模块 |
| 模块名称 | moduleName | VARCHAR2(200) | 非空 |  |  |
| 模块显示名称 | displayName | VARCHAR2(200) | 非空 |  | 界面上显示这个名称 |
| 功能链接 | url | VARCHAR2(400) | 可空 |  | 相对于上下文环境的路径 |
| 样式 | style | NUMBER | 可空 |  | 1一般；2portal |
| 图标 | icon | VARCHAR2(400) | 可空 |  | 相对于上下文环境的路径 |
| 说明 | descn | VARCHAR2(2000) | 可空 |  |  |
| **补充说明** | 1-外键不建立，只是说明关系  **2-types=1系统模块，此类模块只能修改显示名称。**  3-只有叶结点模块，才能有url，其他只作为分级处理。 | | | | |
| **索引** | 1. id，主键 2. index\_code，IDX\_MODULE\_ICODE，唯一索引   由于模块信息进行缓存，因此除主键外，不建立其他索引 | | | | |
| **基本操作** |  | | | | |
| **程序使用** |  | | | | |

* + 1. 系统模块授权[PLAT\_MODULE\_ORG]

系统模块授权表，可以对用户/用户组/角色授权

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **表名** | **PLAT\_MODULE\_ORG（系统模块授权）** | | | | |
| **字段中文名** | **字段名** | **数据类型** | **空否** | **约束条件** | **说明** |
| 模块授权关系ID | id | VARCHAR2(36) | 非空 | 主键PK | UUID |
| 系统模块ID | module\_id | NUMBER | 非空 | 默认值0，无意义 | 系统模块表中的ID(外键) |
| 授权类型 | types | NUMBER | 非空 | 默认值1 | 类型：1-用户组;2-用户;3-角色 |
| 授权对象ID | obj\_id | NUMBER | 非空 | 默认值0，无意义 | 授权对象Id：  types=1、ID=用户组ID；  types=2、ID=用户ID；  types=3、ID=角色ID； |
| **补充说明** | 1-外键不建立，只是说明关系  **2-注意：**  **目前权限对象是为扩展所用** | | | | |
| **索引** | 1. id，主键 2. module\_id,types,obj\_id，IDX\_PMO\_MIDTYPESOID | | | | |
| **基本操作** |  | | | | |
| **程序使用** |  | | | | |

* 1. 权限类(1)

系统平台中的权限相关表分散在各个类表中，这里作为总结。

**PLAT\_MODULE\_ORG（系统模块授权）：模块授权给角色、用户、用户组。**

1. 数据逻辑视图



1. 数据库管理与维护说明