# 富力信息系统代码应用规范

目录

[一、外系统对接 2](#_Toc58250184)

[1）外系统对接开发与接口文档 2](#_Toc58250185)

[2）统一接口调用日志 2](#_Toc58250186)

[3）中间表的使用要求 2](#_Toc58250187)

[二、代码注释 4](#_Toc58250188)

[1、注释形式统一 4](#_Toc58250189)

[2、注释内容准确简洁 4](#_Toc58250190)

[3、基本注释（必须加） 4](#_Toc58250191)

[4、特殊必加注释（必须加） 4](#_Toc58250192)

[5、注释格式： 5](#_Toc58250193)

[6、例子 5](#_Toc58250194)

[三、数据字典补充说明 6](#_Toc58250195)

[四、代码管理工具 7](#_Toc58250196)

# 一、外系统对接

## 1）外系统对接开发与接口文档

 当与外系统有业务关联发生，需要由开发顾问与实施顾问编写开发文档与接口文档：

1、开发文档业务说明；

2、接口设计说明：

2.1、传入参数名称与参数说明；

2.2、返回输出数据说明（接口必须需要返回调用失败或成功说明）

2.3、返回结果字段与字段结果描述；

3、接口测试方案；

4、接口修改成可在前台配置型或后台更新，争取不需要补丁就可以修复简单的接口变更；

5、接口调用需要有日志可供查询，日志具体记录。

## 2）统一接口调用日志

 为业务需要，查询调用日志时，统一的接口调用日志为内部日志不可向外提供

1、接口名称；

2、调用接口的外系统名称；

3、接口所属工程包，尽量详细到类与方法；

4、调用角色，01当前用户或02定时任务调用或03系统管理员调用；

5、调用时间，当前服务器录入时间；

6、接口返回调用信息，判断日志是否是异常信息 Y代表异常信息 N代表非异常信息；

7、日志详细信息。

## 3）中间表的使用要求

 与外系统对接时，有时会使用中间表对接系统，若中间表建在NC端这边，必须遵守以下要求：

1. 外系统只允许分配中间表的查询权限；
2. 编写中间表数据字典，详细说明中间表字段，规范可查看第三大点数据字典补充说明；
3. 外系统需要更新中间表时，需要调用接口，不允许直接修改中间表数据。

# 二、代码注释

## 1、注释形式统一

在整个应用程序中，使用具有一致的标点和结构的样式来构造注释。如果在其它项目中发现它们的注释规范与这份文档不同，按照这份规范写代码，不要试图在既成的规范系统中引入新的规范。

## 2、注释内容准确简洁

内容要简单、明了、含义准确，防止注释的多义性，错误的注释不但无益反而有害。

## 3、基本注释（必须加）

3.1、类（接口）的注释

3.2、构造函数的注释

3.3、方法的注释

3.4、全局变量的注释

3.5、字段/属性的注

备注：简单的代码做简单注释，注释内容不大于10个字即可，另外，持久化对象或

VO对象的getter、setter方法不需加注释。

## 4、特殊必加注释（必须加）

4.1、典型算法必须有注释；

4.2、在代码不明晰处必须有注释；

4.3、在代码修改处加上修改标识的注释；

4.4、在循环和逻辑分支组成的代码中加注释；

4.5、为他人提供的接口必须加详细注释。

备注：此类注释格式暂无举例。具体的注释格式自行定义，要求注释内容准确简洁。

## 5、注释格式：

5.1、单行(single-line)注释：“//……”

5.2、块(block)注释：“/\*……\*/”

5.3、文档注释：“/\*\*……\*/”

5.4、javadoc注释标签语法

@author 对类的说明 标明开发该类模块的作者

@version 对类的说明 标明该类模块的版本

@see 对类、属性、方法的说明 参考转向，也就是相关主题

@param 对方法的说明 对方法中某参数的说明

@return 对方法的说明 对方法返回值的说明

@exception 对方法的说明 对方法可能抛出的异常进行说明

## 6、例子

# 三、数据字典补充说明

 新增或修改数据库表结构时，请发送更新后的数据字典到数据库管理员，字典包括

1、字段名；字段名请使用标准英文，自定义项才允许使用英文+数字形式

2、字段中文名；中文注释

3、数据类型；

4、长度；

5、是否主键；

6、是否允许为空

7、默认值；

8、备注，也叫注释，说明字段用途，如枚举型档案，需提供枚举值对应的注释

例子：

# 四、代码管理工具

用友开发顾问会使用代码工程管理工具SVN，上传与下载最新代码到指定管理服务器上；

1. 用友开发顾问只需每次修改源代码前，需要下载最细代码；
2. 代码上传前，需下载最新代码与当前开发代码做校验，是否只是更新了当次的内容，才可以上传；
3. 由于NC代码不断更新迭代，因此每次更新会保留更新前备份；
4. 保留每个补丁代码到管理服务器上，以便代码回滚。