# [接口测试总结](https://wiki.intra.inzwc.com/pages/viewpage.action?pageId=17368157)  1 接口的管理和定义

     目前我们公司的接口使用wiki进行管理和维护。接口的定义比较完善，主要包含以下要素：接口名、URL、支持格式(json)、http请求方式(Post&Get)、是否

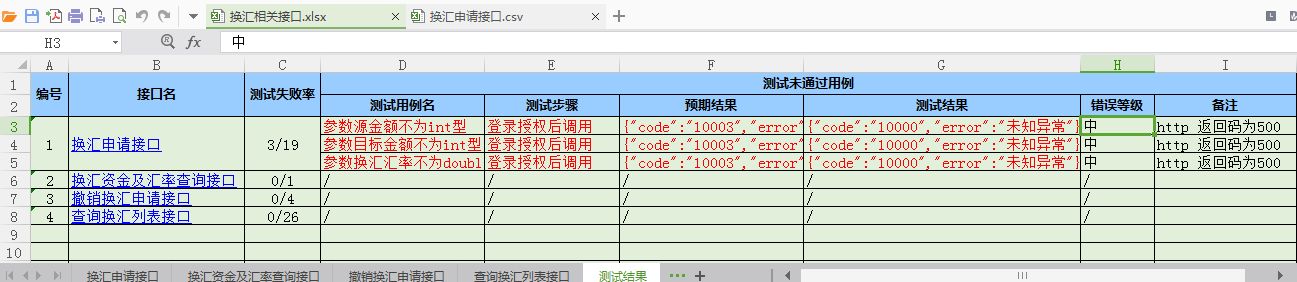
需要登录、请求参数、返回结果及样例。其中请求参数的要素有：参数名、参数类型、是否可选、参数说明。

# 2 接口测试用例的设计、管理和断言

## 2.1 测试用例的管理

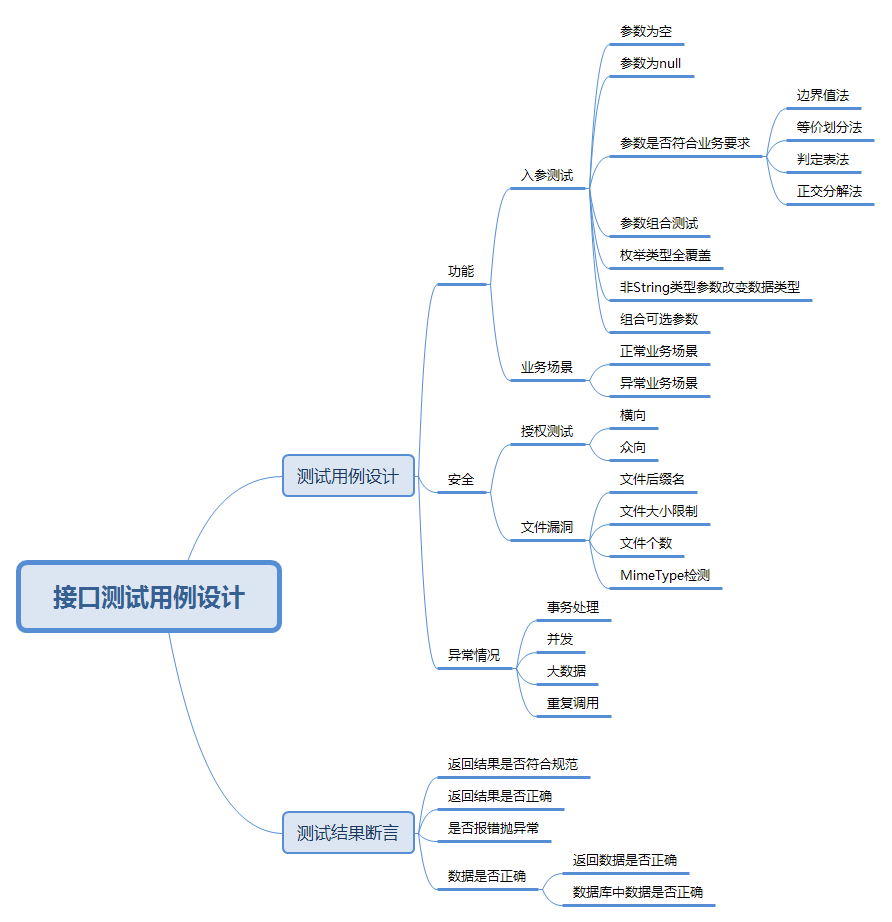
        目前测试用例的管理主要采用excel方式，将测试数据和对应的测试用例按照接口的组织结构存放于代码对应文件夹下，跟随代码存放于svn中。每个接口

测试用例的测试情况都会在该模块的测试结果页展示出来。该方式的优点是测试数据和测试结果一目了然，缺陷是维护比较麻烦，效果图如下：



## 2.2 测试用例的设计和断言

           下图是测试用例的设计和断言思维导图：



在实际测试过程中，测试用例很少涉及到异常情况。考虑到效率，目前只对正常的业务场景做后台数据校验，其他都是根据返回的结果进行断言。如果数据存放

在数据库中优先使用数据库中的数据进行校验，否则只能使用其他接口获取相关数据进行校验。

# 3 测试数据的创建与维护

      系统的数据一部分来源于用户的输入，一部分来源于第三方，通过网络爬虫爬取或者调用第三方接口获取。而接口数据的输出也可以根据使用对象划

分为两类，一种是用户相关数据，一种是大众化数据。数据的存储主要有两个地方，一个是我们公司的数据库，另外就是第三方数据库(如：柜台系统)。

用户输入的数据跟随用户走，其输出必定是用户相关数据。而大众化数据对于所有用户都一样，随着业务逻辑展示出来。综上分析可知，对于用户相关

接口，我们可以维护好用户的输入数据，这样就意味着我们可以获取到固定的输出数据来进行校验。基于上面的思想指导，我们可以在系统中针对不同的

接口和测试用例创建不同的用户数据，测试完毕后还原测试数据，只要输入的用户数据稳定，我们就可以愉快的进行接口数据校验了。但是由于某些数据

存放于第三方系统中(如：柜台系统)，这部分用户数据我们不能作任何还原处理也无法通过访问数据库获取，目前想到的方法是通过其他接口获取数据进行

校验，缺点是对于某些查询类的接口数据无法校验。至于大众化数据只要根据业务逻辑来获取数据库中的数据和接口返回的数据进行对比校验即可。

       理论是丰满的，然而实践起来确是残忍。如果针对单独的接口和测试用例使用单独的用户数据来进行测试，这无疑是个噩梦。目前的做法是尽量共用

用户数据，尽量避免测试用例数据冲突。



# 4 接口自动化测试持续集成方案

       目前持续集成方案采用的是Ant+Jenkins+jmeter，测试报告采用的是官方的xslt模板改造而成，详情可以查看：[接口自动化测试报告](http://192.168.91.45:8080/job/%E5%8D%8E%E7%9B%9B%E8%AF%81%E5%88%B8%E6%8E%A5%E5%8F%A3%E6%B5%8B%E8%AF%95/job/%E5%8D%8E%E7%9B%9B%E6%B8%AF%E8%82%A1%E7%B3%BB%E7%BB%9F%E7%A7%BB%E5%8A%A8%E6%8E%A5%E5%8F%A3%E6%B5%8B%E8%AF%95/HTML_Report/)。

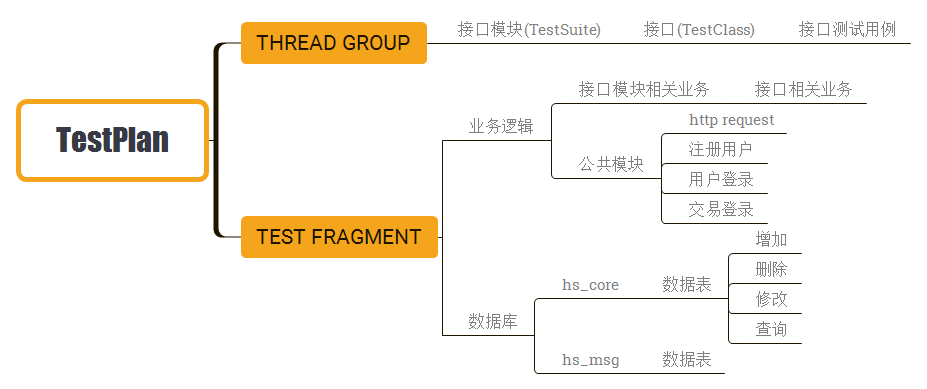
该报告优点是接口测试用例一目了然，能够过滤出失败的测试用例，也能查看详细的测试情况。缺陷是样式不够友好，测试总结情况不够直观也不够全面，

统计的数据都是按照sampler来进行的，而不是根据实际的测试用例。另外jenkins上面安装了一个performance trend插件可以查看这段时间的性能和错误趋

势，设置了html report保存每日的测试报告用于对比和排查测试结果。

# 5 测试脚本框架

   整体的测试脚本框架如下图所示：



# 6 总结

## 6.1  目前的问题

       目前已经基本实现接口的自动化和持续集成，尚有部分接口用例尚未实现。目前存在的问题如下：

* 添加发送邮件功能，将相关测试情况发送邮件给相关人员；
* 优化统计界面，去掉统计页面中不相关的sample信息，将相关的错误用例以及错误详情链接展示在页面中，方便查看；
* 模拟实现绑定第三方登录(微信、微博)；

## 6.2 今后的规划

     后续如果时间和精力充足的话会按照以下内容进行优化：

* 添加发送邮件功能，将相关测试情况发送邮件给相关人员；
* 优化统计界面，去掉统计页面中不相关的sample信息，将相关的错误用例以及错误详情链接展示在页面中，方便查看；
* 模拟实现绑定第三方登录(微信、微博)；
* 错误用例重跑功能；
* 第三方数据模拟系统，针对某些存在第三方系统的数据，我们也许可以通过模拟该系统来进行数据校验。对于一个纯  
  净的用户，只要我们操作相关的第三方系统接口，我们就同时操作模拟系统，并将相关数据存入系统中。后续我们需  
  要校验相关接口的时候我们就可以取这里面的数据来跟对应接口返回的数据进行对比即可；