

图测

朱月飞

京东商城-Y事业部-测试开发工程师

下一代
软件研发
SOFTWARE
DEVELOPMENT

A UI自动化测试的苦恼

像用户一样去测试 **B**

C 图测实践

展望未来 **D**

UI自动化测试的苦恼

01
part

1. 测试排期马上要到了
2. 这次测试要覆盖前后端的所有功能点
3. 据说明天还要发布一个新版本
4. 我的自动化测试脚本又跑不通了
5.

01

测试时间有限制

02

应用复杂

03

需求不固定



艰难的UI自动化测试高手之路

01

手上功夫好

- ① 精通编程语言
- ② 精通自动化测试工具
- ③ 精通元素定位技巧

02

内功深厚耐力足

- ① 唯一不变的就是变化
- ② 脚本数量爆炸

03

记忆力超群

- ① 非结构化
- ② 业务与脚本的孤立
- ③ 无法统计覆盖率



像用户一样去测试

02
part

像用户一样去测试



- I. 关注页面上显示的内容
- II. 关注整体业务的正确性和稳定性
- III. 对于页面的变化快速适应

像用户一样去测试

- 高层抽象，关注测试本身需求
- 从整体上显示自动化测试的内容

- 上手门槛低，操作简单
- 快速实现自动化测试

- 最大程度自动化
- 快速应对页面的变更



图测实践

03
part

图测-像用户一样测试

一款MBT自动化测试工具，将测试脚本与FSM状态图有机结合为基础，保证测试用例随时处于最新状态，减少了自动化测试中令人头疼的脚本维护工作。

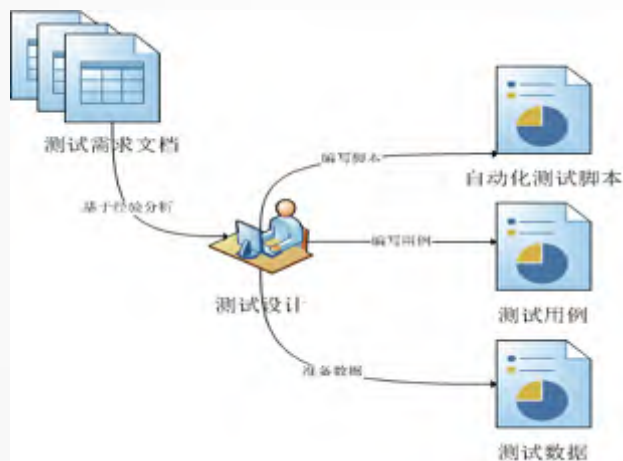
图测宗旨



1. 维护成本小，提升应用范围，从回归测试到日常功能测试；
2. 快速实现自动化，扩大应用人员，从测试人员到研发人员；
3. 分层思想灵活扩展，扩展测试类型，从UI测试到接口测试；

图测的测试模式

传统自动化测试



- 测试脚本维护成本高
- 测试用例覆盖程度无法科学统计
- 维护工作重复性高

基于模型测试



- 维护测试对象模型
- 科学统计测试用例覆盖程度
- 自动生成测试脚本，大大节省人力资源
- 需求阶段即开始建模工作，提前沟通发现问题
- 克服人工带来的风险疏忽
- 支持敏捷，拥抱变化
- 增加测试继承性和规范性

图测的框架结构

chrome extension浏览器插件



http request

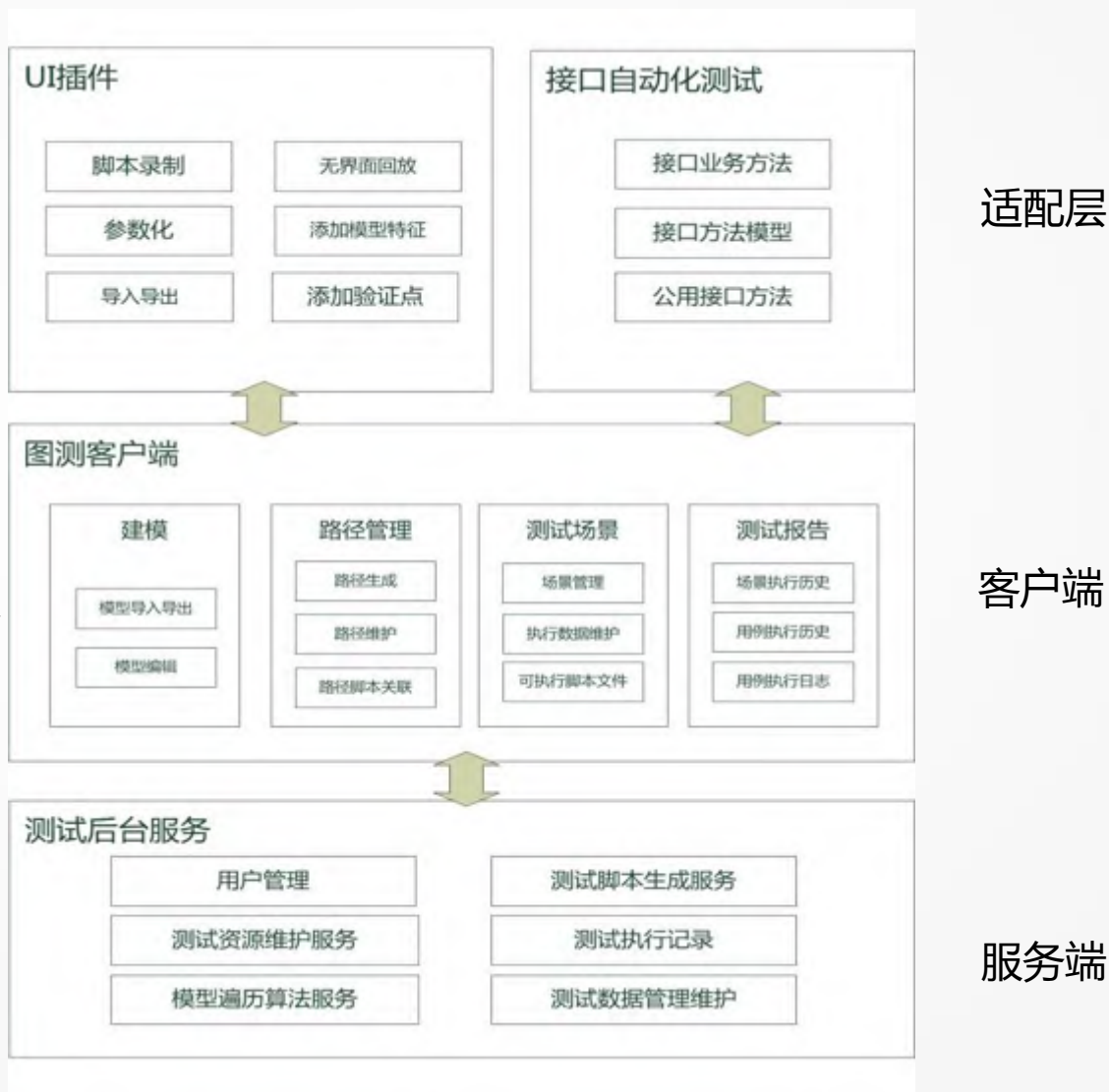


NW+Nodejs+express本地服务



http request

Nodejs+express后台服务



图测的关键技术点分析之模型表现形式

1、graphml文件

一种表示图的XML文件，支持有向图、无向图、混合图。

<http://graphml.graphdrawing.org/>

2、FSM状态图

用来描述对象在其生命周期中的状态转换，已经触发转换的条件以及响应的理论。

WEB应用系统特性恰好符合。

测试人员通过模型展示把关注点从复杂的被测应用脱离出来，专注测试业务本身



```

<?xml version="1.0" encoding="UTF-8"?>
<graphml xmlns=""
  xmlns:xsi=""
  xsi:schemaLocation=""
  >
  <graph id="0" edgedefault="undirected">
    <node id="n0"/>
    <node id="n1"/>
    <node id="n2"/>
    <node id="n3"/>
    <node id="n4"/>
    <node id="n5"/>
    <node id="n6"/>
    <node id="n7"/>
    <node id="n8"/>
    <node id="n9"/>
    <node id="n10"/>
    <edge source="n0" target="n2"/>
    <edge source="n1" target="n2"/>
    <edge source="n2" target="n3"/>
    <edge source="n3" target="n0"/>
    <edge source="n4" target="n6"/>
    <edge source="n6" target="n5"/>
    <edge source="n5" target="n7"/>
    <edge source="n8" target="n8"/>
    <edge source="n8" target="n7"/>
    <edge source="n8" target="n9"/>
    <edge source="n8" target="n10"/>
  </graph>
</graphml>
  
```

图测的关键技术点分析之路径生成

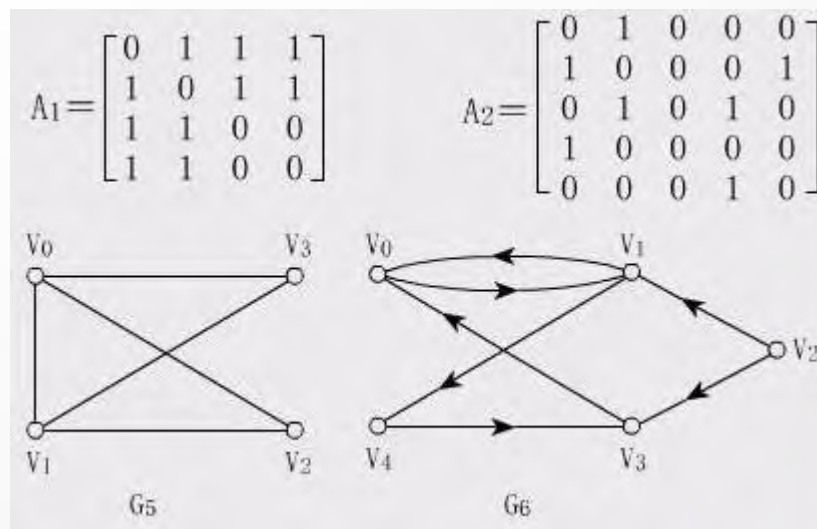
1、有向图转换矩阵

Graphml模型文件 \longrightarrow 邻接矩阵数据结构 \longrightarrow 遍历算法

2、简单易理解的路径生成策略

全覆盖模式

自定义模式



图测的关键技术点分析之辅助工具

脚本生成器：按照指定执行脚本模型，自动拼装可执行用例脚本

场景执行文件夹：场景为单位，自动生成可执行文件内容，快速方便执行用例

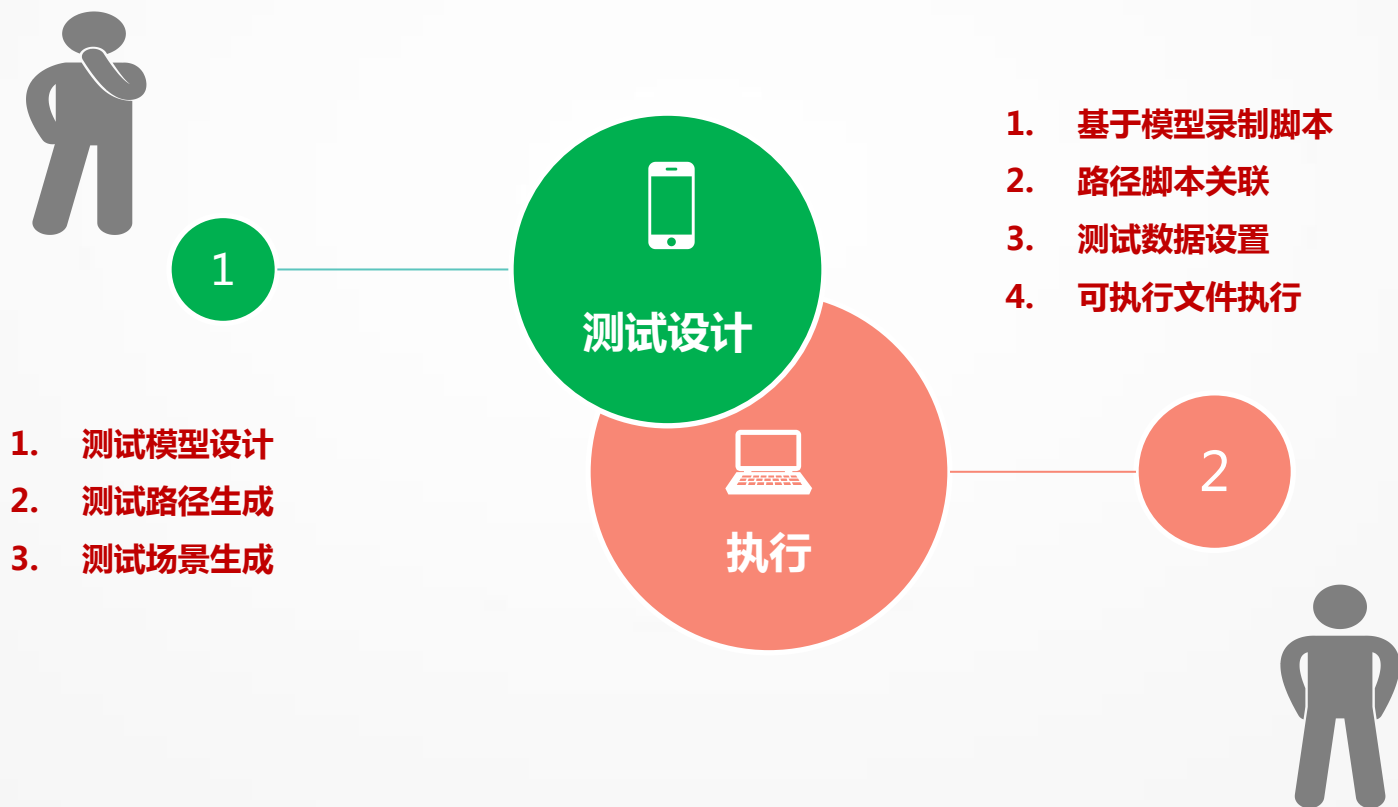


录制插件：根据页面操作自动生成脚本命令

测试数据：管理测试数据，支持脚本参数化

测试执行：测试框架、日志记录、截图、测试报告

图测的分层工作



模型设计

路径生成

场景生成

测试案例：京东购买手机促销流程测试
分析方法：页面状态迁移分析



表单提交



动态页面

模型设计

路径生成

场景生成

测试设计

页面状态 (S)					
START					
京东登录页					
京东首页					
手机品类首页					
锤子促销页					
华为促销页					
苹果促销页					
小米促销页					
手机单品页					
购物车					
结算页					
END					

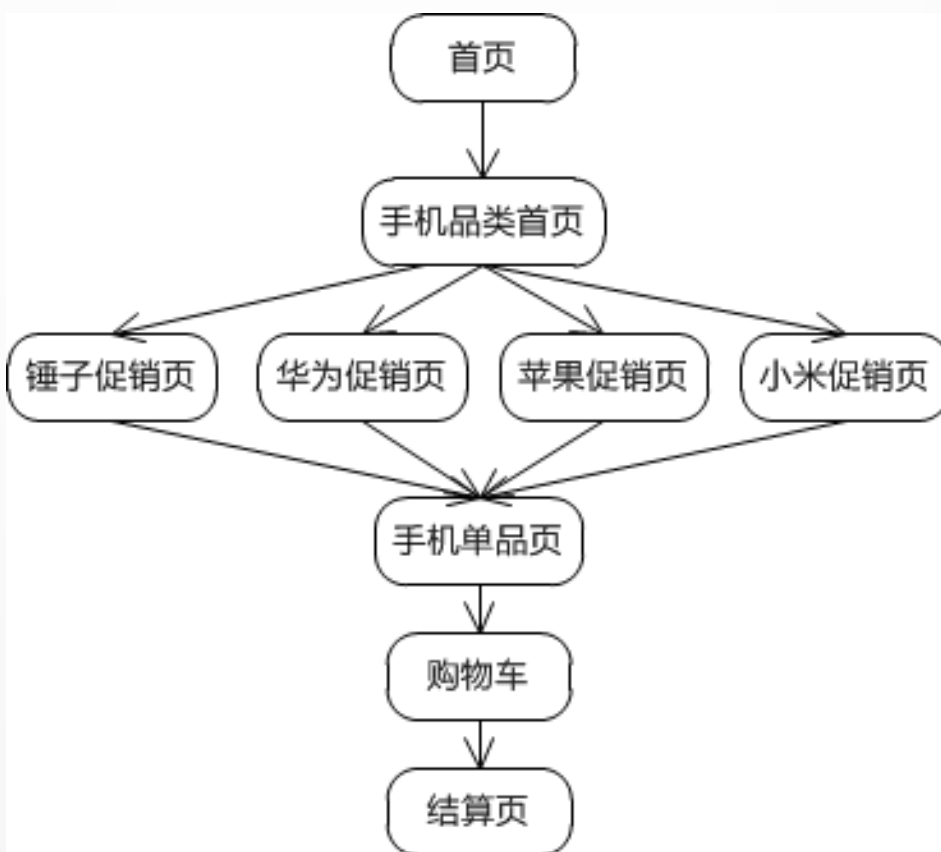
页面状态迁移表					
边名称	起始状态	目标状态	事件名称	类型	参数
e0	START	京东首页	进入首页	导航	无
e1	京东首页	手机品类首页	进入手机品类	导航	无
e2	手机品类首页	锤子促销页	进入锤子促销页	导航	无
e3	手机品类首页	苹果促销页	进入苹果促销页	导航	无
e4	手机品类首页	小米促销页	进入小米促销页	导航	无
e5	手机品类首页	华为促销页	进入华为促销页	导航	无
e6	锤子促销页	手机单品页	抢购锤子	导航	无
e7	苹果促销页	手机单品页	抢购苹果	导航	无
e8	小米促销页	手机单品页	抢购小米	导航	无
e9	华为促销页	手机单品页	抢购华为	导航	无
e10	手机单品页	购物车	加入购物车	导航	无
e11	购物车	结算页	结算	导航	无
e12	结算页	END	关闭窗口	导航	无

- 列举页面状态
- 分析页面状态迁移关系
- 转换模型文件

模型设计

路径生成

场景生成



模型设计

路径生成

场景生成

```
<?xml version="1.0" encoding="UTF-8"
standalone="no"?>
<graphml>
<graph edgedefault="directed" id="G">
  <node id="START"></node>
  <node id="首页"></node>
  <node id="手机品类首页"></node>
  <node id="锤子促销页"></node>
  <node id="华为促销页"></node>
  <node id="苹果促销页"></node>
  <node id="小米促销页"></node>
  <node id="手机单品页"></node>
  <node id="购物车"></node>
  <node id="结算页"></node>
  <node id="END"></node>
  <edge id="进入首页" source="START" target="首页"></edge>
  <edge id="进入手机品类" source="首页" target="手机品类首页"></edge>
  <edge id="进入锤子促销页" source="手机品类首页" target="锤子促销页"></edge>
  <edge id="进入苹果促销页" source="手机品类首页" target="苹果促销页"></edge>
  <edge id="进入小米促销页" source="手机品类首页" target="小米促销页"></edge>
  <edge id="进入华为促销页" source="手机品类首页" target="华为促销页"></edge>
  <edge id="抢购锤子" source="锤子促销页" target="手机单品页"></edge>
  <edge id="抢购苹果" source="苹果促销页" target="手机单品页"></edge>
  <edge id="抢购小米" source="小米促销页" target="手机单品页"></edge>
  <edge id="抢购华为" source="华为促销页" target="手机单品页"></edge>
  <edge id="加入购物车" source="手机单品页" target="购物车"></edge>
  <edge id="结算" source="购物车" target="结算页"></edge>
  <edge id="关闭窗口" source="结算页" target="END"></edge>
</graph>
</graphml>
```

```
<edge id="进入首页" source="START" target="首页"></edge>
<edge id="进入手机品类" source="首页" target="手机品类首页"></edge>
<edge id="进入锤子促销页" source="手机品类首页" target="锤子促销页"></edge>
<edge id="进入苹果促销页" source="手机品类首页" target="苹果促销页"></edge>
<edge id="进入小米促销页" source="手机品类首页" target="小米促销页"></edge>
<edge id="进入华为促销页" source="手机品类首页" target="华为促销页"></edge>
<edge id="抢购锤子" source="锤子促销页" target="手机单品页"></edge>
<edge id="抢购苹果" source="苹果促销页" target="手机单品页"></edge>
<edge id="抢购小米" source="小米促销页" target="手机单品页"></edge>
<edge id="抢购华为" source="华为促销页" target="手机单品页"></edge>
<edge id="加入购物车" source="手机单品页" target="购物车"></edge>
<edge id="结算" source="购物车" target="结算页"></edge>
<edge id="关闭窗口" source="结算页" target="END"></edge>
</graph>
</graphml>
```

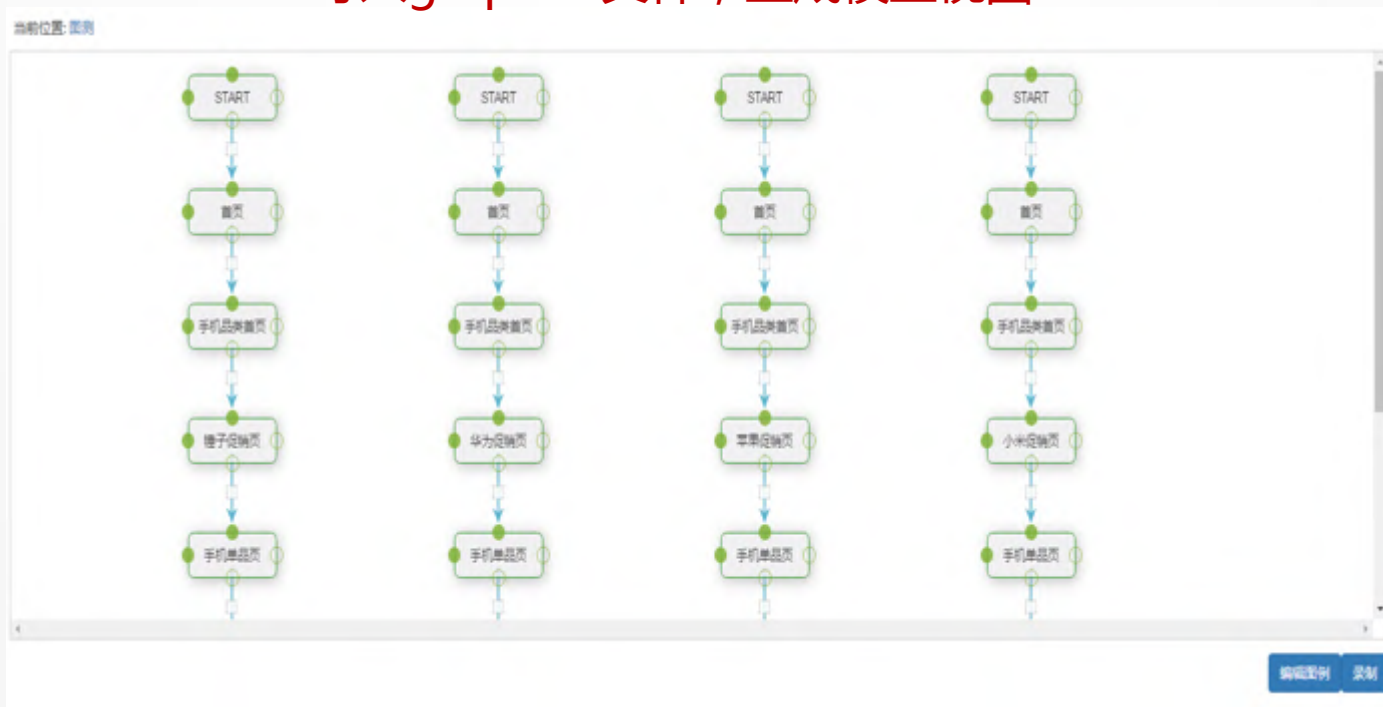
模型设计

路径生成

场景生成

测试案例：京东购买手机促销流程测试

导入graphml文件，生成模型视图

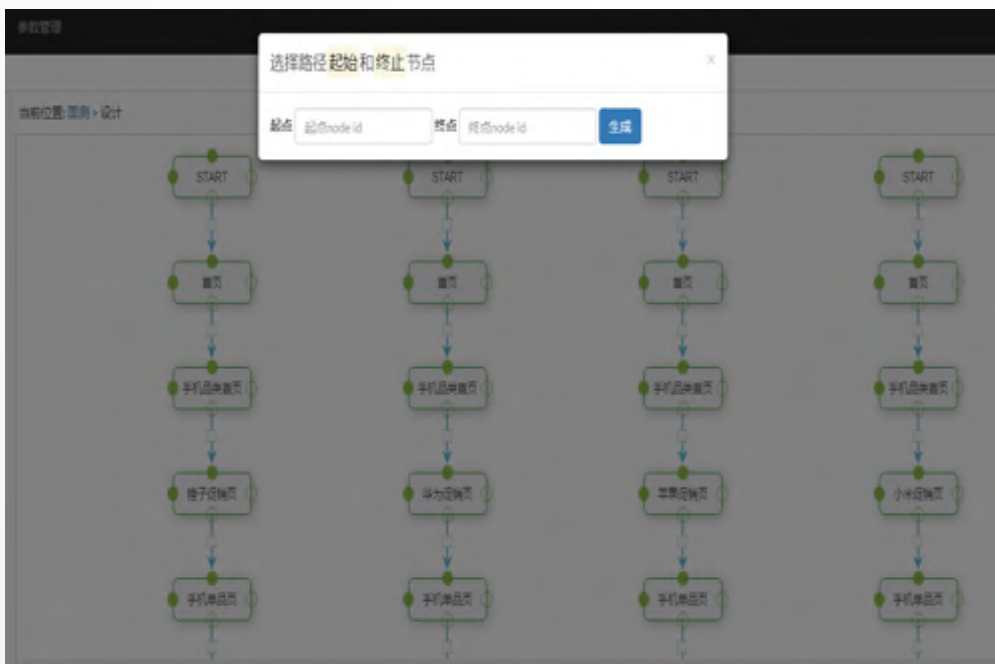


模型设计

路径生成

场景生成

根据起始节点和结束节点的位置，生成满足覆盖条件的路径



当前模型：京东商城购物模型

tid的路径根目录

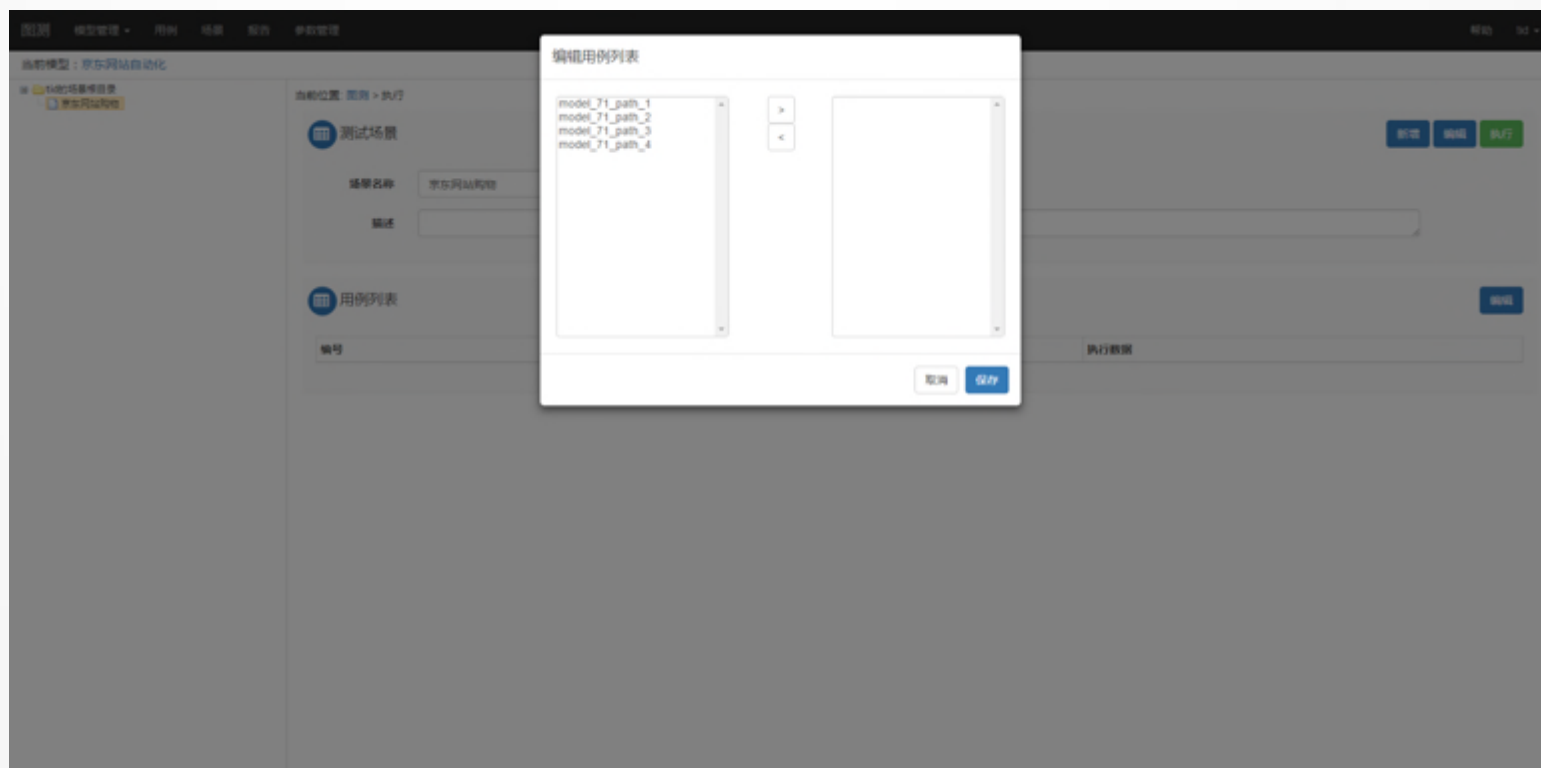
- 手机购物路径集合
 - model_71_path_1
 - model_71_path_2
 - model_71_path_3
 - model_71_path_4

模型设计

路径生成

场景生成

测试设计



模型设计

路径生成

场景生成

测试

模型管理

用例

场景

报告

参数管理

帮助

当前模型：京东商城自动化

140个场景项目

京东商城购物

当前位置： 用例 > 执行

测试场景

新增

编辑

执行

场景名称

京东商城购物

描述

用例列表

编辑

编号	名称	类型	执行数据
0	model_71_path_1	用例	+ 测试数据
1	model_71_path_2	用例	+ 测试数据
2	model_71_path_3	用例	+ 测试数据
3	model_71_path_4	用例	+ 测试数据



启动录制插件，开始自动记录测试步骤，与模型信息同步

测试执行





- 插件功能：
1. 操作命令记录解析
 2. 添加变量
 3. 添加检查点
 4. 调整生成代码



选择路径，操作页面，添加变量，增加检查点



模型脚本制作后，开始拼装路径脚本

当前模型：京东商城购物模型

tid的路径根目录

- 手机购物路径集合
 - model_71_path_1
 - model_71_path_2
 - model_71_path_3
 - model_71_path_4

用例信息

用例名称: model_71_path_1

用例脚本:

```
点击：糖子 ( di:fore1 > dd item-cont > ul clearfix > li:nth-child(5) > a, 13, 9, 0 ), function(){ return driver.sleep(300).wait('di:fore1 > dd item-cont > ul clearfix > li:nth-child(5) > a', 30000).sleep(300).mouseMove(13, 9).click(0); };
```

导出脚本

名称	用例	执行数据
model_71_path_1	用例	+ 测试数据
model_71_path_2	用例	+ 测试数据
model_71_path_3	用例	+ 测试数据
model_71_path_4	用例	+ 测试数据

测试业务与测试脚本分离解耦



对脚本中参数化的内容赋值

测试用例执行数据

测试用例名称: model_71_path_1

+ 新增

Search

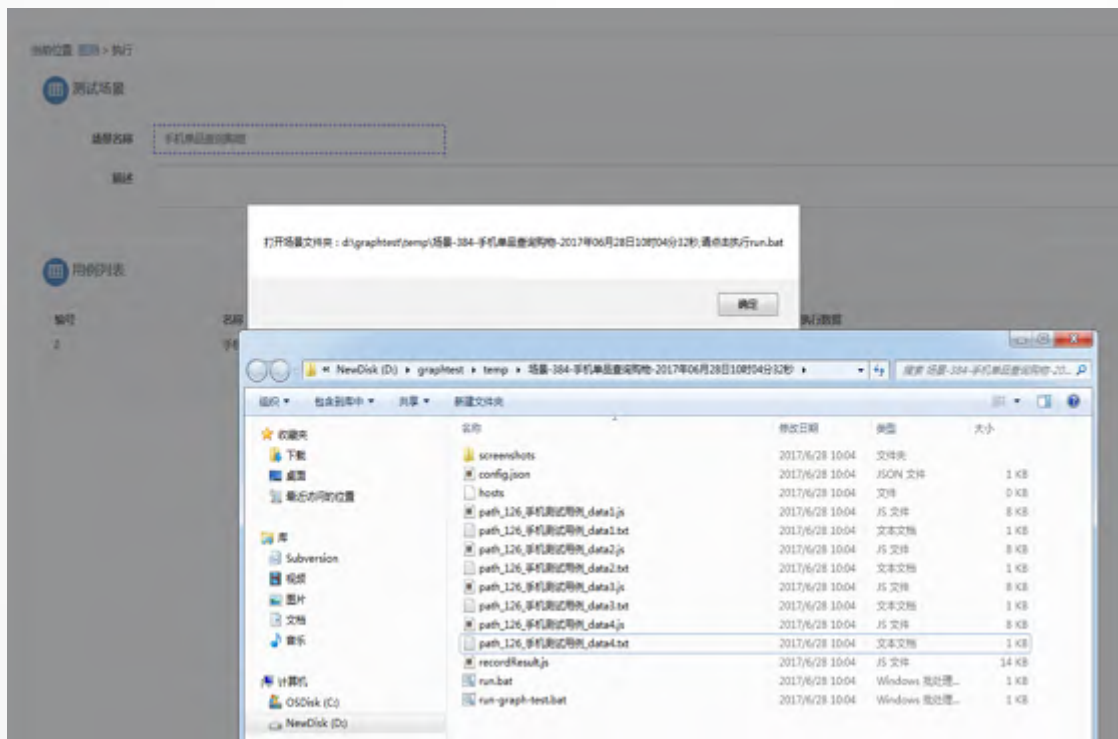
序号	查询手机商品	操作
1	索尼	删除
2	小米	删除
3	华为	删除
4	苹果	删除

Showing 1 to 4 of 4 rows

取消 保存



根据路径与数据的结合，在本地指定目录下生成可执行文件



- ① 脚本本地化
- ② 环境自动化
- ③ 过程系统化
- ④ 资源公共化

1

模型脚本

2

路径脚本

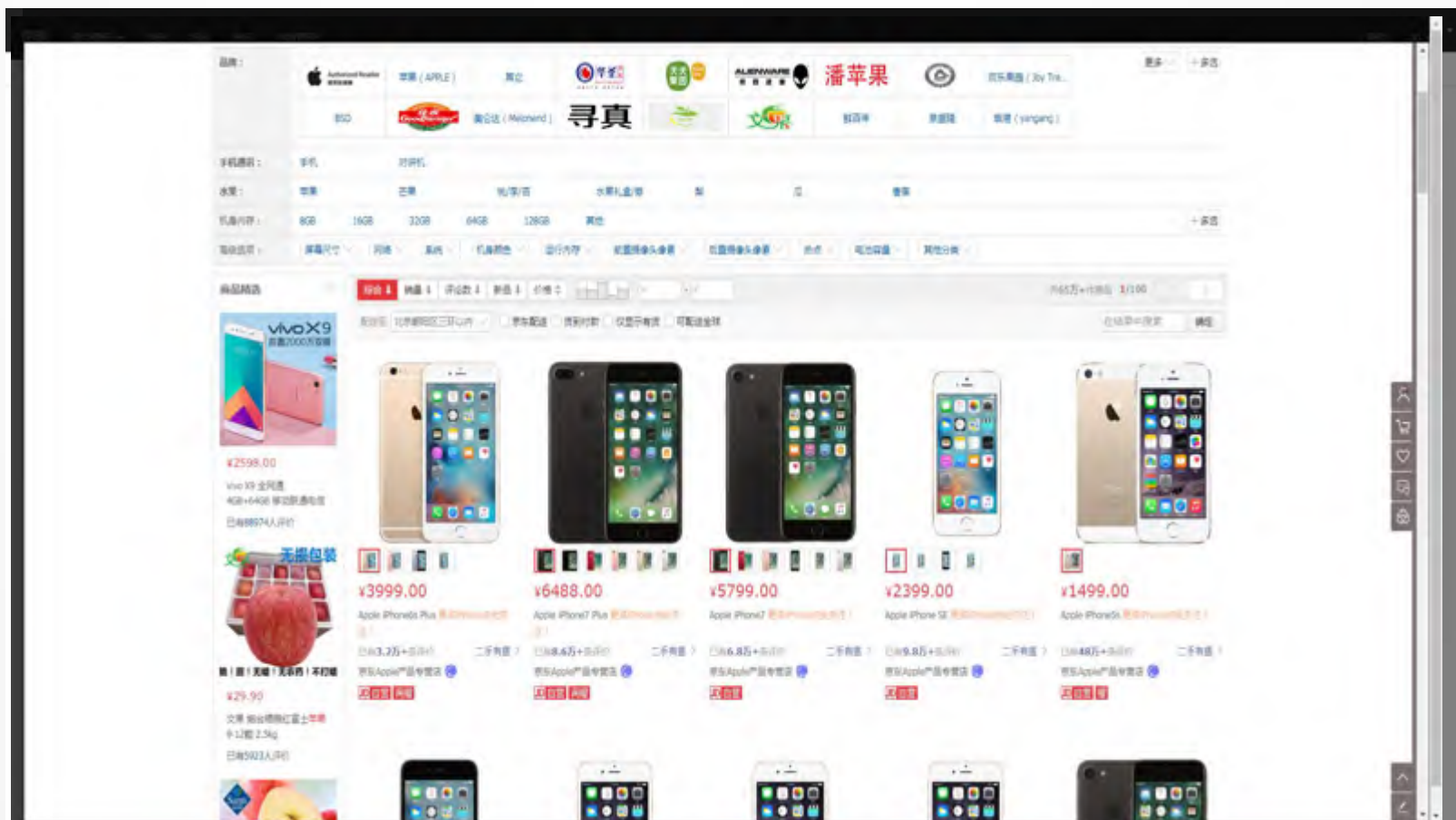
3

测试数据

4

场景执行

在线日志报告，详细记录执行内容以及UI展示截图



展望未来

04
part



遇到的问题！

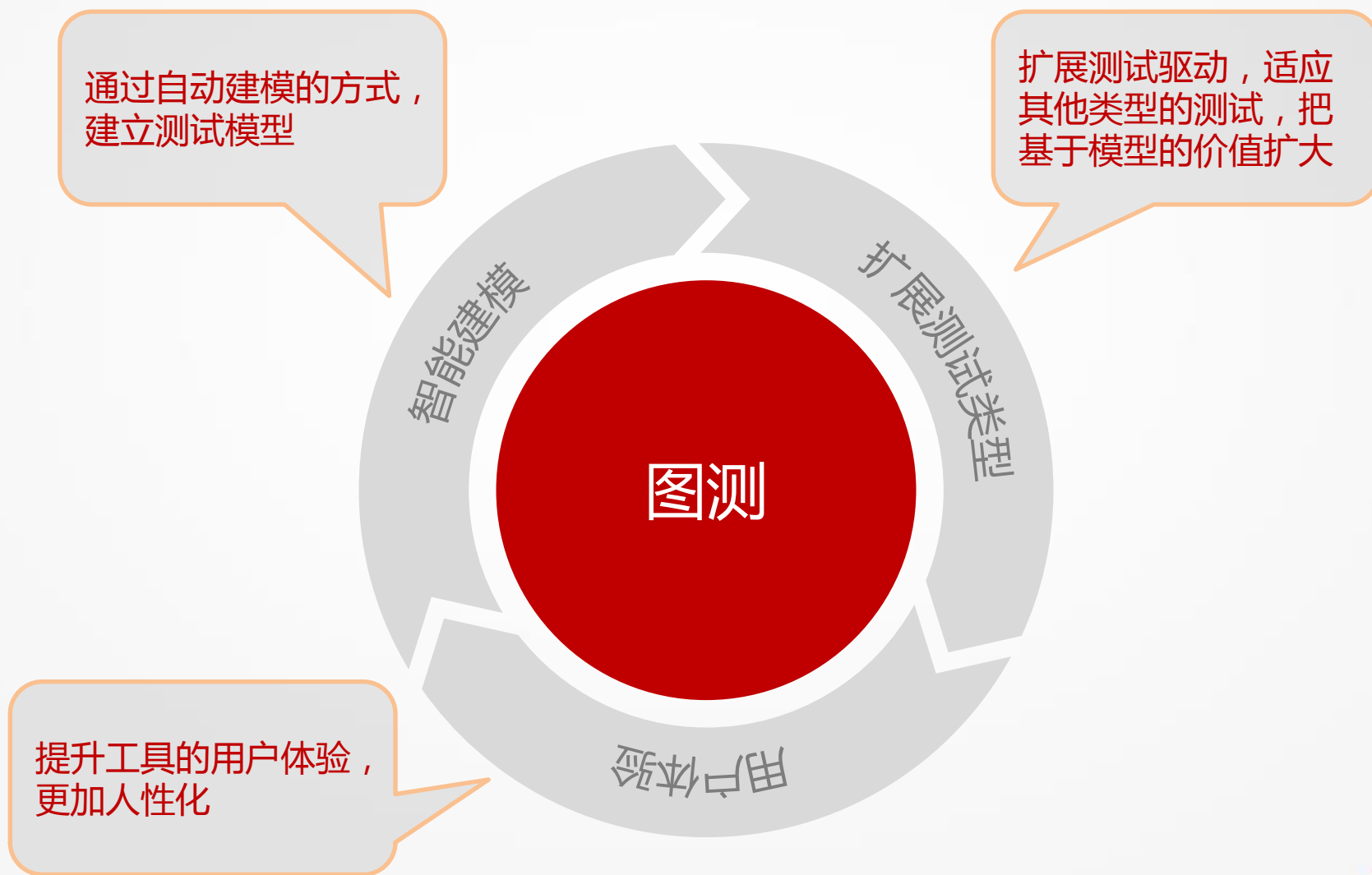
模型的建立

如何建立起符合当前测试业务
需求的模型内容？
建模的成本是否可以接受？



脚本的问题调试

自动生成的脚本执行的问题，
如果快速定位到具体模型内容
中



THANKS

谢 谢 聆 听

