

Axure RP 7.0 从入门到精通 Web+APP 产品经理原型设计

小楼一夜听春雨 编著

人民邮电出版社
北 京

内 容 提 要

本书是一本介绍使用 Axure RP 7.0 软件的图书，通过对基础知识、实际案例以及实战项目的讲解，帮助读者学会如何进行互联网产品原型的开发。

本书共分为 9 章。第 1~2 章主要讲解软件安装及基础功能的操作和使用。第 3~8 章对软件更深层次的应用进行了详细的讲解。特别是对一些功能复杂的元件，以及让新手望而生畏的变量、函数等内容进行了详细的讲解和案例的展示。第 9 章展示了一个比较完整的案例制作过程，从原型的结构、页面的搭建、交互的实现等方面，详细讲述了一个项目原型从无到有的过程。

本书适合零基础的读者，能够让从未接触过原型开发工作的读者迅速、全面地掌握原型开发技能，本书可以作为学校相关课程配套教材，或互联网公司、高新科技企业等新人内训的教程。

第一次写书，也是第一次写序。

写作初衷

2013年6月份开始，我创办了Axure原创教程网（iaxure.com），至今，已经有两年多的时间了。在这期间，我一直专注于Axure的文字教程、案例原型以及各种元件的制作和分享，将iaxure.com打造成了国内专业的Axure原创教程网站。

除了网站的内容，我还录制了3套完整的视频课程。较早的2套在多贝网上免费分享，最后一套“Axure RP 7.0从入门到精通”一直是网上销量最高的课程。另外，我和我的朋友在kouer.com开办了定期的Axure线下培训，进行面授课程。

不管是在网络上的文字分享、视频的录制，还是线下的培训，都让我从中收获了很多。在这些过程中，看到了大量学习者实际的学习需求、学习难点以及学习兴趣所在。我也一直在思考，学习Axure的群体最需要做什么？所以，从今年年初接到出版社的邀约后，我就开始将这些宝贵的收获和经验，融入到自己的第一本书里面。经过长达半年的构思和写作完成了这本书。希望这本书与我以往的作品一样，能够被大家所接受并广为传播。

Axure简介

Axure是一款原型工具，使用这个工具比较多的人群是产品经理。其实，Axure并不局限于制作产品原型，它也能替代PPT、Visio等软件制作演示文稿或者流程图。但是，很多人一提起Axure就将它与产品经理联系到了一起，从而对原型甚至对Axure这款软件产生一些畏难的心理。

在这里我说一下我的观点。

根据我个人理解，用Axure制作原型的时候，Axure应该是面向需求的工具，这样才能有效地控制产品制作的投入与产出。就像有很多同学问我：“小楼老师，我的原型保真程度做到什么样才合适？”这个问题很难回答！因为，这是由原型应用场景决定的。就拿做互联网产品来说，面向开发人员、面向视觉设计、面向市场、面向风投、面向老板、面向客户等，这些群体对原型的理解能力都不在一个层面，因此他们对原型的表现力或者说保真度的需求都不一样。

所以，Axure作为一个沟通的工具，必须要灵活地使用。从学习的角度来说，我认为要全面、熟练地掌握这个工具。只有全面、扎实地掌握基础知识，才能更高效地完成原型设计工作。要用同等的时间做出更好的效果，或者用最短的时间完成同等的效果，都需要有扎实的能力和优秀的逻辑。

而这些技能，都是需要时间，需要一点一滴地去积累，没有捷径！

本书简介

本书共分为9章。根据内容可分为3个部分。

第1部分（第1~2章）：主要讲解软件安装及基础功能的使用。熟练掌握这一部分内容之后，可以

独立完成一般带有少量交互的线框图；

第2部分（第3~8章）：对软件更深层次的应用进行了详细的讲解。特别是对一些功能复杂的元件，让新手望而生畏的变量、函数进行了详细的讲解与案例的展示。这一部分也讲述了如何让原型对不同设备（PC机和移动设备）支持，以及如何制作自定义元件库。熟练掌握这一部分内容之后，可以独立完成带有大量交互的高保真原型的制作。

第3部分（第9章）：这一部分是一个比较完整的案例的制作过程。从知识角度来说，从原型的结构、页面的搭建、交互的实现等方面，详细讲述了一个项目原型从无到有的过程。能够让零基础的读者通过学习拥有实现完整原型开发的能力。同时，整个案例对前面两部分的基础内容进行了贯穿应用，包含大量启发思路、提升逻辑思维能力的案例解析。从应用角度来说，这部分内容中同时包含了PC端以及移动端的不同内容，通过PC端与移动端页面设计、布局呈现的对比，能够让读者更直观地理解不同设备原型制作的不同之处。

本书非常适合零基础的读者，能够让从未接触过原型开发工作的读者迅速、全面地掌握原型开发技能。同时，书中融入的案例以及作者自身总结的一些经验技巧，也能够让具备一定基础的读者从本书中获得基础知识的补充、逻辑思维的提升，以及思路的启发。

最后，作者提醒大家，本书前后内容关联性较强，务必按顺序学习本书的内容，完成相关的任务，切勿跳跃学习，以免产生学习障碍。

小楼一夜听春雨

2015.10.19

一个能改变世界的好工具

初识Axure

2009年底，我怀着做互联网的梦想，离开沈阳来到了上海。因为没有名校背景，并且当时还未大学毕业，所以找工作时处处碰壁。在上海面试了二十家左右的企业，后来因为被面试官看中有多年做站长的背景，我有幸加入一家叫“51.com”（以下简称51）的社区，担任产品运营员一职，负责51同城的运营。

那时，我完全不知道产品经理是什么，后来，我因为一些项目接触到了产品策划人员（每个公司对产品经理这个职位的叫法不一样，在51，产品策划人员就是产品经理），并与之一起做项目，才对这个职位有了一些了解。令我印象特别深刻的是，在51，产品策划输出原型的工具五花八门，有的人用Visio，有的人用Photoshop，有的人用Word，有的人用PPT，还有的人用一些在线原型工具。总之，很少有人用Axure这个工具。记得当时从盛大公司来了一位产品经理，Axure用得特别厉害，大家看到他做的原型图都惊呆了，公司还专门让他给大家做Axure的培训。从那以后我终于知道，原来还有一种工具可以这么神奇，可以把产品创意表现得如此淋漓尽致。

转型路上，工具的痛

2010年年底，我成为了一名产品经理，一名不会用Axure画原型图的产品经理。我至今仍然特别感谢当时给我转型机会的面试官，也是我的直接上司。因为是在创业公司，项目特别多，时间特别赶，我连续四个月“朝九晚十”，一周工作六天，经常通宵加班。入职后没有任何缓冲和学习的过程，就直接开始做项目。我每天逼着自己学习用Axure做产品原型，当时特别渴望有一套完整的Axure教程，然而市场上却一直没有，所以每天只能去查百度知道，泡论坛，看一些碎片化的文字教程。还好我对工具型软件比较感兴趣，因此学得还算比较快。

产品设计师，你需要一个好的工具

做原型图的工具很多，经常看到来自不同公司的同事使用着不同的工具，坦诚地讲，谁也不愿意放弃一个已经很熟悉的工具，而去学习新的工具。但是，很多时候你是没有选择的。比如你从一个光杆司令产品经理变成一个产品团队、交互团队，此时就需要考虑团队协作，考虑输出作品的兼容性。对于这些需求，Axure无疑是做得最好的。用Axure可以做APP、网站、WAP、H5的产品原型设计，可以做出高保真的交互效果，甚至还有很多产品经理拿他当Photoshop使用。在团队协作方面，Axure的表现也特别出色。总之，Axure太强大了。（前些日子“人人都是产品经理”网站的某位专栏作家还用Axure画出了动态的挖掘机。）

成长的路上，感谢有你

认识小楼老师也有两三年了，“人人都是产品经理”社区曾经也发布过很多小楼老师的Axure教程，受到众多产品经理、交互设计师的热情追读。从最早的一个个人博客发展到今天的Axure学习社区，从一些碎片化的Axure原型案例教程到今天的系统性工具讲解与实战教程的结合，小楼老师一步步系统地整理出一套科学的学习进度、内容和学习方法，为小楼老师点赞！感谢小楼老师的Axure教程为广大新手交互设计师、产品经理带来福音。没有枯燥、机械的步骤罗列，只有“一对一”教学式的耐心解答。其中包含了从零起步时的安装指导，到Axure中最难掌握的中继器和函数使用方法的讲授，再到一个完整产品的Web和APP版原型的制作全流程指导。读者可以跟着实际案例学习，举一反三，把它们快速应用到自己的项目中。同时，这本书又能作为案头常备的工具书，日常工作时时查找对应的功能和使用方法。

如果我还是当年那个为产品原型苦恼的产品经理，那么，这就是我需要的那本书。

人人都是产品经理/秒聘网/起点学院创始人&CEO 曹成明

2015年11月30日

简介

小楼是一家IT外包公司的产品经理。因为工作需要，向老板提出用人申请，结果，老板把一个还在上大学的实习生苏苏安排进了公司，让小楼通过培养新人来增加人力。无奈之下，小楼只好争取在最短的时间内教会新人使用原型工具Axure RP 7.0，让她能够独立工作，分担部门工作的压力。就这样，小楼开始了一个Axure新手养成计划。

为保证最好学习效果，请按正文顺序学习。

抱诚守**真**，鼎力之作；

倾尽所**有**，授业解惑；

细枝末**节**，面面俱到；

理论实**操**，完美结合。

购买本图书阅读更多内容，请访问<http://www.iature.com/3384.html>。

【点击此处，马上购买】

第1篇 基础与入门

▶▶ 第1章 认识Axure RP 7.0 12

- 1.1 Axure RP 7.0介绍 13
- 1.2 软件以及相关下载 13
- 1.3 软件的安装与汉化 13
 - 1.3.1 安装Axure RP 7.0..... 14
 - 1.3.2 汉化Axure RP 7.0..... 16
- 1.4 安装谷歌浏览器插件 17

▶▶ 第2章 主要功能详解 19

- 2.1 快捷功能区 20
- 2.2 站点地图 20
 - 2.2.1 站点地图的功能 20
 - 2.2.2 分析网站结构(思维导图)··· 21
 - 2.2.3 创建网站结构 25
- 2.3 元件库 26
 - 2.3.1 元件的使用 26
 - 2.3.2 元件的类别 27
- 2.4 元件属性与样式 29
 - 2.4.1 元件的样式 29
 - 2.4.2 元件的属性 31
 - 2.4.3 属性与样式的使用(案例)··· 32
- 2.5 生成与预览 41
 - 2.5.1 生成与预览的设置 41
 - 2.5.2 生成与预览的区别 41
- 2.6 元件交互与说明 42
 - 2.6.1 元件的名称 42
 - 2.6.2 元件的交互 42

- 2.6.3 条件判断 45
- 2.6.4 元件的说明 75
- 2.7 元件管理 77
- 2.8 页面设置 78
 - 2.8.1 页面样式 78
 - 2.8.2 页面说明 80
 - 2.8.3 页面交互 80
- 2.9 母版管理 81
 - 2.9.1 母版的创建/编辑/删除 81
 - 2.9.2 母版在页面中的使用 82
 - 2.9.3 母版的拖放行为 82
 - 2.9.4 母版的使用情况 83
 - 2.9.5 自定义触发事件 83

第2篇 深入与精通

▶▶ 第3章 变量与表达式 88

- 3.1 变量 89
 - 3.1.1 全局变量 89
 - 3.1.2 局部变量 93
- 3.2 表达式 94
 - 3.2.1 运算符的类型 95
 - 3.2.2 表达式的格式 96

▶▶ 第4章 动态面板 98

- 4.1 动态面板的组成 99
 - 4.1.1 状态管理 99
 - 4.1.2 状态样式 100
- 4.2 动态面板的特性 101

4.2.1 容器	101	6.3.3 分页与翻页	170
4.2.2 固定	104	6.4 中继器/数据集函数	173
4.2.3 状态	106	6.4.1 中继器函数	173
4.2.4 拖动	107	6.4.2 数据集函数	173
4.2.5 缩放	109		
4.2.6 循环	113	►► 第7章 自适应视图与 AxureShare	178
4.3 动态面板的属性	115	7.1 自适应视图	179
►► 第5章 函数	118	7.2 AxureShare	181
5.1 函数介绍	119	7.2.1 APP下载	181
5.2 函数详解	120	7.2.2 注册和找回密码	181
5.2.1 指针函数 (Cursor)	120	7.2.3 工作区的使用	183
5.2.2 窗口函数 (Window)	121	7.2.4 发布	184
5.2.3 页面函数 (Page)	123	7.2.5 移动设备的通用设置	186
5.2.4 数字函数 (Number)	123	7.2.6 iOS 专属设置	188
5.2.5 数学函数 (Math)	124	7.2.7 从主屏幕图标打开原型	189
5.2.6 日期函数 (Date)	126	7.2.8 使用 AxureShare APP 查看 原型	191
5.2.7 字符串函数 (String)	131		
5.2.8 元件函数 (Widget)	136	►► 第8章 自定义元件库	194
►► 第6章 中继器	146	8.1 创建元件库	195
6.1 中继器的组成	147	8.2 设置元件库	196
6.1.1 数据集	149	8.3 使用元件库	197
6.1.2 项目交互	150		
6.1.3 样式设置	153	第3篇 项目与实战	
6.1.4 属性	155	►► 第9章 在线订餐网站及APP 原型	200
6.2 数据集的操作	157	9.1 设置项目结构	201
6.2.1 添加数据	158	9.2 设置视图结构	202
6.2.2 删除数据	159	9.3 设置页面结构	204
6.2.3 更新数据	164	9.4 创建公共模块	204
6.3 项目列表操作	165	9.4.1 Web端导航栏	205
6.3.1 排序	165	9.4.2 移动端导航栏	212
6.3.2 筛选	168		

9.4.3 底部公司信息	214	9.5.7 页面7: 订单页 (Orders) ...	267
9.5 组织页面内容	214	9.5.8 页面8: 结算页 (Payment) ...	294
9.5.1 页面1: 引导页 (Index)	215	9.5.9 页面9: 喜欢页 (Favorite) ...	298
9.5.2 页面2: 启动页 (Loading) ...	216	9.5.10 页面10: 个人中心页	
9.5.3 页面3: 登录页 (Login)	218	(User_info)	302
9.5.4 页面4: 热线电话页		9.5.11 发布与更多尺寸	318
(Hotline)	222		
9.5.5 页面5: 首页 (Home)	227		
9.5.6 页面6: 商品详情页			
(Goods_info)	257		

▶▶ Axure RP 7.0 快捷键 321


第 1 篇

基础与入门

第1章 认识Axure RP 7.0
第2章 主要功能详解

第1章 认识Axure RP 7.0



- 
1. Axure RP 7.0介绍
 2. 软件以及相关下载
 3. 软件的安装与汉化
 4. 安装谷歌浏览器插件

公司的业务发展良好，工作压力也越来越大，我找老板要人，帮产品设计部缓解压力，谁知道居然来了一个什么都不懂的女大学生。但是，再怎么抱怨也无济于事，不如尽快培养好新人。

“苏苏，你知道Axure这个软件吗？”我问新人苏苏。

“不知道。”得到了一个干脆的回答。

“好吧！打开你的电脑，我们从零开始……”

▶▶1.1 Axure RP 7.0介绍

新人不知道这个软件是很正常的事，我们先来看看Axure RP 7.0（简称Axure）到底是款什么软件？

Axure（发音：Ack-sure），代表美国Axure Software Solution公司；

RP则是Rapid Prototyping（快速原型）的缩写。

7.0是Axure RP的当前版本，测试版发布于2013年7月份，正式版发布于2013年12月份。

Axure RP是美国Axure Software Solution公司旗舰产品，是一个专业的快速原型设计工具，让负责定义需求和规格、设计功能和界面的用户能够快速创建应用软件、Web网站以及移动APP的线框图、流程图、原型和规格说明文档。同时，这款工具支持多人协作设计和版本控制管理。

Axure RP的使用者主要包括商业分析师、信息架构师、可用性专家、产品经理、IT咨询师、用户体验设计师、交互设计师、界面设计师等，另外，架构师、程序开发工程师也在使用Axure。

▶▶1.2 软件以及相关下载

新人不知道Axure怎么下载是很正常的事，我们再来看看Axure怎么下载？

Axure官方网站下载地址：

<http://cdn1.axure.com/AxureRP-Pro-Setup.exe>

备注

因为Axure的官网是国外的服务器，国内用户可能会出现访问缓慢的情况，所以下载软件的时候，可以输入上述地址到浏览器地址栏回车进行下载，或者用下载工具进行下载。

作者的网站也为读者提供了下载地址（包含汉化文件下载）：<http://www.iature.com/511.html>。

另外，用谷歌浏览器浏览Axure制作原型生成的HTML文件时，需要安装插件才能显示。这个插件可以在谷歌应用商店中下载。

作者的网站也为读者提供了的插件下载地址：<http://www.iature.com/1534.html>。

▶▶1.3 软件的安装与汉化

新人不知道Axure怎么安装、汉化是很正常的事，我们接下来看看Axure如何安装和汉化？

1.3.1 安装Axure RP 7.0

步骤 1 双击打开下载好的安装程序，弹出下方的带有乱码的界面（图 1-1），等待进度条加载完毕。

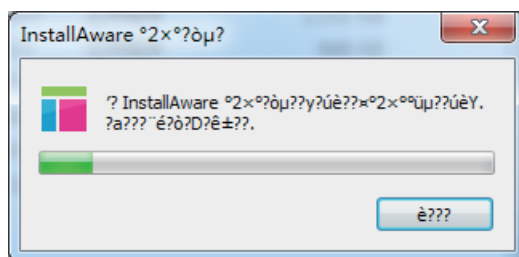


图 1-1

步骤 2 在弹出的安装界面中，单击【Next】按钮，进入下一步。（图 1-2）

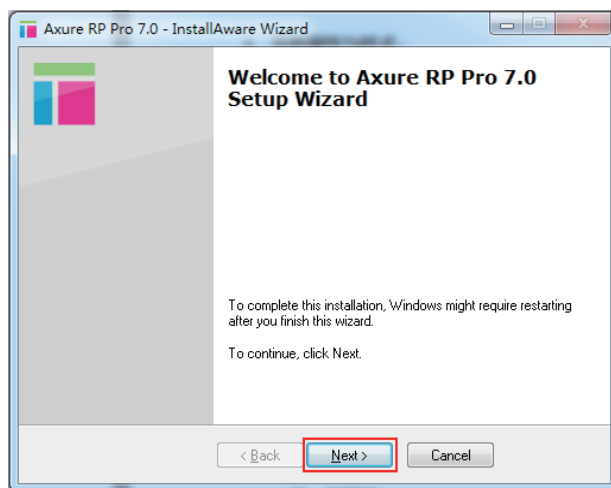


图 1-2

步骤 3 在软件安装协议界面，勾选“I Agree”的复选框，单击【Next】继续。（图 1-3）



图 1-3

步骤 4 通过直接输入软件安装地址或者单击【Browse...】改变安装路径，我们也可以不做改变，直接安装在默认目录即可。单击【Next】继续。（图 1-4）

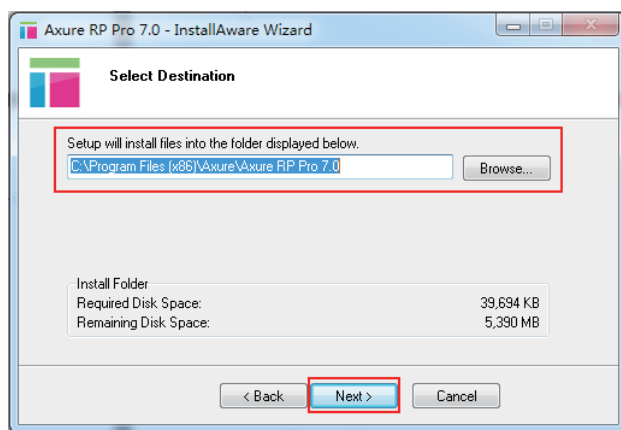


图 1-4

步骤 5 选择为哪些用户创建快捷方式，一般按照默认选项即可。单击【Next】继续。（图 1-5）

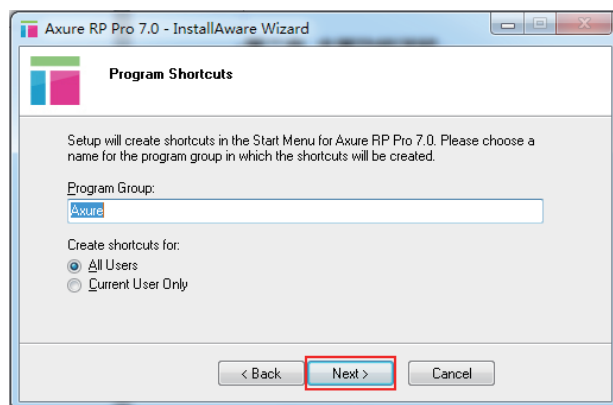


图 1-5

步骤 6 单击【Next】进入最后的安装。（图 1-6）

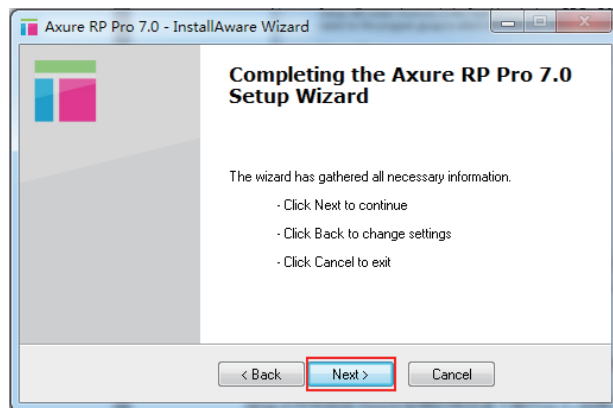


图 1-6

步骤 7 等待安装完毕，如果实在无聊，可以单击【Cancel】再来一次。（图 1-7）

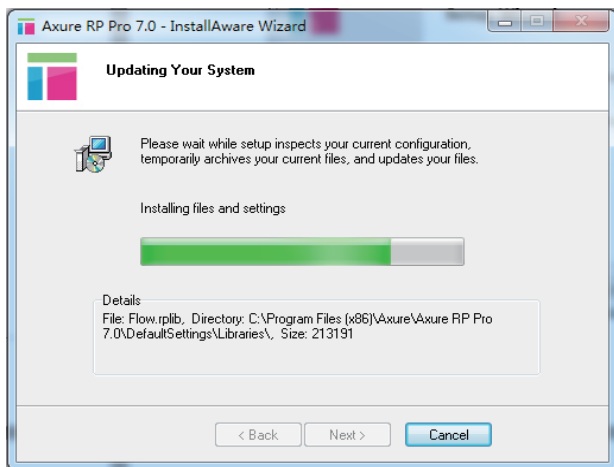


图 1-7

最后：先不要着急启动软件，取消【Run Axure RP Pro】的勾选，单击【Finish】完成安装。接下来，进入汉化步骤。（图 1-8）

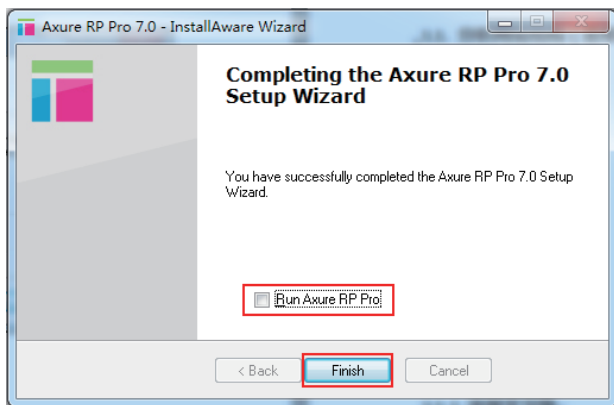


图 1-8

1.3.2 汉化Axure RP 7.0

一般我们下载到的汉化文件都是一个压缩包，解压缩后会有一个名称为“lang”的文件夹，文件夹里面是一个名称为“default”的文件。

汉化步骤非常简单，只需要将解压缩后的“lang”文件夹放进 Axure 安装的根目录下即可。（图 1-9）

注意事项

1. 如果解压缩后只有一个“default”文件，则需要在 Axure 软件的安装根目录下新建一个“lang”文件夹，然后把“default”文件粘贴到里面去。
2. 汉化安装路径（以 Windows 7 为例）
32 位操作系统：C:\Program Files\Axure\Axure RP Pro 7.0
64 位操作系统：C:\Program Files (x86)\Axure\Axure RP Pro 7.0

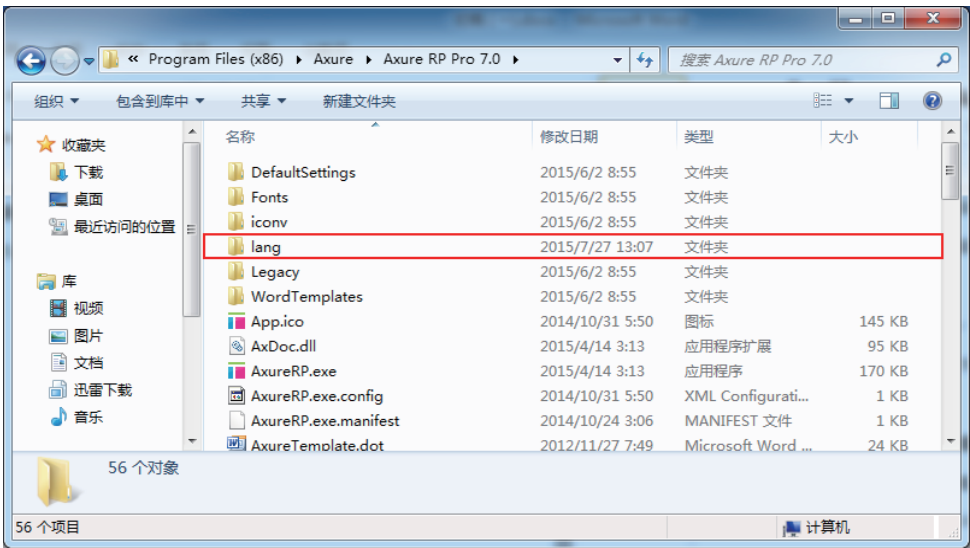


图 1-9

新人不知道的东西会很多，所以需要有耐心地一步一步去学习，不要着急，循序渐进是效率最高的学习方法。

1.4 安装谷歌浏览器插件

安装步骤：

- 步骤 1 下载 “AxureRP_for_chorme_0_6_2.crx” 文件，下载地址见前文：软件以及相关下载小节。
- 步骤 2 打开谷歌浏览器，单击功能列表按钮，在工具菜单中找到扩展程序，单击打开。（图 1-10）

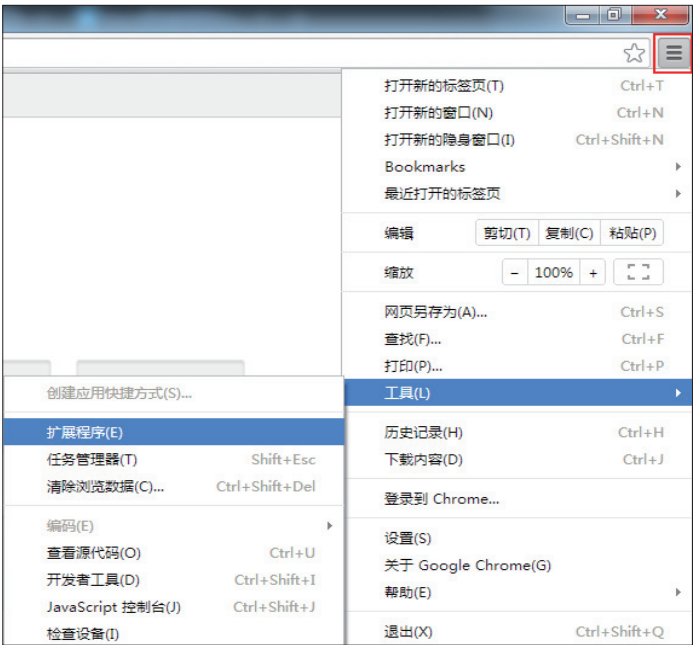


图 1-10

步骤 3 将下载到本地的插件文件拖入到谷歌浏览器中，这时会提示“拖放以安装”。（图 1-11）

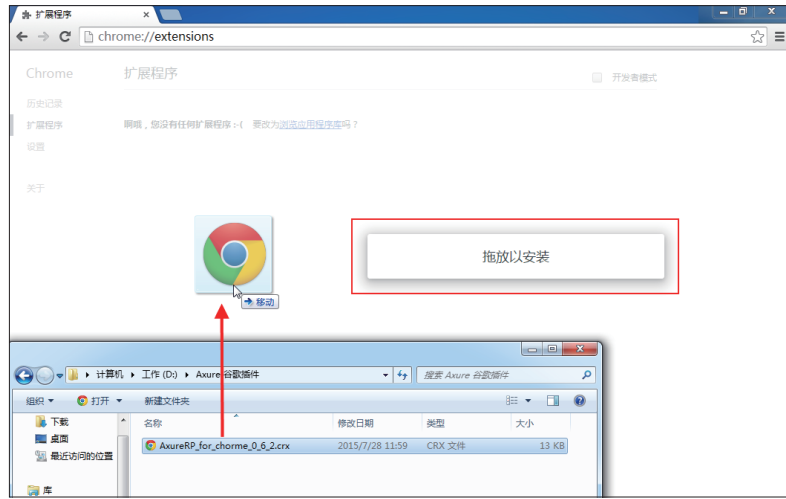


图 1-11

步骤 4 在弹出的对话框中单击【添加】。（图 1-12）



图 1-12

步骤 5 在添加好的扩展程序中勾选“允许访问文件网址”，并且保证“已启用”为已勾选的状态。（图 1-13）




图 1-13

这时，谷歌浏览器就可以正常浏览 Axure 生成的 HTML 文件了。

第2章 主要功能详解



- 
1. 快捷功能区
 2. 站点地图
 3. 元件库
 4. 元件属性与样式
 5. 生成与预览
 6. 元件交互与说明
 7. 元件管理
 8. 页面设置
 9. 母版管理

苏苏，下面我们进入这个软件主要功能的讲解。你一定要仔细了解每一个功能的用途，这对之后原型制作会有很大的帮助。

哦，好的。

2.1 快捷功能区

终于解决了汉化问题，打开了软件界面。首先看到的第一部分就是顶部的快捷功能区。

Axure 这款软件的风格与 Office 系列软件的风格十分相像，软件的顶部把一些常用功能的快捷按钮全部集成在这里。（图 2-1）

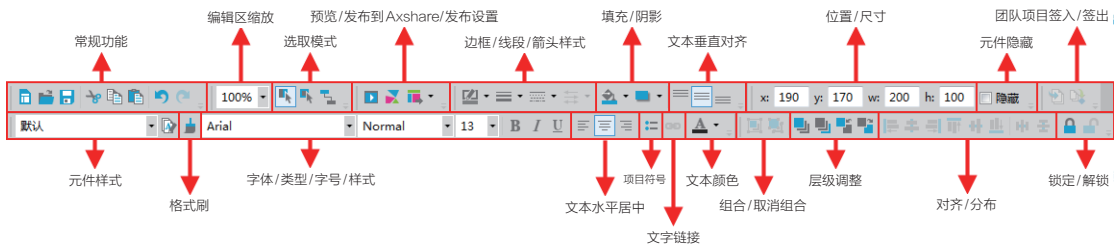


图 2-1

对这一部分需要先有一个大概的了解，不用着急地全部记下来。因为在后面的操作中我们会经常用到，会越来越熟悉这些按钮和功能。

2.2 站点地图

首先，我们来看站点地图。

好！

你打开百度地图做什么？

站点地图与我们常用的地图无关，Axure 中的站点地图包含整个原型中所有的页面。在站点地图中我们可以方便地搭建原型结构与页面层级关系。

2.2.1 站点地图的功能

站点地图位于软件的左上位置，它的作用是管理整个原型中的所有页面，也能够进行页面的添加、删除、命名等操作。当我们双击一个页面名称的时候，这个页面会在主编辑区打开，新建页面里面是空白的，页面的内容是我们自己来编辑的。当我们打开页面后，主编辑区上方也会出现相应的标签，供我们单击切换页面。（图 2-2）

每一个网站，甚至每一个功能模块都有自己的结构，就像在大学里写毕业论文一样，首先要有一个提纲，这样可以帮助我们树立全局的观念，从整体出发，检验每一个部分所占的地位，相互之间是不是有逻辑关系等。提纲能体现出作者的总体思路，优点在于易于掌握论文的全局结构，使论文层次清楚一目了然，只有论文层次一目了然，才能更好地写作。

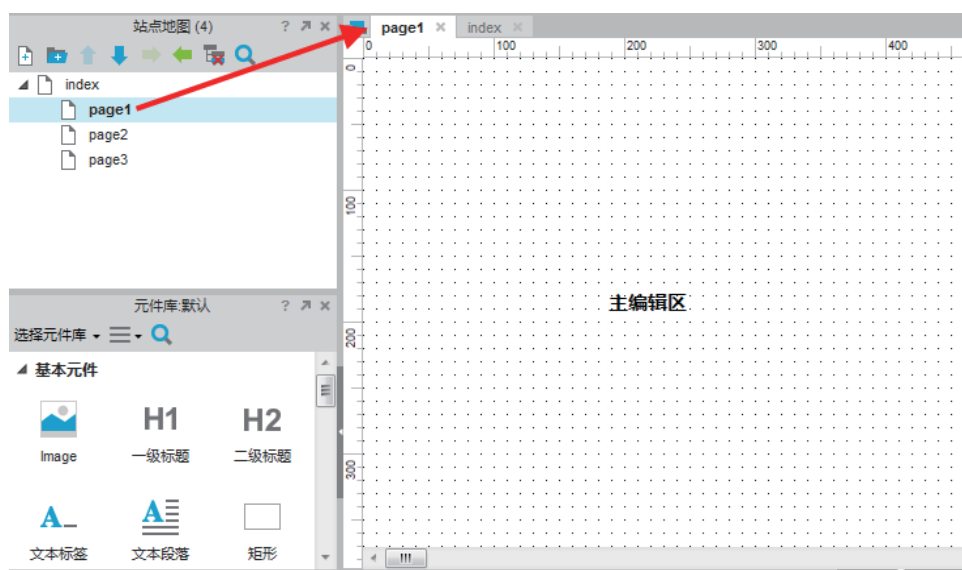


图2-2

我们在进行一个项目的原型开发之前，也要先有一个提纲，就是站点地图。我们在站点地图里搭建整个项目的结构，梳理页面间的层级关系，在完成这个大的框架之后，再做具体的内容填充。这样，就能做出结构合理、思路清晰的原型。

2.2.2 分析网站结构（思维导图）

直接在站点地图中搭建原型结构并不是很方便。而且页面间的逻辑关系及页面中所包含的功能模块、元素也没有办法体现出来。

在这里我们需要使用另外一款软件，在我们搭建原型结构之前，先以图表的形式把我们大脑里面所想到的内容直观地体现出来。

思维导图软件就是我们当前所需要的。

Xmind是一款商业思维导图软件，但是它的基本功能是免费使用的。这里我们就使用Xmind 6中文版（以下简称Xmind）帮助我们整理网站的结构图。

备注

这款软件的安装只需要按照默认安装就可以了，在这里不加赘述。

打开Xmind后，在模板中双击【中心主题】打开主编辑区（图2-3）。

打开后的主编辑区现在只有一个{中心主题}，双击可以更改它的名称，一般就改成项目名称或者网站名称。

在Xmind中有以下快捷键便于我们操作：

- ① 在中心主题上按<Enter>键可以添加子主题；
- ② 在任意子主题上按<Enter>键可以在下方添加同级主题，按<Shift>+<Enter>键可以在上方添加同级主题；
- ③ 在任意主题上按<Insert>键可以添加子主题，按<Delete>键删除当前主题及其子主题。

④ 在任意主题上按 <Ctrl+L> 键可以添加该主题与其他主题的关联。



图2-3

拖动主题，可以调整主题的位置和层级关系，另外，我们可以在任何一个主题上单击鼠标右键，设置当前菜单以下的图表结构，一般我们会使用逻辑图。（图2-4）

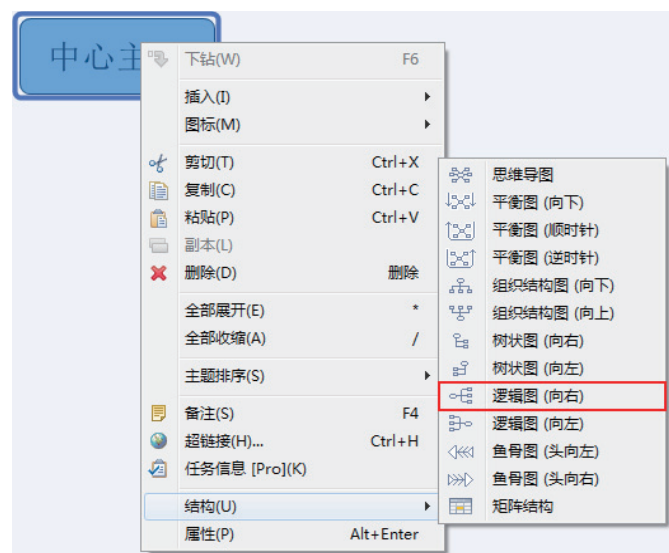


图2-4

苏 怎么才能做出一个网站结构图呢？

晓 在没有需求的情况下，我们不太容易构思一个网站，咱们就以“知乎网”为例，来研究一下这个网站的结构，并制作一个网站结构图。

首先，我们看到的是知乎的首页，里面包含了上方的登录注册的模块、下方的动态信息、底部的版权信息、专题链接还有移动客户端下载链接。（图2-5）

我们就可以把首页的内容归纳出来，形成一个图表。（图2-6）



图2-5

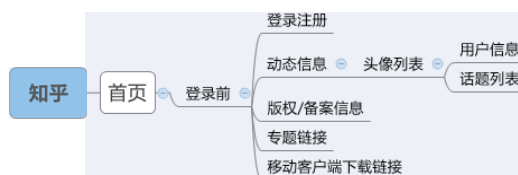


图2-6



图2-7

然后，在首页单击一条用户话题，进入话题的内容页面。这个页面也有它的结构组成。头部是带搜索功能的导航栏，左侧是主要内容区，右侧是侧边栏，底部是版权信息和专题链接。（图2-7）

备注

对于新手，关于Web页面布局也需要了解一下，网上有很多资料，例如常见的16种Web页面布局图（图2-8）。”

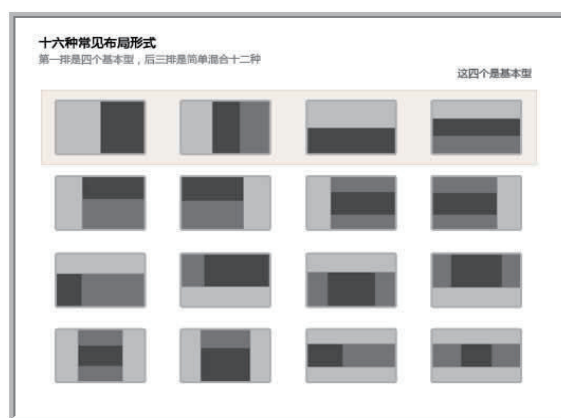


图2-8

页面的头部和底部并不是话题页面专有的，所以我习惯把他们单独放到公共模块。而页面中间部分的内容区和侧边栏是话题内容页独有的，我们把它的内容总结出来。（图2-9）

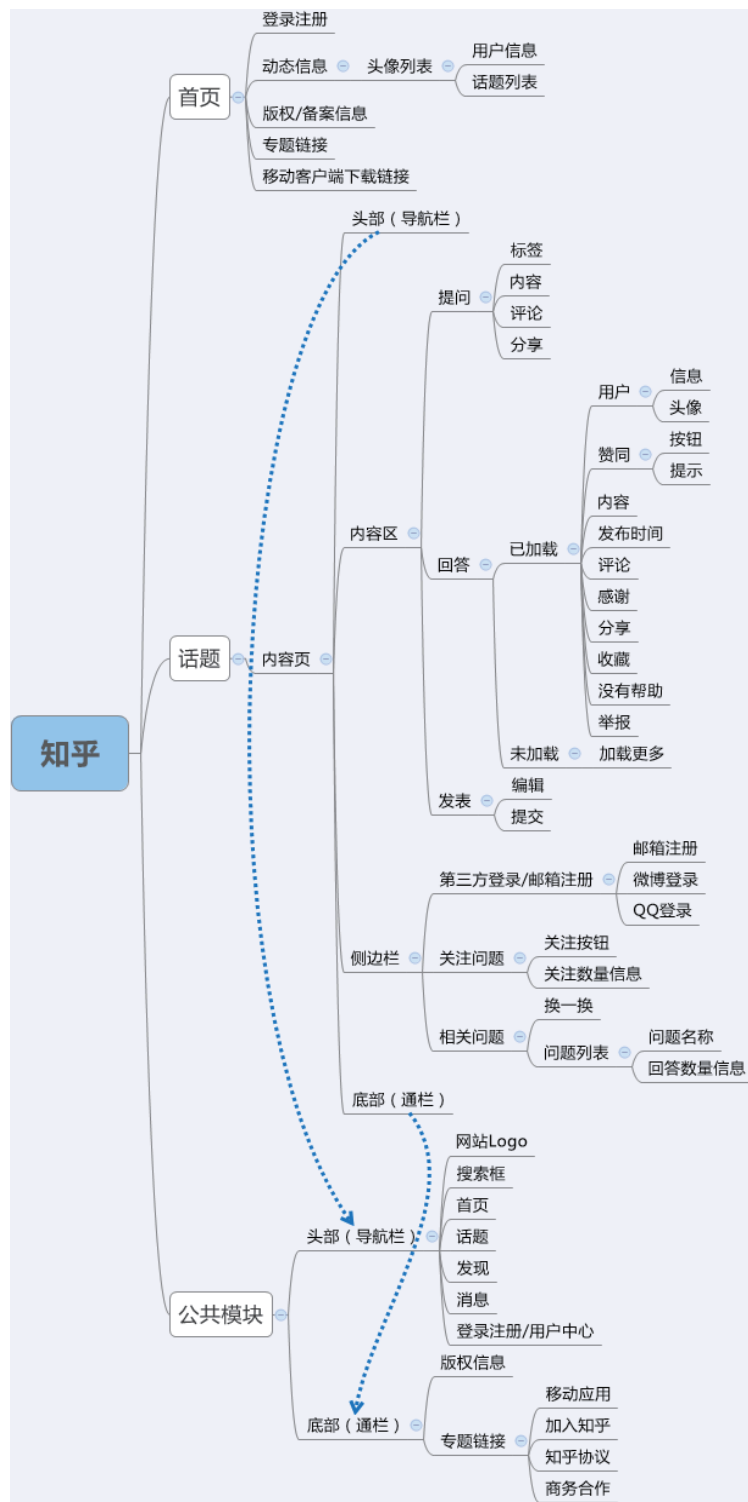


图2-9

这样就把网站结构和页面内容全部归纳了出来。除了这些之外，还有其他的页面我们也能够通过这种方式，全部整理出来，形成一个完整的网站结构图。

不过，有时候，一个页面还会有它的下级页面或者叫子页面，例如话题的首页，右侧有一个话题

广场的按钮，这个按钮单击后会跳转到话题广场页（图 2-10）。



图 2-10

相对于话题首页来说，话题广场页就是它的子页面，所以，在网站结构图中话题广场页是包含在话题首页的下级主题中（图 2-11）。

我们单击每个页面后面的收起按钮，把页面内容部分先隐藏掉，剩下的就是网站的页面结构了。”

2.2.3 创建网站结构



楼哥，这步做完是不是就能够在 Axure 的站点地图里面创建页面了？



是的，如果是通过需求分析来做一个项目，就是要把自己对需求的理解借助思维导图软件，一点一点地把想到的内容整理成这样的结构图，搭建好页面的层级关系及功能之间的逻辑关系，然后把它作为参考去完成原型的开发。当然，这个图也不是尽善尽美的，它的作用是帮助我们最大程度上避免原型开发过程中出现的结构和逻辑错误，如果在之后发现这个图还有不够完善的地方还可以进行后续的修改。

备注

以上为作者在工作中总结出的经验方法，不是行业的规范要求，仅供读者参考，欢迎指正！

下面，我们根据刚才做好的网站结构图把站点地图搭建出来。其中一些权重和关联性较低的页面可以统一放到一个文件夹中进行管理。（图 2-12）

备注

在站点地图中可以通过右键添加同级菜单、子菜单及用来存放一组页面的文件夹；还可以通过拖曳页面来调整页面的层级关系等。

本节任务 ➡ 找一个简单的网站去进行分析，进行结构图的整理练习。

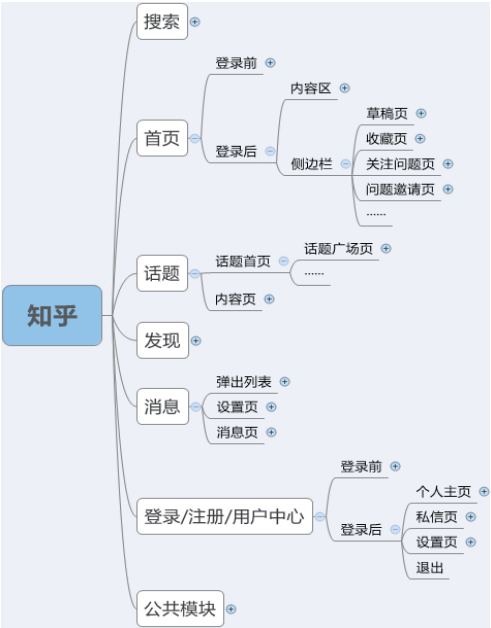


图2-11

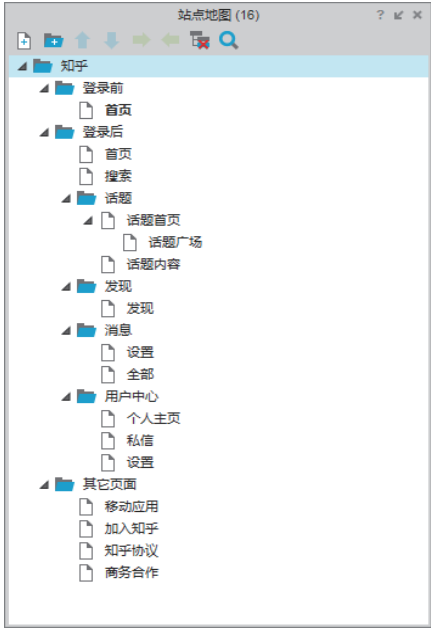


图2-12

2.3 元件库

- 结构搭建完了，就该往里面添加内容了。内容是通过元件来组成的。
- 为什么叫元件呀？
- 这个呀！它是借用实物的命名，我们通常会把机器或者仪器的组成部分叫做元件，例如：电子元件，线路元件等。

2.3.1 元件的使用

元件是软件界面左侧第2个功能模块，这个模块用于对所有元件进行管理。软件本身自带了两个元件库，一个是默认元件库，一个是流程图元件库。第一个“选择元件库”选项列表用来选择元件库，能够帮助我们方便地切换元件库或者显示全部的元件库。在我们进行线框图原型开发时，最常用的是默认元件库。（图2-13）

第二个选项列表，用来管理元件库，在这里可以方便地载入和卸载元件库，也能够创建和编辑自定义元件库。（图2-14）

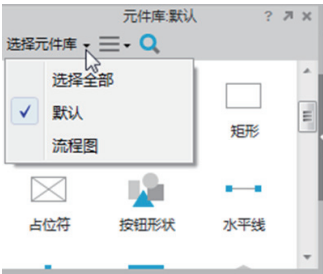


图2-13

备注

卸载自定义元件库时需要先在“选择元件库”选项列表中选择自定义元件库，然后在后面的选项列表中选择【卸载元件库】进行卸载。

我们在进行原型页面制作时，只需要把元件库里面的元件用鼠标左键点住，然后拖动到主编辑区松开，元件就会被摆放在指定的位置上。

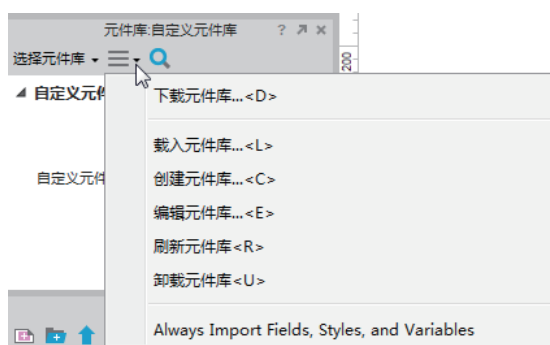


图 2-14

一个页面的内容就是通过一个一个的元件组成的，所以一般网站里面所显示的内容我们都能通过元件的组合搭配模拟出来。（图 2-15）

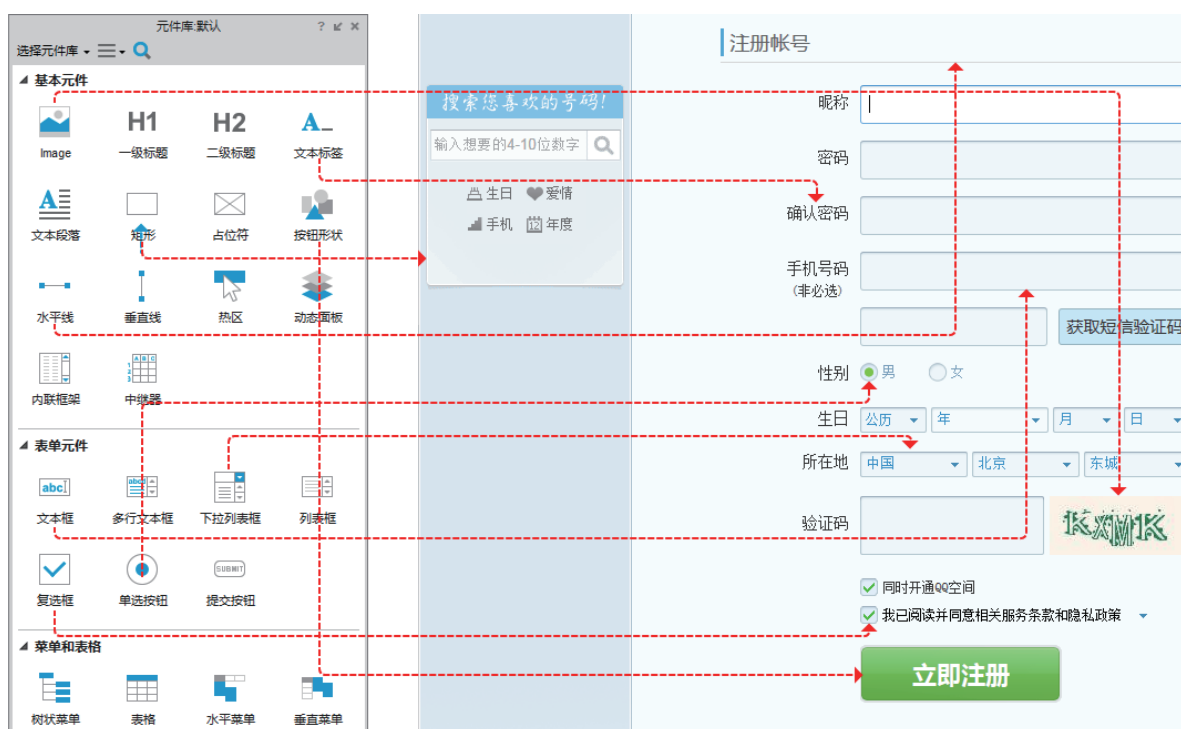


图 2-15

2.3.2 元件的类别

在默认元件库里，有 3 组元件，分别是基本元件、表单元件、菜单和表格元件。

2.3.2.1 基本元件

基本元件是组成各类原型的基本元素。（图 2-16）

前面 10 个是最常用的图片、文本标签、形状以及线段。

其中：

一级标题、二级标题还有文本段落都是文本标签不同样式的变种；

占位符和按钮形状则是在矩形基础上的改变。
线段有两种，分别是水平和垂直。



图 2-16

备注

如果想要斜线的话按住键盘的 <Ctrl> 键，把鼠标指针放在元件的边界点上就能够改变线段的角度（图 2-17），这个方法也适用于大部分元件改变角度，如果想精确改变角度可以在元件的样式中去调整，这个在后文还会讲到。

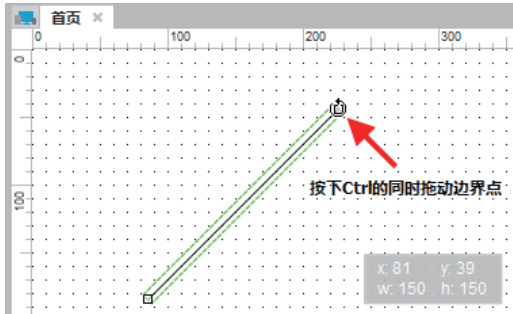


图 2-17

热区：也叫图片热区，网页开发中用于为图片的某个部分添加链接使用。在原型制作中除上述作用外，其透明的特性略有用处。

动态面板：一个容器元件，里面能够包含其他元件，它具备一些其他元件没有的特性，将在后文中着重讲到。

内联框架：它能够用来在页面中嵌入声音、视频、动画等媒体文件以及网页。

中继器：它是一个最特别的元件，通常用来实现重复的列表模块，如果灵活使用的话，还能实现很多让人意想不到的效果。

2.3.2.2 表单元件

表单元件在编程开发中用于向页面中输入数据形成表单并提交到服务器，所以它们是表单元件。（图 2-18）

文本框和多行文本框：用于输入文字。



图 2-18

下拉列表框和列表框：都是用来输入不同的选项。

复选框和单选按钮：分别用于多选和单选的输入。

提交按钮：这个按钮在编程开发中单击时能够完成表单的提交动作，但是在原型制作中不涉及与服务器的交互，所以显得多余，特别是它的样式不能调整又只有几个简单的触发事件，所以一般我们都会用自定义的形状按钮或图片按钮来代替它。

2.3.2.3 菜单与表格元件

菜单和表格元件不是特别常用。（图 2-19）



图 2-19


树状菜单：它是一种比较常用的菜单形式，经常用于一些网站后台的功能列表。

表格：在页面中呈现数据表时会用到，但是 Axure 的表格不支持合并单元格的操作，一般遇到这种需求，通常是用一个矩形进行遮盖，达到看似合并的效果。

水平菜单和垂直菜单：用于网站导航栏或者分类标签等等。

本节任务 ➡ 浏览一些网站，熟悉网站上的内容都是用哪些元件组成的，并且总结出每种元件在网页原型制作中的用处。

▶▶ 2.4 元件属性与样式

 楼哥，为什么这个按钮鼠标放上去会变色？为什么这个性别只能选中一个？为什么这组菜单点了任何一个之后都会变颜色，但只能有一个变颜色。点完再点另外一个，前一个又恢复原样……

 这些效果都是通过设置元件的样式与属性实现的。

2.4.1 元件的样式

在软件界面的右侧中部，是元件的属性与样式功能面板。当我们点中编辑区的元件时，这里就会

显示该元素的属性与样式。(图 2-20)

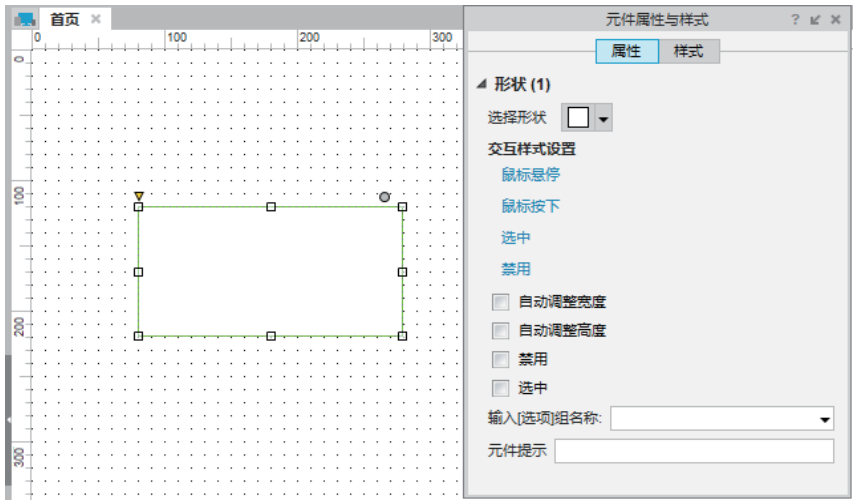


图 2-20

元件的样式包含的内容都一样，不过有个别元件的样式是禁止修改的。这里以矩形和热区元件举例说明（图 2-21）。

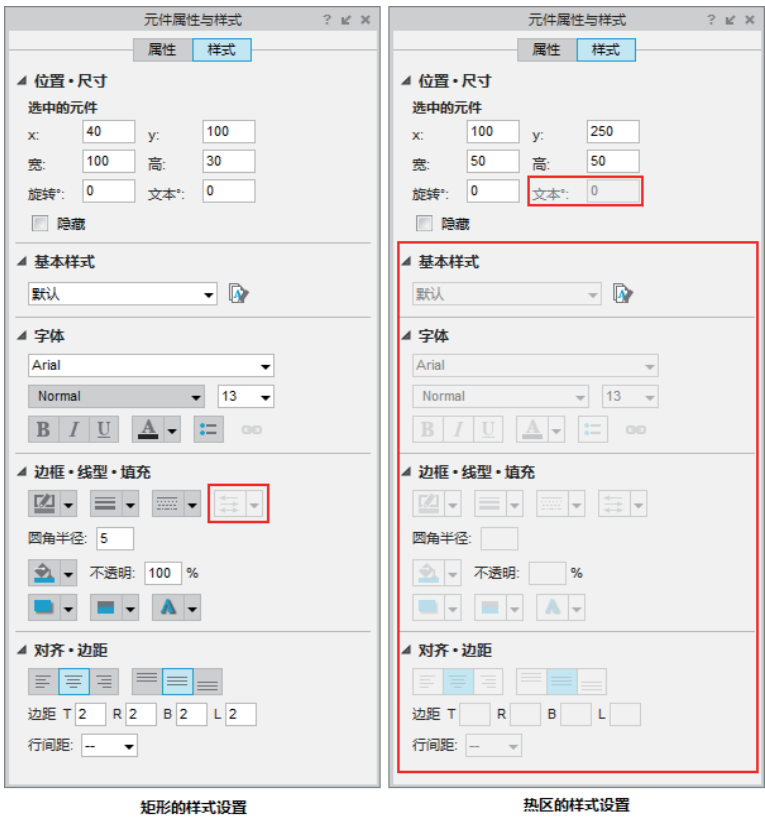


图 2-21

矩形元件的样式设置中只有箭头样式是灰色的状态，说明它是不可设置的；而热区元件的样式设置大部分的样式设置都是灰色不可编辑的。

2.4.2 元件的属性

元件的属性并不是每个元件都一样，他们有的相同，有的部分相同或者完全不同。

我们经常在网页上看到的鼠标进入按钮变色的效果，就可以通过设置属性来实现。只需要在属性面板的{交互样式设置}-【鼠标悬停】中去设置鼠标移入时的另一种样式或者另一个图片就可以了。（图2-22）

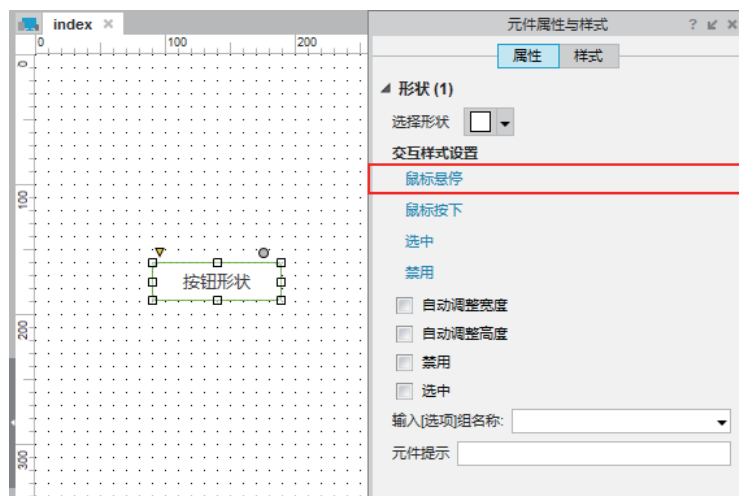


图2-22

这里以按钮形状为例，我们先拖入一个按钮形状，然后把【鼠标悬停】交互样式中【填充颜色】设置为要改变的颜色。当然，也能根据需求同时设置文字的颜色为有反差的颜色。例如把形状颜色设置为蓝色，文字设置为白色。因为默认勾选了【预览】，所以编辑区里面的按钮就变成了鼠标移入时的样子。（图2-23）

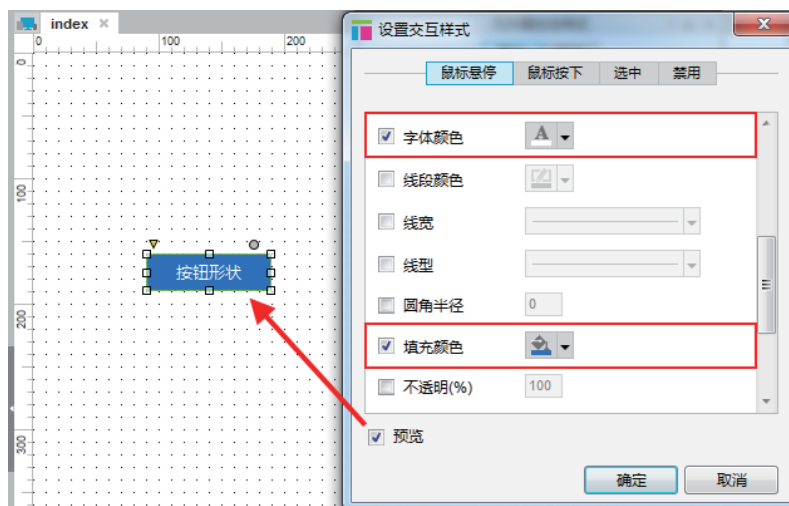



图2-23

单击【确定】以后，按钮颜色又变成默认的颜色。不过，这时在按钮上出现了一个黑白相间的小方块，鼠标移入小方块就会显示我们设置好的效果。

完成上面的编辑后，我们就可以通过按键盘上的<F5>键预览或者<F8>键生成来查看效果。当然也能点快捷功能栏里面的快捷按钮，或者从菜单里面选择。（图2-24）

发布<I> 团队<T> 帮助<H>	
	预览 F5
	预览选项...<O> Ctrl+F5
	发布到AxShare...<A> F6
	生成HTML文件...<H> F8
	在HTML文件中重新生成当前页面<R> Ctrl+F8
	生成Word文档...<G> F9
更多生成器和配置文件<M>	

图2-24

2.4.3 属性与样式的使用（案例）

前面只是简单地说明了一下属性与样式。下面我们再看几种属性与样式的设置。

2.4.3.1 形状的变换

在Axure RP 7.0版本中允许多种文本元件以及形状元件之间的切换。这个操作可以通过在属性中进行{选择形状}的设置，也可以通过单击元件右上方的圆点进行修改。（图2-25）



图2-25

2.4.3.2 圆角与旋转

Axure RP 7.0版本还支持快速将图片或形状元件改为圆角及调整圆角大小。

调整圆角的操作可以通过拖动元件左上方的小三角或在元件样式中改变圆角半径大小来实现。（图2-26）

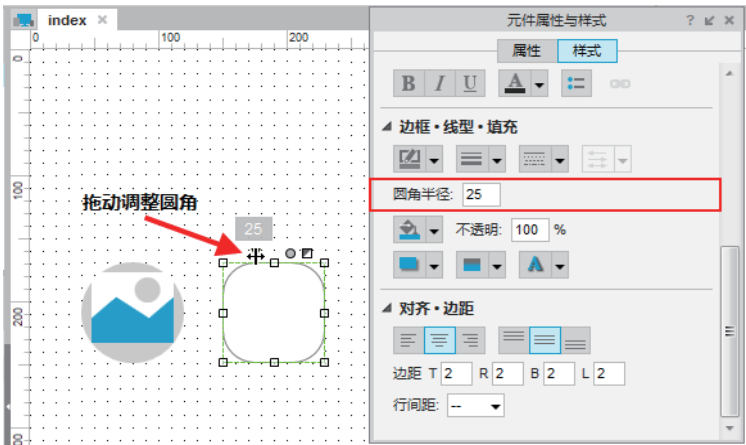


图2-26

前文中我们曾经通过按住<Ctrl>键同时拖动元件边界点实现元件旋转角度，这个操作也可以在样

式中通过设置角度数值来完成。同时，我们看到，元件文本的角度也可以改变，也就是说，在元件角度发生改变时，文本可以保持原来的角度或者另外一个角度。（图2-27）



图2-27

2.4.3.3 添加文字链

这是给元件文字添加链接的功能，注意给元件文字添加链接后，只有在单击文字时可以打开链接的内容，而单击元件其他部分时则无效。

可支持的元件有：形状、图片、文本、表格、菜单。

具体的操作为：在文本编辑的状态下，选中全部或部分文本，单击快捷功能区或者样式中的链接图标，进入链接编辑页面。（图2-28）

添加链接有几个选项可以选择。（图2-29）

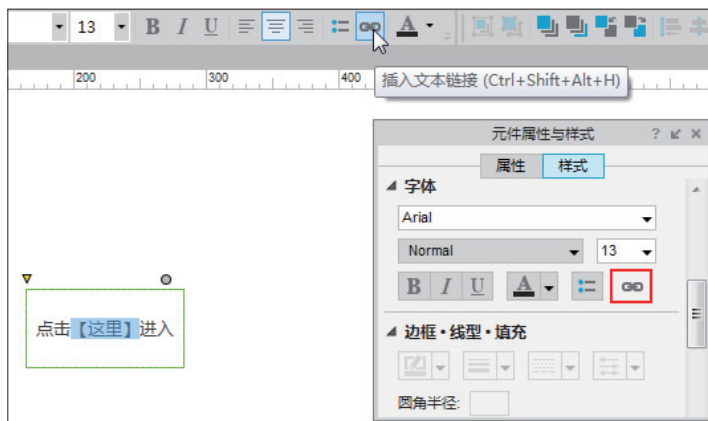


图2-28

第一项是能链接到我们当前项目的站点地图中的任意页面。

第二项是能够链接到外部的网址（URL）或者文件；例如我们输入网址“http://www.baidu.com”，单击文字时就会打开百度的首页。而且，这个选项也可以打开本地磁盘中的文件，例如AVI、Mp3、TXT、Swf等格式的文件。打开的时候只需要在输入网址时输入文件的本地路径就可以了，例如“E:\MP3\xxx.mp3”。不过这个打开文件的功能会受到浏览器的限制，目前，谷歌浏览器的支持是最完美的。

最后两个选项，分别是刷新当前页面和浏览器中的回退效果。

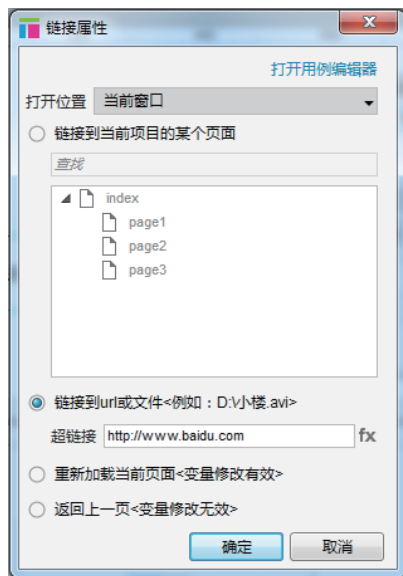


图2-29

备注

除图片需要右键选择【编辑文字】外，其他元件都可以通过双击直接进入编辑状态。

2.4.3.4 切割与裁剪

有时从网上找到一张图片素材，可能不满足我们的需求，例如：只想要里面的一部分，或者需要把整个图片变成几个部分。

我知道，可以用PS。

不用那么麻烦，Axure给我们提供了简单的图片编辑工具。你看这个有3种状态按钮的图片，我们可以通过切割或者裁剪来编辑它们。

1. 切割

切割就是可以把图片分割成多个部分，切割的功能在图片元件的属性中。（图2-30）

单击切割的图标后，鼠标移动到主编辑区就会变成一个小刀的形状，并带着十字虚线，这个十字线就像刀片一样，如果把一张图片切成4份只需把鼠标放到合适的位置单击鼠标的左键。如果要切割成两份的话，切割的时候只需要让一条切割线进入图片区域或者选择只有一条切割线即可。（图2-30）

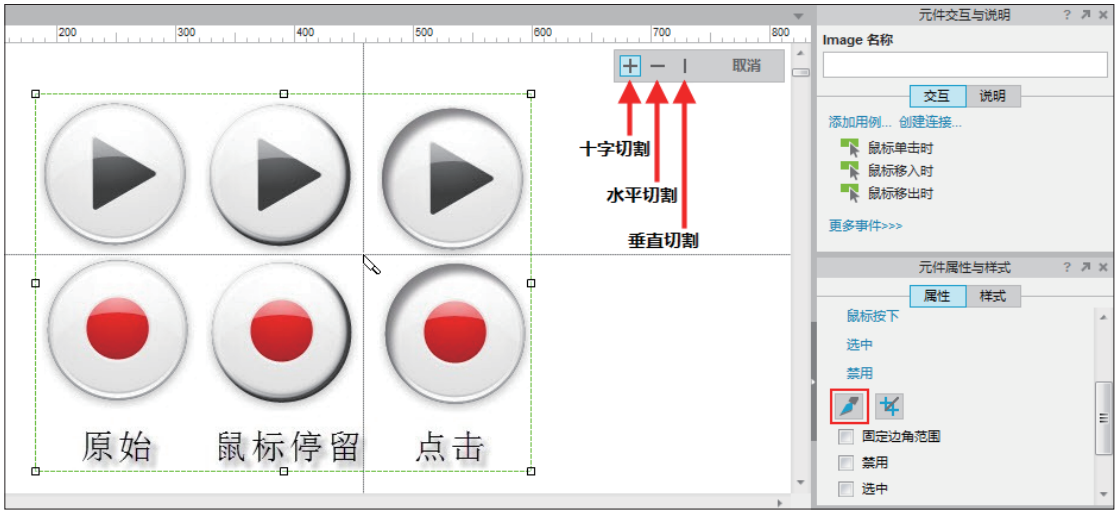


图2-30

Axure还支持批量切割，如果有多张图片需要切割，就把这些图片全部选中再单击切割按钮就可以一次切割多个图片。

不过，像图2-30里面的这种按钮，如果想要把每个按钮单独提取出来，在这里用切割不太方便，因为我们需要把图片里面的按钮按照相同的尺寸提取出来，所以这种情况下我们用裁剪功能更为合适。

2. 裁剪

裁剪能够把图片中的一个区域提取出来。

单击裁剪按钮后，会出现一个方框，拖动方框的边界点可以调整选取区域的大小。裁剪的操作有3种，可以把方框放在适当的区域后，根据需求单击功能按钮进行操作。（图2-31）

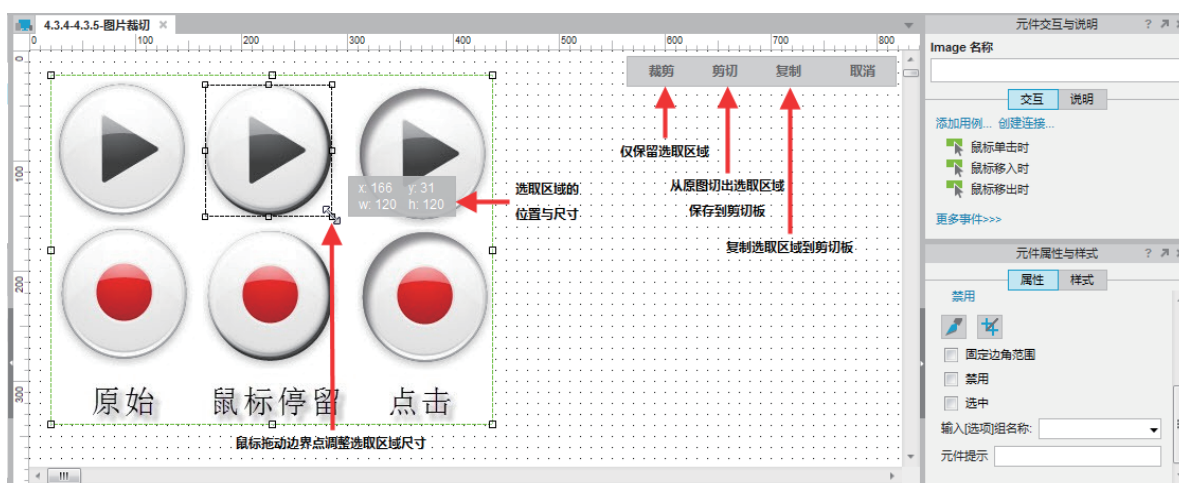


图 2-31

3. 保存原型中的图片到本地

当我们完成图片编辑后，图片不一定能够直接应用到原型中，很多时候我们是需要动态地加载这些图片素材，例如鼠标悬停或者鼠标按下时才需要在交互样式中从本地磁盘上导入这些图片素材。

下面我们就来讲解，如何把原型中的图片保存到本地磁盘。

首先，按键盘上的 <F8> 键生成 HTML 文件，在生成的页面把 HTML 文件的保存路径复制，或者选取一个比较容易找到的路径（图 2-32）。

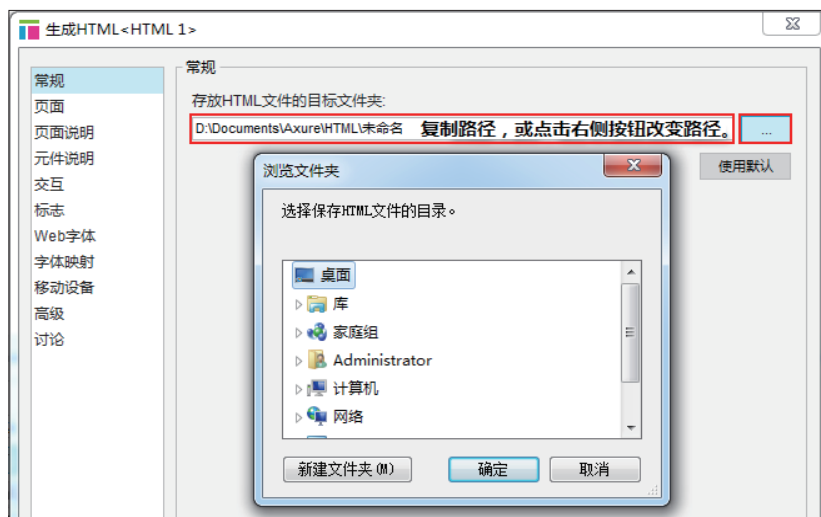


图 2-32

然后单击进行生成，生成之后打开我们在刚才指定的保存 HTML 文件的文件夹，或者把复制的本地路径粘贴到地址栏后按 <Enter> 键进入这个文件夹。（图 2-33）

进入之后打开里面的“images”文件夹，找到与刚才图片所在页面名称相同的文件夹，双击进入。这里面就有我们刚刚编辑过的图片。（图 2-34）

我们可以把图片复制出来进行使用，例如放在桌面上。

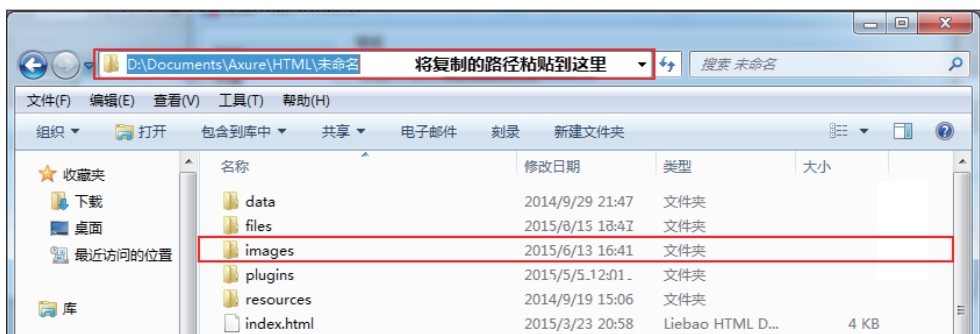


图2-33

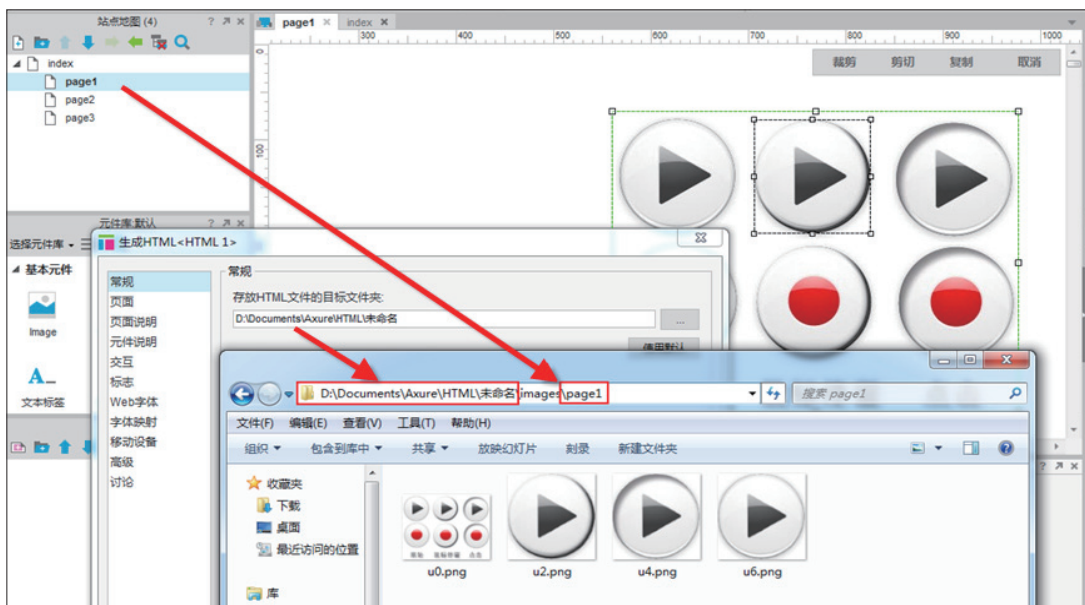


图2-34

4.3.5 图片的变换

一个图片按钮，往往有3种样式，初始、悬停、按下状态都会有不同的样式效果。

在前面我们裁剪的图片就是一个按钮的3种不同样式，我们可以通过在图片属性里设置交互样式，实现一个能够响应鼠标动作并变换样式的按钮。

- 步骤1** 拖入一个默认的图片元件到编辑区，调整成60px × 60px的尺寸。
- 步骤2** 双击图片打开本地资源管理器，找到初始状态的图片并导入进来；单击【确定】之后，由于我们里面的这个图片元件尺寸和导入的图片尺寸不一致，它会弹出一个提示，要求我们确定选择图片原始尺寸或保留编辑区图片元件的尺寸，这里我们选择【否】，让它保留编辑区元件的尺寸。
- 步骤3** 拉动图片左上角的三角形状，调整图片的圆角，我们把它拉动到图片中央，直到不能拉动为止。这样图片就变成了圆形的。
- 步骤4** 在元件属性中单击【鼠标悬停】，把悬停状态的图片导入。然后，单击顶部的【鼠标按下】标签并导入按下时的图片。（图2-35）

这样，我们就完成了一个动态图片按钮的制作。

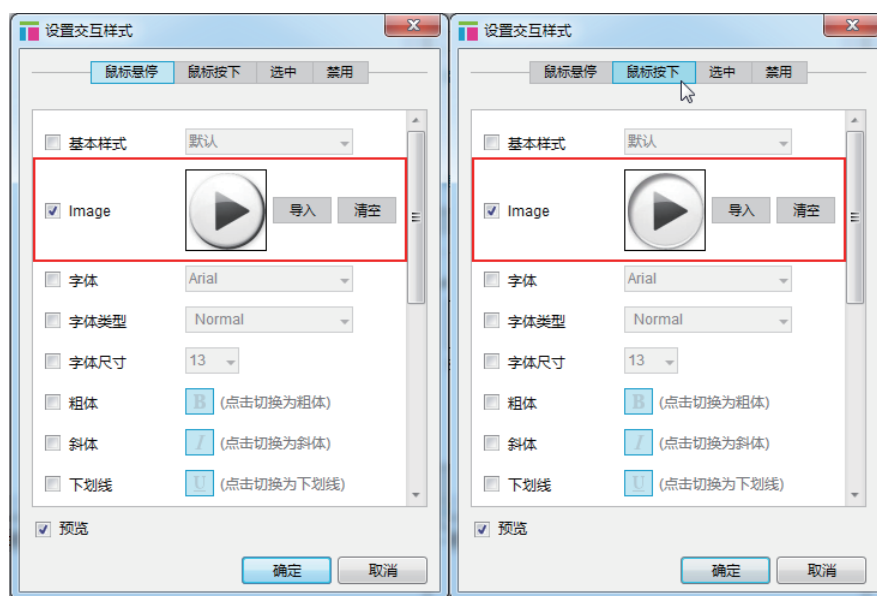


图 2-35

2.4.3.6 单选按钮组

在网页上经常能看到让用户选择性别，例如用户输入注册资料时。这种功能只有一个个性别选项能够被选择，当选择一项时，其他选项就会被取消选择。这种效果在 Axure 中也是通过元件属性来实现的。

因为是输入资料的功能，所以我们在表单元件中去寻找相应的元件。这个元件就是单选按钮。

性别选择的功能实现过程如下：

步骤 1 拖入一个单选按钮，按下键盘的 <Ctrl> 键，同时拖动元件后放开，把元件复制成 3 个；当然，也可以选中元件，通过 <Ctrl>+<C> 键复制、<Ctrl>+<V> 键粘贴来完成复制；复制之后，双击每个元件的文字，改成不同的性别。

备注

完成这一步后，如果我们预览或生成查看效果，你会发现每一个选项都能被选择。单选的关键在后面的步骤，请继续往下看。

步骤 2 按住鼠标左键，圈选这 3 个元件，把它们全部选中；也可以按住 <Ctrl> 键逐一单击这 3 个元件完成选中操作。

步骤 3 在元件属性中找到 {指定[单选按钮]组名称}，在后面的输入框中随意填写一个组名称。

完成第 3 个步骤后，就完成了唯一选中一项的效果。

也就是说，如果我们想让一组这样的元件每次只能其中之一被选择的话，只需要把这几个单选按钮的元件都选中，然后在属性中设置一个单选按钮的组名称，把它们放到一个组里就能实现唯一被选中的效果。最后，如果有需要默认被选中的单选按钮，只需在编辑区单击该元件前面的圆圈，将其改成选中的状态即可。（图 2-36）

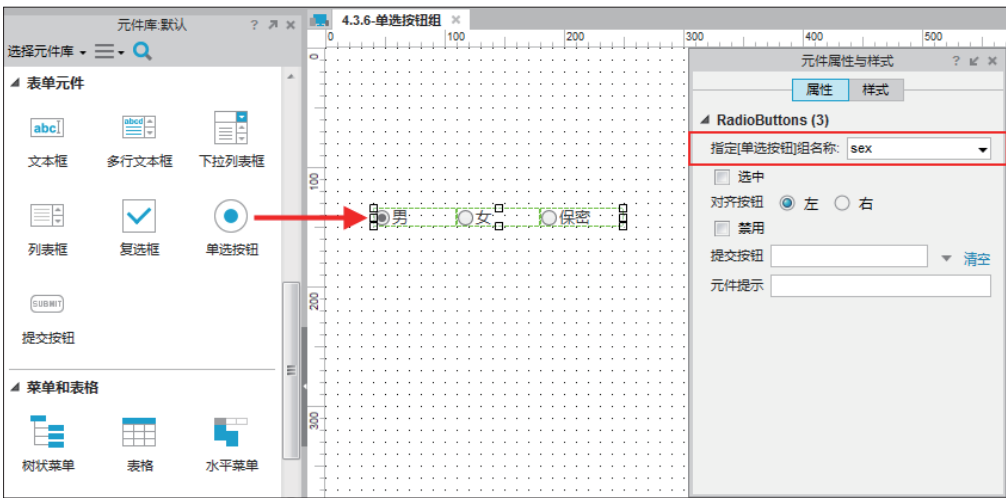


图 2-36

2.4.3.7 文本框类型

在元件的属性中，有些元件会具有自己独特的属性，例如文本框。

文本框的属性中，选择不同的{类型}，在选择生成或预览后会呈现不同的效果。但是这些效果并不是每种浏览器都支持。例如下图中的日期和查找两种类型在IE和火狐浏览器中是无效的。（图2-37）



图 2-37

经过测试，3种主流浏览器的支持结果如下。

IE9：文本、密码、文件。

谷歌：文本、密码、Number、查找、文件、日期、Month、Time。

火狐：文本、密码、Email、Number、Url、文件。

我们能看到IE9的支持类型最少，而火狐和谷歌有各自能够支持的类型。谷歌浏览器对日期类型支持很好，而火狐浏览器则对Number、Email、Url 3种类型提供了验证支持，表现为如果输入错误的内容，文本框边框会变成红色。

2.4.3.8 文本框文字提示

在Axure RP 7.0版本中，文本框的属性中提供了一个文字提示的功能。它能够实现在生成后文本框

中带有提示，当单击文本框时提示并不消失，只有输入文字时，提示才会消失的效果。文字提示也能设置文字的样式，来满足我们不同的需求。（图 2-38）

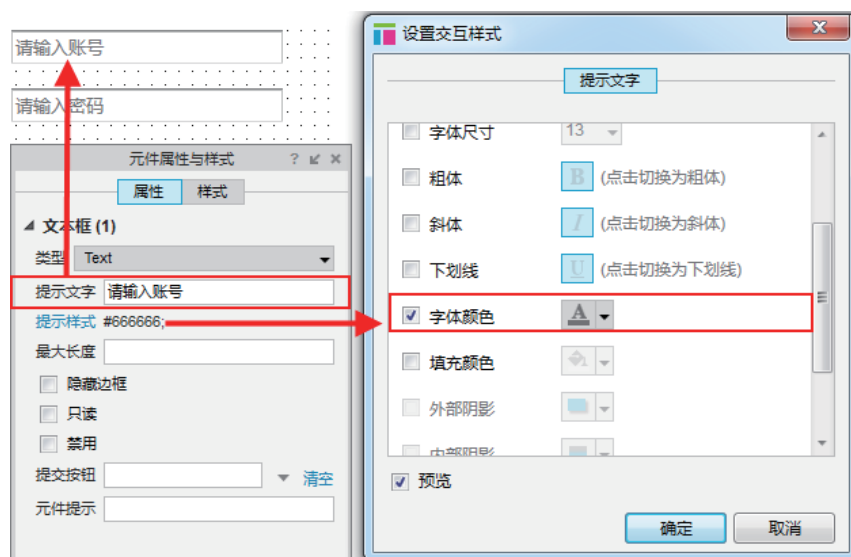


图 2-38

2.4.3.9 文本框文字长度限制

如果想限制一个文本框的文字输入个数，也能在文本框的属性中设置，例如限制输入 6 个字符（图 2-39）。

备注

Axure 中一个汉字按一位字符计算。

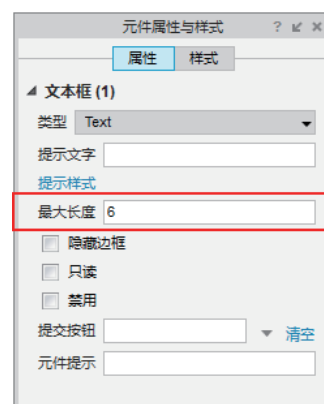


图 2-39

2.4.3.10 页面的镶嵌

在制作原型时，有时候我们需要在页面中展示另外一个页面的内容，或者插入一些多媒体文件。这些需求的实现则需要通过内联框架这个元件来实现。

之所以叫内联框架，就是因为它的功能是能够在页面内部与其他页面、地址、文件进行关联，完成嵌入的效果。

内联框架本身具有边框和滚动条，如果不需要它们可以在属性中进行相应的设置。内联框架在编辑区中因为不会直接显示内容，所以可以通过设置预览图像，让它能够被识别该框架关联的是什么内容。

最后我们能够在编辑区通过双击框架内部区域或者在属性中单击【框架目标页面】设置框架要关联的内容。

示例中，作者选择了【链接到 Url 或文件】，并填入了一个百度地图名片的地址。（图 2-40）

这样设置之后，我们看到的最终效果就是在页面中框架的位置显示的是百度地图名片。（图 2-41）

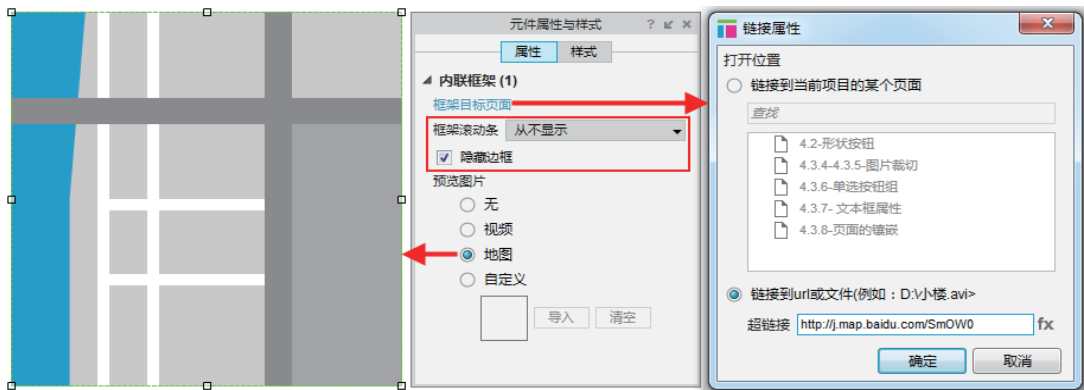


图 2-40



图 2-41

内联框架的内容，还可以通过设置交互动态进行改变。这些内容在后面的部分我们再去了解。

备注

百度地图名片链接制作地址：<http://api.map.baidu.com/mapCard/>。

2.4.3.11 元件的提示

在原型制作中，有时候一些元件需要在用户鼠标指针指向它的时候，显示一些提示。这个效果，也是在属性中设置来实现的。（图 2-42）

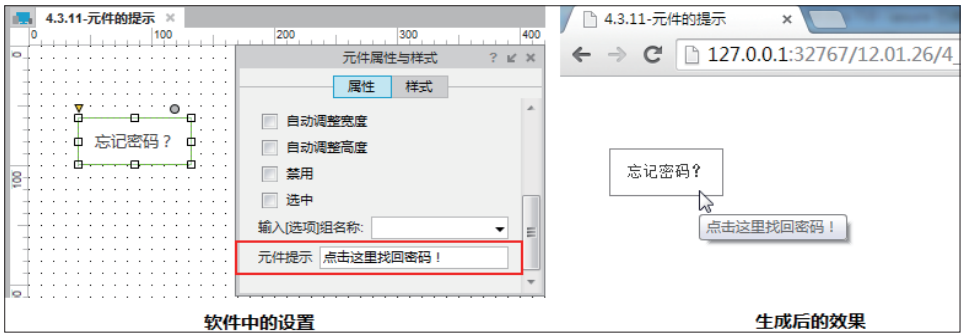




图 2-42

本节任务 → 完成本节中所有案例。

▶▶2.5 生成与预览

 楼哥，你一会儿生成，一会预览，感觉好乱啊！而且，查看效果的时候，那个跟软件里面很
像的站点地图很影响页面的美观……

 生成或者预览都是查看原型的方法，页面中的站点地图也可以进行设置，下面我给你详细
地讲一下。

2.5.1 生成与预览的设置

软件中不管是预览还是生成都有相应的设置。在预览设置及生成的常规设置里，不但可以选择使用哪种浏览器查看效果，还可以设置是否在生成后显示站点地图。（图2-43）

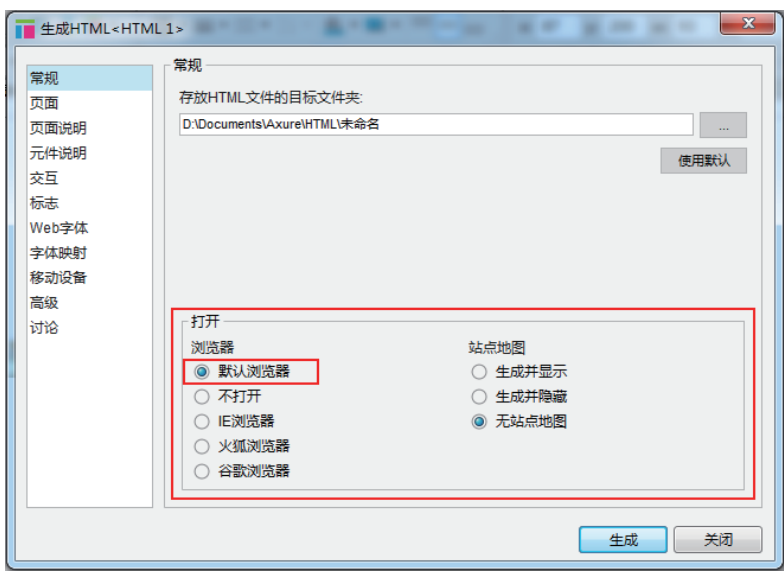


图2-43

国际上主流的浏览器名称基本都能够在选项中显示出来，而国内的浏览器则没有这种待遇。

那么，怎么使用国内的浏览器查看原型效果呢？

我们可以将需要使用的浏览器设置成系统默认的浏览器，然后在预览或者生成的设置里面选择使用默认浏览器就可以了。

在生成的设置当中，还有很多可以进行设置的内容，例如在生成时只生成指定的页面，就可以在【页面】设置中进行设置。

关于设置中的其他内容，我们在之后使用的时候再详细讲解。

2.5.2 生成与预览的区别

关于生成与预览的区别，同样一个原型在预览和生成时在浏览器的地址栏里面的地址是不一样的。

我们可以按键盘上的<F8>键打开生成设置页面，注意常规设置里面的文件夹路径。（图2-44）

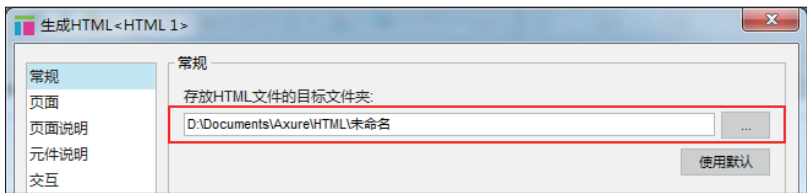


图 2-44

我们在前面也提到过生成HTML文件是保存到本地硬盘的一个文件夹里面，这个文件夹放在什么位置可以由我们自己来指定。那么，当我们单击生成按钮之后，浏览器地址栏显示的就是我们刚才设置的文件夹路径（图 2-45），不过中文部分有时候在一些浏览器中会变成一些特定的字符。

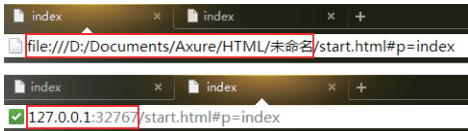


图 2-45

而我们按键盘上的<F5>键进行预览的时候，在浏览器中的地址则是一个以“http://127.0.0.1”开头的网址（图 2-43），“127.0.0.1”就是本机的IP地址，这种方式打开原型就像把HTML文件上传到一个网站服务器，然后通过网址打开的效果。

因为这两种方式的区别，我们在查看原型效果时，如果出现问题可以尝试切换不同的查看方式来尝试解决。

▶▶ 2.6 元件交互与说明

软件界面的右上角，是元件交互与说明功能面板。这是一个非常重要的区域，原型里面的动态效果绝大部分是在这里实现的。而且当一些效果不能实现或者实现起来浪费精力与时间、没有价值意义的时候，也可以通过元件说明进行文字备注。另外，元件说明也能体现在软件生成的文档中，规范、全面的元件说明能够让你生成的文档更容易修改成可用的产品需求文档。

2.6.1 元件的名称

在元件交互与说明模块中的第一项是元件名称的编辑框，为元件添加名称主要是为了让我们能够准确方便地选取元件，特别是一些需要添加交互或者在编辑中经常要用到的元件。所以，给元件命名是我们进行原型设计要养成的一个良好习惯。

备注

元件命名尽量不要使用中文，否则会导致一些不明错误。

2.6.2 元件的交互

在交互面板中首先是【添加用例】和【创建链接】两项。【创建链接】能方便地设置元件被鼠标单击时跳转到当前项目的其他页面。【添加用例】则可以在单击下方的任意一个触发事件后添加打开用例编辑界面。

我们在生活中经常会计划一些事情来应对不同的情况。例如：下雨的时候，如果雨很大就穿雨衣、穿雨鞋，如果雨小就打一把雨伞。这样就会实现一个避免淋雨的结果。

做原型中的交互就像我们计划事情一样，由触发事件（下雨时）、判断条件（雨大或者雨小）、情形（雨大时的动作和雨小时的动作）来组成。其中，每种不同的情形会包含不同的动作。

另外，有时候我们不需要判断，例如：下雨的时候穿雨衣避雨。这种情况在触发事件时就只有一种情形。

在 Axure 的交互中：

触发事件就是面板中的【鼠标单击时】、【鼠标移入时】等事件；

情形就是在触发事件中添加的用例（Case）；

情形的判断和执行的动作，则需要用例编辑界面中进行编辑。

通过上面的步骤，就能够完成一个对交互行为的编辑。例如：为形状按钮添加事件【鼠标单击时】【打开链接—当前窗口】【链接到 Url 或文件】“http://www.baidu.com”，就能够实现形状按钮被单击时在浏览器当前窗口打开百度首页的效果。（图2-46）

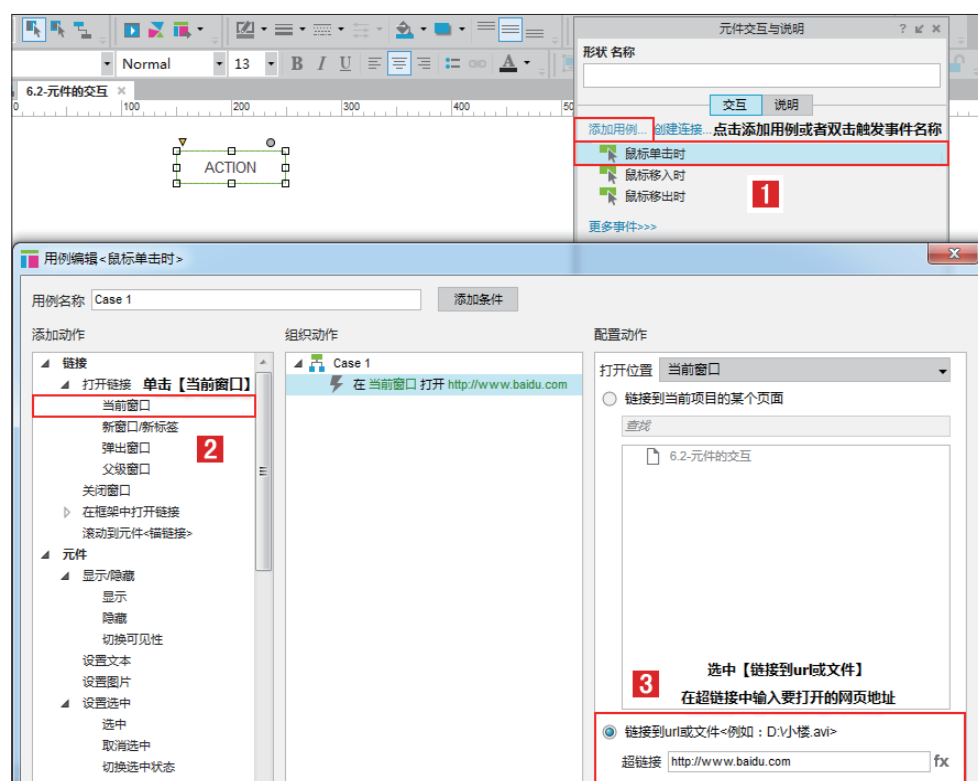


图2-46

在上面的操作中，用例编辑界面，一共包含了5个部分。

- 用例名称：修改用例名称，便于识别或者组织文档。
- 添加条件：为之后的动作添加限制条件，仅在满足条件时执行。
- 添加动作：选择在符合条件时执行的动作；
- 配置动作：选择动作的目标对象，并进行相应的设置。
- 组织动作：为添加的动作调整先后顺序。

这5个部分的作用，我们用生活中的例子来对比一下：晒衣服的时候如果要下雨就要把窗外的衣服收起来。

- 用例名称：下雨收衣服。
- 添加条件：如果要下雨。
- 添加动作：1-打开。
- 配置动作：1-窗户。
- 添加动作：2-收取。
- 配置动作：2-衣服。范围选择为所有。
- 添加动作：3-关闭。
- 配置动作：3-窗户。
- 组织动作：打开窗户>收取所有衣服>关闭窗口。

在Axure用例中的动作是由上至下执行的，所以顺序至关重要，就像我们在收衣服的时候必须先打开窗户，再收衣服，最后关闭窗口，不能打乱这个顺序，否则就会出现问題。

Axure中我们还可以给一个触发事件添加多个用例。（图2-47）例如：

- 步骤1** 再次双击触发事件【鼠标单击时】，或点中【鼠标单击时】再单击【添加用例】添加一个“Case2”；
- 步骤2** 选择动作列表{打开连接}中的【当前窗口】；
- 步骤3** 勾选【链接到url或文件】，填入网站地址“http://www.qq.com”。

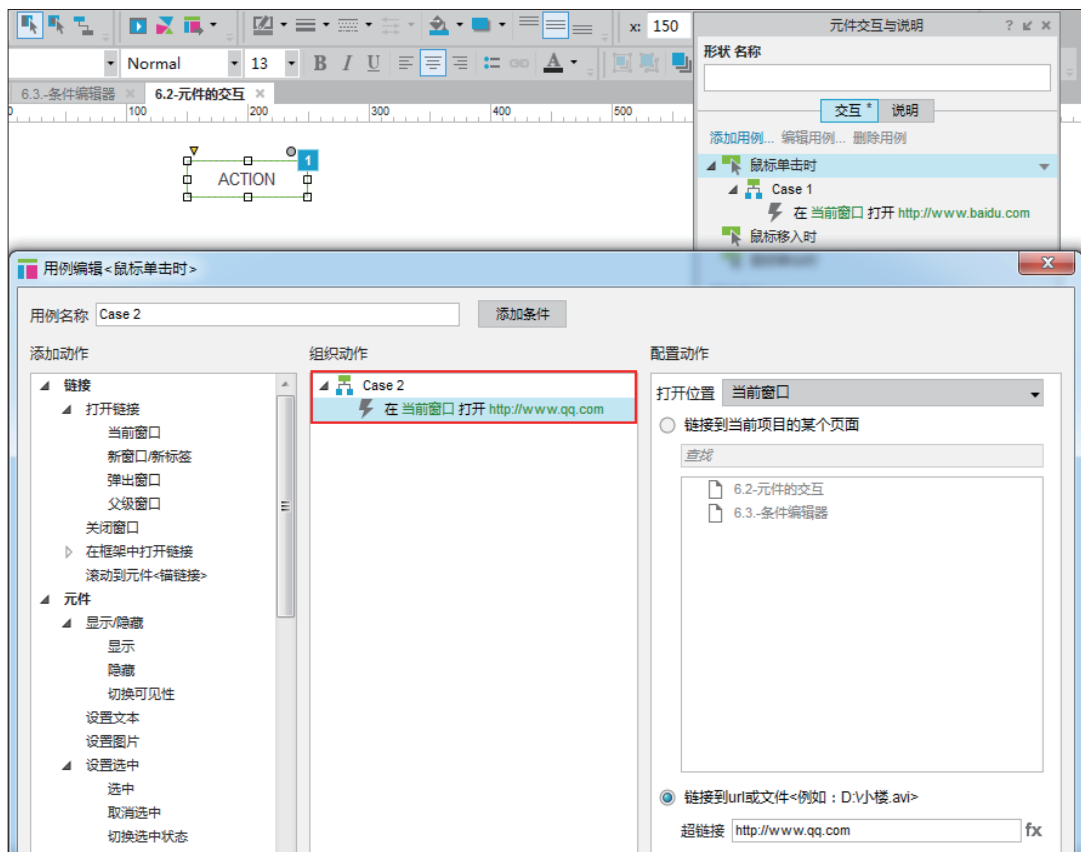


图2-47

备注

在用例编辑器左侧的动作列表中有多种动作类型，每种动作类型下还包含更多该类型的动作。大家要注意每种动作类型前方的三角形图标为白色时，则该类型的动作列表为收起的状态；当动作类型前方的三角形图标为黑色时，则动作列表为展开状态。

这时，我们按键盘上的<F5>键预览或者<F8>键生成HTML文件，查看一下原型的效果就会发现，当单击按钮时会弹出一个菜单，里面有两个选项，分别是两个用例的名称。只有单击其中一个用例的名称后才能执行相应的动作。（图2-48）

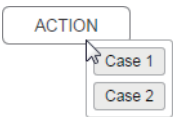


图2-48

这种情况说明，软件无法知道对于这两个用例应该先执行哪个后执行哪一个。所以要通过我们选择才能确定。因为这是在生成时对交互的控制，所以，在生成的设置中有相应的设置。（图2-49）

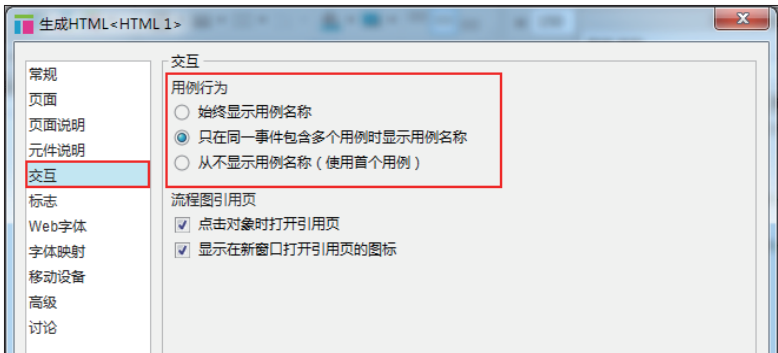


图2-49

生成设置的【交互】设置中，有3个{用例行为}的选项，默认情况下是第二项被选中，即当同一个事件有多个用例时通过显示用例名称让用户选择执行哪一个用例。

第一项是指不管事件有多少个用例，在被触发时都会显示用例的名称，在用户进行点选后执行相应的用例。

最后一项是指不管包含了多少个用例都只执行第一个，无需用户选择。

一般情况下，如果我们需要多个动作都被执行，就要把动作全部添加在一个用例里面，这样软件就会按从上至下的顺序去执行动作。但是一定要注意，例如在当前窗口打开页面或url的动作，在执行时，其后面的动作是不会继续执行的。因为在执行当前动作时已经改变了页面，前一个页面的动作将会全部随着页面被替换而失效。

那么当一个触发事件下有多个用例的时候，只能被用户选择执行其中一个，或者通过设置来只执行第一个吗？

不是的，当用例被添加了条件的时候，同一触发事件下的所有用例就会产生关联，这时候软件就能够自动根据是否满足条件而执行一个或多个用例。

2.6.3 条件判断



楼哥，给元件添加交互的步骤我大概了解了，刚才你说的添加条件怎么做呀？

 条件设置是给触发事件的用例添加的逻辑判断，让程序知道在满足条件时应该怎么做，不满足条件时又应该怎么做。

2.6.3.1 条件的设置

在上一节我们讲过，通过给用例添加条件让软件去自动选择可执行的一个或多个用例。那么如何给用例添加条件呢？

这里我们通过一个例子来了解条件的设置。

例如：选择下拉列表框中的不同选项，打开相应的搜索引擎主页。

下拉列表框是用来输入数据的元件，在表单元件中。在向主编辑区中拖入一个下拉列表框后，点击并为它添加一个名称“search”。

下拉列表框可以通过双击或者单击右键选择【编辑列表项】来添加预置的选项。（图2-50）

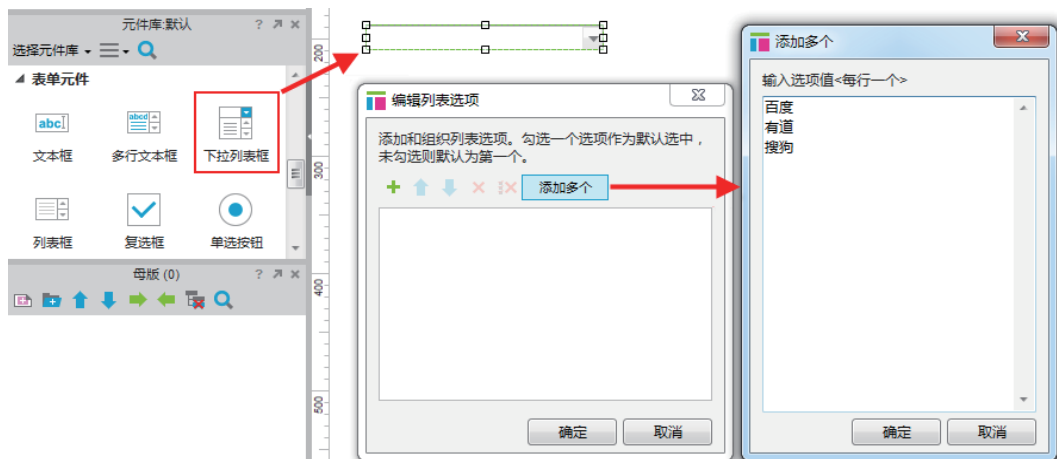


图2-50

选项可以单击对话框中的【+】进行逐个添加，也可以单击【添加多个】。

添加多个选项需要在输入每个选项后，按<Enter>键换行，保证每行一个选项，最后一行输入完毕不要再换行，否则会出现空的选项。

完成选项的输入后，可以在选项列表中通过勾选设置默认显示的选项，如果不选择任何一项的话，元件会默认显示第一个选项。这里我们让“搜狗”为默认选项。（图2-51）

完成下拉列表框的设置后，我们拖入一个形状按钮，双击并修改它的文字为“打开搜索引擎主页”。

接下来，在这个按钮单击的时候，我们就要进行条件判断，通过不同的选项去打开不同的搜索引擎页面。

操作步骤如下。（图2-52）

步骤1 点中形状按钮，右侧的元件交互与说明面板中出现这个元件的各种触发事件。

步骤2 因为是单击按钮时候产生的交互，所以，双击它的触发事件【鼠标单击时】并打开用例编辑界面。

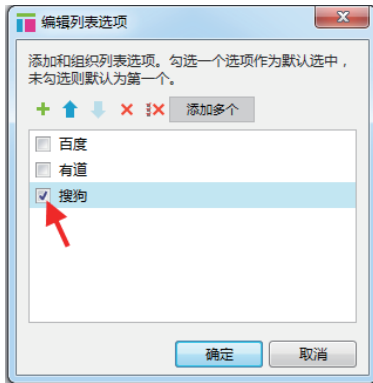


图2-51

- 步骤 3** 在用例编辑界面中单击【添加条件】，打开条件编辑界面。
- 步骤 4** 因为我们需要判断下拉列表框中的被选项是什么，然后执行相应的动作，所以，我们在前面的判断对象列表中选择【被选项】。这时软件会自动把下拉列表框显示在后面的输入框中。
- 步骤 5** 在后面的判断内容列表中，我们选择【选项】。这时软件会自动把下拉列表框中预置的选项显示出来。
- 步骤 6** 选项需要依次判断，这里我们先选择【百度】，单击【确定】。

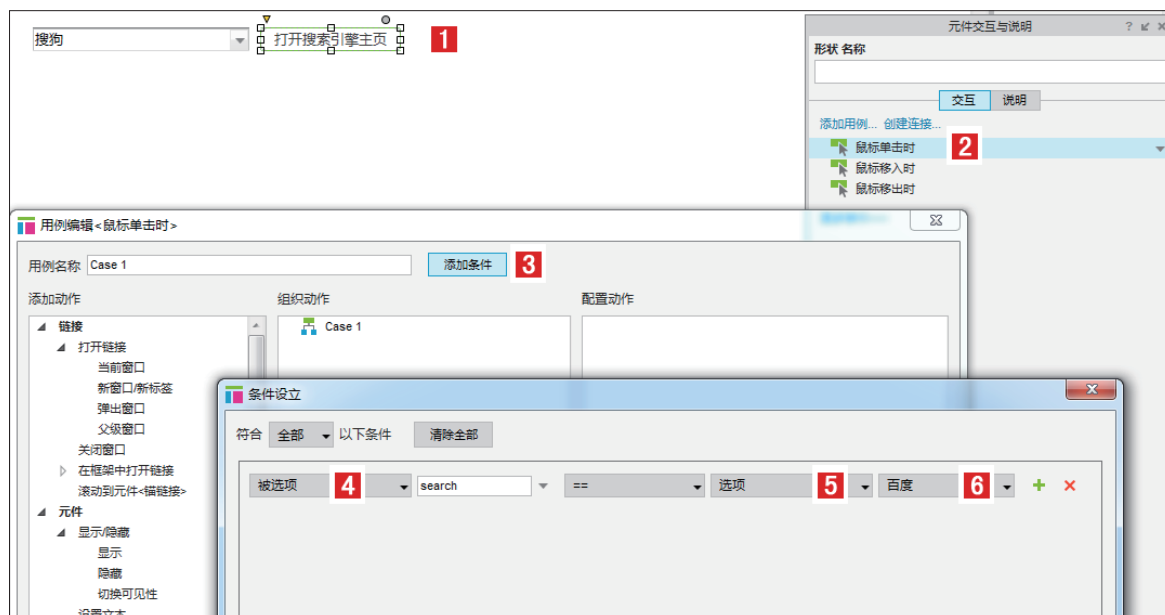


图 2-52

当我们完成上述操作，就完成了为当前的用例添加条件的操作，这时在用例编辑界面的{组织动作}区域中，就会显示出我们刚刚设置的条件。（图 2-53）

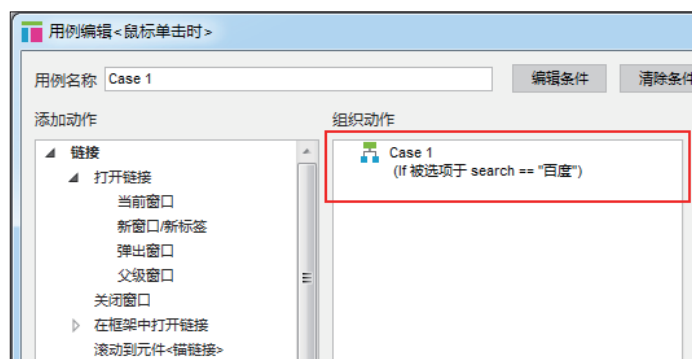


图 2-53

接下来，就是继续设置满足这个条件时需要执行什么动作。（图 2-54）

- 步骤 1** 单击左侧的{打开链接}中的【新窗口/新标签】，添加在浏览器中打开新窗口的动作。
- 步骤 2** 在配置动作中，设置【链接到url或文件】的超链接为“http://www.baidu.com”，然后单击【确定】退出用例编辑界面。

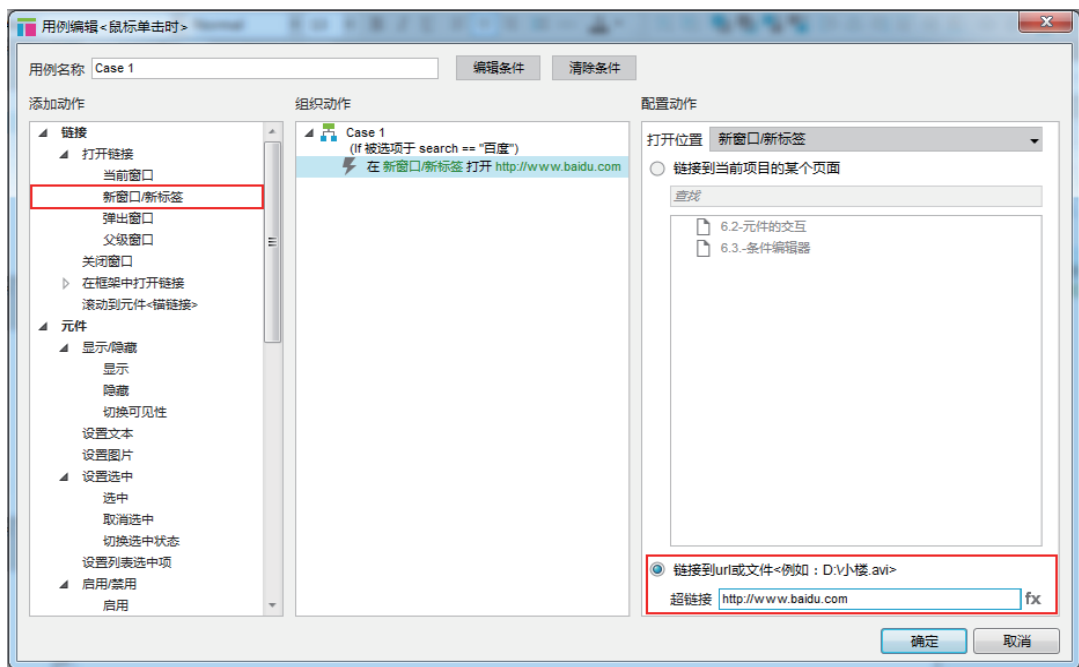


图 2-54

这时，如果我们按键盘上的<F5>键预览或者<F8>键生成HTML查看效果的话，当在下拉列表框中选择【百度】这个选项时，单击按钮就会在新标签（多标签浏览器）或者新窗口中打开百度的首页。但是选择其他选项单击按钮还没有打开页面的效果。

我们需要再次双击触发事件【鼠标单击时】，或者点中【鼠标单击时】再单击【添加用例】来继续添加条件、设置动作，来完成其他搜索引擎页面的打开效果。最终，我们给触发事件【鼠标单击时】添加了3个用例，实现了目标效果。（图 2-55）



图 2-55

2.6.3.2 条件的类型

楼哥，你刚刚举的例子我明白了。不过看上去这个例子只有一个满足条件的用例内容会被执行。如果想要一个触发事件中有多个满足条件的用例内容需要执行呢？

你很细心。刚才我举的例子只是条件判断中的一种类型，关于条件判断还有很多种类型需要了解。

- 1. If、Else If、Else If True 和 If True
If 中文含义是“如果”；
Else 中文含义是“否则”；
True 中文含义是“真”；
还有一个 False 中文含义是“假”，这个之后我们会用到。

If单独出现时，表示如果满足什么样的条件，就执行什么动作；

Else If 会在 If 所在用例之后的用例中出现，表示否则当满足另外一个条件时，会执行什么动作；

Else If True 会在一组条件判断的全部用例的最后一个用例中出现，表示如果前面所列举的条件都未满足的时候，执行什么样的动作；

If True，表示无条件执行动作，它只会在多组条件中出现。



一组和多组条件判断指的是什么？



多组条件判断，是指在一个触发事件中，允许包含多组条件判断。多组条件判断必须包含多个 If，如果只有一个 If，不管有多少用例都只是一组条件判断。如果要在多组条件判断中将一组条件判断分割出来，最简单的划分就是从 If 开始到下一个 If 出现前的所有用例条件为一组条件判断，如果 If 的下方没有再次出现 If，那么从当前的 If 开始到结尾是一组条件判断。



感觉好乱哦！



下面我就给你讲讲单组条件判断和多组条件判断的几种常见类型，分开理解就不会乱了。

2. 流程图的使用

为了更好地理解条件判断的结构，我们可以通过流程图来帮助我们理解。Axure 本身也具备画流程图的功能，我们只需要选择元件库为【流程图】，并且把快捷功能区中的选取模式改为【连接模式】，就可以进行流程图的制作了。（图 2-56）

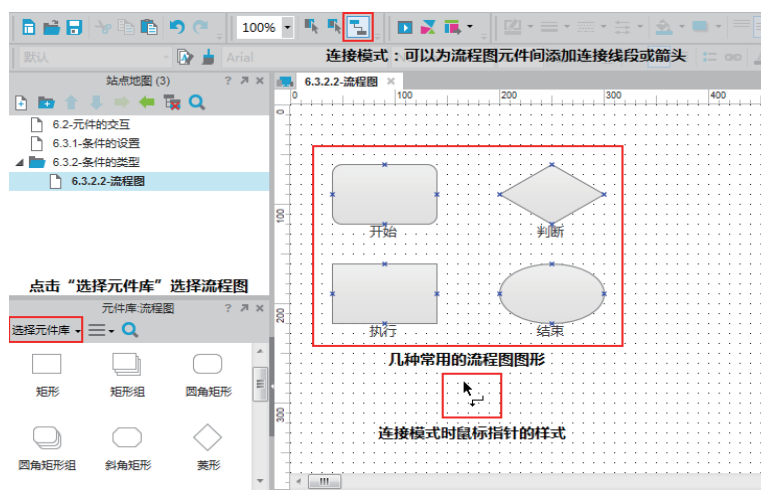


图 2-56

流程图中的图形一般都需要用线段或箭头连接，来展示一个有顺序的流程结构。连接图形只需要在图形的蓝色*边界点上，这时连接点会变红，我们按下鼠标左键不放，拖动到另外一个图形的边界点，当边界点变红时松开，即可完成连接。连接线的样式可以在快捷功能区中进行更改。（图 2-57）

流程图不是 Axure 的主要功能，在这里不做过多讲解，

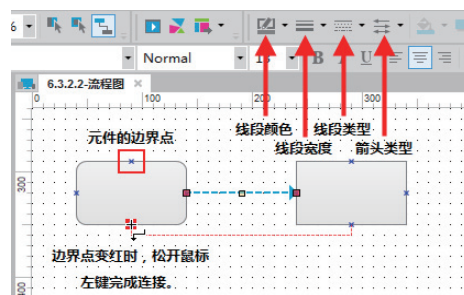


图 2-57

我们只是接下来用流程图协助我们理解条件判断。

流程图中有许多形状，都分别代表着不同的含义。

图片：表示一张图片。		三角形：表示数据的传递	
矩形：一般用来表示执行		梯形：表示手动操作	
圆角矩形：表示程序的开始或者结束		椭圆形：表示流程的结束	
斜角矩形：不太常用，可以自定义		六边形：表示准备或起始	
菱形：表示判断		平行四边形：表示数据的处理或输入	
文件：表示为一个文件		角色：模拟流程中执行操作的角色是谁	
括弧：注释或者说明		数据库：指保存数据的数据库	
半圆形：表示页面跳转的标记			

3. 单组条件判断

类型A：

If <条件>{ 动作1，动作2，……}

这种类型只有一个If，满足这个条件时执行相应的动作。如果不满足条件则没有动作执行。

例如：(图2-58)

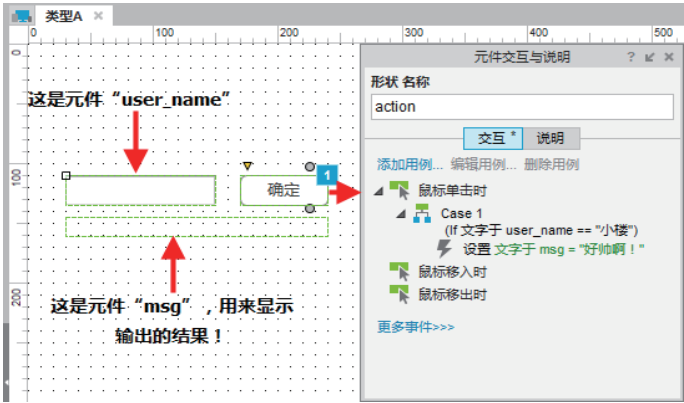


图2-58

我们通过判断文本框元件“user_name”中输入的值，在单击“确定”的时候在文本标签元件“msg”上显示一个结果。

为了实现这个效果，我们给【确定】按钮添加用例，在用例编辑中添加条件判断类型选择【元件文字】，元件选择名称为“user_name”的文本框，判断内容选择【值】，然后在后面的输入框输入要比较的值“小楼”。(图2-59)

这样就完成了对文本框里面所输入的内容是否为“小楼”的判断。

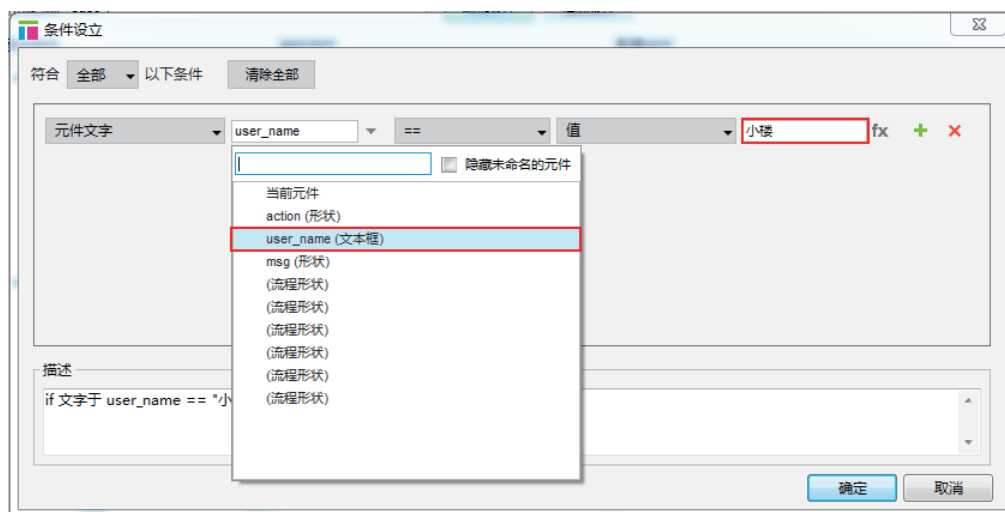


图 2-59

当满足这个条件的时候，我们选择动作【设置文本】，勾选名为“msg”的文本标签，为其添加【值】为“好帅啊！”。（图 2-60）

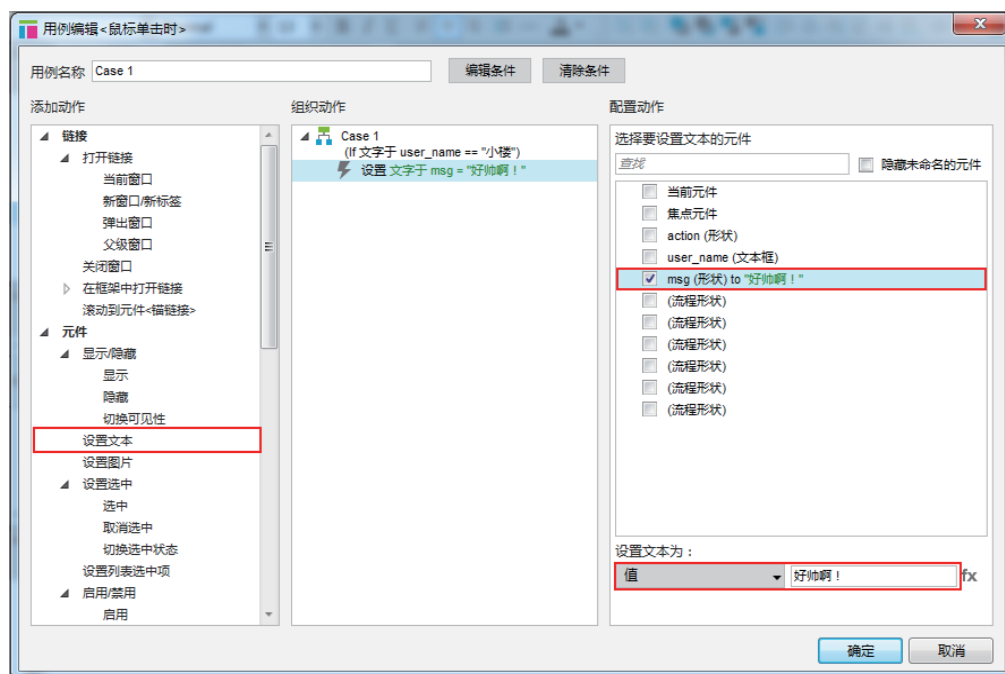


图 2-60

上方这个案例完成后进行测试，我们能看到当在输入框中输入“小楼”时，单击按钮页面上会显示“好帅啊！”，而按键盘上的<F5>键刷新页面后测试输入其他文字，单击确定按钮，则没有任何结果显示。（图 2-61）

我们用流程图来表示上面的步骤（图 2-62）：

通过这个流程图我们能看到，因为只进行了符合条件时的设置，所以只会在符合条件时执行相应的动作。

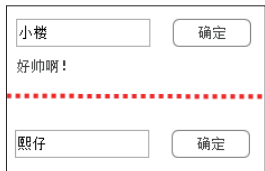


图 2-61

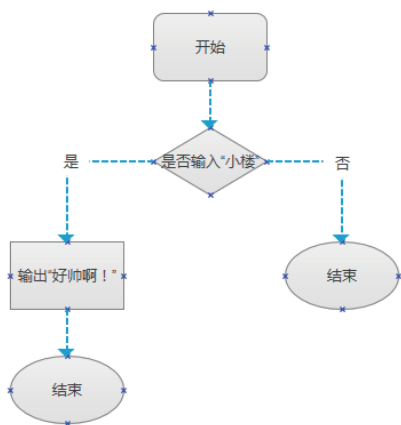


图 2-62

类型 B:

If<条件>{ 动作 1, 动作 2, ……}

Eles If True{ 动作 1, 动作 2, ……}

这种类型相当于在类型 A 的 If 用例后面又添加了一个用例，这时软件会自动把这个用例的条件设置为 Eles If True，我们给这个用例添加上动作，【设置文本】“msg”为【值】“很一般！”。（图 2-63）

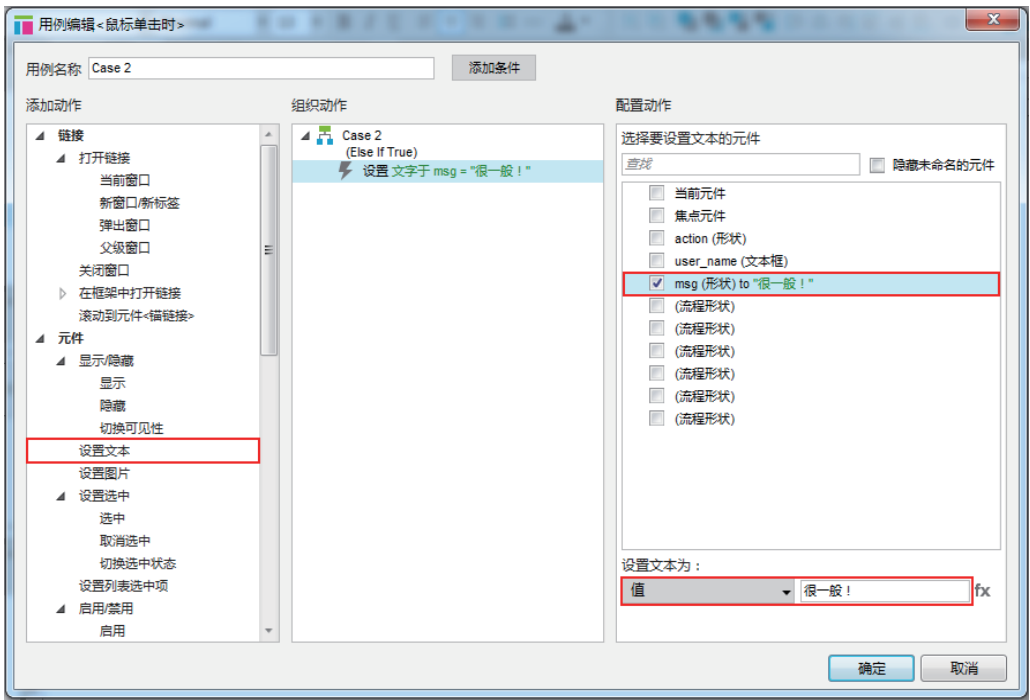


图 2-63

这样的结构就是如果满足 If 中设立的条件就会执行 If 中的动作，否则所有不满足 If 条件的情况就执行 Eles If True 下面的动作。

测试时，我们能看到当在输入框中输入“小楼”时，单击按钮页面上会显示“好帅啊！”，而输入其他文字，单击【确定】按钮，则显示“很一般！”。（图 2-64）



图 2-64

我们用流程图来表示上面的步骤。(图 2-65)

类型 C:

If<条件>{动作 1, 动作 2, ……}

Eles If<条件>{动作 1, 动作 2, ……}

这种类型相当于给类型 B 的第二个用例添加了一个条件，在第一个用例的条件不成立时，会根据用例 2 设置的条件进行判断，如果成立，执行用例 2 的动作，否则没有任何动作执行。

我们在类型 B 的用例 2 编辑界面中，单击【添加条件】添加如果输入框的【元件文字】“user_name”为【值】“熙仔”时的判断，然后为这个判断添加动作，【设置文本】“msg”的元件文字为【值】“很一般！”

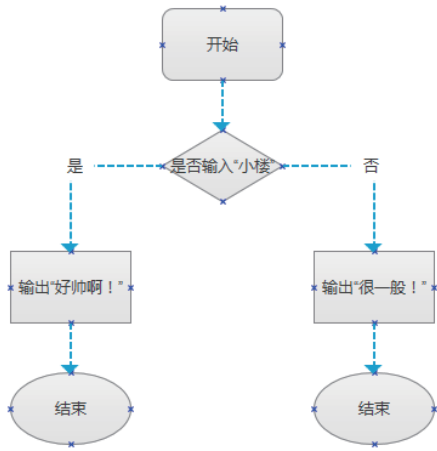


图 2-65

经过这样的设置，我们能够看到测试时，文本框输入“小楼”，会显示结果“好帅啊！”，而输入“熙仔”时会显示结果“很一般”。按<F5>键刷新浏览器，输入其他文字则没有任何动作。(图 2-66)



图 2-66

我们用流程图来表示上面的步骤。(图 2-67)

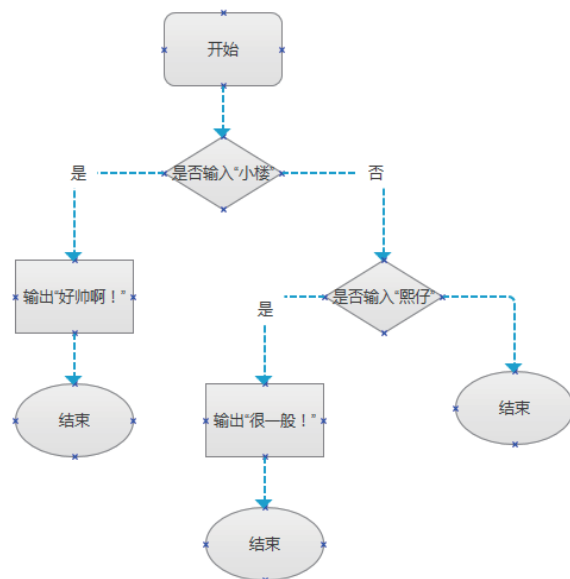


图2-67

类型D:

If<条件>{动作1, 动作2, ……}

Eles If<条件>{动作1, 动作2, ……}

Eles If True{动作1, 动作2, ……}

这种类型是有多个可判断的条件, 由上至下进行判断, 只要被判断的条件符合, 则执行该条件下的动作, 并结束判断的过程。否则, 继续执行下一判断, 当全部可判断条件均不符合时, 执行Else If True的动作。

就像类型C的案例, 我们分别对两个输入的文字进行了条件判断, 当不满足这两种条件时, 并无动作可执行。而当前的这种类型, 就是在Else If True中加上不符合前面所有条件时要执行的动作。(图2-68)



图2-68

我们用流程图来表示上面的步骤。(图2-69)

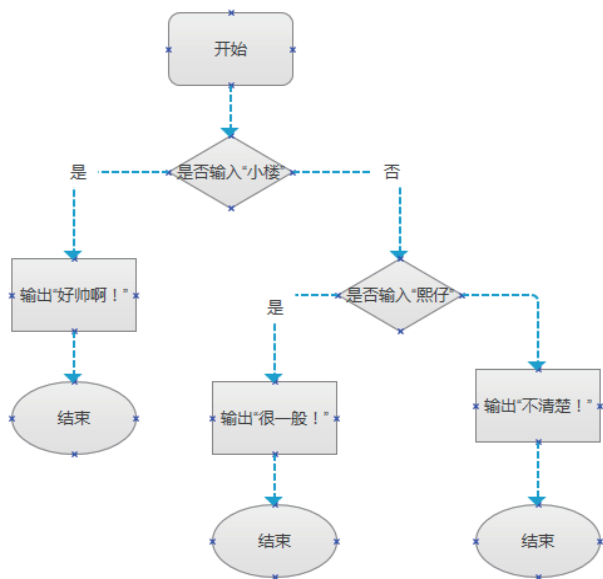


图2-69

以上就是我们在 Axure 原型制作中经常使用到的 4 种比较基本的条件判断结构，其中类型 C 和类型 D 都可以在结构中继续添加 Else If<条件>来增加判断条件，完成更多种条件的判断。

备注

一定要注意的是，单组条件判断只有第一个满足条件的用例中的动作才会被执行，即便有多个条件能够同时被满足，也只会执行第一个满足条件用例中的动作，不会出现执行完一个满足条件的用例后再继续执行下一个满足条件的用例的情况。没有任何条件被满足的时候，如果最后以 Else If true 的用例结尾，则执行最后这个用例中的动作，如果没有 Else If True 则不执行任何动作。

4. 多组条件判断

多组条件判断，是由多个单组条件判断组成。每一个单组条件判断中第一个满足条件的用例都会被执行（Else If true 是最后一个满足条件的用例），没有符合条件的则不会被执行。所以多组条件判断的结构中都会有 0~N 个用例被执行。N 小于条件判断结构中组的数量。

类型 E:

If<条件>{ 动作 1，动作 2，……}

If<条件>{ 动作 1，动作 2，……}

这种类型一般是在同一个用例中，有多种不同的条件分别被满足时都需要执行相应的动作。

例如对一个文本框输入的内容进行判断，该文本框只能输入 6 位以下数字，否则需要针对每种不符合条件的输入给出提示。当单击【确定】按钮时，我们要分别判断，如果不是数字，给出一条不是数字的提示；如果位数超出 6 位也要给出位数超出限制的提示。

这个案例的实现步骤如下。

步骤 1 双击【确定】按钮的【鼠标单击时】事件，打开用例编辑界面；然后，单击【添加条件】。选择及输入条件类型【元件文字】“user_input”的判断内容为【不是】【数字】，完成条件设立。（图 2-70）

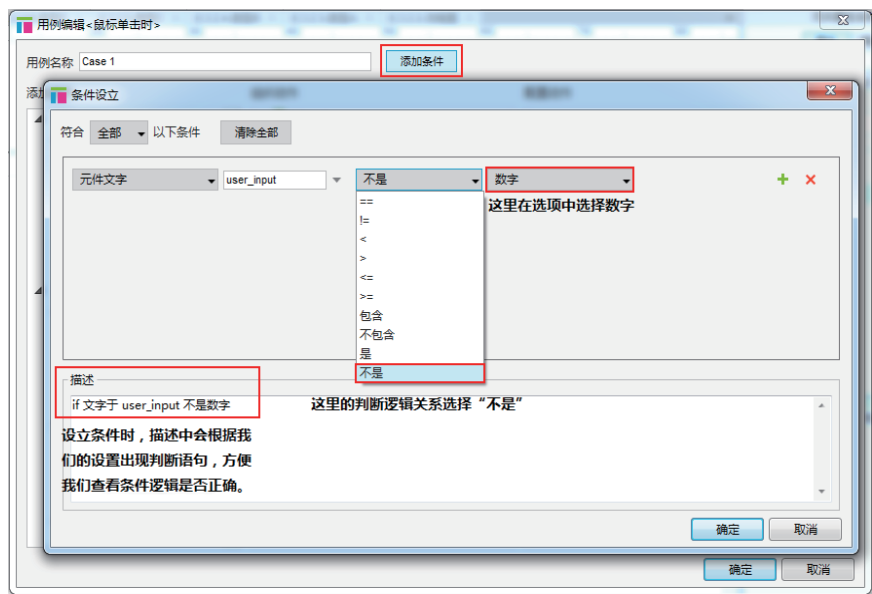


图 2-70

步骤 2 添加动作【设置文字】“msg_01”为【值】“输入的不是数字！”，完成动作设置后，单击【确定】退出用例编辑界面。

步骤 3 在交互面板中单击【添加用例】，继续为【确定】按钮的【鼠标单击时】事件添加用例；在打开的用例编辑界面中，单击【添加条件】进行条件设置，选择及输入判断内容为元件“user_input”的【元件文字长度】【>】【值】“6”，完成第二个条件设立。（图 2-71）



图 2-71

步骤 4 添加动作【设置文字】“msg_02”为【值】“输入的文字超过6位！”，完成动作设置后，单击【确定】退出用例编辑界面。

步骤 5 此时，我们能够看到，在交互面板中，【鼠标单击时】事件中的两个用例是单组条件判断中类型 B 的结构。我们只需要在下面的用例名称上单击鼠标右键，然后在菜单中单击最后一项【切换为

<if>或<Else If>】，就能把这个结构变成我们需要的结构。(图2-72)

完成上述步骤，我们进行测试:(图2-73)

- 步骤1 当输入“abc123”时，满足第一个用例的条件，会提示“输入的不是数字”；
- 步骤2 刷新页面，当输入“12345678”时，满足第二个用例的条件，会提示“输入的文字超过6位！”；
- 步骤3 刷新页面，当输入“abcd1234”时，两个用例的条件都被满足，所以会同时显示上述两个提示。

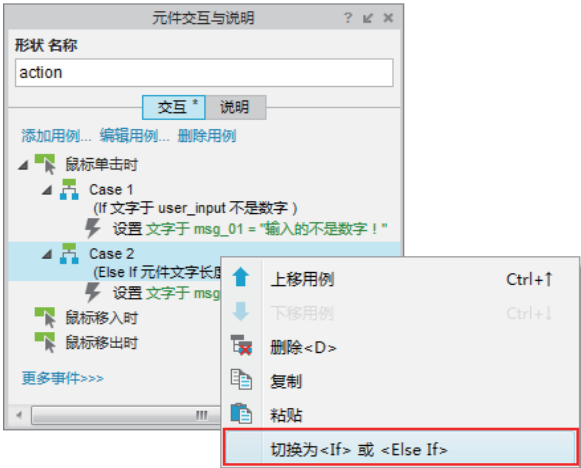


图2-72

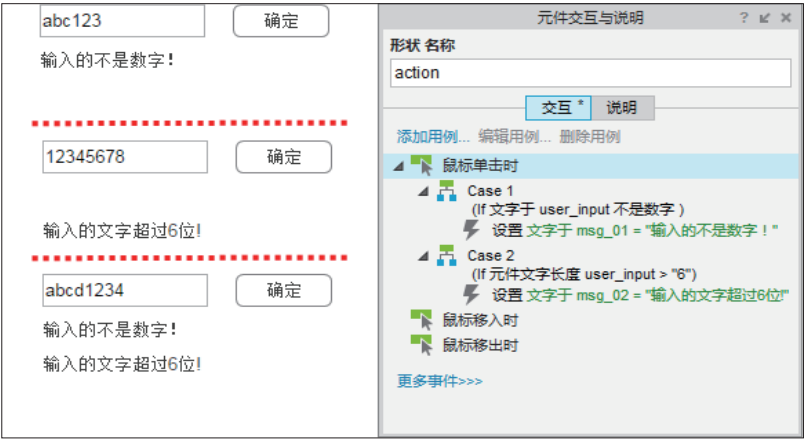


图2-73

我们用流程图来表示上面的步骤。(图2-74)

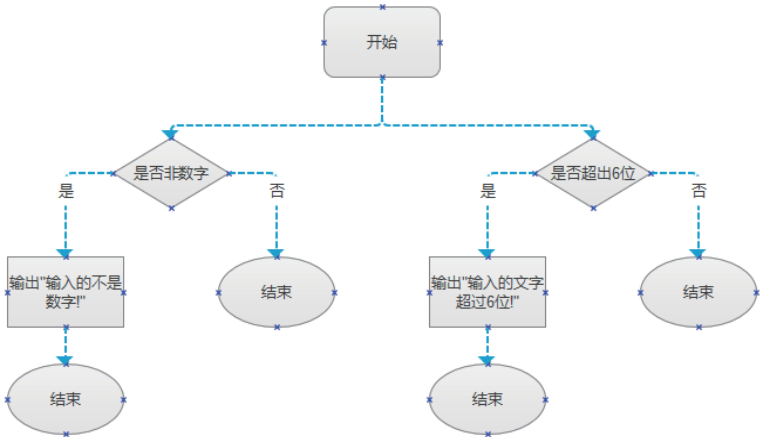


图2-74

类型F:
If<条件>{ 动作1, 动作2, ……}

If<条件>{ 动作1, 动作2, ……}

If True{ 动作1, 动作2, ……}

这是一种比较特殊的多组条件判断,我们能够看到最后一组条件判断只有 If True,并没有其他条件,这种情况是指无条件执行该用例中的动作。而且大家要知道,If True 只在多组条件判断中能够出现。

我们继续类型E中的例子,例如,我们在显示了提示后,等待3秒钟,让提示消失。

其实,我们可以在每种条件判断的动作中分别添加等待3秒钟并让文字为空(即消失)的动作。不过,既然有 If True,我们就直接用 If True 的用例来完成这两个动作。这样无论前面有多少条件判断,最后都会无条件执行等待3秒钟后让提示消失的动作。

我们分别使用两种方法在用例编辑界面中完成以下动作的添加:

步骤1 在动作列表中最下方找到动作类型{其他},选择其中的【等待】动作,然后在配置动作中把默认的“1000”毫秒(即1秒钟),改成“3000”毫秒。

步骤2 在动作列表中单击【设置文本】的动作,在配置动作中勾选两个用来显示提示的元件。这时软件下方可以输入【值】,不过,我们既然是清空提示的动作,在这里不做任何输入,软件执行该动作时就会把两个提示元件的文字清空。

备注

我们在使用 If True 的方法时,需要在类型E的案例基础上继续添加用例,这时系统会自动给出 Else If true,还记得怎么转成 If 吗?是的,只需要在这个新添加的用例上点右键,选择最后一项【切换为 <if> 或 <Else If>】。当切换完成,不添加任何条件就是 If True 了。

我们来看一下,上述两种方法分别添加动作后用例结构的对比。(图2-75)



图2-75

从对比图中我们可以看出,把每个用例中相同的必须执行的部分抽出来并一起使用 If True 用例来执行,能够让我们的用例变得更精简,而最终的执行结果完全相同。

我们用流程图来表示使用 If True 来实现的步骤。(图2-76)

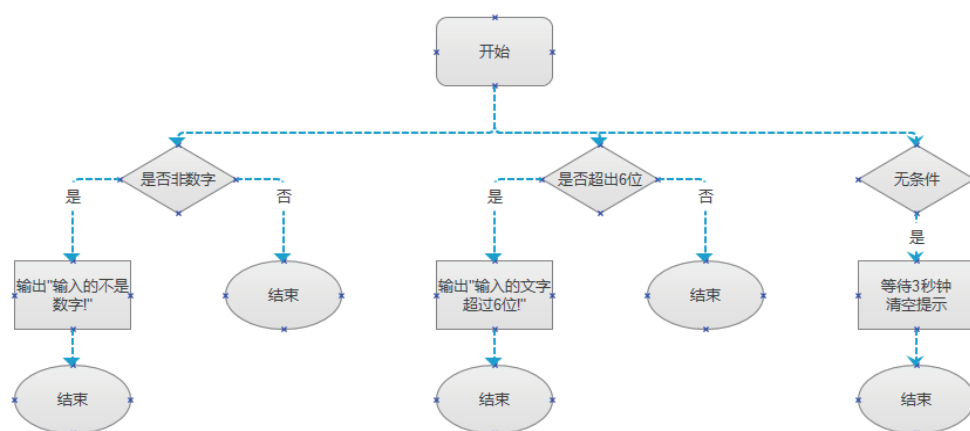


图 2-76

类型 G:

If<条件>{动作 1, 动作 2, ……}

If<条件>{动作 1, 动作 2, ……}

Eles If<条件>{动作 1, 动作 2, ……}

上面这种类型由类型 A 和类型 C 组成, 比我们看到的类型 E 在第二个用例里多出了一个具有条件判断的用例。那么, 我们就在类型 E 的基础上, 将第二个条件修改为输入的内容只能为 6 位, 这样当输入的内容超过 6 位和小于 6 位的时候, 都要给出提示。

我们来看一下添加了一个用例后, 新的用例结构。(图 2-77)



图 2-77

通过上图中的编辑, 就能够实现我们所需要的判断效果。

我们用流程图来表示使用上面用例的执行步骤。(图 2-78)

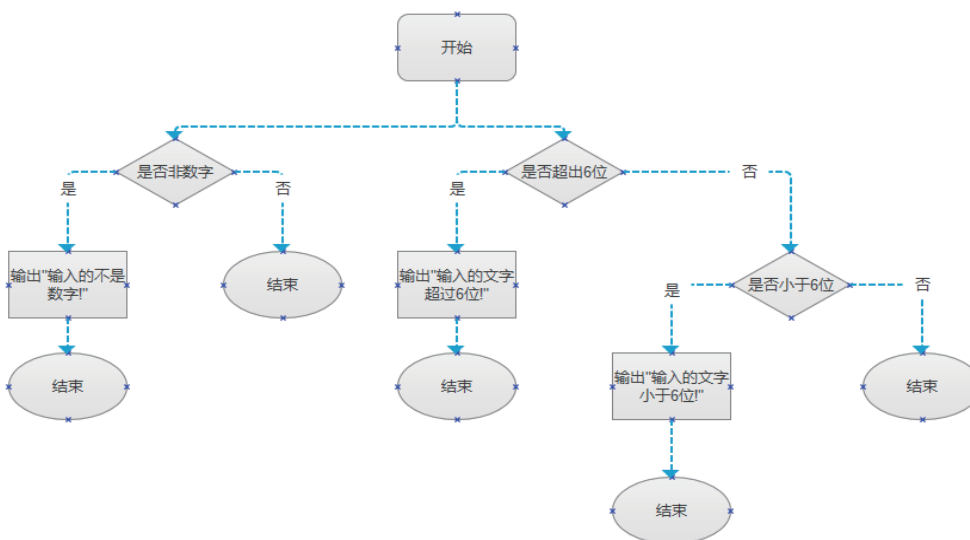


图 2-78

类型H:

If<条件>{ 动作1, 动作2, ……}

Eles If True { 动作1, 动作2, ……}

If<条件>{ 动作1, 动作2, ……}

Eles If True{ 动作1, 动作2, ……}

我们再来看一个类型，这个类型由两个类型B组成。我们通过一个简单的用户名密码验证案例来演示这种类型。我们放置两个文本框分别命名为“user_name”和“password”。然后，每个文本框下方各放置一个文本标签“msg01”和“msg02”，用来显示提示文字。在完成两个文本框的输入后，单击【登录】按钮，对这两个文本框依次进行判断，如果错误就显示红色的错误提示，否则显示绿色的正确提示。错误的依据为文本框输入的不是字母或数字。

我们先把必须的元件放入编辑区并进行命名（图2-79），然后在【登录】按钮上添加触发事件的用例与动作。

在添加用例动作时，我们可以参考前几个用例。不过有几点我们要注意：

步骤 1 添加条件时，要分别判断类型【元件文字】“user_name”及“password”【不是】【字母或数字】；（图2-80）



图2-79

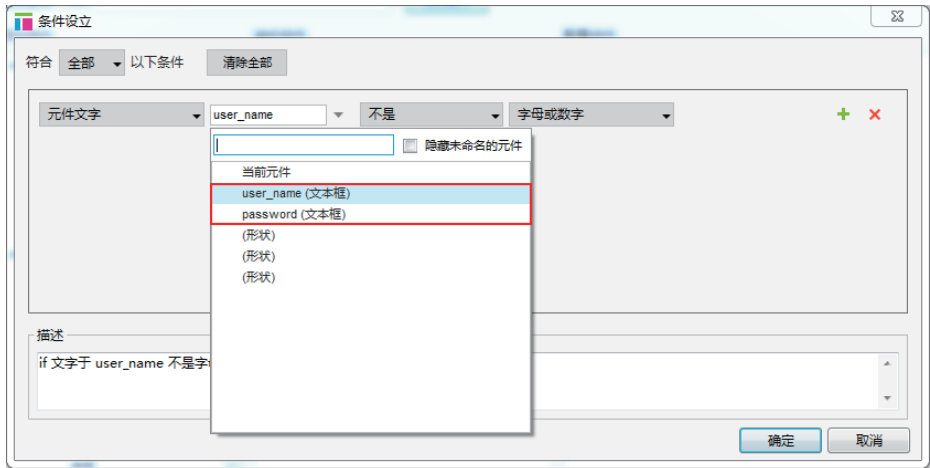


图2-80

步骤 2 设置提示文字时，要在配置动作中选择【富文本】，然后在编辑文字界面的样式设置中，给提示文字加上不同的颜色。（图2-81）

当完成所有用例的编辑后，完整的用例结构和动作的组成如图所示。（图2-82）

通过测试能够看到，输入数字和字母的组合时，没有满足不是数字或字母的条件，所以显示了绿色的正确提示；而输入几个特殊字符时，满足了条件，则显示了红色的错误提示。（图2-83）

总结:

以上只是列举了几种基础的条件判断组成，以及将基础条件判断任意组合形成的几种多组条件判断。其实，通过基础的条件判断进行自由搭配，还可以组成更多不同的多组条件判断类型。而且，我们可以在一个触发事件中完全可以根据自己的需求，设置几组甚至十几组条件判断。在这里我们就不一一列举，在之后的案例中，我们可以继续去体会。

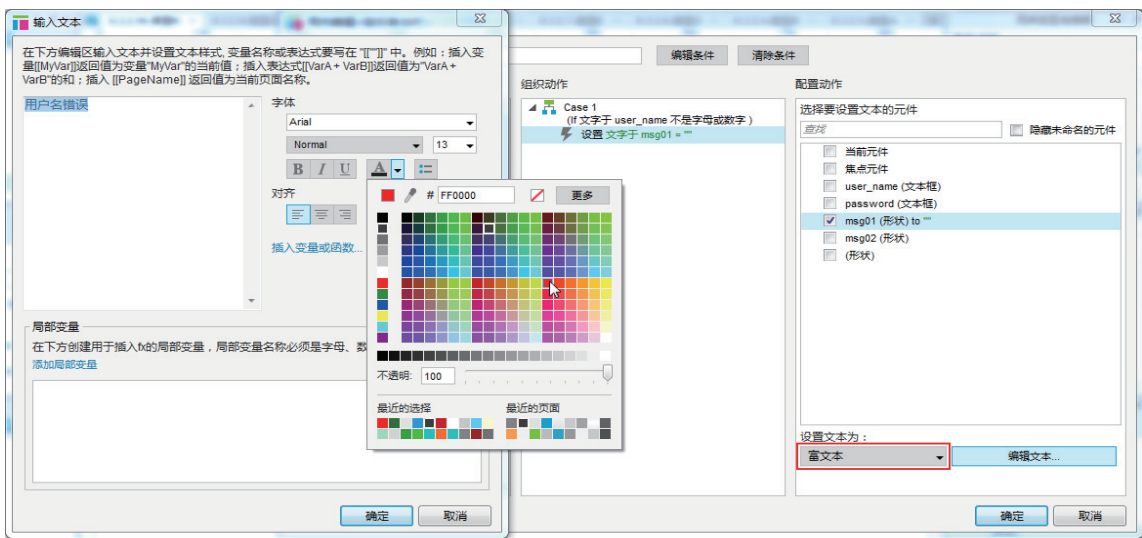


图 2-81



图 2-82

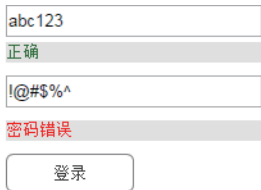


图 2-83

备注

需要大家注意的是，如何在多组条件判断中准确地分辨哪些用例为一组条件判断。其实很简单，每一组用例都是If开始，从上至下，一直到下一个If出现前的所有用例即为一组（图2-84）。如果之后没有If出现则一直至结尾都为最后一组条件判断。（图2-84）

本节任务 → 独立完成本节案例，理解各种条件类型。

5. 全部与任何

条件判断不但有以上的结构，在条件编辑器界面中还可以设立多种条件，在这些条件全部被满足或者任何一个被满足时去执行用例中的动作。



图 2-84

下面我们通过两个例子来演示满足全部条件和满足任何条件的不同之处。

(1) 满足全部条件

假设一个用户名输入框中输入的字符必须是4位以上字母或数字才显示正确提示图标，否则显示错误提示图标。

首先，先准备好元件。(图2-85)

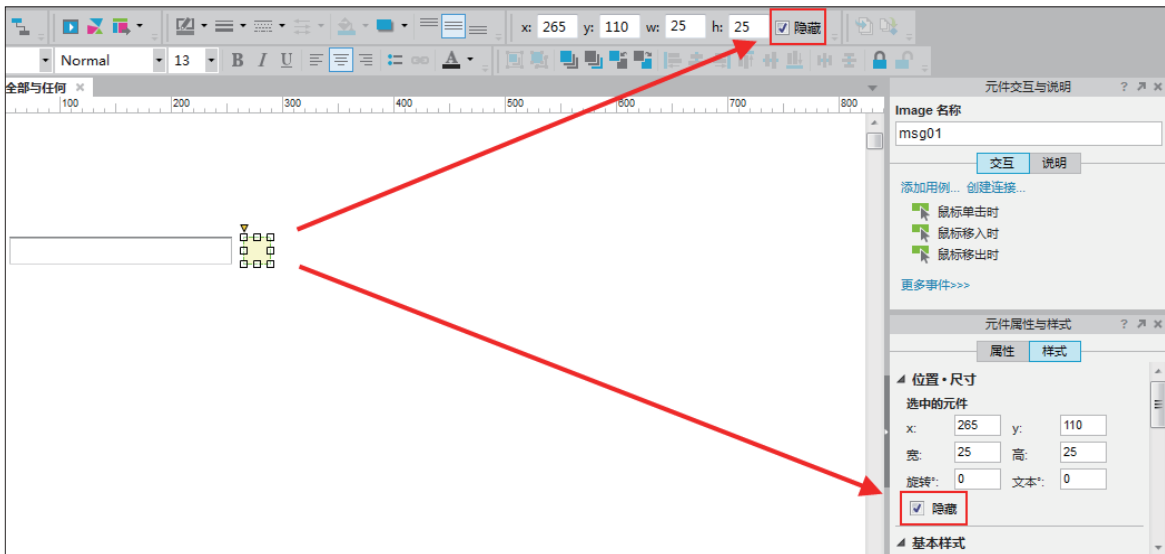


图2-85

步骤1 在编辑区先放入一个文本框，命名为“user_name”。

步骤2 在文本框后方，放入一个图片元件。因为提示图标在文本框未输入文字前不显示，所以我们选中这个图片元件，在快捷功能区或者样式中先勾选【隐藏】。勾选【隐藏】后图标变成淡黄色的矩形。

备注

如果不想让隐藏的元件显示为淡黄色，可以在【视图】中的【遮罩】选项列表中取消【隐藏对象】的勾选。这样被隐藏的元件，在编辑区也变成不可见的状态。

然后，我们进行条件设置。(图2-86)

步骤1 对于这个条件的触发，我们选择【失去焦点时】这个触发事件。

备注

失去焦点是指光标从文本框中消失，与其对应的是获取焦点，即光标进入文本框，呈现闪烁的状态。

步骤2 在【失去焦点时】用例编辑界面单击【添加条件】。在条件编辑界面，我们可以选择符合【全部】以下条件。

步骤3 添加第一个条件判断【元件文字】“user_name”【是】【字母或数字】，然后，单击后方的【+】，继续添加第二个条件，判断【元件文字长度】“user_name”【>】【值】“4”。完成这些设置后，单

击【确定】退出条件编辑界面。

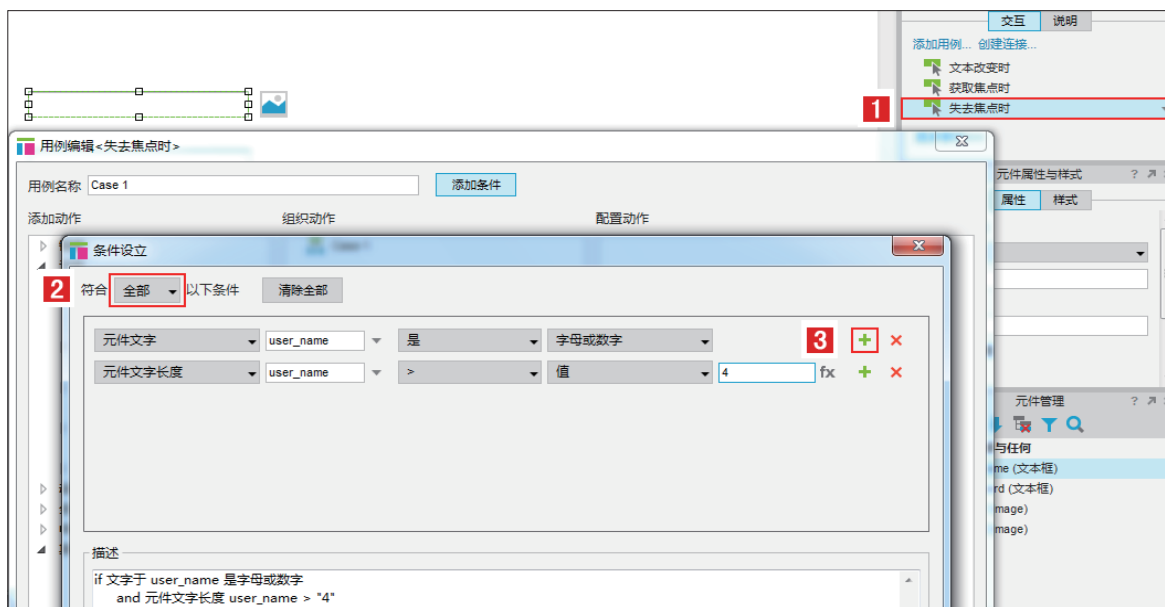


图 2-86

接下来，为用例添加符合条件时要执行的动作。

步骤 1 在动作列表中选择动作【设置图片】，在动作配置中勾选图片元件“msg01”，并在下方的{默认}设置中导入正确提示的图标。（图 2-87）

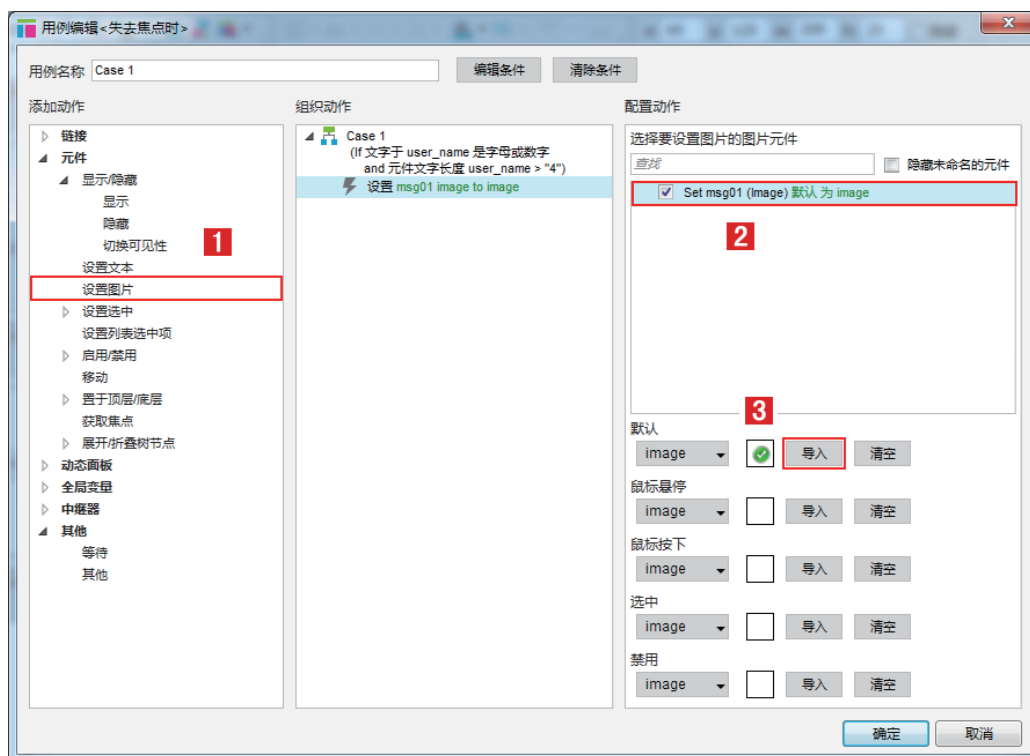


图 2-87

步骤2 因为提示图标初始为隐藏，这时需要让其显示出来，所以，我们再添加一个动作。在动作列表中选择【显示】，然后，在配置动作中再次勾选“msg01”。元件在显示或隐藏的动作中还可以选择各种动画效果，这里我们使用时长为“500”毫秒的【逐渐】显示效果。（图2-88）

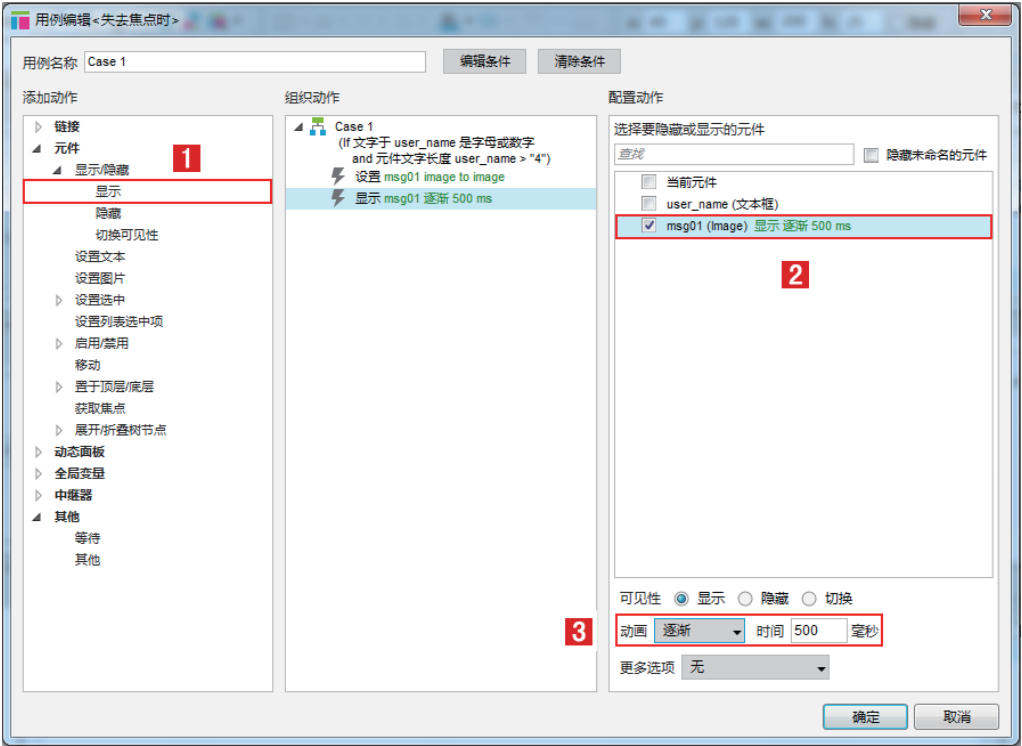


图2-88

步骤3 继续添加一个用例，为不符合条件时的用例配置动作。所有动作就像前一个用例的设置一样，不同的是【设置图片】“msg01”为错误提示的图标。（图2-89）

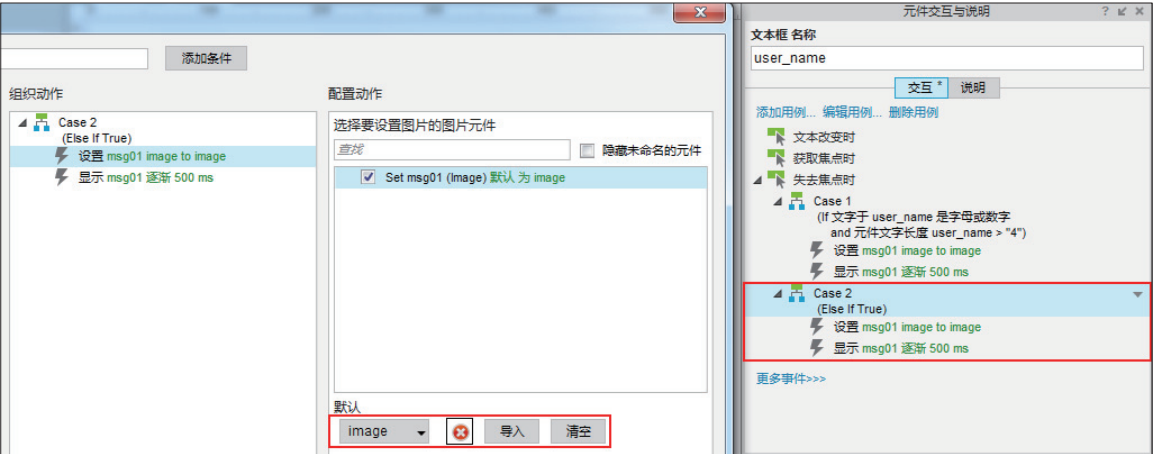


图2-89

通过这个案例说明我们在为用例设置条件时，可以添加多个条件同时验证，只要在条件编辑界面选择符合【全部】以下条件，那么只有设置的条件全部满足的情况下才能够执行相应的动作，

否则，即便有再多的条件符合，只要有一条未能符合，都会导致验证失败，无法执行该用例中的动作。

(2) 满足任何条件

假设对一个电子邮箱地址的输入框进行验证，为输入框失去焦点时，如果输入的电子邮箱地址不包含“@”符号，或者不包含“.”则显示错误提示图标，否则显示正确的提示图标。

参照上一个例子，我们在编辑区继续放入文本框和图片元件，并且分别命名为“email”和“msg02”。我们把图片元件设为【隐藏】，然后，在文本框【失去焦点时】的触发事件中添加用例进行判断。（图2-90）

- 步骤1 条件设立界面中，选择符合【任何】以下条件；
- 步骤2 添加条件判断【元件文字】“email”【不包含】【值】“@”；
- 步骤3 单击【+】继续添加，判断【元件文字】“email”【不包含】【值】“.”。

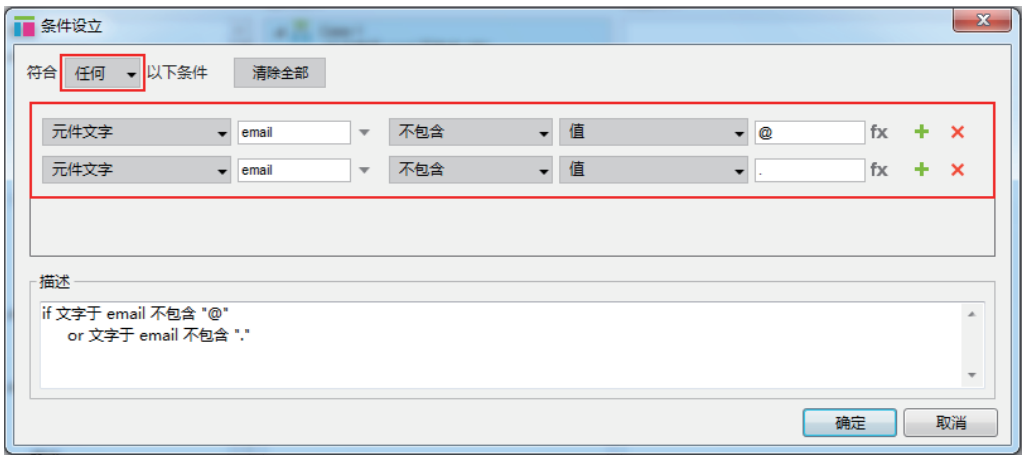


图2-90

在完成以上条件设立后，首先添加满足条件时要执行的动作，设置“msg02”为错误提示图标，并将其显示出来。然后，继续添加用例，设置未满足条件时执行的动作，设置“msg02”为正确提示图标，并显示出来。（图2-91）

通过这个案例说明，我们在条件编辑界面选择符合【任何】以下条件的选项时，只要设置的条件中有任何一条满足，就会执行相应的动作，否则，只有所有条件均不满足时，才会执行不满足条件用例中的动作。

本节任务 → 假设有文本框A、文本框B、提交按钮和文本标签。单击提交按钮时，判断文本框A、B内容是否为空，如果有文本框输入内容为空则用文本标签显示错误提示。请用“全部”与“任何”两种方法完成任务。




图2-91

2.6.3.3 元件交互的使用（案例）

元件的交互决定着原型中的绝大部分效果的呈现，所以使用好交互事件、动作，是原型设计中非常重要的部分。下面就让我们用一些很有代表性的案例，来看看怎样高效、清晰地完成交互效果。

1. 唯一选项组

 楼哥，前面你讲了性别的唯一选择，那个是单选按钮的效果。像有些网站的一些菜单项鼠标进入的时候唯一菜单项会变色，其他的恢复原色。这种效果怎么实现呢？

 可以用矩形做菜单项，然后设置每个矩形鼠标悬停时的样式就可以呀！

 啊？不对不对！我说错了，是点击菜单的时候只有一个变色的。

 这个呀？这个就不能单纯在属性里面设置交互样式了。还需要配合交互用例来实现。

菜单项唯一被选中的效果也是通过组实现的，不过，菜单项不同于单选按钮，可以在被单击时直接变成选中的状态，而且菜单项被选中时还涉及到颜色改变。这些特殊的需求，需要属性里的交互样式、组名设置以及“交互面板”中设置用例的配合来实现。

具体实现步骤如下：

（1）属性中的设置（图2-92）

- 步骤1** 同时选中3个元件，然后在属性中，为这3个元件添加统一的【选中】交互样式，用来实现按钮被选中时变色。例如：我们让3个菜单项的选中时颜色为蓝色，字体颜色为白色。
- 步骤2** 在属性中我们还要为这3个按钮添加同一个组名称，与单选按钮不同的是，这里的组名称叫“选项组名称”。
- 步骤3** 如果有需要默认被选中的元件，可以单独选中元件后，勾选属性里面【选中】的复选框（图中箭头所指位置）。

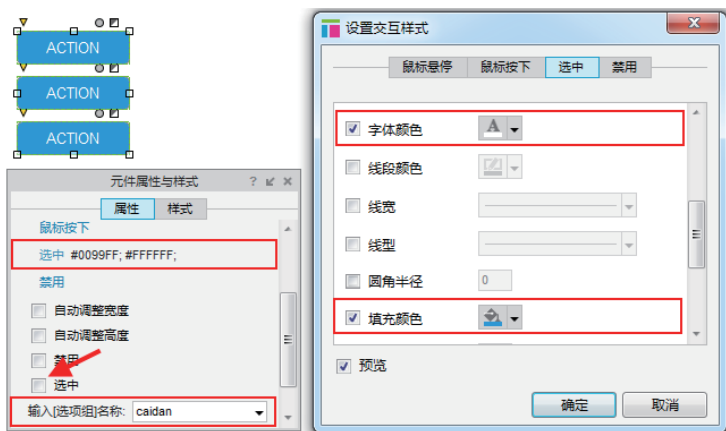


图2-92

（2）交互中的设置

- 步骤1** 选中第一个菜单项，双击【鼠标单击时】事件，打开用例编辑界面；从左侧的动作列表中找到【选中】这个动作，单击过后在右侧的【配置动作】中勾选【当前元件】；然后，在下方设置选中状态为【值】【true】；最后，单击【确定】退出用例编辑界面（图2-93）。

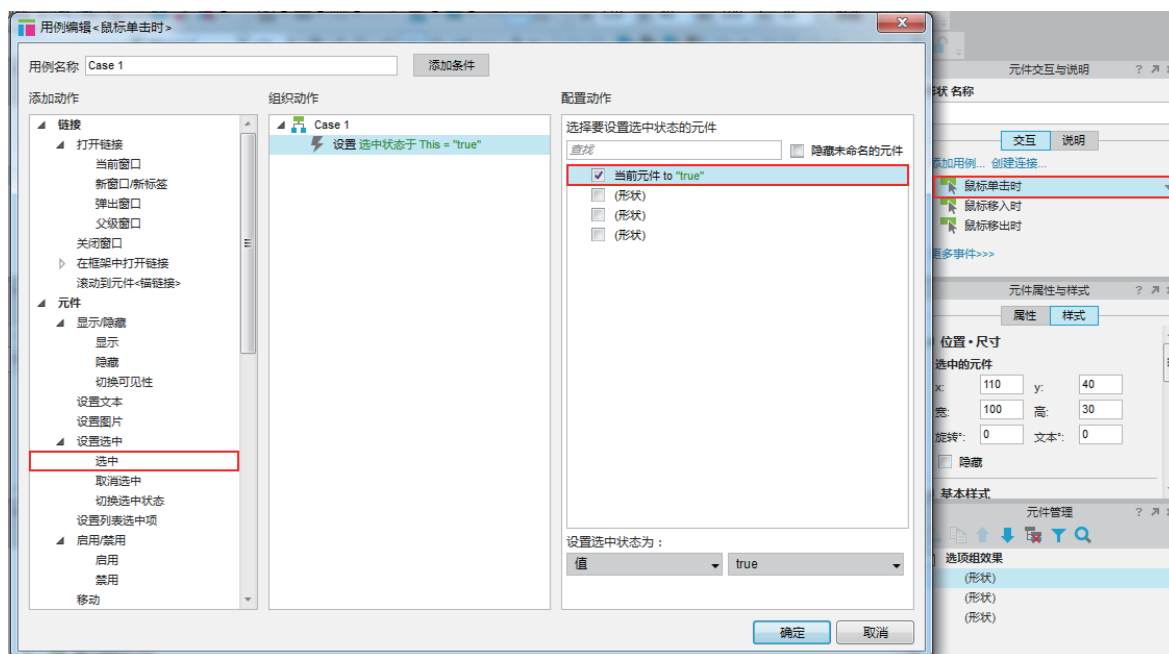


图 2-93

步骤 2 因为菜单项都是单击时选中“自己”，也就是当前元件，而刚刚在第一个菜单项上编辑的交互用例，恰恰也是单击时选中【当前元件】。所以，可以直接复制第一个菜单项上的交互用例到其他菜单项上使用。我们在第一个菜单项的触发事件【鼠标单击时】上单击右键进行复制，然后分别点中其他两个菜单项元件，用快捷键<Ctrl>+<V>键进行粘贴。（图 2-94）

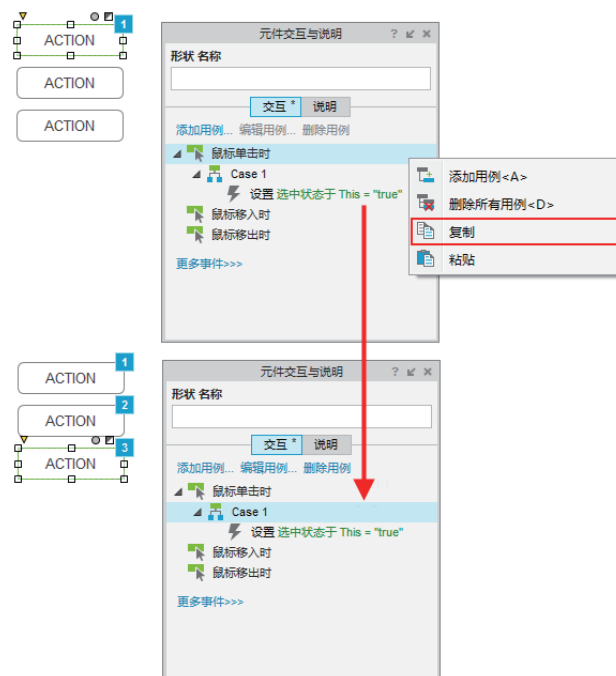


图 2-94

这时，菜单项唯一被选中的效果就制作完成了。

备注

- 1. 当前元件指当前被添加用例的元件；与之对应的还有目标元件，就是在交互中被控制的元件；
- 2. 选中状态值为【true】，表示元件被选中或者说选中状态为“真”。在Axure中通常会有3种状态true、false和toggle，true代表真或成立，false代表假或不成立，toggle代表在true和false两种状态中切换。

2. 框架页面切换

我们在前面讲到，内联框架可以将其他页面嵌入到当前的页面，其实，在动作中我们还可以控制内联框架打开其他页面。例如：我们可以制作一个案例，在单击不同的按钮时，可以在框架中打开不同的搜索引擎界面。

具体步骤如下：

- 步骤1 首先，放入一个内联框架，命名为“myframe”；
- 步骤2 设置框架宽度为整个屏幕的大小，以作者的电脑为例，把框架宽度设置为1366；高度可以随意设置，例如580；
- 步骤3 拖入几个形状按钮，并为每个按钮的【鼠标单击时】事件添加用例，在框架中打开链接【内联框架】“myframe”为【链接到url或文件】，在{超链接}中填入搜索引擎的地址。注意要以“http://”开头。（图2-95）

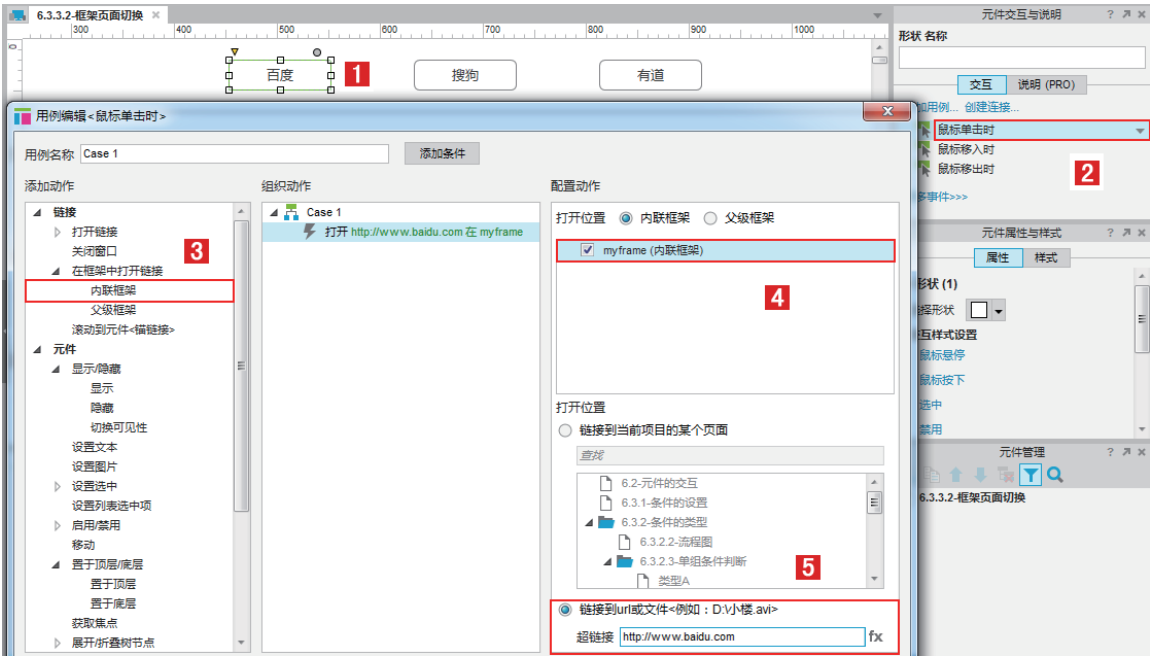


图2-95

通过以上几步，就完成了单击按钮时打开搜索引擎页面的效果。

3. 元件焦点控制

我们常常能够看到这样一个功能：一个用户名输入框，没有输入文字，单击【登录】按钮，会给出提示要求输入用户名，并且光标回到用户名输入框中闪动。想要实现这样的效果，我们需要使用让元件【获取焦点】的动作。整个例子的实现过程如下。

思路关键点。

- 触发：【登录】按钮鼠标单击时。
- 判断：文本框为空。
- 执行：显示提示；设置焦点到文本框。
- 包含元件：文本框“username”、文本标签“msg”、形状按钮“login”。

实现步骤：(图2-96)

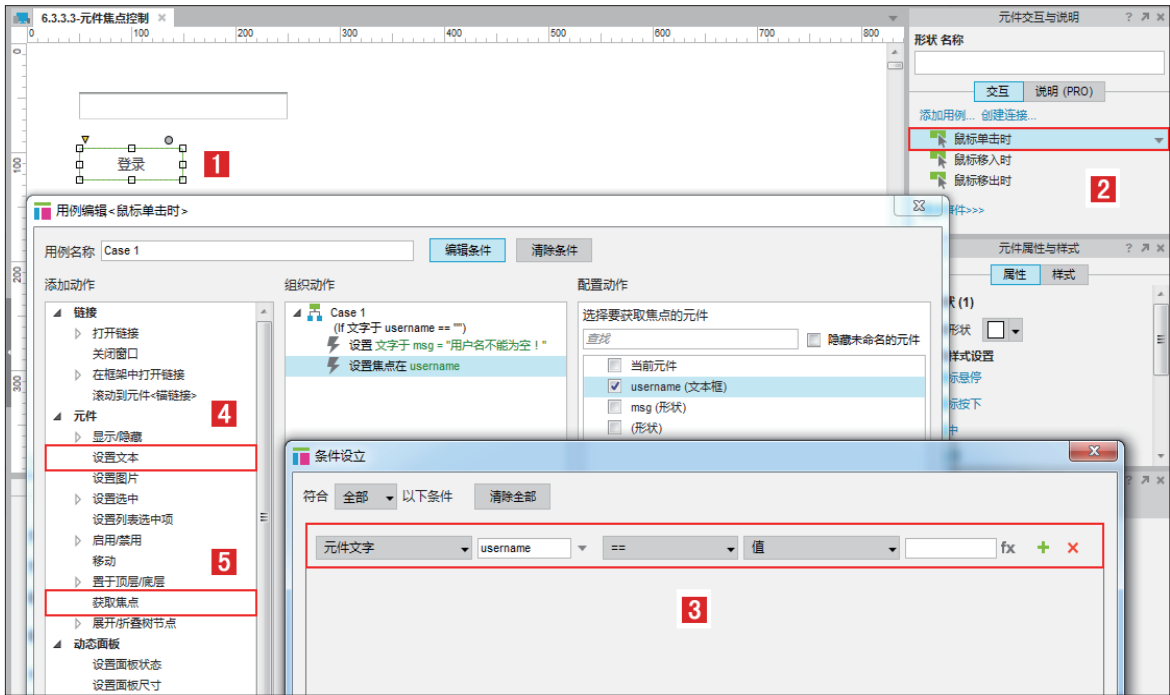


图 2-96

- 步骤 1** 点中【形状】按钮；
- 步骤 2** 双击【鼠标单击时】事件，打开用例编辑界面；
- 步骤 3** 单击【添加条件】进行条件设置，选择条件判断内容为【元件文字】“username”【==】【值】“”，单击【确定】，返回用例编辑界面；

注意

这里在值后面的文本框中不输入任何内容。

备注

“==”是关系运算符，用来判断左右条件内容相等，即“等于”。在之后还会有相关的讲解。

- 步骤 4** 设置动作1为：【设置文本】“msg”为【值】“用户名不能为空！”；
- 步骤 5** 设置动作2为：【获取焦点】到“username”；单击【确定】关闭用例编辑界面。

经过以上5个步骤，就实现了我们想要的效果。

4. 元件移动控制

有这样一个交互效果：菜单栏的某项被单击时，会有一个滑块作为背景，滑动到被单击的菜单项底层。这个交互如果没有滑动的效果，就可以参照“唯一选项组”的案例，但是有了滑动的效果就需要用到元件的【移动】动作了。

在这里我们需要接触几个概念。

坐标：指编辑区中的位置，编辑区的左上顶点为坐标的原点（x0，y0）。

x：x轴坐标，指水平坐标，编辑区的左侧边界x坐标为0。水平坐标可以为负数。

y：y轴坐标，指垂直坐标，编辑区的顶部边界x坐标为0。垂直坐标可以为负数。

元件坐标：指元件的左上顶点，在编辑区中的实际坐标。（图2-97）

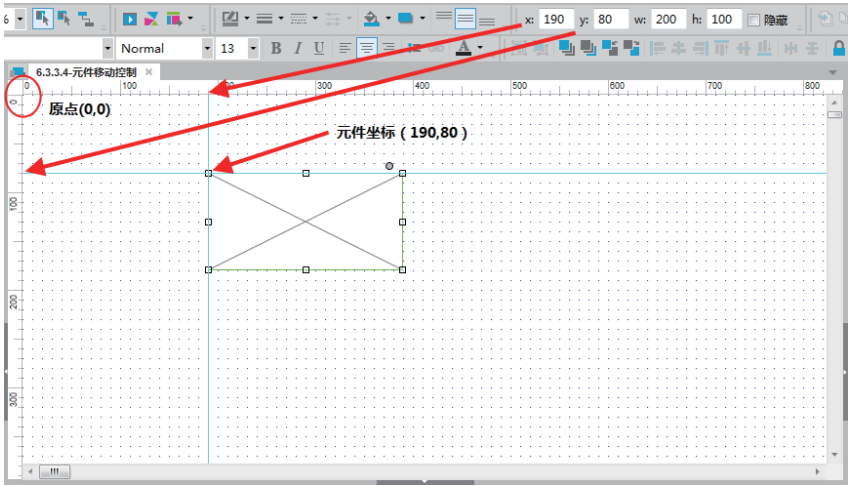


图2-97

备注

元件坐标为负数时会有部分或者全部被边界遮盖。

绝对位置：编辑区中某1点的实际坐标。例如：x58，y162。

相对距离：指编辑区中两点之间的距离。例如：A（x120，y80）相对于B（x50，y140）的相对距离为（x70，y-60）。而B相对于A的相对距离为（x-70，y60）。一般相对距离用于元件移动时是指元件移动后相对于移动前的相对距离。（图2-98）

备注

关于图中的垂直辅助线可以在左侧标尺按住鼠标左键拖至编辑区松开得到，水平辅助线则是在上方标尺上按住鼠标左键拖至编辑区松开即可得到。而背景中的小圆点，我们叫他们网格，是通过在编辑区空白处鼠标右键选择【栅格和辅助线】—【显示网格】显示出来的。并且，网格也可以在【栅格和辅助线】—【网格设置】中进行设置，例如调整网格的填充（间距）、样式、颜色等。

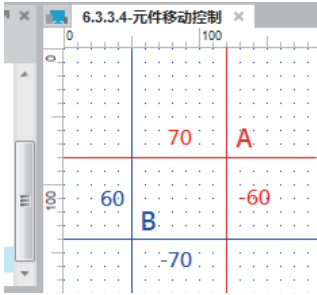


图2-98

了解了以上概念之后，我们知道每个元件在编辑区中都有自己的坐标那么我们要做的效果中的每个菜单项按钮也都有自己的坐标。我们只需要在单击按钮的时候让滑块移动到当前元件的坐标就能实现了。

实现步骤：

步骤 1 选中所有矩形菜单项，取消背景色或者设置填充色不透明度为“0”。（图2-99）

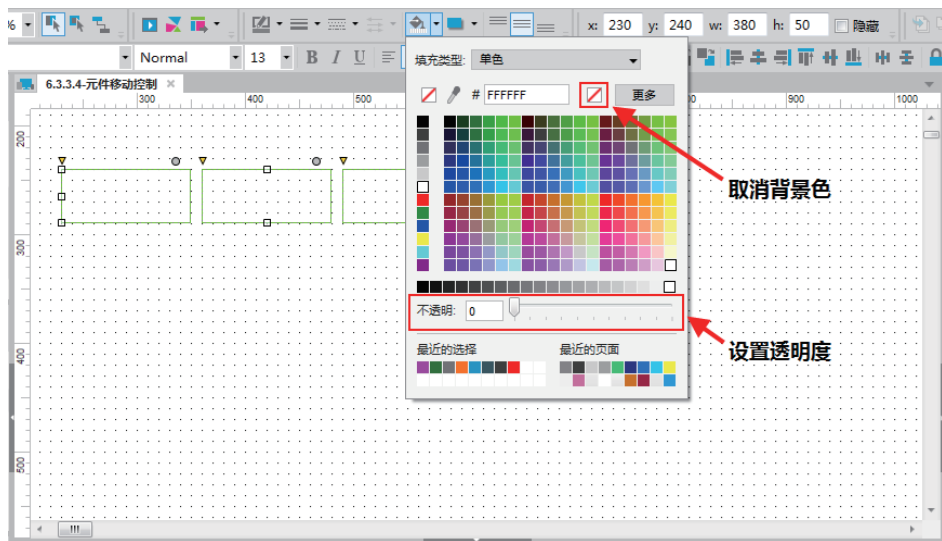


图2-99

步骤 2 将矩形菜单项添加需要的文字。然后，放入一个与矩形菜单项相同尺寸的矩形，命名为“slider”，设置其背景色为蓝色，并单击快捷功能区中的【置于底层】按钮，将其置于底层。然后，将它与首个矩形菜单项按钮重合摆放。（图2-100）



图2-100

步骤 3 为每个矩形菜单项的【鼠标单击时】事件添加用例，设置动作为移动元件“slider”到【绝对位置】x为当前矩形菜单项的x轴坐标，y为当前矩形菜单项的y轴坐标。例如：第2个菜单项的坐标为x360，y240，我们就设置移动元件“slider”到【绝对位置】x360，y240。

这里为了能有滑动效果，还需要设置一个移动时的动画【线性】“500”毫秒，是指滑块由移动开始至结束为耗时500毫秒的匀速移动。（图2-101）

以上就是元件移动的操作方法。这个效果在我们学习了函数之后还可以进行优化。

5. 列表选项交互

在本章2.6.3.3节中的2小案例中，我们通过单击不同的按钮可以打开不同的搜索引擎页面。但是，在一些浏览器的搜索工具栏中，有一个包含多个搜索引擎名称的下拉列表框，选则其中一项之后单击搜索，就可以打开被选择的搜索引擎主页。这样的效果怎么实现呢？（图2-102）

实现思路：

下拉列表包含多个选项，如果是百度就打开百度的首页，如果是搜狗就打开搜狗的首页，所以我们需要判断当前下拉列表框被选中的项是什么，然后在单击【搜索】按钮时执行在框架中打开连接的

动作，打开相应的链接（url 地址）。

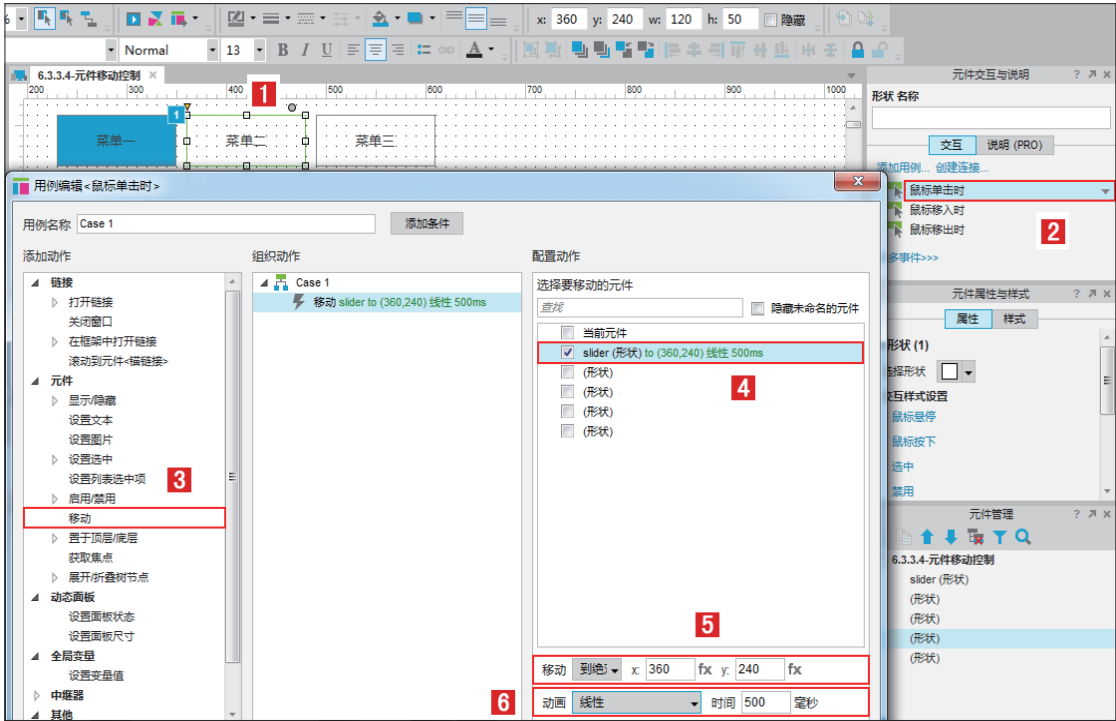


图2-101



图2-102

实现步骤：

- 步骤 1** 拖入需要的元件并命名，下拉列表框为“list”，【搜索】按钮为“search”，内联框架为“myframe”；为了页面美观，我们在内联框架的属性中将【隐藏边框】勾选上；
- 步骤 2** 设置下拉列表框，在编辑选项中添加多个搜索引擎的名称；
- 步骤 3** 双击【形状】按钮“search”的【鼠标单击时】事件打开用例编辑界面，然后添加条件判断内容为【被选项】“list”【==】【选项】【百度】，单击【确定】返回用例编辑界面；（图2-103）
- 步骤 4** 设置动作为在框架中打开链接【内联框架】“myframe”为【链接到url或文件】，在{超链接}中填入搜索引擎的地址“http://www.baidu.com”。

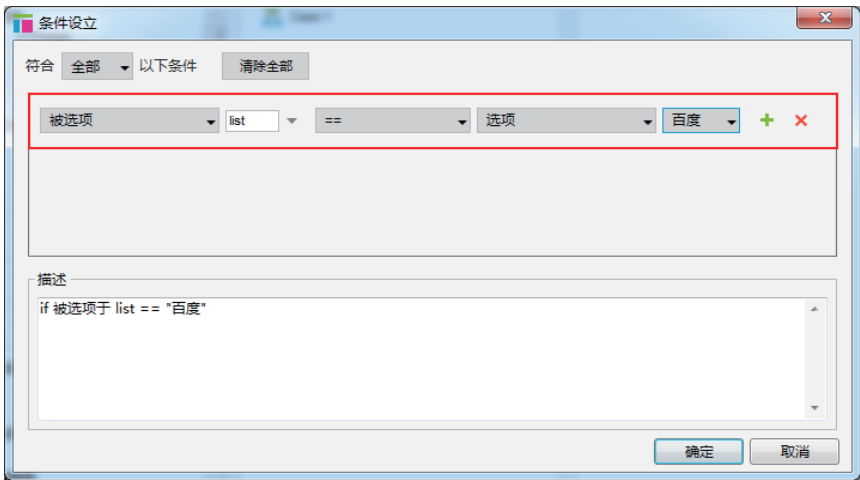


图2-103

步骤 5 参照3、4两个步骤继续添加用例，判断选项为“搜狗”时在框架中打开搜狗首页的超链接，以及选项为“有道”时，在框架中打开有道首页的超链接。（图2-104）

通过以上几个步骤，就实现了在列表选择不同的搜索引擎时，单击【搜索】按钮可以打开不同的搜索引擎的效果。

备注

这里的文本框暂时没有作用。在以后学习局部变量之后我们可以完善这个案例，实现打开指定关键词搜索结果页的效果。



图2-104

6. 焦点元件控制

有些网站或应用比较注重安全性，会在用户输入数据的时候提供虚拟键盘供用户进行输入。例如一个6位的密码框，每次单击小键盘中的数字时，就能向密码框中输入相应的数字，输入后焦点自动进入下一个密码框。（图2-105）

元件准备：

（1）需要6个文本框用来接收密码，我们将它们按顺序命名为“A”、“B”、“C”、“D”、“E”、“F”；

（2）需要用来显示文字的9个矩形，不过因为矩形单击时会获取焦点，有时会对输入产生影响，所以我们不在矩形的事件上添加用例，而是在矩形的上面覆盖一个热区，调整成与矩形相同的大小，单击的时候通过热区来触发事件。

思路分析：

① 任何一个文本框进行输入时，它的文字都会发生改变。文字框文字改变时，如果文本框的内容不是空白的，说明文本框已经输入内容，这时我们就让焦点进入下一个需要输入内容的文本框中。最后一个文本框输入内容后，无需对焦点进行控制，所以可以省略对它的判断。

② 单击热区时，需要让当前获取焦点的文本框中输入与热区相对应的矩形上的数字。

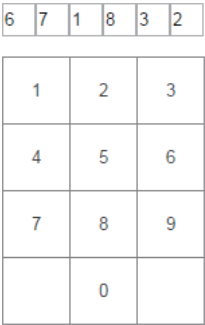


图2-105

实现步骤：

步骤 1 分别给前5个文本框【文本改变时】事件添加用例，判断【元件文字】“this”【!=】""时，执行动作【获取焦点】到下1个文本框。（图2-106）

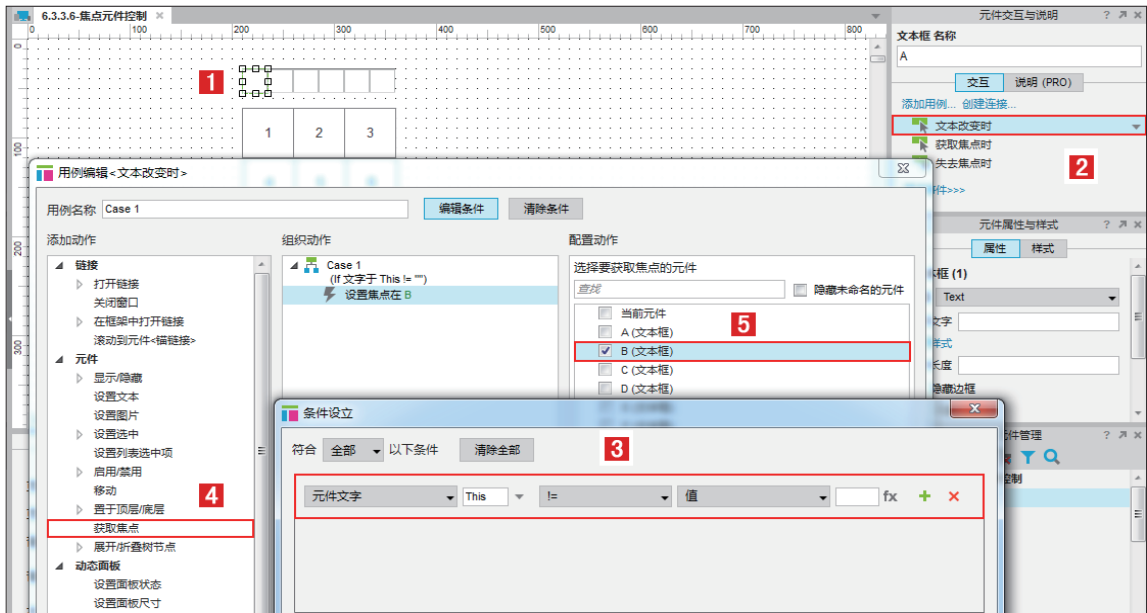


图2-106

备注 “!=”是关系运算符，用来判断左右条件内容不相等，即“不等于”。

步骤 2 我们先用一个矩形和一个热区进行设置。首先，给矩形命名“num1”，并且双击矩形添加文字“1”；然后，给热区设置【鼠标单击时】事件的用例，添加动作【设置文本】到“焦点元件”为【元件文字】“num1”。（图2-107）

备注 焦点元件不是一个固定的元件，是指当前获取焦点的元件。例如有多个文本框，光标在哪个文本框中闪烁时，则哪个文本框是焦点元件。

步骤 3 将设置好事件用例的热区覆盖到矩形“num1”上，然后同时选中这两个元件进行复制粘贴。复制粘贴后，更改元件的名称为“num2”~“num0”，矩形上的文字也对应更改为2~0。然后，将所有元件排布整齐，就完成了我们想要的效果。生成后，单击第一个文本框就可以进行文字输入了。（图2-108）

备注 元件重合时，单击第一下能选中第一层元件，再次单击选中第二层，以此类推。当选中最底层元件后，再次单击则重新选中第一层。案例中修改矩形的名称及文字时可以使用这种方法。我们在之后元件管理部分会讲到更简便的方法。

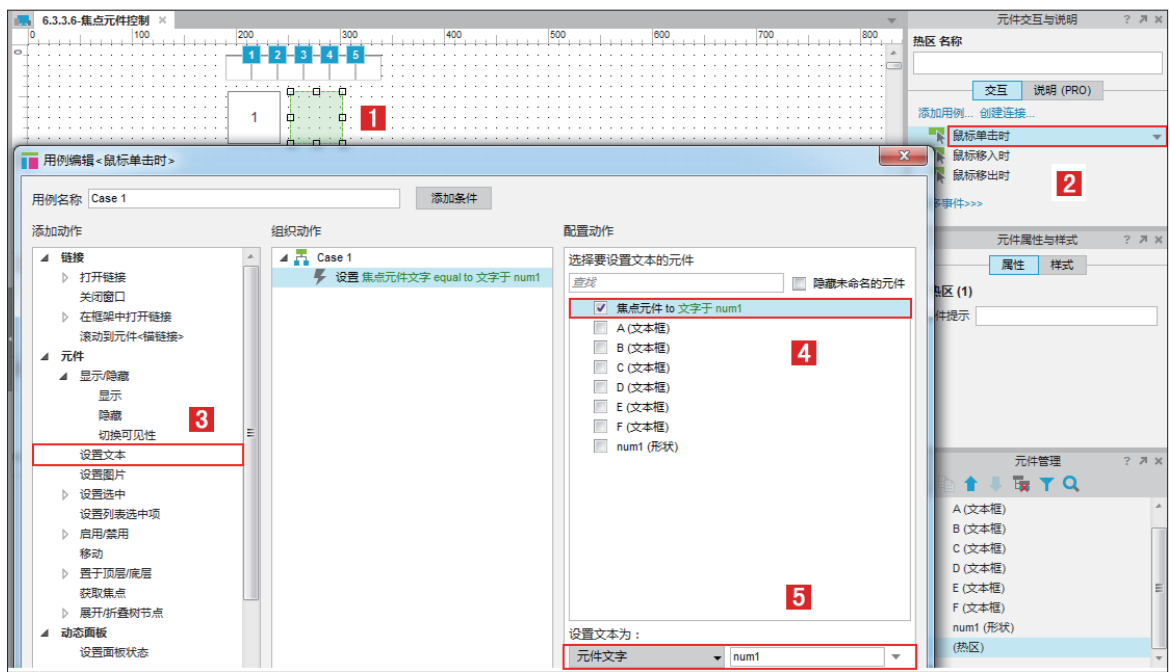


图2-107

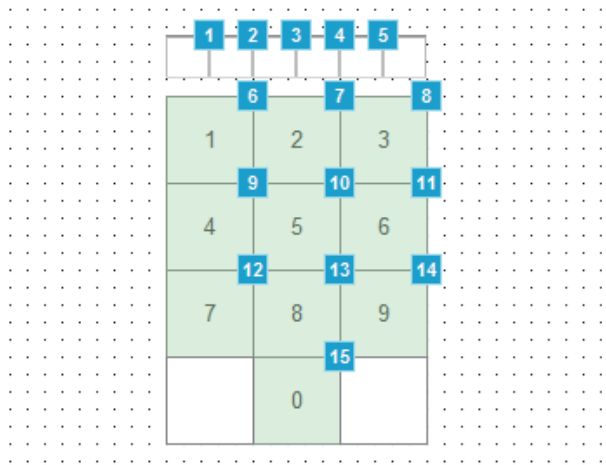


图2-108

本节任务 → 独立完成本节所有案例。

2.6.4 元件的说明



楼哥，我想做一个可以检测输入框中输入的身份证号码是否有效的判断。



苏苏，在我们原型制作的过程中，有时并不是所有的功能都要通过给元件添加交互用例、动作来实现。像你所说的这种身份证号码的验证，在Axure中实现会非常复杂，而对于开发来说，这种验证却往往是通用的，甚至都有可重用的代码。所以，我们完全没必要在原型中实现真正的验证，把精力耗费在这个功能上面非常不值得，这种情况我们只需要给这个文本框加相应的文字说明就好了。

2.6.4.1 添加元件说明

【说明】是用来给元件添加注释说明的。例如我们在绘制原型的时候有一个用户名的输入框，这个输入框会有一些验证条件的限制，例如不能有特殊字符等等。但是在原型上实现一些复杂的验证不是很方便，也很耗费时间。所以这时我们可以直接给元件添加说明，在说明中写出验证要求。（图2-109）

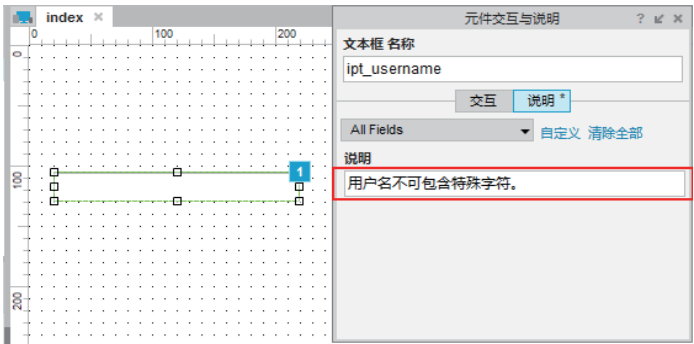


图2-109

当我们给元件添加说明后，在浏览器中查看原型时，网页中添加了说明的元件的右上角会出现说明的图标，单击图标就能看到我们在Axure中编辑时添加的说明。站点地图里面还有方便我们查看原型的功能，我们先记住前面两个的作用，第一个就是元件说明的开关，它能够控制页面中元件说明的图标是否显示。第二个是能够高亮显示页面中有交互的元件，方便浏览原型时查看页面中的交互。（图2-110）



图2-110

2.6.4.2 自定义说明字段

在前面的介绍中，我们只是给元件添加了一个说明，其实在原型制作过程中，往往一个元件需要多个说明，例如一个添加商品信息的按钮，它需要包含业务描述、需求描述、行为人、前置条件、后置条件、其他说明等等。这个时候，我们就需要添加多个说明。

单击功能面板中的【自定义】打开说明编辑界面，在【添加】的列表中选择一种类型就可以添加一个说明字段，新添加的字段就会出现在【说明】的功能模块中，我们再写入对应的说明文字就可以了。添加的说明字段一共有4种：文本、列表选项、数字和日期。一般比较常用的就是文本和列表选项，例如商品信息这个例子中的行为人是几个固定的部门，我们就可以将说明字段设置为列表选项，把这些部门预置进去，供我们选择使用，免去每次输入的烦琐。（图2-111）

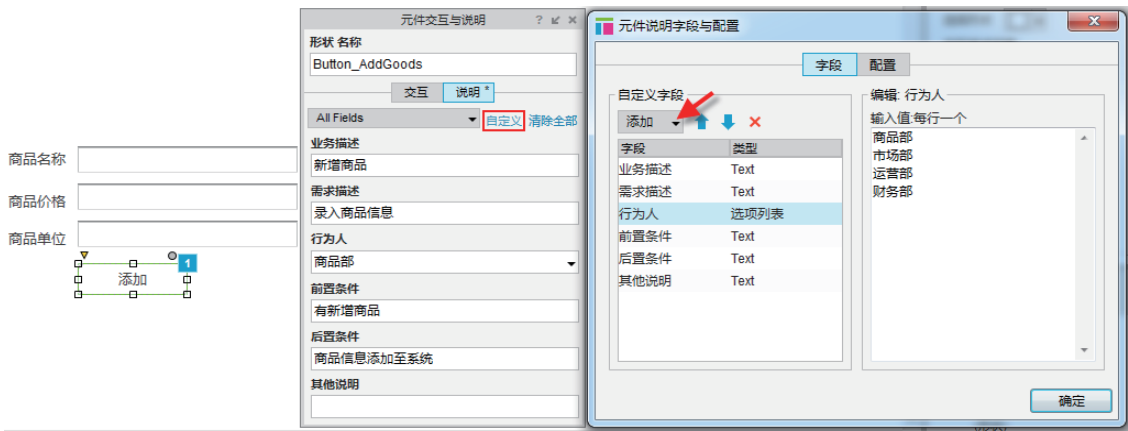


图2-111

2.7 元件管理

楼哥，你看编辑区里面这么多元件，有的还被别的元件盖住，这怎么编辑啊？

别急，苏苏！Axure给我们提供了一个元件管理的功能，就是页面右下角这里。通过元件管理功能我们能方便地对元件进行各种操作。

在界面的右下角，有一个元件管理的模块。这个模块也是很常用的，页面中所有的元素都能在这个列表中找到。（图2-112）

在图2-112中，我们能看到编辑区中所有的元件都在元件管理界面列表中。点中编辑区的元件时，列表中相对应的项也被选中。同样，在单击列表中的某一项时，编辑区中对应的元件也会被选中。所以，当页面中元件过多会互相遮挡的时候，通过元件管理能方便地选中元件。

在本章2.6.3.3节中的6小案例中，热区覆盖在矩形上，我们对矩形的重命名和编辑文字都可以在元件管理中完成。点中元件列表中的某项，再次单击该列表项元件名称则变为可编辑状态；而点中某个列表项后双击，则可以直接编辑元件上的文字。

另外，元件管理还提供过滤和搜索的功能，可以只显示满足过滤条件的元素，也可以通过输入元素名称搜索到元素。（图2-113）

楼哥，元件管理面板中有这么多元件，有的在上面有的在下面，是按什么规则进行排序的呢？

是按层级排序的。就像堆放木板一样，元件也是一层一层叠起来的，在元件管理模块中，层级越低的元件越靠前显示，底层的元件排在第一位，顶层的元件排在末尾。

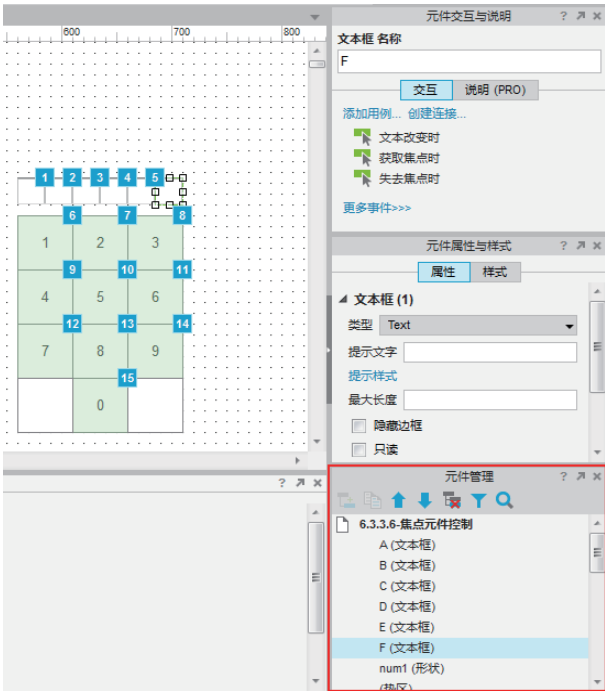


图2-112

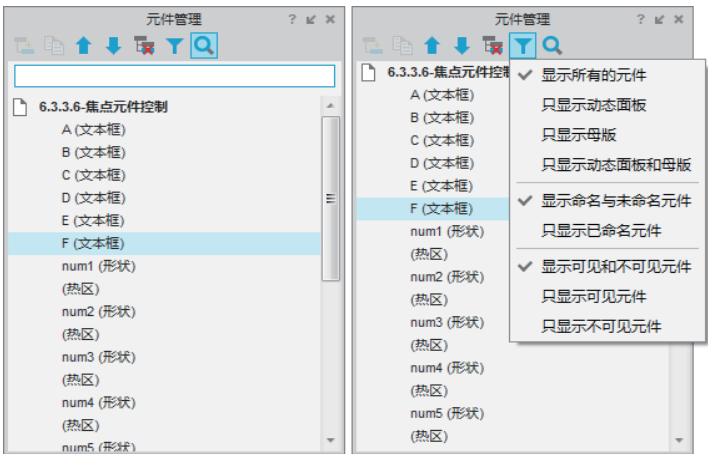


图2-113

苏：那怎样调整元件的层级呢？

峰：真笨！你忘啦？在快捷功能区中有调整层级的按钮啊！不过，在元件管理中也可以调整。我们可以选中元件列表中的某项，然后单击上下箭头的功能按钮调整元件的层级。也可以直接上下拖动元件列表中的某项来调整它的层级。

最后，有一些特殊元素，例如动态面板和母版也会显示在这个列表中，这两个特殊元素还能够仅在编辑区中进行隐藏，而不影响预览或者生成后在页面中的正常显示。（图2-114）

如图2-114所示，单击元件列表项后面的复选框，可以切换该项在编辑区的显示与隐藏。

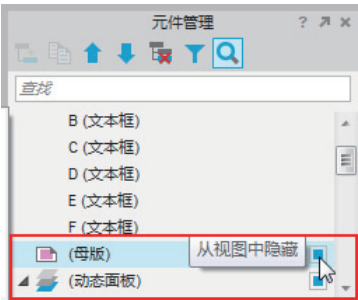


图2-114

苏：动态面板和母版是什么？

峰：……后面讲，后面讲。

2.8 页面设置

前面讲了很多内容都是和元件有关的，其实关于元件的使用还有很多内容，我们先保留一下，看看在软件的主界面中还有哪些功能。

苏：楼哥，生成的页面怎么居中啊，我生成后都在左半边啊？还有，页面上怎么加背景图啊？放一张图片拉大吗？

峰：这些设置在软件的中下方。

在主界面的中下部是页面设置模块，用来给页面设置样式、添加交互、撰写说明。

2.8.1 页面样式

首先，我们来看页面样式。

第1项：选择页面样式。

我们可以单击页面样式编辑的按钮，设置多种页面样式的方案，保存在页面样式列表中，然后在页面样式列表中选择使用。（图2-115）

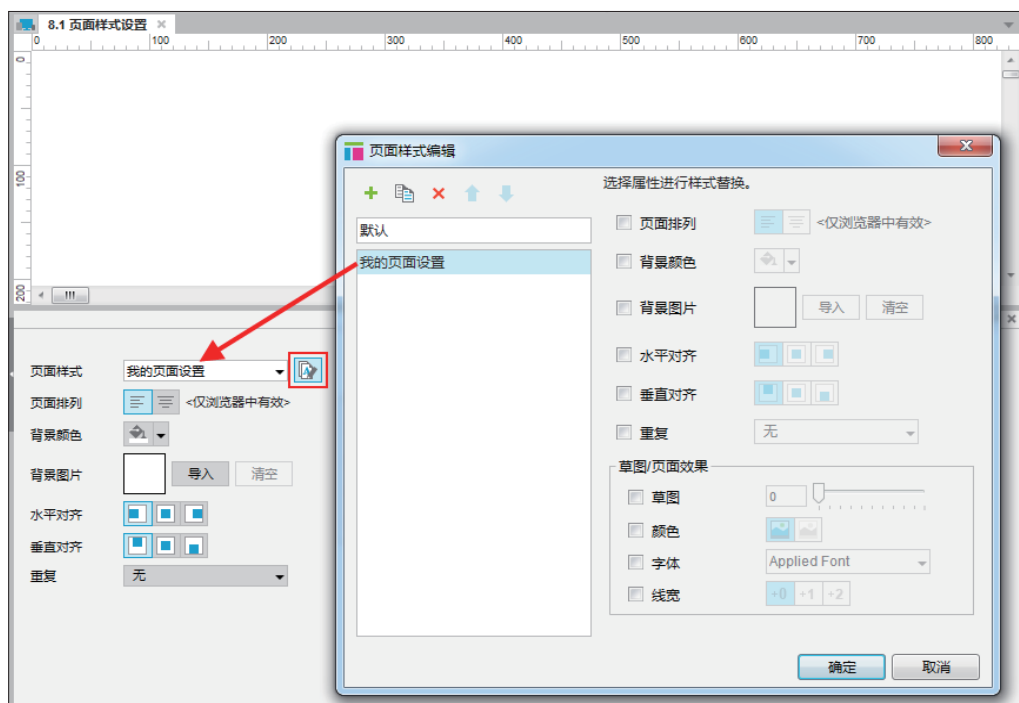


图2-115

第2项：页面排列。

一共有两个选项，默认为居左显示，可以更改为另一种居中显示。一般在设计Web原型时，我们都会选择居中显示。不过，这两个选项的效果，只有在页面预览或者生成后进行查看时才能看到相应的效果。

第3项：背景颜色。

我们可以像给元件设置填充颜色一样，给整个页面添加背景色。

第4、5、6、7项：背景图片及对齐、重复的设置。

通过单击【导入】导入本地的图像文件，我们可以进行水平和垂直的对齐设置。如果图像需要重复，还可以进行重复的设置。重复效果里面有两个需要注意的选项就是填充和适应。

填充是指根据图片的原始比例（宽：高）对应浏览器窗口的当前比例，当宽高比变大时，图片宽度与窗口宽度保持一致，而高度按原始比例进行缩放。当宽高比缩小时，图片高度与窗口高度保持一致，而宽度按原始比例进行缩放。例如：图片原始比例16：9，当浏览器尺寸为1200px×900px时，宽高比变小，这时背景图片尺寸为1600px×900px。当浏览器尺寸为1200px×600px时宽高比变大，这时背景图片尺寸为1200px×675px。

适应与填充相反，是指根据图片的原始比例（宽：高）对应浏览器窗口的当前比例，当宽高比变大时，图片高度与窗口高度保持一致，而宽度按原始比例进行缩放。当宽高比缩小时，图片宽度与窗口宽度保持一致，而高度按原始比例进行缩放。例如：图片原始比例16：9，当浏览器尺寸为1200px×900px时，宽高比变小，这时背景图片尺寸为1200px×675px。当浏览器尺寸为1200px×600px

时宽高比变大，这时背景图片尺寸为1067px × 600px。

以上是页面设置中左侧的部分功能，下面我们来看右侧还有什么？

第1项：草图效果。

拖动标尺能够让页面上的一些元件变成手绘草图效果，标尺越向右侧拖动草图效果则越明显。

第2项：页面颜色。

能够设置页面的颜色效果。共有两个选项，第1个是彩色效果，第2个是黑白效果。

第3项：字体系列。

能够统一设置页面中的字体系列，例如：宋体或微软雅黑。

第4项：线段宽度。

能够统一增加页面中元件边框以及线段的宽度。

2.8.2 页面说明

页面设置面板中最左侧一项是页面说明。页面说明像元件说明一样，可以为当前页面添加注释说明，以便他人了解页面内容。页面说明直接在下方的文本框中键入内容。如果需要有多个说明，可以单击【自定义说明字段】，完成添加后在下拉框中选择不同的字段名称，即可添加不同的说明内容。（图2-116）

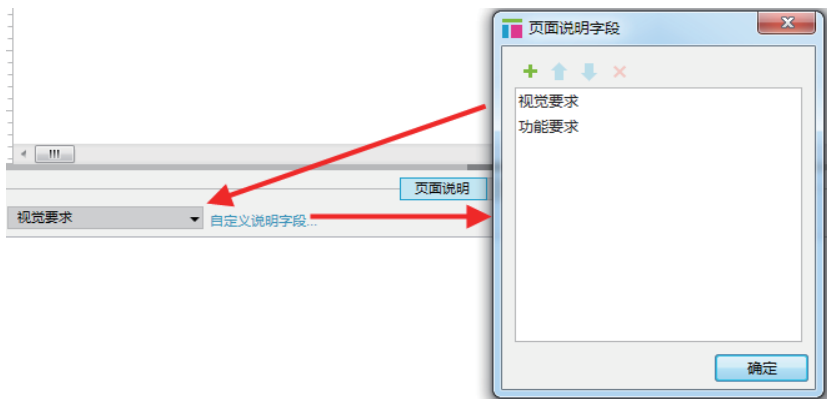


图2-116

2.8.3 页面交互

页面交互中包含页面的各种触发事件，可以为页面的触发事件添加用例，来执行指定的动作。

例如，在本章2.6.3.3节中的6小案例“焦点元件控制”中，如果需要在页面打开时可以直接向第一个文本框中输入文字，就能通过页面的触发事件来实现。我们只需要在【页面加载时】事件中添加一个用例，设置动作为【获取焦点】在文本框“A”上，即可实现需要的效果。（图2-117）

注意

页面加载时获取焦点的这个效果，在IE浏览器中会因为安全性限制而失效，所以不要使用IE浏览器查看该效果。



图2-117

2.9 母版管理



苏苏，你知道什么是母版吗？



嗯，以前用过PPT的软件，知道一些。



那你说说，在PPT中母版的作用是什么？



嗯……，PPT中的母版就是做完之后能够显示在全部或者指定的幻灯片中。



是的，在Axure中的母版也是能够在做完后添加到全部或者指定的页面中。一般来说，我们会把多个页面中都相同的部分抽出来做成母版，达到一次制作多处使用的目的，这样可以节省很多重复的工作。例如一个页面的顶部导航，或者底部的公司信息、版权信息等等，都可以做成母版，然后添加到各个页面中使用。而且，当我们对母版进行编辑时，页面中的内容也会随之同步改变。

2.9.1 母版的创建/编辑/删除

新建：单击面板中的【新建】按钮或者单击面板空白处按快捷键<Ctrl>+<Enter>完成母版的创建。创建后可以对母版名称进行编辑，双击母版名称则进入母版的编辑界面。

编辑：编辑区中可以像组织页面内容一样，拖入元件、添加交互等组成模板的内容。

删除：如果需要删除一个母版，需要先将母版从所有关联的页面中移除，才可以删除。如果被删除的母版有下级母版，则该母版被删除时，下级母版也被同时删除。

2.9.2 母版在页面中的使用

2.9.2.1 添加到页面中

在母版上单击鼠标右键，打开右键菜单，其中有一项是【添加到页面中】，选择该项后，则可以将母版指定放到哪些页面中使用。指定页面的时候软件提供了一些快捷按钮给我们使用。（图2-118）

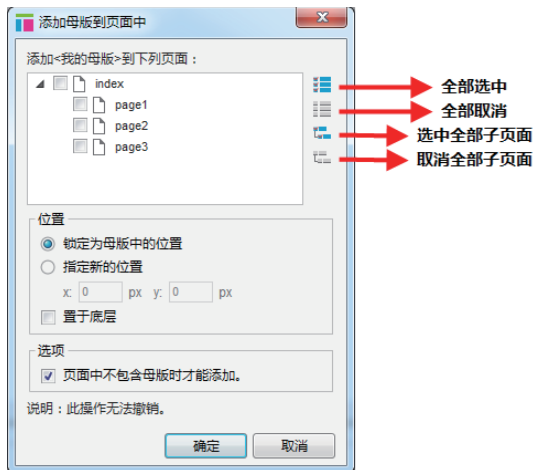


图2-118

一般情况下，我们通过这个操作将母版添加到页面时，默认是在页面相同的位置上添加母版的内容，即选项中的【锁定为母版中的位置】。我们也可以通过【指定新的位置】来添加母版中的内容到页面指定的位置。

最后，在我们向页面添加母版时，如果要求一个母版只能向同一个页面中添加一次，可以选择【页面中不包含母版时才能添加。】的选项加以限制。

2.9.2.2 从页面中移除

如果页面上不再需要母版，或者在向页面添加母版时添加错了，我们可以通过在母版上单击鼠标右键，打开右键菜单，选择【从页面中移除】，在打开的页面中选择要移除的页面后，单击【确定】即可完成移除。

在移除的页面中也提供了快捷选择的按钮，方便我们快速选取页面，按钮功能参考（图2-118）。

2.9.3 母版的拖放行为



楼哥，像网站底部信息这种内容可以做成母版使用吗？



可以呀！



但是，我刚刚试了一下，因为页面长度都不一样，底部信息的母版添加到页面时只能一个地添加，还要指定添加后的位置，好麻烦！



这个你得用拖动添加的方式。

母版除了使用前面的方法添加到页面中，还可以通过点住母版名称拖放到编辑区的方式添加到某

个页面中。而且为了满足不同情况的需要，软件提供了3种拖放方式。拖放方式可以通过在母版名称上单击鼠标右键，打开右键菜单，在【拖放行为】的二级菜单中进行选择。（图2-119）

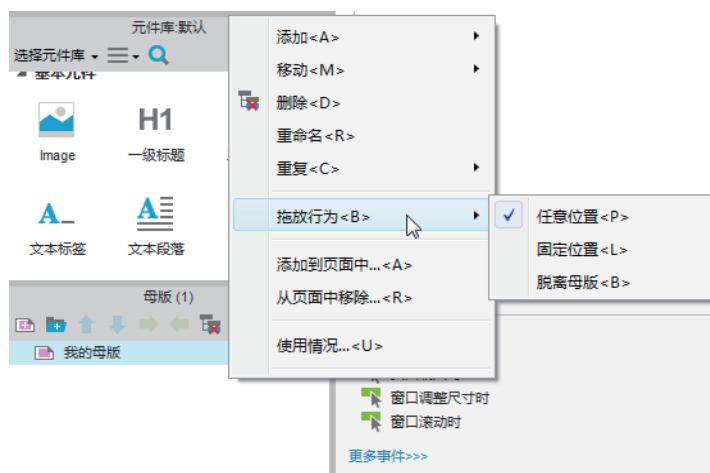


图2-119

任意位置：允许将母版内容拖至编辑区中的任意位置。

固定位置：母版内容拖入到编辑区并松开鼠标后，母版内容会自动放置到页面中的固定位置上，这个固定位置与母版页面中该内容所在的位置一致。

脱离母版：允许将母版内容拖入编辑区中的任意位置上摆放。但是，当拖放结束后，这些内容与母版脱离联系，变成普通元件存在于页面中。母版被编辑并发生改变时，这些内容不会同步发生变化。

2.9.4 母版的使用情况

当页面比较繁杂时，如果想知道一个母版都在哪些页面中被使用，逐一打开页面查看是很不方便的，所以软件提供了查看母版使用情况的功能。我们只需要在某个母版名称上单击鼠标右键，选择【使用情况】，即可打开使用该母版的页面名称列表，让我们一目了然地看到这个母版都被添加到了哪些页面中。

2.9.5 自定义触发事件

母版用来实现内容的重用、同步管理，来提高我们制作原型的效率。但是有时母版中的某些内容，我们可能需要在不同的页面中有不同的交互效果。

例如，同一个【登录】按钮，在一般的页面弹出一个注册模块，而在登录页面单击这个按钮，是把焦点设置到用户名输入框中。这样就产生了一个需求，母版中的同一个按钮需要在不同页面被单击时执行不同的动作。但是，在母版中直接给【鼠标单击时】事件添加用例动作的话，所有页面都会执行母版中设置的动作，满足不了我们的需求。

那么，怎么解决这个问题呢？

举个例子，就像我们家里用电，如果只有一个供电总线，那么开关供电总线的电闸会影响家中所有的供电。但是，我们希望每个房间的供电都能单独控制，所以，需要安装一个设备叫“配电箱”，通

过配电箱就可以对每个房间的供电进行单独管理，不会互相影响。

在 Axure 中，母版中的每个事件都像供电总线，影响所有使用了这个母版页面。如果在添加了母版的页面上给母版中的某个事件单独添加用例动作，这就需要我们给母版中的这个事件接上一个“配电箱”，这个“配电箱”的名字叫“自定义触发事件”。

就如同给家里安装配电箱一样，我们需要设置两个步骤，才能拥有和使用“配电箱”。

① 创建一个自定义的触发事件；（购买一个配电箱）

创建自定义触发事件需要在导航菜单【布置】列表最后一项【管理母版触发事件】中进行创建。例如：单击【+】添加一个事件名称，并命名为“MyWidgetClick”。（图2-120）

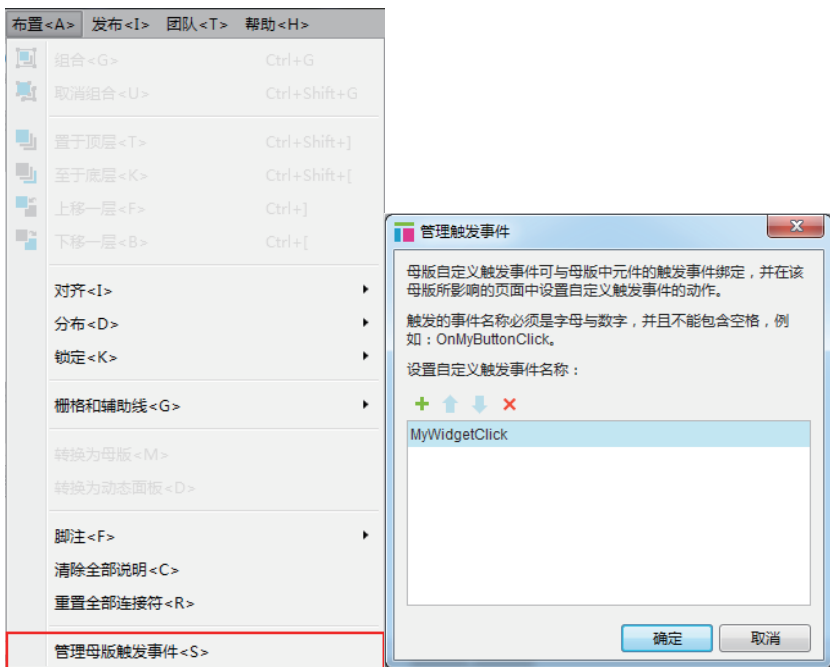


图2-120

注意

只有当编辑区为母版的编辑区时，才能在导航菜单中打开【管理母版触发事件】的选项。否则，该选项为灰色的禁用状态。

② 将母版中的某个事件与自定义触发事件绑定。（将供电总线接入配电箱）

假设，母版中的一个按钮“ACTION”需要将它的【鼠标单击时】事件在不同的页面触发时有不同的用例动作，我们就在编辑区中点选这个按钮，然后双击它的【鼠标单击时】事件，在动作中选择【触发事件】，将它与我们创建的自定义触发事件“MyWidgetClick”进行绑定。（图2-121）

完成以上两步，在添加了母版的页面中选中母版，则会在交互的功能界面中出现这个母版中绑定的所有自定义触发事件，就像从配电箱中接出的分线。例如在图2-121中设置的自定义触发事件，就出现在图2-122的触发事件列表中。为这个自定义触发事件添加的用例动作就是为按钮“ACTION”【鼠标单击时】事件添加了用例动作。（图2-122）

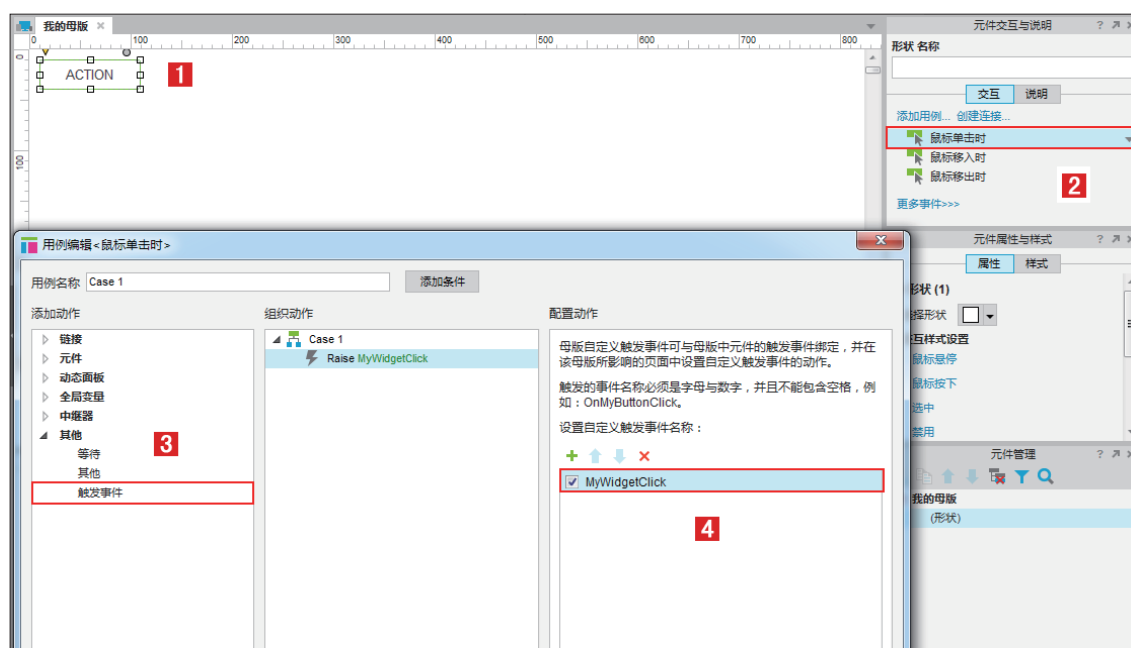


图 2-121

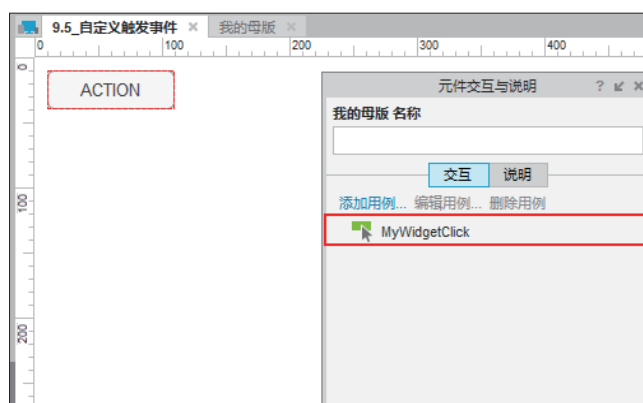
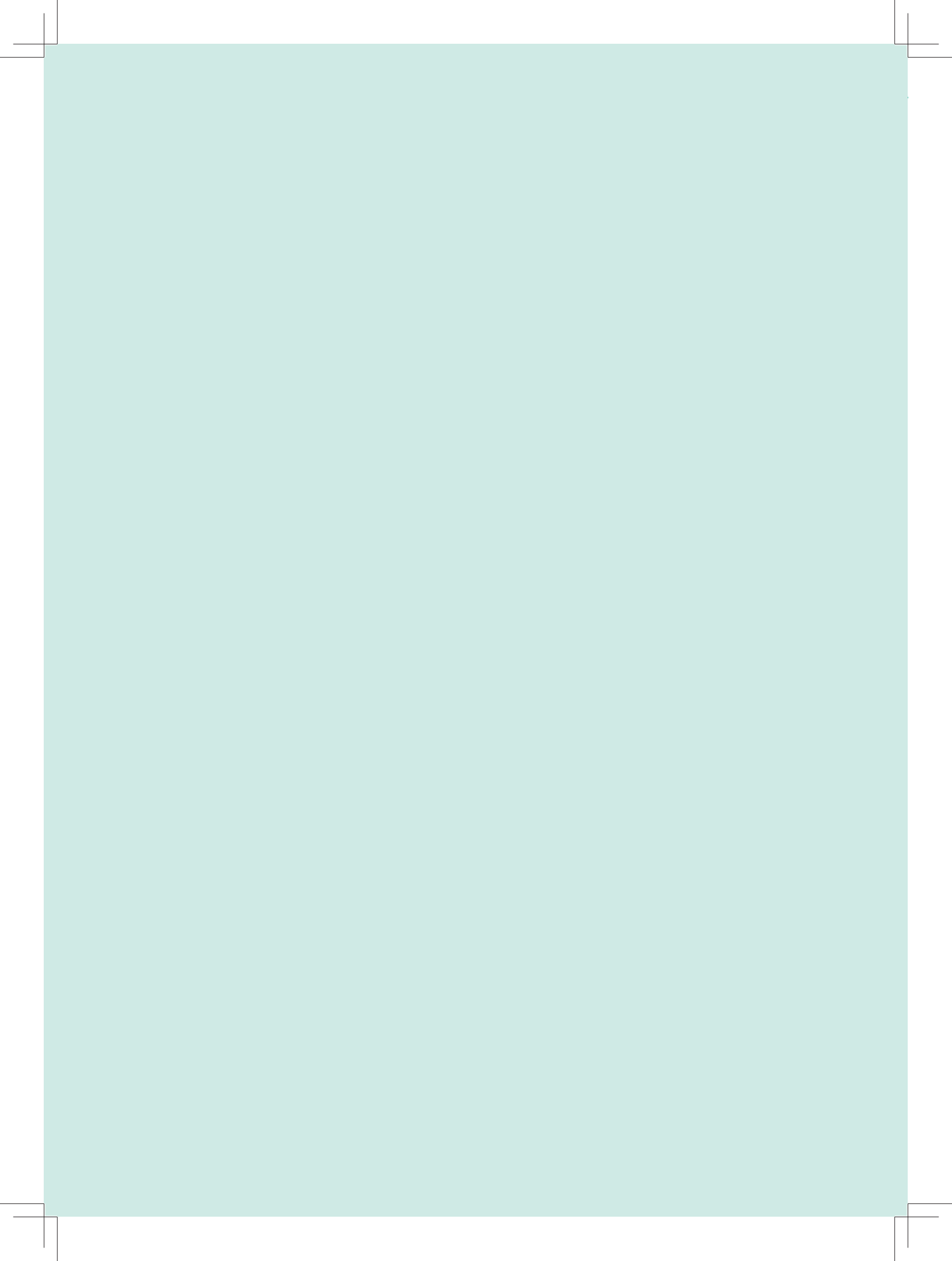


图 2-122




第 2 篇

深入与精通

- 第3章 变量与表达式
- 第4章 动态面板
- 第5章 函数
- 第6章 中继器
- 第7章 自适应视图与AxureShare
- 第8章 自定义元件库

第3章 变量与表达式



- 
1. 变量
 2. 表达式

▶▶3.1 变量



楼哥，我这两天在一些 Axure 群里经常看到有人提到变量，变量到底是什么呢？



变量，只看名称可能觉得非常难懂。其实，如果知道它的作用，抛开它的名称，就会变得很容易！就如同你不知道“谷氨酸钠”是什么一样？但一说“味精”你就清楚了。变量是一个数据的存储容器，有用的数据可以存储在里面，需要的时候去使用。它只有两个操作：存入和读取。

Axure 中的变量有两种，一种是全局变量，另外一种是局部变量。具体的使用方法，我们下面做详细的介绍。

3.1.1 全局变量

全局变量是变量中的一种类型。

我们知道，在乘坐交通工具时可以办一张通用的公交卡，然后可以在所有的公交车、地铁线路中使用，它的使用范围就是整个公共交通系统。就像公交卡有它的使用范围，变量同样有使用范围。公共交通系统就像我们当前的整个原型项目，地铁、公交就像是每个页面。全局变量就像通用公交卡一样能够在整个原型的所有页面中使用。

举例：在 A 页面中输入“小楼”并且选择性别“男”，单击【跳转】按钮，跳转到 B 页面显示“欢迎你，小楼帅哥！”，而输入“苏苏”选择性别“女”的时候，跳转到 B 页面时显示“欢迎你，苏苏美女！”

首先，我们新建 A 页面“pageA”放入元件并命名文本框为“username”，性别选项我们使用单选按钮，男性为“male”，女性为“female”，并在属性中设置为一个单选按钮组，组名为“sex”。跳转按钮使用形状按钮，可以不加命名。

然后我们来分析一下，如果需要在 B 页面获取到 A 页面中的数据，必须通过全局变量进行传递。所以，我们在 A 页面的跳转动作之前，必须要将页面上的数据保存到变量中，否则就丢失了。但是，变量在哪里呢？

我们需要先创建全局变量才能够使用。在这个案例中，我们需要两个全局变量，一个用来存放输入的用户名，另外一个用来存放输入的性别。

下面我们就创建全局变量。

打开导航菜单【项目】的列表，选择倒数第二项【全局变量】，打开全局变量的编辑页面，这里默认创建了一个名为“OnLoadVariable”的全局变量。我们可以删除它，也可以重命名。新建全局变量只需要单击界面中的【+】就可以了。

在这里，我们鼠标右键单击默认的全局变量名称并为其重命名为“uname”，然后，新建一个全局变量，命名为“usex”。大家注意在这里，还可以为全局变量设置默认值。（图 3-1）

创建完变量后单击【确定】关闭全局变量管理界面。

我们知道要保存的数据中，性别的数据是通过单选按钮选择的，所以，当任何一个单选按钮被选中时，全局变量“usex”保存的数据都应该随之变化。那么我们就给每个单选按钮添加事件用例。

但是，使用哪个触发事件呢？我们来想象一下，单选按钮被单击时一定会改变为被选中状态，所以，当单选按钮被单击时，我们让全局变量中存入相应的数据就可以了。

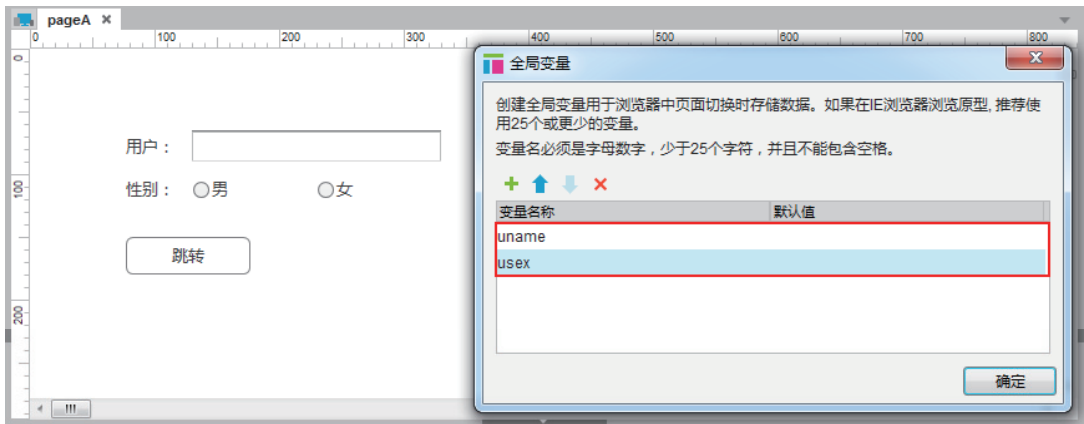


图3-1

单选按钮的【鼠标单击时】事件，没有直接在触发事件列表中显示，而是在列表下方的【更多事件】中选择。

我们先为单选按钮“男”添加用例动作，选择了触发事件之后，在动作列表中找到【设置变量值】，并勾选记录性别的全局变量“usex”，为其设置【值】为“帅哥”。（图3-2）

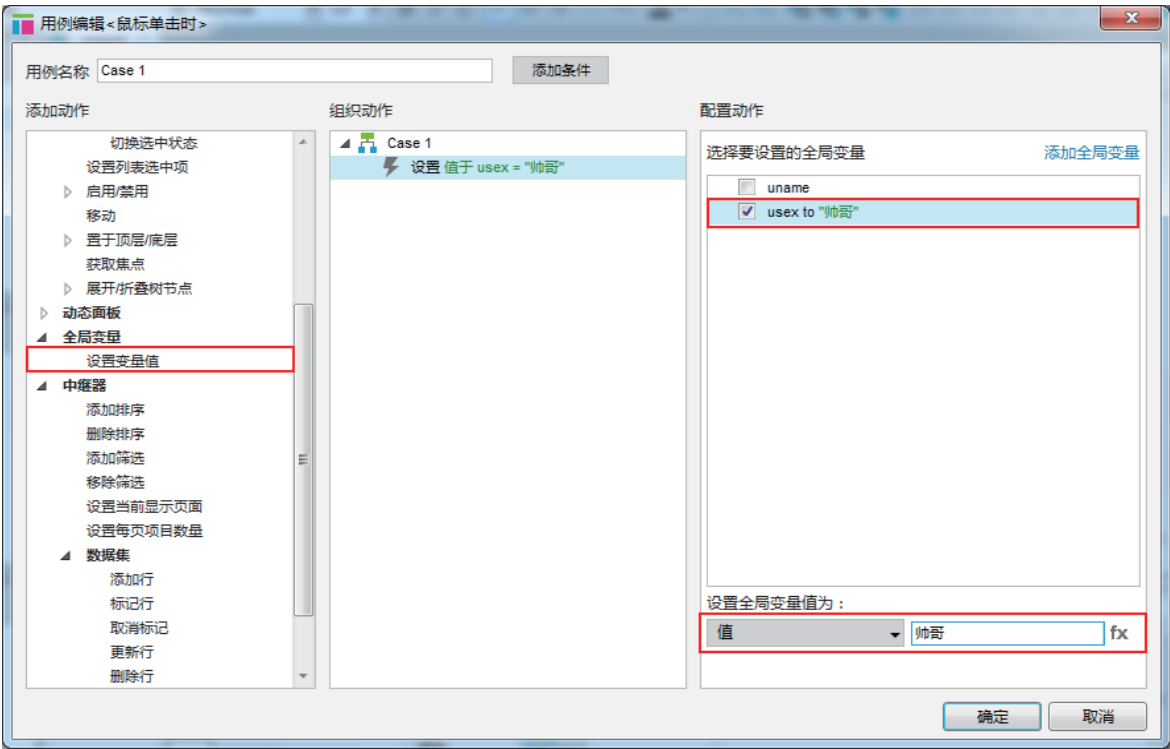


图3-2

参考上面的操作，我们也为单选按钮“女”添加用例动作，设置全局变量“usex”的值为“美女”。完成了单选按钮的事件设置，接下来就是跳转按钮了。

跳转按钮被鼠标单击的时候，除了跳转页面，还要在页面跳转之前将文本框中输入的用户名保存到全局变量“uname”中。这里大家千万要注意在用例编辑页面【组织动作】的列表中，一定要把跳转

页面的动作放在最后。因为在 Axure 中，程序的执行顺序是由上至下的，当前页面的动作不能到跳转后的页面去执行，页面跳转的动作会直接影响到后面的动作，使他们失效。

知道了要做什么，我们在跳转按钮的【鼠标单击时】事件上添加相应的用例动作就可以了。

首先，设置全局变量“uname”的值为【元件文字】“username”。（图3-3）

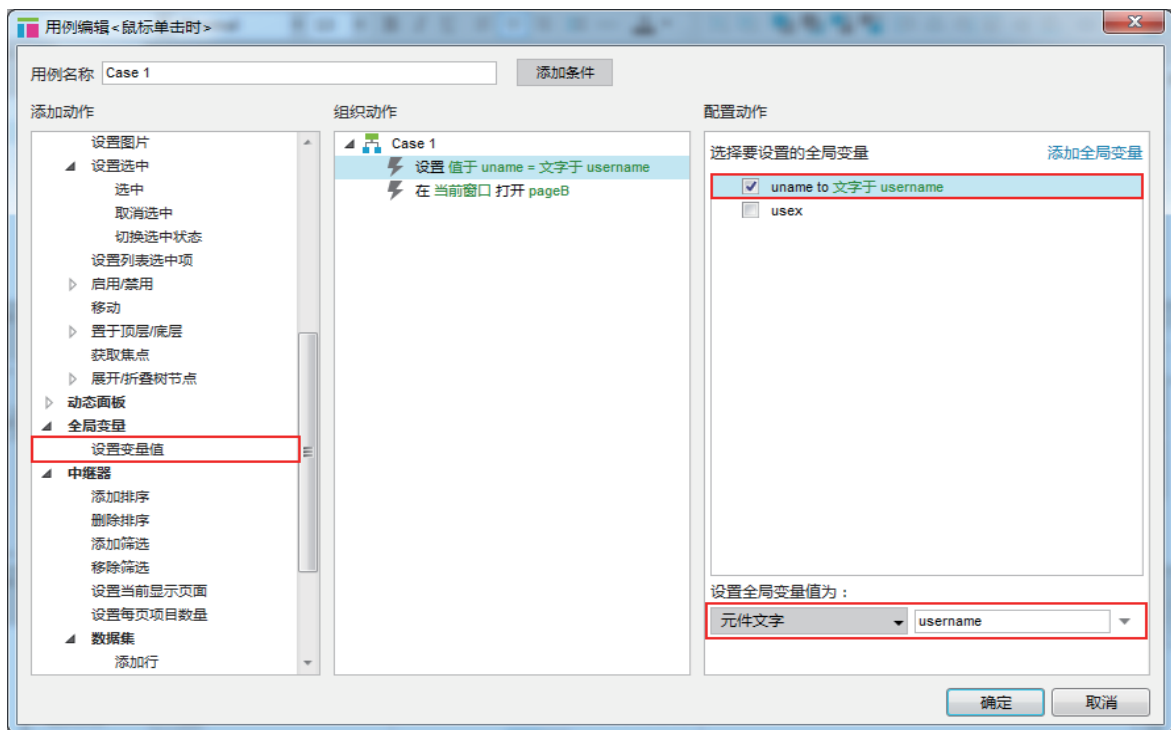


图3-3

然后，在当前窗口打开【pageB】。

通过以上的操作，我们就完成了A页面中的操作，下面我们来设置B页面“pageB”。

B页面要在页面打开后显示一句欢迎词：“欢迎你，XXYY！”。XX代表用户名，YY代表根据性别的不同称谓。

根据上面的描述，我们能看出需要一个元件来显示欢迎词，这个元件我们可以用一个没有文字的文本标签，命名为“welcome”。

显示欢迎词是在页面打开的时候完成的。所以，我们需要在【页面交互】的触发事件【页面载入时】中添加用例动作。设置元件“welcome”的文本为【值】，对于值的内容，我们先键入“欢迎你！”。（图3-4）

在前面我们已经提到，用户名和性别对应的称谓都是通过全局变量传递的，那么到了设置元件“welcome”的文本时，怎么把变量值嵌入到这句欢迎词当中呢？很简单，一共有3个步骤（图3-5）：

- 步骤1** 单击【值】的输入框后面的【fx】，打开编辑文本的界面；
- 步骤2** 单击【插入变量或函数…】，打开变量与函数列表；
- 步骤3** 单击列表中全局变量的名称，将其插入到写好的文字当中去。当然，按照[[变量名]]格式手工输入也是可以的。

完成这两个步骤后，保存并返回到主界面，就完成我们想要的效果了。

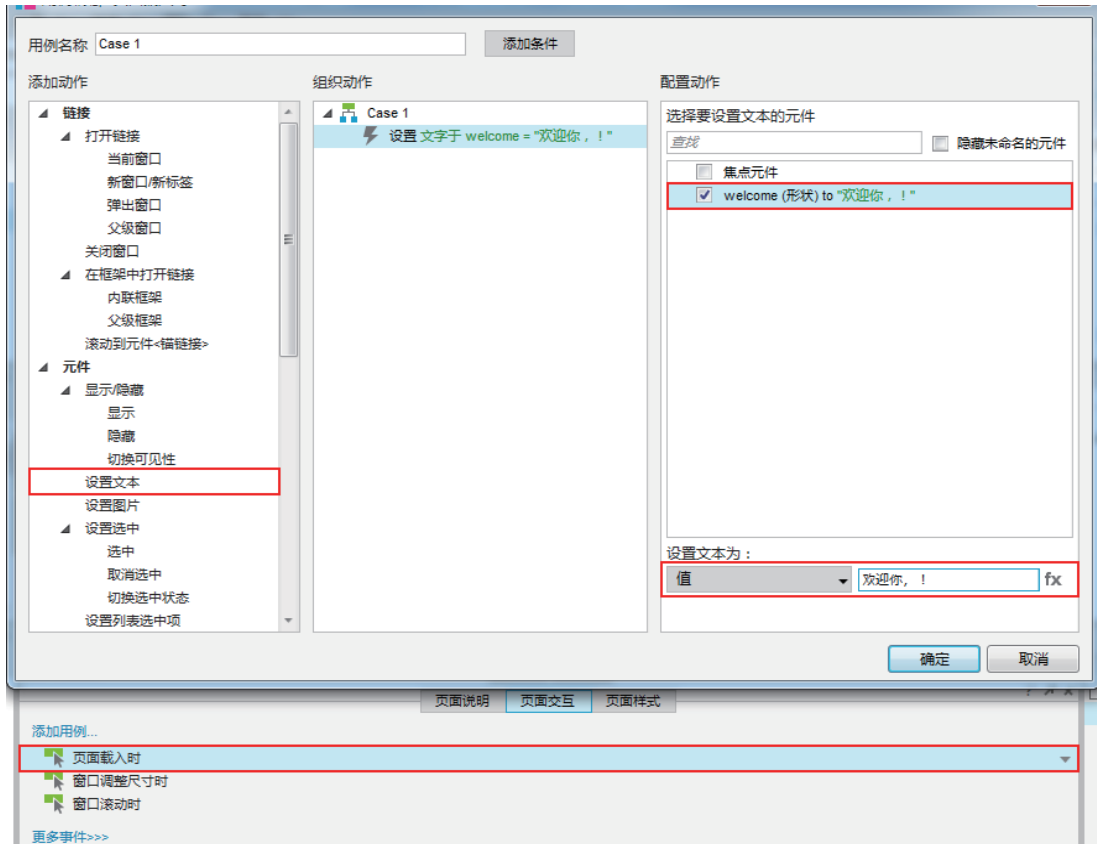


图3-4

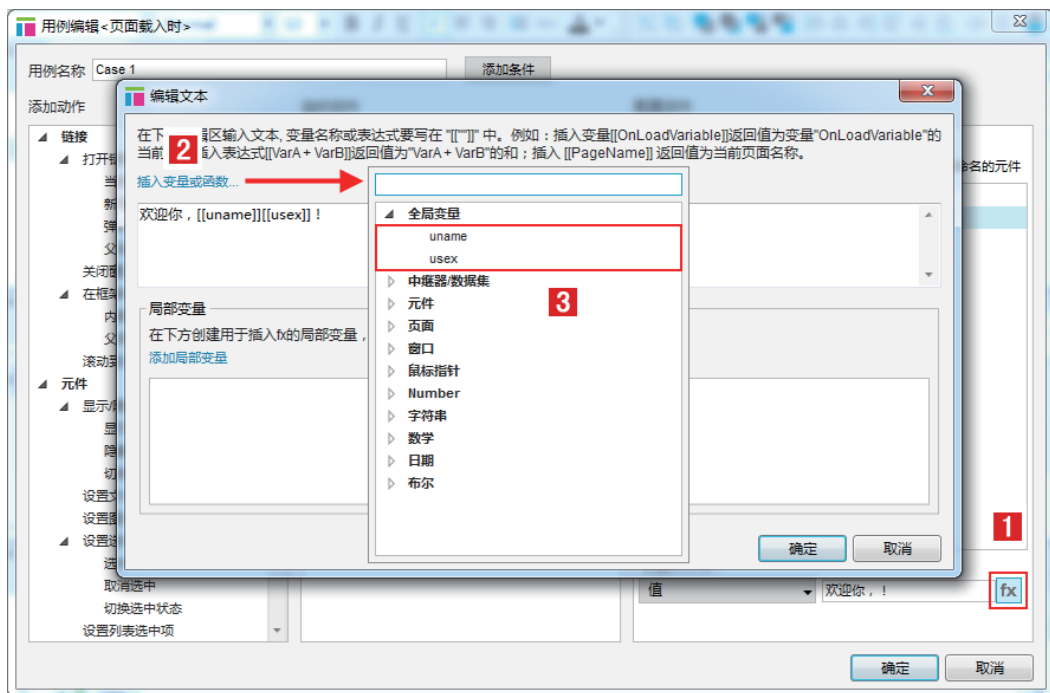


图3-5

备注

在浏览器中查看原型时，如果带有站点地图，我们就能够实时看到全局变量的值。站点地图中第3个功能图标“X=”相当于一个变量跟踪器，它能够实时将项目中所有的变量以及当前的值显示出来。

本节任务 → 不参考教程完成上述案例。

3.1.2 局部变量

局部变量是另一种类型的变量。

如果全局变量像是一张通用的公交卡，局部变量又是什么呢？它像一张临时车票。临时车票是在购买车票时获取并录入了乘车信息，在乘车、检票、出站时会读取这张车票的信息。它的信息只在当前这一次乘车的过程中有效，一旦结束，这张车票即被回收。并且，临时车票在使用过程中不可充值、扣费。

从这个比喻能看出，局部变量只存在于特定的范围，只在这个特定范围内有效，只能够被一次写入，但可以多次被读取。

那么，局部变量如何使用呢？

在第2章第67页“列表选项交互”的案例中，我们做了一个改变下拉列表框的选项，通过单击【搜索】按钮在框架内打开不同搜索引擎主页的效果。（图3-6）



图3-6

下面我们将结合局部变量，实现输入关键字打开搜索结果页的效果。

这里我们先来看几个链接：

<http://www.baidu.com/s?wd=iaxure>

<http://www.sogou.com/sogou?query=iaxure>

<http://www.youdao.com/search?q=iaxure>

上面3个链接，分别是使用这几个搜索引擎搜索“iaxure”时，搜索结果页的关键地址，大家很容易看出来这几个地址最后一部分是搜索关键字。那么，我们是否可以在设置框架打开的超链接中嵌入搜索的关键字，实现打开不同的搜索结果呢？

下面，我们就通过使用局部变量嵌入自己输入的关键字来实现这个效果。

首先，我们将文本框命名为“keytext”；

然后，修改【搜索】按钮的触发事件【鼠标单击时】中的用例动作，以百度的搜索地址为例（图3-7）：

- 步骤1** 单击超链接的输入框后方的【fx】进入编辑值的界面；
- 步骤2** 单击【添加局部变量】按钮，创建局部变量；
- 步骤3** 将局部变量命名为“key”；
- 步骤4** 通过选择列表中的文本框元素“keytext”，完成把它的元素文本赋值给局部变量“key”的操作；
- 步骤5** 修改超链接文字为“http://www.baidu.com/s?wd=”；
- 步骤6** 单击【插入变量或函数…】，打开变量与函数列表，单击我们刚刚创建的局部变量名称，将它插入到超链接文字的下方。这样程序就能读取局部变量中的内容并与当前的超链接文字一起组成搜索结果页的地址。

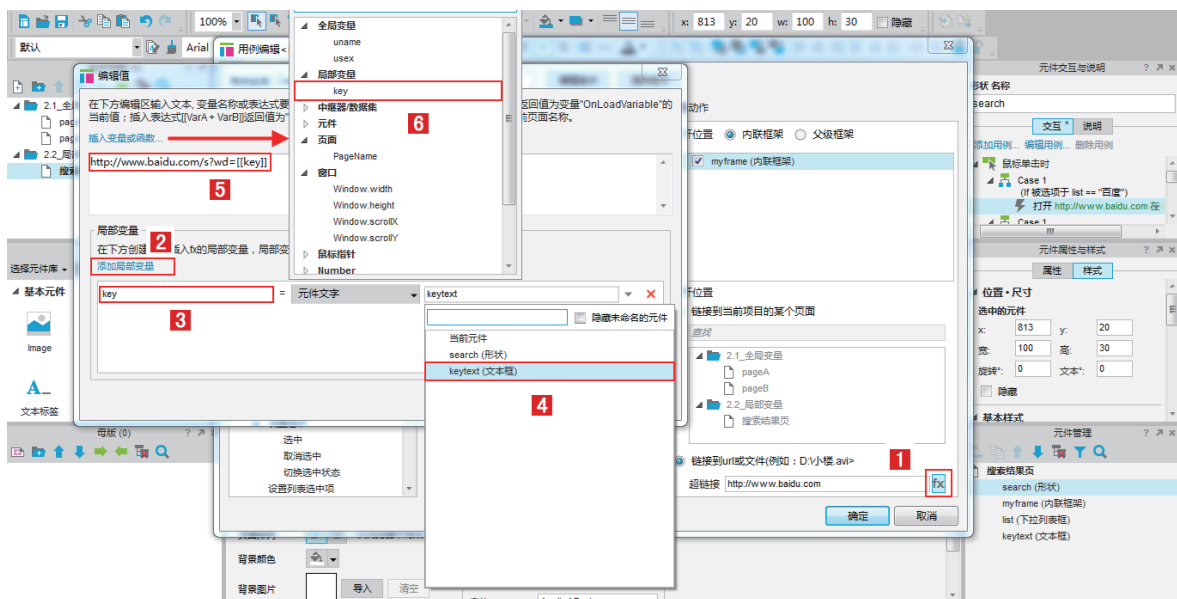


图3-7


通过以上几步操作，我们就完成了在单击【搜索】按钮时，能够打开以输入框“keytext”中文字为搜索关键字的百度搜索结果页面。其他两个搜索引擎，也按照以上步骤完成即可。

最后，提醒大家注意的是，局部变量只在“编辑值”/“编辑文本”的界面中能够创建和使用。我们可以创建局部变量（步骤3），然后通过局部变量获取元件文字、选中状态、被选项等内容（步骤4），并将局部变量的内容读取出来填写到【值】的输入框中参与计算或与其他字符串链接（步骤6）。

本节任务 → 完成其他两个搜索引擎的搜索效果。

3.2 表达式


楼哥为什么我在【fx】的编辑文字界面里写变量名，显示的还是变量名，而不是变量保存的值呢？

 因为你没有写“[]”，你要记住只有把变量写在“[]”中，程序才能通过运算获取到变量值。还有，表达式也要写在[]中才能正确地运算。

 表达式是什么？

 表达式是由数字、运算符、数字分组符号（括号）、变量等组合成的公式。

 运算符是什么？

 运算符，用于执行程序代码运算，会针对一个以上操作数项目来进行运算。例如：2+3,其操作数是2和3，而运算符则是“+”。Axure中的运算符大致有4种：算术运算符、关系运算符、赋值运算符和逻辑运算符。


3.2.1 运算符的类型

算术运算符：

算术运算符就是我们常见的加减乘除符号，符号是“+”、“-”、“*”、“/”。例如：a+b、b×c等。

另外，还有一个算术运算符是取余数，符号是“%”。

 楼哥，什么是取余数？

 例如17%5，它的结果是2，而17%4的结果是1。取余数是指将前面的数字中完整包含了后面的部分去除，只保留剩余的部分。17%5就像把17粒豆子，每次拿掉5个，最后不足5个的留下，结果就是2了。

赋值运算符。

Axure中的赋值运算符是“=”，注意这个是单等号。赋值运算符能够将其右侧的表达式运算结果赋值给左边一个能够被修改的值，例如：变量、元件文字等。（图3-8）

关系运算符。

Axure中的关系运算符一共6种：分别是“<”、“<=”、“>”、“>=”、“==”、“!=”。

关系运算符对其两侧的表达式进行比较，并返回比较结果。它的比较结果只有两种“真”或“假”，也就是“True”和“False”。图3-9是百度百科中给出的例子，给大家作为参考。

逻辑运算符。

Axure中的逻辑运算符有两种，分别是“&&”和“||”。逻辑运算符能够将多个表达式连接在一起，形成更复杂的表达式。“&&”表示并且的关系。而“||”表示或者的关系。

例如：数值a大于b并且不等于c，用表达式来表示的话就是“a>b&& a!=c”。当a为4、b为2、c为5时这个表达式成立，返回值为“True”，如果a为5、b为2、c为5或者a为1、b为2、c为5时这个表达式都不成立，返回值为“False”。

其实，在Axure中还有一种逻辑运算符可以使用，它是“!”表示不是，它能够将表达式结果取反。

例如：!(a>b&& a!=c)，假设它的值与前面的举例一致，则得到的返回值完全相反。

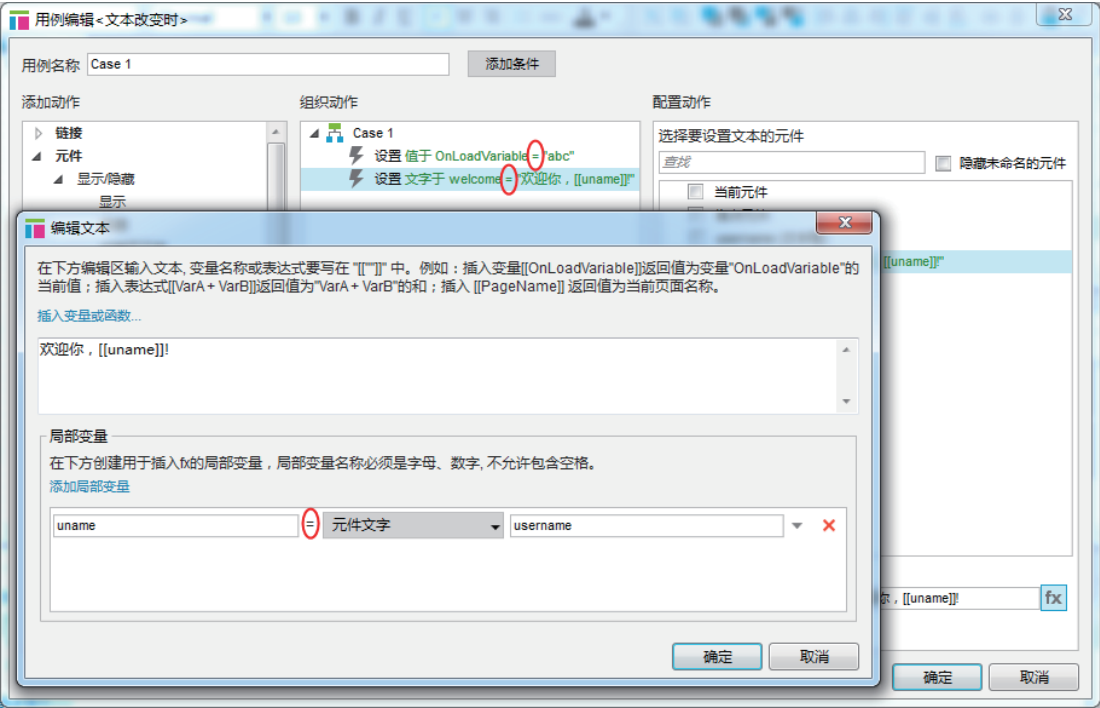


图3-8

运算符	名称	示例	功能
<	小于	a<b	a小于b时返回真；否则返回假
<=	小于等于	a<=b	a小于等于b时返回真；否则返回假
>	大于	a>b	a大于b时返回真；否则返回假
>=	大于等于	a>=b	a大于等于b时返回真；否则返回假
==	等于	a==b	a等于b时返回真；否则返回假
!=	不等于	a!=b	a不等于b时返回真；否则返回假

图3-9

3.2.2 表达式的格式

在运算符的讲解中，大家看到的“a+b”、“a>b”以及“a>b&&a!=c”等，都是表达式。在Axure中只有在值的编辑时可以使用表达式，表达式要写在“[[]]”中。

下面再举几个例子，方便大家理解。

[[uname]]：这个表达式没有运算符，它的返回值是“uname”的变量值。

[[3×5]]：这个表达式的结果为15。

[[uname=='admin']]：当变量uname的值为“admin”时，这个表达式返回值为“True”，否则，返回值为“False”。

[[num1+num2]]：当两个变量值为数字时，这个表达式的返回值为两个数字之和。

 楼哥，如果我想获取两个变量值，把它们连接在一起该怎么写表达式呢？

 你这么快就忘了？不记得全局变量的例子中，那句欢迎语怎么写的了吗？（图3-5）



那句欢迎语是……，“欢迎你，[[uname]][[usex]]！”？



对呀！

如果想将两个表达式的内容链接在一起或者将表达式的返回值与其他文字链接在一起，只需要将它们写在一起就可以了。例如：

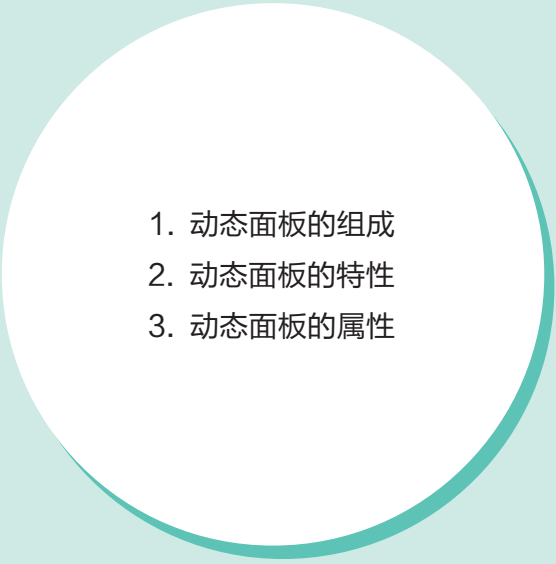
“[[num1]]+[[num2]]”，当num1为3、num2为5时，这个表达式的返回值为“3+5”，而不是“15”。

“[[username]]是帅哥！”，当username为“小楼”，这个表达式返回值为“小楼是帅哥！”。

通过上面的案例，我们能看出，写在“[[]]”里面的内容可以进行运算，而写在“[[]]”外面或者多个“[[]]”的内容写在一起时，则是将返回值与文字或返回值与返回值连接为一个字符串。

第4章 动态面板



- 
- 1. 动态面板的组成
 - 2. 动态面板的特性
 - 3. 动态面板的属性



楼哥，这个叫动态面板的东西是什么？



这个呀！是个容器。



像变量一样吗？



不！这个容器和变量不一样。变量是数据的容器，而动态面板是元件的容器。当然，容器只是它的一个特性，它还有多状态、循环、可改变尺寸……



感觉好复杂！到底要怎么使用呢？



复杂是因为你不了解。其实只要我们知道它的原理，用起来就变得很简单。

►► 4.1 动态面板的组成

不管是在教学视频里，还是在这本书中，我都很喜欢使用比喻。例如把母版的自定义触发事件比喻成配电箱，把全局变量比喻成公交卡。

那么，我们马上接触到的一个比较特殊的元件“动态面板”，我会把它比作什么呢？

这个概念不是特别容易比喻，仅从结构上来讲动态面板就像一个幻灯机，它拥有一个以上的状态，这些状态就像是幻灯片，包含着不同的内容。

动态面板默认会显示排列在第一位状态中的内容，我们能够在元件管理及状态管理中改变状态的次序。另外，动态面板的尺寸决定了显示的内容，默认情况下，如果状态中的内容超出了动态面板的尺寸，则不能够完整显示，就像幻灯片的内容大小超出镜头成像尺寸的话就不能够完全显示。不过，我们也可以通过在动态面板上单击鼠标右键打开菜单，选择【自动调整为内容尺寸】，让动态面板能够自动改变大小，以便让内容完整显示。

4.1.1 状态管理

动态面板拖入到主编辑区后，直接生成或预览查看，你会发现页面中什么都没有。也就是说动态面板其实默认是一个透明的元件，那么如何让它显示内容呢？

双击动态面板，就可以打开“动态面板状态管理”的界面，在这个界面中我们能够看到在状态列表中，显示了一个默认的【State1】。双击这个【State1】的名称，动态面板状态管理的界面消失，主编辑区变成了动态面板中“State1”的编辑区。和普通的页面一样，我们可以向编辑区中拖放任何元件来实现我们想要的内容。不过，大家一定也注意到，编辑区中比一般页面的编辑区多了一个矩形的蓝色虚线框。这个蓝色虚线框与动态面板的尺寸一致，处于蓝色虚线框内的内容才可以被动态面板显示出来；超出虚线框部分的内容，默认情况下在页面中是看不见的。（图4-1）

如图4-1所示，蓝色虚线框大小与页面中动态面板大小一致；图片元件在红框中的部分是在蓝色虚线框内，所以在页面上只能看见图片的这一部分。

以上是对动态面板状态的内容编辑，接下来我们看一下“动态面板状态管理”界面中还有哪些功能。

双击页面中的动态面板，打开“动态面板状态管理”的界面。界面中有几个功能按钮，从左到右依次为新建、重复、上移、下移、编辑状态、编辑全部状态和移除状态。（图4-2）

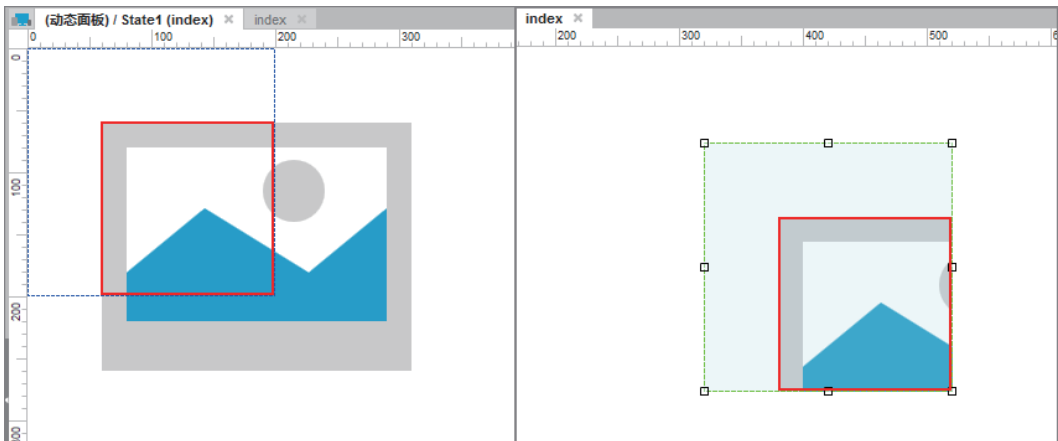


图4-1

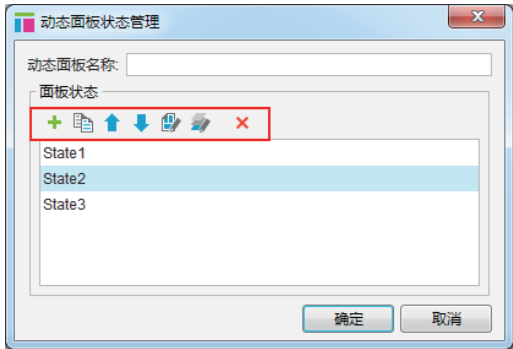


图4-2

- 新建：单击【+】图标，即可新建一个状态，这个状态的内容是空白的。
- 重复：选中一个已有状态，单击【重复】图标，即可新建一个状态并把选中状态中的内容全部复制给新建的状态。
- 上移/下移：选中状态，单击【上移】或【下移】可以调整状态的次序，这个次序影响自动循环切换状态时的显示次序。
- 编辑状态：选中一个状态，单击【编辑状态】图标，则打开状态的编辑界面。一般我们都是通过双击状态名称来完成这个操作。
- 编辑全部状态：自动打开所有状态的编辑界面，并自动切换到第一个状态的编辑界面。
- 移除状态：删除选中的状态。
- 最后，在这个界面中也能够给动态面板添加名称。

4.1.2 状态样式

除了上一节提到的对动态面板状态的管理，我们还可以在每个状态的编辑界面中，为每个状态添加样式。

当我们打开一个状态的编辑界面时，软件中下部分的“页面设置”区域变成了“状态样式”。（图4-3）

在这个编辑区域中，我们能够给动态面板当前的状态添加背景颜色和背景图片。

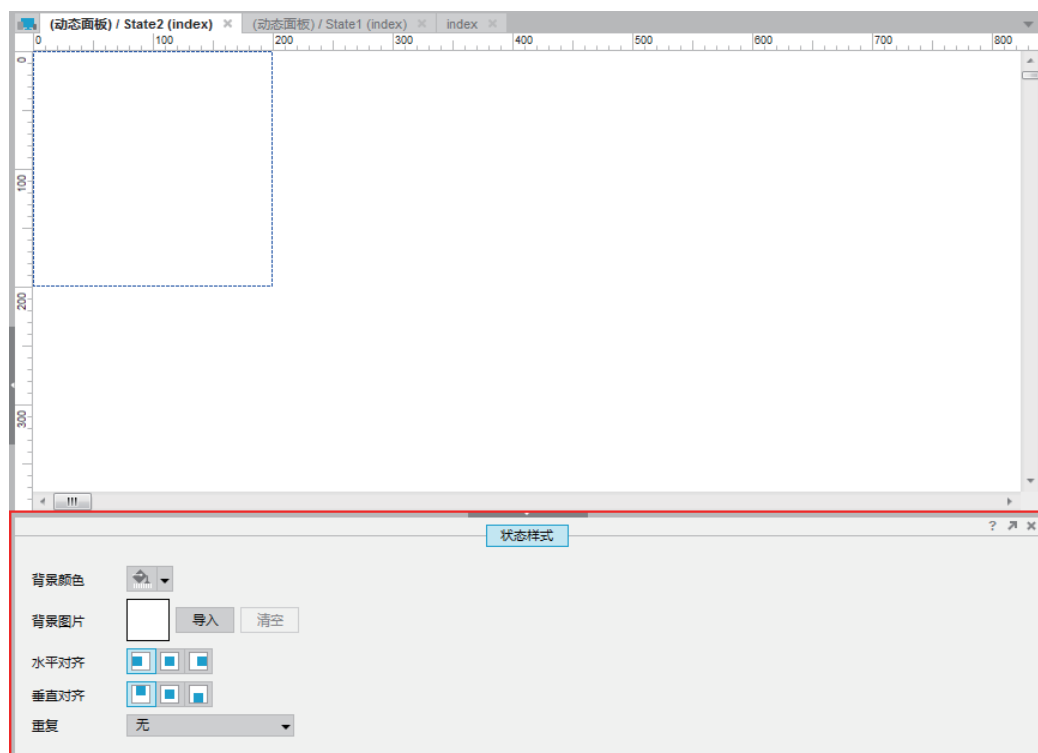


图4-3

需要注意的是，当动态面板改变尺寸时，动态面板添加的背景图片会跟随改变尺寸。当图片的宽高比例与动态面板宽高比例不一致时，可以在{重复}（状态样式中的重复）中设置【填充】或者【适应】的显示规则。显示规则说明参见第2章2.8.1节中的相关内容。

4.2 动态面板的特性



楼哥，你讲了这么多动态面板的内容，我还是不知道它能做什么，怎么用。



不要这么急，一个元件你要先了解它的本质，才能够更好地应用它。下面我就给你讲讲动态面板的几种主要应用。

4.2.1 容器

动态面板是一个透明的容器，它能够包含各种元件在里面。那么，容器都有什么特点呢？

买菜的时候，菜篮就是容器，它能帮我们方便地携带买来的多种蔬菜。

上网聊天的时候，一个聊天窗口也是一个容器，里面包含聊天内容，当隐藏这个窗口时，所有聊天内容也就被同时隐藏。

这样看来，容器可以用来装载内容，这些容器中的内容会随着容器一起移动或者显示隐藏。

动态面板既然是容器，同样包含这些特性。装入动态面板状态中的内容可以随着动态面板一起移动、显示和隐藏。

下面我们基于动态面板容器的特性来做两个案例。

案例1：二级菜单的显示和隐藏

鼠标移入一级菜单时，显示二级菜单；鼠标移出时，隐藏二级菜单。

首先，我们准备这个案例需要的元件。1个表示一级菜单的矩形，和3个表示二级菜单的占位符。

为了让3个二级菜单能够统一显示和隐藏，把它们放进动态面板的首个状态里面。

这里有一个快捷的操作方法，如果动态面板只有一个状态，我们除了可以打开状态的编辑区添加内容，还可以将编辑好的内容全选，单击鼠标右键，在弹出的菜单中选择【转换为动态面板】，软件会在创建动态面板的同时将这些内容变成动态面板状态中的内容。如果操作错误或者想将内容从动态面板中取出，可以选择【从首个状态中脱离】。（图4-4）

将二级菜单封装进动态面板后，我们给动态面板命名为“menu”，在快捷功能区或者元件样式面板中勾选隐藏的复选框，将这个动态面板隐藏。

这个案例的事件是在鼠标移入一级菜单时被触发，所以我们在编辑区点中一级菜单，为其添加【鼠标移入时】事件的内容。在用例编辑界面的动作列表中，选择【显示】，然后勾选动态面板“menu”。

完成上述操作后，大家注意用例编辑界面的右下方，有个【更多选项】，这里面有个动画效果叫【弹出效果】。这个动画效果，能够帮我们直接实现鼠标移出时重新隐藏动态面板“menu”的效果，而不用再去编辑鼠标移出时的事件。我们把它选上后，单击“确定”即完成了这个案例。（图4-5）

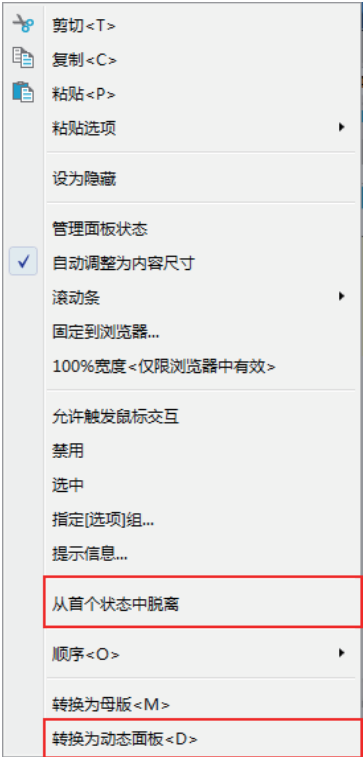


图4-4

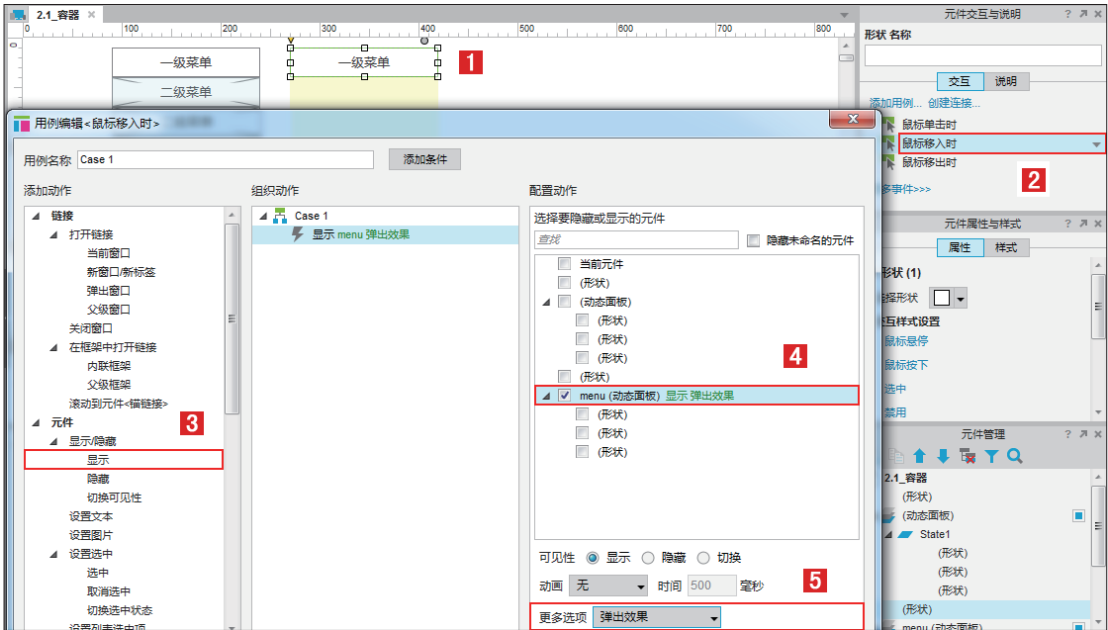


图4-5

为了方便理解，我们来看一下整个实现过程。（图4-6）

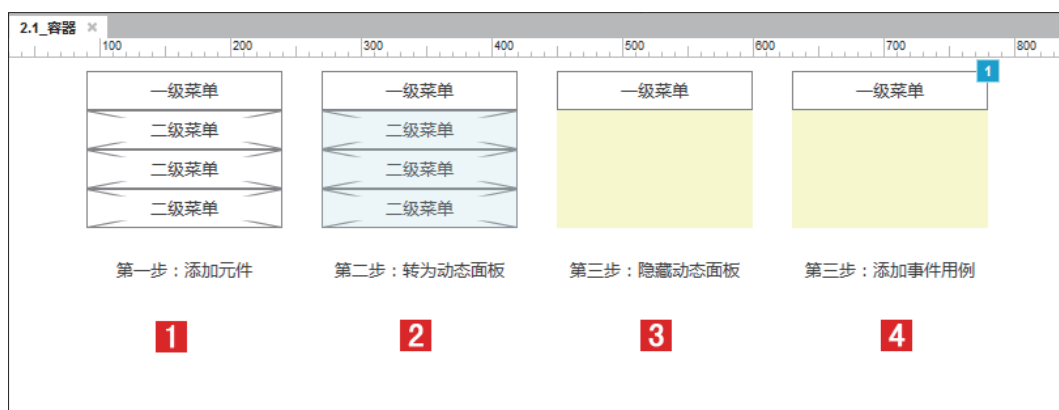


图4-6

如果需要多个菜单组合，可以同时选中已经做好的一级菜单和二级菜单，按住鼠标左键拖动进行复制，然后修改复制后的元件名称即可。

案例2：带遮罩效果的弹窗

单击【登录】按钮，弹出一个登录窗口，窗口周围有半透明遮罩层，遮盖页面原有内容。（图4-7）



图4-7

这个效果，只需把用户登录的这个面板（1个矩形背景+1个文本标签+2个文本框+1个形状按钮）转换为动态面板并命名为“loginpanel”后隐藏。然后单击元件水平菜单上的菜单项“登录”，显示这个“loginpanel”动态面板即可。如果想要半透明遮罩效果的话，就需要在配置动作时在【更多选项】中选择【灯箱效果】。（图4-8）

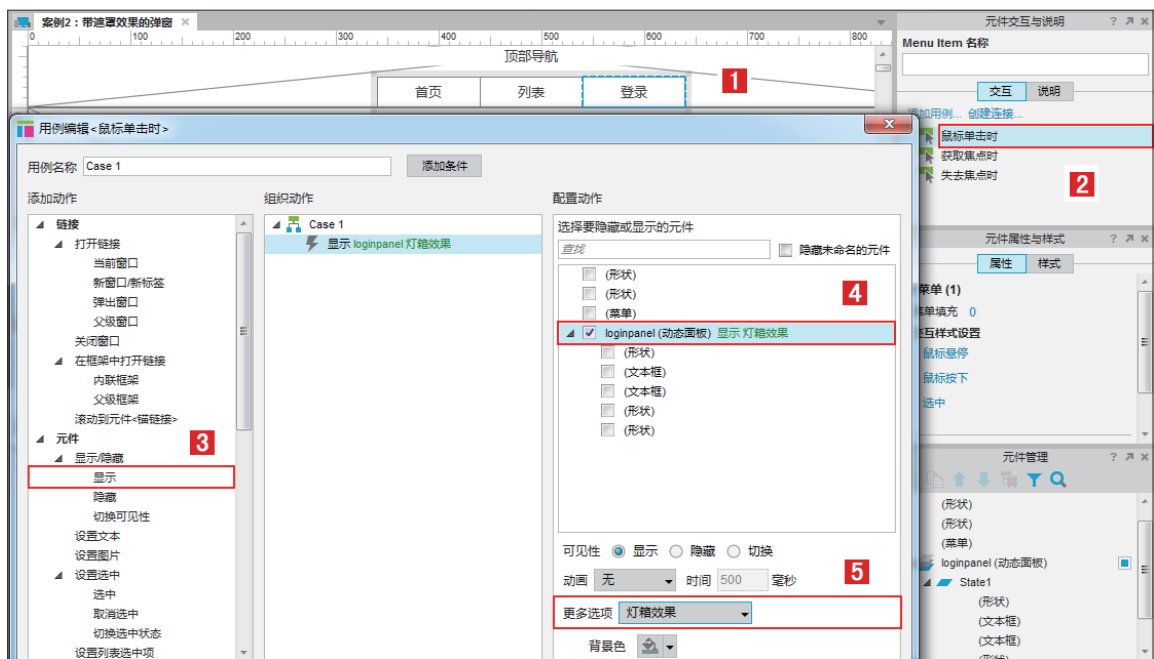


图4-8

4.2.2 固定

动态面板的属性中有个功能，叫【固定到浏览器】，这个功能也可以在动态面板的右键菜单中找到。

如果我们希望页面上的某个部分，在页面滚动时保持在原位不动，就需要用到【固定到浏览器】的这个功能。

例如，一个【返回顶部】的按钮一直固定在页面右下方，距离浏览器右边框100像素和下边框10像素的位置上。单击这个按钮时，整个页面滚动回顶部。这样的效果就需要用到上面提到的功能。

其实，【固定到浏览器】这个功能，并不是把元件在绝对位置固定，它是相对于浏览器的固定，对于页面来说其实元件是在随着页面的滚动而反向移动，形成了一个停留在原位置的视觉效果。所以它是相对固定，相对于浏览器边框的固定。

知道了原理，我们来实现上面说的这个效果。

案例：固定位置的【返回顶部】按钮

首先，我们设定固定按钮的位置。（图4-9）

① 我们先搭建一个简易的页面，然后放入一个矩形，改成箭头样式，并选择【转换为动态面板】，命名为“back”。

② 单击动态面板属性中的【固定到浏览器】，打开设定界面；

③ 勾选【固定到浏览器窗口】的复选框，开启设置选项；

④ 水平固定中选择【右】，并设置边距为“100”px；

⑤ 垂直固定中选择【底部】，并设置边距为“10”px。

然后，添加【返回顶部】按钮【鼠标单击时】事件的用例动作；因为当前动态面板中只有一个元件，所以【鼠标单击时】的事件可以用矩形也可以用动态面板。这里我们使用动态面板的触发事件。

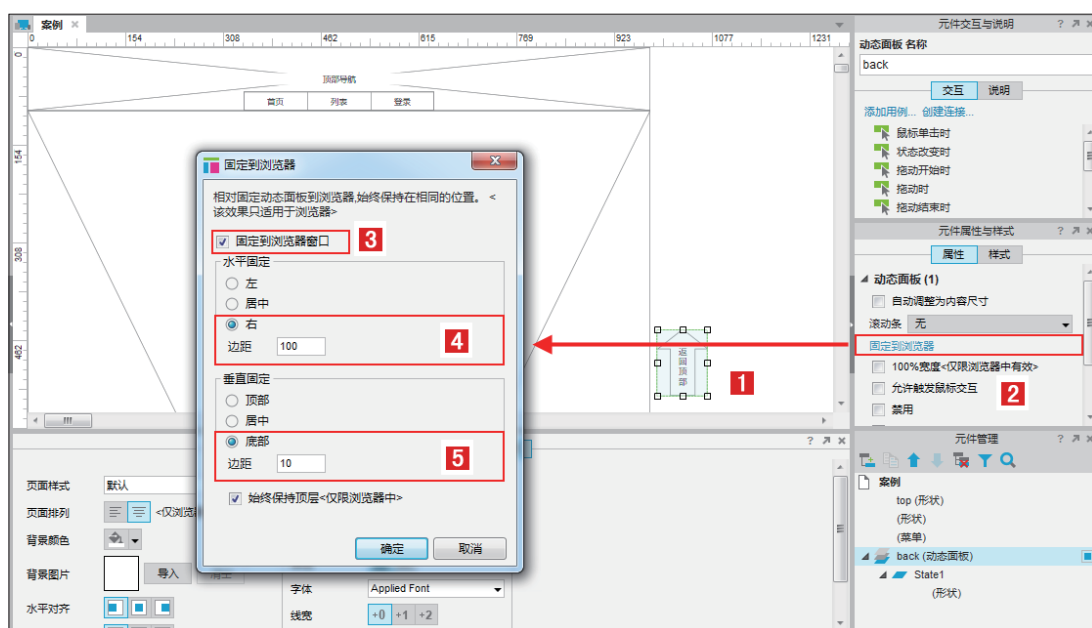


图4-9

如果要滚动回顶部，我们需要用一个元件在顶部进行位置定位，滚动时滚动到这个元件的位置上。页面中，有一个文字为“顶部导航”的占位符元件，它的y轴坐标为“0”，就是在顶部的位置，我们就用它进行顶部位置的定位。

我们将“顶部导航”这个元件命名为“top”，并在用例编辑界面中选择动作【滚动到元件<锚链接>】后，勾选这个元件。在接下来的选项中勾选【仅垂直滚动】，如果需要滚动回顶部时有动态的效果，可以继续选择【动画】中的【线性】效果，时间为默认的“500”毫秒即可。这样就是一个以500毫秒匀速滚动回顶部的效果。（图4-10）

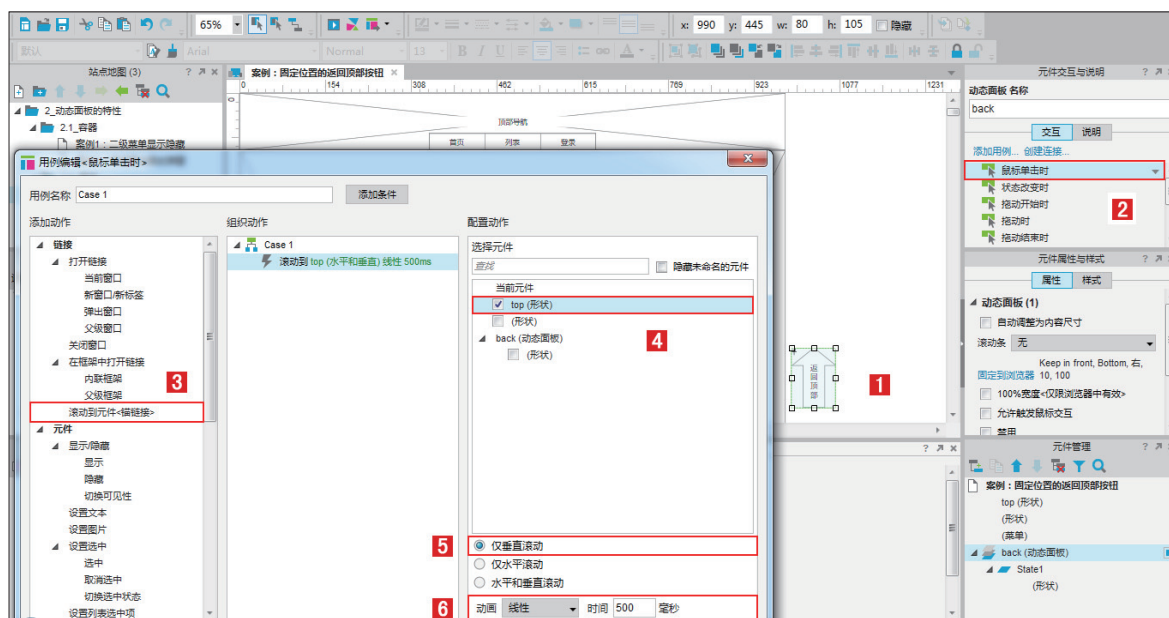


图4-10

4.2.3 状态

在本章 4.2.1 节中我们把动态面板的一个状态当做容器来使用，在本节我们来看一下如何使用动态面板的多个状态。

先举一个例子：一个注册模块，可以分为手机注册和邮箱注册。但是这些注册模块应该都在一个位置上，单击不同的注册标签，显示不同的注册内容。（图 4-11）



图 4-11

如图 4-11 所示，我们需要在单击上方的“手机”和“邮箱”的标签时，下方的内容随之改变。这样的效果就需要用到动态面板的状态切换功能。我们把不同的注册内容放置在不同的动态面板状态中，然后在单击标签时，将动态面板设置为不同的状态。

下面我们就来实现这个案例：多标签的注册模块。

首先，准备好需要的元件。（图 4-12）



图 4-12

步骤 1 先放入一个背景矩形和文本标签；

- 步骤2** 继续放入两个用于单击的标签，并设置为未被选中时的样式；
- 步骤3** 继续放入“手机”注册的相关内容，注意注册内容的背景矩形如图4-12中第6步所示，没有上边框；
- 步骤4** 将“手机”注册内容转换为动态面板，并命名为“register”；
- 步骤5** 双击动态面板“register”，将“State1”的名称修改为“phone”，然后继续添加一个状态，命名为“email”，并双击打开；
- 步骤6** 在“email”这个状态中，添加“邮箱”注册的相关内容。

经过以上几个步骤，我们完成了元件的准备。在本书之后的内容中，我们将不再就准备元件做详细的讲解。

然后，我们先分别为两个选项标签添加选中样式、选项组名称以及触发事件【鼠标单击时】【选中】【当前元件】的用例动作。

接下来，就是最重要的一步，在单击选项标签的时候，我们需要添加用例动作，为动态面板设置相应的状态。以“手机”注册的选项标签为例，我们在这个选项标签【鼠标单击时】的用例中继续添加动作【设置动态面板状态】，然后，在配置动作中勾选动态面板“register”，在下方的选项中，我们为动态面板【选择状态为】【phone】。（图4-13）

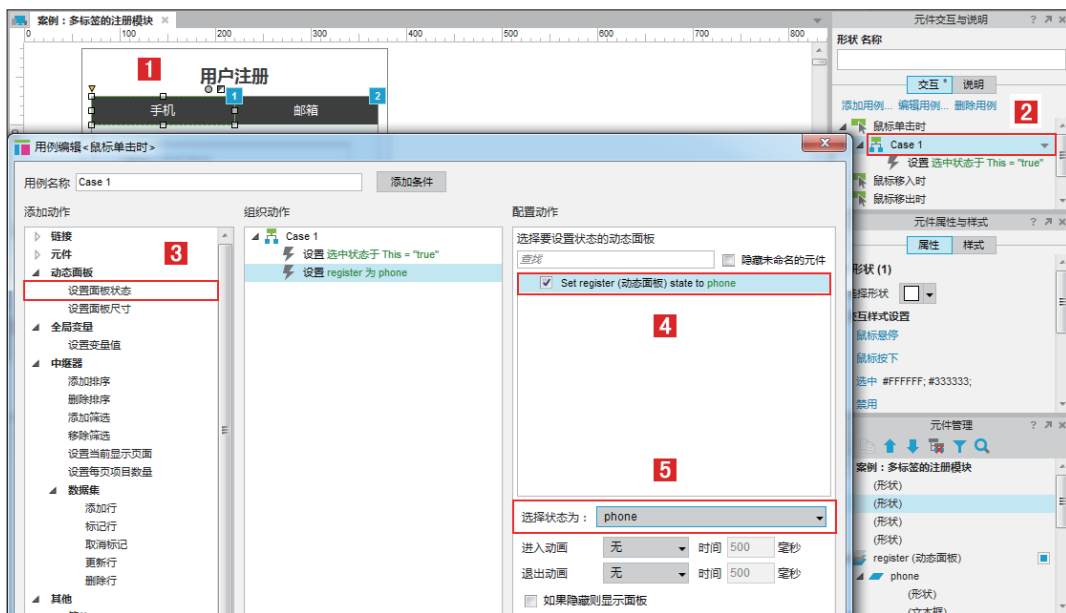


图4-13

最后，参照“手机”注册的设置，也为“邮箱”注册进行相应的设置，当选项标签被单击时，将动态面板状态设置为【email】。

4.2.4 拖动

动态面板是 Axure 中唯一一个能够被拖动的元件。任何元件想要有被拖动的效果，都要放置在动态面板中。

如果需要让一个动态面板被拖动，必须在动态面板的触发事件【拖动时】中添加用例动作，【移动】

动态面板为某一种拖动行为。动态面板的拖动行为包括：拖动（自由拖动）、水平拖动和垂直拖动。

案例：水平滑块

我们制作一个能够水平拖动的滑块，并且在拖动时不得超出标尺两端。（图4-14）

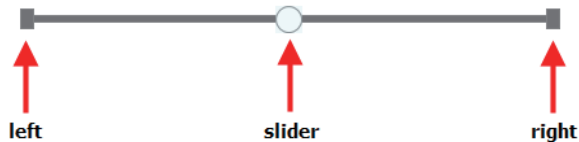


图4-14

我们用矩形做一个标尺，并在两端分别用矩形做好左边界“left”和右边界“right”，然后将矩形改变为圆形并转换为动态面板，作为滑块“slider”。当拖动滑块时，滑块能够水平移动。并且在移动过程中，如果触碰到两端边界就要停留在指定的位置上。

首先，我们先让滑块能够被拖动。双击动态面板的触发事件【拖动时】，添加用例动作【移动】，勾选动态面板【slider】或者【当前元件】，选择【移动】列表中的选项【水平拖动】。（图4-15）

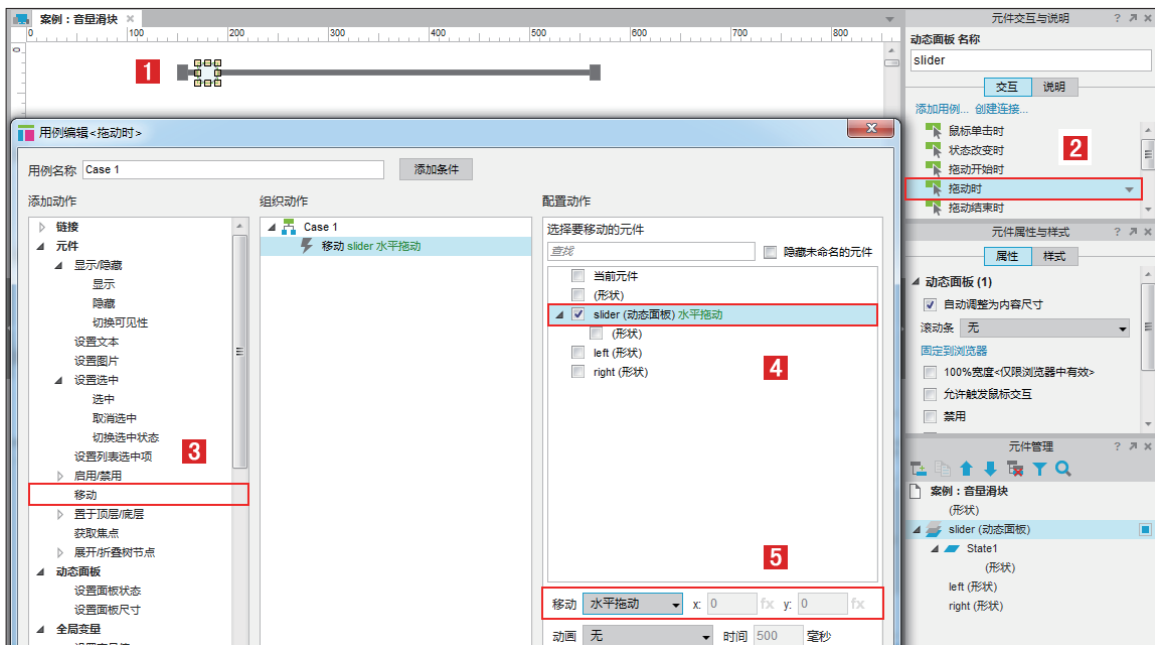


图4-15

然后，继续添加用例，判断拖动时是否超出了边界，如果超出了就让滑块移动到指定位置上。

以左侧边界为例，矩形“left”的x轴坐标为150，宽度为10，如果滑块接触到左侧边界的矩形，我们就要让滑块移动到x轴160的位置上。因为是水平移动，所以y轴坐标不会改变，仍然是当前滑块的y轴坐标20。而滑块接触到右侧边界时，要移动到的位置是右侧边界矩形的x轴坐标-滑块的宽度。右侧边界矩形“right”的x轴坐标为560，减去滑块的宽度20，结果是540。y轴的坐标同样是20。（图4-16）

了解了边界限制滑块的原理，我们就来添加判断和相应的用例动作。仍以左侧边界为例：

步骤1 双击动态面板“slider”的【拖动时】事件打开用例编辑界面，然后单击【添加条件】，设置条件内容为【元件范围】“slider”接触【元件范围】“left”；（图4-17）

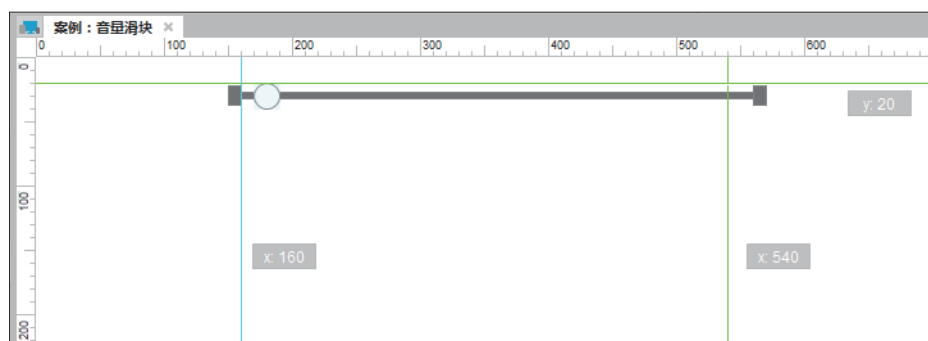


图 4-16



图 4-17

步骤 2 设置满足条件时的动作为【移动】“slider”，移动【到绝对位置】x轴“160”y轴“20”；(图 4-18)

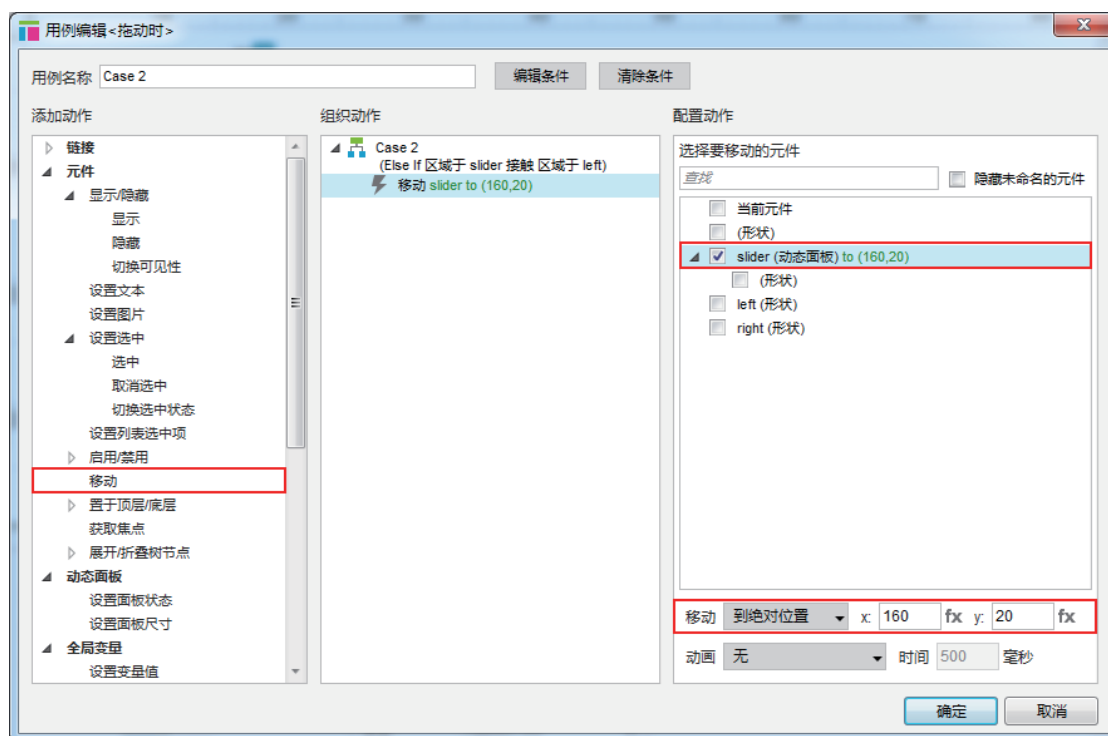


图 4-18

步骤 3 设置当满足条件【元件范围】“slider”接触【元件范围】“right”时，【移动】“slider”【到绝对位置】，{x:} “540”，{y:} “20”；

步骤 4 将动态面板“slider”的触发事件【拖动时】中所有的用例的条件判断，全部切换为If。

4.2.5 缩放

动态面板也是在 Axure 中唯一一个能够调整尺寸的元件。结合动态面板能够编辑状态样式，我们能够实现图片缩放以及简单的进度条加载效果。

案例：图片等比中心点缩放

这个案例效果为鼠标移入图片时，图片按中心点放大；鼠标移出时，图片缩小为原尺寸。放大缩小有动态的效果。

首先，要完成这个效果我们需要先将元件放入动态面板，命名为“zoomimage”，然后设置其背景图片，再设置【重复】为【填充】或【适应】。这样能够保证动态面板改变尺寸大小时，背景图片能够跟随改变尺寸大小。（图4-19）

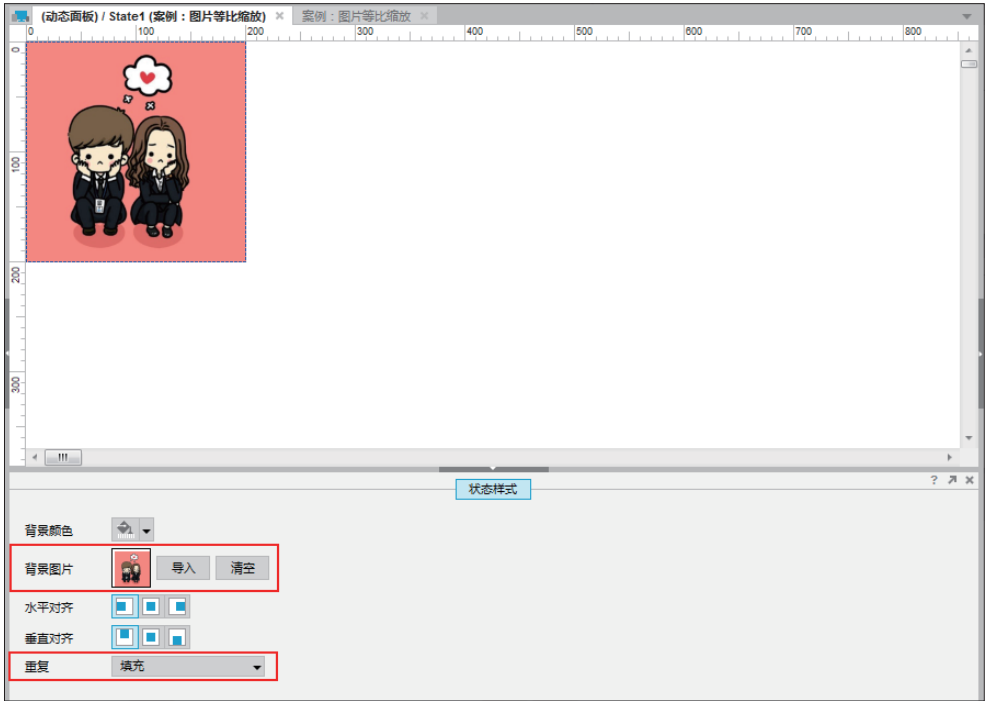


图 4-19

接下来，我们分析一下图片缩放的原理。图片（动态面板背景）放大前尺寸为 200×200 ，放大后尺寸为 300×300 ，因为是中心点缩放，图片放大前与放大后的位置相对移动了 $x50$ 、 $y50$ 。（图4-20）

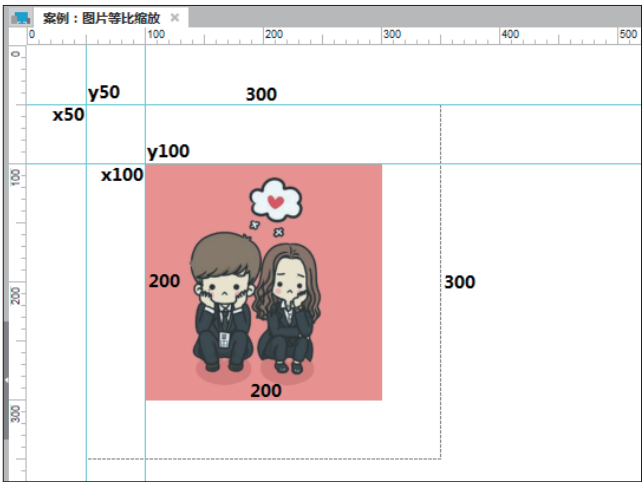


图 4-20

根据分析我们得出，除了要对动态面板进行放大缩小，还需要在放大缩小的同时将动态面板移动到指定的位置上。但是，动态面板在同时缩放与移动的时候，动画效果会互相影响，所以，如果有动态效果，就要把移动和缩放分开进行。解决这个问题的办法，就是在动态面板的外面再添加一个动态面板。外层动态面板负责移动，内层动态面板负责改变图片背景的尺寸。

具体实现步骤如下：

步骤 1 在之前做好的动态面板“zoomimage”上单击鼠标右键，选择【转换为动态面板】完成嵌套，为新的外层面板命名为“moveimage”，然后再次单击鼠标右键设置外层动态面板【自动调整为内容尺寸】；

步骤 2 为外层动态面板“moveimage”添加【鼠标移入时】事件的用例动作为【移动】动态面板“moveimage”【到绝对位置】{x:}“50”{y:}“50”，并为其设置【动画】【线性】“500”毫秒；（图4-21）

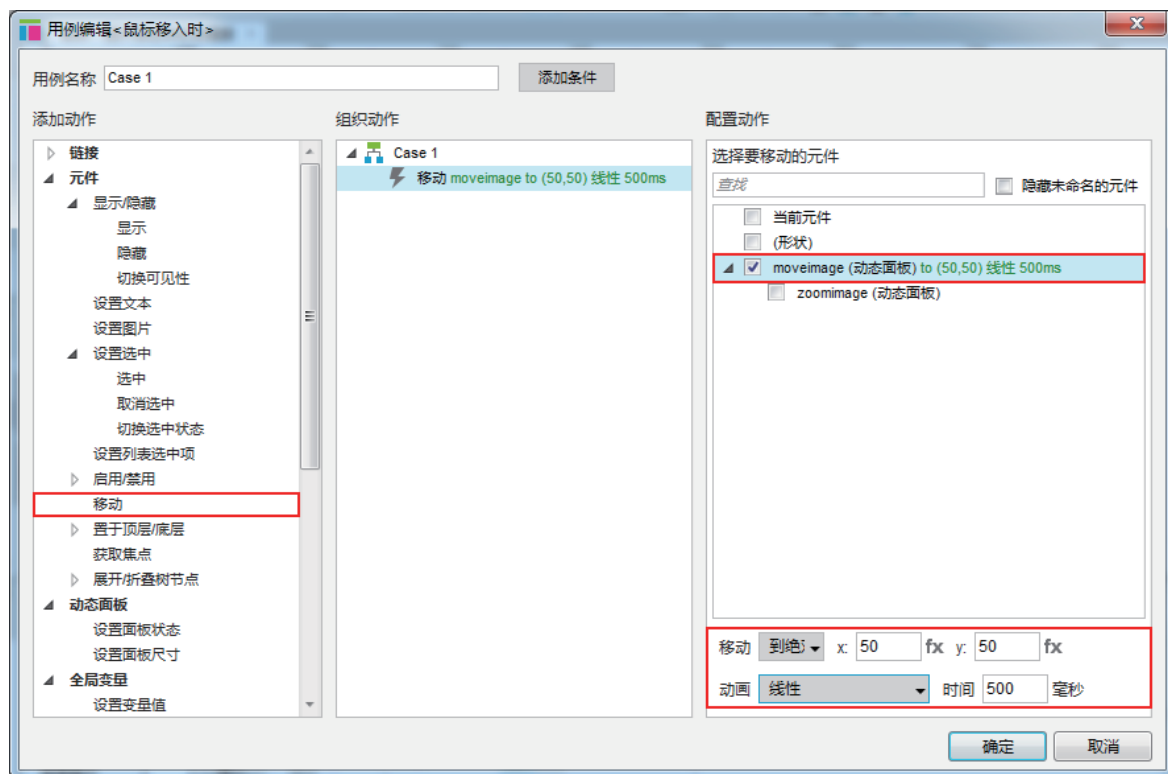


图4-21

步骤 3 继续为上一步中的用例添加动作为【设置面板尺寸】动态面板“zoomimage”，{宽}“300”，{高}“300”，【动画】同样设置为【线性】“500”毫秒。（图4-22）

通过以上步骤，完成了鼠标移入图片时中心点放大的效果。

参照图片放大这个原理，我们继续完成鼠标移出图片时按中心点缩小为原来尺寸的效果。我们设置【鼠标移出时】【移动】动态面板“moveimage”【到绝对位置】，{x:}“100”，{y:}“100”；同时【设置面板尺寸】动态面板“zoomimage”，{宽}“200”，{高}“200”。不要忘记，以上两个动作都要添加【动画】【线性】“500”毫秒。（图4-23）

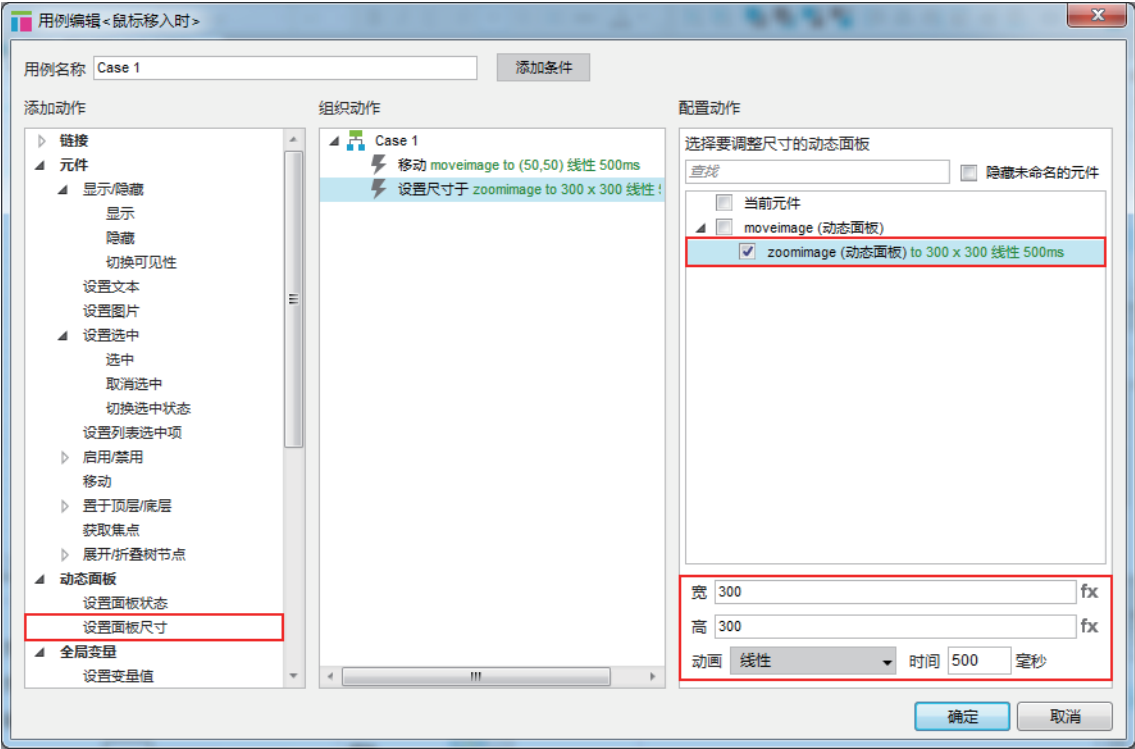


图 4-22



图 4-23

备注

动态面板的【鼠标移入时】和【鼠标移出时】事件，一般都没有直接显示在列表中，需要在【更多事件】中选择。

案例：进度自动加载

相对于上个案例，这个案例更加简单。实现过程为：

- 步骤 1** 放入一个矩形作为进度条外框，尺寸为500×5；然后，放入一个动态面板，为其设置状态样式的背景颜色为红色，并调整这个动态面板的高度为5，左侧与矩形边框对齐。（图4-24）

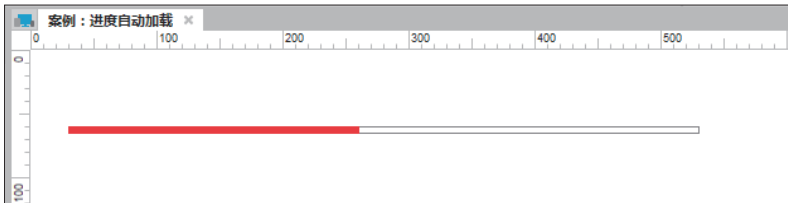


图4-24

- 步骤 2** 点中动态面板，为其添加触发事件【载入时】的用例动作。先设置动作【设置面板尺寸】【当前元件】为{宽}“1”，{高}“5”，让动态面板载入时进度条变为最小的状态。然后，设置动作【设置面板尺寸】【当前元件】为{宽}“500”，{高}“5”，并设置【动画】为【线性】“3000”毫秒，这样就会有一个用时3秒钟，从最小宽度到最大宽度匀速增长的进度条效果。（图4-25）

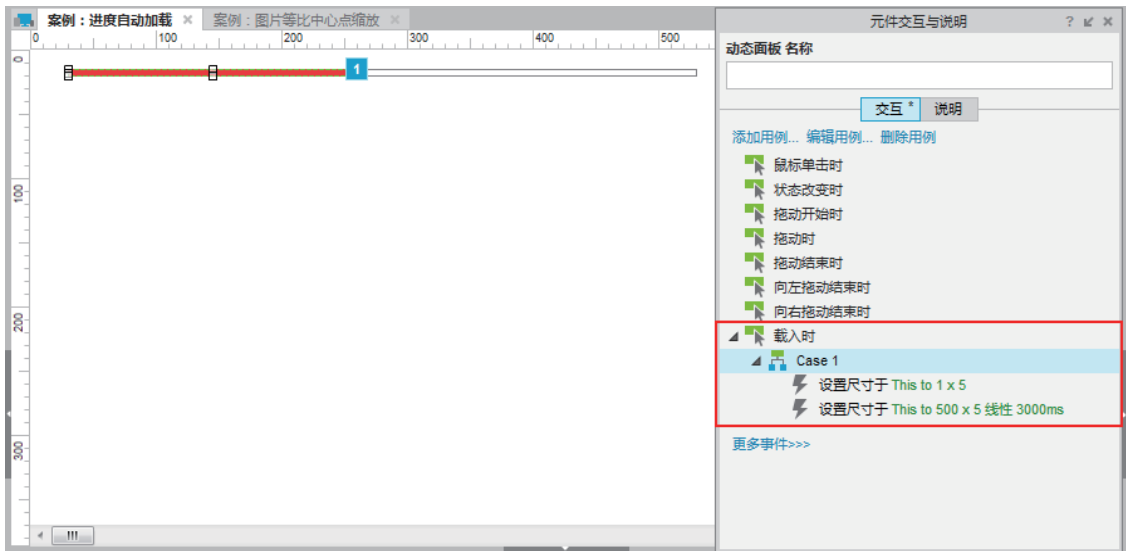


图4-25

4.2.6 循环

在很多网站上，都能看到在某个位置上会有多张图片不停地循环播放。这是幻灯片的自动播放效果，通常称为图片轮播。

在 Axure RP 7.0 版本中这个效果的实现只需简单设置，然后在【载入时】触发事件的用例中添加一个动作即可。

案例：自动图片轮播

我们假设有6张图片进行循环播放。因为6张图片尺寸一致，操作时，我们放入1张图片元件调整好尺寸，其余5张只需进行复制。但是，这6张图片我们需要放置在动态面板的6个状态中，分别在6

个状态中复制粘贴图片元件，显然是件很麻烦的事情，这里我们使用元件管理面板中的【复制状态】功能，让操作变得更简便。

首先，我们先拖入一个图片元件，设置好合适的尺寸，这里是400×240。然后，将其转换为动态面板，并命名为“slide”。这时在元件管理中，我们就能看到这个动态面板及它的默认首个状态【State1】，刚刚的图片元件就包含在这个状态中。（图4-26）

然后，在元件管理的元件列表中，我们点中【State1】，单击5次【复制状态】按钮，这样就完成了状态和图片的同时复制。（图4-27）

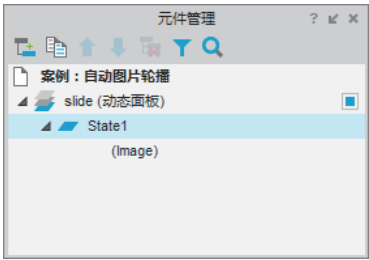


图4-26

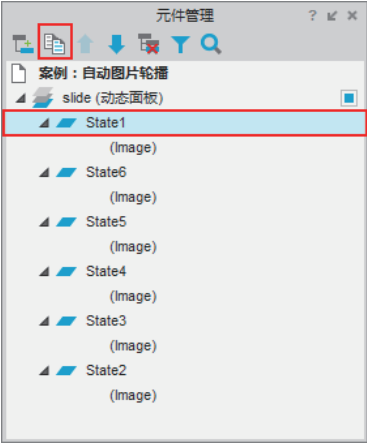


图4-27

接下来，为每个图片元件指定不同的图片，在元件管理的元件列表中，双击两次图片元件即可打开文件选择窗口，选择本地的图片文件。

备注
这一步中图片元件的两次双击，第一次双击会打开动态面板状态，第二次双击打开文件选择窗口。

最后，我们回到页面中，点中动态面板，为它添加触发事件【载入时】中的用例动作为【设置面板状态】“slide”{选择状态为：}【Next】，同时，勾选【向后循环】和【循环间隔】的选项，并设置循环间隔为“2000”毫秒。

这时，就完成了幻灯片自动播放的效果，但是图片的切换是直接改变的，而没有滑动的效果。为了让图片切换有滑动的效果，我们还需要设置{进入动画}【向左滑动】“500”毫秒和{退出动画}【向左滑动】“500”毫秒。（图4-28）

备注
这个案例是向后循环播放幻灯片所以{选择状态为：}【Next】，如果需要向前循环播放，可以{选择状态为：}【Previous】。

备注
进入动画：指即将显示的状态切换进入时的动画效果。
退出动画：指当前已显示状态切换退出时的动画效果。

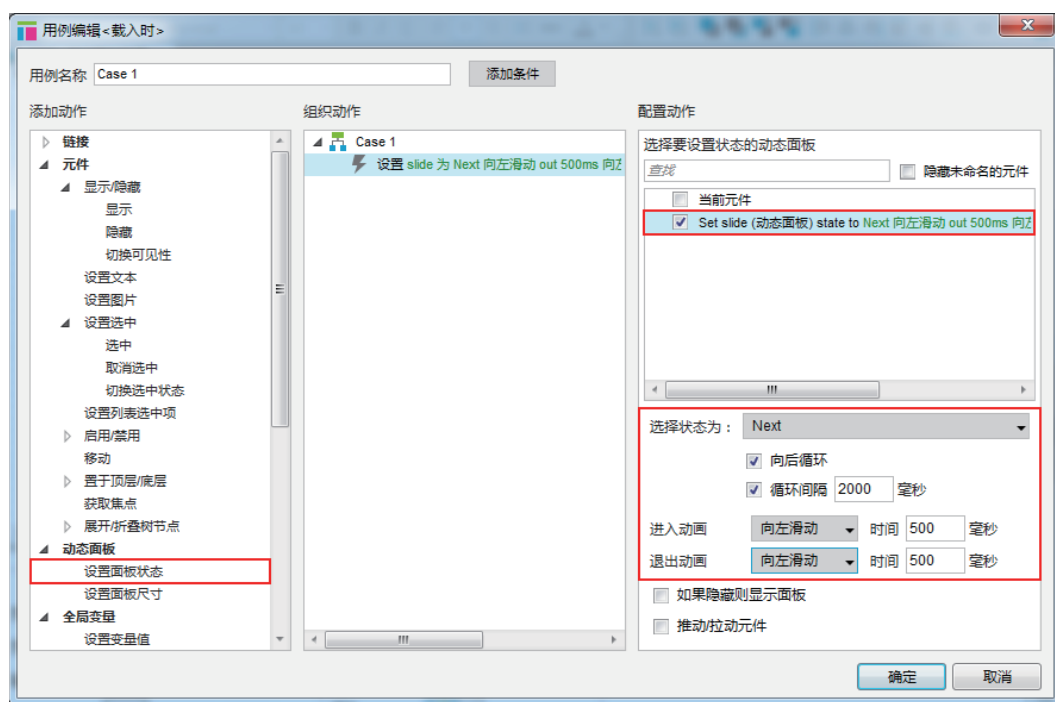


图 4-28

4.3 动态面板的属性

之前我们使用过一些功能：【自动调整为内容尺寸】和【固定到浏览器】，这些功能都在动态面板的属性中。

除了这些功能，动态面板的属性中还有一些其他功能，在这里给大家简单介绍一下：（图 4-29）

1. 滚动条设置

动态面板的滚动条默认为【无】，不显示。它还有其他3个选项，分别为【自动显示滚动条】、【自动显示水平滚动条】和【自动显示垂直滚动条】。

如果动态面板的属性中不勾选【自动调整为内容尺寸】，当面板状态中所包含内容的边界超出动态面板的右边界或者底边界时，就可以出现滚动条。我们可以根据需求在下拉列表中选择设置。

2. 100% 宽度

这个功能，能够让动态面板自适应浏览器宽度。页面打开时在水平方向自动铺满窗口。这个效果只有当页面在浏览器打开时可见，在编辑区没有任何效果。需要注意的是此功能只是动态面板的尺寸自适应浏览器窗口，与其状态中的元件无关，元件不会随着面板改变自身尺寸。

3. 允许触发鼠标交互

这个功能是指：如果为动态面板状态中的元件设置了与鼠标相关的交互样式，那么当鼠标与动态面板产生相同的交互时，是否触发状态中所包含元件的交互样式。

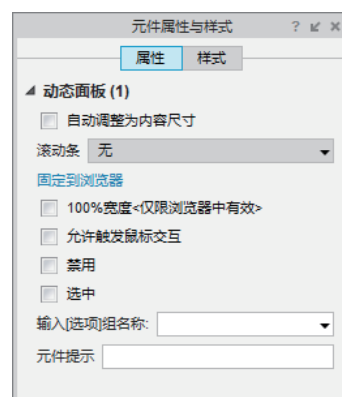


图 4-29

上面这段话，作者自己看了都觉得糊涂。打个比方说吧！例如将动态面板中的白色矩形设置为鼠标悬停时填充颜色变为红色，如果勾选了【允许触发鼠标交互】，当鼠标进入到动态面板范围内但没有进入到矩形范围内时，就会触发交互样式，将矩形变为红色。如果没有勾选则只有鼠标进入矩形范围内时才会触发。

4. 禁用

一般元件勾选禁用，会导致元件自身的交互失效。动态面板的禁用除了会导致自身交互失效，还会导致动态面板中包含的其他元件交互全部失效。

5. 选中

一般元件勾选选中，元件会在页面加载时自动变为被选中的状态，并能够触发元件选中时的交互样式。动态面板的选中除了自身变为被选中的状态，还会将其包含的其他元件全部变为被选中的状态。

基于这个特点，当我们想让一组元件都在【鼠标单击时】变为选中状态的话，可以将它们一起转换为动态面板，然后在动态面板【鼠标单击时】选中该动态面板即可。

案例：单击选中整行

步骤 1 先用多个矩形组成一行。如果想单击任何一个矩形，就让整行变成灰色的话，需要给这些矩形都添加好【选中】的交互样式（图4-30）；

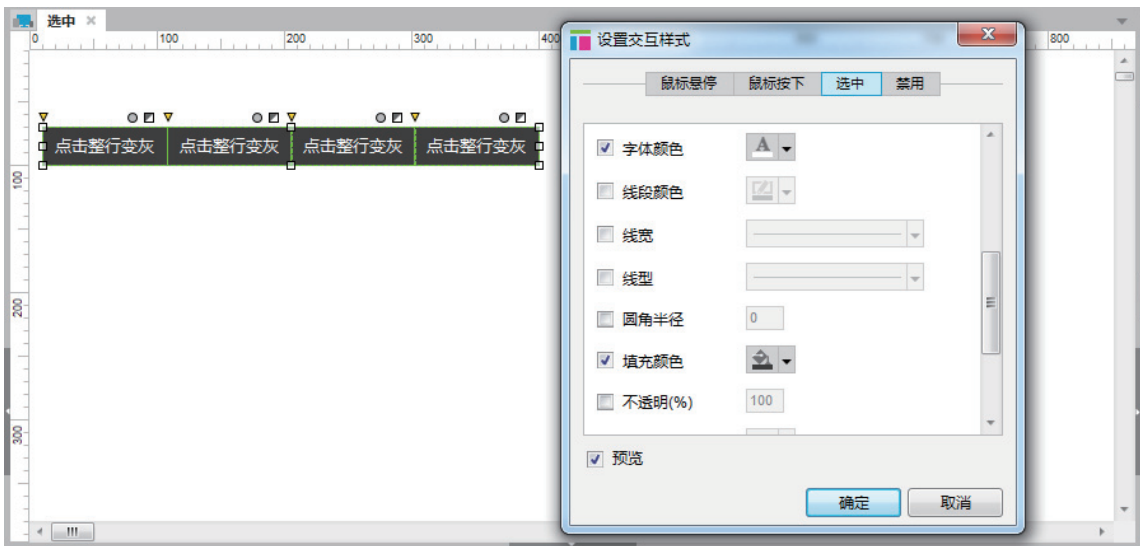


图4-30

步骤 2 选取所有矩形，单击鼠标右键，在菜单中选择【转换为动态面板】；并且，为动态面板添加【鼠标单击时】事件的用例动作为【选中】【当前元件】。（图4-31）

6. 选项组名称设置

与其他元件的选项组功能一样，能够实现唯一被选中的效果。例如将上个案例“单击选中整行”中的动态面板多复制几个，然后给它们在属性中添加相同的选项组名称就实现了唯一一行被选中的效果。

7. 元件提示

这个功能很少使用，在浏览器浏览原型时，能够让元件在鼠标移入时显示一个文字提示。

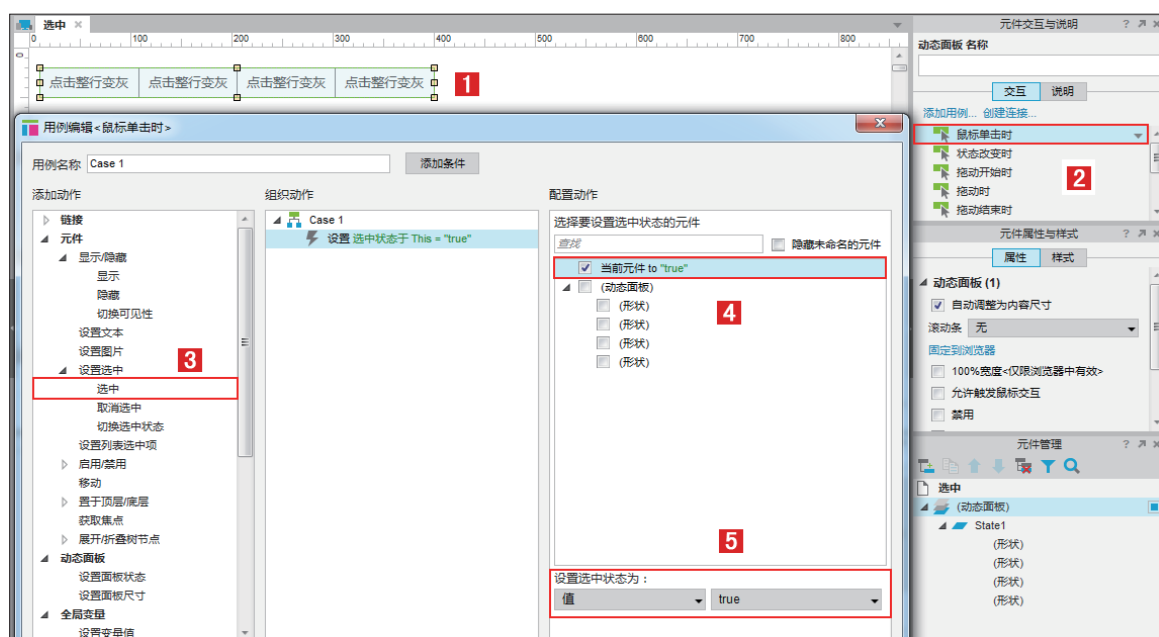


图 4-31

本章任务 → 牢记动态面板的多种特性，完成本章内各个案例并应用这些特性。

购买本图书阅读更多内容，请访问<http://www.iaxure.com/3384.html>。

【[点击此处](#) . 马上购买】