**Axure中级互动设计 1——控制Dynamic Panel**

控制Dynamic Panel  
1. Axure RP实现更丰富的互动设计  
2. 认识Dynamic Panel Widget  
3. 编辑Dynamic Panel 的状态  
4. 预设Dynamic Panel的显示/隐藏  
5. 示范-以Click来开关Dynamic Panel  
6. 教学影片

1. Axure RP实现更丰富的互动设计  
网站开发技术越来越丰富了，这是我们正在经历的过程，也是未来的趋势。  
不管是AJAX或JavaScript，甚至是 Flash、Silverlight、JavaFX，这些都被归类为RIA(Rich Internet Application)的技术，能够让网站介面更丰富，展现更优质的人机互动。   
同时，这个趋势也挑战了传统的文书软体或绘图软体，当这些软体受限于仅能表现单一画面与单一状态的情况下，您很难把心目中最友善的互动方式表达清楚并设计出来。  
Axure RP与其他的wireframe软体，最大的差异在于能够进行互动设计(Interaction Design)，且可以立即将结果呈现于prototype上，而不是只有文字叙述。更棒的一点是，学习Axure RP的互动设计，您并不需要学习HTML或Java Script语法，只要透过一些设定与操作，便可以完成多样而创新的互动介面设计。  
在Axure RP设计出来的Prototype中，最简单的互动设计是网页连结，透过滑鼠的click来串起使用者与网站的互动流程。然而，真正发挥Axure RP在互动设计上的惊人特色，并不是只有连结这么简单的互动，您可以藉由学习Dynamic Panel (动态面板)物件的操作与控制，来做到更丰富更友善的互动介面设计。  
2. 认识Dynamic Panel Widget  
Dynamic Panel (动态面板) 这种物件是专门用在设计Prototype动态功能的Widget，Dynamic Panel可以包含一或多个状态(State)，每个状态本身是一个小页面，藉由控制状态(State)的顺序或隐藏/显示Dynamic Panel来达成互动介面的整个表现。  
就像其他Widget一样，Dynamic Panel可以用拖放的方式从Wireframe窗格加入到画布中。

学习Dynamic Panel的互动设计之前，有2个重要的概念必须先认识：  
(1) Dynamic Panel (动态面板)：  
一种透明的物件，本身可以包含很多个状态 (State)，而每个状态都是一个小网页。被放置在最上层的状态就是该面板的长相。  
(2) State (状态)：  
每个状态都是一个小网页，而这个小网页的边界与大小，与所属的Dynamic Panel大小相同。不同的State可以重叠在同一个Dynamic Panel里头，唯有被控制放在最上层的State，才会呈现于Prototype的画面中。  
以Axure RP所设计出来的多样互动介面，大多藉由各种触发事件(Event)来控制Dynamic Panel显示或消失，或控制哪个State在最上层来模拟出实际的互动介面。  
3. 编辑Dynamic Panel 的State(状态)  
已经摆放到画布中的Dynamic Panel，直接在上头快速点滑鼠左键两下，会开启「Dynamic Panel State Manager」对话方块。在这个对话视窗中，您可以新增State、更改 State名称、调整State排序、移除State与编辑State。

在「Dynamic Panel State Manager」对话方块中选择一个Panel State，然后按下【Edit State】钮，会开启这个Panel State的Wireframe以供设计。或者您可以选择【Edit All States】钮，一口气把所有的状态页面通通开启。  
开启后，您就可以像设计其他Wireframe一样的设计状态页面(State)， 每个状态页面的侧边蓝色虚线外框就表示Dynamic Panel 的边界。  
4. 预设Dynamic Panel的显示/隐藏

Dynamic Panel 可以预设为隐藏(Hidden)，作法是在Dynamic Panel 物件上按滑鼠右键，并选择「Edit Dynamic Panel->Set Hidden」，这样就可以隐藏Panel 的内容，而且Dynamic Panel的遮罩也会从蓝色变成黄色。已经预设隐藏的Dynamic Panel，可以选择「Edit Dynamic Panel->Set Visible」来显示Dynamic Panel。  
Dynamic Panel 可以藉由接下来所介绍的互动模式，动态控制在Prototype 中的显示或隐藏。 Axure RP提供互动设计的实现方式，都在「Interaction Case Properties」对话方块中设定(请参考初级互动设计)。  
其中有5种互动方式(Action)是专门用来控制 Dynamic Panel的，分别是:  
\* Set Panel state(s) to State(s)：将某个Dynamic Panel的State设定为该Panel的显示状态  
\* Show Panel(s)：显示(设为visible) 一或多个Dynamic Panel  
\* Hide Panel(s)：隐藏一或多个Dynamic Panel  
\* Toggle Visibility for Panel(s)：根据目前的显示状态来显示或隐藏Dynamic Panel  
\* Move Panel(s)：移动Dynamic Panel，可根据绝对座标或相对座标来移动  
5. 示范-以Click来开关Dynamic Panel

要了解Dynamic Panel的互动控制，我们先来模拟一个Click的效果- 当滑鼠Click页面上的矩形时，会出现一个讯息视窗，再Click一次时，又把讯息视窗隐藏起来。换句话说，我们将藉由滑鼠Click来控制 Dynamic Panel的显示与隐藏。  
首先我们先在画布上放一个矩形物件。接下来放置一个Dynamic Panel，并且编辑这个Panel唯一的预设状态(State1)。

在 State1(状态)的wireframe上，随意加入一些文字，以供辨认。

再把 Dynamic Panel的显示状态，设定为隐藏(Hidden)。

然后，在矩形物件上增加一个OnClick的Interaction Case。先选择矩形物件，接着click两下“OnClick”，弹出「Interaction Case Properties」互动设计的对话方块。  
分别针对这个对话方块中的 Step 1- 3做互动设计的设定：  
step 1: Description - 预设为Case 1，不用变更。  
step 2: Select Actions - 勾选Toggle Visibility for Panel(s)，此时在step 3会出现Toggle Visibility for Panel。如下图：

接下来…  
step 3: Edit the Action description (click an underlined value to edit) - 这句话的意思是，去点选有加底线的项目进一步编辑。于是我们接着click Panel这个单字，然后会看到弹出一个「Select Panels」对话方块，里头只有一个叫做”My First Panel”的项目，把它勾选起来。此时在step 3会出现Toggle Visibility for My First Panel。 (如下图)

如果您曾经给Dynamic Panel有意义的名称，就可以在Select Panel时看到被您命名过的Panel。选择Dynamic Panel时，Dynamic Panel 会以Annotations & Interactions 窗格中所给予的Label做为识别，如果没有指定的话，Dynamic Panel 会预设标示为”Unlabeled”。  
接下来，就可以输出Prototype了，按下F5或选择「Generate > Prototype」，将刚刚设计过的简易互动介面，输出到浏览器上进一步检视，当滑鼠click时，您所设计的那个Dynamic Panel是否会消失/出现(如下图)。

当您顺利完成这个 Dynamic Panel的设定之后，非常恭喜！您已经成功进阶到Axure RP中级互动设计了，随着Dynamic Panel的各种互动设计技巧的熟练，您将越来越能随心所欲地设计出各种丰富又兼具创意的互动介面。

Axure中级互动设计2—鼠标移入移出等的互动设计

1. 滑鼠移入移出(OnMouseEnter/OnMouseOut)

2. 变换图像(Rollover Image)

3. 变换样式(Rollover Style)

1. 滑鼠移入移出(OnMouseEnter/OnMouseOut)

在先前初级互动设计(Basic Interaction)一章中，介绍多种Axure RP支援的人机介面互动效果。这里我们要介绍其中两个常见的触发事件 (Event)：

OnMouseEnter - 滑鼠的指标移动到物件之上

OnMouseOut - 滑鼠的指标移动出物件之外

有许多Widget可以使用OnMouseEnter 和OnMouseOut触发事件，当滑鼠移到Widget上时会触发OnMouseEnter事件，OnMouseOut事件则发生在滑鼠离开 Widget 时被触发。

最常见的运用方式，则是合并Dynamic Panel来控制比较复杂的功能时，例如：浮动选单、滑过特效和自订tooltips等。

以左图的互动效果来当例子：当滑鼠移到图片上方时，会自动弹出说明文字框(Dynamic Panel)，当滑鼠移出图片时，说明文字框就会消失。这个互动设计就可透过OnMouseEnter 和OnMouseOut触发事件结合Dynamic Panel的控制来达成，设计方式如下图所示。

2. 变换图像(Rollover Image)

Image Widget 以及Button Shape Widget可直接利用Axure RP的功能来设计Rollover Image和Rollover Style，不需要使用到OnMouseEnter、OnMouseOut和Dynamic Panel。

变换图像(Rollover Image)

想要建立变换影像的话，请在Image Widget上按右键，选择「Edit Image->Import Rollover Image」，然后选择一个影像当作变换影像，一旦影像选定后，您可以在Wireframe中，利用滑鼠滑过影像上的黑白色方块来预览变换影像。

3. 变换样式(Rollover Style)

在Axure RP里头，Rectangle，Placeholder，Button Shape这三种Widget可以直接设定变换样式(Rollover Style)，而不需要去特别指定OnMouseEnter / OnMouseOut的触发事件，就可以做到变换样式。

想要建立变换样式的话，请在Rectangle / Placeholder / Button Shape上按右键，选择「Edit Edit Button Shape->Edit Rollover Style」。

此时会开启「Set Rollover Style」对话方块，您可以在这里选择变换样式，勾选「Preview」核取方块来预览设定在Button Shape的变换样式。

变换的样式(如下图)包括：

\* 字型 Font

\* 字体大小 Font Size

\* 粗体 Bold

\* 斜体 Italic

\* 底线 Underline

\* 文字颜色 Font Color

\* 填色 Fill Color

\* 线条颜色 Line Color

\* 线宽 Line Width

\* 线条样式 Line Style

套用了变换样式之后，您可以选择 Preview提早预览效果。也可以在Wireframe中，利用滑鼠滑过Widget左上角的黑白色方块来预览变换状态。

Axure中级互动设计3—设计多层选单

设计多层选单  
1. 认识选单物件 (Menu Widget)  
2. 编辑选单物件  
3. 设定选单的变换样式 (Rollover Style)  
4. 秘技 (Quick Tips)  
1. 认识选单物件 (Menu Widget)  
选单物件(Menu widgets) 常被用来建立简单的多层选单(Flyout Menus)。

您可以在 Widgets窗格中找到两种选单物件：  
\* Menu (Vertical) - 垂直选单  
\* Menu (Horizontal) - 水平选单  
将你所需要的选单物件从Widgets窗格中拖曳到Wireframe就可轻松建立，新的选单会先设置3个选单选项(Menu Item)。  
下图展示的动画则是已经完成设定的多层选单：

2. 编辑选单物件  
编辑选单和选单项目的文字与格式的方法就像编辑其他Widget一样，您可以在选单项目上连续按滑鼠左键两下编辑文字，也可以利用工具列来编辑选单和选单项目的色彩、字型和其他样式。  
选单物件的编辑功能有这些项目：  
\* Edit Menu Padding - 选项间距设定  
\* Edit Rollover Style - 设定变换样式  
\* Add Menu Item After - 往后新增选项  
\* Add Menu Item Before - 往前新增选项  
\* Delete Menu Item - 删除选项  
\* Add Submenu - 新增子选单(如果已经有子选单，则会显示Delete Submenu - 删除子选单)  
在Menu Widget的选项上，按滑鼠右键会出现如下图功能选单：

增加或移除选单项及子选单  
想要增加或移除选单项目的话，请在选单项目上按滑鼠右键，选择「Add Menu Item」和「Delete Menu Item」指令。  
想要在选单项目下加入子选单的话，请在选单项目上按滑鼠右键，选择「Add Submenu」。在Axure RP的画布区域，通常Submenu会自动缩回，要去点一下上层选单，才能再看到前新增的Submenu。如下动画：

设定选项间距 (Padding)  
若是想要编辑选单项目之间的间距(Padding)，请在选单或选单项目上按滑鼠右键，选择「Edit Menu Padding」。

3. 设定选单的变换样式 (Rollover Style)  
就像Button Shape Widget一样，变换样式(Rollover Style)也可套用在Menu Widget的选单项目上。  
想要编辑选单的变换样式，请在选单或选单项目上按滑鼠右键，选择「Edit Rollover Style」，此时会开启「Set Rollover Style」对话方块，您可以在这里选择单一选单项目、选单上的所有选单项目，或是选单和子选单上的所有选单项目的变换样式。  
勾选「Preview」核取方块来预览套用在选单上的变换样式。  
套用了变换样式之后，您可以在 Wireframe中，利用滑鼠滑过选单项目左上角的黑白色方块来预览变换样式。

4. 秘技 (Quick Tips)  
秘技1. 优先设定母选单(Root Menu)的样式：  
当子选单(Submenu)建立了以后，会自动套用母选单(Root Menu)的样式，您可以在建立子选单之前先将母选单样式设好，以节省时间。  
(版权声明：欢迎引用及复制Axure RP 中文教学内容，必须注明出处为悠识数位顾问有限公司及原始文章网址UserXper.com)

Axure高级互动设计1—设定逻辑条件

设定逻辑条件  
1. 认识Condition (逻辑判断)  
2. 认识Condition Builder (逻辑判断编辑器)  
3. 用Condition Builder建立逻辑判断  
4. 秘技(Quick Tips)

1. 认识Condition (逻辑判断)  
虽然可以用文字来描述Prototype操作的动线或互动方式，但有的时候，您还是会需要一个更逼真(High Fidelity)、更互动、更有亲和力的Prototype 。比如，当您在操作Prototype，模拟会员登入帐号，如果能在按下登入按钮时，直接显示登入成功或失败的讯息。是不是比您口头解说或文字描述，还要来得清楚明白呢？  
在Axure RP的互动设计中，「逻辑判断」是最有挑战性，但是也最有趣的一件事情。因为当您在设计网站Prototype时，可以不用寻求程式设计师或Flash设计师的协助，就可以独立完成您想要表达的互动效果。  
清楚的逻辑观念胜过互动设计技巧  
学习Axure RP的条件逻辑设计之前，您必须先知道一个重要的观念：  
成为称职的网站企划人员，不一定要学会Axure RP的高级互动设计技巧，却一定要有清楚的逻辑观念。否则，不只造成网站开发工作上的困扰，更严重的还会造成糟糕的网站使用者经验。只有清楚的逻辑观念，才能带来准确合理的互动设计。

如果您不想学这些高级互动设计技巧，绝对无损于您利用 Axure RP进行网站企划及Prototype 设计。换言之，即使您学会Axure RP的高级互动设计，但是您的逻辑观念如果不清楚，仍然无法成为一位优秀的网站企划。  
要是您希望将Prototype拿来进行 Usability Test，或者在各种沟通讨论会议上，让大家更充分理解操作的互动效果，那么透过Axure RP提供的逻辑判断设定，可以让Prototype更进一步接近真实完成的网站介面与动线。  
物件值(Value of Widget) / 变数值(Value of Variable)  
接下来，您要学习使用变数值/物件值，这两种类型的数值应用于逻辑判断：

a. 物件值(Value of Widget) - 使用Widget本身的值来进行逻辑判断，有4种不同的数值可供应用：  
\* Checkbox或Radio Button的核取状态  
\* Droplist 或 Listbox的选取值  
\* Text Field或Text Area中的文字  
\* Text Field或Text Area的文字的长度  
b. 变数值(Value of Variable) - 使用自定变数的值来进行逻辑判断，有2种：  
\* value of variable - 自定变数的值  
\* length of variable value - 自定变数值的长度  
附带补充，Axure RP本身存在一个全域预设变数(Global Default Variable) - OnLoadVariable，除此之外，变数名称可以自订，最多可以有25个自订变数。您可以透过主选单Wireframe > Manage Variables来检视并编辑所有的自订变数。

您可以在Prototype设计里，将物件值及变数值的逻辑判断加入互动设计，在操作Prototype时，将会根据Web Form Widget所输入的数值，或是变数的值来进行判断。比如，当使用者登入帐号时，自动检查输入的帐号或密码，登入成功则导到欢迎页面，登入失败则显示错误讯息。  
表单型态的物件 (Web Form Widgets)  
由于逻辑判断牵涉到不同物件值的判断，因此经常与表单型态的物件结合运用。 Axure RP的表单型态的物件(Web Form Widgets) 如下图。

2. 认识Condition Builder (逻辑判断编辑器)  
如果您具有程式设计的基础，要学习 Axure RP的变数或逻辑判断，一点都不困难。毕竟Axure RP不是程式开发工具，并没有太多的程式语言要学习。  
为了让没有程式设计基础的 Axure RP使用者也可以设计逻辑判断，因此Axure RP透过三个不同的功能视窗，来协助使用者设定变数以及逻辑判断，这些功能视窗包括：

\* Annotation and Interactions (互动设计)  
\* Interaction Case Properties (互动状况设定器)  
\* Condition Builder (逻辑判断编辑器)  
这三个功能视窗的关连及开启顺序如下图：  
为了让没有程式设计基础的Axure RP使用者也可以设计逻辑判断，因此Axure RP透过三个不同的功能视窗，来协助使用者设定变数以及逻辑判断，这些功能视窗包括：

其中，Annotation and Interactions，以及，Interaction Case Properties，这二个视窗在学习初级互动设计就使用到了。只有Condition Builder对话窗对于学习高级互动设计的人是陌生的，我们就来认识一下这个特殊的功能。  
Condition Builder逻辑判断编辑器的操作介面  
Condition Builder是用来设定逻辑判断的工具，只要透过下拉选单及一些按键，就可以完成一长串复杂的逻辑判断叙述，不需记忆，也不用自行撰写逻辑判断程式。  
您可以在Interactions Case Properties视窗的右上角，点一下Add Condition (或Edit Condition)的文字连结，就会开启「Condition Builder」视窗。  
在「Condition Builder」视窗中，可以建立一行或多行的逻辑判断，只要按下右方的「 」钮，就可以增加一行，按下「-」钮，则会移除一行。  
如果所有的逻辑判断都必须满足， 可以在「Satisfy」下拉选单中选择「Satisfy all of the following」，就会以”and”来连结各个逻辑判断；若是选择「any」，则会以”or ” 来连结。

在逻辑判断中，检查值共有六种型态：  
\* check state of - Checkbox或Radio Button Widget的核取状态  
\* selected option of - Droplist 或Listbox Widget的选取值  
\* value of variable - 自定变数的值  
\* length of variable value - 自定变数值的长度  
\* text on widget - 输入在Text Field或Text Area中的文字  
\* length of widget value - 输入在Text Field或Text Area中的文字长度  
3. 以Condition Builder建立逻辑判断  
您可以自行建立一个Case的逻辑判断，比如：  
If text on Widget 帐号栏位equals ‘guest’ and text on Widget 密码栏位equals ‘12345′ “  
（如果「帐号栏位」输入的是’guest’，而且「密码栏位」输入的是“12345”）  
这个逻辑判断将会在您操作Prototype时，自动判断是否要执行Case所指定的动作。  
完成逻辑判断的设定之后，按下【OK】钮回到「Interaction Case Properties」视窗，就能指定当符合这个Case的逻辑判断时，要执行哪些动作。  
注意：当您增加Case的逻辑判断，Interactions窗格中的Case名称后面会自动加入一个以“If” 开头的逻辑判断描述。同时，这个Case的图示会变成绿色，强调这个 Case使用了逻辑判断。

一旦在 Case中定义了逻辑判断，同一个触发事件的其他Case名称后面，也会自动加入以“Else If” 开头的逻辑判断描述。如果这些Case没有另外设定逻辑判断的话，那么就会直接标示“Else If True”。  
在预设的情况下，Prototype只会执行第一个标示为”True” 的Case。

在Interactions 窗格中的案例上按下右键并选择「Change to IF」，可以将“Else If” 改为“If” ，这么做将可允许在多个状况同时成立的情况下执行多个动作。  
4. 秘技 Quick Tips  
秘技 1. 为Widget命名：  
为逻辑判断使用到的Widget命名，可以方便在建立逻辑判断时做识别。想要替Widget命名的话，只要选择Widget 并在Annotations & Interactions 窗格中的「Label」栏位输入名称就可以了。  
秘技 2. 绘制流程图：  
逻辑判断和流程可以设计得十分复杂，建立一个流程图 (Flowchart)可以帮助您组织与传达使用者可能涉及的各种动线。  
(版权声明：欢迎引用及复制Axure RP 中文教学内容，必须注明出处为悠识数位顾问有限公司及原始文章网址UserXper.com)

Axure高级互动设计2——示范: 登入的逻辑判断

逻辑判断示范  
以旅游网站的企划 (http://userxper.com/blog/archives/36 ) 为例，说明「会员登入页面」如何应用Condition Builder建立逻辑判断。  
这个会员登入要做到的功能是：如果登入成功（即帐号、密码输入正确），则导回首页，并显示招呼讯息。如果登入失败（即帐号、密码输入不正确），则出现错误讯息。  
来到旅游网站Prototype的「范例2-进阶版首页」，点一下”登入”按钮，就进入会员登入的页面。设计以下两种登入的操作状况：   
状况1 输入错误的密码：出现登入错误的讯息。  
状况2 输入正确的密码：登入成功之后，立即导回首页，并显示招呼讯息，同时将登入的文字连结变成登出。

您可以用Axure RP绘制流程图(如下图)，来解释这个功能设计的逻辑判断。

接下来，开始设计页面的操作互动：  
1) Widget命名：针对操作相关的栏位、按钮、动态面板，在Interactions窗格的Label一一填入名称，以利后续的设定识别。

2) 新增Case：针对”登入”这个Button Widget，在Interactions窗格中新增OnClick (点击时触发) 的Case，并在「Interaction Case Properties」视窗中点一下「Add Condition…」连结。

3) 设定Condition：在开启的「Condition Builder」视窗中，您可以使用下拉选单建立如下的逻辑判断：

当您在设定时，下方的Descritpion区块会自动产生逻辑判断的描述：  
If text on widget Username Field equals “UserXper”  
and text on widget Password Field equals “12345”  
（如果「Username Field」输入的是“UserXper”，而且「Password Field」输入的是“12345”）。  
4) 选择Action：按下「Condition Builder」视窗的【OK】钮，回到「Interaction Case Properties」，再选择当符合这个逻辑判断时，所要执行的动作。比如：导回网站首页、显示招呼语，登入的文字连结变为登出。  
当您设定了Case的逻辑判断，您会发现 Interactions窗格中的Case名称后面会自动加入一个以“If” 开头的逻辑判断描述。同时，这个Case的图示会变成绿色，强调这个 Case使用了逻辑判断。  
Axure RP Interactions De.ion  
最后，实际完成的Prototype互动设计如下图动画：

Axure之锚点跳转与Title提示

最近在做一个原型的时候用到锚点跳转与Title提示功能，同事看到HTML之后问我这个是怎么实现的；昨天又看到一篇文章举了个注册表单中的Title 提示示例，问这个是如何实现的，我才发现原来很多人都忽视了axure中一个很常用的组件——图片/图像映射区域（image map region）。

       组件名称：image map region，图片/图像映射区域

       组件介绍：在web页面中制造一片不可见的区域，相当于图片的热区，从而添加说明与互动。在现实的网页中经常会有一张大图中有某个区域是触发按钮，而图像映射区就可以用来说明这个区域的的功能和互动内容。

      可实现功能：链接的Title提示、图片的Alt提示、锚点跳转。类动态面板功能。

       1、Title提示和Alt提示功能实现  
  
       链接的Title提示和图片的Alt提示功能相对比较简单，其实只要简单说一下大家就都能明白了，只是平时没有注意罢了。                                         
       如上图所示，选择图像映射区域组件，覆盖在要映射的图像上方，右键点击“编辑工具提示”，输入Title的文本内容，OK。

       其中，是否给该控件加注标签为可选项，如果不加注标签，则系统默认名称为“unlabeled”，不过，如果你需要使用锚点跳转功能的话则需要加注标签，中英文皆可。因为你要指定锚点的跳转位置，这个位置的判定就是以该标签的名称为准的。

        PS：ALt和Title提示这个功能也可以用动态面板来实现，不过比较笨重，需要设置mouse on和mouse out两个交互动作。

        2、同一页面内锚点跳转

      如上图所示，先在要加入锚点的地方添加一个图像映射区域（操作方法如例1），这个图像映射区域必须要加注标签；在要实现跳转的链接上加到交互动作，动作选 择：滚动到图片映射区域；然后选择你要跳到的锚点位置，在下拉菜单中选择你刚才加注的那个标签名称；可以考虑选择滚动的方式；OK。

     其中，必须要对图像映射区域加注标签；页面的长度一定要足够锚点跳转，如果页面不够长他跳了你也看不到。

     3、不同页面之间的锚点跳转

        如上图所示，这里包含了2个方面的交互：A页面的组件a的交互、B页面（跳转到的页面）的页面交互。这个过程中我们需要借助一个中间变量（新增的变量1）。

        点击线框图，管理变量，新增一个变量（例如1）——给锚点链接添加交互属性，也就是在离开页面A的时候需要先给变量赋值——在跳转到的页面（B）中添加页面交互（在OnPageLoad中添加）：滚动到图片映射区域【注意，不是组件交互而是页面交互！】，OK。

       这个过程的流程是这样的：点击链接a的时候也带入了一个变量1，当a动作完成后，如果在B页面还存在满足变量1的交互动作则继续执行，滚动到图像映射区域。

       4、不同页面之间的切换层状态锚点跳转  
  
       这个需求是这样的，点击页面A中的链接a跳转到页面B中Tab模块T中的第二个tab 2中。

       实际实现原理还是一样的，只不过是在对页面B的页面交互条件中再添加一个“设置动态面板状态跳转”就可以了，此时的条件变成同时满足：变量值为1 and 设置动态面板状态转换到T的tab2 and 滚动到图片映射区域。