1. URL

框架的URL分为2情况：

1.只表示客户端资源：/0-1，代表着$scope.urls的索引，根据这个来获取该数组的某个元素。

这种情况表示要以tag形式展示内容。

2.服务器段资源部分:/0/user/5/page/50

1)基础部分：/0/user

2)条件部分：基础部分之后的所有部分

如果一个URL只有客户端资源而无服务器资源，则表示在tab或dialog中以tag方式展示内容。

根据该URL获取数组元素，如果该元素定义了：target:”dialog”，则在dialog中以tag方式展示内容。

1. 链接与tab

框架加载内容有3中方式：tab、dialog、局部div。

程序在开始时会 创建一个$scope.tabs数组、一个$scope.dialogs数组，分别用来存放所有tab和dialog的基本信息，主要是每个的tab或dialog的模板路径。它们的键名是每个URL的基础部分。而具体的数据则存放在数组的data字段中。

index.html模板中会迭代这2个数组就会展示tab和dialog，同时也不停地监听着这2个数组，从而实时的更新视图。

对于URL有2种处理方式：

2.每次操作只设置$location.path()，其它操作由监听URL变化的部分来完成。

tag应该作为同一数据的局部对比等，而不是像tab那样。右侧只展示数据，不组织结构。

这种方式比较理想，但实现起来比较困难，主要是刷新页面后，要区分当前URL对应的是tab还是dialog，还是里面的tag，还是table的tr。

解决办法：

“查询”:例：/0/user/status/1/&role。当条件部分改变时，获取数据后要更新$scope[target][“/0/user/”]的值；

“添加”：将URL更改成原URL的基础部分，例：/0/user/status/1改成/0/user，获取的数据存放在$scope.dialogs[“0/user/”]

“编辑”：例：/0/user/5。这是的URL只是要更新的资源的地址，同时还要使用该URL 来 获取对应数据。

“删除”：/0/user/5，只是一个状态而RESTful的URL是不能表示状态的，所以不更改URL，只为其添加ng-click=””属性。

数据列表部分的按钮，如“添加”、“编辑”等点击时有2中展示方式：一种是直接在表格里面增加一个新行，这种只要给对应list添加一个元素即可。使用ng-tr指令；二是在dialog中。具体采用那种方式就看使用哪个指令了，

以上2中方式接收到数据后，要更新对应tab或tag或dialog的data字段。

/0/1/put：这里的edit是名词，表示

另外，还有一个$scope.lists数组，

所有需要在tab或dialog中打开的内容均存放在数组$scope.menus当中，迭代这个数组即可输出左侧菜单。

点击菜单的链接时，会监听URL的变化，同时，，然后进行处理。迭代这个数组就可以输出当前tab及其中的tag。之所以创建这个数组而直接不使用$scope.menus是因为，

对URL是这样处理的，URL分成了2部分：前面一部分表示是用来确定在$scope中某个元素的数组的每一级的索引，通过这些索引找到这个元素（就是tab变量）后就获取到了对应的模板的路径。根据该元素是不是数组来决定输出tag还是直接输出tab内容。

输出菜单分两种情况：

1)保留每个的最底层数据，让每个的最底层数据以tag的形式显示在tab中

2)输出所有数据

这是，tab中就没有了tag了。

1. 对于页面只能同时存在一个的视图，比如右键菜单等，直接用

<div id=”contextMenu” ng-view ></div>

实现；注：右键菜单可以在移动平台很好的使用，因为长按屏幕就相当于右击。

1. 对于即将打开的div，应该先给其动态添加控制器（就是在ng-repeat之前），然后再显示（或再compile()对应的HTML）。
2. 使用ReqireJS进行按需加载