1.什么SAP UI5、OPEN UI5？

SAP UI5是SAP UI Development Toolkit for HTML5的简称，它是一套基于HTML5的UI开发组件，是SAP为了快速开发前述的Fiori风格的UI、构建企业级产品的javascript框架， 拥有丰富的界面控件、CSS页面展示模板、上百个工业图标，同时支持控件的扩展。

那什么又是OPEN UI5？SAP基于Apache 2.0协议将这套控件贡献给开源社区，形成的开源版本叫做OPEN UI5，基本上这套版本主要还是由SAP更新，而SAP UI5和OPEN UI5的功能是 比较接近的，或者说OPEN UI5相比于SAP UI5来说，重要功能基本没有减少。

我们接下来不做区分，统称UI5，而由于OPEN UI5是开源的，所以基本上以后用到的库都会是基于OPEN UI5。

2.OPEN UI5 SDK

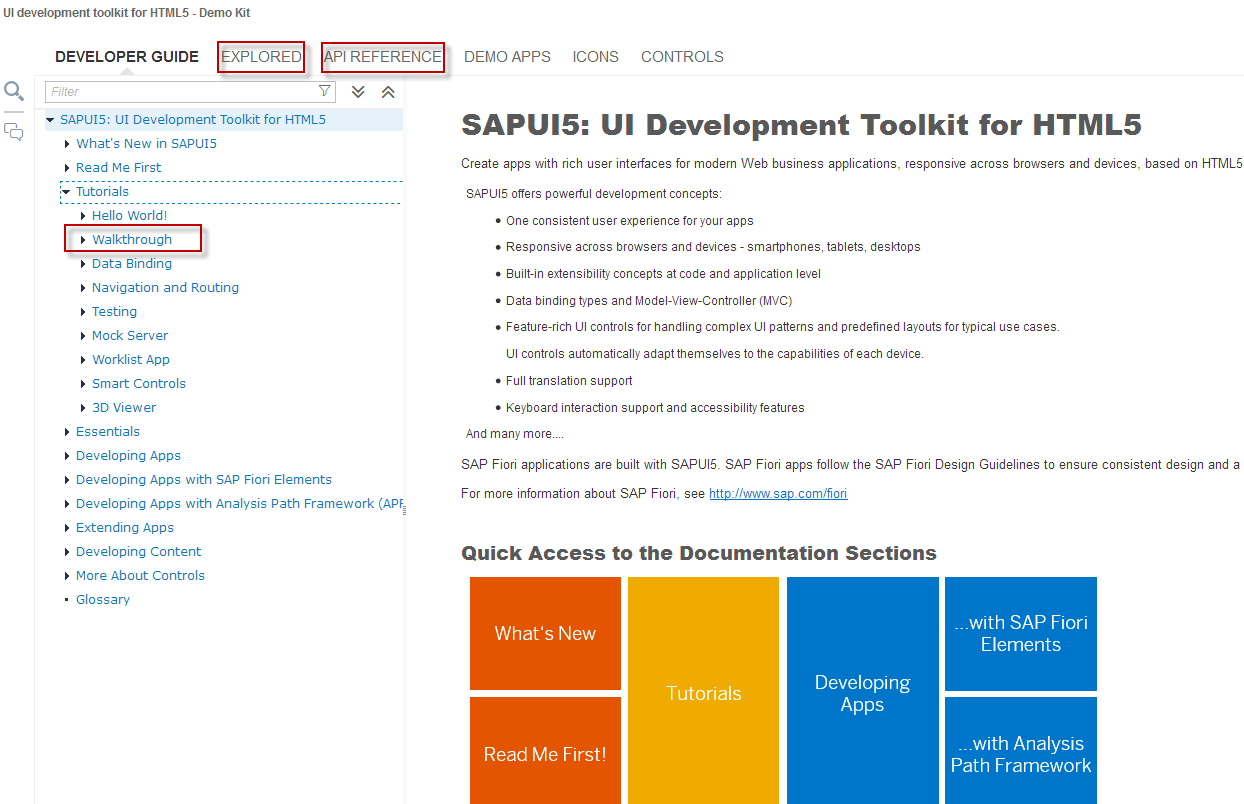
<http://openui5.org/download.html>

本地放在<tomcat path>/webapps/openui5 ，此时打开<http://localhost:8080/openui5>

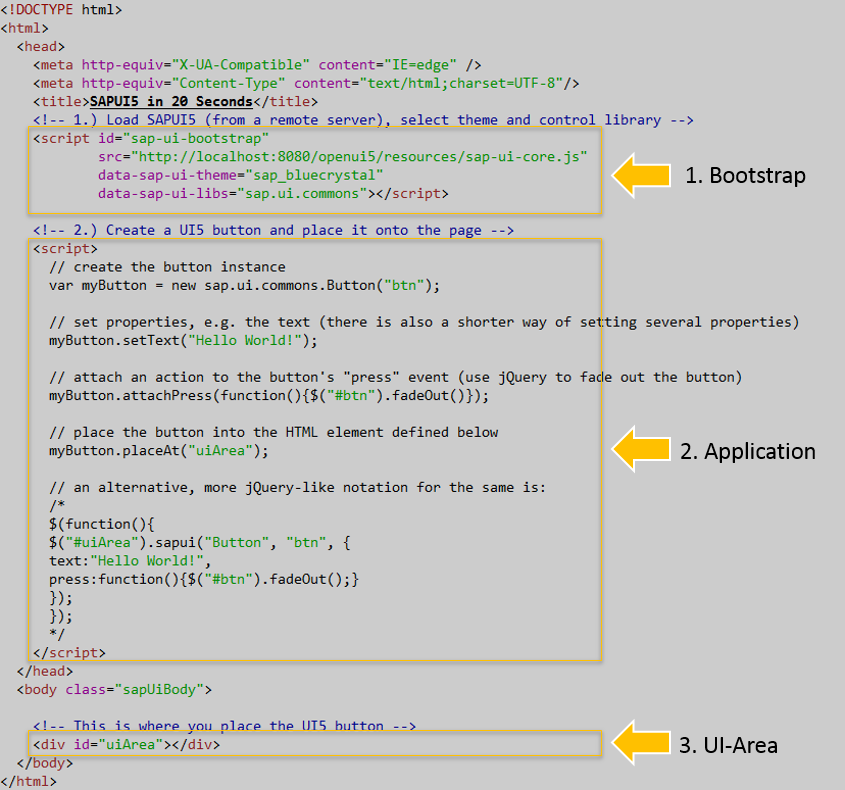
或者在线看：

<http://veui5infra.dhcp.wdf.sap.corp:8080/demokit/#docs/guide/95d113be50ae40d5b0b562b84d715227.html>

<https://sapui5.hana.ondemand.com/>



1. 基本程序结构(Index.html)



Bootstrap是一段嵌入的javascript代码，首先引入了UI5的核心运行时库sap-ui-core.js，接下来代码通过标签设定了两个属性，分别是主题： data-sap-ui-theme ，以及在应用中将会引用到的UI库： data-sap-ui-lib。

Application，我们的应用的主体就在这里了，包含的是如下代码段.这段代码也很容易理解，首先创建一个button的对象，然后设置button对象显示的文本为"Hello World!"，接着为这个按钮注册一个点击事件——按下去的时候按钮自己会消失，最后把这个按钮放在一个叫做uiArea的地方.

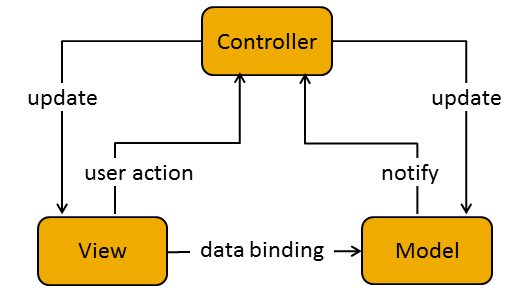
UI-AREA,一个div，并将其id设置为我们之前所用到的uiArea。

第一部分Bootstrap引入运行环境以及一些常用配置文件，第二部分Application中加入HTML控件以及相应的后台数据和业务逻辑，并将其放入到第三部分UI-AREA中。

引用本地：src="http://localhost:8080/openui5/resources/sap-ui-core.js"

引用在线：src="./resources/sap-ui-core.js"

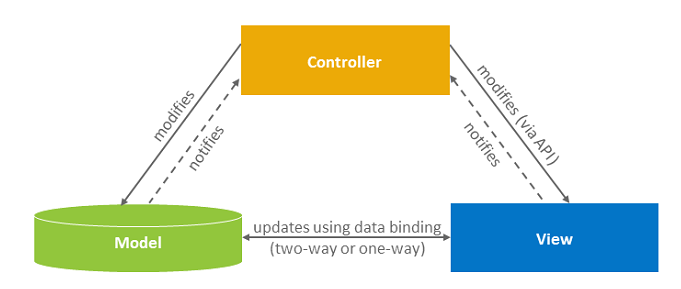
1. MVC基础架构模型



* M 代表Model - 模型   
  一般用来管理数据层，比如绑定后台数据。
* V 代表View - 视图   
  一般用来处理展示层，比如具体前端UI的展示。
* C 代表Controller - 控制器   
  一般用来处理逻辑，可是页面逻辑也可以是协调处理数据的逻辑

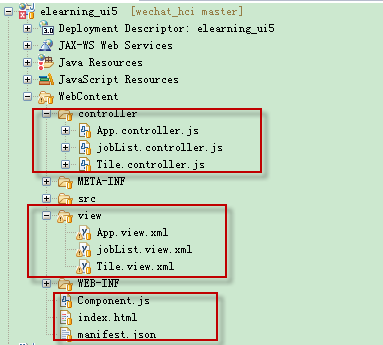
MVC最主要的目的是把展示与逻辑、数据分离开来，使得程序更容易阅读、容易理解，从而也更可以维护， 同时，也增加了可扩展性。

视图和控制器一般是1:1对应的，但是也可以创建一个没有视图的控制器，这样的控制器叫做应用控制器， Application Controller ；同时，也可以创建一个没有控制器的视图。



* Model和View之间的关系：单向绑定或者双向绑定，以及通过绑定更新Model中的数据。（单向绑定：单向绑定意味着模型到视图的绑定；模型中的数据变化将更新相应的绑定和视图。双向绑定：双向绑定意味着模型到视图及视图到模型的绑定；模型/视图中的数据变化将更新相应的绑定和视图/模型。）
* Controller和View之间的关系：View通知Controller，或者Controller使用API来修改View。
* Controller和Model的关系：Model通知Controller，或者Controller修改Model。

1. Case



### WEB-INF是Java的WEB应用的安全目录

Meta-inf: 相当于一个信息包，目录中的文件和目录获得Java 2平台的认可与解释，用来配置应用程序、扩展程序、类加载器和服务

在SAP Fiori应用中，每个应用都有一个配置文件即manifest.json，里面定义了一些列的项目配置信息。可以看到，manifest.json文件定义了包括ui5版本、数据模型等一系列基本信息。在以后的开发过程中该配置文件会被不断完善。注意manifest文件在一个应用中的重要性，manifest.json是app的配置文件。

（1）只有index.html



①引入相关的SAP UI5库文件

②html标签编辑组件的盛放容器：

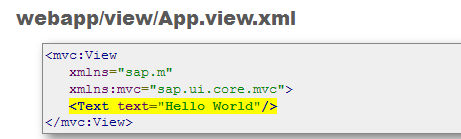
<body class="sapUiBody" id="content"></body>

③编写相关的js代码（这个以后要放到单独的文件中）

在这里我们使用SAP UI5中的control，定义了一个Text。Text组件在sap.m中，所以我们通过sap.m.Text的方法创建一个新的Text。通过text属性设定其内容为Hello World。最后利用placeAt方法，将它放置到content中。这就是最简单的Hello world程序。



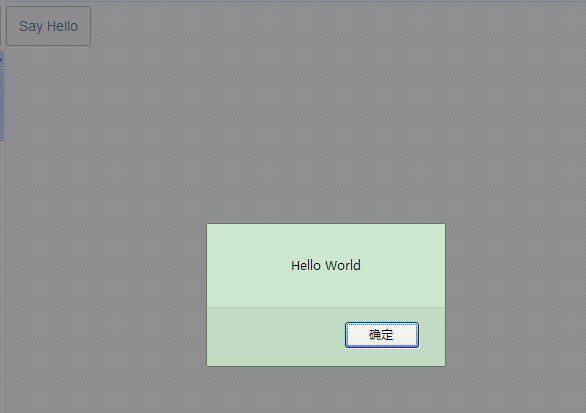
(2) 加入View

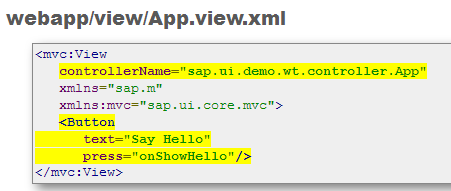




在UI5使用过程中，命名空间的概念很重要。UI5中采用命名空间的机制，对视图进行加载。我们首先创建一个命名空间sap.ui.demo.wt，并在index.html中加载。

(3) 加入Controller：





在头部通过controllerName定义了controller的命名空间。添加了Button标签，定义了一个按钮，并设定了其text属性和press属性。press定义了点击后执行的函数为onShowHello。

视图修改完毕后，我们对应的建立一个控制器文件，App.controller.js



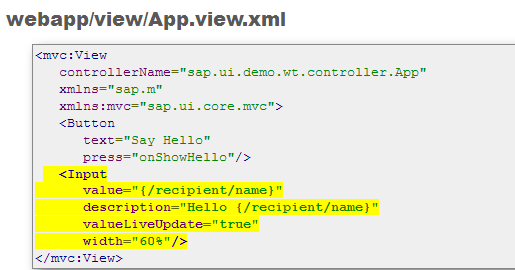
这个代码里，define方法定义一个控制器。其中sap/ui/core/mvc/Controller，大家可以在下载的sdk文件中的resources文件中按照目录结构找到，相当于一个控制器的基础文件。之后定义方法，这里我们之定义了一个onShowHello方法，内容很简单，弹出一个Hello World对话框。

(4)加入Model:





其中，onInit方法相当于构造函数，载入controller后自动调用。在这个function里我们实例化了一个Jason model.包含了一个属性recipient已经一个附加属性name，以上代码构造了一个Model，通过setModel方法在视图中使用setModel方法



这里定义一个input标签（sap.m.Input）,/recipient/name标识调用了的数据内容（和上面模型相对应）。

    经过如上的修改，我们就完成了数据模型的引用。以上也就是一个在SAPUI5中完整的MVC框架结构。

(5)加入Component:

SAPUI5中支持利用Component对**组件**进行封装。分析一下Component框架的代码含义，引用了core中的UIComponent基础空间，组件的编写在UIComponent.extend中进行，即进行扩展。

我们尝试将之前的应用封装成一个组件，新建Component.js文件，代码如下

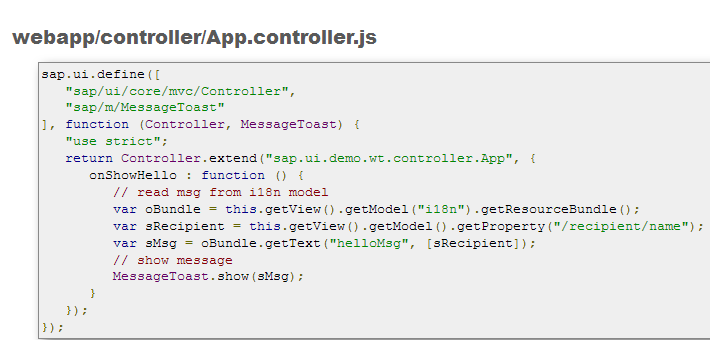


define后每引用sap的一个组件，后面的function就要传入一个对应的参数。

Component的构建流程如上，extendUIComponent这个框架，里面init为初始化函数，里面可以设定其他属性（包括配置模型等）

我们将原来Controller.js文件中的初始化函数、数据模型绑定配置等工作都放到了Component.js当中，相应的修改Controller.js文件

在Controller.js文件中，只保留本项目中需要使用的各个函数，这样使得项目中各个文件的逻辑更清晰了。



在index.html中，我们可以直接调用Component

