**目录：**

[一、Flex页面方法完整调用过程 2](#_Toc29243)

[二、添加二级窗口 3](#_Toc2289)

[三、给输入框添加验证 4](#_Toc2450)

## 一、Flex页面方法完整调用过程

1. xxxModule.mxml
2. xxxPM
3. xxxDelegate
4. xxxMsg
5. xxxModuleConfig

A)在mxml页面中配置config如下

<fx:Declarations>

<parsley:ContextBuilder>

<parsley:ViewSettings autoremoveViewRoots="false"/>

<parsley:Scope name="module"/>

<parsley:MessageSettings defaultReceiverScope="module"/>

<parsley:FlexConfig type="{StationInfoModuleConfig}"/>

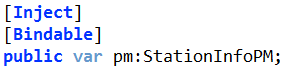
<parsley:FlexConfig type="{StationInfoFormConfig}"/>

</parsley:ContextBuilder>

<parsley:Configure/>

</fx:Declarations>

B)在mxml页面中注入pm如下



C)在mxml中调用方法如

private function loadPage():void

{

pm.loadPage();

}

D)在pm中创建消息传递对象，设置type,对应delegate中的[Command(selector="loadPage")]，发送消息

public function loadPage():void{

var msg:StationInfoMsg= new StationInfoMsg();

condi.language=langMgr.currentLanguage;

msg.type = "loadPage";

msg.args["page"] = page;

msg.args["parms"] = condi;

sendMessage(msg);

}

E)在delegate中，定义与delegate中相对应的”type”，传入对应的msg对象,然后调用call,其中call的第一个参数是service中的方法名

[Command(selector="loadPage")]

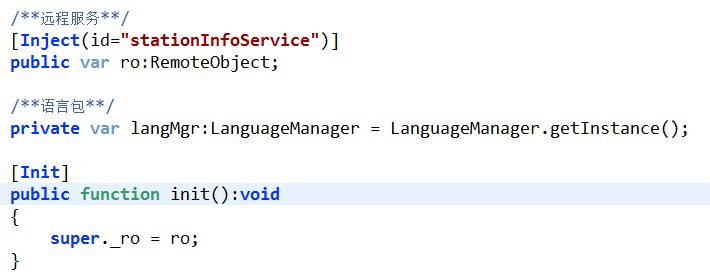
public function getTxnDayStatPageByOrg(p\_msg:StationInfoMsg):AsyncToken

{

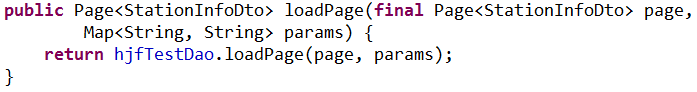
return call("loadPage", p\_msg.args["page"], p\_msg.args["parms"]);

}

在delegate中需要与后台交互则需要注入service，设置RemoteObject对象，如下



F)创建service对象并创建loadPage方法,service调用dao，dao返回给service



G)service返回给delegate，定义对应的[CommandResult(selector="loadPage")]接收service的结果，会话入ResultEvent对象和Msg对象，然后发送给pm（或mxml页面）,发送的时候定义对应的”type”,在pm或mxml中是通过type接收的（loadPageResult）

[CommandResult(selector="loadPage")]

public function getPageByOrgResultHandler(p\_result:ResultEvent, trigger:StationInfoMsg):void

{

var msg:StationInfoMsg = new StationInfoMsg();

msg.type = "loadPageResult";

msg.result = p\_result.result;

sendMessage(msg);

}

H)如下是在mxml页面中定义MessageHandler接收delegate中的返回结果

[MessageHandler(selector="loadPageResult")]

public function getPageHandler(p\_msg:StationInfoMsg):void

{

var list:Page =(p\_msg.result) as Page;

pm.page=list;

if(list !=null && list.result !=null){

var tem:ArrayCollection =list.result as ArrayCollection;

this.db=tem;

}

}

以上是从mxml-->pm-->delegate-->service-->dao-->service-->delegate>mxml的完整调用过程

## 二、添加二级窗口

1.定义一级窗口所需的组件xxxModule.mxml、xxxModulePM、xxxModuleDelegate、xxxModuleMsg、xxxModuleConfig、xxxModuleService、xxxModuleDao

2.定义二级窗口所需组件xxxForm.mxml、xxxFormPM、xxxFormDelegate、xxxFormMsg、xxxFormConfig、xxxFormService、xxxFormDao、xxxFormVo、xxxFormConstants

3.在一级mxml中配置两个config,在二级mxml中注入二级pm,其实注入同一级注入，参考（[Flex页面方法完整调用过程](#_Flex页面方法完整调用过程)）

注：在配置config的时候需要在两个页面中都添加<parsley:Configure/>（原因不明）

1. 在一级中调用openWin方法，内容如下，其中需要额外定义VO和Constants，PopUpManager为common中的对象，formType为Vo中额外的属性，其它像titleKey和\_context为父类中的属性

public function add():void

{

var vo:StationVo = new StationVo();

vo.formType=StationInfoConstants.ADD\_STATIONINFO\_FORM;

openWin(StationInfoConstants.ADD\_STATIONINFO\_FORM, vo);

}

public function openWin(p\_winName:String, p\_vo:StationVo):void

{

//dispatch event

var win:StationInfoForm = new StationInfoForm();

win.titleKey = p\_winName;

//加入到parsley上下文

\_context.viewManager.addViewRoot(win);

PopUpManager.showForm4Parsley(IFlexDisplayObject(win), this);

win.pm.vo = p\_vo;

win.pm.menuId = this.menu.id;

}

1. 注：其它方法同一级窗口调用过程，其中一级mxml继承的是TabModule，二级窗口继承的是TitleWindow，config配置只能配置在一级mxml中，需要用到PM的时候各自注入即可。

## 三、给输入框添加验证

* 1. 添加红色星号,在<grg:FormItem>中添加required=”true”>

<grg:FormItem id = "lineIdItem" locale = "lineId" required="true">

<grg:TextInput id = "lineId" name="lineId" width = "150" maxChars="32" vtype="maxchars" required="true"/>

</grg:FormItem>

b) 添加内容非空校验

1. 在<grg:TextInput>中添加required=”true”
2. 在相应的方法中添加如下内容

if (VControl.validateAll(form, validators).length > 0){

return;

}

c)<grg:TextInput>的其它属性

width：指定组件在父坐标中的宽度的数字,默认值为0，可以使用百分比，如果不能则使用percentWidth

locale：国际化

maxChars：用户可以输入的字符数，如果为0则可以无限输入，超过长度则无法输入

maxnum：用户可以输入的字符数，超过长度可以输入，但无法通过校验

required：是否为必需，如果为true，那么输入框为空是不能通过校验

restrict：匹配正则

text：输入框默认内容，例：text="@{pm.termOperatorVo.contactMobilePrefix}"

enabled：是否可以让光标定位到该输入框，如为false则不能输入，不能选中

visible：是否可见

includeInLayout：指定此组件是否包含在父容器的布局中，false则有受父容器布局约束

vtype：输入框校验规则，如ip,maxchars,email等

minnum：未知

name：未知