eFID系統關於導航欄的管理和控制(包含權限管控)

類AbstractDomainObject是dom4j來對xml進行讀寫的。系統的所有的菜單都是以XML的形式保存在menu的文件夾下，他們的格式大多如下：

<?xml version="1.0" encoding="UTF-8"?>

<section sid="section.am">

<role>fid.director,fid.am.manager,fid.pm.manager</role>

<labelKey>module.am</labelKey>

<action actionId="action.am">

<mode>RELOAD</mode>

<href>/C110.do?action=getList&amp;showAll=false</href>

<dwrScript/>

<jsTemplate/>

</action>

<section-list selected="section.am.user">

<section sid="section.am.user">

<role>fid.director,fid.am.manager,fid.pm.manager</role>

<labelKey>module.am.user.management</labelKey>

<action actionId="action.am.listUser">

<mode>RELOAD</mode>

<href>/C110.do?action=getList&amp;showAll=false</href>

<dwrScript/>

<jsTemplate/>

</action>

<section-list selected=""/>

</section>

<section sid="section.am.guest">

<role>fid.director,fid.am.manager,fid.pm.manager</role>

<labelKey>module.am.guest.management</labelKey>

<action actionId="action.am.listGuest">

<mode>RELOAD</mode>

<href>/C110.do?action=getList&amp;showAll=false&amp;type=guest</href>

<dwrScript/>

<jsTemplate/>

</action>

<section-list selected=""/>

</section>

<section sid="section.am.role">

<role>fid.director,fid.am.manager,fid.pm.manager</role>

<labelKey>module.am.role.management</labelKey>

<action actionId="action.am.listRole">

<mode>RELOAD</mode>

<href>/C120.do?action=getList</href>

<dwrScript/>

<jsTemplate/>

</action>

<section-list selected=""/>

</section>

<section sid="section.am.rate">

<role>fid.director,fid.am.manager,fid.pm.manager</role>

<labelKey>module.am.exchange.management</labelKey>

<action actionId="action.am.listRate">

<mode>RELOAD</mode>

<href>/C220.do?action=getExchangeRateList</href>

<dwrScript/>

<jsTemplate/>

</action>

<section-list selected=""/>

</section>

<section sid="section.am.logger">

<role>fid.director,fid.am.manager,fid.pm.manager</role>

<labelKey>module.am.log</labelKey>

<action actionId="action.am.logger">

<mode>RELOAD</mode>

<href>/C310.do?action=getLogger</href>

<dwrScript/>

<jsTemplate/>

</action>

<section-list selected=""/>

</section>

<section sid="section.am.kpi">

<role>fid.director,fid.am.manager,fid.pm.manager</role>

<labelKey>module.am.kpi</labelKey>

<action actionId="action.am.kpi">

<mode>RELOAD</mode>

<href>/C410.do?action=query</href>

<dwrScript/>

<jsTemplate/>

</action>

<section-list selected=""/>

</section>

<section sid="section.am.web">

<role>fid.director,fid.am.manager,fid.pm.manager</role>

<labelKey>module.am.web</labelKey>

<action actionId="action.am.web">

<mode>RELOAD</mode>

<href>/C410.do?action=queryWebSite</href>

<dwrScript/>

<jsTemplate/>

</action>

<section-list selected=""/>

</section>

</section-list>

</section>

首先，每一個菜單節點都是一個section，其中sid是這個節點的ID，role標誌了可以訪問該菜單的角色，labelKey指定了該菜單的標籤名稱，actionID標誌了要執行的對做的ID，mode指定來菜單加載的模式，href標誌來點擊菜單的時候，指向的鏈接，dwrScript應該是指定了點擊菜單要執行的dwrScript代碼，jsTemplate制定來點擊菜單後要套用的JSP模板。 section-list指定了該菜單下的子菜單，elected="section.am.user"標識當點到父菜單的時候，默認的要選中的子菜單。

菜單類的定義

當程序運行的時候，spring工廠首先常見配置在xml文件中的bean。

在導航欄這方面，首先，創建spring.common.menu.xml文件中定義的bean，

commonMenuManager是iPLM.bean.menu.manager.MenuManager的實例，它的作用是用來導入定義菜單的xml文件(首先，resource指明了要導入的菜單所在 的xml文件，menus要管理的菜單視圖，hasSectionRef指明了是否有section的ref，hasActionRef指明了是否有action的ref，MENU\_ROOT\_SELECTED指明了被選中的菜單根目錄，MENU\_ACTIONS指明了menu的連接，SECTION\_LIST指明了子菜單的清單)。其次創建了commonMenuService對象，他是iPLM.bean.menu.manager.MenuDWRService的實例，(首先，該類定義了MENUS\_SELECTED，用來指明給選中的菜單，customMenuService是一個MAP，用來指定每一個模塊所對應的取得菜單的bean)，通過以上倆個對象，生成menuDelegate對象，該對象是iPLM.bean.menu.web.MenuController的實類。當對象初始化完成之後，首先執行index方法（？？sorry，目前為止，我還不太清楚，初始化頁面的時候，是哪裏指引程序去執行index方法），該方法會返回一個菜單視圖和一個頁面，

logo.jsp頁面，在logo頁面就會根據菜單視圖生成對應的菜單。

現在來看一下，菜單視圖裏面到底存放來哪些內容：

1.開始執行index方法的時候，通過request請求來加載菜單視圖,而菜單視圖是通過loadMenus函數根據menuManager屬性來加載的，menuManager私有屬性是MenuManager的實例，在程序開始的時候，通過spring工廠已經對其進行了創建