JIRA系统的插件体系（1）：总体一瞥

http://simg.sinajs.cn/blog7style/images/common/sg_trans.gif(2010-07-10 00:13:14)

[[http://simg.sinajs.cn/blog7style/images/common/sg_trans.gif](javascript:;)转载▼](javascript:;)

|  |  |
| --- | --- |
| 标签：  [jira](http://search.sina.com.cn/?c=blog&q=jira&by=tag)  [杂谈](http://search.sina.com.cn/?c=blog&q=%D4%D3%CC%B8&by=tag) | 分类： [快乐的工作](http://blog.sina.com.cn/s/articlelist_1047762895_1_1.html) |

JIRA系统是一个非常优秀的项目管理系统，在JIRA系统上可以进行流程定义、表单定义、丰富的权限管理等等。但其最有特色的，我认为还是其高可配置、灵活的插件体系。在JIRA所提供的插件体系下，我们可以进行二次开发，通过功能扩展以满足差异化的需求。如：自定义菜单、扩展自定义属性、扩展流程的验证规则、扩展流程的流转条件、扩展流程的Post Function等等。

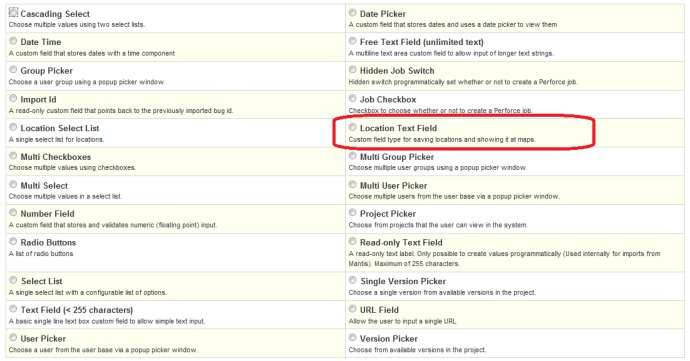
JIRA的插件体系是基于OSGI可插拔式的，每一个插件实际上就是一个jar包。这个插件jar包放到 %JIRA\_HOME%\atlassian-jira\WEB-INF\lib 目录下，重启JIRA系统，即可实现插件的动态部署。

那么JIRA是怎么能识别出插件的呢？  所有的JIRA插件jar包中，都必须要包含一个以 atlassian-plugin.xml 命名的配置文件。下面是一个 atlassian-plugin.xml 样例。（以jira-suite-utilities插件的配置文件为例子，红色部分是注释说明）

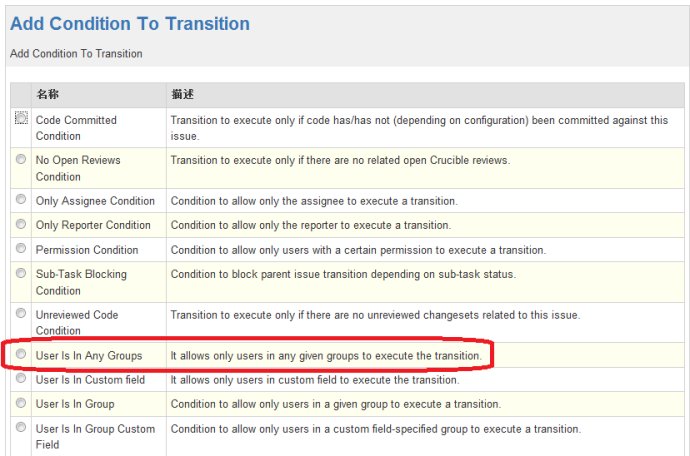
|  |
| --- |
| <atlassian-plugin key="com.googlecode.jira-suite-utilities"                    name="JIRA Suite Utilities"                    plugin-version="2">        <plugin-info>          <description>Many objects to extend JIRA</description>          <version>0.7.7</version>          <application-version min="3.5" max="4.1"/>          <vendor name="Quadratica SRL" url="http://www.quadratica.com.ar"/>      </plugin-info>    <!—在JIRA中增加自定义属性，如这里定义了一个 “Location Text Field” 类型的自定义字段 -->      <customfield-type key="locationtextfield"                        name="Location Text Field"                        class="com.atlassian.jira.issue.customfields.impl.TextCFType"                        i18n-name-key="custom.field.type.text\_field.name">  <resource type="i18n" name="i18n" location="com.googlecode.jsu.maps.resources" /> <!—国际化资源文件 -->            <description key="custom.field.type.text\_field.description"> Saving locations and showing it at maps.</description>          <!—自定义字段的查看、编辑的显示模板 -->          <resource type="velocity" name="view" location="templates/jira/fields/view/view-location.vm"/>          <resource type="velocity" name="edit" location="templates/plugins/fields/edit/edit-basictext.vm"/>          <resource type="velocity" name="xml" location="templates/plugins/fields/xml/xml-basictext.vm"/>      </customfield-type>    <!— 流程流转的条件规则定义，如这里定义了一个“用户是否属于某个群组”的判断条件定义 -->  <workflow-condition key="userIsInAnyGroups-condition" name="User Is In Any Groups"          class="com.googlecode.jsu.workflow.WorkflowUserIsInAnyGroupsConditionPluginFactory">          <description>It allows only users in any given groups to execute the transition.</description>          <!—条件定义的具体实现类 -->  <condition-class>com.googlecode.jsu.workflow.condition.UserIsInAnyGroupsCondition  </condition-class>            <resource type="velocity" name="view" location="templates/jira/workflow/userIsInAnyGroups-condition-view.vm"/>          <resource type="velocity" name="input-parameters" location="templates/jira/workflow/userIsInAnyGroups-condition-edit.vm"/>          <resource type="velocity" name="edit-parameters" location="templates/jira/workflow/userIsInAnyGroups-condition-edit.vm"/>  </workflow-condition>    <!— 流程处理方式在提交时的校验定义，下面的例子是定义了一个时间比较的验证类 -->  <workflow-validator key="dateCompare-validator" name="Date Compare"  class="com.googlecode.jsu.workflow.WorkflowDateCompareValidatorPluginFactory">  <description>Compare two date fields during a workflow transition.</description>    <validator-class>  com.googlecode.jsu.workflow.validator.DateCompareValidator  </validator-class>    <resource type="velocity" name="view" location="templates/jira/workflow/validator/datecompare-validator-view.vm"/>  <resource type="velocity" name="input-parameters" location="templates/jira/workflow/validator/datecompare-validator-input.vm"/>  <resource type="velocity" name="edit-parameters" location="templates/jira/workflow/validator/datecompare-validator-edit.vm"/>  </workflow-validator>    <!— 流程在提交后的处理函数定义，下面的例子是一个从其他字段中拷贝值的方法定义 -->  <workflow-function key="copyValueFromOtherField-function" name="Copy Value From Other Field"  class="com.googlecode.jsu.workflow.WorkflowCopyValueFromOtherFieldPostFunctionPluginFactory">          <description>It copies the value of one field to another.</description>            <function-class>                  com.googlecode.jsu.workflow.function.CopyValueFromOtherFieldPostFunction          </function-class>            <orderable>true</orderable>          <unique>false</unique>          <deletable>true</deletable>          <default>false</default>    <resource type="velocity" name="view" location="templates/jira/workflow/copyvaluefromfield-function-view.vm"/>  <resource type="velocity" name="input-parameters" location="templates/jira/function/copyvaluefromfield-function-input.vm"/>  <resource type="velocity" name="edit-parameters" location="templates/jira/workflow/copyvaluefromfield-function-edit.vm"/>      </workflow-function>      <!— 下面的XML是定义了Dashboard上可显示的Portlet -->      <issue-tabpanel key="transitions-summary-tabpanel"                                      name="Transitions Summary Tab Panel"                                      class="com.googlecode.jsu.transitionssummary.issuetabpanel.TransitionsSummaryTabPanel">    <description key="transition.summary.description" />  <label key="transition.summary.transitions" />                <resource type="i18n"                                name="i18n"                                location="com.googlecode.jsu.transitionssummary.transitionssummary" />          <resource type="velocity"                            name="view"                            location="templates/jira/issuetabpanel/transitionssummary/transitions-summary-view.vm" />      </issue-tabpanel>  </atlassian-plugin> |

当然，在这个JIRA插件配置文件中，定义的自定义属性、流程验证器、流程流转条件以及流程流转后处理方法等，也是需要按照JIRA的规范实现相关的接口的。这里，暂时先不展开描述。在后续的连载中，尽量详细介绍（因为我很懒，不知道什么时候会更新Blog）。

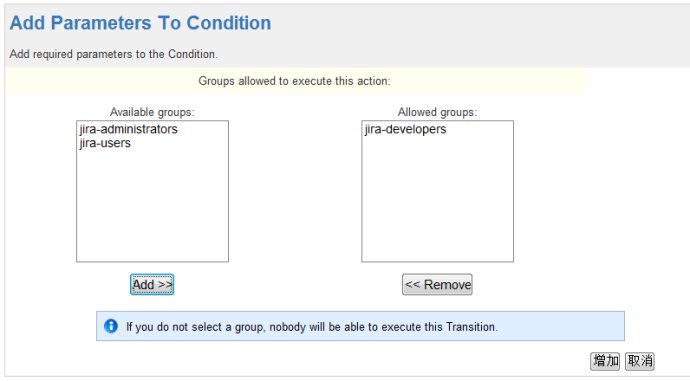
还是以刚刚的jira-suite-utilities插件为例。将jira-suite-utilities-0.7.7.jar拷贝到  %JIRA\_HOME%\atlassian-jira\WEB-INF\lib 目录下，重启JIRA系统。 登陆JIRA系统，我们在自定义属性中，就可以看见在原有的自定义属性类型中新增加了一种“Location Text Field”类型的字段。如下所示：

[](http://photo.blog.sina.com.cn/showpic.html#blogid=3e7397cf0100k395&url=http://s4.sinaimg.cn/orignal/3e7397cfn8af6b8af3e33)

同样，我们在流程的验证器中，也能发现系统中多了一些原来所没有的验证器，如“User Is In Any Groups”验证器。

[](http://photo.blog.sina.com.cn/showpic.html#blogid=3e7397cf0100k395&url=http://s2.sinaimg.cn/orignal/3e7397cfn8af6b727bd71)

选中该验证器后点击下一步：

[](http://photo.blog.sina.com.cn/showpic.html#blogid=3e7397cf0100k395&url=http://s7.sinaimg.cn/orignal/3e7397cfn8af6bb127636)

以上的这些功能，都是通过插件方式提供的，而没有对JIRA系统做任何破坏性、侵入式的修改！