**简介**

设计工具(Activiti Designer)、**流程引擎**(Process Engine)、监控平台(Activiti Explorer)

Activiti流程引擎实际是一个状态机(state machine)，实现BPMN2.0规范。

ProcessEngineConfiguration

IdentityService(设置流程参与者)；TaskService(流程任务)

部署流程，生成ProcessDefinition

创建ProcessInstance

RuntimeService

RepositoryService

ProcessEngine

流程部署到数据库中。

Task：ScriptTask(支持多种脚本，默认Groovy)，UserTask(人工任务)

**BPMN(Business Process Model & Notion)**

Pool：一个pool对应一个流程，pool的名称和流程名称保持一致。

Lane：将一个role或者organization的task放在一个区域。

Task：可以是自动或人工的activity，分别对应Service Task和User Task。

Parallel Gateway：多分支同时运行。

Exclusive Gateway：只有一条分支可运行。

Message Flow：在不同Pool之间发送信号或消息。

Sequence Flow：同一个Pool内，连接activity、gateway和event。

End Error Event：特殊的结束event，可以抛出异常。

Start Error Event：特殊的开始event，可以捕获End Error Event。

流程包是一个BAR文件。

**Activiti Engine**

组成：engine本身，包含一个流程虚拟机抽象，将BPMN2.0流程逻辑转换为状态机模型，所在的jar包是activiti-engine-version.jar；activiti-spring-version.jar是与spring集成的jar。

七大interface

FormService：生成用户任务表单。

HistoryService：查询已完成任务。

IdentityService：提供认证服务，主要用于指定流程参与者。

ManagementService：查询Activiti的表和执行job。

RepositoryService：部署、查询、删除、取回流程定义。

RuntimeService：启动、查询流程实例，设置和获取流程变量，也可将流程设为等待状态。

TaskService：提供和用户task相关的服务。

完成一个User Task，需要先claim它再complete它。

Java Service Task(自动任务)

当是同步时，和流程在同一个线程和事务中；添加async属性后为异步，不同事务，不同线程。async属性几乎支持所有类型的task。

1、实现JavaDelegate接口中的execute方法。

所有的Service Task都是单例的，因此，它里面的类属性要保证线程安全性。Activiti Engine提供了一个Expression类来获取流程变量。

2、JavaBean方式

需要和流程同名，同时实现Serializable接口。

使用Spring

配置：参见《Activiti in Action》第92页。

问题：如果为processEngineConfiguration配置了deploymentResources，每重启一次都会重新部署吗？

Service Task：不用实现JavaDelegate接口，就是一个简单的Spring Bean，要调用的方法的参数是DelegateExecution即可。

**模块、元素**

Script Task：支持Groovy、Clojure、Jruby、Jython、Javascript等。execution是内置元素，可以直接用来设置流程变量。

表单：可以用在开始图元和User Task图元上，有两种方式生成表单。

1、将表单元素列出来，通过Activiti Explorer自动生成。

2、自定义，需要使用activiti:formKey指定表单。

3、表单中可以使用JUEL表达式，一种类似于EL的表达式。

子流程：分为嵌入式子流程和独立子流程。独立子流程有自己的生命周期，主流程通过call的方式调用它。

page 131，activiti in action