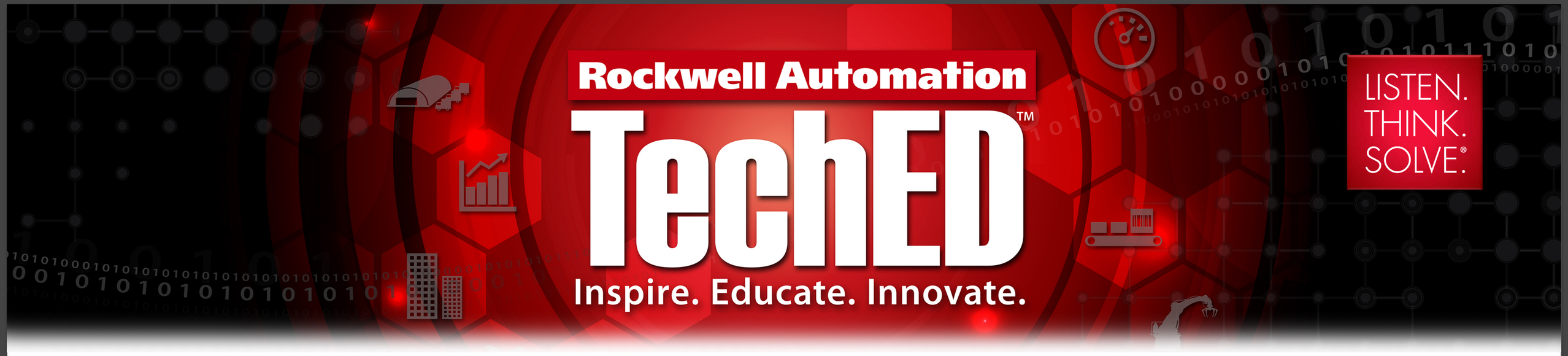


FactoryTalk ProductionCentre培训

应用开发 – Activity

ISPB (信息软件部门)
2016年9月18日

PUBLIC INFORMATION

Allen-Bradley • Rockwell Software

Rockwell Automation

课程说明

- **课程名字：**

  FTPC培训 – Activity & Activity Set

- **课程描述：**

  本课程介绍FactoryTalk ProductionCentre的Activity和Activity Set开发方法。

- **目标受众：**

  开发人员

- **培训目标：**完成该课程之后，学员将能够
  - 创建Activity类
  - 创建Activity Set对象

课程说明

- **课堂纪律：**
  - 关闭手机铃声，设置为静音模式
  - 举手提问、发言

内容

Rockwell Automation

- Activity开发
- Activity Set开发
- QA

Activity开发

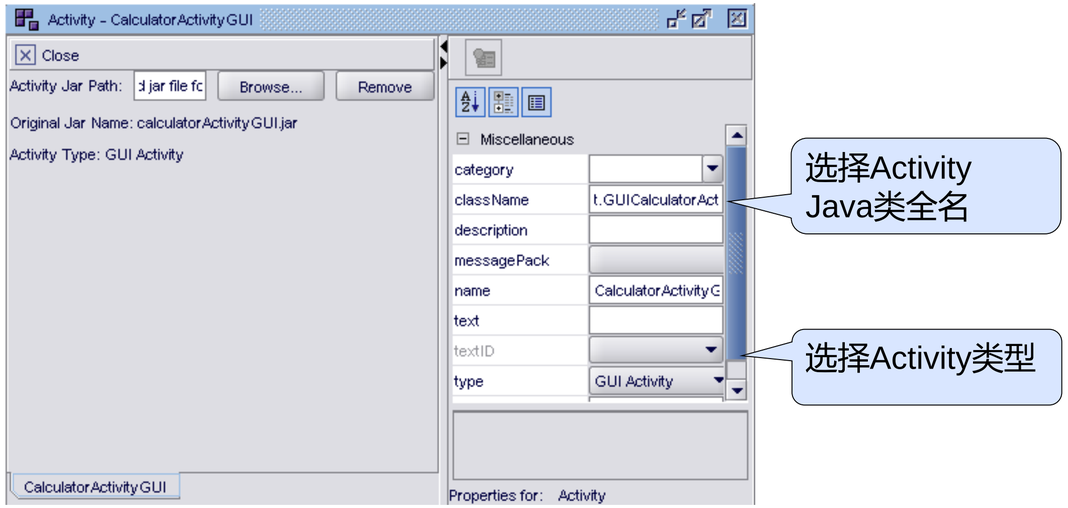
- Activity**开发环境**
  - 搭建Eclipse环境
  - Activity的Java类在Eclipse工厂中开发
  - Java包导入到FTPC中以Library对象存在
  - Activity对象在FTPC中创建，引用Java类

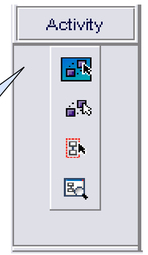
Activity开发

- Activity**介绍**

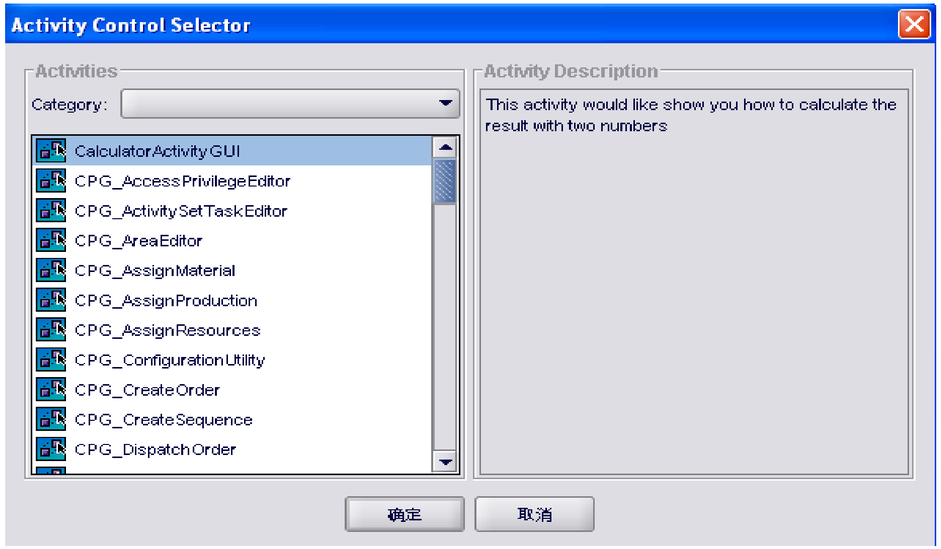
Activity是FTPC中可重用的组件，可以用来创建基于Java用户界面或无界面的后台处理应用。

- 2种Activity
  - GUI Activity：用在Form或Activity Set中构建用户界面
  - 非GUI Activity：其他场景





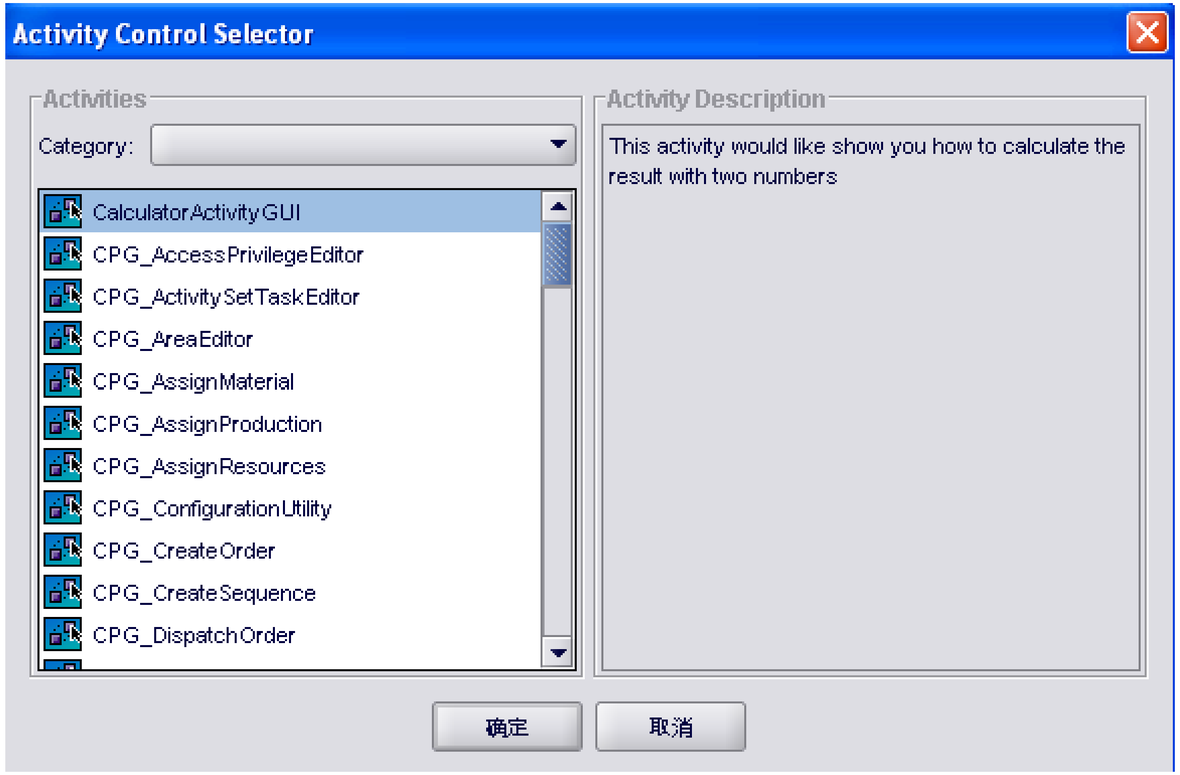
选择Activity，添加到Form或Event Sheet中使用



Activity开发

- Activity**开发介绍**

Activity是FTPC中可重用的组件，可以用来创建基于Java用户界面或无界面的后台处理应用。



- 2种Activity
  - GUI Activity：UI控件，只能使用在Form中
  - 非GUI Activity：非UI控件，可以用在Form或其他环境中

- GUI Activity

  继承自 com.datasweep.compatibility.ui.ActivityControl

- Non-GUI Activity

  继承自 com.rockwell.activity.Activity

Activity开发

- Activity**的同步、异步方法**

- activityExecute(): 该方法为同步调用方法。
- activityStart(): 该方法会启动一个新线程（异步调用），之后再调用activityExecute()方法。
- complete(): 如果该Activity在Activity Set里，这个Activity线程将会解除阻塞获得执行。如果该Activity是在一个Form中，则该方法会调用activityExecute()方法。

业务逻辑应该总是放在activityExecute方法中被执行

如何选择使用同步方法还是异步方法

- 该Activity是否执行时间很长
- 用户是否需要在执行Activity时处理其他任务

如果以上答案都是真，建议采用异步执行方式*

Activity开发

- Activity**的同步方法**

- 如果Activity是在同步线程中执行，该activity的执行是在当前程序的同一主线程中执行。
- 当前执行只有在Activity返回response时结束。
- 在同步执行时，Activity的状态模式是不适用的，并且activity的生命周期事件也不会触发。

Activity开发

- Activity**的异步方法**

- 如果Activity是在异步线程中执行，当前程序的线程继续执行，Activity在它自己的新的线程中执行。
- 在异步执行时，Activity的状态模式是适用的，并且activity的生命周期划分为几个状态。
- 但Activity进入一个新状态，相关事件会被触发。
- 程序开发人员可以编写事件处理脚本，亦可以编写客制化事件处理脚本。
- 程序开发人员可以控制Activity通过调用activityStart（ ）或activityHold（ ）使Activity进入特定状态。

Activity开发

- Activity**本地化支持**
  - 不硬编码消息文本，使用FTPC的message对象
  - 避免硬编码GUI控件文字，使用FTPC的message对象
  - 使用满足用户区域和语言的习惯的日期、货币格式，使用FTPC的本地化数字和日期函数
  - 本地化Response对象的错误消息文本，使用FTPC的message对象

Activity开发

- Activity**类方法**

方法	要求
activityExecute()	业务逻辑主要在该方法中实现。最好在该方法中设置activity的合适状态。
configurationDescription()	如有需要可实现该方法提供配置项描述
configurationItemSet()	如有需要可实现该方法来获得配置项
configurationLoaded()	Implement it to get configuration if needed
getActivityDescription()	Implement it to provide detail description to the activity
getActivityEvents()	Implement it to get custom events if needed
getBaseName()	Implement it to give an unique default name to the activity
inputDescriptors()	Implement it to specify input parameter type if needed
inputItemSet()	Implement to get input items if needed
outputDescriptions()	Implement it to specify out descriptions if needed
Shutdown()	Implement it if there is resource needed to be release
startup()	Implement if if anything is required to be instantiated after the form or event sheet opens

Activity开发

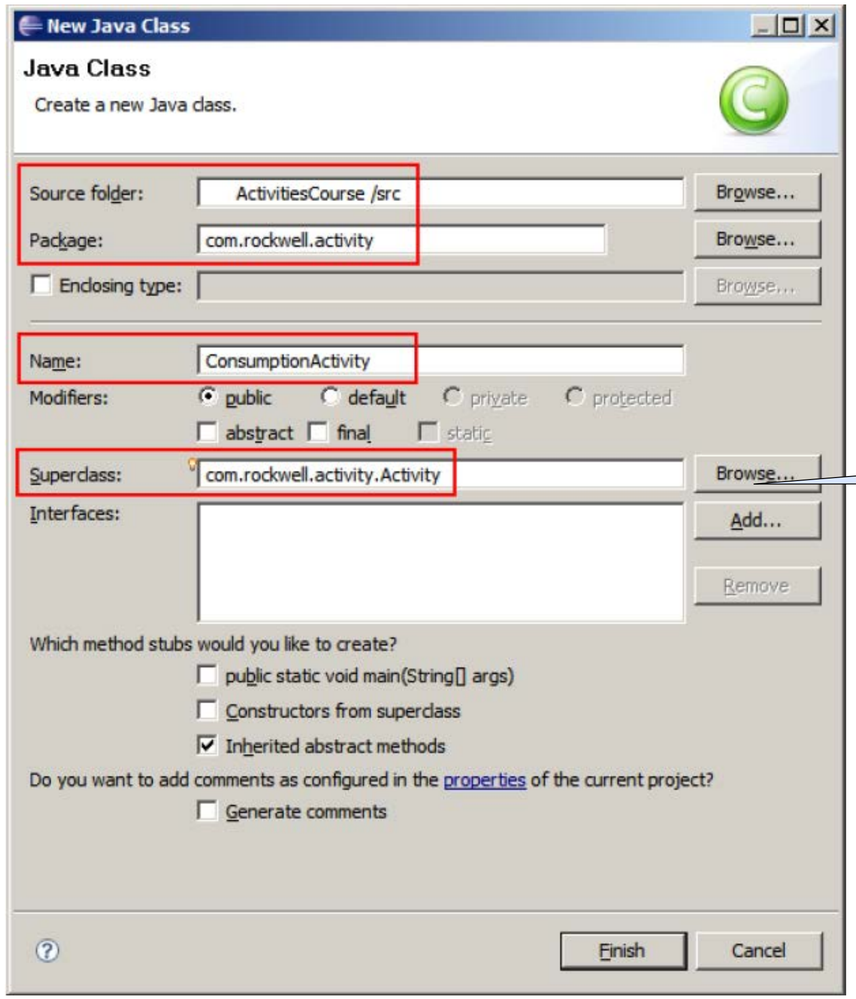
Rockwell Automation

- **如何创建**Activity

	创建步骤	备注
1	创建一个新工程，导入ProductionCentreClient.jar文件，设置使用正确的JRE	ProductionCentreClient.jar版本与FTPC版本一致
2	创建Activity包和类	类必须继承自ActivityControl (GUI) 或Activity (NON GUI)
3	编写Activity描述	实现getActivityDescription()方法
4	定义配置参数	实现configurationDescriptors()方法
5	定义输入参数	实现inputDescriptors()方法
6	定义输出参数	实现outputDescriptors()方法
7	创建客制化事件	实现getActivityEvents()方法
8	添加成员变量	
9	如果是GUI Activity, 定义layout，添加控件，并编写UI控件事件代码	
10	实现业务逻辑	实现activityExecute()方法或activityStart()方法

Activity开发

- **如何创建**Activity



指定Activity基类

- 如果是Non-GUI Activity，继承 com.rockwell.activity.Activity 类
- 如果是GUI Activity，继承com.datasweep.compatibility.ui.ActivityControl 类

Activity开发

- **如何测试、调试**Activity

	调试、测试步骤	备注
1	编译Activity类	
2	将所有类打包到jar文件	如果是调试，忽略该步骤
3	从IDE中启动Process Designer运行该Activity	Main class: com.datasweep.compatibility.buildtime.Buildtime Program arguments: jnp://localhost:1099 http://localhost:8080 DEBUG DEBUGACTIVITY VM arguments-Dcom.datasweep.plantops.j2eevendor=JBoss
4	Create an activity object in PD and define the properties in the properties frame.	Jar file path:If debug, ignore; className:activityPackage+activityClassName; Type:GUI or NON GUI.
5	Add the activity to a form or eventSheet.	
6	Define the configuration items in customizer dialog.	If not provided by activity developer, ignore this step.
7	Set inputItems and call the activityExecute() with pnuts.	If not required by activity developer, ignore this step.
8	Test or debug the activity.	

Activity开发

- Activity **样例代码**(1)
  - Input Descriptor
  - Output Discriptor

```
public ItemDescriptor[] inputDescriptors() {
        return (new ItemDescriptor[] {
                ItemDescriptor.createItemDescriptor(
                GUICalculatorActivity.class, "Number1",
                String.class, new Object[] {
                ItemDescriptor.SHORTDESCRIPTION,
                "Please input number1" }),

                ItemDescriptor.createItemDescriptor(
                GUICalculatorActivity.class, "Number2",
                String.class, new Object[] {
                ItemDescriptor.SHORTDESCRIPTION,
                "Please input number2" }) });


public ItemDescriptor[] outputDescriptors() {
        return (new ItemDescriptor[] { ItemDescriptor.createItemDescriptor(
                GUICalculatorActivity.class, "Result", String.class,
                new Object[] { ItemDescriptor.SHORTDESCRIPTION,
                "This is the result" }) });
}
```

Activity开发

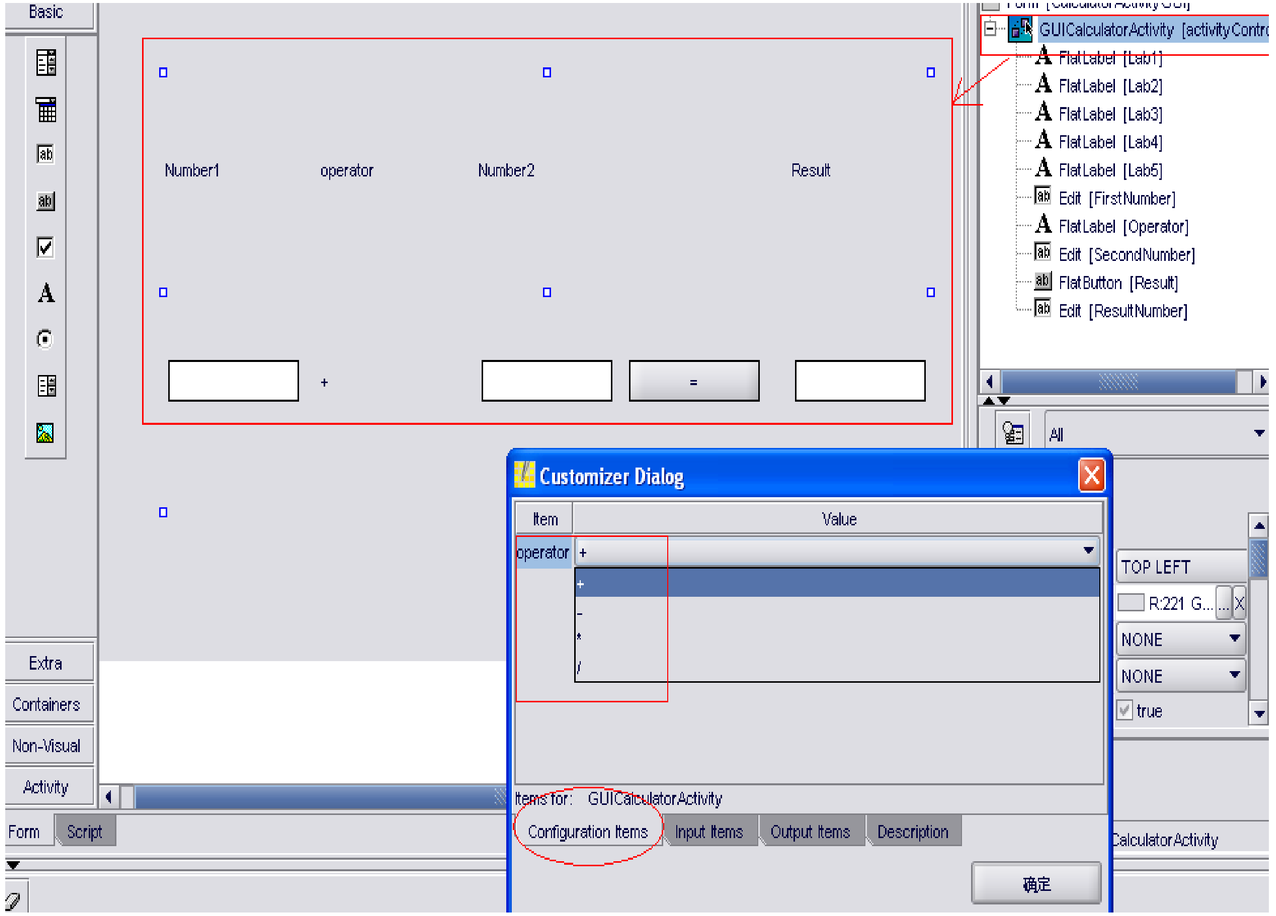
- Activity **样例代码**(2)
  - Configuration Descriptor
  - activityExecute

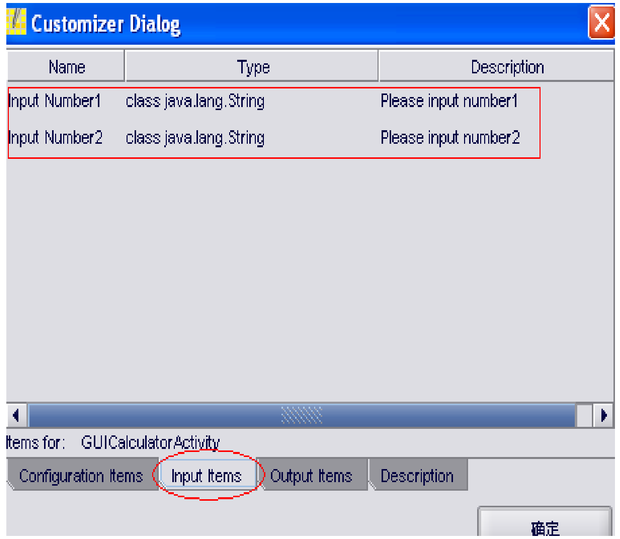
```
public ItemDescriptor[] configurationDescriptors() {
        Object enumOper[] = { "+", add, "GUICalculatorActivity.add", "-",
                        minus, "GUICalculatorActivity.minus", "*", multiple,
                        "GUICalculatorActivity.multiple", "/", divide,
                        "GUICalculatorActivity.divide", };
        return (new ItemDescriptor[] { ItemDescriptor.createItemDescriptor(
                GUICalculatorActivity.class, "operator", String.class,
                new Object[] { ItemDescriptor.ENUMERATIONVALUES, enumOper,
                ItemDescriptor.SHORTDESCRIPTION,
                "Please choose an operator." }), });
}

public Response activityExecute() {
        try {
                String number1 = getInputItem(“Number1").toString().trim();
                String number2 = getInputItem("Number2").toString().trim();
                String result = showNumber(number1, number2);
                setOutputItem("Result", result);
                Response r = new Response();
                return r;
        } catch (Exception e) {
                return createErrorResponse(e);
        }
```

Activity开发

- Activity **样例**





内容

Activity开发

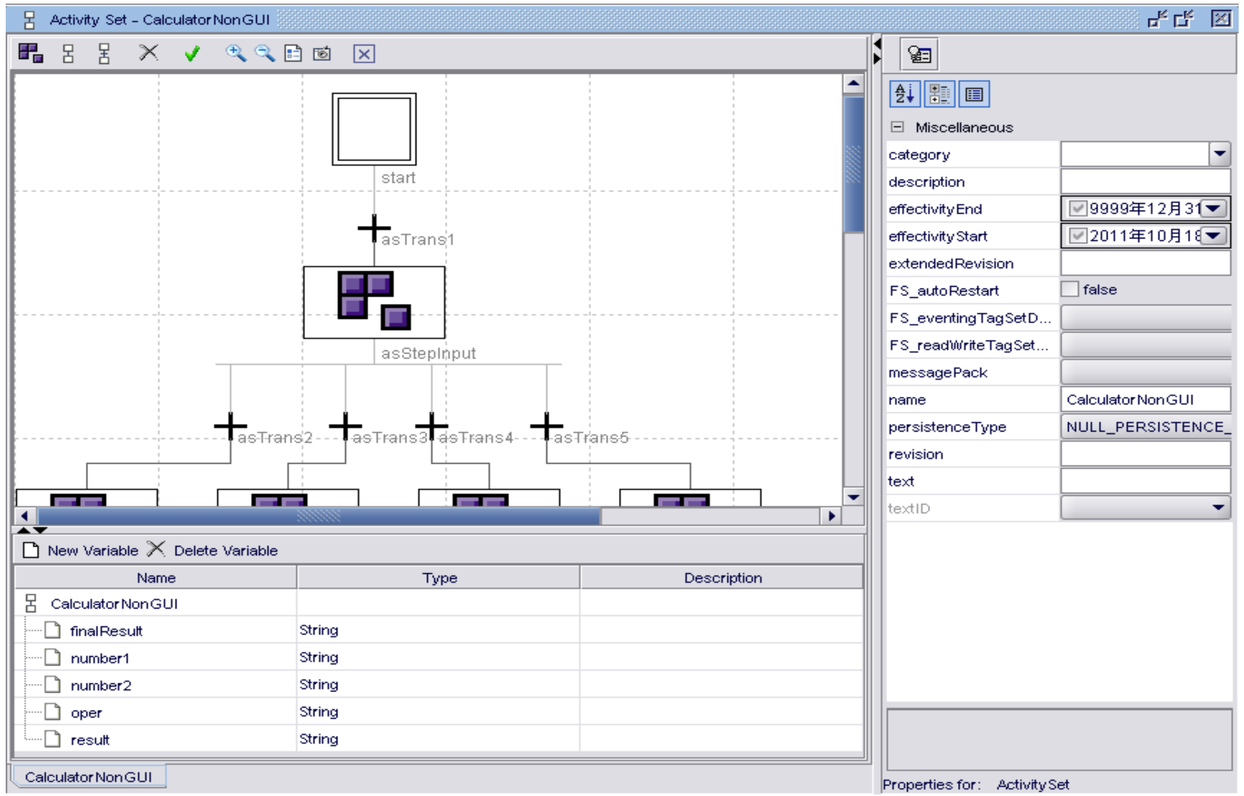
Activity Set开发

QA

Activity Set开发

- Activity Set**介绍**

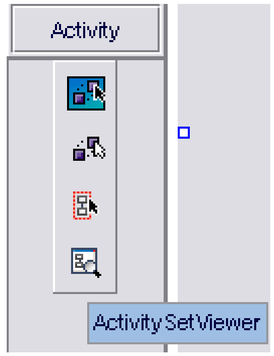
- Activity Set是采用顺序功能图（SFC)来表达和管理的工作流
- 使用顺序化Activity
- 可以用来管理操作员任务
- 或管帮助管理或记录自动化系统的行为



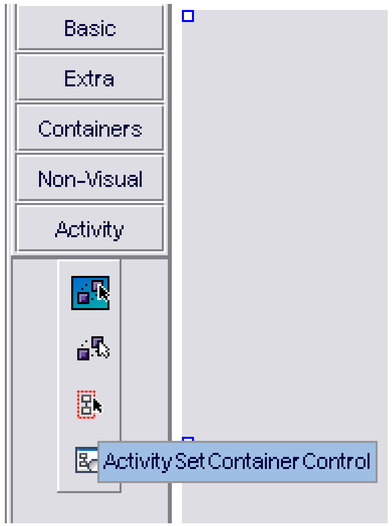
Activity Set开发

- Activity Set**的使用场景**

❑ Form



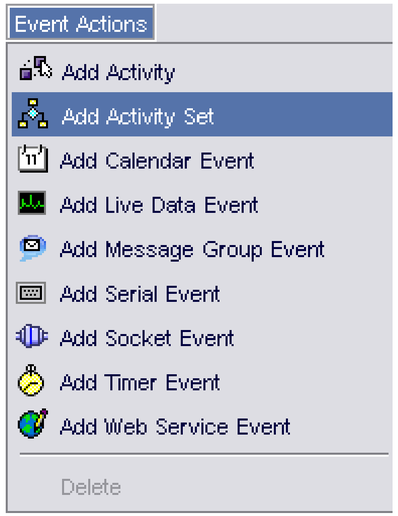
- The ActivitySet Viewer Control will provide a control that displays the SFC of the ActivitySet as a form control. The Execution control will provide methods that allow the user to highlight ActivitySet steps and the provide some text associated with each step. This control will allow application developers to provide a facility to view the execution of the ActivitySet. As noted earlier the ActivitySet Step may have an optional DsImage that is associated with each Step, when an image is assigned it is by default displayed in the interior of the Step, additionally the execution control will provide an API method to assign an image at runtime (the image may change depending of the state of the step).

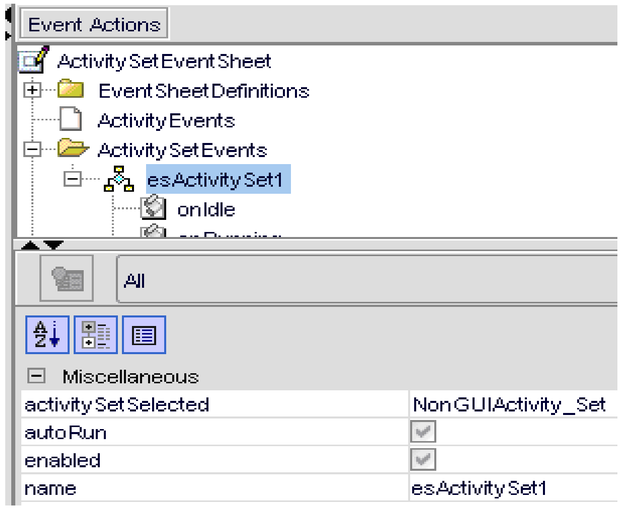


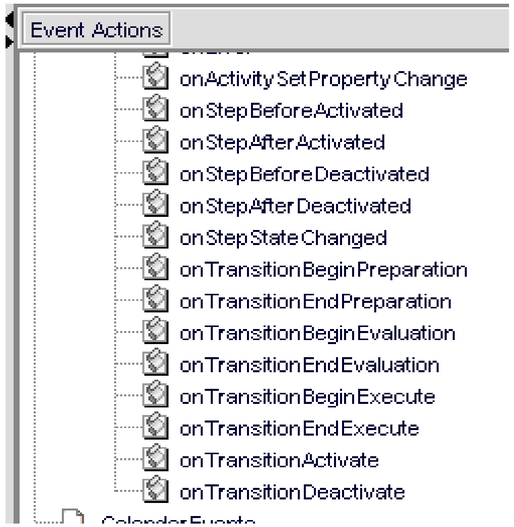
- A new non-visual control will be provide for Forms. This control will manage the execution of an ActivitySet

Activity Set开发

- Activity Set**的使用场景**
  - Event Sheet







To facilitate running an ActivitySet on an event sheet a new event container will be provided. The event container will provide the same set of events and behavior as the new non-visual form control. Significantly, an ActivitySet containing any GUI-Activities cannot be executed in an EventSheet using this control. If this control is presented an ActivitySet that contains a GUI-Activity it will refuse to execute.

Activity Set开发

- Activity Set**的使用场景**

- Event

- The events include the same lifecycle events available for Activities:

  onIdle
  onRunning
  onStopping
  onStopped
  onAborting
  onAborted
  onComplete
  onError
  onHolding
  onHeld

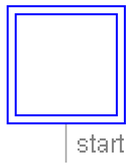
Additionally a set of events will be published that allow monitoring the execution of the SFC.

onActivitySetPropertyChange
onStepBeforeActivated
onStepAfterActivated
onStepBeforeDeactivated
onStepAfterDeactivated
onStepStateChanged
onTransitionBeginPreparation
onTransitionEndPreparation
onTransitionBeginEvaluation
onTransitionEndEvaluation
onTransitionBeginExecute
onTransitionEndExecute
onTransitionActivate
onTransitionDeactivate
onActivitySetPropertyChanged

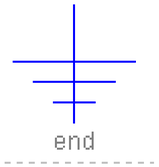
Activity Set开发

- Activity Set**介绍**

- 开始步骤

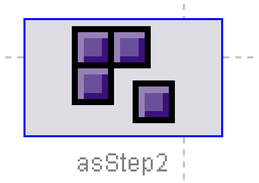


- 结束步骤

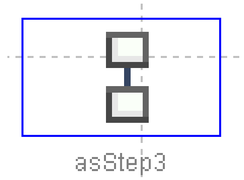


- 转换

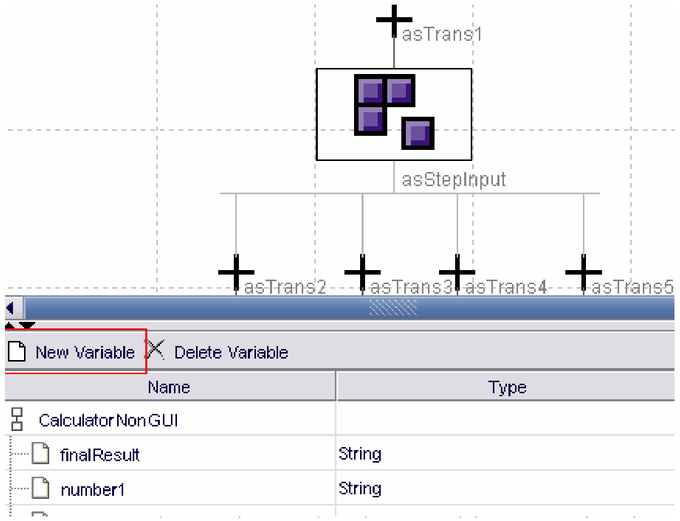
- 常规步骤



- 宏步骤



- 变量



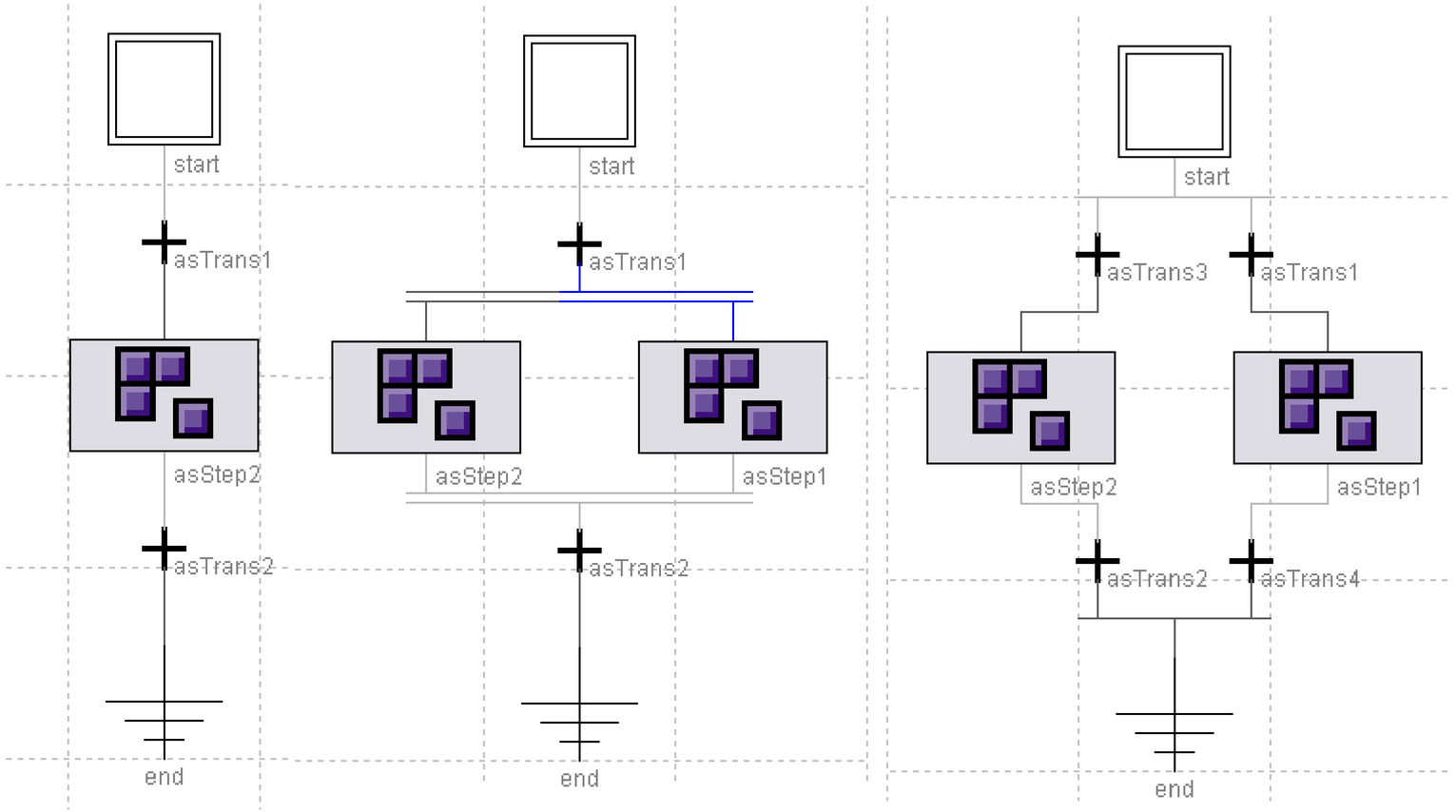
说明：

1. 变量可以用来关联某个步骤上的Activity的输入和输出项
2. 如果你将某个步骤的输出项与另一个步骤的输入项都指向同一个变量，这样就可以在步骤间传递值
3. 你还可以创建变量，该变量不与Activity的输入、输出关联。该变量可以用作转换条件

Activity Set开发

- Activity Set**介绍**

- SFC类型

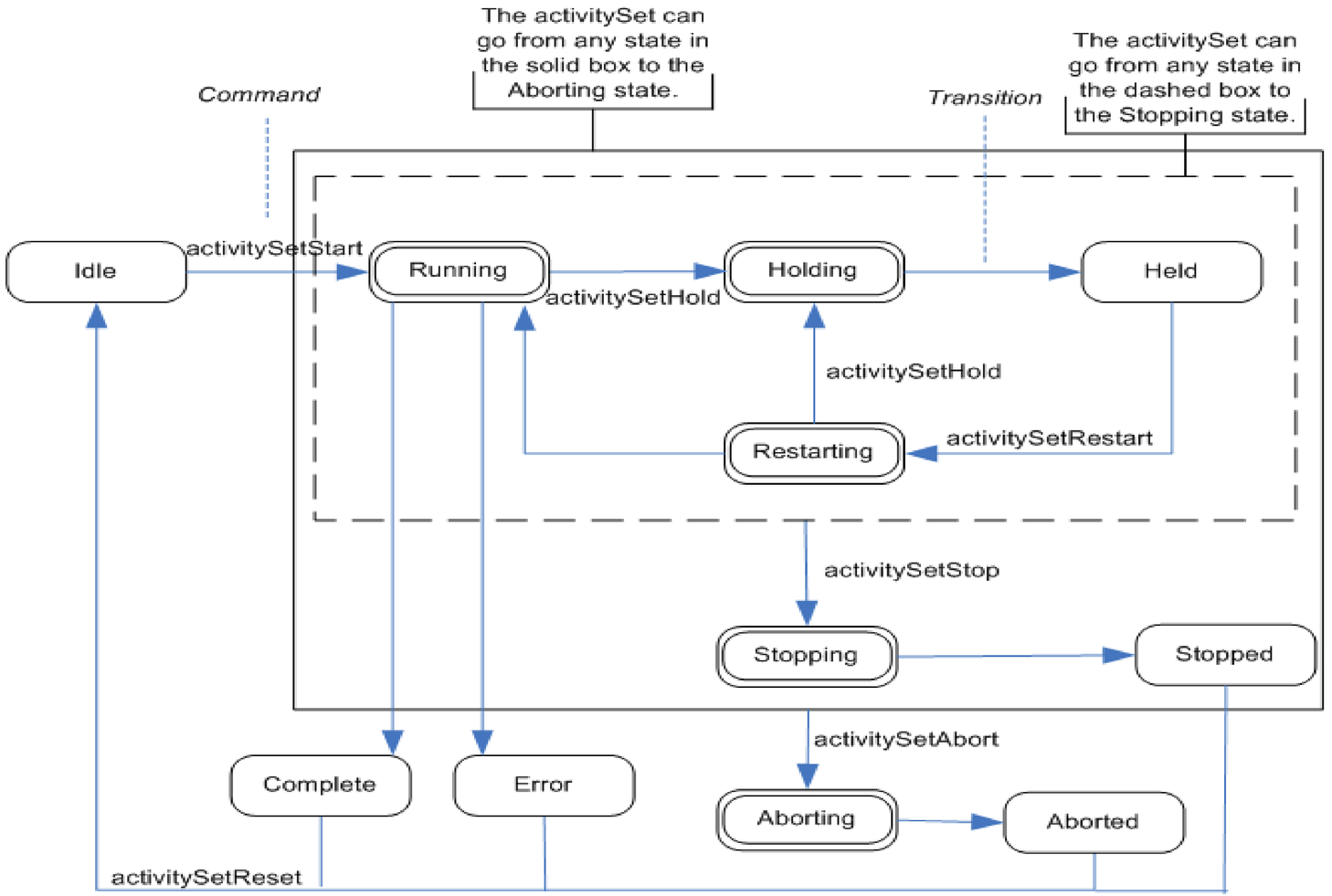


顺序　　并行　　分支

Activity Set开发

- **S88状态模型**

- S88状态模型



Activity Set开发

- Activity Set**开发样例（1）**

- 步骤一

	Activity类	备注
1	CalculatorInputActivity	2个输入项：Input Number1, Input Number2; 2个输出项：Operand1, Operand2;
2	CalculatorOperAddActivity	2个输入项：Operand1,Operand2 1个输出项：Result
3	CalculatorOpeMinusActivity	2个输入项：Operand1,Operand2 1个输出项：Result
4	CalculatorOperMultiActivity	2个输入项：Operand1,Operand2 1个输出项：Result
5	CalculatorOperDivideActivity	2个输入项：Operand1,Operand2 1个输出项：Result
6	CalculatorOutputActivity	1个输入项：Result 1个输出项：Result

Activity Set开发

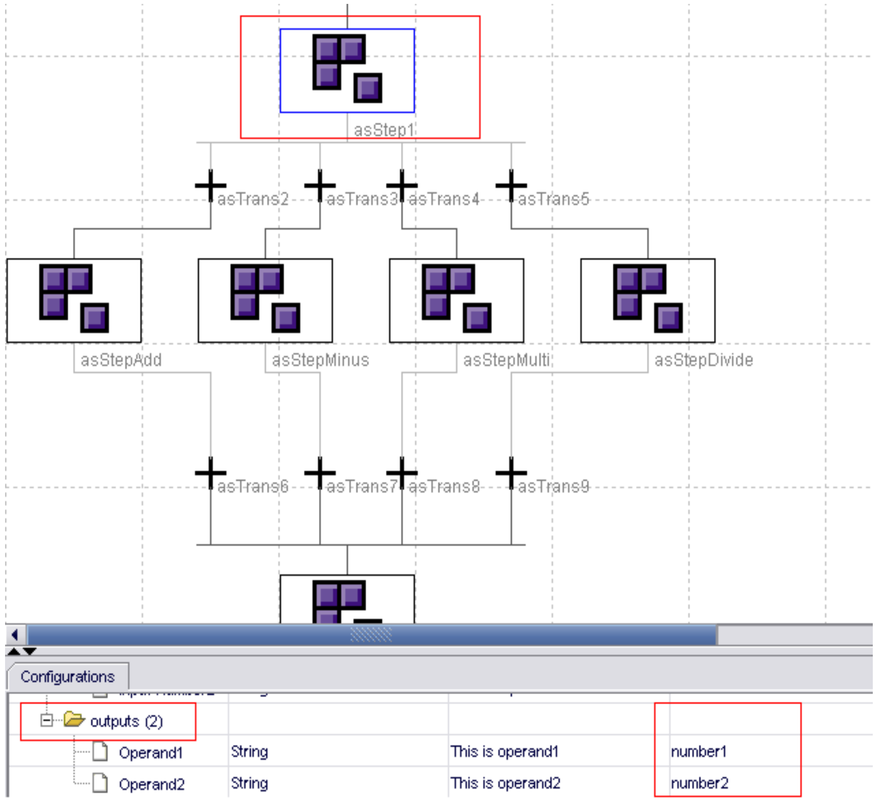
- Activity Set**开发样例（2）**

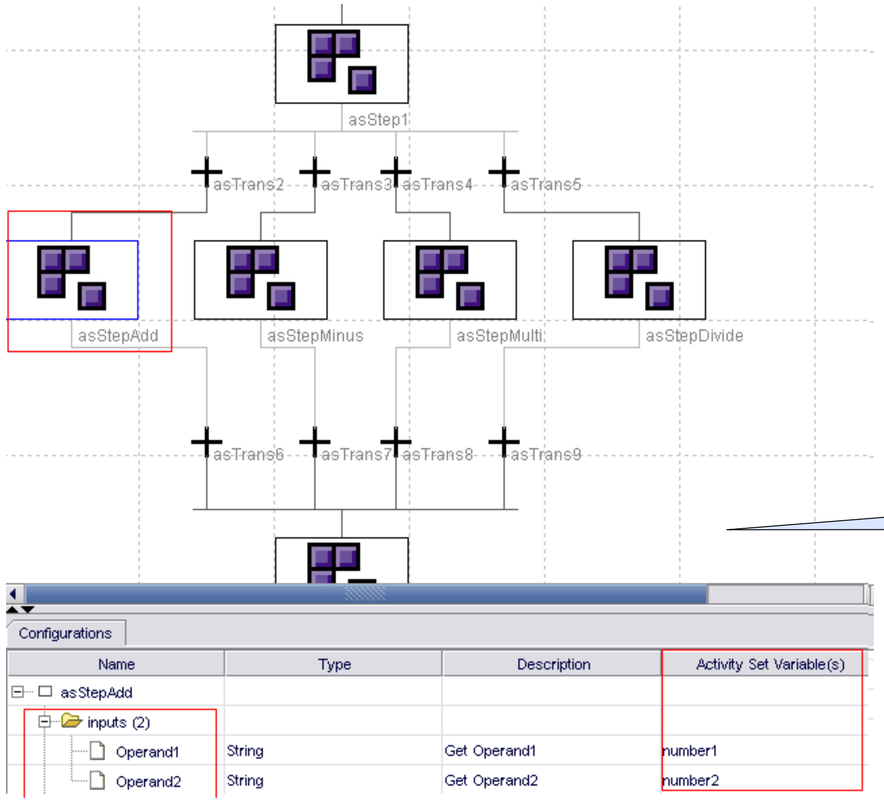
- 步骤二：将6个Activity打包成6个jar文件
- 步骤三：在PD中创建6个Activity对象
- 步骤四：在PD中创建1个Activity Set对象
- 步骤五：在Activity Set中画好图像
- 步骤六：创建变量
- 步骤七： 对每个Activity，映射它的输入、输出项到Activity Set的变量
- 步骤八： 定义转换条件
- 步骤九： 将Activity Set添加到Form
- 步骤十： 启动Activity Set

```
number1 = formatText(editNumber1.getText())
number2 = formatText(editNumber2.getText())
oper = formatText(combobox1.getSelectedItem())
activitySetContainerControl1.setValue("number1", number1)
activitySetContainerControl1.setValue("number2", number2)
activitySetContainerControl1.setValue("oper", oper)
activitySetContainerControl1.activityStart()
i = true
while(i){
    result =
activitySetContainerControl1.getOutputItem("finalResult")
    if(result!=null&&(!result.equals(""))) {
        editResult.setText(result)
        i = false
    }
}
activitySetContainerControl1.activityStop()
activitySetContainerControl1.activityReset()
```

Activity Set开发

- Activity Set**开发样例（3）**

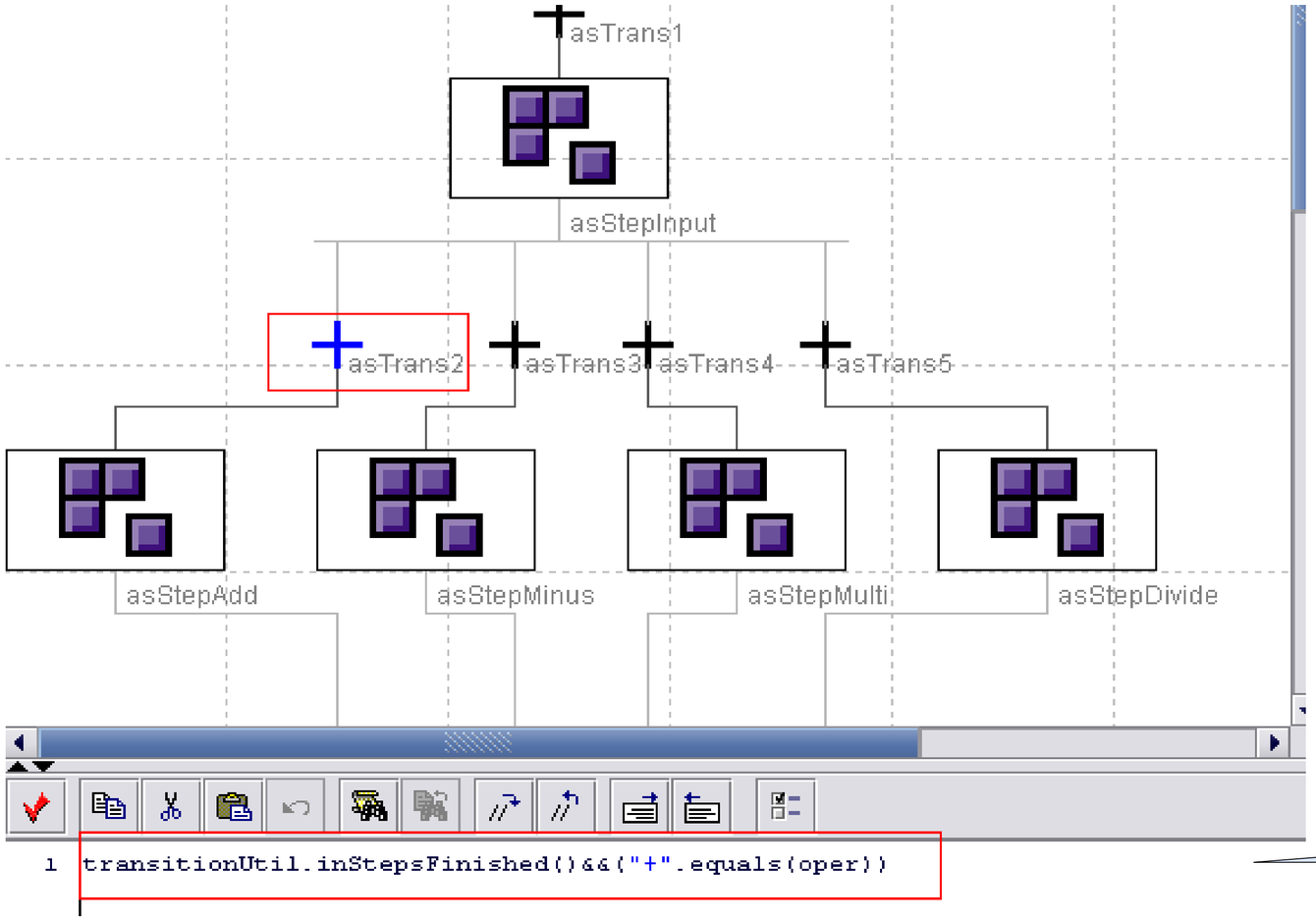




使用变量来连接“输入”与“输出”

Activity Set开发

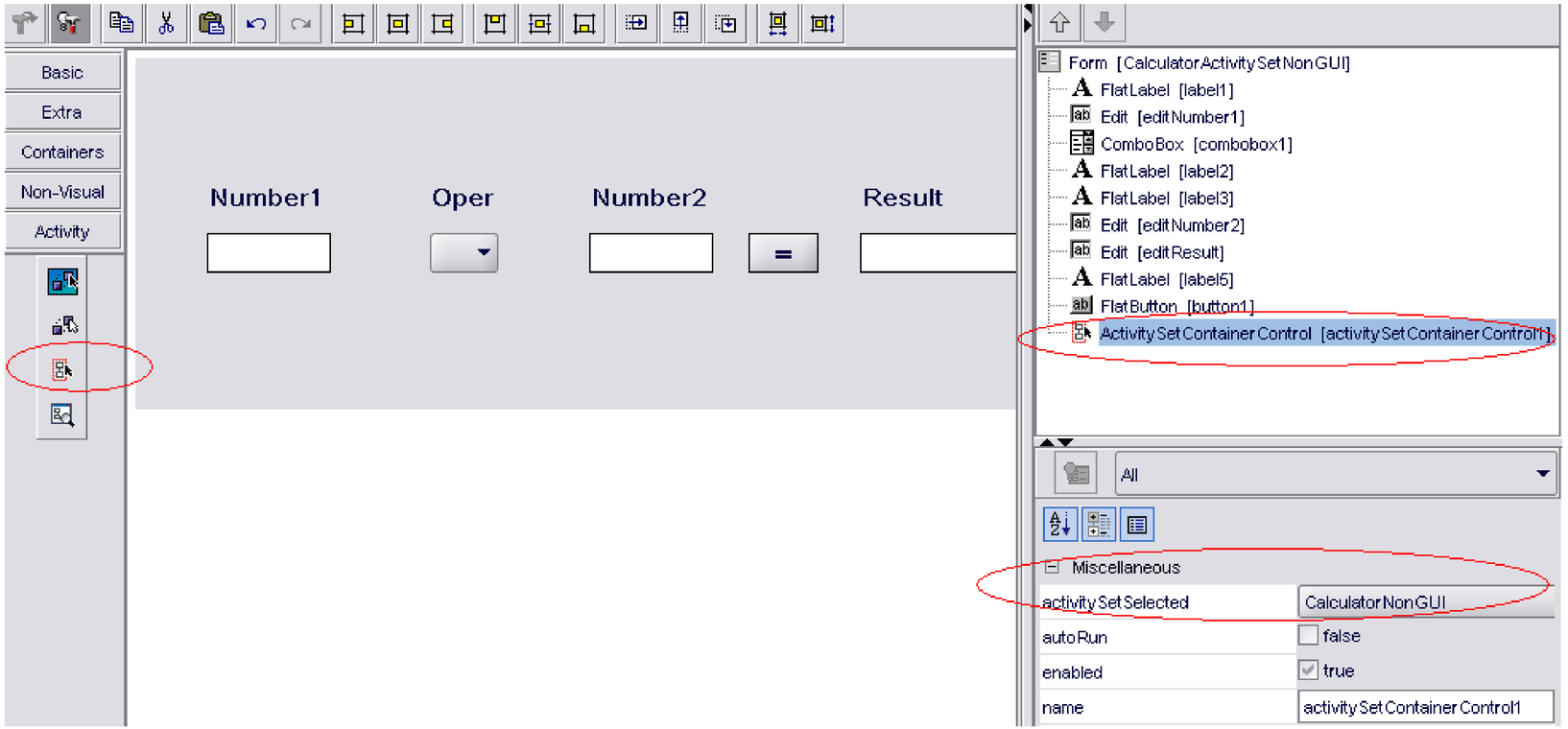
- Activity Set**开发样例（**4**）**



在“转换”中使用变量

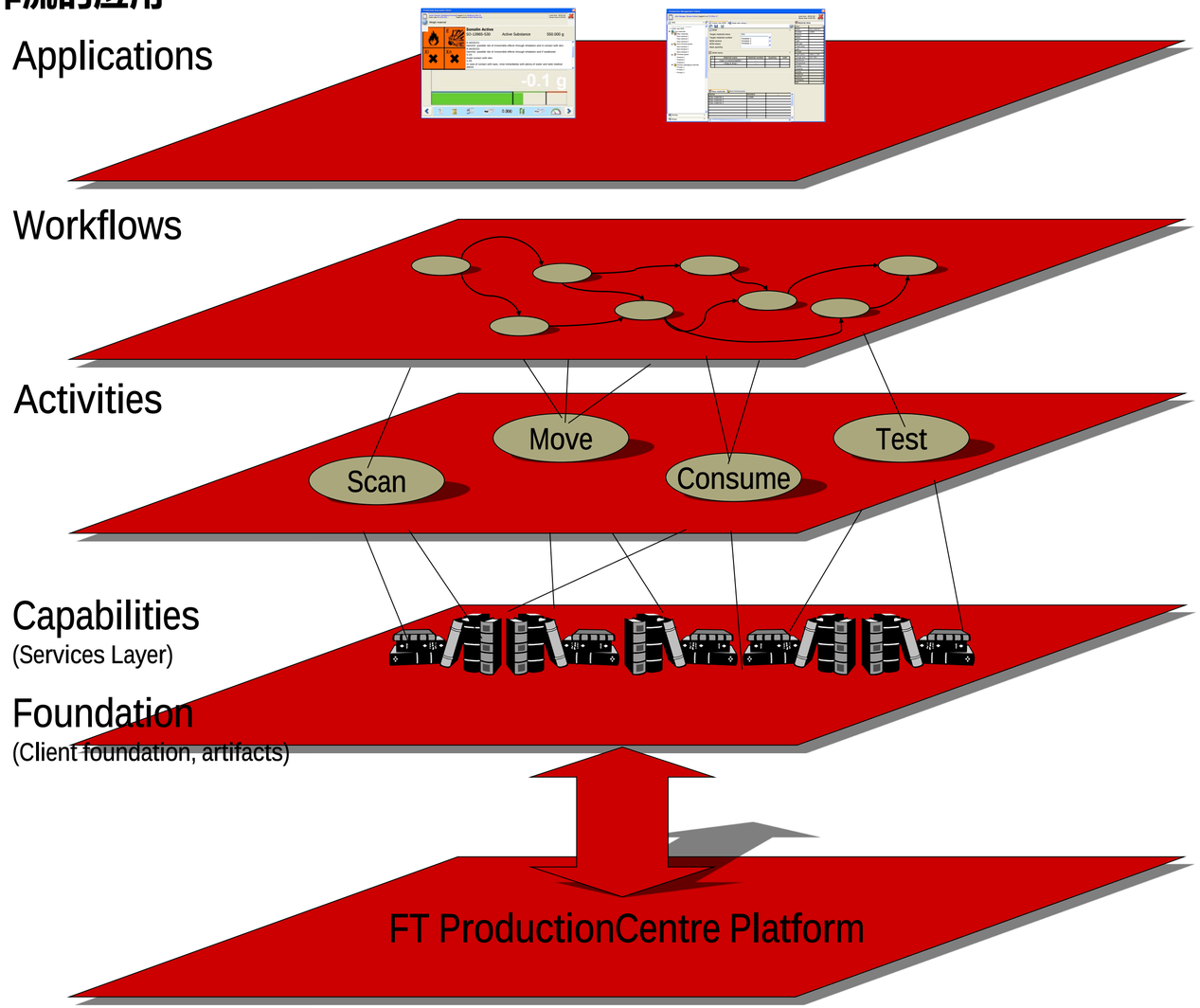
Activity Set开发

- Activity Set**开发样例（5）**



Activity Set开发

- **构建基于工作流的应用**



- Application executes workflow, provides UI based on current steps and allowed tasks
- Workflows model operator actions, automated processes and the business processes
- Reusable components that represent the business logic for any step in a work flow
- Platform Capabilities extended as a library
- Common clients and other foundation artifact ensure reusability

内容

Activity开发

Activity Set开发

QA

QA

Rockwell Automation

TechED™

Inspire. Educate. Innovate.

LISTEN. THINK. SOLVE.®

谢谢

PUBLIC INFORMATION

敬请关注罗克韦尔自动化微信平台

www.rockwellautomationteched.com

Allen-Bradley • Rockwell Software

Rockwell Automation