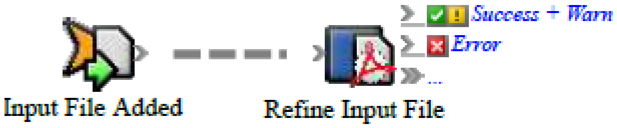
Assigning parameters

You can assign parameters that are not available in Rule Parameters Editor by editing the code.



The code for assigning a parameter to an action looks like this:

```
Public Sub ConfigureAction( _
ByVal triggerEvent As
Creo.PWS.Automation.PrinergyDataModel.InputFileCreatedEvent, _
ByVal action As Creo.PWS.Automation.PrinergyDataModel.RefineAction)
' --- Assign the value of the action parameter "InputFiles" ---
action.InputFiles = triggerEvent.InputFiles
' --- Assign the value of the action parameter "ProcessTemplatePath"
---
action.ProcessTemplatePath = "Storage:Archive:Archive:ArchiveAll"
' --- Assign the value of the action parameter "Comment" ---
' action.Comment = some value expression
End Sub
```

If a parameter is not assigned a value in the user interface, it appears in the code as a comment. You must un-comment it to give it a value.

Assigning static values

A *static* value is a value that is the same every time that a rule is fired, such as the process template to use, the e-mail address to send to, the priority to use, and so on.

```
'--- Assign the value of the Action Parameter "Comment" ----
action.Comment = "This is a test comment"
'--- Assign the value of the Action Parameter "Priority" ----
action.Priority = 2
```

Assigning event properties

Event properties are *dynamic* values—they change every time a rule fires. Assign a property of the event to a parameter of the action:

```
'--- Assign the value of the Action Parameter "Job" ---
action.Job = triggerEvent.Job
'--- Assign the value of the Action Parameter "UserName" ---
action.UserName = triggerEvent.Process.UserName
```

Assigning strings with embedded properties

You may want to construct a string that combines static values with event property values. This is particularly useful for e-mail messages.

```
'--- Assign the value of the Action Parameter "Subject" ---
action.Subject = ("Failed refine in Job " + triggerEvent.Process.Job.Name
+ ".")
```