2. Update Asset workflow post function

Adding post function

Add "[AIP] - Update Asset workflow postfunction" to the workflows to update an asset object on transition. If the condition passes, post function is executed for each asset separately. If an asset is removed from the asset custom field, it will be executed as well. You can control it with assetStatus context parameter. If you do want to do nothing for the removed assets you can control assetStatus == 'removed' and then return null. Please see context parameters details in this document.

Return null if you do not want to change the asset attribute value. Return empty string "" if you want to clear value.

See examples, inventory management and common classes.

Workflow Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condition</td>
<td>Groovy script for the post function condition. Function must return true or condition must be empty to pass. If condition fails post function will be ignored.</td>
</tr>
<tr>
<td>Target Asset custom fields</td>
<td>Specify the custom field to update assets. Leave it blank to update all Asset custom fields.</td>
</tr>
<tr>
<td>Default groovy script</td>
<td>This is the default script to be executed for the options that has no specific groovy script. Type script returning the value to update asset attribute, e.g return issue.summary</td>
</tr>
</tbody>
</table>
| Attributes to be updated (one or more) | **Form:** Define specific Form or select "Any form" to update the assets having the attribute.  
**Attribute:** Target attribute to set it's value |
Groovy Scripts to execute
(one for each attribute definition)

There are 2 options to use as source value:

- **Use default groovy script**: "Default groovy script" result will be set to the attribute value
- **Custom groovy script**: Specify attribute specific groovy script to set as value.

JIRA Custom field value will be added later as 3rd value source.

## Context parameters for Groovy Scripts

Please see [Sample Groovy Scripts to Create/Update Asset workflow post function](https://confluence.snapbytes.com/x/-YZ2AQ) for more examples.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>asset</td>
<td>Access AIP asset class instance. <strong>Only valid for Default Groovy Script and Attribute Groovy Script.</strong> See <a href="https://confluence.snapbytes.com/x/-YZ2AQ">https://confluence.snapbytes.com/x/-YZ2AQ</a>.</td>
</tr>
</tbody>
</table>
| assetStatus   | String value gives status of edit operation of each assets for Asset Custom fields. This value may help for the increase or decrease decision for inventory tracking.
  - **noChange**: asset already exists before transition and it still exists
  - **removed**: asset is removed in this transition
  - **added**: asset was not exist before transition and it is added in this transition.
  - **unknown**: a serious error occurred and failed to determine the status. |
| customFieldManager | Access JIRA Custom Field Manager class. See [https://docs.atlassian.com/software/jira/docs/api/latest/index.html?com/atlassian/jira/issue/CustomFieldManager.html](https://docs.atlassian.com/software/jira/docs/api/latest/index.html?com/atlassian/jira/issue/CustomFieldManager.html) |

**Example:**

```java
loggedInUser == issue.getAssignee()
```

<table>
<thead>
<tr>
<th>DefaultIssueChangeHolder</th>
<th>Default implementation of a change holder. It is used to update a custom field.</th>
</tr>
</thead>
<tbody>
<tr>
<td>aipUtils</td>
<td>Helper class for the post function groovy script. See aipUtils for details</td>
</tr>
</tbody>
</table>

---

### Sample Post Function configuration page
You can add multiple post functions or multiple attributes for one post function.

Try Groovy Scripts

You can immediately execute groovy script so see result. This will let you write and try your groovy scripts faster. Please keep in mind that scripts will be actually executed, if you modify anything please use test objects (issue, asset, etc.)!

Groovy examples for Asset Attribute Types

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Groovy example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckboxList</td>
<td><code>return &quot;@@@ada@@@@@@ist@@@&quot;</code></td>
<td>Return valid option values wrapped with three @ characters. i.e: <code>@@@@ada@@@@@@ist@@@@</code> For a single option value no need to wrap with @ characters.</td>
</tr>
<tr>
<td>DatePicker</td>
<td><code>import java.time.* LocalDateTme t = LocalDateTme.now(); return t.toString()</code></td>
<td>Result must be in ISO format (&quot;yyyy-MM-dd&quot;), i.e: &quot;2018-12-26&quot;. You do not need to do formatting if you use LocalDateTme class as it returns in ISO format by default. Jira Issue's date field example: <code>issue.created.format(&quot;yyyy-MM-dd'T'HH:mm&quot;)</code></td>
</tr>
<tr>
<td>Attribute Type</td>
<td>Example Code</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| DateTimePicker    | ```java
import java.time.*
LocalDateTime t = LocalDateTime.now();
return (t as String)
``` | Result must be in ISO format ("yyyy-MM-ddTHH:mm"), i.e: "2018-12-26T20:20".  
You do not need to do formatting if you use LocalDateTime class as it returns in ISO format by default. If you need to format a date to string, use the format as: "yyyy-MM-dd'T'HH:mm" (Please notice extra single quotes!)  
Jira Issue's date field example:  
- `issue.created.format("yyyy-MM-dd'T'HH:mm")` |
| DropdownList      | `return "ada"`                                   | Return a valid option value.                                                                                                             |
| InventoryList     | `return "3"`                                     | Return reference asset ID                                                                                                               |
| InventoryListByForm| `return "10"`                                   | Return reference asset ID                                                                                                               |
| IP                | `return "10.0.0.2"`                              | Any text is possible, there is no format control.                                                                                       |
| IPv6              | `return "2001:0db8:85a3:0000:0000:8a2e:0370:7334"` | Any text is possible, there is no format control.                                                                                       |
| ListBox           | `return "izm"`                                   | Return a valid option value.                                                                                                             |
| ListBoxMultiple   | `return "@@@ada@@@@ist@@@"`                      | Return valid option values wrapped with three @ characters. i.e: @@@ada@@@@@@ist@@@@  
For a single option value no need to wrap with @ characters. |
| RadioButtonList   | `return "ist"`                                   | Return a valid option value.                                                                                                             |
| Text              | `return issue.summary`                          | Any text is possible.                                                                                                                    |
| TextArea          | `return issue.description`                      | Any text is possible.                                                                                                                    |
| URL               | `return "http://www.snapbytes.com/"`            | Any text is possible, there is no format control.                                                                                       |
| UserPicker        | `return issue.reporter.username`                | Any text is possible, there is no control. You may use issue.reporter.username or issue.assignee.username                             |

**Another post function example for All Attribute types**

You can update all fields of an attribute in one post function.
5. [API] - Update asset's attribute(s), below attribute(s) will be updated:

**Condition:**

```groovy
LoggedInUser == issue.assignee
```

**Default groovy script:**

```groovy
import java.time.*
LocalDateTime t = LocalDateTime.now();
return (t as String) + " " + issue.assignee.username
```

**Target Asset Custom fields:**

- Assets[10000], Assets3[10001], Assets5[10003]

**Form:**

- **All Forms, Attribute:** CheckBoxList Field[64], **Value source:** Custom groovy script = return "bal"
- **Form:** all attributes[11], **Attribute:** DatePicker[65], **Value source:** Custom groovy script = import java.time.*

```groovy
LocalDateTime t = LocalDateTime.now();
return (t as String)
```

- **Form:** all attributes[11], **Attribute:** DateTime Picker[66], **Value source:** Custom groovy script = import java.time.*

```groovy
LocalDateTime t = LocalDateTime.now();
return (t as String)
```

- **Form:** all attributes[11], **Attribute:** DropDownList Field[63], **Value source:** Custom groovy script = return "adv"
- **Form:** all attributes[11], **Attribute:** InventoryList (All Assets List)[67], **Value source:** Custom groovy script = return "3"
- **Form:** all attributes[11], **Attribute:** InventoryListByForm (Assets List By Form(s))[68], **Value source:** Custom groovy script = return "19"
- **Form:** all attributes[11], **Attribute:** IP[69], **Value source:** Custom groovy script = return "16.8.0.2"
- **Form:** all attributes[11], **Attribute:** IPv6[70], **Value source:** Custom groovy script = return "2001:6db8:85a3:0008:8008:8a2e:8378:7334"
- **Form:** all attributes[11], **Attribute:** ListBox[71], **Value source:** Custom groovy script = return "izm"
- **Form:** all attributes[11], **Attribute:** ListBoxMultiple[72], **Value source:** Custom groovy script = return "12001:6db8:85a3:0008:8008:8a2e:8378:7334"
- **Form:** all attributes[11], **Attribute:** Radio Button List[73], **Value source:** Custom groovy script = return "lst"
- **Form:** all attributes[11], **Attribute:** Text[74], **Value source:** Custom groovy script = return "issue.summary"
- **Form:** All Forms, **Attribute:** TextArea[75], **Value source:** Custom groovy script = import inventoryplugin.entity.JiInventory Import

```java
inventoryplugin.entity.JiInventory JIInventoryItem def result = result + "n" + "Asset name: " + asset.getname() result = result + "\n"
```

- **Form:** all attributes[11], **Attribute:** URL[76], **Value source:** Custom groovy script = return "http://www.snapbytes.com/"
- **Form:** all attributes[11], **Attribute:** User Picker[77], **Value source:** Custom groovy script = return issue.reporter.username

Put this post function after issue updates and before GenerateChangeHistoryFunction and re-index post functions.

6. Add a comment to an issue if one is entered during a transition.

7. **Type:** class

```java
Class: com.atlassian.jira.workflow.function.Issue.GenerateChangeHistoryFunction
```