Item State in FOLIO

- What is Item State?
- Considerations for Implementers in 2020 and 2021
  - Alternative: Check In or Check Out Notes
  - Alternative: Dummy Users
  - Alternative: Service Points and Locations

What is Item State?

FOLIO is implementing a three-part item state. The three factors included are Availability; Needed for; and Process.

While initially developed with system-provided values, the goal is to eventually provide libraries who are using a workflow engine the ability to customize these values (and behavior) for their needs.

As of Q1 2020, only Availability is currently implemented, as a field labeled Item status. Note that a few availability items are still being developed - see Availability Implementation List for more details.

Needed for and Process are not yet implemented.

Once developed, the three factors for item state will interact together to drive functionality, and display status information on the item record.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Behavior in FOLIO that is affected by this element</th>
<th>Triggered by</th>
<th>Can you have more than one value?</th>
<th>Is this a required value?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>May describe information about where an item is (available; recently returned) or if it can be circulated.</td>
<td>loan policies, requesting</td>
<td>FOLIO workflows. Availability is not a field that is edited directly - it’s changed by choosing FOLIO workflow options - for example, selecting “Mark as Missing” from the item menu to mark an item as missing.</td>
<td>No</td>
<td>Yes (for physical items)</td>
</tr>
<tr>
<td>Needed For</td>
<td>Allows libraries to assign a path for an item to follow when it is needed for specific things. These may be for staff needs, or for patron needs. Needed For might include Requests; Course Reserves; Binding; etc.</td>
<td>requesting</td>
<td>FOLIO workflow for some values; manually set for other values.</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Process</td>
<td>Describes a staff process that an item is in. Processes might include Digitization; Repair; Cataloging; etc.</td>
<td>loan policies, requesting</td>
<td>FOLIO workflow for some values; manually set for other values.</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Considerations for Implementers in 2020 and 2021

Because Item State is in development, implementing libraries will need to analyze their workflows and identify scenarios where they need to track staff processes without the ability to use item state. In particular, because Needed For and Process are not yet implemented, libraries that need to track behind-the-scenes work like binding will most likely need to use an alternative approach.

Possible features that could be used as alternatives include:

Alternative: Check In or Check Out Notes

Notes are implemented on the item record in the Loan and Availability section. Adding a Check In note displays a pop-up message when an item is checked in; adding a Check Out note displays a pop-up message when an item is checked out.

Advantages:
- This is relatively easy to implement and most library staff already are used to using loan notes in other systems.

Disadvantages:
- Notes must be manually managed on the item record.
- Notes that are created and removed on the same day will likely not be captured in the LDP for reporting purposes, because the LDP (as of Q12020) updates once a day overnight.

Alternative: Dummy Users

"Dummy" users are user records that are not for real people but used for workflow.
So for example, a library that sends items for binding could create a user named “Binding Processes”. Then, when an item is being sent to Binding, it is checked out to that user. When it returns it from binding, you check it in and it removes it from that user.

Advantages:
- You have one place to look for items that are in a particular process state without using a report - simply look at the loans for the dummy user record.

Disadvantages:
- Loaning an item to a user will not in and of itself stop that item from being requestable for other users.
- You may also want the items that are loaned to dummy users to have longer loan periods than to real users.
  - These needs can be taken care of through construction of specific patron groups and circulation rules, but libraries will want to be careful in constructing those rules to make sure FOLIO behaves as they are wanting it to behave.

Alternative: Service Points and Locations

FOLIO has the concept of a permanent location and temporary location for an item (see Effective Location Logic for details). Libraries could use Temporary Locations on items to indicate when items are being handled in staff processes.

So for example, a library that needs to send an item to the Bindery may set a temporary location on that item of “Bindery”.

When the item is returned, the staff member removes the temporary location on the item, and checks the item in. This would put the item in route to its home location.

Advantages:
- This alternative workflow leverages the Circulation Rules system. Rules can be written so that the temporary locations do not allow for requesting or loans.
- Requiring staff to use temporary locations plus Check In and Check Out to move items preserves up-to-date transit information - so you should know if an item is moving in between behind-the-scenes processes and locations.
- For those who export location information to a discovery layer, the use of the temporary location can provide an easy way to signal to patrons where an item is and why it's not available.

Disadvantages:
- This requires staff to be able to comfortable navigate the four-level location hierarchy, and for libraries with lots of processes to manage, it may get unwieldy.
- In order for routing to work, this also requires the creation of behind-the-scenes service points (though you could have one service point for many locations – it is a one to many relationship.)
- As of Q12020, there is no way to do a bulk edit of item records in FOLIO inventory, so managing the temporary location process requires touching the item record directly.