Access to Memory (AtoM) - Bug #13567

Regression: treeview sibling arrangement no longer working

09/23/2021 01:29 PM - Dan Gillean

Status:	Verified		**Start date:**	09/23/2021
Priority:	High		**Due date:**	
Assignee:			**% Done:**	100%
Category:	Treeview		**Estimated time:**	0.00 hour
Target version:	Release 2.7.0			
Google Code Legacy ID:			**Tested version:**	2.7
Sponsored:	No		**Requires documentation:**	No

Description

Works in 2.6.x - broken in qa/2.x currently.

Both treeview versions support the ability to drag-and-drop sibling nodes to rearrange them. Currently this functionality appears to work, but when the page reloads the changes don't stick.

To reproduce

- Log in and find a description hierarchy with multiple siblings under a common parent
- Drag a sibling to a new order in the hierarchy
- Click on the Jobs message that appears to go to the Jobs page - confirm that the job appears to have finished successfully
- Use the job page link back to the description

Resulting error

- The treeview node's new position has not been maintained
- Sibling rearrangement is not working via drag and drop in the treeview

Expected result

- Users can drag and drop sibling nodes to trigger a rearrangement job
- The rearrangement works as expected

History

#1 - 10/19/2021 04:14 PM - Steve Breker

PR: https://github.com/artefactual/atom/pull/1454

This commit fixes an issue where full width treeview move no longer functions due to the ES index (which populates the treeview) lft values becoming out of sync with the database lft values when nodes are moved via the UI. This action was not triggering re-index of affected nodes resulting in treeview errors: 'Mismatch in current position' when the database lft ordering did not match the ES index lft ordering. When this occurs the only fix is to run a search:populate.

The range of treeview node objects that a move action encompasses need to be re-indexed after a treeview move event. Since reindexing the entire parent node could involve a large number of records, this fix only reindexes those objects affected by the move.

The performance of this fix is directly related to the number of nodes affected by the move event - the 'further' the move, the more objects will need reindexing, and the longer the move job will take. If a user refreshes the treeview while the move job is re-indexing it is possible for the UI ordering to appear out of sync temporarily with the DB.

#2 - 10/20/2021 02:14 PM - Steve Breker

- Status changed from New to QA/Review

#3 - 10/29/2021 09:37 AM - Dan Gillean

- Status changed from QA/Review to Verified

- Assignee deleted (Steve Breker)

- % Done changed from 0 to 100

#4 - 11/04/2021 02:32 PM - Steve Breker

PR for B5 fwtv fixes:
https://github.com/artefactual/atom/pull/1469

B5 fixes:

Fix fwtv dnd move by changing the 'is_draggable' callback to correctly
use arrow function.

Fix the 'More' and 'Reset' buttons so they will now appear by removing
B5 button class names from the id section of the button definitions in
the _treewView.php template.

Wrap 'More' and 'Reset' in a div so they could be positioned and styled.