Access to Memory (AtoM) - Bug #13582

CSV import matches wrong archival descriptions when using 'match and update' mode

11/10/2021 05:47 AM - Matthew Addis

Status:	New	**Start date:**	11/10/2021
Priority:	Medium	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
Google Code Legacy ID:		**Tested version:**	
Sponsored:	No	**Requires documentation:**	

Description

We have encountered several cases where CSV import of archival descriptions appears to match the wrong entries in AtoM. This occurs when using the 'match and update' mode both using the CLI and UI. The issue manifests when the identifiers for archival descriptions are quite long and contain forward slashes, e.g. MJA/1/4/2/5/2/10 and when the titles for the archival descriptions are not unique and contain just one or two words that are the same for many archival descriptions, e.g. 'book pages', 'photograph', 'video'. It seems that when AtoM can't match on legacyId/source_name that it tries to match on a combo of identifier/title/repository and whilst the identifiers for new archival descriptions are actually unique they erroneously get matched to existing descriptions. My guess is this is due to the way ElasticSearch is used, e.g. the identifier gets tokenised because of the / and that then results in matches which shouldn't happen. I tried setting the 'Escape special chars from searches' field in the AtoM global settings, but that didn't make a difference.

The problem can be replicated using the attached metadata.csv. This was generated by a script and contains a deep tree of archival descriptions. Create a repo in AtoM called 'Repo1' and then upload the CSV using 'match and update'. The first 500 archival descriptions are created OK. Then the matching problem occurs, e.g. the job report contains lines like this:

Row 501: Matching description found, updating in place; row (id: 7102, culture: en, legacyId: MJA/1/2/1/5/2/6)...; Row 502: Matching description found, updating in place; row (id: 7103, culture: en, legacyId: MJA/1/2/1/5/2/7)...;

A look in the keymap table in mysql shows entries like this:

```
mysql> select * from keymap where target_id = 7102;
+-----------------+-----------+-------------+--------------------+------+---------------+
| source_id       | target_id | source_name | target_name        | id   | serial_number |
+-----------------+-----------+-------------+--------------------+------+---------------+
| MJA/1/1/2/5/2/6 |      7102 | metadata.csv | information_object | 3739 |             0 |
| MJA/1/2/1/5/2/6 |      7102 | metadata.csv | information_object | 3911 |             0 |
+-----------------+-----------+-------------+--------------------+------+---------------+
```

This shows that MJA/1/1/2/5/2/6 and MJA/1/2/1/5/2/6 are both mapped to the same target archival description. The first entry corresponds to row 331 in the CSV file and AtoM correctly created an archival description with identifer=MJA/1/1/2/5/2/6. The second entry is from row 501 in the CSV and is when AtoM erroneously matched MJA/1/2/1/5/2/6 to the existing description for MJA/1/1/2/5/2/6. Note that the identifiers are similar and have the second and third digits in a different order, but they are unique. Therefore, AtoM should have created a new archival description rather than matched to the existing one. The end result is that the first archival description is overwritten by the second. You can see this by going to the slug (mja-1-1-2-5-2-6) which is left unchanged, and seeing that the identifier is overwritten, i.e. is set to MJA/1/1/2/5/2/6.

The problem occurs on AtoM 2.5.4 and 2.6.

I haven't done enough testing to narrow down the minimum case that will cause the error. It doesn't seem to occur unless the identifiers are quite long (in the sense of lots of / separators) and there are already lots of archival descriptions loaded into the system. As noted above, my guess is that ElasticSearch is tokenising on / because this isn't escaped (

https://github.com/artefactual/atom/blob/qa/2.x/lib/model/QubitInformationObject.php#L2162)

From an end-user point of view, the mismatch problem is a real issue. For example, we have AtoM users that do things such as export a CSV from the clipboard for a hierarchy of archival descriptions, edit some of the metadata, e.g. scope and content, and then upload the whole CSV back to AtoM as a CSV import. They do this as a way of bulk editing a set of archival descriptions. However, some of archival descriptions are wrongly matched and the wrong archival descriptions are updated. This then results in a mismatch between slugs and reference codes. Archival descriptions appear to have 'moved to random other places in the tree'. This makes detection and fixing of the problem hard (unless you have direct access to the mysql database and can dig around in the keymap table).

The only workaround I have found so far is to make sure that the title field is unique for all archival descriptions, e.g. by embedding a UUID or reference code etc. But this isn't a workable solution for many AtoM users and they can't/don't want to edit thousands of existing entries.

Files

metadata.csv	111 KB	11/10/2021	Matthew Addis