

## EDIT - feature request #9111

### Handle Malformed URIs

06/30/2020 03:39 PM - Katja Luther

<b>Status:</b>	Rejected	<b>Start date:</b>	
<b>Priority:</b>	Highest	<b>Due date:</b>	
<b>Assignee:</b>	Katja Luther	<b>% Done:</b>	90%
<b>Category:</b>	taxeditor	<b>Estimated time:</b>	0:00 hour
<b>Target version:</b>			
<b>Severity:</b>	normal		
<b>Description</b>			
mail WB:			
ich habe hier wieder so eine URL die ich nicht als URI abspeichern kann: <a href="https://bioone.org/journals/Cactus-and-Succulent-Journal/volume-78/issue-2/0007-9367(2006)78[66:FKASLA]2.0.CO;2/Finders-Keepers-and-some-Lavranian-Adjustments-in-Mesembryanthema/10.2985/0007-9367(2006)78[66:FKASLA]2.0.CO;2.full">https://bioone.org/journals/Cactus-and-Succulent-Journal/volume-78/issue-2/0007-9367(2006)78[66:FKASLA]2.0.CO;2/Finders-Keepers-and-some-Lavranian-Adjustments-in-Mesembryanthema/10.2985/0007-9367(2006)78[66:FKASLA]2.0.CO;2.full</a>			
Wenn ich das richtig verstehe, ist der Grund, dass sich bioone.org nicht an die Regeln hält. Das ist aber leider eine wichtige Quelle für botanische Publikationen, open access oder [noch] nicht (siehe <a href="https://bioone.org/browse/subject/Plant-Sciences">https://bioone.org/browse/subject/Plant-Sciences</a> ) und meine Browser können alle damit umgehen. Da sie auch eine DOI vergeben, kann ich die URL dort ablegen, aber nur bei der Referenz, nicht als Original Publication in Details.			
<b>Related issues:</b>			
Related to EDIT - feature request #9114: Handle malformed URIs in user type		<b>Closed</b>	

### Associated revisions

#### Revision cc71e44c - 06/30/2020 02:59 PM - Katja Luther

encode invalid character in URI

#### Revision 9916da4a - 01/06/2021 02:18 PM - Andreas Müller

ref #9111, ref #9114 remove TaxEditor specific URI encoding code as encoding is handled in cdmlib now

### History

#### #1 - 06/30/2020 03:40 PM - Katja Luther

suggestion of AK:

ich halte es schon für richtig, generell URIs zu verwenden, sollten aber nicht erlaubte Zeichen im hierarchischen Teil der URIs encodieren um Probleme beim erzeugen der URI zu vermeiden.

Wenn new URI(uriStr) scheitert könnten wir vielleicht folgendes machen:

1. url = new URL(uriStr)
2. wenn das gut geht: decodedPathElements = url.getPath().split("/").stream().forEach(e -> URIDecoder.encode(e)).collect(Collectors.toList()); für die Fragment und Query Teile der URL entsprechend das selbe
3. die url neu zusammen bauen und toURI() ausführen.

#### #2 - 06/30/2020 03:41 PM - Katja Luther

We already have the UrlUtf8Coder in cdmlib.commons

#### #3 - 06/30/2020 03:42 PM - Katja Luther

- Status changed from New to Resolved

- Assignee changed from Katja Luther to Andreas Müller

please review.

#### #4 - 06/30/2020 04:23 PM - Andreas Müller

- Related to feature request #9114: Handle malformed URIs in user type added

**#5 - 06/30/2020 04:25 PM - Andreas Müller**

- *Status changed from Resolved to Closed*
- *Assignee changed from Andreas Müller to Katja Luther*
- *Priority changed from New to Highest*
- *% Done changed from 0 to 100*

This works well. I created a new ticket for moving the code to cdmlib: [#9114](#)

**#6 - 01/06/2021 02:20 PM - Andreas Müller**

- *Status changed from Closed to Resolved*
- *% Done changed from 100 to 90*

I removed the related code again as [#9114](#) is now more or less implemented and URI encoding is now done in cdmlib. Please verify this is correct.

**#7 - 04/19/2021 02:35 PM - Katja Luther**

- *Status changed from Resolved to Rejected*
- *Target version deleted (Release 5.16)*

this is handled in cdmlib now, therefore I close this ticket as rejected.