

EDIT - bug #1119

[PARSER] Duplicate inferences created during parsing are NOT merged when data gets saved

09/30/2009 10:46 AM - Andreas Müller

Status:	Duplicate	Start date:	
Priority:	Priority14	Due date:	
Assignee:	Andreas Müller	% Done:	0%
Category:	taxeditor	Estimated time:	0:00 hour
Target version:		Found in Version:	
Severity:	major		
Description			
Probably Niels knows this.			
Needs to be implemented if not yet done			
Related issues:			
Related to EDIT - bug #7829: Improve deduplication of parsed names and refere...			Closed
Is duplicate of EDIT - feature request #9085: Improve deduplication of parsed...			Closed

History

#1 - 10/12/2009 06:43 PM - Niels Hoffmann

- Status changed from New to In Progress

No, I don't know this. Can I get a little more insight?

#2 - 10/14/2009 12:32 AM - Andreas Müller

What I meant is the following: if you parse a name you create a name, a nom ref some authors and may be a nom status. The nom ref may have an inref (e.g. a journal for an article). Do you check if this inref does already exist (can be matched) before you save the name?

#3 - 10/14/2009 05:21 PM - Niels Hoffmann

- Target version deleted (TaxEditor Release 2.0)

I looked into this. Implementation is postponed to after the migration to the new reference model.

Reason for this is mainly the non genericness of how to get the inReference. Every reference class has a differently named method for it which makes it firstly cumbersome to implement and secondly everything has to be redone once the new reference model is in place.

#4 - 01/25/2010 03:04 PM - Niels Hoffmann

InReferences are troubling.

This is what I found out so far:

- The inReference itself does not get persisted correctly. A Journal is correctly recognized during parsing but gets persisted with cacheStrategy set to generic. DatePublished is another field that does not get persisted though it was recognized correctly during parsing. Because of this, the default match strategy does not match. We could set the fields to ignore in the match strategy, but I'd rather want to know what is happening there in the first place.
- Duplicate resolving of the author is not taking place when checking references with inReferences. I have not found out so far what is going wrong here.

Do we have unit/integration tests that guarantee proper behavior of the matching mechanism running against references with inReferences?

#5 - 01/26/2010 06:23 PM - Niels Hoffmann

- Tracker changed from task to bug

I am refiling this as a bug, because inReferences are producing duplicate references as well as duplicate authors.

#6 - 01/26/2010 06:24 PM - Niels Hoffmann

- Subject changed from *Check if duplicate inferences created during parsing are merged when data get saved* to *Duplicate inferences created during parsing are NOT merged when data get saved*

#7 - 01/26/2010 06:24 PM - Niels Hoffmann

- Subject changed from *Duplicate inferences created during parsing are NOT merged when data get saved* to *Duplicate inferences created during parsing are NOT merged when data gets saved*

#8 - 01/28/2010 02:37 PM - Andreas Müller

Replying to [n.hoffmann](#):

InReferences are troubling.

This is what I found out so far:

- The inReference itself does not get persisted correctly. A Journal is correctly recognized during parsing but gets persisted with cacheStrategy set to generic. DatePublished is another field that does not get persisted though it was recognized correctly during parsing. Because of this, the default match strategy does not match. We could set the fields to ignore in the match strategy, but I'd rather want to know what is happening there in the first place.
- Duplicate resolving of the author is not taking place when checking references with inReferences. I have not found out so far what is going wrong here.

Do we have unit/integration tests that guarantee proper behavior of the matching mechanism running against references with inReferences?

Hi Niels,

habe mir gerade nochmal das Ticket genauer angeschaut und deinen Kommentar vom 25.1. richtig gelesen, wo du ja einen Teil meiner gerade gestellten Fragen bereits beantwortest.

Dennoch müsste ich auch erstmal ziemlich genau schauen, inwiefern es ein Matching Fehler ist.

In eu.etaxonomy.cdm.persistence.dao.hibernate.common.CdmGenericDaoImplTest findest du den Test findMatching. Da ist einiges abgehandelt u.a. auch DatePublished und Autoren. Du kannst ja mal schauen ob dein Fall auch abgedeckt ist.

Den Satz: "A Journal is correctly recognized during parsing but gets persisted with cacheStrategy set to generic." Verstehe ich nicht. Wie kann cacheStrategy to generic gesetzt werden?

So weit erstmal.

Gruß,

Andreas M.

#9 - 02/01/2010 05:25 PM - Niels Hoffmann

- Priority changed from Priority14 to Priority11

"title" was set to @MatchMode.EQUAL_REQUIRED@. For newly created (just typed in) references with inReferences, the title for the inReference was still null and therefore did not match. Also lots of issues with the correct cache strategy.

The problem seems solved, but I will leave this ticket open until it is finally confirmed successfully fixed.

#10 - 04/30/2010 07:54 PM - Niels Hoffmann

- Status changed from In Progress to New

#11 - 05/21/2010 05:44 PM - Niels Hoffmann

- Priority changed from Priority11 to Priority08

#12 - 12/20/2010 12:13 PM - Niels Hoffmann

- Priority changed from Priority08 to Highest

#13 - 12/20/2010 12:13 PM - Niels Hoffmann

- Status changed from New to In Progress

#14 - 01/07/2011 06:10 PM - Niels Hoffmann

- Subject changed from *Duplicate inferences created during parsing are NOT merged when data gets saved* to *[PARSER] Duplicate inferences*

created during parsing are NOT merged when data gets saved

#15 - 06/29/2011 10:55 AM - Niels Hoffmann

- Priority changed from Highest to Priority14

#16 - 09/23/2017 02:18 PM - Andreas Müller

- Description updated

- Assignee changed from Niels Hoffmann to Andreas Müller

- Target version changed from TaxEditor Next Major Release to Release 4.11

If this is still open we should try to fix it soon.

#17 - 11/08/2017 11:26 AM - Andreas Müller

- Target version changed from Release 4.11 to Release 4.12

#18 - 01/20/2021 06:14 PM - Andreas Müller

- Private changed from Yes to No

#19 - 01/20/2021 06:23 PM - Andreas Müller

- Status changed from In Progress to Duplicate

- Target version deleted (Release 4.12)

This has been fully fixed in [#9085](#) so close this ticket as duplicate

#20 - 01/20/2021 06:23 PM - Andreas Müller

- Is duplicate of feature request #9085: Improve deduplication of parsed names added

#21 - 01/20/2021 07:25 PM - Andreas Müller

- Related to bug #7829: Improve deduplication of parsed names and references added