

## EDIT - feature request #7932

### Implement subject resolver for CdmPreferences

11/29/2018 01:07 PM - Andreas Müller

|   |                |                        |           |
|---|----------------|------------------------|-----------|
| <b>Status:</b>  | Resolved       | <b>Start date:</b>     |           |
| <b>Priority:</b>  | Priority14     | <b>Due date:</b>       |           |
| <b>Assignee:</b>  | Andreas Müller | <b>% Done:</b>         | 30%       |
| <b>Category:</b>  | cdmlib         | <b>Estimated time:</b> | 0:00 hour |
| <b>Target version:</b>  | Release 5.46   |                        |           |
| <b>Severity:</b>  | normal         |                        |           |
| <b>Description</b>  |                |                        |           |
| <p>Finding the best matching preference for preferences with same predicates but differing subjects is currently not yet implemented or only limited for subjects with taxonomic subtree preferences.</p> <p>As handling preferences in application has shown that there is a need for caching preferences it is preferred to implement a best matching subject resolver that is DB independent.</p> <p>This creates difficulties for tree filters like the taxonomic subtree filter where the subject may include the subtree uuid or the tree index. Using the subtree uuid is easier to maintain but requires on the fly creation of the tree index because only with the tree index one may be able to decide if a taxon is included or not. Using the tree index as part of the subject avoids this problem but is more difficult to maintain if a taxon node is moved or if it's higher taxonomy changes.</p> <p>There are multiple possibilities to implement resolving</p> <ol style="list-style-type: none"><li>1. easiest: all matching subjects must completely be included in the query subject, the longest subject wins</li><li>2. middle: all parts of a matching subject must be in the query subject in the same order, but the query subject may include additional parts, not only at the end but also in the middle; for finding the best matching we start from the back, if the last part exists it is the best matching otherwise call recursively with the query subject shortened by the last part</li><li>3. general: to match the query subject all parts of the subject must also be parts in the query subject <b>no matter which order</b>; finding the best matching works like in 2), the query subject is read from behind and whenever a subject includes the last part read it wins, otherwise start recursively</li></ol> |                |                        |           |
| <b>Related issues:</b>  |                |                        |           |
| Related to EDIT - feature request #7920: Possibility to define area specific ...  |                | <b>Feedback</b>        |           |
| Related to EDIT - feature request #8256: Show all status preferences as table   |                | <b>Feedback</b>        |           |
| Related to EDIT - feature request #8309: Implement preferences with predicate...  |                | <b>New</b>             |           |

### Associated revisions

#### Revision b45c1f2d - 11/29/2018 01:18 PM - Andreas Müller

ref #7932 first implementation of PreferenceResolver and some adaption to PreferenceService API.

- Correct handling of subtree subjects still open.
- Also creation of user defined subjects still open.

#### Revision fa1dc0a8 - 03/29/2019 12:50 PM - Katja Luther

ref #7932: add getBestMatching method using preferenceResolver in preferenceCache

#### Revision 12b0669d - 03/29/2019 12:50 PM - Katja Luther

ref #7932: add getBestMatching method using preferenceResolver in preferenceCache

### History

#### #1 - 11/29/2018 01:08 PM - Andreas Müller

For now we implement solution 2.

#### #2 - 01/18/2019 08:56 PM - Andreas Müller

- Related to feature request #7920: Possibility to define area specific status selection added

#### #3 - 01/18/2019 08:58 PM - Andreas Müller

- Description updated

**#4 - 01/25/2019 12:59 PM - Andreas Müller**

- Status changed from New to In Progress

**#5 - 01/25/2019 01:44 PM - Andreas Müller**

- % Done changed from 0 to 30

Katja, Andreas, could you please integrate the resolver into your DBpref stores/caches? It is mostly implemented.

**#6 - 02/14/2019 10:38 PM - Andreas Müller**

- Target version changed from Release 5.5 to Release 5.6

**#7 - 02/28/2019 10:54 AM - Andreas Müller**

- Priority changed from New to Highest

**#8 - 04/12/2019 12:58 PM - Andreas Müller**

- Target version changed from Release 5.6 to Release 5.7

This is mostly done but untested therefore move to 5.7

**#9 - 05/03/2019 02:57 PM - Andreas Müller**

- Target version changed from Release 5.7 to Release 5.8

**#10 - 07/18/2019 04:19 PM - Andreas Müller**

Andreas K., do you already use the subject resolver in the preference cache you use for vaadin?

**#11 - 07/18/2019 04:23 PM - Andreas Müller**

- Related to feature request #8256: Show all status preferences as table added

**#12 - 07/19/2019 07:44 AM - Andreas Müller**

- Related to feature request #8309: Implement preferences with predicate and subject in local preferences added

**#13 - 07/19/2019 08:32 AM - Andreas Kohlbecker**

Andreas Müller wrote:

Andreas K., do you already use the subject resolver in the preference cache you use for vaadin?

I guess you are referring by this to the PreferenceResolver. This is not being used in cdm-vaadin

**#14 - 07/19/2019 09:13 AM - Andreas Müller**

Andreas Kohlbecker wrote:

Andreas Müller wrote:

Andreas K., do you already use the subject resolver in the preference cache you use for vaadin?

I guess you are referring by this to the PreferenceResolver. This is not being used in cdm-vaadin

So we should create a ticket for this. Using preferences without handling the subjects correctly may create problems. And we already started with using different subjects for the same predicate in some databases.

**#15 - 07/09/2021 09:46 AM - Andreas Müller**

- Status changed from In Progress to Resolved

- Target version changed from Release 5.8 to Release 5.25

Needs to be reviewed and then moved back to 5.8. Open issues (also the one from note-14 should be moved to new ticket).

**#16 - 07/16/2021 02:48 PM - Andreas Müller**

- Target version changed from Release 5.25 to Release 5.45

**#17 - 09/20/2021 12:20 PM - Andreas Müller**

- Priority changed from Highest to Priority14

**#18 - 10/26/2021 01:03 PM - Andreas Müller**

- Tags set to preferences

**#19 - 10/26/2021 02:26 PM - Andreas Müller**

- Tags deleted (preferences)

- Description updated

**#20 - 10/26/2021 02:26 PM - Andreas Müller**

- Tags set to preferences

**#21 - 02/10/2022 11:20 AM - Andreas Müller**

- Target version changed from Release 5.45 to Release 5.46