

Arvados - Story #15067

[Workbench 2] Update property editing to use IDs

04/03/2019 05:37 PM - Tom Morris

Status: Resolved	Start date: 11/12/2019
Priority: Normal	Due date:
Assigned To: Lucas Di Pentima	% Done: 100%
Category:	Estimated time: 0.00 hour
Target version: 2019-11-20 Sprint	
Description See parent ticket for description of what needs to be done.	
Subtasks: Task # 15597: Review 15067-tag-editing-by-ids (wb2 repo) & 15067-wb2-vocabulary-doc (ar... Resolved	
Related issues:	
Related to Arvados - Bug #15407: [WB2] Property keys on collections are getti...	Resolved 08/19/2019
Related to Arvados - Story #15071: Design new vocabulary file format	Resolved
Blocks Arvados - Story #14151: Extend vocabulary support for properties to su...	Resolved

Associated revisions

Revision 3e56db62 - 11/15/2019 05:12 PM - Lucas Di Pentima

Merge branch '15067-tag-editing-by-ids'
Refs #15067

Arvados-DCO-1.1-Signed-off-by: Lucas Di Pentima <ldipentima@veritasgenetics.com>

Revision aae18e11 - 11/15/2019 08:32 PM - Lucas Di Pentima

Merge branch '15067-wb2-vocabulary-doc'
Closes #15067

Arvados-DCO-1.1-Signed-off-by: Lucas Di Pentima <ldipentima@veritasgenetics.com>

History

#1 - 04/03/2019 05:47 PM - Tom Morris

- Parent task deleted (#14151)

#2 - 04/03/2019 05:47 PM - Tom Morris

- Tracker changed from Task to Story

#3 - 04/03/2019 05:47 PM - Tom Morris

- Blocks Story #14151: Extend vocabulary support for properties to support strong identifiers and multiple labels added

#4 - 04/03/2019 05:47 PM - Tom Morris

- Description updated

#5 - 04/03/2019 05:48 PM - Tom Morris

- Target version changed from To Be Groomed to Arvados Future Sprints

- Story points set to 3.0

#6 - 05/29/2019 01:50 PM - Tom Morris

- Subject changed from Update property editing to use IDs to [Workbench 2] Update property editing to use IDs

#7 - 07/31/2019 03:15 PM - Lucas Di Pentima

- Related to Bug #15407: [WB2] Property keys on collections are getting translated from snake_case to camelCase added

#8 - 08/28/2019 02:58 PM - Tom Morris

- Target version changed from Arvados Future Sprints to 2019-09-11 Sprint

#9 - 08/28/2019 02:58 PM - Tom Morris

- Assigned To set to Lucas Di Pentima

#10 - 08/28/2019 02:59 PM - Tom Morris

- Related to Story #15071: Design new vocabulary file format added

#11 - 08/28/2019 07:10 PM - Lucas Di Pentima

- Status changed from New to In Progress

#12 - 09/11/2019 02:58 PM - Lucas Di Pentima

- Target version changed from 2019-09-11 Sprint to 2019-09-25 Sprint

#13 - 09/25/2019 03:01 PM - Lucas Di Pentima

- Target version changed from 2019-09-25 Sprint to 2019-10-09 Sprint

#14 - 10/09/2019 03:12 PM - Tom Morris

- Target version changed from 2019-10-09 Sprint to 2019-10-23 Sprint

#15 - 10/23/2019 02:46 PM - Lucas Di Pentima

- Target version changed from 2019-10-23 Sprint to 2019-11-06 Sprint

#16 - 11/05/2019 02:18 PM - Lucas Di Pentima

Updates at commit:e92ec73 - branch 15067-tag-editing-by-ids (wb2's repo)

I'm really struggling trying to make the view show the key/value labels but use their ids when adding a tag. I believe the handleSelect callback may be useful to somehow stash the selected key/value's id somewhere so it can be retrieved at the moment the Add button is clicked.

Already tried the other way around: handling the IDs but I think it's not possible to make the Input component to show the labels because it needs a string while calling onChange and that string is what it'll show on the view.

Any idea will be most appreciated!

#17 - 11/06/2019 04:11 PM - Lucas Di Pentima

- Target version changed from 2019-11-06 Sprint to 2019-11-20 Sprint

- Story points changed from 3.0 to 1.0

#18 - 11/07/2019 05:15 PM - Eric Biagiotti

My initial impression is that there has to be a way to have form submit with an object. Maybe this will help?

<https://redux-form.com/7.0.2/docs/faq/customcomponent.md/> Also maybe take a look in view-components/form-fields. These might already do something similar.

If this is impossible, we could connect with the dispatcher and update some other state with the the data we need and use that when the form s submitted. <https://redux-form.com/7.0.2/docs/faq/howtoconnect.md/> This should probably be our last resort though.

#19 - 11/12/2019 03:00 PM - Lucas Di Pentima

Updates at commit:796a3ce0 (WB2 repo)

I've been making lots of progress: The properties' ids are saved on hidden fields on the form so they're available on submission at the same time that the visible fields keep showing the labels. The property "chips" render the properties by their labels when possible, keeping the "copy to clipboard" feature as before, that is, copying the real data to the clipboard (it's trivial to change it).

Also, when some key or value has more than 1 label, the suggestion dropdown list show all the possibilities, and uses the right id when selected. On chip rendering, the displayed label will be the first occurrence as declared on the vocabulary.

#20 - 11/13/2019 04:03 PM - Eric Biagiotti

Lucas Di Pentima wrote:

Updates at commit:796a3ce0 (WB2 repo)

I've been making lots of progress: The properties' ids are saved on hidden fields on the form so they're available on submission at the same time that the visible fields keep showing the labels. The property "chips" render the properties by their labels when possible, keeping the "copy to

clipboard" feature as before, that is, copying the real data to the clipboard (it's trivial to change it).

Also, when some key or value has more than 1 label, the suggestion dropdown list show all the possibilities, and uses the right id when selected. On chip rendering, the displayed label will be the first occurrence as declared on the vocabulary.

Works great for collections! I know you're still working on it, but here are my observations.

- src/common/config.ts - has a FIXME.
- When clicking on the tag to copy, it copies the IDs not the labels (Looks like you mentioned this above).
- Looks like according to this, you might not need the actual <Field> tag for the hidden field. <https://stackoverflow.com/questions/44218104/how-to-set-hidden-fields-in-redux-form-in-react-native/45351256#45351256>. If we still need the tags, the hidden fields will expose the ID via browser developer tools. Is this a problem? My initial thought is that as long as we document it for users who are writing a vocabulary, it should be fine.
- What about other places where you can select properties, like the project details panel and from the search drop down? Are these a part of this ticket?
- Do we need a wiki page for how to write a vocabulary with an example and instructions for how to add it to WB2, etc?
- The the more complicated functions moved to vocabulary.ts (getTags, getTagValues) seem pretty isolated and could benefit from unit tests with some vocab permutations. tree.tests.ts might be handy as an example.

#21 - 11/14/2019 06:51 PM - Lucas Di Pentima

Updates at commit:2f93708c (wb2 repo)

Documentation branch pending.

Eric Biagiotti wrote:

Works great for collections! I know you're still working on it, but here are my observations.

- src/common/config.ts - has a FIXME.

Whoops, fixed.

- When clicking on the tag to copy, it copies the IDs not the labels (Looks like you mentioned this above).

Yes, now it copies whatever the user is reading on the screen.

- Looks like according to this, you might not need the actual <Field> tag for the hidden field. <https://stackoverflow.com/questions/44218104/how-to-set-hidden-fields-in-redux-form-in-react-native/45351256#45351256>. If we still need the tags, the hidden fields will expose the ID via browser developer tools. Is this a problem? My initial thought is that as long as we document it for users who are writing a vocabulary, it should be fine.

You were right, hidden Field weren't necessary, removed!

- What about other places where you can select properties, like the project details panel and from the search drop down? Are these a part of this ticket?

The search part is being addressed on [#15069](#), but I've reused these components on the project detail panels. Also took the opportunity to make a PropertyChipComponent so it's now being used on the 3 places instead of repeating almost the same code.

- Do we need a wiki page for how to write a vocabulary with an example and instructions for how to add it to WB2, etc?

Will do on the arvdos repo.

- The the more complicated functions moved to vocabulary.ts (getTags, getTagValues) seem pretty isolated and could benefit from unit tests with some vocab permutations. tree.tests.ts might be handy as an example.

Done, thanks for the pointer!

#22 - 11/14/2019 11:42 PM - Lucas Di Pentima

Documentation added at [0102502](#) - branch 15067-wb2-vocabulary-doc (arvdos repo)

- Added a new page on the Admin section with a small example of a vocabulary file, explaining its format and how to configure the cluster so that

Workbench2 finds it.

#23 - 11/15/2019 04:30 PM - Eric Biagiotti

Lucas Di Pentima wrote:

Documentation added at [0102502](#) - branch 15067-wb2-vocabulary-doc (arvados repo)

- Added a new page on the Admin section with a small example of a vocabulary file, explaining its format and how to configure the cluster so that Workbench2 finds it.

Code and functionality is great and LGTM!

Some doc comments/suggestions/nits:

- The Workbench2 Configuration text can be simplified. Maybe something like this? "Workbench2 retrieves the vocabulary file URL from the cluster config as shown:"
- "The JSON file describes the available ~~options on~~ keys and values and if the user is allowed to enter ~~other kind of~~ free text not covered or defined by the vocabulary."
- Suggestion for line 33: "Keys and values are indexed by identifiers so that the concept of a term is preserved even if vocabulary labels are changed."
- Line 41: Don't need a colon after e.g.
- Line 43, the sentence starting with "Also strict and values members". I would split this sentence/concepts up.
- "when they're displayed the label shown will be the first one of each group defined ~~en~~in the vocabulary file"
- "because those labels are the first ~~en~~in the vocabulary definition.
- The final sentence, maybe something like this would be a bit more clear? "Internally, Workbench2 uses the IDs to do property based searches, so if the user searches by Animal: Human or Species: Homo sapiens, both will return the same results."

#24 - 11/15/2019 06:19 PM - Lucas Di Pentima

Eric Biagiotti wrote:

Code and functionality is great and LGTM!

Thanks! merged.

Some doc comments/suggestions/nits:

Thanks for all the suggestions, they were very useful as my grammar is far from perfect :)

I've updated the documentation at [a8ea767df](#) and also added a link to it from the Workbench2's install page.

#25 - 11/15/2019 07:25 PM - Eric Biagiotti

Lucas Di Pentima wrote:

Thanks for all the suggestions, they were very useful as my grammar is far from perfect :)

I've updated the documentation at [a8ea767df](#) and also added a link to it from the Workbench2's install page.

Happy to help! Thanks for listening to my nits :) I have more!

- install-workbench2-app.html.textile.liquid - should say "in the Admin section".
- workbench2-vocabulary.html.textile.liquid
 - Line 41, "saving property keys other than the ones defined in the vocabulary"
 - In general, I think we shouldn't use "properties" interchangeably with "tags". My suggestions below try to address this.
 - Line 43, "Inside the tags member, IDs are defined (IDTAGANIMALS, IDTAGCOMMENT, IGTAGIMPORTANCES) and can have any format that the current application requires."
 - Line 45, "The strict flag inside a tag definition operates the same as the strict_tags root member, but at the individual tag level. When strict is true, a tag's value options are limited to those defined by the vocabulary"

Doing the rest of my suggestions in a big chunk. This might be easier, and you can copy paste it and look at the diff.

"The @values@ member is optional and is used to define valid key/label pairs when applicable. In the example above, @IDTAGCOMMENT@ allows open-ended text by only defining the tag's ID and labels and leaving out @values@.

When any key or value has more than one label option, Workbench2's user interface will allow the user to select any of the options. But because only the IDs are saved in the system, when the property is displayed in the user interface, the label shown will be the first of each group defined in the vocabulary file. For example, the user could select the property key @Species@ and @Homo sapiens@ as its value, but the user interface will display it as @Animal: Human@ because those labels are the first in the vocabulary definition.

Internally, Workbench2 uses the IDs to do property based searches, so if the user searches by @Animal: Human@ or @Species: Homo sapiens@, both will return the same results."

#26 - 11/15/2019 08:11 PM - Lucas Di Pentima

Updates on documentation at [83ff3cf35](#)
Thanks!

#27 - 11/15/2019 08:21 PM - Eric Biagiotti

Lucas Di Pentima wrote:

Updates on documentation at [83ff3cf35](#)
Thanks!

Just need to add highlighting for (IDTAGANIMALS, IDTAGCOMMENT, IGTAGIMPORTANCES) on line 43. Otherwise, this LGTM. Thanks!

#28 - 11/15/2019 08:40 PM - Lucas Di Pentima

- *Status changed from In Progress to Resolved*

Applied in changeset [arvadoslaae18e1164b31dc835e41784263d25b79893a6b3](#).

#29 - 01/21/2020 09:21 PM - Peter Amstutz

- *Release set to 22*