

Morpho - Bug #1211

add serialization to steps in DataPackage Wizard

11/17/2003 12:50 PM - Dan Higgins

Status:	Resolved	Start date:	11/17/2003
Priority:	Immediate	Due date:	
Assignee:	Jing Tao	% Done:	0%
Category:	morpho - general	Estimated time:	0.00 hour
Target version:	1.8.0	Spent time:	0.00 hour
Bugzilla-Id:	1211		
Description There are potentially many steps in the DataPackage wizard, especially when one is entering metadata for a table with many columns. There should be some automatic mechanism for serializing the data already entered in case of some failure before a new package is saved. Serializing the DataPackage/TextImportWizard as each screen (or column) is changed seems like a useful approach that should be implemented.			
Related issues: Is duplicate of Morpho - Bug #3548: import data table hangs up on large tables			
		Resolved	10/22/2008

History

#1 - 02/18/2005 04:49 PM - Veronique Connolly

Comments from Laura Downey's Morpho Usability Report: "Add ability to do interim saves while using the wizard."

This would take care of bug#1708 and bug#1944 which are critical bugs.

#2 - 02/18/2005 04:51 PM - Veronique Connolly

For both the New Data Package and Data Table wizards.

#3 - 03/09/2006 11:55 AM - Callie Bowdish

Today a Morpho user who had spent 3 hours entering in information regarding her data set had Morpho crash when trying to import a table for the code values. She had not had any earlier opportunity to save during the table attribute description time so lost hours of work. I think that being able to save during the attribute description for the tables is very important. I think we are losing people's willingness to work with the Morpho EML tool because of loss of work time.

Callie

#4 - 05/24/2007 08:45 AM - Callie Bowdish

Templating support for short-cutting the description of virtually identical structures for describing multiple data sets could also help with the amount of time it takes to enter attribute information.

#5 - 08/24/2009 04:45 PM - Jing Tao

Hi, devs:

Interim saving is one of the most wanted Morpho features. It can save the user's work when Morpho crashes. Or user can temporarily terminate the data package wizard/import text wizard during a time-consuming process. Please also see bug: http://bugzilla.ecoinformatics.org/show_bug.cgi?id=1211

Based on the two requests, I think Morpho should provide two functionalities: Automatically interim saving and manually interim saving. Comparing to regular saving, the interim saving can produce uncompleted (and invalid) EML documents.

1. Automatically interim saving:

When user moves from one wizard page to another one (or from one column to another column), morpho will automatically serialize the ALL existing page data into a file (is this efficient?). So the worst case is user will lose one page meta data (or one column metadata) if morpho crashes.

The file will be stored in the directory named "uncompleted" which is the sibling of data, cache, and temp under .morpho directory. If user successfully completes the process and saves his/her job, the file in "uncompleted" directory will be removed.

Morpho may also have a new menu item (or icon) which can list all crashed eml data packages (and uncompleted packages which will be described below). When user clicks one package, morpho will open package wizard/text import wizard to the place when morpho crashed (this part may need some work to be done). So user can continue their work without losing large amount of data.

2. Temporarily terminate wizard (manually interim saving):

This functionality is similar to the "Automatically interim saving" except this is intended. I would like to add a new button named "Interim Saving" (or something else) just beside Cancel, Back, Next, Finish buttons. When user clicks the button, the existing metadata will be serialized into a file under "uncompleted" directory, which was describe above. And the new package wizard (text import wizard and data package frame) will disappear.

User can use the same way as opening a crashed package to open the temporarily saved the package.

Hi Jing,

Your proposal sounds good. A few comments:

- 1) why do we need a new menu item for opening the saved docs? They should be able to just show up in the 'Open' dialog box listing, right?
- 2) Its better to use the word 'incomplete' for the directory name
- 3) The name of the button for generating an interim save might be better as 'Save for Later' or something like that. Also, the button could probably be arranged on the left side of the wizard dialog so that it is clearly different from the rest of the wizard buttons.
- 4) Suggestion: We should probably have an indication when Morpho crashes. By writing a flag into config, and then clearing it on a successful quit, we could detect when Mopho has shutdown unexpectedly. During the next startup, if Morpho was terminated improperly, scan the 'incomplete' dir, and if there are saved documents there, present an open dialog to the user that says:
"Morpho did not exit cleanly the last time it was run. The following documents were automatically saved in an incomplete state. You may open one of them and continue from the point of the last automatic save, or choose 'Cancel' to return to Morpho."
Then give the normal Open and Cancel buttons.

Matt

Hi, Matt:

Thank you for your comments.

"why do we need a new menu item for opening the saved docs?" Here are reasons:

1. Now we have two savings -regular one and interim one (or crash one). Regular saving generates valid document and interim may generate invalid document. If we use same "Open" dialog box listing to show both of them, user may right click the invalid document and try to upload it to metacat. This will cause a rejection from metacat. So it is better to place them in different places. Moreover, a new menu item can help user easily to find out what documents he/she hasn't finished.

2. The opening behaviors of regular saved documents and interim saved documents (or crashed documents) are different. Opening a regular saved document will bring a morpho data package frame, but opening a interim saved documents (or a crashed document) will bring a new data package wizard or importing text

wizard (and data package frame). So it is better to list the two different types of packages in different places - "Open" button will bring a dialog listing regular data packages, another button (or menu item under file menu) will bring a dialog listing interim saved packages and crashed packages.

Of course we can put interim saved documents (or crashed documents) and regular saved documents into the "Open" dialog box list, and use different icons to distinguish them. This is another approach.

Any comment and suggestion will be highly appreciated.

Thanks,

Jing

#6 - 09/03/2009 03:31 PM - Jing Tao

- Bug 3548 has been marked as a duplicate of this bug. ***

#7 - 12/08/2009 12:22 PM - Jim Regetz

Great progress! Several issues/concerns:

1. In the New DP wizard, clicking "Save for Later" produces a "Save ?" window with a misspelling -- "cointue" should be "continue". Also, I'd vote for dropping the space between "Save" and "?" in the title bar.
2. Choosing to "Save for later" multiple times during the New DP wizard results in a new, duplicate DP (with new docid) each time. E.g., the first time I say "Save for Later", it might save the DP as regetz.1.1. If I then resume working and again do "Save for Later", it creates a new regetz.2.1, etc. IMHO this definitely shouldn't happen (and actually, I would argue that it shouldn't change the rev number either, i.e. how it works if morpho crashes during the wizard).
3. What is the expected behavior if there are two or more "crashed" documents? Note the following:
 1. Start New DP wizard, then force morpho to crash
 2. Reopen morpho, click cancel in the recovery window
 3. Start New DP wizard, then force morpho to crash
 4. Reopen morpho: both incomplete packages appear. However, if you click to open one of these DPs, the dialog box disappears, so you don't have an opportunity to work with the other DP (unless you close and restart Morpho).
4. If a document is incomplete, is there a way to gray out the "synchronize" option from the right-click menu in the Open DP table? Right now, going down that path just leads to a cryptic error from metacat.
5. It strikes me as essential to have a rapid way of seeing **which** documents are incomplete (whether by saving or by crashing). It would probably be sufficient to have some indication in the Open Package table (different icon, or red/italic text, etc).
6. I think an incomplete state should **not** count as a revision. This seems how the crash recovery mechanism works, but not the manual "Save for later" option. If there is no clean/easy way to achieve the saving functionality without using the revision mechanism (in the sense of incrementing the rev number), I understand. But even in that case, I don't think these "revisions" should be preserved once the wizard is completed (or canceled). For example, from the "Open Previous Version" menu, you should never be able to choose an incomplete DP (which basically means opening a document in the middle of a wizard, something that IMHO is nonsensical once you've completed or canceled whatever you were doing with that wizard).
7. The docid identifier number behaves unexpectedly in some cases. Not a big deal, but I'll mention it anyway. Examples:
 - a. Start New DP wizard when the next available docid should be regetz.1.1, and click "Save for Later". After clicking Yes to confirm, morpho saves the DP as regetz.2.1, not regetz.1.1. Why skip one?
 - b. Start New DP wizard when the next docid should be regetz.1.1, then force morpho to crash. Reopening morpho presents regetz.1.1 as an incomplete DP. Makes sense. However, after completing the wizard (but before saving), it now says the docid is "temporary.1.1". Then after saving, it (surprisingly?) becomes regetz.2.1.

#8 - 12/08/2009 01:35 PM - Jing Tao

Bug listed 1 and 2 are fixed. For bug 2, morpho will increase docid revision number rather than assign a new id.

#9 - 12/17/2009 01:13 PM - Jing Tao

For item 7:

- a. now it will increase revision number rather than assign a new package id.
- b. it still assigns new docid number. Do we want to increase revision number?

#10 - 12/17/2009 05:22 PM - Jing Tao

Here are some summary and thought about the bug 1211:

1. Terminology:

Tracing-change document: a document created to trace the change when user clicks the next button during the wizards.

Crashed document: the left tracing-change document when morpho crashes.

Saved-incomplete document: a document created when user click Save for Later button.

2. Both tracing-change documents and saved-incomplete documents will locate in incomplete directory. In tracing-change documents, there is a tag to indicate it hasn't been saved. When user click the Save for Later button, the tag will be removed. When user finishes the wizard and click save button, tracing-change documents/saved-incomplete documents will be removed.

3. For new package wizard, the ids for tracing-change document and saved-incomplete document will be in temp space, i.e. temporary.34.1. For entity wizard and code definition wizard, the id will be its original document id.

4. When morpho starts, it will search incomplete directory to find out the list of documents which have the unsaved flag (crashed documents). Then it will be display to user. User can choose one to open. (Jim, don't worry, open dialog and search result frame will also display the crashed document :))

5. Open dialog and search result frame will display crashed document, saved-incomplete document and complete documents with different icons.

Question 1: the tracing-change document will be displayed when user click the open button while he is working on a new data package wizard. How can we not display the working tracing-change document? For instance, when user is working on a new package wizard and temporary.23.1 will be created on incomplete directory. At same time, he click open button and the dialog will display temporary.23.1 too. Since the temporary.23.1 has the unsaved tag and morpho will think it is crashed document.

Question 2: open dialog or search result frame detects foo.34.2 in incomplete dir as crashed or saved-incomplete document, it also detect foo.34.2 in local/metacat, morpho will display the foo.34.2 as crashed or saved-incomplete document. However, if morpho detects foo.34.3, which has newer version, in local/network as a complete document, which one we should display? This can happen when someone insert a new version of document into metacat from other computer.

6. In open dialog box or search result frame, the synchronize and export menu on right click for crashed/saved-incomplete documents will be disabled.

7. When user is working on entity wizard/code definition wizard, the synchronize and export button on morpho package frame should be disabled.

8. Open previous version command will not display the saved-incomplete documents/crashed documents.

9. New docid or new revision number will be assigned only when user click the save button.

#11 - 12/18/2009 10:20 AM - Jim Regetz

Great! Some responses/ideas:

(In reply to comment [#10](#))

1. Terminology:

Tracing-change document: a document created to trace the change when user clicks the next button during the wizards.

Crashed document: the left tracing-change document when morpho crashes.

Saved-incomplete document: a document created when user click Save for Later button.

Instead of "tracing-change", how about "autosaved incomplete"? Then the two fundamental incomplete doc types are "user-saved" and "autosaved". Crashed documents are just autosaved documents that Morpho discovers on startup. I suggest this because it has relevance to my answer to your first question...

<snip>

Question 1: the tracing-change document will be displayed when user click the open button while he is working on a new data package wizard. How can we not display the working tracing-change document? For instance, when user is working on a new package wizard and temporary.23.1 will be created on incomplete directory. At same time, he click open button and the dialog will display temporary.23.1 too. Since the temporary.23.1 has the unsaved tag and morpho will think it is crashed document.

I think this is fine, and it becomes clearly with my slight redefinition of things above. For this document that has an in-progress wizard, Morpho will regard it as "autosaved incomplete". So yes, the DP will show up as "autosaved incomplete" (along with any crashed docs) in the open dialog, but if you click on it you'll just get the "wizard already running" message.

Whatever the "user-saved incomplete" icon is, the "autosaved incomplete" icon could be the same thing but with a red exclamation point superimposed on it or something? Or maybe both types of incomplete icons are the same as the usual icons, but with an black exclamation point superimposed for user-saved and a red exclamation point for autosaved?

Question 2: open dialg or search result frame detects foo.34.2 in incomplete dir as crashed or saved-imcomplete document, it also detect foo.34.2 in local/metacat, morpho will display the foo.34.2 as crashed or saved-incomplete document. However, if morpho delects foo.34.3, which has newer version, in local/network as a complete document, which one we should display? This can happen when someone insert a new version of document into metacat from other computer.

The simplest behavior would be to always open the incomplete document. If/when the user completes the wizard and attempts to save, I assume they'll get the dialog about a docid conflict, and will then have the option to save as a new revision or as a new doc altogether. Correct? A nicer behavior would be first to notify the user that a newer doc version exists, and provide an option to discard incomplete changes or continue editing; if they continue, same deal as above in terms of whether they eventually choose to save it as a new revision or a new doc altogether. Same outcome, but less of a surprise at the end.

Would my assumptions about docid conflict detection still hold if, say, the incomplete version was foo.34.2, the newer version downloaded from metacat was foo.34.4 (or higher), and no local foo.34.3 existed? If not, that case should be handled too.

<snip>

8. Open previous version command will not display the saved-incomplete documents/crashed documents.

Agreed -- and this should be the case automatically as long as incomplete documents are never saved as distinct revs in the data directory, right?

#12 - 01/07/2010 12:19 PM - Jing Tao

Question 2: open dialg or search result frame detects foo.34.2 in incomplete dir as crashed or saved-imcomplete document, it also detect foo.34.2 in local/metacat, morpho will display the foo.34.2 as crashed or saved-incomplete document. However, if morpho delects foo.34.3, which has newer version, in local/network as a complete document, which one we should display? This can happen when someone insert a new version of document into metacat from other computer.

The simplest behavior would be to always open the incomplete document. If/when the user completes the wizard and attempts to save, I assume they'll get the dialog about a docid conflict, and will then have the option to save as a new revision or as a new doc altogether. Correct? A nicer behavior would be first to notify the user that a newer doc version exists, and provide an option to discard incomplete changes or continue editing; if they continue, same deal as above in terms of whether they eventually choose to save it as a new revision or a new doc altogether. Same outcome, but less of a surprise at the end.

Would my assumptions about docid conflict detection still hold if, say, the incomplete version was foo.34.2, the newer version downloaded from metacat was foo.34.4 (or higher), and no local foo.34.3 existed? If not, that case should be handled too.

Hi, Jim:
Thank you for the response. They are very helpful.
But I still particularly have concern about above question. I am not sure if I got your point or not.

My question is about the how to display search result on the search result frame. Currently, if local has package foo.2.1, foo.2.2 and metacat has foo.2.3, the search result will only show foo.2.3 and indicate is on metacat. The local foo.2.1 and foo.2.2 will be hidden.

Now we will introduce auto-saved and user-saved documents, things get more complicated.

case 1: local has complete package foo.1.1, metacat has foo.1.1 (metacat always has complete package), and incomplete directory has foo.1.1. In this case, I think morpho will only display foo.1.1 as incomplete package (either auto-saved or user-saved).

case 2: local has complete package foo.1.2, metacat has complete package foo.1.2 and incomplete directory has foo.1.1. What should we do? The incomplete foo.1.1 will be hidden and display local and metacat package foo.1.2? Or display both incomplete foo.1.1 and complete foo.1.2(local and metacat)?

So my question is if the incomplete package will be hidden when morpho detects higher revision complete package in local or metacat during the search?

#13 - 01/14/2010 03:41 PM - Jing Tao

Hi, Ben:

Thanks for the input. Yeah, I agree to display both the incomplete foo.1.1 and complete foo.1.2.

Here are how the case 2 happen:

1. User works on computer A to create foo.1.1 by new package wizard and saves it to both local and metacat.
2. User uses entity wizard to import a file raindata in foo.1.1 and don't finish it. User saves it as foo.1.1 in incomplete dir.
3. User works on computer B to find a foo.1.1 in metacat. The foo.1.1 is downloaded into computer B. User import the file raindata to foo.1.1 and saves it to both local and metadata as foo.1.2.
4. User works on computer A again. He searches both local and metacat, he will find both incomplete foo.1.1 and complete foo.1.2 on metacat.
- (5. If foo.1.1 is displayed, user opens it. After finishing wizard, user will save it to metacat as foo.1.2. However morpho detects a foo.1.2 already on metacat (docid conflict), will ask user to either increase revision or assign a new id).

Thanks,

Jing

When will Case 2 happen?

Someone else saved a valid foo.1.2 to metacat while I was working on my incomplete foo.1.1 and now I have a doc conflict?

I guess it should show the incomplete foo.1.1 just in case it's an honest id conflict. Even if we are working on "the same" datapackage, it's still probably a good idea to show both the incomplete foo.1.1 and the complete metacat version foo.1.2 in the search results. And from that other enhancement there will be nifty new icons to show how they differ!

-ben

#14 - 01/20/2010 12:53 PM - Jing Tao

Copy an email which described the morpho behavior:

1. When user clicks the new data package wizard button, a incomplete file, i.e., temporary.2.1 will be generate in incomplete folder. If users click the cancel button of the wizard at some point, wizard will disappear and temporary.2.1 will be removed. If user finishes the wizard (clicks the finish button), the morpho data package frame will show up and temporary.2.1 will still stay in the incomplete folder. If user click save button, the temporary.2.1 in incomplete folder will be removed. If user closes the frame without saving, the temporary.2.1 will be removed as well.
2. If user opens a user-saved incomplete file, i.e. temporary.3.1, which comes from new data package wizard, morpho will show a new data package wizard. The temporary.3.1 in incomplete folder will be transformed from a user-saved file to an auto-saved file(adding a tag in additional metadata part). Morpho will have the same behavior as the above one.
3. When user clicks the entity wizard from a morpho frame, i.e. package foo.4.1, incomplete file foo.4.1 will be generated. If users click the cancel button of the wizard, wizard will disappear and foo.4.1 in incomplete dir will still exist, but current data package will be dumped to foo.4.1 in incomplete folder. The reason why morpho doesn't delete foo.4.1 in incomplete dir is in case foo.4.1 has been added a saved entity. For example, user added a data table to foo.4.1 successfully, but click the cancel button when he/she is adding the second table. If the foo.4.1 was removed, the information of the first table will be missed. If morpho crashes now, user couldn't recover the first table. If user finish the wizard and save the package, the foo.4.1 and associated data files in incomplete dir will be removed. If user finished the wizard and close the morpho wizard without saving, the foo.4.1 and associated data files in incomplete dir will be removed as well.
4. If user opens a user-saved incomplete package, i.e. foo.5.1, which comes from entity wizard, morpho will show a data package frame and an entity wizard. The foo.5.1 in incomplete folder will be transformed from a user-saved file to an auto-saved file. Morpho will have the same behavior as the third case.

#15 - 01/26/2010 11:47 AM - Jim Regetz

Morpho version used: morpho-1.8.0-RC1

This issue is related to behaviors [#3](#) and [#4](#) above.

Given the following steps:

1. Open existing DP
2. Begin New Data Table wizard
- 2b. Save for Later (or crash)
- 2c. Re-open DP, resuming where wizard left off
3. Cancel wizard (and confirm yes)
4. Close DP

If you only do steps 1, 2, 3, and 4, you are not prompted to save when closing; this seems right, as the DP should be unmodified. However, if you also do steps 2b and 2c, then you **are** prompted to save when closing (step 4), even though the DP should be unmodified. If you elect not to save, the outcome is the same as the normal case (steps 1-4 only). If you elect to save, the rev number is incremented, unnecessarily producing a new revision that is AFAICT identical to the previous revision.

Of course, if you had made any other changes to the DP (e.g., you previously completed an import of another table, but hadn't yet saved it), then you **should** be prompted to save; this currently works fine.

#16 - 01/26/2010 02:36 PM - Jim Regetz

Morpho version used: morpho-1.8.0-RC1

Minor suggestions about the "Open Crashed Documents" dialog:

1. It looks like a space is needed between the second and third sentences.
2. Maybe change "Open Crashed Documents" to "Open Recovered Documents" ?
3. Maybe replace the last sentence with this: "You may open one of them now to continue from the point of the last automatic save. If you cancel, you will still be able to access these documents later by using the Open dialog."

#17 - 01/26/2010 02:44 PM - Matt Jones

Jim's are good suggestions in comment [#16](#) and I suggest we heed them for usability's sake.

#18 - 01/26/2010 02:58 PM - Jing Tao

I will change it.

#19 - 01/26/2010 04:21 PM - Jim Regetz

Morpho version used: morpho-1.8.0-RC1

As a naive user, I would expect that if I open an incomplete document, then click cancel, I'd be back to having the same incomplete document. In reality, if you resume a wizard after saving for later (or recovering from crash), clicking cancel does more than that: it reverts all the way back to the pre-wizard state. I fully understand why, but it's slightly surprising nevertheless, and could cause users to lose their work.

Suggestions:

1. Strengthen the confirmation dialog text, e.g.: "If you cancel, **all** work done in this wizard will be lost. Are you sure you want to cancel?"
2. As a convenience, but also to make it clearer that there are three possible actions, perhaps add a "Save for Later" button directly to this confirmation dialog? (Would this cause problems if the Cancel button had been pressed on a wizard page that had some required fields not yet filled out?)

An alternative to [#2](#) would be to communicate this option in words, but maybe this makes the text unwieldy: "If you cancel, **all** work done in this wizard will be lost. If you want to resume the wizard later, choose "No", then click on the "Save for Later" button. Are you sure you want to cancel?"

(I'm not sure how many places this confirmation dialog is used, so we'd need to confirm that the modified dialog makes sense in all cases.)

#20 - 01/26/2010 04:44 PM - Jing Tao

Thanks, Jim. I already changed the wording according your suggestion on comment 16. I will take a look at the comment 19.

#21 - 02/08/2010 03:08 PM - Jing Tao

On comment 19, I used the way Jim suggested:

"An alternative to [#2](#) would be to communicate this option in words, but maybe this makes the text unwieldy: "If you cancel, **all** work done in this wizard will be lost. If you want to resume the wizard later, choose "No", then click on the "Save for Later" button. Are you sure you want to cancel?"

This feature is done. If we find any bugs, we should add a new bug report.

#22 - 03/27/2013 02:16 PM - Redmine Admin

Original Bugzilla ID was 1211