[LDO-327] HPTN071 - Issues with the "Lab Submitted Assay Data"

Created: 14/Jun/18 3:05 PM - Updated: 01/Nov/18 10:00 AM

Status:	Open
Project:	LDO-PopART 071
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Bug	**Priority:**	Major
Reporter:	Lei Weng	**Assignee:**	Kalkidan Lebeta
Resolution:	Unresolved	**Votes:**	0
Remaining Estimate:	0 minutes		
Time Spent:	2 days, 21 hours		
Original Estimate:	Not Specified		

Development:

{summaryBean=com.atlassian.jira.plugin.devstatus.rest.SummaryBean@43ea6190[summary={pullrequest
build=com.atlassian.jira.plugin.devstatus.rest.SummaryItemBean@7de99d6d[overall=com.atlassian.jira.plu
review=com.atlassian.jira.plugin.devstatus.rest.SummaryItemBean@78c50129[overall=com.atlassian.jira.p
deployment-environment=com.atlassian.jira.plugin.devstatus.rest.SummaryItemBean@6e3b8199[overall=com.atlassia
repository=com.atlassian.jira.plugin.devstatus.rest.SummaryItemBean@2f84a9e1[overall=com.atlassian.jir
branch=com.atlassian.jira.plugin.devstatus.rest.SummaryItemBean@17ace23[overall=com.atlassian.jira.pl
devSummaryJson={"cachedValue":{"errors":[],"configErrors":[],"summary":{"pullrequest":
{"overall":
{"count":0,"lastUpdated":null,"stateCount":0,"state":"OPEN","open":true},"byInstanceType":
{}},"build":{"overall":
{"count":0,"lastUpdated":null,"failedBuildCount":0,"successfulBuildCount":0,"unknownBuildCount":0},"byIns
{}},"review":{"overall":
{"count":0,"lastUpdated":null,"stateCount":0,"state":null,"dueDate":null,"overDue":false,"completed":false},"
{}},"deployment-environment":{"overall":{"count":0,"lastUpdated":null,"topEnvironments":
[],"showProjects":false,"successfulCount":0},"byInstanceType":{}},"repository":
{"overall":{"count":0,"lastUpdated":null},"byInstanceType":{}},"branch":{"overall":
{"count":0,"lastUpdated":null},"byInstanceType":{}}}},"isStale":false}}

Filename(s):	Please refer to the Source tab of the ticket for the list of commits.
Folder Location:	Please refer to the Source tab of the ticket for the list of commits.
Impact FR?:	Yes
Peer Review:	• Assess the Programmer’s code by reviewing the programming logic and confirming that the production log (e.g., SAS log, R .rout file) does not produce error, unexpected warning, or unexpected note messages. • For results, review the output for obvious nonsensical or unexpected values.
Rank:	1|i0134i:
Rank (Obsolete):	9223372036854775807
Repository Revision Number(s):	Please refer to the Source tab of the ticket for the list of commits.
Verification Procedure:	[Independent Verification and Targeted Checking, Peer Review]

HCA03_2018-06-15_14-07-46.xlsx (7.35 MB)
missing_files_2018_06_15.csv (4 kB)

Based on the information from LC, "Lab Submitted Assay Data" tables are supposed to list all the uploaded assay results from site labs and LC. And the tables were separated by assay file types. These tables are on Atlas LC page.

View HCA02 Results
View HCA03 Results
View HCA04 Results
View HSE01 Results
View HSE02 Results
View HSE03 Results
View HCB01 Results
View HCB02 Results
View HCB03 Results
View HSG01 Results
View VLR06 Results
Denni used these tables pulling site results and she didn't have any issues previously. But she couldn't find several PC36 hca results on 14-Jun-2018. I compared assay dataset among HCA02, HCA03 and HCA04 tables, there are some PC36 site results in HCA03 and HCA04, but the results are not complete (A lot of files submitted early this year were not listed in the table).
[~mdemilia] Could you take a look at this?
FYI [~hjiang23][~mbemer]

Lei Weng added a comment - 14/Jun/18 3:09 PM

Michael DeMilia
If it helps, I could give you some example files that lab submitted but not listed in these tables.
I only checked HCA and not sure if the problem applied to other assay types.

Lei Weng added a comment - 15/Jun/18 3:25 PM

I used HCA03 as an example.

a. Exported "HCA03_2018-06-15_14-07-46.xlsx" from the portal - "View HCA03 Results ".
(https://atlas.scharp.org/cpas/query/HPTN/Specimen%20Management/071/executeQuery.view?query.queryName=HCA03&schemaName=assay.HIV071.HIV01&query.containerFilterName=CurrentAndSubfolders)
b. Merged with current hca dataset (subset hca03 files only) – /trials/hptn/p071/s001/data/l071hca.sas7bdat
c. Generated the output "missing_files_2018_06_15.csv" that files listed in "l071hca.sas7bdat" but not in the portal table.

It looks like files submitted between AUG-2017 and APR-2018 were not updated on the portal table properly.

Michael DeMilia added a comment - 21/Jun/18 3:10 PM

Example 1:
"HCA03_p071_535_201804111606",2148,"12APR2018"

Should appear right here...
https://atlas.scharp.org/cpas/query/HPTN/Specimen%20Management/071/executeQuery.view?query.queryName=HCA03&schemaName=assay.HIV071.HIV01&query.containerFilterName=CurrentAndSubfolders&query.sort=-filename&query.offset=1853

Between:
"HCA03_p071_535_201803281541",2200,"09MAY2018"
"HCA03_p071_535_201804201449",2168,"20APR2018"

There is nothing distinctive at all between the bad file and the good .csv, .tsv, and numbered text files produced by the upload and perl script in /dmz/services/atlas-a/data/HPTN/071/Specimen\ Management/assaydata/ldo_files/.

They were loaded out of sequence, though.

```
-rwxrwxr-x 1 labkey labkey 19788 Apr 11 07:08 HCA03_p071_535_201804111606.csv
-rwxrwxr-x 1 labkey labkey 57257 Apr 11 07:10 HCA03_p071_535_201801041605-1.csv
-rwxrwxr-x 1 labkey labkey 58845 Apr 20 05:47 HCA03_p071_535_201804201449.csv
-rwxrwxr-x 1 labkey labkey 19639 Apr 23 06:07 HCA03_p071_535_201804231507.csv
-rwxrwxr-x 1 labkey labkey 19479 Apr 25 06:06 HCA03_p071_535_201804251508.csv
-rwxrwxr-x 1 labkey labkey 19530 Apr 30 00:44 HCA03_p071_535_201804271310.csv
-rwxrwxr-x 1 labkey labkey 19543 Apr 30 01:04 HCA03_p071_535_201804271310-1.csv
```

```
-rwxrwxr-x 1 labkey labkey 29269 May 2 06:09 HCA03_p071_535_201805021457.csv
-rwxrwxr-x 1 labkey labkey 29236 May 2 06:16 HCA03_p071_535_201805021457-1.csv
-rwxrwxr-x 1 labkey labkey 39111 May 9 06:42 HCA03_p071_535_201803281541.csv
```

Example 2:
"HCA03_p071_300_201708041734",1637,"05AUG2017"

Should appear right here...
https://atlas.scharp.org/cpas/query/HPTN/Specimen%20Management/071/executeQuery.view?query.queryName=HCA03&schemaName=assay.HIV071.HIV01&query.containerFilterName=CurrentAndSubfolders&query.sort=-filename&query.offset=2109

Between:
"HCA03_p071_300_201708031540",1631,"04AUG2017"
"HCA03_p071_300_201805080956",2198,"09MAY2018"

Huge gap here and most of the files on the list fall in between:

```
-rwxrwxr-x 1 labkey labkey 6817 Aug 3 2017 HCA03_p071_300_201708031540.csv
-rwxrwxr-x 1 labkey labkey 13199 Aug 4 2017 HCA03_p071_300_201708041734.csv
-rwxrwxr-x 1 labkey labkey 8585 Aug 18 2017 HCA03_p071_300_201708181729.csv
-rwxrwxr-x 1 labkey labkey 3480 Aug 28 2017 HCA03_p071_300_201708280851.csv
-rwxrwxr-x 1 labkey labkey 20548 Aug 29 2017 HCA03_p071_300_201708291506.csv
-rwxrwxr-x 1 labkey labkey 20548 Aug 29 2017 HCA03_p071_300_201708291506-1.csv
-rwxrwxr-x 1 labkey labkey 20228 Aug 30 2017 HCA03_p071_300_201708291506-2.csv
-rwxrwxr-x 1 labkey labkey 16483 Sep 12 2017 HCA03_p071_300_201709121520.csv
-rwxrwxr-x 1 labkey labkey 1825 Sep 15 2017 HCA03_p071_300_201709151614.csv
-rwxrwxr-x 1 labkey labkey 22082 Sep 18 2017 HCA03_p071_300_201709181124.csv
-rwxrwxr-x 1 labkey labkey 1217 Sep 20 2017 HCA03_p071_300_201709201210.csv
-rwxrwxr-x 1 labkey labkey 8089 Oct 30 2017 HCA03_p071_300_201710301409.csv
-rwxrwxr-x 1 labkey labkey 7929 Oct 30 2017 HCA03_p071_300_201710301409-1.csv
-rwxrwxr-x 1 labkey labkey 7769 Oct 30 2017 HCA03_p071_300_201710301409-2.csv
-rwxrwxr-x 1 labkey labkey 547 Oct 30 2017 HCA03_p071_300_201710301556.csv
-rwxrwxr-x 1 labkey labkey 389 Nov 1 2017 HCA03_p071_300_201711011514.csv
-rwxrwxr-x 1 labkey labkey 13029 Dec 8 2017 HCA03_p071_300_201712081000.csv
-rwxrwxr-x 1 labkey labkey 549 Dec 8 2017 HCA03_p071_300_201712081001.csv
-rwxrwxr-x 1 labkey labkey 10535 Dec 20 2017 HCA03_p071_300_201712200814.csv
-rwxrwxr-x 1 labkey labkey 9893 Dec 20 2017 HCA03_p071_300_201712201516.csv
-rwxrwxr-x 1 labkey labkey 16253 Dec 22 04:33 HCA03_p071_300_201712220730.csv
-rwxrwxr-x 1 labkey labkey 15932 Dec 22 07:18 HCA03_p071_300_201712220730-1.csv
-rwxrwxr-x 1 labkey labkey 8267 Dec 22 11:05 HCA03_p071_300_201712221404.csv
-rwxrwxr-x 1 labkey labkey 7302 Jan 1 12:15 HCA03_p071_300_201801011512.csv
-rwxrwxr-x 1 labkey labkey 1193 Jan 1 12:16 HCA03_p071_300_201801011507.csv
-rwxrwxr-x 1 labkey labkey 23748 Jan 10 10:35 HCA03_p071_300_201801101333.csv
-rwxrwxr-x 1 labkey labkey 5267 Jan 12 12:35 HCA03_p071_300_201801121530.csv
-rwxrwxr-x 1 labkey labkey 28728 Jan 12 12:36 HCA03_p071_300_201801121532.csv
-rwxrwxr-x 1 labkey labkey 378 Jan 26 04:37 HCA03_p071_300_201801260735.csv
-rwxrwxr-x 1 labkey labkey 1010 Jan 30 08:54 HCA03_p071_300_201801301150.csv
-rwxrwxr-x 1 labkey labkey 16181 Jan 31 03:24 HCA03_p071_300_201801310622.csv
-rwxrwxr-x 1 labkey labkey 184443 Jan 31 08:10 HCA03_p071_300_201801311108.csv
-rwxrwxr-x 1 labkey labkey 184443 Jan 31 08:11 HCA03_p071_300_201801311108-1.csv
-rwxrwxr-x 1 labkey labkey 20957 Feb 1 10:00 HCA03_p071_300_201802011300.csv
-rwxrwxr-x 1 labkey labkey 184288 Feb 7 08:14 HCA03_p071_300_201801311108-2.csv
-rwxrwxr-x 1 labkey labkey 5959 Feb 15 05:39 HCA03_p071_300_201802150839.csv
-rwxrwxr-x 1 labkey labkey 4574 Feb 21 05:21 HCA03_p071_300_201802210820.csv
-rwxrwxr-x 1 labkey labkey 13956 Feb 21 10:36 HCA03_p071_300_201802211334.csv
-rwxrwxr-x 1 labkey labkey 2942 Feb 27 07:59 HCA03_p071_300_201802271057.csv
-rwxrwxr-x 1 labkey labkey 21496 Mar 15 04:49 HCA03_p071_300_201803150747.csv
-rwxrwxr-x 1 labkey labkey 14375 Mar 19 12:03 HCA03_p071_300_201803191502.csv
-rwxrwxr-x 1 labkey labkey 14053 Mar 20 08:18 HCA03_p071_300_201803191502-1.csv
-rwxrwxr-x 1 labkey labkey 10972 Apr 3 06:55 HCA03_p071_300_201804021601.csv
```

```
-rwxrwxr-x 1 labkey labkey 36228 Apr 6 08:54 HCA03_p071_300_201804061154.csv
-rwxrwxr-x 1 labkey labkey 2724 Apr 10 08:33 HCA03_p071_300_201804101133.csv
-rwxrwxr-x 1 labkey labkey 310947 Apr 10 08:37 HCA03_p071_300_201804101133-1.csv
-rwxrwxr-x 1 labkey labkey 310947 Apr 10 08:39 HCA03_p071_300_201804101133-2.csv
-rwxrwxr-x 1 labkey labkey 2858 Apr 10 08:42 HCA03_p071_300_201804101133-3.csv
-rwxrwxr-x 1 labkey labkey 1847 Apr 11 09:49 HCA03_p071_300_201804111247.csv
-rwxrwxr-x 1 labkey labkey 7625 Apr 11 09:49 HCA03_p071_300_201804111248.csv
-rwxrwxr-x 1 labkey labkey 13909 Apr 12 12:12 HCA03_p071_300_201804121421.csv
-rwxrwxr-x 1 labkey labkey 28715 Apr 18 15:07 HCA03_p071_300_201804181805.csv
-rwxrwxr-x 1 labkey labkey 5694 Apr 18 15:13 HCA03_p071_300_201804181813.csv
-rwxrwxr-x 1 labkey labkey 5372 Apr 19 07:05 HCA03_p071_300_201804181813-1.csv
-rwxrwxr-x 1 labkey labkey 28549 Apr 19 07:08 HCA03_p071_300_201804181805-1.csv
-rwxrwxr-x 1 labkey labkey 10359 May 8 06:57 HCA03_p071_300_201805080956.csv
```

Michael DeMilia added a comment - 22/Jun/18 3:20 PM

Weekly summary: Curioser and curiouser. Spent the week down the rabbit hole and still don't have a clue what's going on. I put in a ticket with support saying could use Tobin's (or whomever) help figuring out what's happening with a query in Atlas and cross referencing it to the Postgres table where the data is being stored. Trying to figure out why past uploaded CSV files are missing in the Atlas data. First step is to see if the query is loading all the data in the table. If the data is all there, then the query or the data needs to be fixed. If the data isn't all there, then need to figure out why it didn't load upon upload or got deleted.

Michael DeMilia added a comment - 26/Jun/18 10:39 AM

I think the problem is between April 2017 and April 19 2018 and an internal Labkey change messed things up. They had me do a temporary fix for HCB03 and HSG01 in March 2018, but the other modules were not fixed. It turns out I guessed correctly the beginning of last week that the problem was the same, however, the solution was no longer relevant. The problem was resolved with LabKey release 18.1 r56529 so new data coming in after that was fine. I really won't know for sure until I run some queries on the internal atlas tables in Postgres after we figure out which ones they are exactly. While the problem was fixed, the old uploaded data is still wrong. It seems that the old data needs to be fixed internally in the atlas Postgres tables.

Michael DeMilia added a comment - 29/Jun/18 6:31 PM

I'm still waiting on getting schema group permission changes in atlas-test postgres so that I have the permissions to view the data in the internal tables. Lloyd couldn't just make the change without discussion with others. I asked Tobin if he has an atlas based solution for running the queries I need without messing anything up, but still haven't gotten a response. So I started digging into the LabKey documentation and the SAS interface to it to see if I could get the data another way. The macros and Java that they call are in /trials/sas/atlasmacros/.

I started experimenting with it and it's far from an ideal solution since LabKey is really picky and it's kind of like using SAS to pull results from a LabKey generated webpage. If you try to pull back too much data at once then the Java class instantiation fails.

For example, the full query Atlas is using for the report is really complicated and can be found:
Also, to get to the query details for the HCA03 query:
1. View the query - https://atlas-test.scharp.org/cpas/query/HPTN/Specimen%20Management/071/executeQuery.view?query.queryName=HCA03&schemaName=assay.HIV071.HIV01&query.containerFilterName=CurrentAndSubfolders&query.sort=-filename&query.offset=2109
2. Click the "assay.HIV071.HIV01 Schema" link at the top of the page
3. Click "Edit Source"

Now I theoretically want to find the records that are being left out because id_file is null as a result of the join:
INNER JOIN hptn071_lab.v_assay_file_upload A on Data.Run.ID_File=A.id_file

This works to pull back a few records in the table:

```
%include '/trials/sas/atlasmacros/labkeyexecutesql.sas';
options mprint symbolgen;
%let schema=assay.HIV071.HIV01;
%let path=/HPTN/Specimen Management/071/;
%let offset=;
```

```
%let nobs=1000;
%let sql=select Data.Run.ID_File, Data.* from Data;
%labkeyExecuteSql(dsn=junk,sql="&sql",baseUrl="https://atlas-test.scharp.org/
cpas",folderPath="&path",schemaName="&schema",rowOffset=&offset,maxRows=&nobs,showHidden=1,containerFilter="CurrentAndSubf
```

This fails because of too much data:

```
%let nobs=;
```

These are all bad LabKey queries:

```
%let sql=select Data.Run.ID_File, Data.* from Data where Data.Run.ID_File=.;
%let sql=select Data.Run.ID_File, Data.* from Data where Data.Run.ID_File is null;
%let sql=select Data.Run.ID_File, Data.* from Data where Data.Run.ID_File=";
```

This brings back 0 records:

```
%let nobs=;
%let sql=select Data.Run.ID_File, Data.* from Data where Data.Run.ID_File<0;
```

So that makes we wonder if the data just wasn't loaded into the table and it's not a matter of a null id_file. I just can't tell for sure yet, because I can't see the full table and LabKey is so picky about what queries it will execute. It would be a whole lot easier to just run postgres queries on the table if I had the permissions.

Still investigating...

Michael DeMilia added a comment - 29/Jun/18 6:35 PM

Looked into how to cross reference LabKey internal data to the Postgres tables without a whole lot of luck. Dug into the LabKey documentation trying to get a better understanding of how the internal data gets stored. Dug into what the transform script that gets called by our Atlas upload script for each module is doing.

Michael DeMilia added a comment - 02/Jul/18 1:00 PM

Tobin doesn't have an answer for me on a better way to run the queries using Atlas than the way I'm doing it using SAS. Looks like I really need to get access to the postgres tables to pursue this further.

This gives me 0 records and the other ways of trying the same things crash the Java:

```
%let sql=select Data.Run.ID_File, Data.* from Data where Data.Run.ID_File<0;
```

Yet this clearly shows me that there are records with missing id_file:

```
%let sql=select Data.Run.ID_File, Data.* from Data where dataid = 331921;
```

Michael DeMilia added a comment - 12/Jul/18 11:19 AM

Just so you're clear; the problem with the past records with the missing CSV file upload association (null id_file) was caused by an internal LabKey bug from an upgrade that Shane installed around either April or August (not sure which yet) 2017. It was fixed by an LabKey upgrade that Shane installed around April 2018. There was never an issue with any of our code itself. So we just need to fix the records during that period.

Michael DeMilia added a comment - 12/Jul/18 11:20 AM

If Lloyd/Shane have concerns about adding cpas-read permissions to the assayresult schema because it is an internal LabKey schema, then an alternate solution is to add me as a user to the cpas schema in atlas-test (not production) for just a few days to run my queries then remove me after. I didn't want to go that route since it will give me write permissions and not just read. I definitely don't want them doing that on production, just atlas-test. Lloyd or someone in xApps would need to make the record fixes in production (good idea to try it first in atlas-test); not me.

Michael DeMilia added a comment - 19/Jul/18 2:49 PM

Shane confirmed that we can't be messing with the assayresult schema or the cpas group users even on atlas-test since it's internal to Atlas functionality. I need to take a look at the LabKey documentation about the APIs and how to change data using the options available in the interface then discuss with Tobin.

Michael DeMilia added a comment - 23/Jul/18 2:31 PM

The problem is between April (or possibly August) 2017 and April 2018 and an internal LabKey change messed things up. They had me do a temporary fix for HCB03 and HSG01 in March 2018, but the other modules were not fixed. The LabKey upgrade that we applied to production in April 2018 fixed the problem. The records are in atlas, but they are missing id_file so they don't appear in the reports since they rely on an inner join with hptn071_lab.v_assay_file_upload on id_file. So what happened was that when the labs uploaded a CSV in Atlas, our module perl script called an internal Labkey transform script which had a bug during that period, and the id_file value got set to null such that the association with the CSV file information got lost.

I had originally hoped to run queries and figure out what needed to be done exactly directly in postgres, but the problem is I don't have read permissions for the assayresult schema. Lloyd and Shane have nixed the idea of adding cpas-read group permissions to that schema or temporarily add me as user to the cpas group. I've been using the LabKey SAS macros to try to figure things out. LabKey is really picky and it's kind of like using SAS to pull results from a LabKey generated webpage. If you try to pull back too much data at once then the Java class instantiation fails. It's been really hard to do things this way. You would think that selecting records where id_file is null would be easy, but all my queries using the SAS API trying do so using different logic (null, missing, '', <0) just fail. So when I get the chance, I need to dig further into the LabKey API documentation and discuss with Tobin if there's an easier way to do this.

Decision #1: Is it okay to fix the data? If so, then this is the easiest solution, we don't have to mess with the existing reports, we don't risk breaking anything, and the labs will probably be the happiest. If there's an override to the decision that we can't change assayresult schema permissions or users in postgres, then that's the easiest way to do things; I would suggest that atlas-test be taken down while I'm doing my queries and experimenting with fixing the data to avoid any of Shane's concerns; and once it's all figured out, atlas production be taken down while Lloyd repeats the process there. If not, then Tobin and I have to come up with a solution to do things in pieces using the SAS API, another API, or via some new report we create in atlas where the data can be edited to fix the id_file.

Decision #2: If it's not okay to fix the data, then do we need to fix the existing reports for all the 071 assays in atlas? The easiest and safest solution would be to not mess with the existing reports. 071 is a house of cards; every time we try to pull a card out and fix it, we are risking a collapse. There have been so many times where we made a change to something and later discovered we messed something else up that was seemingly unrelated. If we must do it this way, then we may need to use SAS to generate a fixed set of data then load it into a new postgres table that the atlas reports use.

Decision #3: If it's not okay to fix the data and we don't need to fix the existing reports, can we provide the information to the labs via separate reports for the non-viewable records? SAS would probably be the best way to generate the reports as CSVs and we can put the links in atlas. An alternative method, would be create web reports directly in Atlas using a modified version of the SQL used for the existing reports and possibly requiring a new postgres table generated by SAS. The alternative method would probably be more work, though, there's a chance we will have no choice if we can't get all the problem records using the SAS API.

Michael DeMilia added a comment - 06/Aug/18 4:53 PM

Been experimenting with the LabKey SAS API.
https://www.labkey.org/Documentation/wiki-page.view?name=sasMacros

It's more complicated to insert or update data then to just select data. To select I can call %labkeyExecuteSql like is done in 071 SAS processing when data is pulled from Atlas. For example:
%let sql=select Data.Run.ID_File, Data.* from Data where dataid = 331921

Trying to use %labkeySelectRows and %labkeyMakeFilter is more complicated. Particularly since it requires use of an existing query (instead of an SQL statement) and in this case where we can't use the existing query that's producing the report since the it doesn't pull back the records with the null ID_File due to the join:
https://atlas-test.scharp.org/cpas/query/HPTN/Specimen%20Management/071/executeQuery.view?query.queryName=HCA03&schemaName=assay.HIV071.HIV01&query.containerFilterName=CurrentAndSubfolders

So I'm pretty sure I'll need to add a user defined query in the folder structure just to run the SAS API, even if it's not used for any reports.

It's even more complicated that we need Data.Run.ID_File which is the null value that needs to be filled in. I'm not sure how to reference this in the query creation. I sure hope it's possible to make use of Data.Run.ID_File this way.

The goal is that once I get %labkeySelectRows to select the records I want then use %labkeyUpdateRows to update then after Data.Run.ID_File is filled in the interim dataset using the file lookup information found by SAS using %labkeyExecuteSql.

Initially I'm going to try it using %labkeyInsertRows with a duplicate of a record to make sure the insert produces an identical record to the original except with Data.Run.ID_File filled in.

Just getting a single record updated is complicated. I then would need to loop through it for all the records, which is problematic since the Java object creation fails when I try to pull back too much data.

Going to put in a ticket with LabKey for help since it's getting even more complicated than expected.

Lei Weng added a comment - 01/Nov/18 10:00 AM

Kalkidan LebetaDebbie GollnickHaoping Jiang

Alexis replayed LC's request on the status of these table. LC need to be able to see architect uploads , and sounds like this is the next priority project.

Michael DeMilia logged work : 1h - 15/Jun/18 10:52 AM

Michael DeMilia logged work : 8h - 22/Jun/18 3:13 PM

Michael DeMilia logged work : 8h - 22/Jun/18 3:13 PM

Michael DeMilia logged work : 8h - 22/Jun/18 3:14 PM

Michael DeMilia logged work : 7h - 22/Jun/18 3:14 PM

Michael DeMilia logged work : 7h - 22/Jun/18 3:14 PM

Michael DeMilia logged work : 5h - 28/Jun/18 11:26 AM

Michael DeMilia logged work : 8h - 28/Jun/18 11:26 AM

Michael DeMilia logged work : 5h - 28/Jun/18 11:26 AM

Michael DeMilia logged work : 4h - 29/Jun/18 6:43 PM

Michael DeMilia logged work : 8h - 29/Jun/18 6:43 PM

Lei Weng made changes - 14/Jun/18 3:09 PM

assignee	Lei Weng	Michael DeMilia

Michael DeMilia made changes - 15/Jun/18 10:52 AM

timeestimate		0
timespent		3600
WorklogId	28843	

Lei Weng made changes - 15/Jun/18 3:25 PM

Attachment		HCA03_2018-06-15_14-07-46.xlsx
Attachment		missing_files_2018_06_15.csv

Michael DeMilia made changes - 22/Jun/18 3:13 PM

timespent	3600	32400
WorklogId	28879	

Michael DeMilia made changes - 22/Jun/18 3:13 PM

timespent	32400	61200
WorklogId	28880	
Michael DeMilia made changes - 22/Jun/18 3:14 PM		
timespent	61200	90000
WorklogId	28881	
Michael DeMilia made changes - 22/Jun/18 3:14 PM		
timespent	90000	115200
WorklogId	28882	
Michael DeMilia made changes - 22/Jun/18 3:14 PM		
timespent	115200	140400
WorklogId	28883	
Michael DeMilia made changes - 28/Jun/18 11:26 AM		
timespent	140400	158400
WorklogId	28890	
Michael DeMilia made changes - 28/Jun/18 11:26 AM		
timespent	158400	187200
WorklogId	28891	
Michael DeMilia made changes - 28/Jun/18 11:26 AM		
timespent	187200	205200
WorklogId	28892	
Michael DeMilia made changes - 29/Jun/18 6:43 PM		
timespent	205200	219600
WorklogId	28895	
Michael DeMilia made changes - 29/Jun/18 6:43 PM		
timespent	219600	248400
WorklogId	28896	
Lei Weng made changes - 30/Aug/18 10:52 AM		
assignee	Michael DeMilia	Kalkidan Lebeta