

Curator guidelines for the AmP collection

Starrlight Augustine^{1*}, Bas Kooijman², Konstadia Lika³, Nina Marn⁴

April 10, 2021

^{1*}Akvaplan-niva, FRAM Center, Tromsø, Norway

²Department of Theoretical Biology, VU University, Amsterdam, The Netherlands

³Department of Biology, University of Crete, Heraklion, Greece

⁵Ruder Boskovic Institute Zagreb, Croatia

Abstract

This document presents the current guidelines for curators when reviewing AmP (add-my-pet) submissions. It first describes the curator tasks, then gives an overview of the curation process (from submission to upload). Next, the document describes how to curate an entry. Specific points to pay attention to are listed in the final section. Curation is supported by AmPtool function `curator_report` which is available on GitHub.

1 Curator board: tasks

Curators are expected to:

1. participate in regular curator board Skype meetings, with an interval of some 2 weeks, for 1 or 2 hours, where priorities, challenges and problems and fixes around AmP are discussed
2. regularly curate incoming AmP submissions
3. report any problem with AmP, DEBtool, websites and DEB theory in general, actively searching for improvements and innovations, to be discussed in board meetings
4. stimulate the scientific community to contribute to and use the AmP collection

2 AmP curator file sharing system

Curators make use of a file sharing system: Dropbox/DEBtool_dev which comprises the following 2 directories:

1. AmP_review
2. AmP_upload_queue

Each curator may use a personal folder for developments, but should take care to not use a lot of heavy files. The AmP dropbox is administered by Starrlight.

3 AmP upload queue

Current web-uploader: Bas

The web-uploader is responsible for taking entries from the upload_queue directory and uploading to the web.

4 Curation procedure: overview

When an entry e.g. Pelagia_noctiluca is submitted to the AmP collection it needs to be curated before uploading to the web. This section describes the workflow from submission to web-upload.

1. All entries which are submitted to the collection are placed in the 'AmP_review' folder
2. A curator is assigned an entry
3. When the Curator starts the review she/he renames the entry to Pelagia_noctiluca_Name. It is obligatory to append the initials of the reviewer (_Name) to the folder the moment the entry is under review.
4. The curator curates Pelagia_noctiluca on her/his local computer. The function of the AmP_review folder is to:
 - (i) keep track of who is curating which entry
 - (ii) keep track of how long the curation process is going on (via the modification date)
 - (iii) keep track of which entries have not yet been assigned a reviewer
5. There is a **14 day** deadline for reviewing an entry

6. Once the entry is accepted, add to the mydata-file:

```
metaData.curator = {'YourFirstName YourLastName'};  
metaData.email_cur = {'myname@myuniv.univ'};  
meataData.date_acc = [year month day];
```

7. Place `Pelagia_noctiluca` in the `AmP_upload_queue` folder. (make sure no extra file are in the folder.
8. Delete `Pelagia_noctiluca_Name` from the `AmP_review` folder.
9. The webmaster uploads to the web and deletes from the `AmP_upload_queue` once it is successfully online.
10. The Curator notifies the person who submitted an entry that the entry is online. The person who submitted the entry is invited to inspect the entry and is instructed that any modifications and/or corrections must be performed on the newly curated files.

5 Curating an entry: Protocol/ How to

The previous section described the general workflow: from submission to web-upload. This section describes the curation process itself.

1. File check: All submissions must comprise 4 files:

```
'mydata_my_pet.m'  
'pars_init_my_pet.m'  
'predict_my_pet.m'  
'run_my_pet.m'
```

(The submission should include csv files if the mydata file uses them.)

2. Open the 4 files in the matlab console
3. Make sure that all function names match their file names
4. Strip all commented text that was meant to guide the user (and was functional in the basic templates, but not in actual submissions). Be aware that all files are copy-paste-modified versions of other files, so material that should have been removed, might still be present, including old bib's and old refs of data. Delete all that.

5. Check the spelling of the species name as well as if it is already in the collection by running:

```
curator_report(speciesnm, 1);
```

Note: This executes the first module of the curator_report which calls function 'check_speciesnm' that checks the species name and lineage against what is present in the Catalog of Life. Keep in mind that deviations are not always wrong, and that the classification should always be consistent with the one used in AmP (which are in the lists of lists in AmPtool/taxa).

6. Check if the ecoCodes that are used occur on the ecoCode-page.

```
curator_report(speciesnm, 2);
```

7. If the submitted entry is a revised version of an already existing entry : (i) make sure that it started with the already existing entry and has the correct author history. (ii) Make sure there is/ are (a) discussion point(s) explaining the key improvements of the new entry
8. check if any of the data are guessed in the mydata-file - if so, make sure this is justified in a comment and check the validity of the justification
9. check the goodness of fit. Are the predictions sufficiently close to the data and, if not, does the author discuss that adequately in a discussion-point?
10. Check the data labels and completeness levels by running Step 2 of the curator_report:

```
curator_report(speciesnm, 3);
```

Warnings are printed in case that zero- or uni-variate data types are unknown on DEB-wiki (http://www.debtheory.org/wiki/index.php?title=Zero-variate_data and http://www.debtheory.org/wiki/index.php?title=Univariate_data). If the labels are correct, these tables on DEBwiki should be edited. Clear all after editing to make sure that the labels are read again from DEBwiki by check_data. If expected COMPLETE is NaN, the combination COMPLETE and data-types should be added in function COMPLETE_data. If there are no uni-variate data, we must have data_1 = {};, not data_1 = {"};.

11. Check and correct the bibliography by running Step 3 of the curator_report:

```
curator_report(speciesnm, 4);
```

This step of the curator_report function runs Bibtex; make sure that you have it; the Miktex version works well. Function bbl2html converts Latex code to html and handles quite a few

special characters, but new ones might be added, if needed. The tilde above character still gives problems, because Matlab refuses to write txt that contains this character. Also the percentage symbol in e.g. web-address gives problems; make sure to double them. Reference to doi's should only be done in the doi-field, containing the number only, not as part of a web-address in a howpublished field. The `curation_report` opens the Bibliography in your systems browser to check the result and the proper functioning of the web links. To do this, the `curation_report` produces several files in the same directory as the `mydata-file`.

12. Escaped characters, such as `{}` might only occur in the Bibliography, not in comments, dicussion or facts

```
curator_report(speciesnm, 5);
```

13. Checking additional parameters by running

```
curator_report(speciesnm, 6);
```

Entry-specific parameters such as `E_Hh` are actually used in `predict`. If not, remove. Keep in mind that extra maturity levels are scaled automatically in `parscomp_st` and that the scaled version is used in the `predict` file as input to `DEBtool` routines. This step will also notify you if automatic chemical parameters have been overwritten. If this is the case, check how and why and make sure it is consistent.

14. Look at which parameters have been fixed and freed by running step 5:

```
curator_report(speciesnm, 7);
```

15. Generate implied properties and examine those carefully:

```
curator_report(speciesnm, 8);
```

16. When the curation is over and all is acceptable, make sure that all extra files (`.bib`, `.html` etc) are deleted from the folder. Furthermore, make sure the figures are saved. Then put the ready to be uploaded entry in the `upload_queue`.

6 Curation: specific points to watch out for

6.1 Mydata_my_pet.m

1. Check the structure of the file; if incorrect, adapt. Discussion should be on model-application/data-interpretation, facts on literature (so related to references). The sequence should be
set metaData: 0-var data, 1-var data, weights, pseudodata
metaData-field should be filled in the sequence: taxon-info, T_typical, data_0, data_1, COMPLETE, author-info, author_mod-info, curator-info
1-var data-fields should be filled in the sequence: data, units, label, temp, bibkey, comment
pack auxData, txtData
set rest of metaData: grouping plots, discussion, facts, links, references
2. Do all data that require temp-specification have that? Naturally, all rates and times, but also think of all L or W that do not have an associated maturity level. If not, return to author
3. In the case of a bad fit and if possible, check that L^3/W does not vary too much over events (h, b, j, p, m) . If they do, discuss with author.
4. Check that Ri is well documented
5. Is COMPLETE setting reasonable (cf. <http://www.debtheory.org/wiki/index.php?title=Completeness>)? If not, adapt.
6. Is temp_typical in the neighborhood of temp-settings of 0-vars data? If not, adapt.
7. Check that comments are added to txtData if assigned
8. Check that title is specified for each grouped plot. The grouped plots need to have: females before males ! From high to low food, temperature, elevation
9. Check that all units are standard and, if not, are dealt with properly. If not, return to author.
10. Check that all the changes in the structures 'weight', 'data.psd' and 'weight.psd' are properly justified with discussion points. If not, contact author.
11. when references are a web address: add a note = {Accessed : 2015-04-30} filed to the reference if you are motivated. it might be worth looking at the content of the links to internet pages and see if the content matches what the author wrote about it.

12. Check that a ref to Wikipedia is included, if possible
13. Check that there is no trailing space between the closing `}}` and `''` for the last field of the bib-item, else the generation of the bib-file will fail.

6.2 Predict_my_pet.m

1. Is the structure of the file standard: after unpacking: tempcorr, 0-var (life cycle, hatch, birth, metam, puberty, ultimate, reprod, death, packing), 1-var.
2. If there are extra parameters, check if this allows for new constraints that should be added as customized filters
3. Is temperature correction properly indicated (with T)
4. Are dry and wet weights clearly distinguished (with Ww and Wd)?
5. Are physical and structural lengths clearly distinguished (with Lw and L)?

7 DEBwiki: editing pages

DEBwiki uses Wiki markup, so same rules for e.g. tables, links, sections etc. apply.

A lot of the pages are setup as templates. This means that they need to be edited externally (at the template page) and then are loaded in the main page. When a template is used (for example, for Upcoming events, and Past events) in the Edit view of the wiki page there will be a comment explaining and pointing to the corresponding template. For example, the page http://www.debtheory.org/wiki/index.php?title=Upcoming_events comprises of two sections – each one a template. When one clicks the Edit option on either section, e.g. Upcoming events, an editor opens where you can see which template is used (in this example, the executed part is: `"{{Template:Upcoming events}}"`) with a comment on the location of the template: in this example, you will see `"<!-- To edit upcoming events:`

`http://www.debtheory.org/wiki/index.php?title=Template%3AUpcoming_events -->"` .

(Note: Sections of text in the editor which are enclosed by `<!-- ... -->` are commented out, so you do not see this in the 'Read'(or Published) view)