DOI minting

During the EBAS DOI meeting 2019-09-17 a separate task was kicked off to investigate at which point in time a DOI should be minted in EBAS. This document should provide more information.

Before going into consequences of minting the DOI at different points in the data lifecycle, we need to

- have a clear overview about the current processes involved in the EBAS data lifecycle and
- know which meta information about the state of a data submission is needed for triggering the DOI minting

Requirement: What are the exact Processes in the EBAS data Lifecycle today?

In an approach to collect the current processes in EBAS data curation work, an EBAS Data Lifecycle Flowchart was generated.

The analyses showed several shortcomings in our current processes, the one relevant for the DOI coining is that the lifecycle state information of data in the database is unclear. We have no clean concepts of data embargo and the QA state of the data.

Requirement: What do we need to know about the State of a Data Submission in the Data Lifecycle?

Data Quality (QA state):

Each submission inserted into EBAS should undergo a manual inspection.

- There is a formal process for VOC data (data are first QA'ed by ACTRIS, before being inserted into EBAS)

- For all other data types and projects we lack a formal definition of the QA process. The data should at least be plotted and inspected by a human.

  - sometimes this is performed before insert
  - sometimes after insert (which means bad data are published already)
  - sometimes the inspection is not performed at all (which means bad data published and we will maybe not discover it)
  - no common rules or standardised process (depends on person who processes the data)

- Should we think about systematic review by the scientific responsible EBAS person? --> supported by e-reports, closure-report etc.
- There is no formal status in the database whether a submission in quality assured or not.
- Are there plans for formal QA in some projects (ACTRIS)?

Embargo Periods:

Some projects have a *preliminary vis-à-vis project (e.g. EMEP*preliminary). This is mainly used to control data access to open data during data embargo periods.

Possible use of _preliminary Projects for QA Status

The use of the _preliminary projects is not strictly defined so far. Although they were introduced for embargo periods and the related data access restrictions, it is often also seen as a QA status.

But is the combination of embargo and QA state a correct model for EBAS?

In some cases yes. E.g. in EMEP, the embargo period is used for processing the data and writing the data report. After the data report is published, the embargo period ends. At the same time the data are considered quality checked.

But is this valid fo all projects?

Control questions:

- Is there (will there be) a project with defined QA process and an embargo period, where not both are set at the same time?
- Does "not QA'ed" imply the same access restrictions as the data embargo state? (The fact that data are not QA'ed, does not necessarily mean they should not be accessible, on the other hand a data embargo means exactly that)

It could be an advantage to keep those two concepts apart.

Alternative: separate QA Status

If we decouple QA from embargo (i.e. _preliminary): Do we need a QA flag per project or a global QA flag? If different projects will have different QA procedures and requirements, it could happen that:

- Different QA's for different projects are performed at a different points in time (then one is passed, the other one is pending)
- One QA is passed while the other is failed

If we want to open for different QA procedures for each project we need to accommodate this in the data model as well. -> Diffferent QA states for each project which the submission is assigned to.

Finally -- the DOI minting!

When should we mint a DOI?

1. On insert (regardless of QA state and embargo periods).

   The digital object starts to exists with insert into the database.

   Pro:

   - Easy to implement

   Con:

   - In case of corrections, re-submissions etc we potentially collect a lot of deleted objects.
   - Insert into database (important core process) depends on external system (i.e. the Registration Agency). Is the service down or we miss internet access, we cannot insert data locally in our database. We would need to find a workaround for those situations (which adds to complexity).

2. On approved QA status

   Only "final" data get a DOI.

   Pro:

   - More stable DOI system, less changes and corrections as with (1.).

   Con:

   - In case of multiple projects - DOI would need to be minted on first projects QA approved. Thus, there is a possibility that the data fail another projects QA.
   - Analog to (1.): Setting QA state depends on external system (probably less severe than insert)

3. When published (the embargo period ends)

   The DOI is minted when the digital object is published, i.e. when a _preliminary project is changed to a non preliminary. In case of multiple projects the DOI would be minted on first projects publication.

   Pro:

   - More stable DOI system, less changes and corrections as with (1., 2.).

   Con:

   - Analog to (1. and 2.): Setting QA state depends on external system (probably less severe than insert)

4. Manually on demand

   The DOI minting is initiated manually on demand. This can be when data are published (e.g. for EMEP), or when QA is passed. Bulk minting can be done with the same arguments as e.g. publishing.

   Pro:

   - More stable DOI system, less changes and corrections as with (1. and 2)
   - No dependency on external system for other EMAS processes
   - More flexible, minting can happen at different points in time for different data types, projects etc.

   Con:

   - Separate manual step which must be administrated

5. Automatically but separated from other processes

   The DOI minting is done automatically depending on other state information. There may be different rules for different data types, projects etc.

   For example a default trigger for DOI generation could be defined as: When a data submission is in the database, has a "QA passed" state and is published (not _preliminary), the DOI will be generated automatically.

   Pro:

   - No dependency on external system for other EBAS processes
   - More flexible, minting can happen at different points in time for different data types, projects etc.
   - No manual step is needed, no administration

   Con:

   - More effort for implementation needed (at least more than 1.)

Proposed changes (DRAFT):

- Define formal QA rules for each project
- Keep _preliminary projects for controlling embargo periods

  - improve ebas_insert to automatically set _preliminary according to rules (e.g. EMEP data younger then 2 years ago -> insert as preliminary automatically)

- Add a separate QA status for each project which is assigned: values can be passed, failed, not performed, N/A (no formal QA procedure for this project)

    - imporve ebas_insert to set QA status according to rules (e.g. set QA passed for ACTRIS VOC files)

- Data access is only controlled by _preliminary (embargo), data which are not QA'ed but are in the database are available.

- Status falg should be visible in the metadata in EBAS web and should also be exported in data files

- DOI minting is triggered according to alternative 5. above