Validation Changes to Enable Metadata Quality Improvements

Problem: We all want to improve our metadata quality, but even seemingly simple things like correcting platforms, instruments, sensors and campaigns is very difficult (if not impossible) when those collections have granules.

Details: With the aim of maintaining integrity of the metadata, the CMR (and previously ECHO) has some validation rules regarding what fields can be updated on a collection and on a granule. Some validation rules are very restrictive and enforce referential integrity. Additional attributes and Projects /Campaigns are two examples where the CMR will not let providers modify the collection in a way that breaks referential integrity of the granules.

Example:

1. *Collection A has 1 project listed, "my project". It has 500 associated granules which also specify "my project".*

2. *A metadata curator attempts to improve the collection and changes the project to "Correct Name". This change is rejected by the CMR as it attempts to change a project name referenced by existing granules.*

Currently the only way to modify/remove existing Projects/Campaigns or Additional attributes would be to reingest the corrected collection and granules.

Platform, instrument, and sensor validation works differently. Collection level platforms, instruments and sensors are allowed to be changed at any time, even if it that means that the collection level information no longer matches the granule level information. However, on granule ingest, granule level platform, instrument, sensor must either match information in the collection or be blank. Blank values "inherit" the information from their parent collection.

Example:

1. *Collection A has 1 platforms listed, "AM-1". It has 500 associated granules which also specify "AM-1". And 250 granules that don't specify a platform at all (thus inheriting their parent collection information).*

2. *A metadata curator corrects the collection and changes the platform to "Terra". This change ingests without error into the CMR.*

3. *At this point, 500 of the granules will only be returned by searching for "AM-1". Searching for "Terra" will return 250 of the granules.*

4. *The next granule is ingested contains the platform "AM-1". This granule is rejected by the CMR because its platform is not one of the ones associated with the collection.*

For a very detailed understanding of the current rules, see: https://wiki.earthdata.nasa.gov/download/attachments/68387948/gcmd_granule_collection_field_relationship_demo.pdf?version=1&modificationDate=1456955342937&api=v2

Potential Options:

Given the situation described above, there are several potential options being discussed to reduce to complications in updating metadata. We want to hear thoughts from the providers regarding why each of these would or would not work. This list is not exhaustive and we would love to hear other ideas too. Please add your comments to this page to provide feedback.

1) **Relax Validation Rules**: Change all referential integrity validation rules to "warnings" only to allow for quick changes to be made to metadata. This would allow more inconsistent metadata into the CMR. The onus would then be on the providers to follow-up on restoring collection and granule level integrity in a timely manner.

Question 1A: Are DAAC clients relying on the granule level metadata to match the collection level metadata?

2) **Smart Validation Rules**: Change the logic of the validation rules to check if any granules would be negatively impacted by a collection level change. If no granules are impacted, allow all changes to go through. Otherwise, reject the change. Note: While this would make some changes easier, it would most likely make others harder. This would have to be used in conjunction with some other option(s) below.

3) **Increased Validation Rules + Auto Bulk Update**: Enforce all referential integrity by having collection level changes trigger a background job to update all associated granules. This would require coordination between the CMR team and the provider to ensure that all incoming granules are updated to match the newly modified collection AS SOON as the collection is modified.

Question 3A: In this case, the granule level metadata in CMR would not match the metadata (or data) at the DAACs. Users could see metadata in a search client showing "Terra", but download the original data and it would show "AM-1". How big of a problem is this for users?

4) **Remove Granule Fields**: Remove platform, instrument, sensor and campaign from the granule level metadata. Granules would inherit this information from their parent collections for search purposes. This would solve the initial problem. However, it would also break the use case of a collection having multiple sensors listed and each granule having a subset of those sensors.

Question 4A: How often are DAACs listing multiple sensors at the collection level and having the granules only list a subset of the sensors?

SDPS uses MISR sensor information to process MISR Browse granule.

5) **Aliases: The CMR could maintain a list of** "known aliases" for certain fields. For example, "AM-1" could be a known alias for the canonical "Terra". Metadata could be ingested and updated using any known alias and still be considered valid. The CMR's search system could convert from aliases to the canonical form and vice-versus so that users could search for "AM-1" and find "Terra" and search for "Terra" and find "AM-1". Facets could always show the canonical form of "Terra", but then be able to group "AM-1" data into that facet since it is a known alias.

Question 5A: In this case, the native metadata in CMR would not always match user's search terms. For example, a user could search for "Terra" and then open up the native version of the metadata (or download the data) and see "AM-1". How big of a problem is this for users? Are users very familiar with the known aliases in their fields?

6) **Same Old:** Reingest collection + all granules

7) **Support multiple values**: Is it possible to add new collection values so that the collection can accurately reflect the granule inventory? For example, the collection currently uses Platform "AM-1", the metadata curator now adds Platform "Terra". The Data Provider starts ingesting granules referencing "Terra" and also re-ingests the old granules that reference "AM-1" with "Terra." This avoids the need to delete the entire collection. Question: is it possible to update a collection that contains a invalid value for Platform while still preventing new collections from referencing the invalid Platform (i.e. grandfather in the invalid values). Eventually it should be possible to update the collection and remove the "AM-1" Platform once all the granules referencing it are removed; removing the Platform would be rejected if granules still reference it.