Deduplication Workflow

Because NGDS is a system of largely independent, federated data providers, it is likely that a given resource (document, dataset, application) will be registered in more than one catalog, and that particular data items might be published in different datasets. The first case is referred to as metadata duplication--multiple metadata records that describe the same resource. The second case is data duplication, in which the same feature or observation is described by records in different datasets.

Ideally metadata records that document the same resource would be merged such that there is only a single metadata record (identified by a unique gmd:fileIdentifier) in the system that documents the resource, and that record would include all the available distributions for the resource. Given the distributed nature of the system and the cost of merging such records and propagating the updates, this is unlikely to happen. Practical options for aggregating catalogs to deal with metadata duplication in harvested metadata include identifying duplicate records and merging them in their metadata collection, or keeping both harvested records and adding an explicit link between them to indicate that they document the same resource.

Data duplication is a trickier issue. Features like wells may be described in multiple databases, with information collected at different times by different observers. Data compiled into a dataset that is to be deployed as a feature service may be sourced from one or more such databases. Different data compilers may assemble data for a feature service from different source databases. The problem is exacerbated by the lack of intrinsic identifiers for things like wells, rock samples, or observations--different observers may use equally valid schemes to identify things for their intended application. Feature location and observation time stamps are about the best thing we have to go on, but those may be reported with different precision or be inaccurate, making establishment of equivalence a probabilistic exercise. Our proposed solution in this situation is to provide a link between records that describe features that might or are known to be the same feature.

This document describes a workflow to identify and correlate duplicate metadata in harvested metadata collections and duplicate data in related feature datasets. We refer to this process as deduplication. Multiple iterations of deduplication will likely be necessary, in which each iteration uses different approaches to detecting duplication.

**Identifying Duplicate Data**

Iteration 1

With tier 3 data services or documents metadata, a CSV dump of a feature type of interest is an easy way to begin deduplication. As these datasets are standardized, expected field headings exist in all federated catalogs which can be queried for correlation. Correlation of databases can begin in Microsoft Access or SQL, joined based on a foreign key. By successive tests, search for duplication by:

1. IDs (API numbers, labels, other IDs)
2. Titles (if assessing documents metadata)
3. Location information (Latitude, Longitude)
4. Names (well names, operator names, spring names)
5. Dates

This will generate a list of *likely* and *certain* duplicates. Some may remain uncertain due to slight differences in records due to data collection techniques, for example. Location information is inevitably expressed at varying levels of certainty and with a varying number of significant digits. To cull the list of certain duplicates down further, additional iterations of deduplication efforts are at this point necessary.

Iteration 2

Place shapefiles of the list of possible duplicates into ArcMap or another spatial analysis tool. If using ArcMap, perform a Spatial Join between the two datasets. This correlates data from two source databases based on relative spatial location. A series of joins can be performed:

1. The default ‘match\_option’ of a join is INTERSECT, which only matches features based on intersection. These duplicates will likely have been caught in the operations from Iteration 1.3.
2. WITHIN\_A\_DISTANCE ‘match\_option’ allows features to be joined if they are within a specified distance. This is likely the most useful option for NGDS deduplication based on the location uncertainty that is sometimes associated with field data. This is also the most time-consuming option, which requires the worker to manually inspect the data in matching records to assess if locations of nearby point data are likely to be the same. For example, this may be evidenced by a well drilling pad visible in ArcMap base layers or GoogleEarth imagery near a well header data point.

A spatial join can also be performed in an open-source GIS application such as QGIS. With both datasets loaded into QGIS, from the Vector menu, choose *Data Management Tools* and *Join attributes by location*.

**Relating Duplicates**

Once a list of *certain* duplicates is found within the system, a correlation table must be set up to capture those relationships. See <http://modeling.sourceforge.net/UserGuide/design-rels-many-to-many-principle.html> for a visual representation of a correlation table. Since NGDS is structured so that each resource or data point has a URI (unique resource identifier) which represents an individual record, these will be entered into the correlation table and used as foreign keys to any other associated URI. This table will allow complex queries, such as "show everything that is related to x". Finally, the found and indexed related URIs may be indicated as an additional related resource for end users to access easily, being exposed to the foreign key’s additional documents and other available data.

NGDS tier 3 data services have a RelatedResouces field which houses additional documents (usually live links) as well as foreign keys to related URIs within the system. For these data types, the related URI can be entered into the actual data in the RelatedResources field. This increases the amount and quality of data for any particular resource or data point available to the end user, accessible by browser.

**Appendix A: NGDS workflow for exporting all WFS features as CSV**