

The basic steps are:

1. identify what's being moved and to where

-Are they datasets only or dataverses? do they have child and further nested/grandchild objects.
Browse the UI to gain a quick understanding of how simple or complex the structure is.

2. look up id's of objects being moved (and their child objects) and id of destination object

select id, name, alias from dataverse where alias='<dv alias>';

select * from dataset where identifier='<identifier in doi string>';

select dtype, id from dvobject where owner_id=<parent id> and dtype!='DataFile'; (repeat for each child level)

3. update owner_id of objects being moved to be id of destination object (has to be a dataverse)

update dvobject set owner_id=<destination parent id> where id=<object being moved>;

4. index all objects moved and their child objects using curl and the index commands for datasets and dataverses.

curl http://localhost:8080/api/admin/index/datasets/<db id>

curl http://localhost:8080/api/admin/index/dataverses/<db id>

Moving datasets from one dataverse to another is the simplest case -no child objects.

Moving dataverses is most complicated, especially if they have sub dataverses since you need to index all child objects by id but only change the owner_id of the parent being moved.

I usually create a simple text doc with the output of the select dtype, id from dvobject query, then find/ replace the dtype part with the curl part and run that on the web server, after first updating the owner_id in the db. In complex moves, I list each dv and sub dv and move them separately, verifying in UI that it was moved correctly.

Kind of tedious but we have this as a feature request:

<https://github.com/IQSS/dataverse/issues/2278#issuecomment-247366379>