# Scenarios for identifying data needs

1. Test migration from AT to AS. Note errors.
   1. Install AT-AS migration plugin.
   2. Conduct a test migration of all data currently in AT to AS.
      1. Includes assessment records.
      2. See #4 below for subjects.
      3. See #5 below for names.
   3. Create a list of migration errors encountered.
      1. Things that cause migration to fail
      2. Mapping errors
   4. Create a list of style/content errors to fix.
2. Run EAD checkers over EAD files. Note errors.
   1. Yale Schematron file
   2. Harvard EAD checker
      1. The Harvard tool is supposed to also check for container list errors.
   3. Create a list of data errors that the checkers return.
   4. Note if any style/content errors are found.
3. Test import of Manuscript EADs to check for container list errors.
   1. Create a list of migration errors.
   2. Note if any style/content errors are found.
   3. This will supplement anything not caught by Yale or Harvard tools.
4. Subject database ingest to AT database. Migrate to AS.
   1. Create a backup copy of the current subjects in AT including the resources that they are linked to.
   2. Scrub all subjects in AT.
   3. Import subjects from the “cataloger’s” database to AT.
   4. Include in test migration from AT to AS (#1 above).
5. Name authorities evaluated by node export. (Names are already in AT)   
   Run report so that you can see all the names and what fields they are mapped to – that info would generate the error list
   1. Extract the name authorities from the SQL database. (backend of Collection Guides database)
   2. Push into AT.
   3. Include in test migration from AT to AS (#1 above).
   4. Create list of migration errors.

# Migration sketch

1. Extract data from SQL database (backend of Collection Guides database). Clean data.
2. Clean additional data directly in AT.
   1. Data that cannot easily be extracted from SQL database or cannot be cleaned in bulk.
3. Import subjects from the “cataloger’s” database to AT.
4. Push AT-AS migration.
5. Fix remaining errors (hopefully few) using API.