

Moving moving data between authoring platform instances.

The authoring platform is a system of systems and as such one can not simply take a backup of the terminology server from prod, restore it to UAT & everything will work.

Todo: look at creating a script to do all this in one.

Systems:

- Termserver - Use the ansible restore script and set the latest backup either in the hostvar file (e.g. [uat-ms-authoring.ihtsdotools.org](#)) or on the command line e.g:
`ansible-playbook -i ../ansible-inventory/uat -u ansible terminology_server_dbContent.yml -l uat-ms-authoring.ihtsdotools.org`
the relevant settings are:
`import_zip_base_url e.g. : import_zip_base_url: "https://s3.amazonaws.com/prod-ms-authoring.backup.ihtsdo/"`
`import_term_serv_filename e.g. : import_term_serv_filename: "snowowl_prod-ms-authoring_20170207_0202"`
- JIRA - If the relevant project(s) already exist then check the project counts are high enough :
[Bumping JIRA Issue Counter for the Authoring Platform](#).
If creating a new project and thus magic ticket etc first see:
[Creating An SCA Project in Jira](#) or [creating an SCA project \(JIRA 7\)](#)
Another technique is to backup prod JIRA to an xml/zip file, transfer this from prod to uat and then create just the required projects (no magic ticket) or empty the projects if they already exist.
Then use the JIRA import project functionality to import each project individually.
- CIS - There is a pull job for pulling data out from a cis instance (in this case prod) which is used thus:
`ansible-playbook -i ../ansible-inventory/prod -u ansible cis_sync_pull.yml -l cis.ihtsdotools.org`
and which would then be followed on the same machine by the push to uat thus:
`ansible-playbook -i ../ansible-inventory/uat -u ansible cis_sync_push.yml -l uat-cis.ihtsdotools.org`