Texas A&M University Libraries experiences

Current infrastructure and deployment layout:

- Duplicate instance of Folio Q3.2 on K8s infrastructure – in its own namespace using dedicated databases for Okapi and the Folio modules.
- Three tenants with varying amounts and types of data:
  1. Diku tenant - with default reference and sample data
  2. Tamu tenant - with various data types from our migration work
  3. Tern tenant - with various data types migrated from Voyager

Upgrade steps:

- There is no current documentation outlining the exact steps or processes by which the data upgrade should happen. Used Taras’ working Google Doc as a starting point:

  Automatic migrations dev and prod rollout plan

- Built and deployed an upgraded Okapi v2.36.1 container.
- Deployed Q4 modules on K8s infrastructure alongside the running duplicated Q3.2 containers in the same namespace, using new URLs for each.
- Built the Folio Q4 Stripes front-end containers for each tenant and deployed on K8s infrastructure.
- Create backup of Okapi and Folio module databases.
- Built and ran update-deploy-<tenant> K8s Jobs, one for each of our tenants. The K8s Job runs a script that:

  1. Pulls in the latest module descriptors from Index Data's module registry to Okapi's \_/proxy/pull/modules endpoint
  2. Forms and Posts the new Q4 module's deployment descriptors to Okapi's \_/discovery/modules endpoint
  3. Posts an install.json file with a list of the Q4 modules to be upgraded to, to the Okapi tenant endpoint with variables set to false per tenant:

    \_/proxy/tenants/$STUDENT_ID/install?deploy=false\&spreRelease=false\&tenantParameters=loadSample%3D$SAMPLE_DATA%2CloadReference%3D$REF_DATA

Outcomes:

- No tenant upgraded successfully. Each was left in an incomplete and unworkable state with no way to easily roll back without restoring a backup of the databases for Okapi and the Folio modules.
- You could still log in to each tenant, but dependencies were not being met in the Settings page, and some app pages would not load without error.
- Once the process errors out, no way to undo changes or have it continue.
- Each tenant failed during the upgrade, logs from the create-deploy-<tenant> K8s Job erroring out differently:

  Diku tenant error (MODFISTO-83 - Migration script improvement to support transition of non-unique index to unique one CLOSED):

  POST request for mod-finance-storage-4.1.1 \_/tenant failed with

  \*
  CREATE UNIQUE INDEX IF NOT EXISTS budget_name_idx_unique ON diku_mod_finance_storage.budget ( lower (f_unaccent(jsonb->>'name')) ) ;
  \*

  Key (lower(f_unaccent(jsonb -&gt; 'name':text)))=(gifts-one-time-fy19) is duplicated.

See workaround described in the Edelweiss Release Notes

Tern tenant error (ERM-675 - mod-license upgrade from Q3.2 to Q4 fails CLOSED):

  POST request for mod-licenses-2.0.0 \_/tenant failed with

Tamu tenant error (ERM-675 - mod-license upgrade from Q3.2 to Q4 fails):
Recommendations: