DRAFT

VIVO DSpace Integration Project: Potential Next Steps: Follow-up Phases

Some potential next steps are outlined below.

Phase 2
- Requirements
  - Completion of exchange information task
    - Transformation of other entities - repositories, communities, collections
    - Integration of project within VIVO codebase
      - Moving sh files to Java code
      - Moving https://github.com/vivo-community/DSpace-VIVO to https://github.com/vivo-project/VIVO
  - VIVO UI for communication with DSpace
- Usage scenario
  - A new fresh and empty instance of VIVO is installed. There is a running instance of DSpace. **Log into the VIVO as an admin, configure and run the migration process from the VIVO UI.** After initial migration of data from DSpace, VIVO UI for editing and adding new items has been turned off (VIVO is used in read only mode). Weekly or daily, there is the update of information in VIVO during the night by dropping triplestore, and running migration with a fresh copy of DSpace data. **The update is scheduled through the VIVO UI**. **Metadata about Repositories, communities and collections can be migrated as well.**
- Estimation of effort: 150 hours

Phase 3
- Requirements
  - Establishing stable links between DSpace and VIVO items and terms
    - Based on added metadata field in DSpace
    - Based on added metadata field in VIVO
    - Based on unique automatic generation of VIVO ID by using DSpace item fields
    - Based on external mapping file (VIVO instance – DSpace instance)
- Usage scenario
  - A new fresh and empty instance of VIVO is installed. There is a running instance of DSpace. Log into the VIVO as an admin, configure and run the migration process from the VIVO UI. **As part of configuration, the admin can configure strategy for establishing links between DSpace and VIVO items and terms.** After initial migration of data from DSpace, VIVO UI for editing and adding new items has been turned off (VIVO is used in read only mode). Weekly or daily, there is the update of information in VIVO during the night by **updating only changed metadata in the DSpace instance**. **The established link between DSpace and VIVO items is mandatory for the update.** The update is scheduled through the VIVO UI. Metadata about Repositories, communities and collections can be migrated as well.
- Estimation of effort: 200 hours

Phase 4

- Requirements
  - Description of crosswalks/mapping in formal language
    - Adjusting default crosswalks for customized VIVO and DSpace instances
- Usage scenario
  - A new fresh and empty instance of VIVO is installed. **VIVO might be based on localized/customized VIVO ontology (for instance German VIVO ontology).** There is a running instance of DSpace. **DSpace might be based on a localized/customized internal DSpace model. Moreover, it might extend the default set of item types.** Log into the VIVO as an admin, configure and run the migration process from the VIVO UI. As part of configuration, the admin can configure strategy for establishing links between DSpace and VIVO items and terms. **Moreover, as part of configuration, the admin can modify default crosswalks.** After initial migration of data from DSpace, VIVO UI for editing and adding new items has been turned off (VIVO is used in read only mode). Weekly or daily, there is the update of information in VIVO during the night by updating only changed metadata in the DSpace instance. The established link between DSpace and VIVO items is mandatory for the update. The update is scheduled through the VIVO UI. Metadata about Repositories, communities and collections can be migrated as well.
- Estimation of effort: 200 hours

Phase 5
- Requirements
  - Sending notifications to DSpace from VIVO
    - Depositing pdf files from VIVO UI into DSpace
- Usage scenario 1
  - A new fresh and empty instance of VIVO is installed. VIVO might be based on localized/customized VIVO ontology (for instance German VIVO ontology). There is a running instance of DSpace. DSpace might be based on a localized/customized internal DSpace model. Moreover, it might extend the default set of item types. Log into the VIVO as an admin, configure and run the migration process from the VIVO UI, **as well as access parameters with adding files privileges**. As part of configuration, the admin can configure strategy for establishing links between DSpace and VIVO items and terms. Moreover, as part of configuration, the admin can modify default crosswalks. After initial migration of data from DSpace, **DSpace UI for editing and adding new items has been turned off (DSpace is used in read only mode). VIVO users can deposit, download or update files through VIVO UI.**
- Usage scenario 2
  - A new fresh and empty instance of VIVO is installed. VIVO might be based on localized/customized VIVO ontology (for instance German VIVO ontology). There is **NO running** instance of DSpace. **Install the new instance of DSpace.** DSpace might be based on a localized/customized internal DSpace model. Moreover, it might extend the default set of item types. **Log into the VIVO as an admin, configure the DSpace server location and access parameters with adding files privileges.** As part of configuration, the admin can configure strategy for establishing links between DSpace and VIVO items and terms. **DSpace UI for editing and adding new items has been turned off (DSpace is used in read only mode). VIVO users can deposit, download or update files through VIVO UI.**
- Estimation of effort: 150 hours

<u>Phase 6</u>

- Requirements
  - o Sending notifications to DSpace from VIVO
    - ▪ Depositing pdf files from VIVO UI into DSpace workflow
  - o Receiving notifications from DSpace
    - ▪ Changes in DSpace items reported to VIVO
      - ▪ DSpace notify VIVO about change in status of an item (e.g. First deposit, Published); when DSpace item is published, the URL is visible to VIVO users
  - o Communication between VIVO and DSpace should be based on
    - ▪ COAR notify (https://notify.coar-repositories.org/)
    - ▪ CSRF Tokens (https://github.com/DSpace/RestContract/blob/main/csrf-tokens.md)
- Usage scenario 1
  - o A new fresh and empty instance of VIVO is installed. VIVO might be based on localized/customized VIVO ontology (for instance German VIVO ontology). There is a running instance of DSpace. DSpace might be based on a localized/customized internal DSpace model. Moreover, it might extend the default set of item types. Log into the VIVO as an admin, configure and run the migration process from the VIVO UI, as well as access parameters with adding files privileges. As part of configuration, the admin can configure strategy for establishing links between DSpace and VIVO items and terms. Moreover, as part of configuration, the admin can modify default crosswalks. **Log into the DSpace instance and configure access parameters for the VIVO instance.** After initial migration of data from DSpace, **DSpace and VIVO continue to coexist in parallel. VIVO users can deposit, download or update files through VIVO UI. DSpace users (probably librarians) can apply institutional OA mandates. VIVO and DSpace instance data are synchronized at any moment by exchanging notifications between platforms.**
- Usage scenario 2
  - o A new fresh and empty instance of VIVO is installed. VIVO might be based on localized/customized VIVO ontology (for instance German VIVO ontology). There is NO running instance of DSpace. Install the new instance of DSpace. DSpace might be based on a localized/customized internal DSpace model. Moreover, it might extend the default set of item types. Log into the VIVO as an admin, configure the DSpace server location and access parameters with adding files privileges. As part of configuration, the admin can configure strategy for establishing links between DSpace and VIVO items and terms. **Log into the DSpace instance and configure access parameters for the VIVO instance. DSpace and VIVO continue to coexist in parallel. VIVO users can deposit, download or update files through VIVO UI. DSpace users (probably librarians) can apply institutional OA mandates. VIVO and DSpace instance data are synchronized at any moment by exchanging notifications between platforms.**
- Estimation of effort: 400 hours