Newspaper digitisation Home

Navigate space

The Danish newspaper digitisation project (http://en.statsbiblioteket.dk/national-library-division/newspaper-digitisation/newspaper-digitization) is an in-production example of Minimal Effort Ingest using Autonomous Preservation Tools. In this project we receive scanned newspaper pages in batches of about 25,000 pages along with MIX (https://www.loc.gov/standards/mix/), MODS (https://www.loc.gov/standards/mods/) and ALTO (https://www.loc.gov/standards/alto/) metadata. We receive two batches a day and a total of about 30 million newspaper pages throughout the duration of the project. All ingest, validation and enrichment preservation actions are performed with the Autonomous Components.

Each new batch of scanned newspaper pages must be ingested in our repository system, undergo a large number of quality checks and have access copies generated.

In keeping with the Minimal Effort Ingest model, we first ingest the batch of pages, and then perform the quality checks. Metadata is stored in DOMS, our Fedora Commons (http://fedorarepository.org/) 3.x based repository, whereas the data files are stored in our Bit-Repository (http://bitrepository.org/). We use Solr (http://lucene.apache.org/solr/) to index the content of DOMS for our discovery platforms.

We store an additional object in DOMS, the batch object, which represents the batch of scanned pages, rather than any single page. In this object, we store PREMIS (http://www.loc.gov/standards/premis/) (Events detailing which actions and transformations have been performed on the batch. This information is also indexed by Solr.

We implemented the Autonomous Preservation Tool model in what we call Autonomous Components. Each component corresponds to a single action, such as "Ingest batch into repository" or "Schema-validate all XML files in batch".

All autonomous components are Linux executables and have the following characteristics:

- Can query Solr for batch objects having specific combinations of PREMIS Events.
- Registers a component-specific PREMIS Event on the batch object after execution.

The current location of a batch in the workflow is determined by the set of PREMIS events present on the batch object - in other words which components have processed the batch so far. Each component knows which PREMIS events must be present or absent on a given batch for it to be ready to be processed by the component.

We have created Tree Iterators as a framework for autonomous components to handle batches in a storage-agnostic way. Tree iterators allow you to iterate through complex directory structures, whether in the repository or on disk, in a uniform way. With this framework, the autonomous components are able to work identically on batches not yet ingested, and batches inside the repository. This gives us great flexibility when testing, and allows us to easily re\-arrange which components should be run before ingest, and which should be run after.

System

Description of the platform used for ingesting and storing the digitised newspapers

- Architecture
  - A Batch Event System for simpler objects, such as newspaper titles
  - Autonomous components that trigger on non-newspaper batches
  - DOMS object model creation from tree — Description of the general method of creating the initial object hierarchy in DOMS based on a tree structure in a file system
  - Testing — Details the 5 levels of tests used in the project
- Autonomous Components
  - Autonomous Component Harness
  - Batch structure checker — Checks the structure of a batch
  - Metadata checker
  - Template Autonomous Component — twitter style description
  - Prompt Doms Ingester — This an autonomous component https://sbforge.org/display/NEWSPAPER/Autonomous+Components responsible for ingesting the metadata from a newly-uploaded batch-run into DOMS tree-structure, as described here https://sbforge.org/display/NEWSPAPER/DOMS+object+model+creation+from+tree.
  - Jpylyzer Hadoop component
  - Histogrammar Hadoop Component
  - Newspaper presentation copies Hadoop component
  - Manual QA flagger
  - Newspaper repository cleaner — This an autonomous component https://sbforge.org/display/NEWSPAPER/Autonomous+Components responsible for cleaning up after a roundtrip has been approved.
  - Doms Enricher
  - Roundtrip Approver
  - Edition records maintainer component
  - Title records maintainer component
  - Newspaper edition pdf presentation copies component
- Batch Description

        - Appendix 2B - JPEG2000 specifications
        - Appendix 2C - metadata per page MODS
        - Appendix 2D - metadata per publication and edition
        - Appendix 2E - metadata per film v2
        - Appendix 2F - nomenclature of files and file structure v2
        - Appendix 2J - metadata per page ALTO v2
        - Appendix 2K - metadata per page MIX v2
- Components — Lists the components making up the newspaper ingest system together with source code repositories ands responsible.
    - Batch-event-framework
    - Common Properties for Autonomous Components
    - Delay alerter
    - Newspaper process monitor — Process monitor for monitoring the process of the newspaper digitisation flow. The processmonitor is a webservice deployable in a Tomcat.
    - Statistics module — Provides functionality for generating statistics for a batch as xml, and accessing these through a web frontend.
    - Workflow restart trigger
- Release Procedures
- Systems — This is the common parent for the system pages.
    - Bit Preservation System — The Bitrepository https://sbforge.org/display/BITMAG platform is used for longterm preservation of the newspaper data
    - Digital Object Management System — The role and design of the DOMS system, our metadata storage system
    - JPEG 2000 image server
    - Newspaper Batch Event Framework
    - Newspaper Digitisation Process Monitor — A graphical web interface to show the current state for each batch. Wireframe http://htmlpreview.github.io/?https://github.com/statsbiblioteket/newspaper-digitisation-process-monitor/blob/master/aviswireframe/aviswireframe.html http://htmlpreview.github.io/?https://github.com/statsbiblioteket/newspaper-digitisation-process-monitor/blob/master/aviswireframe/aviswireframe.html
    - Newspaper MfPak Integration — This is a webservice component which pulls data from the mfpak database and presents it via a REST API to the UI System.
    - Ninestars QA Suite
    - Zookeeper lock server for the autonomous components

User stories

Describes the functionality (existing and proposed) of the Newspaper Digitisation platform through user stories.