Email Processing with TOMES Software

December 31, 2018

Introduction

The Transforming Online Mail with Embedded Semantics (TOMES) collaborative project is a multi-state initiative to identify state positions by function that generate archival email, based on guidance known as the Capstone approach promulgated by the National Archives and Records Administration (NARA). Additionally, the project aims to develop methodologies to transfer archival email accounts out of their native system and develop predictive coding and libraries to assist archivists to process the email accounts. The project team includes staff from the state archives of North Carolina, Utah, and Kansas.

One of the grant's performance objectives was to process at least ten email accounts designated as containing permanently valuable correspondence based on Capstone roles. State Archives of North Carolina (SANC) staff were able to successfully process ten email accounts held in the State Archives of North Carolina's digital repository. They used software developed in the TOMES project to transform, tag, and package the accounts into Archival Information Packages (AIPs) with metadata, which met the objectives to produce a cross platform PST to EAXS XML parser and to publish and NLP dictionary to flag named entities unique to state and local government.

To learn more about the project, visit the project website. For more details on the software developed, see the State Archives of North Carolina GitHub page.

Accounts Processed

To test the TOMES tool workflow using real data, SANC staff processed copies of ten email accounts from positions appraised as being Capstone positions from Governor Pat McCrory's administration (2013-2017) in September of 2018. Sixteen PST files were processed, due to some email accounts being too large to be exported from the eDiscovery module of Office653 as one PST file. The following email accounts were processed:

1. pat.mccrory@nc.gov
2. svc_gov.plm1@nc.gov (a secondary service account for Governor McCrory)
3. christiane.newell@nc.gov
4. joseph.nelson@nc.gov
5. tiffany.howell@nc.gov
6. catherine.truitt@nc.gov
7. trish.smothers@nc.gov
8. sue.breckenridge@nc.gov
9. april.riddle@nc.gov
10. jeff.mixon@nc.gov

Procedure

The TOMES processing workflow uses modules to process an email account through each of the following steps. For more details, please see the software user guide on the GitHub page.

1. Convert PST to EML
2. Convert EML to EAXS

3. Convert EAXS to a tagged EAXS file
4. Create an archival information package (AIP) structure consisting of source and derivative files as well as basic METS files

SANC staff accomplished the first three steps using the TOMES Docker application created for in-project processing. The Docker application provides a web-based interface to all but the final processing step. The application and data files were hosted on Amazon cloud servers. This allowed them to upload the source PST files to the cloud and then launch each processing step through my web browser. Actual data processing occurred on remote Amazon servers.

For each account staff converted each source PST file to EML with the TOMES PST Extractor module. Next, they used TOMES DarcMail module to convert each EML structure to an EAXS file. They then generated a tagged EAXS file from each EAXS file with the TOMES Tagger module.

For the final steps of the procedure, they packaged the processed files together into an AIP. Because this functionality was not available through the TOMES Docker application, they created the AIP by logging into the Amazon server and running the TOMES Packager module from the command line. They then generated checksums for each AIP using the Library of Congress's Bagger tool per the State Archive of North Carolina's policy.

Results

The initial PST files ranged in size from 10 MB to 6 GB, while the final bags ranged from 21MB to 13 GB due to the inclusion of new EML, EAXS, tagged EAXS, and METS data.

The time to convert files from PST to EML ranged from one minute to 18 minutes depending on the file size, with an average of about 5.5 minutes. Conversion to EAXS took 13 minutes or less, with an average of about 6.5 minutes. The creation of tagged EAXS files was the most time-intensive process, lasting 15 minutes to 1.5 hours and averaging 52 minutes. Creating AIP packages averaged about 10 minutes.

It should be noted that the processing was conducted on dedicated, remote Amazon servers with 64 GB of memory. Processing these accounts on less powerful machines would have taken up considerably more time.

Analysis

The processing met the stated objectives of the grant to process archival email accounts and provided a successful proof-of-concept for semantically tagged messages in the EAXS structure. The experience brought to light some usability challenges with the current implementation using Docker, and more importantly some higher-level issues regarding utility of the software and access to the output.

First, project staff encountered some minor issues with the web application that are due to the coding of the Docker app and not the underlying TOMES software. The version of TOMES Docker being used included default naming conventions for the processed files that were incompatible with the packager module, a useful discovery. They also noted that the ability to automatically move files to the next module and to process multiple files simultaneously would be helpful features.

More importantly, the TOMES team needs to evaluate the feasibility of using TOMES software on less powerful machines than those used for the in-project processing. Furthermore, they should consider how to lower the barrier of entry for use. Given the limited resources available to many state and local

archives, they may not have the hardware or staff with the technical expertise to facilitate use of the TOMES software as is.

Finally, there is room for more investigation into the utility of the EAXS schema and the tagged EAXS files for archival work. While the TOMES team was able to produce tagged EAXS files, the resulting XML is not very human readable or searchable. To search and meaningfully use the processed files, further software development is needed to interact with the output of the TOMES software. Now that messages can be semantically tagged, archivists will still need to filter, search, and make sense of the results.

The EAXS schema was developed as part of a previous North Carolina grant to convert proprietary files to a sustainable open source language; however, more work is needed in the archival community to set standards and best practices for email preservation. The EAXS schema could be updated and improved and discussed as a preservation format or as an access format, but these are questions that must be answered in collaboration with the larger archival community.