Terms

- *folder*: Logical container for messages in a mail account.
- *account*: Named set of folders.
- *mbox file*: A host file in mbox format (see https://www.loc.gov/preservation/digital/formats/fdd/fdd000383.shtml), whose file name ends in ".mbox". One mbox file holds the messages in one folder.
- *directory*: Host file directory.
- *account directory*: A directory containing directly or indirectly all the mbox files that correspond to folders in a single email account. Any mbox file contained directly or indirectly in an account directory is deemed to represent a folder in that email account.
- *folder directory*: The directory in which an mbox file is located.
- *DArcMail folder name*: A partial path name, constructed from the absolute path name of the corresponding mbox file by removing the account directory prefix and the ".mbox" suffix from path of the mbox file.
- *CSV file*: A host file in comma-separated-values format (see https://www.loc.gov/preservation/digital/formats/fdd/fdd000323.shtml)
- *DArcMail database file*: An SQLite host file initialized with the DArcMail database schema. For SQLite format, see https://www.loc.gov/preservation/digital/formats/fdd/fdd000461.shtml.
- *email address*: A minimal internet-standard email address, e.g., john.doe@xmail.com, that appears in a message address field ("From", "To", "Cc", "Bcc").
- *email name*: Any string appearing in a message address field that is not part of a minimal internet-standard email, e.g., "Doe, John". An email name may be paired with an email address, as in '"Doe, John" <john.doe@xmail.com>'. Because of differences in individual contact lists or email directories or email systems, or because of changes in one system over time, multiple email names may be associated with a single email address; for example, the names "Doe, Pat", "Pat Doe" may both be associated with the address patdoe@zmail.com It can also happen that an address field contains an email name with no paired email address; this can occur with internal email systems whose communications do not follow all internet protocols.

Inputs

Mbox files, each containing the messages from an email folder.

Persistent Outputs

- DArcMail database file: Each database file holds folder and message data for one account. On loading the database, the user has the option to store each message attachment in the database file or in separate "raw attachment" file, which is referenced in the database.
- ".raw attachment file: A host file representing an externally stored attachment of a message in the database. The ".raw" file does have any XML markup.
- XML files: XML representation of the folders and messages in an email account. Each email folder is represented as one or more XML files. The maximum size of an output XML file is set at approximately 2GB; if the XML representation for a single folder would be larger than this limit, the XML representation consists of more than one XML file. Regardless of folder size, each message attachment -- regardless of attachment size -- is represented in a separate "XML attachment file", which is referenced from the main XML file.

- CSV file: Every output XML file (main XML file, not attachment XML file) has a corresponding CSV file with one row per message, recording the contents of standard message header fields -- “From”, “To”, “Date”, “Subject”, “MessageID” -- as well as the SHA-1 checksum, the number of errors encountered during message-parsing, and the text of the first error encountered.
- mbox files: user-selected subsets of input mbox files.
- Operation log file: Each operation on persistent inputs and persistent outputs produces a text log file.

Functions

- Load data from mbox files into a database.
- Search the database for messages and email addresses.
- Export message sets into custom mbox files.
- Delete data from the database.
- Convert a mbox file to an XML file [NEED REFERENCE TO THE XML SCHEMA]

Components

- Database: DArcMail uses SQLite, a file-based SQL database.
- DArcMail.py program: DArcMail.py is a Python3 program with graphical UI. It does not rely on web services; rather, it runs on a computer to which the user is currently logged in. DArcMail.py is the only DArcMail program that uses the database. The SQLite database file must reside in a file system mounted on the computer on which the user is running the DArcMail.py program. DArcMail.py implements the load, delete, browse, and export functions.
- DArcMailXml.py program: DArcMailXml.py is a Python3 program with graphical UI. It does not rely on web services; rather, it runs on a computer to which the user is currently logged in. It does not use the database. DArcMailXml.py implements the convert function.
- CmdDArcMailXml.py program: This is a Python3 command-line implementation of the convert function; it lacks a graphical UI.

Functions of DArcMail.py

- Connect to a DArcMail database. The connect screen has one required parameter, the path of a SQLite database file. If the file does not already exist, it will be created and initialized with the DArcMail database schema. If the file exists but is not formatted as a SQLite database file, then the connection attempt fails. A DArcMail database file may have any path that is legal in the host operating system.

- Load data into a DArcMail database file. The load screen has four parameters: email account name, account directory, folder selection, and storage for message attachments. If a user has connected to an already existing DArcMail database, and if an account name and account directory have already been created in that database, then these parameters are pre-populated in the load, delete, and export operations. A single DArcMail database file can store only one account and one account directory; once these have been created, they cannot be changed, although the database can be reinitialized via the delete operation. Load parameters:

  - Email account name. There is no limit on the length of the name, and there are no illegal characters in an email account name. There is no requirement that the account name have any degree of similarity to the database file name.
  - Account directory. This must be the path of an existing directory. The account will not be entered into the database unless there is at least one mbox file contained directly or indirectly in the account directory.
  - Folders. The default is to load “ALL FOLDERS” in the account. But the user may specify that only one mbox file will be loaded. If the user chooses to load only one mbox file, other mbox files in the account directory can be loaded on subsequent load operations. No mbox file can be loaded twice; if only one or more mbox files have already been loaded, then “ALL FOLDERS” cannot be specified on a subsequent load operation unless all earlier loaded mbox files are first deleted from the database.
  - Store attachments externally. The default is to load all attachments into the database. See **Persistent Storage**, below.

- Delete data from a DArcMail database. The delete screen has one parameter: folders. The default is to delete “ALL FOLDERS” in the account. But the user may specify that only one folder will be deleted. Deleting all folders, or deleting the last remaining folder, also causes the account itself (account name and account directory) to be deleted from the database file. Using the DArcMail delete operation ensures that any externally stored attachment files are also deleted. Simply deleting the database file itself, without having first run the DArcMail delete operation, may result in orphaned host files whose names have no obvious connection to individual messages. See **Persistent Storage**, below.

- Export message sets into custom mbox files. In constructing the exported mbox files, DArcMail pulls all message headers, parts, and attachments directly from the original mbox files. Export parameters:
  - Folders. The user can export messages in a message set from one folder or from “ALL FOLDERS”
  - Export directory. To prevent inadvertent overwriting of original mbox files, the export directory must not be the same as the account directory. A mirror directory hierarchy will be created, as needed, to account for the folder structure of exported message sets, underneath the export directory.
  - The user can choose to export messages in the selected message set or messages not in the selected message set; see **Message Set**, below.

- Browse data already loaded into a DArcMail database. Browsing data creates no persistent outputs. There are four entry points to browsing: “Account”, “Message”, “Address”, and “Results”.
  - Account. The browse account page gives high-level information for an account: name, directory, count of messages, date range for set of messages in the account, the number of distinct email addresses (tallied separately for “From”, “To”, “Cc”, “Bcc”), and the number of externally stored attachments. The page also lists high-level information for each loaded folder: folder name, number of messages, and date range for the set of messages in the folder.
  - The “Message” and “Address” tabs are search forms. The result of a message or address search is a message or address list. Each entry in a message or address list is hyperlinked to

a message or address info page. List pages and info pages are accumulated under the "Results" tab. Result pages can be individually deleted from the Results tab. Search, list, and info pages are described further below.

- Results. This tab holds the accumulated set of message and address searches, as well as the results generated from hyperlinks in the results of the primary message and address searches.

Function of DArcMailXml.py and CmdDArcMailXml.py

Convert an mbox representation of the messages in an email folder to XML representation, the Email Account XML Schema (EAXS). The EAXS scheme is available at https://raw.githubusercontent.com/StateArchivesOfNorthCarolina/tomes-eaxs/master/versions/1/eaxs_schema_v1.xsd.

Message Search

- Messages can be searched by specifying one or more of these parameters: the MessageID field (Global Id), terminus post quem ("Date From"), terminus ante quem ("Date To"), containing folder (or "ALL FOLDERS"), keyword in Subject field, address or email name in the From, To, and Cc fields, attachment name, and keyword in the text of the message body.
- For all of these parameters except the dates, user-supplied strings are searched as substrings. Dates are given as YYYY-MM-DD; however for "Date From", "2019-01" and "2019" are treated as "2019-01-01", while for "Date To", "2019-12" and "2019" are treated as "2019-12-31".
- There is one additional search parameter, Message Status, with values "Any", "Selected", and "Unselected"; see **Message Set**, below.
- Search parameters with non-null values are logically ANDed to determine the set of messages included in the result set.

Message List

Each line of a message list has (a) a select/deselect checkbox; (b) the internal numeric id of the message, which is hyperlinked to a message info page; (c) the date of the message; (d) the subject of the message. A message list longer than 30 items is paged. A pulldown at the top of the list lets a user either select or deselect all the messages on the current page or all the messages resulting from the search; see **Message Set**, below.

Message Info

A message info page displays message header data (internal numeric id, account name, folder name, "MessageID", "Date", "Subject", "From", "To", "Cc", "Bcc") at the top and message content at the bottom. The header data also includes "In-Reply-To" and "Has-Replies"; if the replied-to message is part of the loaded account, then hyperlinks connect the reply chain. In the message content portion, each message part is shown on a separate line consisting of six fields:

- Action: (a) "view" if the message part has content-type "text/plain" or "text/html" and the Content-Transfer-Encoding of the part is not base64-encoded; or (b) "download" if the message cannot be viewed directly in DArcMail. A message part can be "downloaded" if it is stored

internally in the database or if it is an attachment stored externally. A part whose Content-Transfer-Encoding is base64 will be automatically decoded in the download process. The ability to view message parts with content-type “text/html” in the DArcMail application is limited: while most basic html renders correctly, embedded images are not rendered.

- Storage: “internal” or “external” depending on how the message part is stored.
- Content-Type: the “Content-type” of the message, such as “text/plain”, “text/html”, “multipart/alternative”, or “image/gif”.
- Length: length of the message part, in bytes.
- Original Name: the original name of a part that is an attachment.
- Part Id: the internal numeric id of the message part.

Message Set

The DArcMail program maintains a set of user-selected messages as long as the DArcMail program is running; it is not saved in the database, so is valid only during the current program session. Every message list lets the user either select or deselect individual messages, all messages on the list page currently being displayed, or all messages retrieved by the search command. The message set can be used to narrow subsequent searches and to export a custom subset of messages as an mbox file.

Address Search

The scope of the address search is all email address (e.g., “john.doe@xmail.com”) and email names (e.g., “Doe, John”) that have appeared in an address field (“From”, “To”, “Cc”, or “Bcc”) of any message that has been loaded into the database. The message search has only one parameter, the search string, which is automatically wild-carded at beginning and end.

Address List

Each line of an address list has (a) the internal numeric of the address, which is hyperlinked to an address info page; (b) the email address; (c) the email name. The email name may be identical to the email address. An address list longer than 30 items is paged.

Address Info

An address info page displays the account name, the email address, the internal numeric id of the address, the set of email names associated with the address, and the number of messages with this address in each of the “From”, “To”, “Cc” and “Bcc” fields, with each non-zero number hyperlinked to a message list. If an email address is not paired paired with an email name, DArcMail will display the email address as the email name; conversely, if an email name is not paired with an email address, DArcMail will display the email name as the email address.

Persistent Storage

The original account directory and its subdirectory structure is used to store

- The original mbox files. These are not changed by DArcMail.
- DArcMail log files produced by the load, delete, and convert functions; these are placed in the account directory.

- The ".raw" files produced by the load function, if the user has specified external storage of email attachments. The simple file name of each attachment is a 36-character UUID with the ".raw" suffix -- for example, "171204ad-06aa-4836-b23a-24383e3e85a4". Each "raw" external file is placed in one of 256 subdirectories of the folder directory. Each of the 256 subdirectories has a two-character name composed of hexadecimal digits: "00", "01", ..."0f", "10", "11", ..., "1f", ..."ff".
- The main xml and the csv files produced by the convert function for a given folder are placed in the folder directory . The xml attachment files are placed in one of 256 subdirectories of the folder directory, the subdirectories being named as described above for storage of the ".raw" files.

Each exported mbox -- consisting of a user-selected subset of messages from an original mbox -- is stored in a subdirectory of the export directory, where the subdirectory structure of the export directory mirrors the subdirectory structure of the account directory.

The database file may reside anywhere in a file system that the user can access. Once created and loaded, the database file may be move to another location and still remain valid.