Attachment Converter Workshop, iPres 2023

Nishchay Karle, Obi Obetta, Matt Teichman*

September 6, 2023

Welcome to the Attachment Converter Workshop at iPres 2023! In this session, we'll walk you through our new open source application, Attachment Converter, which batch-converts all attachments in an email mailbox to preservation formats.

Then next few sections of the handout include background on the project for your reference, but when the workshop starts, we'll be working through the material on this handout starting with section 4.

1 Project Website

The project website is located here:

`https://dldc.lib.uchicago.edu/open/attachment-converter`

2 How to Participate in This Workshop

There are two ways to participate in this workshop. If you're feeling tech-savvy, we would encourage you to install the required software in advance of the workshop and type along with us as we walk through some illustrative examples of the email format. If you aren't feeling tech-savvy, you should be able to just watch and follow along. Either way, we are really looking forward to engaging with your questions and comments as we show you how to use our new tool. If you aren't sure how tech-savvy you're feeling, the question to ask is whether you're comfortable opening the Terminal application on your computer and working at the command prompt.

In the next section, we'll go through how to install the software you'll need if you want to participate in the workshop by typing along on your own machines. If you're planning to simply attend, watch, listen, and ask questions, please feel free to skip to section 4, which is what we'll be working off of during the workshop—you won't need to set anything up on your computer in advance.

*teichman@uchicago.edu

3 Advance Preparation

If you're planning to type along with us on your computer during the workshop, then this is the section for you!

The software you'll need to install for the workshop is slightly different, depending on whether you're working in Windows or macOS. Either way, you will need to have privileges on your machine that allow you to install software, so if you're attending this conference from a work machine, that might be something worth looking into with your system administrator.

3.1 macOS

If you're on a Mac, you'll need to open a Terminal, then install an open-source package manager, the `git` version control system, the GNU Make build tool, and the `libpst` package. We'll go through those steps next, but if you're on Windows, please skip to section 3.2.

Remember: to follow these instructions, you'll need to have the ability to install software on your machine, so if you don't, you may want to reach out to your system administrator to see whether they can grant you the appropriate privileges for doing so.

3.1.1 Install Homebrew

There are various options for open source package managers on macOS, but we recommend using Homebrew. If you've never used it before, you'll first need to install XCode Command Line Tools, which you can do by running this command in your Terminal:

```
$ xcode-select --install
```

Then you can install Homebrew by following the instructions here:

`https://brew.sh/`

Or, equivalently, by typing this command:

```
$ /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

3.1.2 Install Libpst

The last thing we'll ask you to install is Libpst, the software we will use to convert from Outlook `.pst` to MBOX format during the workshop. To install that, run:

```
$ brew install libpst
```

Once you've reached this point on your Mac, you can skip the next section—which is our Windows-specific setup instructions—and proceed straight to section 3.3.

3.2 Windows (Debian WSL)

Attachment Converter is a UNIX application, which means that in order to run it on Windows, you'll need to install the Windows Subsystem for Linux. We chose Debian as a Linux distribution for this purpose, because Debian has full out-of-the-box support for OCaml, the programming language that Attachment Converter was written in.

So first, you'll install the Debian WSL. Once that's set up, you'll open up a Debian WSL Terminal and do everything else from inside that Terminal, including installing a few more utilities, as well as running Attachment Converter itself.

Note that you need to have privileges to install software on your machine to follow these instructions. If you don't, check with your system administrator about how to get them.

3.2.1 Set up the Debian WSL

To set up the Debian WSL:

- open up the Microsoft Store application using your Start Menu
- there should be a search box at the top of the window that opens
- type "Debian" in the search box and hit Enter
- a list of search results will come up; double-click on the one called Debian
- click through the installation buttons, prompts, etc. that come up
- you will eventually be asked to choose a username and password for your Linux subsystem
- don't forget to write those credentials down and keep them available for reference
- the installer will ask you to reboot your machine, which completes the process

Once you've rebooted and logged back in, you should be able to open a Debian WSL Terminal by running an application called "Debian" from your Start Menu. It will ask you to log in using the username and password you chose during the installation process.

3.2.2 Install Version Control Software

Now that your UNIX environment is set up, the next step is to install version control software, which in this case is Git. To do that, run this command:

```
$ sudo apt install git
```

Note that the first time you run a command with the prefix `sudo`, you will be prompted for a password. If that happens, use the password that you chose for your UNIX account when you set up the Debian WSL.

This is the utility we will use to get the latest version of the source code for Attachment Converter, later in these setup instructions.

3.2.3 Install GNU Make

The software we're going to use to compile Attachment Converter is called Make. To install it, run this command:

```
$ sudo apt install make
```

3.2.4 Install Libpst

Finally, we're going to ask you to install Libpst, which is a freely available utility for converting Outlook `.pst` files to MBOX format—the email mailbox format that Attachment Converter uses. To install it:

```
$ sudo apt install libpst
```

Once you've reached this point on your Windows machine, you're ready to go to the next section, in which we show you how to compile Attachment Converter into an executable that you can run.

3.3 Compile Attachment Converter

Now that you're set up with the basic software you need, whether you're on Windows or a Mac, the next step is to download the source code for Attachment Converter, compile it into an executable you can run, and put the executable in a location where your Terminal can see it.

3.3.1 Get The Code

The first thing we need to do is download the source code for Attachment Converter. The simplest way to do that is by using Git.

First, go to the directory on your computer where you would like the source code to get downloaded to. If you aren't sure where that should be, you can make a new directory to keep your source code in by running these commands:

```
$ cd ~
$ mkdir src
$ cd src
```

To then download the source code for Attachment Converter using Git, run:

```
$ git clone https://github.com/uchicago-library/attachment-converter.git
```

As an aside, if you're on Windows and want to view the contents of a directory you're in using Windows explorer, you can run this command to open up an Explorer window in the current directory:

```
$ explorer.exe .
```

If you're on a Mac, you can do the same thing—i.e. view the directory you're in in Finder using the `open` command:

```
$ open .
```

Now that you have the source code for Attachment Converter, the next step is to compile it into an executable program.

3.3.2 Compiling, the Semi-Automated Way

The utility we're going to use to compile Attachment Converter is called Make. If you're on Windows, we told you to install that in the previous section. If you're on a Mac, then you already have Make installed on your computer.

Attachment Converter has a lot of moving parts, which means that installing it involves installing some more standard utilities and copying a bunch of different files to a bunch of different places in your home directory. When you run Make, the full list of things it will do is:

- install all the free software that Attachment Converter uses to convert file attachments
- install `opam`, the package manager for the OCaml programming language
- create a location in your home directory for all `opam` files to go in
- install `dune`, the OCaml build tool, to that location
- install all third-party OCaml libraries that are necessary to compile Attachment Converter
- put a number of different configuration files in places where Attachment Converter expects them to be, in order to run

To compile Attachment Converter and then install it, run:

```
$ make home-install
```

You'll see a whole bunch of stuff get printed to the screen, which should give you an idea of what part of the installation process is happening. It may pause at one point to ask you to type in your administrator password. When the installation process is done, it should print a message that looks like this:

```
Attachment Converter has been installed to ~/bin/attc.
Please ensure that ~/bin is on your path.
```

Once the installation process is finished, `~/bin` needs to be on your shell path in order for Attachment Converter to run. If you don't know what that means, run this command if you're on Windows:

```
$ echo "export PATH=~/bin:$PATH" >> ~/.bashrc
```

And run this command if you're on a Mac:

```
$ echo "export PATH=~/bin:$PATH" >> ~/.zshrc
```

Then close and reopen your Terminal.

3.3.3 Compiling, the Manual Step-By-Step Way

If you get an error while running Make, another thing you can try is to do all the steps that our Make configuration does indivdually. Following all these steps should work, if there's an unexpected error in our Make configuration. (Though if you do encounter an error, we would love to hear about it, so that we can fix it and update these instructions!)

The full instructions for setting Attachment Converter up in the non-automated way can be found on our website here:

`https://dldc.lib.uchicago.edu/open/attachment-converter/docs/`

That concludes our setup instructions! The rest of this handout reflects what we will cover during the workshop proper.

4 During The Workshop

Attachment Converter is a command-line utility that batch-converts all attachments in an email mailbox to preservation formats. You give it your email in the form on an MBOX file, and it creates a new MBOX file with copies of all the attachments in preservation formats, next to the original attachments in the emails from which they originated.

Let's open the workshop with a quick demo of Attachment Converter.

4.1 Quick Demo

In this demo, we:

- run Attachment Converter on a small example MBOX containing five emails
- the example MBOX contains attachments in the following formats:
  - DOC
  - DOCX
  - XLSX
  - JPEG
  - PDF
- those attachments are then converted to, respectively:
  - TXT, PDF-A-1b
  - TXT, PDF-A-1b

- TSV, PDF-A-1b
- TIFF
- PDF-A-1b

4.2 Background

4.2.1 The MBOX format

4.2.2 The Anatomy of an Email

1. MIME types
2. Base64 data

4.2.3 How To Convert an Outlook `.pst` to `.mbox` Format

4.3 More Detailed Demo

4.3.1 Installing Attachment Converter

4.3.2 A Simple Example of Running Attachment Converter

4.3.3 Looking At The Output

1. The Headers that `attc` inserts
2. The Data that `attc` inserts

4.4 Advanced Configuration

4.4.1 Attachment Converter's Configuration File

4.4.2 A Glance at our Shell Scripts

1. where they go
2. rough overview of what they do

4.4.3 Show and Tell: Here is How To Add a new Utility to Attachment Converter