

Mastodon for Apple II

User guide

Version 1.7.1

October 2023

Table of Contents

Index of Figures.....	iii
Index of Tables.....	iii
Introduction.....	4
Startup and initial configuration.....	5
Serial communications setup.....	5
Logging in to your Mastodon account.....	6
Using Mastodon for Apple II.....	8
The main interface.....	8
Navigating in the Mastodon client.....	8
Contextual actions.....	9
Searching.....	10
The Profile view.....	10
Viewing images.....	11
Writing, editing or replying to a toot.....	12
Attaching images.....	13
Configuration.....	14
Internationalization of special characters.....	14
Frequently asked questions.....	15
The need for a proxy.....	15
Installation of the proxy.....	15
Support of Fujinet.....	15
Support of Uthernet.....	16
Links and resources.....	17
Thanks and acknowledgments.....	18

Index of Figures

Figure 1: Serial connection failure screen.....	5
Figure 2: Serial connection settings screen.....	5
Figure 3: The login screen.....	7
Figure 4: The main interface, showing the home timeline.....	8
Figure 5: The profile view (boost has been blanked out for seriousness).....	10
Figure 6: The image loading screen.....	11
Figure 7: The toot composition screen, showing a reply.....	12
Figure 8: The image selection UI.....	13

Index of Tables

Table 1: Serial ports by Apple II type.....	4
Table 2: Possible serial communication errors.....	4
Table 3: Actions opening new lists.....	7
Table 4: Contextual toot actions.....	7
Table 5: Contextual profile actions.....	9
Table 6: Image viewer commands.....	9
Table 7: Toot composition commands.....	10
Table 8: Replacement characters for international Apple II computers.....	12

Introduction

The Mastodon for Apple II project started during the long winter nights of 2023 as an experimentation. My goal was to figure out if I could get my Apple //c "online", how, and learn things in the process.

Mastodon seemed like a good playground for this, as the Fediverse started to thrive following the mess that Twitter became around the same time.

Mastodon for Apple II started its existence as a simple program, barely capable of displaying a list of toots, and gradually got more features, up to the point where some naive decisions stopped me from adding anything - every new thing I added made me hit one limit or another: not enough RAM, not enough space on the floppy disk, not enough screen real estate, slow performance, etc.

As I had hoped, hitting these limits pushed me towards learning things I only dreamt of learning as a kid experimenting with AppleSoft BASIC, and Mastodon for Apple II gradually became an almost full-featured client for Mastodon, adding features and reducing footprint at the same time. It allowed me to look under the hood, develop debugging tools, learn 6502 assembly, participate in a cross-compiler's development, and re-discover the joy of achieving difficult things on a computer.

Mastodon for Apple II is now a viable, albeit less comfortable, way of browsing Mastodon, and a good showcase of what can be done with the serial-to-network proxy that I wrote during this project. The proxy provides a framework for simple network transfers and is protocol-agnostic. It has no knowledge of anything Mastodon-related and can be reused for other projects, and, probably, other 8-bit computers.

This user guide is written to give the curious reader a sense of what can be done today with an Apple II - a machine built forty years ago! - and maybe give them the motivation to go through the initial setup and give this a real test, and marvel at the 280x192 green-and-black images that once were state of the art.

It may also be a way for me to declare this project complete and put an end to the hundreds of late-night hours, but I would not bet on it - maybe a GS/OS GUI version would be fun to do?

Startup and initial configuration

Serial communications setup

When starting, Mastodon will expect the serial communication to be setup with the proxy and will test it before proceeding. If your serial setup is different than the default one, you will be met with a Timeout failure and the possibility to reconfigure the client.

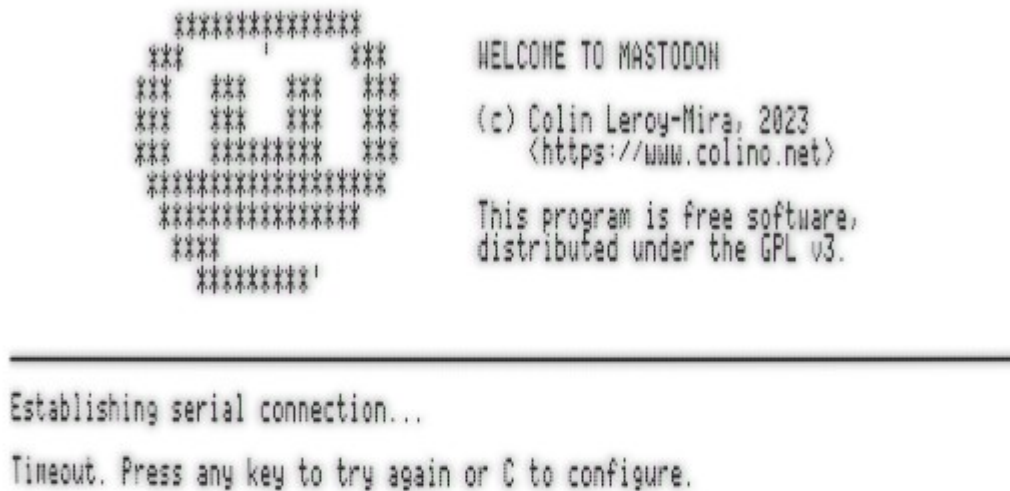


Figure 1: Serial connection failure screen

In this case, press the C key. Choose the serial port your cable is attached to using the left and right arrows, then use the Down arrow to configure the serial baudrate with the Left and Right arrows.

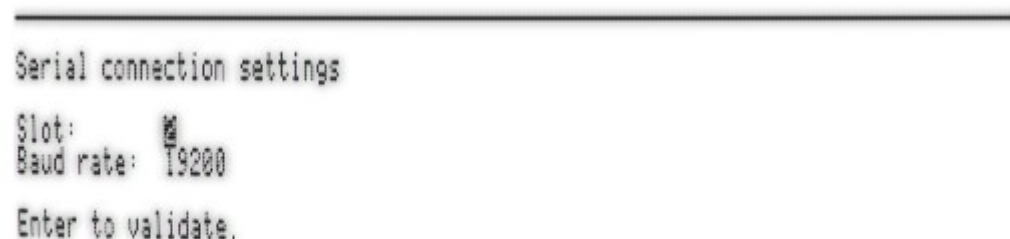


Figure 2: Serial connection settings screen

Note: by default, the proxy is set up at 19200bps.

The serial slot to use differs with different iterations of the Apple II computer :

Apple II type	Serial ports
Apple //c and //c+	Slot 1: Printer port Slot 2: Modem port
Apple IIe	Physical installation slot of the Super Serial Card
Apple IIgs	Channel A: Printer Channel B: Modem

Table 1: Serial ports by Apple II type

The client will try to explain precisely what is not working to help fixing the problem:

Error description	Possible causes
Timeout	Cable not connected; cable in the wrong port; wrong baudrate setting; interrupts disabled on the SSC card (DIP switch SW2-6)
Can not open serial port	The card in the selected slot is not a Super Serial Card.
Wrong protocol version	The proxy and the client run different versions of the wire protocol. Either upgrade or downgrade both to the same release.
Unknown error	Something unexpected. Try rebooting the proxy and the Apple II.

Table 2: Possible serial communication errors

Logging in to your Mastodon account

If no configuration pre-exists on the floppy disk, Mastodon will ask you for your login details the first time it is launched. It will ask you for your instance URL, your login (your email address, generally), your password and, if necessary, the OTP code from your two factor authentication application.

Note: your instance URL must start with the protocol (https:// generally).

Important note for Apple][+ users: This computer can not do lowercase. This means your password needs to be typable without lowercase characters.

Logging in on Mastodon is done via HTML forms. Consequently, it is possible that a future release of Mastodon changes specific parts of the login pages

layouts, making it impossible for Mastodon for Apple II to log you in correctly. If this happens, check for an update of the client.

Note: If you mistype your password or your OTP code, Mastodon will ask for them again. If you mistype your login or instance URL, on the other hand, reset your Apple II to re-enter these informations, using the Ctrl + Open-Apple + Reset key combination.

```
*****
***      ***
***      ***
***      ***
***      ***
***      ***
*****
*****
*****'

WELCOME TO MASTODON

(c) Colin Leroy-Mira, 2023
<https://www.colino.net>

This program is free software,
distributed under the GPL v3.

-----

Your instance: https://piaille.fr
POST /api/v1/apps... Done.
If on a non-US keyboard, use @ instead of arobase.
Your login: colin@colino.net
GET https://piaille.fr/oauth/authorize?response_type=code&client_id=IGLGyB4HWagU
FyIAKJ-OSHni_3B6KK4ahgEFB0rHjD4&redirect_uri=urn:ietf:wg:oauth:2.0:oob&scope=rea
d+write... OK.
Login required.
Enter password: *****
```

Figure 3: The login screen

Note for non-english users: if your Apple II has an international keyboard, chances are that the '@' character is not typable. This is because the character map is quite limited on the Apple II, and characters deemed not very useful in a language are replaced by other, more important characters. Mastodon informs you of this just before asking for your login, and indicates which character to substitute '@' with. On a french Apple II, this will be 'à'.

For the same reason, the '@' and '#' characters will be substituted in the main interface. You can use better alternative characters for those by configuring Mastodon - see **Configuration**.

Using Mastodon for Apple II

The main interface

```
Colin McMillen                               2023-09-19 19:32:58
@colin_mcmillen                               @Serenity
Commands:                                     Vous vous rappelez quand le gouvernement avait voulu faire u
View toot : Enter                             n site qui validait si un journal etait acceptable ou pas?
Scroll   : Up/dn                               0 replies, 0 boosts, 0 favs, 0 images
Search   : S
Notifs.   : N
Timelines : H/L/G                             ? muhorlogerie ?                               2023-09-19 19:32:39
Bookmarks : K                               @muhorlo
Configure : O                               Ca, c'est la pendule en question. Elle fonctionne a nouveau
Back      : Esc                               ?
Toot:                                         Marbre blanc, zinc dore pour les ornements et un sacre
Reply     : R                               ange en bronze patine qui pese une tonne.
Favourite : F                               A l'arriere du support de cadran, "le 16 9bre 1877"
Boost     : B                               Elle va donc feter 146ans dans moins d'un mois.
Bookmark  : M
Author:                                         0 replies, 0 boosts, 0 favs, 2 images
Profile   : P
Writing:
Compose   : C
58158 free
```

Figure 4: The main interface, showing the home timeline

The main interface is split in two vertical panes on Enhanced Apple IIs (//c, //c+, IIe enhanced, IIe Platinum and IIgs). The left pane lists the available commands. They will depend on the context, although some are always present. The right pane displays the content you are currently viewing. When starting Mastodon, you will see the last toots of your Home Timeline.

On older Apple IIe without a 80-column card and on Apple][+, the menu is hidden, and can be displayed using Ctrl-Y. The commands are also usable while the menu is hidden.

Navigating in the Mastodon client

The Home timeline is the first list you will see, but it is not the only one. Each time you navigate in Mastodon, a new list will open, or you will close the current list to go back to the previous one. An exception to this is when you want to refresh the current list: the refreshed list will replace the current list.

The following actions will open or refresh a list:

Key	Action
H	Loads the Home Timeline
L	Loads the Local Timeline
G	Loads the Global Timeline
N	Loads your notifications
K	Loads your last ten bookmarks
S	Loads your search's results (after filling in your query)
P	Open a profile
Enter	Open a toot and its replies

Table 3: Actions opening new lists

To go back to the previous list, use the Escape key.

You can scroll through the current list with the Up and Down arrows.

Note: When entering a list, minimal metadata is being kept in memory to allow returning to the point you were at. This means that opening too many lists will end up depleting the available memory, and bugs may happen if memory allocation fails unchecked. There is a free memory counter at the bottom-left part of the screen to help keep track of that.

Contextual actions

You can interact with toots in multiple ways: Favourite, Boost, etc. These actions are listed in the Toot and Author section of the left interface pane.

They will apply to the "active toot", which is the first one displayed. In other words, to favourite or reply to a toot, bring it to the top of your screen, then hit the 'F' or 'R' key.

The following actions are possible on a toot:

Key	Action
R	Reply
I	View images
F	(Un)Favourite
B	(Un)Boost
M	(Un)Bookmark

Key	Action
E	Edit (on your own toots only)
D	Delete (on your own toots only)
P	Open the toot's author profile
W	Toggle displaying sensitive toots (off by default)

Table 4: Contextual toot actions

The left menu will update accordingly to only display what is possible with the current active toot.

Note: the current state of your interaction(s) with a toot will be shown in its statistics line, by displaying a star next to the boost and/or favourite count if you boosted or favourited that toot.

Searching

To search for messages or a person, press 'S'. Input your query and validate with Enter, or cancel the search with Escape. You can toggle the search to "messages" or "account" by using the Open-Apple + M, or Open-Apple + A key combinations.

Note: the Account search will directly open the best-matching account returned by the search. Messages search will open a list of matching messages. As usual, you can leave both of those with the Escape key.

The Profile view

The profile view is a special list, starting with the person's bio and other information about them, then displaying the list of their toots and boosts. In the Profile view, the left contextual menu will be different when the active item is the person's bio instead of a toot.

From this view, you will be able to perform a few actions:

Key	Action
I	View the person's profile and header images
F	(Un)follow the person
B	(Un)block the person
M	(Un)mute the person

Table 5: Contextual profile actions

Note: Muting a person makes you unfollow them. This happens server-side.

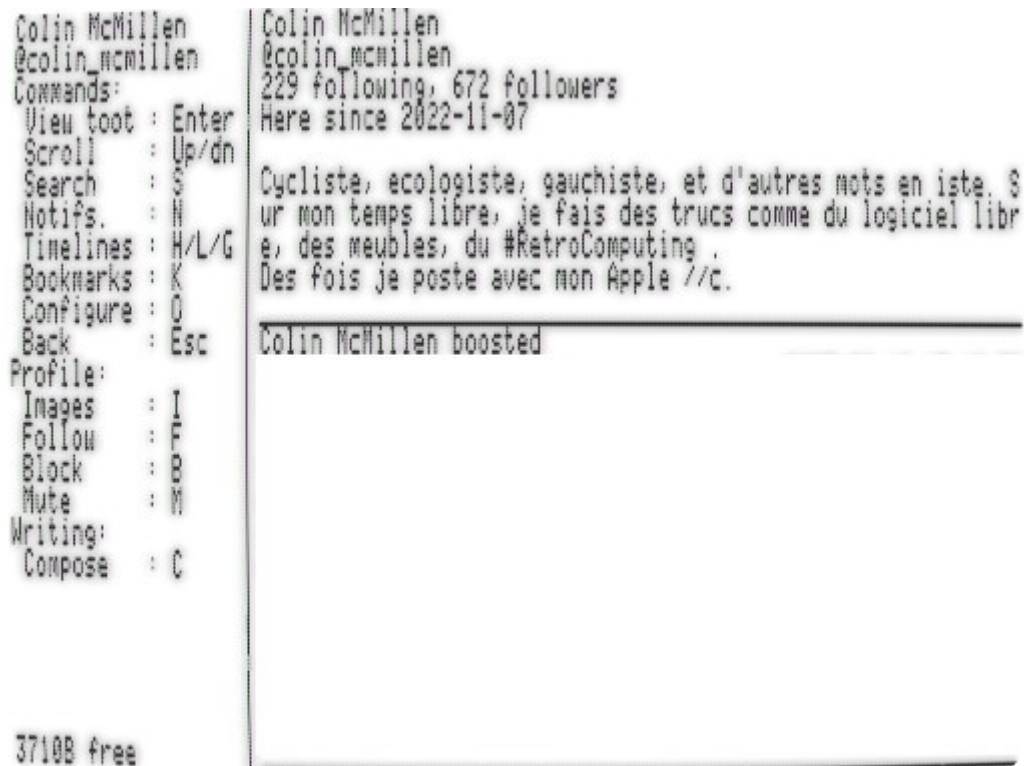


Figure 5: The profile view (boost has been blanked out for seriousness)

Viewing images

Whether the images you want to view are profile images or a toot's images, the interface to view them is the same. Hitting the 'I' key will open the image viewer.

During loading of the first image, you will see the available commands:

Key	Action
Escape	Quit image viewer and return to navigation
L	Toggle image / legend
S	Save image to disk
Any other key	Loop through the images

Table 6: Image viewer commands

To save an image, hit S, then provide a file name. You can either provide just a file, or a full path (/IMAGES/DOG for example) to save on another floppy.

Warning: There is a filter for toots marked sensitive in the navigation UI, but no additional confirmation before displaying their sensitive images.

Image 1/1:
Tabby cat chilling in a drawer labeled "Box".



Figure 6: The image loading screen

Writing, editing or replying to a toot

The toot composition interface is quite simple and also consists of two vertical panes, with the left one listing the possible commands. A notable difference to the navigation interface is that in the composition interface, commands all are key combinations, as single key strokes are directed towards your toot in progress.

These commands are:

Key	Action
Open-Apple + S	Send the toot, and return to the navigation interface
Open-Apple + Escape	Return to the navigation interface without sending, nor saving, the toot in progress
Open-Apple + I	Open the images attachment interface
Open-Apple + C	Edit the Content Warning
Open-Apple + P	Set the audience to Public (the default)
Open-Apple + U	Set the audience to Unlisted
Open-Apple + R	Set the audience to Private (followers only)
Open-Apple + M	Set the audience to Mentions (mentioned persons only)

Table 7: Toot composition commands

Note: On Apple II without a 80-column card, the menu is hidden, displayable using Ctrl-Y, and the commands are usable with Ctrl instead of Open-Apple.

Your toots' length will be limited to 500 characters, the same limit a standard Mastodon server applies, regardless of your instance's maximum length setting.

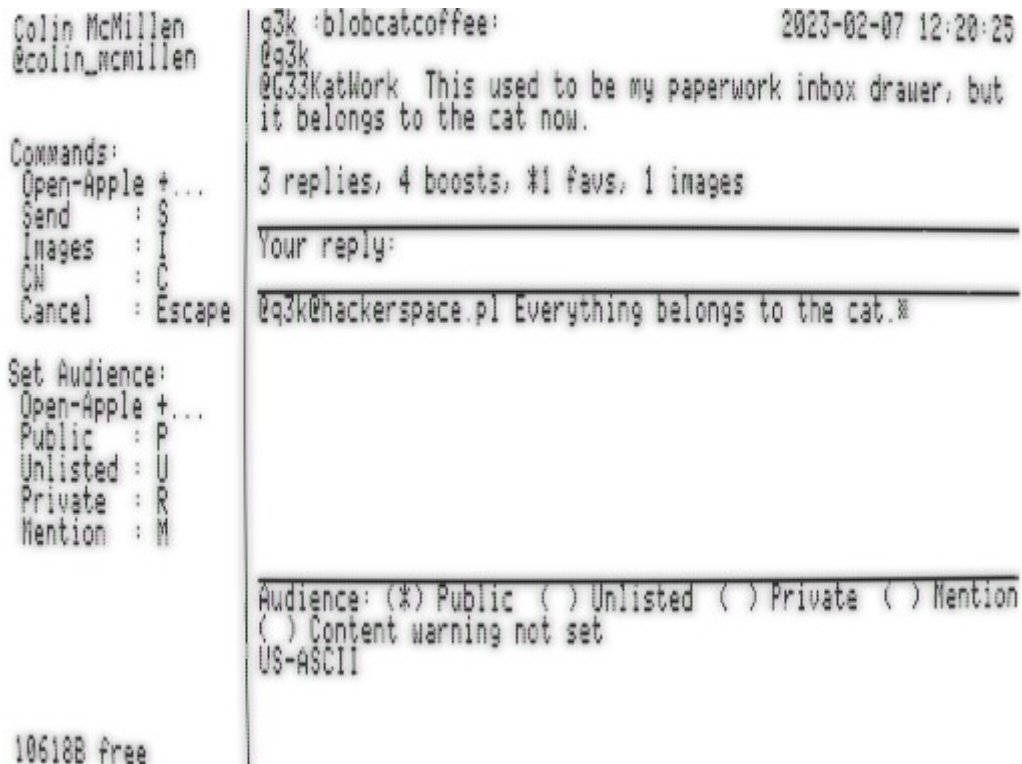


Figure 7: The toot composition screen, showing a reply

Attaching images

After pressing Open-Apple + I, you will see your toot's currently attached images. The list will be empty at first.

When you press Enter, you will be able to select an image file. Use the Up and Down keys to navigate the volumes / files list ; Use the Right key to enter a directory, and the Left key to go back to the parent directory. Use the Enter key to select an image, or the Escape key to cancel.

You will then be asked to provide a description to your image, which is an important thing for the some people relying on screen readers to browse Mastodon.

Upon validation of the image's description, you will return to the list of images.

The 'S' key in the attached images list will toggle the images' "sensitive" flag, and the 'R' key will remove the last attached image.

When you are done attaching images, choose Escape to return to the toot edition.

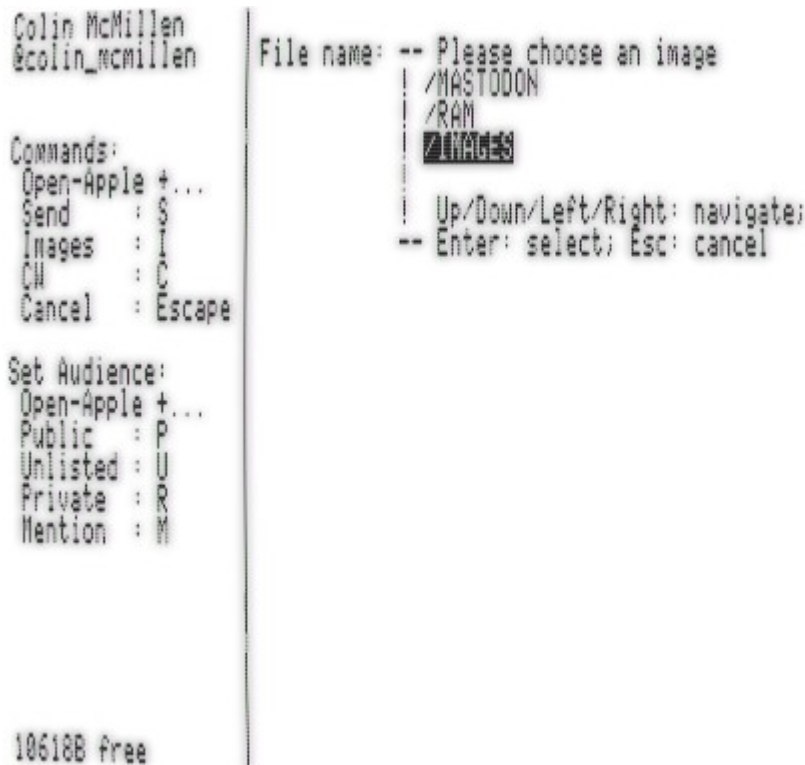


Figure 8: The image selection UI

Configuration

The configuration dialog will allow you to configure your internationalization option. Choosing the correct one is important for character set translation from Mastodon servers' native UTF-8 to the Apple II character set. Verify this setting if, for example, 'é' displays as '{'.

The second option to configure is whether your monitor is color or monochrome. This setting is used to scale and dither images. The Apple II has a 280x192 monochrome resolution, and only a 140x192 color resolution. If your monitor is monochrome, configuring Mastodon this way will allow for sharper images.

Internationalization of special characters

Two characters have a special meaning on the Mastodon platform: '@' for handles, and '#' for hashtags. Unfortunately, these characters are not represented in all internationalized Apple II computers. When you specify your Apple II localization in the Configuration screen, the Mastodon client and proxy will work together to replace these two characters, both ways, so that you will be able to mention people or use hashtags.

Unfortunately, this means that you will not be able to use these replacement characters as-is in your toots.

Country/charset	Character	Replacement character
FR / ISO-646FR1	@	\$
FR / ISO-646FR1	#	£

Table 8: Replacement characters for international Apple II computers

This table will be expanded with other variants if/when users from the relevant countries will have helped with the choice of good replacements.

Frequently asked questions

The need for a proxy

It can feel annoying that Mastodon for Apple II is not self-sufficient and requires the use of a helper proxy to perform its tasks. This design choice has been made for a few reasons. First of all, few Apple II computers are equipped with Ethernet cards. Another important reason is that old computers, even those with Ethernet connectivity, need proxies anyway to connect to +95% of the Internet : Cryptography is everywhere now, and cryptography is very hard to do on a platform with so little computing power compared to 2023's standards.

Enthusiasts on internet maintain a Crypto Ancienne library, which can be used by old computers to natively connect to services encrypted with modern crypto algorithms. Their README states that a Motorola 68030 CPU at 25MHz requires 22 seconds of prime number computations to connect to a server, and most servers hang up the connection during that time. I can not imagine doing anything with a 1MHz 65c02.

The proxy that Mastodon for Apple II uses also helps the Apple II handle computational-heavy things, like JSON parsing or image format conversion from PNG, JPG etc to HGR format. These are things that would be unbearably slow if done on the Apple II - if there was enough memory to even try!

This proxy software is written to be reusable for other projects.

Installation of the proxy

Installing the proxy is a quite standard process for anybody used to work and play with the Raspberry Pi platform. Unzip the image file, dump it to a microSD, and you're basically done. The proxy's installation is out of this document's scope, but if more information is needed, please refer to the project's homepage in the [Links and ressources](#) section.

Support of Fujinet

The Fujinet is a nice piece of hardware that plugs into the SmartPort of Apple II computers, and provide, among other things, WiFi connectivity and JSON parsing. At the beginning of the Mastodon for Apple II development, I considered using it instead of writing my own serial-to-network proxy. I decided against it for a few reasons: I wanted something being able to run

on any stock Apple II with a Language Card. The Fujinet device does not work on a ROM255 Apple //c, which ROM lacks SmartPort support.

I also wanted to lower the requirements a bit: it is already quite an involved process, requiring a very specific serial cable, ADTPro image transfer, and a Raspberry (although the Raspberry is in fact not a requirement, the proxy being able to run on any computer running a POSIX OS like GNU/Linux if one builds it themselves). I didn't want people to have to buy really specialized hardware.

Finally, I didn't want to have to get involved in another Free Software project, sending and defending pull requests in order to implement things I would inevitably need.

Support of Uthernet

The lack of support for Uthernet derives from similar reasons: the hardware is uncommon, and it can not be installed in an Apple //c. In addition to that, handling TCP from the Apple II would require handling other very difficult things from the Apple II, things I didn't think I could realistically achieve: encryption, JSON parsing, character set translation, and image conversion.

Links and resources

The project code lives at <https://github.com/colinleroy/a2tools/>

Its homepage is at <https://www.colino.net/wordpress/en/binary-release-of-mastodon-for-the-apple-c/>

It is built using cc65, <https://github.com/cc65/cc65/>

Thanks and acknowledgments

Thanks to the #RetroComputing community, full of awesome people. They push me towards doing more and better, especially Scott Small (@smallsko@oldbytes.space) and SuperIlly (@dec_hl@mastodon.social).

Many thanks to Oliver Schmidt (<https://github.com/oliverschmidt/>), long-time cc65 contributor and Apple II hacker, for his patience and guidance when I need help to learn things, from linker configurations to 6502 assembly to acceptable-pull-request coding-style.

Heartfelt thanks to my soulmate Abigail for her encouragements, her incredible patience and tolerance to my deep-focus mode, the mode in which I can spend full week-ends doing nothing else and basically don't live here (or anywhere) anymore. I love you.