



*Postilion mailbox view*

Many years ago, around 1996, I wanted a better MUA, Mail User Agent — What most people would call an email client, or mail reader. I ended up writing my own, and it was called [Postilion](#). I haven't done anything with this program since some time in 2001, but was surprised to wake up this morning to a message from a user. Here is that message, and my reply.

As a note, "curses" is a library for manipulating standard old text displays (not graphical programs) and OpenBSD is an operating system similar to Linux.

Here, then, is the original message, from Mayuresh:

Hello,

I used to use "Postilion" under Linux running WindowMaker.

I found your application to be quite well designed on the UI front.

I migrated over to OpenBSD 5 years back and found it to be exactly what the team claims it to be, simple, clean and secure.

The OpenBSD team has just delivered a SMTP daemon;

<http://www.openbsd.org/cgi-bin/man.cgi?query=smtpd&sektion=8>

It would be nice to have a well designed MUA too.

Would you be interested in designing and developing a curses based MUA?

Best.

Here is my reply:

Mayuresh,

Thanks for your kind words about Postilion. I haven't had a message about that software in years. I stopped working on it in 2001, and haven't even used it myself for years.

I don't think I'm the one to handle making a new (text based) curses MUA. I am not a C programmer (Postilion was mostly TCL/Tk) and I don't really have any interest in using a curses MUA. That's really key — I can't see developing a software project that I myself am not interested in using.

When I first started work on Postilion, I had just read Eric Raymond's (esr) "[The Cathedral and the Bazaar](#)". I was impressed by the Mail.app program available from NeXT, which I could run on NeXT/Step on a Sun system I had. Mail.app, however, did not support IMAP, and had other limitations. I was itching to have an MUA which matched the elegance of Mail.app but with more modern features. This being in 1996, that meant IMAP, ACAP, multiple folders open at once, multiple identities, etc.

After reading Cathedral, I decided to take esr's words to heart. He talked about how important it was that the development projects a programmer took on had to scratch his own itch, and my itch was for an elegant MUA. Secondly, he talked about finding a program which is close to what you want, and using that as a starting point. I had already found [TkRat](#), and figured that was a good place to start — the interface was written in TCL/Tk, an interpreted language (I am more comfortable with interpreted languages than with compiled ones) and the interface was fairly close to what I wanted. Also, the author, Martin Forssén, was friendly and helpful, had a really good, robust C language core for handling the messy parts of the protocols (thanks to Mark Crispin's fabulous c-client) and he seemed flattered that I was basing my work on his. Our two projects gained a sort of symbiosis over time.

Sure enough, as I worked on the project, I found that the Bazaar part of the process kicked into gear. I soon had hundreds, then thousands of users. My community was helpful, providing translations, testing, etc. One user, Marco van Hylckama Vlieg, contributed the entire set of custom made icons used in the GUI. At its peak, Postilion was in use on every continent (except, perhaps, Antarctica), in English, French, German, Italian, Dutch and Swedish. It was favorably written up in a French computer journal and adopted as part of a civic computing infrastructure project in a German village.

I was honored by all of that. It was exciting to get email messages from Abu Dhabi or Moscow, Bucharest or Lima. I sometimes got well over a hundred messages a day from people thanking me for my work, offering suggestions or requesting help. I was hired by a call center management company to make custom additions to the core program, just to suit their environment. My own employer at the time, an early ISP, paid me to add many of the features the software offered.

What made this all possible, however, was my own desire to use the software I was writing. If you really want a good, curses based mail reader, Mayuresh, you should think about taking the journey I took! Find something close to what you want, Mutt, say, or Alpine; something with a license you can abide; something in a language you're either comfortable with, or feel you can learn, and have at it. Take Eric Raymond's advice — scratch your own itch, and start by building on the work of others. I think you'll find it a worthy mission.

I wish you well,

-nic

Just my little contribution to the Open Source world. Not Postilion, that's old news. No, I mean prodding someone else to take the journey themselves.