

With Software As a Service, Is Only the Network Luddite Free?

Bradley M. Kuhn

Saturday 21 February 2009

In The Beginning . . .

- There was a computer.
- There were terminals.
- There were users.
- . . . and the users had freedom.

Ye Ol' Four Freedoms

- to learn.
- to copy and share.
- to modify.
- to share modified versions.

. . . Then Freedom Eluded Us

- They built licensing walls.
- By separating the software from the computer.
- Who Invented Licensing?

Who Invented Licensing?

Some think he did: Hai Hobbyists! Softwearz gived free-domes, but I ated them.

- Gates convinces IBM to license rather than acquire DOS.

Who Fixed this Problem?

Who Fixed this Problem?

RMS: My Old Boss

- Job at MIT's AI lab in late 1970s and early 1980s.
- Programming their PDP-10!
- In the Golden Age.

MIT AI Lab: The Golden Age

- academic computing.
- software sharing.
- no licenses!

MIT AI Lab

MIT changed.

MIT AI Lab

patents == \$\$\$

MIT AI Lab

spin-offs == \$\$\$

GNU

- Changes the world.
- For the user of individual computers.
- No reason now not to have freedom on your own machine.

Internet changes little (to start)

- Client/Server computing.
- Freedom implications are basically the same.
- Example: The Email System.

Email: The MTA

- Yours has freedom: (postfix, exim, sendmail).
- Plenty of proprietary ones: (exchange).
- RFCs define interoperability.
- We reverse-engineer RFC-less protocols: (OpenChange).
- UR SERVERZ CAN HAS FREE-DOMES.

Email: The MUA

- Yours has freedom: (Thunderbird, mutt, Gnus)
- Plenty of proprietary ones: (outlook)
- UR CLIENTZ CAN HAS FREE-DOMES.

What changed?

- The Browser as application delivery platform.
- AJAX.
- Whose computing is done where?
- Computing in the Cloud.
- Stall-menz gived u free-domes, but Cloud ated them.

Email becomes Gmail

- “The experience”
- Effectively: Thin Client.
- Mixes Up Free again: Price vs. Freedom.
- RFCs no longer enough.
- We’re back to proprietary “lock-in”.

Users and “The Cloud”

Users and “The Cloud”

Even *Worse* Than Proprietary Apps and OS’es

Meanwhile, My Old Boss Says . . .

“[Cloud Computing] is stupidity. It’s worse than stupidity: it’s a marketing hype campaign.”

- That’s his way of telling the next generation to deal with challenge.
- What’s that challenge?

What’s The Challenge?

- Not merely a question of code and licensing.
 - Although that’s part of it (more later).
- A question of community source+data control.
- Power to transport: applications, data, human connections.
- Autonomy: of code, of data, of community.

How Do We Start?

- Infrastructure that respects freedom.
- Empower the user: give them **all** the code.

Can't Stop There

- User communities need ability to transplant.
- Reclaim your data; relocate your community.
- This is **tough** programming.

Projects to Look At

- Evan Prodromou's identi.ca (laconi.ca).
 - Microblogging: (Follow me there: 'bkuhn')
- Jesse Vincent's Prophet.
- Yourselves.
 - make it even easier for developers to respect freedom.

License To Look At

- Affero GPL.
- Extends copyleft to network service world.
- Handles the code side well.

Techniques to Look At

- Deployed applications auto-give users source.
- Data is downloadable in community-chunks.

“Future” of the Cloud

- Disjoint but Integrated.
- Portable and Autonomous.
- Developers decide next direction.
- Ask **the** question in Cloud projects:
 - Are we respecting the users' freedom and autonomy?
- The data is theirs; we are *their* custodians.

More Info and This Talk's License

More Info? See: <http://autonomo.us/>

This talk and the slides are

Copyright © 2008, 2009, Bradley M. Kuhn.

These slides, this talk, and audio/video recordings thereof are under a Creative Commons Attribution-Share Alike 3.0 United States License.